Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : Mathématiques et Informatique

Arrêté ministériel : 25 mai 2016

Présentée par

Alessandro FENICIO

Thèse dirigée par **Gaëlle CALVARY**, Professeur, Grenoble INP
et codirigée par **Yann LAURILLAU**, Enseignant-Chercheur, UGA
préparée au sein du **Laboratoire Laboratoire d'Informatique de Grenoble**
dans **l'École Doctorale Mathématiques, Sciences et technologies de l'information, Informatique**

Ingénierie de l'Interaction Homme-Machine et Persuasion Technologique : étude du concept de Chemin Persuasif

Engineering Human Computer Interaction and Persuasive Technology: study of Persuasive Paths

Thèse soutenue publiquement le **23 septembre 2019**,
devant le jury composé de :

Madame Gaëlle CALVARY
Professeur, Institut polytechnique de Grenoble , Directeur de thèse
Monsieur Yann LAURILLAU
Maitre de conférences HC, Université Grenoble Alpes, Co-directeur de thèse
Monsieur Patrick GIRARD
Professeur, Université de Poitiers, Examinateur
Monsieur Patrick REIGNIER
Professeur, Institut polytechnique de Grenoble, Examinateur
Monsieur Guillaume RIVIERE
Maître de Conférences, ESTIA, Examinateur
Monsieur Christian BASTIEN
Professeur, Université de Lorraine, Rapporteur
Monsieur Jean VANDERDONCKT
Professeur, Université catholique de Louvain, Louvain School of Management, Rapporteur

Abstract

Societal challenges are an international concern. Daily advertising campaigns rise attention of people to make them change: "Smoking kills", "Drinking or driving, choose", "Eating five fruits and vegetables a day", etc. However, these campaigns have limited effect.

Persuasive technologies have been explored for fifteen years to orient technology on the difficulty of changing behavior. Monitoring devices such as bracelets or watches of physical activities and applications are multiplying obtaining commercial successes. However, despite the potential capabilities of technology of delivering personalized strategies, the incentive to change remains limited. The difficulty lies in the multidisciplinarity of the field: designing persuasive interactive systems requires mastering the fundamental concepts and the advances in cognitive and social psychology, which makes the persuasive practice extremely ambitious.

This thesis contributes to the engineering of persuasive interactive systems. It deals with the process of behavior change and proposes the concept of persuasive path to stimulate users in their behavior change. The persuasive path is a succession of events designed to pave the progression of the user toward the change among the set of possible behaviors. This set is modeled with state machines describing all the possible transitions between behaviors. Transitions between behaviors are triggered when the determinants of the corresponding behaviors are satisfied in the current user's context. A persuasive architecture is proposed to orchestrate the state machines and the persuasive paths. The formalism of state machines also allows the characterization and comparison of change processes in the literature.

An incremental design method is proposed to design, step by step, the state machine and the persuasive path. The steps proceed in order to actuate design choices that make the system little by little more dependent: problem dependent, domain dependent, task dependent and context dependent. This structuring progressive conception allows a revision of the design choices according to the observed performance of the persuasion.

The conceptual contributions (concepts and design method): CRegrette, an application aimed at stopping behavior (smoking); on the other hand, Mhikes, an application aimed at reinforcing behavior (walking). A complete implementation of Mhikes (concepts and architecture) is made available to show the technical feasibility

of the approach. The technological maturity of this approach allow the deployment of the application at real scale and an experimental evaluation of the contributions.

The evaluation results confirms the relevance of the models and of the architecture, allowing the introduction of software probes (1) to identify the roles endorsed by users, 2) to follow the possible changes and 3) to produce personalized notifications. The notifications resulted more efficient than the communication campaigns operated by Mhikes. However, the role changes remains complex, with extra-transitions that are more difficult to actuate than intra-transitions.

In conclusion, the thesis delivers a complete set of methods, models and tools for the engineering of persuasive interactive systems. More broadly, this set can be used by other communities to progress in the compression of human interaction.

Résumé

Les défis sociétaux sont une préoccupation internationale, avec des incantations quotidiennes au changement : “Fumer tue”, “Boire ou conduire, il faut choisir”, “Manger cinq fruits et légumes par jour”, etc. Toutefois, ces campagnes publicitaires restent à effet limité.

Les technologies persuasives sont explorées depuis une quinzaine d’années pour mobiliser le numérique sur ces difficultés de changement de comportement. Les dispositifs et applications de monitoring se multiplient avec des succès commerciaux comme les bracelets ou les montres d’activités physiques. Toutefois l’incitation au changement reste limitée malgré le potentiel du numérique pour des stratégies personnalisées. La difficulté tient à l'interdisciplinarité inhérente au domaine : concevoir des systèmes interactifs persuasifs requiert de maîtriser les fondamentaux et les avancées en psychologie cognitive et sociale, ce qui rend l’exercice extrêmement ambitieux.

Cette thèse contribue à l’ingénierie des systèmes interactifs persuasifs. Elle traite du processus de changement de comportement. Elle propose le concept de chemin persuasif pour stimuler l’utilisateur dans son changement de comportement. Le chemin persuasif est une succession d’événements incitant l’utilisateur à cheminer d’une certaine façon dans son ensemble de comportements possibles. Cet ensemble est modélisé en une machine à états explicitant l’ensemble des

comportements et des transitions possibles entre comportements. Les transitions sont déclenchées en contexte, lorsque les déterminants des comportements correspondants sont satisfaits dans le contexte courant de l'utilisateur. Une architecture persuasive est proposée pour opérer les machines à état et chemins persuasifs. Le formalisme des machines à état permet aussi la caractérisation et la comparaison des processus de changement de la littérature.

Une méthode de conception est proposée pour concevoir, étape par étape, la machine à états et le chemin persuasif. Les étapes procèdent, pas à pas, à des choix de conception rendant le système petit à petit réalité dépendant, problème dépendant, domaine dépendant, tâche dépendant et contexte dépendant. Cette conception progressive est structurante et permet une révision des choix de conception selon la performance observée de la persuasion.

Les contributions conceptuelles (concepts et méthode de conception) sont illustrées sur deux cas d'étude complémentaires : d'une part, CRegrette, une application visant à stopper un comportement (fumer) ; d'autre part, Mhikes, une application visant à renforcer un comportement (marcher). Une implémentation complète de Mhikes (concepts et architecture) est détaillée pour démontrer la faisabilité technique des propositions. Sa maturité technologique a permis un déploiement de l'application en grandeur réelle et une évaluation expérimentale des contributions.

Les résultats d'évaluation confirment la pertinence des modèles et de l'architecture pour placer des sondes logicielles permettant (1) d'identifier les rôles joués par les utilisateurs, 2) d'en suivre les éventuels changements et 3) d'émettre des notifications personnalisées. Les notifications s'avèrent plus performantes que les campagnes de communication aujourd'hui pratiquées par l'entreprise Mhikes. Toutefois le changement de rôle reste difficile, avec des transitions extra-rôles plus difficiles à franchir que les transitions intra-rôle.

In fine, la thèse livre un ensemble complet de méthode, modèles et outils pour l'ingénierie des systèmes interactifs persuasifs. Plus largement, cet ensemble peut servir à d'autres communautés pour progresser dans la compression de l'humain en situation d'interaction.

Table of contents

1 Introduction

The thesis contributes to the engineering of persuasive interactive systems, a particular class of interactive systems designed to change the behavior and/or the attitude of users. Performing changes is hard for people: sometimes they are not motivated, sometimes not capable.

The mission of persuasive systems is to accompany individuals in a process of self-development aimed at boosting awareness, motivation, and ability to accomplish the targeted objectives. The work deals with the engineering of such systems. It tackles the difficulty of designing, developing and evaluating persuasive systems, proposing new means to bridge the gap between the theory and the operationalization of persuasion.

1.1 Research domain

The research domain of this work is **persuasive technology,** defined as a particular class of interactive systems intentionally designed to alter or change the attitudes and/or behaviors of users avoiding any form of coercion. The investigations of persuasive technologies is particularly important since their principles are currently employed by society to help individuals in engaging into the most difficult **personal and social challenges**.

Examples in which persuasive technologies are used embrace **health** (e.g. reducing alcohol and tobacco consumption, promoting physical activity and healthy food), **productivity** (e.g. increasing quality of work), **self development** (e.g. learning new skills, achieving personal objectives), but also **sustainability** (e.g. reducing waste of resources, increasing recycling of materials), **climate changes** (e.g. reducing the CO2 emissions, encouraging adoption of electric vehicles, sustaining green-industry), and many others.

Nowadays, advertising campaigns are the most diffused means to challenge individuals to overcome social problems. Messages as **“Smoking kills, quit!”**, **“Avoid junk food, eat healthy!”** are delivered to thousands of individuals, however, this seems to **not be sufficient** since just a small part of them ultimately changes their habits. Often individuals cannot progress on their personal changes: they may not have an action plan, or they may postpone the engagement. In other cases, adopting the wrong strategy to undertake the change may lead individuals to fail and to enter in a state of frustration.

Persuasive technologies operate in such difficult scenarios, offering a new alternative, for a higher engagement of individuals by using technology, able to provide **personalized strategies** that guide the user toward the achievement of his/her objectives.

Persuasive technologies are a cross field domain, concerning different research areas such as **psychology**, **sociology**, **systems engineering**, **human-computer interaction**. Researchers in these fields have provided their contribution to the state of the art in different forms. Psychologists and sociologists produced different theories and models, identifying steps and emotional states that individuals traverse during the change. Technical researchers and practitioners instead have targeted the operationalization of these models, translating them into concrete persuasive interactive systems.

Resulting from this intrinsics pluri-disciplinarity, the literature in the field is huge, making engineering persuasive interactive systems particularly difficult as it needs to account the diversity of these disciplines. Ultimately there is a need today to define concrete means to operationalize the plethora of theories and models, into reliable implementations.

1.2 Research problem

The design of persuasive technology is often **craft production**: developers do not have sufficient notions on the complex domain of persuasion to methodically transpose persuasive principles into concrete artifacts. This is not the case in other domains. For example, for engineering interactive systems, developers can rely on conceptual methods that structure the development. Task models, design patterns, architectural frameworks guide developers, providing them with steps to be followed to pass from the system requirements to the final implemented system and its verification.

In persuasive technology, at present, there is a lack of conceptual methods structuring the development, which causes a gap between the objectives targeted by persuasion and their final software implementation and evaluation.

For this reason, persuasive design is often achieved by using a “pick and mix” cocktail of strategies. Unfortunately, this approach is fundamentally wrong because the persuasive models rely on theories that cannot be composed and mixed opportunistically. The current alternative is to rely on experts in both persuasion and development, but this is not always feasible, especially in small companies.

1.3 Research question

In this work, we investigate the possibility of paving the way of engineering persuasive interactive systems by defining a **unifying concept** able to guide developers toward the design, operationalization and evaluation of interactive persuasive systems, preventing practitioners from falling in opportunistic and possibly incorrect compositions of psychological theories.

This question reveals to be crucial since the persuasive field is also required to deal with a vast set of systems, targeting different types of attitudes and/or behaviors, such as increasing a behavior (e.g. increase the physical activity) versus suppressing a behavior (e.g. quit smoking).

The research question thus is extended at investigating the capability of this hypothetical **unifying concept** to deal with the variety of behaviors targeted by persuasive systems, the ultimate goal being to master the intrinsic complexity of multidisciplinarity for engineering persuasive interactive systems.

1.4 Thesis statement

We answer the research question by analyzing persuasion under the perspective of a **process**. We assume two perspectives:

- On one hand, **user-centered**, focusing on describing the suite of user behaviors as a process,
- On the other hand, **system-centered**, dealing with the engineering of the technology supporting the user change.

For the user-centered perspective, we use the **state-machine** formalism to represent the behavioral processes reported in the literature. We show that it makes it possible to represent domain independent (e.g., Fogg's behavior model) as well as domain dependent processes (e.g., CRegrette and Mhikes). We also show that the literature is poor of conceptual tools permitting to turn these user-centered processes into a system point of view. For this reason, we propose the concept of **persuasive path** as structuring tool to frame system-centered processes and to integrate the literature models.

We claim that the **persuasive path** may be the missing conceptual tool in persuasion, because, besides its modelling capability, it provides developers with an

implementable structure, bridging the gap between the modeling and the operationalization.

To model persuasive paths, we propose the concept of **persuasive event**, as basic unit of persuasive paths. With persuasive events, it is possible to refine the missions targeted by the individuals into smaller elements, associated with context and linked to the system and user tasks. This association permits the definition of dedicated software probes, capable of evaluating the performance of the persuasive system and of dynamical adaptation.

1.5 Approach

In this thesis, we combine a theoretical and practical approach.
Theoretically:

- We analyse the literature theories and models from the perspective of the process;
- We identify an operalization between the process of change and the persuasive features;
- We provide the definition of new concepts including persuasive path and persuasive events, interactive role and role-switching;
- We provide a conceptual tool to refine the user's targeted mission in smaller elements easier to be modeled and evaluated.

Practically:

- We provide a practical persuasive architecture to operationalize the persuasive paths;
- We provide illustrated case studies to discuss the proposed concepts;
- We implement our theoretical concepts on real interactive systems;
- We carry on the experimental evaluation of the implemented systems by recruiting real users.

1.6 Case studies

In order to picture the variety of different behaviors that persuasive systems are called to deal with, we present two case studies based on two opposite types of targeted behaviors:

- **CRegrette**: a smoking-cessation persuasive system based on a mobile application and on a digital brooch. The targeted behavior of the system is

reducing/suppressing the behavior (smoking). The study demonstrates the importance of driving an in-context persuasion.

- **Mhikes**: a web and mobile interactive system to perform outdoors activities. The initial system has been made persuasive by implementing the approaches proposed in this thesis. The targeted behavior of the Persuasive Mhikes is increasing/performing the behavior (performing outdoors activities).

1.7 Outline

In this thesis, we will first introduce the reader to persuasion and persuasive technologies, exploring the state of the art from the point of view of the process of change. Then in Chapter 2 two we will introduce the concept of persuasive path as structuring tool to describe persuasive models and targeted behaviors. Successively, we will present the software architecture that permits to operationalize persuasive paths giving a concrete implementation example on the real interactive system Mhikes, produced by Easy Mountain which financed this work. Finally, we report on a set of pilot studies and experiments providing insights on the application of persuasion via persuasive paths.

2 State of the art

Persuasive technology is a cross field domain with copious literature. Precedent works have thus focused on providing structured reviews of the state of the art. An example is the work of Pinder et al., which synthesizes different theories into the “Habit Alteration Model” (Pinder et al. 2018), used as explanatory framework to synthesise the state of the art in persuasion.

In line with Pinder’s et al.’s approach, we focus on the procedural nature of behavior changes. By contrast, we structure this review aiming at observing the basic properties of the processes of change to support their future implementation. We discuss the different approaches with a common framework, based on elementary state machines. The objective is to highlight for each theory which are the different states that users have to travel, what are the transitions making users progress toward the change, and what are the determinants on which these transitions rely on.

In order to organize the review, we place three sections in this chapter:

- The first one presents the state machines based **formalism** we use to structure the systematic review of the state of the art,
- The second one is dedicated to the state machines-based **analysis** of the state of the art,
- The third one focuses on the **design** of the process of change aiming at finding a relationship between the persuasive models and features.

2.1 State machines as systematic formalism

To describe behavior changes, the formalism needs to be able to express: what is the behavioral status of the individual, how the individual got to this state, what are the alternatives for further behavioral evolutions, and what may cause these evolutions.

State machines are a powerful computational model used in mathematics, logic, computational theory and computer science. A state machine (or automaton) is an abstract machine that operates transitions among a set of states in consequence of external inputs.

State machines are appropriate to describe the causality of the transitions between the different behavioral states that bring an individual to change behavior.

Investigating this possibility, we have ultimately formulated a formalism that permits to apply these mathematical objects in the context of our research.

In the following section, we first introduce elementary state machines, then we present how this formalism can be applied to behavioral changes, and we conclude with some illustrative examples.

2.1.1 Elementary concepts

The theory of computation uses different types of automata to describe the expressivity and the complexity of languages. State machines are one class of these automata, and are defined by using a set of **states,** connected by **transitions,** and a set of external input **conditions** that determine the passage from one state to another.

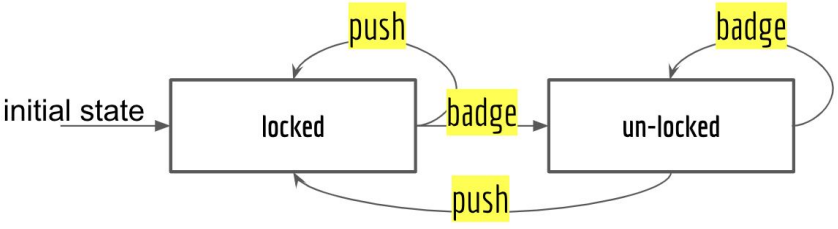


Figure 2.1	Example of finite automaton describing a badge operated turnstile. The turnstile is locked until a person uses a badge. Once the badge is used, the turnstile unlocks and it is possible to push it.

In Figure 2.1, we provide a basic example of automaton that describes the functioning of a turnstile. There are two **states** corresponding to the situations where the turnstile is locked, or unlocked. There are four **transitions**, represented by the arrows connecting the states or looping on one state. The **conditions**, which are the external inputs that determine the transition, are the user's actions: push or use the badge.

The initial state indicates that the turnstile is initially locked. No final state is indicated since the turnstile is supposed to operate continuously. The automaton describes the functioning of the turnstile from the point of view of the system (the turnstile itself): the states represent the states of the turnstile.

2.1.2 Application to persuasion

In our instantiation of the state machines on behavior change, the **states** correspond to the behaviors performed by the user during the process of change, the **transitions** connect the states, and the **conditions** are the factors causing the transitions between behaviors.

In Figure 2.2, we provide an example of a generic process of behavior change. The automaton describes the process of change from the user point of view, representing what are the behavior the individual performs during the change, what are the targeted behaviors, what are the possible transitions from one behavior to another, and what are the factors that determine these transitions. We keep the model domain-independent to review the state of the art in a generic manner.

The **states** of the automaton correspond to the behaviors that users may perform during the process, and are represented by boxes in the automaton.

The possible **transitions** between the states are represented by oriented arrows linking the states. The color (green, blue or red) reflects the extent to which the transition is desired to achieve the targeted state. Green arrows indicate a progression toward the change; blue arrows represent a loop on a state or an iteration over different states; red arrows reflect regressions.

The **conditions** of a state machine capture the elements that **determine** the transition between states. They are noted "**Di**" in the automaton, and associated with a list of tentative strategies placed below the automaton.

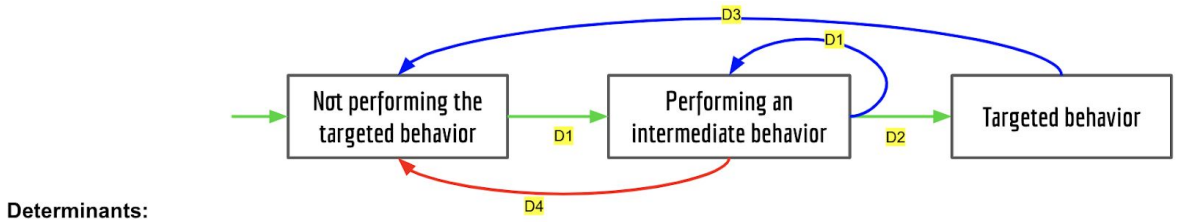


Figure 2.2	The concepts of **states** (boxes), **transitions** (arrows), and **determinant conditions** (Di) as systematic diagram to analyze the state of the art.

2.1.3 Focus on the conditions

The conditions that permit the transition between the states of automata for behavior change deserve a particular focus since they are related to important concepts including the **context**, the **behavior determinants**, and the **persuasive strategies** that may lead to these determinants.

Context is key. For this reason, in this dissertation on the conditions that determine the transitions, we cannot avoid an introduction on the context giving some examples of how it is accounted in the literature.

Albert Bandura, which in 2002 was ranked the fourth most-frequently cited psychologists of all times (Haggbloom et al. 2002), explains that environmental factors, cognitive factors, and personal factors have a reciprocal causality in determining the human behavior. Each one of these three elements influences the other two with an intensity that varies according to the situation and the current activity (Bandura 1986) (Wood and Bandura 1989).

Despite the importance of context is acknowledged, there is not a unique definition of context. This is due to the complex relationship between entities and factors that form the context. This complexity becomes more evident when designers are demanded to define context for a certain system. As a support of this affirmation, we provide the definition provided by AK Dey of context, which is one of the most cited in literature:

Context is "any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction

between a user and application, including the user and applications themselves." (Dey 2001).

The definition, from one side, denotes the complex relationship between the entities involved in the context, from the other, does not explore these relationships in depth to remain generic.

In literature, several frameworks have been proposed to account context. Oinas-Kukkonen and Harjumaa (Oinas-Kukkonen and Harjumaa 2008)(Oinas-Kukkonen 2010), for example, operate a definition of context aiming at the implementation of interactive systems in the Persuasive System Design model, basing on the analysis of the events, intents and persuasive strategies. Coutaz et al. (Coutaz et al. 2005) describe the context definition for interactive systems on three layers of abstraction: a lower "sensing layer" (e.g. sensors), a "perception layer" (providing symbolic interpretations) in the middle and a higher one "situation and context identification layer" for moving between situations and contexts. Calvary et al. (Calvary et al. 2002) in work "chameleon framework" define three dimensions to perform an adaptation of the interactive features: the "users" of the system (who are intended to use the system), the "hardware and software platforms" (the computational and interaction devices used by users), and the "environment" (the physical conditions where the interaction can take place).

During our investigation we have constantly faced the importance of context and for this reason in the following chapters we always dedicate a particular attention to this subject, defining and commenting on our approaches and choices to characterize this aspect.

The **behavior determinants** are the factors that lead an individual to a given behavior. They combine genetic and environmental features, accounting the behaviors precedently performed. Advances in technology (e.g., sensors) offer possibilities for capturing the context and inferring these determinants. Therefore rather than eliciting the behavior determinants in the nodes (which model the behaviors), we associate them with transitions, to explicit the determinants in the context of each transition.

The **persuasive strategies** are expected to favor the targeted behavior in the context of the given state, thereby possibly satisfying the determinants and enacting the transition. As further developed in the work, literature identifies different strategies that can be employed in specific situations to persuade the behavior change.

In the proposed automata describing the models of the literature, we will then use the term **determinant** to capture both these perspectives: the **behavior determinants,** which may be more dependent on the individual, and the **tentative persuasive strategies** that are supposed to be more generic. For simplicity, we use the same label (e.g. “D1”) for transitions that are favored by the same strategies. Also, we do not elaborate on the determinants of regressions, since these transitions should be avoided.

2.1.4 Illustration of behavioral state machines

Before characterizing the state of the art, we illustrate our formalism on our two case studies, CRegrette and Mhikes, presented in Section 1.6. We illustrate the basic concepts of states, transitions and determinants, focusing also on possible strategies.

Illustration on CRegrette

States:

1. The individual is performing the smoking behavior
2. The individual avoids/reduces the smoking behavior
3. The individual is not performing smoking behavior

Transitions and **determinants**:

- Progressions:
  - From **1** to **2** thanks to a commitment to the change
  - From **2** to **3** stimulus control
- Loops**:**
  - From **1** to **1** because insufficient awareness of the risks of smoking
  - From **2** to **2** rising motivation, self-control, commitment to the change
  - From **3** to **3** motivation from experiencing the benefits of not smoking
- Regressions**:**
  - From **2** to **1** desire caused by other people smoking in proximity
  - From **3** to **1** not being capable of handling abstinence.

Illustration on Mhikes

States:

1. The user searches or downloads hikes without performing them
2. The user performs hikes with Mhikes

Transitions and **determinants**:

- Progressions:
  - From **1** to **2** unsatisfactory well-being conditions
- Loops**:**
  - from **2** to **2** motivation gain from experiencing the benefits hiking
- Regressions**:**
  - from **2** to **1** focus loss, or frustration for not being able to hike

Next section characterizes the existing behavioral processes as state machines.

2.2 Theoretical frameworks

Change is not instantaneous. It is a process that takes time, evolving through different stages. This section reviews a set of behavioral and persuasive models aiming at analysing the process of change.

2.2.1 The transtheoretical model of behavior change

Description

In order to change, individuals need to be aware of the problem, to have a plan, and finally to avoid relapsing in the old behavior they wanted to change. Prochaska et al. represent these steps in the stages of the transtheoretical model of behavior change (Prochaska and Velicer 1997).

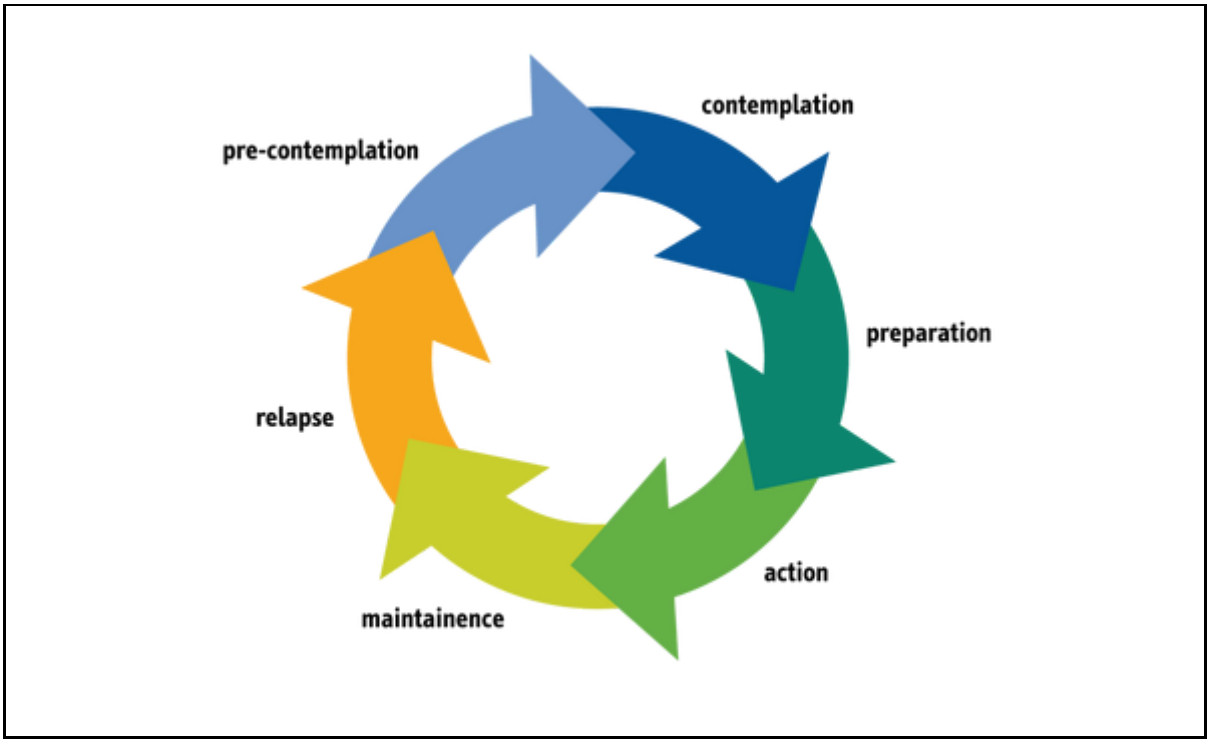


Figure 2.3	The stages of the transtheoretical model of behavior change (Prochaska and Velicer 1997)

The transtheoretical model of behaviour change formalizes the steps of the change into a progression that can be achieved using motivation, effort and energy (Miller and Rollnick 2002). The transtheoretical model has been used to promote behavior change in a variety of behaviors including: quitting smoking (DiClemente et al. 1991), physical activity (Marcus et al. 1992) (Marshall and Biddle 2001), dietary fat consumption (Greene et al. 1994) and others. Prochaska and their colleagues explain that, according to their research, changing appeared to be more effective if performed by using stages. In particular, they propose a list of core factors which appeared to be fundamental for the individuals: a set of **stages of change** (to describe their progression), their growing **decisional balance** (awareness that changing is bringing advantages in their lives), their **self-efficacy** (confidence in carrying on with the change) and a set **processes of change** (strategies to progress toward the stages).

The set of stages of the model, pictured in Figure 2.3, are the following:

- **Pre-contemplation**: in this stage, the individuals are not yet considering the idea of changing. They may be rejecting the change, not be interested or not be aware that performing the change is possible.
- **Contemplation**: in this stage, individuals are aware of the change. However they do not do anything to prepare to change, they may just consider the pros and cons of engaging the change.
- **Preparation**: in this stage, individuals start to get informed on how to engage the process of change. They may eventually build an action plan and collect further information on how to practically start to change.
- **Action**: in this stage, for the first time individuals perform the behaviors associated with the change. They try to respect their action plan and seek the help of other people that are performing the same behaviors. In this stage, individuals try to build habits aiming at permanently integrating the change in their lives.
- **Maintenance**: in this stage, the behavior associated with the change has been performed several times by the individual and he or she fights to

avoid any possible behaviors that may induce him or her to relapse to the previous stages.

A set of processes of change describing the strategies to progress from one stage to another was also formulated and it is the following:

- **Consciousness-raising:** it consists in getting informed on the targeted behavior in order to increase the awareness.
- **Dramatic relief:** this process involves the feelings connected to the change. For example, the fear and anxiety are feelings caused by approaching the change. Inspiration and hope are feelings that may be caused by interacting with individuals that succeeded in the change.
- **Self-reevaluation:** this process brings the individuals to build a new self image of themselves. In this vision, they are tasting how they will be after changing, their new lives and possibilities.
- **Environmental reevaluation:** this process brings individuals to realize that the operated change improves their lives and the lives of other people around them.
- **Social liberation:** individuals realize that society appreciates and supports their engagement to the change.
- **Self-liberation:** this process brings individuals to realize their self-ability and to produce an always more strong commitment to the change.
- **Helping relationships:** individuals look for other people performing the same behavioral change in order to get support and progress.
- **Counter-conditioning:** individuals find solutions to perform the targeted behavior as a substitute of other behaviors that they want to avoid/suppress.
- **Reinforcement management:** this process aims at reinforcing the positive behavior with a reward and at punishing the performing of any behavior that the individual wants to avoid/suppress.
- **Stimulus control:** individuals manage their environment in order to facilitate the performing of the targeted behavior. They may introduce facilitators for performing the behavior or cues that remind the positive behaviors and avoid the negative ones.

West (West 2005) lists several empirical challenges to the Transtheoretical Model, and argues that it should be discarded because it contains fundamental theoretical flaws. One key flaw in considering habits is that the Transtheoretical Model assumes

that people make stable rational choices, rather than being subject to nonconscious influence such as impulses or habits. Another critic moved by Bandura concerns the stages of the model, which appear as ‘arbitrary pseudo-stages’ rather than genuine stages (Bandura 1998).

State-machine-based model

The transtheoretical model of behavior change intrinsically represents the process of change as a process. A state machines-based model (Figure 2.4) can be operated considering five states related to the five stages described in the model. The states are sequentially connected by transitions from the initial state to the final one, with the processes of change as determinants or strategies. The last looping transition (D4) captures the maintenance behavior where individuals fight to avoid the relapse. The relapse phenomenon, represented by a regression, may bring individuals from performing the target behavior to non performing it.

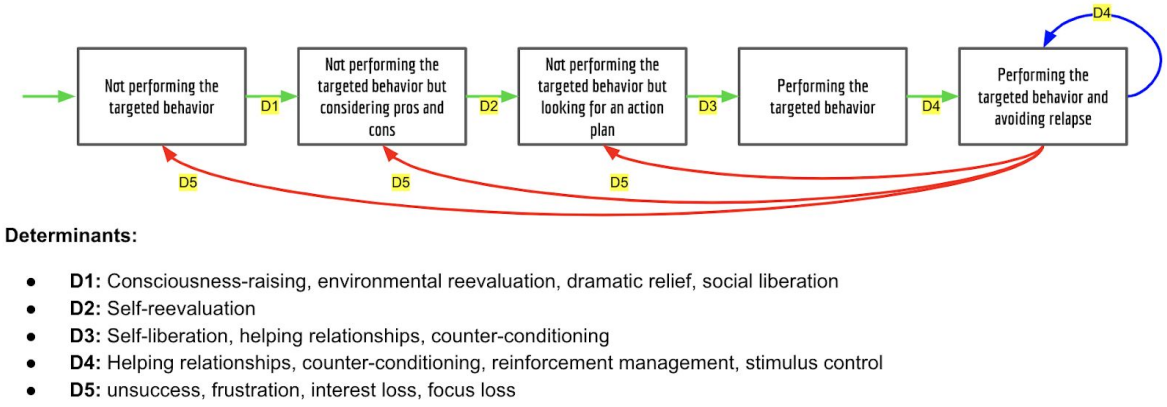


Figure 2.4	Possible state-machine-based representation of the Transtheoretical Model of Behavior Change.

2.1.2 Conditioning and Influencing

Description

The operant conditioning describes how a behavior can be constructed by using stimulus-response pairs formed outside conscious decision making (West and Brown 2013). This theory has been first investigated by Skinner (Skinner 1963), and can be applied to persuade users to perform targeted behaviors.

The theory presents two main approaches: punishing and reinforcing a given behavior.

1. **Reinforcement** is used to increase the behavior and can be of two types:
    a. **Positive** consisting in adding repetitive stimulus following correct behavior (e.g. prizing the runner to have completed a new run),
    b. **Negative**, which can ulteriorly be decomposed into:
        i. **Escape** consisting in removing noxious stimuli following correct behavior (e.g.. clicking on an unread notification; users often found them annoying, but actually by clicking they perform the action)
        ii. **Active avoidance** consisting in performing behaviors just to avoid noxious stimulus (e.g. in order to prevent the generation of a notification that the user may find annoying, he/she performs the action before the system creates the notification).
2. **Punishment** is used to decrease the behavior and can also be of two types:
    a. **Positive** consisting in adding noxious stimuli following the behavior (e.g. decreasing the rating of runners if they do not achieve their fixed goal)
    b. **Negative** consisting in removing an appetitive stimulus following the behavior (e.g. impeding the runners to chat with other runners in the application if they do not attend their fixed goal).

In persuasion, the reinforcement strategy is often preferred, since the punishment can cause frustration to the user, constituting a huge risk for the achievement of the change. The causality effect on reinforcement is found to be effective in several contexts involving repetitive actions such as marketing (buying goods as action), education (learning as action), gambling (risking/playing/betting as action) and others.

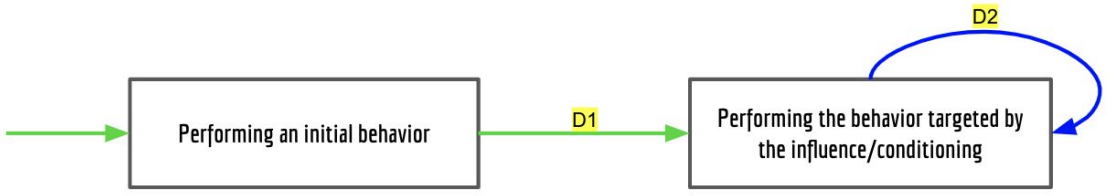
The operant conditioning investigates also on the efficacy of providing stimuli in different modalities. A stimuli, indeed, can be provided by using a fixed or variable ratio, or by using a fixed or variable interval. For example salaries given each month have a fixed ratio and a fixed interval, but considering the winning of a slot machine, it has a variable ratio (the winnings) and variable interval (the times we pull the lever) (Staddon and Cerutti 2003; Bijou 1957).

When a person's emotions, opinions or behaviors are affected by others intentionally or unintentionally, social influence theories apply. Robert Cialdini, expert in influence, identified a set of principles altering the causality of behavior (Cialdini and Garde 1987):

- "**Reciprocity**": the tendency to return a favour,
- "**Scarcity**: the desire of having things that are in short supply, unique, rare,
- "**Consistency**": if people commit, orally or in writing, to an idea or goal, they are more likely to honour that commitment to be congruent with their self-image,
- "**Consensus**": when not sure on what to do, people look at others' behavior,
- "**Liking**": people are easily persuaded by people they like, and
- "**Authority**": people tend to obey authority figures (e.g. experts, officers, etc.). Authority has been initially investigated by the psychologist Milgram, who measured the willingness of subjects to obey an authority figure. This person was instructing subjects to perform acts conflicting with their personal conscience (Milgram and van Gasteren 1974).

State-machine-based model

The instantiation of the concepts of influence and of operant conditioning on the state machine focus more on the transition aspect rather than on the whole process. A main transition captures the passage from an initial state, where the behavior is not performed, to a final one where the behavior is performed. The operant conditioning and influence provide the determinants to perform the behavior, and to maintain it. We underline that the determinant punishment cannot be used as a determinant to maintain the behavior, since it is aimed to decrease a behavior. Receiving a penalty because exceeding the speed limit (initial behavior), for example, conditions the individual to reduce the speed (targeted behavior). Once the individual drives below the speed limit, the punishment can't be used anymore, reward instead continues to be an alternative.



Determinants:

- **D1:** influence (e.g. commitment), operant conditioning (e.g. reinforcement, punishment)
- **D2:** influence (e.g. commitment), operant conditioning (e.g. reinforcement)

Figure 2.5	Possible state-machine-based representation of the influence and conditioning theories

2.2.2 The Fogg Behavior Model

Description

Bj Fogg is one of the precursors of the persuasive technology field. With his article Captology (Brian J. Fogg 1998) for the first time he defined persuasive technologies and three different roles for persuasive systems:

- The role of **tool:** the persuasive system changes the attitude of users helping them to accomplish their targeted results more easily (e.g. a calculator). These systems increase the individual capabilities;
- The role of **medium:** the persuasive system conveys either symbolic content or sensory content (e.g. a flying simulator). These systems provide experience to the individual;
- The role of **social actor:** the persuasive system adopts animated characteristics, plays animated roles, or follows social rules or dynamics (e.g. a personal e-assistant). These systems create relationships between the individuals and the technology.

Figure 2.6	The BJ Fogg triad (B. J. Fogg 2002)

As mentioned, the action stage is the stage of the transtheoretical model, where the individual decides to convert his/her plan into change into a physical commitment. Bj Fogg in his Fogg Behavior Model describes the likelihood of a behavior to become action based on three variables: the motivation, the ability and the trigger (Brian J. Fogg 2009).

In the Fogg Behaviour Model the two core characteristics of ability and motivation are used to describe individuals attempting a behavior change: their **motivation** to change and their **ability** in doing so. These two variables can assume different reciprocal values in respect to the considered change. This relation is described by the activation threshold line representing the fact that a positive combination of these two variables is necessary as a prerequisite for performing a new behavior. The third element is the trigger which can be of three types:

- A **spark** motivates the behavior performing (e.g. a notification on the mobile phone saying what to do next),
- A **signal** indicates or reminds something (e.g. a statistic on the steps done per day to motivate the person to have a walk),
- A **facilitator** is something that makes the behavior easier to be done (e.g. buying separated trash bin to adopt the practice of recycling).

Indeed the behavior is likely to happen if the target behavior is sufficiently motivated, if the individual has the ability to perform the behavior, and is triggered to perform the behavior. These three factors must simultaneously occur and reach a certain threshold to trigger an effective change. Otherwise the behavior is likely not to happen (Brian J. Fogg 2009).

For example in order to persuade people to fill an online satisfaction survey, they should be having some free time to complete the form (ability), they should be interested in improving something or eager to give their feedback (motivation) and they should be informed that the online survey is available, for instance by receiving a newsletter (trigger). If one of these three components is not present, it is unlikely that the person will fill the survey.

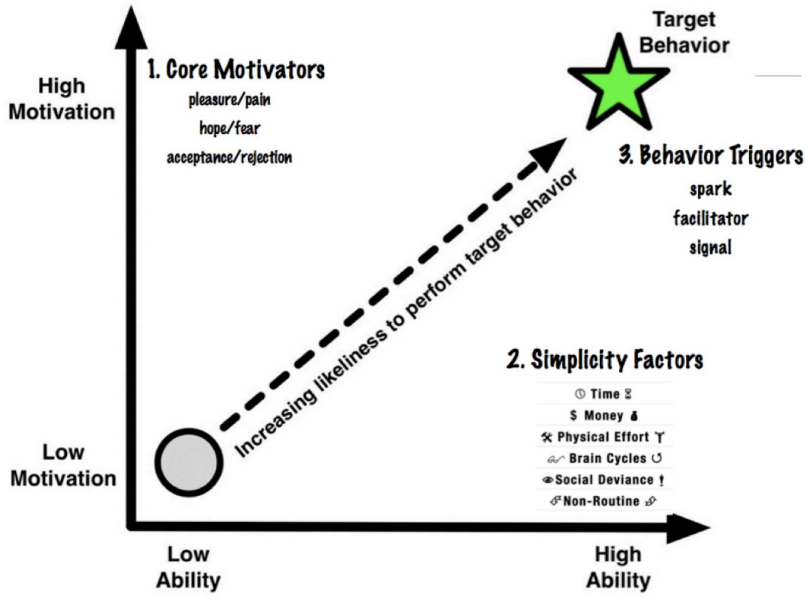


Figure 2.7	The Fogg Behavior Model (Brian J. Fogg 2009)

Some specific elements can affect the level of ability and motivation. Elements affecting motivation are central to the human experience and are: Pleasure/Pain, Hope/Fear, Social Acceptance/Rejection. The elements that affect ability the most is simplicity, in other words, make the path to a change easier. The six factors making simpler the change are: time, money, physical effort, brain cycles, social deviance, and non routine (Brian J. Fogg 2009). According to Fogg, these simplicity factors are linked like the rings of a chain: if any one is weak, then the complete chain may break and, simplicity would be lost (Brian J. Fogg 2009).

The Fogg Behavior model captures the fact that motivation and ability are variable factors. This variability is of particular interest when related to the evolution over time. For example, individuals may learn new things during the change improving their abilities: practice hard a certain sport, for example, leads to develop tricks and reflex that improve the performances. Motivation also may vary. For example, when individuals attempt a behavior change their initial motivation is often very high. For instance, people often target a change in their diet to improve their health. Initially, when highly motivated, they simply avoid fats and excessive carbohydrates. Unfortunately it is possible that after a first period in which they are very motivated, they may miss the taste of food they used to eat. The new alimentation may be found unsatisfactory and then they quit the change. The '**motivational waves**' approach explains that the high motivational peaks should be used to build motivational tools to be exploited in the future in case the motivation lowers. In the

alimentation example a high motivation may be used also to learn new recipes to improve the way of cooking healthy. In case the motivation decreases because of a lack of enthusiasm, a new recipe can provide a new input to pursuive the change. If the recipe is also found tasty for the individual, he or she will get a self rewarding.

Along with the motivational waves, Bj Fogg explains that the ability may be increased using the approach of '**tiny habits**'. For example, for a non expert runner who wants to reach the 20 Km of running per week, starting with a first itinerary measuring 10 Km may cause trouble: motivationally (because in case of failure he or she could develop a frustration feeling) and physically. Tiny habits is a persuasive strategy that suggests to start with simple objective and to increase progressively the difficulty in order to reach the desired objective. The runner for example could first try to master itineraries of 2 or 3 Km and when comfortable with that pushing their effort by 1 Km until reaching the 20 km. Accomplishing the easier objectives (2/3 Km in the example) helps in developing their own self-confidence to indeed increase their abilities. Motivation can also be boosted by this process associating a reward for each completion of the easy behavior. Fogg suggests also to follow this pattern when applying the tiny habits: "Everytime <context> I will <action> and then celebrate". For example "Every time I can choose between elevator and stairs I will chose the stairs and then celebrate being proud of my choice".

Bj Fogg developed a method to classify 15 different targetable behaviors. This contribution is known as the Behavior Wizard (Fogg and Hreha 2010). Behaviors changes are characterized by using two main axis, the types of behavior (also called flavours by Fogg) and their scheduling. 5 flavours for the behavior are described: performing a new unfamiliar behavior, performing a familiar behavior, increasing a behavior, decreasing a behavior or stopping a behavior. For the scheduling, 3 main types are listed: performing the behavior one time, for a period of time or from now on[1]. The combination of these 3 schedules and of the 5 flavours generates the table illustrated at Figure 2.8 .

[1] A successive extension of this study aimed at mapping the behavior goal from Facebook onto the framework extended the number possible scheduling for behaviors from 3 to 7 types (Fogg 2009).

	Green behavior Do new behavior, one that is unfamiliar	Blue behavior Do familiar behavior	Purple behavior Increase behavior intensity or duration	Gray behavior Decrease behavior intensity or duration	Black behavior Stop doing a behavior
Dot behavior is done one-time	**GreenDot** Do new behavior one time *Install solar panels on house*	**BlueDot** Do familiar behavior one time *Tell a friend about eco-friendly soap*	**PurpleDot** Increase behavior one time *Plant more trees & local plants today*	**GrayDot** Decrease behavior one time *Buy fewer bottles of water now*	**BlackDot** Stop doing a behavior one time *Turn off space heater for tonight*
Span behavior has specific duration, such as 40 days	**GreenSpan** Do new behavior for a period of time *Carpool to work for three weeks*	**BlueSpan** Do familiar behavior for a period of time *Bike to work for two months*	**PurpleSpan** Increase behavior for a period of time *Take public bus for one month*	**GraySpan** Decrease behavior for a period of time *Take shorter showers this week*	**BlackSpan** Stop a behavior for a period of time *Don't water lawn during summer*
Path behavior is done from now on, a permanent change	**GreenPath** Do new behavior from now on *Start growing own vegetables*	**BluePath** Do familiar behavior from now on *Turn off lights when leaving room*	**PurplePath** Increase behavior from now on *Purchase more local produce*	**GrayPath** Decrease behavior from now on *Eat less meat from now on*	**BlackPath** Stop a behavior from now on *Never litter again*

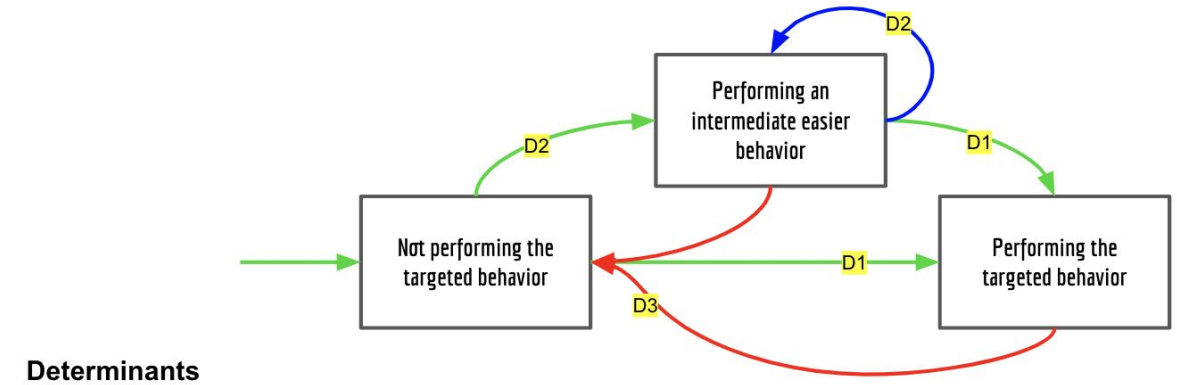
Figure 2.8	The Fogg Behavior Wizard (Fogg and Hreha 2010)

State-machine-based model

The Fogg behavior model, similarly to operant conditioning and influence, focuses on the transition between doing and not doing a certain behavior, rather than considering the whole process. Relating it to the transtheoretical model (if possible), for instance, we would not mapp the five stages on it but maybe just the contemplation, preparation and action one.

The FBM model can be represented into a state machine by using three states. An initial one where the behavior is not performed because there is not sufficient motivation and/or ability to cross the action threshold. From the theory of baby steps, we can place a second state in which the user performs an intermediate behavior aiming at increasing the motivation and ability. A looping transition on this state captures that several intermediate behaviors may be necessary to gain the demanded motivation or ability to progress. When the motivation and ability are sufficient to cross the action threshold, then the individual reaches the last state,

where the targeted behavior is performed. The regression to initial state can happen when motivation or ability decreases.



Determinants

- **D1:** ability and motivation > action threshold
- **D2:** motivation and ability are low
- **D3:** motivation or ability lowers

Figure 2.9	Possible state-machine-based representation of the Fogg Behavior Model.

2.2.3 The Hook model

Description

Nir Eyal (Eyal 2014) produced a set of guidelines to enhance technological products with habit forming. His investigation concerned the process that brings people to make an always increasing use of a certain class of products such as software applications and social networks. In particular, he claims that such products may evolve within two main stages: a first in which there is no real need for using them (this stage is called vitamin), a second one in which not using them causes pain to the user (this stage is called painkiller). Social media such as Facebook, Instagram or timekiller games for smartphones are pertinent examples of this process: initially they are used occasionally (as vitamins), then in some cases they became addictive and not using them cause pain (using them is like taking a painkiller). Eyal theorizes a model made of four stages, the Hook model.

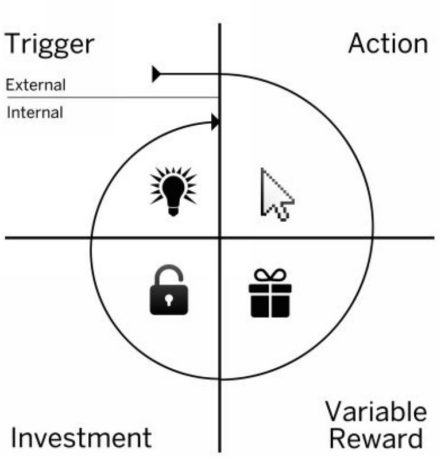


Figure 2.10	The phases of the Hook model (Eyal 2014)

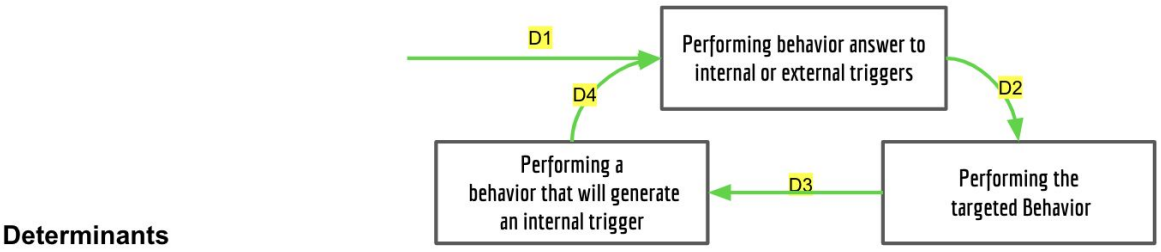
The stages are the following ones:

- The **trigger** stage: it recalls the ideas of B.J. Fogg describing an event that makes the user react. Two different types of triggers can be prompted to the user: the external ones (including the ones theorized by Fogg), where the system tells the user what to do, and the internal ones, where the user that knows what to do. According to Eyal, internal triggers are often produced by negative emotions. For example, to overcome a sense of loneliness, individuals may use products such as social network to get in touch with other people;
- The **action** stage: it follows the principles of the Fogg Behavior Model, i.e. to induce the behavior a combination of sufficient of motivation and ability, and the presence of a prompt are necessary;
- The **reward** stage: it is used to acknowledge the behavior performed in the action stage.
  Eyal theorized three types of rewards:
  a. The “reward of the hunt” (materialized by resources and goods such as money, food),
  b. The “reward of the self” (represented by personal gratification for being consistent with their one’s ideas, mastering an ability or completing a particular task),
  c. The “reward of the tribe” (which is given by another user through social-likes, comments, ratings, reviews).

- The **investment** stage: is where users perform a small action that will generate an internal trigger in the future to restart the process. For example, after reviewing a restaurant in a food-application (investment), the system may ask to review other restaurants (trigger) in order to stay in the loop of the application usage.

State-machine-based model

The Hook model targets the engagement of people in technology. The four phases (trigger, action, reward, and investment) explain what the system should do from the point of view of the system. Taking the perspective of the user we can evidence three states. A first where the user receives the trigger (internal or external) and feels the necessity to perform the associated behavior. The behavior is then performed in the second state. A successive iteration target to induce the user to perform a behavior that later will generate the new trigger to induce an iteration of the states. In Figure 2.11 we have chosen the green color to represent the transition related to determinant (D4), meaning that it is not simply a loop-transition (blue-colored) but that the iteration is considered as a real progression. Not having this transition indeed would break the chain of the four phases described in the model.



Determinants

- **D1:** necessity of responding to an internal or external trigger
- **D2:** motivation and ability > action threshold (see Fogg Behavior Model)
- **D3:** Eyal's rewards (Hunt, Tribe and Self), operant conditioning, reward theories
- **D4:** emotions and experiences trigger the user to use the system again

Figure 2.11	Possible state-machine-based representation of the Hook Model.

2.2.4 MOA and COM-B model

Description

The Motivation Opportunity Ability (MOA) model proposed by Ölander, F. & J. Thøgersen (Ölander and ThØgersen 1995) comes initially from the marketing field

but has also been used to explain behaviors (Hamari, Koivisto, and Pakkanen 2014) and persuasive techniques (Hughes 2007). Also this model mentions the concept of motivation (combination of attitudes and social norms that forms an intention) and ability (habit and knowledge needed to perform the behavior). Opportunity includes all factors external to an individual that 'make the behaviour possible or prompt it' (Michie and Prestwich 2010). MacInnis (MacInnis, Moorman, and Jaworski 1991) explains that the adoption of a behavior by subjects is influenced by their motivation to adopt the behavior, by their capacities and by the opportunities offered by the environment.

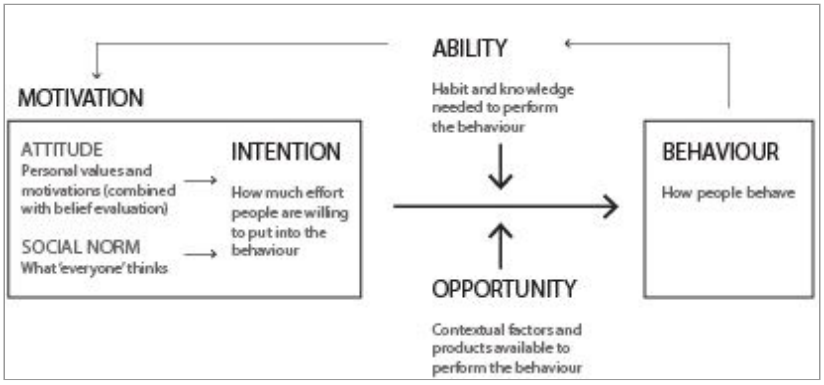


Figure 2.12	The Motivation Opportunity Ability (MOA) model (Ölander and ThØgersen 1995)

The COM-B Model considers the causal factors of motivational opportunities and (cap)abilities encountered in the MOA model. Additionally to the MOA approach, the COM-B comprehends also a framework that helps to use the different persuasive strategies. The framework is the Behavior Change Wheel (Michie, van Stralen, and West 2011) and was produced thanks to the literature review operated by its authors on 1267 scientific articles.

The Behavior Change Wheel combines the **sources of behavior** (motivations, abilities and opportunities) with a set of **intervention functions** (restriction, education, persuasion, incentivisation, coercion, training, environmental restructuring, modelling and enablement) and a set of **policies** (communication marketing, guidelines, fiscal, regulation, legislation, environmental/social planning

and service position). Then the framework provides the “links” between these components, such as which intervention function can be used for each source of behavior, or which policy categories works with each determinate intervention function. These links, finally help at choosing the intervention to be actuated in order to trigger the behavior change and to be implemented in the interactive persuasive system.

State-machine-based model

The instantiation of the MOA model on a state machine focuses on how specific transitions happen, rather than describing the whole process. A main transition captures the passage from an initial state, where the individual is not performing the behavior (but is preparing for it) toward a state where the behavior is performed. The first state includes a loop transition happening until the preparation is not completed. The opportunities offered by the environment permit the individual to perform the targeted behavior in the final state. This state gives new opportunities and makes discovering new abilities to individuals, bringing them to prepare a new behavior.

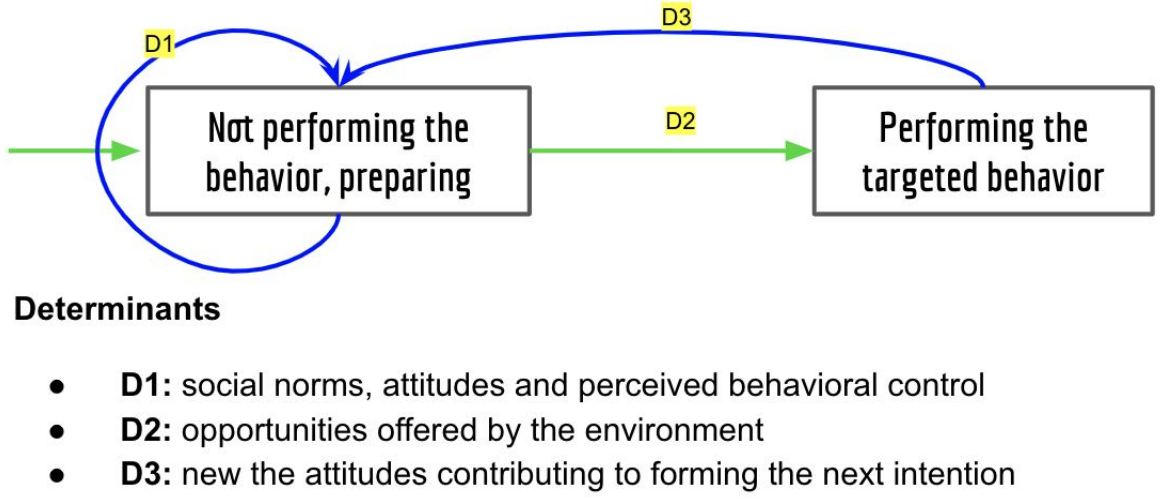


Figure 2.13	Possible state-machine-based representation of the Motivation Opportunity Ability Model.

2.2.5 The habit alteration model

The Habit Alteration Model (HAM) theorized by Charlie Pinder et al. (Pinder et al. 2018) synthesises the Dual Process Theory, modern habit theory, and Goal Setting

Theory in a common model. The aim is to simplify these theories highlighting how the internal and external factors generate both habitual and non habitual behaviors.

The model considers the two types of processes from the Dual Process Theory: type1 processes are fast, automatic, nonconscious, associative; and type 2 processes are slower, deliberative, conscious (Kahneman 2011) (Strack and Deutsch 2004).
According to the theory, four components generate a behavior: (1) type 1 processes, which are the ones that relate cues to behavioral impulses, (2) type 2 processes generated by explicit intentions, (3) the cues of context, and (4) the individual differences (e.g. the impulsivity). The model is based on three phases: filter, prepare, and act as pictured in Figure 2.14. The dashed lines represent the processes that may run whilst the solid ones represent the continuously running processes.
In the first phase (**filter**), the processes of type1 and type2 generate a set of cues that are the inputs for the second phase (**prepare**). In the prepare phase, the type 1 processes are memorized in the implicit memory and will generate a stack of impulses; the type2 processes are memorized in the explicit memory and will generate a stack of intentions.
Impulse and intentions will compete between each other and the ones crossing an act threshold will be delivered to the act phase.
The last phase (**act**) will take the impulse and intentions to generate a response and an outcome which will constitute new inputs of type1 and type2 processes for the first phase.
The repetitions of the three phases in stable context becomes more automatic, and may bring people to pass from type 2 processes (slow and conscious) to type1 ones (faster and associative).

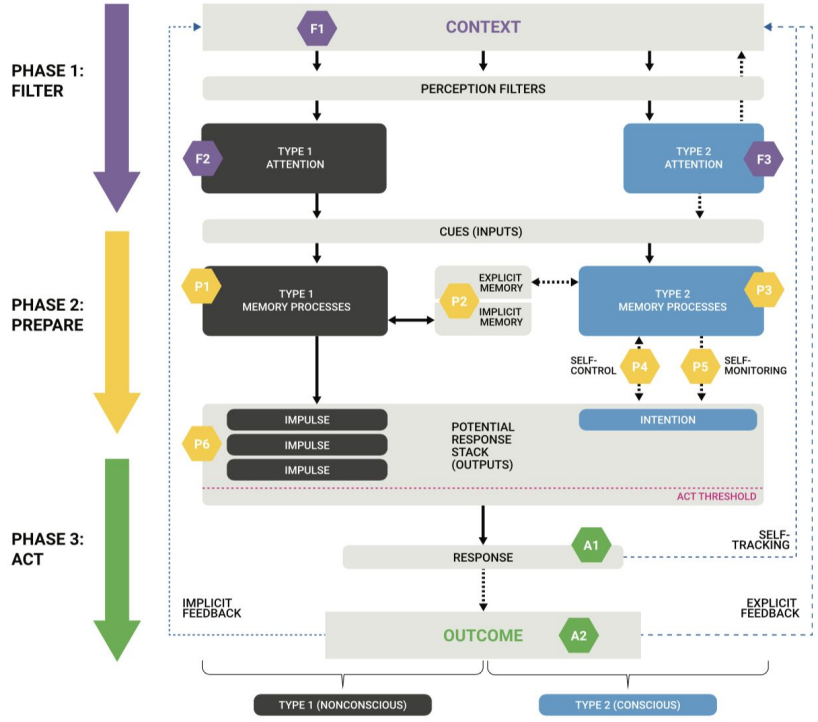


Figure 2.14	The Habit Alteration Model by Pinder et al. (Pinder et al. 2018)

State-machine-based model

The HAM model is explicitly based on three iterative phases: filter, prepare and act. During the first two phases, the individuals are not performing the targeted behavior, but they are attentive to the inputs provided by the context (first state). These inputs bring individuals to build a stack of priorities (preparing) in the second state. These priorities made of impulses and intentions will be translated into behaviors in the last state. In the final state, individuals perform the behavior but analyse also the response and the outcomes of their behavior. This analysis provides them with a mechanism of feedback and self-tracking that may produce new contextual inputs. This may bring the individual to iterate the process, and this possibility is captured by the blue-colored looping transition in Figure 2.15 from the final to the initial state.

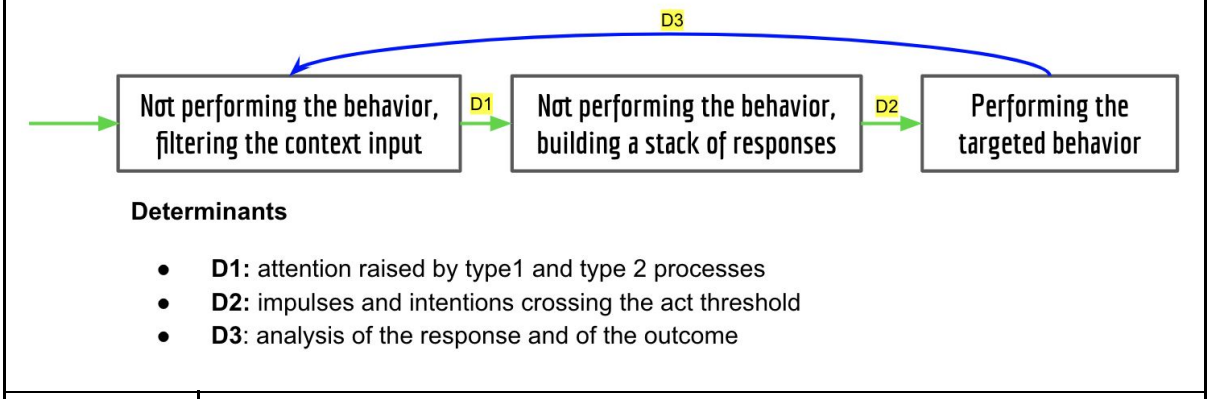


Figure 2.15	Possible state-machine-based representation of the Habit Alteration Model.

2.2.6 Discussion

We have performed a structured review of a set of models and theories coming from the state of the art in persuasion. In this review we have highlighted how the process of change and the evolution of the behaviors are considered in each theory, by using different state machines. This approach was motivated by the need of having a general structure to operationalize these models which are numerous and which present several diversities, such as their main focus or their level of granularity.

Some theories for example seem to mix the analysis of behaviors of individuals with the strategies that the system has to employ to support the change. This is the case of the Hook model which includes both a stage of action (where the user performs the behavior) and a stage of reward (where the system provides the user with reward). In this case, it is not possible to instantiate a state machine, because the states represent different entities (behavior and features) unless a unification of the perspective is performed (as we have done in the presented automata in Figure 2.11).

The transtheoretical model of behavior change, instead, describes the whole evolution of the change and focuses on the individual. The model gives a global picture of the behaviors performed, of the emotional states and of the transitions for all the process. In this case, tracing the state machine automation interests the three components of the automaton: states, transitions, and determinants.

In other models, the focus is on how the transitions between the behaviors are realized. In the influence theory, in the motivation opportunity ability model, in the habit alteration model and in the operant conditioning theory, the specific behavior is

not analyzed, but the determinants that bring to reach it. In other words, these models focus on the transitions and determinants that bring from not-performing the behavior to performing it.

This review finally shows that, in order to operationalize the models of the state of the art, it is necessary to have a general structure able to represent the behavior processes and their evolution over time. This structure needs to capture the granular differences between the approaches, modelling just small parts of the change (such as the transition) or bigger parts of it (the entire process or different states). Finally it should provide means to model all aspects of the process, such as the behaviors, the context, the determinants, being ultimately able to associate these information to concrete persuasive features to be driven to user.

In Chapter 3 we will be proposing a means to fulfill these requirements, introducing the concept of persuasive paths, persuasive events and interactive roles.

2.3 Design the process of change

In previous section, we have presented models and approaches of the state of the art analyzing how they approach the process of change. In this section, we aim at investigating on how these processes can be operationalized. Having this information would permit designers to associate specific persuasive features to these models making a link between the system design and the system implementation.

In order to investigate this operationalization, we introduce the Persuasive System Design model of Oinas-Kukkonen and Harjumaa (Oinas-Kukkonen and Harjumaa 2009)(Oinas-Kukkonen 2010) as reference frame, which is one of the most investigated frameworks for the design of persuasive technologies. Successively we propose some features that can be used in some specific states of the change, and other features of general purpose. Finally, we conclude the section with a synthesis on this analysis.

2.3.1 Design persuasive features for the process of change

The Persuasive System Design (PSD) of Oinas-Kukkonen and Harjumaa (Oinas-Kukkonen and Harjumaa 2008) is a framework designed to answer to two specific needs: (1) making psychology-related theories and models accessible to designers and computer scientists, and (2) providing powerful tools and methods for designing and implementing persuasive technologies. The framework first defines a set of **postulates** to be addressed when designing persuasive systems,

successively defines the notion of **context** and then enunciates a set of **qualities** for the persuasive systems associated with specific implementable features.

The seven **postulates** that need to be addressed when designing or evaluating persuasive systems mentioned by the authors are the following:

- The information technologies have always an influence on users: they are not neutral,
- Users need to have consistent representations of their personal values in the system,
- Persuasion can be driven directly or indirectly to the user, according to the theory of likelihood (Petty and Cacioppo 1986),
- Persuasion is incremental (e.g. Tiny habits of Fogg (Brian J. Fogg 2009)),
- Persuasion has to be transparent to user, mentioning the real purpose for which it was developed,
- Persuasion should respect the privacy of user,
- The system needs to be easy to use and to bring a real benefit to users.

The **context** is analyzed by the authors, by considering three principles: the intent, the event and the strategy.

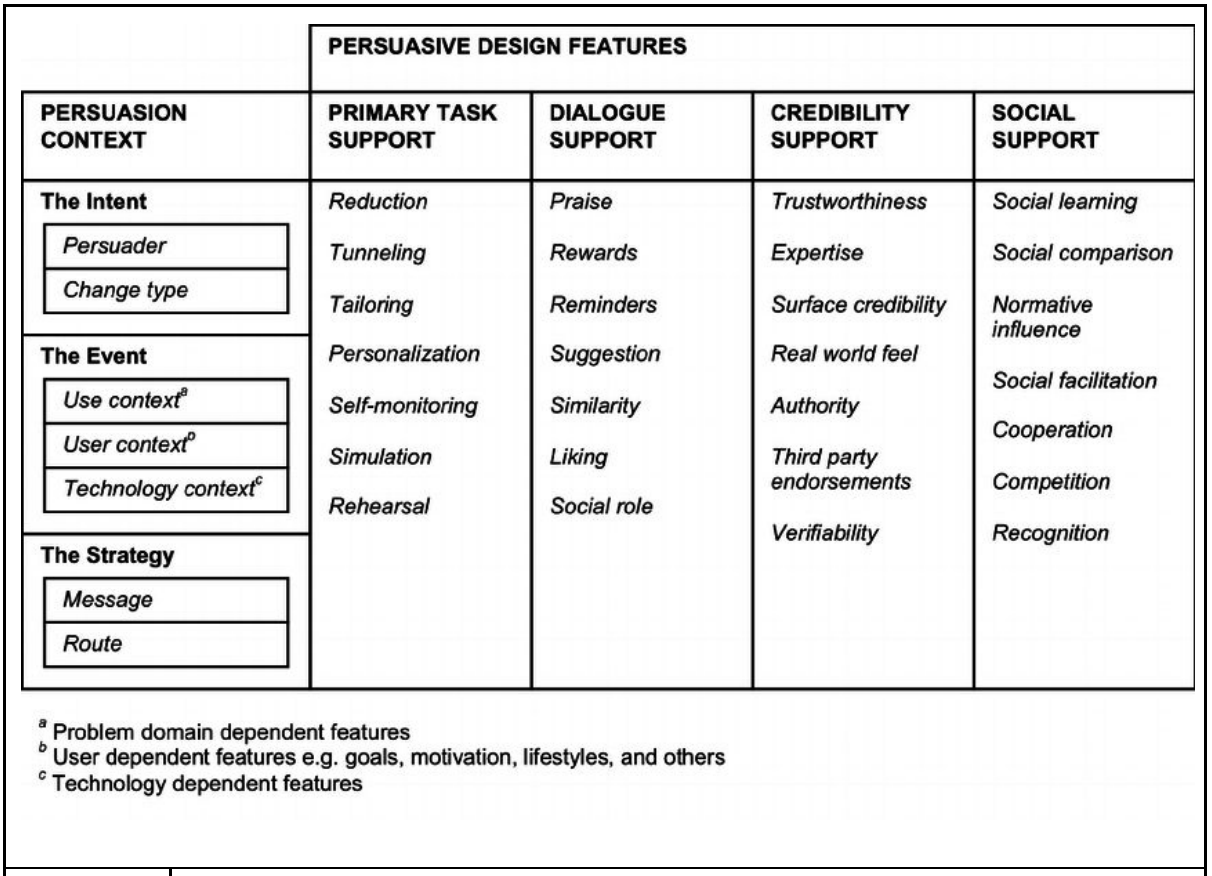
- The **intent**. It identifies where the intent of persuading comes from. Fogg details three possibilities (Brian J. Fogg 1998):
    - An **endogenous** intent: coming from the producer of the technology (e.g. a game designed to teach ecology to children),
    - An **exogenous** intent: caused by external factors (e.g. a mother giving a pet to her son in the hope that the son becomes more responsible),
    - An **autogenous** intent: caused by the users themselves (e.g. download calories counter in the hope of managing the weight).
- The **event**. It specifies the domain of persuasion. To define the event, it is necessary to define which will be the domain of use (health, sport, education), what will be the contextual information of the user (the type of goal, the needs, the cultural implication) and the context of the technology (which platform will be used, which advantage has a device in respect to another).
- The **strategy**. It consists in identifying the persuasive *message* to be delivered to the user and the route that permits the delivery.

Finally authors propose four design qualities for the persuasive system:

- The **primary task** support carrying out the user's primary tasks,

- The **dialogue** support defining how to implement the computer-human dialogue,
- The system **credibility** to confer more credibility to the system,
- The **social** support to motivate users by leveraging social influence.

These system qualities are associated with a set of features detailed in the work (Oinas-Kukkonen and Harjumaa 2009), and reported in the table in Figure 2.16.



PERSUASION CONTEXT	PERSUASIVE DESIGN FEATURES: PRIMARY TASK SUPPORT	DIALOGUE SUPPORT	CREDIBILITY SUPPORT	SOCIAL SUPPORT
The Intent *Persuader* *Change type*	*Reduction* *Tunneling* *Tailoring*	*Praise* *Rewards* *Reminders*	*Trustworthiness* *Expertise* *Surface credibility*	*Social learning* *Social comparison* *Normative influence*
The Event *Use context*[a] *User context*[b] *Technology context*[c]	*Personalization* *Self-monitoring* *Simulation*	*Suggestion* *Similarity* *Liking*	*Real world feel* *Authority* *Third party endorsements*	*Social facilitation* *Cooperation* *Competition*
The Strategy *Message* *Route*	*Rehearsal*	*Social role*	*Verifiability*	*Recognition*

[a] Problem domain dependent features
[b] User dependent features e.g. goals, motivation, lifestyles, and others
[c] Technology dependent features

Figure 2.16	Persuasive System Design Model (Oinas-Kukkonen and Harjumaa 2009)(Lehto and Oinas-Kukkonen 2011)

Users involved in the process of change travel different behavioral states and perform different behaviors over time. We have represented this evolution with the finite state automaton described in Section 2.1 . Persuasive systems supporting behavioral change need thus to adapt the persuasive strategies to each one of these states. In this section, we aim at presenting some examples of these persuasive strategies and to explain how they can be used along the process of change.

In order to illustrate some of the PSD framework features, we simplify the behavioral change into three generic states: the user not performing the targeted behavior, the user performing some intermediate behavior, and the user performing the targeted

behavior. Following the semantic introduced at Section 2.1, we can picture this generic behavioral change process with the following automaton.

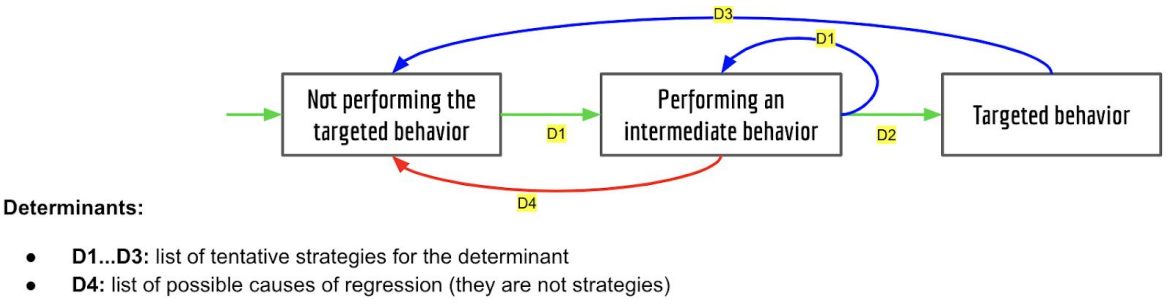


Figure 2.17	Example of a generic behavioral process.

The process evidences two fundamental states, the initial one, in which the targeted behavior is not performed and the final one, which is the objective of the behavior change. Since behavioral change is a process we have inserted an intermediate state that represents this evolution. This state may be decomposed in sub states with more intermediate behaviors (such as it happens in the TTM see Section 2.2.1).

The features detailed in the PSD framework are not linked to specific states of the process of change by the authors: they may work in a specific state or in more than one. In the following, however, we are going to propose a set of persuasive features for each state. For this reason, reading the following paragraph, the reader should be reminded that (1) the proposed features are just an example of the ones applicable at each state, that (2) the features may work also in different states, and that (3) the objective of this dissertation is to illustrate how to operationalize the passage from one behavioral state to another.

2.3.1.1 Initial state: the user is not performing the targeted behavior

Persuasive strategies associated with this initial state need to target the awareness of the user: the user needs to be aware of the possibility of changing. The system should facilitate the user in engaging the change and it should give means to teach the user on how to approach the intermediate behavior.

We have chosen for this stage the following three features as examples:

- **Reduction (Primary Task):** the persuasive system should as much as possible reduce the effort of the user in performing the first steps toward the

change.

Example: a web-blog on healthy alimentation including a button to locate the nearest bio-shops.

- **Simulation (Primary Task):** the persuasive system provides a simulation showing the benefits of adopting the change.

  Example: a smoking-cessation persuasive system showing how health values can vary after one month of non smoking (e.g. blood pressure reduction).

- **Third party endorsements (Credibility support):** the system may show that there is an endorsement from well-known sources.

  Example: a persuasive system designed to improve physical activity may mention that the process of design has involved famous athletes.

In order to give a concrete example of one of these principles, we provide a screenshot of the website of the Smokefree60+[2] program of the U.S. Department of Health and Human Services. The program aims to sensibilise smokers on the risks deriving from smoking

[2] https://60plus.smokefree.gov/quit/health

Figure 2.18	Picture taken from Smokefree60+ program website showing the benefits of non smoking over time.

In Figure 2.18 we can see an application of the principle of **simulation**: the website shows simulated improvements in health after 20 minutes of non smoking up to 10 years.

2.3.1.2 Intermediate state: the user is performing an intermediate behavior

Persuasive strategies associated with the intermediate state need to persuade the users to refine their behavior. Users need to develop their own strategies toward the targeted behavior and may need tips and advice. Users may permane in this state for some time performing these intermediate behaviors until they do not feel prepared to perform the final targeted behavior.

We have chosen the following three features as examples for this stage:

- **Reminders (Dialogue support):** the persuasive system should remind the user to perform the behavior. This is particularly important to help the user in developing routines. The employment of this feature should concern the first states of the change, since in the last ones the individual is supposed to be able to perform the behavior more autonomously.
  Example: in order to save energy the system reminds the user to check if all the lights are turned off before leaving the apartment.
- **Tunneling (Primary Task):** the system should be able to guide the user toward the intermediate behavior approaching more and more to the targeted one.
  Example: a system for improving sport activities may order the behaviors by difficulty to facilitate the user progression from easy behaviors to difficult ones.
- **Social learning (Social support):** the system should provide users with means to learn from other people performing the same behavior.
  Example: a smoking system reporting the tips and the advice of people that quit.

In order to give a concrete example of one of these principles, we provide a screenshot of the application QuitNow available for smartphones.

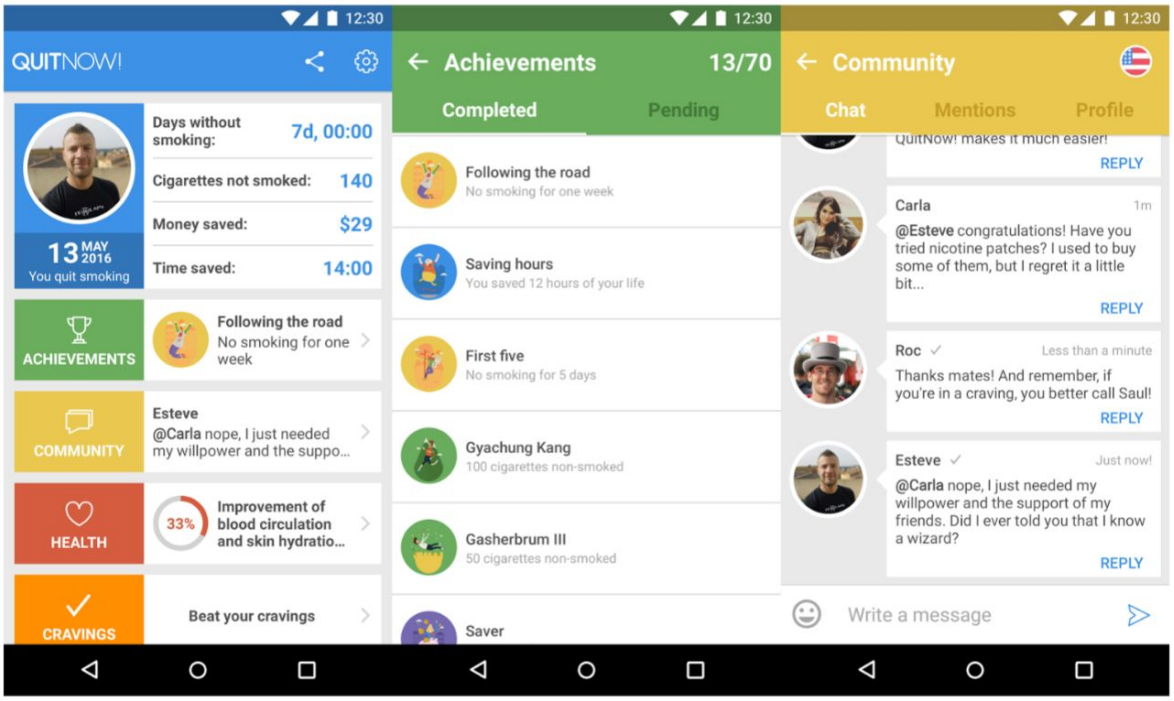


Figure 2.19	Screenshots of the application QuitNoW!

In Figure 2.19 we can see an application of the principle of Social learning used in QuitNow. Users learn how to perform the behavior from the advice of other members of the community.

2.3.1.3 Final state: the user is performing the targeted behavior

In this stage, individuals are performing the targeted behavior. They have practiced intermediate behaviors that brought them to develop automatisms and habits supporting the targeted behavior. The challenge of this state is to maintain the targeted behavior.

We have chosen the following three features as examples for this stage:

- **Rewards (Dialogue support):** the persuasive system reinforces the performing of the targeted behavior with a reward.
  Example: users reducing the consumption of water in one month get a silver virtual trophy in the system application. In order to get a gold one, they have to maintain the challenge for 3 months.
- **Social comparison (Social support):** the system should provide means for comparing performance with the performance of other users.

Example: comparing the personal water consumption with the one of the neighbors.

- **Recognition (Social support):** the persuasive system provides a social reward for users performing the behavior.
  Example: users consuming less water during one year in each neighborhood are awarded as the neighbors of the year by the major in the town's website.

In order to give a concrete example of one of these principles, we provide the reader with a screenshot of the application bActive application of Tim Harries et al. (Harries et al. 2013) designed to increase the walking activities of users available for smartphones.

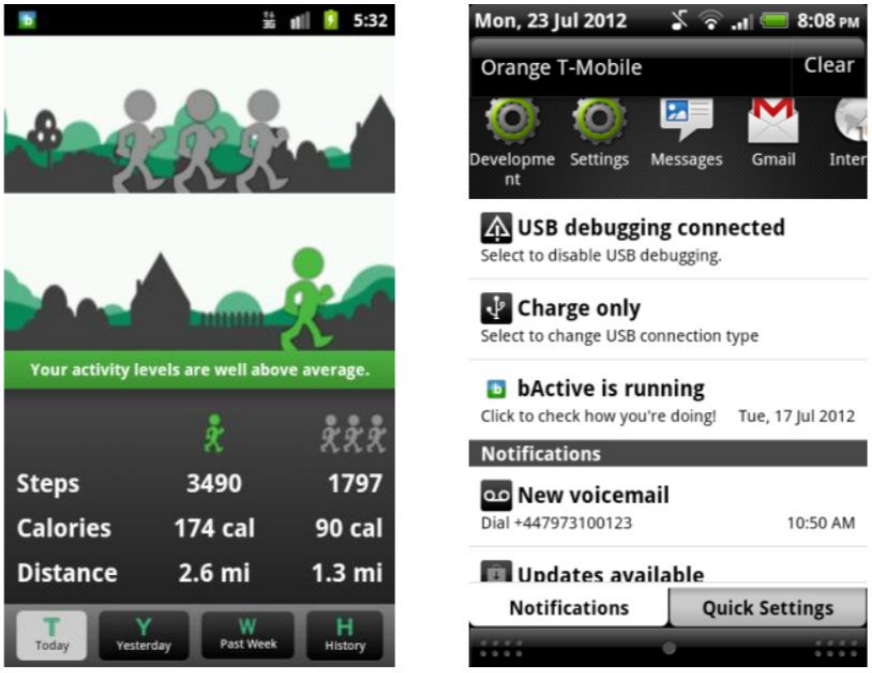


Figure 2.20	Screenshots of the bActive application of Tim Harries et al. (Harries et al. 2013)

In the screenshot in Figure 2.20, it is possible to see an application of the principle of **social comparison**: the performance of the steps done, of the travelled distance, and of the calories burned by the user is compared with the values obtained by other users with similar profiles.

As mentioned in the beginning of the section, the features detailed in the PSD framework are not linked to specific states of the process of change: they may work

in specific states or in all of them. For this reason we will now provide four examples of features that may be applicable during the whole process of change.

2.3.1.4 Generic states: example of features applicable to the whole process

The persuasive system needs to accompany the user during the whole process of change. Some persuasive features may be applied during the whole behavioral change. For example users need always to have feedback on their progress, they need to receive suggestions on how to perform the behavior and need to be encouraged to keep progressing between the states. The system should adapt to users, providing a personalized interaction based on their context.

We have chosen to describe three examples of features supporting the whole change:

- **Suggestion (Dialogue support):** the system suggests to perform a certain behavior.
  Example: the user receives a notification to take few steps for 10 minutes in order to reactivate the blood circulation.
- **Self-monitoring (Primary task support):** the system provides users with statistics and reports to inform them about their progress toward the change.
  Example: the persuasive application shows the number of steps done during the day and if the user has achieved a minimum number.
- **Praise (Dialogue support):** the system provides the user with a positive feedback via words, symbols or images to make them more open to persuasion.
  Example: the system provides the user with virtual badges every time that a targeted goal is achieved.
- **Tailoring (Primary task support):** the system should provide the user with personalized content tailored to the potential needs, interests, personality, and usage context.
  Example: an application to support physical activity may propose personalized workout based on the physical ability of users and on their past workouts.

In order to give a concrete example of one of these principles, we provide the reader with some examples on concrete interactive systems.

Figure 2.21 shows an example of **suggestion** on the platform Duolingo. Duolingo is a web and mobile application to learn languages. Users are required to indicate how much time they wish to dedicate to learn a new language per day and the application reminds them to perform the activity over time. If users stop the learning activity for a

given period of time, they get notified by the system to start again to perform the behavior. The message aims at persuading the user in doing the behavior and the yellow button facilitates the task bringing the user directly on a language learning session.

In the same figure we can also find an example of **tailoring**. The system proposes language challenges and activities that are related to his or her specific profile: the level of expertise in a certain argument, or the words that he or she has learned in previous lessons.

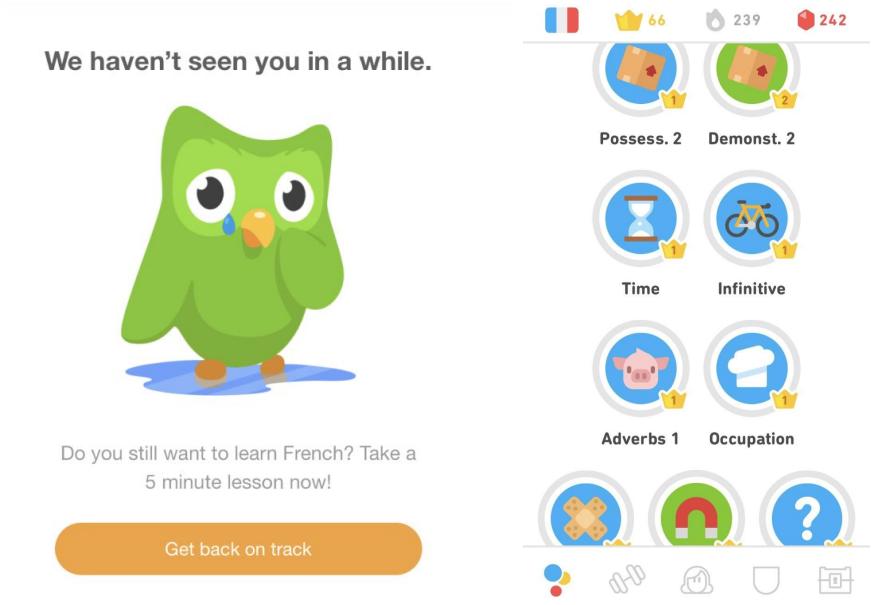


Figure 2.21	Screenshots of the Duolingo application

The work of Consolvo et al. "UbiFit Garden" (Consolvo et al. 2008) can be taken as an example to show the features of **praising** and of **self monitoring**. UbiFit garden was designed by authors to track the physical workout of users. The application alters the smartphone background inserting flowers and butterflies related to different activities performed by the user. In Figure 2.22, the part (a) shows the background at the beginning of the week: small butterflies indicate recent goals attainment and the absence of flowers means no activity during the current week. In the part (b), the background is changed, showing a garden with flowers. The different flowers represent the different activities performed, explained in the part (c) of the image. Large butterfly indicates that the current week goal was met by the user.

The **praising** feature is represented by the flowers and butterfly that positively reinforce users. The **self-monitoring** feature is represented by using the technique of metaphor, often used in persuasive technology (e.g. “Monnalisa bookshelf” or “Virtual aquarium” by Nakajima and Lehdonvirta (Nakajima and Lehdonvirta 2013)). With metaphors, instead of providing numerical data (maybe too difficult to be understood by some users), the system uses familiar objects (in this case the flowers and butterflies) to provide feedback to users on their performances, for example counting the number of the metaphorical objects (in this case the flowers and butterflies).

Figure 2.22	Images of the work of Consolvo et al. “UbiFit Garden” (Consolvo et al. 2008)

2.3.2 Discussion

In this section we have proposed an operationalization of a generic process of change with specific persuasive features. This operationalization aimed at analyzing how to approach the feature design according to the different states. This analysis permitted to recognize some discussion points in line with the general investigation of this work that we synthesize in the following.

First of all, taking the perspective of the process, we conclude that the persuasive features have a direct link with the transitions between the states. We have motivated how each transition is associated with a set of behavior determinants. These determinants are thus the factors that permit the designer to choose which feature can be driven at each state of the process to enact transition. The persuasive

features, thus, trigger the behavior determinants and allow the user to progress on the change.

A second and fundamental consideration is that the features can not exclude the context analysis. This information is crucial to decide which persuasive feature should be used and what is the adequate situation. To make an example it is not possible to use a reward-based strategy feature in a state where the behavior is not performed, exactly as it is not possible to tailor the interaction to users if not sufficient information is available on them.

From these considerations we conclude that modeling the process of change is not sufficient: the models/frameworks/structures required to operationalize persuasion need also to integrate a reasoner, that constantly updates and queries the context information to effectively operationalize the process, the features, and the interaction.

In the next chapter will present our proposition in order to fulfill these requirements and to propose a solution to design and operationalize persuasive technologies accounting the context.

3 Conceptual contributions

Focusing on the operationalization of behavior change processes, Chapter 2 uncovers that existing behavior change models represent either the complete process of change (i.e. TTM) but without providing elements for its operationalization, or focus at smaller scales (e.g., FBM, HAM) on transitions between two intermediate behaviors. In this thesis, by contrast, we propose a unifying concept, the notion of persuasive path, to embrace all these aspects.

Before formalizing the persuasive path, its properties and elements, we detail the two case studies, CRegrette and Mhikes introduced at Section 1.6, by modeling their processes of change. This will help in pinpointing the strength of persuasive paths for moving from user- to system-oriented perspective.

3.1 Illustrative case studies

We begin this chapter by modeling the two case studies presented in the introduction (Section 1.6) as state machines, aiming at performing a more accurate analysis claiming for persuasive paths.

3.1.1 Case study 1: CRregette, persuading to avoid smoking

Description

CRegrette investigates the effects of different types of persuasive triggers targeting smoking cessation. We have instantiated the three triggers described by Fogg (Brian J. Fogg 2009): we print stickers to be pasted on the cigarette packet or on the lighters to enact users' self monitoring (signal), we created a mobile application able to notify the persuasive strategies to the user (spark), and we built a wearable device able to detect the smoking activity, to simplify tracking of smoked cigarettes (facilitator). The results of the experiments brought to conclude that the combination between facilitator and signal were the most effective strategy to reduce the smoking behavior. A second showed that notifying the persuasive messages in different contexts had different outcomes. In particular notifying the persuasive message in some contexts (e.g. during work activity) reminded the negative behavior of smoking, while reminding to not smoke in other contexts (e.g. the morning) was taken as a challenge for users and more effective.

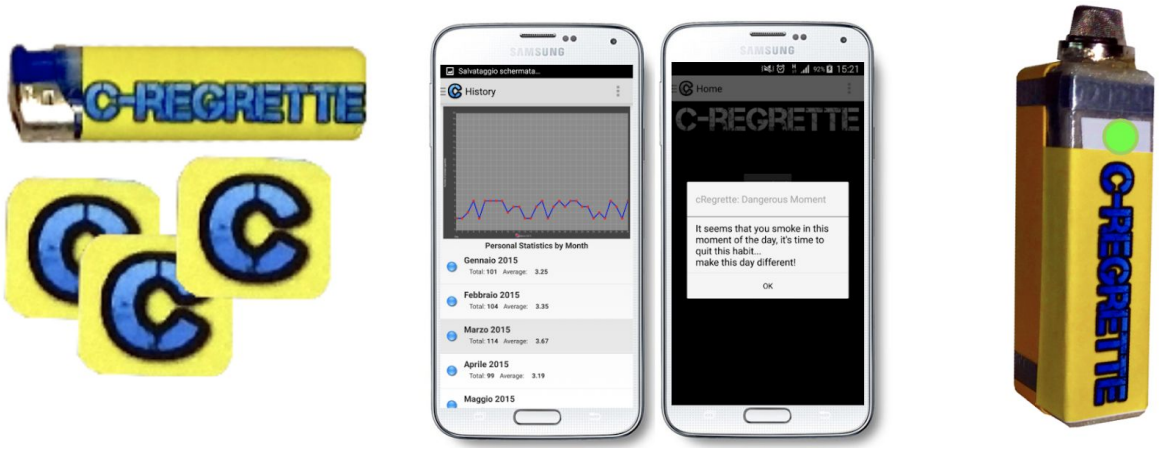


Figure 3.1	The CRegrette Persuasive Ambient Device. On the left the stickers, on the right the wearable device, in the center the mobile application connected to the device. (Fenicio and Calvary 2015)

State-machine-based model

Analysing the process of smoking cessation, and in particular of CRegrette, we noticed that initially individuals smoke obviously: they do not realize the number of smoked cigarettes nor the place or the time where they smoke. Sometimes they just smoke as reflex or to employ their time.

In Figure 3.2, the first transitions (associated with determinants **D1**) aimed at making users aware of the reasons, the places, the moments, the people connected to their smoking activity, in other words the unfavorable contexts. In order to gather these contexts we asked the participants to note down the context when they were smoking. For this reason, we reminded them to note down this information applying on the cigarette packets and on the lighters a set of stickers, that they could not avoid seeing while smoking.

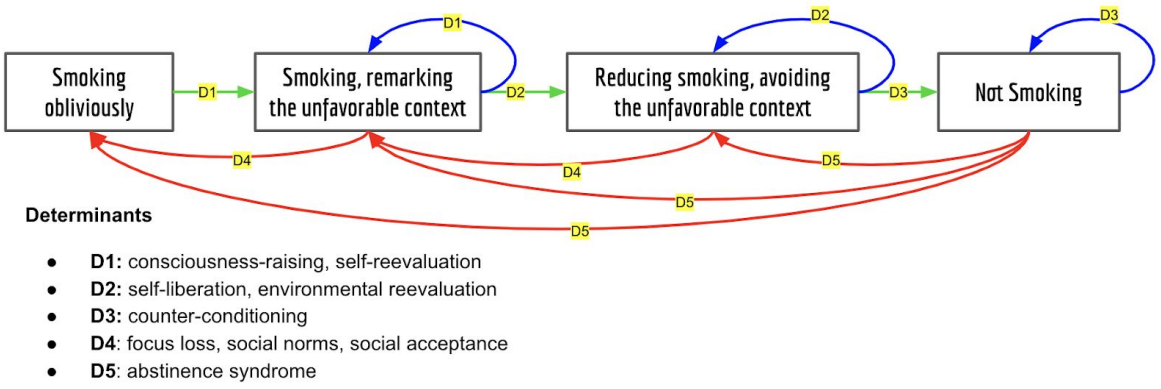


Figure 3.2	State-machine-based representation of CRegrette

The successive transitions (associated with determinants **D2**) aimed at persuading users in avoiding the unfavorable contexts that induced them to smoke. This was made possible by notifying the user to avoid certain places or to make effort during specific moments of the day where he or she had previously smoked. This information was gathered by the brooch, that was storing the places and the moments connected to the smoking activity.

The last transitions (associated with determinants **D3**) aimed at the maintenance.

3.1.2 Case study 2: Mhikes, persuading to increase outdoor activities

Description

Mhikes is a free web and mobile platform designed to accompany users in the exploration of natural and urban environments. The platform is available on the web version at the url www.mhikes.com or in the mobile application version for Android (play.google.com/store) and OSX (www.apple.com/itunes).

The application provides a catalogue of itineraries situated all over the world. The itineraries are associated with outdoor activities such as hiking, mountain biking, walking, snow racketing, and others. Each itinerary is composed of a GPS trace made of points of interest, whose information (directions / historical contents / curious facts) is shown when the user is in proximity of one of these points.

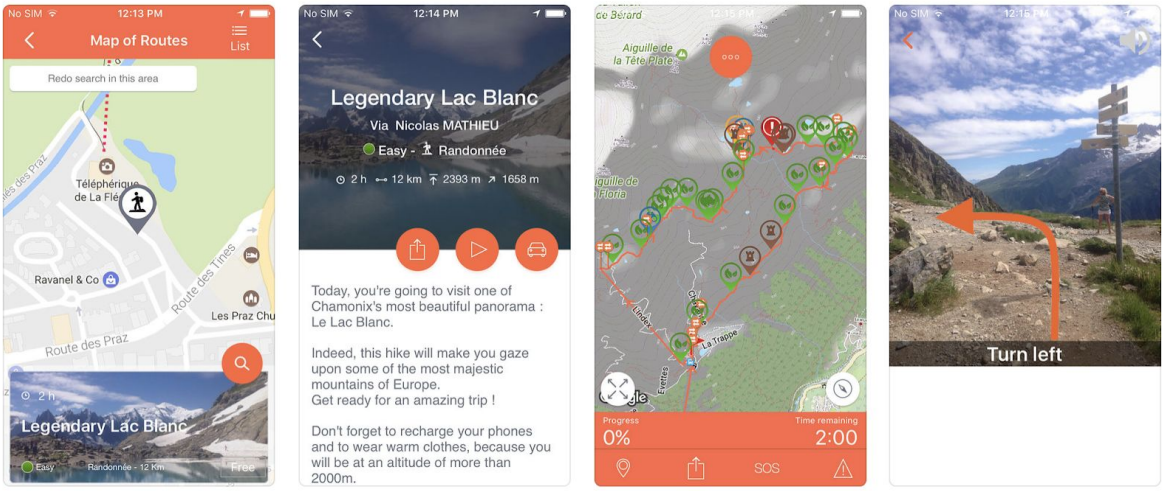


Figure 3.3	Screenshots of the Mhikes application for Iphone. From left to right: the world map showing the available itineraries, the detail of an itinerary where it is possible to perform its download, the itinerary on the dedicated map, an example of guidance of the system.

Users can consult the itinerary catalogue and download the itineraries on their smartphones to be guided during the outdoor activity. We name these users the **walker users** of Mhikes.

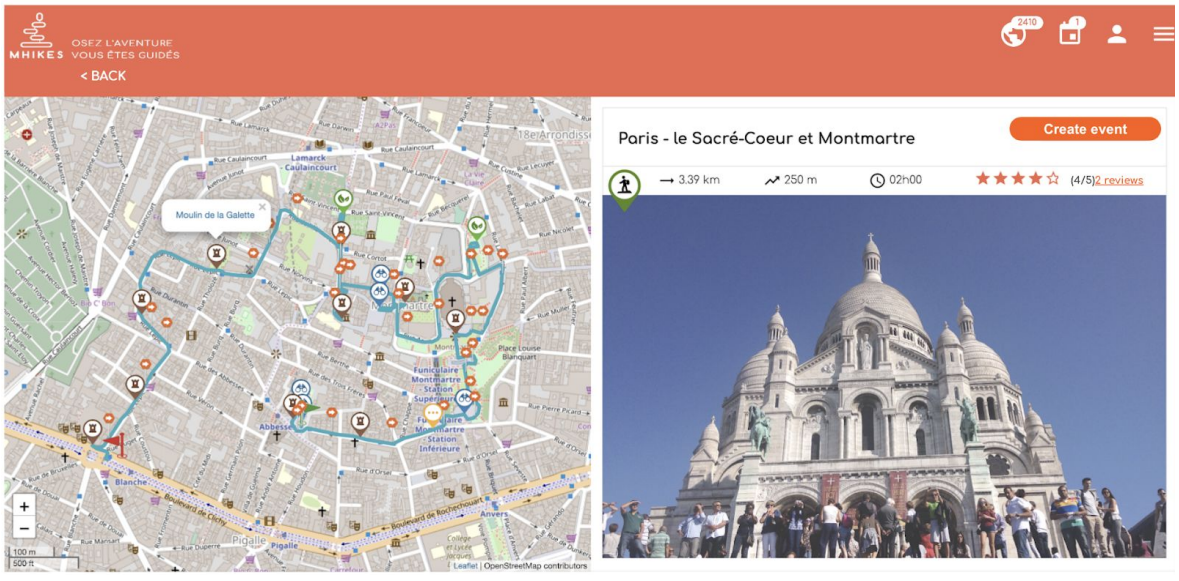


Figure 3.4	Details of a Mhikes itinerary in the web version. To be noticed the button “create event” on the top right.

Itineraries may also be created by users themselves by recording the GPS trace “on the field”, or by editing the trace on a dedicated web-online editor. We name these users the **tracer users** of Mhikes.

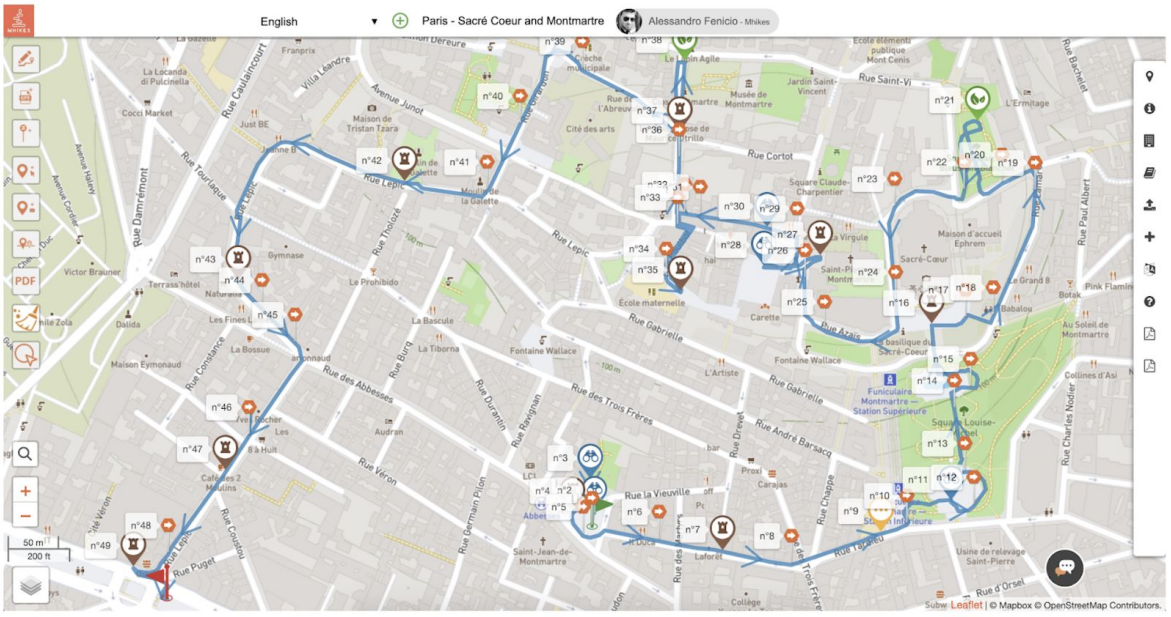


Figure 3.5	In the figure the Mhikes web Editor that permits to trace, edit, upload and publish an itinerary on the Mhikes catalogue.

Users are motivated by different reasons to download the application and to register to the platform. For example they may be interested in being guided on mountain hikes because of a lack of experience, or they may look for new itineraries to be explored during the weekend with their family or friends. Another scenario is represented by users eager to share their experiences and advice, tracing their own itineraries to be shared in the community of Mhikes.

State-machine-based model

The initial version of Mhikes did not include any persuasive intent. For this reason, initially any process of change was defined for the system. From a back-end analysis, the Mhikes team was able to tell that many itineraries (e.g. “Walking on the Mont Blanc”) were consulted by users and that the relative traces were also downloaded on their mobile phones. However these actions did not bring them to perform the itinerary: users showed an initial intention not concretized into a physical action. For this reason we used this application as a case study to implement

persuasive features, with the objective of transforming this intention of performing a behavior into its actual realization.

In the non persuasive version, three behaviors were observed by the Mhikes team: users searching and downloading the itineraries, user tracing the itineraries, and users performing the itinerary (behavior less performed). However there is a need to understand the determinants leading to perform the hikes and, in general, to maintain these behaviors.

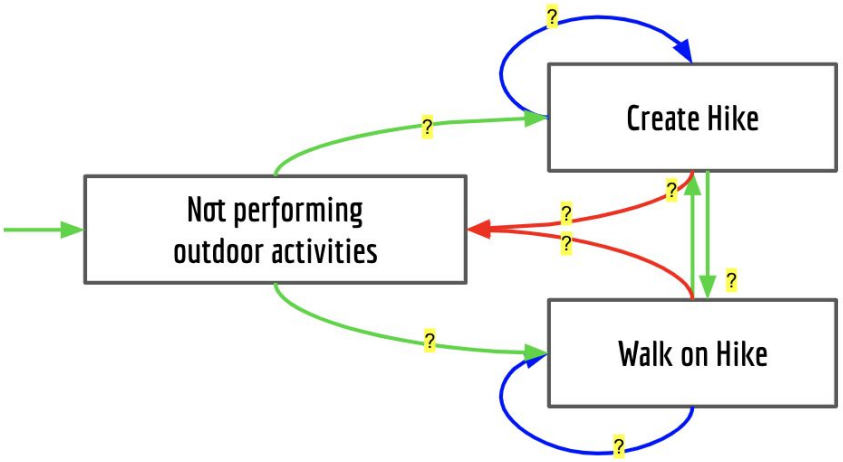


Figure 3.6	State-machine-based representation of the initial version of Mhikes

3.1.3 Discussion

In the previous paragraphs, we have described the process of change of CRegrette and Mhikes using state machines. We have used the same approach in Chapter 2, where assuming a user-centered perspective we have described how individuals change behavior over time according to the different models of literature.

Using this approach (**user-centered** with **state-machines** formalism) to describe the process of change permitted to have a clear vision of behavioral evolution. However, for engineering persuasive interactive systems, it is necessary to switch from a user-centered perspective to a **system-centered** perspective in order to accomplish the operationalization of persuasive features.

In this perspective, we propose the concept of persuasive path to bridge the gap between user-centered modeling and system-oriented implementation.

3.2 Concept of persuasive path

In this section, we discuss and articulate the concept of persuasive path. We will be presenting the concepts of **Persuasive Events**, of **Interactive Role** and we will detail how the two concepts are related to the notion of **Persuasive Path**.

After the definitions we will provide the reader with a conceptual example of persuasive path for the CRegrette and Mhikes case studies. Later we will discuss conceptual model and design.

3.2.1 Definitions

Persuasive paths are a succession of persuasive events associated with the different behaviors performed by an individual involved in a process of change.

3.2.1.1 Persuasive event

Dictionaries provide several definitions for the word **event**: an *occurrence*, *something that happens or takes place*. For example it is defined in the Oxford Dictionary as "**a single occurrence of a process**". This generic definition refers to any action that can be observed in any given **context**. A second definition from the Oxford Dictionary states that an event is ***"a planned public or social occasion"***. This second definition focuses on the social facet of the concept of event. The words *public* and *social* suggest that people may be involved in the event with different **roles**. For example individuals may just participate to an event while others may endorse the role of organization. From a persuasive perspective, events have exploitable characteristics to leverage persuasion.

The social dimension gives a set of persuasive strategies exploitable with events. For example using the Cialdini's principles (Cialdini and Garde 1987), events may convey influence because being invited or taking part in a social event is proven to have an impact on influencing people by using the principle of mechanism of **social proof** and **commitment** (Cialdini and Garde 1987). Participating in an event because a friend is participating (**liking** principle), accepting to go to an event to return a favour (**reciprocity** principle), or accepting to go to an event because advised by an expert (**authority** principle) are other examples of applied influence. The principle of Social **acceptance/rejection** states that "people are motivated to do things that bring them to be socially accepted" (Brian J. Fogg 2009). Similarly accepting an invitation for an event can figure as the opportunity to do something to

confirm a membership (e.g. going to a tribute day). Persuasion offers several techniques oriented to social interactions that can be used in persuasive events (e.g. **social support** (Oinas-Kukkonen and Harjumaa 2008)) or that come from the **reward**/**operant** conditioning theories (Skinner 1963) or from other kind of reward (e.g. tribe, hunt, and personal rewards (Eyal 2014)). Events also support one of the fundamental steps in habit forming, **having a plan/following a routine**. BJ Fogg explains that users try to create their own everyday routines to embed change in their lives (e.g. "tiny habits" (Brian J. Fogg 2009)).

The characteristics detailed in the previous paragraph thus theoretically demonstrate the persuasive potential of events. We thus define a **persuasive event** as an event associated with a behavior performed by an individual involved in the process of change. The event may be caused by a certain persuasive strategy or can be spontaneously performed by the user. A persuasive **event** is always associated with the **context** in which the user performs the behavior.

Event may also be related to other individuals which according to the behavior targeted by the event may endorse different **roles**. The concept of roles will be also used later in the work to convey specific persuasive features. In order to give a definition for roles associated with persuasive paths, we illustrate and detail this concept in the next paragraph.

3.2.1.2 Interactive role

The concept of role in an interactive system is usually associated with the type of task that users perform. This dualism role/interaction has been investigated under different perspectives according to the field of research. In the CSCW community, roles are often associated either to a business role in an organization (e.g., manager, employee) or to the access rights of users to features or resources of the system (both are often interrelated). In the "Role-Based Access Control" (Sandhu et al. 1996), roles usually assume a hierarchical shape (e.g. administrator, moderator, user, guest). By contrast, Jourde et al. in their work propose a different approach using "interactive roles" to describe how users' tasks are dynamically allocated depending on who is achieving the task using a certain modality (Jourde, Laurillau, and Nigay 2010). BJ Fogg, from the persuasive research field, overturns the perspective by analyzing the user interactions according to the final "purpose of the system" expressed through three different roles: the system as a social actor, as media and as tool (Brian J. Fogg 2009). Schneider et al. quantitatively analyzed the attitudes, beliefs, and values of mobile fitness coach users with the theory of planned

behavior, identifying distinct user groups to as target of the persuasive design (Schneider et al. 2016).

In order to discuss the concept of role in persuasion, in the article (Fenicio et al. 2016b) we have proposed an instantiation of the paradigm of **producer and consumer**. This paradigm in computer science is used to analyze the synchronization problem of the two actors (the producer and the consumer) competing for resources (Tanenbaum 2009). Bringing this paradigm to the persuasive perspective brought to define these roles as following:

- The **producers** are users that create new resources in the system.
- The **consumers** use the resources produced by the producers.

An **example** of the paradigm consumer producer can be found in Mhikes. The producers are the users that trace hikes; the consumers are the ones performing them.

We have extended this paradigm to capture another characteristic of interactive systems: the social facet. In this perspective, we can decompose the interactive roles using two axes as shown in Figure 3.7. A first axis (vertical) describes if users produce new resources in the system (**Producer**) or if they use the ones that are already available (**Consumer**). The second axis (horizontal) describes if users produce the resource to be used in group (**Co-Producer**) or if users will just consume group-resources (**Co-Consumer**). We can indeed define four main roles out from the initial two:

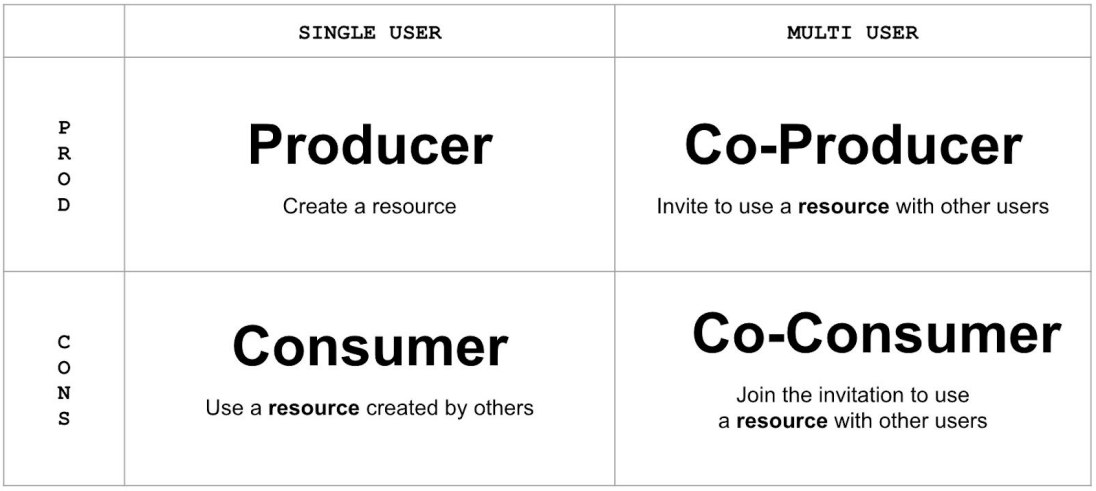


Figure 3.7	The four interactive roles and their two main axes: the vertical one related to production and consumption, and the horizontal one related to the interactive modalities: single user and multi user.

- **Producers** add new content as resources, information or experience tips.

  Profile example: they are experts in something and eager to share their experiences with others. However, their creation task does not involve other users in principle, they are already satisfied by the productive task.

- **Consumers** use the contents provided by other users.

  Profile example: they take advantage of technology and of the content it provides. They may also use the technology to increase their abilities.

- **Co-Producers** are users that create resources to be used with other users.

  Profile example: they are eager to share resources through social activities like events. Their production is no longer oriented to hedonism but assumes a real social and multi-user dimension.

- **Co-Consumers** are users interested in consuming resources with other users.

  Profile example: they may not be sufficiently motivated to consume contents autonomously, nor to engage the productive tasks. However, interacting with co-producers, they are motivated in consuming the resources offered by the system.

Users of an interactive system may change role repeatedly, by performing transitions from the behavior associated with one role to the one associated with a different role.

This may lead to perform a more specific characterization of the transitions at the operationalization time, defined as **extra-role** and **intra-role** transitions. In particular:

- **Intra-role** transitions demand less effort to user because the user permanes in the same type of task.

- **Extra-role** transitions demand more effort to users. They may need to learn how to endorse the new role.

In Chapter 4 we will go further on this aspect defining all the four roles for the Mhikes system and detailing how changing role may be used as persuasive feature to sustain the behavior change.

3.2.1.3 Persuasive path

A **persuasive path** is a chain of persuasive events describing the behaviors performed by individuals over time. In Chapter 2 we have used states, transitions, and determinants on state machines to structure the process of change. Persuasive paths go further, aiming at the operationalization of this process, by associating to

each state the current contextual information of users, devices and of the environment, the employed persuasive features, the element to evaluate the persuasion. These components permit to the persuasive architecture to orchestrate the succession of events and to build a personalized plan toward the change.

3.2.2 Illustrated case studies

In the following, we illustrate persuasive paths on the case studies of CRgerette and Mhikes.

3.2.2.1 CRegrette

We take as starting point Figure 3.2, which describes the behavior change of CRegrette by using the state-machine formalism.

The **first state** of the automaton is related to the smoking behavior: we want to avoid it.

In order to make users aware of the context in which they smoke (**second state**), we can schedule persuasive events based on their smoking activity. Thanks to software probes inserted in the digital brooch of CRegrette, the persuasive architecture could capture the smoking activity and notify users the unfavorable context (e.g. pop-up notification saying: “It’s 11 am and you are smoking in this location. This is the third time this week that this happens, did you realize it?”).

The unfavorable context is an important information since using this context the architecture can schedule persuasive events for the **third state:** “avoiding the smoking behavior”.

Knowing the unfavorable context, the persuasive architecture can schedule a set of events in targeting behaviors not related to smoke. For example it may propose to the user to have a fruit at that time/location. A pop message in the interface may say “It is proven that having one apple per day improves your health, lowers diabet, reduces heart attacks chances and provides several vitamins”, with a confirmation or cancel button to indicate if the user has accepted or not the suggestion. If the strategy was successful the persuasive architecture stores the information on the context associating it to a favorable context for this behavior.

Differently from the initial version of CRegrette, the persuasive message does not remind the smoking activity since they are related to another topic (e.g. eating an apple).

Finally to deal with the **fourth state** of the behavior change, related to the maintenance of correct behaviors, the architecture can schedule persuasive events

to enact a self evaluation of users. At the end of the day for example notification showing the progress of the day can be used as reinforcement for the effort of user.

3.2.2.2 Mhikes

We take as starting point Figure 3.6, which describes the behavior change representation of the first version of Mhikes by using the state-machine formalism.

The **first state** is represented by the behavior that users initially perform: consulting itineraries and the resources of Mhikes without actively performing the outdoors activities. The persuasive architecture in this state collects relevant information on the user context (in this case favorable contexts): for example, the moments of the day in which they consult the itineraries, the used devices, and specific information related to the itineraries (e.g. where the hike is located, which kind of activity it involves, the associate difficulty).

Knowing this information, the architecture can propose persuasive events to persuade users to perform activities similar to the one they have browsed in the previous state.

Events based on the **second and third states** (creating hike, walking on hike) thus are events related to perform the physical activities such as performing a hike. Associated persuasive features to persuade users in performing this behavior can be notifications. The delivery time for these notifications can be inferred by the persuasive architecture on the basis of the favorable context. For example the same moment of the day in which the user autonomously consulted itineraries could be a favorable moment for the persuasive system to provide the same type of information.

We conclude remarking that in figure Figure 3.6 two transitions connect the states of hike creation and hike performing. In the following of this work we are going to propose specific determinants for these transitions based on the mechanism of role switching.

3.2.3 Discussion

We highlight two peculiarities of these persuasive paths, further developed in Chapter 4.

First of all, we have taken the perspective of a user that “consumes” the contents of the system: the user of the example performs existing hikes. A different path would be created by the persuasive architecture if the user was “producing” content in the

system, for example creating new hikes as initial behavior. In this case the event would have aimed to reinforce the hiking creation or to induce the user in becoming a consumer, for example trying some hikes similar to the ones that he or she usually traces. This will have brought to a **role-switching** from producer to consumer.

A second aspect is that the architecture could schedule events that involve more users. For example a tracer user could organize an event inviting other users to test his or her hike. In this case we will have an implementation of the concept of **social event**. Therefore two new roles could be defined: the role of organizer and participant.

In Chapter 4 we will detail in depth the concepts of role-switching and of social event as persuasive features to persuade Mhikes users.

Before passing to the implementation, in the next sections we focus on how to model and conceive persuasive paths.

3.3 Model of persuasive path

In order to show the relationship between the concepts presented so far in the definition of persuasive path, we have realized a UML diagram. The diagram separates and models three levels: the “target level” where targeted behaviors are defined (blue colored), the “physical level” where behaviors happen and are observed (green colored), and the “persuasive level”, linking the precedent two, where persuasive paths operate (yellow colored).

We provide a more detailed presentation of each level:

The **physical level** (green colored in the UML) is where the reality is observed. This level includes two main components: the **context** and the **behavior**.

> The context is defined and decomposed into three dimensions: the user, the environment and the device. This choice is in line with previous formulations in the state of the art (e.g. the PSD framework, and other approaches cited at Section 2.2). The **user** refers to all the possible information characterizing users such as their profile, their interests, their past actions and their behavioral status. The **environment** refers to the information related to the physical and social conditions where the interaction takes place. For example, the characteristic of a certain place (urban district, meteorology, temperature, type of terrain). The **device** refers to the technical surroundings supporting computation and interaction: the type of device, the platform, the software version, but also the status of the device sensors such as accelerometers and

microphones, and of the user interface (e.g. the checkbox is checked, a certain view is displayed).

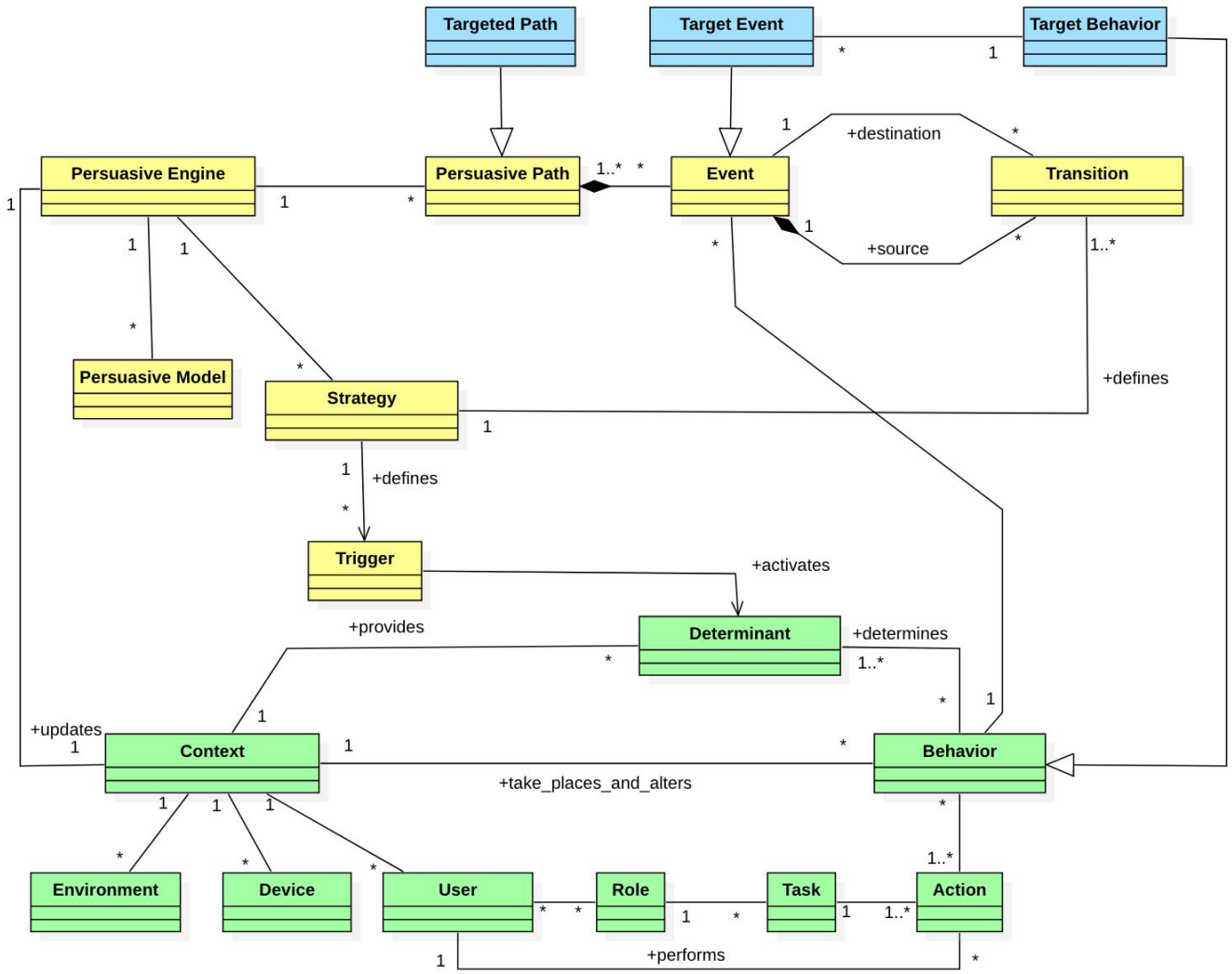


Figure 3.8	A UML diagram representing the three modeled levels: the “target level” where targeted behaviors are defined (blue colored), the “physical level” where behaviors happen and are observed (green colored), and the “persuasive level”, linking the precedent two, where persuasive paths operate (yellow colored).

The **determinants** that bring the user to perform a certain behavior are related to the context. This justifies their direct association in the UML diagram.

The **behavior** is indeed defined as one or more user actions taking place in defined contexts.

Actions are related to specific **tasks**, associated with user **roles**.
For example in Mhikes the task of “download an itinerary” is associated with the action of “clicking the download button” and with the associated consumer role.

The **persuasive level** (yellow colored in the UML) maps the physical level into persuasive paths.

For the moment, the **persuasive engine** must be seen as a “black-box” component in which the reasoning algorithms take places (we will provide concrete examples of their functioning in Section 3.4).

The **persuasive engine** has a constantly updated vision of the context, of all the users, devices and environmental information available in the system. Persuasive models (and related state machines) can be associated by persuasive experts, according to the domain of application and to the type of behavior targeted. The persuasive engine manages a **persuasive path** linking the persuasive events associated with the user behaviors. The mechanism described by **persuasive events** and transitions is the one modelled by state machines: an event can be the source of a transition if another event figures as destination. The transitions are associated with a persuasive feature built on triggers to be driven to users, satisfying or not their **behavior determinants**.

The **targeted level** (blue colored in the UML) models the targeted behaviors interested in the process of change.

The difference between the **targeted path** and the persuasive one is that the targeted one is made only by the behaviors targeted by the user. Users indeed may not always perform the targeted behaviors. For example in CRegrette even if users are targeting the non-smoking behavior, they may relapse and smoke. This case event should be accounted in the persuasive path, since it may provide contextual information to build further strategies (e.g. a given trigger did not work). The smoking behavior is thus included in the persuasive path, but not in the targeted one. On the other hand, the action of eating a fruit (instead of smoking) will figure in the blue targeted path since it is associated with a targeted behavior.

In the previous paragraphs we have completed the definition of the relationships between the elements of persuasive paths. We have explained how persuasive paths orchestrate the persuasive events to sustain the process of change. In the next section we perform a focus on the persuasive architecture components associating a set of implementation guidelines.

3.4 Persuasive architecture

In the following paragraphs we present the components of the persuasive architecture to engineer the persuasive path in an interactive system. In the UML graph presented at Section 3.3 we have given a model of the persuasive path. In the graph, the persuasive architecture is implicit, hidden behind the component "**persuasive engine**" which figured as a "black-box". In this paragraph we "open the box" and explicit its components.

The architecture follows a Model View Control pattern (MVC). In the MVC pattern the users interact with the user interface of the system, the **view** generating the inputs for the controller. The **controller** contains the logic of the system. It is a set of algorithms dedicated to retrieve the data stored in the model and to aggregate them to produce new data or to build the system output to be sent to the user through the view. To compute the answer, the controller queries the **model** which contains all the data related to the entities involved in the system. These data are usually stored in a physical memory such as a database.

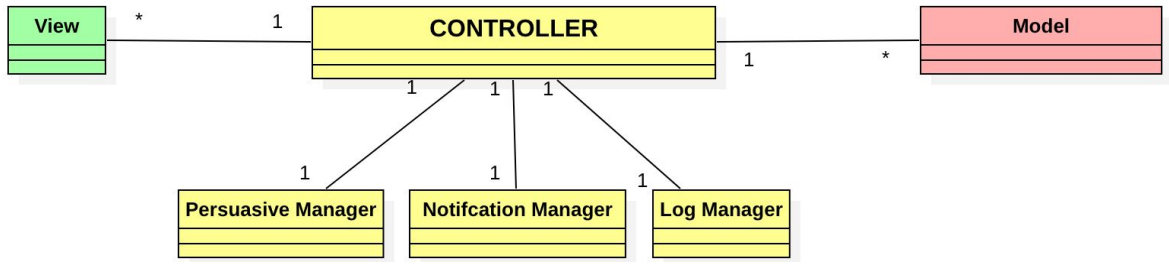


Figure 3.9	The components of the persuasive architecture used in Mhikes.

We provide the reader with a more detailed explanation of the MVC components for persuasion:

- **View** (rendering of the system)**:** this component represents the rendering through which the users interact with the system. The interactive components of the view (e.g. buttons, links…) embed a set of software probes detecting the user interaction. When an interactive event is detected, the Log Manager stores the information in the database.
- The **Controller** is divided into three parts:
    - **Log Manager:** this component is dedicated to track the interactive events that users performed in the view, physically storing the contextual information associated with the user interaction.
    - **Persuasive Manager:** this component uses the input generated by the user, the information stored in the database, and the associated persuasive models to build the persuasive strategy (a more complete explanation is provided at Section 3.4.3 when the component "persuasive manager" of the UML graph is detailed). Once the persuasive strategy is built, the persuasive manager stores this information in the database waiting to be treated by the notification manager.
    - **Notification Manager:** this component delivers the persuasive strategies scheduled by the persuasive manager. It retrieves strategies from the database and builds a dedicated view to be shown to the users. Once the trigger has been delivered, the Log Manager will be in charge of detecting via its software probes if the user reacted to the trigger with the expected interaction on the interface.
- **Model:** the model is a representation of the organized collection of data stored in the database. The database contains all the information about the entities of the system (e.g. the user profile, the application contents, the context information, etc.). The database may be managed by using dedicated database manager or administration tools.

In order to orient the implementation, we provide a set of design general guidelines related to the main components of the persuasive architecture.

3.4.1 View

The View of the architecture is the rendering of the system. The interface includes a set of interactive components (e.g. buttons, icons, tangible components, links...) permitting the user interaction. The components, on which the user interacts (e.g.

clicking, scrolling, dragging, pinching, etc..), must be associated with software probes. In order to fulfill this requirement, two steps need to be performed:

- **Associating each user with an identifier**
  The architecture will be able to build the persuasive path only if the interaction is reconducible to defined user profile. For this reason the system needs to implement an account system providing an authentication means for (e.g. username/password, log-in with google, fingerprint). Once this step has been completed it has to be possible to retrieve the user profile form an associated unique identifier (e.g. email, userId, nickname…).

- **Add a listener to the components**
  In order to retrieve the information on the interaction, each component needs to implement a listener. In software engineering, this can be achieved by using ActionListeners[3] for example. A pseudo-code implementation example in Java can be the following:

```
//INSTANTIATING THE BUTTON
JButton button = new JButton();
//LISTEN TO THE CLICK
button.addActionListener(new ActionListener()
{
 //PERFORMING THE RELATED ACTION
 public void actionPerformed(ActionEvent e)
  {
 //Code of the action to be done
  }
});
```

Figure 3.10	The definition of a JButton component and its associated action-listener.

Every time the button is clicked, the listener has to store all the available information related to the interaction. Two types of information can be modelled:

- **Domain-independent information**, such as the unique user identifier, the unique component identifier, the action associated, the device type, the current time (e.g. the device system clock), the location (it can be the GPS position if the device has a GPS receiver, an approximate

[3] https://docs.oracle.com/javase/7/docs/api/java/awt/event/ActionListener.html

location derived by the IP of the device if it is connected to the internet);

- **Domain-dependent information**, which covers other general data gatherable by the specific system: in CRegrette for example the air quality; in Mhikes the difficulty of a hike, its duration, etc...

3.4.2 Log manager

The Log Manager is the component responsible for storing the information related to the interaction in the persuasive architecture. The listeners of the View components trigger the logging operation passing to the Log-System the information on the users that interact with a given component (along with the associated role). The Log Manager is composed by a physical storage memory (e.g. database, file) associated with a controller responsible for inserting reading, updating or deleting the information. Taking MySQL as an example, the database needs to include a table composed as follows: in the rows there are all the interactive events triggered by the front-end components listeners, in the column there is all the contextual information passed by the triggers, where the interaction_id is a unique identifier of the events. An example is provided below:

domain-independent information						domain-dependent information	
interaction_id	**user_id**	**component_id**	**device**	**timestamp**	**location**	**altitude**	**blood_pres**
2	23	click _link_ 3	web	1/3/19 2:10:16	paris	35m	110-70
3	24	button_ok_4	mobile	1/3/19 2:10:17	berlin	34m	120-80

Figure 3.11	Example of how the Log Manager stores contextual information on the model of the persuasive architecture.

For the log manager we do not have specific guidelines to propose, developers can choose any implementation that is compatible with the system as far as it is capable of storing the domain-independent and dependent information.

3.4.3 Persuasive manager

The persuasive manager is the software component responsible for aggregating all the data available in the Log Manager to produce new knowledge about the users,

used to build the personalized persuasive paths. The persuasive manager is composed by a set of data analysis algorithms. These algorithms depend on the specific data provided and associated by each interactive system and for this reason a generic implementation code cannot be proposed. However a set of guidelines may help developers to implement this software component and the related algorithms.

- **Aggregating the information**
  The Log Manager will provide the information relative to the user's interaction with the system. This information needs to be aggregated to derive new knowledge. For example knowing the users most performed actions on the interface components, their most used devices, and the times of the day where the interaction is maximum, may be used to create a specific strategy for these users: for example, suggesting to users similar content on their device at that moment of the day or putting them in communication with users with similar profiles.
- **Having an updated user context**

  The amount of data collected by the log manager will increment along the time. For this reason it is necessary to implement an algorithm able to update (or recalculate) a more recent, user-context each time that is needed.
- **Building and scheduling a personalized notification**
  Once the user context is available, the persuasive manager has to compute the set of possible transitions from the actual user behavior, to create a personalized trigger to be delivered to the user. The trigger can be built for example on a pop-up message, by using an email, an in-built mobile notification (e.g. android notification) or by using other interface components capable to deliver the persuasive message. The trigger needs to be personalized for each user, meaning that they must include for example their names, their recent activities, their favourite activities to fulfill the tailoring principle (see Paragraph 2.2.1.4).
- **Scheduling the triggers**

  Once the persuasive trigger and message are built the persuasive manager chooses a favourable moment to send the message. If the favorable moment cannot be computed (or on the contrary is known a priori), it can be manually specified. Finally the triggers and their scheduling are stored in a physical memory (e.g. database) to allow the Notification Manager to manage and send them.

3.4.4 Notification manager

The notification manager reads from the database the information on the triggers and on the scheduled time computed by the persuasive manager. The notification manager cares about the delivering of the triggers and is in charge of detecting if they are correctly sent, received and perceived by users. In order to be able to evaluate the triggers the notification manager needs to implement the following requisites:

- **Pooling the scheduled triggers**

  The triggers may not need to be delivered in the moment in which they are built. For this reason the system needs to query periodically the associated database. For example, the system may be programmed to check each minute if there are any available triggers to be sent. Another option is to use context sensing to choose the moment in which the manager has to check the database. For example checking the database every time that the user comes back home. This information can be retrieved by the GPS position of the device or by the fact that the device is connected to the home wifi network.

- **Tracking and evaluating the triggers**

  The notification manager needs to be able to determine the status of the trigger. A trigger can be scheduled (programmed to be sent), sent (driven to the user), delivered (received by the user), perceived by the user (e.g., read), executed (the user has performed the action associated with the trigger) or expired (not valid anymore). In order to be able to that, the persuasive manager needs to be associated with specific listeners that update of the trigger. Let's take an example. A persuasive trigger can be made of a persuasive message combined by a call-to-action button sent by email. The notification manager needs to implement a listener that ensures that the email has been delivered (e.g. querying the mail server), opened (e.g. Delivery Receipt), and if the call-to-action has been clicked or not (e.g. Button listener).

  Once trigger is executed, the Logging System will be able to track if the trigger has caused the performing of the related action in the system (e.g. interacting with an interface component associated with the triggered behavior). The persuasive triggers that were performed (or not performed) by the users may then be considered by the persuasive manager to complete the users contextual information in order to drive always more effective persuasive

strategies. Finally a policy on when the trigger is considered not anymore valid has to been chosen at design time.

3.5 Design of persuasive path

In the previous sections, we have proposed a set of concepts for operationalizing persuasion. In this section, we provide practitioners with a method to apply these concepts.
The method presents five steps, inspired by the Most4P model (Fenicio et al. 2016b):

1. (Reality-dependent) Observing the users **behaviors**,
2. (Problem-dependent) Defining the **mission** to be achieved,
3. (Domain-dependent) Identifying domain dependent **objectives** to achieve the mission,
4. (Task-dependent) Expliciting the objectives in **strategies** on the system and user tasks,
5. (Context-dependent) Associating concrete **tactics** to strategies based on the context.

In the following, we explicit these five steps illustrating them on the concrete case studies of CRegrette and Mhikes, which can be visualized in Figure 3.12.

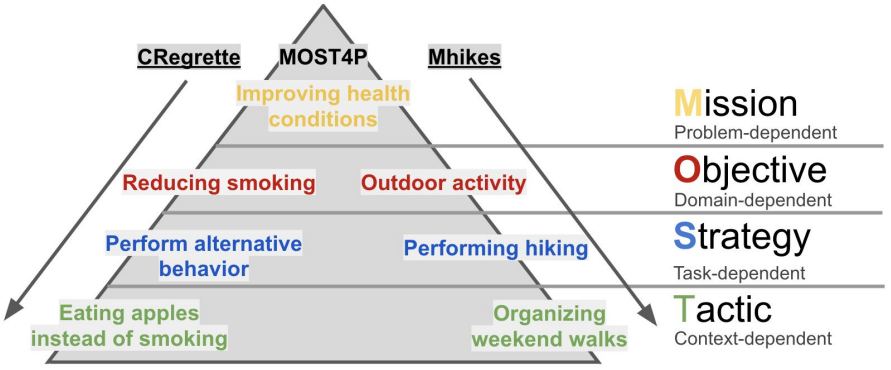


Figure 3.12	Example of structuration of CRegrette and Mhikes on the Most4P model.

Going through the steps, we will be highlighting which components of the persuasive architecture are interested by each step.

Step1: Observing the users **behaviors**
This step concerns all the possible observable behaviors of users in the physical world. These behaviors include both the targeted behaviors (e.g. quitting/reducing smoking in

CRegrette) and the non targeted behaviors (e.g. smoking in CRegrette). The reason why the non targeted behaviors have to be considered is that they are observed, thus, part of the reality and impacting on the desired change.

> → At the end of Step1, it is possible to get a state machine with the observed behaviors (e.g. Figure 3.2 for CRegrette). The behaviors are the **states** and the **transitions** are links between them. In case of sufficient information from the observation also the **determinants** may be defined. We can associate the observation of the user behavior to the **physical level** of the persuasive architecture (Section 3.3). This means that at this stage of the method practitioners need to identify what are the observable data associated with these behaviors.

Step2: Defining the **mission** to be achieved
This step aims at identifying the problem to tackle, the ultimate mission of the persuasion. The mission can be addressed in different domains, for example in the case of CRegrette and Mhikes, a common mission for the persuasive system could be of “improving the general health condition”.

Step3: Identifying domain dependent **objectives** to achieve the mission
The general mission of step2 is refined into domain-dependent objectives. For example in order to “improve the general health condition” CRegrette sets as objective of “reducing smoking”, while Mhikes targets the “performing of outdoors activities”.

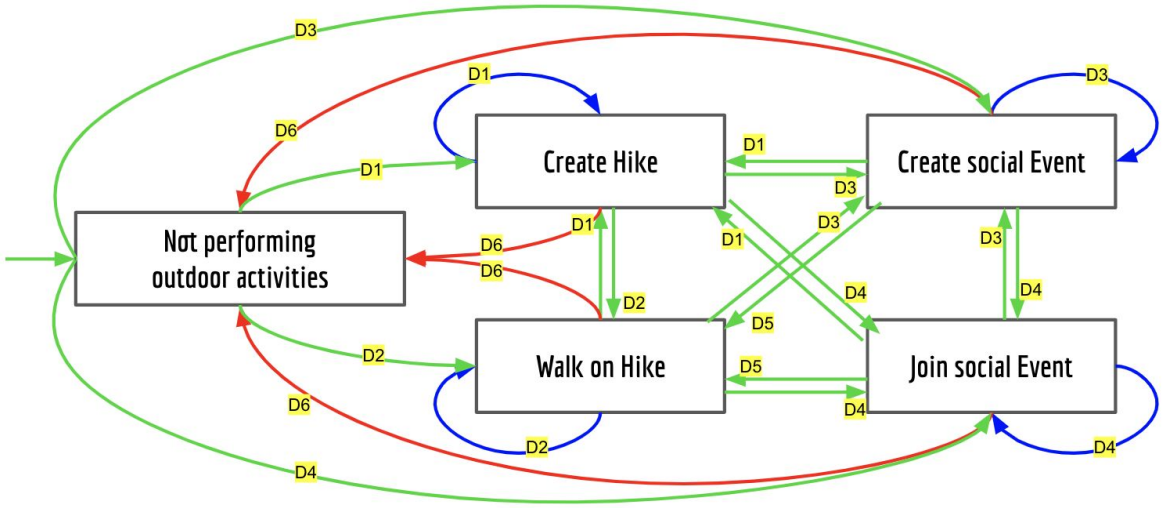
> → At the end of Step3 it is possible to define the **target level** of the persuasive architecture (Section 3.3). The mission is the general objective of the **targeted path** (Section 3.3) while the objectives are the **targeted events** (Section 3.3) since they refer to the targeted behaviors of the application domain.

Step4: Expliciting the objectives in **strategies** on the system and user tasks
The objectives identified at step3 are general solutions to achieve the targeted mission. However the system may have different tasks to approach the realization of these objectives using strategies that are related to their personal preferences. In CRegrette a person may decide to “reducing smoking” by “finding an alternative behavior” as strategy, another strategy may have been to reduce each day the number of cigarettes by one. In Mhikes users may approach the targeted objective of “performing outdoors activities” by performing hiking or by creating new itineraries.

> → The end of Step4 may bring practitioners to revisiting the initial state machine formulated at Step1, since with Step4 they may discover new

alternatives in the system and user tasks to actuate the objectives. This happened during the application of the method to Mhikes.

In the initial Mhikes state machine (Figure 3.6), we have introduced two new states as strategies, giving users the possibility of creating social events to motivate them in performing collectively the outdoor activities. Figure 3.13 shows the updated state machine, with the two new states “Create social event” and “Join social event”.



Transitions determinants (A different examples for each transition)

- **D1:** pleasure of creating a own hike
- **D2:** wish of improving physical well-being
- **D3:** pleasure of organizing a social activity
- **D4**: desire of hiking without organizing
- **D5**: desire of hiking alone
- **D6**: unsuccess, frustration, interest loss, focus loss

Figure 3.13	The process of change in Persuasive Mhikes

The introduction of these new tasks, also impacted the set of available interactive roles, introducing, beside the initial ones of **Tracer** (Producer, creators of itineraries) and **Walker** (Consumer, performers of itineraries), two new roles related to the social dimension: the roles of **Leaders** (Co-Producers, creators of events) and of **Followers** (Co-Consumer, participants of events).

Step5: Associating concrete **tactics** to strategies based on the context
The strategies become concrete tactics once the context in which they are performed is defined. In CRegrette a tactic for the previous strategy “an alternative behavior” in context can be “eating a fruit at 11 am”; in Mhikes a tactic for the strategy “performing outdoors activities” in context can be “performing a hike next weekend".

> → At the end of Step5 it is possible to define the **persuasive level** of the architecture, and indeed to apply the **persuasive paths.** The association between behaviors and context operated by the tactics permits to design specific persuasive features to be operationalized by the persuasive paths (e.g. building persuasive notification). In addition to the observables defined at **Step1,** dedicated sensors (software probes) and actuators (e.g. the listeners presented in Section 3.4.1) can be added to capture and evaluate the performing of behaviors and to update the context of the persuasive architecture.

Once the five steps are achieved, the practitioners get a complete definition of the elements present in the persuasive architecture and thereby are able to operationalize the persuasive path associated with the initial behavior change process through specific persuasive features.

This section focused on providing practitioners with a method to operationalize persuasive paths. We wish to complete this method with a further discussion on specific points that deserve attention and that are transversal to the five steps listed above.

3.6 Discussion

Before concluding the chapter, given that the reader now has a complete vision on the subject, we explicit some aspects of persuasive paths that we consider deserving an additional discussion.

Enacting collaboration in social dimension

Progressing on a behavior change is not easy and often people search the support of other individuals to be helped in their cause. Users may be not sufficiently motivated or capable of progressing autonomously and this may generate frustration for not being able to progress in the behavior change. The role of Co-Consumer captures this users’ need. Persuasive path thus may enact the interactive system social dimension.

From the cooperation between Co-Producer and Co-Consumer an entire community mechanism can be developed and designers may consider to implement these roles aiming at the community growth. This is an additional reason that motivated the implementation of social events in the Mhikes platform, as demonstrated in the next chapter.

Providing the context for the behavior change

In the introduction we have mentioned the importance of delivering in-context strategies to the persuadee. Our preliminary investigation CRegrette confirmed this as a key point of persuasion. A side-effect of the experimentation indeed showed that sending a notification to an individual suggesting to avoid smoking is counterproductive when the person is in the office while it is effective in the morning before going to work. Persuasive paths offer the possibility to account context along the behavior change process by using three contextual dimensions (user, device, and environment).

However beside the capability of providing the context, persuasive paths allow to build new knowledge about the context. In particular they allow the definition of the favorable context in which persuasion should be driven to users. This is made possible by the aggregation of the information operated by the algorithm of the persuasive manager, and by the feedback on the delivery of persuasive triggers that the notification manager and the log manager are able to evaluate as successful or unsuccessful.

We may thus conclude that persuasive paths, provided with a sufficient amount of reliable data, have the theoretical capability to infer what to trigger to users and when. This capability, without doubt, constitutes a remarkable persuasive potential which will be later tested on the conducted evaluation.

Providing elements to operate the role switching

The role switching may be triggered by the persuasive architecture in different ways. For example, Co-consumers may be persuaded to become Co-producers, with the aim of being socially rewarded. This example is based on a reward strategy, but other possibilities are offered by the persuasive architecture. We propose two factors that persuasive paths can use to operate the role switching:

- The **experience factor**. Users may positively respond to a role switch suggestion according to the experience they have with the application and with the associated behavior. For example a non expert hiker in Mhikes may use the application just as consumers, to have tips on where to hike. After having performed different hikes their experience increases. Persuasive paths

can use this factor to persuade users in becoming producers, “doing more”, for example tracing their own hikes.

- The **similarity factor**. Users may positively respond to a role switch if similar users have already operated the switch. The notion of similarity is related to context: similar users have common contextual characteristics. Example of similarity in context may be related to the user profile (e.g. age, physical characteristics, work, interest, cultural beliefs) to their personal skills (e.g. background, ability with technology) to the environment (e.g. same location of use). Persuasive paths account these factors to produce persuasive features targeting these similarities (e.g. proposing the user to join some friend activities, or mentioning that similar users have previously succeeded in performing the targeted behavior).

Targetable behaviors using persuasive paths

We have discussed that the targetable behaviors are not all the same. The two proposed case studies, CRegrette and Mhikes, explicit this difference: the first aims at avoiding a behavior (smoking); the second aims at increasing a behavior (performing outdoor activities).

Persuasive paths allow the instantiation of all the five flavors of behavior identified by Fogg (Fogg and Hreha 2010). This is made possible by the persuasive manager which is able to schedule different persuasive events in the persuasive path.

We give an example of how the persuasive manager operates the scheduling of each of the types of behavior identified by Fogg:

- Performing a **familiar** behavior (blue flavour): the persuasive manager schedules a persuasive event that the user has precedently performed. For example to persuade a consumer user to perform a consuming task (e.g. download an item) the persuasive manager can schedule one event associated with that task. A way to achieve this is to use intra-role transitions.

- Performing a **new** behavior (green flavour): the persuasive architecture has access to all the possible tasks offered by the persuasive interactive system. For this reason the architecture can propose to the users events associated with a behavior that they are not familiar with. For example, if users have performed only behaviors related to consuming tasks (e.g. read a post on a web-blog) they may not be familiar with writing post. The persuasive architecture can schedule events associated with this task, achieving the result. A way to achieve this is to use extra-role transitions.

- **Increasing** a behavior (purple flavour): the persuasive architecture has access to all the contextual information associated with users and to their precedent activities. In order to increase a behavior the persuasive manager can analyse the frequency on which each behavior occurs and schedule a set of persuasive events in the path in order to increase that frequency. For example, to persuade a consumer user to increase the performing of a consuming task (e.g. download an item) the persuasive manager can schedule more events associated with that task in the persuasive path in order to increase the occurrence frequence.
- **Stopping** and **decreasing** a behavior (black and gray flavour): as we have explored on CRegrette notifying users aiming at non-performing or avoiding a behavior may recall the negative behavior. In order to stop a behavior thus the persuasive architecture can schedule events related to alternative behaviors, such as “having a fruit” at the same time in which the smoking activity took precedently place. If the users keep performing alternative behaviors they consequently stop the unwanted behavior and eventually they may develop new habits based on events suggested by the persuasive architecture.

Persuasive paths offer thus the theoretical means to structure the system and user missions in the process of change, and to operationalize the persuasion.

In the next section we translate the theoretical and modelling approach to a concrete implementation operated on the Mhikes system. The implementation is oriented not only to demonstrate and validate the concepts related to the persuasive paths, but to definitively introduce persuasive features into the system that is used by the Mhikes community.

4 Implementation

In the previous chapter we have defined a set of new concepts and finally we have provided a methodological approach to operationalize them using the five step method described in Section 3.5.

In this chapter we go further giving concrete details on the implementation and of the operationalization of persuasive paths in Mhikes. First, we introduce a set of preliminary studies on the ergonomy and on the understanding of the concept of social events by final users. Later, we move to the actual implementation of the interface and of the components of the persuasive architecture.

4.1 Preliminary study on non persuasive Mhikes

The objective of this pilot was to correct any possible ergonomic issues with the interface of the initial system before introducing new features. This is a good practice to avoid introducing bias when aiming at assessing the effectiveness (or ineffectiveness) of new features. Indeed even if the introduced features work correctly the result may be anyway negative because of ergonomic flaws.

This pilot study was conducted on real participants: a group of students of informatics engineering recruited at the University Grenoble Alpes. The students were instructed on the functionality of Mhikes application and then on how to evaluate the ergonomics in an interactive system. In particular we have presented during the univeritary lessons the "Ergonomic Criteria for the Evaluation of Human-Computer Interfaces" formulated by C. Bastien and D. Scapin (Bastien and Scapin 1993), and on the basis of these criteria we have asked the participants to evaluate and report on the initial ergonomic status of the Mhikes application.

The pilot was conducted on two sessions: the first concerned a critical analysis of the graphical interface of the application and of the relatives functions. A second session permitted to extend the critical analysis on a practical scenario: participants were asked to download and walk on an itinerary in the city center of Grenoble with the objective of collecting feedback on the bug and on the ergonomy of the application and of any possible issue on the real time execution.

The table below reports the functionality issues encountered along with other details such as the device used by the user, its current operating system, and the context of the bug expressed through pre-condition (context of interaction), reported bug (what did not work) and post-condition (consequences of the bug).

Manufacturer	Model	OS Version	Pre Condition	Reported BUG	Post Condition
Huawei	P9 Plus	Android 6.0	Beginning of route	Missing quiz pop-up.	Impossible to use quiz
Huawei	P9 Plus	Android 6.0	Arriving to a POI	The description show up, but it closes monogamously after some seconds	Impossible to read POI description
Samsung	Galaxy S6	Android 6.0	Going out of route	“Away from path” message doesn't show up	
Huawei	P9 Plus	Android 6.0	Going out of route	“Away from path” message doesn't show up	
Motorola	X Style	Android 6.0	Answering to the quiz	“about:blank” as title of the message for the score rank	
OnePlus	One	Android 5.0	Answering to the quiz	“about:blank” as title of the message for the score rank	
Motorola	X Style	Android 6.0	Answering to the quiz	“about:blank” as title of the message for the score rank	
Samsung	Galaxy S6	Android 6.0.1 Android 5.0.2	Arriving to a quiz point	“votre nom” instead of names of participants to the quiz	Impossible to use quiz
Sony	Xperia	N.A.	Arriving to a quiz point	“votre nom” instead of names of participants to the quiz	Impossible to use quiz
OnePlus	One	Android 5.0	Arriving to a quiz point	“votre nom” instead of names of participants to the quiz	Impossible to use quiz
Samsung	Galaxy S6	Android 6.0	Not logged, download map, redirect to login	Login with google	Crash
Samsung	Galaxy S6	Android 6.0	Minimizing the app	On re-opening, app is lagging	Finally it crashes
Samsung	Galaxy S5	Android 6.0.1	Minimizing the app on the “Your GPS seems to be disabled, do you want to enable it?” message	On re-opening from minimization infinite messages “check you GPS settings” are created	Crash
Sony	Xperia	N.A.	Arriving at the End of route	“End of Hike” windows shows wrong statistics (or at least not understandable) 0.00/39	

Figure 4.1 Report on the bugs encountered in the initial version of Mhikes

For handing the ergonomic report, participants were asked to indicate the issue, the violated category of the Bastien and Scapin’s criteria and to eventually indicate a proposed solution. The result is shown in the table below.

ERGONOMIC ISSUE	CAT	PROPOSITIONS
Text size too small	1.4	- Increase the text size - Use text style (ex. bold)
Not possible to disable audio immediately	3.2	
Menu icon not conform respect to the icon in the home screen (≡) vs (...)	6 7	Uniform to the menu icon in the home (≡)
Menu icon not conform respect to the icon position in the home screen	6	Uniform to the menu icon position in the home
Clicking on any of the menu choices (except for GUIDANCE) user looses all the progress on the actual task (ex. Quiz points)	5.1	- Ask confirmation before leaving the route - Save the information to be restored later (in that case "pause" button (||) should be uniformed)
Not pointing at the actual POI	3.2 5.1	Differentiate the previous/actual/next POI (ex. bold)
Not content in the POI → Empty/Transparent	3.2 5.1	Put default images for a POI
Check mark instead used as confirmation button to search	5.1 6 7	-Follow IOS/Android design guidelines -Use "Search" or magnifying glass
Selection of difficulty using - +, but no numeric values	7	-Use seek bar
0 Ratings and four 4 = Impossible	6 7	- 0 ratings → no stars shown
Clicking on explore → results on map first Clicking on Search nearby → results on list first Clicking on the slideshow → results on list and not possible to choose map	6	Unify the default results presentation
Upper part of the screen there is a not signalled slideshow and in the remaining 2/third two image buttons	1.2 6 7	The slideshows should use dots . . . . to inform on which slide user is looking (As in the mhikes website). It also permits to distinguish between slideshow and image buttons.
Clicking on the magnifying glass, on search and on explore, user arrives in views where the menu (≡) disappears (see 31). Instead using entry "search" "explore" or "MyHikes" from the menu of the screen 28 users arrives in views where the menu is present (see 30).	3.1 6	View 30 and 31 should be the same (with or without the menu). Same for "Myhikes", explore, or search nearby
Menu present the "home" button which is used only in the case of view n°31, concerning the previous issue	2.1.1	Not useful or wrong to be compared with previous issue

Figure 4.2	Example of ergonomic issues encountered in the initial version of Mhikes

Most of the remarks were targeting the **consistency** criterion. Consistency refers to "the way interface design choices (codes, naming, formats, procedures, etc.) are maintained in similar contexts, and are different when applied to different contexts" (Bastien and Scapin 1993). One of the violations occurring in Mhikes was that different procedures were used to access the menu options. Other frequent violations were related to the **significance of codes** criterion: "the relationship between a term and/or a sign and its reference. Codes and names are significant to the users when there is a strong semantic relationship between such codes and the items or actions they refer to" (Bastien and Scapin 1993). An example of the violations was associating the symbol of check mark (usually used to represent a completed task or a correct status of a variable) to the function of searching. This also violates the consistency criterion since the search function was also associated in other screens to a magnifying glass icon.

The complete report was finally delivered to the development team of Mhikes which fixed the encountered issues. Ultimately, the Mhikes team adopted these ergonomic criteria as a reference for their future developments.

4.2 Preliminary study toward persuasive Mhikes

During Step4 of the design method proposed in Section 3.5, we updated the state machine of Mhikes to provide users with the possibility of creating social events on the Mhikes itineraries. This also resulted in updating the existing roles of the application.

In order to ensure that the implementation of an event system for outdoors activities could make sense for final users, we performed a preliminary study to ensure the understanding of users about this functionality. This is the report of the study:

Evaluation objective: in this evaluation we evaluated if the concept of social event was accepted by potential users of the Mhikes system and if they were spontaneously referring about the hypothesized interactive roles.

Evaluation methods: we interviewed N=25 participants with mixed backgrounds aged between 21 and 78 years during a public festival that took place in Grenoble. The “Transfo” festival promotes Digital Society. It is a one-week series of events, during which anyone can propose discussions and activities related to the technology topic.

On this sample we have used the following **protocol**:

1. Presenting the general objective of the study and collecting the personal data.
2. Semi-structured interview on social events.
3. Introducing the Mhikes event system: the users were informed on the Mhikes platform to organize social events for outdoor activities.
4. Semi-structured interview on event organization in Mhikes.
5. Analysing the answers.

Semi-structured interview:

- The **first set** of questions aimed at understanding the participants’ knowledge on generic types of events:
    - What is your experience in organizing/attending events? (ice-breaking)
    - Which expectations do you have from an organizer?
    - Which communication channels do you use to get informed/organize events?
- The **second set** of questions aimed at understanding the participants’ capability of applying their knowledge of events on the Mhikes system:

- What could motivate you to organize an event in Mhikes?
- What motivates you to participate in an event Mhikes?
- Would you lead a hiking event with users that you don't know?
- Would you join a hiking event with users that you don't know?

We report the synthesis of the verbatim analysis.

The first set of questions confirmed that all the users had a clear personal definition of events. They mentioned to have participated and organized few of them (e.g. birthday parties, vacations, meeting with friends). The participants reported in particular that they expect organizers to provide a **vision** of the event before the event (e.g. a detailed plan, timing, material organization), secondly they have to care about the **security** (e.g. guarantee, safety instructions, knowing the participant experience) and third they need to target the **entertainment** of participants during the event (e.g. caring about adaptation of activities along the events, flexibility, managing communication between participants).

During the second set of questions, participants mentioned that organizing an outdoor social event in the system could be motivated by **sharing a moment** with friends on the platform, by meeting new user of the community, by **sharing knowledge** on a particular place/activity, or by the implicit **pleasure of creating** something for the community. Participants described their motivation in attending an event on the platform to **overcome their organizative issues** (e.g. low experience in activities, no transportation means) and to meet people with the same **experience level**.

Participants made explicit references to the words **organizers**, **participants** as **roles** for events, nevertheless they often pointed out the fact that performing outdoor activities may not be an activity group, but "a moment of reconciliation with nature", "an occasion to disconnect from other people". This confirmed the fact that the four roles had a concrete representation in the user mind and that events activities may be performed in group (leader/follower) or alone eventually by using technology as a support (tracer/walker)

Conclusion: this experiment conducted through interviews permitted to strengthen our vision of the social events and of the roles, confirming that a possible implementation on the Mhikes system would have made sense for the final users.

As post-experiment contribution of this pilot, we resume the complete vision of the interactive roles in Mhikes in the following points:

- **Tracer** (Producer)

In Persuasive Mhikes producers use the application to trace itineraries. They physically walk on the hike or trace it by using the Mhikes web editor.
They are expert in tracing and hiking and eager to share their experiences with others. Their may be a personal achievement or be destined to other users.

- **Walker** (Consumer)
  Mhikes consumers use the application to safely discover and being guided on the outdoor activities, augmenting the experience of multimedia contents. Pop-up messages indeed show photos of crossroads with arrows and voice messages to choose the appropriate path. Once in proximity of a Point Of Interest (POI), dedicated contents are shown to learn about plants, flowers, and other environmental aspects.

- **Leader** (Co-Producer)
  Mhikes leaders are the users that organize social events such as weekend-walks and/or nature-related group activities to be done collectively. They use the technology to achieve their primary objective of discovering/enjoying exploring nature and urban environments, and secondly they take advantage of the activity to gather other people such as friends, or other users with common interests or similar profiles. They are eager to share through social activities. Jointly, leaders may also share useful resources such as transportation means and/or specific equipment dedicated to explorations (e.g. technical clothes).

- **Follower** (Co-Consumer)
  In Mhikes, followers are users that participate in social events available on the platform. The social events facilitate connections with leaders and to carry on activities. Followers are indeed users interested in hiking, walking but not sufficiently motivated to create content or to organize a social event. On the other hand, invited by a leader or a friend, they are happy to participate in such kind of activities. They are usually enthusiastic about sharing social activities and thankful to the organizers.

4.3 Persuasive Mhikes

In this section we give a complete description on how persuasive paths have been implemented and of how they operate in Mhikes.

4.3.1 View

After applying the method of Section 3.5 on Mhikes, we have designed the new user interfaces targeting the new tasks involved in the persuasive version.

We have implemented a second catalogue page dedicated to the events browsing and creation. On this catalogue page user can search for an itinerary and then pass to the event creation. The hike catalogue and the event catalogue were successively linked in order to facilitate users to switch between the different tasks and roles.

The main feature linking events and hike is that events are organized on a specific hike of the catalogue. Once the itinerary has been chosen, the user can create a social event implementing the event properties.

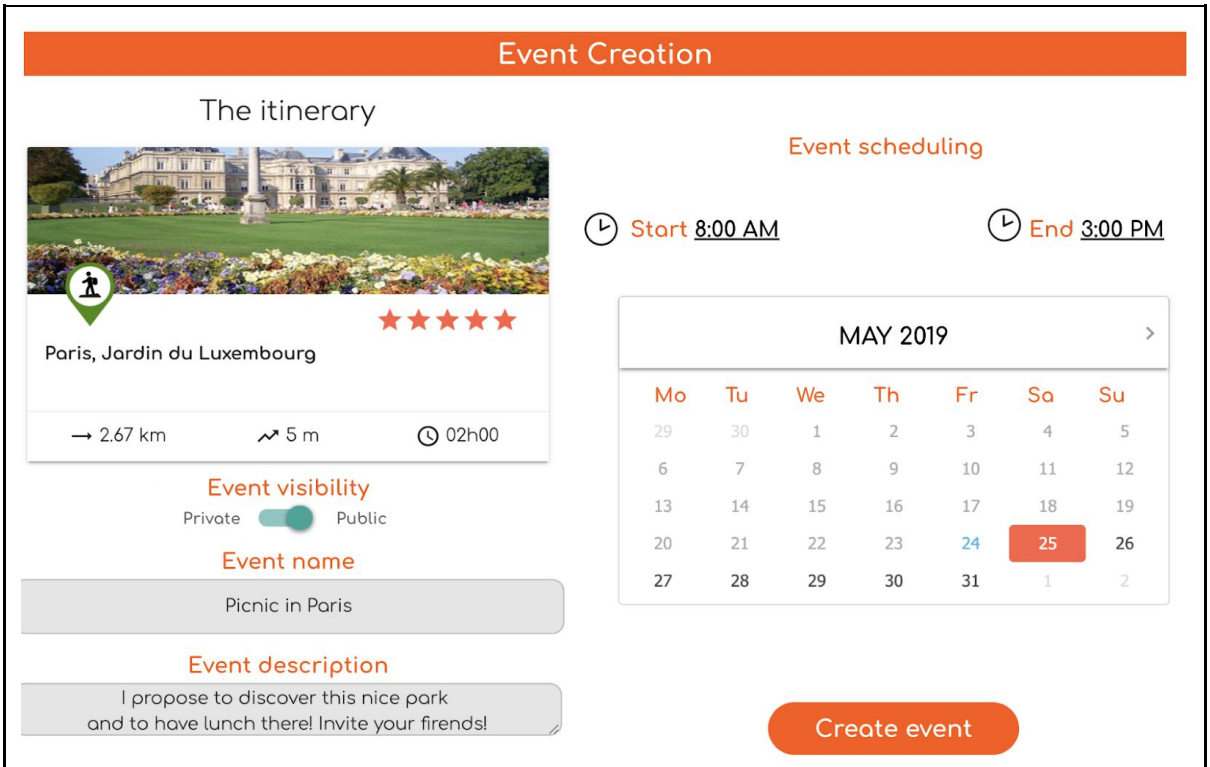


Figure 4.3	Social event creation in Mhikes.

Once the event creation is completed the event appears in the event catalogue and it will be visible to the other users of Mhikes (unless the event is not marked as private).

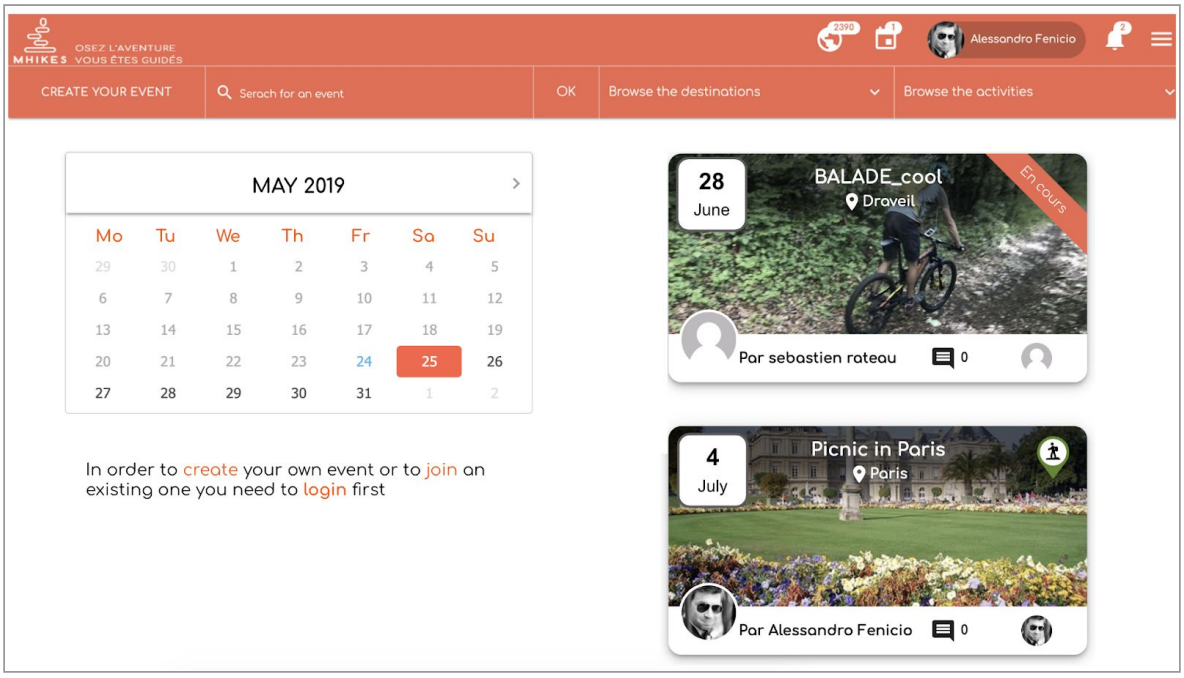


Figure 4.4	Event list in Mhikes. On the top the research bar and the button to create an event, on the left a calendar to consult the events per day and on the right the event list starting from the selected day.

In Figure 4.4 we show the event catalogue page, were two events are present “Picnic in Paris” and “BALADE_cool”. Clicking on events, users can consult theirs details (Figure 4.5), or create new events clicking on the top left corner button (Figure 4.3).

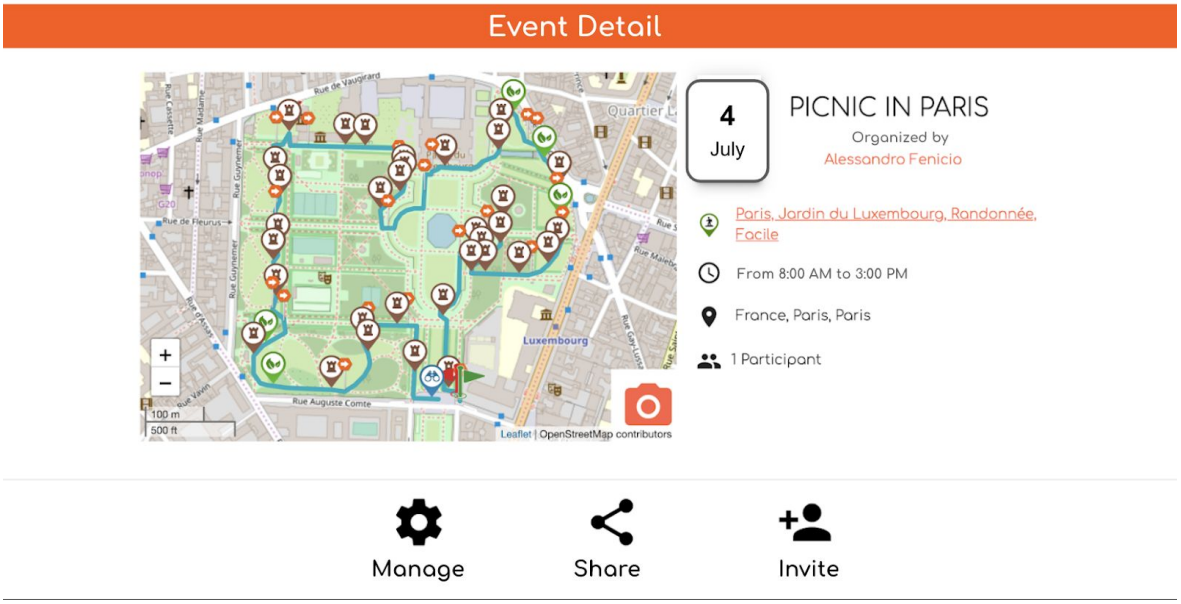


Figure 4.5	Detail of a social event created in Mhikes.

The design approach based on reviewing different online platforms that use social events to provide their services, aiming at gathering the key points to design the Mhikes event system. From this analysis we have evidenced five key points characterizing the social event design: the **location**, **scheduling**, **privacy**, **personalization** and **role**.

In the following we provide a short description of these characteristics illustrating how we relate them to the interfaces of persuasive Mhikes.

- **Location:** it represents the geographical context where the event happens. Two different characterizations are possible: a unique location, or a set of locations associated such as for example a starting point and an ending point. In Persuasive Mhikes the location is represented by the GPS trace of the itinerary. The social event starts and ends where the first and the last point of the trace are placed.
- **Scheduling:** it represents the temporal context in which the event happens. The event systems available online, typically indicate the beginning/end time or the duration of the event. Similarly to the location characterization, an event may be unique in time or may evolve over time. In Persuasive Mhikes it refers to the time in which the Mhikes social event happens. The starting time can

be chosen by the organizer while the end time is suggested in function to the length and difficulty of the itinerary.

- **Privacy:** it represents the rights that the event organizer gives to the event participants and to users that consult the social event system. For example, an event can be private if the organizer is the only one able to invite other users. Another possibility is that the event is open only to people having a secret password. Privacy deals also with the contents associated with the event which can be visible/editable only by participants to the events. In Persuasive Mhikes events are visible to all the users. If the organizer sets it as private only the invited participants can visualize it.
- **Personalization:** it specifies the details of the event such as the event name, the description. It can include a set of multimedia contents associated with the event, to provide more information or just for decorative purposes. It may also specify the maximum/minimum number of participants or the price of the event if necessary.
  In Persuasive Mhikes the events are free, and the organizer can set a description for the social event indicating details on the meeting point, on the needed material and more in general on the organization.
- **Role**: it specifies the role of users involved in the event. The main role is the organizer which organizes and manages the event. Other roles can be defined according to the type of event, co-organizers, participants, invited, special guest.
  In the event system of persuasive Mhikes two main roles are implemented, the organizer and the participant which refer respectively to the role of Co-Producer and Co-Consumer.

In the following screenshot the reader can appreciate which components of the interface, where interested by the properties gathered from the analysis of existing online event systems.

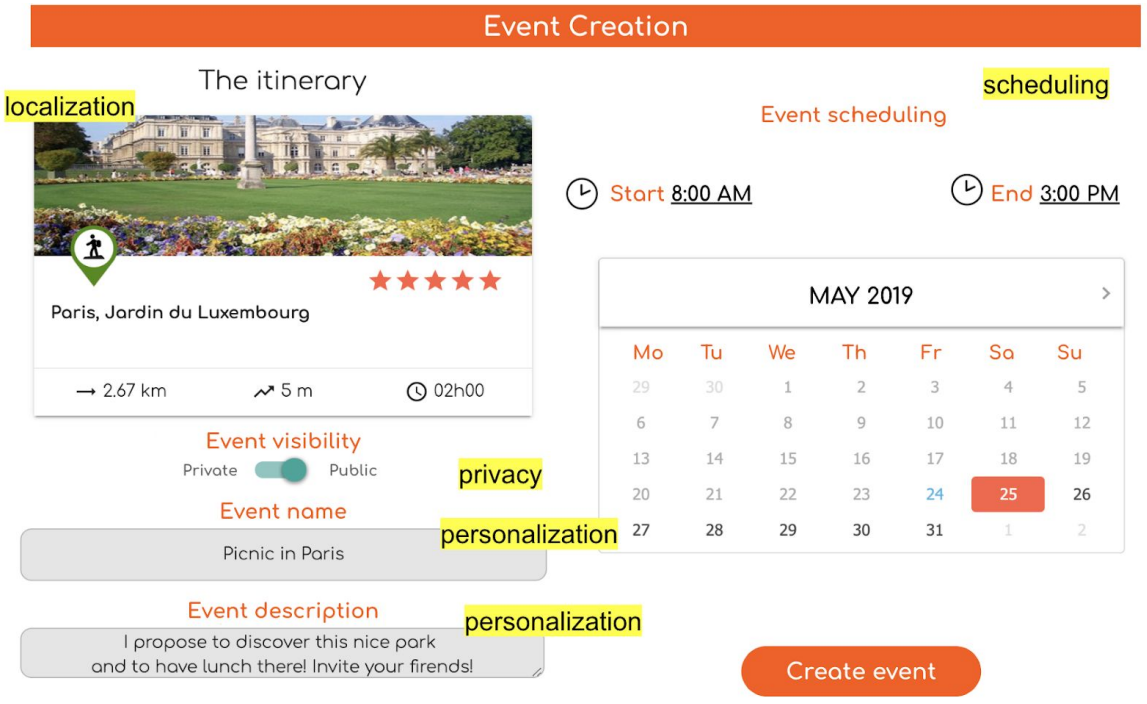


Figure 4.6	The properties of social events illustrated on the event creation of Mhikes.

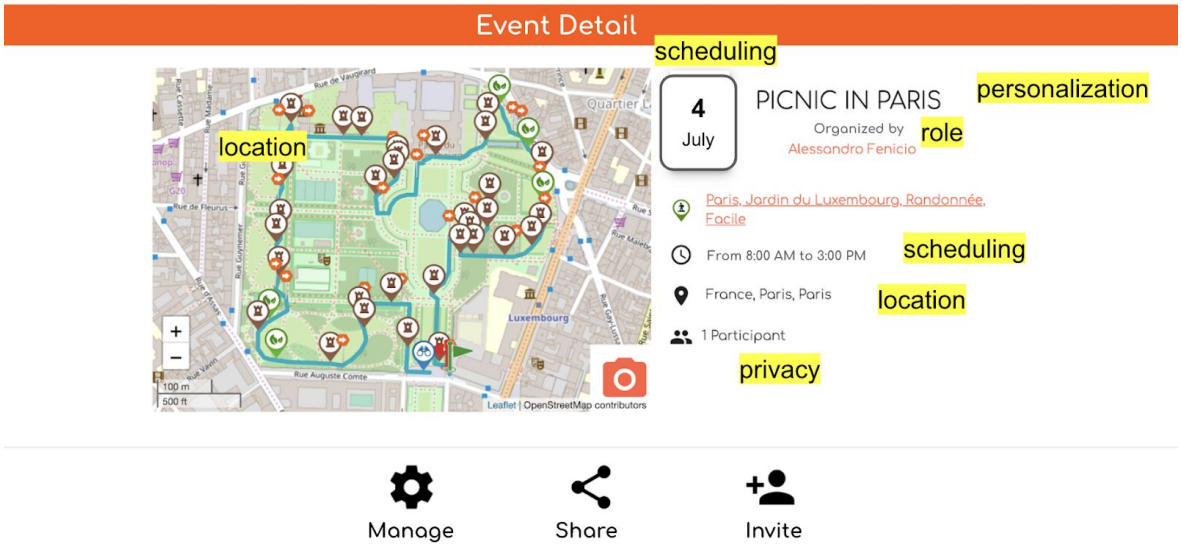


Figure 4.7	The properties of social events illustrated on a Mhikes event.

Once the design of the user interfaces of the persuasive version was completed, we passed to the implementation of the software probes, used by the Log Manager.

4.3.2 Log manager

The logging manager permits to track the interaction of the Mhikes users by using software probes. This is made possible by placing a dedicated logging functions that create an entry on a database table. For example, every time the user validates the creation of a social event (button “create event” Figure 4.6) the logging function for the event “create_event” is called.

```
…
$eventId = 23;
log_function(“create_event”, $eventId, $content)
…
```

Figure 4.8	Example of software probe associated with the create event button.

The second parameter is an identifier for the event ($eventId), the third ($content) is an optional parameter to complete the information to be stored in the database. Once the function is called the following algorithm collects the additional contextual information.

```
function log_function($descriptor, $eventId, $content) {

  //Contextual information
  $userId = getUserId($session)      //Retrieving a unique
UserId
  $ip = getIp($server)               //Retrieve   the   ip
address
  $device = getDevice($session) //Retrieve the device type
  $timestamp = time()           //Retrieve   the   current
timestamp

  //Save on the database
  $userHistory  =  new  UserHistoryManager($userId,  $ip,
```

```
    $device, $timestamp);
      $userHistory->save($descriptor, $eventId, $content);
        }
```

Figure 4.9	Detail of the logging function used by the Log Manager

Below we report the list of the tracked interactions that concern this investigation[4].

$descriptor	Event Description	Associated action on the interface component
search_hike	The user searches for itineraries in the itinerary catalogue	Using the research text bar Using the activity selector Using the location selector
consult_hike	The user opens an itinerary of the catalogue	Clicking on one of the itinerary displayed in the itinerary catalogue
create_hike	The user creates a new itinerary	Clicking on the creation button in the Mhikes editor
search_event	The user searches for events in the events catalogue	Using the research text bar Using the activity selector Using the location selector Clicking on the interface calendar
consult_event	The user opens on event of the catalogue	Clicking on one of the event displayed in the event catalogue
create_event	The user creates a new itinerary	Clicking on the creation button in the event system
join_event	The user participates to an event	Clicking on the participate Button in the event description
user_profile	The user consults the profile of another user	Clicking on the avatar of the user

Figure 4.10	The interactive events tracked by the software probes managed by the Log Manager

The events are finally stored in a database to be ready for the persuasive manager analysis.

[4] additionally to the mentioned events, other interactions were tracked but we avoid entering in the details of these ones since they are not strictly connected to this investigation work.

id	user_id	hashIp	ip	device	timestamp	action ▾ 1	content
507	27090	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1539096426	create_hike	5bbcbf6ac31e5
508	27090	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1539096478	create_hike	5bbcbf9dc5b7f
531	2	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1539153783	create_hike	5bbd9f7738d8e
532	2	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1539153798	create_hike	5bbd9f863d7bb
533	2	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1539154065	create_hike	5bbda08fe720f
570	47636	20327b1f7aa0d60ddca3966d09a73e6a20745f09c88ffca1a5...	92.184.117	mobile	1539340583	create_hike	5bc0792727b7e
617	47127	3e29c9543423c509595f932f5fd5f12055f7831358f3174643...	83.113.23	web	1539507704	create_hike	5bc305f8d8e19
628	24706	3970418037b5582c807fe645f61e1cfce5d89b704cef9bcb6b...	88.187.238	web	1539542673	create_hike	5bc38e91435a4
629	24706	3970418037b5582c807fe645f61e1cfce5d89b704cef9bcb6b...	88.187.238	web	1539542737	create_hike	5bc38ed110558
659	2	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1539690389	create_hike	5bc5cf952c2a3
13	46984	9f5f5e174d1fe8dd0f9571598b0a59818d6fba103eb9c80bb2...	*NULL*	web	1537875465	create_event	37
53	47028	4532daeb8c4435d5b7f0f36ccd2e59aa19e6353d58696a9bdb...	*NULL*	web	1537950427	create_event	38
233	46396	51b755574449a28db3189c66ec74f90ef056f68d3899ae52a0...	*NULL*	web	1538422112	create_event	39
273	2	539e79f89bcdd5ac07be377ddc606e99265a7e1bac074d101d...	92.154.24	web	1538473402	create_event	40
341	47127	ae4b34ab1ba918f46f6255adee1b5df182e85cf3f78760e903...	86.205.69	web	1538650589	create_event	41
410	47397	470e28e30bd3d51e197c3426098899db8ee94bd1ae0bddd6ec...	185.171.114	ios	1538812283	create_event	42
516	47837	56ad3ee666f21f48534f384e86b3ae57c0a28beab7a98b3d11...	88.184.241	android	1539119024	create_event	43

Figure 4.11	Extract of the database table where the events are stored.

4.3.3 Persuasive manager

The persuasive manager is the architectural component responsible for analysing the contextual information of users to engineer personalized persuasive strategies.

The persuasive manager reads the database table created by the logging system, aggregating the information to produce new knowledge exploitable for persuasive purpose.

In the persuasive manager different back-end tools have been implemented to analyse the available aggregated information. An example is a visualization tool permitting to have a graphical vision of the users' interaction occurred in the system in a precise day/month.

We clarify that the back-end graphical tools that we will be presenting in the following are tools developed four our personal understanding of the data, and that, once the persuasive architecture was deployed, the analysis of the interaction and the aggregation of the information was performed automatically by the persuasive manager of the persuasive architecture.

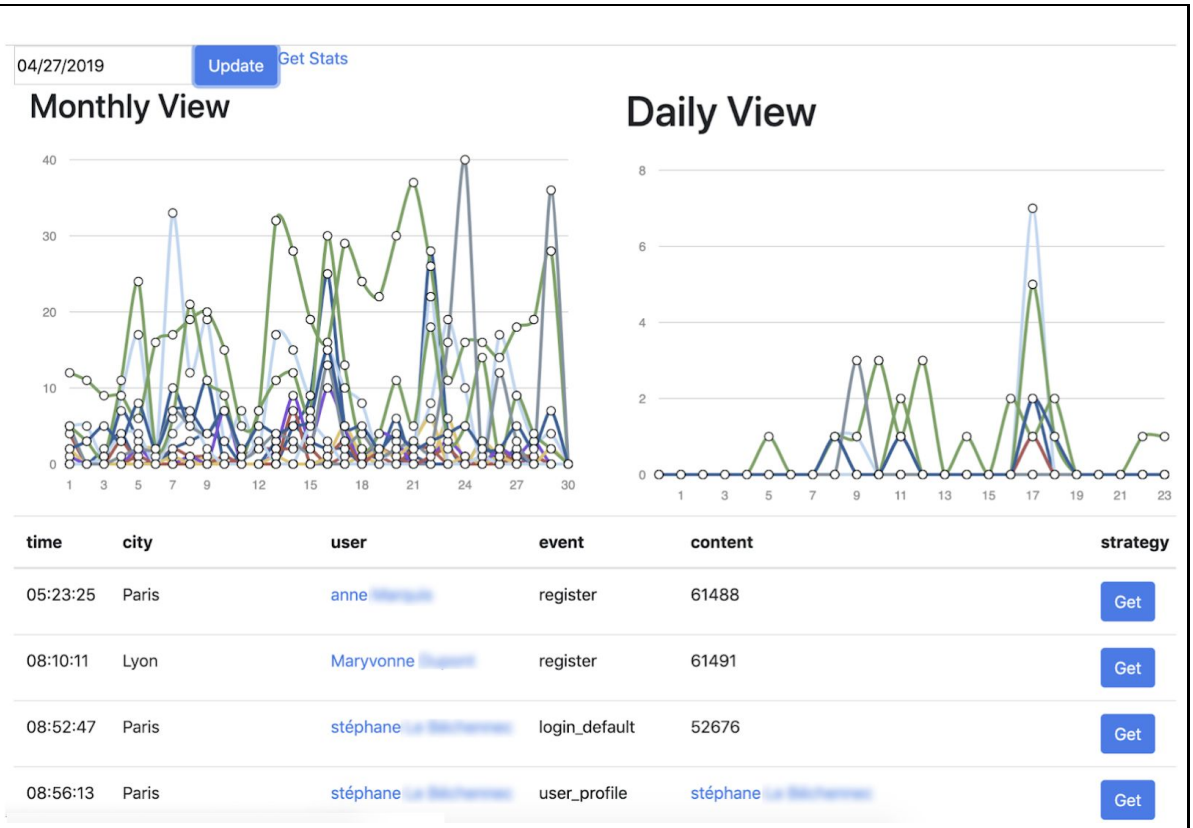


Figure 4.12	Back-office view of the persuasive manager showing the events recorded per month (on the left) and per selected day (on the right). The detailed events list on the bottom shows some associated contextual information

In the picture above, each interaction (e.g. register, login, user_profile, create_hike, create_event, etc.) is represented by a different color in the graph. On the left graph, it is possible to follow the evolution of the interactions on the whole month (April in the example) while on the right the details of a precise day are shown (the 27th of April in the example). Below the graphs, a list of actions ordered by time permits to have more detailed information: the user that performed the action, his or her approximative location (retrieved by their ip address) and other information associated with the action (if the user consulted an itinerary, the name of the itinerary; if the user created an event, the name of the event; etc.).

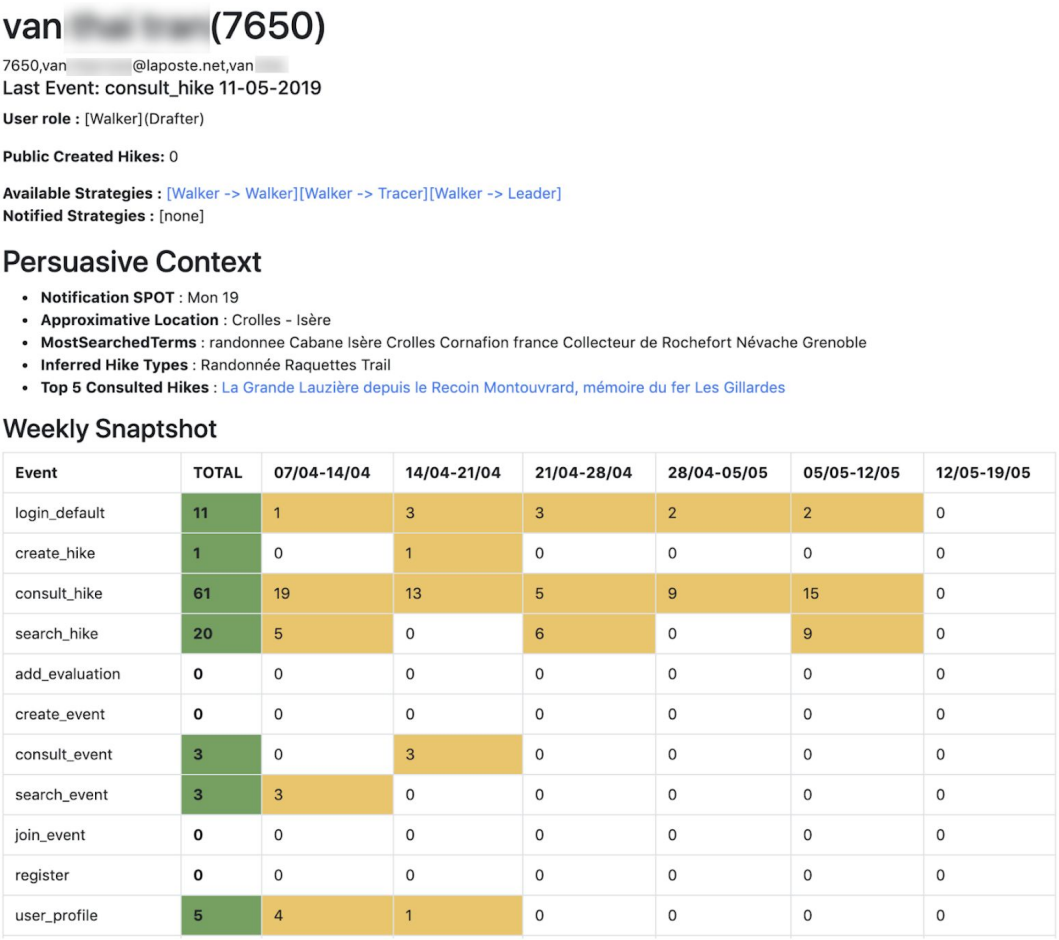
The button “Get” strategy permits to manually launch the aggregation of the information regarding a single user to have an instantaneous vision of his/her context and to check what are the dedicated persuasive strategy, that can be currently employed.

Let's give an example of how the persuasive manager produces the persuasive strategies from the users' contextual information.

In Figure 4.13 we can see the contextual information associated with the user 7650 (whose personal details have been anonymized):

- The **last event** performed in the system (consult_hike) and when it was performed (11 May 2019),
- The **role:** in the picture the user is classified as **Walker** since he or she has performed actions related to consume contents in the system (consulting itineraries). The word "drafter" indicates that the user has tried to create an itinerary to become a producer (a Tracer) but that for the moment that itinerary is still not public for the other users. This is also captured by the indicator Public Created Hikes = 0.
- The **notification spot**: it is the moment of the week in which the user has the bigger interaction with the system. In this case the user mostly uses the system on Monday at 7pm.
- **Approximative location**: the system does not record the full IP of the user for the privacy[5], but even with a partial information on the IP, it is possible to know his/her approximate location, from where the user is interacting (e.g. Crolles - Isère).
- The **most searched terms:** they are a list of searched terms on the system ordered by occurrence (in the example the user has searched 'randonnée' (hike in english) the majority of time).
- The **inferred hike type:** it represents the type of itinerary the user has consulted the most, ordered by occurrence (in the example the user has consulted a majority of itineraries of type randonnée (hike in english), this is coherent with the most searched terms).
- The **top 5 consulted hikes:** they are the itineraries that the user has consulted the most.

[5] for example the IP 113.203.167.143, was stored as 113.203.167.***



van [redacted] (7650)

7650,van [redacted]@laposte.net,van [redacted]

Last Event: consult_hike 11-05-2019

User role : [Walker](Drafter)

Public Created Hikes: 0

Available Strategies : [Walker -> Walker][Walker -> Tracer][Walker -> Leader]
Notified Strategies : [none]

Persuasive Context

- **Notification SPOT** : Mon 19
- **Approximative Location** : Crolles - Isère
- **MostSearchedTerms** : randonnee Cabane Isère Crolles Cornafion france Collecteur de Rochefort Névache Grenoble
- **Inferred Hike Types** : Randonnée Raquettes Trail
- **Top 5 Consulted Hikes** : La Grande Lauzière depuis le Recoin Montouvrard, mémoire du fer Les Gillardes

Weekly Snaptshot

Event	TOTAL	07/04-14/04	14/04-21/04	21/04-28/04	28/04-05/05	05/05-12/05	12/05-19/05
login_default	11	1	3	3	2	2	0
create_hike	1	0	1	0	0	0	0
consult_hike	61	19	13	5	9	15	0
search_hike	20	5	0	6	0	9	0
add_evaluation	0	0	0	0	0	0	0
create_event	0	0	0	0	0	0	0
consult_event	3	0	3	0	0	0	0
search_event	3	3	0	0	0	0	0
join_event	0	0	0	0	0	0	0
register	0	0	0	0	0	0	0
user_profile	5	4	1	0	0	0	0

Figure 4.13	Contextual information of user 7650, available strategies, persuasive context and details of performed actions by week.

The **weekly snapshot** gives also a concrete vision of how the events performed by the user were distributed along the time. The total count of events confirms that the most performed ones are **consult_hike** and **search_hike**. This is coherent with the fact that the inferred user role is **Walker.** With this information, the persuasive manager is capable of creating two personalized strategies for the user: **Walker->Tracer** and **Walker->Leader**. These strategies indicated the possibility of notifying to the user a role-switching to a different role, respectively to Tracer (of itinerary) or to Leader (of a social event), or to remain in the current role **Walker->Walker** (the strategy Walker -> Follower (consumer of social events) is not mentioned in Figure 4.13 since any event to be joined was available in the system).

Taking for example the strategy **Walker->Leader**, the system can suggest to the user to create a social event on an itinerary matching his/her interests.

For example, “hikes” near to “Crolles - Isere” may be relevant for the user and the most consulted itineraries suggest directly an itinerary to be used in the persuasive trigger.

The last information missing is when the message should be triggered to the user. The persuasive context shows that the user is interacting mostly at 7pm on Monday. This may be a favorable moment to send the trigger.

In Figure 4.14 we give an example of how the persuasive manager adapts the information to create a persuasive trigger related to the user 7650 in the analysis.

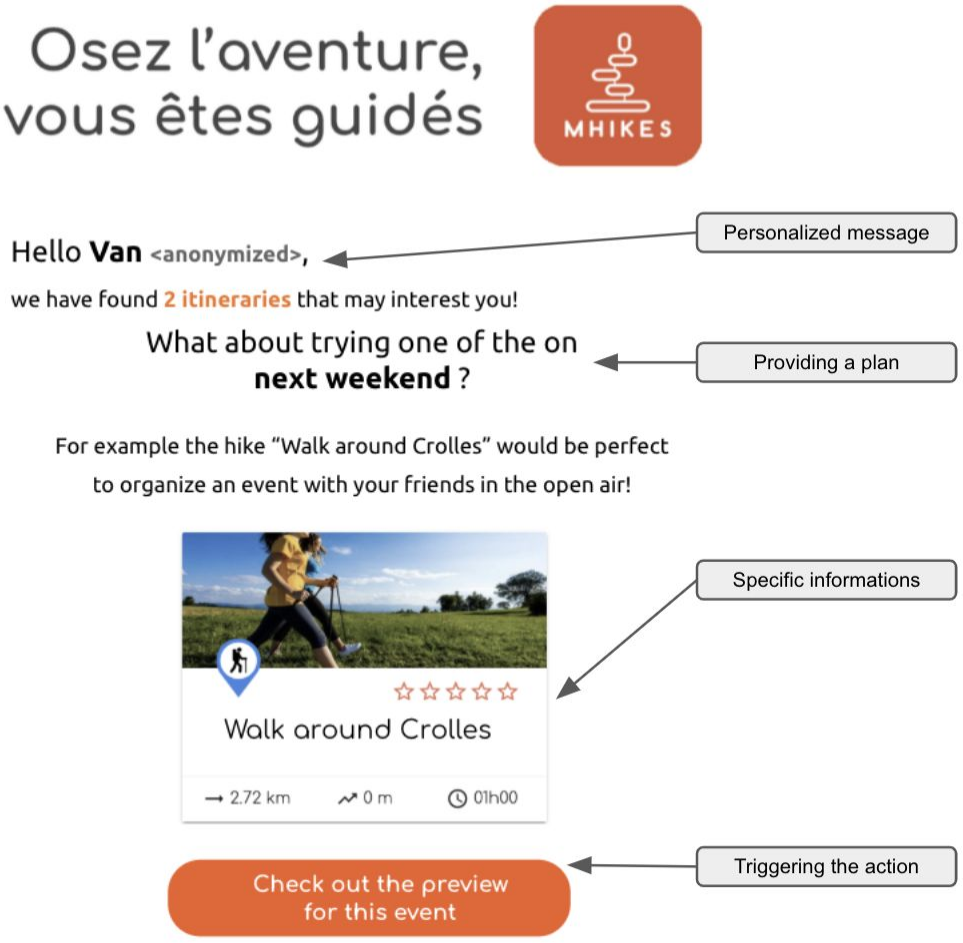


Figure 4.14	Example of persuasive trigger to be sent to the user 7650, aiming to switch his or her role from walker to leader.

The trigger refers to the role-switching strategy **Walker->Leader** to persuade the user to become organiser of social events.

The trigger (an email in our case) begins with greeting the user using his or her first name and by providing an itinerary that may match his or her profile. In order to facilitate the action the system provides a scenario and a preview of the social event. Finally a button permits the user to engage the role-switching bringing him or her directly on the task. In the case of our example the user is redirected to the event creation (Figure 4.6).

4.3.4 Notification manager

This component is responsible for scheduling the persuasive triggers. The notification manager is associated with a set of channels through which the persuasive triggers are delivered. Example of these channels are newsletters, emails, notifications of the applications (e.g. android notifications) etc. In the Mhikes case study we have chosen to drive the persuasive messages using emails. This choice is due to the fact that notifications were not available in the Mhikes application at the moment of the experiment design because their implementation was not compatible with the business plan of the company.

ID	user_id	strategy	url	hike_id	programmed	status
11	50747	T2L	cMVc	5c14caae4a7aa	5th March 2019 09:00	2
13	54067	T2L	cMVe	5c4ef1d782f45	5th March 2019 09:00	1
14	48705	T2L	cMVm	5bd1eacb5b243	5th March 2019 09:00	1
15	52676	T2L	cMVg	5c33a85688a49	5th March 2019 09:00	1
16	53884	T2L	cMVi	5c745e0de68fb	5th March 2019 09:00	0
17	55252	T2L	cMVk	5c63379ecbb9a	5th March 2019 09:00	0
18	46984	L2T	cMVq	5c34e304a60b7	5th March 2019 09:00	0
19	54623	T2L	cMVw	5c59ad14bcce8	5th March 2019 09:00	1
22	18483	T2L	cMVC	5c52fcf9dd37a	5th March 2019 09:00	1
23	50638	W2L	cMVA	5c02a786965fd	5th March 2019 09:00	1
24	52724	T2L	cMVG	5c34e304a60b7	5th March 2019 09:00	0
25	52239	T2L	cMVu	5c2c902636b5b	5th March 2019 09:00	0
26	55281	T2L	cMVE	5c6312ab469e8	5th March 2019 09:00	1
27	47010	W2L	cMVs	5baa9003aa8f2	5th March 2019 09:00	0
28	50329	T2L	cMVy	5bf50a390e4d3	5th March 2019 09:00	1
29	55127	T2L	cMVo	5c61778fbc69b	5th March 2019 09:00	1
30	48864	L2T	cMW6	5c6312ab469e8	5th March 2019 09:00	1
31	51678	L2T	cMVO	5c34e304a60b7	5th March 2019 09:00	1

Figure 4.15 Extract of the table where the Notification Manager stores the scheduled strategies

The strategy table of the database stores all the persuasive triggers made available from the persuasive manager. In Figure 4.15, each line represents a different trigger: the user_id column identifies the persuadee; the strategy column represents the type of switch suggested by strategy (e.g. L2T means Leader to Tracer, W2L means Walker to Leader); the "programmed" column identifies the scheduled delivery time. The "url" indicates an univoque code that permits to track the status of the strategy, made explicit in the last column status. The status code respectively represents: 0 the strategy has been delivered (email sent), 1 the strategy has been red by the user (email opened), 2 the user has clicked on the persuasive trigger (click on the email trigger), 3 the strategy has produced the targeted behavior (the persuasive log has registered the targeted action on the system), and -1 the user has refused the trigger (e.g. email marked as spam).

The strategy status represents an important information for the persuasive manager: it can indeed check if a strategy has been successful or not for a given user. This information influences the next persuasive strategies produced, accounting what are the strategies that have to be privileged and the ones to be avoided. This generalization may be done for each user but also extended at the system level to become a characterizing factor of the specific domain of application. In the case of CRegrette for example, once we noticed that persuasive notifications were counterproductive during the working time, we modified the delivery time to the early morning, which revealed to be a more effective strategy.

4.3.5 Architecture in action: insights from software probes

The persuasive probes intanciated on Mhikes immediately permitted to collect insights on the users of the system and their interaction. These insights were used to validate and test the effectiveness of the implemented **software probes**, but besides the validation we also discovered new interesting information on the way that users were using the system, giving clues on the **behavior determinants** that lead them to interact.

For example a very first analysis of the persuasive probes permitted to characterize how the interaction was distributed along the week. In Figure 4.16 the number of registered actions of creating hikes, consulting hikes, creating social events and consulting social events has been plotted according to the day of the week.

Focusing for example on the consulting activity, it is immediately possible to observe an increment from Thursday to Saturday. This may be due to the fact that users are **looking for a last-minute plan** these days. Indeed on Sunday the consultation decreases, meaning that the plan should already be defined. This probably means

that scheduling a persuasive trigger that suggests a plan for the weekend may be less effective on Sunday. Tuesday the consultation also appears to increase, but since the weekend is far, probably this kind of users operates a **more accurate planning**: they are really defining all the details of their next outdoor activity. In this case the same notification may be more effective.

This is an example of how looking at very basic data collected by the persuasive architecture may lead to the design of more effective persuasive strategies.

Software probes may also be used to get hints on the design of the persuasive interfaces.
In Figure 4.17 different software probes associated with different interactions on the system are plotted on a pie chart. Our expectation with this plot was to have the majority of actions related to the itinerary creation and on the itinerary consultation, since the social events system was recently introduced and not completely mastered by users.

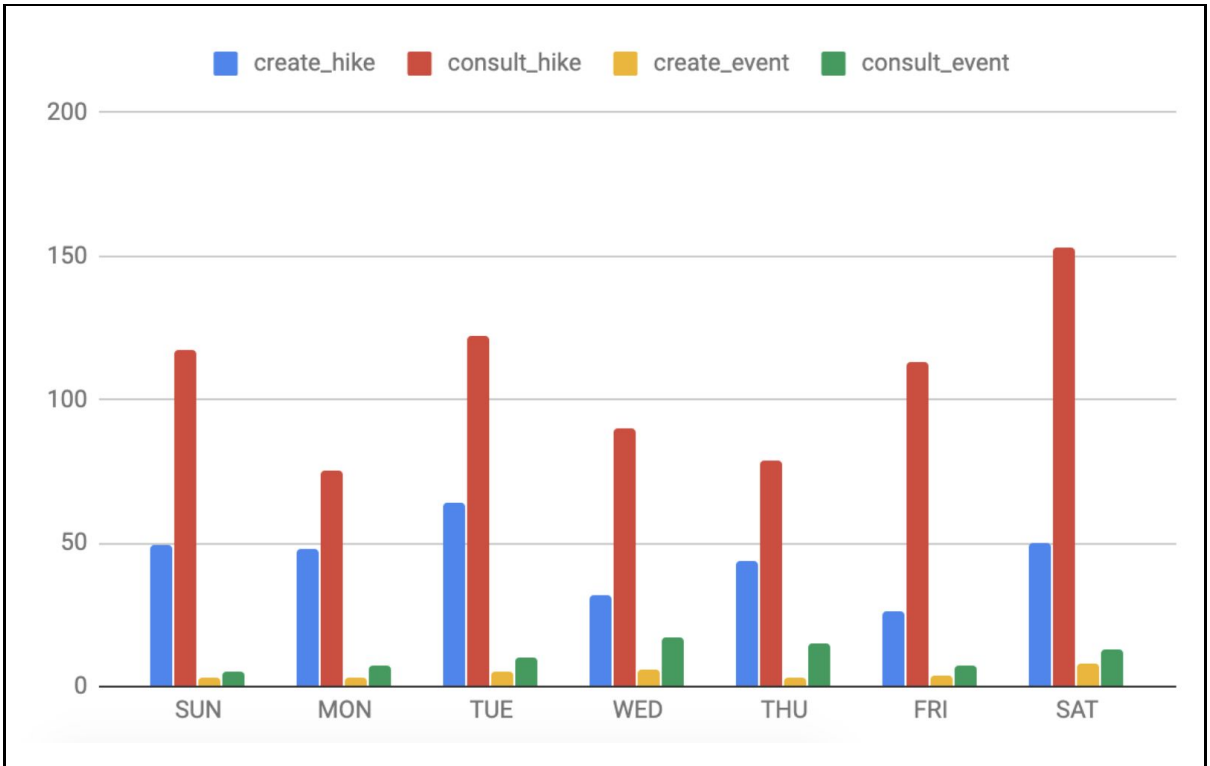


Figure 4.16	Distribution of hiking creation, hiking consultation, social event creation and social event consultation on the day of the week

What we found confirmed our hypothesis, indeed more than 62% of users performed actions related to searching, consulting or creating itineraries on the system. What was not expected was to find among the data a 23,8% of users that were consulting user profile pages, which we supposed to be a minor task. An example of profile page is provided in Figure 4.16.

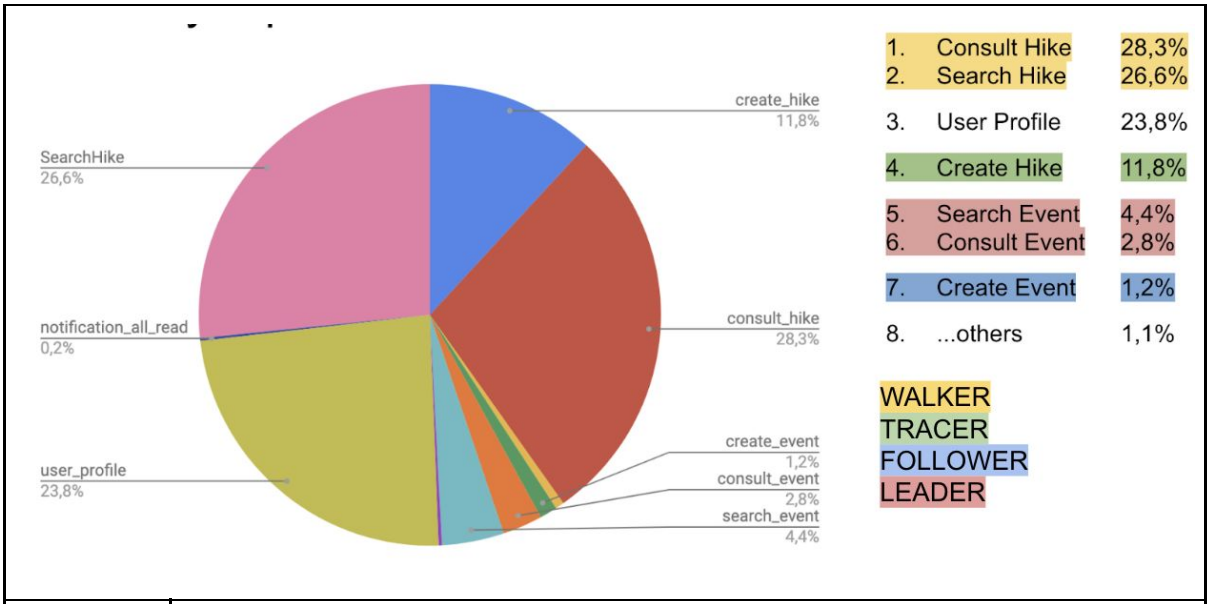


Figure 4.17	Cumulative representation of different software probes related to the system features

This unexpected data brought to perform a deeper investigation of the phenomenon, and we found that all the user profile consultations were related to consulting his/her own profile, which appeared to be even more unexpected.

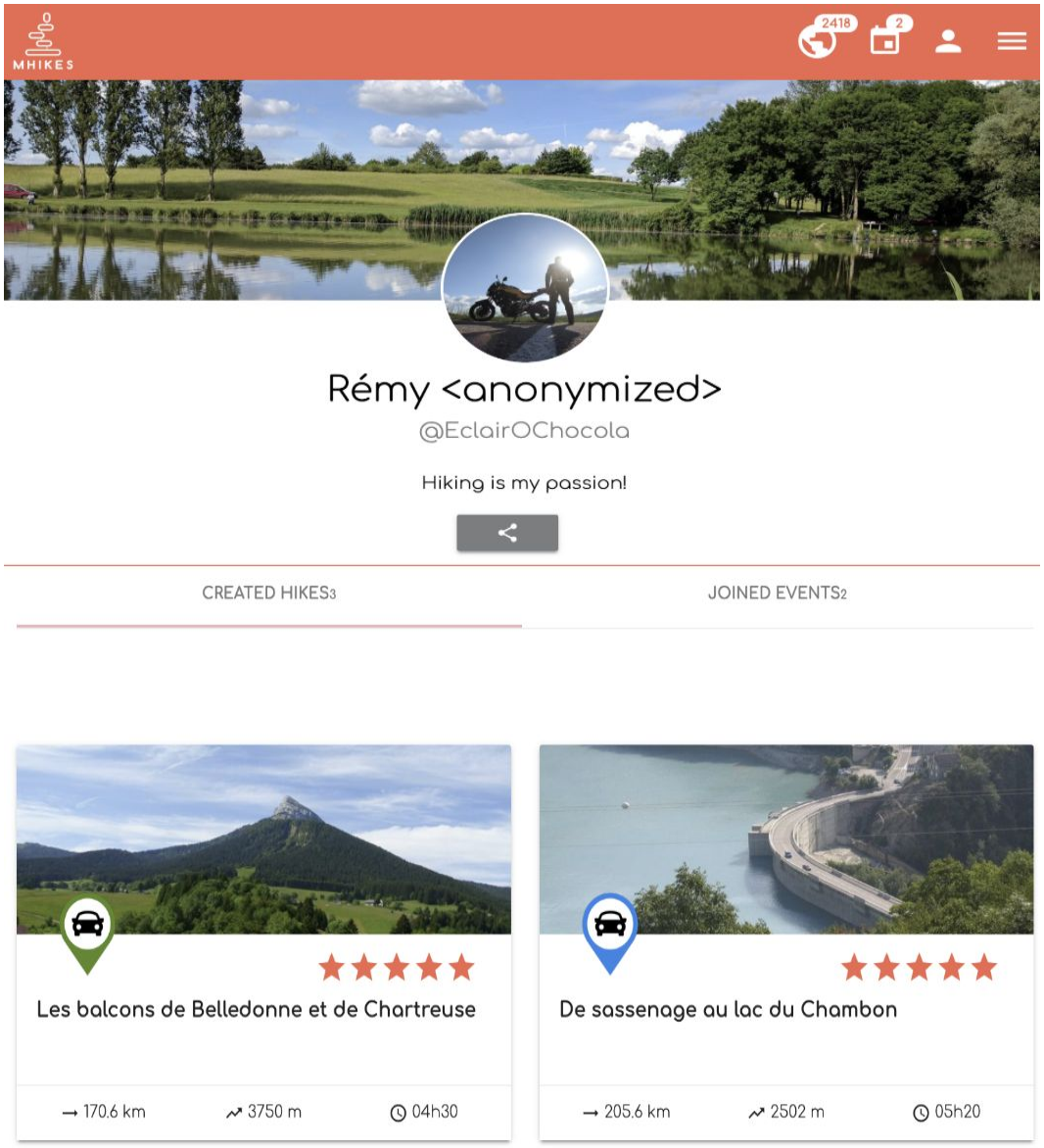


Figure 4.18	Example of a user-profile page.

Finally it was possible to find the explanation in the design of the homepage of the website.

Figure 4.19 shows the execution of the logging task on the Mhikes website. On the top the user is not connected, and clicks on the upper-left bar to perform the login. The login is performed inserting the email and password (central picture), finally the same webpage is displayed to the user but including an avatar and the name of the user to confirm that the login was successful (bottom image).

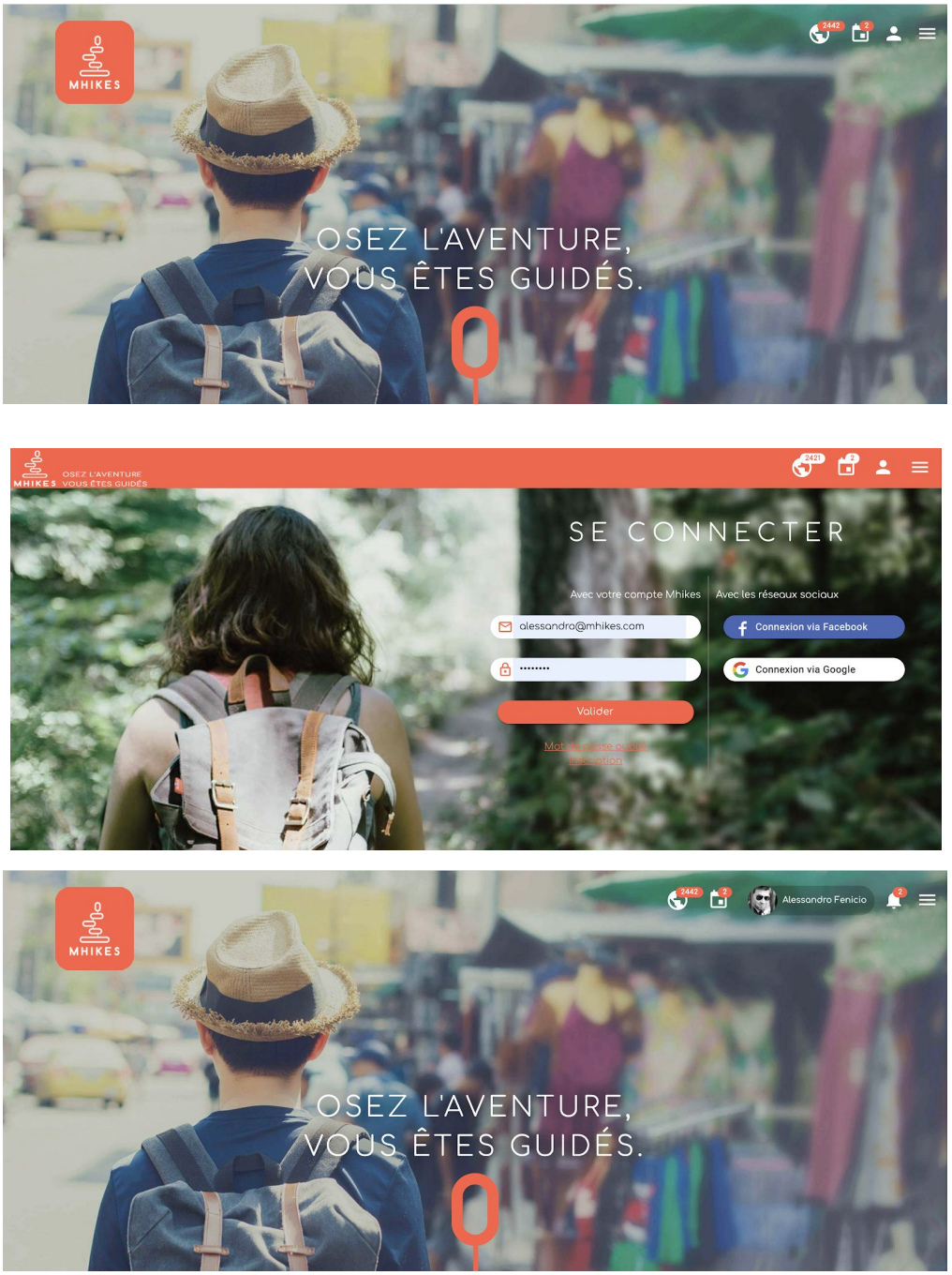


Figure 4.19	Screenshots of the logging task performed on the Mhikes website

Users indeed were implicitly persuaded in clicking on the avatar, since this was the only element that changed after the login. Clicking on this component was thus bringing them to self-consult their user profile. This example shows how software probes may be used to better understand the **behavior of the user** on the system

and to **observe possible issues** in the interface that can be corrected. This understanding may also improve the detection of ergonomic flaw in the system interface.

Persuasive design can also take advantage of this kind of information. For example the action analysis that brought to plot the chart on Figure 4.17 was performed with the objective of finding a possible user task to be associated with the delivery of persuasive notification.

As a result, having that the user profile was often consulted after the login brought to formulate the hypothesis of including the trigger notification directly in the profile page, integrating the delivering of triggers directly in the usual user task routine.

4.4 Synthesis

In this chapter we have exemplified how to translate the conceptual approach presented in Chapter 4 into software artefacts. We have followed a progressive approach: 1) non persuasive Mhikes, 2) towards persuasive Mhikes, 3) persuasive Mhikes, 4) persuasive Mhikes in action.

First we have conducted an analysis of ergonomy of the **non persuasive Mhikes**, providing a concrete method that developers can follow to ensure these properties on any system, based on the Ergonomic Criteria for the evaluation of Human-Computer Interfaces (Bastien and Scapin 1993). Successively we have conducted a preliminary study **toward the Persuasive Mhikes**, to evaluate the user understanding of the persuasive feature identified to be implemented. The chapter continues by explaining the **persuasive version of Mhikes** in detail, giving concrete examples on the implementation. All the components of the architecture are detailed to describe how persuasive paths operate and of how **persuasion is operationalized**.

The implementation on Mhikes showed all the potentiality of persuasive paths. The investigation provided a complete approach starting from the modeling of behavior, toward a conceptual design of persuasion, arriving up to the final operationalization.

The architecture implementation fulfilled our operationalization expectations and gave new significant clues to characterize the behavior determinants associated with the process of change.

The implementation of the social events system was delivered in the production version of Mhikes (the latest released online) and all the Mhikes users were able to access from these features from all over the world.

In the next chapter, we proceed to an evaluation of the Persuasive Mhikes, made by testing the system features on real users. A set of experiments and pilots will provide insights and discussion points on the feasibility and performances of our approach.

5 Evaluation

The Mhikes implementation described in the previous chapter is a practical proof-of-concept, demonstrating the technical feasibility of persuasive paths. In this chapter we cover an evaluation from the user point of view, based on the system usability and on the persuasive effectiveness. The needed time to evaluate if persuasion has permanently changed behavior of individuals does not fit within the duration of a thesis, for this reason. In the work we have explained how the role switching mechanism may enact and speed-up the process of change. For this reason we have focused our evaluations on the interactive roles, on the triggering strategy in order to obtain as soon as possible clues on the behavioral evolution.
The evaluations, thus, are twofold in order to assess:

- The usability of the proof-of-concept and its technical ability to distinguish roles, and
- In terms of persuasion, the effectiveness of role-based persuasive triggers and their effect on role switching.

For this reason we continue this evaluation chapter providing practical on-the-field studies corroborating this thesis. In the following we will be experimenting in Mhikes:

1. The feasibility of detecting the roles in an interactive system,
2. The impact of persuasive triggers on the users,
3. The impact of the role switching strategy on the users.

The following experiments and pilots will involve real users, for this reason, before digging into them, we have conducted a preliminary study on the usability of the implemented event system on which the experiments are based. The objective of this pilot study, thus, is to ensure that the implemented features satisfy the usability criteria, in order to avoid any bias on the successive experiments.

5.1 Preliminary usability pilot

A persuasive interactive system needs to be first of all usable and encounters the satisfaction of the final users. Evaluating usability is propaedeutic to a correct assessment of persuasion. If usability issues, it is not possible to discern between these issues and the ineffectiveness of the implemented persuasive features. Similarly, if the system extended with the new features is found to be more persuasive, it is not possible to discern between the effectiveness of persuasion and improvements caused by a better usability.

For this reason, the first step is to evaluate the usability and to remove any issue to be sure to correctly evaluate the persuasive aspects.

In this experiment we evaluated the usability, and if the implementation of the events system on Mhikes satisfied the expectations of final users.

Evaluation method: we recruited a group of N=11 participants with mixed backgrounds aged between 22 and 42 years and with an average self-estimated Information technologies level 2,8 out of 4.

On this sample we have used the following **protocol**:

1. Presenting the general objective of the study and collecting the personal data.
2. Brief informal interview to understand the user knowledge of information technologies and their knowledge on outdoor activities.
3. Introducing the main functionalities of Mhikes and clarifying any doubts.
4. Providing a scenario.
5. Asking to perform a set of actions on the Mhikes System.
6. Providing a CSUQ Questionnaire form to be filled after the tasks completion.
7. Collecting any ulterior post-experiment feedback.

This is an example of scenario provided on Step4:

> *It's spring, you live in **Grenoble** and you are in **normal physical condition**.*
>
> *The good season motivates you to perform your favorite activities: **mountain biking** and **hiking**. Your friend **Bob June** suggested Mhike, a platform that he uses to participate in these activities. Mhikes platform permits indeed to participate and organize open-air social events.*

This is an example of Step5:

1. Next weekend you are free, and you decide to use the Mhikes system to find an event that may interest you. The event should be in Grenoble, related to hiking and it should take place next weekend. **How do you proceed?**
2. You have found an event interesting. Before participating, you decide to ask to the organizer if a nearby parking is available to park your car. **How do you proceed?**
3. The organizer answered your question: the system generated a **notification** in the interface, **have you found it?**

4. You have finally decided to participate to the event. You have now to confirm your presence to the event through the system. **How do you proceed?**
5. You want to invite your friend **Bob June** to participate to the same event. **How do you proceed?**
6. You have decided to create an event for your next birthday.
   The event should be restricted to your friends and should happen in Lyon. **How do you proceed?**

During all the experience participants were asked to speak aloud and to comment on why they were interacting with each component of the interface. Next step was to ask them to fill a usability satisfaction questionnaire.

We relied on the IBM Computer Satisfaction Usability Questionnaire (CSUQ) (Bravata et al. 2007), an empirically-validated 19-questions-based questionnaire benefiting from an alpha[6] = 0.89 reliability coefficient related to usability, thus meaning that answers provided by participants to this questionnaire demonstrate a high correlation with the usability of the system being evaluated. Each IBM CSUQ closed question was measured using a 7-point Likert scale (1=strongly disagree, 2=largely disagree, 3=disagree, 4=neutral, 5=agree, 6=largely agree, 7=strongly agree) and was phrased positively as follows:

Q1: Overall, I am satisfied with how easy it is to use this model.

Q2: It was simple to apply this model.

Q3: I can effectively complete my task applying this model.

Q4: I am able to complete my task quickly applying this model.

Q5: I am able to efficiently complete my task applying this model.

Q6: I feel comfortable applying this model.

Q7: It was easy to learn to apply this model.

Q8: I believe I became productive quickly applying this model.

Q9: The model provides me with structured guidance on how to fix problems.

Q10: Whenever I make a mistake using the model, I recover easily and quickly.

[6] The most common way to estimate the reliability of these types of scales is with coefficient alpha (Nunnally, 1978). Coefficient alpha can range from 0 (no reliability) to 1 (perfect reliability). Measures that can affect an individual's future, such as IQ tests or college entrance exams should have a minimum reliability of .90, and preferably a reliability of .95. For other research or evaluation, measurement reliability should be .70 to .80 (Landauer 1988)(Nunnally, 1978).

Q11: The information provided by the model and its accompanying method is clear.

Q12: It is easy to find the information I needed.

Q13: The information provided for the model is easy to understand.

Q14: The information is effective in helping me complete the tasks and scenarios.

Q15: The organization of information on the model screens is clear.

Q16: The interface of this model is pleasant.

Q17: I like using the interface of this model.

Q18: This model has all the functions and capabilities I expect it to have.

Q19: I am satisfied in using this model.

In addition to the Likert scale items we inserted also an after-scenario questionnaire consisting in an open question phased as "List the most negative aspect(s):","List the most positive aspect(s):" and "Provide some optional general comments on your experience". Then, for each participant, the answers were anonymously stored into a database. The data analysis was carried out as indicated in the documentation of the CSUQ questionnaire computing the four indicators that provide the specific values of the **system usability** (related to questions 1 to 8), of the quality **of the information** (related to questions 9 to 15), of the **quality of the interaction** (related to questions 16 to 18) and of the **overall satisfaction** of the user (related to questions 1 to 19). Then we have computed the average of the answer for each indicator along with the standard deviation and the confidence interval computed with a confidence interval set on a critical value of 99%.

The results were the following.

Indicator (Questions)	Average	St.Deviation	Error on c.i. 99%
System Usability (Q1-Q8)	6,027597403	0,8726412682	0,6777296483
Quality of Information provided (Q9-Q15)	5,663203463	0,9002319956	0,6991577593
Quality of Interaction (Q16-Q18)	6,393939394	0,6466697907	0,5022307628
Overall Satisfaction (Q1-Q19)	5,966944992	0,7651708136	0,5942636055

Figure 5.1 CSUQ indicators results

Below we report a plotted version of the data provided in the previous table.

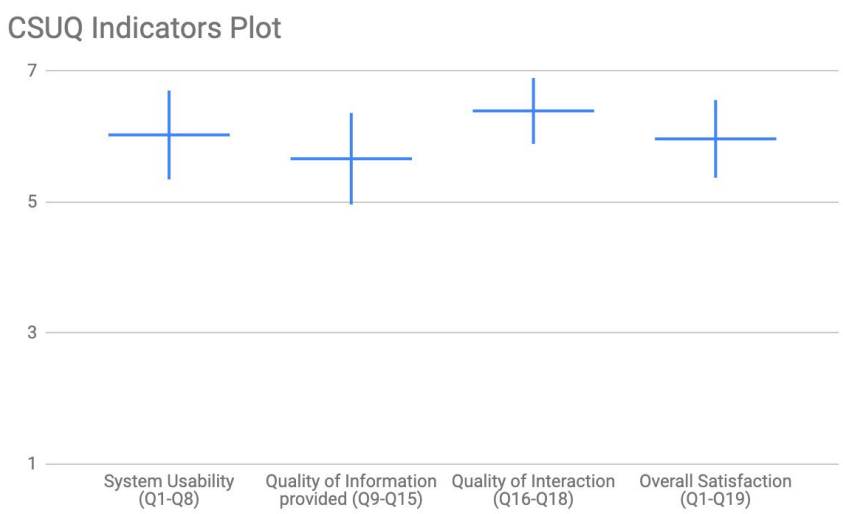


Figure 5.2	CSUQ indicators plotted

Results: results show that the four CSUQ indicators have all values greater than neutral likert value of 4 (even accounting confidence intervals).

Conclusion of the preliminary study on the usability: the pilot confirms that no usability issues were encountered by the recruited user, whose CSUQ indicators highly score over the neutral likert value of 4. The general comprehension of the event system was achieved by the participants who were able to correctly explain and complete the assigned tasks.

5.2 Detecting and tracking the roles of users

The **objective** of this experiment was to prove that (1) the interactive roles defined in the contribution section were detectable by using the software probes instrumented in the system and (2) to confirm that the same software probes were also able to detect the switching role mechanism over time.

Method: we have defined four roles in the contribution: walker, tracer, leader and follower. In this experiment we have restricted the role analysis to the **walker** and the **tracer** since analysing these roles was a business priority for the Mhikes enterprise.

In order to support our experiment, we performed an instrumentation of the Mhikes system. In particular, we set up a logging system capable of tracking which actions users were performing on the interface by using a log function storing their ID, the action performed, the timestamp and the device they were using. In the example below a snippet of pseudo code including the log function applied to the action of consulting an itinerary (see Figure 5.3).

```
# The User clicks the itinerary `i' in the interface
# The System loads the itinerary `i' from Database
# Logging the action (line below)
log('consult_itineary',i);
# The System shows the itinerary `i' in the interface
```

Figure 5.3	Snippet of the code that permits the log of user's interactive events

Each one of the interface components was then associated (at the code level) to two labels (tracer or walker) indicating if that given component belongs to the task of tracer or to the task of walker (e.g. button "create new itinerary" → label = Tracer, "click on an itinerary to see the contents" → label = Walker) as shown in Figure 5.4.

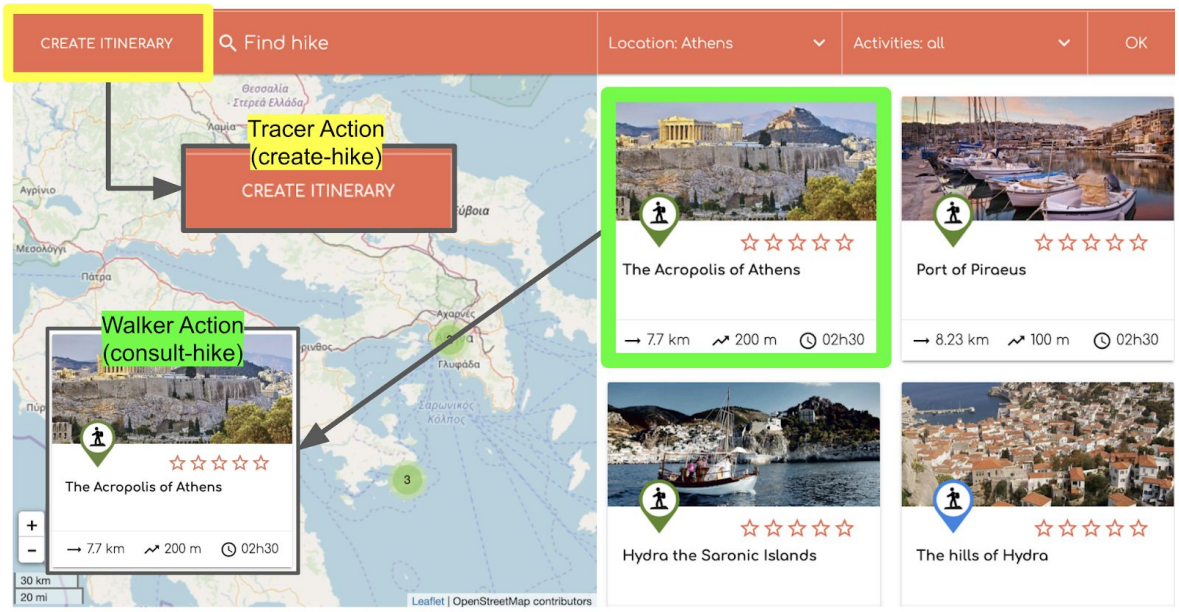


Figure 5.4	Example of components of the interface connected with software probes to track the user interaction

Using this instrumentation, before the introduction of the persuasive features, we stored around 5000 interactions from around **260 real users** using the website of Mhikes system[7]. Each interaction of the users was associated with a specific role and this made possible to track wherever the user was behaving for example as Tracer or as Walker. Extending this analysis over more weeks made possible to understand how users were switching roles.

In order to make a cumulative analysis of how roles were distributed before the introduction of persuasive strategies, we then constructed a contingency table grouping users that used or not the **create_hike** and **consult_hike** functions. The result is shown in the table below whose χ^2 value test is 6.9328 (degree of freedom = 1) and it's p-value is equal to 0.008463.

[7] Users were not recruited, data were taken from a real usage of the system during 25 weeks.

	The user **never consulted**	The user **consulted** (at least one time) = Walkers
The user **never traced**	84	122
The user **traced** (at least one time) = Tracers	34	21

Figure 5.5	The table shows the results of the data analysis dividing users into users without a role (84), users endorsing only the tracer role (34), users endorsing only the walker role (122) and users endorsing both roles (21)

Several considerations may be obtained analysing the data:

1. The 60% of users endorse a unique role (156 / 261)
2. The 78% of users endorsing one role is a **Walker** (122 / 155)
3. The 22% of users endorsing one role is a **Tracer** (34 / 155)
4. The 68% of users with at least one role is **Tracer** (122 / 177)
5. The 19% of users with at least one role is **Walker** (34 / (177)

Result: the aforementioned analysis **confirms** that the majority of users endorse one role among Tracer and Walker, and that most endorsed role is the role of Walker. The analysis of the logs confirms the feasibility of introducing software probes in the system Mhikes to support a role-based engineering approach. In particular we can confirm that: (1) the two distinct roles are detectable and that the users mainly chose to endorse just one of the interactive roles at the time, and (2) to tell whether or not a user switches between roles over a given period of time.

Conclusion: the experiment confirms the feasibility from a system point of view of introducing a set of software probes to (1) detect the main role that users are endorsing in the system, and (2) to detect and track the evolution of the roles over time.

5.3 Role-based persuasive triggers

The **objective** of this experience was to investigate if the triggers generated by the persuasive architecture performed better in terms of view and click rate of users.

Evaluation protocol:

1. Selecting a set of notifiable users
2. Let the architecture computing their possible next steps in their persuasive path
3. Creating a personalized trigger
4. Scheduling the notification of the trigger
5. Sending the trigger
6. Evaluating the 'opening and click rate' of the triggers.

Method details:

The persuasive architecture presented in Section 3.4 permits a full deployment of the persuasive strategies passing through the following stages:

1. The logging system loads the available users in the persuasive manager. In particular are discarded the users whose contextual information is not sufficient to build a persuasive message (e.g. incomplete information on the type of activity to be notified or to the favorable persuasive moment).
2. The persuasive manager computes the next event of the persuasive paths for each user and the associated targeted action (e.g. create an itinerary, create a social event).
3. The persuasive manager prepares a personalized trigger to be sent to the users.
4. The notification manager schedules the trigger and sends it in a given date/time.
5. The persuasive manager tracks the clicks and the opening rate of the triggers.

For this experiment we have considered two role-based strategies:

- A switching trigger from the **Tracer role to the Leader one**.
  With this trigger we aimed at persuading the creators of itineraries to organize an event on one of their itineraries.

- A switching trigger from the **Walker role to the Tracer or Leader one**.
  With this trigger we aimed at persuading the users that used to consult itineraries to create their own hikes or to organize an event.

These two strategies have been later compared with the usual non-role-based strategy operated by Mhikes staff before designing these experiments.

We have chosen to send the triggers in two defined days of the week: Tuesday and Thursday. This choice has been made according to the community manager of Mhikes which recognized these days as the most effective to have the users reaction. However these two days figure as favorable days to send emails also in other field studies. We cite as support of this affirmation a marketing analysis study[8] published on internet where 14 different studies are compared to find the most effective day of the week to send emails. The result of the study found as most favourable days respectively: Tuesday, Thursday and ultimately Wednesday. This result is coherent with an ulterior test we have performed. We have analysed the number of performed actions of users in the system per each day. Excluding the days Friday, Saturday and Sunday where the time is too short to organize a weekend-plan (moment in which the system is used the most) the three days Tuesday, Thursday and Wednesday resulted to be the most suitable for the trigger delivering.

We have sent 351 emails using the two aforementioned trigger strategies on **Tuesday** and **Thursday**, collecting the opening and the click rate for each email. An example of email can be seen at Figure 5.6.

[8] https://coschedule.com/blog/best-time-to-send-email/

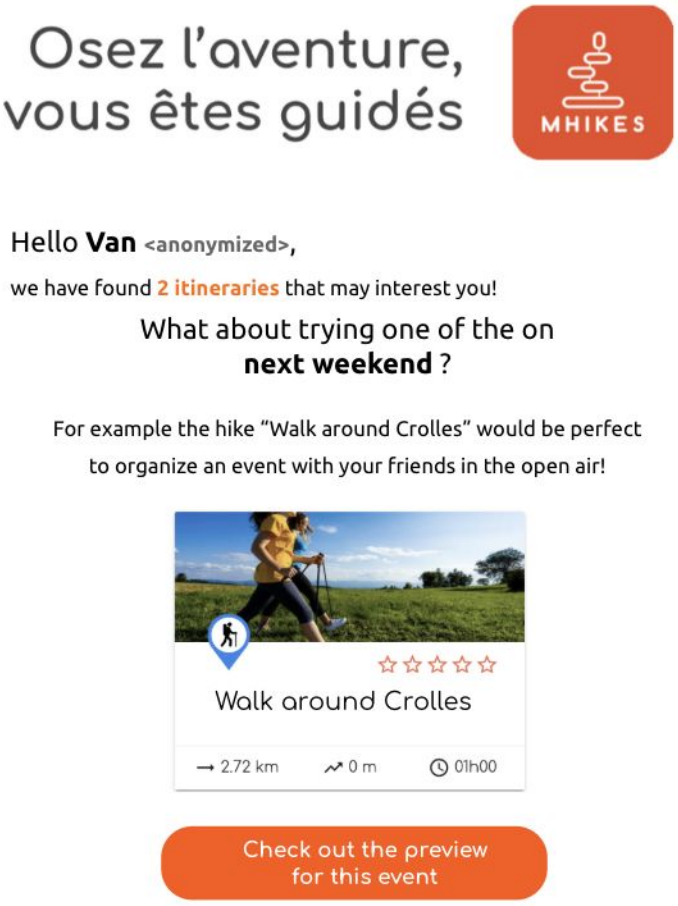


Figure 5.6	Example of persuasive trigger for switching from walker to leader.

Figure 5.7 below reports the results of the experiment. The first four rows show the combinations between the different triggers and different days. The following two report a cumulative view of the results obtained and of the historical data of Mhikes on the email delivering while the last row compares the variation of the percentage of the two cumulative views at the two rows above.

	DAY	N	Open Click	%	Open NoClick	%	NoOpen	%
Switch Tracers to Leaders	TUE	72	6	8,33	26	36,11	40	55,56
Switch Tracers to Leaders	THU	29	1	3,45	12	41,38	16	55,17
Walkers to Tracer or Leader	TUE	172	10	5,81	62	36,05	100	58,14
Walkers to Tracer or Leader	THU	78	6	7,69	33	42,31	39	50
D1: HISTORICAL DATA MHIKES	---	4636	204	**4,4**	1091	**23,53**	3341	**72,07**
D2 : Trigger Campaigns (cumulative)		351	23	**6,55**	133	**37,89**	195	**55,56**
Δ% = (D2-D1)/D1				48,86		61,03		-22,91

Figure 5.7 Summary of the statistics collected on the different persuasive triggers.

Results

A first consideration brings to observe that the percentage of opening and clicks is relatively low when sending emails to customers. However the triggered action may be completed successively to the email reception without clicking on the button and directly going to the Mhikes system. However despite the values are found to be of small volume, we can observe that the triggering campaigns performed better than the classic campaign method usually carried out by Mhikes: Open rate = +61%, Open+Click rate = +48%, and not Opening rate = -22.91%. Users reacted more on the persuasive triggering campaigns and the rate of not opening decreased. Speaking in detail about the triggers we can observe that both the triggers on the two days, compared with the historical data of Mhikes, performed better on the opening rate. For the clicking rate we globally found the same result, except for the switching trigger from tracer to leader on Thursday.

Conclusion

The triggers created by the persuasive architecture performed globally better than the generic triggers previously adopted by Mhikes on clicking and opening rate. As a limitation of this experiment we mention that, due to the limited database of notifiable users, we could send just 351 emails. An alternative to increasing the number of triggers sent could have been to integrate them into the Mhikes mobile application. However, when this experiment was designed, implementing this option was not aligned with the company development plan, and therefore it could not be tested.

5.4 Role switching

The **objective** of this experiment was to investigate if after delivering the persuasive triggers generated by the proposed persuasive architecture, the user switched of role in the Mhikes system.

In particular for this experiment we have considered the two role-switching triggers investigated at the previous experiment: switching **from tracer role to the leader one** and switching **from walker to tracer or leader role** on Tuesday and Thursday. This choice is motivated (as in the previous experiment) by the necessity of considering a consistent datasets for the analysis on which it was possible to conclude.

Evaluation protocol:

1. Retrieving all the users notified with the mentioned switch-role strategies.
2. Retrieving for each user their actions list related to the week before and after the delivery of the first trigger.
3. Retrieving the list of actions performed in the week after the delivering of the trigger.
4. Discarding all the users that did not have at least one action in the two weeks.
5. Comparing if the total number of actions performed after the delivery of the triggers was greater than the total number of actions performed before the delivery.

We provide a more detailed explanation of the experimentation method.

Let's consider the case of a given strategy delivered to a user by using a given trigger. In Figure 5.8 the timeline shows the moment at which the trigger was delivered. The week before this delivery is considered the background week (B) and the week after is the observing week (O). In these two weeks the sets of main actions are respectively calculated.

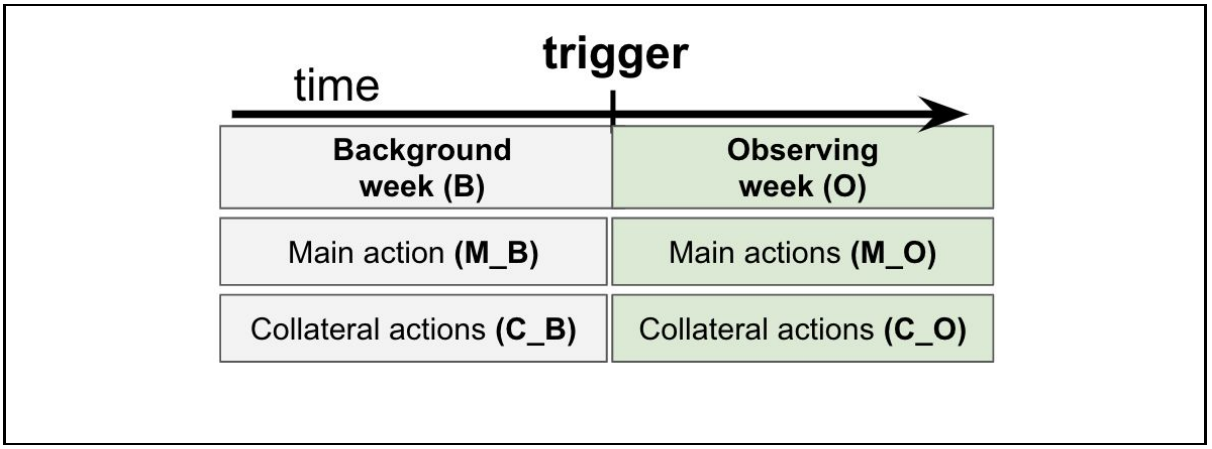


Figure 5.8	Protocol used to evaluate the persuasive strategies.

The **full set** of actions for the two considered strategies is: “creating a hike”, “creating a event”, “searching for a hike”, “consulting a hike”, “searching for an event”, “consulting an event”, “starting the creation of an event”. Among these actions we define two **main actions** for the role of Tracer and Leader, respectively: “creating a hike” and “creating an event”. Consequently, we define the set of **collateral actions** the ones that are not the main ones for each role. For example for the tracer role, the collateral actions are all actions except “creating a hike”.

Let’s have a pictured example on the user id=55252, a tracer:

User 55252

Tracer to Leader	26/02/19 to 05/03/19	05/03/19 to 12/03/19	Total before	Total after	Delta
Collateral actions	create_hike 27/02/19	consult_hike 10/03/19 create_hike 10/03/19 create_hike 10/03/19	1	3	2
Main actions			0	0	0

Figure 5.9	Evaluation of collateral and main actions for user 55252

The user 55252 has been triggered with a “Tracer to Leader” strategy on the 5/03/19.

Figure 5.9 shows the set of main and collateral actions computed on the background week (26/2 to 5/3 2019) and the set of main and collateral actions computed on the observation week (5/3 to 12/3 2019).

The strategy is considered valid at least one action is present in the observation week and the background week. In the example in Figure 5.9 it is possible to see that there is an increment of the collateral actions while there is no increment for the main action.

However, we remark that this constraint drastically reduces the volume of the triggers that could be considered valid, passing from a total of 340 triggers to a usable set of 25.

Results

We have analyzed 25 triggers meeting the postulated consistency requirements delivered to 21 Mhikes users.

The trigger “Tracer to Leader” tested on n=10 triggers persuaded 5 times the users to perform collateral actions in the system, but no increment on the main action “create event” was observed.

	Tracer to Leader Switch	Walker to Tracer/Leader Switch
Collateral Events	5	2
Main Events	0	1
Total Events	10	15
Averages		
Collateral Events	0.5	0.13
Main Events	0	0.07

Figure 5.10	Cumulated result of main and collateral actions realized after triggering the switch role strategy from Walker to Tracer/Leader and switch role strategy from Tracer to Leader

The trigger “Walker to Tracer or Leader” tested on n=15 triggers brought just in one occasion to increment the main associated action and in two cases was observed an increment of the collateral actions. Figure 5.10 reports the aforementioned results.

Conclusion

The comparison of the behaviors of users, looking at the actions performed on the system before and after the triggers delivery, showed that no significant role switches were observed. The users rather performed collateral actions related to the main one, but this was not sufficient to complete the switch. This seems to indicate that role-switching is a ceiling for persuasion in Mhikes.

5.5 Discussion

In this chapter we have focused our evaluation of the persuasive features integrated in the Mhikes system. First we have analyzed the usability of the implemented proof-of-concept, successively we have evaluated its technical ability to distinguish roles and then, in terms of persuasion, we have evaluated the effectiveness of role-based persuasive triggers and their effect on role switching.

After having validated the proof-of-concept as usable and understood by users, the evaluation has interested three experiments on the feasibility of detecting the roles in an interactive system, on the impact of persuasive triggers on the users, and on the impact of the role switching strategy on the users.

The experiments confirm that the architecture was capable of introducing a set of software probes to (1) detect the main roles that users are endorsing in the system and (2) to detect and track the evolution of the roles over time. Thanks to these probes the persuasive architecture was able to create a set of personalized triggers. The triggers performed globally better than the communication strategy previously adopted by Mhikes in terms of “clicking” and “opening rate”. The triggers were then analyzed to discover if role- switching was eventually performed by users. Looking at the actions performed on the system before and after the triggers delivery, we observed that the **intra-role-transitions** (transition related to the same role) worked in some cases, while we did not observe any success for the **extra-role-transitions** (transition related to the role switching). This seems to indicate that role-switching is a ceiling for persuasion in Mhikes.

6 Conclusion and perspectives

This work of thesis addresses societal challenges from the perspective of persuasive technology. It tackles the ambitious research problem consisting in filling the gap between the theory of persuasion and its operationalization.

The thesis analyzes all the possible facets of the problem, investigating on the modeling of behavior, on the representation of the process of change, as well as on the conception to arrive to a complete set of methods, models and tools for practitioners in charge of engineering persuasive interactive systems.

The intrinsic complexity of the field and the variety of the arguments explored deserve a recall of the contributions, which open the way to the future research directions that this work can take.

6.1 Recall of the contributions

This research work aims at **bridging the gap** between the huge amount of persuasive models in the literature and their implementation to design persuasive interactive systems.

Tackling this problem is not easy: the investigation brought immediately to face the difficulty of accounting the **variety of these models**, belonging to different and interdependent research areas such as sociology, psychology, software engineering and human computer interaction.

Previous approaches gave partial solutions of the problem: providing behavioral models, providing a set of implementable features, or giving general frameworks to design the systems. However, considering the difficulty of the subject, the persuasive system implemented by non expert in the domain often resulted in a “pick and mix” cocktail of persuasive strategies.

We found in literature the **lack of a complete approach**, taking both the user point of view and the system point of view, and providing concrete guidelines to pass from the modeling of behavioral changes to the operationalization of persuasion.

Our investigation brought to the formulation of **Persuasive Paths** to overcome this gap between theory and practice. Persuasive paths are able to provide:

- The **structuring** capability to represent the process of change,

- The **conceptual** method to pave the path toward the change,
- The **guidelines** to associate persuasive features to the process of change,
- The **architecture** to operationalize persuasion, and
- The **evaluation** tools to verify the effectiveness of the persuasive approach.

Thanks to the **generic formalization** of persuasive paths, they can be used in any domain of application, and also on several domains at the time. This is an important societal aspect since, nowadays **society** relies on different systems to deal with different behavior changes: one application to quit smoking, one for dieting, another to improve the physical activities. The human behavior is sliced in **"watertight compartments"**, such as improving the physical activities would not be influenced by quitting smoking.

Persuasive paths permit to describe the behavior change in its totality, aiming at understanding what are the **behavior determinants** that bring individuals to act and react. Investigating the behavior determinants is ambitious, they may be considered the **grail** of psychology, sociology and human computer interaction. With our research we did not arrive at a complete understanding of these determinants, but we claim to have posed the basis for further investigation.

We claim indeed that persuasive path, thanks to its **persuasive architecture**, may help in characterizing these determinants **observing the reality** (e.g. the performed actions on the physical world or on interactive systems), **modeling the context** (e.g. accounting as much as possible the information on the individual and on the surrounding environment) and **producing new knowledge** (e.g. favorable contexts to drive persuasion) to describe the behavior mechanisms.

The transitions of the persuasive paths are an example of how our investigation can contribute to the characterization of **behavior determinants.** The transitions between the persuasive events (and their associated behaviors) capture the contextual information that actuates the transition. Analysing and aggregating this information, may help in identifying the most frequent factors enacting the transitions, giving new clues on the characterization of the behavior determinant.

In Mhikes for example we have discovered that **extra-roles** transitions (targeting a behavior related to a different role from the actual one) happen less frequently than **intra-roles** transitions (continuing with the actual behavior, staying in the same role). This phenomenon indicates that the **difficulty** of switching between tasks associated to different roles is a behavioral determinant for the Mhike users, which in this case discourages change.

We claim that our investigation may produce further contributions to all the interconnected field of persuasive technologies. For this reason in the next paragraph, we propose different research perspectives.

6.2 Perspectives

We organize the perspectives into short-term and long-term perspectives.

Short-term perspectives are the ones that we have considered for the development of this work but that due to the limited time of this thesis we could not explore further. They may be oriented at performing different types of experiments on the existing instantiation, in order to have a more reliable global picture of what it should or should not further be investigated.

Testing different types of triggers

In our experiments we used the email as delivering channels for the persuasive triggers.

During the thesis it was not possible to introduce the mobile notifications in the Mhikes system, for business choices. However, we have estimated that instancing these notifications would have brought to considerably increase the number of notifiable users and indeed to produce more knowledge on the user. A mixed approach in the communication channel (e.g. email + mobile notification) could also give insights on which is the user preference for triggers and if this preference varies on different application domains.

Extending the experiment time

Persuasive paths are composed of persuasive events and of transitions between them. Due to the limited time to be dedicated to the experiment in the context of the thesis, we could not experiment long paths. Sometimes the data collection could capture just few transitions. Longer experiment could be able to observe ‘repeating patterns’ in the persuasive events of the paths (e.g. user that loops on a set of recurring events). In case this hypothesis would be found to be true, this could become new contextual information on users. The perspective in this case would be to associate particular ‘trajectories’ of paths to particular user profiles based on their context. In this way, in case of lack of information on the user profile, a default persuasive path could be inferred, based on similar users.

In the **long term**, we propose some more general perspectives that if deeper investigated may lead to significant contributions in the state of the art.

Persuasive path as validator of theoretical models

A first general observation regards the reliability of the psychological models. The theory in literature are models conceived to be applied in sociology and psychology. Often developers and designers of persuasive systems take these models as if they were completely reliable, assuming that their principles are supposed to work always and generically.

This is one of the main misunderstandings that leads to an uncontrolled pick and mix cocktail of persuasive features operated by novices in designing persuasive solutions.

Investigation in persuasion should thus consider two big stages: the first based on **understanding** and a second one based on **applying**.

Persuasion field needs to **understand** the reliability of employed models, and technology can help researchers in this task, validating these theories from a conceptual point of view, before thinking at their application to persuade.

Persuasive paths may be used in this perspective to instantiate and test different behavioral models investigating their relationship with the existing models in the theory. The result would be a study on the common determinants that lead individuals to progress in their changes.

The **application** instead concerns the objective of this thesis: finding a strong relationship between modeling the process of change and its operationalization.

Investigating on the behavior determinants at larger scales

We have performed the practical instantiation of the persuasive architecture on Mhikes. An interesting research direction would be of instantiating the paths on multiple domains of application. Concretely we can imagine a persuasive system based on persuasive paths targeting both the challenges of CRegrette and Mhikes. Instead of proposing the user to eat a fruit to avoid smoking, the system could propose him or her to perform an itinerary. This investigation would permit to evaluate the concept of transition between the paths of the two systems and so on behavioral determinants at larger scales.

Modeling the difficulty of transitions as a new behavior determinant

We chose to design our experiments in a way that users had no notion of the persuasive path, they were using the application normally but receiving the persuasive strategies. Another option, not investigated in this work, is to allow the users to visualize their paths and eventually to give them the possibility to directly

alter them, inserting or removing events. This could make them aware of their past activities and persuade them in participating actively to design behavioral processes toward their targeted changes.

During the last experiments we evaluated the switching role strategy. Users were globally positively reacting to the trigger but remained hooked on their actual roles without completely performing the switch. The analysis of these results thus seems to suggest that **extra-role-transitions** were found too difficult for users. This may be due to the fact that switching to a different role comports to learn how to perform the new associated behavior and maybe in Mhikes this was found to be too complex for users.

This result brings to consider a modelization of a “threshold difficulty” of transitions between the events of a persuasive path. At the moment this difficulty is implicitly modelled by intra-role-transitions and extra-role-transitions. A different approach could consider a wider range of values (e.g. from 0 easiest, to 9 most difficult). The definition of this threshold could figure as a parameter for transitions, and could be used to estimate not only the difficulty of a targeted behavior (from a system point of view) but also the level of confidence in progressing over the change (from the user point of view), proposing more challenging transitions to more motivated users and easier transitions to beginners.

Bibliography

Bandura, Albert. 1986. Social Foundations of Thought and Action: A Social Cognitive Theory. Prentice-Hall. https://market.android.com/details?id=book-HJhqAAAAMAAJ.

Bandura, Albert. 1998. “Health Promotion from the Perspective of Social Cognitive Theory.” Psychology & Health. https://doi.org/10.1080/08870449808407422.

Bastien, J. M. Christian, and Dominique L. Scapin. 1993. “Ergonomic Criteria for the Evaluation of Human-Computer Interfaces.” Inria. https://hal.inria.fr/inria-00070012/.

Bijou, S. W. 1957. “Patterns of Reinforcement and Resistance to Extinction in Young Children.” Child Development 28 (1): 47–54. https://www.ncbi.nlm.nih.gov/pubmed/13404656.

Bravata, Dena M., Crystal Smith-Spangler, Vandana Sundaram, Allison L. Gienger, Nancy Lin, Robyn Lewis, Christopher D. Stave, Ingram Olkin, and John R. Sirard. 2007. “Using Pedometers to Increase Physical Activity and Improve Health: A Systematic Review.” JAMA: The Journal of the American Medical Association 298 (19): 2296–2304.

Calvary, Gaëlle, Joëlle Coutaz, Laurent Bouillon, M. Florins, Quentin Limbourg, L. Marucci, Fabio Paterno, et al. 2002. “The CAMELEON Reference Framework.” Deliverable D1 1.

Chomsky, N. 1956. “Three Models for the Description of Language.” IRE Transactions on Information Theory 2 (3): 113–24. https://doi.org/10.1109/TIT.1956.1056813.

Cialdini, Robert B., and Nathalie Garde. 1987. Influence. Vol. 3. A. Michel.

Consolvo, Sunny, David W. McDonald, Tammy Toscos, Mike Y. Chen, Jon Froehlich, Beverly Harrison, Predrag Klasnja, et al. 2008. “Activity Sensing in the Wild: A Field Trial of Ubifit Garden.” In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1797–1806. ACM.

Coutaz, Joëlle, James L. Crowley, Simon Dobson, and David Garlan. 2005. “Context Is Key.” Communications of the ACM. https://doi.org/10.1145/1047671.1047703.

Dey, Anind K. 2001. “Understanding and Using Context.” Personal and Ubiquitous Computing 5 (1): 4–7.

DiClemente, Carlo C., James O. Prochaska, Scott K. Fairhurst, Wayne F. Velicer, Mary M. Velasquez, and Joseph S. Rossi. 1991. “The Process of Smoking Cessation: An Analysis of Precontemplation, Contemplation, and Preparation Stages of Change.” Journal of Consulting and Clinical Psychology 59 (2): 295.

Eyal, Nir. 2014. Hooked: How to Build Habit-Forming Products. Penguin. doi:10.1007/s00779-011-0412-2

Fenicio, Alessandro, and Gaëlle Calvary. 2015. “Persuasion Through an Ambient Device: Proof of Concept and Early Evaluation of CRegrette, a Smoking Cessation System.” In European Conference on Ambient Intelligence, 252–67. Springer.

Fenicio, Alessandro, Gaëlle Calvary, Yann Laurillau, and Jean Vanderdonckt. 2016a. “ProSPer: A MOST Model Extension Applied to Persuasive Interactive System.” In Proceedings of British HCI 2016 Conference (HCI 2016).

Fenicio, Alessandro, Gaëlle Calvary, Yann Laurillau, and Jean Vanderdonckt. 2016b. “ProSPer: Modeling the Change, Driving the Persuasion.” In Actes de La 28ième Conférence Francophone Sur l’Interaction Homme-Machine, 59–69. ACM.

Fogg, B. J. 2002. “Persuasive Technology: Using Computers to Change What We Think and Do (Interactive Technologies).”

Fogg, B. J., and Jason Hreha. 2010. “Behavior Wizard: A Method for Matching Target Behaviors with Solutions.” In Persuasive Technology, 117–31. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-13226-1_13.

Fogg, Brian J. 1998. “Persuasive Computers: Perspectives and Research Directions.” In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 225–32. ACM Press/Addison-Wesley Publishing Co.

Fogg, Brian J. 2009. “A Behavior Model for Persuasive Design.” In Proceedings of the 4th International Conference on Persuasive Technology, 40. ACM.

Fogg, B. J. 2009. “The Behavior Grid: 35 Ways Behavior Can Change.” In *Proceedings of the 4th International Conference on Persuasive Technology*, 42:1–42:5. Persuasive ’09. New York, NY, USA: ACM. https://doi.org/10.1145/1541948.1542001.

Greene, Geoffrey W., Susan R. Rossi, Gabrielle Richards Reed, Cynthia Willey, and James O. Prochaska. 1994. “Stages of Change for Reducing Dietary Fat to 30% of Energy or Less.” Journal of the American Dietetic Association 94 (10): 1105–12.

Haggbloom, Steven J., Renee Warnick, Jason E. Warnick, Vinessa K. Jones, Gary L. Yarbrough, Tenea M. Russell, Chris M. Borecky, et al. 2002. “The 100 Most Eminent Psychologists of the 20th Century.” Review of General Psychology: Journal of Division 1, of the American Psychological Association 6 (2): 139–52. https://doi.org/10.1037/1089-2680.6.2.139.

Hamari, Juho, Jonna Koivisto, and Tuomas Pakkanen. 2014. “Do Persuasive Technologies Persuade? - A Review of Empirical Studies.” In Persuasive Technology, 118–36. Springer International Publishing. https://doi.org/10.1007/978-3-319-07127-5_11.

Harries, Tim, Parisa Eslambolchilar, Chris Stride, Ruth Rettie, and Simon Walton. 2013. “Walking in the Wild--Using an Always-on Smartphone Application to Increase Physical Activity.” In IFIP Conference on Human-Computer Interaction, 19–36. Springer. https://link.springer.com/chapter/10.1007/978-3-642-40498-6_2.

Hughes, J. 2007. “The Ability-Motivation-Opportunity Framework for Behavior Research in IS.” In 2007 40th Annual Hawaii International Conference on System Sciences (HICSS’07), 250a – 250a. https://doi.org/10.1109/HICSS.2007.518.

Jourde, Frédéric, Yann Laurillau, and Laurence Nigay. 2010. “COMM Notation for Specifying Collaborative and Multimodal Interactive Systems.” In Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing Systems, 125–34. ACM.

Kahneman, Daniel. 2011. Thinking, Fast and Slow. Macmillan. http://www.math.chalmers.se/~ulfp/Review/fastslow.pdf.

Landauer, Thomas K. 1988. “Chapter 42 - Research Methods in Human-Computer Interaction.” In Handbook of Human-Computer Interaction, edited by Martin Helander, 905–28. Amsterdam: North-Holland. https://doi.org/10.1016/B978-0-444-70536-5.50047-6.

Lehto, Tuomas, and Harri Oinas-Kukkonen. 2011. “Persuasive Features in Web-Based Alcohol and Smoking Interventions: A Systematic Review of the Literature.” Journal of Medical Internet Research 13 (3): e46.

MacInnis, Deborah J., Christine Moorman, and Bernard J. Jaworski. 1991. “Enhancing and Measuring Consumers’ Motivation, Opportunity, and Ability to Process Brand Information from Ads.” Journal of Marketing 55 (4): 32–53. https://journals.sagepub.com/doi/abs/10.1177/002224299105500403.

Marcus, Bess H., Stephen W. Banspach, R. Craig Lefebvre, Joseph S. Rossi, Richard A. Carleton, and David B. Abrams. 1992. “Using the Stages of Change Model to Increase the Adoption of Physical Activity among Community Participants.” American Journal of Health Promotion: AJHP 6 (6): 424–29.

Marshall, Simon J., and Stuart J. H. Biddle. 2001. “The Transtheoretical Model of Behavior Change: A Meta-Analysis of Applications to Physical Activity and Exercise.” Annals of Behavioral Medicine: A Publication of the Society of Behavioral Medicine 23 (4): 229–46.

Michie, Susan, and Andrew Prestwich. 2010. “Are Interventions Theory-Based? Development of a Theory Coding Scheme.” Health Psychology: Official Journal of the Division of Health Psychology, American Psychological Association 29 (1): 1–8. https://doi.org/10.1037/a0016939.

Michie, Susan, Maartje M. van Stralen, and Robert West. 2011. “The Behaviour Change Wheel: A New Method for Characterising and Designing Behaviour Change Interventions.” Implementation Science: IS 6 (April): 42. https://doi.org/10.1186/1748-5908-6-42.

Milgram, Stanley, and L. van Gasteren. 1974. Das Milgram-Experiment.) Rowohlt.

Miller, W. R., and S. Rollnick. 2002. “Motivational Interviewing: Preparing People for Change Guilford.” New York.

Nakajima, Tatsuo, and Vili Lehdonvirta. 2013. “Designing Motivation Using Persuasive Ambient Mirrors.” Personal and Ubiquitous Computing 17 (1): 107–26. https://doi.org/10.1007/s00779-011-0469-y.

Nunnally, Jum. 1978. “Psychometric Methods.” New York: McGraw-Hill.
Oinas-Kukkonen, Harri. 2010. “Behavior Change Support Systems: A Research Model and Agenda.” In International Conference on Persuasive Technology, 4–14. Springer.

Oinas-Kukkonen, Harri, and Marja Harjumaa. 2008. “A Systematic Framework for Designing and Evaluating Persuasive Systems.” In International Conference on Persuasive Technology, 164–76. Springer.

Oinas-Kukkonen, Harri, and Marja Harjumaa. 2009. “Persuasive Systems Design: Key Issues, Process Model, and System Features.” Communications of the Association for Information Systems 24 (1): 28.

Ölander, Folke, and John ThØgersen. 1995. “Understanding of Consumer Behaviour as a Prerequisite for Environmental Protection.” Journal of Consumer Policy 18 (4): 345–85. https://doi.org/10.1007/BF01024160.

Petty, Richard E., and John T. Cacioppo. 1986. “The Elaboration Likelihood Model of Persuasion.” In Communication and Persuasion: Central and Peripheral Routes to Attitude Change, edited by Richard E. Petty and John T. Cacioppo, 1–24. New York, NY: Springer New York. https://doi.org/10.1007/978-1-4612-4964-1_1.

Pinder, Charlie, Jo Vermeulen, Benjamin R. Cowan, and Russell Beale. 2018. “Digital Behaviour Change Interventions to Break and Form Habits.” ACM Transactions on Computer-Human Interaction: A Publication of the Association for Computing Machinery 25 (3): 15.

Prochaska, James O., and Wayne F. Velicer. 1997. “The Transtheoretical Model of Health Behavior Change.” American Journal of Health Promotion: AJHP 12 (1): 38–48.

Sandhu, Ravi S., Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. 1996. “Role-Based Access Control Models.” Computer 29 (2): 38–47.

Schneider, Hanna, Kilian Moser, Andreas Butz, and Florian Alt. 2016. “Understanding the Mechanics of Persuasive System Design.” Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI ’16. https://doi.org/10.1145/2858036.2858290.

Skinner, B. F. 1963. “Operant Behavior.” The American Psychologist. https://psycnet.apa.org/record/1964-03372-001.

Staddon, J. E. R., and D. T. Cerutti. 2003. “Operant Conditioning.” Annual Review of Psychology 54 (1): 115–44. https://doi.org/10.1146/annurev.psych.54.101601.145124.

Strack, Fritz, and Roland Deutsch. 2004. “Reflective and Impulsive Determinants of Social Behavior.” Personality and Social Psychology Review: An Official Journal of the Society for Personality and Social Psychology, Inc 8 (3): 220–47. https://doi.org/10.1207/s15327957pspr0803_1.

Tanenbaum, Andrew S. 2009. Modern Operating System. Pearson Education, Inc.

West, Robert. 2005. “Time for a Change: Putting the Transtheoretical (Stages of Change) Model to Rest.” Addiction 100 (8): 1036–39. https://doi.org/10.1111/j.1360-0443.2005.01139.x.

West, Robert, and Jamie Brown. 2013. Theory of Addiction. John Wiley & Sons. https://market.android.com/details?id=book-sWtwAAAAQBAJ.

Wood, Robert, and Albert Bandura. 1989. “Social Cognitive Theory of Organizational Management.” AMRO 14 (3): 361–84. https://doi.org/10.5465/amr.1989.4279067.