

Persuasion API

Master's Project

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science in
the Graduate School of The Ohio State University

By

Naveenraj Nagarathinam

Graduate Program in Computer Science and Engineering

The Ohio State University

2015

Master's Examination Committee:

Dr. Rajiv Ramnath, Advisor

Dr. Jayashree Ramanathan, Co-Advisor

Copyright by
Naveenraj Nagarathinam
2015

Abstract

Persuasion – the concept of influencing people towards attitude and behavioral change has slowly made its way into technology. Technology applications today increasingly use strategies of persuasion to influence the users of the application to perform target behavior. These range from commercial applications such as eCommerce looking to sell products better to applications such as fitness that helps bringing lifestyle changes. To perform this in a particular application, this functionality has to be embedded into the system in addition to the core functionality of the application. The idea of this project is to study the design of persuasive systems and build a generic configurable persuasion engine that can be plugged into any such application and used readily. This will allow the developers of the applications to concentrate on building the core functionality of the system. This will also benefit any existing systems built without persuasion in mind to add elements of persuasion to help achieve better results.

In this project, we study existing research on psychology of persuasion, behavioral change and design of persuasive systems to understand the aspects of persuasion and how they can be applied in technology. Then we propose a set of features to build a generic persuasion engine. Then we pick an initial set of features and provide an implementation, thus providing a platform for adding further persuasive elements.

Acknowledgments

I would like to thank my advisor Dr. Rajiv Ramnath for proposing this idea, his time, support and guidance throughout this project. I would like to thank Dr. Jayashree Ramanathan, who agreed to be in the committee and review my Master's project.

I would like to thank the people at Sentosa Technologies, the developers of FitUprising for their support, providing a platform to brainstorm and build our ideas on persuasion and being an early adopter to evaluate the implementation.

Vita

July 2005 to May 2010M.S. Software Engineering, Anna University
July 2008 to Nov 2008.....Software Engineer Intern, Intel Corporation
Nov 2009 to May 2010Software Engineer Intern, Intel Corporation
June 2010 to June 2013Software Engineer, Intel Corporation
Aug 2013 to presentGraduate Student, Department of Computer
Science and Engineering, The Ohio State
University
May 2014 to Aug 2014Software Engineer Intern, Amazon Inc.

Fields of Study

Major Field: Computer Science and Engineering

Table of Contents

Abstract	ii
Acknowledgments	iii
Vita	iv
Fields of Study	iv
Table of Contents	v
List of Figures	viii
Chapter 1: Introduction	9
1.1 Background and Motivation	9
1.2 Approach	10
1.3 Organization of the Report	11
Chapter 2: Persuasion in Technology	12
2.1 Aspects of Persuasion	12
2.2 Ontology of Persuasion	17
2.3 General Decision Process and the Feedback Loop Model	17
2.4 Understanding User Persuadability	18

2.5	Application of persuasion in technology.....	19
Chapter 3: Persuasion API: Background		21
3.1	Improved user involvement.....	21
3.2	Improved content acceptance	21
3.3	Badges and Recognition System	22
Chapter 4: Persuasion API Architecture		25
4.1	Client Parameters	25
4.2	The Rules.....	26
4.3	Rule Actions	27
4.3.1	Email Notifications	27
4.3.2	Social Feed Notifications	27
4.3.3	Badges.....	27
4.3.4	Custom Rule Actions	28
4.4	Rule Processing	29
4.5	Client-API Interaction	30
4.5.1	Forward Communication	31
4.5.2	Backward Communication.....	32
4.6	Persuasion API Architecture	35
4.6.1	System Components.....	35

4.6.2	Data Model.....	37
4.7	Persuasion API Client	37
4.7.1	Service Proxies.....	37
4.7.2	JMS Listeners.....	38
4.8	Technology and Design Decisions	38
4.8.1	Client-API interaction.....	38
4.8.2	Client Data Duplicated in the API	38
4.8.3	Technology	38
4.8.4	Off-the-shelf vs Custom Rules Engine	39
	Chapter 5: Conclusions and Future Work.....	40
5.1	Conclusion.....	40
5.2	Future Work	40
	References.....	42
	Appendix A: Appendix	45

List of Figures

Figure 1: Ontology of Persuasion	17
Figure 2: Cognitive Human Decision Process	17
Figure 3: Feedback Loop Model for Persuasion	18
Figure 4: Feedback Loop for Badges	23
Figure 5: Feedback Loop for User Appreciation	24
Figure 6: Feedback Loop for Public Recognition	24
Figure 7: Triggers for Rule Processing	29
Figure 8: Interaction Model for Client to API Data Flow	31
Figure 9: Interaction Model for User Email Notifications	32
Figure 10: Interaction Model for Badges	33
Figure 11: Interaction Model for User Social Notifications	34
Figure 12: Interaction Model for Custom JMS Notifications	35
Figure 13: Persuasion API System Components	36

Chapter 1: Introduction

1.1 Background and Motivation

Persuasion is a process of trying to convince other people to change their attitudes or behavior [1]. While this might sound complex at first, persuasion can be seen as an attempt of influence to do something that people normally wouldn't or to do more of what they do. Every day, people are subjected to persuasion so often that we easily overlook how they are influenced by outside sources. Think for a moment how many advertisements one encounters on a daily basis. People are being persuaded in every aspect, from the products they buy, the movies they watch to the places they go. Persuasion is a powerful force and has a major influence on society and a whole.

Persuasive technologies are defined as a class of technologies that are intentionally designed to change a person's attitude or behavior [2]. It's getting more exposure as technology has allowed products and services to have an increasingly pervasive role in people's lives [3]. We are surrounded by digital products that are designed to change what we think and do - from websites to mobile applications to smart television applications.

The development of such applications needs to include the persuasive features as an integral part of the application, along with its core functionality. This can be a challenge, given that many people have little or no experience in developing production with a persuasive goal. Though there is a lot of research in the areas of persuasive system design,

it takes a substantial effort from the developer to understand the dynamics behind the persuasive elements, analyze the persuasion context and apply them. Development of such understanding plays an important role in the effectiveness of persuasive systems [4]. In case of an existing application that is built without a persuasive goal, it is even more demanding.

Now, consider the availability of a plug and play component that can bring in persuasive features to these applications. Such a component would be more effective and save lot of effort for the developers and allow them to focus on building the core functionalities of the application.

1.2 Approach

We intend to start by studying the existing literature on psychology of persuasion and persuasive technology to develop a deeper understanding of the persuasion strategies and how those can be applied in technology effectively. Then we identify the persuasive techniques that are effective in technological intervention and research the existing systems for application of those techniques. This will help understand the use and effectiveness of these techniques. Then we pick up the initial set of most influential strategies and implement them in a generic API that can cater different kinds of applications. Then integrate the API with a consumer and measure the actual effectiveness of the strategies implemented.

1.3 Organization of the Report

Chapter 2 gives a summary of our understanding of the strategies that applies to persuasion in technology from studying existing research in persuasion and persuasive technology. This chapter also explains how individuals differ in persuadability and what a system can do to tailor their attempts of influence to individuals and application of persuasion in existing applications of relevance. Chapter 3 explains the concept of Persuasion API and what we plan to implement from our understanding of persuasion and how they can be effective. Chapter 4 explains the architecture of Persuasion API, the technology and the implementation details of various components. Conclusions and ideas for future work are presented in Chapter 5.

Chapter 2: Persuasion in Technology

Influencing others isn't luck or magic – its science [5]. There are proven strategies to help make you more successful as a marketer and an office politician. Similarly, one can be successful in persuading through technology with a clear understanding of such strategies. There's a lot of academic research around the areas of persuasive technology and design for behavior change. Existing literature on persuasion have been explored to understand the aspects of persuasion in general and how those apply to technology. This chapter presents our understandings from this process.

2.1 Aspects of Persuasion

The Persuasive System Design (PSD) model [6] is a major contributor for our understanding of design and development of persuasive systems. This, along with the study of previous research [5] [7] [8] [4] [9][add reference numbers here] on persuasive technology leads us to a fine understanding of the persuasion strategies that can applied on technology and the approach of developing a persuasive system. The following section discusses such strategies.

Every system that wishes to attract user base needs to offer something of importance to the user. These are general aspects of the system that influences people to use the system.

Reduction: People incline towards applications that break up their target goal into simpler incremental tasks.

Tracking & feedback: People always like to track and hear how they perform. People consistently see positive performance and providing feedback for bad performance creates a dissonance and motivates them to perform better.

E.g. a fitness program showing user's current progress towards their goal, and appreciating or motivating to perform better.

Facilitation: People like a platform to collaborate.

E.g. a weight loss management system facilitates people living nearby for group fitness. Or a forum in an eCommerce site to discuss on products.

Personal Goal Setting: People like to have control over what they do and like to do at their own pace (though some motivation can improve this).

Predictions/rehearsal: Visually seeing their path to goal has its own impact.

E.g. a fitness program showing their 5 weeks weight loss based on their current status and performance. Or an eCommerce site showing complete shipment tracking.

Tunneling: Long term attitude change is always incremental. People expect the system to guide them step by step to get them closer to their goal.

While the features of a system can help to get the users to start using the system, it needs more effort from the system to retain them, keep them engaged and motivate them to perform their (or the application's) target behavior. Some of the direct and indirect persuasion strategies that help achieve this are discussed below.

Appreciation: People like to be appreciated. It has a feel good factor and they like to consistently receive it.

E.g. appreciating for being under the calorie goal for the day motivates them to do it daily.

Rewards: Rewarding people makes them feel accomplished and motivates them to progress further. In some cases, this also improves their self-efficacy.

E.g. providing them a virtual trophy or a badge for achieving the week's weight loss goal.

Reminders: Unless people are determined, they tend to forget their goal in every step. [10]

E.g. as soon as a person reaches home, a reminder to take the stairs instead of elevator will make them think of taking the stairs, which they won't do normally.

In the era of information overflow, people often use heuristics to process the information they see, because of the abundance of information to be handled. When an individual sees relevant cues, heuristics are triggered [6] and it motivates them to process the related content. Systems are more likely to make users accept their content by presenting them with relevant cues or stereotypes for evaluating the information. Some strategies of such indirect persuasion are discussed below.

Consensus/Social Proof: People tend to accept and do what most other people accept and do. This relies on people's sense of 'safety in numbers'.

E.g. a diet routine with a message '90% people said this helped them achieve their goal' next to it has more chances of being followed.

Liking: People tend to be influenced by people they like. This can be their friends, some public figure they like or just someone they trust.

Expertise & Authority: Expert opinions and opinions from people in authority are more likely to have an impact.

E.g. a dietary routine with additional information that is suggested by so and so expert is more likely to be noticed.

Attitude of content: There are differences in how people perceive positive and negative messages/enforcement. Positive enforcement is more likely to have effect.

E.g. a health tip of advantages of having a fruit snack (may be with advantages over fried snack) has more impact than a tip of disadvantages of fried snack.

Simulation: Content which shows the cause-effect relationship is more likely to have an impact.

E.g. a diet-exercise routine is more likely to be followed with an additional information that it helps reduce 5lb in 3 weeks.

Tailoring & Personalization: People pay attention to content (only) those that are relevant to their interests and more to those that are relevant at that point of time.

E.g. a fitness website suggesting exercise routine according to the current physical activity level of user. Or in a pet care website, if a person is into dogs, a cat care is more likely to be ignored.

Our social relationships are increasingly maintained through technology-mediated communications [4]. This shift in the ecosystem comes in the advantage to persuasive technology. Systems can leverage this social influence to persuade users to steer them towards the target behavior.

Public recognition: Everyone likes to be recognized. Again, this attacks the feed good factor of the person. A system should provide public recognition for users who perform the target behavior.

E.g. displaying the virtual trophy they received next to their profile and showing their current weight loss next to it.

Social comparison: People like to compare their performance with others and this increases their motivation to perform to be better than others.

E.g. frequently publishing their peer's weight loss progress a user's feed.

Social influence: People tend to change to meet the expectations of others. Peer pressure and motivation helps them perform better.

E.g. supporters in a weight loss program can motivate a person to perform better.

Though most of the examples here are related to fitness applications, these are some of the general strategies, most of which can be applied to any application aiming to achieve a certain user behavior. For example, an eCommerce application displaying related products when a user is trying to purchase a product has more chances of selling the related product as well(Tailoring), or attaching a message '70% of the users(who view the product) buy this' next to product description may influence the person to consider buying the product(Consensus). Every application has different methods for using a certain persuasion technique and the parameters used to identify if a certain technique can be applied, varies.

2.2 Ontology of Persuasion

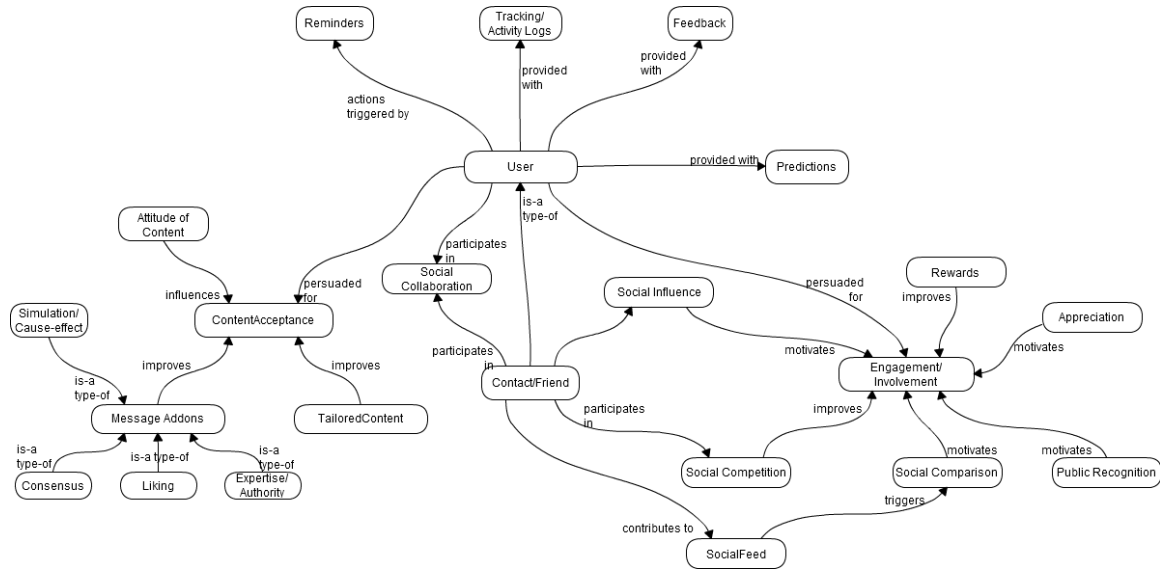


Figure 1: Ontology of Persuasion

2.3 General Decision Process and the Feedback Loop Model

Every human action is a result of some other action, either by the same person, or induced by someone. The outcome of the previous action is a feedback to the person which induces the successive action [11]. This action that generates the outcome can be of multiple stages. In software terms, the system is the one inducing the action based on the data collected from previous user actions. The system collects data, which is turned into information, which tells a story, which in turn, influences how the person makes decisions and acts on the knowledge [12].



Figure 2: Cognitive Human Decision Process

This general decision process can be extended to form a feedback loop to induce further user actions. The person makes a decision which leads to performing some actions. The system can capture this action and use it as data to create meaningful visualizations in order to induce further actions, and the loop continues.

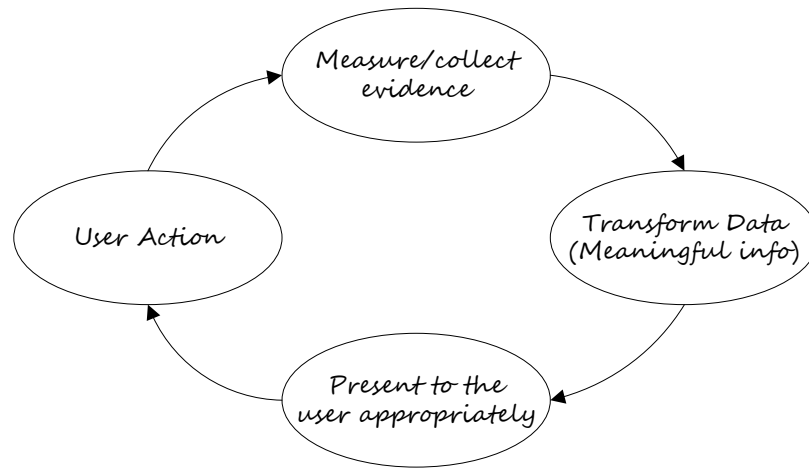


Figure 3: Feedback Loop Model for Persuasion

2.4 Understanding User Persuadability

Every user is different and needs different levels of cognition to be successfully persuaded [13]. A persuasive strategy that works well on a user need not necessarily work well on another user. Human persuaders often tailor their influence attempts to their audiences and hence they are usually successful in persuading. Hence, static application of persuasion is not likely to produce the desired result in a persuasive application. To understand the persuadability of the user and the persuasion strategy the user is more susceptible to, the best way for a persuasive system is to learn from the responsiveness of the user to the

system's attempts at persuasion. It can use this information to build a persuasion profile for the user indicating the susceptibility of the user to various persuasion strategies.

However, when a new user starts using the system, the system starts with a blank slate with no profile information for the user. One way to build the user's profile is to perform a standard A/B testing. During the course, the user can be attempted to be influenced by persuasion strategies selected randomly and the profile can be build based on the user's responsiveness. A better way would be to do some initial user profiling by presenting the user with a questionnaire with questions that help measure their susceptibility to various persuasion strategies [14] [15]. For example, if the user strongly agrees with 'If someone from my social network notifies me of a good book, I tend to read it', then the user is more likely to be influenced by consensus. If the user strongly disagrees with 'When a professor tells me something, I tend to believe it is true', then the user is not likely to be influenced by authority.

Such a questionnaire can be provided to the users, for example, at the time of registration to the website and the initial user profile be built by calculating a persuasion score for each strategy based on the user's response. The users can then be attempted to be persuaded with the most influential strategy from the user profile and the system can continuously learn and adapt based on the user's responsiveness.

2.5 Application of persuasion in technology

Application of persuasion in technology, particularly in weight loss applications, has been studied to understand the effect of various persuasion strategies and pick the most

influential strategies to be implemented in Persuasion API. A matrix denoting the websites and the usage of different persuasion strategies was formed. Though there are some strategies that are more commonly used by most of the websites, they may not necessarily be influential. These persuasive features have been built into those applications and there is no clear data as to how implementation of a particular strategy has helped influencing the users. Some studies [16] analyzing the weight loss websites have attempted find similar details and have found no success in such attempts.

The matrix of weight loss websites and the persuasion strategies used by them can be found in [Appendix A](#).

Chapter 3: Persuasion API: Background

From the study of previous research in persuasive technology, we intend to achieve two major category of things with Persuasion API – better user involvement and improved content acceptance.

3.1 Improved user involvement

Appreciation and recognition can be quite influential when it comes to persuasion. Sometimes, people do what they do for recognition, to stand out in the crowd, or for themselves [17]. They do it because it improves their level of self-worth. We aim to improve user involvement in the system by leveraging these aspects of persuasion. Section 3.3 explains as to how we intend to do this and how they can be influential.

3.2 Improved content acceptance

Peripheral cues are known and proven to have an impact in the way people respond to information content [18]. These cues are be additional information that help us perceive the importance of the content. For example, in advertising, although the informational content of the advertisement may be the most important determinant of a product, the credibility of the product endorsers or the celebrity status may be influential to get the product noticed to make the person process further information.

In Persuasion API, we intend to improve content acceptance by attaching one line add-on messages to the displayed content as peripheral cues. For example, a message next to an advertised product that says ‘#1 best seller in this category’ or a message next to a recommended weight loss routine that says ‘98% of the users found this helpful’ helps influence the users to notice and process more information of the content. We are proposing to use consensus, liking, authority and scarcity as strategies of framing these add-on messages. See section 2.1 for details of these strategies.

We also propose building individual profiles for users of the system in the API to guide with most influential strategies that work on them. Initial profiling can be done using the method of user questionnaire discussed in section 2.4. The system can then present content to the user with appropriate add-on messages using the strategies suggested by the user profile and continue to learn and adjust the profile from the user’s responsiveness to such interventions.

List of possible questions related to the strategies planned to be implemented in Persuasion API can be found in [Appendix A](#).

3.3 Badges and Recognition System

Our first focus is on improving user involvement by recognizing and rewarding the users. And this is what the current implementation accomplishes with the badges and recognition system.

The system captures the user's actions and do one or more of the following as an appreciation of performing the action. These actions are based on rules configured in the Persuasion API, details of which are explained in Chapter 4.

Award the user a badge

Badges are awarded to users in recognition for achieving something substantial. This is one of the persuasive system's way of showing appreciation to the user and it can also be used to build the user's public profile and recognize them in their social network.

Rewarding users with badges creates a positive feedback loop by inducing the user to perform more of the behavior to obtain the next level badge, either to be proud of oneself or to get more public recognition.

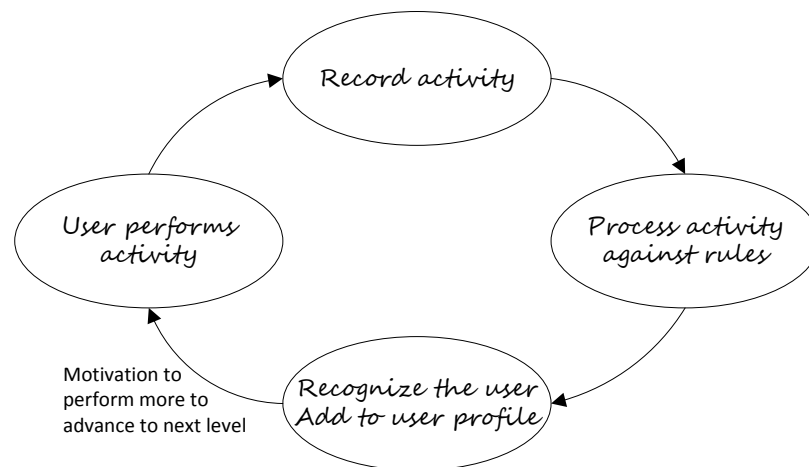


Figure 4: Feedback Loop for Badges

Send user an appreciation

Appreciation messages are sent to the users in recognition of some activity performed by the user. This affects the feel-good factor of the user and creates a positive feedback loop by inducing the user to perform the same action to keep this coming.

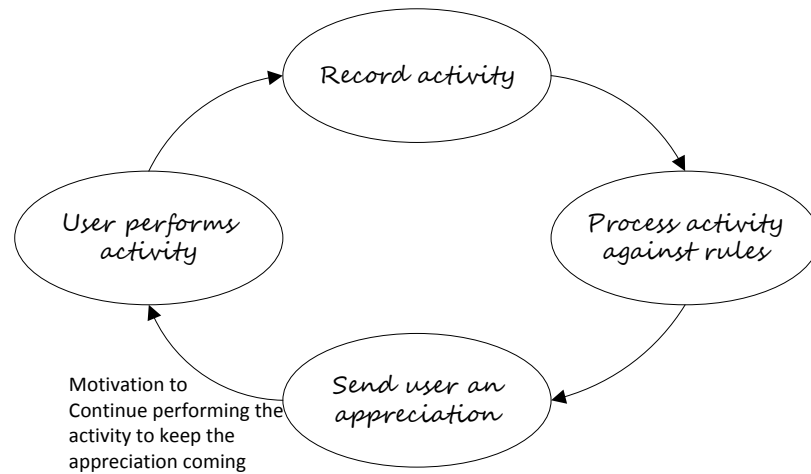


Figure 5: Feedback Loop for User Appreciation

Publicly recognize the user

The user's achievements are posted in the user's social network. This improves public recognition of the user and creates a positive feedback loop to keep performing to get noticed and stand out in the crowd.

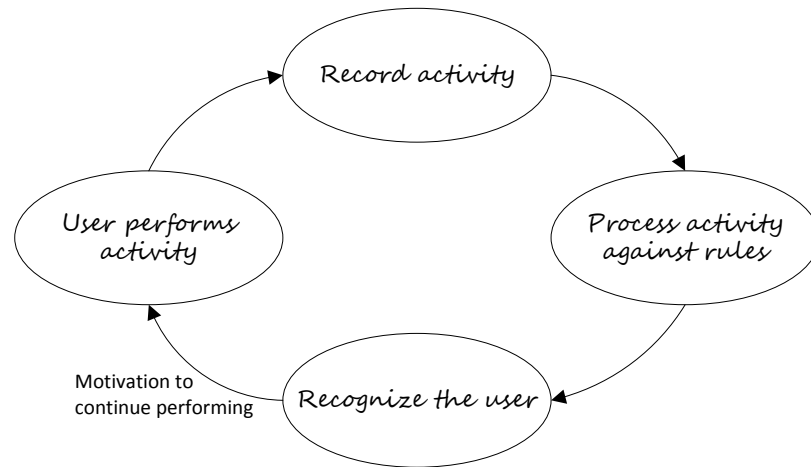


Figure 6: Feedback Loop for Public Recognition

The next chapter provides the technical details of how these are implemented.

Chapter 4: Persuasion API Architecture

The Badges and Recognition System is a rule based system that captures user activity and processes them against the configured rules and performs actions based on the outcome of the rules.

4.1 Client Parameters

There are three major parameters that are obtained from the client system that form the basis of the Badges and Recognition System – the user, user activity and user attribute.

User is the user of the client system who is persuaded to perform the target action. User Attribute refers to a characteristic of the user that can hold a certain value. For example, the type of a user or the user's location.

User Activity is an action of relevance performed by the user. For example, the user of a travel website has written a review for a hotel or the user of a fitness website has recorded a completed workout. Two attributes related to the user activity is captured in the API – number of times the user has performed the activity so far and an optional value related to the activity. For example, the optional value for recording a workout could be the number of calories burnt.

4.2 The Rules

Rules are the integral part of the Badges and Recognition system. These are typical hierarchical rules, with each rule containing one or more conditions that need to pass in order for the rule to succeed. These conditions can be one of user attribute value, user activity value or the number of times the user has performed an activity compared to a rule specific value.

The rule specific comparison value can be a static value. For example, in a travel review website, you intend to send the user an appreciation when the user has posted 50 reviews. The rule for this can be that if *review_posted*<count_greater_than>50, send user an appreciation. Here, *review_posted* is activity and we are comparing the number of times the user has performed the activity of posting a review.

The comparison value also be one of activity count, activity value or attribute values. This can be configured using the notations @@*activity_name*#count, @@*activity_name*#value and @@*attribute_name* respectively. For example, to configure a rule in a fitness application to check if the user has exceeded exercise goal for the day, the rule can be *daily_workout_completed*<value_greater_than>@@*daily_workout_goal*. Here, the activity is *daily_workout_completed*. We are checking the value associated with the activity to the user attribute *daily_workout_goal*.

Each rule has a configurable set of actions to be performed if the rule passes.

4.3 Rule Actions

There are four possible actions that can be performed – the user can be assigned a particular badge, the user be sent an email notification, a notification be posted to the user’s network’s social feed and/or it can be posted to one or more JMS queues for performing a custom action.

4.3.1 *Email Notifications*

Every rule when passed can be configured to send the user an email notification – for example, an appreciation email message when the user of a fitness website has recorded his food intake and has stayed under his calorie goal for the day. A plain text email subject and a HTML email body can be configured to be sent. Both email subject and email body can use any values of user activity count, user activity value or user attribute value in the email subject or email body. This can be configured using the notations @@activity_name#count, @@activity_name#value and @@attribute_name respectively.

4.3.2 *Social Feed Notifications*

Every rule when passed can be configured for a social feed notification to be posted to the user’s social network – for example, the user of a fitness website has lost 10lbs since her last weigh in. A plain text social notification can be configured to be posted. These notifications can use similar notations as the email configuration to use values of user activity count, user activity value or user attribute value in their body.

4.3.3 *Badges*

Different kinds of badges can be configured – for example, a seller in an eCommerce website can have different ratings for trustworthiness, prompt shipments and product

quality. We call these *badge classes* in Persuasion API. And there can be different *levels* of badges in each class. These levels are the actuals badges that the user will be awarded – for example, the number of stars the seller gets for prompt shipments. Every user in the system can be awarded one or more badges, but limited to one per badge class. The reason for this is logical, i.e. one can carry only one rating for trustworthiness, but can have another rating for product quality.

Each badge can be also be configured an optional badge image, which the client system can use it as required, for example the crown attached to the user’s profile image based on his all-time top score in an online game. Also, each badge can be configured an optional email message to be sent to the users and an optional social notification to be posted to the user’s social network when the user is awarded a badge. Similar notations using @@ can be used to consume values of user activity count, activity value or attribute value in the email or social notifications.

These badge related email and social notifications are in addition to those that can be configured for the rule actions. And hence one must be careful in configuring both assignment of badges and posting notifications as actions for the same rule so as to avoid duplicate communications.

4.3.4 Custom Rule Actions

There can be cases where the client system needs to perform some action other than assigning the user a badge, sending emails or posting social notifications. For example, there is a seasonal promotion in an online fashion store and they would like to award the customer a 10% discount for spending \$100 or more. These voucher codes are generated

on demand and cannot be configured in a static email message. To cater these, rule actions can be configured to post notifications to JMS queues, which can be captured by the client system and perform the required action. The names of these JMS queues can be configured on the fly and the notifications can be posted to more than one JMS queues, in case multiple different custom operations needs to be performed if a particular rule passes.

4.4 Rule Processing

Every user action that is of relevance to rule processing is reported by the client system to the API as and when it happens. For example, the user of a weight loss application has recorded a weigh-in and has lost 10 pounds since his last weigh-in. The client system also reports any user attribute value changes to the API. The API then processes the corresponding user's activity/attribute values in the background against the configured rules and performs appropriate action, say, send the user an appreciation email and so on.

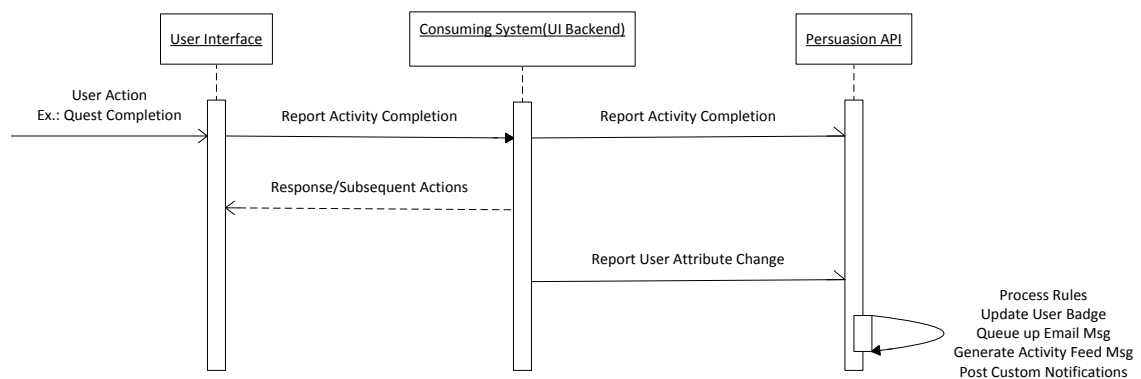


Figure 7: Triggers for Rule Processing

All the rules are processed for every user irrespective of the activity or attribute that is updated. This is to ensure consistency despite the intricacy of rule configurations possible. Recall that the rules are hierarchical and that they can contain parent-child relationships. Rule Processing begins with the top level rules. Once a top level rule is processed and if it passes, the processing continues to the next level in the hierarchy. If a parent rule fails, none of the child rules are processed. Depending on the outcome of every rule, the rule actions are processed. Badges assignments are stacked and processed at the end. This is to ensure only highest level of badge in each badge class is assigned to the user.

The 'Notify Always' configuration

Email, social and custom JMS notifications are processed based on the outcome of the child rules. If the current rule is the lowest in the hierarchy to pass, then the notifications of the current rule are processed and sent. If any of the child rule passes, it means that finer rules succeeds and it is logical to process those notifications instead. None of the notifications of the current rule are processed. *<add some examples here>*. However, this can be overridden by the *Notify Always* configuration associated to each rule. By turning on this configuration for a rule, it makes the rule processor to post the configured notifications irrespective of the outcome of the child rules.

4.5 Client-API Interaction

Given the fact that this API is generic and the consumers may use different technical platforms, it is an architectural decision that all interactions be initiated from the client and

communicated through platform independent protocols like RESTful services and Apache ActiveMQ based JMS messages.

4.5.1 Forward Communication

The client communicates with the API through the REST based services provided by the API. The client uses these services to report user activity and user attributes. These are synchronous services that only record the information provided by the client. Though the clients are not to expect any information from the service in return, the activity or attribute information needs to be successfully recorded in the API in order for the call to succeed.

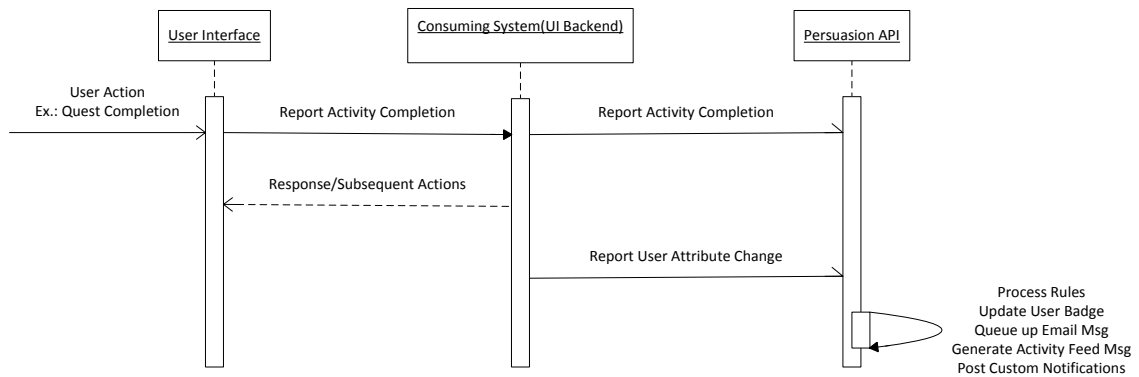


Figure 8: Interaction Model for Client to API Data Flow

This information recorded by the services are then picked up by the asynchronous processor and processed in the background against the configured rules and appropriate actions are performed. This information is communicated to the clients through the backward communication channel.

4.5.2 Backward Communication

As a result of processing by the background rules engine, there can be badges assigned to the user or notifications to be sent out. But there is direct communication channel for the API to invoke the client to pass this information on. Most such information are queued up and it is left to the consumer to be picked up and processed at its own pace.

Email Notifications

Email notifications are something that are supposed to be sent to the user. But the Persuasion API does not store any user information other than the user name such as email address or the communication preferences set by the user in the client system. Hence the API does not make any direct communication with the user. Instead, the email messages are queued up with formatted subject and body and are left to be picked up by the client and sent to the user accordingly. Figure 9 explains this.

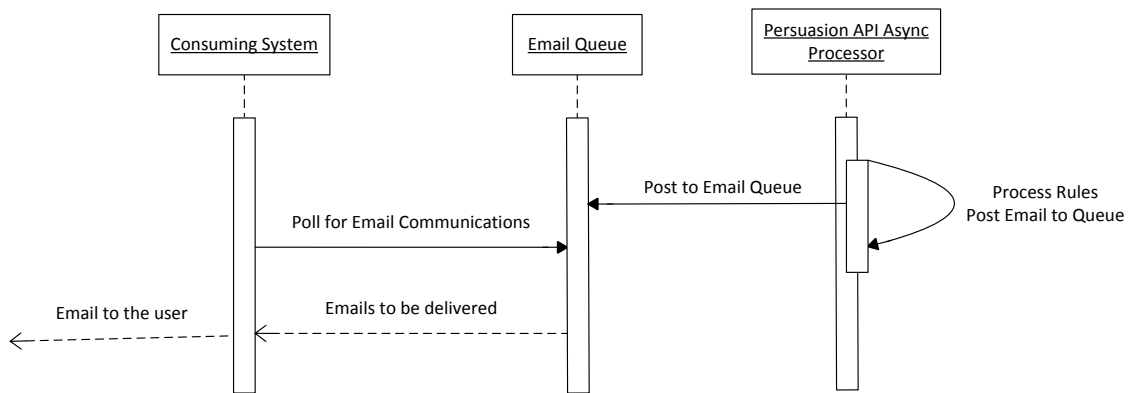


Figure 9: Interaction Model for User Email Notifications

Badges

Badges can be assigned to the user as a result of some user action based on the rules. As opposed to email notifications, badges information stay in the API and are not passed on to the client. The client calls the API services to retrieve information about badges for a particular user as and when required.

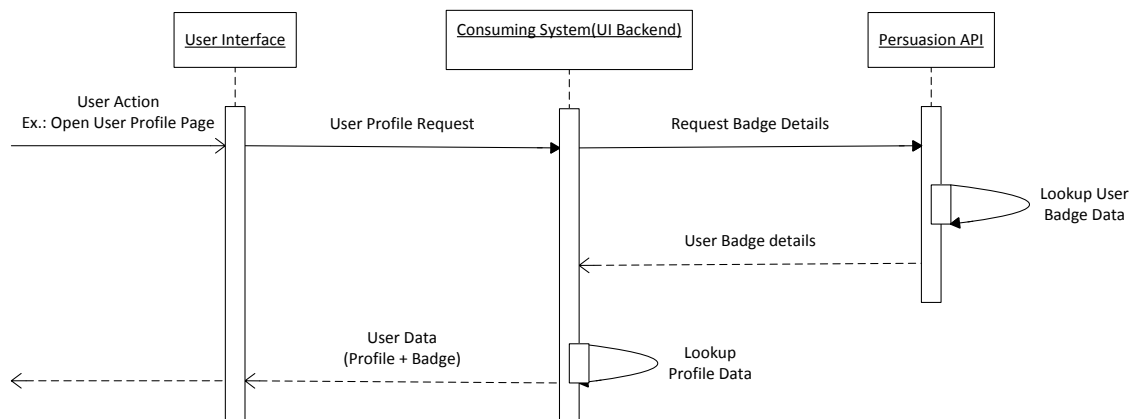


Figure 10: Interaction Model for Badges

For example, the user's profile is accessed and the client logic displays the badge next to the user's profile. In this case, the client makes a call to the API before displaying the user profile page to retrieve the badge information. This is depicted in Figure 10.

Social Notifications

Similar to badges, social notifications are to be pulled from the API using the provided services. Social notifications are a bit tricky than email notifications or badges, which correspond to a particular user. Social notifications, on the other hand, are usually displayed to a user's network. Whereas the notifications generated the API pertain to a particular user. And the API does not store any information of the user's social network.

And hence to pull the notifications to be displayed to a particular user, the client is expected to send the list of IDs of users in the network for the API to pull the associated social updates.

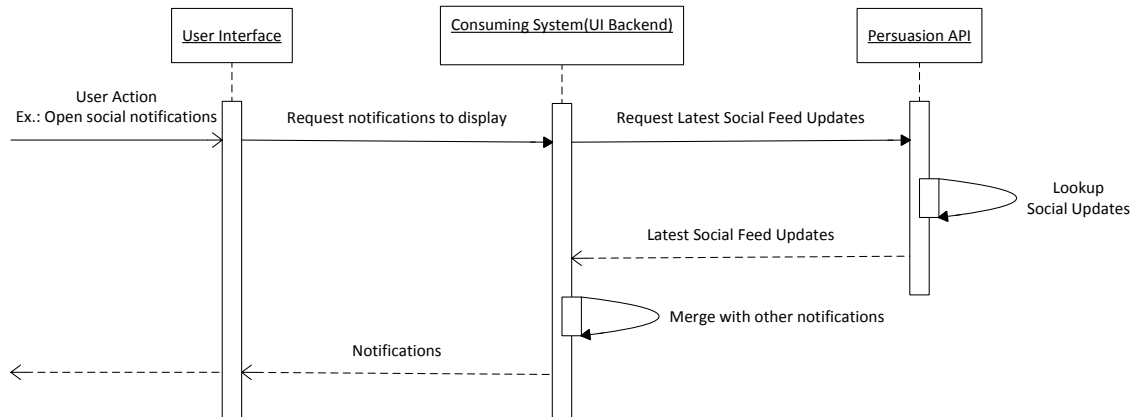


Figure 11: Interaction Model for User Social Notifications

Figure 11 shows a user action where she opens the notification panel in the client system user interface and the sequence of calls that happen to provide the user with the notifications.

Custom JMS Notifications

JMS notifications for rules helps the client perform a custom action. These messages are posted to the specified queues and are left to the client to be picked up and processed.

Figure 12 depicts the flow of calls to process a custom notification.

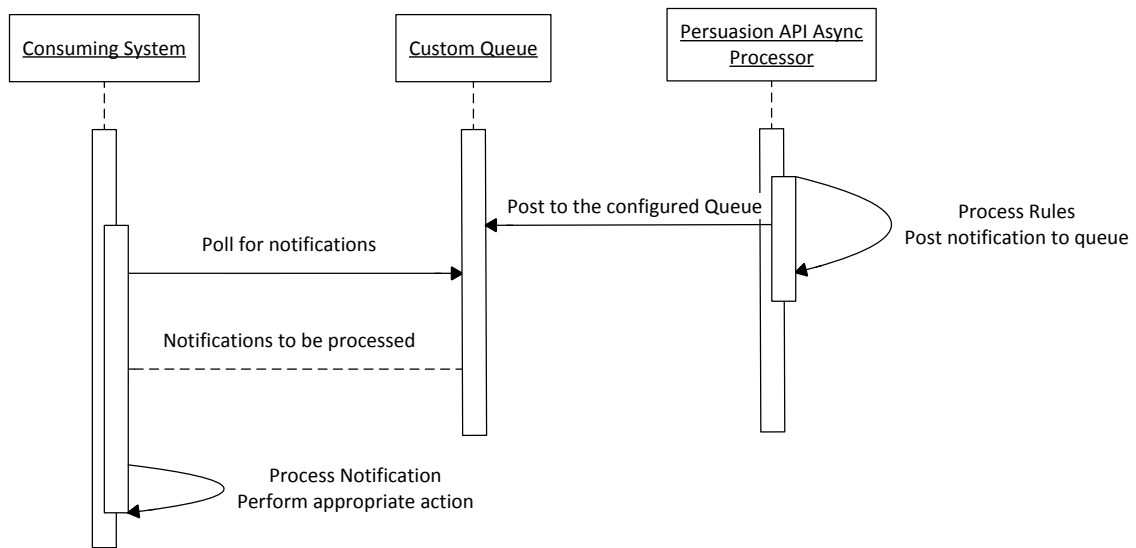


Figure 12: Interaction Model for Custom JMS Notifications

The JMS messages carries along with them, all the activity and attribute information used to process the rule that posts the notifications. These data can be used in the client's implementation for further processing.

4.6 Persuasion API Architecture

4.6.1 System Components

Persuasion API is divided into different components that perform specific functionalities.

Figure 13 shows the different components and how they interact.

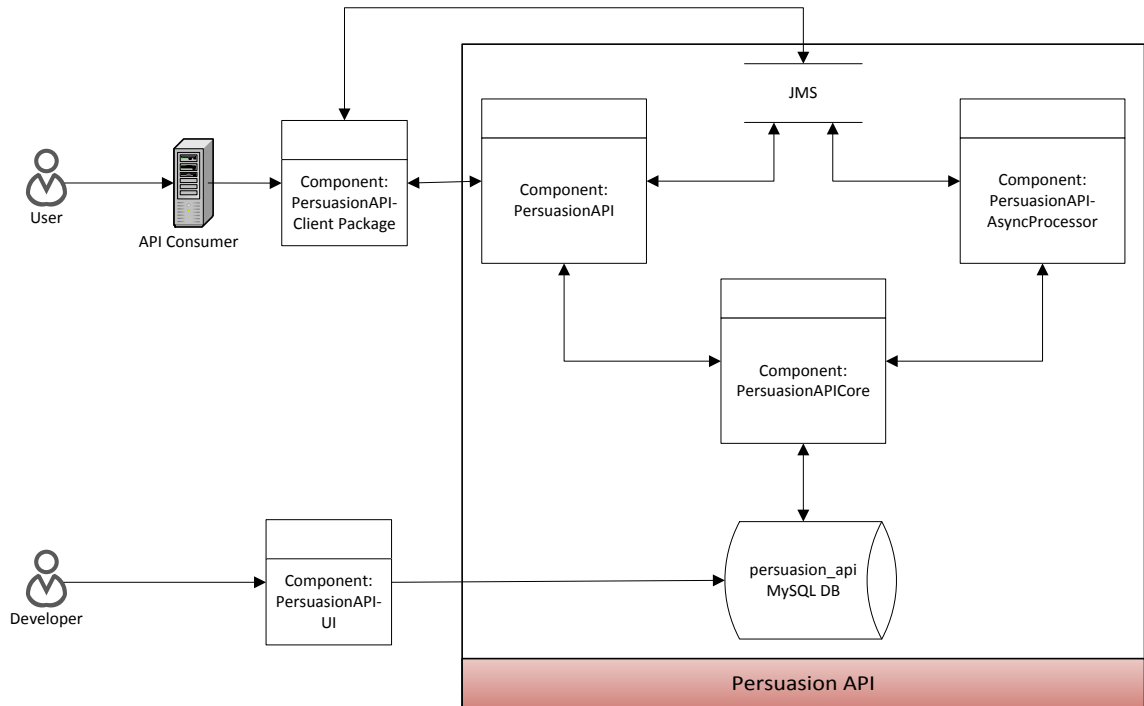


Figure 13: Persuasion API System Components

PersuasionAPI is the service layer containing the web services that cater both the client systems and the configuration UI. These services are REpresentational State Transfer (REST) based POST services using JSON for data interchange. General request and response formats, service contracts and error handling details can be found in [Appendix A](#).

PersuasionAPICore is the core component of Persuasion API. This contains the core logic, operations, interfaces to database and JMS components.

PersuasionAPIAsyncProcessor is the asynchronous processing layer of the system that contains the rules engine. It listens for the user activity and attribute updates performed by the clients in the service layer and triggers rule processing.

PersuasionAPIClient is a Java based client package that helps Java based consuming systems to easily integrate with the API. This component is explained in detail in next section.

4.6.2 *Data Model*

<Insert diagram and provide explanation>

Detailed information of the tables and fields, code structure and the organization of classes can be found in [Appendix A](#).

4.7 Persuasion API Client

Persuasion API Client package contains proxies that can be used to call the API services and interfaces that helps the clients listen to JMS messages from the API and provide custom implementations.

The client package needs to be initialized before it is utilized to call services or listen to JMS messages. The consumers are expected to provide the context root to the URL where the API services are installed and the address to the host where the Apache ActiveMQ JMS service is installed that handles the related queues.

4.7.1 *Service Proxies*

The client contains proxies to all the API services consumed by the client system. The proxy methods accept defined request and response types. The URL to call the services are built using the context root initialized by the client and the path information that comes

along with the proxy. JSON conversion to and from the services are handled by the proxies and a custom exception is thrown in case a failure response is returned by the service. These request, response and exception type comes defined in the package itself.

4.7.2 JMS Listeners

The client package also provides JMS listener interfaces. Consumers are expected to provide implementations to a defined abstract interface. These implementations can be registered to be listened to a JMS queue. Queue names are to be provided at the time of listener registration and the server details are picked up from the details provided during the initialization phase. Listener registrations are limited to one per queue. An attempt to register more than one listener will fail. Listeners can be deregistered, or temporarily paused from listening to a queue and started again later.

4.8 Technology and Design Decisions

Some of the major decisions were to decide the type of interactions between the API and the consuming system, the data relating to the consuming system that need to be duplicated and stored in the API, and the technology used to develop the API.

4.8.1 Client-API interaction

<Add content>

4.8.2 Client Data Duplicated in the API

<Add content>

4.8.3 Technology

J2EE Spring was chosen for the implementation framework and MySQL was chosen as the database platform due to the features<explain differently?> and prior expertise. RESTful services with JSON for passing data were used for implementing services that will be consumed by the client. Java Message Service (JMS) is used for backward communication from API to the client. JMS is also used for communication between different components of the Persuasion API i.e. from the service layer to the asynchronous processing layer.

4.8.4 Off-the-shelf vs Custom Rules Engine

<Add content>

Chapter 5: Conclusions and Future Work

5.1 Conclusion

With this project, we have not intended to perform research on any new methods of persuasion. There is already a vast amount of great work in the field of persuasive technology. We have attempted to understand the dynamics of persuasion from the available literature and create a platform that can be useful for a wider audience. This can help incorporate elements of persuasion in various new and existing applications to get more value out of their system without substantial efforts from the developers. This also helps the customers to better achieve the purpose they intended to use the system. And I would consider that we have succeeded in our attempt to create one with a limited set of features, providing a platform to add more persuasive elements and make this API more useful.

5.2 Future Work

With the current status of implementation, we see a good amount of work that can help extend the feature and make the application better. A user interface for creating and maintaining the configuration is being worked on. Research on user questionnaire and building user profile can be an immediate next which lays a foundation for implementing message add-ons and further learning to keep the user profile up-to-date.

More importantly, integrating this API with potential applications and measuring the impact of persuasion will help contribute to further research to understand the effectiveness of various persuasion strategies.

References

- [1] R. M. Perloff, *The Dynamics of Persuasion: Communication and Attitudes in the 21st Century*, 2nd ed., Lawrence Erlbaum Associates, Publisher, Mahwah, New Jersey, 2003.
- [2] B. Fogg, "Persuasive technologies: Introduction," in *Communications of the ACM archive. Volume 42 , Issue 5*, 1999.
- [3] B. Fogg, "Creating Persuasive Technologies: An Eight-Step Design Process," in *Persuasive '09*, Claremont, California, USA, 2009.
- [4] T. Lehto, "The Importance of Persuasive Systems Design in Enhancing Consumers' Perceptions and Adoption of Health Behavior Change Support Systems," 2013.
- [5] R. B. Cialdini, *Influence: The Psychology Of Persuasion*, Harper Business, 2006.
- [6] H. Oinas-Kukkonen and M. Harjumaa, "Persuasive Systems Design: Key Issues, Process Model, and System Features," *Communications of the Association for Information Systems*, 2009.
- [7] A. Stibe and H. Oinas-Kukkonen, "Designing Persuasive Systems For User Engagement In Collaborative Interaction," in *Twenty Second European Conference on Information Systems*, 2014.

- [8] D. Yoganathan and S. Kajanan, "Persuasive Technology for Smartphone Fitness Apps," 2013.
- [9] K. Torning and H. Oinas-Kukkonen, "Persuasive system design: State of the art and future," in *4th International Conference on Persuasive Technology*, ACM New York, NY, USA, 2009.
- [10] M. Maheshwari, S. Chatterjee and D. Drew, "Exploring the Persuasiveness of “Just-in-time” Motivational Messages for Obesity Management," in *Persuasive '08*, 2008.
- [11] Website Editor, "Feedback loops Design Pattern," 2003. [Online]. Available: <http://ui-patterns.com/patterns/Feedback-loops>.
- [12] C. Risdon, "Behavior Change as Value Proposition," 2012. [Online]. Available: <http://www.adaptivepath.com/ideas/behavior-change-as-value-proposition>.
- [13] M. Kaptein and D. Eckles, "Heterogeneity in the Effects of Online Persuasion," *Journal of Interactive Marketing*, 2012.
- [14] M. Kaptein, P. Markopoulos, B. d. Ruyter and E. Aarts, "Can you be persuaded? Individual differences in susceptibility to persuasion," 2010.
- [15] M. Kaptein, J. Lacroix and P. Saini, "Individual Differences in Persuadability in the Health Promotion Domain," 2010.
- [16] T. Lehto and H. Oinas-Kukkonen, "Persuasive Features in Six Weight Loss Websites: A Qualitative Evaluation," in *Persuasive Technology, 5th International Conference*, 2010, pp. 162-173.

- [17] Aoife, "Persuasion is about Appreciation," 2015. [Online]. Available:
<http://zerocamouflage.com/persuasion-is-about-appreciation>.
- [18] P. W. Miniard, P. R. Dickson and K. R. Lord, "Some Central and Peripheral Thoughts on Routes to Persuasion," in *Advances in Consumer Research Volume 15*, 1988, pp. 204-209.

Appendix A: Appendix

<Add content>