**Final Year Project**

**Final Evaluation Report**

**Rel-Event**

**A Predictive Event Finder Application**

**FYP Team**

### **Rana Zubair (14i-0285)**

### **Muhammad Aqib (14i-0224)**

### **Aneeb Hassan (14i-0329)**

**Supervised by Dr. Mirza Omer Beg**

**FAST School of Computing**

**National University of Computer and Emerging Sciences**

**Islamabad, Pakistan**

**2019**

**Intellectual Property Right Declaration**

###### This is to declare that the work under the Title:

Rel-event – A predictive event finder application **carried out in partial fulfillment of the requirements of BS FYP,**

is the sole property of National University of Computer and Emerging Sciences, and is protected under the intellectual property right laws and conventions. It can only be considered/used for purposes like an extension for further enhancement, product development, adoption for commercial/organizational usage, etc., with the permission of the University.

This above statement applies to all students and faculty.

Date: 10th June 2020

Student 1

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student 2

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student 3

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor (Faculty)

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Anti-Plagiarism Declaration**

This is to declare that the above publication produced under the title:

###### Rel-event – A Predictive Event Finder Application

is the sole contribution of the author(s) and no part hereof has been reproduced on as it is basis (cut and paste) which can be considered as Plagiarism. All referenced parts have been used to argue the idea and have been cited properly. I/We will be responsible and liable for any consequence if violation of this declaration is determined.

Date: 10rd June, 2020

**Authors’ Declaration**

It is hereby declared that the work presented in the report is our own, and has not been submitted/presented previously to any other institution or organization

**Executive Summary**

Rel-event is an event recommendation and searching application which aims to has combine the latest influx of Natural Language processing and machine learning algorithms to open doors for predictive analysis on event searches and popularity of an event.

In this of age of information, its problematic, time consuming, and frustrating to find information which is relevant to you and to your needs, especially in finding events that match your type around the city that you live in or visit.

This project aims to scrap live data from event websites and categorize those events into 6 different types such as Sports, Music, Food etc. There will be 2 types of users:

1. Customer
2. Organizer

As the Customer uses the application with time, it will learn the user’s choices by saving user activity in database. The engine will apply model-based recommendation and collaborative filtering to recommend events related to his/her personality choice.

As the Organizer, a user can predict the popularity/success of his event by entering details of the event he plans to market through sentiment analysis.

Table of Contents

[Chapter 1 - Introduction 1](#_bookmark0)

1.1 [Overview 1](#_bookmark1)

1.2 [Motivation 1](#_bookmark2)

[Chapter 2 – Project Vision 2](#_bookmark3)

* 1. [Problem Statement 2](#_bookmark10)
  2. [Business Opportunity 2](#_bookmark11)
  3. [Objectives 3](#_bookmark12)
  4. [Constraints 3](#_bookmark12)

[Chapter 3 – Software Requirements Specifications 4](#_bookmark9)

* 1. [List of Features 4](#_bookmark10)
  2. [Functional Requirements 4](#_bookmark11)
  3. [Non-Functional Requirements 5](#_bookmark12)

3.4 [Project Process Flow Diagram 6](#_bookmark12)

[Chapter 4 – Use Cases 7](#_bookmark14)

* 1. [Use Case Diagram 7](#_bookmark15)
  2. [High Level Use Cases 8](#_bookmark16)

[Chapter 5 – Iteration Plan 10](#_bookmark21)

* 1. [Timeline 10](#_bookmark17)

[Chapter 6 – Iteration 1 and 2 11](#_bookmark21)

6.1 [Extended Use Cases 11](#_bookmark17)

6.2 [Domain Model 18](#_bookmark17)

6.3 [System Sequence Diagram 19](#_bookmark17)

6.4 [Operation Contracts 20](#_bookmark17)

6.6 [Class Diagram 24](#_bookmark17)

[Chapter 7 – Implementation Details 25](#_bookmark21)

7.1 [Data Scrapping, Extraction and Classification 25](#_bookmark17)

7.2 [Recommendation Engine 27](#_bookmark17)

7.3 [Sentiment Analysis 28](#_bookmark17)

7.4 [User Interface 29](#_bookmark17)

Chapter 1 - Introduction

## 1.1 Overview

The domain of event management and searching has combined well with the latest influx of Natural Language processing and machine learning algorithms to open doors for predictive analysis on event searches and the popularity of an event. The huge of amount data that is being generated from the field of event management itself has led to a lot of research and development work on predicting the popularity of an upcoming event, narrowing and classifying the events into different categories, and patterns, and feature extraction through user’s history.

## 1.2 Motivation

In this age of information, its problematic, time consuming and frustrating to find information that is relevant to you and your needs, especially in finding events that match your type around the city that you live in or visit. Also, will that event be successful or not? For example: Debating events, Sports events, Corporate events, etc.

# Chapter 2 – Project Vision

#### 2.1 Problem Statement

Till this date, event categorization and popularity prediction of an upcoming event to specifically cater to the users' needs which have not been up to the mark. Machines are yet to beat human expertise in the relatively uncertain field of football analysis. The aim, therefore, is to use historical and realtime event-related data to effectively feed and tune intelligent models to get predictive insight into future events. Given enough data to the latest machine and deep learning algorithms, the goal would be to effectively predict whether the event would be successful to a specific percentage, The research aims to explore various machine and deep learning algorithms that apply to the problem structure, fed by previous events history data, their positive or negative reviews, and the number of people associated with it.

* 1. **Business Opportunity**

This event finding app helps a large number of outgoing and social people, and not only them but those people as well who want to attend events that will have a smaller number of people and also match their personality.

This app would also be a tremendous help to event organizers to have

an insight into their upcoming events and what they can do to improve it.

* 1. **Objectives**
* Gathering data from multiple websites and datasets from social media
* Extraction of relevant information and its categorization.
* Sentiment Analysis for success/failure of an event
* Developing an android app with proper UI/UX techniques to display events.
* Providing the functionality to the app to learn from the user’s choices

## Constraints

## Managing the large history of user activity

* Continuous model training on user interests with respect to history
* Optimizing the new event’s suggestion through sentimental analysis
* Integrating a huge amount of training data continuously with our app
* Predict the popularity of an upcoming event

# Chapter 3 – Software Requirements Specification

## List of Features

## We will implement a user-friendly interface.

## The system will be able to classify the events by extracting useful features from the user database and past events and then classify them into their respective categories.

## Our system will work on the following criteria:

## Optimization

## Interaction

## Predictability

## Functional Requirements

* The user of the website will be admin of the website who will put/upload the images and the algorithm will do the rest of the job for him.
* Users can also search for the product by writing its name in the search box to see if the product has been classified successfully.
* The system shall process the images and by analyzing the data provided, will produce efficient results for the user which he can use in his business techniques.
* The user will be able to view the product and the details of that product once the algorithm has finished classifying it successfully.

## Non-Functional Requirements

* **Usability**This section lists all of those requirements that relate to or affect the usability of the system.
* **Design for ease of use**

The user interface for users shall be designed for ease of use and shall be appropriate for any community that owns a phone.

* **Reliability**

A maximum effort will be made to make sure the predictability rate is above a satisfaction rate.

* **Availability**

Our product is for all android phone users; he can use it for his purpose as long as he is working on it. No constraint on the number of hours.

* **Performance**

For the user experience to be as smooth as possible, the majority of the classification and application page loading should happen in less than 5 seconds. However, some flexibility is acceptable for parts of the pages that require to automate the main process however it should be reduced to a minimum. Our goal is to provide the user with ultimate system performance for most of the time.

* **Supportability**

Efficient tools and coding methods will provide the best possible support to make sure the system works in the best possible way.

## Project Process Flow Diagram

## 

# Chapter 4 – Use Cases

## Use Case Diagram

## High-Level Use Cases

|  |  |
| --- | --- |
| Name: | Login to application |
| Actor: | Customer and Organizer |
| Type: | Primary |
| Description: | The user enters the required information and selects the desired functionality to enter into the application. |

## 

|  |  |
| --- | --- |
| Name: | Add Event |
| Actor: | Organizer |
| Type: | Primary |
| Description: | The user selects the option and is asked to fill in mandatory event details. |

|  |  |
| --- | --- |
| Name: | Delete Event |
| Actor: | Organizer |
| Type: | Primary |
| Description: | The user selects an event which he desires to remove. |

|  |  |
| --- | --- |
| Name: | View Event |
| Actor: | Customer / Organizer |
| Type: | Primary |
| Description: | The user selects from a list of recommended events he wishes to know more details about. |

|  |  |
| --- | --- |
| Name: | View Notifications |
| Actor: | Customer / Organizer |
| Type: | Primary |
| Description: | The user checks the designated tab for updates on chosen events. |

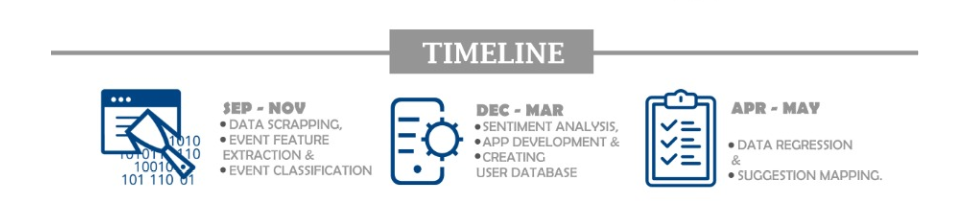
|  |  |
| --- | --- |
| Name: | Search Events |
| Actor: | Customer / Organizer |
| Type: | Primary |
| Description: | User types in the event name or chooses an event category to find desired events. |

|  |  |
| --- | --- |
| Name: | View Profile |
| Actor: | Customer / Organizer |
| Type: | Primary |
| Description: | The user is taken to his customized page with all his account details. |

|  |  |
| --- | --- |
| Name: | View History |
| Actor: | Customer / Organizer |
| Type: | Primary |
| Description: | The user is shown all his feedback on selected events. |

# Chapter 5 – Iteration

## 5.1 Timeline



# Chapter 6 – Iteration 1 and 2

## 6.1 Extended Use Cases

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | Login into application | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 30/09/2019 | Last Revision Date: | 01/09/2019 |
|  |  | | |
| **Description:** | User opens the application on phone and is shown starting screen asking to choose to fill in the required information | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has the application installed on the phone 2. Phone location services are activated | | |
| **Postconditions:** | 1. User has successfully opened the home page of the application | | |
| **Flow:** | 1. The user enters his information such as Name, Age, etc. 2. The application will ask for user request between recommendation or creation 3. The user chooses to select recommendation 4. The system accepts user input | | |
| **Alternative Flows:** | 5. In step 3 of the normal flow, chooses creation:       1. The system accepts user input | | |
| **Exceptions:** | In step 3 of the normal flow, if the user did not enter the correct event details:      1. He will be asked to re-enter details | | |
| **Requirements:** | The following requirements must be met before the execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | View Event | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 30/09/2019 | Last Revision Date: | 01/09/2019 |
|  |  | | |
| **Description:** | The user opens the application on phone and is shown the list of categories and by opening a category user is shown a list of events. | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has the application installed on the phone 2. Phone location services are activated | | |
| **Postconditions:** | 1. User has successfully opened a list of categories, selects the desired event and is shown details of a selected event | | |
| **Flow:** | 1. The user selects a category from the home page 2. The application will show a list of events from the selected category 3. The user chooses an event 4. Application opens and shows details of events | | |
| **Alternative Flows:** | 5. In step 1 of the normal flow, user changes selected city:       1. The system accepts user input | | |
| **Requirements:** | The following requirements must be met before the execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

## 

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | Search Events | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 25/05/2020 | Last Revision Date: | 30/05/2020 |
|  |  | | |
| **Description:** | The user opens the application on phone and is shown a list of categories and by opening a category user is shown a list of events. He then clicks on the search option to initiate the desired search. | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has the application installed on the phone 2. Phone location services are activated | | |
| **Postconditions:** | 1. User has successfully opened the application | | |
| **Flow:** | 1. User is waiting on the home screen  2. The user pressed the search option to initiate a search 3. User types input required for the desired event to be searched.  4. Application opens and shows details of events | | |
| **Requirements:** | The following requirements must be met before the execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | View Profile | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 25/05/2020 | Last Revision Date: | 30/05/2020 |
|  |  | | |
| **Description:** | The user opens the application on the phone and is shown a list of categories. The user then goes onto the navigation tab and selects settings where he can view profile details. | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has the application installed on the phone 2. Phone location services are activated  3. The profile has been made | | |
| **Postconditions:** | 1. User has successfully opened settings from the navigation tab, is shown details of the profile | | |
| **Flow:** | 1. The user selects the navigation tab on the top left of the screen from the home page 2. The application will show a list of commands  3. The user chooses on settings command 4. Profile details are shown | | |
| **Requirements:** | The following requirements must be met before the execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | View History | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 25/05/2020 | Last Revision Date: | 30/05/2020 |
|  |  | | |
| **Description:** | The user opens the application on phone and navigates to My Events placed in the navigation tab. | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has the application installed on the phone 2. Phone location services are activated  3. User has a profile created on application  4. User has added events to list | | |
| **Postconditions:** | 1. The user has successfully opened My Events tab, and if available, is shown an entire list of past events. | | |
| **Flow:** | 1. The user selects My Events from the navigation tab 2. The application will show a list of events from past rated endeavors. | | |
| **Requirements:** | The following requirements must be met before the execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | Add Event | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 25/05/2020 | Last Revision Date: | 30/05/2020 |
|  |  | | |
| **Description:** | The user opens the application on phone and is shown a list of categories and by opening a category user is shown a list of events. He selects on the desired event and rates it which will add the selected event in the My Events section. | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has the application installed on the phone 2. Phone location services are activated | | |
| **Postconditions:** | 1. User has successfully opened a list of categories, selects the desired event and is shown details of a selected event | | |
| **Flow:** | 1. The user selects an event from the chosen category 2. The application will show details of events  3. User rates an event 4. The application adds event into My Events section of the application | | |
| **Requirements:** | The following requirements must be met before the execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Use Case: | Delete Event | | |
| Created By: | Aneeb Hassan | Last Updated By: | Aneeb Hassan |
| Date Created: | 25/05/2020 | Last Revision Date: | 30/05/2020 |
|  |  | | |
| **Description:** | User opens the application on phone. He opens the navigation tab and clicks on My Events.He clicks on the events he wants to delete and the application removes it. | | |
| **Actors:** | Customer, Organizer | | |
| **Preconditions:** | 1. User has application installed on phone 2. Phone location services are activated | | |
| **Postconditions:** | 1.User has successfully opened list of categories. | | |
| **Flow:** | 1. User selects My Events from navigation tab 2. Application will show list of events from selected category 3. User chooses an event to delete 4. Application deletes selected event | | |
| **Requirements:** | The following requirements must be met before execution of the use case 1. Active internet connection 2. Active phone location services (GPRS) | | |

## Domain Model

## System Sequence Diagram

## Operation Contracts

## Operation Contract 1: Login (Name, Email)

## Cross References: Log in to application use case

## Pre-Conditions:

## User has a working internet connection

## User wants to sign into application

## Post-Conditions:

## User instance is created

* Profile instance is created
* User and profile association is formed

## Operation Contract 2: Authenticate (Profile)

## Cross References: Log in to application use case

## Pre-Conditions:

## User has entered correct Name and Email information

## Pre-Conditions:

## Name and Email attributes modified

## Operation Contract 3: ShowCategory (Events)

## Cross References: View Events

## Pre-Conditions:

## User has a working internet connection

## User has successfully signed into the application

## Post-Conditions:

## Event catalog instance is created

* Event instance is created
* User and event catalog association is formed
* Event catalog and event association is formed
* Name, Location and Type attributes are modified

## Operation Contract 4: RequestDetails(Name, Location, Type)

## Cross References: View Events

## Pre-Conditions:

## User has a working internet connection

## User has successfully signed into the application

## User has selected category

## Post-Conditions:

## Event catalog instance is created

* Event instance is created
* User and event catalog association is formed
* Event catalog and event association is formed
* Name, Location and Type attributes are modified

## Operation Contract 5: ViewHistory (MyEvents)

## Cross References: View History

## Pre-Conditions:

## User has a working internet connection

## User has successfully signed into the application

## Post-Conditions:

## Event catalog instances are fetched

* User and event catalog association is formed
* Event catalog and eventlsit association is formed
* Name, Location and Type attributes are modified

## Operation Contract 6: RequestEvents(Name, Location, Type,Rated)

## Cross References: View History

## Pre-Conditions:

## User has a working internet connection

## User has successfully signed into the application

## User has opened My Events section

## Post-Conditions:

## Event catalog instance is created

* Event instance is created
* User and event catalog association is formed
* Event catalog and event association is formed
* Name, Location and Type attributes are modified

## Operation Contract 7: SearchEvent (Name,Location,Type)

## Cross References: Search Event

## Pre-Conditions:

## User has a working internet connection

## User has successfully signed into the application

## Post-Conditions:

## Event catalog instances are fetched

* User and event catalog association is formed
* Event catalog and eventlsit association is formed
* Name, Location and Type attributes are modified

## Operation Contract 8: RequestEvents(Name, Location, Type,Rated)

## Cross References: View History

## Pre-Conditions:

## User has a working internet connection

## User has successfully signed into the application

## User has opened My Events section

## Post-Conditions:

## Event catalog instance is created

* Event instance is created
* User and event catalog association is formed
* Event catalog and event association is formed
* Name, Location and Type attributes are modified

## Class Diagram

## 

# Chapter 7 – Implementation Detail

## 7.1 Data Scrapping, Extracting and

## Classification

We are using a library called beautiful soup to gain access to the information provided on the website. It’s a python library used for pulling data out of HTML and XML files.

This is the basic syntax of an HTML webpage. Every <tag> serves a block inside the webpage:  
1. <!DOCTYPE html>: HTML documents must start with a type declaration.  
2. The HTML document is contained between <html> and </html>.  
3. The meta and script declaration of the HTML document is between <head>and </head>.  
4. The visible part of the HTML document is between <body> and </body>tags.  
5. Title headings are defined with the <h1> through <h6> tags.  
6. Paragraphs are defined with the <p> tag.

Other useful tags include <a> for hyperlinks, <table> for tables, <tr> for table rows, and <td> for table columns.

Also, HTML tags sometimes come with id or class attributes. The id attribute specifies a unique id for an HTML tag and the value must be unique within the HTML document. The class attribute is used to define equal styles for HTML tags with the same class. We are finding relevant data from all the information provided on a webpage using these HTML tags, which include the event name, description, time, venue, and source. After getting relevant data from the webpage, we are sending it to a list.

Using **xlwt module**, we performed multiple operations on the spreadsheet. For example, writing or modifying the data. Also, we might have to go through various sheets and retrieve data based on some criteria or modify some rows and columns and do a lot of work.

We need to share our extracted data with our team and Google Drive is always our first choice. One major issue over here is we have to do it on a weekly or even daily basis, which is very boring. All of us hate repetitive tasks, including me.

Fortunately, Google provides API for most of its service. We are going to use [**Google Drive API**](https://developers.google.com/drive/) and **[PyDrive](https://pythonhosted.org/PyDrive/" \t "_blank)** to manage our files in Google Drive.

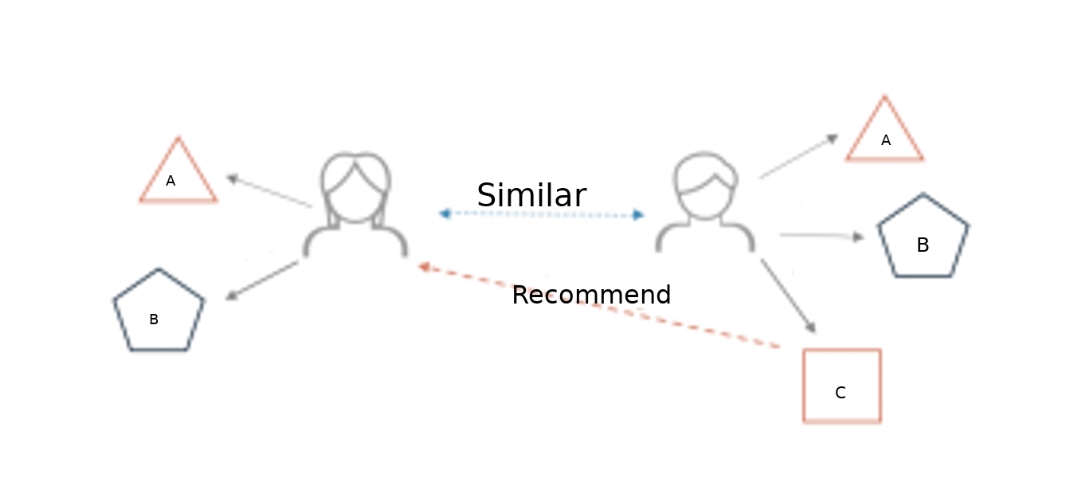
After getting the relevant data from the Scrapping, we need to classify the event into cities and different type. We categorized the event type into seven different types which include Workshops, Art & theater, and music. We wrote an algorithm that checks every word and classifies the event in type and venue based on some tags or words. We update the excel file again using xlwt module.

## Recommendation Engine

A recommender system is a simple algorithm whose aim is to provide the most relevant information to a user by discovering patterns in a dataset.

There are 2 main types of engines we plan to combine and apply in this project. They are model-based collaborative filtering and memory-based collaborative filtering using root mean squared error and cosine similarity.

**Memory-Based Collaborative filtering:** In this model, events are recommended to a user based on the fact that the events have been liked by the user.

**Model-based Collaborative filtering:** these methods are based on matrix factorization and are better at dealing with sparsity. They are developed using data mining, machine learning algorithms to predict users’ ratings of unrated items. In this approach techniques such as dimensionality reduction are used to improve accuracy.

## Sentiment Analysis

Sentiment analysis works by analyzing text with different polarities (example positive or negative), whether a whole document, paragraph, clause.

We went with aspect-based sentiment analysis in our project

Usually, when analyzing sentiments of texts, i.e. event reviews, you’ll want to know which particular aspects or features people are mentioning in a positive, neutral, or negative way.

The algorithm our system uses is a Hybrid approach that combines rule-based and automatic approaches. One benefit of this type of system is that results are often more accurate.

A rule-based system uses a set of human-crafted rules to help identify subjectivity, polarity, or the subject of opinion. An automatic-based system doesn’t work with manually crafted rules but rather depend on machine learning techniques. A task is modeled in the form as a classification problem. A classifier has inputted a text and returns the desired category such as positive or negative.

The training process:

* The model learns to relate a particular input (i.e. a text) to the corresponding output (tag) based on the test inputs used for training. The feature extractor transfers text input to a feature vector. Pairs of feature vectors and tags (such as positive or negative) are fed into a machine learning algorithm to create a model.

The prediction process

* Feature extraction from the training process is used to convert unseen text entries into feature vectors. These feature vectors are then inputted into the model, generating predicted tags (positive, negative).

## User Interface

## 

## Splash Screen Log-In Page

## Home Screen Navigation Tab

## 

## Workshop Category Event Details

## 

**Sentiment Analysis**