Informatics Institute of Technology
In collaboration with
University of Westminster, UK.

"Software Agent Based News Recommender System for an Online News Environment"

A dissertation by
Sriyal Himesh Jayasinghe

Supervised by
Ms. Naomi Krishnaraja

Submitted in partial fulfilment of the requirements for the

M.Sc in Advanced Software Engineering

Department of Computing

August 2014

Declaration

I hereby certify that this project report and all the artefacts associated with it is my **own** work and it has not been submitted before nor is currently being submitted for any degree programme.

Full Name of Student: Jayasinghe Arachchilage Sriyal Himesh Jayasinghe

Registration No : 2009077

Signature : ..... Date : .....

Abstract

News has grown to be an essential in the contemporary world and online news media plays a major role in the news domain by publishing the latest news and allowing the readers to access them from any place at any time. Online news companies generally sort news articles according to the time it is published and due to the high volume of news articles and instance updates most online readers face a limitation in finding news articles which would match their specific preferences. The only facility provided by most online news domains is to overcome this limitation is to provide a searching facility for the readers to manually search for preferred news articles which would not impress all readers due to the time and the effort that has to be invested in the searching process. Previous work done in the field of news recommendation has only looked into providing a solution either supporting a web based presentation or a mobile based presentation without looking in to the prospect of providing one presentation layer for both the mediums. Another drawback of the current solutions is not considering the scalability of the system as a key success factor which would have a negative impact on the performance of the system due to the vast amount of articles and readers of online news environments.

To overcome the above mentioned limitations of online news environments, NewsRec system was introduced which uses a hybrid recommender engine to achieve a high accuracy in the recommendation. NewsRec system was designed and implemented to have only one presentation layer to support any type of a device and platform and software agents were utilized to induce better collation and communication between different activities of the recommender system thus making the scalability of the system to increase.

All agents were implemented using JADE agent framework and a RESTful API exposed through NodeJS and Redis server were used to read and store data coming through the clients and Java Persistence and Hibernate to store the processed data. Implemented system was tested thoroughly under different conditions and the NewsRec system was evaluated by evaluators of various domains. Eventually, the test results attested that the analysis, design, implementation and documentation have been carried out in an effective and in an efficient manner.

Subject Descriptors:

H.3.3 Information Search and Retrieval

I.2.11 Multiagent systems

Key Words:

Information Filtering, Personalization, JADE framework

Acknowledgement

The effort that was put by me in to completing this project has been a very tough expedition throughout but at the same time it was a wonderful experience. However, it would not have been possible without the kind support, assistance and courage of many people who stood behind me. I would like to extend my sincere thanks to all of them.

I am highly indebted to my supervisor Ms. Naomi Krishnarajah, Head of the Department of Computer Science, Informatics Institute of Technology for her constant guidance and supervision as well as for encouragement and ideas given me to strive to greater heights and to achieve my fullest potential.

I hereby express my gratitude to Prof. Alfred Perera, Dean of the Informatics Institute of Technology and the management of Informatics Institute of Technology for providing me the opportunity for engaging this work.

I would like to express my special gratitude and thanks to everyone who assisted me by participating online surveys, interviews and contributing their ideas for requirements gathering and critical evaluation.

Finally I would like to express my great gratitude towards my wife for her kind co-operation and encouragement which helped me to complete the project successfully.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Definition
CF	Content Filtering
CBF	Collaborative Filtering
NLP	Natural Language Processing
OOADM	Object-Oriented Analysis and Design Method
SSADM	Structured Systems Analysis and Design Method
HTTP	Hypertext Transfer Protocol
SOAP	Simple Object Access Protocol
REST	Representational State Transfer
SRS	Software Requirement Specification
AI	Artificial Intelligence
RAM	Random Access Memory
JSON	JavaScript Object Notation
SOA	Service Oriented Architecture
JADE	Java Agent Development
UML	Unified Modelling Language

Chapter 1: Introduction

Contents

1.1 Chapter Overview

The purpose of this chapter is to provide an overview of the project undertaken. It starts by outlining a brief introduction along with the current limitations faced by the online news domain followed by justifying the importance of personalizing online news feeds to suit different reader preferences. Additionally previous work done in relation to designing and developing online news recommendation systems is evaluated along with the aim and objectives that were laid out for the successful completion of the project are also included. Furthermore the high level features of the system are described and the chapter concludes with an overview of the chapters contained in this report.

1.2 Problem Domain

News have always been a part of a human's life and even with today's busy life style people somehow would find at least a few minutes to read some sort of news feed which they have access to. Popularity of News feeds have changed over the course of time and according to the statistics obtained from a survey done for the annual report on American Journalism by Pew Research Centre (The state of the media, 2013) popularity of online news is on the rise and would be the most popular news feed in the very near future. Statistics derived from another survey done in UK by UK Telecoms (Techcrunch, 2014) claims that online news viewing has taken over the printed newspapers throughout UK in the year 2014.The above mentioned surveys highlight the main reason for this popularity as the wider use of digital devices such as smart phones and tablets which give the reader the freedom of accessing news at times that are convenient to the reader.

According to Yafooz, *et al* (2011)and Yuanhua, *et al* (2011)with the amount of news that get generated and circulated around the world each moment viewers/readers would face the problem of finding the most suitable news that suits their preferences. As argued by Wartenberg and Holmqvist (2005) news publishers should not consider that news publishing is a just process of publishing set of texts with images and expect the readers to find the most preferable news articles and read them. Major limitations of TV, Cable news feeds and printed news papers are that those sources lack the opportunity of presenting its readers/viewers with news that would match to the reader/viewers interests or providing a searching facility to search for preferred news items. As highlighted by Arora and Shah (2011) and Thurman and Schifferes (2012) even nowadays online news sites too contain huge amount of news articles and it is a tedious effort for a reader to filter out preferred news articles from a collection of more than 1000 articles that can be found in an online news service compared to a few dozen of articles of a physical newspaper. This inadequacy can ultimately lead to situations in where the reader waste time searching through a heap of news articles to find what interests the reader and end up with frustrations. As suggested by then Google CEO Eric Schmidt in an interview *"At its best, the on-line version of a newspaper should learn from the information I'm giving it -- what I've read, who I am and what I like -- to automatically send me stories and photos that will interest me "* (Times, 2009)online news environments should make an effort in presenting news in a more reader friendly manner by presenting more preferable articles to satisfy individual news reader interests rather than providing the same news feed for each user. But this will

pose a huge challenge for news publishers as they will have to manually identify each reader's best interests and then find articles that would match those interests and afterward publish those articles through a customized view for each reader.

Kompan and Bieliková (2010) highlights that the above mentioned limitations of where the readers have to manually search through a pile of news articles to find the most interesting articles and the news publishers have to make an extra effort in making news publication more personalized can be avoided in an online news environment by providing personalized news feeds to its readers with a use of an automated recommendation system which would learn each reader's different preferences and filter the most appropriate or suitable news feeds that matches the readers' interests. Research work done by Yuanhua, *et al* (2011) and Wang and Chen (2010) too emphasizes the need for up to date recommendation systems for online news environments.

While the majority of the work done on recommendation systems for various domains, including the online news recommendation domain has focused mainly on improving the accuracy of the prediction, very little effort being put into make recommendation system architectures scalable. Software architectures of available news recommendation systems are not scalable enough to support different key activities such as identification of reader preferences, clustering of readers with similar preferences, reading news feeds from different sources and ranking them, carrying out of various filtering phases discretely and then in combination work as one unit. Rafsanjani *et al* (2013) and Morais, Oliveira and Jorge (2012) highlight software architectures of present day recommender systems should be made more scalable by using software agent approach to induce better collation and communication between different activities of the recommender system.

As highlighted by Tavakolifard*et al* (2013) another aspect that has to be considered when developing online news recommendation systems is the rising trend of using mobile apps to logon to news sites. Nowadays smart phones come with different operating systems and architectures and any proposed news recommendation systems should have a scalable architecture which would support both standard web and different mobile platforms without having different formats of news presentations.

It is evident that by providing personalized news feeds to its readers which suit their interests' on a scalable approach online news environments can attract a wider audience and could use the news personalization as a critical success factor for its growth.

1.3 Previous Work

Since it is suggested that online news environments should provide personalization features to its readers, many research have been done in finding ways to facilitate personalization in online news environments. Google News (Google news, 2014) is a free news aggregator service provided by Google Inc which gathers the most up-to-date news from various news sources using an automatic aggregation algorithm for collecting news articles. This service provides a feature where the readers can search for news articles through content matching but it lacks a powerful reader personalization feature. The current personalization feature provided by this service will suggest news articles for readers based on the search history of the readers. Das *et al* (2007) have suggested a scalable online recommendation system for Google News based using collaborative filtering. This

research focuses mainly on making the system more scalable to support the large volume of news that is aggregated and tries to predict interest of the readers based on the user clicks. Bhatpara (2008) argues that this system has not taken content based filtering of news articles or article clustering in to consideration and suggests a hybrid recommender approach to improve the accuracy of the system which currently hangs around 30%.

Tatar *et al* (2011) proposes a solution for an online news recommendation system based on the number of comments received by a particular news article. According to the number of comments received by a particular news article within a time interval the system would try to predict an expected popularity. Then each news article will automatically be ranked according to the predicted popularity and the highest ranked news articles will be categorized under different categories and will be recommended to readers. Major drawback of this solution is that there are no mechanisms used to filter out positive comments from the negative comments resulting system suggesting articles even which are having a high ranking as a result of getting high number of negative comments.

Li *et al* (2011) has suggested a novel approach for news recommendation which uses a hybrid recommendation approach. This system tries to rate news articles based on the number of user clicks received for a particular news article. There are various other implicit and explicit user feedback parameters like reader rating of an article, no of likings given by readers for an article, no of sharing of the article on social networks that can be used in combination to calculate the preference for a particular article.

None of the above mentioned work has looked in to coming up with a scalable approach that would support both mobile and web approach for news presentation. A research carried out by Tavakolifard *et al* (2013)have proposed a mobile news recommender system but that system only supports iPhone platform and it hasn't considered in making the system architecture scalable enough to support any of the other smart phone platforms nor supporting a standard web view.

All of the above mentioned works haven't considered the possibility of using the software agent approach to increase the scalability of the systems by inducing better collation and communication between different activities of the recommender system.

Since all the above mentioned previously attempted solutions haven't come up with a software agent based different platform supportive recommendation system for recommending news articles from an online news environment this research attempts to address that inadequacy and fill it.

1.4 Project Aim

To design, develop and evaluate a software agent based recommender system to recommend news articles of selected categories to suit individual preferences of readers who would use different types of devices with different platforms to access news articles of an online news environment

Further elaborating on the aim, this project will produce a software agent based recommendation system for an online news environment that would recommend news articles of selected categories like sport or entertainment

that would suit the preferences of the reader. Preferences of a user will be understood by the system through explicit and implicit interactions performed by the user and filtering of news articles will be done by different recommender strategies. The system would support different type of devices like smart phones, tablets and PCs or laptops with different platforms without the system having to have different presentations to suit each device or platform.

The proposed system will not cover,

- Research in to incorporating news feeds provided by 3rd party news agencies like BBC.
- Research in to reading and extracting keywords from a news article
- Research in to eliminating any drawbacks of the tools used for the development

1.5 Project Objectives

In order to achieve the above mentioned aims of the project successfully the following objectives have been identified.

Objective 1	Prepare Terms of Reference
Prepare the Terms of Reference which defines the aims and objectives, features of the prototype, background, resource requirements, project deliverables and activity schedule which will be a guide throughout the project and evaluation process.	
Objective 2	Literature Survey
Carryout an in-depth literature survey on the following subject areas ▪ Characteristics of online news environments – to understand how an online news environment works ▪ Existing online news recommendation systems – to understand the techniques, technologies and algorithms used in existing online news recommendation systems along with their strengths and weaknesses ▪ Various recommender and clustering algorithms – to determine the most appropriate recommender and clustering concepts and algorithms that can be used to develop an online news recommender system ▪ Software agents – to understand the software agent concept and determine how best the agent approach can be used in an online news environment to make the system more scalable ▪ Different development technologies, frameworks, libraries, APIs, tools and methodologies to determine the most appropriates to develop the system	
Objective 3	Selection of a software development methodology
Select a software development methodology that would be most suitable to carry out the various phases of the project	
Objective 4	Requirement gathering process

<table>
<tr><td colspan="2">Carryout an in-depth user requirement gathering phase with
End users of an online news recommendation system through a questionnaire and observations of news site accessing to identify the end user requirements and behaviours.
Domain experts of an online news field to identifying standards of news categorization and current processes adapted to categorize news articles along with their suggestion for online news recommender system.
Personal evaluation of the existing online news sites and online recommender systems available on other domain to verify the end user requirements and domain expert ideas and identify any possible new requirements</td></tr>
<tr><td>Objective 5</td><td>Prepare a software requirement specification (SRS)</td></tr>
<tr><td colspan="2">Using the information gathered through the literature review, end user questionnaire and observations, domain expert interviews and personal evaluations prepare the software requirements specification to document the functional and non-functional requirements of the proposed system</td></tr>
<tr><td>Objective 6</td><td>Selection of software and hardware resources</td></tr>
<tr><td colspan="2">Select the most appropriate technologies, tools, APIs, libraries, platforms, algorithms and hardware requirements to implement the prototype</td></tr>
<tr><td>Objective 7</td><td>Prepare software design specification</td></tr>
<tr><td colspan="2">Prepare the design specification for the prototype according to the analyzed requirements gathered from the requirement gathering phase and selected software and hardware resources</td></tr>
<tr><td>Objective 7</td><td>Develop the prototype</td></tr>
<tr><td colspan="2">Develop the prototype using the most appropriate software and hardware resources to full fill the user requirements identified in the requirement specification.</td></tr>
<tr><td>Objective 8</td><td>Testing of the prototype</td></tr>
<tr><td colspan="2">Formulate a testing plan, prepare test cases and conduct an in-depth testing of the system to identify bugs and check whether the required functional and non-functional requirements of the users are achieved from the developed prototype</td></tr>
<tr><td>Objective 9</td><td>Evaluation of the work carried out</td></tr>
<tr><td colspan="2">Carry out a critical evaluation of the prototype using selected user groups of the system and conduct a review of the evaluation findings to determine how far the project has successfully addressed the hypothesis
Carry out a review with domain experts of different research areas used in the prototype to identify areas for future improvements
Perform a personal evaluation to self asses the work carried out</td></tr>
<tr><td>Objective 10</td><td>Documentation</td></tr>
<tr><td colspan="2">Document all findings and key steps involved in the project and submit the project report</td></tr>
</table>

Table1.1- Objectives

1.6 Feature of the Prototype

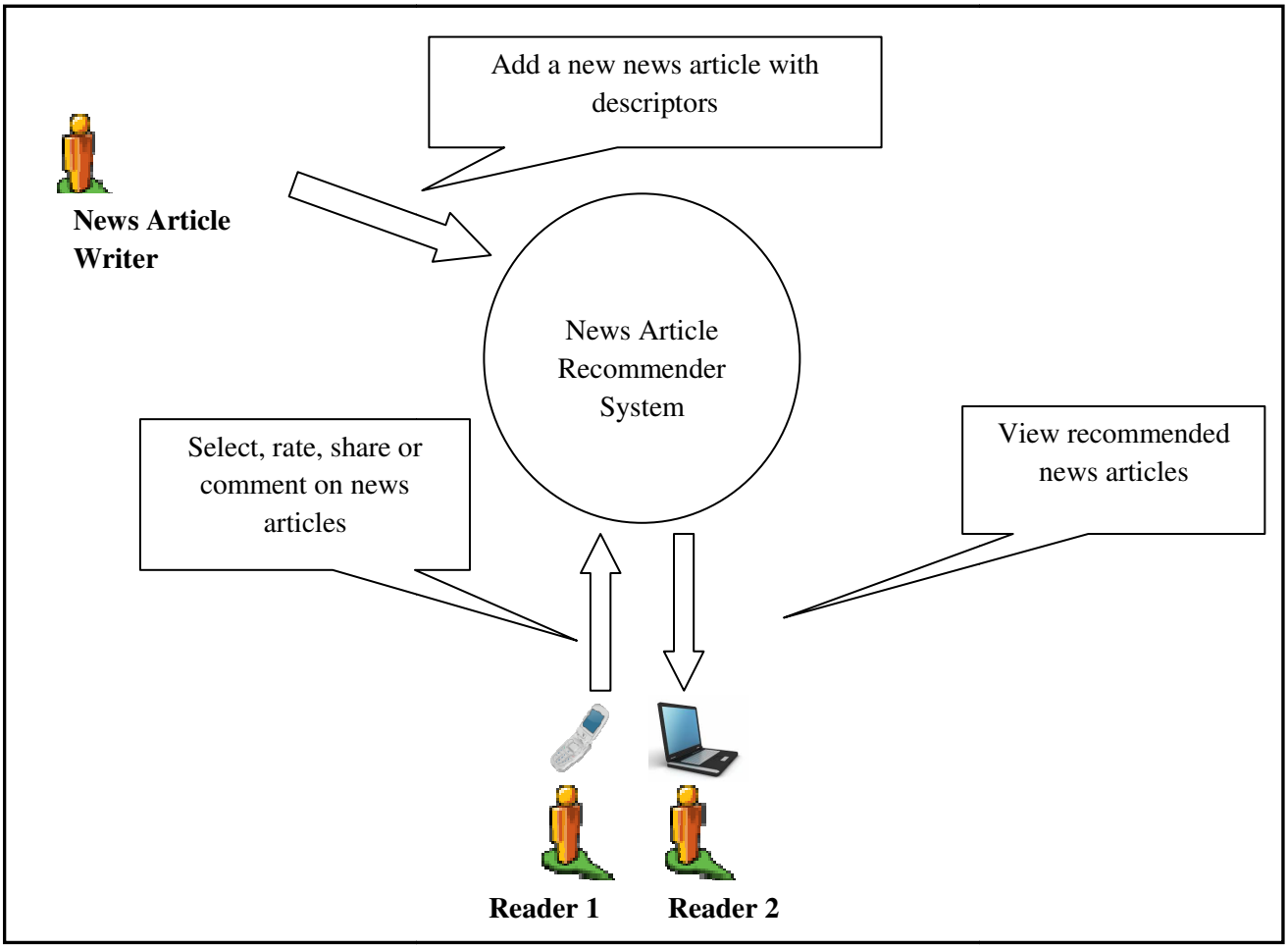


Figure 1.1- Flowchart describing the basic flow of the proposed system

1.7Resource Requirements

*These requirements are subjected to changes

Software Requirements	Hardware Requirements
• Java SDK 1.6 • JADE Framework • NodeJS Library • NetBeans 8.0 IDE • MySql DBMS • Microsoft office package for documenting the report • Rational Rose for documenting the design	• Core i5 2.13 GHz processor • 4 MB DDR2 RAM

Table 1.2 - Software and Hardware requirements

1.8 Project Document Structure

Final document of the project will be containing following chapters. References and Appendix section will be followed by the below mentioned chapters.

Chapter 2 – Literature Review

The Literature review chapter will consist of an in depth review carried out to understand the impact the strengths and weakness of various recommender types would have on the accuracy and the scalability of a recommender system, common problems faced by all types of recommendation systems in relation to accuracy and scalability and previous work done to overcome those problems.

Chapter 3 - Project management

Project management chapter will describe about the project management aspect of the project. It will contain the project plan and risk mitigation plans along with a discussion about the most suitable software development methodology for the project

Chapter 4- Requirements Specification

Requirement specification chapter will cover the process undertaken for stakeholder identification, methods adopted for the requirement elicitation from the stakeholders and the processes undertaken to identify the functional and non-functional requirements.

Chapter 5- System Architecture and Design

System Architecture and Design chapter will contain design decisions, design goals, high level architecture of the system along with various design diagram done to model the proposed system.

Chapter 6- Implementation

Implementation chapter will provide a detail description about how each functional and non-functional requirement was implemented, any problems and challenges encountered during the implementation stage and measures that were taken to overcome those challenges and problems.

Chapter 7- Testing

Testing chapter will contain details about the testing phase of the project. Starting from unit level testing it will go until actual prototype feature testing. All the test results gathered from above mentioned testing will be documented and analyzed.

Chapter 8- Evaluation

Evaluation chapter will be having details about the project evaluation. Expert evaluations, self-evaluation and reviews about those evaluations will be available in this chapter.

Chapter 9 – Conclusion

Analyses how the objectives were successfully achieved followed by modules contributed and problems encountered. Eventually future enhancements and concluding remarks are described.

Chapter 2: Literature Review

Contents

- Chapter Overview
- Recommendation Systems
- Key success factors of recommender systems
- Recommender systems architecture
- Cold starter problem of a new user
- Change of User preferences
- Scalability of recommendation systems
- Review of existing systems

2.1 Chapter Overview

Previous chapter contained a brief introduction in to the project, by presenting the problem domain along with previous work done on the news recommender domain, project aim and objectives which are to be achieved for the successful completion of the project. Literature Review chapter will discuss the effectiveness of automated news recommendation for an online news environment along with a critical evaluation of the related work done on the problem domain. Then the chapter will present the reader with a critical review of the possible techniques, technologies, algorithms, methodologies and various already available tools can be used to develop the proposed news recommender system in an effective and efficient manner.

2.2 Recommendation Systems

As highlighted by Zeng*et al* (2013) information filtering systems assisted with different types of user interactions would try to filter the best matching information that a user is likely to find interesting or useful. When an information filtering system presents the filtered information in the form of suggestions such information filtering systems are called as recommender systems. Recommender systems would collect details, behaviours and patterns of users in implicit and explicit ways to better understand users' preferences and then use that knowledge to filter out the best matching content to impress users. Over the years usage of recommender systems has increased significantly in commercial domains Linden, Smith and York (2003) and Koren, Bell and Volinsky (2009) has managed to ignite a huge interest in the academic and research domains to find out better ways recommender systems can be utilized in domains such as e-learning, movie, music, news, book, research article recommendation.

2.3 Key Success Factors of Recommender Systems

Naturally in any recommendation system accuracy would be considered as the main key success factor but researches by Felfernig and Burke (2008), Ziegler *et al* (2005), Beel *et al* (2013) and Cosley *et al* (2003) have highlighted several factors like diversity, recommender persistence, privacy, robustness, serendipity and system scalability as other key success factors for a successful recommendation system. Though most of the work done on recommendation systems has emphasized on improving the accuracy of the recommendations Basilico and Hofmann (2004) has argued that scalability is as important as accuracy for a recommendation system. As time goes on most recommender systems have failed to contend with the massive amount of items and users that it would have to contend with. As highlighted by Birukov, Blanzieri and Giorgini (2005) accuracy of a recommender system is referred to as the ability of the system to precisely predict items to match different user preferences and scalability is referred to as the ability to cater the needs of a large number of users and items. Unfortunately these two critical success factors have a negative correlation between them, where when one increases the other decreases and it's a difficult task to maintain both of them at an acceptable level. This research focuses mainly on identifying the best approaches for building a recommender system for an online news site considering accuracy and scalability as key success factors.

2.4 Recommender System Architecture

According to Ghazanfar and Prugel-Bennett (2010) any recommender system will be consisting of four basic entities known as the user, item, user profile and recommender engine. Table 2.1 provides an overview of the role played by those 4 entities in a recommender system along with a description for each entity.

Entity	Description & role played in a recommender system
User	The user entity can be a person or an external system that would receive recommendations from the recommender system
Item	Item is an entity that would be recommended to users by the recommender system.
User Profile	Information such as the preference knowledge of a user about different items and personal information of the user which can be used for the recommendation process is stored in an entity known as a user profile
Recommender Engine	This will be the core engine of any recommender system and recommendations will be carried out by aligning the rating value of the items which is calculated based on the item description with the user preferences

Table 2.1 - An overview of recommender System components and their roles

Hanani, Shapira and Shoval (2001) have identified that a recommender engine would be consisting of 3 components known as the user profile learner, item content analyser and the filtering engine. Table 2.2 provides an overview of the role played by those 3 entities in a recommender engine along with a description for each component.

Component	Description & role played in a recommender engine
User profile learner	User profile learner would try to learn the preferences of a user based on the user explicit and implicit interactions and the captured user knowledge along with any user demographic data will be stored and in a user profile and be maintained by the user profile learner.
Item content analyser	Item content analyser would try to rate an item based on the content descriptions of the item combined with any user preferences shown for that item.
Filtering engine	Filtering engine would try to filter or align items which would suit the user preferences the most by using various filtering techniques.

Table 2.2- An overview of recommender System components and their roles

According to Ricci, Rokach and Shapira (2011) recommender systems can be classified into 3 basic categories based on the filtering approach used. They are content based filtering systems (CBF), collaborative filtering (CF) systems and hybrid filtering systems.

2.4.1 Content-Based Recommender Systems

Content-based filtering, also known as cognitive filtering, recommends items to users based on similarities between the content of items and a user profile (Van Meteren and Van Someren, 2000).The content of an item is characterized as a set of terms known as descriptors which describes the nature of the item and the user profile contains information of the content of the items of which the user has previously shown an interest on.

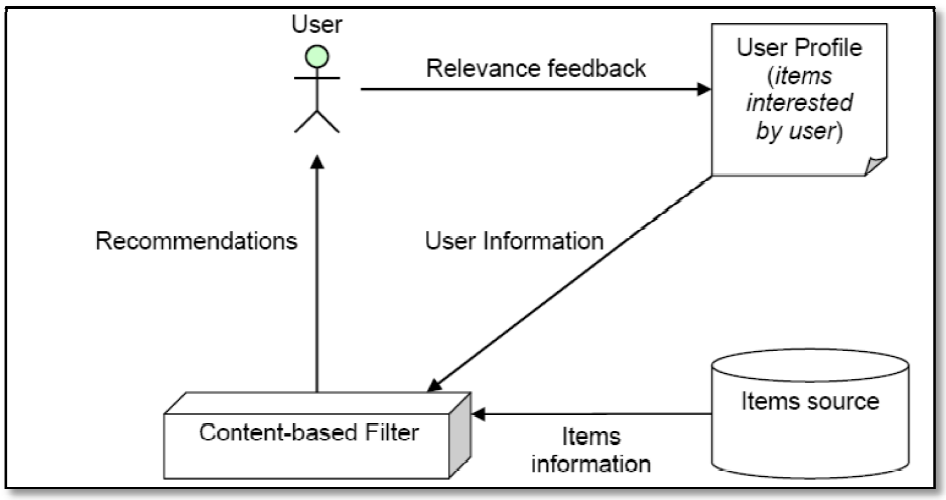


Figure 2.1- Architecture content-based recommender systems - (Adomavicius, G.; Tuzhilin, A., 2005)

Adomavicius, G. and Tuzhilin, A. (2005) have highlighted two key aspects that have to be considered when implementing a content-based filtering system. Table 2.3 provides an overview of those two aspects.

Aspect	Meaningful representation of item descriptors & the user profile information for easy and better comparison
Description	Since the accuracy of the recommendations depends on the depth of the successful alignment between the user profile information and item descriptors, both should be represented in a way that it would be easy for comparison and provide a better accuracy
Aspect	Approach used to assign descriptors to an item
Description	In content based recommendation systems the content descriptions should be assigned to each item at the time adding the item to the item repository. The assignment process can either be an automated or a manual process

Table 2.3– Meaningful representation of item descriptors & the user profile information for easy and better comparison

Item descriptors & the user profile information representation approaches

Gauch*et al* (2007) have identified 3 approaches that can be utilized to model item descriptors and user profile. Tables 2.4, 2.5 and 2.6 provide an overview of those two approaches along with the impact each would have on the accuracy and the scalability of the system.

Approach	Keyword profiles
Description	In this approach user profile is built using a set of keywords extracted from web pages visited, bookmarked, saved or explicitly provided by the user.
Impact on accuracy	Since it can't match all possible polysemy, synonymy, and multi-word representations of a keyword the accuracy will be limited
Impact on scalability	With a text based representation the scalability will increase as it requires only a text matching module for the comparison

Table 2.4 - Overview of keyword based profiles (self composed)

Approach	Semantic network profiles
Description	In this approach user profile is represented by a weighted semantic network in which each node contains a particular word found in the corpus and arcs are created representing co-occurrences of the two words in the connected nodes
Impact on accuracy	This approach would comparatively have a higher accuracy compared to the keyword based profiles due to its ability to weight relationships among words.
Impact on scalability	The computational resources required for building and maintain the semantic relationship network will be higher than keyword based profiles

Table 2.5- Overview of Semantic Network Profiles (self composed)

Approach	Concept profiles
Description	This approach is similar to semantic network-based profile with nodes and arcs but the nodes represent abstract topics considered interesting to the user, rather than specific words or sets of related words.
Impact on accuracy	This approach would provide the highest accuracy for the recommendation process when compared with the previously discussed two approaches
Impact on scalability	Since this approach suggests hierarchical concepts, rather than a flat set of concepts, creating a broad and deep concept hierarchy is an expensive process and will lead to negative impact on the scalability of the system

Table 2.6- Overview of Concept Profiles (self composed)

Since the proposed system is intended to be developed for an ordinary online news site an approach which would provide an intermediate level of support for the accuracy of the recommendation and the scalability of the system have to be chosen. When reviewing the 3 approaches that can be utilized to model item descriptors and user profiles it can be distinguished that though the keyword based approach comparatively provides the utmost flexibility for the scalability of the system it can't be suggested for the proposed system due to its negative impact on the accuracy of the recommendations. Though the concept profile approach provides the highest positive impact on the accuracy of the recommendation process it too can't be suggested as the most

suitable approach for the proposed system due to its negative impact on the scalability of the system. As a result semantic network profile can be suggested as the most suitable approach to model item descriptors and user profiles of the proposed system which provides a mid-level support for the accuracy of the recommendation and the scalability of the system.

Item descriptor assignment approaches

delOlmo and Gaudioso (2008)have identified two approaches that can be used to assign descriptors to an item and tables 2.7 and 2.8 provide an overview of those two approaches along with the impact each would have on the accuracy and the scalability of the system.

Approach	Automated descriptor assignment	
Description	In this approach the system would try assign descriptors to an item using automated processes like NLP or ontology alignment	
Advantages		**Disadvantages**
1.Less user interaction would make the system more user friendly. 2. Higher accuracy rate compared to manual descriptor assignment approach		1. Requirement of additional software modules to read and extract the correct item descriptors. 2. Increased software coupling with 3rd party environments. 3. Increase on software maintainability and operational costs. 4. Comparatively it would require more time to assign descriptors.
Impact on accuracy	The accuracy will be increased if an automated approach like NLP or ontology alignment is used as such approaches would help to solve the limitations of polysemy, synonymy, and multi-words.	
Impact on scalability	The scalability of the system would decrease in an automated approach as for each domain or category of item a separate NLP & ontology engine would be required	

Table 2.7- Automated descriptor assignment approach (self composed)

Approach	Manual descriptor assignment	
Description	In this approach the entity who would add the item to the item repository	
Advantages		**Disadvantages**
1. Ease of development and maintainability of the system. 2.Ability to use in uncommon domains 3.Comparatively it would require less time to assign descriptors.		1. May reduce the user friendliness of the system. 2. Lower accuracy rate compared to automated descriptor assignment approach
Impact on	In a manual process it would be hard to identify possible polysemy, synonymy, and	

accuracy	multi-word representations to a higher degree and it will have negative impact on the accuracy of the system
Impact on scalability	The scalability of the system would increase with the use of a manual approach as it won't require any additional modules

Table2.8- Manual descriptor assignment approach (self composed)

When comparing the approaches that can be used to assign item descriptors to an item it can be observed that the accuracy of automated assignment process is higher than the manual approach. But while trying to achieve that accuracy level it would have to face serious implications on the scalability of the system. In online news environment there would be many categories and sub categories of news and maintaining a separate NLP and ontology engine for each sub category would be practically infeasible for an ordinary news website. The significance of a news article will be high as soon as the incident took place and an automated approach would have an impact on the latency of the article and the best individual who would know the most suitable descriptors for an article would be the editor of that article. Due to the above mentioned reasons it can be suggested to utilize the manual approach of assigning descriptors to items to the proposed system.

Advantages and limitations of content based recommendation systems

Melville, Mooney and Nagarajan (2002) have highlighted several advantages and limitations of a content based recommendation system. Table 2.9 provides an evaluation on the impact the advantages would have on the accuracy and the scalability of the system and table 2.10 provides an evaluation on the impact the disadvantages would have on the accuracy and the scalability of the system.

Advantage & description	Impact on accuracy	Impact on scalability
User independence - Content-based recommender systems have the capability of building a user profile solely based on the feedback provided by an active user	This wouldn't have a major negative impact on the accuracy as the accuracy of the recommendations would only be restricted to the gathered knowledge. But the system would not be able to recommend any items on a different category to the user.	The scalability of the system will be high as the recommendation and user preference learning processes will be simpler and not resource intensive
Transparency – Content-based recommender systems can provide explanations for the recommended items by listing content descriptions that caused an item to be recommended	The feedback of a selection can easily be mapped to better understand the preferences of a user which will be helpful to carry out more accurate recommendations	The scalability of the system would not be highly challenged as it requires only a small component which would require average level of computational power to display the reasons
Recommending of unseen item to users – Content-based recommender systems have the capability of recommending items of the preferred category which are not yet rated by any user	This advantage would make the system more dynamic as it can make more alignments between users and items which will improve the accuracy.	If the item descriptors and user profile information are based on a mathematical approach it will have negative impact on the scalability.

Table 2.9 - Impact the advantages of a content based recommendation system would have on accuracy and scalability (self composed)

Limitation & description	Impact on accuracy	Impact on scalability
Limited content analysis – Irrespective whether an automated or manual process is used to assign descriptors to an item content-based recommender system have a native limitation in the number and the types of descriptors that can be used to describe an item	Since the accuracy of content based recommendation systems is depended on a successful matching between the item descriptors and user profile this limitation will have a negative impact on the accuracy of the system	This limitation can be handled to a satisfactory manner with the use of NLP and ontology techniques but use of such techniques would have a negative impact on the scalability due to the additional resource requirements
Over specialization – This limitation will make items that are going to be recommended to a user will be similar to those already seen thus making the recommendations to limited extent of novelty	This will not have a major impact on the accuracy of the prediction and only be a limitation in recommending hot items from different categories to a user	To overcome this limitation concepts like user clustering and collaborative filtering techniques can be used but those would have a negative impact on the scalability of the system as those would require additional computational resources
Cold Starter problem of a new user – A new user will need to rate sufficient number of items for the system to gauge the preferences accurately to provide reliable recommendations	This will have a negative impact on the accuracy of the recommendations done for a new user which may even lead to new users not trusting the system	This limitation partially can be overcome by using demographic data or social media information of a new user to do the recommendation but those techniques will have an additional overload on the system thus making the scalability going down.
Grey sheep – When the preferences of a user is changed collaborative recommender systems would take time in adjusting to the new preferences of the user	This will have a negative impact on the accuracy for a while until the system understand the new preferences of the user	This limitation can be limited by using techniques like AI but those techniques will have a negative impact on the scalability of the system due to the additional resource requirements

Table 2.10 - Impact the disadvantages of a content based recommendation system would have on accuracy and scalability (self composed)

As discussed above by making recommendations more user-centric content based recommendation systems have made the learning process of user preferences simple to implement. Another advantage is the making of user experience more independent which would lead the system to find similarities or alignments faster. Content based recommendation systems have the capability of recommending unseen items to users which would make items actively being recommended without being ignored. In the context of an online news site this would lead to new news articles being recommended to users. Content based recommender systems would also work with sufficient accuracy provided that enough information is captured about an item. Though content based recommendation systems are more reliable with text based descriptors like an online news articles they are often affected with the drawback of limited content analysis though by introducing more accurate ontology and NLP modules that drawback can be restricted. But this would require additional implementation and maintaining overhead as for each domain a new ontology and a NLP module will have to be implemented & maintained thus making the scalability of the system a major concern. Still content based recommender systems would face the problems when it has to interact with users who are new to the system. Considering the fact that online news site would always have to contend with new readers and even the already existing readers would be with different reading interests it can be concluded content based recommender approach solely would not be sufficient enough for an online news recommender system.

2.4.2 Collaborative Recommender Systems

Collaborative filtering Shang *et al* (2014) also referred to as social filtering, tries to recommend items for a particular user by considering the recommendations given to items by other users with similar preferences. In this approach the recommender system would first try to understand the preferences of a particular user and then cluster users or items in to similar preference groups. Recommendation process would be carried out based on the ratings given by the group on a particular item.

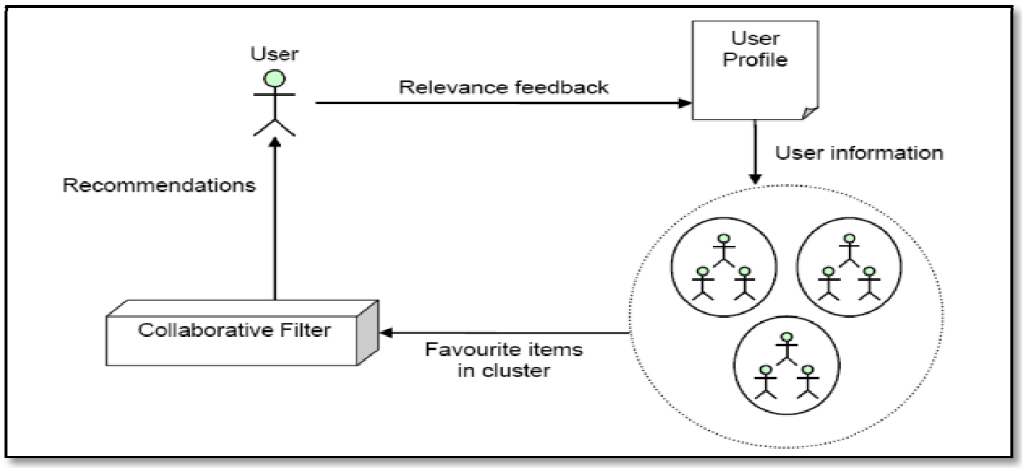


Figure 2.2 - Architecture of collaborative recommender systems - (Hahsler, 2011)

Collaborative filtering approaches

One way of distinguishing collaborative recommendation systems is to consider whether the recommendations are done based on a user profile or items (Papagelis and Plexousakis, 2005). Table 2.11 provides an overview of the two approaches.

Approach	Overview
Item based approach	This approach would recommend items by considering how similar the target items are to what the user has already rated. But it faces the limitation of failing to recommend items which are different to what the user has already seen. Item based approach is more suitable for static systems which would not be acquiring item frequently
User based approach	This approach would recommend items which are rated by other users with similar preference but user based approach has the limitation of time taken to successfully group a user but it is more suitable for dynamic systems which would have to handle frequent item additions

Table 2.11- Overview of item based and user based approaches (self composed)

Since online news sites have to face frequent additions of new news articles it can be argued that item based approach would not be suitable for the proposed system and additionally item based approach would require more computational resources compared to user based approach which would be having a negative impact on the scalability of the system.

Pagare and Patil (2013) have highlighted two approaches to distinguish collaborative recommendation systems. Table 2.12 provides an evaluation on the impact those methods would have on the accuracy and the scalability of the system.

Approach	Impact on Accuracy	Impact on Scalability
Model based approaches	As the raw data is processed offline those may not reflect the latest rating for an item or user leading to lower accuracy	Since the data is processed offline recommendation and rating processes can abstracted from each other to make the system more scalable.
Memory based approaches	As the raw data is kept in the memory and processed these approaches would reflect the latest ratings for an item or user which would make the recommendations more accurate	Since the processing of the data is done on a near real time basis the system will have to accommodate with better memory & load management techniques which would lead to lower scalability

Table 2.12 - Impact the two methods of a collaborative recommendation system would have on accuracy and scalability (self composed)

The choice of choosing between these two approaches will be a tradeoff decision that has to be taken by considering the accuracy requirement and the level of users and items the system would have to manage. According to Gupta *et al* (2013)most of the social networks like twitter and facebook use memory based approaches for the recommendations due to the vast amount of items and users such systems have to deal with. But for an online news site with a comparatively smaller number of users and items it's not advisable to opt for memory based approaches due to the large investments that had to be incurred on the hardware.

Approaches for model based collaborative recommendation systems

Pagare and Patil (2013) have highlighted that decision trees, aspect models, latent factor models and clustering methods are some of the model-based approaches that can be used for collaborative filtering. Out of those approaches most widely used approach is the clustering approach due to its ease of implementation, dynamic nature and greater support for scalability of the software.

Review on clustering algorithms

Clustering is the process of partitioning items into groups and each group is known as a cluster. Each cluster would consist of objects with similar patterns among themselves than they are to a pattern belonging to a different cluster. As highlighted by Abbas (2008) K-means, Hierarchical, Self organization map and Expectation maximization are 4 of the most widely used and powerful clustering algorithms. Table 2.13 provides an overview of the impact the above mentioned clustering algorithms would have on the accuracy of the recommendation while table 2.14 provides an overview of the impact the above mentioned clustering algorithms would have on the scalability of the recommendation system. Facts for the review were gathered from reviews about clustering algorithms by Abbas (2008), Rauber, Pampalk and Paralič (2000), Kaur and Kaur (2013) and Sharma, Bajpai and Litoriya (2012).

Clustering Algorithm	Impact on accuracy
K-means	• Lowest initial accuracy rate provided in classification but since the algorithm follows an iterative process the quality of the grouping can be improved as time goes on • More dynamic clustering approach
Hierarchical	• As the number of clusters increase the accuracy of the classification increases • Since the algorithm doesn't support an iterative process changes to the patterns of items can't be absorbed.
Expectation maximization map	• Provides a higher accuracy rate compared to K-means algorithm but as the cluster number increase the accuracy of the grouping decreases
Self-organization map	• Shows the highest accuracy in classifying most objects in to their suitable clusters • As the number of clusters increase the accuracy of the clustering decreases • Less sensitive to noise of the data set

Table2.13 - Impact on accuracy by clustering algorithms (self composed)

Clustering Algorithm	Impact on scalability
K-means	• Computational power required for the clustering is very minimal compared to the other three • Has the ability to work with larger data sets without affecting accuracy or resources
Hierarchical	• A very resource greedy algorithm due to smaller data set with in a cluster and as a result would make the scalability to down
Expectation maximization map	• A relatively less resource greedy algorithm compared to hierarchical and self-organization maps • Has the ability to work with larger data sets without affecting accuracy or resources but not as strong as K-means approach.
Self-organization map	• A very resource greedy algorithm due to smaller data set with in a cluster and as a result would make the scalability to down

Table 2.14- Impact on scalability by clustering algorithms (self composed)

When evaluating the above mentioned clustering algorithms it can be suggested that K-means clustering technique as the most suitable clustering technique for the proposed system due to it's scalability and dynamic nature in understanding pattern changes. Though hierarchical and self-organization maps provide better accuracy due to their lack of support for larger data sets makes them incompetence to be selected as it can be expected that online news sites will have to deal with larger number of readers and articles. When considering the expectation maximization clustering approach though in accuracy it is similar to K-means it can't be suggested due to its lower capability to work with larger data sets compared to the K-means algorithm.

Advantages and disadvantages of collaborative recommendation systems

According to Rafsanjani*et al* (2013) and Herlockeret *al* (2004) collaborative recommender systems would face the below mentioned advantages and limitations. Table 2.15 provides an evaluation on the impact the advantages would have on the accuracy and the scalability of the system and table 2.16 provides an evaluation on the impact the disadvantages would have on the accuracy and the scalability of the system.

Advantage & description	Impact on accuracy	Impact on scalability
Applicability to open domains – As Collaborative recommender systems don't represent items in terms of descriptors and the recommendations are done based on a rating of a user group CBR systems can be applied to virtually any kind of domain.	This wouldn't have any impact on the accuracy of the recommendation system	This will have a negative impact on the scalability compared to a content based recommender system which would be using textual representation for item and user preference matching as a textual representation based recommender system would have better scalability compared to a mathematical represented content based or collaborative recommender system
Improved maintainability – Since the collaborative filtering techniques do not require any tagging descriptions or features for items maintainability will be high.	This wouldn't have any impact on the accuracy of the recommendation system	This will have a negative impact on the scalability compared to a content based recommender system which would be using textual representation for item and user preference matching as a textual representation based recommender system would have better scalability compared to a mathematical represented content based or collaborative recommender system
Cross-genre recommendation – Collaborative recommendation systems can make predictions of entirely different items to the user who have never rated such items in the past	This would lead to better alignment between different domains and as a result improved accuracy on the recommendations	Since this is a native advantage of collaborative recommender systems it won't require additional techniques and as a result it won't generate a negative impact on the scalability of the system.

Table 2.15- Impact the advantages of a collaborative recommendation system would have on accuracy and scalability (self composed)

Limitation & description	Impact on accuracy	Impact on scalability
Cold starter problem – A new user will need to rate sufficient number of items for the preferences to be gauge accurately and new products need to be rated by a substantial amount of users before it can be	This will have a negative impact on the accuracy of the recommendations done for a new user which may even lead to new users not trusting the system. Certain better matching items would be hidden	If actions are taken to overcome the early rater problem using any implicit sources like social media engineering it would require additional computational requirements and will have a negative impact on the scalability of the system
Grey sheep – When the preferences of a user is changed collaborative recommender systems would take time in adjusting to the new preferences of the user	This will have a negative impact on the accuracy for a while until the system understand the new preferences of the user	This limitation can be limited by using techniques like AI but those techniques will have a negative impact on the scalability of the system due to the additional resource requirements
Popularity bias issue- Collaborative filtering systems will tend to recommend only popular items making less room for users with unique interest.	This will lead to lower accuracies or totally ill relevant recommendations for users with unique tastes making certain users go away from the system	To overcome this limitation AI techniques can be adopted to better understand user preferences and make more user clusters but it will lead to poor scalability as it would require additional computational resources
Shilling attack – People can rate their items more to stop the competitor items being recommended	This may pose a huge threat to a product selling site like Amazon but for online news site this may not be pose a huge threat as no one would gain any substantial gain by rating a particular item high.	If required this limitation can be overcome by using AI technologies to understand the rating patterns and take precautions to stop shilling attacks but that would require additional computational resources and will have a negative impact on the scalability

Table 2.16- Impact the disadvantages of a collaborative recommendation system would have on accuracy and scalability (self composed)

When evaluating collaborative based recommender systems it can be seen that inherently such systems don't require domain knowledge of an item like in the content based recommender systems to carry on with the predictions, which would make it more suitable for systems with different item categories. Since news sites contain articles on different domains it can lead to better recommendations to users. Another advantage is the ease of system maintenance as the system architecture would be based on few components compared to content based recommender system which would give the similar accuracy. A further advantage of a collaborative recommender system is the capability of recommending items of another category even though the user has not shown an interest on such items. As accuracy of the recommendations is a key aspect when developing a recommender system this can lead to better recommendations to users it can help the system to enhance the understanding of different interest of a user. But since new items will not be recommended to users until a certain level of rating is achieved new items can simply be ignored by the system. For news site this can lead to a disadvantage as by nature news sites are dynamic which would have to contend with new items on a regular basis and typically readers will be interested on new news articles. As discussed above due to the dynamic nature of online news sites they would keep on accumulating new news articles and it can be accepted that as time goes on more and more new readers would subscribe for the recommender service and as a result scalability of the system would be hindered. As scalability of a recommender system is a key aspect, approaches like software agents or more advanced approaches like distributed clustering would have to be incorporated. With the above evaluation it can be concluded that a collaborative approach itself would not be enough for a successful implementation of an online news recommender.

2.4.3 Hybrid Recommender Systems

Since both CBF and CF systems independently provide lower recommendation accuracies due to the inherent limitations of the two approaches hybrid recommendation systems have gained popularity in the recent past. The key inspiration behind a hybrid recommendation system is to combine at least two recommendation engines to negate the limitations of one with the other. Felfernig and Burke (2008) and Adomavicius and Tuzhilin (2005)have demonstrated that combining multiple techniques together would achieve some synergy between the techniques used and as a result would try to provide more accurate recommendations than pure approaches and it can be suggested that the most suitable recommendation approach for the proposed system would be a hybrid recommendation system.

Limitations of hybrid approach

Though the expectation of combining a content based recommendation system with a collaborative recommendation system is to negate the disadvantages of one with the advantages of the other certain disadvantages can't be negated (Burke, 2002). Table 2.17 provides an overview of the disadvantages of a CBF system that can be negated by combining with a CF system while Table 2.18 provides an overview of the disadvantages of a CF system can be negated by combining with a CBF system.

Disadvantage of CBF systems	Advantage of CF systems that could negate
Limited content analysis	Cross-genre recommendation
Over specialization	Cross-genre recommendation
Grey sheep	No advantage of CF to negate this disadvantage
Cold starter problem of a new	No advantage of CF to negate this disadvantage

Table 2.17- Negation of disadvantages of a CBR system (self composed)

Disadvantage of CF systems	Advantage of CBF systems that could negate
Cold starter problem of a new	User independence
Popularity bias issue	Recommending of unseen item to users
Shilling attack	Recommending of unseen item to users
Grey sheep	No advantage of CF to negate this disadvantage
Cold starter problem of a new	No advantage of CBR to negate this disadvantage

Table 2.18 - Negation of disadvantages of a CF system (self composed)

Though the most suitable recommender approach for the proposed system is a hybrid recommender engine it can be seen that cold starter problem of a user and grey sheep disadvantages are common to both the recommender types and can't be negated with the advantages of the other. Since cold starter problem of a user and grey sheep limitation or change of user interests can affect the accuracy of the system in a negative manner approaches that can be taken to overcome those limitation will have to be further studied.

2.5 Cold Starter Problem of a New User

Chen *et al* (2013), Rashid, Karypis and Riedl (2008) and Lin *et al* (2013) highlight work done on finding approaches that can be utilized to overcome the cold starter problem of a new user and have suggested three approaches that can be adopted to solve the cold starter problem of a new user. Table 2.19, 2.20 and 2.21 provides an overview about the impact those approaches would have on the accuracy and scalability.

Approach	Implicit user preference capture
Description	In this approach the system would try to understand the preferences of a new user through the stored user preferences of another environment like social networks
Advantages	**Disadvantages**
1. Less user interaction which would make the system more user friendly.	1. Not all users will have social network accounts 2. Requirement of additional software modules to read and understand the stored knowledge. 3. Lower accuracy rate compared to explicit user preference capture. 4. Increased software coupling with 3rd party environments.

Impact on accuracy	Though the accuracy is comparatively low, it can be improved by using additional components like NLP and ontology modules.
Impact on scalability	The scalability of the software would go down as a result of the software components that had to incorporate for reading and understanding of knowledge.

Table 2.19- Implicit user preference capture (self composed)

Approach	Explicit user preference capture	
Description	In this approach the system would try to understand the preferences of a new user by directly capturing the preferences by means of a question and answer session.	
Advantages		**Disadvantages**
1. If the user provides correct information a higher accuracy rate compared to implicit user preference capture.		1. May reduce the user friendliness of the system for a new user. 2. Since the number of questions asked from the user should be limited it may not be sufficient to understand unique preference of a new reader.
Impact on accuracy	Accuracy of this approach is comparatively higher compared to the implicit approach as it directly involves the user.	
Impact on scalability	Scalability of the software would be higher compared to implicit approach as it won't involve additional software components.	

Table 2.20 - Explicit user preference capture (self composed)

Approach	Hybrid user preference capture
Description	The system would try to understand the preference of a user by combing implicit and explicit user preference capture approaches.
Impact on accuracy	This approach provides the best solution for capturing unique preferences of a new user and the highest accuracy rate compared to implicit and explicit user preference capture approaches.
Impact on scalability	The scalability of the software would go down as a result of the software components that had to incorporate for reading and understanding of knowledge.

Table 2.21- Hybrid user preference capture (self composed)

When comparing the explicit user preference capture approach with the implicit user preference capture approach it can be seen that the explicit user preference approach has a better accuracy and scalability level. But when comparing explicit user preference approach against the hybrid user preference approach it can be seen that though the hybrid user preference approach has a better accuracy rate it can have a negative impact on the scalability of the system. As most of the researches have suggested cold starter problem as a separate research area it can be concluded that finding the most appropriate solution for the cold starter problem is beyond this

research scope and it is suggested to use the explicit user preference approach for the current problem due to its comparatively higher accuracy and support for scalability.

2.6 Change of User Preferences

As highlighted above another limitation that can't be countered with the adaptation of a hybrid recommender approach is the change of user preferences. As highlighted by Billsus and Pazzani (2000) change of user preferences can occur due to two scenarios.

1. Due to lack of items that matches with the interests of a user the recommender engine doesn't have any items to recommend
2. Over time the preferences change of a user.

The first scenario would occur with seasonal news articles like news articles related to a league level completion of a sport which would take place during a specific time period in the year but would generate a huge hype over the readers (Yu *et al*, 2006). But once that specific season is over the system would not be accumulating any news articles specifically related to the season to be recommended to the interested readers though the reader preference has not changed.

The first scenario would not generate a major negative impact on the accuracy or the scalability of the recommender system but in a dynamic user preference model such interests can be removed from the preference model and the preference information can be saved to carry out recommendation back again with the accumulating of seasonal news articles. Decline in the number of articles can be detected by keeping a track of number of articles generated for a particular interest and once the article generation gets reduced to zero the preference details can be stored for future use. This approach would have a fair accuracy level and will have a positive impact on the scalability of the system. Another approach would be to use data mining combined with an AI technology to understand the time period such news articles are generated and once the time period expires the system would take out the preference details from the user profile. For this approach to be implemented the system should be mature enough with a substantial level of data to be used for the data mining approach and would have a negative impact on the scalability. Due to those limitations the first approach of keeping a track of number of articles generated for a particular interest is suggested as the most suitable approach.

The other scenario would happen due to various changes in the readers interests like change of beliefs the user has changed the previous preferences (Chen, Jiang and Zhao, 2010). In this scenario if measures have not been taken to track changes to user preferences the system would not be able to understand the change of interest of the user and would continue to recommend undesired articles to the user which would have a major negative impact on the accuracy.

Decline in the reader preference can be detected by keeping a track how a reader would be reacting to the recommended news articles. If a reader continuously keeps ignoring news articles that match a user preference then it can be identified as a user preference change. One approach this can be achieved is by comparing the interest rating value change for a specific shorter period of time which would have a fair accuracy level and will

have a positive impact on the scalability of the system. Another approach would be to use pattern identification approach combined with an AI technology to understand the change of user preference. For this approach to be implemented the system should be mature enough with a substantial level of data to be used for the pattern identification and it would have a negative impact on the scalability. Due to those limitations the first approach comparing the interest rating value change for a specific shorter period of time is suggested as the most suitable approach.

2.7 Suitability of Software Agents to Improve Scalability

As highlighted by Beel *et al* (2013) and Takács *et al* (2009) a major overlooked critical success factor of recommendation systems is the scalability. The scalability of a software can be achieved at various level and according to Mircosoft (2014) the least effective and costlier approach to improve scalability would be at the hardware level while the most effective and cheaper approach would be at the design and the software runtime level. As highlighted by Tan *et al* (2013) one simple but a powerful approach of achieving scalability of software is to induce better collation and communication between different activities. This can be easily achieved through software agents as the design and runtime view of software agents emphasize better collation and communication between different activities of the software.

2.7.1 Multi Agent Systems

Multi-Agent Systems are used to solve complex issues which needs large number of resources and accurate information in a quick time. By using multi-agents, the developer will be able to assign tasks to each and every agent which will increase the efficiency of the program. Multi-agents are programmed to adapt to the behavioural changes of the environment and to respond to that change. Therefore, the dynamic nature of agent's behaviour will help them to dynamically achieve complex tasks in changing environments.

There are several agent categories in multi-agent system as Resource Agent and Request Agent. Resource Agent is responsible of interacting with resources and handling the operations regarding resources. The Request agent is the agent who is requesting information from the other agents. When requesting information, either an agent can broadcast a message to the other agents or put the message in a message space. The message space is a message area which each and every agent will periodically read the messages displayed and check if the message is relevant to that agent. If the received message or the message in the message space is relevant to that agent it will act according to that message and update the message space with its status. This ensures the effective communication and negotiation between agents.

2.7.2 Characterises of a Multi Agent System

The main three characteristics exist in a multi-agent system are negotiation, collaboration and communication.

- Communication - Communication happens between two agents.
- Negotiation - Discussion to come into an agreement about a request sent by the agents.
- Collaboration - Work together and discussing a common request among the agents.

Multi-agent system basically consists of request agent, resource agent, message space and the ontology. Ontology stores the initial knowledge to initiate task on each agent. Request agent get the request to the system and resource agents represents different resources that would help to complete the task. Each agent updates the message space to specify their current state to other agents.

2.8 Review of Existing Systems

<table>
<tr><td>Research or service</td><td colspan="2">Google News ((Google news, 2014)</td></tr>
<tr><td>Features</td><td colspan="2">• Works as a portal by aggregating news from various sources
• Presentation of news articles to the readers with a dateline
• Clustering of similar news articles based on the keywords of the articles and date of publication
• Only recommendation feature is recommending of articles which match the reader's Google News search history and the articles the reader has clicked on</td></tr>
<tr><td colspan="2">Accuracy</td><td>Scalability</td></tr>
<tr><td colspan="2">• The personalization is provide only if the readers have to signed up for the "search history" feature making the personalization not available on common basis</td><td>• No steps taken to maintain the scalability of the service</td></tr>
<tr><td>Research or service</td><td colspan="2">Das. A, et al (2007)</td></tr>
<tr><td>Features</td><td colspan="2">• A scalable online recommendation system for Google News based on collaborative filtering
• Content independency of the system which makes it easily extendible to other domains</td></tr>
<tr><td colspan="2">Accuracy</td><td>Scalability</td></tr>
<tr><td colspan="2">• Currently hangs around 30% (Bhattarai A, (2008))
• Currently have only a collaborative recommendation process
• Need to convert to a hybrid recommendation systems (Gunawardana and Meek (2009))</td><td>• Scalable only for the news aggregation process
• Haven't considered scaling the system for the number of readers (Bhattarai A, (2008))</td></tr>
<tr><td>Research or service</td><td colspan="2">Tatar, et al. (2011)</td></tr>
<tr><td>Features</td><td colspan="2">• Based on the number of comments received by a particular news article
• Has tried to tackle the limitation of online content has a disproportionate distribution of user interest of publication</td></tr>
<tr><td colspan="2">Accuracy</td><td>Scalability</td></tr>
</table>

• Major drawback of this solution is that there are no mechanisms used to filter out positive comments from the negative comments	• No steps taken to maintain the scalability of the service

Research or service	Tavakolifard et. al (2013)
Features	• System for the mobile news domain which uses both content-based and collaborative techniques in combination with semantic and linguistic theories • Uses user profiles which are constructed based on implicit feedback of the reader to map a particular reader's interests

Accuracy	**Scalability**
• Higher accuracy as the recommendation is based on a hybrid approach • Since explicit reader feedback to identify user interests are not considered user modelling is not accurate	• Currently it only supports iPhone platform • No steps taken to maintain the scalability of the service

Research or service	Li et. al (2011)
Features	• System for the mobile news domain which uses both content-based and collaborative techniques in combination with semantic and linguistic theories • Uses user profiles which are constructed based on implicit feedback of the reader to map a particular reader's interests

Accuracy	**Scalability**
• Higher accuracy as the recommendation is based on a hybrid approach • Since explicit reader feedback to identify user interests are not considered user modelling is not accurate	• Currently it only supports iPhone platform • No steps taken to maintain the scalability of the service

Table 2.22 - Review of existing system

As highlighted by the Table 2.22 it is evident that the current work done on online news recommendation most systems haven't considered much in making those systems scalable to handle the increasing number of users nor article of a typical online news environment with the use of software agents techniques and they haven't considered in making them support both the standard web view and mobile platforms using a single presentation layer. Most of the work done have limitations on the accuracy of the recommendations due to either they have considered only one type of a recommendation engine or they have failed to identify all possible user feedback types and validate them.

2.9 Chapter Summary

This chapter mainly focused on finding the best approaches, concepts and techniques to provide a solution to the problem domain. The chapter started by highlighting the suitability of a recommendation system to online news environment. Then the chapter provides an overview of the key success factors of a recommendation system and mentioned that this research would focus in identifying the best approaches that can be used to achieve a trade off between the accuracy and the scalability of the recommender system. Then the chapter moves on to describe the components of a recommender system. Since it was highlighted that the heart of any recommender system is the recommender engine further research was carried on to the architecture of a recommender engine. There it was identified that recommender engines can be classified in to three types based on the recommendation approach utilized. After wards the chapter provides an evaluation on content based recommender approach and it was decided to use the semantic user profiling approach to model the user profile and manual approach to assign item descriptors to the articles due to their support for higher level of accuracy and scalability and the chapter is followed by an evaluation on the advantages and limitations of CF approach and the impact those would have on the accuracy and the scalability of the system. Next the chapter provides an evaluation on collaborative based recommender approach and it was decided to use K-Means clustering approach cluster the readers due to its simplicity and support for higher scalability and then the chapter is followed by an evaluation on the advantages and limitations of CBF approach and the impact those would have on the accuracy and the scalability of the system. Since both CF and CBF approaches have their unique weakness the hybrid approach was evaluated and it was found out that even the hybrid approach would not be providing and solution to the cold starter problem of a new user and user preference changes. Afterwards the chapter provides a review of the approaches that can be taken to solve the cold starter problem and it was decided to use the explicit approach of identifying user interest for a new user due to its higher accuracy. After that the chapter provides an overview of the reasons that can lead to user interest changes followed by a review on the approaches that can be utilized to solve the user preference changes. Then the chapter provides an overview to the importance of scalability management in a recommender system followed by a review on the approaches that can be taken to achieve a higher scalability in a recommender system and it was highlighted that the most effective and cheaper approach would be at the design and the software runtime level . After wards the chapter would provide a justification that use of software agents would help to increase the scalability of software as software agents by nature try to induce better collation and communication between different activities of the software. Lastly the chapter provides a review of the existing solutions both in the commercial and research levels to identify the successfulness of those systems in solving the problem and also to identify any possible positive features that can be adapted in to the proposed system. The review of the existing systems proved that those systems don't provide a comprehensive solution with either limitations on the accuracy or the scalability of the recommendation engines used. It was also highlighted at that review that those systems either support only a web view or a mobile app view.

Next chapter will be the Project Management, which would discuss about the overall project management process of the project.

Chapter 3: Project Management

Contents

- Chapter Overview
- Project management methodology
- Development methodology
- Research methodology
- Chapter summary

3.1 Chapter Overview

Previous chapter focused on discussing literature related to the problem domain and this chapter contains information related to the project management process of the project. It will start from identifying a suitable project management methodology and moving on to time and resource allocations of the project. Then it will discuss about the possible risks along with appropriate mitigation plans. Finally it will lead to a discussion about selecting a suitable software development methodology for the project.

3.2 Project Management Methodology

Any project would have scope, time and cost as the constraints of the project and it is important to manage the above mentioned constraints properly to produce a quality output from the project undertaken. In a research project, managing those constraints going to be harder compared to an industrial project due to frequent requirement fluctuations, which highlights the importance of a suitable project management methodology. PRINCE2 is a widely used project management methodology which encompasses the high level management, control and organisation of a project. It was decided to adopt PRINCE2 as the project management methodology due to the previous exposure and experience with it.

3.2.1 Time Allocation

Following table 3.1 represents an overview of the identified main tasks of the (Refer appendix A for complete Gantt chart). Research projects require the literature review process to be carried till the end of testing phase due to frequent requirement fluctuations of research projects. Subsequently the documentation of project will too spread throughout the project life time as it will be the ultimate deliverable of this exercise

No	Task	Duration	Start Date	Finish Date
T1	Initial research	14 days	08 Jan 2014	21 Jan 2014
T2	Preparation & Submission of TOR	28 days	18 Jan 2014	14 Feb 2014
T3	Literature review phase	224 days	08 Jan 2014	22 Aug 2014
T4	Requirement specification phase	36 days	14 Mar 2014	20 Apr 2014
T5	Designing phase of the system	30 days	10 Apr 2014	10 May 2014
T6	Implementing phase of the system	90 days	06 May 2014	5 Aug 2014
T7	Testing phase of the system	67 days	06 Jun 2014	12 Aug 2014
T8	Critical evaluation phase of the system	20 days	01 Aug 2014	21 Aug 2014
T9	Preparation & submission of the final document	221 days	14 Jan 2014	23 Aug 2014

Table3.1 - Time allocation

3.2.2 Constraints and Dependencies

The successful completion of project will be dependent upon the following constraints and dependencies.

- Time constraint - considered as a major challenge for the successful completion of the project as the whole software development life cycle has to be carried individually.

- Lack of prior knowledge – due to lack of knowledge on automated recommender concepts and software agents the development of the proposed system may be hindered.

3.2.3 Potential Risks and Mitigation Plan

Risk Id 1	Failure to keep with the up-to-date domain knowledge & technology changes		
Risk level	High	Occurrence Frequency	High
Description	Automated recommendation is a very active research area where new ideas and concepts are brought forward on a regular basis making it difficult to keep up with the most up to date knowledge of the domain. It has to be accepted that the surrounding technologies too may change over the course of the project time schedule .This will present a risk that the system is not be developed using the most up to date domain knowledge nor using the latest technologies		
Mitigation	• Check for any new work done on the domain area on a weekly basis • Check for technology changes on a weekly basis. • Keep frequent communication with domain experts about the latest developments.		
Risk Id 2	Constant and repeated changes to requirements		
Risk level	High	Occurrence Frequency	High
Description	As the nature of a research project there would be constant changes to the requirements in every phase of the software life cycle and. This is considered as a risk project as it may be difficult to manage all the requirement changes within the allocated time period.		
Mitigation	• Prepare ahead for the changes • Prioritize the changes and try to tackle only the critical changes		
Risk Id 3	Not being able to achieve the expected accuracy level for the recommendation		
Risk level	High	Occurrence Frequency	Low
Description	Due to time constraints and changes to requirements and technologies the developed system may not be able to achieve the expected accuracy rates for its users		
Mitigation	• Try to keep up with the latest developments • Finish the prototype early to have enough time to do changes if required.		
Risk Id 4	Data lost due to hardware, software and system failures		
Risk level	High	Occurrence Frequency	Low
Description	During the course of the project the data of the project can get lost due to hardware, software and system failures and it may prove to be a critical threat to the successful completion of the project		
Mitigation	• Maintain daily backups • Carryout continuous testing and debugging of the software.		

Table 3.2 - Identified risks for NewsRec System

3.3 Development Methodology

In order to avoid schedule and cost overruns and to mitigate frequent requirement changes of the project a suitable development methodology has to be adopted. With the aim identifying a suitable development methodology, key characteristics of several software development methodologies are critically evaluated. Following table shows the characteristics of several frequently used software development methodologies and the below evaluation is based on the available resources, nature of the project and ability to meet the identified requirements

	SOFTWARE DEVELOPMENT METHODOLOGIES				
	Waterfall Methodology	**Rapid Application Methodology**	**Spiral Methodology**	**Agile Methodology**	**Prototype Development Methodology**
CHARACTERISTICS	Top Down development with independent rigid phases	Iterative development	Iterative development	Iterative & Incremental development	Prototype is developed based on currently known requirements
	Linear Module	Software prototyping	Continuous refinement of the final software product	Agility for rapidly changing	Higher rate on customer interation
	All the requirement should be known in the upper hand	Fast development with minimum planning and high quality systems	Emphasizes Risk analysis	Accelarated delievery	
	Heavy Documentation			Customer Interaction	

Table 3.3 - Characteristics of the Software development methodologies (Self composed)

Waterfall method is suitable for projects with a stable set of requirements but in a research project by nature requirements change during each phase of the software development life cycle. Since linear model with rigid phases doesn't support requirement fluctuations of experimental and research development, traditional waterfall approach is not suitable for a research project. Therefore, it is evident that this project needs to select a methodology where it has more iterative approach than the waterfall methodology. When considering the Rapid Application Development methodology, it can be seen that it uses minimal planning and then quickly moves in to prototype development at an early stage of the project duration. It is evident that it is a must to have a certain level of understanding of the final products' functionalities to start developing the prototype at an early stage. But in a research project functionalities of the prototype can't be identified with a higher degree of certainty at the start of the project and it can be concluded that this methodology is not suitable for a research project due to the constraint of not knowing the functionalities of the prototype in an early of the project. As listed in the table 3.11, Agile methodology uses iterative and incremental development strategy there will be lot of testing, customer interactions and discussions throughout the development phase which may result in time overrun. Additionally Agile methodology is more suited to handle projects with projects with massive scopes and larger number of stakeholders on various levels of an organization. Since the project undertaken has a major constraint

on the time availability and neither involved with a huge scope nor a larger number of stakeholders it can be concluded that Agile methodology isn't suitable to handle the project in hand. Similar to the Rapid Application Development methodology Prototype methodology too focus on moving into the development of the prototype at an early stage of the project and poses the same problem like in the Rapid Application development methodology as the functionalities of the prototype can't be determined at an early stage of a research project. Additionally it may increase the complexity of the system and may lead to scope enhancements of the system due to the higher rate of customer interaction leading to time overruns. Therefore it can be deduced that Rapid Application Development methodology isn't suitable to conduct the project undertaken.

The Spiral methodology supports continuation of cycles of the software development life cycle without clear termination conditions which would help to counter frequent requirement fluctuations which would lead to timely delivery of the prototype. Additionally the Spiral methodology helps to analyse the risks that would affect the final product and finding those risks in advance can lead to speed up the development process. As the major challenges of the project in hand are the time constraint and frequent requirement fluctuations it can concluded that Spiral methodology would be the most suitable development methodology.

3.4 Research Methodology

As highlighted by Dubois and & Gadde (2002) a research can be categorized in to two categories known as inductive researching and deductive researching. Deductive approach is aimed at proving and testing a hypothesis while the inductive approach is concerned with generation of new theory emerging from data. The project in hand falls in to the deductive researching approach as the aim of the project is to prove and test that a software agent based recommender system can solve the limitations mentioned in the introduction chapter.

3.5 Chapter Summary

This chapter initially highlighted the importance of a suitable project management methodology to get best result out of the project and PRINCE2 which is a widely used project management methodology was chosen as the most suitable project management methodology due to the previous experience and exposure with it. Next the entire project was broken down into different task and proper time allocation was done, so that work can be done accordingly and parallel. Then followed a descriptions of identified constraints of the research and possible risks that can occur during the course of the project were identified and appropriate mitigation plans were discussed. After that to identify a suitable development methodology for the project several development methodologies were evaluated and it was decide to adopt the Spiral development methodology due to its flexibility to support frequent requirement changes that can occur during the project life cycle. Finally this chapter discussed about choosing a research methodology that have to be adopted for the successful completion of the project and deductive research methodology was adopted over inductive methodology due to the reason that this project was based on proving the hypothesis of adopting software agents for a online news recommendation system. The next chapter is the System Requirement Specification, it will discuss about the entire requirement engineering aspect of the proposed project

Chapter 4: Requirements Specification

Contents

- Chapter Overview
- Requirement elicitation process
- Stakeholders
- Use Case Diagram
- Functional requirements
- Non – Functional requirements
- Chapter Summary

4.1 Chapter Overview

The previous chapter presented a detailed review of the problem domain & existing literature in relation to online news recommendation. This chapter starts by presenting the process untaken to gather requirements from identified stakeholders for the proposed NewsRec System. Next it divulge deeper in to several requirement analysis models carried out to analyze the raw data gathered and identify requirements of the system. Lastly it discusses how the project scope will be refined and restructured to accommodate the identified requirements.

4.2 Requirement Elicitation Process

In order to ensure efficiency of the requirement gathering process and counter the limitations associated with each requirement elicitation method several approaches were utilized parallel to gather requirements from various stakeholders. Table 3.1 provides an overview of the factors that lead to the selection of each utilized requirement elicitation method.

Method No 1	Literature Review
The literature review carried out on the current status and limitations of online news environments and various recommender system assisted to identify many problematic areas concerning to accuracy and scalability of a online news recommender system	
Advantages	• Facilitate to identify and better understand crucially evaluated and well documented advantages and limitations concerning to news recommendations • Paved way to identifying certain areas that have be included in the questionnaire and interview process for better clarifications
Disadvantages	• Due to delays in publications literature material may not present the latest developments and limitations of the related domains • Time consuming effort due to the time and effort required review the vast amount of literature available
Method No 2	Questionnaire
A questionnaire focusing on identifying the end user requirements for an online news recommender system was prepared. Since news is interested by various cross sections of the society hard copies of the questionnaire along with an online survey was made available.	
Advantages	• Comparatively time saving method to other elicitation methods • Ability to cover geographically, communality dispersed users • Ease of tabulation and comparison of the feedback due to the standardization of questions
Disadvantages	• Success of identifying the correct requirements depends on the honesty of the participants • May encumber participants from sharing additional information due to

	standardization of the questions. • Difficulties faced when comprehending answers given to open ended questions
Method No 3	Formal interviews
A series of formal interviews were carried out with the domain experts of the online news domain with the aim of identifying standards of news categorization and current processes adapted to categorize news articles along with their suggestion for online news recommender system.	
Advantages	• Ease of clarifying doubts with the experts while the interview is going on • Enabling the elicitation of measurable requirements and guideline for the implementation
Disadvantages	• Inability to reach a wider audience of domain experts due to time taken to conduct an interview • Subjective interpretation of the problem by the experts
Method No 4	Observations of end user operations
Several observation sessions of users accessing online news sites using various devices were conducted to identify the difficulties faced by users of online news site in relation to finding preferred news articles. Additionally another set of observation sessions were carried out to observe how the users are using online site like Amazon which provides a recommendation facility.	
Advantages	• Direct insight into the practical limitations faced by users • Ability to identify new limitations that may not have been identified or ignored by the research experts
Disadvantages	• Participant behaviour may be affected by observer presence
Method No 5	Self-evaluation
Several self-observation sessions in accessing online news sites using various devices were conducted to self-identify the difficulties faced by users of online news site in relation to finding preferred news articles. Additionally another set of observation sessions were carried out to observe how the users are using online site like Amazon which provides a recommendation facility	
Advantages	• Direct insight into the practical limitations faced by users • Ability to identify new limitations that may not have been identified or ignored by the research experts. • Can be used to validate the identified requirements
Disadvantages	• Certain requirements may be ignored be subjective to self-experiences

Table 4.1 - Evaluation of requirement elicitation methods

4.2.1 End User Questionnaire

Structure of the questionnaire

The questionnaire distributed among the potential end users of the system contained questions to identify

- User behaviour patterns in an online news environment – to gather information as to how end users would be using an online news environment
- End user preferences in an online news environment – to understand end user preferred news categories along with how they would like to imply their preferences to news articles
- End user expectations from an online news recommendation system – to understand the non-functional expectations of end user when using an online news recommendation system.

Limitations of questionnaire process

Limitation Id 1	Limited response
Though the questionnaire was made available to a larger number of potential end users of different cross sections of the society not all of the questionnaire were returned back and a substantial amount of the returned were not adequately completed. Therefore the gathered responses may not represent the absolute view of all the end users.	
Limitation Id 2	Feedback credibility
Since the questionnaires were filled by individuals at their own free will there is no way to authenticate the credibility of the feedbacks given. Therefore the gathered responses may not represent the absolute view faced by the end users.	

Table 4.2 - Limitations of the questionnaire

Requirement elicitation of questionnaire

$$\boldsymbol{Successful\ Response\ Rate} = \left(\frac{\text{Surveys Completed}}{\text{No of Surveys Started}}\right) * 100$$

$$\boldsymbol{Successful\ Response\ Rate} = \left(\frac{93}{113}\right) * 100$$

$$\boldsymbol{Successful\ Response\ Rate} = \mathbf{82.30}\ \%$$

Figure 4.1- Normal user successful response rate

According to the above Figure 4.1 the successful response rate was above 82% for the normal user questionnaire and with the respondent quantity and percentage it can be concluded that the questionnaire process was successful.

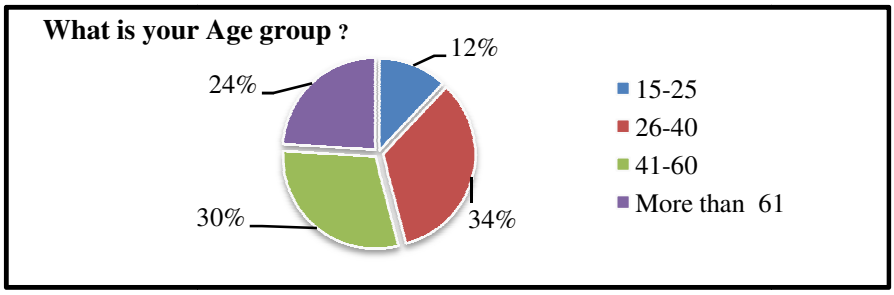


Figure 4.2-Age group representation

The above Figure 4.2, highlights that the questionnaire was mostly answered by the participants of 26-40 age group with a 34% contribution along with 41-60 age group with 30% and more than 61 age group with 24% contributions. These statistics shows that the questionnaire has managed to gather requirements from major age groups.

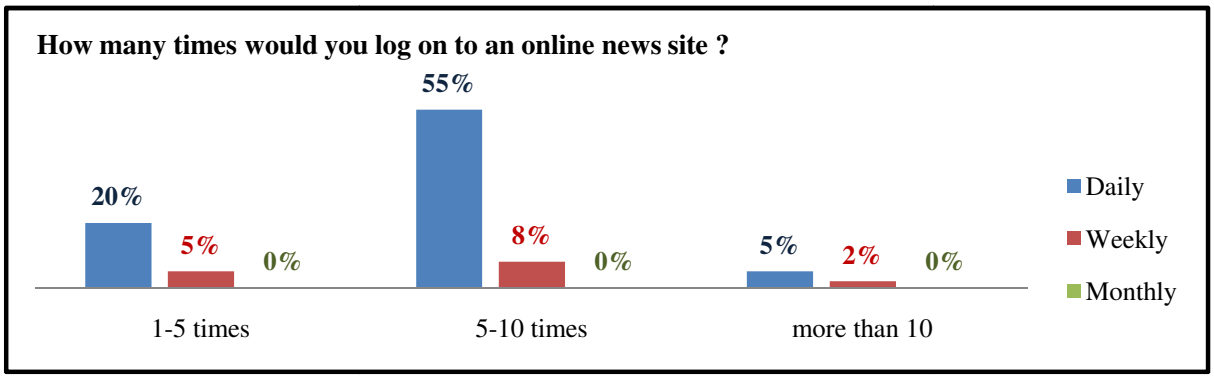


Figure 4.3 - News site access frequency

According to the results shown on Figure 4.3 it can be concluded that the daily log on time is greater than the weekly log on time. Therefore, the results show that there is a tendency on referring online news site than weekly.

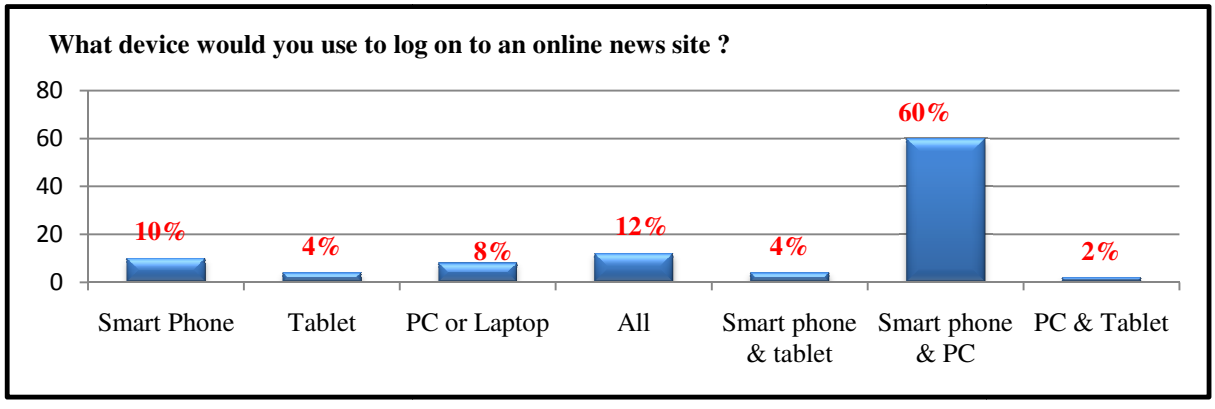


Figure 4.4 - Devices used to access news sites

Figure 4.4 shows that 60% uses smart phones and PC to log on to online news sites and it highlights that the proposed system should support both a standard web view and a mobile view

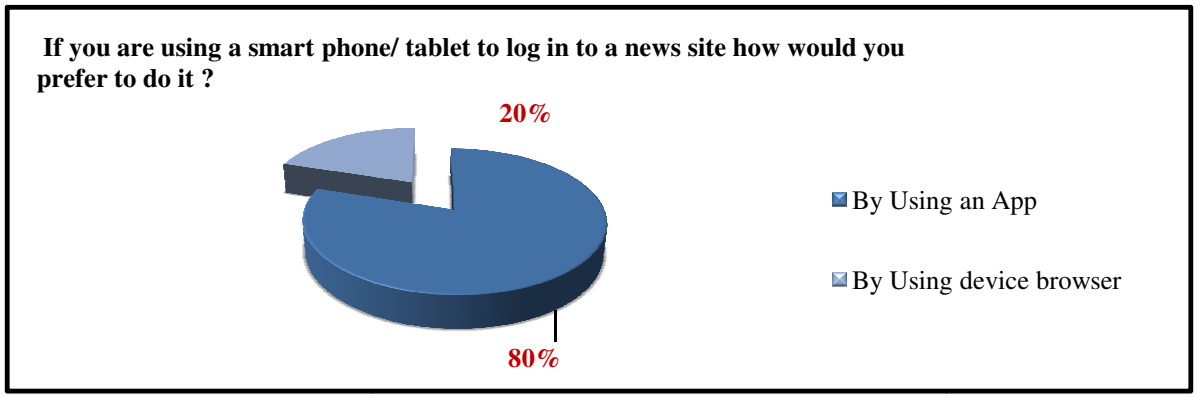


Figure 4.5 - Smart phone user access preference

As shown in the figure 4.5 , it can be clearly seen that the readers prefer to access online news site through an App , rather than using a default browser when using an smart phone

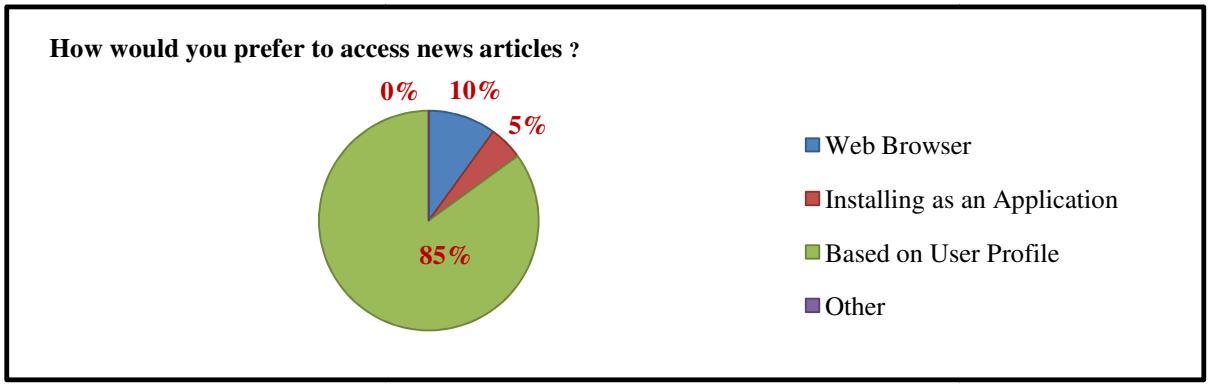


Figure 4.6 - Preference of news article access

As per figure 4.6 it can be seen that the best option is to access online news site is through the user profiles.

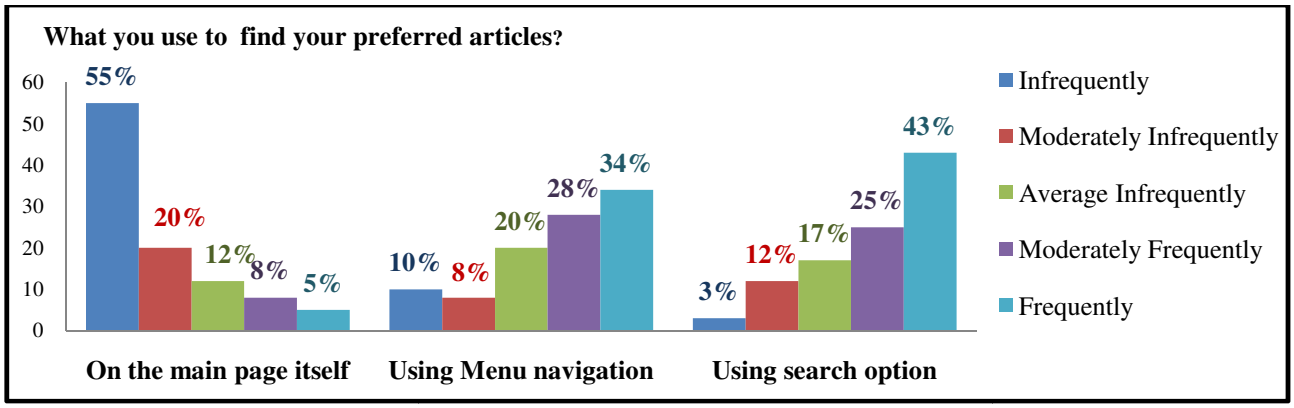


Figure 4.7 - Preferred news article finding method

As shown in the above Figure 4.7 it can be seen that most people find preferred news articles by using search option (43%). This is due to that a reader's preferred news articles usually can't be found on the menu navigation or in the home page.

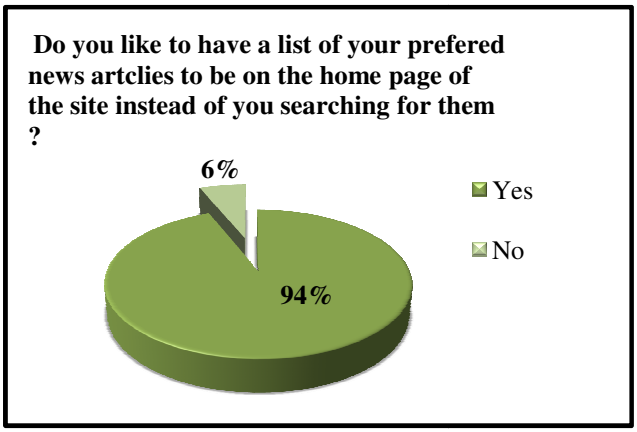


Figure 4.8 - Preference for a personalized news site

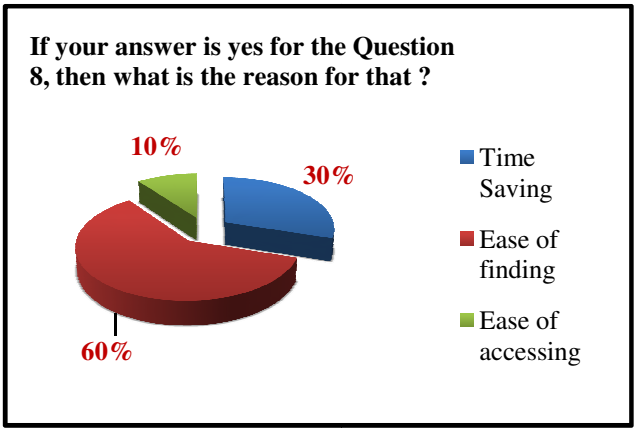


Figure 4.9 - Reasons for a personalized news site

Figure 4.8 highlights the participants liking for a personalized news site and figure 4.9 highlights the reasons for a personalized news site

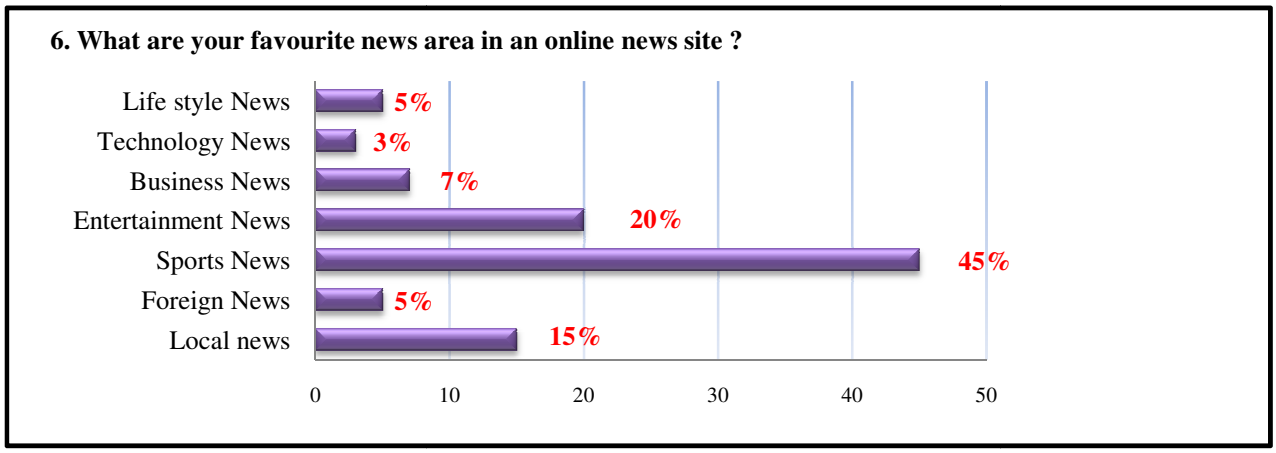


Figure 4.10 - Preferred news categories

Figure 4.10 shows that most readers who participated for the survey prefer to read on sports news (45%) and entertainment news (20%) other than other categories of the news.

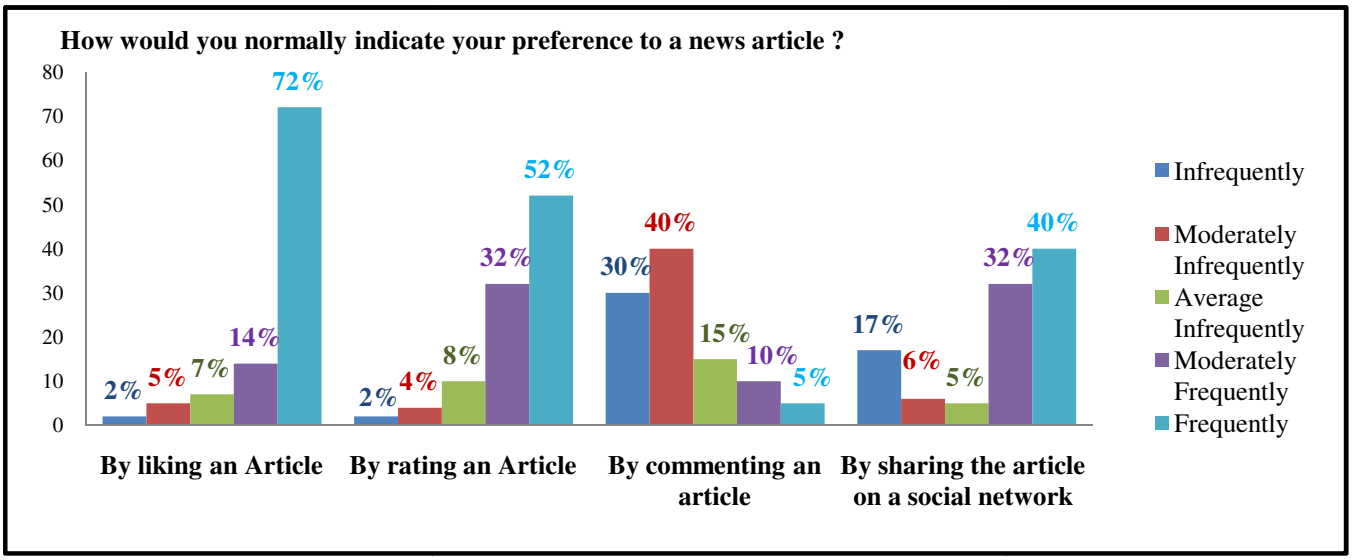


Figure 4.11 - Reader preference indication

According figure 9 readers prefer to indicate their preferences by liking (72%), by rating (52%), by sharing (40%) and by commenting (5%) on an article. So the proposed system should at least have the liking and sharing functionalities for an article.

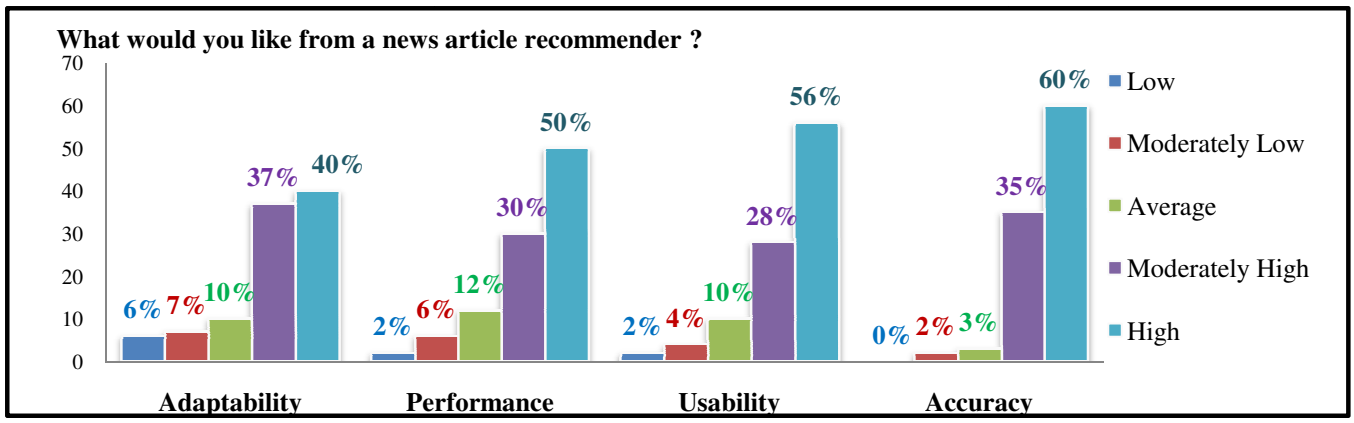


Figure 4.12 - Non-functional requirement rating

Figure 4.12 shows that the readers are mostly concerned on the accuracy and Usability of the news site than the other two features.

4.2.2 Formal Interviews

Structure of the interview questions

Several formal interviews were conducted with the online news site editors and online news page designers of different organizations with the target of identifying the role played in publishing of online news articles, methods practiced for article categorization, standards adopted in the news industry to describe meta data of a news article and general opinion of the online news industry about an online news recommendation system.

Limitations of the interviews

Limitation Id 1	Preservation of self-interests
Certain online news editors and online news page designers were not cooperative in answering certain question as they felt that those questions were trying to meld with their business operations. Therefore the gathered responses may not represent the absolute view of the whole online news industry.	
Limitation Id 2	Lack of knowledge
Certain online news editors were not technically competent enough to understand the functionalities of recommendation system. Therefore they were unable to provide a clear feedback on a number of technical questions.	

Table 4.3 - Limitations of the interviews

4.2.3 Findings Derived From Requirement Elicitation

Table 4.4 provides a summarized set of findings gathered through all requirement elicitation methods discussed.

Id	Finding	Literature review	Questionnaire	Formal interviews	Observations	Self-evaluations
1	Should be scalable enough to support both mobile and standard web users		✓		✓	✓
2	Should use a hybrid recommendation algorithm for the recommendation	✓				
3	Reader profile representation should be text based	✓				
4	Article descriptor representation should be text based	✓		✓		
5	Should be developed specifically to provide a higher accuracy rate and maintain a higher scalability level	✓			✓	✓
6	Should be simple enough for online news page designers to operate the system comfortably			✓		
7	Cold starter problem should be solved using explicit user preference indication	✓			✓	
8	Should be providing above 70% accuracy rate		✓			✓
9	Should capture multiple user interaction types to rate a user	✓	✓			
10	Change of user preference can affect the accuracy of the recommendation	✓				✓

Table 4.4 - Summary of findings

4.3 Stakeholders

4.3.1 Stakeholders and Roles

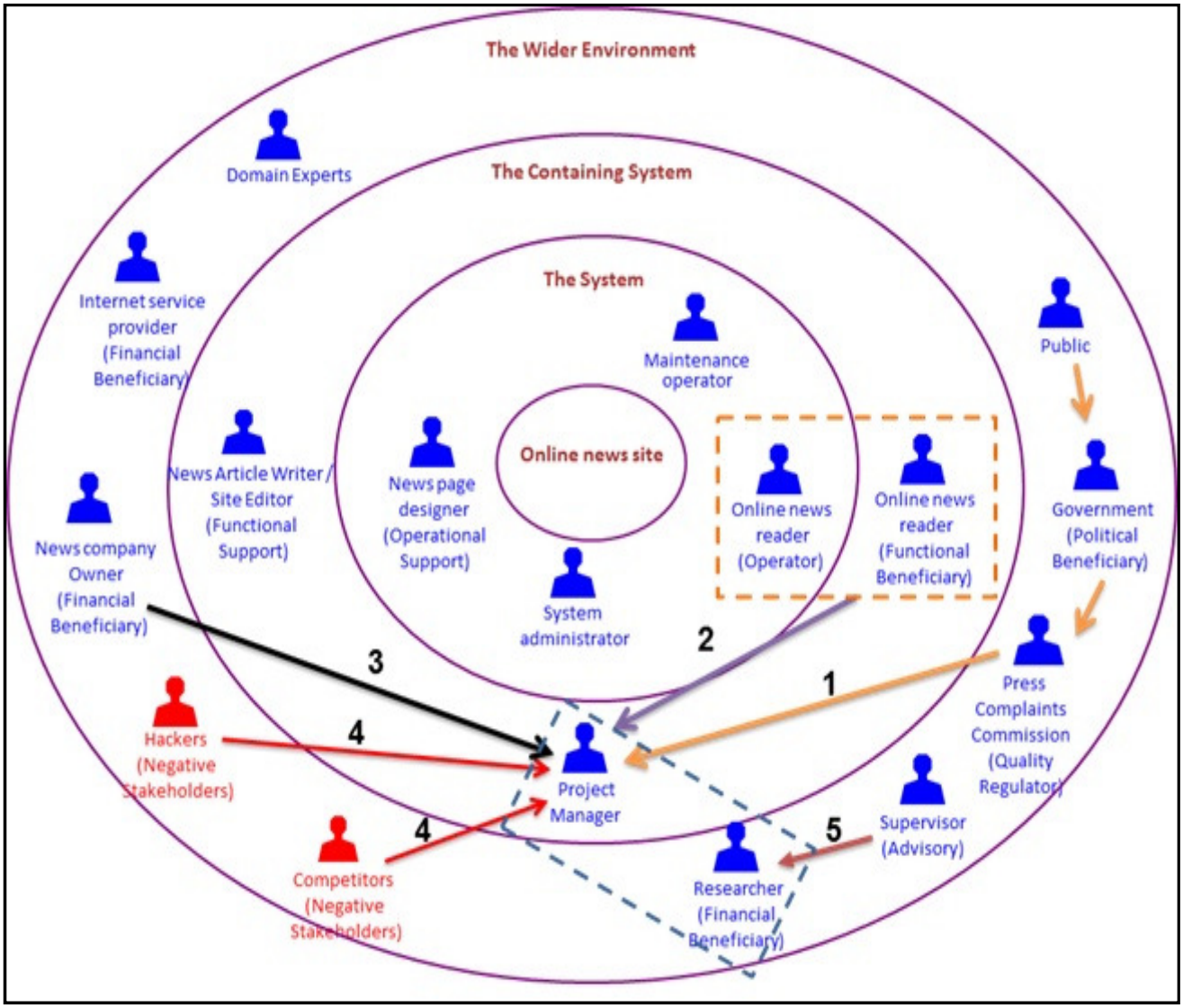


Figure 4.13 - Onion diagram for the proposed system

Above onion diagram represents the identified stakeholders and their roles in proposed solution and below listed are the pressure points of the diagram:

1. Project manager should make sure that the system is met with required standards and qualities.
2. Project manager should make sure that the system is user friendly.
3. Project manager should make sure that the system will be financially beneficial for the on line news company owner.
4. Project Manager should make sure that the system is superior and secure
5. Researcher should make sure that the whole process goes on according to plan and is be able to finish it on time.

Table No 3.4 lists the roles and viewpoints of each identified stakeholder

Stakeholder	Role	Viewpoint
Online news readers	Operator/ Functional Beneficiary	Expects to receive recommended news articles to suit preferences. View, rate, comment on and share news articles.
News page designer	Operational role-support	Wants to publish news articles to online news articles
Maintenance operator	Operational role-Maintenance	Easy maintainability of the system. Easily recover the system on a system failure.
System administrator	Operational role-Administration	Easily administrator the whole process of the system. Easily recover the system on a system failure
News article writer/ Site Editor	Functional support	To publish the article on the news site based on the requirements given.
News site owner	Financial beneficiary	Make money by attracting more readers and use that customer base to attract advertisements.
Internet service provider	Financial beneficiary	Increase the profits by providing efficient Internet service for the readers
Competitors	Negative	Would want to identify drawbacks of the system and implement a better solution.
Hackers	Negative	Hack the system and make it unresponsive
Public	Negative	To point out the weaknesses of the solution and expect the researcher to correct it.
Domain experts	Expert	To provide expert opinion about the technologies and methodologies used for the project.
Government	Political Beneficiary	To make sure that articles published on the site are adhering to the standards and code of ethics set.
Press complaints commission of Sri Lanka	Quality regulator	To make sure that articles published on the site are adhering to the standards and code of ethics set. Accepts any complaints from the public against published articles
Supervisor	Advisory	Provide advices and guidance to successfully finish the project.
Project Manager/ Researcher	Managerial/ Financial beneficiary/ Intellectual	Wants to manage the constraints to assure smooth flow of the project Want to develop a bug free system which meets the requirements

Table 4.5 - Stakeholders and roles

4.3.2 The Context Diagram

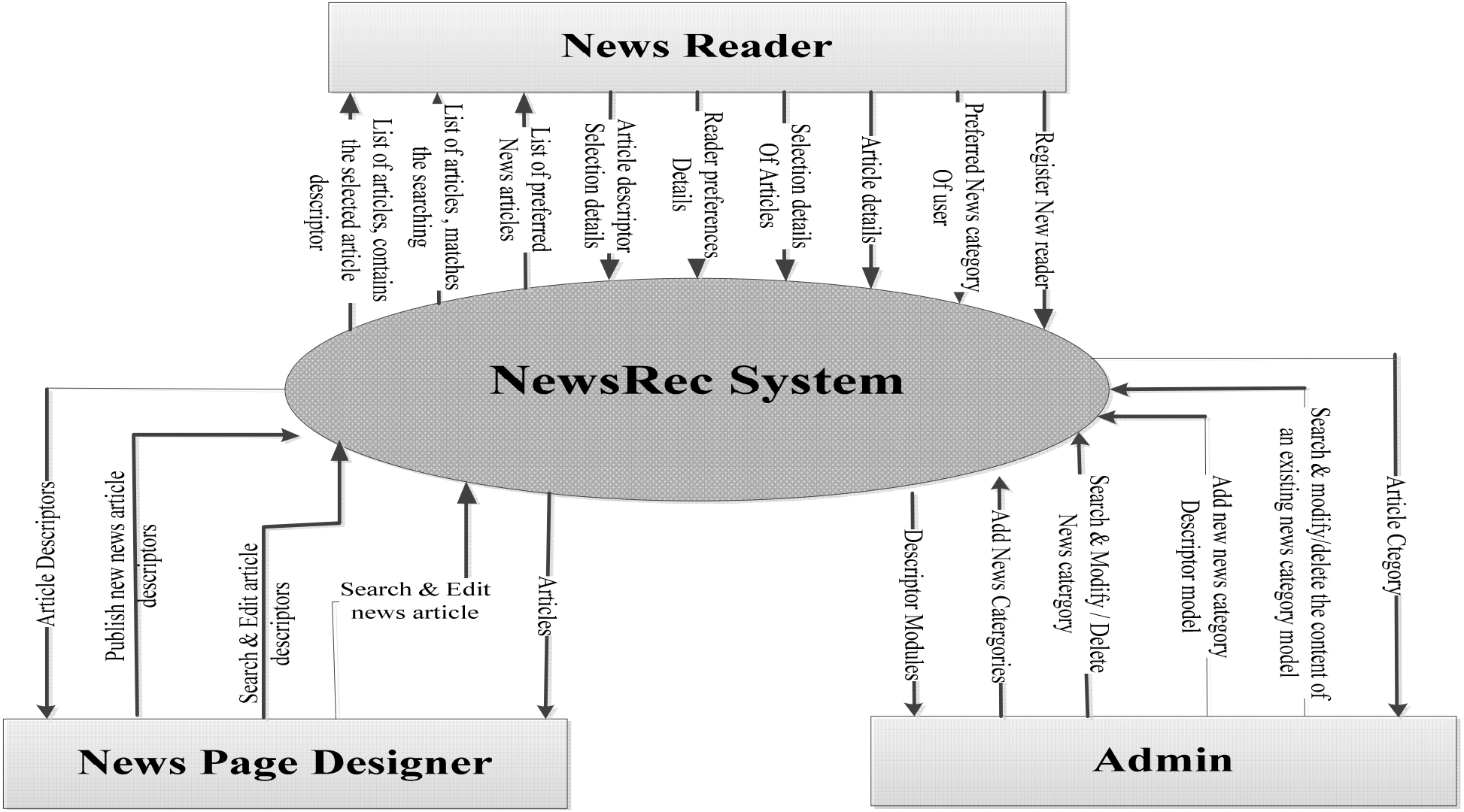


Figure 4.14- Context diagram for NewsRec system

4.4 Use Case Diagram

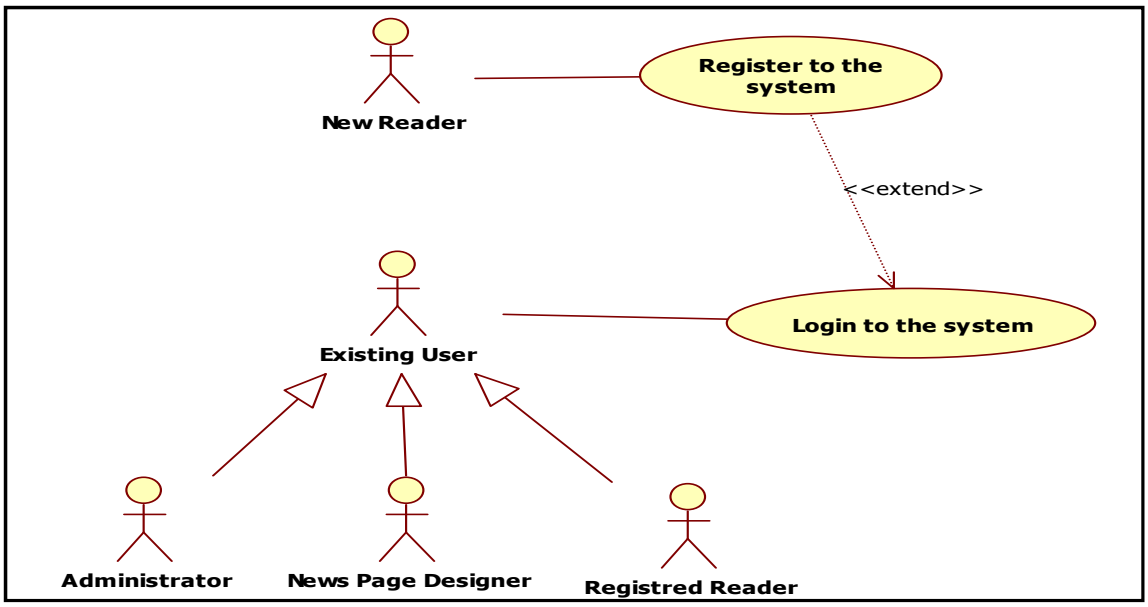


Figure 4.15 - Use case diagram for System login

The above diagram illustrates the 4 types of actors that could interact with the NewsRec system and their login process to the system. A new reader initially would have to register with the system by providing the readers' demographic information along with a unique username and a password of choice. After a successful registration the new reader will become a registered reader will be able to login to the system. All the 3 types of existing user roles should login to the system to use unique features provided to each user role. The below diagram provides an overview of the features that will be provided to each existing user role. A registered user will be able to search for news articles by clicking article descriptors provided or by providing a keyword for the system to search and will be provided with a list of articles which contains the article descript or the keyword in the article header. At each login the system will generate a list of recommended articles that matches the readers current reader profile interests using content based and collaborative filtering. When an article is selected from either of the lists the reader can indicate reader preference by various actions and the system would capture the selection and preference indication method information to update the user reading profile and the rate value for that selected article. Administrator will be able to carry out tasks like adding a new news category or a new article descriptor model and updating and deleting of existing news categories or news descriptor models. The main task of a News page designer is to add a new news article to the system along with an appropriate article descriptor model and descriptor values. Additionally a News page designer can delete or modify the content of a published news article or modify the article descriptor values of a published news article.

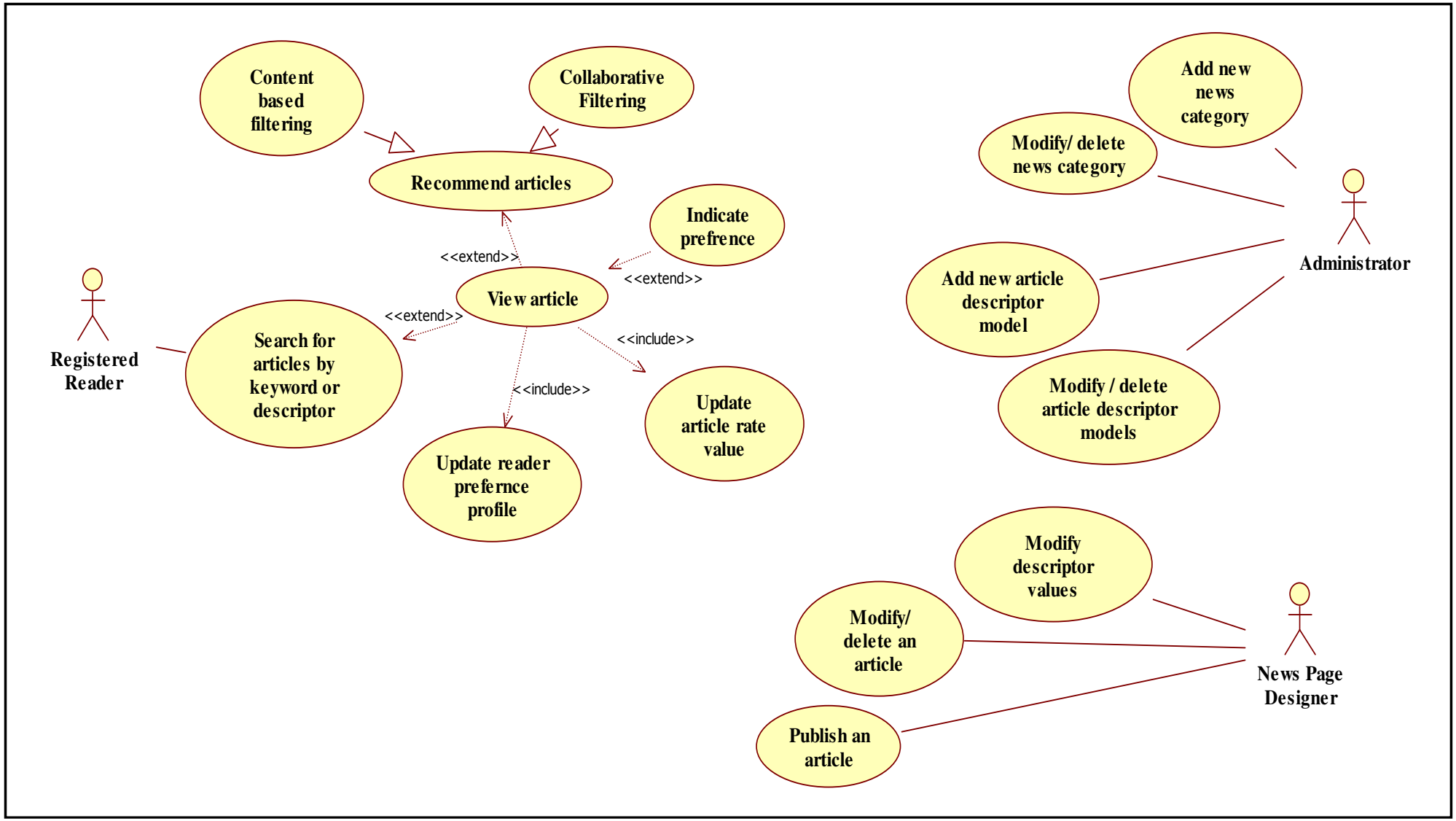


Figure 4.16 - Use case diagram for NewsRec system

4.4.1 Use Case Descriptions

Use case name	Register to the system
Description	This use case is to show the process where a news reader register to the system
Participating actors	Unregistered news readers
Pre-conditions	1. System should be functional 2. Reader should not have registered to the system
Extended use cases	Login to the system
Included use cases	None
Main flow	1. Reader enters registration details along with a unique username and a preferred password. 2. Reader specifies the preferred news category 3. System verifies that the user name doesn't exist in the database. 4. System verifies the password. 5. System accepts user registration 6. System generates a user preference profile with the demographic data of the user and the preferred news category. 7. Use case terminates.
Alternative flows	None
Exceptional flows	E1: Reader enters registration details along with an existing username and a preferred password. 1. An error message is thrown, and the use case ends in failure. E2: During the execution of the use case Internet connection interrupts or fails. 1. Use case ends in failure.
Post conditions	System stores the registration details of the reader and the generated user preference profile.

Table 4.6- Use case description for register to the system

Following use case description represents the scenario where a user tries to login to the system.

Use case name	Login to the system
Description	This use case describes the flow of events and activities that will be carried out every time when a user tries to login to the system.
Participating actors	1. Registered reader of the system 2. Administrator of the system 3. News page designer
Pre-conditions	1. System should be functional.
Extended use cases	None
Included use cases	None

Main flow	1. User tries to login to the system by providing a user name and a password to the system. 2. System will query for the username and fetch the password for that username 3. System will compare the given password value with the recorded password. 4. System authenticates the user and direct the user to appropriate user panel 5. Use case terminates
Alternative flows	None
Exceptional flows	E1: In step2, system finds that the given username doesn't exist. 1. An error message is thrown, and the use case ends in failure. E2: In step 3, the system finds that the given password doesn't match 1. An error message is thrown, and the use case ends in failure. E3: During the execution of the use case Internet connection interrupts or fails. 1. Use case ends in failure.
Post conditions	If the user was a 1. Registered user : the system will display a list of recommended news articles 2. Administrator : the system will direct the user to the administrator panel 3. News page designer : the system will direct the user to the news uploading panel

Table 4.7 - Use case description for login to the system

Please refer to Appendix B of this document for rest of the use case descriptions.

4.5 Functional Requirements

Requirement Prioritization

Due to time and resource restrictions it might be difficult to implement every identified requirement in the requirement elicitation process. So the identified requirements are prioritized in order to identify most important and less important requirements. Following table shows the priority levels and the descriptions of it.

Priority level	Description
Critical (**C**)	Functional requirements of the 'Critical' category represent the core functionality of the system, and it is mandatory to be implemented
Important (**I**)	The functional requirements of this category are not essential, but they are considered to be necessary.
Desirable (**D**)	The requirements which are intended to implement in further developments or out of the scope for the project.

The following table shows the identified functional requirements along with the priority levels

FR No	Requirement description	Priority Level	Use case mapping
FR1	Any new reader should be able to register with the system by providing demographic data of the reader along with a unique password and a preferred password	**C**	Register to the system
FR2	At the time of the registering to the system any new reader should be able to specify his/her preferred news category	**C**	Register to the system
FR3	Any registered reader should be able to login to the system using his/her username and password and the system should authenticate the user.	**C**	Login to the system
FR4	If the authentication fails the system should display an error message to the reader citing the reason for the authentication failure	**C**	Login to the system

FR5	After a successful login for each registered reader the system should display a list of preferred news article titles sorted according to the rating values of the articles along with article descriptors.	**C**	Recommend articles
FR6	A registered user should be able to search for news articles by entering a string of characters. The entered string will be matched against the article title or article descriptor values	**C**	Search for articles by keyword or article descriptor
FR7	For a successful search the system should display a list of articles titles sorted according to the rating values along with their article descriptors.	**C**	Search for articles by keyword or article descriptor
FR8	If the searching fails due to non-existence of the entered string an error message must be displayed.	**C**	Search for articles by keyword or article descriptor
FR9	A registered user should be able to click-on an article descriptor value that is listed on the recommended or searched news article title list and the system should display a list of article titles containing that article descriptor	**C**	Search for articles by keyword or article descriptor
FR10	A registered user should be able to click-on and view the content of an article that is listed on the recommended, searched, article descriptor filtered news articles lists.	**C**	View article
FR11	A registered user should be able to rate a viewed news article	**C**	Indicate preference
FR12	A registered user should be able to comment on a viewed news article	**I**	Indicate preference
FR13	A registered user should be able to share a viewed news article on a social network like facebook.	**D**	Indicate preference
FR14	A registered user should be able to recommend a viewed news article to another reader.	**D**	Indicate preference
FR15	The System should be able to track reader's article selection details	**C**	Search for articles by keyword or article descriptor
FR16	The System should be able to track reader's article description selection details	**C**	Search for articles by keyword or article descriptor
FR17	The system should update user's preference interest	**C**	Update user profile
FR18	The system should update rate value of articles	**C**	Update article rate value

FR19	News page designer should be able to publish a new news article	**C**	Publish a new news article
FR20	News page designer should be able to assign article descriptors to a newly published news article and descriptor values	**C**	Publish a new news article
FR21	News page designer should be able to modify the assigned article descriptors of a published news article	**I**	Modify descriptor values
FR22	News page designer should be able to update the content of an existing news article.	**D**	Modify/ delete an article
FR23	News page designer should be able to delete an existing news article.	**D**	Modify/ delete an article
FR24	Administrator should be able to add new news categories to the system along with suitable article descriptors.	**C**	Add new news category
FR25	Administrator should be able to create new article descriptor models	**D**	Add new article descriptor model
FR26	Administrator should be able to modify article descriptor models	**D**	Modify/delete article descriptor models
FR27	Administrator should be able to delete article descriptor models	**D**	Modify / delete article descriptor models
FR28	Administrator should be able to delete existing news article categories	**D**	Modify/ delete news category

Table 4.8 - Function requirements of NewsRec System

4.6 Non Functional Requirements

Following are the identified non-functional requirements of the **NewsRec** System. The purpose of non-functional requirements is to provide a better user satisfaction, improve performance and to enhance maintainability.

NFR 1	Accuracy
In any recommender system the main non-functional requirement is the accuracy of the recommendations given to its users. **NewsRec** system too should focus on achieving a higher accuracy rate as the trust of the readers towards the system depends highly on the accuracy of the recommendations done. For a newly registered user the system expects to provide accuracy above 40% and for an already registered user who has substantially interacted with the system it expects to provide accuracy above 70%.	
NFR2	Scalability
Another major non-functional requirement of the proposed **NewsRec** system is to maintain a higher level of scalability. It's expected that the workload of the system will increase as time goes on with the number of readers subscribing for the system and the number of articles that will get accumulated. So the system should be scalable enough to handle to increasing workload in a satisfactory manner.	
NFR3	Performance
System should respond quickly for the user requests and additionally the system should be light weight to support resource constraint devices like smart phones and work properly without crashing on all types of devices and platforms.	
NFR4	Usability
The system should ensure higher level of usability by making the prototype usable for new users with minimum training. The user interface has to be simple and visually appealing to use. Help messages will be provided where necessary to improve usability. A new user should take maximum of 1 day to get familiar with the functionalities and the capabilities of the **NewsRec** system.	

Table 4.9 - Non functional requirements of NewsRec System

4.7 Chapter Summary

This chapter mainly focused on gathering correct and adequate requirements expected from the proposed online news recommends system. Chapter started with a detailed description of the various techniques that were used to gather requirements from identified stakeholders. Questionnaires and interviews were the two main elicitation techniques used, other than the literature review, observations and self-evaluation techniques. Later all the collected results were analysed and documented to refine user requirements, identify the project scope and formulate a solution to the problem. The key stakeholders and their roles were identified via an onion diagram while a context diagram paved way to better understand the data flows between the proposed NewRec system and its stakeholders. Use case and Use case descriptions were included into this chapter to clearly identify the stakeholders' interactions and expectations from the system. Then the chapter went to document a comprehensive list of functional and non-functional requirements. Next chapter will be the System Architecture and Design, which will be focusing on the detailed system design of the prototype based on the requirements identified from this chapter.

Chapter 5:System Architecture & Design

Contents

5.1 Chapter Overview

Having discussed requirement elicitation process in the previous chapter, this chapter focuses on the architecture and the design of the NewsRec system. Chapter will first discuss the high level architecture and workflow of the system, introducing the sub components which form the complete system. This will be followed by the component level design of the system which focuses the functionalities of each sub component in order to meet the requirements in requirement specification. Finally, integration process of these sub components into main system will be discussed

5.2 High Level Design

The high-level design provides a fairly accurate picture of the end goal and the process that is needed to get there, allowing filling in the details later on the exact steps and configurations that will be needed for everything to fit together

5.2.1 Rich Picture of the NewsRec System

A Rich Picture is an approach to explore a situation and define a high-level solution and then express the solution through diagrams to create a preliminary design model that can be further expanded with an appropriate design methodology. Figure 5.1 illustrated the Rich picture diagram of the NewsRec system.

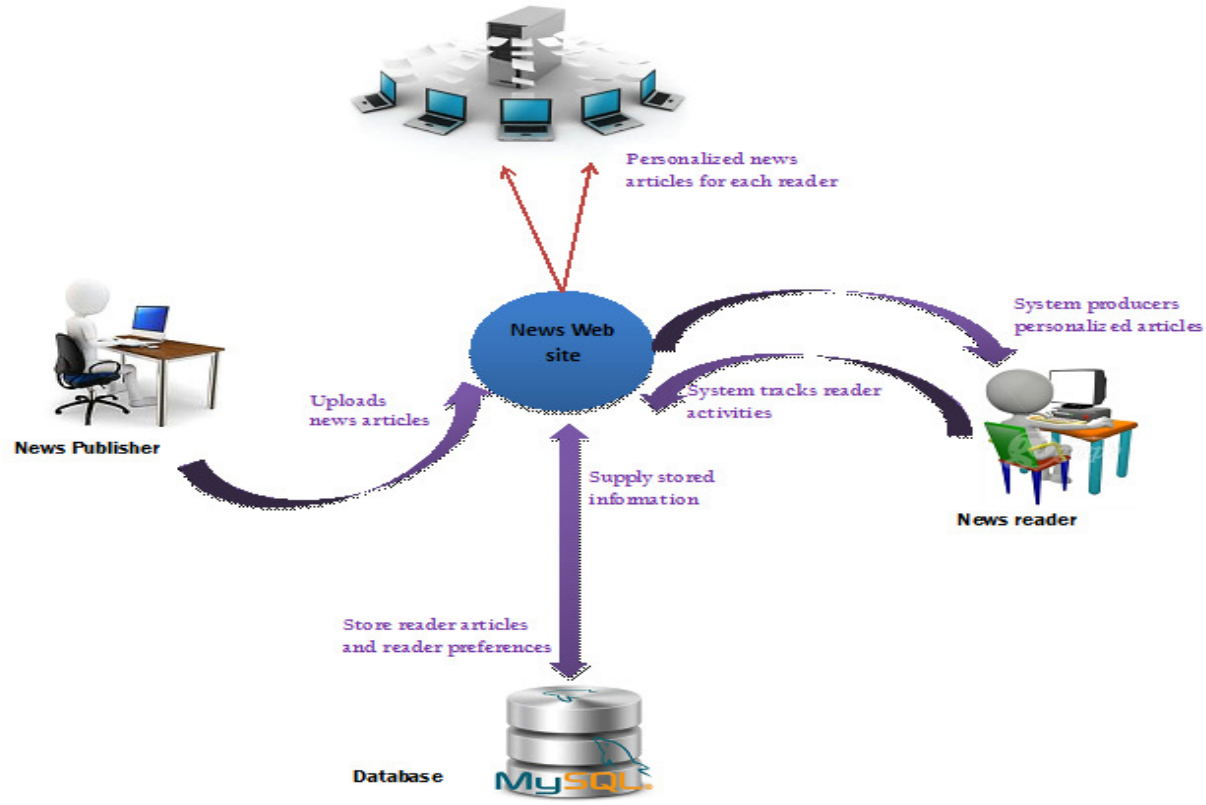


Figure 5.1-Rich picture diagram of the NewsRec system

According to the above Figure 5.1 the main stakeholders of the NewRec system are the news publisher and the news reader. News publisher would upload news articles to the system along with suitable article

descriptors and all the uploaded articles and article descriptors will be saved in a database. When news readers log in to the NewsRec system, news articles will be recommended for the logged on readers by the system based on the previous reader preferences. In order to understand the preferences of readers, reader activities like selection of article and liking of articles will be tracked and saved in the database.

5.2.2 High Level Architecture

Three - Tier architecture is an industry-proved software architecture model which support modelling of enterprise-level client/server applications. Main advantage of the three-tier architecture is the independency of the three tiers from allowing any of the three tiers to be upgraded or replaced independently. According to Marston(2014) three-tier architecture is capable of resolving issues like scalability, security, fault tolerance and due to the above mentioned reasons the high level architecture of the proposed NewsRec System was modelled using the three-tier architecture.

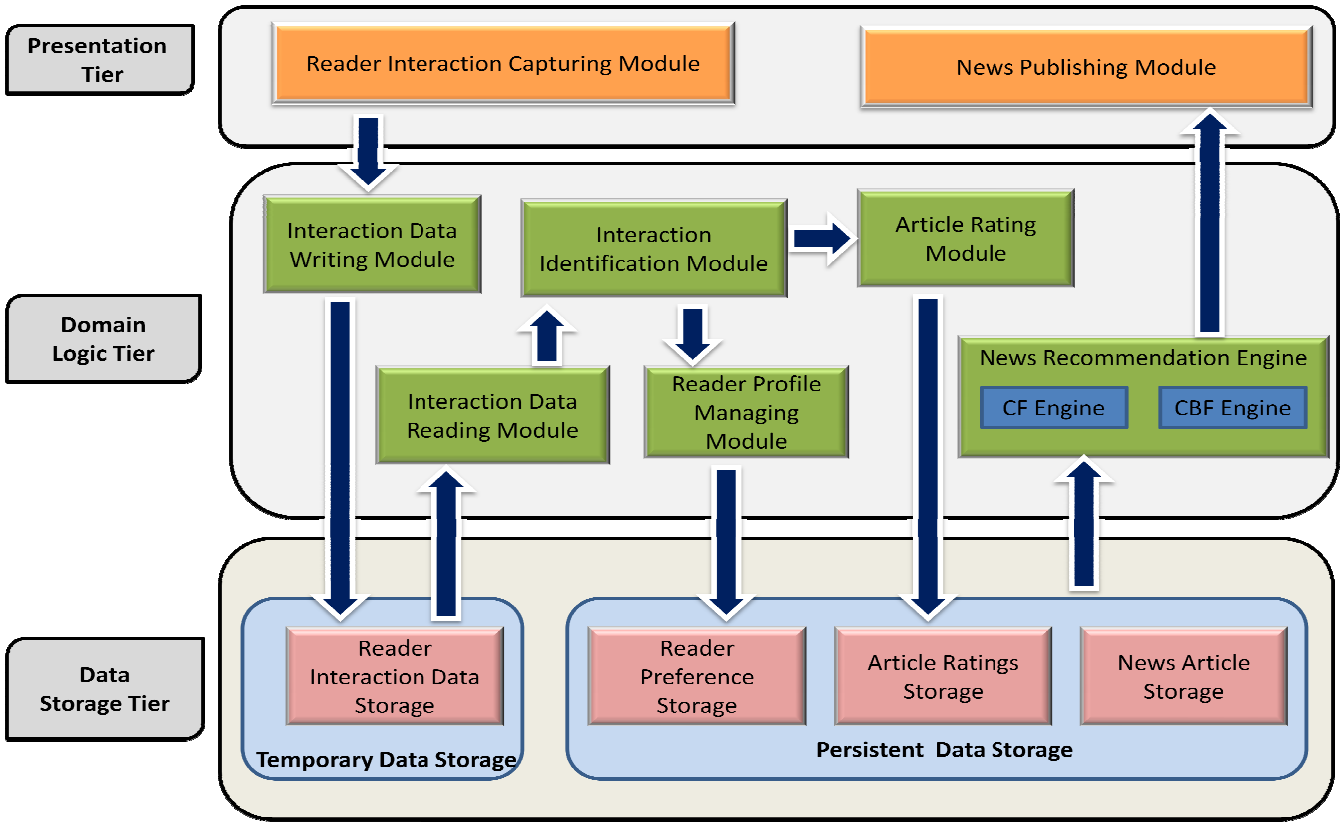


Figure 5.2: High Level Architecture Diagram

According to the above diagram presentation tier contains 2 modules which would be exposing the system to the outside world.

- **Reader Interaction Capturing Module:** This module will be used to capture the various interactions the reader would be performing on the presented news articles.
- **News Publishing Module**: This module is used to present news articles to various devices and platforms.

The domain logic tier contains several modules which would carry out the business logic of the proposed system.

- **Interaction Data Writing Module:** This module is used to write the reader interactions captured through the reader interaction capturing module to dynamic data storage.
- **Interaction Data Reading Module:** This module is used to read the stored reader interactions from the dynamic data storage periodically and feed those interaction details to the interaction identification module
- **Interaction Identification Module:** The purpose of this module is to filter out the unwanted user interactions of the reader and then categories those filtered interactions in to reader profile related interactions and article rating related interactions**.** Reader profile related interactions will be fed to the reader profile managing module while the article rating related interactions would be fed to the article rating module.
- **Reader Profile Managing Module**: This module would manage the preference details of the registered readers and cluster the readers with similar interests and save the updated preference details and cluster details.
- **Article Rating Module:** This module would handle the rating functionalities of an article based on the article rating interactions performed by the reader and save the new article rate values.
- **News Recommendation Engine:** The recommendation of news articles to the users will be performed by this module and this module contains two sub modules known as the CF engine and the CBF engine.CF engine would carry out the recommendation based on the content alignment of the user profile and article descriptors and CBF engine would carry out the recommendations based on the user clusters and article rate values. .

Data storage tier contains with 4 data storages which would store the information that have to be used for the recommendation process.

- **Reader Interaction Data Storage –** This storage would store the various interactions the readers would be performing on the system.
- **Reader Preference Storage –** This storage would store all the reader preference information along with reader clustering information
- **Article Ratings Storage -** This storage would store all rating information about the articles

  News Article Storage – This storage would perform the functionality of a repository where the actual articles will be stored at.

5.2.3 Article Descriptor and Reader Profile Modelling

Article descriptor modelling

Article descriptors are modelled as either unique interest or common interests and a given article can be described using a set of unique or common interests. Common interests are keywords that can be used to describe an article and which would be able attract the interest of substantial amount of readers like the word

"Cricket". Unique interests are keywords that can be used to describe an article but wouldn't be able attract the interest of substantial amount of readers like the name of an upcoming sport star.

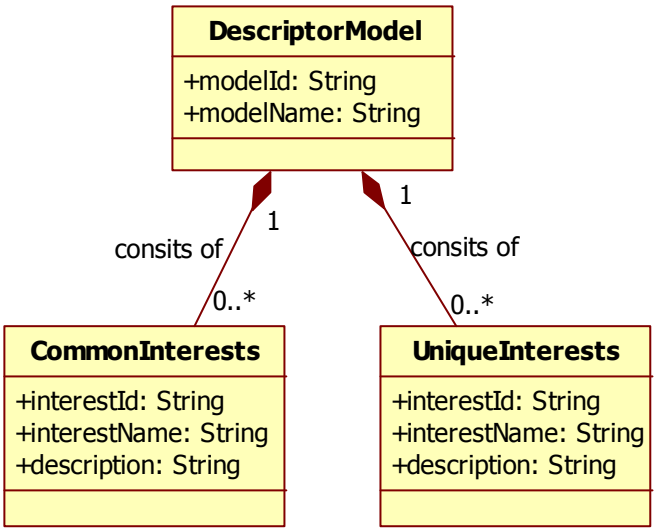


Figure 5.3-Article descriptor model

Reader profile modelling

A reader profile is modelled as a set of established interests, recent interests and previous interests where each interest would be consisting of several common and/or unique interests.

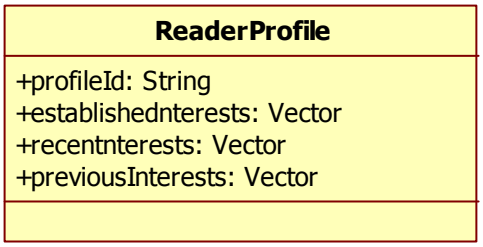


Figure 5.4-Reader profile model

5.2.4 Recommender Algorithm Design

According to Burke, R. (2007) there are 7 basic design approaches that can be considered when designing a hybrid recommender system and it was decide to adopt the Cascade hybrid approach where one recommendation systems purify the suggestions and recommendations that are presented by another recommendation system. In adopting this approach to the NewsRec system the collaborative filtering approach would filter the highest rated articles within the cluster by the specified common interests of the reader making the collaborative filtering suggestion to be filtered back again using the content based filtering approach.

5.3 System Design

5.3.1 Selection of Design Methodology

Design methodology is a set of procedure that one follows from the beginning to the completion of the software development process. The nature of the methodology is dependent on several main factors like software

development environment, type of the software being developed, the requirements of the users, and the time schedule. Though several design methodologies are available the most widely used two methodologies by both academia and industry are.

- Structured Systems Analysis and Design Method (SSADM)
- Object-Oriented Analysis and Design Method (OOADM)

Application of structured systems analysis and design methodology

SSADM approach supports development of function oriented software. Though the SSADM is not widely used in the industry as key design methodology due its incapability in modelling complex and larger software it is a respectable analysis and design methodology when it comes to modelling fairly small and simple software.

The reader interaction capture and interaction data writing modules are two simple and small modules which would be used to capture user interactions coming into the system and store that information on temporary data storage. Since It was decided to implement those two modules using function oriented Java Script language (please refer to Chapter 7.2) and due to their simplicity it was decide to model those two modules using the SSADM approach.

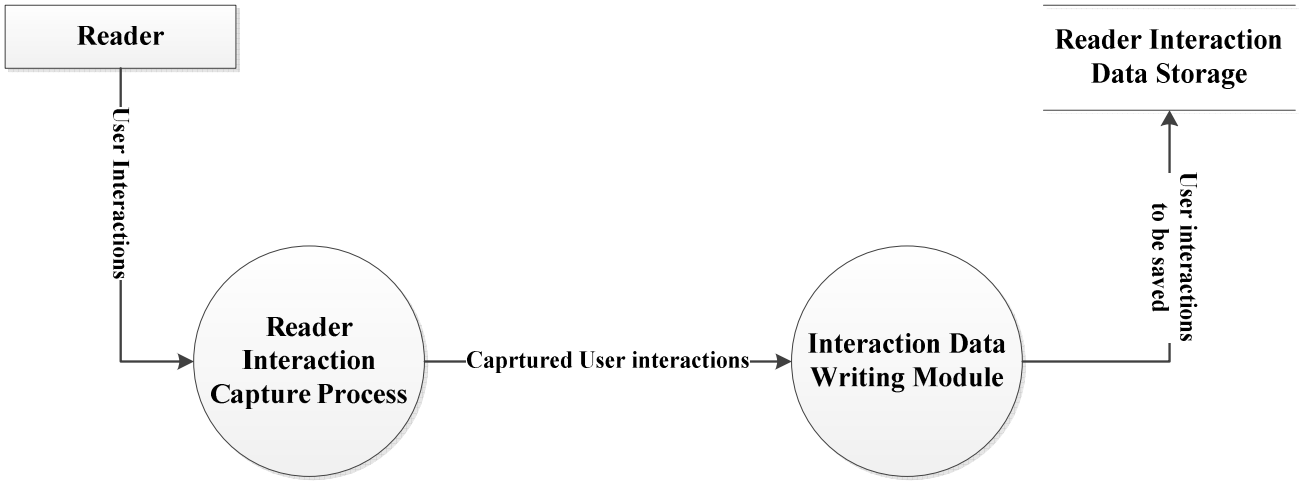


Figure 5.5-DFD diagram of interaction capturing and storing

Application of object oriented analysis and design methodology

OOADM approach supports development of object oriented software. OOADM approach is widely used in the industry as key development methodology due its capability in modelling complex and larger software. OOADM is capable of breaking the complex software system down into its various objects, combining the data and the functions that operate on the data into a single unit

Since the implementation of the rest of the other modules is dependent on the JADE framework (please refer to Chapter 7.2) and complex algorithms it was decided to model the rest of the system using the OOADM approach and Unified Modelling Language (UML) was selected as the technique for representing the model.

5.4 Domain Model of the NewsRec System

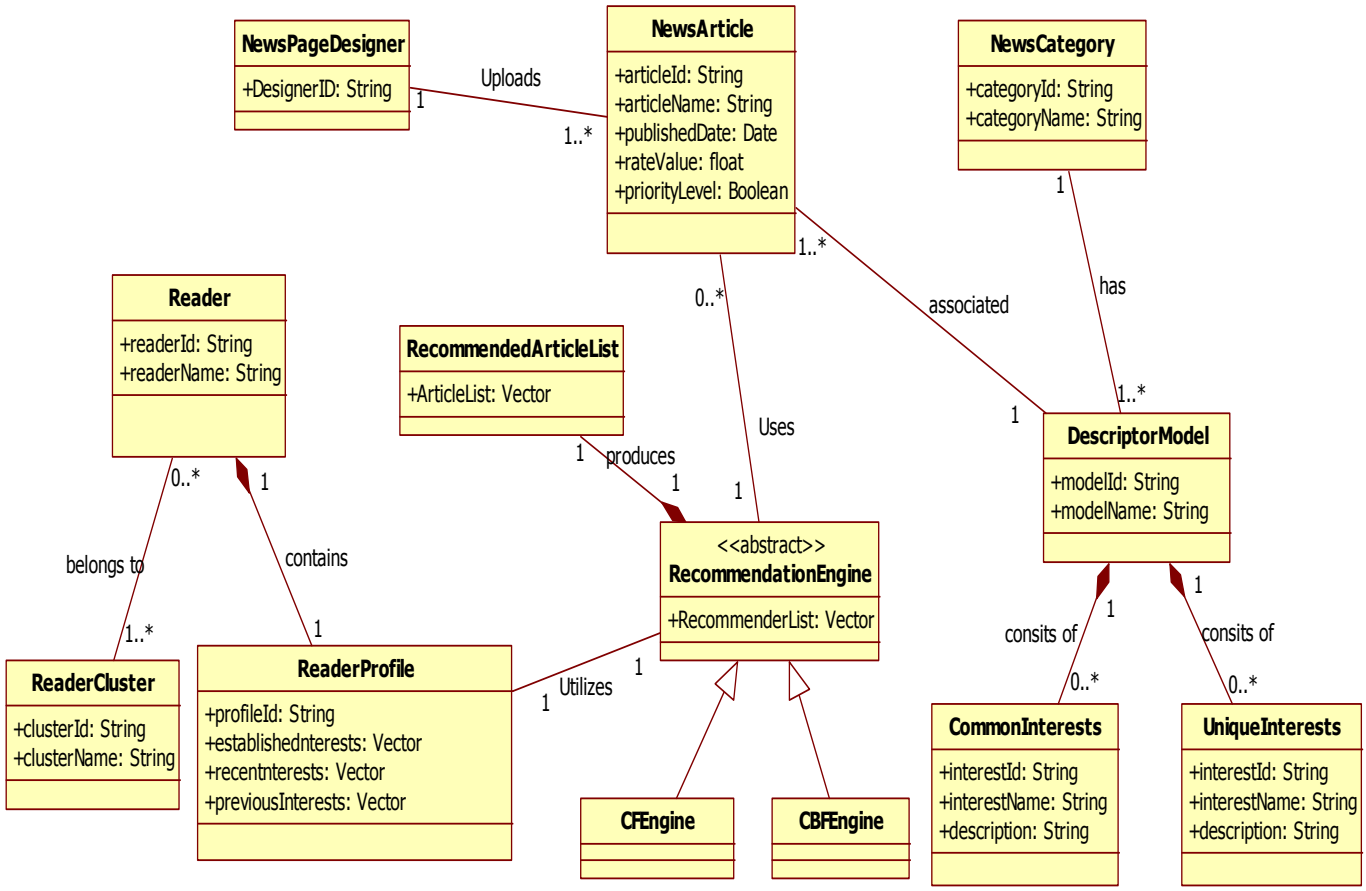


Figure 5.6-Domain Model of NewsRec System

Table 5.1 provides an overview of the classes contained in the domain model.

Class	Description
Reader	A news reader is modelled through this class
ReaderCluster	This class represents the different reader clusters created by the clustering
ReaderProfile	The reading preference details of a reader is modelled through this class
NewsArticle	The details of a news article is modelled by this class
NewsCategory	Different news categories of the online news environment are modelled thorough this class
DescriptorModel	Various descriptors for a news article are modelled through this class
CommonInterests	Common interests defined are modelled through this class
UniqueInterests	Unique interests defined are modelled through this class
RecommendationEngine	This class works as an abstract class specifying how the recommendation process should be carried out

CFEngine	Content based recommendation is carried out by this class
CBFEngine	Collaborative based recommendation is carried out by this class
RecommendedArticleList	This class is used to represent the list of recommended articles

Table 5.1-Overview of the domain model classes

5.5 Sequence Diagrams

New reader registration

Figure 5.7 represents the sequence of events that would take place when a new reader tries to register with the system. The UserRegistertrationListener class which listens to the incoming data stream for new reader registrations would create a new UserRegisterAgent object when a request for a new reader registration arrives. Then that object would register with the RedisClient and get the user registration details from the Redis server. With the user registration details the UserRegisterAgent would create a new object of Reader class. The Reader object then will be passed to the GenericReaderManager class and that object will be passed to HiernateReaderManager class to be written to the persistence data storage.

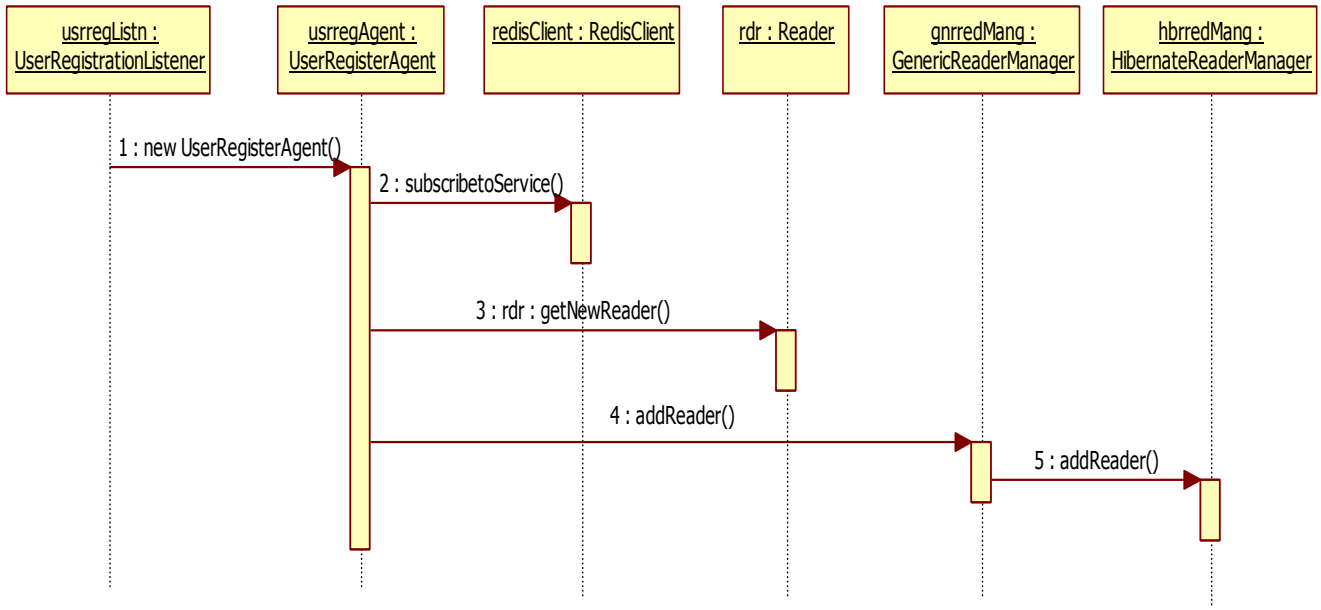


Figure 5.7-Sequence Diagram for New Reader Registration

Update article rate value

Figure 5.8 describes the sequence of events that would take place when the system would tries to update the rate value of an article. The ArticleRateValeUpdateListener class which listens to the incoming data stream for article rate value updates would create a new ArticleRateUpdateAgent object when a request for a new article rate update arrives. Then that object would register with the RedisClient and get the article update details from the Redis server. With the article update details the ArticleRateUpdateAgent would create a new object of Article class. The Article object then will be passed to the GeneicReaderManager class and that object will be passed to HiernateReaderManager class to be written to the persistence data storage.

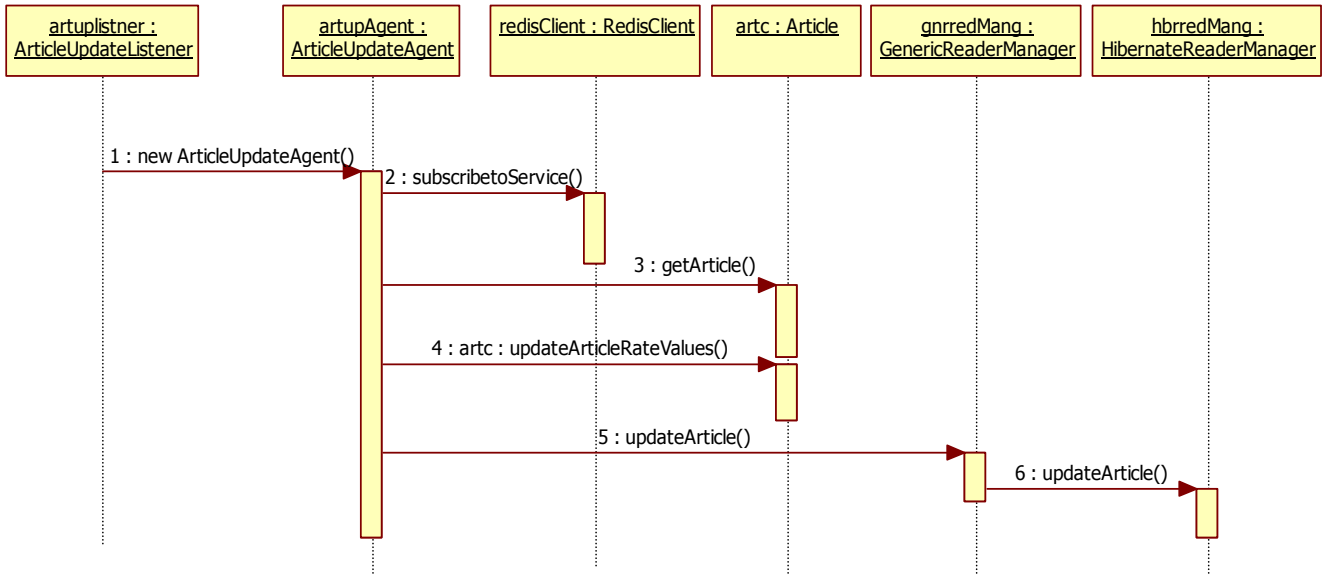


Figure 5.8-Sequence Diagram for Update Article Rate Value

Update reader profile

Figure 5.9 describes the sequence of events that would take place when the system would tries to update the reader profile. The UserProfileUpdateListener class which listens to the incoming data stream for user profile updates would create a new ReaderProfileIndexerAgent object when a request for a new user profile update arrives. Then that object would register with the RedisClient and get the user profile update details from the Redis server. With the user profile update details the ReaderProfileIndexerAgent would create a new object of Reader class and the Reader class would create a ReaderProfile class. The reader profile will be updated through the updateEshtablishedInterests and updateRecentInterests methods and after that The Reader object then will be passed to the GenericReaderManager class and that object will be passed to HiernateReaderManager class to be written to the persistence data storage.

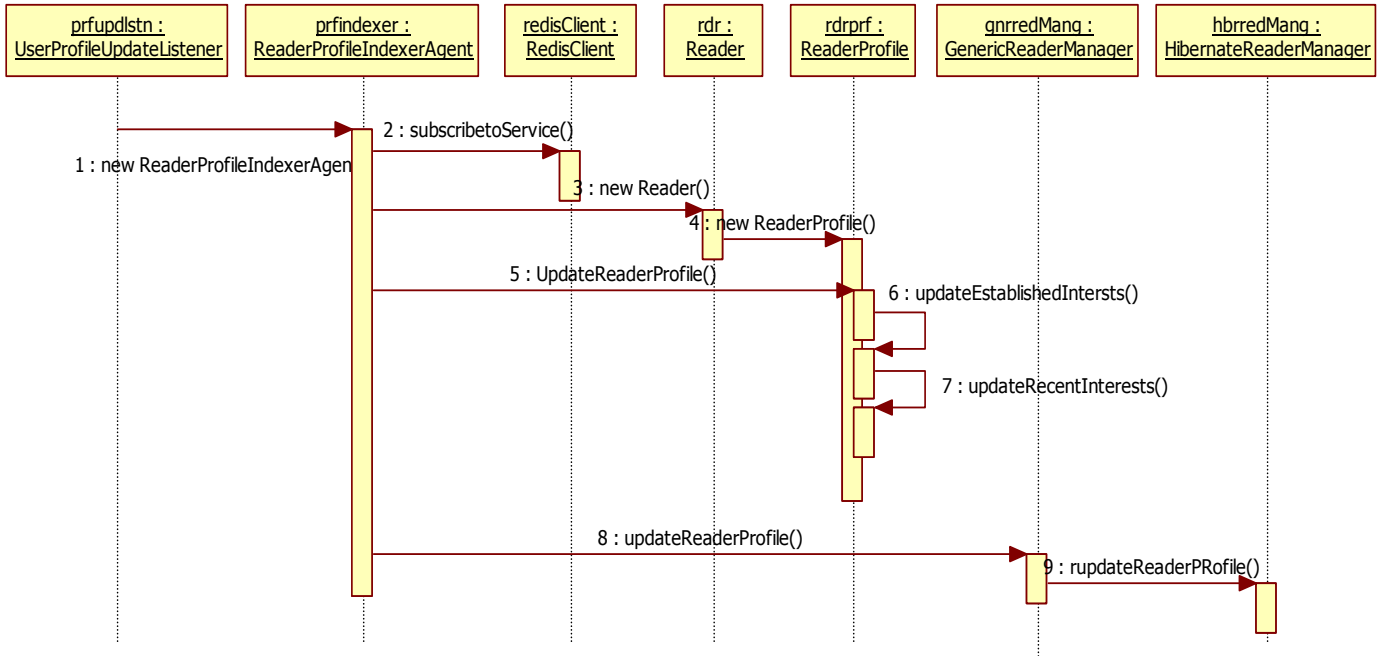


Figure 5.9-Sequence Diagram for Update Reader Profile

Recommendation of articles

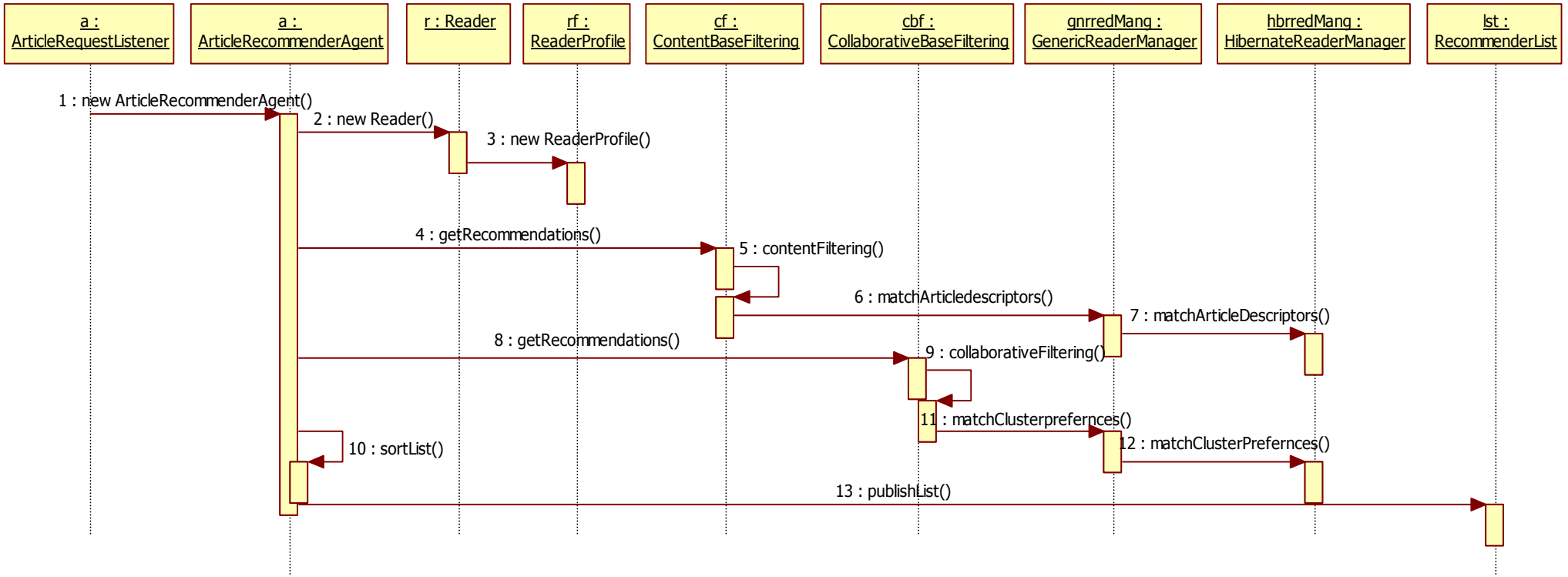


Figure 5.10-Sequence Diagram for Recommendation of Articles

5.6 Design and Architecture Optimization

In order to optimize the design and the architecture 3 design patterns and one architecture pattern was used.

Singleton design pattern

Singleton pattern was used to model the RedisCilent, HiberateReaderManger and HibernateWriterManager classes. As the functionality of those three classes is to interact with a data store only one instance will be required throughout the application runtime.

MVC design pattern

The REST services were modelled using the MVC pattern to clean separation between the business logic involved in those REST service from the presentations to make those layers independent to support any future enhancements that can happen in those layers.

Pipes and fitters design pattern

Pipes and fitters design pattern was adopted when implementing the recommendation algorithms, which would provide flexibility in modifying the existing recommender logic or to fit in another recommender algorithm to the existing system without modifying the rest of the other classes**.**

Pubsub Architectural pattern

Since the flow of user interactions in to the system channel is asynchronous notifying the listeners about any user interactions are handled by the pubsub architectural pattern. Additionally the pubsub pattern allows the freedom of supporting new user interaction formats is without modifying the logic of the pattern reading classes.

5.7 Packages and Package Dependency Diagram

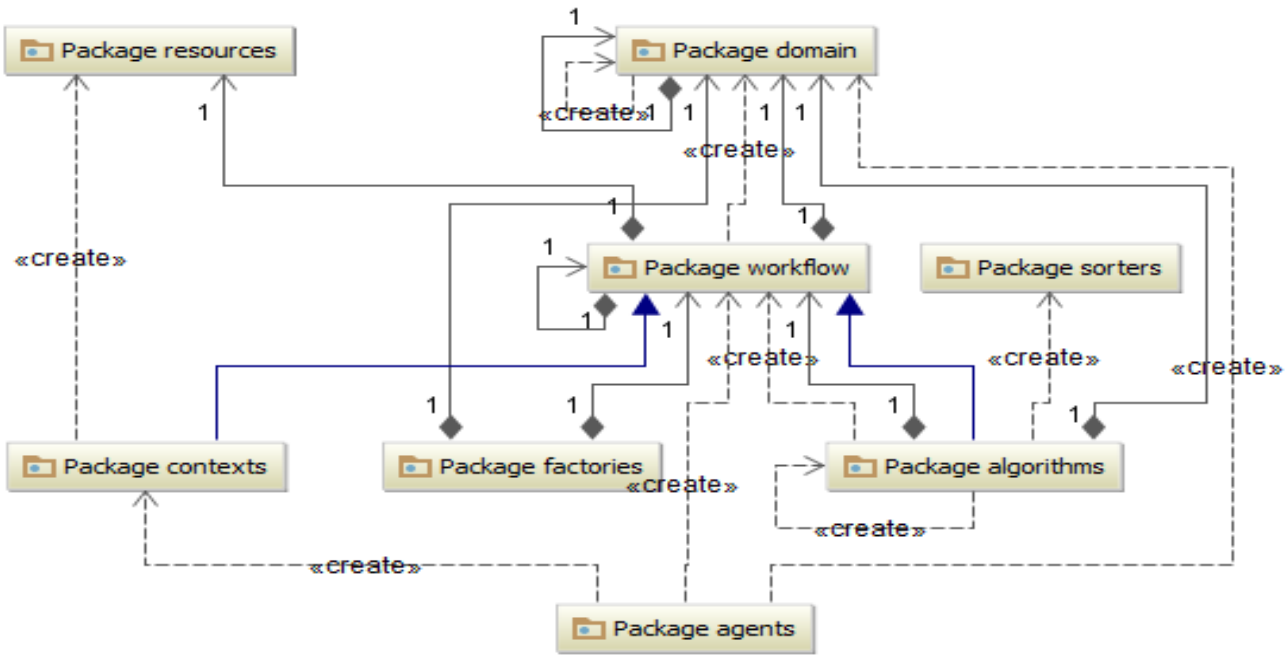


Figure 5. 11-Packages and Package Dependency Diagram

5.8 ER Diagram

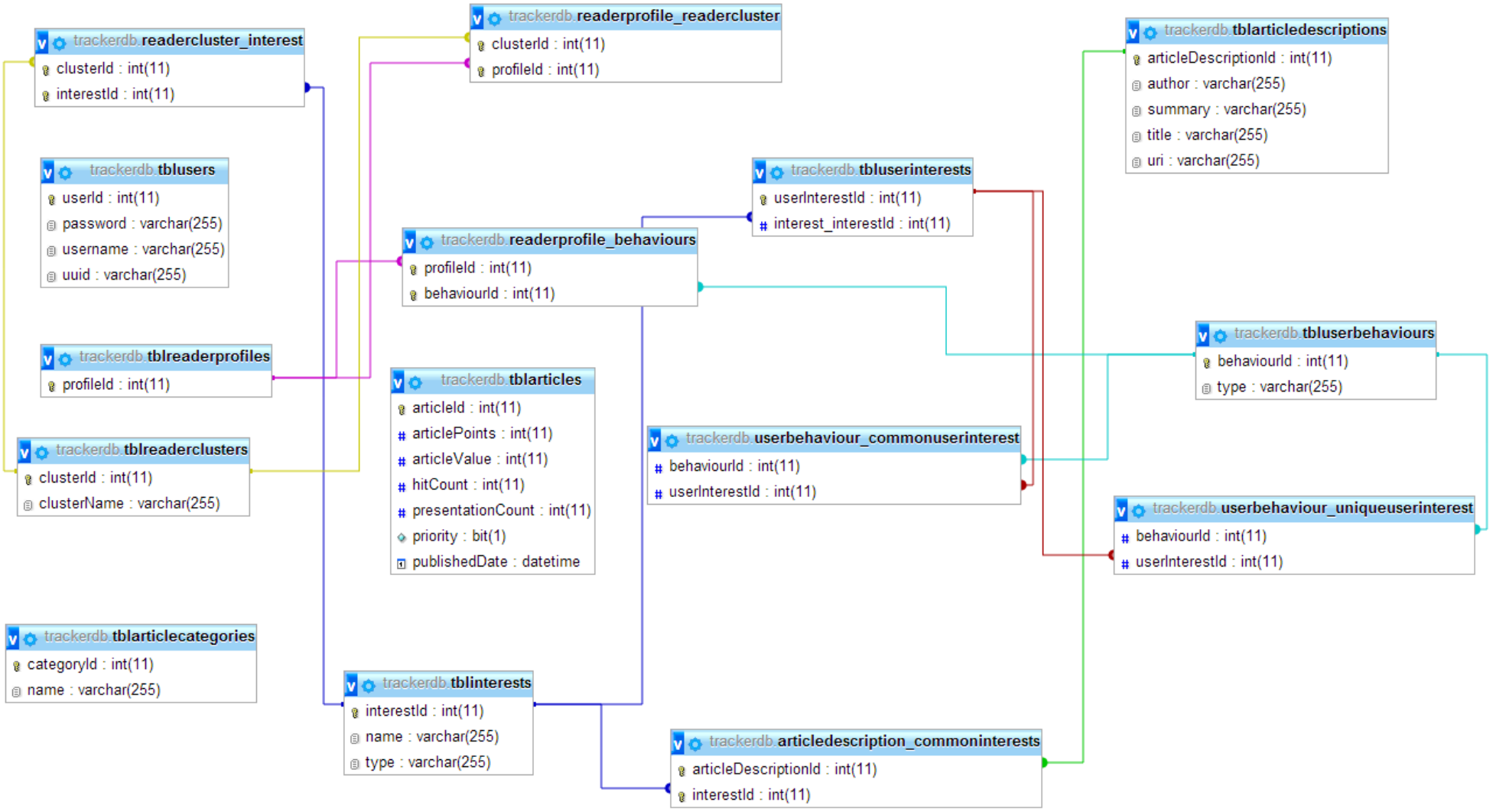


Figure 5.12-ER Diagram

5.9 Design Goals for Overall Solution

Following are the main design goals that were adhered during the design phase of the overall system.

Accuracy

Accuracy of all the modules is vital to achieve successful recommendations given and to achieve the highest possible accuracy the article rate value update module and reader profile update modules should provide the highest possible accuracy rate for recommendation module to carry out its recommendation process successfully. During the design phase careful consideration was given to achieve the highest accuracy rate for the NewsRec system

Scalability

Since scalability is one key success factor for a recommendation engine major emphasise was given to achieve the highest possible scalability for the NewsRec system

Adaptability

The users should be able to adapt to the functionalities of the system within limited time period. Constant help should be provided to the users with steps to complete tasks which are accessible by the users.

5.10 Chapter Summary

This chapter discussed the architecture and the design of the NewsRec system. It started off with the High-level system architecture which was modelled and discussed in terms of modules. The user interaction capturing and saving, user profile updating, article rate value updating and recommender engine modules were identified as the main modules of the NewsRec system. Then the chapter went on to discuss the designing of reader profile and article descriptor models with a design approach for the proposed hybrid recommender engine. The low-level design diagrams such as domain model, sequence diagrams are discussed followed with an explanation of the design and architectural patterns that were adopted. Then the chapter moved in to highlighting the package and package dependency and ER diagrams of the NewsRec system. Finally, accuracy, scalability and adaptability were identified as the overall design goals for the NewsRec solution. The next chapter will focus on the prototype implementation of the NewsRec project

.

Chapter 6: Implementation

Contents

- Chapter Overview
- Technology Selections
- Implementation of REST Service
- Core System Implementation
- Chapter Summary

6.1 Chapter Overview

Having discussed the design of the proposed system in the previous chapter, in this chapter it will be focused on implementation process of the NewsRec system with different frameworks, environments and APIs. It starts off with a discussion on the technology selection for the implementation of the prototype followed by detail account of the implementation process, problems encountered and solutions agreed upon will also be discussed in using appropriate code snippets and screenshots.

6.2 Technology Selections

Selection of software agent development framework

Since the concept of the prototype depends on software agents the key success factor of the implementation would be the selection of the correct software agent framework. Table 6.1 provides a comparison of few available software agent development frameworks.

Agent Development Toolkits → / Features ↓	Aglet	Voyager	JADE	Anchor	Zeus
Nature of Produce	Free, Open source	Commercial	Free, Open Source	Available in BSD license	Free, open source
Standard implemented	MASIF	----	FIPA Compliant	SSL, X.509	FIPA compliant
Communication Technique	Synchronous , Asynchronous	All methods	Asynchronous	Asynchronous	Asynchronous
Security Mechanism	Poor	Weak	Good	Strong security	Good
Agent Mobility	Weak	Weak	Not-so-weak	Weak	Do not support
Agent Migration Mechanism	Socket	RMI	RMI	Socket	null

Table 6.1-Comparisons of Agent Frameworks Singh, Juneja and Sharma (2011).

With the above mentioned information it is evident that JADE framework is more efficient in developing multi agent software due to its support for security and agent mobility factors

Selection of programming language for development of the core components

Since the JADE framework is developed using the Java language and due to its' facilitation in developing multi agent systems using Java language it was decided to use the Java language for the development of the key components of the NewsRec system.

Approach to capture reader interactions from various devices

Since presenting a common layer to capture user interactions from different devices and platforms and exposing a common layer to publish news to different devices and platforms is a part of the project aim it was decided to

expose a web service to capture the user interaction due to its support for platform interoperability. Each client who would consume the hosted web service is expected to write their own web client.

Web service exposure approach

There are two web service exposing approaches known as the SOAP and REST. When comparing SOAP and REST, it can be decided that the most suitable approach to expose the web service is to use the REST approach due to the below mentioned reasons.

- REST services can be consumed by any type of a client
- REST services are light weight and consumes less bandwidth
- Advanced security mechanisms adopted
- Ease of learning and ease of expanding the service with another B2B service

Selection of programming language for the web service

It was decided to write the web service and the writing of captured user interaction storage module using NodeJS server side Java Script framework due to its speed and ease of use. Though NodeJS supports functional oriented approach since the business logic of the service is to just perform data storage writing there would not be any tight coupling of the system.

Selection of temporary data storage

Since the main objective of the temporary data storage is to perform as a data buffer for the core system to work it was decide to use Redis server, which is an open source key value repository as the temporary data storage due to ease of use and better compliance with Java language and NodeJS framework.

Selection of persistence data storage

Since using persistence approach in storing data would be beneficial for the scalability and migration factors of database accessing it was decide to use a persistence layer for the data base access. The persistence layer was modelled using Java Persistence API and developed using the Java Hibernate API as the core system would be developed using the Java language. The physical database was developed using the MySQL DBMS.

Selection of a JSON object serializer and de-serializer

Since the REST web services communicate using serialized JSON objects the core engine should have an inbuilt module for JSON object serialization and de serialization. For this it was decide to use the GSON an open source library.

Selection of an IDE and a deployment environment

It was decide to use IntelliJ as the IDE and Apache Maven as the deployment environemt. IntelliJ was selected due to previous experience and Apache Maven was selected due to its capability of resolving the project dependencies intelligently.

6.3 Implementation of REST service

6.3.1 Implementation of user interaction capturing module

The user interaction capturing module was decided to be implemented as a web service which would be developed using the NodeJS framework. Figure 6.1 describes how the web service was created.

```
/**
 * Creates the server for the pinpoint web service
 * @param {int} port: Port for the server to run on
 */
  exports.createServer = function (port) {
    var server = http.createServer(function (request, response) {
        var data = '';

        winston.info('Incoming Request', { url: request.url });

        request.on('data', function (chunk) {
            data += chunk;
        });

        response.writeHead(501, { 'Content-Type': 'application/json' });
        response.end(JSON.stringify({ message: 'not implemented' }));
    });

    if (port) {
        server.listen(port);
    }

    return server;
};
```

Figure 6.1-User Interaction Capturing REST service

As highlighted in the above mentioned figure first the NodeJS framework specifies a function which would read any http request that are coming in and then parser the http content to identify the JSON object information. It also specifies a port for the REST service consuming clients to communicate with the service. The de-serialization of the JSON objects are handled by the NodeJS framework itself.

Complexity of the code: Hosting a REST service through NodeJS is not a very challenging task as all the key required functionalities are inbuilt in to the framework and the implementation effort would only be in reading the http request and extracting of the JSON object.

Problems faced: None

6.3.2 Implementation of user interaction writing module

The user interaction writing module which works as the business logic of the exposed REST service was decided to be implemented using the NodeJS framework and the Redis server was decided as the data storage. Figure 6.2 describes how the extracted JSON object information are saved in to the Redis server.

```
const redis = require('redis');
const client = redis.createClient();

const io = require('socket.io');

if (!module.parent) {
    app.listen(PORT, HOST);
    console.log("Express server listening on port %d", app.address().port)

    const socket  = io.listen(app);

    socket.on('connection', function(client) {
        const subscribe = redis.createClient();
        subscribe.subscribe('pubsub'); //    listen to messages from channel pubsub

        subscribe.on("message", function(channel, message) {
            client.send(message);
        });

        client.on('message', function(msg) {
        });

        client.on('disconnect', function() {
            subscribe.quit();
        });
    });
```

Figure6.2-User Interaction Saving

To save the data first a connection to the Redis server is established. Afterwards the NodeJS framework register with a data transferring channel to send the data to the Redis server and through the channel the data is sent to the Redis server.

Complexity of the code: Since NodeJS doesn't support concurrency in its environment writing data to a Redis server through a NodeJS app wrapped with a certain amount of complexity. The concurrency would have to be handled by the developer through a mechanism known as channels.

Problems faced: None

6.4 Core System Implementation

The core system is totally developed using the Java language and major emphasis is given to the agents of the system. There are 9 agents running within the system and each agent is allocated with a specific task. Each agent would collaborate with rest of the business logic classes to carry out the task assigned. **Table** provides an overview of the agents and their specific roles.

Agent	Role
ArticleClickAgent	Role of this agent is to collaborate with the business logic classes when a reader has selected an article from a list
ArticleLikeAgent	Role of this agent is to collaborate with the business logic classes when a reader has clicked the like button for an article
ArticleRateValueUpdateAgent	The responsibility of this agent is to communicate with the related business logic classes to update the rate value on an article
ArticleRecommenderAgent	This agent is responsible for collaborating with the business logic classes to provide a list of recommended articles for the reader.
ReaderProfileUpdateAgent	With the details of the users' interactions this agent is responsible in updating the reader profile interests.
UserClusterAgent	The role of this agent is to cluster the users appropriately to the correct user cluster
UserSpawnerAgent	When a reader logs in to the system making the environment ready for is the task of this agent

Table 6.2-Agents used in the system

6.4.1 Recommender Engine Implementation

Recommender engine was developed with the help of 4 classes. The content base and collaborative base filtering were abstracted to two different classes and one class works as the integrator between the two algorithms and the other works the class which would build the suggestions in to one unique list.

```
public static RecommendationAlgorithm build(WorkContext algorithmContext){
    RecommendationAlgorithm algorithm= new RecommendationAlgorithm(algorithmContext);
    algorithm.addAlgorithmSteps(new CommonInterestHandler());
    algorithm.addAlgorithmSteps(new UniqueInterestHandler());
    algorithm.addAlgorithmSteps(new SuggestionsBuilderHandler());
    return algorithm;
}
```

Figure 6.3-Calling of recommender algorithms

As highlighted in figure 6.4 first the collaborative filtering triggered through the CommonInterstHandler and content filtering is triggered through the UniqueInterestHandler classes. Then the SuggestionBuilderHanadler would put the two list together by eliminating repeats and sorting the list according to the publish time and article rate values. Figures 6.4 and 6.5 would be highlighting the implementation of the recommender algorithms.

```
public boolean process(WorkContext context){

    UserBehaviour recentBehaviour=getReaderProfile().getRecentBehaviour();

    if(recentBehaviour.hasCommonInterests()){
        Collection<UserInterest> interests = recentBehaviour.getCommonInterests();
        Collection<Article> articles = context.getArticleManager().getArticlesByCommonInterests
                                        (UserBehaviour.getInterests(interests));
        context.getArticleManager().sort(articles, new ArticleHitsNRateSorter());
        context.addArticlesToBuffer(articles);
    }
    return WorkTask.SUCCESS;
}
```

Figure 6.4-Collaborative Filtering

```
public boolean process(WorkContext context) {

    UserBehaviour establishedBehaviour = getReaderProfile().getEstablishedBehaviour();

    if (!establishedBehaviour.isEmpty()) {

        if (establishedBehaviour.hasUniqueInterests()) {
            Collection<UserInterest> interests = establishedBehaviour.getUniqueInterests();
            Collection<Article> articles = context.getArticleManager().getArticlesByUniqueInterests
                                        (UserBehaviour.getInterests(interests));
            context.getArticleManager().sort(articles, new ArticlePublishedNPrioritySorter());

            context.addArticlesToBuffer(articles);  //Track based on which behaviour we got it from

            return WorkTask.SUCCESS;
        }

    }

    UserBehaviour recentBehaviour = getReaderProfile().getRecentBehaviour();

    if (recentBehaviour.hasUniqueInterests()) {
        Collection<UserInterest> interests = recentBehaviour.getUniqueInterests();
        Collection<Article> articles = context.getArticleManager().getArticlesByUniqueInterests
                                        (UserBehaviour.getInterests(interests));
        context.getArticleManager().sort(articles, new ArticlePublishedNPrioritySorter());

        context.addArticlesToBuffer(articles);
    }
    return WorkTask.SUCCESS;
}
```

Figure 6.5-Content Based Filtering

Complexity of the code: The recommender engine was the highest complex part of the implementation where several agent, business and persistence classes had to coordinated together. The algorithm for the whole recommender engine was developed from the root level without the help of any other library or framework other than the JADE framework.

Problems faced: Since this process involved interaction between several agents together several problems were faced. Major problem faced was managing the concurrency of the communication between the agents and making the agents ready before hand to take up the task.

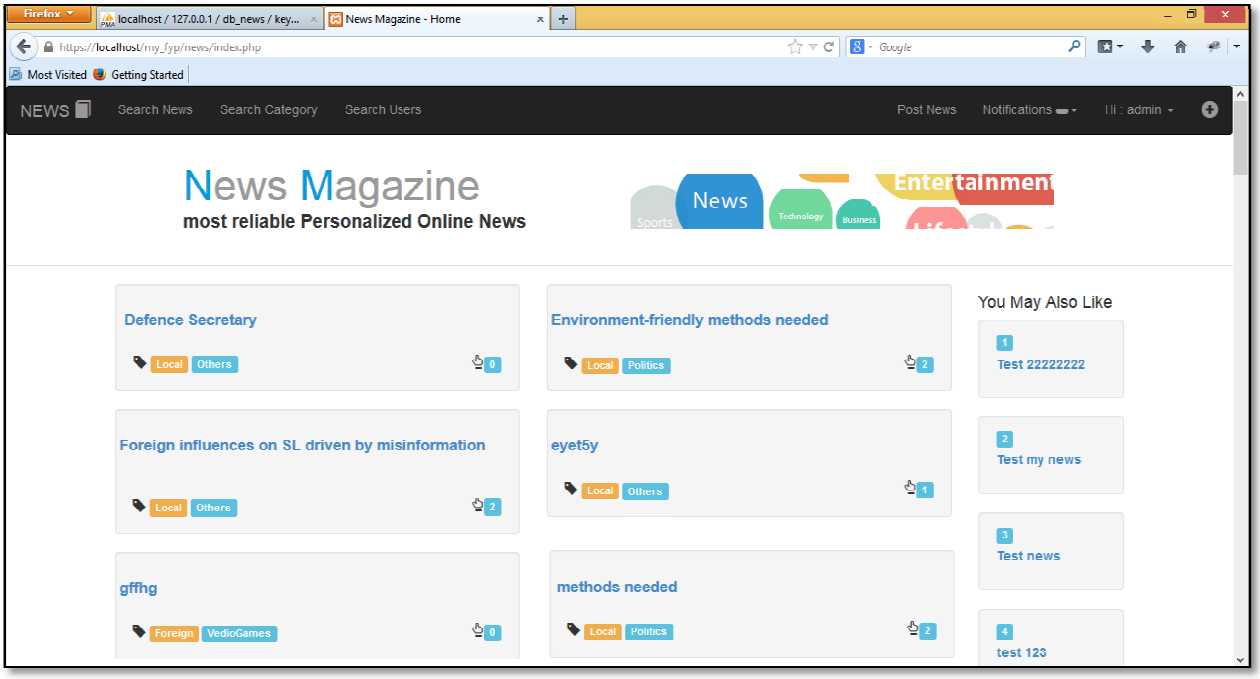


Figure 6.6-Recommended News Articles

As shown in the above figure in the middle of the webpage it will be display the personalized news articles. Addition to that more recommendations will be displayed under the 'You May Also Like' section

6.5 Chapter Summary

This chapter discussed about the implementation process of the NewsRec system. Chapter started with justifying the use of JADE framework as the agent development framework and Java as the core engine development language. Then the chapter went on to justifying the use of NodeJS framework for the development of the REST service and its' business logic and Redis server as the temporary storage for the NewsRec System. After wards the chapter provided an overview of the software agents of the NewsRec system and their roles and negotiations. Then the chapter went on to describe the various steps that were implemented with explanations through code snippets for better understanding and with each step complexity and the problems encountered were discussed.

Next chapter would be the testing details of the NewsRec system

Chapter 7: Testing

Contents

- Chapter Overview
- Objectives & Goals of Testing
- Testing criteria
- Functional requirement testing
- Unit & Integration testing
- Non-Functional requirement testing
- Chapter Summary

7.1 Chapter Overview

Having discussed about the implementation of the proposed NewsRec System in the last chapter, this chapter will be on testing of the functional and non-functional requirements of the implemented system with the intension of making certain that all the implemented requirements were completed to the expected levels. This chapter first outlines the testing criteria, testing methods and testing levels and finally will provide an evaluation of the testing results.

7.2 Objectives and Goals of Testing

Software testing is performed to verify that the completed software package functions according to the expectations defined by the requirements.

The main objectives of the testing process for NewsRec are:

- To verify and validate the functional requirements of the NewsRec system.
- To verify and validate the non functional requirements of the NewsRec system
- To identify the errors and defects of the system in order to make sure that the final product contains fewer amounts of errors and bugs.
- To further enhance the system based on the test results.

7.3 Testing Criteria

Testing the implemented system is the process of an examining an application to ensure that it satisfies the functional and non-functional requirements and meets quality expectations. Software quality can be measured in two ways as described below

1. **Software functional quality**

Mainly focus on the combination of the product development characteristics with the technical requirements of the given design based on the functional requirements.

2. **Software structural quality**

This uses to measure the performance of the functional requirements of the product with the identified non-functional requirements.

7.4 Functional Requirements Testing

Adopted testing method for functional requirement

Spiral methodology, which is the adopted software development methodology, allows the flexibility and freedom of carrying out the testing of the software parallel with the implementation phase. Hence the testing of the implemented functional requirements was carried out parallel to the implementation of the NewsRec system with the black box testing approach.

Table 7.1 shows a summary of the functional requirement testing results. Please refer to **appendix** for test cases and detailed description.

Functional Requirement	**Pass Rate**	**Status**
Any new reader should be able to register with the system by providing demographic data of the reader along with a unique password and a preferred password	**100%**	**Pass**
At the time of the registering to the system any new reader should be able to specify his/her preferred news category	**100%**	**Pass**
Any registered reader should be able to login to the system using his/her username and password and the system should authenticate the user.	**100%**	**Pass**
If the authentication fails the system should display an error message to the reader citing the reason for the authentication failure	**100%**	**Pass**
After a successful login for each registered reader the system should display a list of preferred news article titles sorted according to the rating values of the articles along with article	**100%**	**Pass**
A registered user should be able to search for news articles by entering a string of characters. The entered string will be matched against the article title or article descriptor values	**100%**	**Pass**
For a successful search the system should display a list of articles titles sorted according to the rating values along with their article descriptors.	**100%**	**Pass**
If the searching fails due to non-existence of the entered string an error message must be displayed.	**100%**	**Pass**
A registered user should be able to click-on an article descriptor value that is listed on the recommended or searched news article title list and the system should display a list of article	**100%**	**Pass**
A registered user should be able to click-on and view the content of an article that is listed on the recommended, searched, article descriptor filtered news articles lists.	**100%**	**Pass**
A registered user should be able to rate a viewed news article	**100%**	**Pass**
The system should update rate value of articles	**100%**	**Pass**
News page designer should be able to publish a new news article	**100%**	**Pass**
News page designer should be able to assign article descriptors to a newly published news article and descriptor values	**100%**	**Pass**
Administrator should be able to add new news categories to the system along with suitable article descriptors.	**100%**	**Pass**

Table 7.1 - Tested functional requirements

7.5 Module and Integration Testing

NewsRec system had many software modules in its architecture (refer to section 5.2.2 of the Design chapter) where one module was coupled with another in the form of the output of one unit is consumed by the coupled module. Since the functionality of one unit is essential for the other module to perform it was decided to carry out a module and integration testing of the NewsRec system with the black box testing approach. Each module was tested for 10 times and table 7.2 provides a summary of unit and integration testing results of the NewsRec system

Module	Input	Expected Output	Actual Output	Status
User interaction capturing and saving module	News link selection and liking of articles	The interaction data should be captured successfully	The interaction data was captured successfully	Passed
Captured user interaction saving module	Captured user interactions	The interaction data should be stored in the Redis server	The interaction data was stored in the Redis server	Passed
Interaction reading module	User interaction saved on the Redis server of the system	Interaction data should be successfully read from the Redis server	Interaction data was successfully read from the Redis server	Passed
Interaction identification module	Read user interactions from the Redis server	Interaction type should be correctly identified	Interaction type was successfully identified	Passed
Reader profile managing module	User profile updating data from the interaction identification module	User profile should be updated accordingly and the new preference data should be saved on the MySQL database	User profile was updated accordingly and the new preference data was saved on the MySQL database	Passed
Article rating module	Article rate updating data from the interaction identification module	Article rate should be updated accordingly and the new rating data should be saved on the MySQL database	Article rate should be updated accordingly and the new rating data was successfully saved on the MySQL database	Passed
CF engine	User profile data and saved article data	A list of recommended articles should be produced	A list of recommended articles was produced	Passed
CBF engine	Article rate values, user profile data and saved article data	A list of recommended articles should be produced	A list of recommended articles was produced successfully	Passed
News publishing module	Recommended article list produced by CF and CBF engines	The module should publish the recommended article list both to a standard view and mobile app view	The module successfully published the recommended article list to a standard view and a mobile app view	Passed

Table 7.2- Summary of unit and integration testing

7.6 Non Functional Requirements Testing

7.6.1 Accuracy Testing

A major non functional requirement of the NewsRec system was the accuracy of the recommendations done for the news readers and at the time of the non functional requirement identification it was mentioned that for a newly registered reader the system should provide an accuracy rate above 40% and for an already registered user who has substantially interacted with the system an accuracy rate above 70% should be maintained.

Accuracy of the recommendations was determined by using the following equation as recommended by Lindgaard and Chattratichart, (2007).

$$\text{Accuracy percentage of the output} = \frac{\text{Number of passed recommended data}}{\text{Number of all recommended data}} * 100$$

For the testing purpose, ten different users were selected and were allowed to create user accounts and access the NewsRec System for six days where they were asked to perform 15 interactions with the system per day. Then at the end of each day an accuracy rate of each user was calculated based on the above mentioned equation. Figure 7.1 shows the statistic of the accuracy testing performed on the NewsRec system.

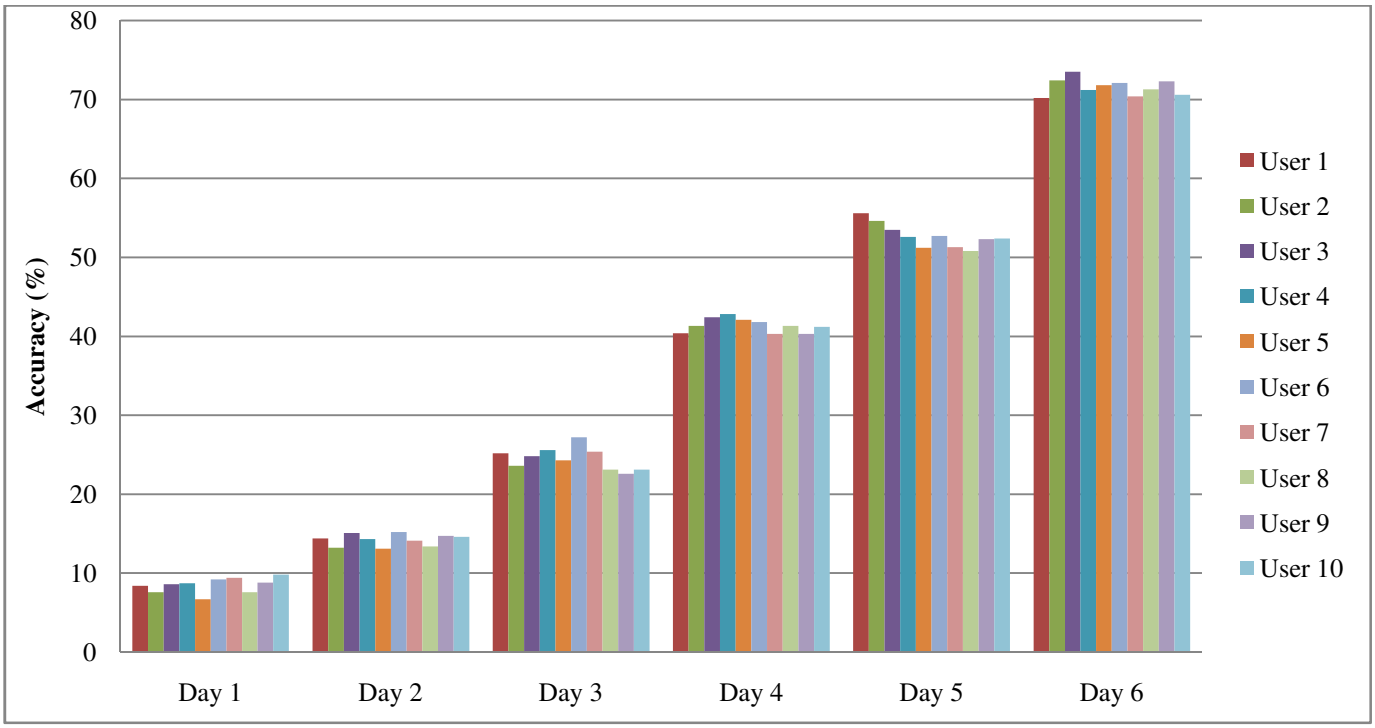


Figure 7.1- Accuracy rate of the recommendations

With the statistics of the above figure it can be seen that though the initial accuracy rate was very low as the readers interact supplementary with the system the accuracy of the recommendations gradually increases and have achieved the expected accuracy rate of above 70% but the to achieve an accuracy of at least a 40%, it would take around 4 days or minimum of 60 interactions with the system. Same test results were analyzed to see the accuracy of the recommendations done for common and unique interests of a reader. At the end of each day the average for common and unique interest recommendations of all the ten readers were calculated

separately. Figure 7.2 shows the statistic of the recommendations of unique and common interests recommended by the NewsRec system.

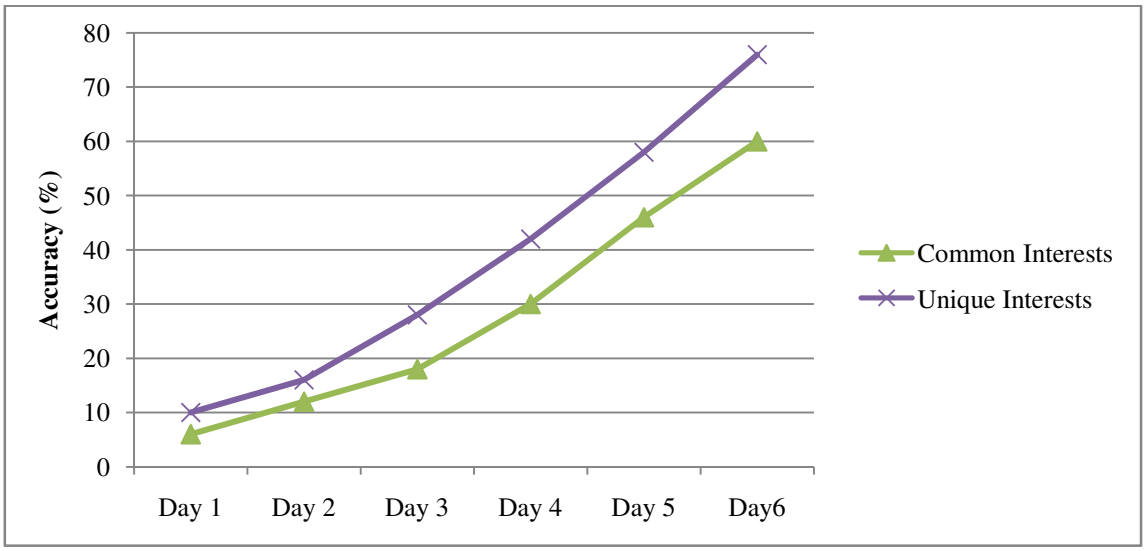


Figure 7.2- Accuracy rate of unique and common interest recommendations

With the above statistics it is evident that the system flavours the recommendation for unique interest more compared to common interests. The reader preference model was modelled for unique interest to be used for content based filtering and common interests to be used for collaborative filtering. The lack of accuracy for common interests may be because at the time of the testing the system doesn't have enough readers to cluster them more logically.

Another testing was carried out to compare the hybrid recommender approach against the content based, collaborative and no recommender approaches. Another ten different users were selected and were allowed to access the NewsRec System with the hybrid recommender, content based, collaborative and no recommender approaches for six days where they were asked to perform 15 interactions with the system per day. The averages of each approach were calculated on every day and was analysed. . Figure 7.2 shows the statistic of the accuracy rate of recommender approaches

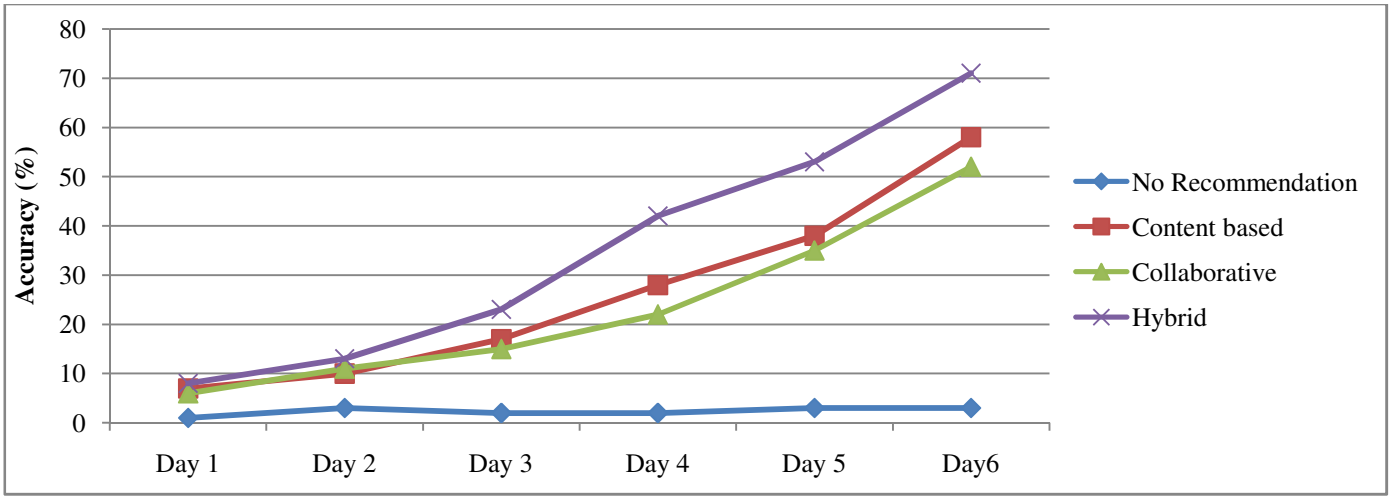


Figure 7.2-Accuracy rate of the recommender approaches

With the above mentioned statistics in can be seen that the hybrid recommender approach provides a higher accuracy rate compared to the content based or collaborative approach. It is astonishing to see the content based filtering approach provides a better accuracy compared to the collaborative approach. It may be due to the fact that at the time of the testing the system doesn't have enough readers to cluster them more logically.

7.6.2 Performance Testing

A performance testing of the NewsRec system was carried out to test the response time of the system and Table 7.3 shows the statistics of the performance testing process carried out using a standard web browser and each test case was executed for twenty times and the average time was taken for the analysis.

Test case	Purpose	Input data	Expected result	Actual result (s)	Comment
1	Loading the home page for the 1st time	NA	three seconds or less	2.42	Pass
2	Redirect home page	NA	three seconds or less	0.19	Pass
3	Logout from the website	NA	three seconds or less	0.92	Pass
4	login to the website	Username and the password	three seconds or less	1.10	Pass
5	Signup with the website	Required user details	three seconds or less	2.46	Pass

Table 7.3 - Performance testing of NewsRec system using a Web Browser

Rusu, *et al* (2011) hasstated that the general load time of a webpage should be between one to seven seconds and main page should be loaded around 2.45 seconds. As shown in the above table time taken to load the main page of the NewsRec system is less than 2.45 seconds and other interactions were performed below the expected time limit and it can be concluded that the system has a good response speed for user requests.

Table 7.4 shows the statistics of the performance testing process carried out using an Android app and each test case was executed for ten times and the average time was taken for the analysis

Test case	Purpose	Input data	Expected result	Actual result (s)	Comment
1	Loading the home page for the 1st time	NA	three seconds or less	2.88	Pass
2	Redirect home page	NA	three seconds or less	0.34	Pass
3	Logout from the website	NA	three seconds or less	1.12	Pass
4	login to the website	Username and the password	three seconds or less	1.32	Pass
5	Signup with the website	Required user details	three seconds or less	2.97	Pass

Table 7.0.4- Performance testing of NewsRec system using an Android app

Though the mobile app takes a longer time to respond compared to the normal web browser it can be seen that the time taken to load the main page of the NewsRec system is less than 2.45 seconds and other interactions were performed below the expected time limit and it can be concluded that the system has a good response speed for user requests.

Though the performance testing results were positive, it's has to be noted that the testing was not carried out in an actual client- server network where other network devices such as routers would have increased the latency of the requests and response.

7.6.3. Load and Scalability Testing

Load and scalability testing is used to test the level and efficiency a system would be handling the load and the resources allocated to the system. There are many tools available to carry out this type of testing. ApacheBenchmark is one of widely used load and scalability testing tool due to its ease of use and presentation of results. Therefore, Apache Benchmark tool was used in order to test the load and scalability of the NewsRec system where the system was running on a machine with an i3 processor of 2.2 GHz and 2GB of RAM and Windows 7 32bit version as the operating system.

Stated below is the statistics derived from by the Apache Benchmark tool of the load testing process carried out.

1. Total data transferred is 571000 bytes for 1000 requests. Which is close to 517 per page and it is in line with the value of home page size.
2. Test completed in 12.270s.
3. Requests per seconds were 81.50.
4. Time per request was 12269.823 ms (for 1000 concurrent requests). So across all requests it is 12269.823 ms/1000 = 12.270ms
5. Transfer rate is received as: 45.45 (Kbytes/sec).
6. In connection time stats, there were many requests had to wait for few seconds. This may be due to apache putting requests in wait the queue.

When analysing the results of the load testing, it can be concluded that the NewsRec System is capable in handling at least 1000 users at a time, which can be considered as an acceptable rate for load and scalability testing. In addition to this, load testing was done by connecting and disconnecting the local host connection and noticed that the system operates in a normal way.

7.7 Limitations of the Testing Process

- **Testing was not carried out on a real client server environment**

Testing process for the performance testing was carried out by using a web server running on the same network as the client machines. As a result the current performance values may not represent the true performance

values as real world bottle necks that could have made a negative impact on the system performance were not faced during the testing process.

- **Not enough time for the system to mature**

A major requirement of collaborative filtering is that the system should be mature with enough users and items for it to produce accurate results. But since the testing process was conducted with only 10 readers on 6 days the results generated for the collaborative filtering may not represent the true picture of collaborative filtering approach.

7.8 Chapter Summary

This Chapter focused on the testing aspect of the prototype and the chapter started with outlining the purpose of the testing phase, testing criteria, testing methods and testing levels. There were two main types of testing methods Software functional quality testing and software structural quality testing. Under software functional quality testing functional requirement testing and unit and integrating testing were carried out using the black box testing approach. The testing results emphasized that the functional and unit and integration of the system was completed to the expected level. Then the testing moved on to software structural quality testing which mainly focuses on testing the identified non-functional requirements. Testing of the performance, accuracy, load and scalability of the prototype were carried out under software structural quality testing phase. Overall software structural quality testing phase was at a satisfactory level with the testing results managing to achieve the determined testing threshold limits except for the threshold limit of above 40% of accuracy for a newly registered user. It was concluded that it will take around 4 days or minimum of 60 interactions with the system to achieve the expected accuracy for a newly registered user. Next chapter which is the evaluation chapter will describe the evaluation process carried out on various evaluation criteria of the NewsRec project.

Chapter 8: Evaluation

Contents

8.1 Chapter Overview

Previous chapter discussed about the testing process carried out on the NewsRec system with the intension of making certain that all the implemented functional and non-functional requirements were achieved to the expected levels. This chapter focuses on the evaluation process carried with various evaluators on different evaluation criteria and the self-evaluation in order to find out strengths and weakness of the NewsRec project.

8.2 Evaluation Criteria

The following criteria were identified for the evaluation process and these criteria were selected with the aim of covering the major phases of the whole project.

Criteria	Description& purpose
Overall concept and whole project	It is believed that the NewsRec project should obtain comments, views and evaluation with constructive criticism on the perception of the concept
Scope and depth of the project	Since both online news domain and recommender systems are broad, it is important to get views and comments about the scope of the project from domain experts
System design, architecture and implementation	Evaluate whether the design, architecture and implementation of each module is completed properly providing proper justifications.
Solution and Prototype	Assessment should be done on the prototype to determine whether the prototype acts as a proof of concept of the NewsRec project.
Usability, performance and accuracy of the prototype	Evaluate the non-functional requirements of the NewsRec to determine the extent the non-functional requirements were implemented.
Recommender Engine	Effectiveness and suitability of the proposed recommender engine for the hypothesis have to be evaluated
Limitations of the solution and future enhancements	Identification of limitations of the NewsRec system and potential future enhancements that needs to addressed.

Table 8.1-Evaluation criteria

8.3 Selection of Evaluators

Below mentioned evaluator categories were identified for the evaluation process of the project and high priority will be given to the domain experts of recommender systems as a high emphasis of this project was given to the recommendation engines and software agents and end users as they would be the group ultimately using the developed system.

Evaluator Category	Description & criteria evaluated
End users	Group of online news readers and the system operators were selected for the evaluation of the overall concept, usability, performance and accuracy, prototype and future enhancements criteria of the NewsRec system.

Online news industry experts	Group of experts who are well experienced and currently working in news industry were selected to evaluate the overall concept, prototype and future enhancement criteria
Software engineers & architects	Group of well experienced and currently working software engineers and architects were selected to evaluate the System design, Architecture and implementation and future enhancement criteria
Domain experts of recommender systems	Group of prominent experts on the recommender system domain were selected to evaluate the recommendation engine of the NewsRec system future enhancement criteria.

Table 8.2-Evaluator groups

8.4 Evaluation Methodology and Approach

Evaluation process of a project exposes the success of the project and provides a feedback with regards to implementation and major phases of the project life cycle such as problem, analysis, design and implementation. The project was evaluated using a combination of qualitative and quantitative methods. The quantitative evaluation is more or less achieved through the testing phase of the project. Hence more emphasis was given to the qualitative evaluation of the project this section. Evaluation of the NewsRec project was carried out through questionnaire and interview approaches, but major emphasis was given to questionnaire approach due to time constraints faced at the end of the project. One questionnaire was prepared on the qualitative measures of the recommender algorithm and usage and applicability of software agents and distributed to experts in the field of recommender systems and software agents with the source code, design and the implementation chapters. Another questionnaire was prepared on the qualitative measures of the design, architecture and implementation of the NewsRec system and was distributed among the software engineers and architects. An additional questionnaire was prepared on the qualitative and quantitative measures of the non-functional requirements of the NewsRec system and overall concept of the project and was distributed among the end users of the NewsRec system. Interview process was carried out with the domain experts of online news domain to evaluate the qualitative aspect of the overall concept, scope as well as the prototype.

8.5 Summary of Evaluation Survey Questions

The below table 8.3 provides a summary of the questions that were asked from various evaluator groups during the evaluation survey.

No	Question
User Related Questions	
1	How many years of experience do you have in recommender systems and software agents?
2	How many years of experience do you have in the online news domain?
The project concept and the project as a whole	
3	What is your general idea about the NewsRec project?
4	What would be the impact this solution would have on the selected user groups?
Scope and depth of the project	

5	Do you think the scope of the project is acceptable for postgraduate level?
6	What depth the solution should have addressed the recommendation systems?
System design, architecture and implementation	
7	What are your comments about the design and architecture with regards to the project concept?
8	Do you think the decisions made in the implementation phase are acceptable and justifiable?
9	What are your suggestions on the design and the implementation?
The solution and prototype	
10	Do you think the presented solution is having the depth in solving the problem?
11	Do you think the system provides a solution to the identified problem?
12	What are your comments on the features offered?
Usability, performance and accuracy of the prototype	
13	How would you rate the usability, performance and accuracy of the prototype?
Recommender Engine	
14	What is your opinion on the hybrid approach used for the NewsRec system?
15	What are the areas that have to be improved in the presented recommender engine?
Limitations of the solution and future enhancements	
16	What are the general limitations you see in the solution and what are your recommendations for those?
17	What are the features do you think can be added to NewsRec project?

Table 8.3-Evaluation Survey Questions

8.6 Evaluation Survey Findings

The evaluation results are presented as a summary with comments and suggestions of the evaluators.

8.6.1 Overall Concept

"I believe your project concept highlights what the news industry should provide to its readers. Basically, it creates a whole new level and a set of opportunities for the online news industry and to have an edge over the other news sources"

Mr AriyanadaDomagahawatte (B.A)

Editor – Lankadeepa news paper

"This solution would deliver the correct and expected content in a timely manner to the reader through their choice of device."

Mr.S.NadarajahPillai (B.A)

Paper Monitoring Head - Media Ministry

"This system would save a lot of time and effort of a reader, especially mobile readers and it would help readers with unique interests immensely"

Ms.UdaraBenaragama

Medical student

Evaluation module	Summary of feedback
The concept	The feedback from the evaluators about the concept of the project was positive. Especially the general users were pleased to have a system like NewsRec which would help in saving time and effort and they were highly satisfied that the same features were provided to the mobile platform as well. And domain experts were keen on the financial benefits this system would bring in to the online news domain.
Review on feedback	Recommendation is a functionality which lacks in the current online news domain and it is a good concept to work on will be helpful for news companies and for news readers to earn mutual benefits. Most of the evaluators are stated that this is project has a good potential towards the success and most of the evaluators valued the effort

Table 8.4-Summary of Evaluation feedback on project concept

8.6.2 Scope & depth of the project

"The depth of the system in recommendation aspect is quite outstanding and I believe this effort has covered enough scope for a postgraduate level research"

Prof. M.E. Paul De Bra (PhD)

Eindhoven University of Technology, Netherlands

"Recommendation systems are a very broad area and it consists of a substantial depth as well. The depth of the application is quite satisfactory for a postgraduate level research"

Dr.Shantha Jayala (PhD)

University of Kelaniya, Sri lanka

"The results provided by the system is sufficient for the online news domain and its readers and I think this is a good academic achievement"

Mr Ariyanada Domagahawatte (B.A)

Editor – Lankadeepa news paper

Evaluation module	Summary of feedback
The scope	Overall comment was that the project scope is a challenging and the addressed depth of the project is sufficient enough for a post graduate level research. Also the elevators praised the adaptation of mobile devices to the solution.
Review on feedback	Automated recommendation systems are a broad area which has a vast depth and due to those reason it was in doubt to the depth the research should go in to. With the feedback of the domain experts it can be concluded that the depth addressed in this research is sufficient enough for a post graduate level research.

Table 8.5-Summary of Evaluation feedback on project scope and depth

8.6.3 System Design, Architecture and Implementation

"Modularization and use of design pattern like Observer and Singleton have contributed on a positive note towards the system and the hybrid design approach is also well thought of"

Mr. Yohan Seimon (M.Sc.)

Technical Architect - Navanthis

"The adopted three tier architecture would be sufficient enough at the moment, but it would have been better if the architecture was thought for a SaaS (Software as a Service) platform"

Mr.Mohamed Sifan (B.Sc.)

Software Design Engineer - Imagination Technologies, UK

"I believe use of NodeJS and Redis server has contributed immensely towards the scalability of the project as both are light weight components. But use of software agents doesn't make any sense.'

Mr.Nuwan Dias (B.Sc.)

Software Engineer - WSO2

"Since a news site would transfer huge amount of information it is better to use a simple and lightweight method to do it. REST is a good option for that as used in the news site and it supports many formats such as JSON."

Mr.Gavin Theekshana (M.Sc.)

Senior Software Engineer- Millennium Information Technologies

Evaluation module	Summary of feedback
System Design	Most of the reviewers were satisfied with the design methodology and design goals and decisions that were taken during the designing phase. Some criticized the use of SSADM for the designing of the data interaction writing module.
Review on feedback	The decision of using SSADM approach for the designing of the data interaction writing module was taken by considering the technologies involved in that module. Both NodeJS and Redis access layer supports structural programming and the best design methodology for structural programming is the SSADM approach.
Evaluation module	Summary of feedback
System architecture	Though most of the reviewers praised the architecture of the NewsRec system, one software design engineer has raised a point by saying the architecture of the system will not be supporting a cloud level solution.
Review on feedback	The argument made that the current architecture would not support a SaaS platform is a valid point but the current architecture was derived for typical client server architecture and the most suitable architecture approach was chosen for the modelling of the system. If this system is going to be converted to a SaaS then the architecture of the system would have to be thought from the beginning.

Evaluation module	Summary of feedback
Implementation	Though most of the reviewers praised the technologies used for the implementation of the NewsRec system saying that those used technologies are light weight and cutting edge technologies, one software engineer has raised a point by saying the use of Software agents doesn't add any value to the system
Review on feedback	The technologies used for the implementation of the NewsRec system were reviewed for their advantages and limitations and the most suitable technologies were chosen. Software agents are more of a research level concept than a software engineering practice and research publications have proven that many benefits can be achieved by using software agents.

Table 8.6-Summary of System Design, Architecture and Implementation

8.6.4 Solution and Prototype

"The solution seems to be addressing the identified problem properly, but the prototype can be improved by capturing more user interactions in both explicit and implicit approaches"

Dr. Shantha Jayala (PhD)

University of Kelaniya, Sri lanka

"For the current solution to be a more rounded solution more emphases needs to be given to the needs of online news publishers as well"

Mr Ariyanada Domagahawatte

Editor – Lankadeepa news paper

Evaluation module	Summary of feedback
Solution and Prototype	Overall comment for the solution and prototype evaluation is that the solution presented and the prototype addresses the problem and tries to give a solution. But the evaluators have raised concerns over the lack of parameters taken to gauge the preference of a reader and limited functionalities provided for the online news publishers
Review on feedback	The current system was developed as a proof of concept that a recommender system can be incorporated with online news environment and due to time restrictions not all types of user interactions were captured and taken for consideration. This suggestion can be highlighted as a future enhancement for the current system. Other criticism that the current system doesn't provide sufficient functionality to the online news publishers was again due to the time restrictions faced during the project execution and this suggestion too can be taken up as a future enhancement.

Table 8.7-Summary of Solution and Prototype

8.6.5 Usability Accuracy and Performance

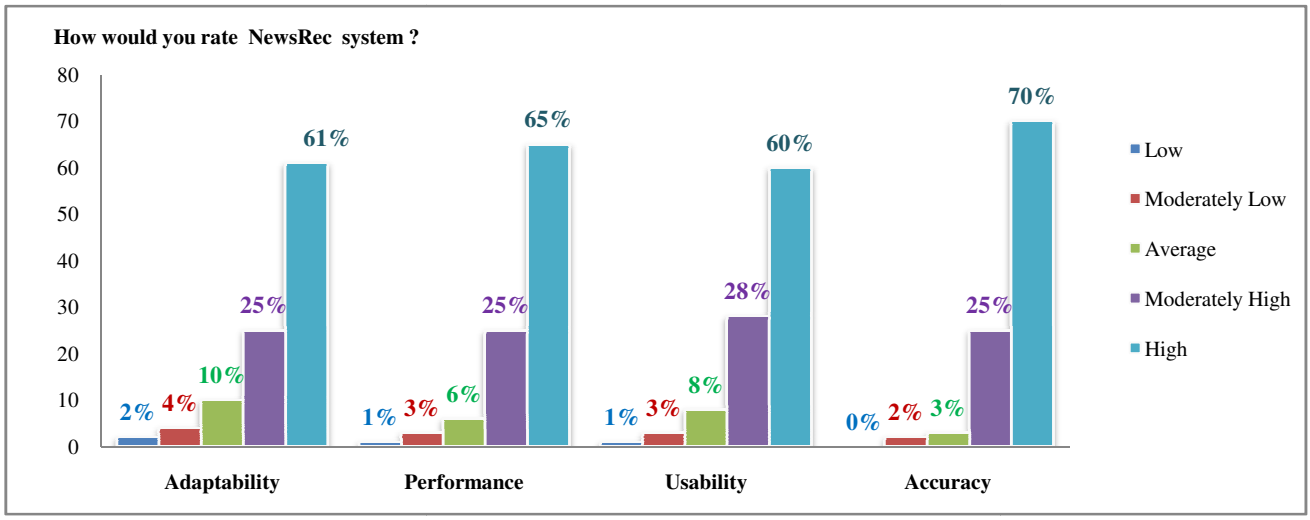


Figure 8.1-Non functional requirement rating

Figure 8.1 shows that the readers were highly satisfied with the non-functional requirements provided by the NewsRec system with all the evaluated non-functional requirements were collectively above 80% in the high and moderately high categories.

The online news publishers were not satisfied with the usability and the performance provided to them by the NewsRec system and this point was raised when evaluating the prototype as well. As highlighted when reviewing the prototype these suggestions can be taken up as future enhancements of the system.

8.6.6 Recommender Engine

"NewsRec is all about the experience in personalization and most of the users will be impressed with the personalization option because it can deliver relevant content based on user behaviour. I believe that the use of the hybrid recommender solution to develop the system is a sensible choice because it can give recommendation even for a new article that anybody has seen yet. But I believe that web data mining should have been a better approach than the hybrid approach."

Prof. M.E. Paul De Bra (PhD)

Eindhoven University of Technology, Netherlands

"There are several recommender systems with unique behaviour moreover; they carry many advantages as well as disadvantages. Before selecting a filtering system to use, it is better to check the suitability of it. I believe that the hybrid solution will be more effective than using the filtering systems separately because it will reduce the number of limitations with the both the filtering systems and it will help to generate more accurate results. But the approach for the collaborative filtering should have been an item based than being a user based, since news site carries more articles."

Dr.J. Ben Schafer(PhD)

Associate Professor - University of Minnesota, United States

Evaluator's comment	Review about the comment
"I believe that web data mining should have been a better approach than the hybrid approach."	Web data mining approach uses server logs to make recommendations and the major limitation of this approach that it won't consider content of the page for the recommendations making the recommendations limited to collaborative recommendation. The current approach is a hybrid approach, which is a better recommendation approach and web data mining can be taken up as a future integration for the current approach.
"But the approach for the collaborative filtering should have been an item based than being a user based, since news site carries more articles."	According to a comparison done by Hahsler (2011) in item-item approach the performance average is 0.003sec while in the user-based approach the performance average is 0.001sec. According to these statics it concluded that it is better to use user based over item based.
"I think the most suitable clustering technique for this scenario would be Hierarchical clustering technique"	The Hierarchical clustering technique has several drawbacks compared to the K-means clustering. As highlighted in chapter 2.4.2 it is evident that hierarchical clustering technique requires more resources and since it's not a iterative technique it will be hard to cluster users who have changed their preferences.

Table 8.8-Review on evaluation of recommender engine feedback

8.6.7 Limitations of the solution and future enhancements

"Especially with the different audience segments a news site would have this concept could benefit the marketing in a massive range by personalizing the advertisements."

Ms.Poornima Fernando (B.Sc)

Business Analysis - IFS

Identified future enhancements are discussed in the conclusion chapter. Please refer to chapter 9.7 for detailed discussion of the future enhancements.

8.7 Self Evaluation

This section will be focusing on self-evaluation of the above identified evaluation criteria along with an evaluation on the overall project.

Self evaluation on identified evaluation criteria

Criteria	Self evaluation
Overall concept	The project hypothesis was a valid and a timely one with a good business prospective and providing a solution to such a problem was an achievement. The concept could have further expanded to recommendation of other news presentations formats like videos and audios to provide a complete solution

Scope and depth of the project	Though recommender systems domain is a broad area with a substantial depth, the area and the depth covered during the NewRec project is at a satisfactory level.
System design, architecture and implementation	The design approach applied along with design decisions and design goals and the system architecture are industry accepted standards and the design of the system was specially focusing of achieving the highest possible scalability. The design of the system can be taken up as a high point of the project. The techniques and technologies used during the implementation of the NewsRec system are critically evaluated cutting edge techniques and technologies and can be satisfied with the implementation of the NewsRec system
Solution and Prototype	The solution put forward to solve the problem has addressed issues related to both online news domain and recommender systems domain and the solution presents a novel approach in modelling reader profiles and articles. The prototype provides a more than satisfactory solution to the problem from the readers' perspective but the prototype could have provided more user friendly functionalities for the online news publishers. Though those features were identified during the requirement elicitation process they were not implemented due to time constraints but it can be taken up that both the solution and the prototype are functioning as a proof of concept to the problem.
Usability, performance and accuracy of the prototype	The non functional requirements were identified ahead of the implementation and the literature review concentrated heavily in finding ways to achieve the highest level of accuracy and the scalability for the proposed solution. The current accuracy rate of the recommendations will be further improved with time and other implicit and explicit can be incorporated to improve the accuracy. Usability and performance of the NewsRec system were achieved to an exceptional level.
Recommender Engine	Testing results have shown that the recommender approach and the engine adopted for the NewsRec system was capable of achieving an accuracy of above 70% with a minimum of 90 interactions performed on the system. Due to those reasons the recommender engine used for the NewsRec system can be classified as an achievement and valued solution for the problem. It was visible during the testing phase that the system was not yet matured to carry out sensitive recommendations using the collaborative filtering but with enough readers and interactions the recommendation accuracy would increase. The recommender engine could have been further improved to support more explicit and implicit user feedbacks that would help the accuracy of the recommendations to further improved.

Table 8.9-Self evaluation of evaluation criteria

Reflection on the Functional Requirements

Functional Requirement	Priority	Status
Any new reader should be able to register with the system by providing demographic data of the reader along with a unique password and a preferred password	**C**	Implemented
At the time of the registering to the system any new reader should be able to specify his/her preferred news category	**C**	Implemented
Any registered reader should be able to login to the system using his/her username and password and the system should authenticate the user.	**C**	Implemented
If the authentication fails the system should display an error message to the reader citing the reason for the authentication failure	**C**	Implemented
After a successful login for each registered reader the system should display a list of preferred news article titles sorted according to the rating values of the articles along with article	**C**	Implemented
A registered user should be able to search for news articles by entering a string of characters. The entered string will be matched against the article title or article descriptors descriptor	**C**	Implemented
For a successful search the system should display a list of articles titles sorted according to the rating values along with their article descriptors.	**C**	Implemented
If the searching fails due to non-existence of the entered string an error message must be displayed.	**C**	Implemented
A registered user should be able to click-on an article descriptor value that is listed on the recommended or searched news article title list and the system should display a list of article	**C**	Partially Implemented
A registered user should be able to click-on and view the content of an article that is listed on the recommended, searched, article descriptor filtered news articles lists.	**C**	Implemented
A registered user should be able to like a viewed news article	**C**	Implemented
The system should update rate value of articles	**C**	Implemented
News page designer should be able to publish a new news article	**C**	Partially Implemented
News page designer should be able to assign article descriptors to a newly published news article and descriptor values	**I**	Partially Implemented
News page designer should be able to modify the assigned article descriptors of a published news article	**D**	Not Implemented
Administrator should be able to add new news categories to the system along with suitable article descriptors.	**D**	Not Implemented

Table 8.9- Reflection on Functional Requirements

Many obstacles had to be overcome during the course of the project and there were many successes and failures as well. To overcome the obstacles extensive level of research material were referred and feedbacks were obtained from experts on various fields and as a result project plan had to be revised several times to accommodate the extra time that was required to overcome the problems. But all in all in the researcher's opinion, the project was successful to a great extent. When comparing the proposed system with the existing system, the newly designed system was researched, designed and developed specifically to achieve the highest possible level of scalability through the design and run time levels of the system while trying to achieve the highest possible accuracy for the recommendations. The testing and domain expert evaluations have justified the novelty of the NewsRec system but there are a set of possible future enhancement that can be incorporated to the NewsRec system (refer to chapter 9.7) that would help to increase the accuracy and the scalability even to a greater level. Throughout the project many academic and technical concepts and soft skills were either learnt or enhanced (refer to chapter 9.3, 9.4 and 9.5) that could be beneficial for future engagements.

8.8 Chapter Summary

This chapter started with describing evaluation criteria, evaluation methodology and selected different types of evaluators to evaluate the different phases of the NewsRec project along with the justifications for those selections. Then Questionnaire and Interviews were selected as the evaluation approaches with justification followed by the evaluation feedbacks discussion and reviewing. According to the received feedback it was highlighted that the concept of the project was a timely one and the idea has a good potential towards the being successful and most of the evaluators valued the effort. When reviewing the feedback for the depth and the scope of the project it was highlighted that the depth and the scope is at an acceptable level for a postgraduate level research. Most of the evaluators praised the design of the NewsRec system along with the design goals and decision and the selection of technologies for the implementation was highly appreciated as well. The solution and the prototype evaluation too yielded positive remarks from the evaluator and the evaluators highlighted that the non functional requirements of the NewsRec system are also at an acceptable level. Selection of a hybrid solution for the recommender engine was praised by domain experts and it was further mentioned that recommender module is capable of recommending readers with accurate personalized content. Several negative comments from the reviewers challenged with valid reasons and some were identified as potential future enhancements. The critical evaluation chapter concluded with a self evaluation covering all the aspects of the NewsRec project. Next chapter will be the conclusion chapter where it will discuss how successfully the project aim and objectives were achieved and moving on to justifying learning out comes, highlighting the challenges faced and future enhancement of the project and as well as wrapping up of the project.

Chapter 9: Conclusion

Content

- Chapter Overview
- Achievement of Aim & Objectives
- Utilizing of knowledge from course modules
- Use of existing skills
- Learning outcomes
- Problems & challenges faced
- Limitations of the research
- Future enhancements
- Contribution
- Concluding remarks

9.1 Chapter Overview

The previous chapter presented the results of the evaluation process carried out on the NewsRec System. This chapter will focuses on concluding the project by highlighting the achievement of the aim and objectives, the problems and challenges faced during project life cycle, limitations of the project, identified future enhancements and closing remarks.

9.2 Achievement of Aim and Objectives

Aim

To design, develop and evaluate a software agent based recommender system to recommend news articles of selected categories to suit individual preferences of readers who would use different types of devices with different platforms to access news articles of an online news environment

The aim was successfully achieved with in the allocated time period and the prototype was qualitatively and quantitatively evaluated through a self-critique process, domain experts and end users.

Objectives

Objective 1	Prepare Terms of Reference
The Terms of Reference defined (TOR) the aim and the objectives set for the research along with an activity schedule which was a used as a guideline for efficient time management. TOR is included as chapter 1 of the report.	
Objective 2	Literature Survey
The literature survey carried out provided a through insight and understanding of various recommender types and architectures, the strengths and weaknesses of those recommender types and the impact those strengths and weaknesses would have on the accuracy and scalability of a recommender system. It paved pay to broader exposure on various approaches of user profile modelling and clustering, user model and item based recommendation, the cold starter problem, interest changes of users and use of software agents on a recommender system. The literature survey is included as chapter 2 of this report.	
Objective 3	Selection of a software development methodology
After reviewing several software development methodologies Spiral methodology was selected due to its flexibility in handling frequent requirement changes. For detailed explanation of this selection refer to chapter 3 of this document	
Objective 4	Requirement gathering process
Requirement gathering for the NewsRec system was done through end users, domain experts and self-evaluations and various requirement gathering techniques were employed to gather requirements. For the detailed requirement elicitation process please refer to chapter 4 of this document.	
Objective 5	Prepare a software requirement specification (SRS)

Information gathered through various requirement gathering techniques such as online surveys, interviews, requirement reuse and literature review were analysed to identify the functional and non-functional requirements of the system. For the detailed requirement elicitation process please refer to chapter 4 of this document.	
Objective 6	Selection of software and hardware resources
After analysing the requirements for the NewsRec system most suitable technologies, tools, APIs, libraries, platforms, algorithms and hardware requirements were determined. A detailed description of the software and hardware resources selection in included into chapter 6 of this report.	
Objective 7	Prepare software design specification
Designing of the system was carried out to satisfy the most appropriate techniques identified from the literature review and requirements identified through the requirement analysis. This information is added to this report as Chapter 5	
Objective 7	Develop the prototype
Prototype was implemented using factors identified in LR, SRS and Design chapters, this chapter contains the relevant code fragments and problems and solutions found during implementation phase. A detailed description of the prototype development is included to this report as Chapter 6	
Objective 8	Testing of the prototype
Testing of the prototype was done using through a devised testing plan and testing was done in quantitative and qualitative aspects. This information is added to this report as Chapter 7	
Objective 9	Evaluation of the work carried out
Evaluation of the project was carried out to evaluate the project using industrial experts and self-critique, then conduct a review of the evaluation findings to determine how far the project has successfully completed. For detailed explanation of the evaluation of the work carried out refer to chapter 8 of this document.	
Objective 10	Documentation
Documentation of each phase that was carried throughout the project life cycle was done with due thoroughness Chapters from 1 to 9 contain the documentation of the each objective of the project.	

Table 9.1- Contribution of objectives towards the completion of the project

9.3 Utilizing of Knowledge from Course Modules

Even though some modules of the M.Sc. program were not directly connected to the project research area, the knowledge and experience gathered through those contributed immensely toward the project success. The following modules distinguished themselves by providing significant contributions to the project completion.

Module	Web Services & SOA
Web services module contributed for the designing and development of the user interaction capturing and news publishing REST web services of the prototypes' presentation layer.	
Module	Project Management
Project management module helped to plan the project timeline, resources, and deliverables and come up with a risk mitigation plan.	

Module	Software Architecture
Software architecture module contributed to the successful requirement analysis and designing phase of the project.	
Module	Research Methods
Research methods module contributed to the initial project hypothesis identification and validation and later for the reviewing of various aspects related to the project.	

Table 9.2-Contribution of modules towards the completion of the project

9.4 Use of existing skills

- Existing structured design skills such as context diagrams and flow charts were exploited successfully for the designing process of the system.
- Existing programming skills on the Java platform such as Java persistence and java Hibernate technologies and Java Script language were utilized successfully for the completion of the project.

9.5 Learning outcomes

- Though the course content offered during the M.Sc. program provided knowledge on various software engineering related topics they didn't cover software agents or automated recommendation systems which were the basic foundation of the project. Therefore self-learning, online documentation and discussion with domain experts were used to gain the necessary knowledge to complete this project.
- Key technologies used in the project such as Redis server, JADE framework, ACL communication and NodeJS were self-learnt through online documentation and tutorials for the successful completion of the project.
- Evaluating the project qualitatively and quantitatively required sound knowledge on software quality assurance and testing skills and these knowledge were enhance through self-learning process.
- Critical thinking and formal documentation skills were developed through gradual learning and hands on experience.

It can be concluded that knowledge gained through the M.Sc. curriculum, existing skills and self-learned skills combined resulted in the successful completion of the project.

9.6 Problems and Challenges Faced

The extensive scope

The automated recommender system domain is a very broad and deep area. Though it was thought at the start of the project that the scope would be too small for a postgraduate level research with the in-depth research carried out it was evident that the scope would increase to a level which would not be able tackle within the allocated

time period. Literature review and requirement elicitation processes were used to scale down the scope to a manageable level.

Lack of academic publications on software agent scalability

Since software agent is a concepts mostly used in the academic arena it was difficult to find information about adaptation of the software agents in an enterprise level applications for scalability management. This problem was overcome by detailed self-study and experimentations.

Lack of knowledge and learning resources of key technologies

Lack of knowledge with key technologies such as JADE framework, Redis server and NodeJS was a major problem face during the implementation of the NewsRec system. Though there were online tutorials for those technologies only a handful was there describing the integrating of them together. This problem was overcome by detailed self-study and experimentations.

Time Constraint

Being a research project the inherited risk of frequent requirement fluctuations and lack of domain knowledge threatened to overrun the time allocated to the project. A possible solution to this problem was found by adapting the spiral development approach, which promises to keep the development iterative and support requirement fluctuations. It immensely helped to lower the time constraints and lead the project to a successful completion.

9.7 Limitations of the Research

Restriction on the news presentation format

This research was solely carried out focusing on recommending text based news articles but in the real world on line news articles can be presented in various other formats such as videos and audio. This research can be further expanded to explore in to possibility of recommending news articles of other presentation formats.

Restriction on the key success factors of a recommender system

The current research was carried out only taking accuracy and scalability of recommender system as key success factors. But apart from accuracy and scalability there are various other key success factors for a recommender system and this research can be further expanded to explore the accommodation of those left out key success factors of a recommender system in the online news domain.

9.8 Future Enhancements

Enhancement ID	ENH1	**Priority Level**	Medium
Enhancement	**Hybrid user feedback approach to solve reader cold starter problem**		
Description	Currently the NewsRec system adopts an explicit user feedback approach to solve the cold starter problem of a new user where the system prompts the reader to specify the preferred news category. But the accuracy of the recommendations done for a new user solely depends on the honesty of the user in specifying the preferred news category. Implicit user feedbacks which can		

	be gathered through social networks would reflect the interests of a reader more accurately. As highlighted at the literature review a hybrid user feedback approach which would combine explicit and implicit user feedback would give a better solution to the cold starter problem.		
Enhancement ID	ENH2	Priority Level	Medium
Enhancement	**Data mining approach to solve the user preference changes**		
Description	Currently the system is using a comparison of preference changes of a reader over a shorter perioc of time to understand any user preference changes. A better approach of solving this limitation would be use data mining approaches combined with an AI technology to understand reader preference changes. For this the system should be matured enough having faced various preference changes of users with different interests and the research should be expanded to see the possibility of the depth the data mining and AI approaches can be accommodated.		
Enhancement ID	ENH3	**Priority Level**	High
Enhancement	**Hybrid approach to assign descriptors to news articles**		
Description	The current approach adopted to assign descriptors to a news article is a manual approach and as the number of articles would increase the current manual approach would not be sufficient to assign descriptors to all the news articles and the approach would have to be further enhanced to support an automated descriptor assignment process with the use of NLP and ontology managements combined with the current manual approach..		
Enhancement ID	ENH4	**Priority Level**	Low
Enhancement	**Expansion to other news presentation formats**		
Description	As of now the system is developed to recommend only text based news articles but in reality news articles can be presented in video and audio formats as well. Another key future enhancement would be to extend the current solution to recommend video and audio news articles to readers. This proposed further enhancement can be taken another step forward to identify the preferred news presentation format of a reader and recommend news articles from that formats. For future enhancement to be accommodated further research have to carried on other news presentation formats		
Enhancement ID	ENH5	**Priority Level**	Low
Enhancement	**Expansion to a distributed system**		
Description	The current system focuses to achieve scalability through the software itself with the use of software agents but it can be further enhanced to the level of a distributed system which would be able handle the load of readers and articles in a more efficient manner with the use of both hardware and software level scalabilities.		
Enhancement ID	ENH6	**Priority Level**	Low
Enhancement	**Use of intelligent software agents**		
Description	In the current approach software agents are used to induce better collation and communication between different activities of the recommender system but agents can be made intelligent to		

	accommodate data mining and better load management processes to increase the accuracy and scalability of the developed system.		
Enhancement ID	ENH7	**Priority Level**	High
Enhancement	**Use of other user feedback sources to better understand reader preferences**		
Description	The developed system is only using explicit reader interaction such as link selections to determine the preference of a reader. But other user feedback sources like server logs, client agents and web cookies can be utilized to gather further user interactions such as time spent on reading an article to better understand and determine user preferences.		

Table 9.3-Contribution of modules towards the completion of the project

9.9 Contribution

NewsRec system aims to provide a solution to the current limitation faced by the online news industry where a reader would have to manually search for articles which matches their preferences. All most all the evaluated online news site either doesn't support any recommendation mechanism or if any recommendation mechanisms were provided they were having major limitation. The developed NewsRec system have contributed to the online news domain by solving the limitation of searching for preferred news articles through a generic feasible solution that would automatically recommend a list of preferred news articles to readers. When comparing online news environments with other online domains one striking feature is the rapid updation of news articles. Another limitation faced by any recommendation system is managing the scalability of the system with the increasing number of users and items. NewsRec system has contributed to the recommender domain by providing its own recommender algorithm and adaptation of software agents to manage the increasing level of scalability.

9.10 Concluding Remarks

NewsRec system presents a novel recommender system for the online news domain which would support different types of devices with different platforms through one presentation interface and which would try to manage the scalability of the system and improve the collation and communication between different activities of the recommender system by using software agent approach. The presented solution can be vastly beneficial for the online news domain to attract more news readers and manage the increasing number of users and articles in a more cost effective manner. The current solution opens up new research areas for the research community to further enhance the solution by trying out the future enhancements mentioned above.

REFERENCES

1. Abbas, O. A. (2008)'Comparisons Between Data Clustering Algorithms' *The International Arab Journal of Information technology* [online]. *5*(3), pp 320-325. Available at <http://www.ccis2k.org/iajit/PDF/vol.5,no.3/15-191.pdf> [Accessed 15 February 2014]

2. Adomavicius, G., and Tuzhilin, A. (2005)'Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions'*Knowledge and Data Engineering, IEEE Transactions on* [e-journal] 17(6), pp. 734-749. Available through IEEEXplore [Accessed 18 February 2014]

3. Arora, A. & Shah, P. (2011) Personalized News Prediction & Recommendation. *CS 229:Machine Learning Final Projects, [online].*Available<http://cs229.stanford.edu/proj2011/AroraShah-personalizedNewsPredictionandRecommendation.pdf> [Accessed 02 February 2014]

4. Bhattarai A. (2007) 'Google news personalization : scalable online collaborative filtering : paper review'[online]. Availablefrom<http://www.mathcs.emory.edu/~lxiong/cs570s08/share/slides/paper_google.pdf> [Accessed 18 February 2014]

5. Billsus, D., & Pazzani, M. J. (2000) 'User modeling for adaptive news access.*User modeling and user-adapted interaction*' *10*(2-3), pp.147-180.

6. Birukov, A., Blanzieri, E., andGiorgini, P. (2005) 'Implicit: An agent-based recommendation system for web search'*In Proceedings of the fourth international joint conference on Autonomous agents and multiagentsystems*[e-journal] pp. 618-624. Available through ACM [Accessed 15 February 2014]

7. Basilico, J., and Hofmann, T. (2004) 'Unifying collaborative and content-based filtering'*InProceedings of the twenty-first international conference on Machine learning* [e-journal] pp. 9. Available through ACM [Accessed 10 February 2014].

8. Beel, J., Langer, S., Genzmehr, M., andNürnberger, A. (2013) 'Persistence in Recommender Systems: Giving the Same Recommendations to the Same Users Multiple Times' In *Research and Advanced Technology for Digital Libraries* [e-journal] pp. 386-390. Available through Springer Berlin Heidelberg [Accessed 15 February 2014].

9. Beel, J., Genzmehr, M., Langer, S., Nürnberger, A., & Gipp, B. (2013) 'A comparative analysis of offline and online evaluations and discussion of research paper recommender system evaluation'

In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation* [e-journal] pp. 7-14. Available through ACM.[Accessed 26 March 2014]

10. Burke, R. (2002) 'Hybrid recommender systems: Survey and experiments'*Usermodeling and user-adapted interaction*, [online]12(4), pp. 331-370. Available at <http://213.55.83.52/ebooks/Photography/504705.pdf> [Accessed 20 February 2014]

11. Burke, R. (2007)'Hybrid web recommender systems' *InThe adaptive web*[online]. pp. 377-408. Available at < http://www.dcs.warwick.ac.uk/~acristea/courses/CS411/2010/Book%20-%20The%20Adaptive%20Web/HybridWebRecommenderSystems.pdf>
[Accessed 18 February 2014]

12. Cosley, D., Lam, S. K., Albert, I., Konstan, J. A., andRiedl, J. (2003) 'Is seeing believing?: how recommender system interfaces affect users' opinions'*In Proceedings of the SIGCHI conference on Human factors in computing systems* [e-journal] pp. 585-592. Available through ACM [Accessed 10 February 2014]

13. Chen, M., Yang, C., Chen, J., and Yi, P. (2013)'A Method to Solve Cold-Start Problem in Recommendation System based on Social Network Sub-community and Ontology Decision Model' [online] pp 160-166. Available from <http://www.atlantis-press.com/php/download_paper.php?id=10372> [Accessed 11 February 2014]

14. Chen, Z., Jiang, Y., & Zhao, Y. (2010) 'A Collaborative Filtering Recommendation Algorithm Based on User Interest Change and Trust Evaluation' *JDCTA*, *4*(9), pp.106-113.Available at < http://www.aicit.org/JDCTA/ppl/13_JDCTAS19-557001.pdf> [Accessed 22 March 2014]

15. Das, A. S., Datar, M., Garg, A., and Rajaram, S. (2007) 'Google news personalization: scalable online collaborative filtering' *In Proceedings of the 16th international conference on World Wide Web.* [online] pp. 271-280. Available at <http://www/ra/ethz.ch/CDstore/www2007.org/papers/paper570.pdf > [Accessed 05 February 2014]

16. Dubois, A., &Gadde, L. E. (2002)'Systematic combining: an abductive approach to case research' *Journal of business research* [online] *55*(7), pp. 553-560. Available at <https://noppa.aalto.fi/noppa/kurssi/tu-22.z/materiaali/readings_for_the_case_studies_course_6.pdf> [Accessed 03 March 2014]

17. Felfernig, A., and Burke, R. (2008)'Constraint-based recommender systems: technologies and research issues'*InProceedings of the 10th international conference on Electronic commerce*[e-journal] Available through ACM [Accessed 18 February 2014]

18. Gauch, S., Speretta, M., Chandramouli, A., and Micarelli, A. (2007) 'User profiles for personalized information access'*In The adaptive web*[e-journal] pp. 54-89. Available through Springer Berlin Heidelberg [Accessed 15 February 2014]

19. Ghazanfar, M. A., andPrugel-Bennett, A. (2010) 'A scalable, accurate hybrid recommender system'*In Knowledge Discovery and Data Mining, WKDD'10. Third International Conference* [ejournal] pp. 94-98. Available throughIEEEXplore [Accessed 16 February 2014]

20. Google news (2014) Available at <https://news.google.com/> [Accessed 02 February 2014]

21. Gunawardana, A., and Meek, C. (2009)'A unified approach to building hybrid recommender systems' In *Proceedings of the third ACM conference on Recommender systems* [e-journal].pp. 117-124. Available through ACM. [Accessed 03 March 2014]

22. Gupta, P., Goel, A., Lin, J., Sharma, A., Wang, D., andZadeh, R. (2013)'Wtf: The who to follow service at twitter'*In Proceedings of the 22nd international conference on World Wide Web* [e-journal] pp. 505-514. Available through ACM [Accessed 15 February 2014]

23. Hanani, U., Shapira, B., and Shoval, P. (2001)'Information filtering: Overview of issues, research and systems'*UserModeling and User-Adapted Interaction*,[online]11(3), pp. 203-259. Available at<http://www.researchgate.net/publication/220116306_Information_Filtering_Overview_of_Issues_Research_and_Systems/file/9fcfd50745a87e9bc0.pdf> [Accessed 12 February 2014]

24. Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004)'Evaluating collaborative filtering recommender systems'*ACM Transactions on Information Systems (TOIS)* [e-journal]. 22(1), pp. 5-53. Available through ACM [Accessed 18 February 2014]

25. Hernández delOlmo, F., and Gaudioso, E. (2008) 'Evaluation of recommender systems: A new approach' *Expert Systems with Applications* [online] *35*(3), 790-804. Available at < http://163.17.12.2/drupal/sites/default/files/Evaluation%20of%20recommender%20systems%20A%20new%20approach.pdf> [Accessed 11 February 2014]

26. Kaur, M.andKaur, U (2013)‘ Comparison between K-mean and hierarchical algorithm using query redirection’ *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)* [online] 3(7),pp 1454-1459. Available at <http://www.ijarcsse.com/docs/papers/Volume_3/7_July2013/V3I7-0565.pdf>[Accessed 11 February 2014]

27. Kompan, M., and Bieliková, M. (2010) ‘Content-based news recommendation’*E-commerce and web technologies* [e-journal] pp. 61-72. Available through Springer Berlin Heidelberg [Accessed 02 February 2014]

28. Koren, Y., Bell, R., &Volinsky, C. (2009) ‘Matrix factorization techniques for recommender systems’ *Computer*[online]. *42*(8), pp. 30-37. Available at <https://datajobs.com/data-science-repo/Recommender-Systems-%5BNetflix%5D.pdf> [Accessed 05 March 2014]

29. Lin, J., Sugiyama, K., Kan, M. Y., and Chua, T. S. (2013) ‘Addressing cold-start in app recommendation: latent user models constructed from twitter followers. *In Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* [e-journal] pp. 283-292. Available through ACM [Accessed 18 February 2014]

30. Lindgaard, G. and Chattratichart, J. (2007)‘Usability testing: what have we overlooked?’*In Proceedings of the SIGCHI conference on Human factors in computing systems*[e-journal]. pp. 1415- 1424. Available through ACM [Accessed 15 March 2014]

31. Linden, G., Smith, B. and York, J. (2003) ‘Amazon.com recommendations, item –item collaborating filtering’ Internet *Computing,* [e-journal] *7*(1), pp.76-80. Available through IEEEXplore [Accessed 10 February 2014]

32. Li, L., Wang, D., Li, T., Knox, D., and Padmanabhan, B. (2011) ‘Scene: a scalable two-stage personalized news recommendation system’ *In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*.[online] pp. 125-134. Available at <http://users.cs.fiu.edu/~taoli/pub/p125-li-sigir2011.pdf> [Accessed 10 February 2014]

33. Marston, T. (2012). What is the 3-Tier Architecture?. [online]. Tony Marston. <http://www.tonymarston.net/php-mysql/3-tier-architecture.html> [Accessed: 05th February 2014].

34. Melville, P., Mooney, R. J., and Nagarajan, R. (2002)‘Content-boosted collaborative filtering for improved recommendations’*In Proceedings of the Eighteenth National Conference on Artificial*

Intelligence(AAAI-2002) [online].pp. 187-192. Available at <http://www.aaai.org/Papers/AAAI/2002/AAAI02-029.pdf>[Accessed 16 February 2014]

35. Morais, A. J., Oliveira,E. and Jorge, A.M. (2012) 'A multi-agent recommender system.'*Distributed Computing and Artificial Intelligence*.[online] pp. 281-288. Available through Springer Berlin Heidelberg [Accessed 02 February 2014]

36. Papagelis, M., andPlexousakis, D. (2005) 'Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents' *Engineering Applications of Artificial Intelligence* [e-journal]*18*(7),pp 781-789. Available through ScineceDirect [Accessed 12 February 2014]

37. Pagare, R., and Patil, S. A. (2013) 'Study of Collaborative Filtering Recommendation Algorithm-Scalability Issue'*International Journal of Computer Applications*, 67(25), pp. 10-15

38. Rafsanjani, A.H.N., Salim, N., Aghdam, A.R. and Fard, K.B. (2013) 'Recommendation systems: A Review' *International Journal of Computational Engineering Research.* 3(5), pp. 47-52.

39. Rashid, A. M., Karypis, G., and Riedl, J. (2008)'Learning preferences of new users in recommender systems: an information theoretic approach' *ACM SIGKDD Explorations Newsletter* [e-journal]. *10*(2), 90-100. Available through ACM [Accessed 12 February 2014]

40. Rauber, A., Pampalk, E., and Paralič, J. (2000)'Empirical Evaluation of Clustering Algorithms' Journal *of information and organizational sciences*, *24*(2),pp 195-209.

41. Ricci, F., Rokach, L., and Shapira, B. (2011)'*Introduction to recommender systems handbook'* [online] (pp. 1-35)Newyork : Springer. Available from <http://link.springer .com/chapter/10.1007/978-0-387-85820-3_1> [Accessed 14 February 2014]

42. Rusu, C., Roncagliolo, S., Rusu, V., and Collazos, C. (2011)'A methodology to establish usability heuristics'*In ACHI 2011, The Fourth International Conference on Advances in Computer-Human Interactions*, Gosier, Guadeloupe, France: IARIA, 2011

43. Singh, A., Juneja, D., & Sharma, A. K. (2011) 'Agent Development Toolkits.*arXiv preprint arXiv:1111.5930*'

44. Shang, S., Hui, Y., Hui, P., Cuff, P., and Kulkarni, S. (2014) 'Beyond Personalization and Anonymity: Towards a Group-Based Recommender System'*InProceedings of the twenty ninth Symposium on*

Applied Computing (SAC)[online]. Available at <http://digitalmilieu.net/documents/Shang_Hui_Beyond%20Personalization%20and%20Anonymity.pdf > [Accessed 16 February 2014]

45. Sharma, N., Bajpai, A., and Litoriya, M. R. (2012) 'Comparison the various clustering algorithms of weka tools.' *International journal of Emerging technology and Advanced engineering*[online]. 2(5), pp73-80. Available at <http://www.ijetae.com/files/Volume2Issue5/IJETAE_0512_13.pdf> [Accessed 12 February 2014]

46. Takács, G., Pilászy, I., Németh, B., & Tikk, D. (2009) 'Scalable collaborative filtering approaches for large recommender systems' *The Journal of Machine Learning Research*, *10*, pp 623-656. Available at < http://www.sze.hu/~gtakacs/download/jmlr_2009.pdf> [Accessed 22 March 2014]

47. Tan, W., Xu, W., Yang, F., Xu, L., & Jiang, C. (2013) 'A framework for service enterprise workflow simulation with multi-agents cooperation' *Enterprise Information Systems*[e-journal].*7*(4), pp.523-542.Available through Taylor & Fransis Online. [Accessed 21 March 2014]

48. Tatar, A., Leguay, J., Antoniadis, P., Limbourg, A., De Amorim, M. D. and Fdida, S. (2011) 'Predicting the popularity of online articles based on user comments'. *In Proceedings of the International Conference on Web Intelligence, Mining and Semantics*[e-journal] pp. 67. Available through ACM [Accessed on 10 February 2014]

49. Tavakolifard, M., Gulla, J. A., Almeroth, K. C., Ingvaldesn, J. E., Nygreen, G., and Berg, E. (2013) 'Tailored news in the palm of your hand: a multi-perspective transparent approach to news recommendation'. *In Proceedings of the 22nd international conference on World Wide Web companion*.[online] pp. 305-308. Available at <http://telemedicina.unifesp.br/pub/Events/2013-05%20-%20WWW2013/www2013/www2013.org/companion/p305.pdf> [Accessed 10 February 2014]

50. Techcrunch (2014) 'Digital News finally as popular as News papers in the UK' Available at <http://techcrunch.com/2014/06/25/print-vs-digital/>[Accessed 03 February 2014]

51. The state of the media (2013)'An Annual report on American Journalism'Available at <http://stateofthemedia.org/2013/overview-5/key-findings/>[Accessed 03 February 2014]

52. Thurman, N., and Schifferes, S. (2012) 'The future of personalization at news websites: lessons from a longitudinal study'. *Journalism Studies*[online] *13*(5-6), pp.775-790. Available at <http://www.tandfonline.com/doi/full/10.1080/1461670X.2012.664341> [Accessed 05 February 2014]

53. Times (2009) Available at <http://business.timesonline.co.uk/tol/business/industry/technology/article6858991.ece> [Accessed 02 February 2014]

54. Van Meteren, R., and Van Someren, M. (2000)'Using content-based filtering for recommendation'*InProceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop*[online] Available at <http://users.ics.forth.gr/~potamias/mlnia/paper_6.pdf> [Accessed 14 February 2014]

55. Wang, J., Li, Q., & Chen, Y. P. (2010) 'User comments for news recommendation in social media' *In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval* [e-journal] pp. 881-882. Available through ACM [Accessed 04 February 2014]

56. Wartenberg, C. and Holmqvist, K. (2005) 'Daily Newspaper Layout-Designers, Predictions of Readers, Visual Behaviour-A Case Study' *ACM Lund University Cognitive Studies126: 1101-8453*[e-journal]Available through ACM [Accessed 04 February 2014]

57. Yafooz, W. M. S., et al. (2011) 'Challenges and issues on online news management' *Control System, 2011 IEEE International Conference on Computing and Engineering (ICCSCE).*BatuFerringhi, Penang, Malaysia. 25-27 November 2011

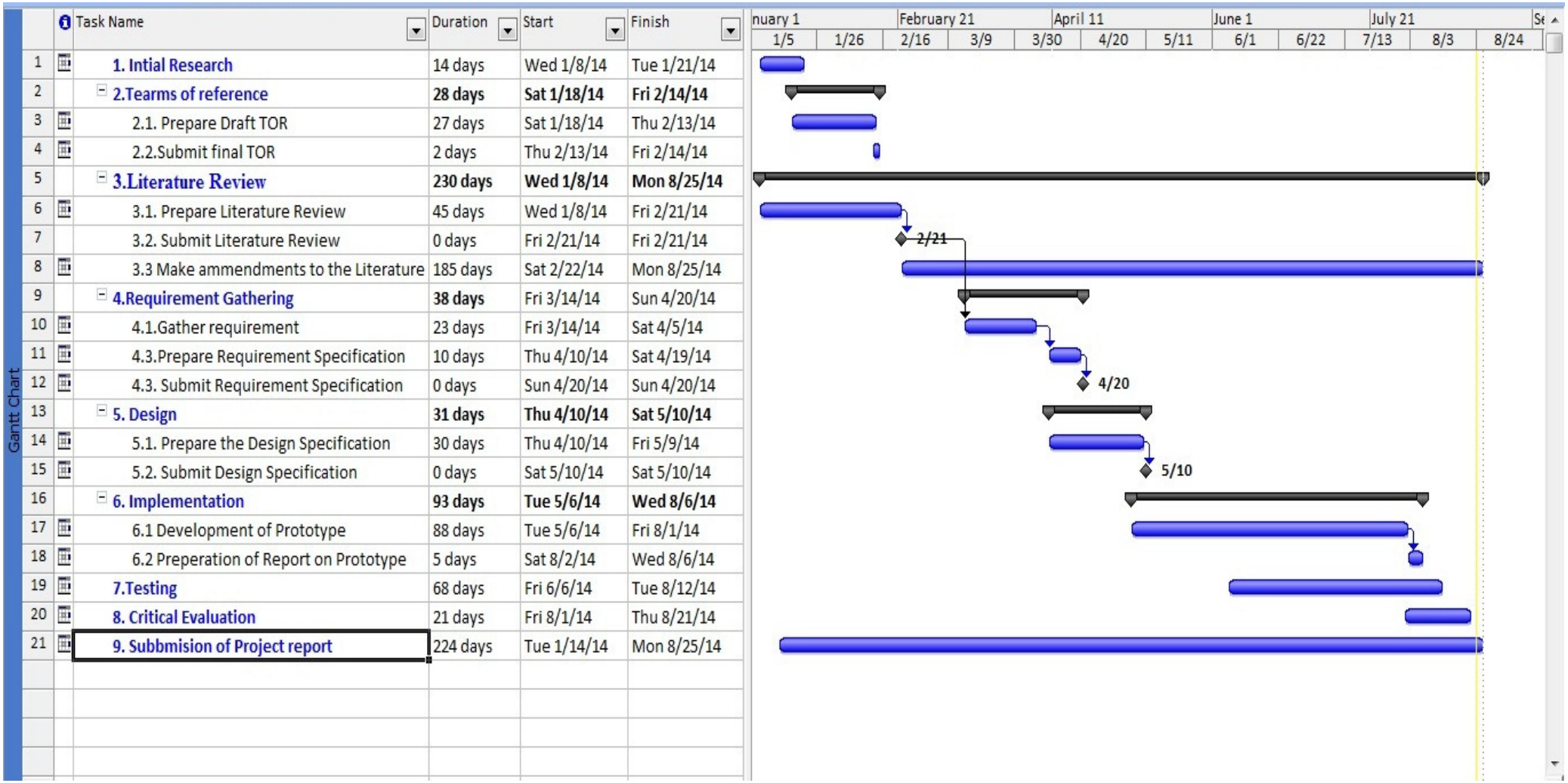
58. Yu, H., Zheng, D., Zhao, B. Y., & Zheng, W. (2006) 'Understanding user behavior in large-scale video-on-demand systems' *In ACM SIGOPS Operating Systems Review*[e-journal].40(4) pp. 333-344.Available through ACM. [Accessed 25 March 2014]

59. Yuanhua, Lv., Taesup, Moon.,Pranam, Kolari., Zhaohui, Zheng., Xuanhui, Wawng. & Yi, Chang. (2011) 'Learning to Model Relatedness for News Recommendation' *WWW-11 Proceedings of the 20th international conference on World wide web.* Hyderabad International Convention Centre(HICC), Hyderabad, India. 28th March

60. Zeng, W., Zeng, A., Shang, M. S., and Zhang, Y. C. (2013) 'Information filtering in sparse online systems: recommendation via semi-local diffusion'*PloS one* [online]. 8(11) Available at <http://www/plosone.org/ article/info%3Adoi% 2F10.1371%2Fjournal. pone.0079354#pone-0079354-g009> [Accessed 12 February 2014]

61. Ziegler, C. N., McNee, S. M., Konstan, J. A., andLausen, G. (2005) 'Improving recommendation lists through topic diversification' *In Proceedings of the 14th international conference on World Wide Web* [e-journal]. pp. 22-32. Available through ACM [Accessed 18 February 2014]

Appendix A – Gantt Chart



	Task Name	Duration	Start	Finish
1	1. Intial Research	14 days	Wed 1/8/14	Tue 1/21/14
2	2.Tearms of reference	**28 days**	**Sat 1/18/14**	**Fri 2/14/14**
3	2.1. Prepare Draft TOR	27 days	Sat 1/18/14	Thu 2/13/14
4	2.2.Submit final TOR	2 days	Thu 2/13/14	Fri 2/14/14
5	3.Literature Review	**230 days**	**Wed 1/8/14**	**Mon 8/25/14**
6	3.1. Prepare Literature Review	45 days	Wed 1/8/14	Fri 2/21/14
7	3.2. Submit Literature Review	0 days	Fri 2/21/14	Fri 2/21/14
8	3.3 Make ammendments to the Literature	185 days	Sat 2/22/14	Mon 8/25/14
9	4.Requirement Gathering	**38 days**	Fri 3/14/14	Sun 4/20/14
10	4.1.Gather requirement	23 days	Fri 3/14/14	Sat 4/5/14
11	4.3.Prepare Requirement Specification	10 days	Thu 4/10/14	Sat 4/19/14
12	4.3. Submit Requirement Specification	0 days	Sun 4/20/14	Sun 4/20/14
13	5. Design	**31 days**	**Thu 4/10/14**	**Sat 5/10/14**
14	5.1. Prepare the Design Specification	30 days	Thu 4/10/14	Fri 5/9/14
15	5.2. Submit Design Specification	0 days	Sat 5/10/14	Sat 5/10/14
16	6. Implementation	**93 days**	**Tue 5/6/14**	**Wed 8/6/14**
17	6.1 Development of Prototype	88 days	Tue 5/6/14	Fri 8/1/14
18	6.2 Preperation of Report on Prototype	5 days	Sat 8/2/14	Wed 8/6/14
19	7.Testing	68 days	Fri 6/6/14	Tue 8/12/14
20	8. Critical Evaluation	21 days	Fri 8/1/14	Thu 8/21/14
21	9. Subbmision of Project report	224 days	Tue 1/14/14	Mon 8/25/14

Appendix B – Gantt Chart

Following use case description represents scenario where the reader search for an article.

Use case name	**Search for news articles by keyword or descriptor**
Description	This use case maps the process of a reader searching for articles using a sting of characters or by clicking a article descriptor value
Participating actors	Registered readers of the system
Pre-conditions	1. System should be functional 2. Reader should be logged on the system. 3. News articles with a title and article descriptor values should be available in the system.
Extended use cases	View article
Included use cases	None
Main flow	1. Reader enters a string of characters to be searched in the article title or article descriptors 2. System traverse through all the article titles & article descriptors to identify matches against the entered string. 3. System prepares a list of article titles which contains the entered string in the article title or in the article descriptors. 4. System sorts the articles in the list according to the rating values of those articles and presents the sorted list to the readers. 5. Extended use case “View articles” may get executed 6. Use case terminates
Alternative flows	1. Reader clicks on a displayed article descriptor value 2. System traverse through all the article titles & article descriptors to identify matches against the entered string. 3. System prepares a list of article titles which contains clicked article descriptor in the article title or in the article descriptors. 4. System sorts the articles in the list according to the rating values of those articles and presents the sorted list to the readers. 5. Extended use case “View articles” may get executed Use case terminates
Exceptional flows	E1: In step1, the reader enters a string which is unavailable in the article title or the article descriptors.

	1. An error message is thrown, and the use case ends in failure. E2: During the execution of the use case Internet connection interrupts or fails. 1. Use case ends in failure.
Post conditions	System displays a list of article titles for the reader to choose a one to view its' content.

Use case description for search for news articles by keyword or descriptor

Following use case description represents the scenario where a user selects an article to view.

Use case name	**View article**
Description	This use case is to show the scenario where a user selects an article from a list to view the content of the selected article.
Participating actors	Registered news readers
Pre-conditions	1. System should be functional. 2. Reader should be logged on the system. 3. Reader should be presented with a list of article titles
Extended use cases	Indicate preference
Included use cases	Update article rate value, Update user preference profile
Main flow	1. Reader clicks on an article title displayed on the list presented by the system 2. System fetches the content of the article selected by the user. 3. System displays the content to the reader. 4. Reader may execute the extended use case "Indicate preference". 5. System keeps a track about the selection under reader's dynamic preference list. 6. Included use cases will be executed 7. Use case terminates.
Alternative flows	None
Exceptional flows	E1: During the execution of the use case Internet connection interrupts or fails. 1. Use case ends in failure.
Post conditions	System will use the use the selection information to update the user preference profile and update article rate value

Use case description for Search for view article

Following use case description represents the scenario where the system updates the reader preference profile of a reader.

Use case name	**Update reader preference profile**

Description	This use case describes flow of events and activities that will be carried out when the system tries to update reader preference profile.
Participating actors	Registered news readers
Pre-conditions	1. System should be functional
Extended use cases	None
Included use cases	None
Main flow	1. System reads the dynamic preference list of the reader. 2. System calculates the new preference value for each element of common interests of established interests and updates the user preference with new values. 3. System calculates the preference floating indicator for each element of common interests of established interests. 4. System decides which common interests of established interests to be moved to the common interests of previous interests and moves those. 5. System calculates the new preference value for each element of unique interests of established interests and updates the user preference with new values. 6. System calculates the preference floating indicator for each element of unique interests of established interests. 7. System decides which unique interests of established interests to be moved to the unique interests of previous interests and moves those. 8. System calculates the new preference value for each element of common interests of recent interests and updates the user preference with new values. 9. System calculates the preference floating indicator for each element of common interests of recent interests. 10. System decides which common interests of recent interests to be removed and removes those. 11. System calculates the new preference value for each element of unique interests of recent interests and updates the user preference with new values. 12. System calculates the preference floating indicator for each element of unique interests of recent interests. 13. System decides which unique interests of recent interests to be removed and removes those. 14. Use case terminates.
Alternative flows	None

Exceptional flows	E1: During the execution of the use case Internet connection interrupts or fails. 1. Use case ends in failure.
Post conditions	System will save the new reading profile values of the reader.

Use case description for Update reader preference profile

Following use case description represents the scenario where the system updates the rate value of an article

Use case name	**Update article rate value**
Description	This use case describes flow of events and activities that will be carried out when the system tries to update the rate value of an article.
Participating actors	Registered news readers
Pre-conditions	1. System should be functional
Extended use cases	None
Included use cases	None
Main flow	1. System reads the dynamic preference list of the reader. 2. System calculates the new preference value for each element of common interests of established interests and updates the user preference with new values. 3. System calculates the preference floating indicator for each element of common interests of established interests. 4. System decides which common interests of established interests to be moved to the common interests of previous interests and moves those. 5. System calculates the new preference value for each element of unique interests of established interests and updates the user preference with new values. 6. System calculates the preference floating indicator for each element of unique interests of established interests. 7. System decides which unique interests of established interests to be moved to the unique interests of previous interests and moves those. 8. System calculates the new preference value for each element of common interests of recent interests and updates the user preference with new values. 9. System calculates the preference floating indicator for each element of common interests of recent interests. 10. System decides which common interests of recent interests to be removed and removes those. 11. System calculates the new preference value for each element of unique interests of

	recent interests and updates the user preference with new values. 12. System calculates the preference floating indicator for each element of unique interests of recent interests. 13. System decides which unique interests of recent interests to be removed and removes those. 14. Use case terminates.
Alternative flows	None
Exceptional flows	E1: During the execution of the use case Internet connection interrupts or fails. 15. Use case ends in failure.
Post conditions	System will save the new reading profile values of the reader.

Appendix C – Test Cases

Test cases and results of functional requirement testing.

Test case	Scenarios	Input data	Expected result	Actual result	Status	Success rate
2.1	Register to the system	Non existing user name and a password of choice	Successful signup to the system.	Successful sign up to the system.	Pass	100%
2.2	Register to the system	Existing user name and a password of choice	Error message displayed about the user name existence	Error message displayed about the user name existence	Pass	100%
2.3	Register to the system	Non existing user name and a password of choice without specifying the preferred news category	Error message displayed about non mentioning of the preferred news category	Error message displayed about non mentioning of the preferred news category	Pass	100%
2.4	Login to the system	Correct user name and password	Successfully login	Successfully login	Pass	100%
2.5	Login to the system	Wrong user name and correct password	Login fails	Login fails	Pass	100%
2.6	Login to the system	Correct user name and wrong password	Login fails	Login fails	Pass	100%
2.7	Logout from the system	Press the logout button	Successfully Logout	Logout from the system	Pass	100%
2.8	Search article with an existing keyword	String values “Cricket”, “Local” and “Environment” with one at a time	Display a list of article which contains “Cricket”, “Local” and “Environment” one at a time	Display a list of article which contains “Cricket”, “Local” and “Environment” one at a time	Pass	100%
2.9	Search article with a non existing	String values “fjehfue123”, “vno123ci” and	Displays a message that articles containing “fjehfue123”,	Displays a message that articles containing “fjehfue123”,	Pass	100%

	keyword	“12kjf&#sd”	“vno123ci” and “12kjf&#sd”can’t be found	“vno123ci” and “12kjf&#sd”can’t be found		
2.10	Select an article from the list	User click on a link	User preference value should be modified	User preference value is be modified	Pass	100%
2.11	Select an article from the list	User click on a link	Article rate value should be updated	Article rate value is updated	Pass	100%
2.12	Like an articles	Click on the like button	Update number of likes for that article	Page refresh and update number of votes	Pass	100%
2.13	Like an articles	Click on the like button	Article rate value should be updated	Article rate value is updated	Pass	100%