

# **Faculty of Computers and Information**

## **Cairo University**

### **Computer Science Department**

**“Watch and Rate”**

Review System using sentiment analysis

**Supervised by:** Dr. Hanaa Bayomi

**Prepare by:**

Esraa Ahmed Fathallah	20120076	CS-IS
Hosam Hesham Abdel Fatah	20120152	CS-IS
Heba Syed Shamardy	20120476	CS-IS
Noha Magdy Saber	20120431	CS-IS



## Contents

Figures .....	4
Tables .....	5
Acknowledgement .....	6
Document Purpose and Audience .....	6
Chapter 1: Introduction .....	7
1.1 Software purpose .....	7
1.2 Software scope .....	7
1.3 Definitions, acronyms and abbreviations .....	8
1.4 Motivation .....	8
1.5 Problem definition .....	8
1.6 Project Objective .....	9
1.7 Project development Methodology .....	9
1.8 Report organization .....	10
Chapter 2: Related Work .....	11
2.1 Existing systems .....	11
2.1.1 IMDB .....	11
2.1.2 Common sense media .....	13
2.1.3 New Movies Review .....	14
2.1.4 Rotten Tomatoes .....	16
Chapter 3: System Analysis .....	18
3.1 Project Specifications .....	18
3.1.1 Introduction .....	18
3.1.2 Purpose .....	18
3.1.2 How it works .....	19
3.2 Application Architecture .....	19
3.2.1 REST .....	20
3.2.2 Service Layer and Database Layer .....	21
3.2.3 Review sentiment Analysis .....	21
3.2.3.1 Annotation phase .....	21
3.2.3.2 Feature Extraction .....	22
3.2.3.3 Features chosen .....	22
3.2.3.4 Tools .....	23



3.3 Planning .....	26
3.4 System Analysis .....	28
3.4.1 Feasibility .....	28
3.4.2 SWOT Analysis .....	28
3.5 Requirements Specification .....	29
3.5.1 Functional Requirements .....	29
3.5.2 Non Functional Requirements .....	32
3.6 System Model .....	35
3.6.1 Use case description .....	35
3.6.2 Use case Diagram .....	37
3.6.3 Sequence diagrams .....	38
3.6.4 Class Diagram .....	40
3.6.5 Entity Relationship Diagram .....	41
3.6.6 System test cases .....	41
 Chapter 4: Snap shots of application .....	44
4.1 Introduction .....	44
 Chapter 5: Result summary.....	61
5.1 Result from database .....	61
5.2 Training Results using Weka .....	65
5.2.1 Introduction .....	65
5.2.2 First Trial .....	66
5.2.3 Second Trial .....	67
5.2.4 Third Trial .....	67
5.2.5 Fourth trial .....	68
5.2.6 Fifth Trial .....	69
5.2.7 Final trial .....	70
5.3 Ranking attributes in the model .....	72
5.4 Future work .....	72

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

---



## Figures

Figure 1 application architecture.....	20
Figure 2 Annotation for negative and positive sentences .....	23
Figure 3 SentiWordNet lexicon.....	25
Figure 4 Basic dependencies Stanford parser .....	25
Figure 5 Training data for WEKA.....	26
Figure 6 WEKA Generated Model.....	27
Figure 7 Time planning.....	28
Figure 8 use case diagram .....	38
Figure 9 sequence diagram 1.....	39
Figure 10 sequence diagram 2.....	40
Figure 11 class diagram .....	41
Figure 12 entity relational diagram.....	42
Figure 13 start screen of application .....	45
Figure 14 Sign in screen.....	46
Figure 15 sign up screen.....	47
Figure 16 home screen.....	48
Figure 17 recently movies from home screen.....	49
Figure 18 recently movies .....	50
Figure 19 recently reviews from home screen .....	51
Figure 20 notification screen .....	52
Figure 21 search screen (movies results).....	53
Figure 22 search screen (reviews results) .....	54
Figure 23 movie details screen .....	55
Figure 24 movie details reviews .....	56
Figure 25 add review screen.....	57
Figure 26 slide panel screen .....	58
Figure 27 user reviews screen .....	59
Figure 28 review details screen .....	60
Figure 29 user profile screen.....	61
Figure 30 first trial Weka training .....	67
Figure 31 second trails Weka training.....	68
Figure 32 third test Weka training .....	69
Figure 33 fourth test Weka training.....	70
Figure 34 fifty test Weka training .....	71
Figure 35 final test Weka training.....	72
Figure 36 Ranking important attributes to less important.....	73



## Tables

Tables 1 Definitions, acronyms and abbreviations	8
Tables 2 Similar features between IMDb and us	11
Tables 3 Different features between IMDb and us	13
Tables 4 Similar features between Common Sense and us	14
Tables 5 Different features between Common Sense and us	15
Tables 6 Similar features between NMR and us	16
Tables 7 Different features between NMR and us	16
Tables 8 Similar features between Rotten Tomatoes and us	17
Tables 9 Different features between Rotten Tomatoes and us	18
Tables 10 Feasibility Study	29
Tables 11 Use Case Description	37
Tables 12 System test cases	44
Tables 13 comments data table	62
Tables 14 notification data table	63
Tables 15 user data table	63
Tables 16 movie data table	64
Tables 17 Review data table	65



## Acknowledgement

First we want to thank Dr. Reem and Dr. Hanaa a lot for their support, guidance and encouragement even when we changed in the idea of the project. We're very thankful for their flexibility, giving us time and helping us through each step. We're also grateful for TA. Marwa Nabil for giving us technical help to do every step right. We're so grateful.

“The team”

## Document Purpose and Audience

### - What is this document?

This document is a Software Requirements' Specification which describes the requirements to build rating and review system.

### - Who is expected to read it?

- Supervisor of the project.
- The team of designers and developers who will work on this project
- Project management team or developers who need to see rating and feedback system requirements.
- Reviewers of the document / Evaluators of the project
- Students of FCI who need to read old graduation projects.



## Chapter 1: Introduction

### 1.1 Software Purpose

The Watch and Rate system is designed to help people who are interested in movies, their news, movies' actors, and reviewers' opinions about these movies. The Watch and Rate system helps its users to search for movie rating, reviews, and regular users' reviews to make a clear opinion about a certain movie. We use the following APIs: IMDb's rating, Rotten Tomatoes' rating and New York Times paper's critics' to retrieve movies' reviews.

The main goals of Watch and Rate system is to:

- a) Allow users to check movies' reviews especially when they would like to new movie.
- b) Allow users to receive movie news, reviews and updates.
- c) Allow users to Share bad, good, and satisfying movies experiences and discuss them with others.
- d) Allow users to Search for movies by its title and get movie's info, reviews, users' reviews, IMDb ratting, Rotten Tomatoes ratting and our user's rating.
- e) Allow users to Make Recommendations according to their favorite movie categories.

### 1.2 Software Scope

It's a web service using android as a frontend and a server as a backend where the user can search for a movie to get its reviews. We're making a review system for movies where any user can write his/her opinion about a certain movie or search for reviews related to a movie.

The Main features of our application is as following:

- Providing movies' info using TMDb.
- Writing reviews, rating them and commenting on them.
- Searching the proposed system for users' movie reviews.
- Showing New York Times for critics' reviews.
- Recommending movies.



- 
- Notifying the user about:
    - Comments on his/her review.
    - Rates-on his/her review.

### 1.3 Definitions, acronyms and abbreviations

<b>TMDb</b>	The Movie Database API.  TMDb is a popular, user editable database for movies and TV shows.
<b>NYT</b>	New York Times paper as to get its critics' reviews for our users.
<b>ERD</b>	Entity relation diagram to make our system database.
<b>API</b>	An API is application program interface.  We have 2 APIs: <ul style="list-style-type: none"><li>- NYT API</li><li>- TMDb API</li></ul>

Table 1: Definitions, acronyms and abbreviations

### 1.4 Motivation

Most of current applications that provide movie ratings aim to provide just normal users' reviews and movie rating based on them for their users. Though these applications provide users' reviews but this reviews' are not useful enough for its users. The provided reviews lack interaction with others through comments or rating how much these reviews were helpful. Hence, we provide movie rating based on sentiment analysis of users' reviews, users interact with each other through commenting and rating each other reviews and a user gets score for his/her helpful reviews.

### 1.5 Problem definition

When people are board, they tend to find something fun to do. This might be a movie that's enjoyable to watch. This's when Watch and Rate interfere to



recommend a movie based on analyzing reviews for their sentiment. It also gives the users a chance to interact with others' ideas through comments. Watch and Rate is basically for people interested in movies.

## 1.6 Project Objective

The objective of the project is to provide an application that anyone interested in movies field can use. It will help users in deciding whether the movie they want to watch is good enough or not.

The features of the developed application are as follows:

- a. A movie API (TMDB API) for extracting movies info and ratings.
- b. A review API (New York Times API) for extracting movies reviews.
- c. Sentiment analysis for analyzing user's reviews.
- d. The server will process the data collected from user's reviews that entered in system.
- e. Make a system rating (based on sentiment analysis) on the data, and save them in application database.
- f. The server will be connected directly with MySQL database.
- g. Android interface will be developed to rad the stored data inside the database and show them in case of the user required a specific movie review.

## 1.7 Project development Methodology

We use agile methodology which is an alternative to project management used basically in software development to help teams to be flexible, more productive I an incremental and iterative work.

The main phases of our system are as follows:

1. Phase one: analysis for choosing the requirements
2. Phase two: designing the diagrams needed to develop these requirements



3. Phase three: development of these requirements
4. Phase four: testing what has been accomplished.
5. Write documentation about this part

Each version of the system will contain these four phases until it's a functioning prototype then we go back to phase one again to put new feature and so on.

## 1.8 Report organization

Chapter 2: introduces the proposed system and the related systems. In each related system the extra features in both systems – Watch and Rate system and the other system – given the reasons why some features are left out in Watch and Rate.

Chapter 3: introduces Watch and Rate's design and analysis including the functional and non-functional requirements, class diagram, use case description and diagram, sequence diagrams, entity relationship diagram and system's test cases.

Chapter 4: introduce screen shots of application and how user interact with it.

Chapter 5: introduce results and summary after finish project.



## Chapter 2: Related Work

### 2.1 Existing Systems

#### 2.1.1 IMDB

Feature Name
Search
Sign up
Sign in/ Sign in with social networks
Show trending movies
Rate movie
Show popular movies by genre
Play trailer
Show coming soon movies
Top rated movies
User history
Critics reviews for movie
Connect with New York times API
Movie info
Notification

Table 2: Similar features between IMDb and us

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



Different features	Watch & Rate	IMDb	Reason
US Box office	No	Yes	Because app is mainly for rating & movie reviews
Show movies news/updates	No	Yes	Because app is mainly for rating & movie reviews
Lowest rated movies	No	Yes	User does not need to know least rated movies
Gallery	No	Yes	App core does not need it
Show all available tickets/show times	No	Yes	Because app is mainly for rating & movie reviews
Show awards central	No	Yes	Because app is mainly for regular people opinions about movie
Voice search	No	Yes	Future work
Watch list	No	Yes	Future work
Link to buy movie	No	Yes	App is not commercial
Share movie	No	Yes	Because app is mainly for rating & movie reviews
User score	Yes	No	
Rate Review	Yes	No	
Write Review	Yes	No	
Comment on Review	Yes	No	
Edit Profile	Yes	No	

Table 3: Different features between IMDb and us



### 2.1.2 Common sense media

<b>Feature Name</b>
Search in our app „, search by name of film
Search in this app , by category or browse by age
Sign up
Sign in
Sign in with social networks
Rate movie
Show popular movies by genre
Show coming soon movies
Top rated movies
User history
Movie info
Notification
Home in this app “ by more than category (not movie only)”
Home in our app “ trending movie „Recent movie , top rated”
Edit profile
View user reviews
Recommendation

Table 4: Similar features between Common Sense and us

<b>Different features</b>	<b>Watch &amp; Rate</b>	<b>Common sense media</b>	<b>Reason</b>
Play trailer	Yes	No	
Critics reviews for movie	Yes	No	
Voice search	No	Yes	Future work
Watch list	No	Yes	Future work

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



Share movie	No	Yes	Because app is mainly for rating & movie reviews
User score	Yes	No	
Rate Review	Yes	No	
Write Review	Yes	No	
Specify age	No	Yes	We use application for all ages and for all types of movie
Comment on Review	Yes	No	

Table 5: Different features between Common Sense and us

### 2.1.3 New Movies Review

Feature Name
Search in our app ,, search by name of film
Sign up
Sign in
Sign in with social networks
Rate movie
Show popular movies by genre
Show coming soon movies
Top rated movies
User history

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



Movie info
Home in this app “ in theaters ,opening , upcoming , wish list”
Home in our app “ trending movie „Recent movie , top rated”
Edit profile
Play trailer

Table 6: Similar features between NMR and us

Different features	Watch & Rate	IMDB	Reason
Watch list	No	Yes	Future work
Critics Review	From NYT API	From lots of papers	NYT provide free API to get movie reviews
Share movie	No	Yes	Because app is mainly for rating & movie reviews
User score	Yes	No	
Rate Review	Yes	No	
Write Review	Yes	No	
Recommendation	Yes	No	
Comment on Review	Yes	No	

Table 7: Different features between NMR and us



#### 2.1.4 Rotten Tomatoes

<b>Feature Name</b>
Search
Sign up
Sign in
Sign in with social networks
Show trending movies
Rate movie through users
Show popular movies by genre
Play trailer
Show coming soon movies
Top rated movies
User history
Movie info
Write review

Table 8: Similar features between Rotten Tomatoes and us

<b>Different features</b>	<b>Watch &amp; Rate</b>	<b>Rotten Tomatoes</b>	<b>Reason</b>
US Box office	No	Yes	Because app is mainly for rating & movie reviews
Gallery	No	Yes	App core does not need it

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



Tickets and show times	No	Yes	Application is currently for rating and reviewing movies
Show all available tickets/show times	No	Yes	Because app is mainly for rating & movie reviews
Top critics' reviews	From New York Times only	From lots of different papers	NYT provide free movie review API to use
Show awards central	No	Yes	Because app is mainly for regular people opinions about movie
Voice search	No	Yes	Future work
Watch list	No	Yes	Future work
Link to buy movie	No	Yes	App is not commercial
Share movie	No	Yes	Because app is mainly for rating & movie reviews
Notification	Yes	No	
User score	Yes	No	
Rate Review	Yes	No	
Comment on Review	Yes	No	
Edit Profile	Yes	No	

Table 9: Different features between Rotten Tomatoes and us



## Chapter 3: System Model

In this chapter Watch and Rate's design and analysis is introduced including the functional and non-functional requirements, class diagram, use case description and diagram, sequence diagrams, entity relationship diagram and system's test cases.

### 3.1 Project Specifications

#### 3.1.1 Introduction

The Watch and Rate system is designed to help people who are interested in movies, their news, movies' actors, and reviewers' opinions about these movies. The Watch and Rate system helps its users to search for movie rating, reviews, and regular users' reviews to make a clear opinion about a certain movie. We use the following APIs: IMDb's rating, Rotten Tomatoes' rating and New York Times paper's critics' to retrieve movies' reviews.

#### 3.1.2 Purpose

The Watch and Rate system is designed to help people who are interested in movies, their news, movies' actors, and reviewers' opinions about these movies. The Watch and Rate system helps its users to search for movie rating, reviews, and regular users' reviews to make a clear opinion about a certain movie. We use the following APIs: IMDb's rating, Rotten Tomatoes' rating and New York Times paper's critics' to retrieve movies' reviews.

The main goals of Watch and Rate system is

- Allow users to check movies' reviews especially when they would like to new movie.
- Allow users to receive movie news, reviews and updates.
- Allow users to Share bad, good, and satisfying movies experiences and discuss-them with others.
- Allow users to Search for movies by its title and get movie's info, reviews, users' reviews, IMDb ratting, Rotten Tomatoes ratting and our user's rating.



- Allow users to Make Recommendations according to their favorite movie categories.

### 3.1.3. How it works

It's a web service using android as a frontend and a server/service as a backend where the user can search for a movie to get its reviews. We're making a review system for movies where any user can write his/her opinion about a certain movie or search for reviews related to a movie.

We have online server and push our application to it and online database to access the application from any smart android phone/ tablet

After users sign up and login, they can use application with all features

If user doesn't make an account, he only can search about movie and see the recently, recommended and higher review movies

**We use OpenShift (PaaS by Red Hat) as a main server for our application**

## 3.2 Application Architecture

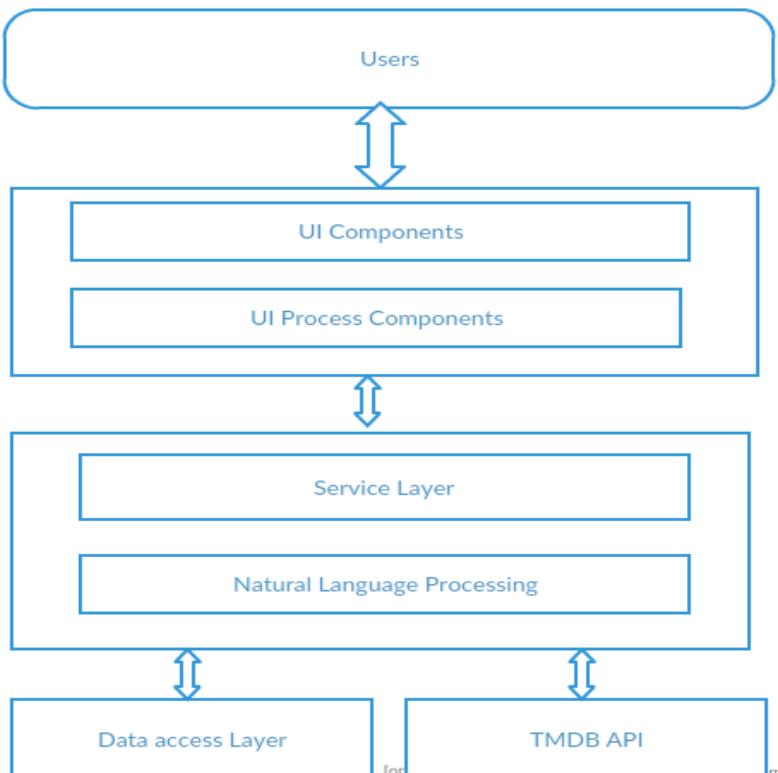


Figure 1: application architecture



---

Movie Review Application is a restful android application consumes web services.

### 3.2.1 REST

REST stands for Representational State Transfer. It is an architectural style that does not require the client to know anything about the structure of the API. But the server and the client must agree to a way of communication which is accomplished through request/response as the client request a certain service through the URL then the server specifies the location of the resources and the required fields to process the request and send back to the client a response.

Web services that are defined on the server can be accessed through many different interfaces, in our case, it's an android application that a user can request any service and get a response.

Example request/response for Watch and Rate application:

**Request:** this is a request to write a comment on a review

```
http://watchandrate-fcigp.rhcloud.com/Write_comment?data={"content":"This is a good movie","hasurl":0,"userid":3,"revid":12}
```

**Response:** json response

```
{"status":"inserted"}
```

**Request:** this is a request to get comment notification to show to a user

```
http://watchandrate-fcigp.rhcloud.com/comment_notification?data={"userid":2}
```

**Response:** json response



```
{"results": [{"movie_id": 4771, "movie_name": "Gone Baby Gone", "username": "hosam", "sender_id": 3, "review_title": "Will Keep You Debating On What Is Truly Right\\n\\n", "commnet_id": 23, "notification_id": 5, "reciever_id": 13, "userimage": "3facepic13.png", "rev_id": 22}, {"movie_id": 4771, "movie_name": "Gone Baby Gone", "username": "Ess", "sender_id": 9, "review_title": "Will Keep You Debating On What Is Truly Right\\n\\n", "commnet_id": 24, "notification_id": 6, "reciever_id": 13, "userimage": "9received_956006441112515.jpeg", "rev_id": 22}], "status": "commentNotification"}
```

### 3.2.2 Service Layer and Database Layer

MySQL database is used to save the data of the application which interacts with service layer through tomcat.

### 3.2.3 Review sentiment Analysis

Sentiment analysis or opinion mining refers to the use of natural language processing and computational linguistics to extract subjectivity from given source material; in our case it's applied to movie reviews

#### 3.2.3.1 Annotation phase

Reviews from IMDB were annotated using sentence level annotation. A sentence could have one of two classes either a positive class or a negative class. Part of our data was annotated manually and another part we got from open data source that had some positive and negative sentences for movies. Yellow color used in annotation manually to annotate positive sentences and red color was used to annotate negative sentences.

# Graduation Project

## Project: Watch and Rate

### Software Documentation



83. Just freakin' awful



Author: [bernie-122](#) from Sydney, Australia

25 June 2006

I am dumbfounded as to how this movie got so many rave reviews from people on this board! I can only suppose it has been swamped by studio supporters, since I see only toward the last pages are the real opinions coming out.

This is about the worst biopic I've ever seen. Phoenix wasn't that bad, but he reminded me of Johnny Cash not at all. Not his looks, not his voice, not his mannerisms, just nothing. I can't say too much about Witherspoon, as I never saw June to make a comparison.

So, the live performances, to which we were treated aplenty, were largely meaningless and, I'm sorry to say, there wasn't much in between them to take our minds off this fact. The story, such as it was, was full of threads that never went anywhere and it otherwise left us alternately either hanging or belaboured by the obvious. The editor must have been on the same pills Johnny was taking, because it made me feel nervous and imitable.

However, there is a saving grace for this film and that is the second story of Ira and Ruth. The tale of the 1940's relationship could have been a movie by itself, truly showing a developing relationship and the troubles they face. This was the type of relationship I like to see, emphasizing care and compassion for the one you want to share your life with it. The gifts were heartfelt, a tribute to the love they truly felt only to be undermined by the struggles they faced. It was real, and an example of the type of relationship I pray many actually get instead of the shallower loves seen in most movies. This relationship was obviously to show how much love has changed, the comparison illustrating the differences between real and superficial love. Obviously this was the meant to drive Sophia and Luke's relationship and overcome their few challenges, but it was the more engaging and emotional of the tales.

Despite the great morals, sweet romance, and good looks though what else does this movie have in store. In terms of positives, the beautiful settings of North Carolina will take your breath away and offer

Figure 2: Annotation for negative and positive sentences

#### 3.2.3.2 Feature Extraction

The second phase was to choose which features can be used to classify a sentence and which feature was important to the support vector machine which was in our case WEKA to train and get a good accuracy to generate a model that can predict.

#### 3.2.3.3 Features chosen

1. Number of positive words in a sentence. A word is considered positive if it has a positive weight in SentiWordNet greater than its negative weight or a negative word that was preceded by negation.
2. Number of negative words in a sentence. A word is considered negative if its negative weight is greater than its positive weight in SentiWordNet or a positive word that was preceded by negation.



- 
3. Positive score of a sentence is the summation of all positive weights of all the words in a sentence.
  4. Negative score of a sentence is the summation of all negative weights of all the words in a sentence.
  5. Number of strong positive in a sentence. A word is considered strong positive if it's positive and the word is preceded by intensifier or its positive weight is greater than 0.5.
  6. Number of strong negative in a sentence. A word is considered strong negative if it's negative and the word is preceded by intensifier or its negative weight is greater than 0.5.
  7. Number of adjectives in the sentence.
  8. Number of adverbs in the sentence.
  9. Number of subjective words.
  10. Number of neutral words.

#### 3.2.3.4 Tools

##### 1. SentiWordNet

SentiWordNet is lexical resource for opinion mining used to determine if a word is positive or negative by reading the sentiword file and initializing Hash Map for the words in it. It's used to get most of the features – positive\_words, negative\_words, strong\_positive, Strong\_negative, positive\_score, negative\_score-

SentiWordNet consists of words – synset terms- and assign to each term posScore –positive score- and negScore –negative score-

```
# SentiWordNet v3.0.0 (1 June 2010)
# Copyright 2010 ISTI-CNR.
# All right reserved.
#
# SentiWordNet is distributed under the Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0) license.
# http://creativecommons.org/licenses/by-sa/3.0/
#
# For any information about SentiWordNet:
# Web: http://sentivordnet.isti.cnr.it
# -----
#
# Data format.
#
# SentiWordNet v3.0 is based on WordNet version 3.0.
# WordNet website: http://wordnet.princeton.edu/
#
# The pair (POS, ID) uniquely identifies a WordNet (3.0) synset.
# The values PosScore and NegScore are the positivity and negativity
```

# Graduation Project

## Project: Watch and Rate

### Software Documentation



```

# score assigned by SentiWordNet to the synset.
# The objectivity score can be calculated as:
# ObjScore = 1 - (PosScore + NegScore)
# SynsetTerms column reports the terms, with sense number, belonging
# to the synset (separated by spaces).

# -----
# POS ID PosScore NegScore SynsetTerms Gloss
a 00001740 0.125 0 able#1 (usually followed by 'to') having the necessary means or skill or r
a 00002098 0 0.75 unable#1 (usually followed by 'to') not having the necessary means or s
a 00002312 0 0 dorsal#2 abaxial#1 facing away from the axis of an organ or organism; "the ab
a 00002527 0 0 ventral#2 adaxial#1 nearest to or facing toward the axis of an organ or organis
a 00002730 0 0 acroscopic#1 facing or on the side toward the apex
a 00002843 0 0 basiscopic#1 facing or on the side toward the base
a 00002956 0 0 abducting#1 abducent#1 especially of muscles; drawing away from the midline of
a 00003131 0 0 adductive#1 adducting#1 adducent#1 especially of muscles; bringing together o
a 00003356 0 0 nascent#1 being born or beginning; "the nascent chicks"; "a nascent insurgenc
a 00003553 0 0 emergind#2 emergent#2 coming into existence; "an emergent republic"

```

Figure 3: SentiWordNet lexicon

## 2. Stanford POSTagger

POSTagger stands for Part-Of-Speech Tagger is a piece of software that reads text in some language and assigns part of speech to each word such as noun, verb, adjective, etc.

POSTagger is used to know the number of adjectives (JJ) and the number of adverbs (RB) in a sentence.

## 3. Stanford Semantic Parser

Semantic Parser that maps natural language sentences to representational form of their meaning. Parser is used to detect negation relation that changes the meaning of a word. If a word is positive, after negation it's considered negative and vice versa.

### Basic Dependencies:

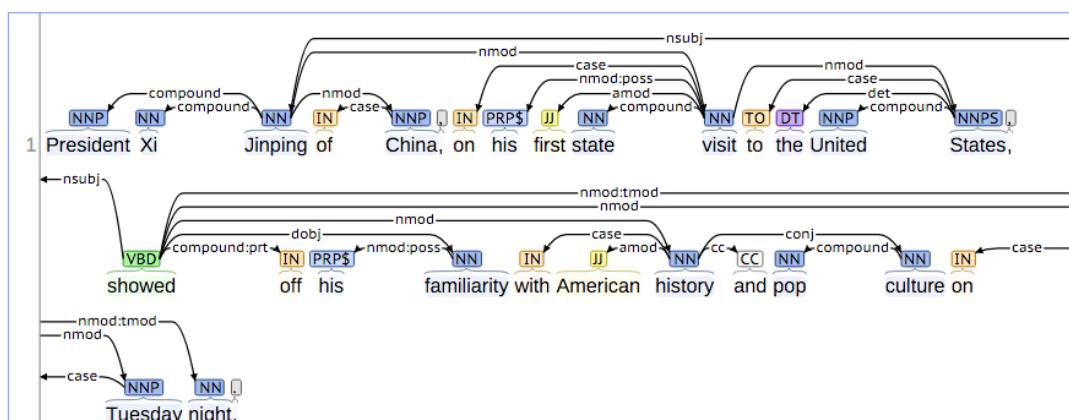


Figure 4: Basic dependencies Stanford parser



#### 4. WEKA

It is a powerful machine learning tool written in java that is used for classification and regression. It is a collection of visualization tools and algorithms for data analysis and predictive modeling.

WEKA is used in the project to be trained to classify sentences to positive or negative.

This is a picture of the .arff file used by WEKA that was generated by java application code.

```
2 @attribute 'positive_words' numeric
3 @attribute 'negative_words' numeric
4 @attribute 'positive_score' numeric
5 @attribute 'negative_score' numeric
6 @attribute 'strongPositive' numeric
7 @attribute 'strongNegative' numeric
8 @attribute 'subjective_words' numeric
9 @attribute 'neutral_words' numeric
10 @attribute 'adj_words' numeric
11 @attribute 'adv_words' numeric
12 @attribute 'class' {negative, positive}
13 @data
14 9,0,1.125,0.75,1,1,11,23,4,1,positive
15 9,0,0.0,0.0,0,0,9,30,4,4,positive
16 0,0,0.0,0.0,0,0,0,4,2,0,positive
17 5,2,0.875,0.75,1,0,8,13,1,1,positive
18 8,0,0.125,0.0,0,0,0,8,13,2,3,positive
19 1,8,0.875,1.25,1,1,11,15,3,1,positive
20 2,0,0.0,0.0,0,0,2,7,1,0,positive
21 1,4,0.75,1.25,1,2,8,12,2,3,positive
22 0,8,0.0,1.375,0,2,10,18,2,2,positive
23 6,0,0.5,0.0,0,0,6,8,2,1,positive
24 6,0,0.0,0.0,0,0,6,9,1,1,positive
25 3,0,0.375,0,0,0,0,3,10,2,1,positive
26 6,0,0.25,0.25,0,0,6,19,0,5,positive
27 2,0,0.0,0.0,0,0,2,8,3,1,positive
28 1,3,0.125,0.25,0,0,4,8,0,3,positive
29 4,0,0.375,0,0,0,0,4,18,2,3,positive
30 0,2,0.25,0.375,0,0,2,4,1,1,positive
```

Figure 5: Training data for WEKA

After WEKA is trained and gets a good accuracy on the train data. A model is generated that is used to predict if the sentences of a review is positive or negative and based on it the review is given a star rate in the range [1,5]

This is the final model used in our application with accuracy 94.5%

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



```
Classifier output
Correctly Classified Instances      10084          94.5789 %
Incorrectly Classified Instances    578           5.4211 %
Kappa statistic                   0.8916
Mean absolute error               0.071
Root mean squared error          0.1946
Relative absolute error          14.1973 %
Root relative squared error     38.9144 %
Total Number of Instances        10662

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area
      0.941     0.05      0.95       0.941     0.946      0.989
      0.95      0.059     0.942      0.95      0.946      0.989
Weighted Avg.      0.946     0.054     0.946      0.946     0.946      0.989

==== Confusion Matrix ===

      a      b  <-- classified as
5017  314 |  a = negative
 264 5067 |  b = positive
```

Figure 6: WEKA Generated Model

To predict for a new review input, the review is broken into sentences, each sentence gets its features from the code –number of positive words, number of strong positive, etc. - and the generated model would predict based on these features.

### 3.3 Planning

# Graduation Project

## Project: Watch and Rate

### Software Documentation



<b>Phase 1</b>		23d	09/10/15	10/02/15
Specifying Idea		3d	09/10/15	09/12/15
Researching		5d	09/13/15	09/17/15
Analysis		4d	09/18/15	09/21/15
Think About Features		5d	09/22/15	09/26/15
Specify Scope		3d	09/27/15	09/29/15
Analysis for similar websites		3d	09/30/15	10/02/15
<b>Phase 2</b>		45d	10/03/15	11/16/15
Time Plan		2d	10/03/15	10/04/15
SRS Documentation		3d	10/05/15	10/07/15
Study features in details		3d	10/05/15	10/07/15
<b>Define Database</b>		32d	10/08/15	11/08/15
Doing Database		4d	10/08/15	10/11/15
Enhance		3d	10/12/15	10/14/15
Documentation		2d	10/15/15	10/16/15
<b>UI part 1</b>		23d	10/17/15	11/08/15
Doing UI		21d	10/17/15	11/06/15
Documentation		2d	11/07/15	11/08/15
<b>Documentation</b>		8d	11/09/15	11/16/15
Mid term		10d	11/17/15	11/26/15
<b>Phase 3</b>		149d	11/30/15	04/26/16
<b>Prototype 1</b>		26d	11/30/15	12/25/15
Services		6d	11/30/15	12/05/15
Development		12d	12/06/15	12/17/15
Integration & Testing		4d	12/18/15	12/21/15
Documentation		4d	12/22/15	12/25/15
First term exam		27d	12/26/15	01/21/16
<b>Prototype 2</b>		22d	02/01/16	02/22/16
Services		5d	02/01/16	02/05/16
Development		10d	02/06/16	02/15/16
Integration & Testing		4d	02/16/16	02/19/16
Documentation		3d	02/20/16	02/22/16
<b>Prototype 3</b>		28d	02/23/16	03/21/16
Services		7d	02/23/16	02/29/16
Development		12d	03/01/16	03/12/16
Integration & Testing		5d	03/13/16	03/17/16
Documentation		4d	03/18/16	03/21/16
Mid term		10d	03/22/16	03/31/16
<b>Prototype 4</b>		22d	04/05/16	04/26/16
Services		5d	04/05/16	04/09/16
Development		10d	04/10/16	04/19/16
Integration & Testing		4d	04/20/16	04/23/16
Documentation		4d	04/23/16	04/26/16
<b>Phase 4</b>		49d	10/5/2016	28/6/2016
Finalizing		9d	10/5/2016	18/5/2016
Second term exam		22d	19/5/2016	9/6/2016
Testing		10d	10/6/2016	20/6/2016
Documentation		7d	21/6/2016	27/6/2016

Figure 7

Time

planning



### 3.4 System Analysis

#### 3.4.1 Feasibility

<b>Feasibility Study</b>	
System:	movie review system to write review and react to other's reviews.
Author:	Hen Team
Product	
	Android Application that is used to interact between movie fans by giving their opinions and interact with each other.
Technical Feasibility	
	For development: Java, Java servlets, WEKA, Tomcat, MySQL, Android
	For deployment: Android devices
Social Feasibility	
	No training for the system needed.
Economic Feasibility	
	The project can be delivered with very little budget.
Market Research	
	This application is beneficial to the market as it's a beginning of social network for movie fans only for movies. Movie fans can interact with each other.

Table 10: Feasibility Study

#### 3.4.2 SWOT Analysis

##### a. Strengths

It's a beginning of a social network for movie fans that can interact, watch or debate over a movie together. Also its



new way of evaluating a movie based on a story, direction or other movie features is new way of evaluation.

**b. Weaknesses**

Interaction among users still needs to have more interacting features. But it's a beginning that can be developed more to have more interactions.

**c. Opportunities**

This application is an opportunity for any movie fan to interact with people with the same movie interests.

**d. Threats**

Non accurate result for the text analysis may drive some users away so it needs to improve the model of text analysis.

## 3.5 Requirements Specification

### 3.5.1 Functional Requirements

**1) Register**

- a. Register to an account using (name, password and email) as any user should determine a name (s)he will use our system through it, password and a unique email.
- b. Register using a social account (Facebook or Google+) and the app retrieve his/her data from that account.

**2) Login**

A user must login to interact with the system

- a. Login using the registered account in our database
- b. Login using social account (Facebook, Google+) which checks if the info matches our database.

**3) Write review**

The user can give his/her opinion/review (whatever positive or negative) on a movie (s)he watched.



Any review should have the following:

- a. Review title.
- b. The body of the review which is free text written in English.
- c. User chooses to enable/disable comments on his review.
- d. User can add image.

#### 4) Comment on review

Any user can reply to a review and can add URL to the comment to interact with the reviewer by supporting his/her opinion or criticizing or suggesting the opinion

#### 5) Rate review

Any review before saving in the system is analyzed so that the system gives it a positive or a negative rating without any user interference.

#### 6) User score

Any user can reply to a review and can add URL to the comment to interact with the reviewer by supporting his/her opinion or criticizing or suggesting the opinion

user score will be a mathematical function to calculate the average of system analysis of the reviews owned by the user and other users' rating for his/her reviews. The mathematical function is the following:

user score =

$$\left( \frac{\text{numberOfPositiveReviewsUsingTextAnalysis}}{\text{totalNumberOfReviews}} \right) / \left( \sum_{i=1}^{\# \text{ofUserReviews}} \frac{\text{numberOfUpvotesOfOneReview}}{\text{totalRatingsOfTheSameReview}} \right)^2$$

Example: if a user wrote 5 reviews and by text analysis we found 3 positive opinions and 2 negative opinions then the first part of addition is 3/5. Other users rated each review owned by that user, consider they all have been up voted by 7



users and down voted by 3 users then we get the summation of all reviews by dividing the up votes number of each review over total number of user rated the same review. Lastly we add both and divide by 2 to get the average.

#### 7) Show higher reviewers

It's based on user score. The user who owns many reviews that are positive reviews or his/her reviews are up voted a lot is considered one of the higher reviewers in the system.

#### 8) Search

Any user can write in the search bar free text then we use text analyzer to know what is (s)he is searching for and extract the reviews related to his/her search. If we couldn't extract enough information and we couldn't get any reviews based on free text we show him/her filters to choose to write (name of the movie, year of production, name of actor/actress, type of the movie or name of director)

#### 9) Recommend movie

The system recommends movie to the user based on TMDb API which has feature discover with some filter applied .

#### 10) Edit profile

A user can change his/her name, password and email.

#### 11) View reviews

A user can view the reviews s/he wrote before.

#### 12) Delete Account

A user can choose to delete the account.

#### 13) Delete comment

A user can delete his/her comment on a review.

#### 14) Delete owned review

A user can delete his/her review.

#### 15) Report a problem



A user can send to the admin a problem by specifying the title of the problem and a description.

#### 16) **Notification**

Any user will be notified in such cases:

- 1) Someone commented on his/her review.
- 2) Someone rated his/her review.

#### 17) **Home Page**

it contains:

- 1) News feed which has the trending reviews and recent reviews in our system that will be shown to the user and can be categorized by the type of movies that (s)he wants to see.
- 2) New notifications.
- 3) Users' with highest scores.
- 4) Search bar.
- 5) Go to profile.

#### 18) **Recently movie**

The system shows recent movies to the user based on the date of movies

#### 19) **Recently Reviews**

The system show recent reviews added to system based on the date of review

### 3.5.2 Non Functional Requirements

- **Quality Requirements**

Quality requirements ensure the system possesses quality working fine according to usability, efficiency, reliability, maintainability and reusability. These requirements *constrain* the design to meet specified levels of quality.



- **Response time**

The system gives results or feedback to the User in a certain minimum time. And this requirement is related to software systems that process a lot of data. The fewer response times the more efficient software system. In our system we need it to be 40 requests every 10 seconds.

- **Throughput**

For systems which have computations or transactions per minute, it's better to the application's throughput to be few minutes as the average of operations per minute. In our system we need it to be 50 transactions/minute for example.

- **Availability**

Is a measure to the amount of time that a server is running and available to respond to users? Application must have a high availability to many users anywhere and anytime.

- **Recovery from failure**

Is the ability of the system to recover the error/failure in few minutes and with a certain minimal loss of data, if a failure is happened to the system so, the application should be able to recover any error in few seconds such as 60 seconds.

- **Allowances for maintainability and enhancement**

In order to achieve this, we should make sure that system can be adapted in the future, and then you should describe the expected changes for the next versions, this constraints design and improves its



quality and improve its performance too, without adding new obvious functional requirements. For example: modifying the movies API later will not effect on the other application parts.

- **Resource usage**

For applications that use a huge amount of memory and network bandwidth, as these are considered as resources, you should specify the maximum amount of each resource that the software system will use. This provides an ability to plan hardware upgrades. For example: one terabyte for memory.

- **Platform requirements**

This type of requirement constrains the environment and technology of the System:

- **Computing platform**

Is to know the hardware and operating system, to know that it will be able to work on it or not, and this is important to protect the social software from breaking down suddenly and specify any powerful or bad performance components. Then we should choose the best operating system and hardware to work on. For example: movies API may not work on Linux, but it may work on windows perfectly.

- **Process requirements**

- **Development process to be used**

To get a high quality application, you should do a lot of development steps which assure to you the quality of the software. Such as particular approaches to testing.



After every testing step you should get the customer opinion to save your time. For example: doing user login and feedback part and testing it on a lot of machines.

## 3.6 System model

### 3.6.1 Use case description

<b>Use case name</b>	<b>Description</b>
Create account	To interact with application ,user has to create an account (if he does not has one) with his/her information such as name ,email ,password ,age ,...etc.
Sign in	User login with his\her email and password after creating an account.
Write a feedback	User can write his negative\positive opinion about any movie in the application database.
Get notification	After signing in the application, user will be able to get any comment on his\her feedback notification or rate notification anytime.
See higher reviewers	User can see the most trusted users' feedbacks as trending feedbacks in home page.
Gain user score	For every feedback the user write, he will get a score for it, for example: +1 for positive feedback, -1 for negative feedback, 0.5 for neutral one.
Rate movie	User can rate any movie in system.

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



Comment on feedback	User can post a comment on any another user feedback.
Search	User can view reviews for any movie in application database by entering movie name in search bar.
View owned reviews	User can view his\her own reviews anytime.
Edit profile	User can edit his profile info: name, email, password, age, city, etc.
Delete account	User can delete his account then his\her own reviews, comments will be deleted also, he\she won't be able to interact with application anymore. Then he\she needs to create new account to interact with application if he wanted to.
Delete owned reviews	User can select any review to be deleted from his\her own saved reviews.
Delete owned comments	User can select any comment to be deleted from his\her own comments on other users reviews.
Recently movie	User can show list of last movies add to API
Recently reviews	User can show list of last review users added to system

Table 11: Use Case Description



### 3.6.2 Use case Diagram

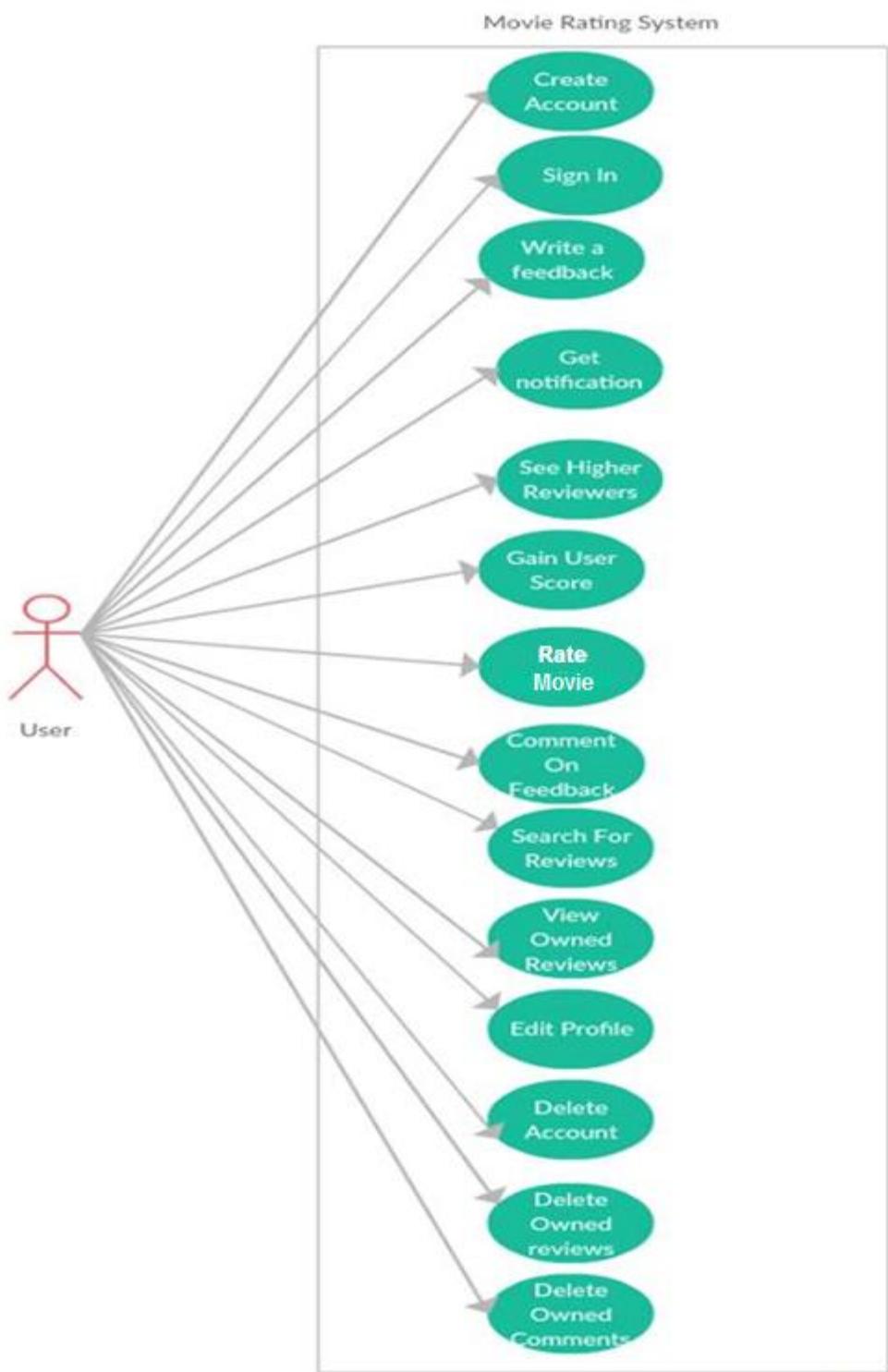


Figure 8: use case diagram



### 3.6.3 Sequence diagrams

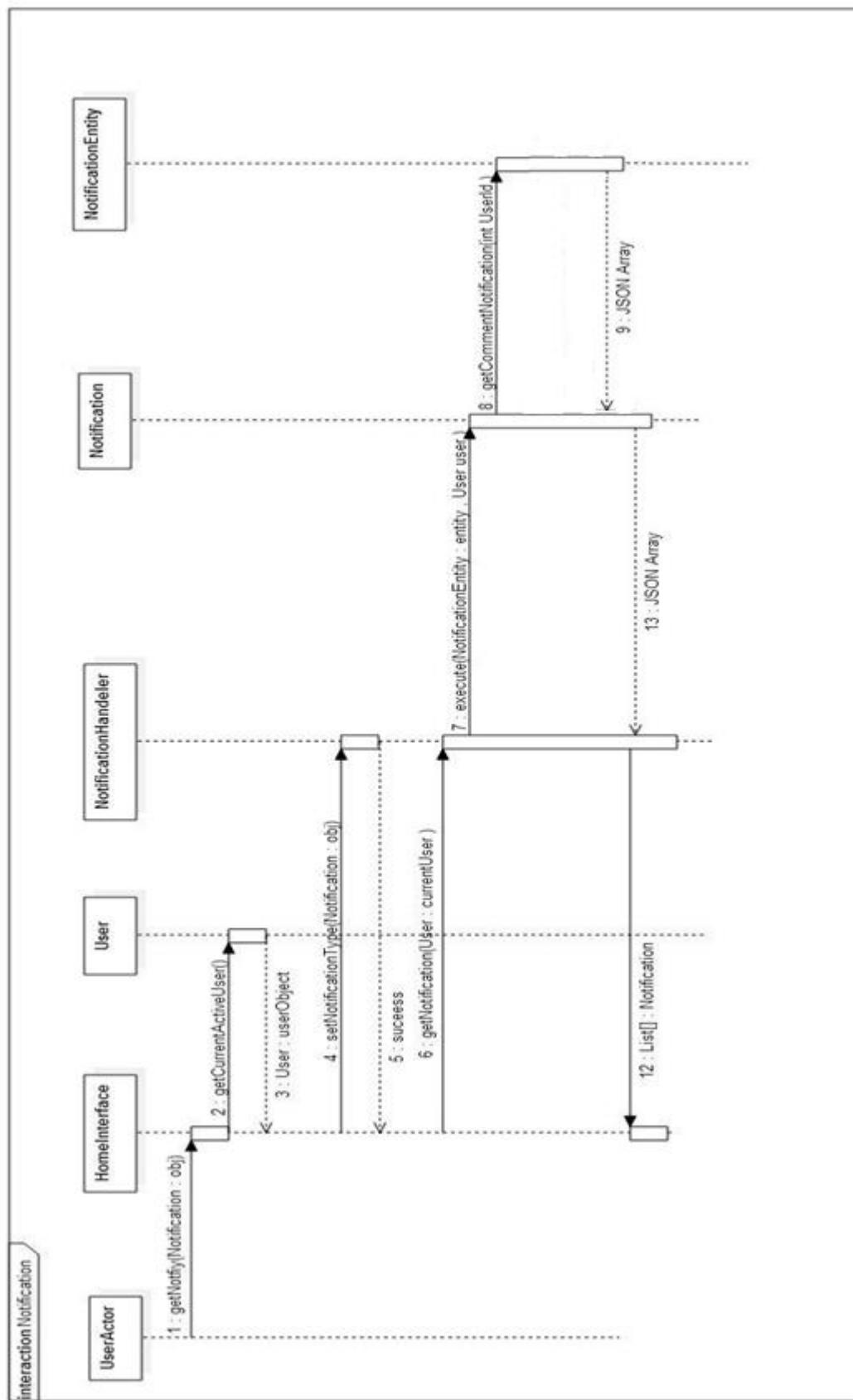


Figure 9: sequence diagram 1

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

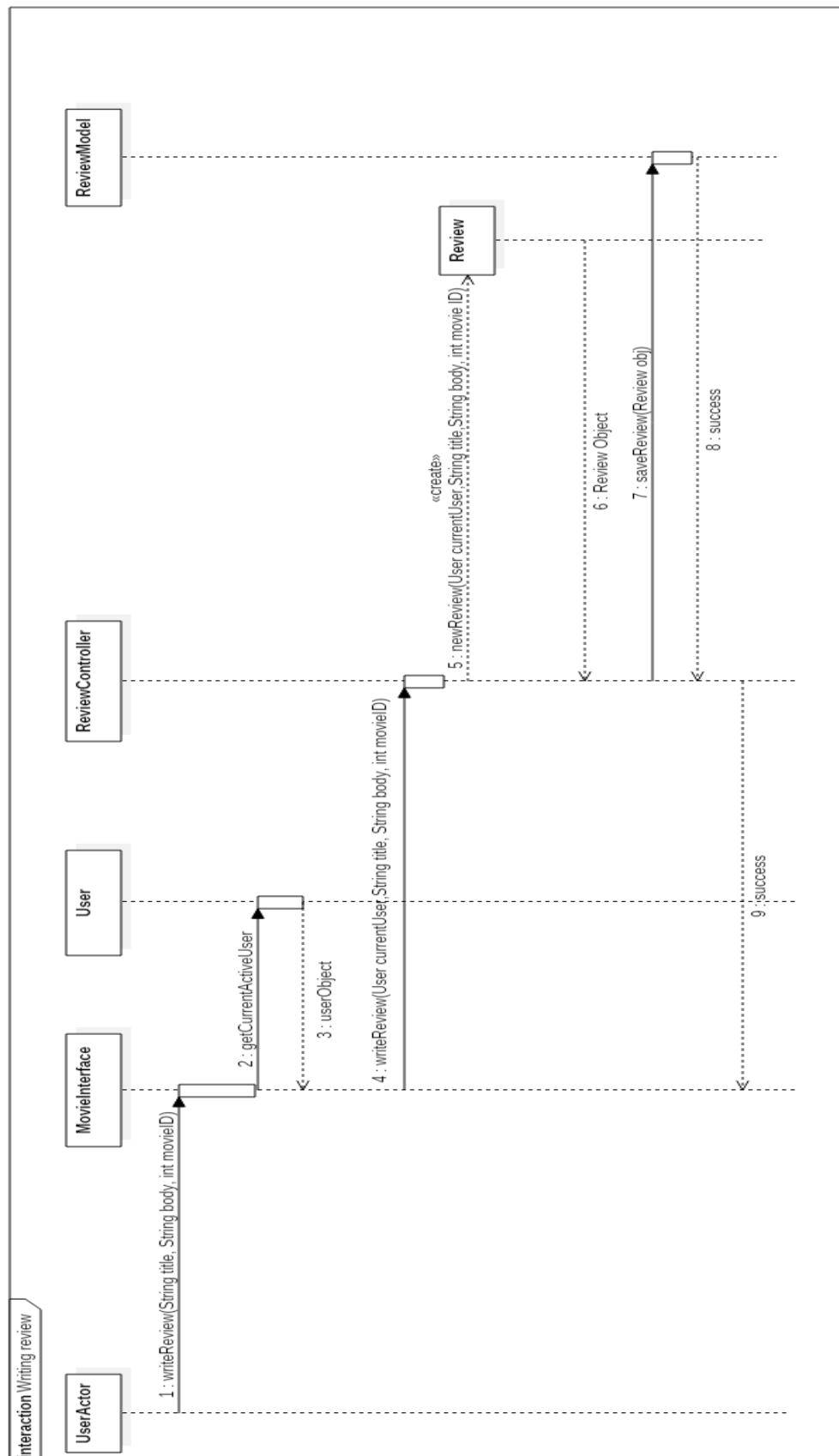


Figure 10: sequence diagram 2

# Graduation Project

## Project: Watch and Rate

### Software Documentation



#### 3.6.4 Class Diagram

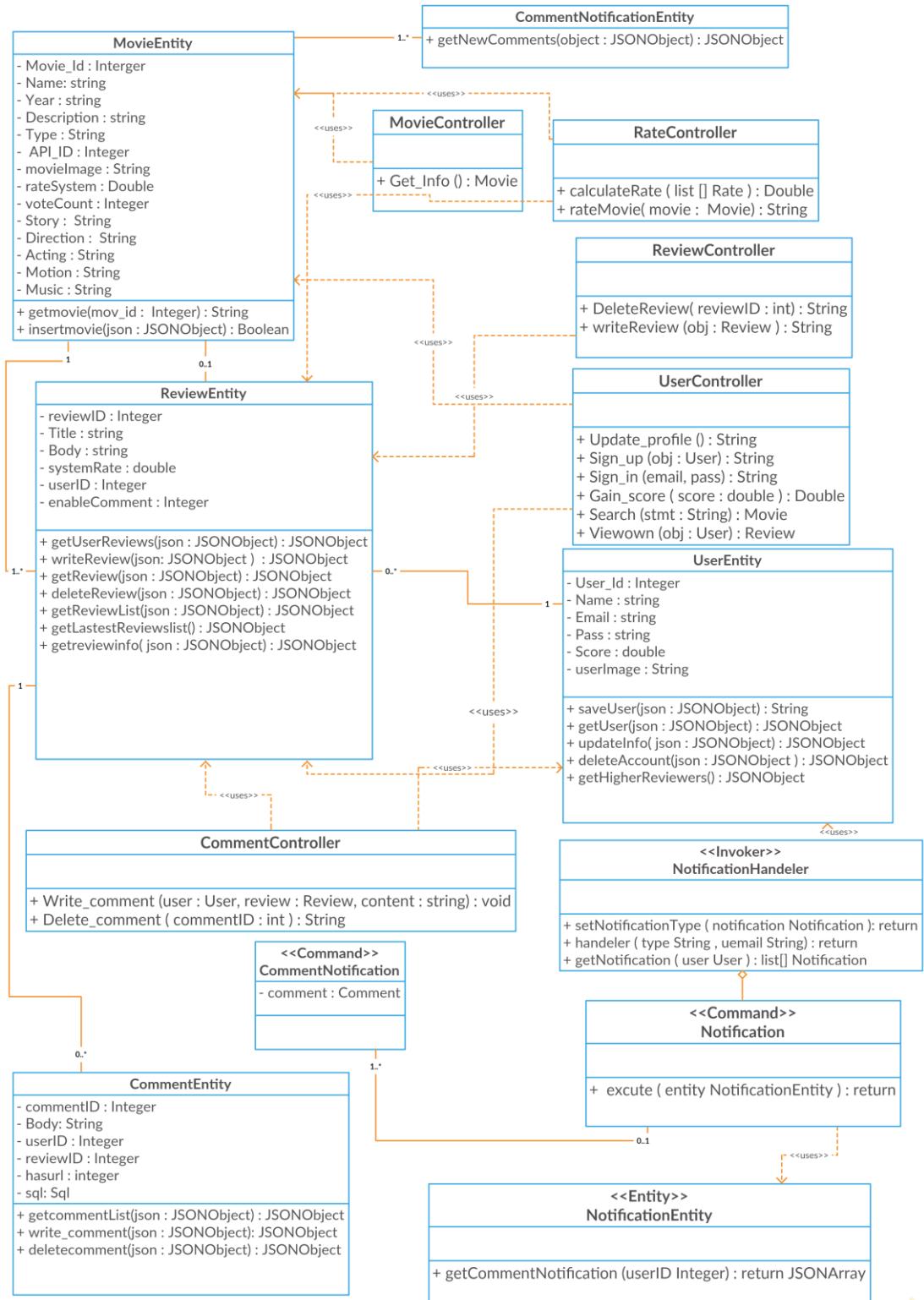


Figure 11: class diagram



### 3.6.5 Entity Relationship Diagram

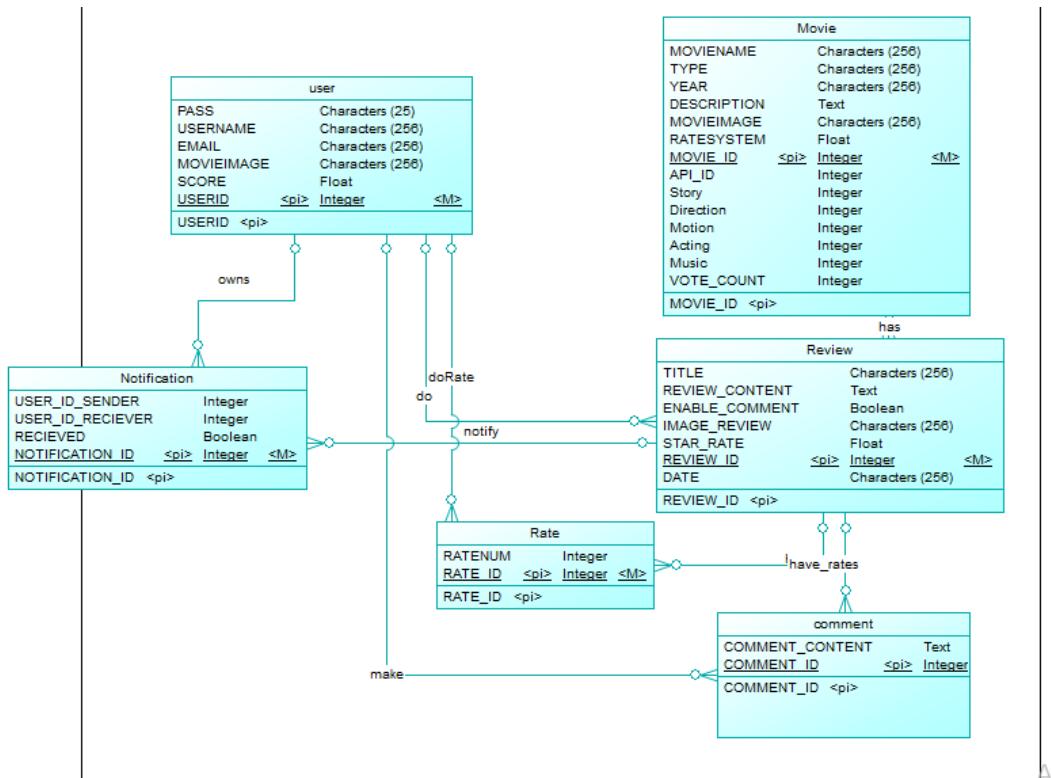


Figure 12: entity relational diagram

### 3.6.6 System test cases

ID	Test Objective	Preconditions	Steps	Test data	Expected result	Post-condition
#1	sign up for first time	1-just open the application 2-choose sign up from home	1- Register to an account using (name, password and email), unique email must be used.	"A valid Name"	1-The user is signed up successfully. 2- New user is added in data base. 3- User can log in.	1-Message to confirm sign up 2-Return to home screen 3-User can now login
			2. Enter the conform Password	"A valid and strong Password"		
			3. Click "sign up" button			

**Graduation Project  
Project: Watch and Rate  
Software Documentation**



#2	log in	1. -User name to login to be available 2. Internet connection.	1. In the login Panel, enter the username	"A valid username"	A message is displayed and the user is logged in to the account	the home page of the application "watch and rate " will appear and now he can use the app
			2. Enter the Password for the User account in the password field	"A valid Password"		
#3	login unsuccessful user	1. A User name to login to be available 2. Internet connection	1. In the login Panel, enter the username	"A valid username"	An Error message is displayed and the user is not logged in to the app, and return to login page again with "error message which is wrong"	No precondition in this case.
			2. Enter the Password for the ESS-User account in the password field	"An invalid Password"		
			3. Click "Login" button			
#4	search for movie	1. User must be logged in with correct email.	1. In the search bar , enter the name of the movie	"A valid name"	the list of reviews about the film which the user entered	List of reviews. User can write feedback , or comment and rate the currently feedback
			3. Click "search" button			
#5	Personal details-modification with valid values- "First Name"	1. User must be logged in with correct email.	Click on the Image displayed at the top left corner of the page	"A valid new value for the first name field" or password or Email or new photo	The users information is displayed	No precondition in this case.

**Graduation Project  
Project: Watch and Rate  
Software Documentation**



			Click on "Choose a file" button			
			Choose an image file of type "JPG"		now show the new value entered	
#6	write feed back	1- must login with a correct Email 2- search for a correct movie	1- write legal feedback about the movie  3. Click "sign up" button	"Not be empty"	1-The feedback is added successfully. 2- New feedback will appear in data base.	1- new review will be added
#7	write comments	1- must login with a correct Email 2- search for a correct movie	1- choose the review which he/she wants to write comment  3. Click "sign up" button	"Not be empty"	1-The comment is added successfully. 2- New comment will appear in data base.	1- new review will be updated
#8	Notifications	1- must login with a correct Email 2- user which will receive notification must write review	1-user comment on you review  "A valid and strong Password"	"comment not be empty"	1-The user will receive notification with the type of notification comment) 2- new notifications is added in data base.	1-user open and see the notification 2-the notification will mark as read
#9	View owned reviews	1. -User name to login to be available 2. Internet connection.	1. Click on "review button"	"no data will insert"	All user's review will appear	user can review his reviews

Table 12: System test cases



## Chapter 4: Snap shots of application

### 4.1 Introduction

In this chapter we will take a closer look to the final design of the application.

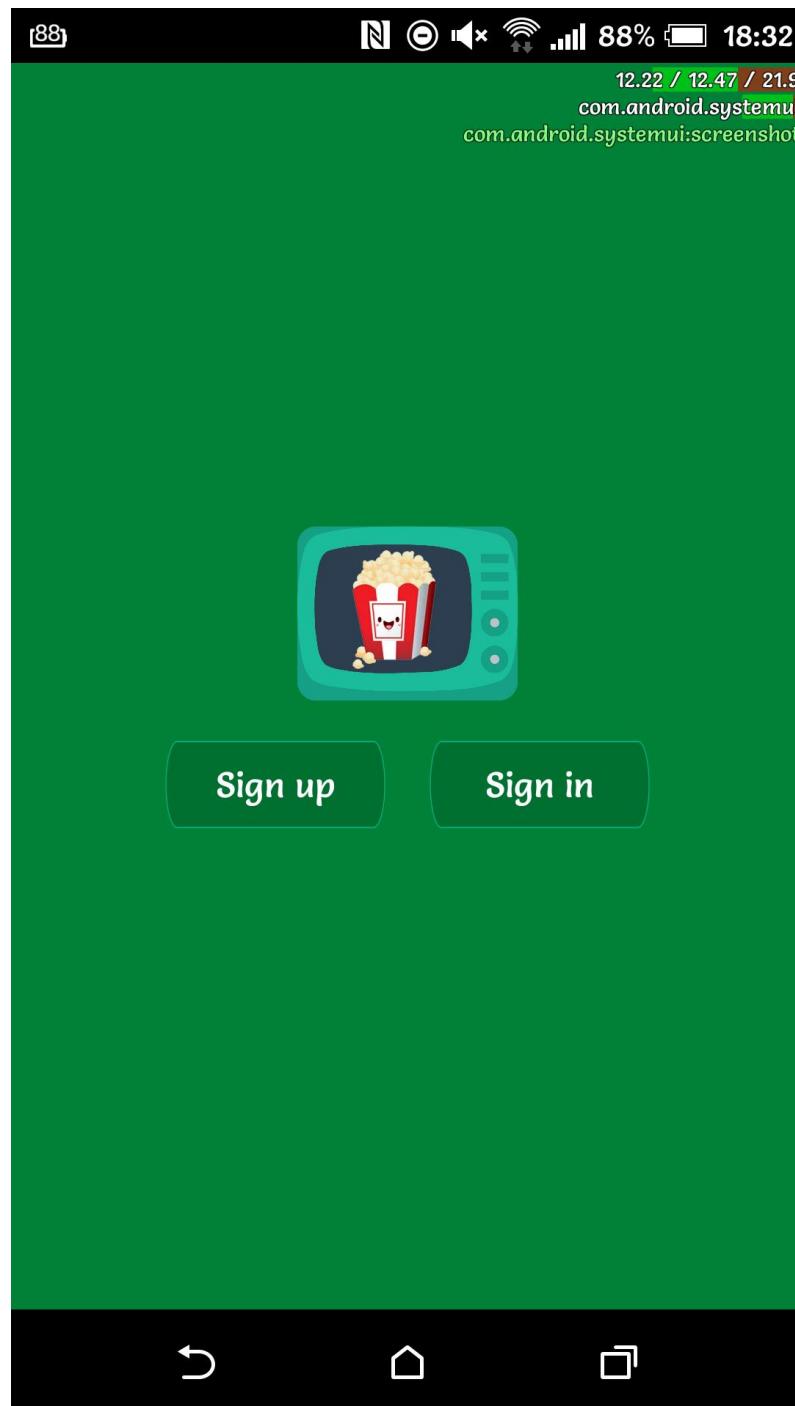


Figure 13: start screen of application

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

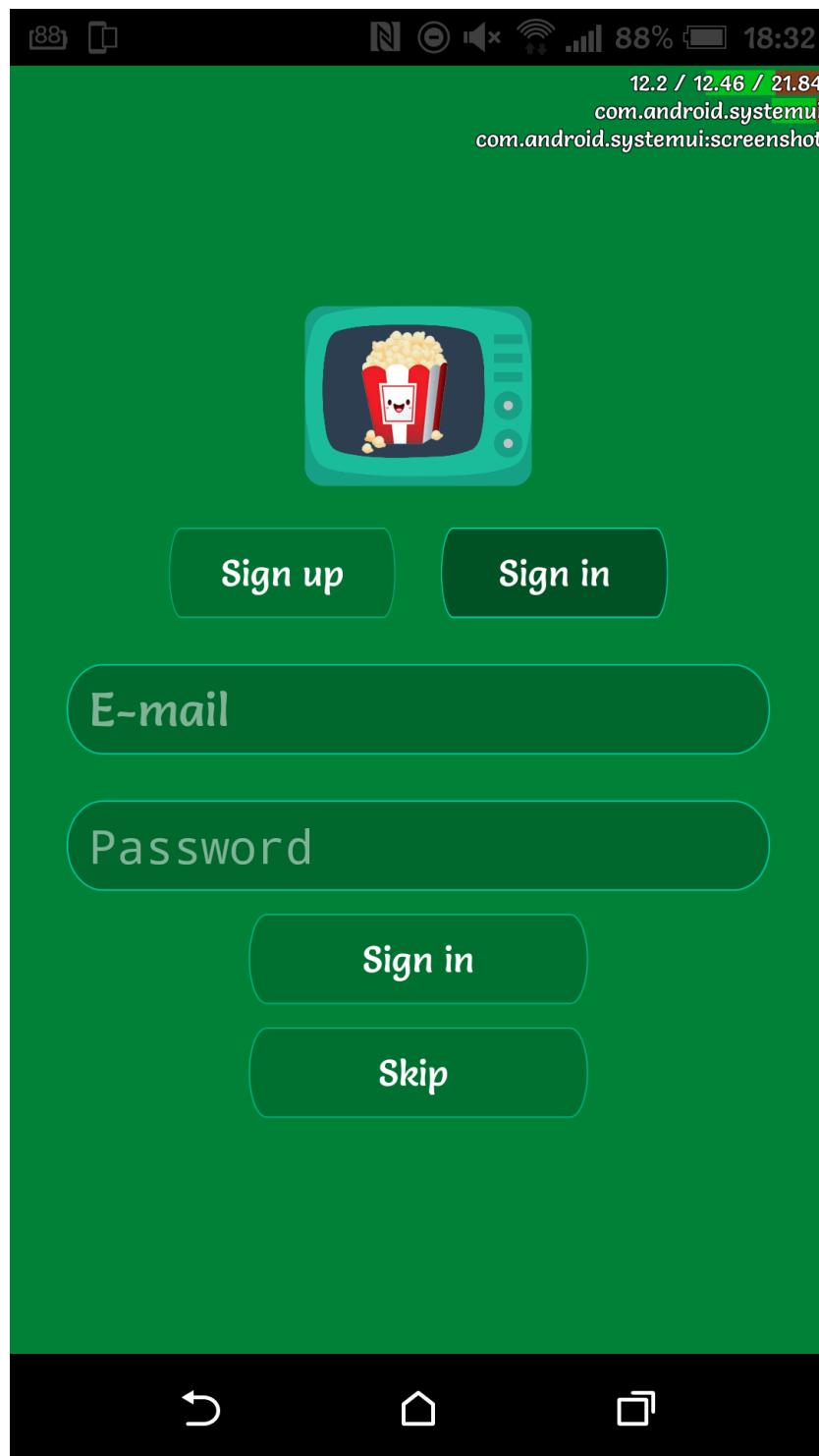


Figure 14: Sign in screen

The sign page of “watch and rate“ if the user has an account he can login

If he/she doesn't have an account and wants to access an application, he can press skip but he doesn't permission to have all our features

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

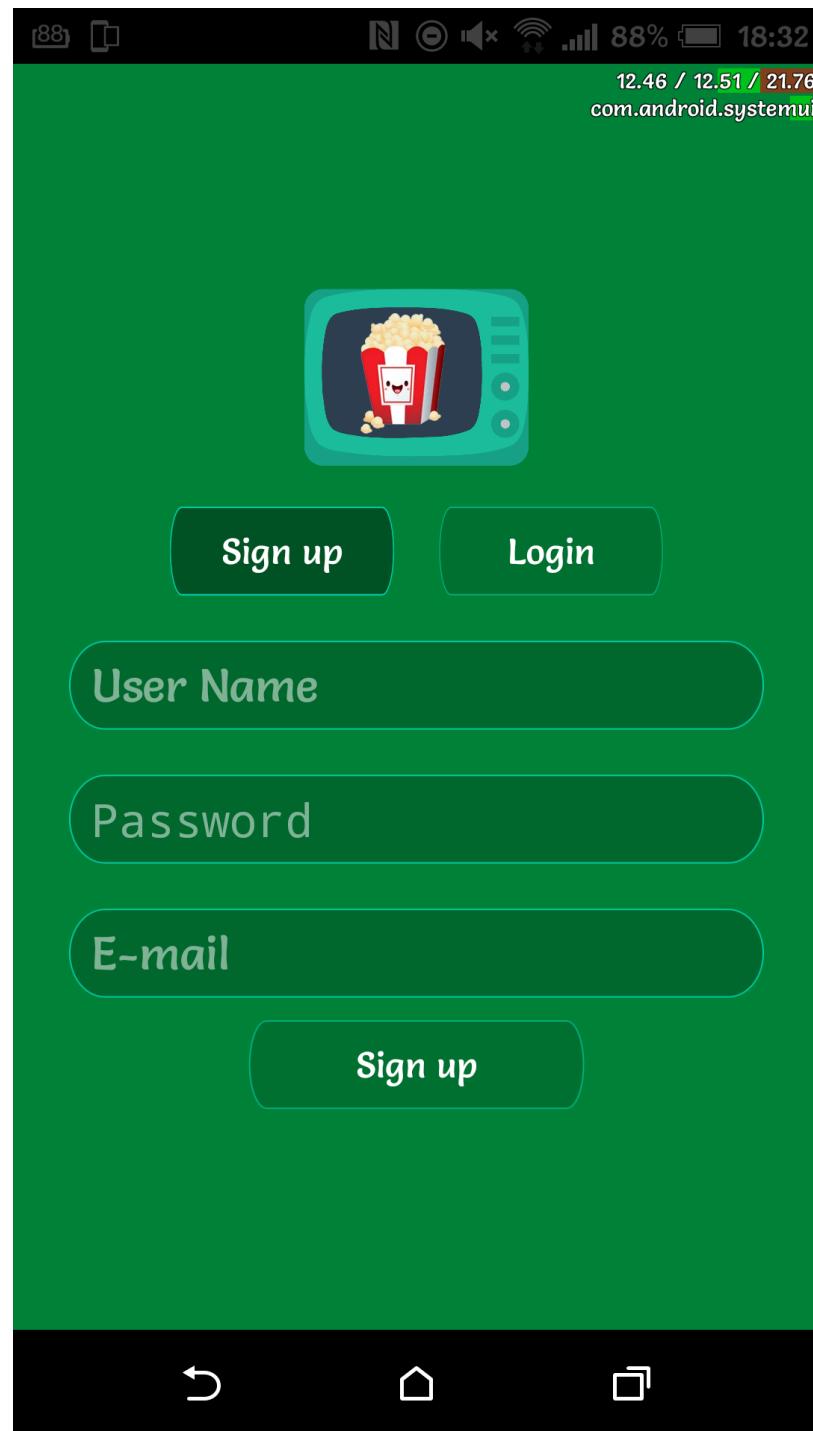


Figure 15: sign up screen

Sign up page, if the user wants to create an account he only need to provide User name / Password / E-mail

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

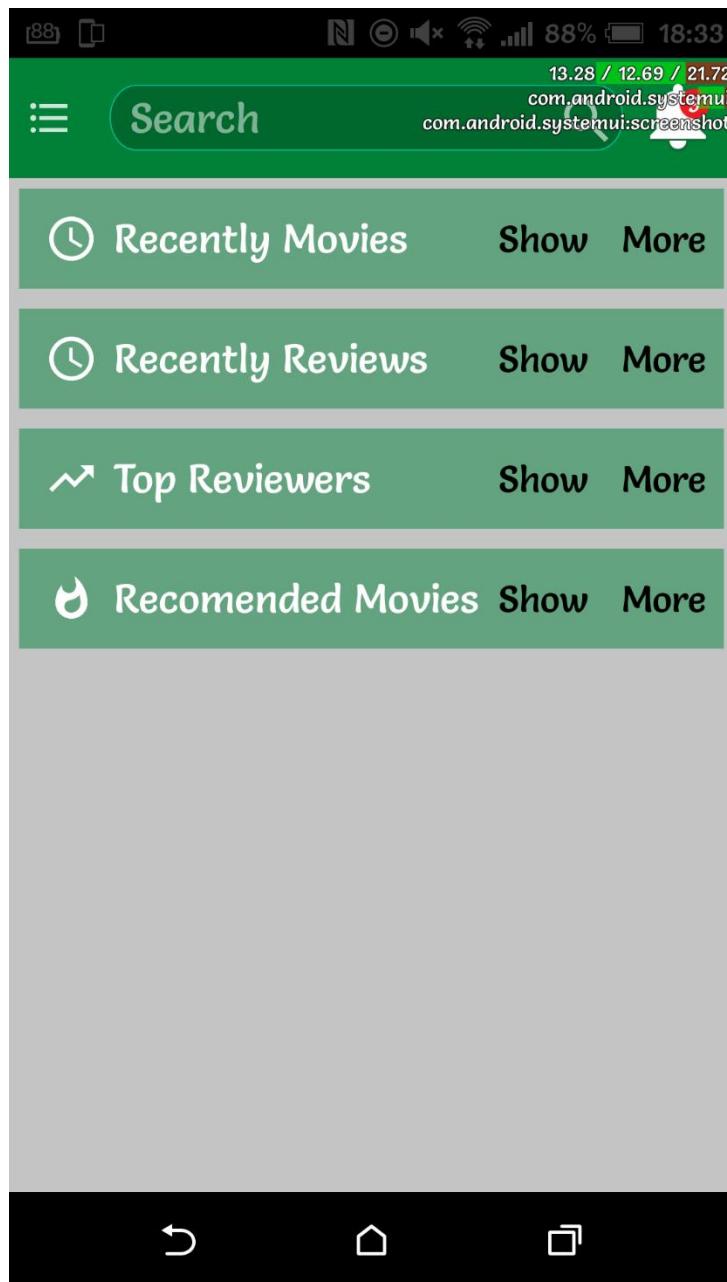


Figure 16: home screen

This is the home of our application and the main page for the application

Which contains

- 1- (Recently Movies / Recommend Movie /Recent Reviewers / top Reviewers)
- 2-Slide panel which have some features, the features will describe later
- 3- Search Bar to search for favorite movie
- 4- notification for user if he already login

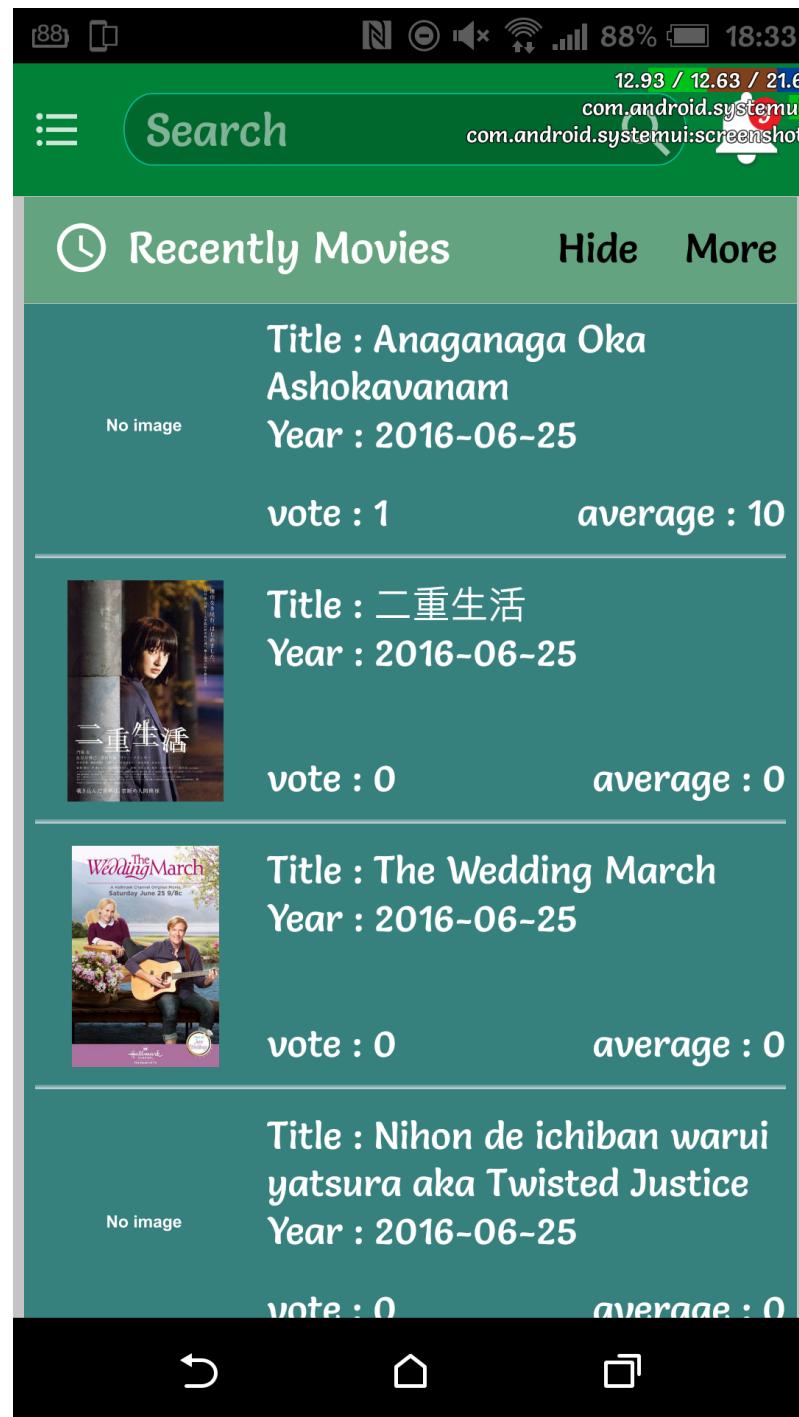


Figure 17: recently movies from home screen

Recently movie header which has 2 button

- 1.button shows the recent 5 movies
- 2.button shows all recent movies



Figure 18: recently movies

Shows recently movies added



Figure 19: recently reviews from home screen

Recently Review header the same concept of recently movies  
And the Same for the 4 header in the home page

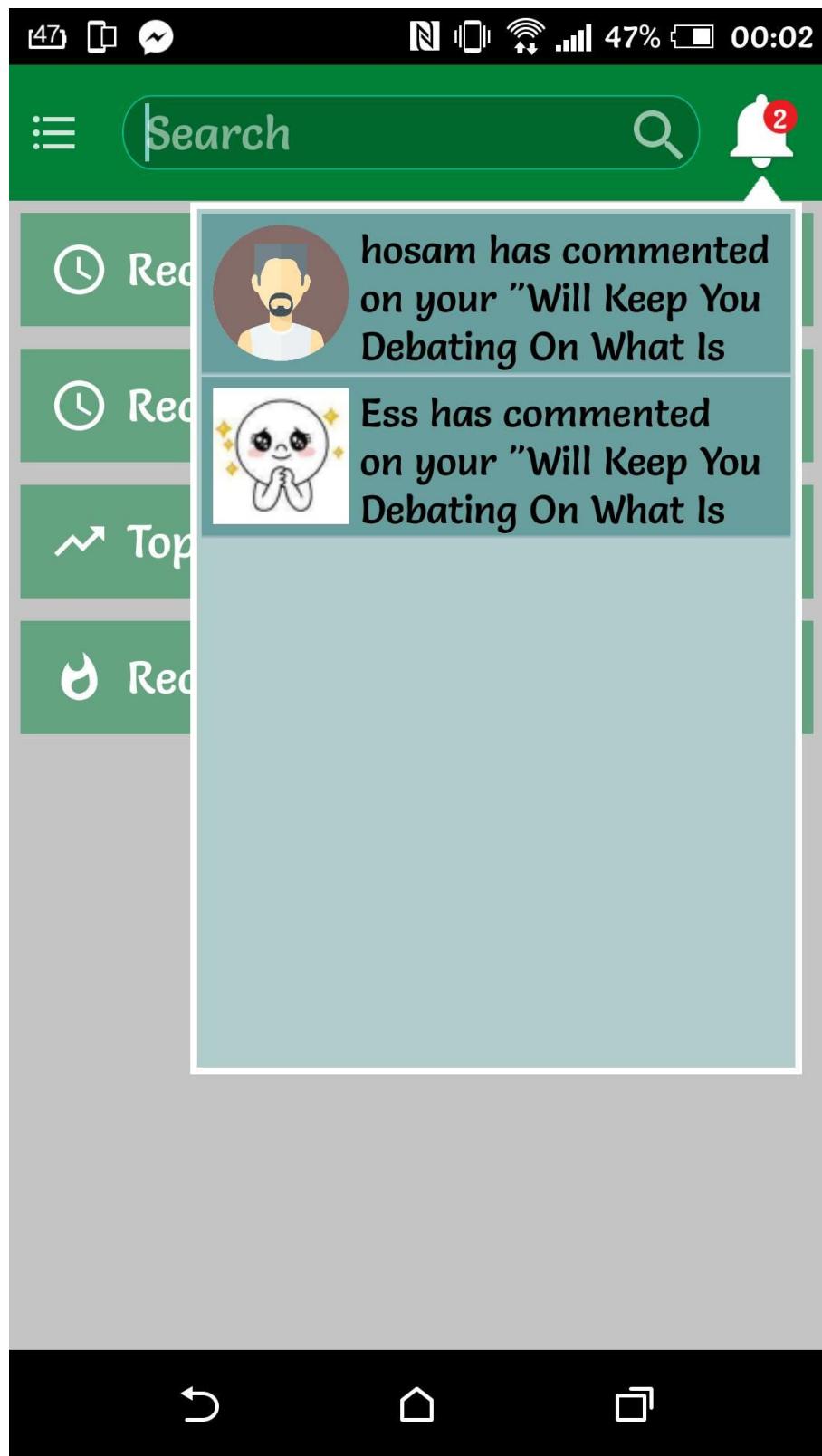


Figure 20: notification screen

Show notification from who comment in your reviews

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



Figure 21: search screen (movies results)

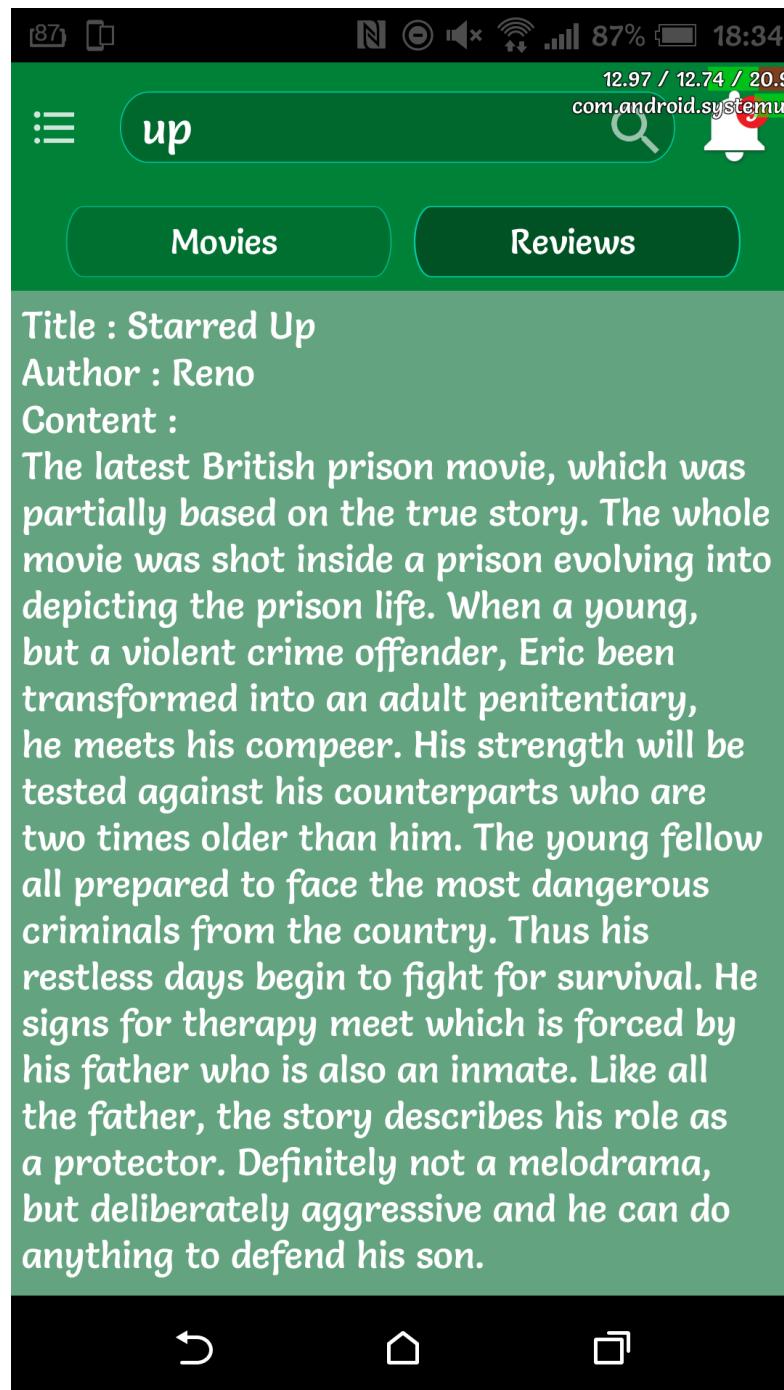


Figure 22: search screen (reviews results)

The search bar, if I want to search for movie: ex “up “

The suggestion movie which contains up word will appear

The 2 button will appear

First button for movies

The second button is for review about movie which come from “nyt API “



Figure 23: movie details screen

The full information about movie will appear As

- 1- Description
- 2- Vote count / average count
- 3- Year
- 4- The new features are special for “watch and rate “which show to the use the main features about movie by features every features have percentage



The screenshot shows a mobile application interface for a movie. At the top, there are several icons: a Facebook icon, signal strength, battery level at 35%, and the time 00:44. Below this is a header bar with three horizontal lines on the left, the text "Details and reviews" in the center, and a plus sign icon on the right. Underneath the header are two rounded rectangular buttons: "Rate" on the left and "Reviews" on the right. The main content area contains the following text:  
**Movie : Gone Baby Gone**  
**Title : Will Keep You Debating On What Is Truly Right**  
  
**Date : 2016-06-25**  
**Author : heba**  
**Description :**  
I had wanted to see this movie for sometime now. I have finally gotten to see it and can tell you it is no disappointment. This movie had a particular touch that gave a genuine and authentic feeling to it.  
  
Ben Affleck has a way of writing about life on the streets in Boston that just grabs you. Good Will Hunting was a fine example of that but Gone Baby Gone displayed his directing abilities as well. This movie appeared that it could fall apart any second but yet the story held up and managed to



Figure 24: movie details reviews

There are 2 important buttons in the bottom page of full information

First button to show rate for movie and allow to user to Rate movie if he already  
Login

Second button to show review about film which made by users of “watch and rate”

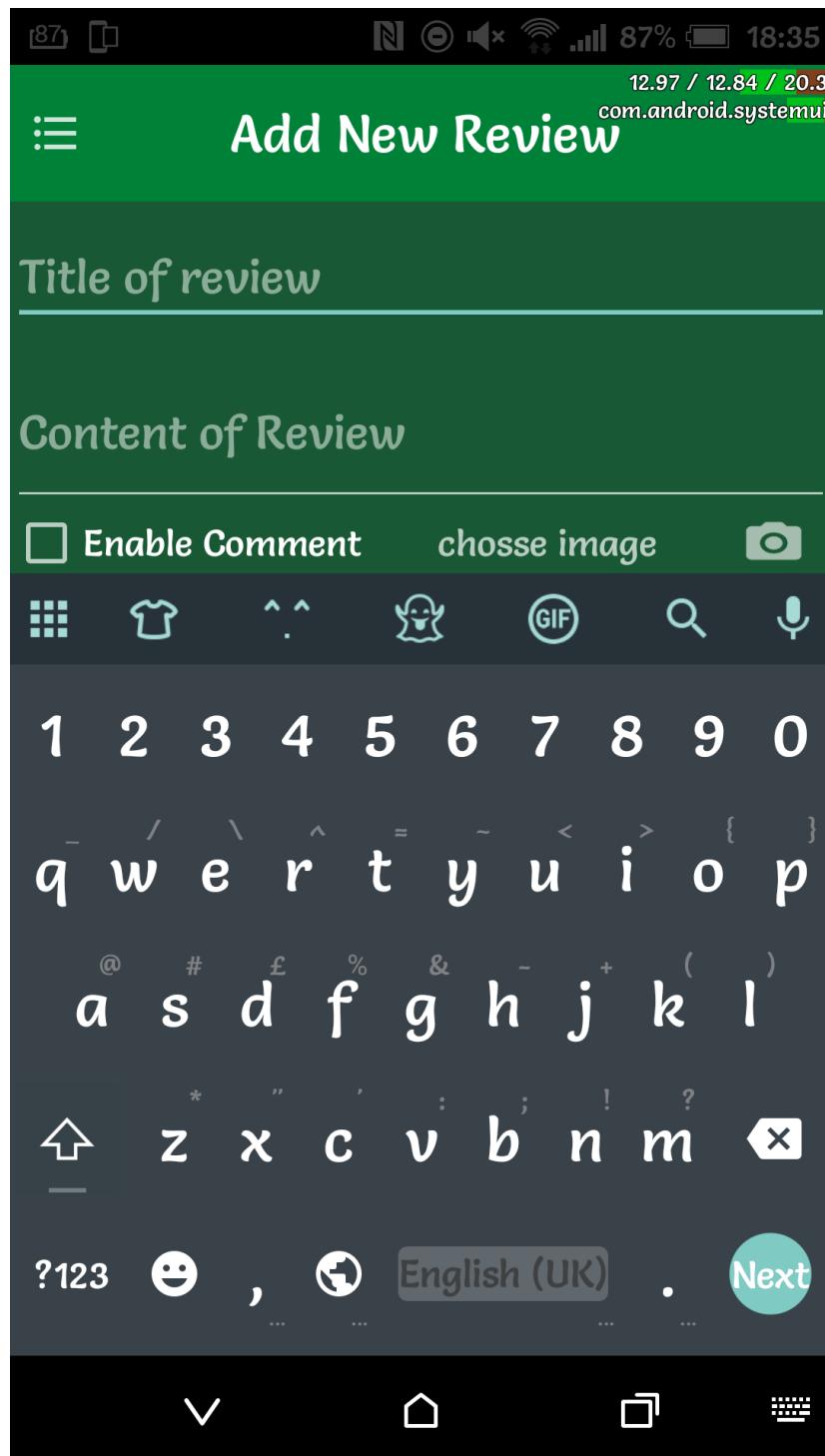


Figure 25: add review screen

Add Review page, this page will appear only for user which is already login

New Review may be having A picture

The user allows to receive comment for his/her review or not

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

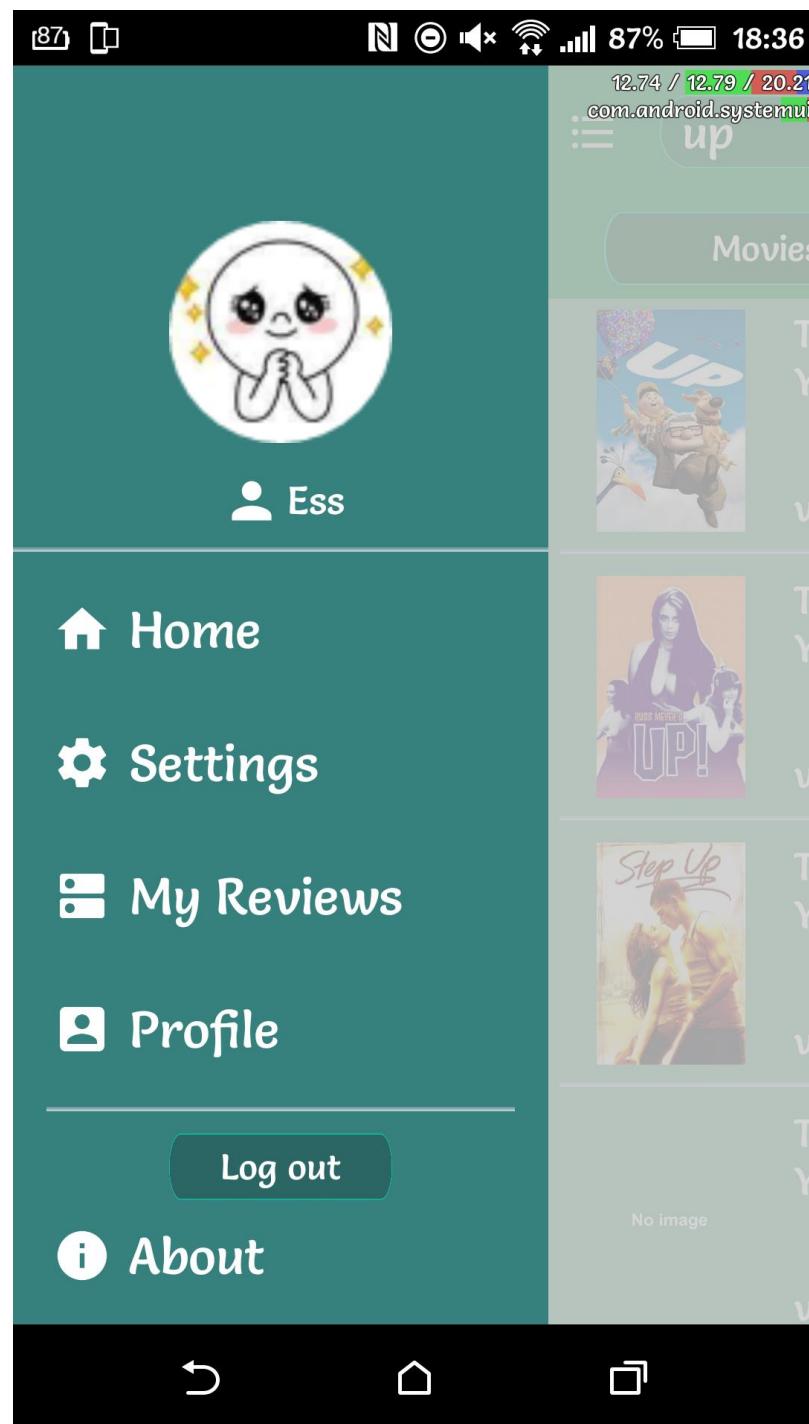


Figure 26: slide panel screen

The slide panel in the home page which contains a personal profile for user/  
The review for user / about our application / setting for application / return to home  
page again



The screenshot shows a mobile application interface with a green header bar. The header contains social media icons (Facebook, Twitter, etc.), signal strength, battery level (31%), and the time (00:56). The main title "My Reviews" is centered in white text. Below the title, there is a review entry for the movie "Gone Baby Gone". The review text reads:  
**Movie : Gone Baby Gone** Edit  Delete   
**Title : Will Keep You Debating On What Is Truly Right**  
  
**Date : 2016-06-25**  
**Description :** I had wanted to see this movie for sometime now. I have finally gotten to see it and can tell you it is no disappointment. This movie had a particular touch that gave a genuine and authentic feeling to it.  
  
Ben Affleck has a way of writing about life on the streets in Boston that just grabs you. Good Will Hunting was a fine example of that but Gone Baby Gone displayed his directing abilities as well. This movie appeared that it could fall apart any second but yet the story held up and managed to get its message out. Ben Affleck seems to be very creative and knows how to get your attention right when he want you to.

Figure 27:user reviews screen

This is the page which shows the all of user reviews for all movies



The screenshot shows a mobile application interface for a movie review. At the top, there's a navigation bar with icons for Facebook, signal strength, battery level (31%), and time (00:56). Below this is a green header bar with the title "Review Details". The main content area has a dark green background with white text. It displays a quote: "Is the right thing to do always the right decision to make and do the ends justify the means? I advise you not to miss this movie and definitely watch this with a friend or family member." Below the quote, there are two comment sections. The first comment is by a user named "hosam", represented by a circular profile picture of a man with a beard. The text reads "greats review ;)" followed by "hosam". The second comment is by a user named "Ess", represented by a circular profile picture of a cartoon character. The text reads "nice go one" followed by "Ess". At the bottom, there's a green input field with the placeholder "Write your comment" and a large black arrow icon pointing to the right. The footer of the screen features three black navigation icons: a circular arrow, a house, and a square.

Figure 28: review details screen

In this page the comments on review will appear if the user enable comment

“this user turns of the comments “

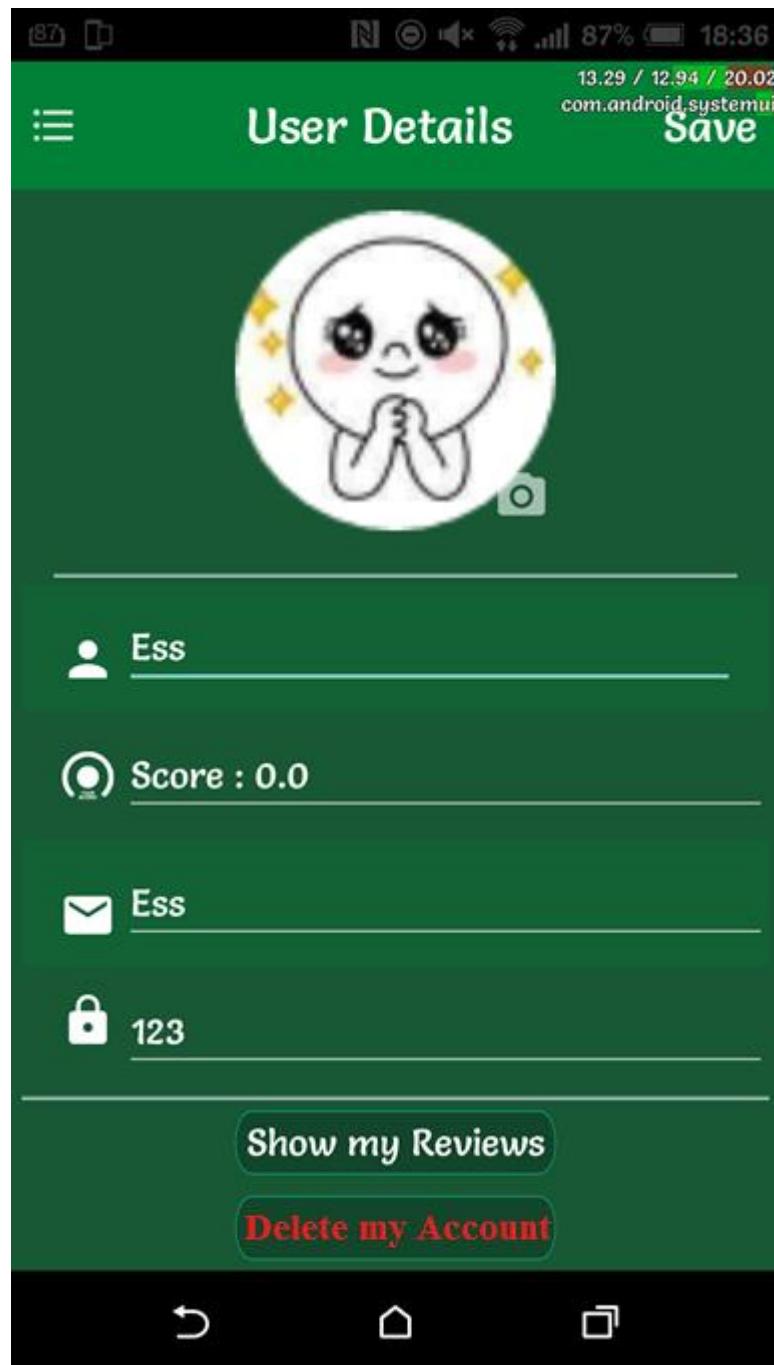


Figure 29: user profile screen

This is personal profile which shows his name / password / E-mail and his score  
Score is beginning at zero and increases one with each review has a rate almost equal to its movie and his profile image

User can change everything except his score

User also can remove his/her accounts



## Chapter 5: Result summary

### 5.1 Result from database

COMMENT_CONTENT	HASURL	COMMENT_ID	USER_ID	REVIEW_ID
no no file w7sh	0	1	2	10
no 7lw :P	0	2	2	10
ummmmm ymkn	1	3	3	10
loooooool xD	1	4	2	10
lool	0	6	3	13
how ,afsh komntat 5als :(	0	7	3	13
loool a5ern	0	8	3	15
lool	0	9	3	16
hey guys	0	10	9	10
@Ess welcome	0	11	2	10
mabrook ^_^	0	12	2	20
hhhhh habl bcomment 3la nafse xD	0	13	2	20
hhhhhhhhh good work	0	14	9	20
lool	0	15	3	10
7elw awy	0	16	4	15
msh awi	0	19	5	12
msh awi	0	20	2	12
wl ya wl	0	21	4	12
w b3deeen	0	22	6	12
greats review ;)	0	23	3	22
nice go one	0	24	9	22
This is a good movie	0	30	2	12

Table 13: comments data table

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



USER_ID_SENDER	USER_ID_RECIEVER	RECIEVED	NOTIFICATION_ID	REVIEW_ID	COMMENT_ID	USER_ID
5	3	1	1	12	19	5
2	3	1	2	12	20	2
4	3	1	3	12	21	4
6	3	1	4	12	22	6
3	13	1	5	22	23	3
9	13	1	6	22	24	9
2	3	1	7	12	30	2

Table 14: notification data table

PASS	USERNAME	EMAIL	USERIMAGE	SCORE	USER_ID
asd	Hosam Azzam	sa	2received_10646	4	2
as	hosam	sa	3facepic13.png	4.5	3
123456	hosam	hos	none	2	4
dsf	islam	gfd	none	3	5
dsf	Ail	gfd	none	2	6
as	as	ss	none	0	7
esso	Safrota	safrota	8facepic5.png	3.5	8
123	Ess	Ess	9received_95600	2.5	9
123	ess	ess	none	0	10
8919661	HassaN	hassanmohmah@hotmail.com	none	0	11
12345	heba	heba@gmail.com	none	0	12
heba	heba	heba@gmail.com	none	0	13

Table 15 user data table

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



MOVIE_NAME	TYPE	YEAR	DESCRIPTION	MOVIE_IMAGE	RATESYSTEM	MOVIE_ID	API_ID	Story	Direction	Acting	Motion	Music	VOTE_COUNT
Home	none	2015-03-18	When Earth is taken over by the overly-confident B...	/nlvfqZkq4nQm	6.87	3	228161	0	0	0	0	0	1235
Up	none	2009-05-13	After a lifetime of dreaming of travelling the world...	/gfFqbcoFW8uc	7.54	4	14160	0	0	0	0	0	3374
Taken 3	none	2015-01-01	Ex-government operative Bryan Mills finds his life...	/ikDwR3i2bczqr	6.12	5	260346	0	0	0	0	0	1350
Tom	none	2016-01-14	An astronaut encounters a major problem upon return...	null	0	6	377750	0	0	0	0	0	0
Hello	none	2014-01-01	When Max (Eric Stoltz), urged on by "Risk Management..."	null	3.5	7	299922	0	0	0	0	0	1
Gone Baby Gone	none	2007-10-19	Two Boston area detectives investigate a little girl...	/4Nxv90So2AH	6.95	8	4771	0	0	0	0	0	441

Table 16: movie data table

Graduation Project  
Project: Watch and Rate  
**Software Documentation**



TITLE	REVIEW_CONTENT	ENABLE_COMMENT	STAR_RATE	IMAGE REVIEW	REVIEW_ID	USER_ID	MOVIE_ID	DATE
zaft	zaften talata 4	1	3	none	10	3	228161	2016-06-20
zaf test image	loo	0	3	316-41-3513428443_1539l	12	3	228161	2016-06-20
lool	msm3tsh 3ano 5als	1	3	none	13	3	377750	2016-06-20
twest	test	1	3	318-34-20facepic11.png	14	3	377750	2016-06-20
test final test	uploadae image	1	3	323-44-0513428443_1539l	15	3	377750	2016-06-20
test after solve	ya rab	1	3	300-03-55facepic8.png	16	3	377750	2016-06-21
loo	o!!!	1	3	300-28-2313432274_1372l	17	3	377750	2016-06-21
test	test	1	3	301-23-17facepic4.png	18	3	299922	2016-06-21
test sd	test new server	0	3	316-25-01facepic9.png	19	3	299922	2016-06-21
final review	a5erb nf3t .D	1	2	217-25-50received_106211	20	2	228161	2016-06-21
Will Keep You Debating On What Is Truly Right	I had wanted to see this movie for sometime now. I...	1	13	none	22	13	4771	2016-06-25

Table 17: Review data table



## 5.2 Training Results using Weka

J48 decision tree is used as a classifier to classify train and test data using cross validation 10 folds.

### 5.2.1 Introduction

- **what is the main idea?**

The main idea is how to apply NLP

NLP is Natural language processing is a field of Computer Science, artificial intelligence and computational linguistics used to decrease the gap between human and computer using natural human language

- **how we use it?**

It used for getting the common feedback about a lot of things: movies reviews, recommendation systems, making surveys

- **Attributes meaning which is used through all trials:**

1. positive\_words: number of positive words in the sentence using sentiword.  
a word is considered positive if its positive weight is greater than negative weight.
2. negative\_words: number of negative words in the sentence using sentiword.  
a word is considered negative if its positive weight is less than negative weight.
3. positive\_score: the summation of positive weights of all words in a sentence
4. negative\_score: the summation of negative weights of all words in a sentence



- 5. strongPositive: number of positive words that have positive weight greater than 0.5 or a normal positive word but preceded by an intensifier.
- 6. strongNegative: number of negative words that have negative weight greater than 0.5 or a normal negative word but preceded by an intensifier.
- 7. adj\_words: number of adjective words in the sentence using Stanford POS tagger.
- 8. adv\_words: number of adverb words in the sentence using Stanford POS tagger.
- 9. subjective\_words: any word in the sentence that has emotion positive or negative – Used sentiword -
- 10. neutral\_words: any word in the sentence that doesn't have any emotion in it.
- 11. Class:** negative class or positive class

### 5.2.2 First Trial

Instances number: 10662

Recall= 0.542, Precision = 0.542

Attributes: 3

- 1. positive\_words
- 2. negative\_words
- 3. class

```

Time taken to build model: 0.28 seconds
=====
Stratified cross-validation ===
Summary ===

Correctly Classified Instances           5782               54.23   %
Incorrectly Classified Instances        4880                45.77   %
Kappa statistic                         0.0846
Mean absolute error                     0.4965
Root mean squared error                 0.4985
Relative absolute error                  99.2949 %
Root relative squared error            99.692  %
Total Number of Instances              10662

Detailed Accuracy By Class ===

      TP Rate    FP Rate    Precision    Recall    F-Measure    ROC Area    Class
          0.576     0.492       0.54       0.576      0.557      0.533   negative
          0.508     0.424       0.545      0.508      0.526      0.533   positive
Weighted Avg.      0.542     0.458       0.542      0.542      0.542      0.533

Confusion Matrix ===

      a      b  <-- classified as
3072  2259 |      a = negative
2621  2710 |      b = positive
  
```

figure 30: first trial Weka training



### 5.2.3 Second Trial

Instances number: 10662

Recall= 0.587, Precision = 0.587

Attributes: 5

1. positive\_words
2. negative\_words
3. adj\_words
4. adv\_words
5. Class

```
==== Stratified cross-validation ====
==== Summary ====

Correctly Classified Instances          6257                  58.685 %
Incorrectly Classified Instances        4405                  41.315 %
Kappa statistic                         0.1737
Mean absolute error                     0.4736
Root mean squared error                 0.4916
Relative absolute error                  94.7226 %
Root relative squared error            98.3153 %
Total Number of Instances              10662

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
          0.56       0.386     0.592      0.56       0.575     0.612   negative
          0.614      0.44      0.582      0.614       0.598     0.612   positive
Weighted Avg.      0.587      0.413      0.587      0.587       0.587     0.612

==== Confusion Matrix ====

      a      b  <-- classified as
2984 2347 |    a = negative
2058 3273 |    b = positive
```

Figure 31: second trials Weka training

### 5.2.4 Third Trial

Used less instances to decrease run time of the code. Added number of subjective words.

Instances number: 2000

Recall= 0.566, Precision = 0.566

Attributes: 5

1. positive\_words
2. negative\_words



- 
3. subjective\_words
  4. adj\_words
  5. adv\_words
  6. Class

```

Time taken to build model: 0.04 seconds

==== Stratified cross-validation ====
==== Summary ====

    Correctly Classified Instances      1132          56.6    %
    Incorrectly Classified Instances    868           43.4    %
    Kappa statistic                   0.132
    Mean absolute error               0.4826
    Root mean squared error          0.5022
    Relative absolute error          96.5194 %
    Root relative squared error     100.4374 %
    Total Number of Instances        2000

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
      0.528     0.396     0.571      0.528     0.549      0.564    negative
      0.604     0.472     0.561      0.604     0.582      0.564    positive
Weighted Avg.   0.566     0.434     0.566      0.566     0.565      0.564

==== Confusion Matrix ====

      a     b  <-- classified as
  528 472 |  a = negative
  396 604 |  b = positive

```

Figure 32: third test Weka training

### 5.2.5 Fourth trial

This trial had a problem in the code that raised the accuracy so high. As the class attribute was used in 2 conditions in the code that generated the arff file. These conditions were removed in the sixth and final trial.

Instances number: 2000

Recall= 0.971, Precision = 0.971

Attributes: 9

1. positive\_words
2. negative\_words
3. strongPositive
4. strongNegative



```

5. subjective_words
6. neutral_words
7. adj_words
8. adv_words
9. class

==== Stratified cross-validation ====
==== Summary ====

    Correctly Classified Instances      1941           97.05   %
    Incorrectly Classified Instances     59            2.95   %
    Kappa statistic                   0.941
    Mean absolute error               0.0403
    Root mean squared error          0.1574
    Relative absolute error          8.0522 %
    Root relative squared error      31.4868 %
    Total Number of Instances        2000

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
      0.967     0.026     0.974     0.967     0.97       0.981   negative
      0.974     0.033     0.967     0.974     0.971      0.981   positive
    Weighted Avg.     0.971     0.03     0.971     0.971     0.97       0.981

==== Confusion Matrix ====

      a     b   <-- classified as
  967  33 |   a = negative
   26 974 |   b = positive

```

1.

Figure 33: fourth test Weka training

### 5.2.6 Fifth Trial

Solved the problem that increased run time.

Added to the last trial 3 attributes: positive\_score, negative\_score, neutral words

Instances number: 10662

Recall= 0.994, Precision = 0.994

Attributes: 11

1. positive\_words
2. negative\_words
3. positive\_score

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

---



4. negative\_score
5. strongPositive
6. strongNegative
7. subjective\_words
8. neutral\_words
9. adj\_words
10. adv\_words
11. class

```
==== Stratified cross-validation ====
==== Summary ====


```

Correctly Classified Instances	1941	97.05 %
Incorrectly Classified Instances	59	2.95 %
Kappa statistic	0.941	
Mean absolute error	0.0403	
Root mean squared error	0.1574	
Relative absolute error	8.0522 %	
Root relative squared error	31.4868 %	
Total Number of Instances	2000	

```
==== Detailed Accuracy By Class ====


```

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.967	0.026	0.974	0.967	0.97	0.981	negative
0.974	0.033	0.967	0.974	0.971	0.981	positive
Weighted Avg.	0.971	0.03	0.971	0.971	0.981	

```
==== Confusion Matrix ====


```

a	b	<- classified as
967	33	a = negative
26	974	b = positive

Figure 34: fifty test Weka training

### 5.2.7 Final trial

Removed the 2 conditions that took class type into consideration.

Instances number: 10662

Recall= 0.941, Precision = 0.94

Attributes: 11

1. positive\_words
2. negative\_words
3. positive\_score

Graduation Project  
Project: Watch and Rate  
**Software Documentation**

---



4. negative\_score
5. strongPositive
6. strongNegative
7. subjective\_words
8. neutral\_words
9. adj\_words
10. adv\_words
11. class

```
==== Stratified cross-validation ====
==== Summary ====

Correctly Classified Instances      10017          93.9505 %
Incorrectly Classified Instances    645           6.0495 %
Kappa statistic                   0.879
Mean absolute error               0.0755
Root mean squared error           0.1987
Relative absolute error           15.0939 %
Root relative squared error      39.734 %
Total Number of Instances         10662

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
        0.912     0.033     0.965     0.912     0.938     0.989   negative
        0.967     0.088     0.917     0.967     0.941     0.989   positive
Weighted Avg.      0.94      0.06     0.941     0.94      0.939     0.989

==== Confusion Matrix ====

      a     b  <-- classified as
4864  467 |   a = negative
  178 5153 |   b = positive
```

---

Figure 35: final test Weka training



### 5.3 Ranking attributes in the model

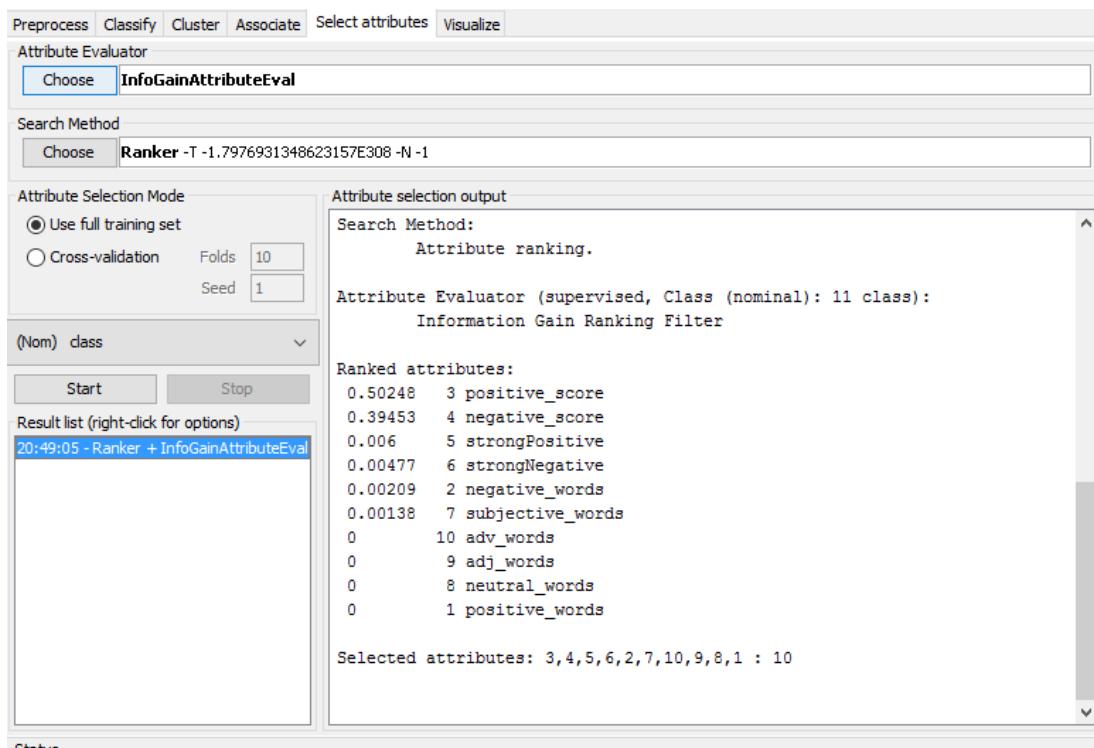


Figure 36: Ranking important attributes to less important

### 5.4 Future work

1. User can rate any review for any movie (working on it)
2. Applying the sentiment analysis for movies on Arabic language also (not only English)
3. Providing messages can user interact with people within it
4. Providing search by category feature (comedy, drama, action, etc. movies)
5. User can see any user's profile including his reviews and score
6. User can control application settings like colors
7. User can use “push notification” feature