FAKULTÄT FÜR INFORMATIK

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Implementation and Evaluation of a Persuasive Mobile Food Recommendation System

Muhammad Kabir Khan

FAKULTÄT FÜR INFORMATIK

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Implementation and Evaluation of a Persuasive Mobile Food Recommendation System

TODO: Titel der Abschlussarbeit

Author:	Muhammad Kabir Khan
Supervisor:	Prof. Johann Schlichter, Ph.D.
Advisor:	M.Sc. Bëatrice Lamche
Submission Date:	TODO: Submission date

I assure the single handed composition of this master's thesis in informatics only supported by declared resources.

Munich, TODO: Submission date Muhammad Kabir Khan

Acknowledgments

I would like to thank to my advisor Beatrice Lamche, who always give me her valuable advice and her guide during the phase of my research. I am also very grateful to Geogh Gorh and Silvia Kolssa for there advises in continuing my research.

I also recognize all the participants how took part in user study to evaluate my work. I am also thankful to my dearest friend Muhammad Mudassir Khan for his valuable advices. Least but not the least my family, because of their prayer I would able to make it.

Abstract

This thesis presents the concept of impact of Persuasive mobile recommender system in lifestyle domain. It combines the previous research in Persuasive recommendation system, Active Learning and critiquing. It is tailored to one the leading smartphone mobile platform, exploits the benefits of mobile environment by following the user interface guidelines to overcome the challenges comes with limited screen sizes.

Explanations in recommender systems play a role of motivator, which influence on consumers to opt recommendation item. In past, explanations were used as one-way communication. System does not use them to get user feedback and doesn't have any impact on user decision. The system developed in this thesis generates personalized explanation that has an impact on user decision. Generated recommendation considers on user's preference and context. In addition to this system allow the user to critique on recommended item so that system can learn more about use and correct its wrong assumption.

An iOS application was developed, that rely upon the Principles named as Cialdini's Influence Principles, a framework that is well tested and verified. The purpose behind using this approach because it enrich the recommendation with explanation and also allows examining their impact on persuasiveness. Also evaluate the explanation in terms of transparency, user control, perceived efficiency and user acceptance. In addition to this application follows existing research of Active Leering Critiquing methodology to learn about user preferences. The resulting application was evaluated by the diverse set of users.

Contents

1. Introduction

The motivation behind this master thesis is to implement and evaluate mobile-based food recommender system by using Persuasive and Active Learning techniques. In the beginning, it provides overview and gives the reasoning about the selection of system, presents the detail of design and development process presented and finally demonstrates the performance of two variants of the system against each other in a real user test.

This chapter will enlighten the motivation behind developed system in Section 1.1. What goals need to be achieved describes in Section 1.2. The last Section 1.3 will provide a brief outline on structure of thesis.

1.1. Motivation

In this thesis, a mobile food recommender system developed, but why choose a mobile platform?. Rapid innovation and significant advancement in the field of technology and scientific research has made smartphone a primary computing and communication device. Smartphone is now become a necessity of life and people use it as an assistant for their day to day work. According to latest survey 60% of internet traffic is accounted by mobile device [15]. Enhancement in communication technology and flexible data option provided by network operators has increase relevance of interactive mobile applications. Packed with hundreds of features smartphone use different applications for variety of functionalities which required internet connectivity. Furthermore, smartphone support touch screen and rich support of multimedia and other application take the user experience to the next level [30].

Next question arise why choose a food domain?. The World Health Organization [49] is predicting that the number of obese adults worldwide will reach 2.3 billion by 2015 and the issue is attracting increased attention. Therefore, electronic food management systems have become a hot topic and, are under consideration to replace traditional paper based program. Idea of using electronic devices for health related matter is not new; similar devices are in use by patients for medical reasons e.g. Glucometer, and blood pressure monitor. People want, to carry better life style and to live healthy.

Therefore popularity of food monitoring systems is getting popular. These systems are not only providing valuable services but hold user preferences and keeps history to provide more personalize recommendations. Recommendations are based on food ratings and browsing histories.

Food recommendations have gotten a tremendous amount of success and still in research phase for further improvement. Along with significant advancement and feature set like similar recipes, recipe nutrition detail, where to buy ingredients from some research, some wholes are still remaining. Indeed recommendation techniques like collaborative, content and knowledge based filtering are good for job done. But food domain is not quite simple. User preference and taste not only change by their mood but much more depend upon their health. Therefore, Active learning and critiquing techniques are required to improve better recommendation. So that user can give their feed back and get what ever his preferences are. Mostly approaches are done critiquing by using rating of recipes and generate their result by using celebrative or content based filtering. Similarly, knowledge based filter digs some more; here rating is based on ingredients. Furthermore, persuasion of recommendation is always not guarantied in all cases. Clearly the system is not able to provide the best recommendations due to its detachment from the current situation; what is lacking in these approaches are intersection of persuasiveness, active learning and critiquing and last but not the least user preference context.

This work focuses on generation of food recommendation on a mobile platform. Recommendation provided the system should be persuasive in nature in terms of user interface and explanation of recommendation. Where explanation helps user to understand what factors that took into account while making a recommendation. Also allows her to correct the wrong assumption about her preference by the mean of Active Leering and critiquing. Generation of recommendations should rely on user context and her preferences.

The following short description of the target scenario will illustrate the driving idea behind this research project.

Amy comes form gym and aiming to cook. However she doesn't want spend her time by searching recipe from different recipe website. She opens up the mobile app that display her recipe. Within few interactions she got some recipe that are according to her deity need and cooking time, along with the explanation why this recipe recommended to her. Feeling confident with the recommendation she cooks that recipe.

1.2. Goals

On the basis of scenario, describes in last section, this work reflects the goals which are stated below:

A recommendation is valuable if it interests the user. To determine the generated recommendation is according to user interest entails to our first primary goal, which is offering Persuasive recommendations with Active Learning and Critiquing. Second, explanations provided by system for recipe should leads more user control and transparent in nature. Finally, provide a high quality, intuitive and sleek user interface design to provide effective user experience. Every aspect of UI should follow the user interface guideline. System should provide recommendation without considering any restriction in terms of login and initial preferences.

To achieve primary goals there are several other interesting secondary goals, which facilitate, how our primary goals should be achieved. Starting with the research phase, which includes question and answers to user how they want to use such system in order to achieve better usability. Next focus on existing search work how the other system implements food recommendation scenarios, finding out what are their weakness and strengths. Food ontologies understanding how they are interrelate with other. What factor in which recipes are dependent on in order to develop strong system. Understanding user context which time he prefers which recipe. Furthermore, it is important to research on what researches and related work are out there under Persuasive and Active learning and Critiquing system to grab the understanding, how we can get inspiration from their valuable approaches and work. Finally focus on user experience of such application is one of challenging task, how and where to show the important aspects of recipe in our interface, so that it is easy to learn and has improved usability in comparison with current market applications.

Once the research phase has done next step to collect the functional and non-functional requirement of the system, which is gathered by observing existing systems in the domain of food and performing pre-test. Once the system is build it has been tested with gathered functional and non-functional requirement and find out the limitation or boundary conditions of system. More over iOS client needs to be test with given requirement additionally user satisfaction should be required for usability test.

Evaluation of the developed prototype should be performing in a real-world scenario. The system should resemble with the existing application in terms of user interface. The employed Persuasive and critiquing system should be available in two variants, one serving as a baseline for evaluation. Both variant should be tested by the diverse set of user in a real world scenario along with real data. Results should reflect both positive and negative point about the both systems. Criteria of evolution should also consider the combination of existing research of Persuasive recommender system. After finishing the evaluation reflected results leads to potential improvements and opens up the new direction of research.

1.3. Outline

Division of this thesis is split up into six chapters. *Chapter1* contains introduced the ideas, motivations and goals.

Chapter2 starts with background in which some definitions and classification of recommendation systems, Followed by different types of profiling and contexts that impact on recommendations. Furthermore, in related work section, pervious work of Persuasiveness, Critiquing and Personalized food recommendation techniques have been discussed.

Chapter3 explains the design decision that made to develop the system. It start with discussing Profile in terms of Food and User. After that in detail discussion about the Context that systems consider in making of recommendation. In the end it will discuss how Critiquing and Persuasion along with algorithms that are used to develop the system..

Chapter4 discuss the System design and architecture phase, which hold the all ERD, components view, servers on which system depend. In the end of the chapter API calls are mentioned which are provided by server.

Chapter5 elaborates how the user study has been conducted by mentioning the goals, methods, and testing framework along with the dataset. In the end of this chapter measured results and discussion is mention.

Chapter6 will summarizes the achievements and gives clues about further development and research.

2. Background and Related Work

This chapter will establish the foundation of Persuasive recommendation system along with active learning and critiquing approach. Prior to in depth analysis, it will provide important background information along with some required definitions. Additionally, related work will presented, as the chapter proceed further to the end.

2.1. Definitions

2.1.1. Recommender System

Recommender Systems (RS) are search tools, which supports user decision-making by providing the suggestion that, are according to their interests. Such systems are in widely use from social networking to e-commerce sites in order to achieve different purposes. In e-commerce site, they help not only to serve the customer by suggesting items according to their preferences but also support business to improve in its sale. On the other hand in social network site, to suggest friends or pages like according to user preferences. According to Ricci [30] "RS are information search tools that have been recently proposed to cope with the "information overload" problem, i.e., the typical state of a web user, of having too much information to make a decision". Proposed solution [29] is an intelligent system that suggests the product or service that fulfill the user's preference in given context or situation. Suggestions provided by such systems are depended on the model how they are keeping information. Majority of RS are typically community based. In this kind of modeling suggestions are depend about item popularity among the user. Where popularity is calculated by ratings. Important question that arise in such systems are to find item accuracy according user preferences. On the other hand Personalized models are used that depends on the various factors which includes user's preferences, history of bought/liked items, or the items the user has ranked in the past. Various techniques are use in the developing of recommender system. Classification of recommendation systems [32] will be discussed as follows.

Content-based filtering

In this technique recommendations are based on user preferences. System recommend items that similar to once's liked by user. Item similarity is calculated by features associated with the compared items [32]. For example, if a user has rated positively, recipe A under the category of sweets then next suggestion that is provided by the system is one which is similar to one user has liked before.

Collaborative-based

Collaborative filtering is a technique in which system find the correlation between item and user, based on other user's feedback having a similar taste in past [32]. Initially system calculates all similar taste users for the current user and calculate the recommended item that contains either rated or liked by other users having similar taste. Importantly in this approach item speciation will not be considered. For instance, if user like recipe A then next recommendation would be recipe that there are other users who liked recipe A also liked recipe B.

Demographic

Recommendations are generated according to user demographic profile. Recommendations can be produced for different demographic niches by combining the ratings of users in demographic clusters [20]. For example, suggestion provided by the systems are shown according to user's age.

Knowledge-based

In knowledge-based systems item recommendation is based on domain specific knowledge, which justifies how certain item features meet according to user's preferences [32]. Importantly, it uses predication techniques namely Case-based reasoning which reuses the cases past cases that are similar to current case in order to identify item set of recommendation.

Community-based

Type of recommendations provided by this kind of system based on preference of user friends. According to Ricci research [32], People tend to rely more on recommendations provided by friends rather than on recommendations from anonymous individual having similar taste. Such type of RS model relies on user's social relations including preference of user's friends. Suggestions depend on rating that is provided by user's friends.

Hybrid Recommender Systems

Hybrid system is a fusion of any two or more techniques motioned above. Ricci [32] explains the motivation behind such system to avoid the limitation of one technique. For instance, Collaborative filtering have cold startup problem i.e. they are unable to suggest those items, which have no ratings. On the other hand Content-based doesn't have such limitation by combination of both approach new hybrid system can be formed. Similarly, Burke and Robin [6] proposed the combination techniques to create a new hybrid system.

2.1.2. Contexts

Recommendation techniques used by traditional system relies on vector of item rating and user preferences. According to Suchman [39] these approaches ignores the notion of "situated actions" which infers that user have particular context and item preference within one context may be different from another context[2]. Absence of context may lose information predictive power because of aggregation of multiple contexts. For instance user wants to buy cloths for his child. Instead of given him child dress system suggests dresses according to user choice because of incomplete contextual information.

Since context is a multi dimension topic therefore vast amount of research has been done in area, narrowing down role of context in recommender system, context can be defined as all information according to given situation. One of the early definitions of context in terms of operation [35] defined as where you are, who you are with, and what resources are nearby. As research further increases, new and most sited definition of context according to Adomavicius [2] "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 an application, including the user and applications themselves". Also, Dourish, P. et al., [9] while observing the uses of context they classified into two views namely *representation* and *interaction* views. They assumed four key assumptions for describing representational view. Context is independent from underlying activity, delineable, stable and in a form of information. According to this view, context can be known prior as it is a set of observable attributes and structure of these attributes does not change with respect to time. Futhermore, [9] while observing the uses of context they classify into two views namely *representation* and *interaction* views. They assume four key assumptions for describing representational view context is independent from underlying activity, delineable, stable and in a form of information. According to this view, context can be known prior, as it is a set of observable attributes and structure of these attributes does not change with respect to time. On the other hand interactional view features are dynamic. It assumes that context and activity have a relationship.

Adomavicius [2] ascertain [9] claims on categories and explains recommender system can have different types of knowledge, which may include the exact list of all the relevant factors, their structure, and their values, about the contextual factors. He classifies the knowledge of a recommender system about the contextual factors into three categories;

Fully observable: refers to explicit knowledge of structure and values of contextual factors of application at the time when recommendations are made. These factors refer as Purchasing Purpose, Shopping Companion and time. For example, User wants to buy shirt, besides having information of selling point and item. Theses may include information about the time, for whom. *Partially observable*: application has some of information about the context explicitly. For instances, Purchasing Purpose, Shopping Companion and time, all information given but the structure is missing. *Unobservable*: no information regarding contextual information is provided explicitly. Utilizing only the latent knowledge of context in an implicit manner makes recommendations. For example, the recommender system may build a latent predictive model, such as hierarchical, linear or hidden Markov models, to estimate unknown ratings, where unobservable context is modeled using latent variables.

Furthermore, Adomavicius [2] find out the dependency of contextual factors over time and classify them into categories. *Static*: The relevant contextual factors and their structure remains the same (stable) over time. *Dynamic*: contextual factors change in some way.

How Contextual Factors Change	Knowledge of the RS about the Contextual Factors		
	Fully Observable	Partially Observable	Unobservable
Static	Everything Known about Context	Partial and Static Context Knowledge	Latent Knowledge of Context
Dynamic	Context Relevance Is Dynamic	Partial and Dynamic Context Knowledge	Nothing Is Known about Context

Figure 2.1.: Contextual Information Dimensions.
[2]

Representing and Modeling Context

Classical recommendation system has the prediction problem in which user's rating for item reflects the degree of user preferences. Therefore, a recommender system tries to estimate a rating function.

$$R : UsersxItems \rightarrow Ratings \tag{2.1}$$

It's a 2D matrix of user-item to an ordered set of rating values. Where "R" a general-purpose *utility* (or preference). Since the value of "R" is partial function therefore rating of all user-item are not known which invokes the predication problem.

$$R : UsersxItemsxContexts \rightarrow Ratings \tag{2.2}$$

In contrast, context aware recommenders have additional evidence to estimate user preference on unseen item. Contextual evidence can be applies to input function and viewed as "multidimensional". Where, any information related to data and user can be refer as Contextual information

Paradigms for Using Contextual Information

According on algorithm approaches of context aware recommendation, represents in form of $U\ x\ I\ x\ C\ x\ R$, Where "U" is for User, "R" denotes rating, "C" is contextual dimension, and produce contextual recommendations list i1, i2, i3 ... for each user "U". Figure 2.2 illustrates the paradigms used in processing Contextual Information. These are categorize as follows:

Contextual prefiltering: In this filtering Context is applied an input. With the help of any classical 2D recommendation data selection approach, applies current context which is used for selection of relevant ratings and dataset.

Contextual postfiltering: Prediction of rating is applied with the help of traditional 2D recommender system technique. On the resultant set of recommendation applies the context for each user.

Contextual modeling: Directly integrates the contextual information in the modeling technique as part of the rating estimation. .

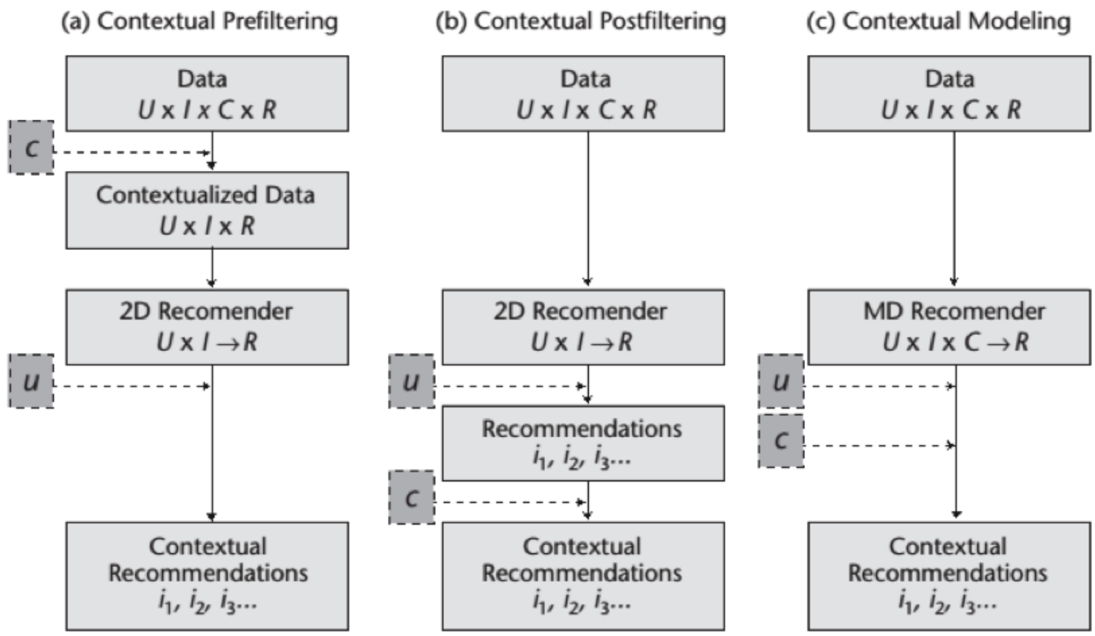


Figure 2.2.: Paradigms for Incorporating Context in Recommender Systems. [2]

2.1.3. User Profiling

User profiling is process of acquiring information about the user, which helps in constructing the user model. Rate of acceptance and effectiveness of recommendation will effect how much system has information about the user. Tang [43] explained, In the context of software applications, a user profile or user model comprehends essential information about the user. It also refers to digital representation of person in a system and hold the person's preferences. Variations in user profile content depend on application. Some applications depend on demographic information of user while other relies on rating, liking, disliking or other preferences. Also, consideration of user interaction and behavior take plays an important role for providing precise recommendations.

Schiaffino [34] stated that discovery of differences and similarities in interests among users is a key to provide personalized recommendations. Which implies that application users have different preferences and interests to achieve their goals which leads to importance of creation of user profile in order to find out relationship between users according to their interests. Content of user profile depends on application domain and variations are presents in application. Categorically there are two methods to gather content of user profile. First method is manually, in this technique users are being interviewed, fill some forms or questionnaires. For instance asking about his demographic profile like where he is from, which dishes he likes or rating based questions like how frequently he eat junk food? On the other hand by mean of implicit learning about user preference which requires artificial intelligence (AI) technique like Case-based reasoning, Bayesian Networks, Artificial neural networks. Most of these techniques are beyond the scope of this work so they will not be described in more details. Fundamentally there are two different alternatives to build a user profile; either the information is obtained explicitly from user or implicitly through the observation of user's actions. The next section describes these alternatives.

Explicit Profiling

Explicit profiling often known as explicit user feedback is the simplest way of gathering information about users. In this technique, user has to fill questioners in order to develop profiles. Profiles developed by this techniques are totally depends on the questioner. Normally, data contains demographic information regarding the user like name, age, location and other preferences. Gauch [34] suggests input methods that allows user to rate item according to their interest. Alternatively system can provide check box and text field in order to get preference of user. *HTML forms, Questionnaires, Rating, User preference and Touch sensors* are some techniques user to obtain explicit profile. Whereas, the limitation of this technique is that it requires user time and willingness to provide their data. Some user has reservation in regards to provide their personal information because of potential privacy concern. However user's preference can always be determine via this technique.

Implict Profiling

Implicit profiling is also known, as implicit user feedback is another approach for building user profile. It is popular and widely used methodology to develop user profile based on user 's acquired information. Mostly profiles are derived form monitoring or observing user activities. Information acquires by the system helps them to ensure the given recommendation is according to user interest. For example popular website for watching videos, "YouTube". It's recommendations are made on similar to those videos that were watched by user in the past. Single-sign-on (SSO) [**hursti1997single**] one the most common methodology for user registration. Since the information is implicit, they may contains user demographic information. Similarly user preference likes, dislikes, locations can be fetch by integration of related third party services. Information obtain from this mechanism plays an important role in personalized recommendation system. *Search logs, Browser cache and User monitoring agents* are some techniques used to generate implicit profile. Main advantage of this technique is user doesn't have to fill the forms and provide their information. Kelly [34] provides an overview of standard techniques that helps to build user profile and information types about the user that can be inferred from user's behavior.

2.1.4. Conversation-based Critiquing Recommenders

Recommender systems may also vary in the function to the extent that user can get engage in the dialogues. In traditional techniques data was collected once and terminate after recommendations are made. Assumption of these approaches are user know all his preferences at the beginning, which was not the case. Whereas, user taste may change over time and he want to interact with different option. Smyth [21] handles this problem and called "Conversational recommendation system" (CRS). He states that CRS have their origins in conversational case-based reasoning (CCBR) which apply similar techniques to elicit query information in problem solving domains and diagnostic tasks." It is an interactive approach in which user preferences are establish through conversation session. At first initial set of recommendations are given to the user. System adapts the user feedback to further enhance the recommendations. Smyth [21] distributed feedback into three categories. *Rating-based* in this approach user provides rating for an specific item. *Critique-based* where user add constrain over item features. *Preference-based* in which user indicates its preference for one particular item over the others.

2.1.5. Active Learning

Active learning (AL) is a methodology to learn about user's preference by asking him/her to rate a number of item known as training point [33]. Data is formed by model that approximates user's preference. It is useful where user's preferences are change with respect to context. Rashid [28] explains that objective of AL may varies according to objectives of recommendation systems. For example, what is important in the recommender system being built? The difficulty of signing-up (user effort)? If the user is happy with the service (user satisfaction)? How well the system can predict a user's preferences (accuracy)? Furthermore, this approach can solve cold startup problem in an effective manner. Figure 2.3 explains how interactive process works in order to obtain training data, unlike passive learning, where data is simply given to system in a linear fashion. Rubens [33] categorized AL method according to their primary goal.

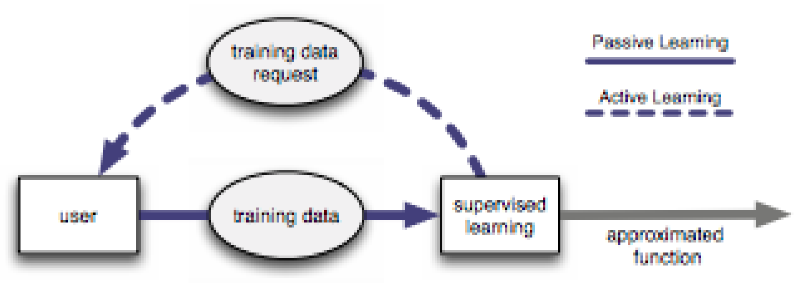


Figure 2.3.: Active vs Passive Learning
[33]

Instance-based Methods

In this kind of approach points selection relies on their properties in an attempt to predict user's rating by the closest match to other user, without have explicit knowledge about underlying model [3]. Whereas, it assumes that under considered model, any data and rating predictions are accessible.

Model-based Methods

In this methodology point selection is based on best construct model that explains data supplied by the user to predict user ratings [3]. Similarly, select point are used to maximize the reduction of expected error of the model. Whereas, it assumes that in addition to any data available to instance-based methods, the model and its parameters are also available.

Modes of Active Learning: Batch and Sequential

Since the expectation of user is to high for the system while they are performing interaction. They expect immediate output from the system. One common approach is to recalculate the rating of item once user rated that specific item, known as sequential mode. However another possibility is that to allow user to rate several features of items or rate several item before model readjustment, known as batch mode. As immediate reflection of data in sequential mode is an advantage but cost of interaction will always effect. Therefore, trade-off exists between Batch and Sequential AL; the usefulness of the data vs. the number of interactions with the user.

2.1.6. Persuasive Recommedations

Traditionally, prevalent research in recommender systems has been focused on algorithms development and evaluation that provide precise recommendations[50]. The presumption behind the algorithm that its accuracy contribute to the quality and acceptance of the recommender systems has been changed lately[24]. Context and profiling techniques are also emerged as an important pillars. Additional factors which are important and should be focused on is presentation of recommendation so that user can interact with the system in more convenient fashion[24], transparency of system or explain working of system to the end user[38], persuasion[27] and recommendation's novelty[8]. Fogg [11] defines persuasion as *the attempt of changing people's attitudes or behaviors or both*.However, explanation of recommendation also have influence on user [23] [14].

Persuasion factors

Aristotle [13] was the first one who talked about persuasion. He claims *ethos/character of the speaker*, *message's receiver pathos/emotions* and textitlogos/argument are the main elements that plays an important role in persuasion. Since, difference of opinion exists in factor of recommendation. The most cited one is Cialdini's [7] known as "6 Influence Principles (also known as Six Weapons of Influence)". If they implemented in a system then effect of persuasion increase. Theses principle include; *Reciprocity* (humans have the tendency to return favours), *Commitment* (or consistency: people's tendency to be consistent with their first opinion), *Social proof* (people tend to do what others do), *Scarcity* (people are inclined to consider more valuable whatever is scarce), textit Liking (people are influenced more by persons they like) and *Authority* (people have a sense of duty or obligation to people who are in positions of authority). Where Figure2.4 indicates effect of Communication Persuasion Paradigm[51].

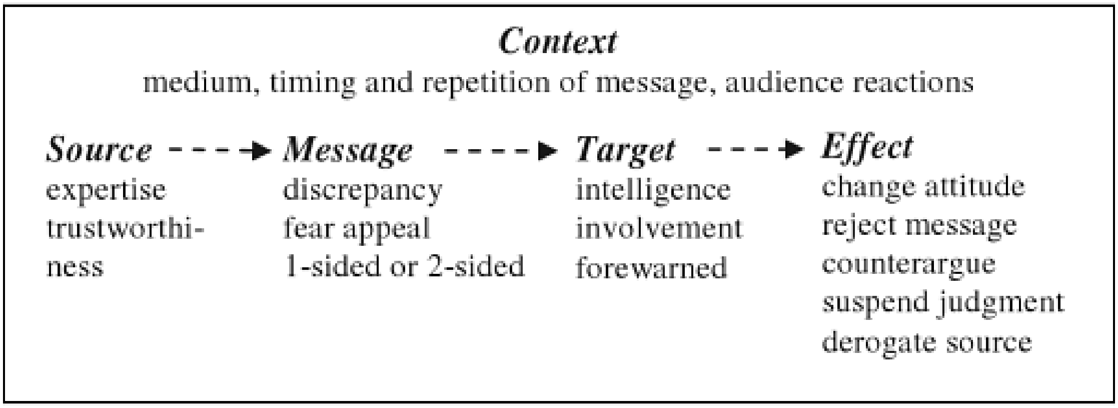


Figure 2.4.: Communication Persuasion Paradigm
[51]

Explanations in Recommender Systems

An explanation consider an additional information provided by system output to achieve certain goals [46]. Explanation persuade user to try or buy recommended item[47]. Several aspects should be considered in order to introduce explanation into the system. Following are the seven goals for explanation in recommendation system[45].

1. **Transparency** in recommendation system helps to understand how the recommendations are generated and how overall system mechanics[45]. Furthermore, it allow user to check the quality of system and in case of anomalies user have an idea why system has given this result. Let's consider an example, suppose system recommends a movie in comedy genre for a particular actor although user is not interested in it. System will provide an explanation you like some movies of this actor that's why it is in your recommendation.

2. **Scrutability** allows user to correct misguided assumptions made by system. Since user preferences may always change due to contextual factors. Therefore, it is important that explanations enable the user to understand what is going on in the system and let them exert control over the type of recommendations. Scrutability is related to the established usability principle of User Control[25].

3. **Trust** increases user's confidence in the system[45]. Trust could also relies on recommendation algorithm[22]. Studies suggested that trust is related to transparency and increase the possibility of interaction with the system.[10]. If users find system trustworthy then they are intent to return back to the system[10].

4. **Persuasiveness** convince users to try or buy [45], where as it may increase user evolution of the system. It allow user to gain benefits form the system rather than relied on other individuals.

5. **Effectiveness** helps user to make good decision [45]. Effectiveness is totally depends on recommendation algorithm. It also helps user to evaluate the quality of recommendation according to their preferences and also discard those item, which are not related to taste.

6. **Efficiency** help users make decisions faster [45]. Nielsen describe efficiency is an establish usability principle and explain it is ability of system to help user in fast decision making while selection of best suited item[25].

7. **Satisfaction** is the ability to make system fun while user is interacting with the system [45]. Similarly, providing poor recommendation tends to decrease a user's interest [42] or acceptance of a system[14].

2.1.7. Mobile Recommender Systems

With the technical advancement and features like computers, mobile phones have become a primary platform. By Combining with recommendation system it become a key tool for business and users. Due to this trend personalization in recommendation become more focused. Hence, it is important to understand the capabilities of mobile and how the information will access and displayed on mobile phone for usability prospective[25]. Schiller [36] describe the mobile recommendation system in three dimensions. According to him each one has its own impact on system usage and functionality. Furthermore, these dimensions are independent but there are examples for all the possible combinations of these dimensions, are stated as follows:

1. **Device portability** describes as information can be access from any location without bounding it to any specific location. Furthermore, user experience will be the same regardless of the location it is being accessed from.

2. **User mobility** refers to information is access from mobile device, like smartphone and tablets. On the other hand there are some constrains regarding the screen sizes and how information will be displayed on such devices.

3. **Wireless connectivity** refers to all the information, which is access by the mobile, is via wireless technology such as wifi, 3G and LTE.

General Issues

Besides, all the advancements and opportunities provided by mobile devices. Some limitations are exists and stated as follows:

1. **Resource limitation**
   Mobile devices offer less memory and power consumption as computer to standard computing machine for example desktop computers.[30].

2. **Connectivity**
   Since the communication in mobile system is depends on wireless connectivity therefore, high level interruption and noise occured. Additionally, data rate is also very low as compare to standard wire communication.[30].

3. **Battery life**
   Although significant research and improvement has been performed in battery life of mobile device but a lot needs to be done for providing longer battery life.[30].

4. **Fragmentation**
   Fragmentation refers to different screen sizes of devices. Since there are more than thousands of devices that exist having different screen size and resolutions. It is a challenge for developer to support all of them.[30].

2.2. Related Work

This section gives an overview of previous work and related papers on the respective domain.

2.2.1. Food Recommender Systems

Mayumi, M. et al.[48], design system to enlighten importance of personalized recipe recommendation system which is based on user's food preferences. In this research user's preferences are gathered by his browsing and cooking history. Recommendations provided by this system is not depends on what user ate in past. Moreover, system does not require particular action of user to collect his preferences. Figure 2.5 illustrate the idea estimating user's favorite ingredients. Their approach breakdowns the recipes into like and dislike ingredients set. It calculate ingredient the score by incorporating the frequency of that ingredient in the dishes that has eaten by user and consider them in like ingredient. Whereas, ingredient is consider as dislike by user if he never browsed the recipe details or never cooked. Considering the research gap, developed system is working on implicit profiling technique therefore unable to consider addition context and preference of user while generating recommendations. Moreover system does not allow user in correction of the wrong assumption made by the system.

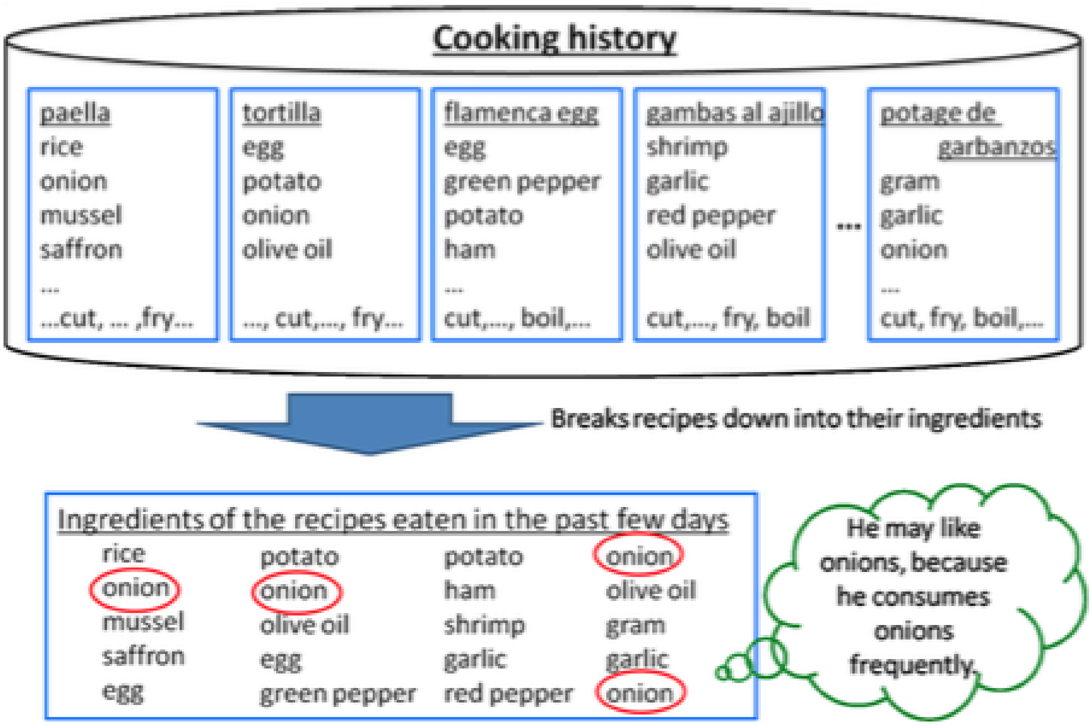


Figure 2.5.: Extracting the favorite ingredients using cooking history [48]

Freyne, J. et al[12] research investigate recipe recommendation system techniques by applying different recommendation techniques [12]. Initially with collaborative filter approach with simple break down to relate recipe and ingredients result were not so good but by applying content based approach there were significant improvements in result. For optimal solution they used hybrid approach of content-based and collaborative filter. Summary of their work is stated as after breaking down the recipes into ingredients, give and compute score ingredient score, applying collaborative filter to narrow down the ingredient score and finally applying content based approach in which prediction of recipe rating is examine by the score of individual ingredients. This approach conveys the basic idea recipe recommendation system. The purpose of our research not only to understand the recipe ontologies but also considering user context like cooking, like/dislike ingredients, recipe's course which fills the gap of this research.

Suksom, N. et al.[40], focus on personalized food recommendation system aims to provide dietary recommendation based on individual diet and preferences by using knowledge base recommendation technique. Where as, knowledge based depends on ontology and rule-based knowledge development. Figure 2.6 gives an overview of system. This research allow user to setup there preferences on startup there is no method by which user can update their preference nor the system allows to correct the wrong assumption made by the system. With help of Active-Learning and Critiquing our research helps to learn about user and his current preferences.

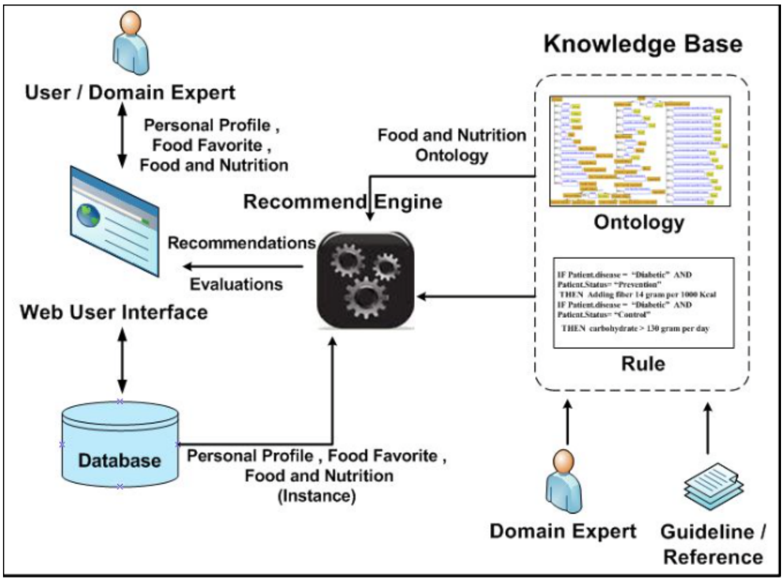


Figure 2.6.: Knowledge-based framework for the food recommender system [40]

Teng, C. et al.,[44] research permits collaborative recipe generation and modification. Recipes data are gathered according to regional preferences and modification is done by individual ingredient preferences. By this approach two kinds of networks are created one is ingredient complement and other is ingredients substitution. The network suggests which substitution of ingredient increases the taste of the recipe and gives those recipes, which is high rated by user. System uses collaborative filter approach along with data mining techniques. System does not have diet specific recipes according to user preference nor consider any user context.

2.2.2. Critique-Based Recommender Systems

Ricci [31] described the critique based mobile base recommendation in the domain of travel. Motivation behind this research to collect collects user preferences via critique with low amount of user effort. According to this research it is an advantage to collect user preference via critiquing and it is relatively fast. He modeled user preferences on the bases of long-term and session-specific preference. Given a user's request, the system exploits the user's preferences model to initialize the search query, which consists of the three following components.

1. The logical query (QL) models must conditions that need to be independently satisfied by any of the recommended products. The logical query is a conjunction of logical conditions (constraint) on single product features.

2. The favorite pattern (p) models wish conditions that should be matched as many as possible. Wish conditions allow trade-offs to be made, i.e., a good match on a condition could compensate a bad match on another one. The favor- ite pattern is represented as a vector in the same vector space of the product representation.

3. The feature importance weights vector (w) is again n-dimensional and models how much each feature is important with respect to the others.

Figure 2.7 shows rest how the critiquing is perform. We apart our research from Ricci's by dealing in domain of food. Also our research improves the system reactiveness by providing the conversational-based recommendation.

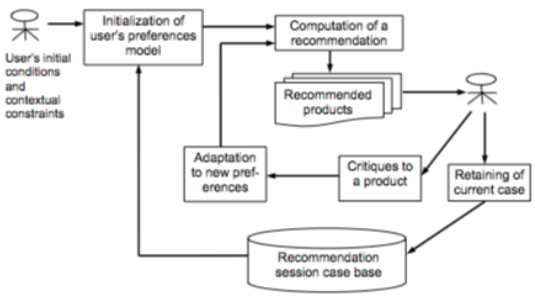
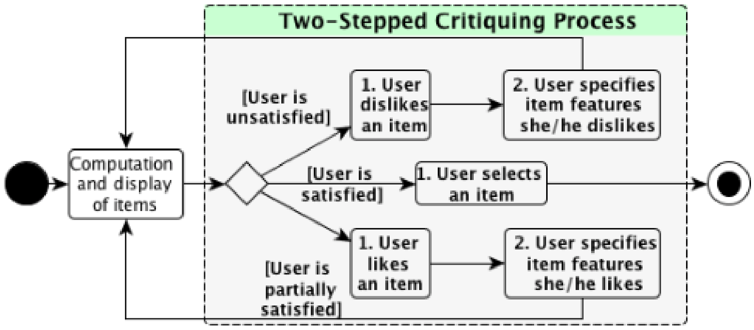


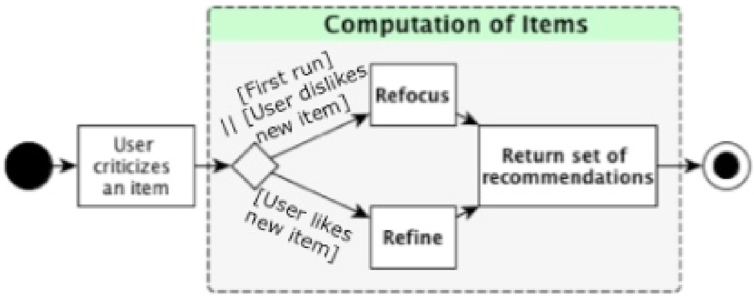
Figure 2.7.: Critique-Based recommendation model
[31]

Lamche, , C. et al. [18], investigated shopping mobile recommendation system by using conversation-based active learning approach. It uses the utility-based context and critiquing for feedback. Recommendations improvements are done by two-step critiquing process which illustrating in at Figure 2.8a. Critiquing process is relies on either positive or negative feedback. Figure 2.8b shows the refine model of recommendation in a system. However, system doesn't consider the user's context in making of recommendation that will be filled by our research. Moreover, both research consider different scenarios of lifestyle, mean one deal with shopping and other food.



(a) The two-stepped critiquing process

[18]



(b) Active Learning recommenation model

[18]

Figure 2.8.: Active Learning Strategies

2.2.3. Persuasive Recommender Systems

Ricci, F. et al.,[13] designed and analyzed a movie recommendation system. The purpose of the system was to investigate the role of persuasion. In that scenario he tried to recommend those movies that are according user's interest. It helps them to understand the persuasion effect that users are willing to accept recommendation or not. System is design to suggest persuasion recommendation based on Kaptein's [16] methodology and follows the approach of six(6) best-matching explanation **??** to provide Persuasion Principle.

Influence Strategy	Explanations
Reciprocity	A Facebook friend, who saw the movie that you suggested him/her in past, recommends you this movie
Scarcity	The recommended movie will be available to view from 15/1/2014 to 31/1/2014 on cinemas
Authority	The recommended movie won 3 Oscars
Social Proof	76% users rated this movie with 4 to 5 stars
Liking	Your Facebook friends like this movie
Commitment	Watch this movie and you may change your mind about this kind of movies

Table 2.1.: Best-matching Explanations on each Influence Strategy

Discussed system does not provided any explanation that are involved in recommendation making, nor dealing the contextual factor, neither allow user to correct wrong assumption create by system. Our research also follows the same approach as Ricci but the dimension of domain is different and considers all the mentioned factors that Ricci's didn't.

3. Design Decisions

This chapter will provide the explanation about the design decisions that we have made to develped our system. It will also highlight how they intereact with each other in order to achieve the goals of this thesis. The chapter begins with explanation of profiling approaches, followed by impacts of context, critiquing and persuasion.

3.1. User Profile

One the core component of a system is to recommend user food that suits to his preferences, which can be gathered by profiling a user. In our system we followed a hybrid approach to build a user profile. Demographic information of a user is implicitly fetched from his Facebook account. The reason behind following implicit profiling approaches is to get user information without bothering them. This allows system to have up to date information about them. However, it has been noticed that people are reluctant to those systems that request permission to access their social network activity information. Knowing these concerns, we only ask users to permit access to their basic information. Following are the acquired attributes from Facebook profile.

1. Birthday
2. Email
3. FirstName
4. LastName
5. Gender
6. Name
7. Profile Link
8. UserId

Moreover, explicit profiling techniques are used to gather users' contextual information and their preferences. While explicit profiling reveals accurate information, there however exist shortcomings in this technique. It demands user's time and willingness to provide the data by filling the long forms, which seems to be tedious to the users. As the system is Knowledge based Personalized recommender, this problem has to be dealt efficiently because the recommendations produced by the system are highly influenced by user feedback. Therefore instead of making a user to provide all the information, we collect this data by using interactive forms based, which includes simple toggling, rating and selection mechanism that also increase the usability of the system.

3.2. Food Profile

Food recommendation is the basic research area of this thesis. Based on this approach our research is to provide recommendation according to both individual's dietary needs and preferences. Understanding food domain is very complex and challenging task when its come to recommender domain. User's selection of a recipe is highly depends on it's ingredients. Also there are some other factors which includes cooking methods, ingredient costs and availability, complexity of cooking, preparation time, nutritional breakdown, ingredient combination effects, as well as cultural and social factors [12]. Our research starts with in finding out how popular websites are dealing in this domain and structuring the recipes. So that we can get inspiration about the important features that user are looking for while he interacts with such system. Next chore of our research is to build a recipe database therefore we need a provider-API that ensures a large number of recipes. Among these APIs two notables with impressive meta-data about recipes are:

1. Yummly API.
2. BigOven API.

Both services are crowd-source driven, highly recommended in food domain and are offering almost the same data set. Next step to find the best suited API for our research therefore we preformed some experiments targeted to comparison between both selected API. Result of this experiment showed that Yummly API is not providing cooking description. On the other hand BigOven API doesn't support recipe's nutritional information and have limited number of calls per hour for student account. Regarding selection of API our focus was, it should provide all the relevant information about the recipes required by our research, in order to avoid any dependency. Considering mentioned fact we decided to choose BigOven API.

Concerning about the attributes of food profile we followed the common approach that recipe have some important key attributes like cooking methods, ingredient preparation time, nutritional breakdown [12]. However we are unable to get nutritional information due to API's constrain, as discussed early, but in our data model we are considering it for future research purpose. Figure 3.1 illustrates the key attributes of food profile which is a common fashion for representing a recipe. We followed an hybrid approach [40] [44] [12] for our personalize knowledge bases food recommendation system. Recipe's ingredients are the primary factor on which recommendations are relied.

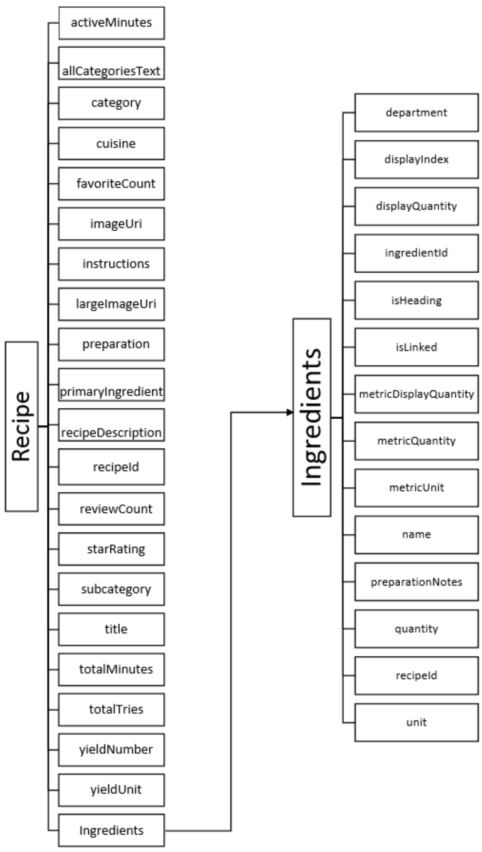


Figure 3.1.: Attributes in Food Profile

Assumption

In order to simplify evaluation of recipe recommendation, System assumed that liking and disliking of ingredients by user is based on his dietary needs and health preferences. Suppose user does not like a particular ingredient let's say "X", therefore system learns from user's critique and eventually avoids such recipes, which have "X" as an ingredient in it.

BigOven API

BigOven API provides all the information about the recipe in a well-structured and well documented manner. Along with the high number of recipes, they offer functionalities including *Search, Display Recipes, Recipe review, Grocery List and Rest-based API support*. For this thesis we focus only few of them to develop a database of your system. Following are some API calls that are implemented in our system.

1. **Reading a Recipe.**

   The Recipe object refers to a recipe within the BigOven collection.

   URL request:
   GET http://api.bigoven.com/recipe/id?api_key="bigOvenApiKey"

Parameter	Description	Required
id	Primary key(ID) of recipe	Yes
api_key	Your api key issued to you by BigOven	Yes

Table 3.1.: Bigoven- Reading a Recipe.

2. **Recipe Search Results.**

   The Recipe Search Result object is a collection of results for a given recipe search query.

   URL request:
   GET http://api.bigoven.com/recipes?title_kw=" keyword"&pg="page"&rpp="resultPerPage" &api_key="bigOvenApiKey"

Parameter	Description	Required
title_kw	Title keyword being searched for	No
pg	Cureent Page to be fetched	No
rpp	Number of results in page	No
api_key	Your api key issued to you by BigOven	Yes

Table 3.2.: Bigoven- Recipe Search Results

3.3. Contexts

Any information that can be used to characterized the situation of entity known as Context. For instance person, place[1]. Incorporation of context in recommendation system leads to improve the quality of recommendation. System that uses context to provide relevant information is called context-awear system. Lee, H. et al., [19] classified context based on existing classification and definition in mobile domain. He categorized contextual information into five categories and further divided in to sub categories Figure3.2 is a illustration of his classification.

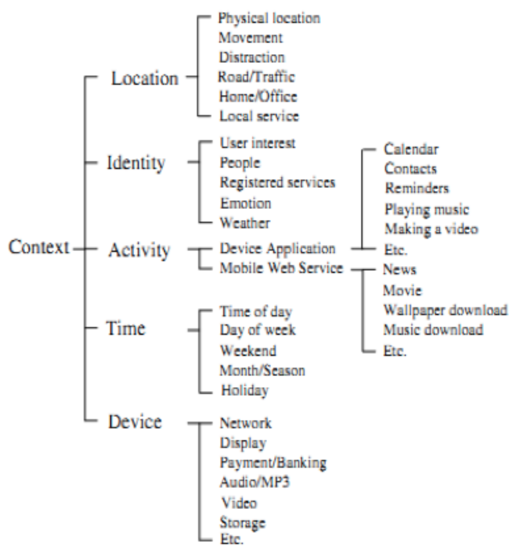


Figure 3.2.: Context hierarchy of the Mobile Web.
[19]

Location not only refers to user's current location but also what objects are nearby to user such as destination, restaurants and local services. Also it tells about the state of the user like he is moving to staying with respect to specific place, such as home or office. *Identity* is the representation of person's interests e.g. emotional state, preferred keyword , usage history and social network. *Activity* describe current usage of a mobile device, for instance which services are using by user. *Time* refers to the current time as per system clock e.g. time of device, also time elaborates in terms of day, week, month of the year. *Device* is a combination of hardware, software and network features that are provided by mobile for example Operating system version, Camera and Color. Network explains as cellular technology and wireless interface such as 3G, LTE and Bluetooth.

Considering Lee, H. et al.,'s classification [19] as a foundation, different attributes of context used in our system are discussed in the following forthcoming section.

3.3.1. User Context

As discussed in earlier section, user context create huge impact while recommendations are made. Briefly, user context refers to the current activities of user. for example, what is the user during a specific circumstances. *Cuisine's Recipe and User Health and taste* are considered as user interest in our system. User need to define *Cuisine's Recipe* while he wants to interact with the system in order to narrow down the recommendation according to selected food type. Since all the recipes are categorize as, drinks, breakfasts and appetizer. Also *User Health and taste* context is gathered by user feedback. Considering user health and taste preferences system will not add those recipes, which do not matches to his profile.

User's Health Context

Elaborating more about user's health. Definition of health is totally depending on in which perspective it is evaluating. Initially we want to calculate health by the help of nutrition information provided by the recipe but due to API constrains we are unable to define in this manner. However we come with another approach, which sound more intuitive and simple to measure health i.e. BMI (Body Mass Index). BMI depends on Person's age, height and weight. It is use to measure body fatness and health of individual. By the BMI we can record how much calories he eat and how much he required maintaining his BMI. We mocked the information of calories and exercise information in our explanation to measure the effect of persuasion.

3.3.2. Accessibility Context

Accessibility context is a combination of “Activity” and “Time” according to above mentioned classification [19]. Following are the attributes which are related to our system: *Cooking Time* indicates that how much time user have for cooking, so that system can recommend him only those recipes which are related to user's preferred cooking time. *Recipe's Next Cooking Time* assume the recipe that user most likely to cook. System will not recommend the particular recipe which is already cooked by the user during a week. Assumption behind one-week cooking gap for a cooked recipe was to provide verity of recipes to user to maintain his interest.

3.3.3. Device Context

Focus of this research is requires a mobile platform. Android, iOS and Windows phones are the three considered option. We selected iOS platform and chose iPhone as a selected device for developing our prototype. The reason behind the selection of iOS, was to develop a high fidelity UI which is intuitive and useable, by considering all the User interface guide lines provided by Apple Inc. iOS version that is required by our app is minimum 8.3. However, client side code written in swift programming language, highly recommended by Apple Inc.

3.4. Critiquing

Elicitation of user preference is the key step of recommendation system. There are many simple approaches for accusation of user preferences and transform it into user model. Traditionally, these were acquired explicitly where user need to fill a form and mentioned his wants and need. However, it has been noticed users avoid in filling information about themselves. Using these approaches result less knowledge about user preference and poor recommendations. To solve these problems two methodologies have been suggested. In first approach accusation of user's preferences takes place by analyzing of user's navigation behavior. This assumption that user always visit his interested item. Advantage is requiring lower user effort. While shortcomings are: (1) It depends understating of domain specific knowledge because user actions are translated into user preferences model. (2) Noise existence because preference and context of a person may differs from another person.

Second is conversational approach a new paradigm for the collection of user preference and redefines human-computer interaction. Such systems are based on interaction cycles in order to gather preferences about the users. At interaction cycle, the system can either ask the user a preference or propose a product to the user. The user can reply either by answering to the question posed or by criticizing the system proposal.[31]

Moreover, Critiquing by conversational approach is not enough in our case. It can answer to the cold startup problem and are unable to deliver quick results. Therefore Active Learning (AL) is the additional approach, which is used by our system in order to quickly deliver good results without preexisting knowledge about user preferences. [18]. We followed Model-based AL methodology in order to construct user model regarding ingredient and recipe selection, avoid expected error in model. As far as AL mode is concerned we followed the Sequential model states as: recalculate the rating of item once user rated that specific item. In our case ingredients and recipe [28].

Following our goal to develop the user profile over time using active learning methods in recipe recommendation scenario. Training points of our application is recipe rating, ingredient like/dislike that will build over time in order to make accurate suggestion. Initially when we have no training point based on conversational approach our system recommends top ten recipes according to given contextual information based on Cuisine and Current Preferred cooking time of user. As it is unlikely that user always wish to eat same type of food and have same cooking time. At some point user have a tough schedule due to other activities and have less time to cook. Similarly eating preferences changes with respect to meal 's time like breakfast, lunch, dinner and drinks. Considering the dynamic behavior of user and interest conversational approach is more suited. We followed Knowledge based recommendation approach, as our system is more specific to user's health and taste and the more knowledge about the user have more strong recommendation would be. We also consider content-based approach in our system in order to select popular recipes among the users to drag the attention of user. Initially when system does not have user preference, it follows collaborative filter approach by suggestion him top ten recipes of system. In order to improve the critiquing we are categorized in two manners, First, How much to like the recipe based on star rating. Second user likes particular ingredient or not which is Boolean in nature. Where system categories each ingredients in three state like, dislike and neutral (these are neither like nor dislike by user). Figure 3.3 explains our appoach of critiquing.

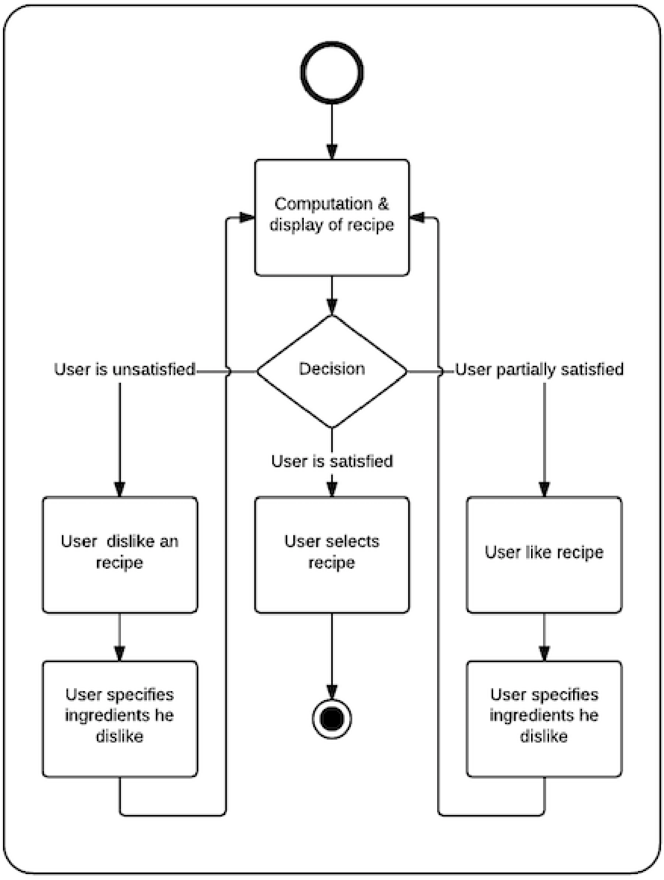


Figure 3.3.: FoodForMe Critiquing Algorithm.

3.5. Persuasion

Developing persuasive recommendation system (RS) is mainstream of this research. Traditionally, RS has been more focused towards algorithmic approaches and relied on them for providing accurate recommendations. As there was an assumption that the accuracy of algorithm is the key factor that affects the quality of recommendation. Recent studies shows there are other factors that plays a significant role in the acceptance of recommendation. Main factors are User-centric design for presenting recommendation, transparency of system (explain user how system works) via message and source. An other important factor that way influenced on acceptance is explanation of recommendation[13]. In order to achieve Persuasion in our system we focus on "Visualization or Presentation [24] [26] and "Explanation of recommendation[7][11]".

3.5.1. Visualization or Presentation

While investigating persuasion impact on recommendation our focus in terms of modality and organization of recommended items in order to achieve user satisfaction. Hence, various recipe and food systems are compared in order to follow the social-orientation methodology, as it is part of user-centric approach and studies parameter that affects user preference and satisfaction [41]. After getting an idea about what are key factors in displaying recipes and what user expect form such system. We structure our recommendations to increase the efficiency in selecting a recommended item and build trust and user satisfaction based on qualitative characteristics. Following are the approaches that we follow in organization of recommendations.

1. Primary factors that consider while structuring recommendation item is user context to increase and maintain user interests in to the system to make system *transparency and efficiency*.
2. Recommended item have recipe review count, rating and category and subcategory of recipe, recipe avatar, ordered by recipe title to achieve *effectiveness and satisfaction*

3.5.2. Explaination

Although presentation and visualization will have an impact on persuasion but the key element in order to persuade something is the explanation about that recommended item. Any type of information additional information along with system's output to achieve certain objective. In our system our task is to persuade recipes, which suits according to user taste and heath preference. However, there is no clear indication in extant literature about what type of explanations can actually lead to persuasion and at what extend. Aim of providing the explanation to measure intensity of trust, credibility, satisfaction, accuracy and transparency of system. We started our research by finding the key element on which our explanation will depends by following Aristotle's elements that helps in persuasion [13]. Following are the explanation of each element with respect to our system. *Ethos/Character of the speaker* refers to motivate user to cooking that recipe by getting the recipe for credible sources and convey message to user that our system cares him a lot by suggest him recipes according to his preferences. *Message's receiver pathos/emotions* Plays a vital role in persuading item. Health, taste and cooking preference are the primary consideration of user emotion. Our assumption was our system user are the diet conscious and want to live a better life. *Logos/Argument* is reasoning why this particular recipe is given to him what are assumptions of system while considering this recipe. After finding out the key elements on which explanation is based on. Next step to apply the Cialdin's Influence Principle[7] broadly used and verified for persuasion. Following are the description how we are applying each of the factor in our experiment.

1. *Reciprocity* describes as humans have the tendency to return favors. In our system we are achieving the mechanism of rating of recipe. We assumed that by rating the recipe user is not only providing his feedback but also helps community in recipe selection. However, we also though about integrating user's friends that will recommend him recipe based on his taste. Furthermore, we want to add dietitian that user can follow that will suggest him heath plan and based on that plan our system will recommend him recipe but this though is out of scope for this thesis.

2. *Scarcity* refers to people are inclined to consider more valuable whatever is scarce. In our system we implement this approach by the help of consumption context or categorizing the recipe according to consumption of recipe for instance breakfast's recipes, juices etc.

3. *Authority* is implemented by the popularity of recipe among that course of recipe how much star it have and how user think about that recipe.

4. *Liking* is implemented in two ways in our system. First by overall recipe rating and second is based on ingredients choices that are according to the taste of user.

5. *Commitment* is implemented by the considering the user preference. All recipes are that recommended to user have at least one of his favorite ingredient and his timing. Furthermore, we also mock exercise time to consume that recipe to keep focus on user's health.

6. *Social Proof* implemented as critiquing on recipes by given them stars. When user rates the recipe, system will recalculate the overall rating of recipe. This helps other users in finding out which recipe is popular in a system.

Explanation of recommended item depends on: (1) How much calories are contained by recommended recipe (2) How much user needs to do exercise to burn such amount of calories. . (3) Factors why this particular recipe in user's recommended item set. Let's discuss each item hypothesis behind each of discuss item. Calories are important factor that person has to major if he wants to live healthier. To deal with health factor we have to provide calories information so that he can keep track of his consumption of calories. After given information about calories it is also important to highlight how much he have to workout to burn calories. Idea behind this to integrate app with another health or medical app and based on his health condition recommend him recipes. Moreover, it is also important to keep aware of user why system is recommending him such item. For this reason we consider user preferences about recipe and ingredients. More specifically every recipe which is recommended to user contains atleast one of his favorite ingredient. Additionally organized by cooking time which is provided by user according to his preference. Figure 3.4 is a graphical representation how our system will generate recommendation's explaination.

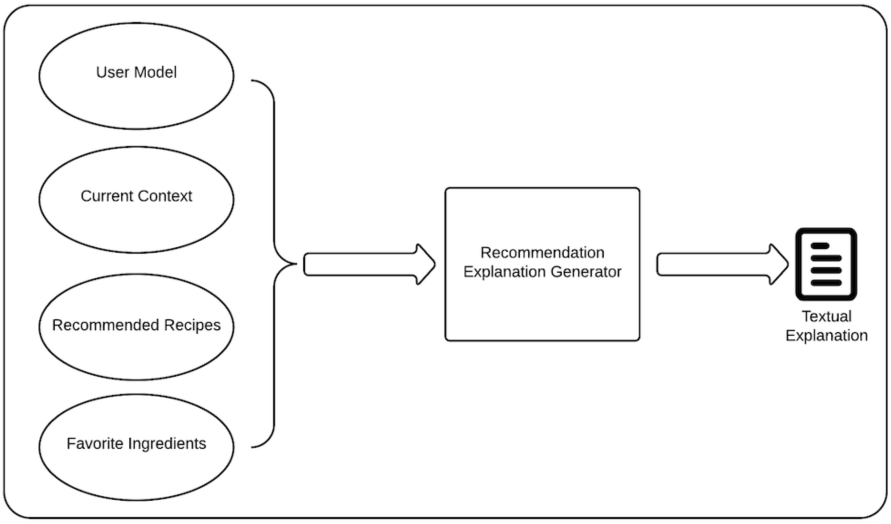


Figure 3.4.: Generation of recommendation explanations.

4. System Design and Implementation

This chapter describes of the system. It starts by providing overview of the system, followed by requirement elicitation to build prototype. Furthermore, it also provide deeper understanding of architectural paradigm by discussing each module and their interconnection.

4.1. Overview

Pervious chapter limits our discussion about the design decision and describe the essential component of the system. Since the concept is quite abstract and does not dictate any implementation details. In order to challenge the relevance and capability of the concept, a prototype app, tailored to a real world scenario, should be developed as a "proof of concept". Our system is divided in to two components: (1) Rest based web application follows modular principle of system design i.e. functionality of a system is divided into multiple concurrent modules. Where coordination of modules depend on database. Each module has its own Data Access Object (DAO) through with communication take place. Module query existing data with the help of DAO perform their task and update afterward. (2) iOS client provides all the required interface to communicate with the server. It aims to collect information that needs to build user profile and allow active learning and critiquing mechanism to update user profile and increase the trust between user and system by conveying the idea how much system cares about user and his need.

4.2. Requirment elicitation

This section represents user's viewpoint of the system. It also describes the purpose of the system by identifying the Functional, non-functional requirements and description of use case in the form of scenarios. It is important to mention here that all the scenarios are developed to evaluate the prototype and are not meant for production purpose.

4.2.1. Functional Requirments

FoodForMe is a mobile food recommender system that user iOS platform. It purpose to facilitate user to find the food what to cook that matches their personal preferences. Idea behind this prototype application is to proof the concept a combination of Persuasion and critique-based recommender system lead to better recommender and have an impact on user decision making process. Therefore all the functionality in a design is bounded to this purpose. There are two cases of interaction with the system. In case one user need login via Facebook so that system can get its demographic profile instead of asking him to fill out his personal information. Demographic information contains name, birthday, email, name and link of his profile picture. By default system keeps his cooking time preference and course selection preference. User can change these preferences from the setting screen. In second case user can interact without login and having same default preference. As it is notice that some user hesitate to provide their information without having a trust in a system. However, in this particular case user can only view the information. Our rest of discussion will relate do case one.

After getting login and change his preference. User will able to view recipes according to his preference. Each recipe shows the name, star rating, main categories, sub categories, number of reviews and recipe picture. Once the user tap on any recipe, user is able to view detail of selected recipe. Detail screen consists on 3 to 4 sections depends on screen type. Section 1 contains the generic information about the recipe same as discuss above accept it provide large Image of recipe. In Section 2 is related to recipe ingredients, each ingredient item have its name and quantity. Section 3 is about preparation/direction means its guide user how to cook that recipe. Section 4 is an option selection and it will appear as per screen type. In this section system will provide why system think this recipe is according to user preference. User is able to see two screens that display recipes list. First one will display the popular recipes of the system. Recipe course and popularity are the factors on which this will depends on. Motivation behind this to aware user what's new and hot in system and allow user to change his taste. On the other hand second list will depends only user preference. On detail screen of each recipe system will provide explanation, which tells user what system will think about him and why these recipe recommend to him.

On the detail screen of selected recipe user can criticized on showed item. User can critique on list of indigents by mentioning them he like that ingredient or not. Also he might be able to critique on recipe by given star according to his choice. Additional system allow user to change his personal preferences these include cooking time and course selection.

4.2.2. Non Requirments

From usability to performance aspect of a system Non-functional requirement can apply in many ways. However it is our assumption that this app is a prototype and will only use for evaluation purpose but we have to consider User interface, performance and supportable requirements [5]. Following are some few non-functional requirements that should guide to development process:

1. App must not be crash.
2. Any mobile user can use the application and have a clear understanding of app without facing any problem.
3. App should provide consistence user interface with respect to colors, fonts and theme.
4. App should follow the Application User interface guideline provided by apple.
5. For app start to critiquing or selection of preference must be reachable at any time.
6. Processing time of app may not exceed to 1 second.
7. Server calls should not take more 30 sec.

4.2.3. Use-Cases

Figure 4.1 illustrates foodforme uses cases. These include importing of user demographic information via Facebook, logout user form applicaiton, user can change his preference according to his convenience, browsing of recipes, critiquing on recipe based on ingredients and rating and selection of recipe to cook. To discuss the use cases we follow the scenario based approach. [4].

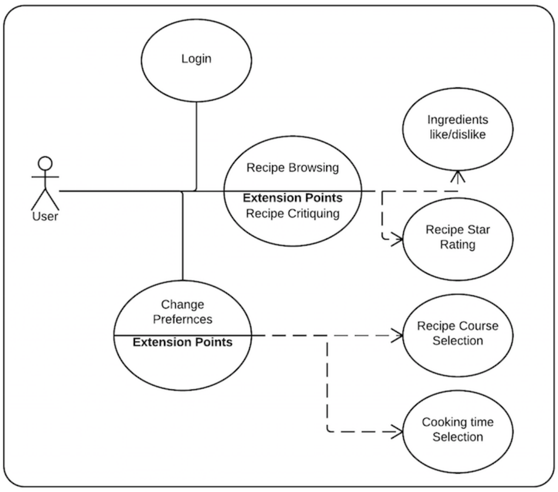


Figure 4.1.: FoodForMe use case diagram

The first use case, determine user login via Facebook to get his Facebook profile to avoid filling his demographic information by his own. Login flow should provide good user experience and follows the standard practices. Use case start any point of time after app launch and user click on main menu and ends once user can see his name and profile picture at menu screen. Table 4.1 provides the scenario description.

Use case name	Determine stereotype
Participating actor	Initiated by User
Flow of events	(1). User starts app. (2). Click on menu button. (3). Tap on login via Facebook button to import his Facebook profile.
Entry condition	User starts app
Exit condition	User can see his Facebook profile image and name on application slide menu.
Quality requirements	(1) Import user profile via Facebook SDK. (2) Profile should be import in one click excluding of Facebook login process. (3) It should take more than 30 seconds.

Table 4.1.: Use case 1:Import user demographic information

The second use case, application allow user to logout from application by providing standard mechanism provided by Facebook. After login he is not able to see his personal information in app. Table 4.2 determines the event flow.

Use case name	Determine stereotype
Participating actor	Initiated by User
Flow of events	1. User starts app. 2. Click on menu button. 3. Tap on logout via Facebook button to logout his Facebook profile. 4. App shows logout confirmation which includes logout and cancle
Entry condition	User starts app
Exit condition	User can logout form applicaiton and unable to his profile picture and name on app menu.
Quality requirements	1. User able to logout from system in two clicks. 2. It should take more than 30 seconds.

Table 4.2.: Use case 2: Logout use

Table4.3 describe use case regarding recipe browsing. According to this user can browse the recipes list that is recommended to him by scrolling up and down. Where recipe list have star rating, recipe name, recipe's category, review count of recipe.

Use case name	Determine stereotype
Participating actor	Initiated by User
Flow of events	1. User can tap on popular/ recommend recipes view. 2. User can see the list of recipes. 3. User can perform scrolling to view more recipes.
Entry condition	User starts app
Exit condition	User can found his favorite recipe to cook.
Quality requirements	1. Scrolling of recipes should be sleek. 2. Each item should have name, star rating, avatar or recipe, review count and primary catagory.

Table 4.3.: Use case 3: Recipe browsing

Use case 4 is regarding the showing the detail of selected recipe. This use case is depended use case 3. On detail screen user can view large image of recipe including all the parameters that is use case 3. Moreover it should display the ingredients along with their quantity. Finally it displays preparation method means how to cook that recipe. Senerios is describe in Table4.4

Use case name	Determine stereotype
Participating actor	Initiated by User
Flow of events	1. User selects recipe from list (Use case 3) 2. User can see ingredients of selected recipe along with quantity. 3. User can cooking method
Entry condition	User select a recipe from recommended item
Exit condition	User can found his favorite recipe to cook.
Quality requirements	1. Scrolling of recipe should be sleek. 2. Each item should have cooking method and set of ingredients along with quantity

Table 4.4.: Use case 4: Recipe Detail

Use case 5 of our system is critiquing a recipe. System allows user to critique on user's recommend recipe so that system will aware about user taste and his health need. In our system critiquing of recipe and its ingredient is down by separately so that we can evaluate user taste more specifically. It might be possible that user doesn't like the recipe but his likes the ingredient and those ingredients are the essential one for his dietary need. Therefore, recipe critiquing is done via star rating where ingredient can be critique by like/dislike. Table 4.5 discuss the event flow of this use case.

Use case name	Determine stereotype
Participating actor	Initiated by User
Flow of events	1. User is viewing recipe detail He wants to give his feed back about recommend recipe. 2. He taps on Critique button at top right corner of recipe detail screen. 3. On critique screen he may like/dislike ingredients. 4. He may rate recipe by selecting stars.
Entry condition	use case 1, 3, 4
Exit condition	User can found his favorite recipe to cook.
Quality requirements	1. Like/dislike have differnt color 2. Rating of recipe is done via stars selection

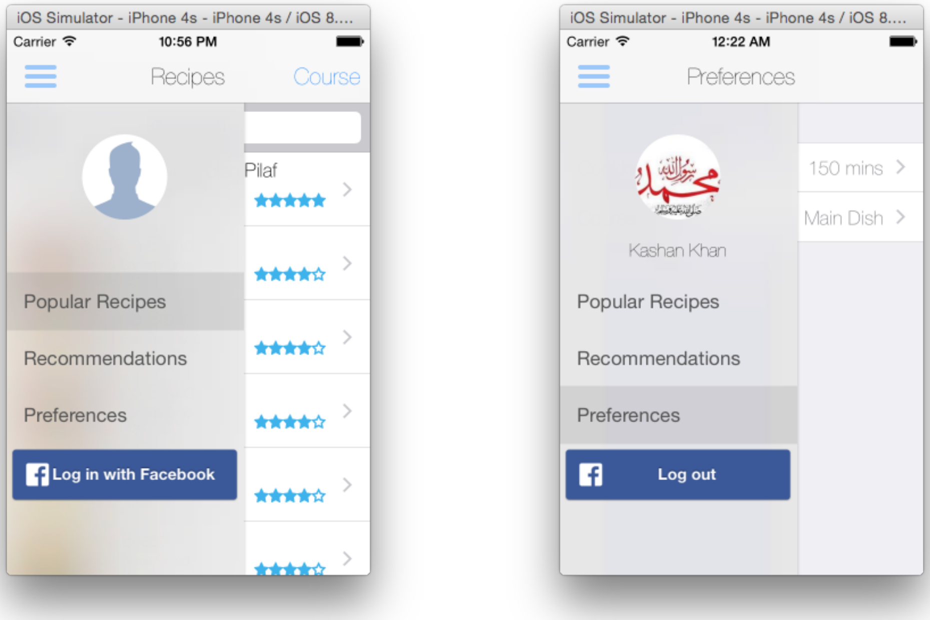
Table 4.5.: Use case 5: Recipe Critique

4.3. User interface

Current section focuses on the implementation of user interface of application, FoodForMe, It is our fundamental requirement to build a high fidelity prototype that will run iOS platform and supported device should be iPhone. Current implementation of prototype will run on iOS 8.3 and above. Last year 2014, when apple announced iOS 8 they introduced a new programing language for developing iOS application called "swift". They also declared that objective-c will not going to be obsolete. Therefore we have a choice to select the preferred language for developing client. We chose swift for our development. Reason behind selecting a Swift over objective-c is swift is 1.3x faster then objective which have an impact on performance factor of our app. Swift also provide many features like closures and etc. which is legged by objective-c. Application UI was designed with the help of Storyboard technique. The entire user interface of application is based on Auto Layout Guideline provided by Apple Inc. Auto layout is a system that helps in developing UI by creating a mathematical description of the relationships between the elements. To achieve the persuasion effect it is very important to focus each and every element of UI. Later in this section we discuss each screen implement and behavior of application. Finally section will close by discussing the data model for app.

4.3.1. Main Menu

FoodForMe main menu allows user to perform navigation inside the application from one screen to another screen. These screens include Popular recopies, Recommendations and Preferences. It is also responsible for representing user login status. Categorically we have two modes and refer as "User Logout Mode" and "User Login Mode". Each mode has in settings value in terms of presentation and behavior of application. Initially app starts with "User Logout Mode" in this status. In this mode app will show "Login Via Facebook" at the bottom of the menu along with default avatar on the top of the menu as show in Figure4.2a. Once user click on "Login Via Facebook" and successfully login then app will enter in second mode i.e. display his Name, Avatar and lower button will change to "Logout" as shown is Figure4.2a. These modes have impact on application interaction cycle. The idea behind introducing two modes is to allow user to have a glance of app without force him login. Once he is comfortable with the app and he trust the app then he can perform login. After login application will provide personalized information that is based on user interest. Second and importantly Login via facebook is one of the common and the quickest method to gather the user demographic information.



(a) Logout user (b) Login user

Figure 4.2.: FoodForMe Main Menu

4.3.2. Popular/Recommended Recipes

Popular and Recommended Recipes are the two important interfaces of our app which looks a like to each other but have slightly different functionality between them. Figure4.3 is representation how it look likes. We put more attention in design phase and tried our best to follow the user-centric design approach. Also finding out the primary factors that will help in structure of recipes to increase and mentation user's interest to achieve transparency, efficiency, effectiveness and satisfaction. To achieve this we display recipe name, rating, category, subcategory of recipe, recipe's avatar and sort of recipe by tile. However all recipes that are recommended to user is based on his preferences and course selection. Considering the fact that user can change his course any time therefore we provide course button at the top right corner to change his course.

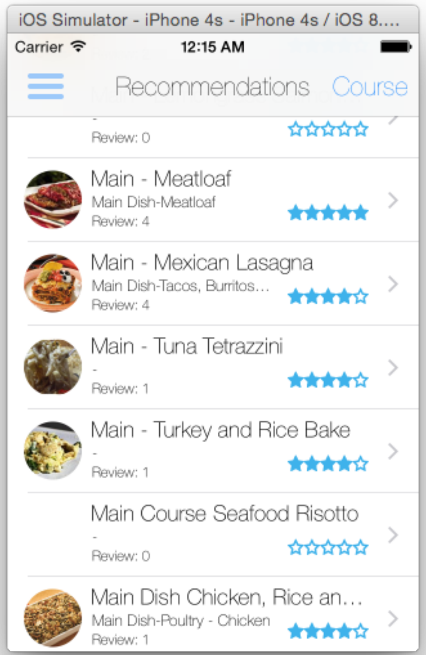
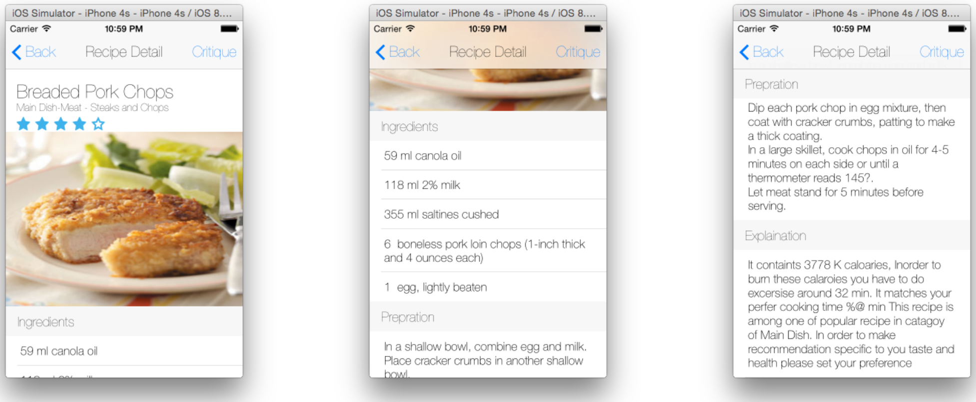


Figure 4.3.: FoodForMe Recipes List

4.3.3. Recipe Detail

Design of recipe detail follows conventional layout practices. These include by providing recipe information at the top with large avatar, ingredients along with quantity and last but not the least preparation description that helps to cook that recipes. Figure4.4 is the visual representation of interface. Keeping in mind that user can critique on recipe and it's ingredient therefore critique button is added to top of screen.



(a) Recipe Header (b) Recipe's Ingredients (c) Recipe Preparation

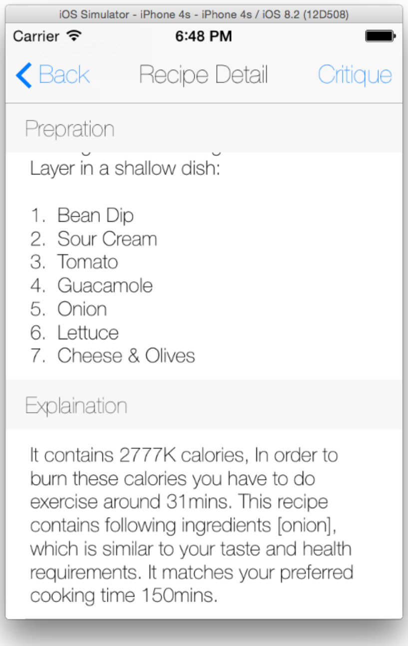
Figure 4.4.: FoodForMe Recipe Detail

Recipe's Explanation

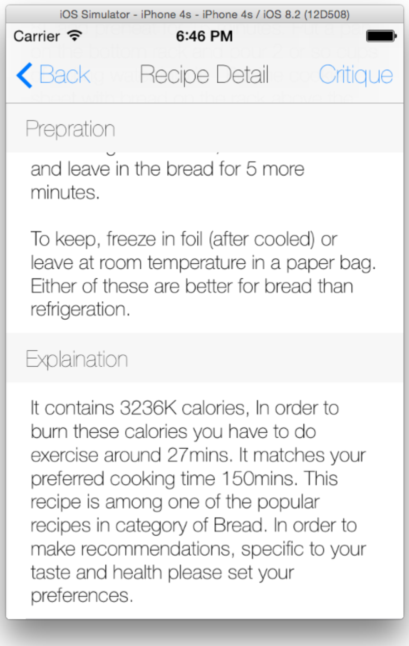
Earlier in this section we discussed that Recommended and Popular recipes have same interface but the difference comes in recipe detail. Popular recipes do not requires any information, as is task to get the knowledge about user by providing him diverse result. However, Recommend recipes should always provide explanation about why system thinks this recipe is suitable for user along with two additional factors i.e. calories and exercise time needs to burn such calories. Where, determination of suitability is integration of course selection, user preferred cooking and his taste and health preferences. Additionally system provides two types of explanations depending on user login status. If user is offline or logout, in explanation system tries to motivate user to login to get accurate recipe according to him. Figure4.5 shows the FoodForMe recipe's explanation.

4.3.4. Recipe critiquing

For Personalize recommender system is a important to know user preferences. More knowledge about the user more accurate the recommendations are. Critiquing recipes allow us to determine the user interests in our system. Focusing on critiquing importance our design should be intuitive, interactive and helps user to perform critiquing fast enough. Our assumption was user may not like some recipes but ingredients in that recipe have an impact on preference. Also it is important to what accent user likes or dislikes the recipe. Therefore we separated recipe and ingredients critiquing.



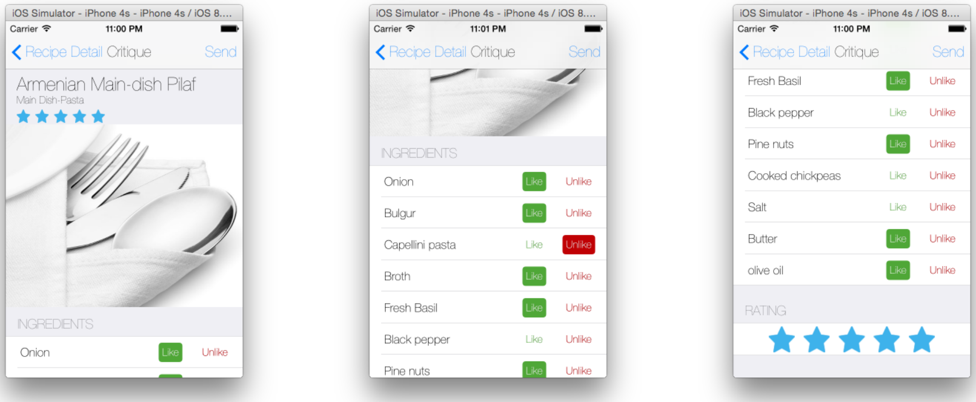
(a) Recipe's explanation For Logout user



(b) Recipe's explanation For Login user

Figure 4.5.: FoodForMe Recipe Explanation

Star rating is preferable for recipe critique. As far as ingredients critique concerns it should be like or dislike. While our implement of this screen have separated into 3 sections. Top one was about the recipe information, Middle deals with ingredients and Last section is about given stars to recipe to selected recipe. Since the critiquing of ingredients is Boolean in nature so we can use default UISwitch presents on iOS sdk but they are not much intuitive nature and may confuse users. Thus, our design follows the simple toggling button which red and green color for liking and disliking ingredients. Figure4.6 show how we implemented this screen.

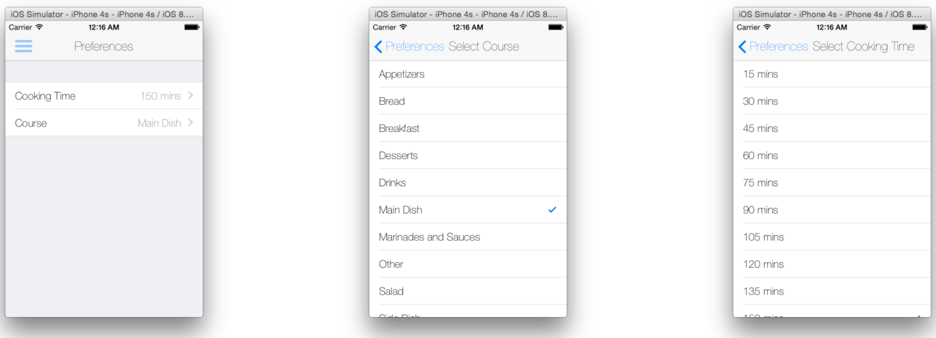


(a) Recipe Header (b) Ingredents Critique (c) Recipe's Critique

Figure 4.6.: FoodForMe Recipe Detail

4.3.5. Preferences

User context is important aspect to refine recommendations. Cooking time and Course are the two important explicit contexts that our system required from the iOS app. In our application Preferences screen allow user to select these context according to his choice. After taping on any item a detail screen will appear that allow user to select the value from required list. Figure4.7 shows the implementation of Preferences.



(a) General Peferences (b) Recipe's Course Selection (c) Cooking Time Selection

Figure 4.7.: FoodForMe Recipe Detail

4.3.6. Core Data Model

Figure4.8 is the representation of iOS Core Data Model. Where Core Data is the best way for non-trivial data storage. Purpose behind use it is to increase responsiveness, reduce the memory overhear and save a lot of code. We use SQLITE version for Core Data. Our model is divided into main components User-Profile and Food-Profile. User-Profile contains all the preferences about user. All the component of User-Profile should be update to server. Similarly Food-Profile is generated by server based on User-Profile it includes recommended recipes, what are favorite ingredients. Importantly explanation of recipe that provides to user is depends on Food-Profile.

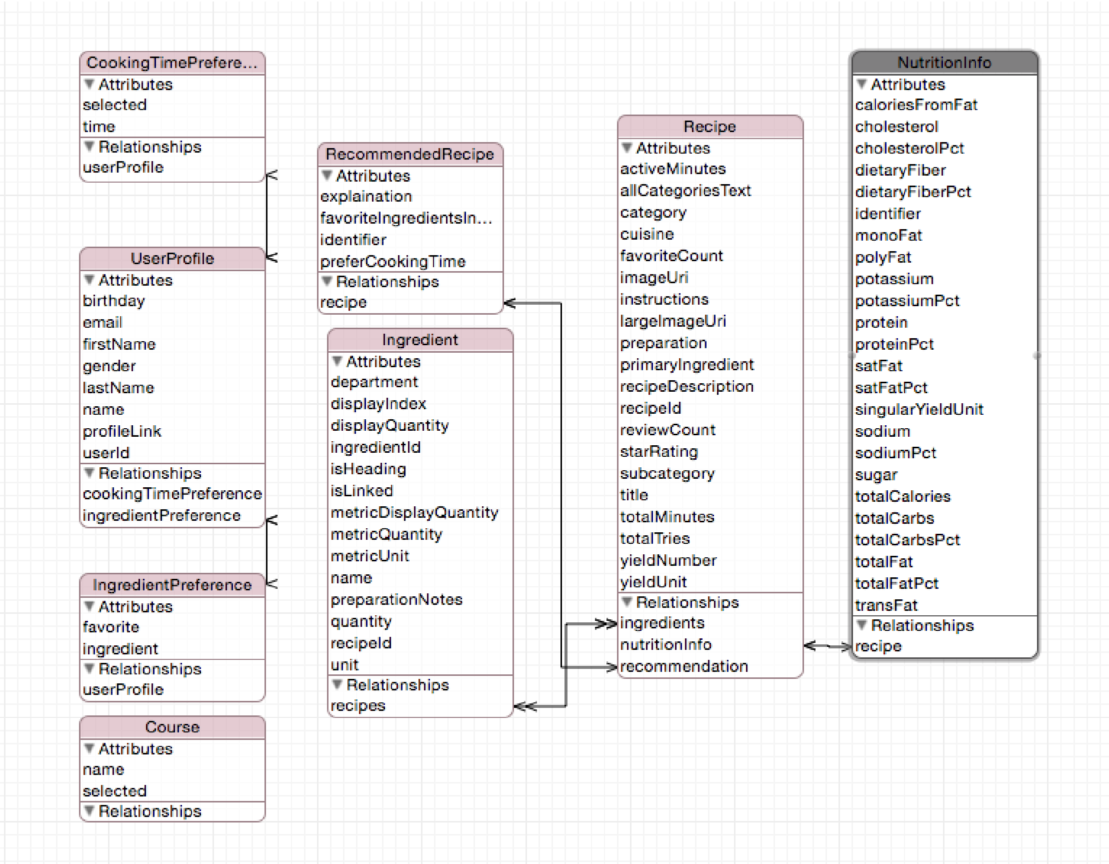


Figure 4.8.: FoodForMe iOS Data Model

4.4. System Architecture

Figure 4.9 represents the overview about the on system including client commination. Before start discussing about each component of system in detail, Let's discuss briefly about the working of system. User needs to perform login to create their account on to our system. If user account exists system will return his profile. However, if the account does not exist system will immediately create his profile and return that created profile. After the login, system will immediately gather contextual information about the user and provide to the server. On receiving contextual information of user, server sends some recommendation via recommendation module. User can critique on any recipe by using recipe service. On receiving the critique, server matures user profile and regenerates the recommendation based current information.

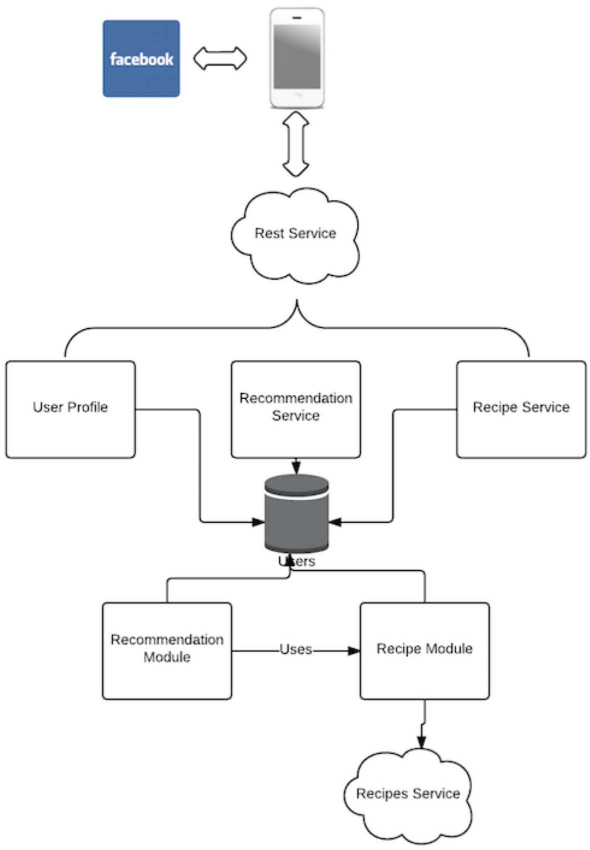


Figure 4.9.: FoodForMe: Overview of Server application Architecture

4.4.1. Class Drigram

Server Data model

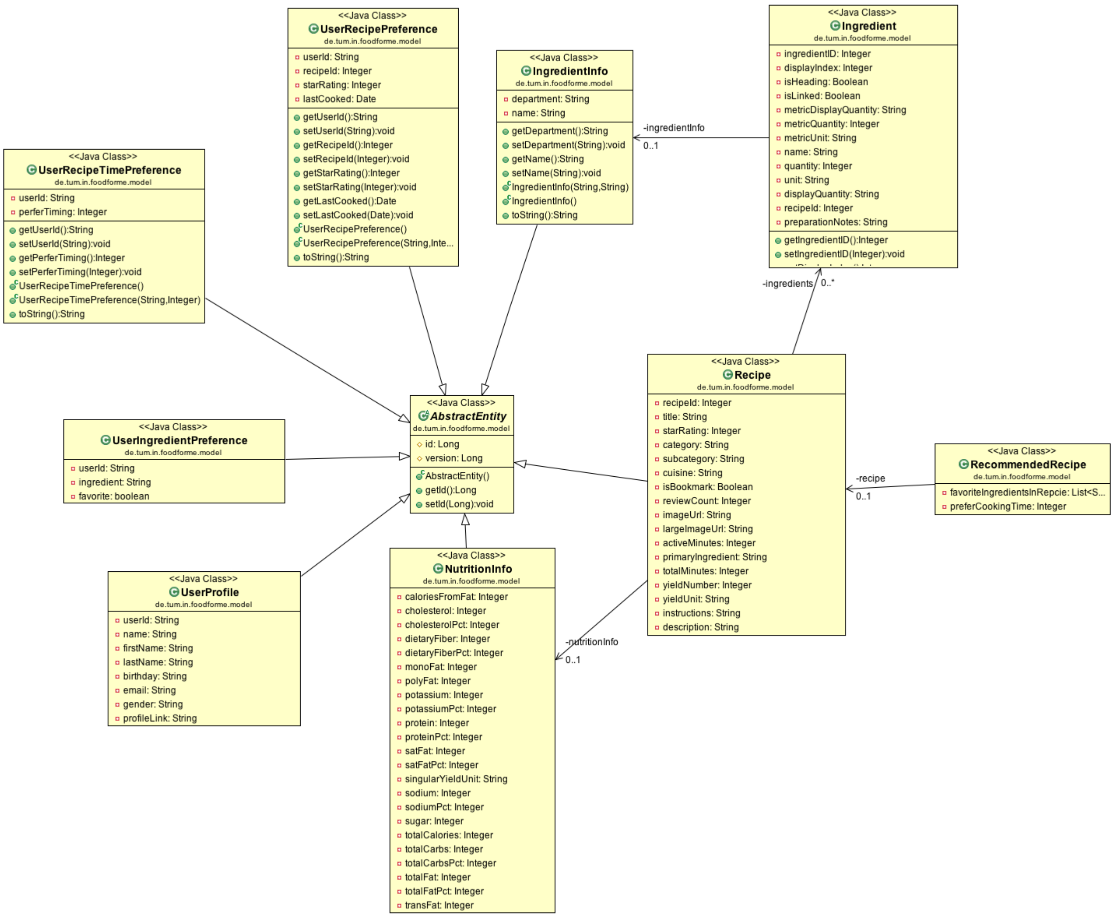


Figure 4.10.: FoodForMe Server Data model

User Profiling Module

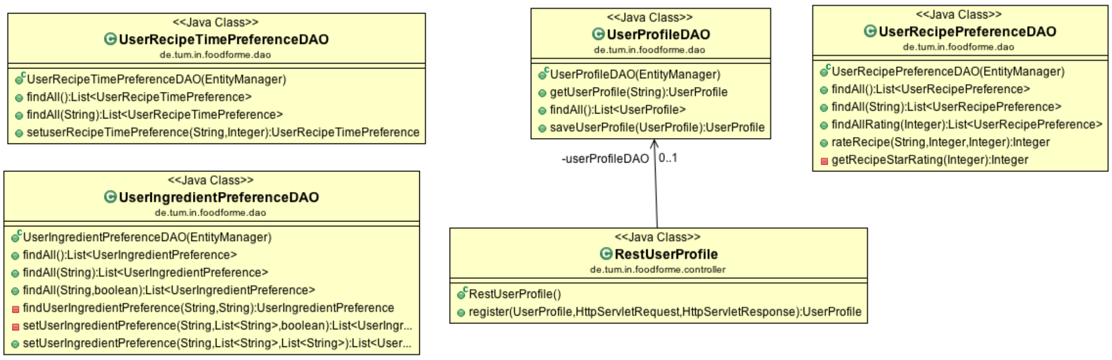


Figure 4.11.: FoodForMe Server User Profiling Module

Recipe Module

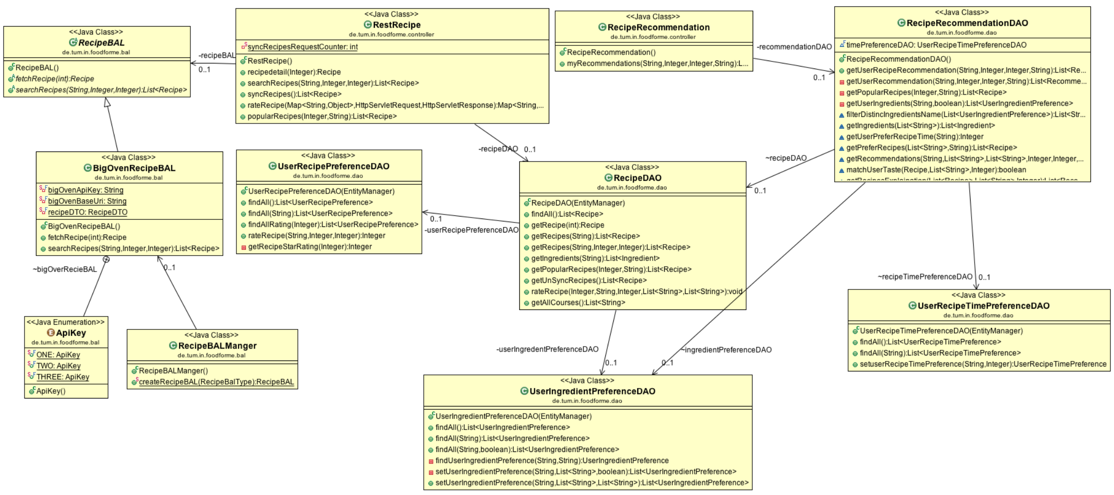


Figure 4.12.: FoodForMe Server Recipe Profiling Module

Recipe Recommendation Module

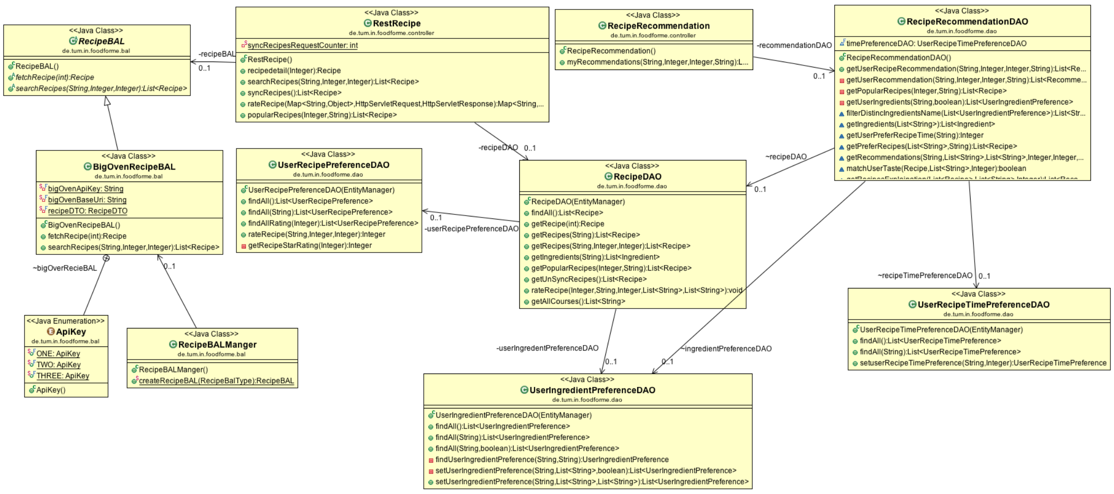


Figure 4.13.: FoodForMe Server Recommendation Module

4.4.2. ERD

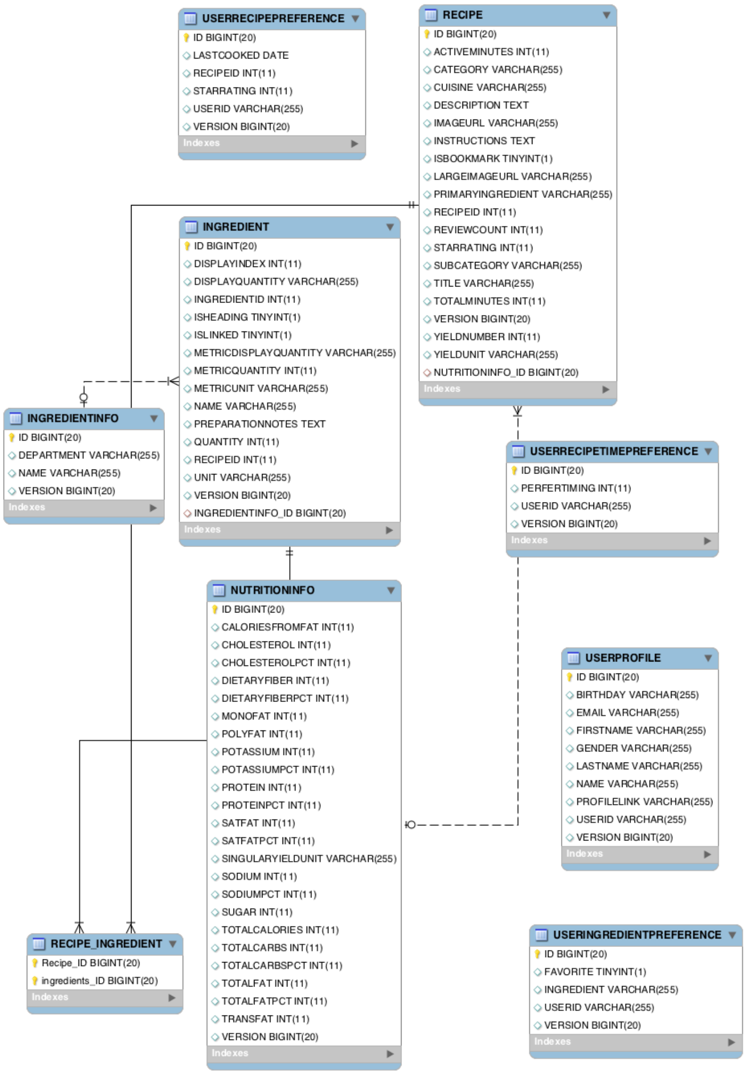


Figure 4.14.: FoodForMe Server ERD

4.5. Authentication and User Data Authorization

FoodForMe uses Facebook as a third party service for iOS client. Such services requires Consumer Key which they allocate to each app to grant authorize access of user's data. User need to grant permission to our Facebook application so that our app can access hi data. Following is the briefly discuss authentication and authorization procedure of Facebook.

4.5.1. Facebook

The authentication method of Facebook is based on Open Authentication protocol (OAuth). Application developers are required to create their app on Facebook by using URL https://developers.facebook.com/apps/. First they need to choose the app platform like in our case iOS. After creating an App they need to specify the application type. App ID and App Secret are the unique identification of each app issued by Facebook. After creating the developer need to install Facebook sdk based on a choice platform and integrate into the app. Figure 4.15 is a graphical representation of Facebook connect.

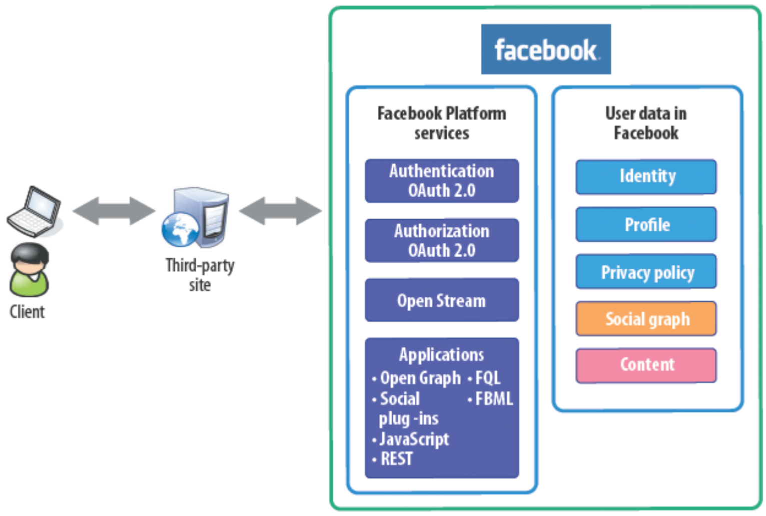


Figure 4.15.: Facebook Connect
[17]

During a login process Facebook ask user that this app requires such permission if the user is satisfied, permissions are given by him. If consumer application wants to change the permission then the login process will start again. Each system call requires App token and App Secret.

4.6. System Services

Profiling, Recipe and Recommendation are main services offer by FoodForMe Server. Our forthcoming discussion is related to the elaboration of the mentioned serveries.

4.6.1. Profiling Service

Profiling service is combination of different classes. Each is responsible for its own functionality. This service builds user profile depending his preferences as discuss in Chapter3. "RestUserProfile" class handles the user login process and profile updation. Where preference are stored separately in UserRecipePreference and UserRecipeTimePreference accorindly. However critiquing about the ingredients are stored in UserIngredientPreference class.

4.6.2. Recipe Service

Recipe Service is backbone of our system and combination of different services. Each service has its own task and module. Recipe Service acts like a factory in our system. It has It is responsible to fetching the recipes from any recipe data bank to create food profile, perform critiquing and gives the result back accordingly if user asks for popular recipes and recipe of specific course. Let discuss each of them independently.

Databank Service

Recipe provided by our system is relying on different databank. From which it will get the recipes and perform recipe sync. One data bank is implemented by BigOven Api. Considering response data format of each API is different for each other. Each data bank has provided its own Data transform Object layer that will convert response to our required format.

Critiquing Service

Critiquing service provides an interface, which allow users to critique. By the mean of critiquing system build user profile gradually. Critiquing service accepts the list of likes and dislikes ingredients set separately, recipe and star rating , provided by user. Once the system receives information it will re-compute star rating of recipe. Build the user profile by trigger profile service on behalf of received parameters.

4.6.3. Recommendation Service

This service is responsible for generation of recommendation based on user preferences and context. Recommendation service accepts the user Id, page size, cooking time and course in order to generate the recommendation. Once all the information is received. If system doesn't have user preference, in the result set system will give popular recipe according to receive context. If system does have user preference it will filter out the recipes according you user preference and generate set of parameters why system give that recipe to user. If page size of generated recommendations are not equal to request page size system will repeat the algoritm.

5. Evaluation

Despite the fact, how good the prototype in terms of algorithmic and design approaches. There will be some probability that elaborates, users are doing what they are not expected to do. This leads to other features(s) that needs to be develop in order to improves the user's satisfaction and increase her willingness to user the system. Therefore it is important that developed system should go through an evaluation proves before it goes live. Focus of evolutions to ensure that product is appropriate and the involvement of user through the design process. This chapter depicts the evaluation of prototype in a real user study and present the result.

5.1. Motivation and Goals

Motivation behind performing evaluation to determine, whether the process of recipe according to user interest in mobile critique-based recommender system can be improved by applying Persuasive Principles. As discussed earlier, purpose of developed Food Recommender System aim to be used in a real world situation. This established some aspects of the user study, including the development of a variant of the proposed application and assessment of the system.

During the study, two variants of the system will need to be tested, one begin the basic interface design, without explanation and works on basic recommender system by providing star rating to the recipes. Whereas, the second system is the main output of thesis, with better infrastructure of recipes, sleeker interface and having explanation about the recipes. Additionally recommender system algorithm supports critiquing on both ingredients and recipes.

The study is designed in a way that each user has to test both variants of the application, which system is more appealing for the users. Focusing on real effect of recommendation depends on factors like user intent, context, way to present recommendations set and other. Thus experiment needs to provide evidence as the true value of evaluation.[37]. Additionally a single irritation of user study should not exceed with more then 20 min to maintain user interest. Considering the fact that user have to test two variant of application. However it is possible that user might have get some less qualitative results which is not due the fault from the system but because users were overwhelmed with long sessions.

The focus of evaluation was to measure the effect of persuasion by providing the recommendation in form of recipes. Also how active learning also system to change itself according to user preferences. However it exclude the non-relevant part like ingredients integration to improve the recipes.

5.2. Data set Generation

Data sets are necessary to create a pragmatic setup to represent a real world objects. Therefore, a data crawler needs to be developed as an open-source project that will crawl the recipes form any recipe data bank. Crawler developed by us is written in java. Crawling recipe form data bank is a two steps process. First fetches the recipes from databank on behave of food type and course. In second step it will sync all those recipes whose details are not present in our database. Additionally recipes images are not been stored in our system instead of saving image we stores their URLs. Extracted data provide international recipes of different cousins. However, it provides functionally to add more different data source that provides recipes. To keep the amount of work reasonable items were associated with the following:

1. Unique identifier for recipes
2. 11 types of courses (e.g,. Appetizers, Bread)
3. 91 Cuisine (e.g. American, Thai, Beverages)
4. Images links of different sizes
5. Preparation
6. List of ingredients
7. Popularity of recipe

Currently our data bank has 1303 of recipes, 10037 ingredients. Furthermore, it can grow more depends on crawling method.

5.3. Setup

This section leads our discussion toward the selection of test hardware, different variants of application and testing framework for the sake of performing evaluation.

5.3.1. Test Hardware

Recommended hardware must have at least 320x 480 resolution and above, running iOS version 8.0 or later. In order words required device to run application is iPhone 4S and above.

5.3.2. Variants

Two variants of system are developed to test. Both variants use the same recommendation algorithm. However they are different to each other by have unique interface design, explanation of recommendation and critiquing methodology.

1. **Experiment (EXP)** This variant refers to the proposed developed system as discuss in chapter 4. It allows users to change their preferences while interacting to the system by critiquing on ingredients and recipes. System provides explanation to each recommended item. Additionally interface follows the user centric design, which contains more information about the recipe to draw user attention towards recipes.

2. **Basic (BASE)** Consider as baseline to compared the needs and effectiveness of actual system. In this system mostly provides textual information about the recipe. User can critique only recipes by given stars. System does not provide any explanation along with recipes.

5.3.3. Testing Framework

Testing framework that is applied in user study is a subset of the aspect that are relevant for persuasion and critiquing in recommend system. It follows the user-centric design approaches [26] and framework that applied for evaluation of recommend recommendation system [37]. Measure data is divided into following areas:

Persuasion's Principle

Persuasion's principle that discussed in section 3.5.2 were related to the how these factors will be implemented in our system. It is essential aspect for this thesis that the recommendations that are suggested to user should ne persuasive in nature and follows the principle. The forth-coming discussion relates whether user might be able to perceive those or not.

1. *Perceived Reciprocity* is allows user to return her favor by critiquing as recipe and select other good food according to particular food course and cuisine. The

participants were asked they think that system helps them to select them a recipe although this recipe is according to her preferences, but the system consider other user feedback on that recipes before making up recommendations.

2. *Perceived Scarcity* measurement indicates by categorizing recommendation with respect to context and consumption time of recipe. Participants were asked whether they find that system helps them to filter out the recipe according to their meal time.

3. *Perceived Authority* refers to the popularity of a particular recipe in a given context of user. To measure the level of perceived authority, Participants were asked to what ascent they felt authority.

4. *Perceived Liking* refers to factors that include liking a recipe. In our system measure matric was based on ingredients and recipe staring. Participants were asked which mechanism like/dislike help more in finding out the recipe.

5. *Perceived Commitment* indicated to what degree system is committed to user preference. In order to measure the degree of commitment metric is defined. Participants were asked to mark if they felt that system is committed to preferences or not.

6. *Perceived Social Proof* denotes the what society think about that particular item in our case recipe. User may or may not thought that social factor is present, By asking the user to what level they consider social factor is present in our system by providing them a scale.

Transparency

Ability of a system that allows user to understand its working and explain system choices and behavior. However it is possible that user's understanding about the system working might be differ form its actual working. For the sake of evaluation user were asked to mark choices that they think about the system when making recommendations.
Perceived Transparency: There is a possibility that user might or not perceive that system is transparent. By asking the factors which users thinks system include while making a recommendation. It is possible learn their perceived transparency.

User Control

Level of control that provides to users while operation a system refers to User Control. Under this sub section we want to measure the following items:

Perceived Overall Control indicates that does user have overall control over recommendation. The participant were asked about what they feel about while using the application by rating that were they able to tell about the system which recipes are the looking for, telling their preference to system and finally to what assent they feel they have control over system.
Perceived Scrutability denotes the ability of the system that allows user to revised their preference and excludes those assumptions that system made on there behaves. Participants were asked to rate till what ascent they feel control over system to correct the wrong assumption made by the system.

Efficiency

Providing good recommendation is not only the task of recommender systems. Also it is essential for a system how quickly it come to decisions that help in selection of item especially in the domain of mobiles. Measuring efficiency is divided into two perspectives. First, as in conversational recommender numbers of cycles were counter. Where each cycle defined as number of times user have to perform update his/her preference in order to complete a task. Each cycle have an impact on user model. Second refers as time that passed from display of first recommendation set to selection and confirmation of required item. Where time measure in seconds.
Perceived Efficiency Duration of item until item found is hard to perceive for user. Therefore for the sake of simplicity user were asked if it requires to much effort to find a recipe they are looking for. Additionally run of cycles are also hard to reminder but since conversation system cycle normally cycle are not be exceed to two cycle. Therefore user were asked to mention in which cycle they received their recipe.

Satisfaction

Satisfaction is the ability to make system fun while user is interacting with the system. Also providing poor recommendation decrease the user interest.
Perceived Satisfaction measure to determine user's feeling while in using system. It allows user to express their preferences about the system in a direct way. Participants were asked to which ascent they were satisfied about they system.

Context

Context defines as any information that can be used to characterize the situation of entity. It is important for recommendation to consider the context while making a recommendation in order to make the recommendation more concrete.
Perceived Context are determine in following ways: calories information about the recipe,

cooking time and course of recipes. Participants were asked to select which contextual information helped them to select to recipe.

UserDemographic

Participants were asked to fill their demographic information for instance age, occupation, how often they cooked.

5.3.4. Testing Procedure

Evaluation's testing procedure was structured in the following order:

1. Initially participants were asked to provide their background information, which include age, occupation. Additionally they need to provide how often they cooked considered demographic information.

2. Next step was to present the idea of system and the purpose of user study to the participant. So they have a clear understanding about both presented items.

3. Instead of requesting user to pick up the random recipe according to their interest present a realistic scenario, which helps user in order to mention their interest like as follows:

   Imagine you just came for gym. You need prepare a meal for yourself. You have prepared a meal according to you taste and diet plan and have an hour for cooking. You are not sure what to cook. You opened FoodForMe app and start looking what needs to prepare. After you find the recipe you need to provide your feedback.
   For first step your task is:
   Find a recipe you would lie to cook if given the opportunity that satisfies the following:

   a) *Cooking time 90min.*

   b) *Type of recipe : Appetizer, bread, breakfast, salads, soups etc.*

   After introducing task, hand over the app to users so that familiarize themselves with the interface and app workflow. They were also asked that this task would be executed twice with two different variants of the app, leading to potentially better or worse recommendations. Finally they need to judge which variant is better.

4. After demonstration of task, users need to understand how to perform critique and setup their preferences.

5. Participants were encouraged to given their verbal feedback about the system. Although it'ss not mandatory. All the verbal feedbacks were noted.

6. Once the tester performed his/her task and selects the recipe, which is satisfied as per his/her preference. The whole process was repeated for other variant of the system.

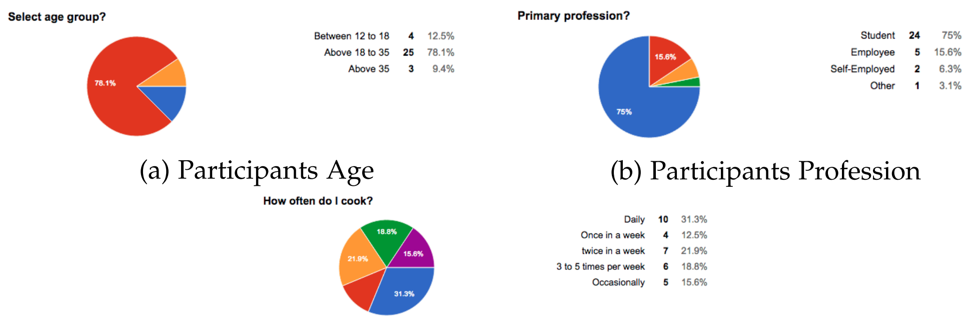
7. Finally each participant was interviewed about which variant they liked the most and why by filing a questioner.

5.4. Results

In this section our discussion leads to the result of data points measures via user study.

5.4.1. Participants

People who performed evaluation was belong to various age groups and have different occupation. Overall 31 people participated, 24 students, 5 employee, 2 self-employed and 1 is housewife. 78.1% of them have age between 18 to 35 years. 12.5% are between 12 to 18 years old. Where people whose ages are above 35 year have a small portion which is 9.4%. Figure 5.1 shows their distribution separately.



(a) Participants Age

(b) Participants Profession

(c) How much Participants cook

Figure 5.1.: Participants Demographic Info

Where as in terms of cooking, about 31.2% of the participants are likely to cook daily, however 21% of them cook twice a week and 18.8% said that they need to cook 3 to 5 timer per week and rest are those how cook occasionally.

5.4.2. Perceived Persuasion

Significant change in user's intension to cook a recipe has been observed form the result. To determine which strategies preform better in terms of persuasiveness, paired t-test were used upon the difference between initial rating and the one for each strategy. The results in Table 5.1 reflects the effectiveness comparison among the strategies.

	Scarcity	Authority	Social Proof	Liking	Commitment
Reciprocity	<0.001	<0.001	<0.001	0.083	<0.001
Scarcity		<0.001	0.169	<0.001	<0.001
Authority			0.089	<0.001	0.325
Social Proof				<0.001	0.083
Liking					<0.001
Commitment					

Table 5.1.: Paired t-test was used to examine significance, where 0.05 is set as the threshold for p-value to evaluate the significance and p-value lower than 0.001 indicates strong significance.

Next we discuss the impact of each of the persuasiveness factor individually and focus on what was the consideration of participant about them.

Reciprocity

Selecting which system is most like by user does measurement of Reciprocity. System that provides only textual information was BASE and on the other hand EXP has more information contains all the graphical information about recipes. People are more reluctant towards EXP i.e. 93.8%. Figure 5.2 represents the reciprocity measurement.

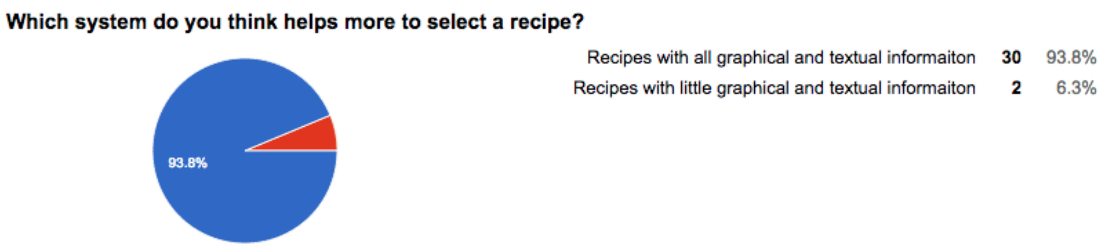


Figure 5.2.: Perceived Reciprocity

Scarcity

Asking the participants what they feel about scarcity by filtering out the recipe with respect to meal by provide the liker scale. Where 1 mean strongly agree and 5 means strongly disagree. 34.3% of them were quite sure about it and 43.8% feel good about that. Additionally 15.6% were moderate about this statement. Where 6.2% how felt negative about this statement.

Figure 5.3.: Perceived Scarcity

Authority

Measuring the authority by asking liker scale question, did they feel that recipe star rating help them in selection of recipe. 93.8% of participants were positive about that. Conversely people how negate that statement were 6.2%. Figure5.4 represents overall statics.

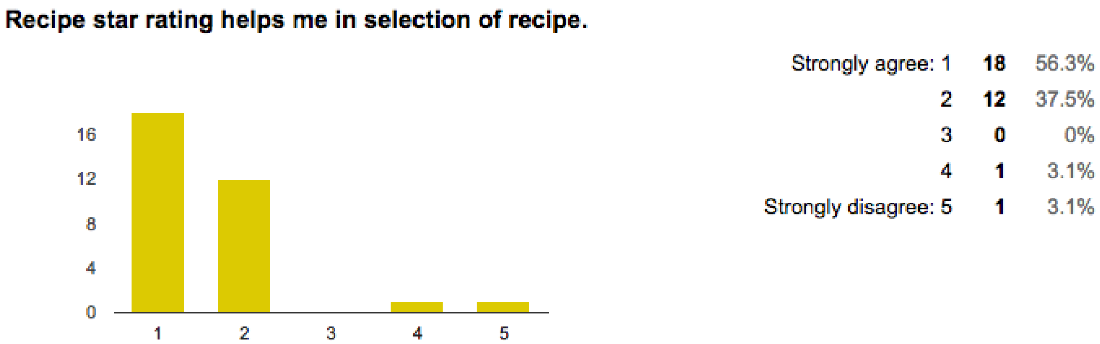


Figure 5.4.: Perceived Authority

Liking

When ask the participants which recipe critiquing technique helps them in finding their recipe. Majority of them voted for critiquing on both ingredients and recipe which in 78.1%. On the hand minor portion of the participant i.e. 21.9% liked the conventional critiquing technique to provide their feedback.

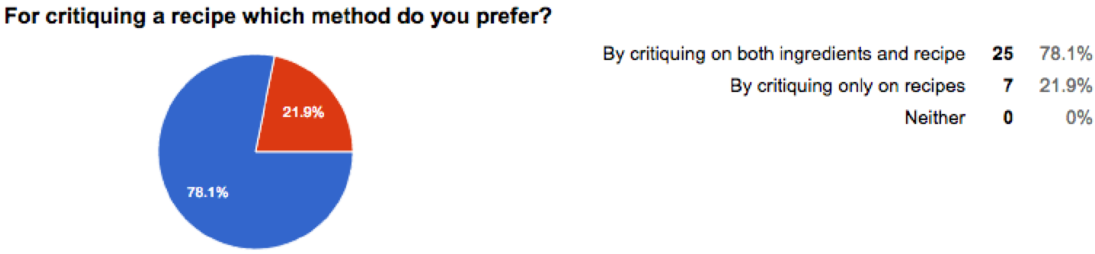


Figure 5.5.: Perceived Liking

Commitment

Participants considered the commitment by providing their answers to liker scale question. In which they have to mention, do they feel explanation of recipe helped them in recipe selection. 90.7% believed in that statement. Conversely, minor amount of dined to the statement. Where as only 3.1% were neither agree nether dined to the statement. Figure 5.6 shows the feedback.

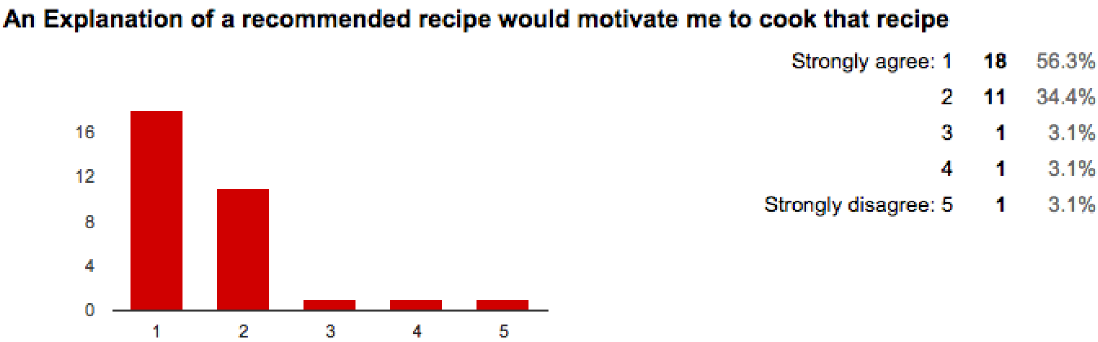


Figure 5.6.: Perceived Commitment

Social Proof

Social proof was measured by asking a likert scale question. In which they had to answered whether participant agreed that he understand that rating a recipe not only helps him to receive better recommendations but also the whole community. 84.4% of participant were positive and endorsed this fact. In contrast to that a small spike in Figure 5.7 represents to those 12.5% participants how were not favor to that.

Figure 5.7.: Perceived Social Proof

5.4.3. Perceived Transparency

Once the user performs their task they were asked about how the underlying recommendation system variants work to measure transparency effect. In general, the percipients were able to explain that based on their preferences system builds a there model in each cycle and generate a recommendation for them. While observing the result there is no clear distinction between the variants. In general, participants felt EXP is more transparent then BASE. Mean average of EXP is 4.344 compared to the BASE which is 4.125. While analyzing the standard deviation EXP σ = 0.7007 and BASE σ = 0.7931. Further analysis suggested that EXP that provides explanation perceived to more transparent (one-tail t-test, $p<0.05$ with p= 6.06E-03). Figure 5.8 shows the rating distribution between two variants. It can be seen clear that in EXP people are more satisfied with respect to BAES. Although there are some participants how feel that system is complex.

5.4.4. Perceived User Control

In our earlier discussion in this chapter, to measure the effect of user control we divided into overall control and suitability perceived by user.

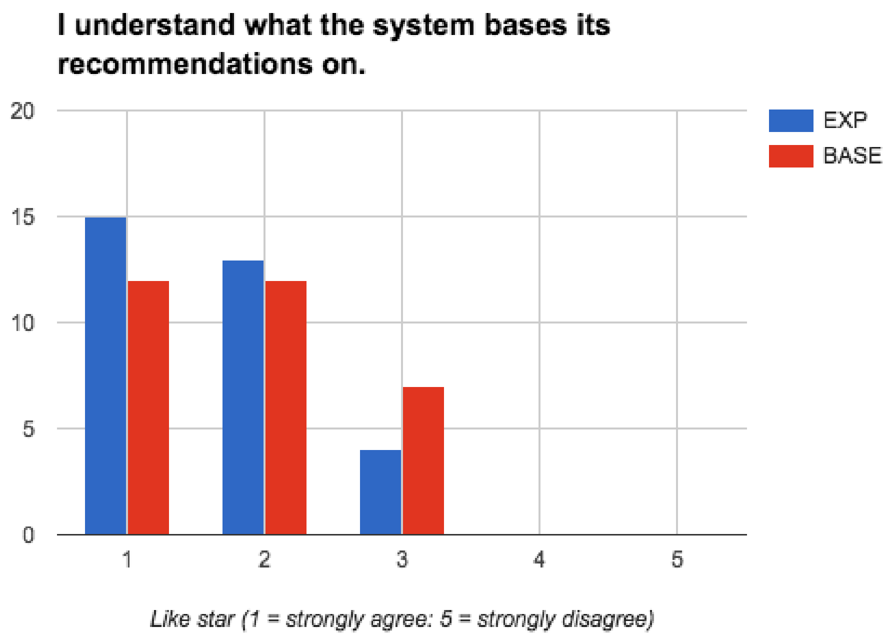


Figure 5.8.: Perceived Transparency

Overall Control

To observe that user were able to perceived overall control or not. They asked about did they felt control while telling the system what they want. Mean average of EXP is 4.219 where BASE = 3.219. Standard deviation of EXP σ = 0.8701 and BASE σ = 0.9750. Also further analysis proves that EXP seems to be significantly better than BASE (one-tail t-test, p> 0.05 with p = 2.31E-12). Figure 5.9 reflects the actual distribution of rating. Where EXP has 87.5% over positive rating, where there are some percentage of people feel neutral about system where 3.1% doesn't feel control.

Scrutability

Scrutability measured by asking the user did they feel ease of correcting mistake made by system. EXP performs a lot better than BASE. Mean average of EXP = 4.000 and BASE 2.313. Consider Standard deviation EXP σ = 0.9504 and BASE σ = 0.9651. On more analysis result also support that Scrutability is perceived on EXP than BASE.(One-tail t-test, p<0.05 with p = 1.91E-19). Actual distribution of rating in Figure 5.10 reveals the significance of EXP over BASE.

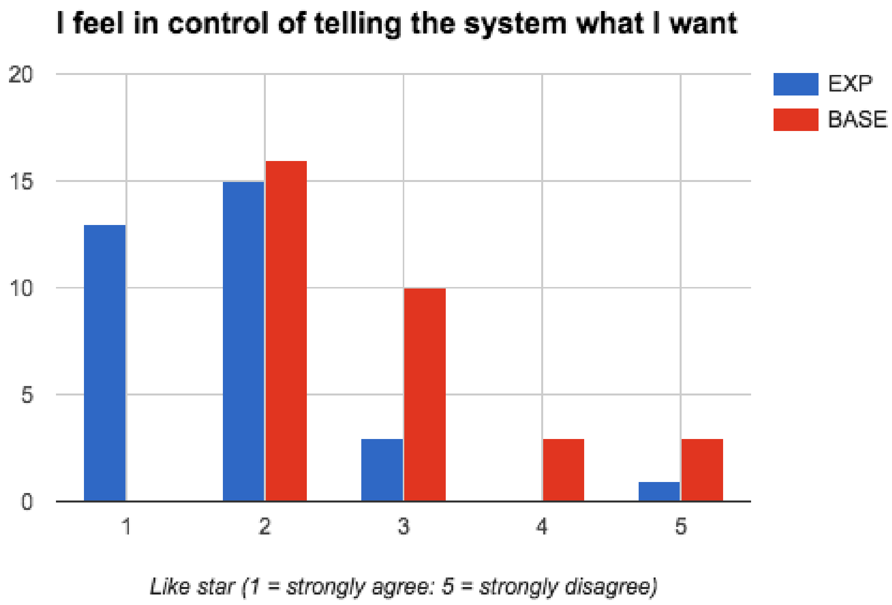


Figure 5.9.: Perceived Overall Control

5.4.5. Efficiency

Efficiency is measured by analyzing the number of critiquing cycle (number of user updates to the preferences by the mean of critiquing).

Cycles

Participants were asked to mention in which cycle they felt they had received their item. On average participants completed their task in min 3 cycles before. EXP mean average = 3.625 and BASE = 7.000 where standard deviation of EXP σ = 0.8328 and BASE σ = 1.1914. However one-tail t-test suggested that EXP is more significant than BASE($p< 0.05$ with p = 4.21E-23). Figure 5.11 illustrates the number of cycles that users had performed to complete their task.

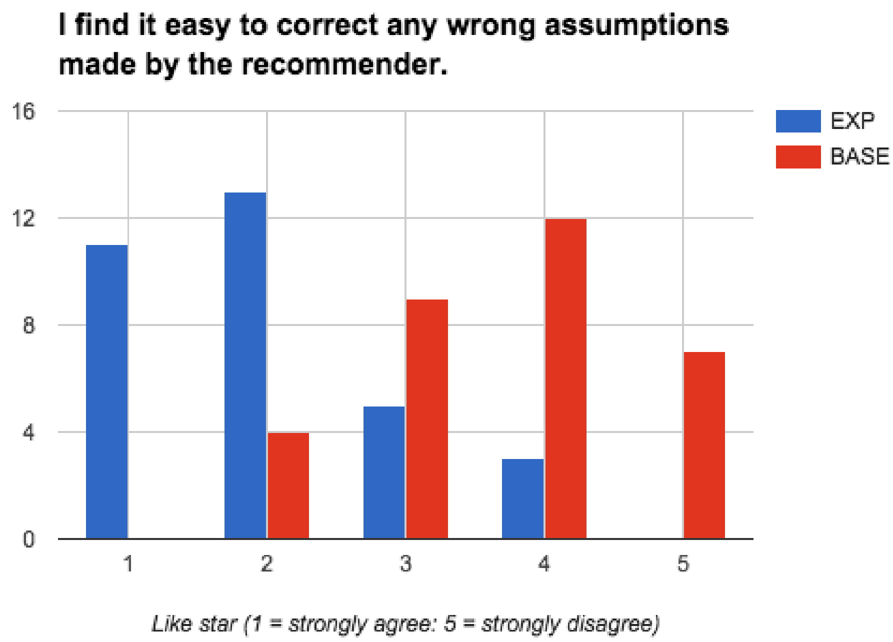


Figure 5.10.: Perceived Scrutability

Perceived Efficiency

Participants were asked about whether they felt that find the recipe requires too much effort for them. The participants felt satisfaction about the EXP. Average rating got by EXP = 1.844 and BASE = 2.500. Consider Standard deviation EXP σ = 0.6278 and BASE σ = 1.1359. One-tail t-test result also in the favor of EMP where $p < 0.05$ with p = 9.13E-06. Figure 5.12 reflects the user choice how they perceived the efficacy in both systems.

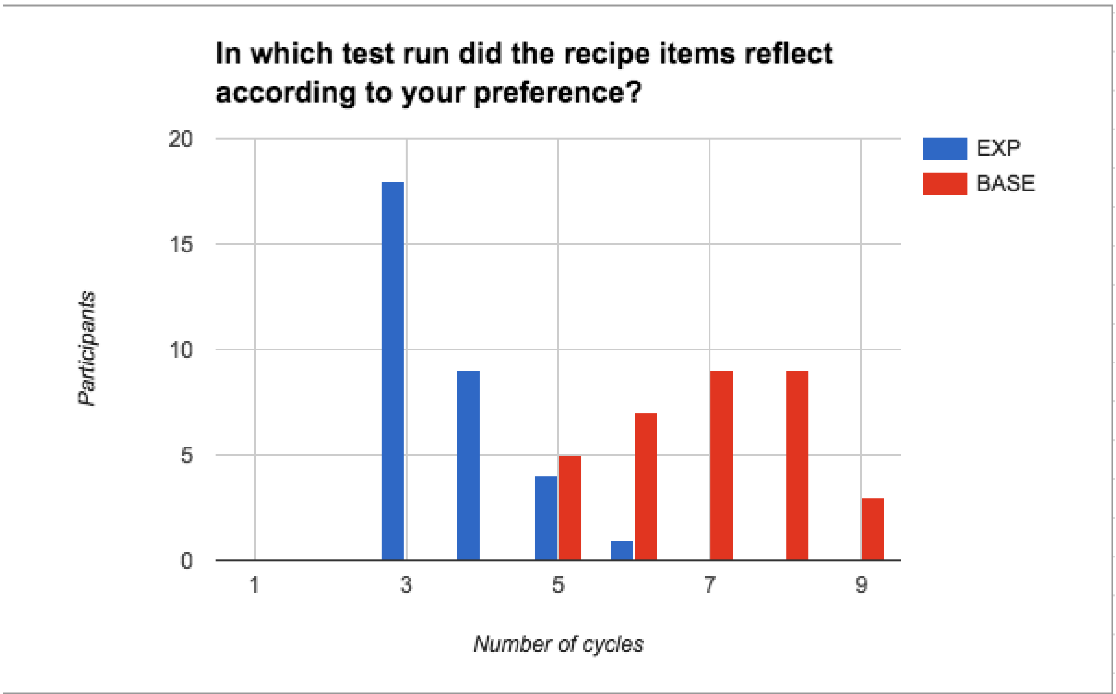


Figure 5.11.: Number of cycles

5.4.6. Perceived Satisfaction

Satisfaction is measures by inquiring users about their though while interacting with the systems. EXP have better results then EXP by having average of 4.313 against 2.469. While Standard deviation EXP σ = 0.8958 and BASE σ = 0.9153. By looking at one-tail t-test result we conclude that results are significant (p< 0.05 with p=4.50E-16). Analyzing the rating distribution in Figure 5.13 make it more clear.

Figure 5.12.: Perceived Efficiency

5.4.7. Perceived Context

Participants were asked which contextual information they considered more while selecting a recipe according to preferences. Contextual factors are categories in two major section Calories information, which further divided calories, contains by recipe itself and amount of exercise need to bee done to burn such calories. On the hand context like cooking time and course of the recipe are consider. Considering the result showing in Figure 5.14 participants that consider calories information in their diet are 75% which is majority of population. While more then half population of participant considers recipe course and preparation of recipe.

5.4.8. Informal Feedback

Our user study was not about feeling questionnaire, Informal feedback given by participant were also encourage and considered as valuable. It is important mention that each feedback was taken separately to each user based on their experience with other apps. Mainly user feedback were around what they feel are missing in our system and existing once. However it fairly possible that other may disagree with these options. Our

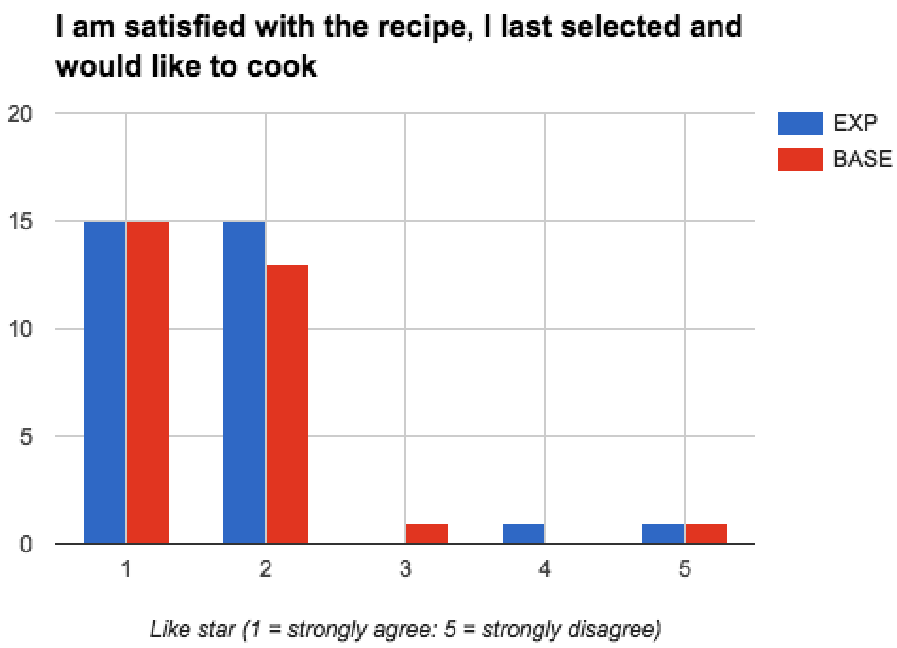


Figure 5.13.: Perceived Satisfaction

discussion in this section focuses on most mentioned and participant's interest feedback.

Explanation Enhancements

- Some participants mentioned that it would be more helpful for them if the context in a explanation are highlight. It will allow them to take a decision without closely reading the whole paragraph.
- While some people think that it would be great if explanation of tells recipe us about how does it taste for instances bitter, spicy, mild.
- Additionally, fewer people said that it would be nice to have origin of recipe in explanation part. Although it has been shown on top header but considered this section, they accept all the information about the recipe.

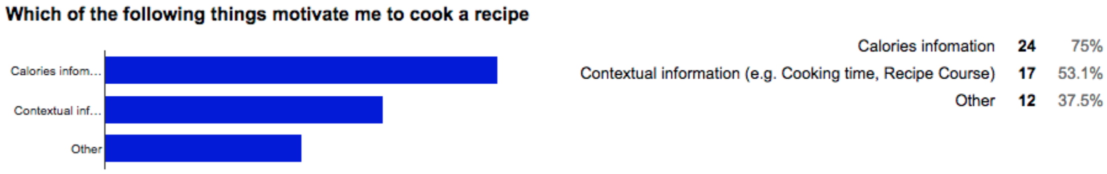


Figure 5.14.: Perceived Context

More Attributes

- Majority of participants suggested after seeing calories information in explanation section, they would love to have a feature that would keep track of deity information by allowing them to set a target like how much they want to put on or lose weight.
- Some participant added the above clause by suggesting that there should be a mechanism that will recommend us a weekly diet plan.
- Fewer participants were raised thought that app should consider also what ingredient they have and what recipes they can cook along with that items.
- In addition to above point 1 or 2 participants suggested idea the app will attract them more if they can edit a recipe and contribute their recipes to the system.

Endless set of recommenadations

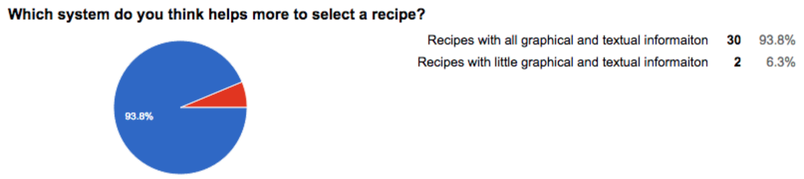
Quite few users reported, instead of given them specific amount recipes it would be quit helpful if app should provide a pagination functionality so that they can scroll and are able to more and more recipes. By providing us examples of some available products.

Allow sorting

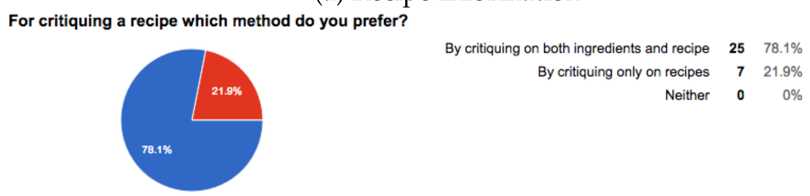
Fewer participants felt that the app should allow them a functionality to sort the recipes according to type for example, sorting based on popularity and reviews.
Finally, it was observed that as if there are so many ingredients that are liked by users, he can get much or more specific result with critiquing again, that he or she looking for because of limit amount of item set. To solve this problem user should provided endless recommendation set.

5.4.9. User Preferences

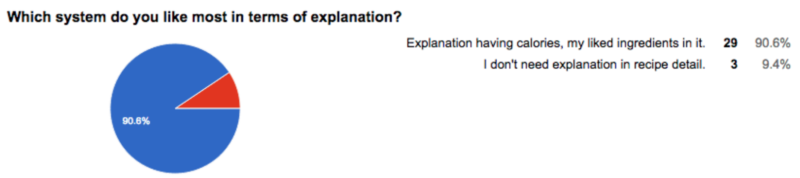
Last part of evolution was to ask the participant which variant they liked most why. More then 90% of participants were in the favor of EXP over BASE. Reasons were BASE has allowed them to critique only the recipes because of this critiquing cycles were increased to 2X to 3X. Furthermore, participants liked to know what the system thinks about their preference and why this recommendation is for them EXP allows them where BASE don't. Last but no the least only providing the recipes details is not sufficient for user, People want more and more information about the item which they select the more information they have the better they previse.



(a) Recipe Information



(b) Recipe Critique



(c) Explanation

Figure 5.15.: User Preferences

5.5. Discussion

After elaborating on the measurements, now each point will be further analyzed to explain the observed results and share the lessons learned for future improvements. Starting our discussion with persuasion, EXP has more positive feedback compared to BASE. As per results 90% of participants in the favor of EXP. Let's narrow down the

discussion to each Principe of Persuasion. EXP achieved more then 93% of *Reciprocity* than BASE. Participant thinks that preferred information that requires selecting a recipe should contains other review about the recipes and user-centric approach, which involved all textual and graphical information about the recipes. Overall *Scarcity* is achieved by the system s 78.8% by filtering out the recipes with respect to consumption time. *Authority* is perceived by 93.8%, participants feel star rating helps them in recipe selection. *Liking* 78.1% people in the favor of EXP which means they want to critique on recipe as well as ingredients. EXP leads BASE in terms *Commitment*, 90.7% of user thought that explanation of recipe motivate them to cook the recipe. Lastly, *Social Proof* achieves 84.4%. Overall looking into the result, we can draw our conclusion that EXP is far more persuasive then BASE. However in order to system more persuasive system should consider more health factors.
In terms of transparency, both system performs well there is no clear winner. Considering the fact system is simpler when it allows small number of items to critique on. However by looking into the results of conducted serve, mostly participant were able to tell the system's behavior. Moreover, EXP seemed to be transparent to participants, because it provides an explanation. In this case we can draw a conclusion that EXP is superior than BASE.
User felt more control over EXP because it provides explanation through which user could grab an idea what factors are involved in making this recommendation. In addition to EXP allows user to correct the wrong assumption that has been made by the system, which relates to the goal scrutablility. While in BASE it was pretty hard to correct the wrong assumption since it dealt which overall weight of the recipes.
Efficiency has been observed in two ways number of critique cycle that user need to perform to get the particular recipe and overall efficiency of the system in which we measured how much efforts required for a user to find a recipe. In terms of number of cycles EXP provide more efficient result in shorter number of cycle than BASE. Since the critiquing criteria followed by BASE is easy but lead to large number of cycle. Where as intuitive critiquing design of EXP allow users to critique on recipes as well as ingredients which lead to shorter critiquing cycle and far better result then BASE. However people find system is efficient in both variant in finding recipes but comfortable in EXP more than BASE. This might due explanation and over all improved design.
Level of satisfaction of EXP user is remarkably higher than BASE. Although both variants got the positive feedback but due to slick and user-centric design of EXP, its immediately captures the user attention.

Considering the contextual information about the recipe. It seemed like people are more focus towards that system which include calories information along with cooking time and course selection. BASE has got 53.1% of positive comment but EXP wins again by gain 23% more votes then BASE. Overall votes for EXP were 75%. Overall, participant prefers EXP over BASE and found more persuasive in nature. Users were comfortable about the explanation about the system and want some more information in it. They found system transparent, efficient, satisfied and provide significant control to them. However, to make system more appealing some changes need to be implemented which includes, consideration of health, sorting, pagination in recommendation list.

6. Conclusions and Future Work

This thesis investigated, the development and impact of Persuasion along with Active-Learning critique-based mobile recommender system in lifestyle domain. The developed concept proposed the factors of Persuasion Principle, generation of explanation not only to make the system transparent but also have an impact of persuasion in it and also provide user controller in the process of recommendation.

The system distinguishes user context and recommendation explanations. Where user preferences and context tells the system about current user interest and explanation justifies recommendation item relevance. For the getting the user context cooking time and recipe course was consider. While explanation part considers the user current context, vector of weighted recipe and liked and disliked ingredients. To make recommendation more persuasive health factor was introduced in explanation which contains the information about calories that recipes contains and how much user have to perform work out to burn such calories. Finally the recommendation are made more interactive and follows the user-centric design and systems provide user option to correct the wrong assumption made by the system.

An iOS mobile application was develop to measure the applicability of concept. Which should followed the user-centric design approach, having consisting UI like other market product. Prototype was presented to 32 real for evaluation. The purposed concept was able to achieve significantly improved result in terms of persuasion, explanation and critiquing on recommend items. In general users of the application appreciated the interface design and concept. Additional to this user are satisfied and want to cook the recommended recipe. In short proposed system was highly acceptable by the system and people want to have such application in market.

6.1. Future Work

Future development may include the creation of more complex recommendation scenarios to test the capabilities of proposed concept. Users are not only allowed to add recipes into the system but also add different cooking method of same recipe by trying our different ingredients. In addition to this more health factor need to be consider in the generation of recommendation like BMI (Body Mass Index), current health status (like diseases and allergy that use may have) and integration of iOS health app so that user can track his health status. To make recommendation more effective system should be consider Healthy Eating Index and nutrition information of each ingredient in recipe.

On algorithmic side research might be to add machine-learning algorithm that would help system to generate more accurate results that would be more effective to the user and produce more diversify results. Which the help of this it would be more easy for the system to generate weekly dietary plan for user and recommend him. User can add his wishlist regarding recipes and system by looking his profile recommend him recipes. Furthermore, if user have some food restriction due to his health system would be recommend his alternative to that recipe ingredients and present to user if it is good for him.

Other improvement might be to make the system pro-active by considering more contexts like location. Suppose on the ways to home form gym system recommend user some drinks that may be important to his health. Last important improvement be to test the application in real world scenario which more users and system is developed on cloud and make the system distributed to measure the effectiveness of developed system in terms of time and cost.

A. Appendix

Algorithm 1 PSO

```
Initialize a population of particles with random values positions and velocities from D dimensions in the search space
while Termination condition not reached do
    for Each particle i do
        Adapt velocity of the particle using Equation ??
        Update the position of the particle using Equation ??
        Evaluate the fitness f(X_i)
        if f(X_i) < f(P_i) then
            P_i ← X_i
        end if
        if f(X_i) < f(P_g) then
            P_g ← X_i
        end if
    end for
end while
```

List of Figures

List of Tables

Bibliography

[1] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles. "Towards a better understanding of context and context-awareness." In: *Handheld and ubiquitous computing*. Springer. 1999, pp. 304–307.

[2] G. Adomavicius and A. Tuzhilin. "Context-aware recommender systems." In: *Recommender systems handbook*. Springer, 2011, pp. 217–253.

[3] G. Adomavicius and A. Tuzhilin. "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions." In: *Knowledge and Data Engineering, IEEE Transactions on* 17.6 (2005), pp. 734–749.

[4] B. Bruegge and A. H. Dutoit. *Object-Oriented Software Engineering Using UML, Patterns and Java-(Required)*. Prentice Hall, 2004.

[5] S. Burigat, L. Chittaro, and L. De Marco. "Bringing dynamic queries to mobile devices: a visual preference-based search tool for tourist decision support." In: *Human-Computer Interaction-INTERACT 2005*. Springer, 2005, pp. 213–226.

[6] R. Burke. "Hybrid web recommender systems." In: *The adaptive web*. Springer, 2007, pp. 377–408.

[7] R. B. Cialdini and L. James. *Influence: Science and practice*. Vol. 4. Pearson education Boston, MA, 2009.

[8] P. Cremonesi, F. Garzotto, and R. Turrin. "Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study." In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 2.2 (2012), p. 11.

[9] P. Dourish. "What we talk about when we talk about context." In: *Personal and ubiquitous computing* 8.1 (2004), pp. 19–30.

[10] A. Felfernig and B. Gula. "An empirical study on consumer behavior in the interaction with knowledge-based recommender applications." In: *E-Commerce Technology, 2006. The 8th IEEE International Conference on and Enterprise Computing, E-Commerce, and E-Services, The 3rd IEEE International Conference on*. IEEE. 2006, pp. 37–37.

[11] B. J. Fogg. "Persuasive computers: perspectives and research directions." In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co. 1998, pp. 225–232.

[12] J. Freyne and S. Berkovsky. "Recommending food: Reasoning on recipes and ingredients." In: *User Modeling, Adaptation, and Personalization*. Springer, 2010, pp. 381–386.

[13] S. Gkika and G. Lekakos. "The persuasive role of Explanations in Recommender Systems." In: *Proceedings of the Second International Workshop on Behavior Change Support Systems (BCSS2014), Padova, Italy*. 2014.

[14] J. L. Herlocker, J. A. Konstan, and J. Riedl. "Explaining collaborative filtering recommendations." In: *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. ACM. 2000, pp. 241–250.

[15] S. Hessinger. *60 Percent of Online Traffic Now Comes From Mobile*. `http://smallbiztrends.com/2014/07/online-traffic-report-mobile.html`. Accessed Jul 8, 2014.

[16] M. Kaptein, B. De Ruyter, P. Markopoulos, and E. Aarts. "Adaptive persuasive systems: a study of tailored persuasive text messages to reduce snacking." In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 2.2 (2012), p. 10.

[17] M. N. Ko, G. P. Cheek, M. Shehab, and R. Sandhu. "Social-networks connect services." In: *Computer* 43.8 (2010), pp. 37–43.

[18] B. Lamche, U. Trottmann, and W. Wörndl. "Active learning strategies for exploratory mobile recommender systems." In: *Proceedings of the 4th Workshop on Context-Awareness in Retrieval and Recommendation*. ACM. 2014, pp. 10–17.

[19] H. J. Lee, J. Y. Choi, and S. J. Park. "Context-aware recommendations on the mobile web." In: *On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops*. Springer. 2005, pp. 142–151.

[20] T. Mahmood and F. Ricci. "Towards Learning User-Adaptive State Models in a Conversational Recommender System." In: *LWA*. Citeseer. 2007, pp. 373–378.

[21] L. McGinty and B. Smyth. "On the role of diversity in conversational recommender systems." In: *Case-based reasoning research and development*. Springer, 2003, pp. 276–290.

[22] S. M. McNee, S. K. Lam, J. A. Konstan, and J. Riedl. "Interfaces for eliciting new user preferences in recommender systems." In: *User Modeling 2003*. Springer, 2003, pp. 178–187.

[23] D. McSherry. "Explanation in recommender systems." In: *Artificial Intelligence Review* 24.2 (2005), pp. 179–197.

[24] T. Nanou, G. Lekakos, and K. Fouskas. "The effects of recommendations' presentation on persuasion and satisfaction in a movie recommender system." In: *Multimedia systems* 16.4-5 (2010), pp. 219–230.

[25] J. Nielsen and R. Molich. "Heuristic evaluation of user interfaces." In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 1990, pp. 249–256.

[26] P. Pu and L. Chen. "Trust building with explanation interfaces." In: *Proceedings of the 11th international conference on Intelligent user interfaces*. ACM. 2006, pp. 93–100.

[27] P. Pu, L. Chen, and R. Hu. "Evaluating recommender systems from the user's perspective: survey of the state of the art." In: *User Modeling and User-Adapted Interaction* 22.4-5 (2012), pp. 317–355.

[28] A. M. Rashid, G. Karypis, and J. Riedl. "Learning preferences of new users in recommender systems: an information theoretic approach." In: *ACM SIGKDD Explorations Newsletter* 10.2 (2008), pp. 90–100.

[29] P. Resnick and H. R. Varian. "Recommender systems." In: *Communications of the ACM* 40.3 (1997), pp. 56–58.

[30] F. Ricci. "Mobile recommender systems." In: *Information Technology & Tourism* 12.3 (2010), pp. 205–231.

[31] F. Ricci and Q. N. Nguyen. "Critique-based mobile recommender systems." In: *OEGAI Journal* 24.4 (2005), pp. 1–7.

[32] F. Ricci, L. Rokach, and B. Shapira. "Introduction to recommender systems handbook." In: *Recommender systems handbook*. Springer, 2011, pp. 1–35.

[33] N. Rubens, D. Kaplan, and M. Sugiyama. "Active learning in recommender systems." In: *Recommender Systems Handbook*. Springer, 2011, pp. 735–767.

[34] S. Schiaffino and A. Amandi. "Intelligent user profiling." In: *Artificial Intelligence An International Perspective*. Springer, 2009, pp. 193–216.

[35] B. N. Schilit and M. M. Theimer. "Disseminating active map information to mobile hosts." In: *Network, IEEE* 8.5 (1994), pp. 22–32.

[36] J. H. Schiller. *Mobile communications*. Pearson Education, 2003.

[37] G. Shani and A. Gunawardana. "Evaluating recommendation systems." In: *Recommender systems handbook*. Springer, 2011, pp. 257–297.

[38] R. Sinha and K. Swearingen. "The role of transparency in recommender systems." In: *CHI'02 extended abstracts on Human factors in computing systems*. ACM. 2002, pp. 830–831.

[39] L. Suchman and H.-M. Reconfigurations. "Plans and situated actions." In: *New York, Cambridge University* (1986).

[40] N. Suksom, M. Buranarach, Y. M. Thein, T. Supnithi, and P. Netisopakul. "A Knowledge-based Framework for Development of Personalized Food Recommender System." In: *Proc. of the 5th Int. Conf. on Knowledge, Information and Creativity Support Systems*. 2010.

[41] K. Swearingen and R. Sinha. "Interaction design for recommender systems." In: *Designing Interactive Systems*. Vol. 6. 12. Citeseer. 2002, pp. 312–334.

[42] K. Tanaka-Ishii and I. Frank. "Multi-agent explanation strategies in real-time domains." In: *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 2000, pp. 158–165.

[43] J. Tang, L. Yao, D. Zhang, and J. Zhang. "A combination approach to web user profiling." In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 5.1 (2010), p. 2.

[44] C.-Y. Teng, Y.-R. Lin, and L. A. Adamic. "Recipe recommendation using ingredient networks." In: *Proceedings of the 3rd Annual ACM Web Science Conference*. ACM. 2012, pp. 298–307.

[45] N. Tintarev and J. Masthoff. "A survey of explanations in recommender systems." In: *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*. IEEE. 2007, pp. 801–810.

[46] N. Tintarev and J. Masthoff. "Designing and evaluating explanations for recommender systems." In: *Recommender Systems Handbook*. Springer, 2011, pp. 479–510.

[47] N. Tintarev and J. Masthoff. "Evaluating the effectiveness of explanations for recommender systems." In: *User Modeling and User-Adapted Interaction* 22.4-5 (2012), pp. 399–439.

[48] M. Ueda, M. Takahata, and S. Nakajima. "User's food preference extraction for personalized cooking recipe recommendation." In: *Semantic Personalized Information Management: Retrieval and Recommendation SPIM 2011* (2011), p. 98.

[49] W. H. O. (WHO) et al. "Information available at http://www. who. int/mediacentre/factsheets/fs311/en/index. html." In: *Accessed November* (2008).

[50] B. Xiao and I. Benbasat. "E-commerce product recommendation agents: use, characteristics, and impact." In: *Mis Quarterly* 31.1 (2007), pp. 137–209.

[51] K.-H. Yoo, U. Gretzel, and M. Zanker. *Persuasive recommender systems: conceptual background and implications*. Springer Science & Business Media, 2012.