Software Requirements

Specification



Decide

**Table of contents**

1. Introduction

1.1 Purpose

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

1.4 Scope

1.5 Definitions, acronyms, and abbreviations

1.6 References

1.7 Overview

2. Overall description

2.1 Product Perspective

2.1 Product Functions

2.3 User Characteristics

2.4 Constraints

2.5 Assumptions and dependencies

2.6 Apportioning of requirements

3. Specific requirements

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.2 Functional requirements

3.3 Performance requirements

3.4 Design constraints

3.5 Software system attributes

4. Release plane

Appendix I

Appendix II

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to provide detailed description to the operation of the **Free-software** “**Decide**”. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

**1.2 Document Conventions**

This Document was created based on the IEEE template for System Requirement Specification Documents.

**1.3 Intended Audience and Reading Suggestions**

* Typical Users, such as students, who want to use Decide for reviewing analysis.
* Programmers who are interested in working on the project by further developing it or fix existing bugs.

**1.4 Scope**

The “Decide” is a Desktop application which helps users specially food business owners to analyze the customer english reviews quickly and efficiently through computer without any human interference.

**Decide** is a tool aims to help users to build data driven business by facilitating the analyzing of data through fast accurate classification and visualization.

The application should be free to download from our GitHub page Restaurant owners can inputs the required reviews to be classified (1:5 when using the default model “SVC.pkl”) , they also can use a pre-trained model of their choosing to classify the data.

Furthermore, the software need not Internet connection to display results.

All results will put in same folder .

**1.5 Definitions, acronyms, and abbreviations**

|  |  |
| --- | --- |
| **Work** | **Definition** |
| User | Someone who interacts with application |
| Restaurant Owner | Someone who has a restaurant and wants to use the application |
| Stakeholder | Any person who has interaction with the system who is not a developer. |
| NLP | Natural language processing |
| Sentiment-analysis | Process of classification of texts to different rating classes |
| Training | Core process to create a learning model |
| Pre-trained | Any model that is not loaded as default in official version |
| Prediction | Process of using an already created model to classify input to desired classes |
| NLTK | Open-source library for natural language processing |
| Prediction set | Set of input reviews that are desired to be classified using a trained model |
| SRS | Software Requirements Specification, documents that contains the needed requirements by the customer. |
| Use Case | Diagram that consists of a series of actions that a user must initiate with the system to carry out some useful work and to achieve his/her goal. |
| DEP | Dependency |
| RAT | Rational |
| DESC | Description |
| MUST | The minimum level required to avoid failure contained in a PLanguage  statement [2] |
| PLAN | The level at which good success can be claimed contained in a PLanguage  statement [2] |
| WISH | A desirable level of achievement that may not be attainable through available means contained in a PLanguage statement [2] |
| DEFINED | The official definition of a term contained in a PLanguage statement [2] |

**1.6 References**

IEEE Template for System Requirement Specification Documents:

<https://goo.gl/nsUFwy>

Decide GitHub page:

<https://github.com/JustAhmed/Reviews-Classifier>

GNU General Public License version 3:

http://www.gnu.org/licenses/gpl.html

**1.7 Overview**

The remainder of this document includes three chapters and appendices. The second one provides an overview of the system functionality and system interaction with other systems. This chapter also introduces different types of stakeholders and their interaction with the system. Further, the chapter also mentions the system constraints and assumptions about the product.

The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences.

The fourth chapter deals with the prioritization of the requirements. It includes a motivation for the chosen prioritization methods and discusses why other alternatives were not chosen.

The Appendixes in the end of the document include the all results of the requirement prioritization and a release plan based on them.

**2. Overall description**

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

Decide was developed for everyone who is interested in graphs and wants either to just experiment with them so that he can understand them or wants to use them as a means of analyzing data.

It is Free Software project and it has a very active developer team to support it and provide feedback to users. It was developed to run on Windows.

**2.1 Product Perspective**

**Decide** is a desktop open-source application that run on windows platform.

It could be used to classify reviews to any desired or intended classes, by default 1:5 stars rating and for the food domain.

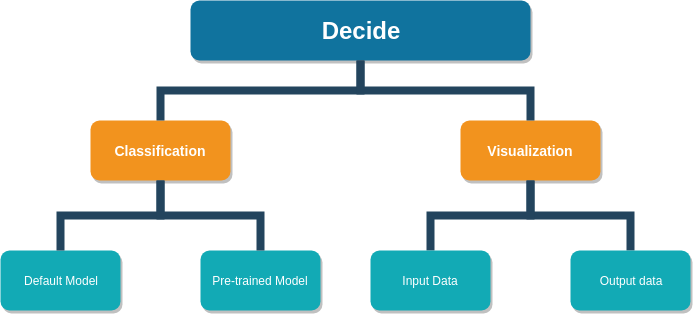
The System will contain two parts the classification part and the visualization part .

User provide the predictions set as a csv file and can use a pre-trained model.

User can choose a visualization technique to display both input and output.

User will have the classified data in csv file and any pre-specified charts.

User are allowed to input prediction set with maximum size of 250 MB



**2.2 Product functions**

**GUI Window**

**Import (model) :**

* import the model the user wants to use
* leave it blank the app will use the default model

**Import ( review data) :**

* import the user data he wants to analysis

**classify :**

* classify the user data either based on his own model or using the default model

**bar chart :**

* display a bar chart graph of the output

**pie chart :**

* display a pie chart graph of the output

**histogram :**

* display a histogram of the output

**most frequently word :**

* display the most frequently used word in user review data

**text length hist :**

* display a histogram of the user review data based on the review length

**text length boxplot :**

* display a boxplot of the user review data based on the review length

**CMD Window**

user user enter all orders in one line

(python Decide.py "Model.pkl" "dataset.csv" [bpflw])

**Decide.py** :main program (Decide)

**Model.pkl :**

import the model the user wants to use leave it blank the app will use the default model

**dataset.csv** : the user data he wants to classify

**b** : display bar chart of the output

**p** : display pie chart of the output

**f** : display a histogram of most frequently used word of the input data

**l** : display text length histogram of the input data

**w** : display text length boxplot of the input data

**2.3 User Characteristics**

There is only one type of users that interact with the system: restaurant

owners .That User has to use **the** system and has his own requirements.

User can use the loaded pre-trained model to predict the classification of texts and input reviews.

User needs to import the data CSV file and can have certain graphs displayed

**2.4 Constraints**

Available computation power must be large enough for the prediction sets.

Some prediction models might require max length of input words.

Note: We advice users to read pre-trained model documentation

Available labeled training set which are used by the developing team for creating the default model

Collect the reviews will also be a constrain

**2.5 Assumptions and dependencies**

One assumption about the product is that it will always be used on windows Desktop that have enough performance. If the computer does not have enough hardware resources available for the application, for

example the users might have allocated them with other applications, there may be scenarios where the application does not work as intended or even at all.

Input CSV file must be arranged in certain form (all reviews under a header called text).

For training model, the user must have available computation resources to the intended prediction process.

**2.6 Apportioning of requirements**

In the case that the project is delayed, there are some requirements that could be transferred to the next version of the application. Those requirements are to be developed in the third release.

**3. Specific requirements**

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features.

**3.1 External interface Requirements**

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

**3.1.1 User interfaces**

There are 2 user interface CMD mode or GUI mode :

**CMD mode :**

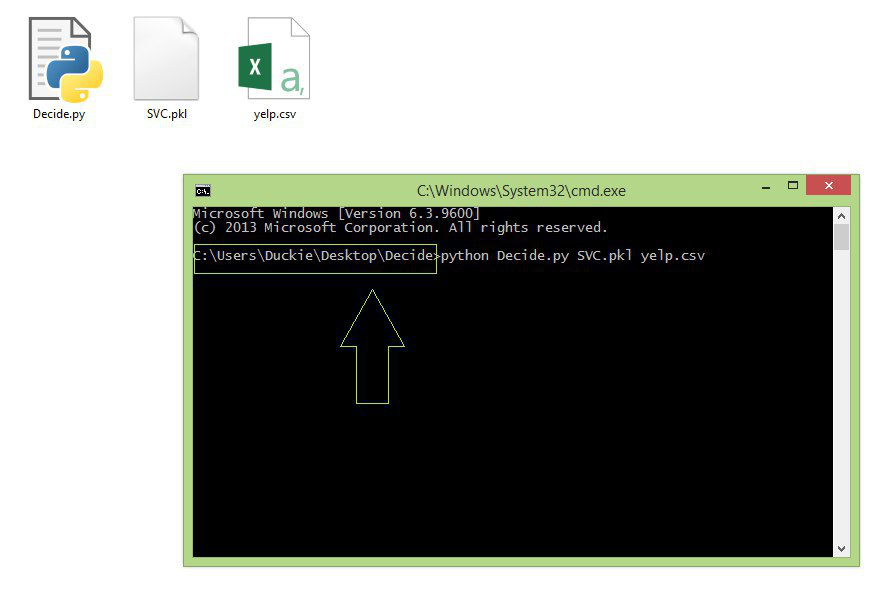
initially the software directory has 3 files

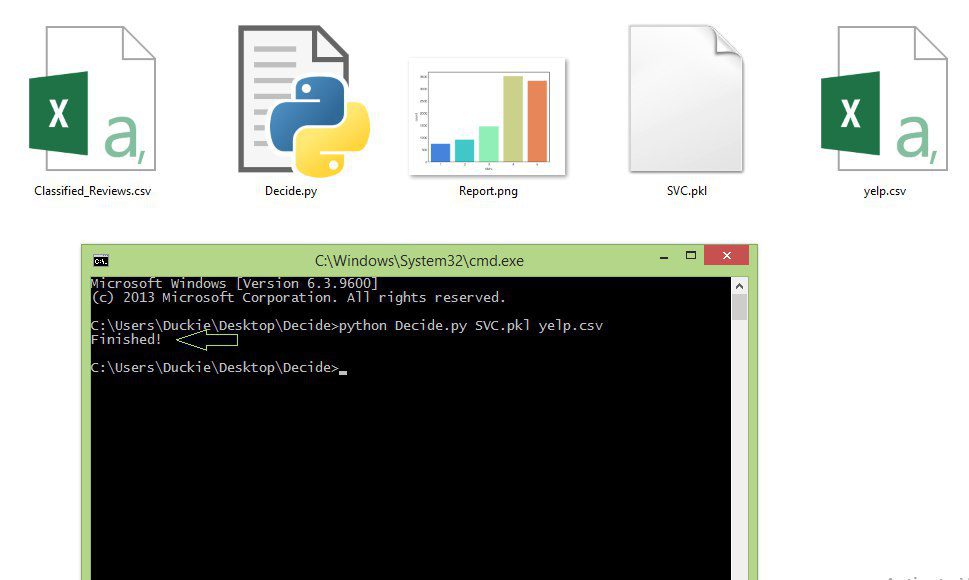
Open CMD from the directory

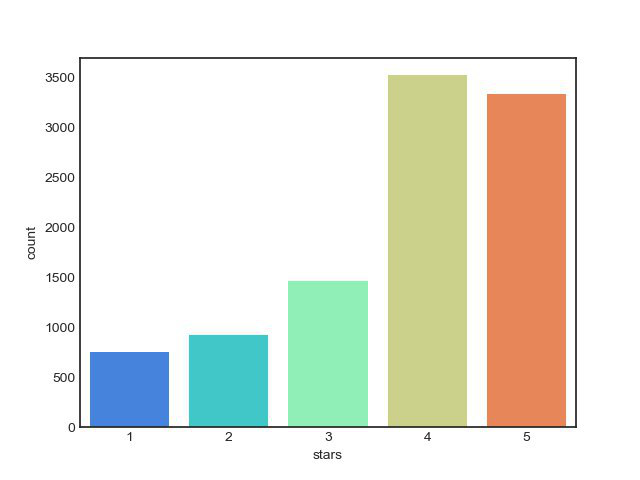
Type in the command as in image

Command → python Decide.py classifier.pkl reviews\_file.csv

After processing time 2 extra files will appear the predicted csv file and the bar chart.

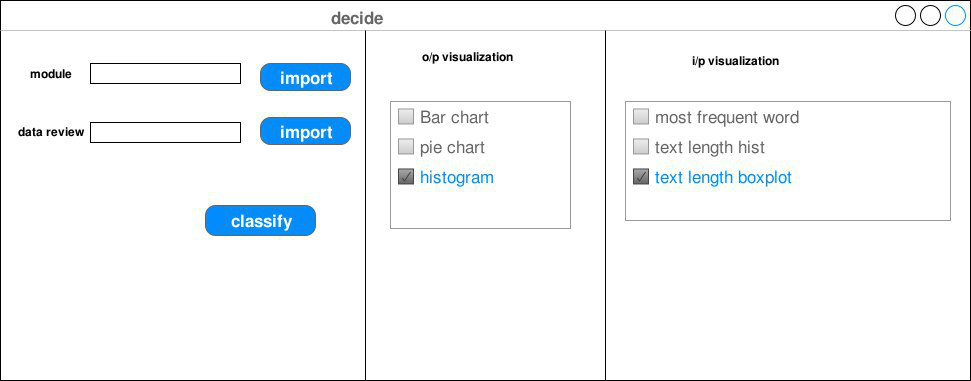






**GUI mode :**

User should see the following window on starting the application, User has model.pkl as the default model User can navigate through import button to import pre-trained model.

User can choose among various visualization techniques for both input and output

**3.1.2 Hardware interfaces**

Recommended requirements:

* 2 GB Ram
* 2 dual core GHz processor

**3.1.3 Software interfaces**

To avoid any unexpected behavior, any pre-trained model should be written using scikit-learn version 0.19.0 or higher or higher.

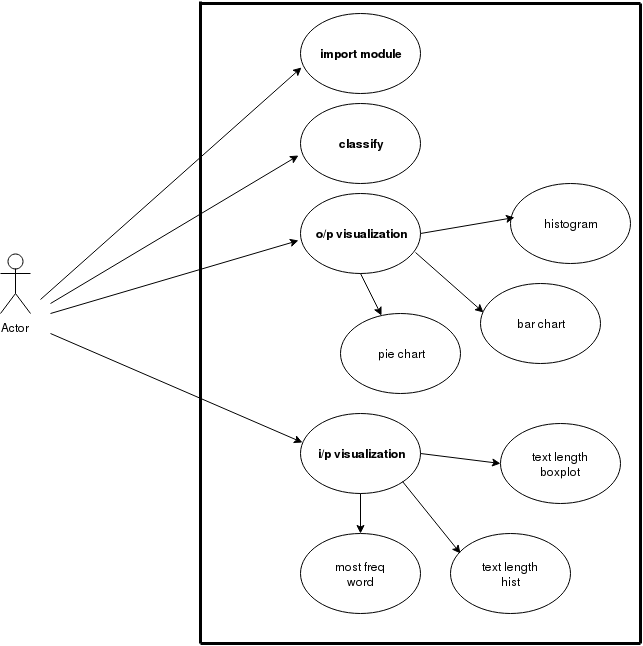
Pre-trained model should be a pickle model.

Reviews are stored in csv file and under a header called text.

**3.2 Functional Requirements**

This section includes the requirements that specify all the fundamental actions of the software system.

The following figure shows use case diagram



**3.3.1.1Functional requirement 1.1**

**the user**

**ID : FR1**

TITLE : classify

DESC : when we want to classify the data we should first import the data file and model to start classification the user can also customize the way data will be classified or the way that data will be printed by I/P visualization or O/P visualization options.

The user can do all of the above either by :

* **CMD** mode : the user must specify the the dataset directory (ex: [C:\](about:blank) data.csv)

or

* **GUI** mode : the user navigate to the directory where he inserted the the date set

DEP : None

**3.3.1.2 Functional requirement 1.2**

**ID: FR2**

TITLE : Import Model

DESC : the user can use the default mode or using his own model, user can import model either by CMD or GUI.

If the user chose CMD he must insert the file directory or he can navigate to the directory by choosing GUI mode

DEP : FR1

**3.3.1.3 Functional requirement 1.3**

**ID : FR3**

Title : data Reviews

DESC : In order to import reviews user has two options either using CMD mode or GUI mode, if the user chose CMD mode he must specify the the dataset directory (ex: [C:\](about:blank) data.csv), or user can choose the GUI mode and then navigate to the directory where he inserted the the date set

DEP : None

**3.3.1.4 Functional requirement 1.4**

**ID: FR4**

TITLE : I/p visualization technique

DESC : user can choose I/P visualization technique by marking any of the following :

* most frequently word
* best length hist
* text length box plot

the user can mark more than one or using the default setting by unmarking the option .

The user can do all of I/p visualization options either by CMD or by GUI.

DEP : None

**3.3.1.5 Functional requirement 1.5**

**ID: FR5**

TITLE: most frequently word

DESC: display the most frequently word used in input date so the user can understand his own data, and what is the most used word in reviews

DEP :FR3

**3.3.1.6 Functional requirement 1.6**

**ID: FR6**

TITLE: Text length hist

DESC: display a histogram of the length of reviews so the user can know the range of reviews length and hence he can understand his own data.

DEP :FR3

**3.3.1.7 Functional requirement 1.7**

**ID: FR7**

TITLE: Text length boxplot

DESC: display a boxplot of the reviews length so the user can know the range of reviews length and hence he can understand his own data.

DEP :FR3

**3.3.1.8 Functional requirement 1.8**

**ID: FR8**

TITLE : o/p visualization

Description : the user choose how the output should be displayed and he has three options:

* Bar chart
* Pie chart
* histogram

the user can choose more than one option and he also can decide if he want to use a default options by unmarking any of the three options.

All of the above can be done either by command line or using graphical user interface .

**3.3.1.9 Functional requirement 1.9**

**ID: FR9**

TITLE: Bar chart

DESC: display bar chart of the output data so that the user can understand the data and the accuracy of our system.

DEP :FR8

**3.3.1.10 Functional requirement 1.10**

**ID: FR10**

TITLE: Pie chart

DESC: display Pie chart of the output data so that the user can understand the data and the accuracy of our system.

DEP :FR8

**3.3.1.11 Functional requirement 1.11**

**ID: FR11**

TITLE: histogram

DESC: display histogram of the output data so that the user can understand the data and the accuracy of our system.

DEP :FR8

**3.3 Performance requirements**

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.

**3.3.1 Usage of classification feature**

**ID: QR1**

TITLE: Usage of classification feature

DESC: The classification should be easy to use and straightforward to understand and also help the user to understand how data works.

RAT: For user convenience.

DEP: none

**3.3.2 Response Time**

**ID: QR2**

TAG: Response Time

GIST: The fastness of the classification

SCALE: Number of reviews classified per second for 2.4 GHz CPU

METER: Measurements obtained from 100 review during classification.

MUST: No more than 2 seconds 100% of the time.

WISH: No more than 1 second 100% of the time.

**3.4 Design Constraints**

**ID: QR3**

TAG: Application Memory Usage

GIST: The amount of Operate System memory occupied by the application.

SCALE: MB.

METER: Observations done from the performance log during classification.

MUST: No more than 20 MB.

PLAN: No more than 16 MB

WISH: No more than 10 MB

Operate System: DEFINED: The mobile Operate System which the application is running on.

MB: DEFINED: Megabyte.

**3.5 Software system attributes**

**3.5.1 Reliability**

**ID: QR4**

TAG: System Reliability

GIST: The reliability of the system.

SCALE: The reliability that the system gives the right classification for reviews.

MUST: More than 50% of the searches.

PLAN: More than 80% of the searches.

WISH: 100% of the searches.

**3.5.2 Maintainability**

**ID: QR5**

TITLE: Application extendibility

DESC: The application should be easy to extend. The code should be written in a way that it favors implementation of new functions.

RAT: In order for future functions to be implemented easily to the application.

DEP: none

**ID: QR6**

TITLE: Application modularity

DESC: Application should be built to allow using different pre-trained classification model.

RAT: In order to load different classifying models.

DEP: none

**3.5.3 Portability**

**ID: QR7**

TITLE: Application portability

DESC: The application should be portable with windows, Linux and mac.

RAT: The adaptable platform for the application to run on.

DEP: none

**4- Release Plan**

The requirements were divided into three releases based on the prioritization and their dependencies. The three different releases were assembled so that each would work as a fully functional application.

In the first release the requirements that build up the foundation of the application were included, together with the most highly prioritized requirements and their dependencies.

The second release also includes important requirements. However, these requirements are not vital for a functional application. They are more suited to act as additional features that can contribute to making the software product more attractive.

The third release includes the requirements that can be afforded to discard if the project gets delayed or overruns the budget

**Appendix I:**

table 1- Select of four most important requirements

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID | Ahmed  abd-el | Amr | hossam | Ahmed  hesham | Hossam | total |
| FR1 | 9 | 7 | 9 | 9 | 8 | 51 |
| FR2 | 3 | 6 | 4 | 7 | 7 | 27 |
| FR3 | 2 | 3 | 2 | 2 | 2 | 11 |
| FR4 | 8 | 7 | 8 | 9 | 5 | 47 |
| FR5 | 4 | 6 | 8 | 6 | 7 | 31 |
| FR6 | 3 | 4 | 6 | 2 | 3 | 18 |
| FR7 | 3 | 5 | 6 | 2 | 2 | 18 |
| FR8 | 10 | 7 | 9 | 8 | 9 | 43 |
| FR9 | 6 | 6 | 7 | 6 | 6 | 31 |
| FR10 | 6 | 7 | 5 | 8 | 6 | 32 |
| FR11 | 7 | 8 | 7 | 6 | 4 | 32 |
| QR1 | 6 | 7 | 8 | 3 | 8 | 34 |
| QR2 | 9 | 10 | 9 | 8 | 10 | 46 |
| QR3 | 3 | 6 | 9 | 4 | 6 | 28 |
| QR4 | 5 | 6 | 6 | 7 | 4 | 29 |
| QR5 | 7 | 7 | 3 | 2 | 9 | 28 |
| QR6 | 3 | 5 | 8 | 3 | 2 | 21 |
| QR7 | 1 | 4 | 6 | 7 | 4 | 22 |

**Appendix II:**

table 2 - four most important requirements

|  |  |  |
| --- | --- | --- |
| Requirement ID | Title | Requirement Type |
| FR1 | classify | Function |
| FR4 | I/p visualization | Function |
| FR8 | O/p visualization | Function |
| QR2 | Response Time | Quality |