Software Requirements Specification

for

Expenses Tracker Application

**Version 1.0 approved**

**Prepared by: Shubham Devkate ( 111508024)**

**Maheshkumar Kenre (111508039)**

**Pawankumar Hage (111508027)**

**College Of Engineering Pune**

**25/10/2017**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 2

1.5 References 2

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 3

2.3 User Classes and Characteristics 3

2.4 Operating Environment 4

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

**3. External Interface Requirements 5**

3.1 User Interfaces 5

3.2 Hardware Interfaces 5

3.3 Software Interfaces 5

3.4 Communications Interfaces 6

**4. System Features 6**

4.1 System Feature 1 6

4.2 System Feature 2 7

4.3 System Feature 3 8

**5. Other Nonfunctional Requirements 8**

5.1 Performance Requirements 8

5.2 Safety Requirements 8

5.3 Security Requirements 8

5.4 Software Quality Attributes 9

5.5 Business Rules 9

**6. Other Requirements 9**

**Appendix A: Glossary 9**

**Appendix B: Analysis Models 10**

**Appendix C: To Be Determined List 10**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Shubham Devkate  Mahesh Kendre  Pawan Hage | 26/10/2017 | The First Version | 1.0 |
|  |  |  |  |

# Introduction

## Purpose

At the instant, there is no as such complete solution present easily or we should say free of

cost which enables a person to keep a track of its daily expenditure easily. To do so a person

has to keep a log in a diary or in a computer, also all the calculations needs to be done by the

user which may sometimes results in errors leading to losses. Due to lack of a complete

tracking system, there is a constant overload to rely on the daily entry of the expenditure and total estimation till the end of the month.So to make all the jobs easy this software will help a lot.

## Document Conventions

*All the necessary libraries, software and hardware requirements have been represented by the* ***bold font****. Important terms in the explanation have been underlined so as to improve the readability. Main headers have been represented by a larger font than the normal text and has also been made bold. The entire document has been divided into comprehensible blocks.*

*Abbreviation used in this document are :*

*GUI - Graphical User Interface**t*

## Intended Audience and Reading Suggestions

As the name itself suggests,this project is an attempt to manage our daily expenses in a more efficient and manageable way. The system attempts to free the user with as much as possible the burden of manual calculation and to keep the track of the expenditure. Instead of keeping dairy or a log of the expenses on the smartphones or laptops, this system enables the user to not just keep the tab on the expenses but also to plan ahead keeping the past budget in mind. With the help of this system , user adds , delete ,change the current entered bill entry efficiently and also can attach the bill image (optional) for his/her own reminder.

*To get the maximum understanding of the product information, read section wise, in order. There is no special order for a specific domain of people. All potential users can go with the natural flow of this document, which is –*

*● Introduction*

*● Overall description*

*● External requirements*

*● System features*

*● Glossary*

*If you are not familiar with certain technical words, you must scan through the glossary (Appendix A) before beginning with the overall description.*

## Product Scope

The system we envisioned comes to practice, providing a centralized log inculcating all daily expenses. No or less manual calculations are required by the user, with efficient and user friendly interface. Apart from keeping a personal log, we are planning to extend this system to incorporate a shared expense group. We are planning to include a service so as to make the direct cash payment within the application itself.

## References

Slides put up at SacCt by Dr. Doan Nguyen for CSC230.

http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller

http://www.sparxsystems.com/platforms/software\_development.html

# Overall Description

## Product Perspective

The product is supposed to be an open source, under the GNU general Public License. It is a web based system implementing client-server model. The Aakash portal System provides simple mechanism for users to share and acquire knowledge.

The following are the main features that are included in Aakash Portal

* Cross platform support : Offers operating support for most of the known and commercial operating systems.
* User account : The system allows the user to create their accounts in the system and provide features of updating and viewing profiles.
* Number of users being supported by the system : Though the number is precisely not mentioned but the system is able to support a large number of online users at a time.
* Search : search is simply local search engine based on key words.
* Discussion Forum : Provides users with a platform to discuss and help each other with their problems
* Ticketing system : Allows user to submit his issue to the admin in case his problems are not solved by FAQs and discussion forums.
* FAQs section : Frequently asked section contain answer of problem which Aakash tablet user frequently faced.

## Product Functions

*The classifier mainly performs following functions:*

* Report Generation : Each and every entry is logged into the database and a user can view monthly or weekly report as per the requirement.
* Adding or deleting an entry : A user can easily modify each and every bill entry along with an option of attaching an image of the entry.
* Graphical representation : A lucrative statistical of the budget is shown to the user for easy understanding
* Add Friends : The system should add other people for collaborating the expenses with the user

## User Classes and Characteristics

*This software can be used by a wide range of people:*

1. *There are several options for handling expense report approval within Track Star. The default method is for managers to approve expense reports for employees they manage. Project Manager approval is an option as well.*
2. *The system can be deployed as an in-house solution mean this application or system can be used by various types of users for their expenses tracking*
3. *Also it will be helpful for the small scale industries or companies to keep track of money which they spend on the different departments*
4. *This system is useful for user class like student, managers, housewife, business oriented peoples class etc*

## Operating Environment

**Apache Cordova** (formerly **Ionic**) is a framework that allows developers to create cross-platform mobile applications using web technologies like **HTML, JavaScript or java and CSS.**

***Minimum Hardware Requirements:***

*Processor Intel® Pentium® D CPU 2.66 GHz*

*RAM 512 MB*

***Software Requirements:***

*Operating system : Windows (XP or newer) / Linux distribution (Mac/Ubuntu)*

## Design and Implementation Constraints

*The design document that we develop during this phase is the blueprint of the software. It*

*describes how the solution to the customer problem is to be built. Back and exit buttons are provided on each page for sake of convenience Error messages will be displayed appropriately. Widgets like calculator and date picker is there to help the user.*

*The design should not repeat the same thing. Systems are constructed using a set of design patterns, many of which have likely been encountered before. These patterns should always be chosen as an alternative to reinvention. Time is short and resources are limited! Design time should be invested in representing truly new ideas and integrating those patterns that already exist.*

## User Documentation

*User documentation refers to the documentation for a product or service provided to the end users. The user documentation is designed to assist end users to use the product or service. This is often referred to as user assistance. The user documentation is a part of the overall product delivered to the customer.*

*A user manual with some “ Getting started “ demonstrations on how to use the application will be included in the software. The manual itself or a link to the manual will be integrated to educate the users about the usage and make them aware of the various functionalities of the application.*

*Technical documentation refers to the documentation that describes how a product or service operates. For example, software code documentation, technical specifications and API documentation. user documentation was provided as a user guide instruction manual or help manual. However, user documentation is increasingly being delivered online today.*

## Assumptions and Dependencies

*Easy to be customized in future, as the client demand some other additional features. The complexity of customer’s company may be different or if mode of business changes then the system has capability to make appropriate modification to suite that change. Customization is key factor of designing this software.*

# External Interface Requirements

## User Interfaces

*-> System users :* The user logs on to the system by inserting Username and password, and can edit details inside the database such as adding or deleting the entry. Back and exit buttons are provided on each page for sake of convenience. Error messages will be displayed appropriately.

*Design of the GUI and the methods of its implementation are yet to be decided.* *Screenshots of the GUI will be attached here when it is developed.*

## Hardware Interfaces

*The system requirements are as mentioned in section 2.4.This application is intended to be stand alone. No hardware devices or interfaces are used.*

*MIC may be used in later versions of the software. No use of it currently.*

## Software Interfaces

***INPUT DATA***

*The input data is basically a “.mp3” file which has to be analysed to give the desired output.*

*The song is selected and opened for analysis with the help of a file manager.*

***OUTPUT DATA***

*After analysis, the output generated is the name of the genre and the language to which it belongs.*

***LANGUAGE USED***

*The language that will be used for the implementation of classification of music by genre using neural networks is python. It is the preferred language since :*

1. *It has many libraries that support large, multi-dimensional arrays and matrices.*
2. *It has a large collection of high-level mathematical functions to operate on these arrays.*

***LIBRARIES USED***

*The following are the python libraries used:*

* ***Librosa*** *for music and audio analysis. It provides the building blocks necessary to create music information retrieval systems. It is also used for Mel-transform( a method for sound processing).*

* ***Keras*** *for Deep Learning. It supports training of the Neural Network on a GPU.*

* ***Numpy*** *is the fundamental package for scientific computing with Python. It contains a powerful N-dimensional array object and useful linear algebra, Fourier transform, and random number capabilities.*

* ***Music21*** *toolkit for computer-aided musicology.It is used to study large datasets of music, to generate musical examples, to teach fundamentals of music theory, to edit musical notation and study music.*

* ***Tensorflow*** *is used to implement classification of songs by genre. It is an open source software library for machine learning which works in backend. It is primarily used for systems capable of building and training neural networks to detect patterns.*

## Communications Interfaces

*TBD*

# System Features

*This section illustrates the functional features of this product :*

## Genre Recognition

4.1.1 Description and Priority

*In this feature, the user will be able to recognize the genre of a song by giving an input file in order to process the data involved. The software can recognize one of the the following 12 genres: classic rock, folk, dance and electronica, jazz and blues, soul and reggae, punk, metal, classical, pop, hip-hop, country, instrumental. The dataset used for training the neural network is ‘Million song dataset’ or ‘MSD’. The priority of this feature is high in terms of benefit (rated 9 out of 10), high in terms of cost (rated 8 out of 10) and medium in terms of risk (rated 5 out of 10).*

4.1.2 Stimulus/Response Sequences

1. *Select the option ‘Know the Genre’ to stimulate the feature.*
2. *Select a valid .mp3 file from computer to be given as input from the Select file option.*
3. *Click on ‘Done’ and the output will be displayed.*

4.1.3 Functional Requirements

REQ-1: *The user must input a ‘valid’ .mp3 file without any discrepancies in the file type, format and extension it possesses. If any error is found in the input file given, the application will not be able to give any valid output and will give an appropriate error.*

REQ-2: *It is also required that the application is installed with all necessary libraries and modules. If not installed properly, the feature may not work and may not show any error message.*

REQ-3: *Internet Connection (TBD)*

## Language Recognition

4.2.1 Description and Priority

*This feature of the product will allow user to know or recognize the language of a song. For this, the user needs to give an input mp3 file to the software in order to process it and get the output. The software can recognize the following languages: English, French, German, English and Hindi. The dataset used for training the neural network is ‘Musixmatch dataset’. The priority of this feature is high in terms of benefit (rated 8 out of 10), high in terms of cost (rated 9 out of 10) and medium in terms of risk (rated 5 out of 10).*

4.2.2 Stimulus/Response Sequences

1. *Select the option ‘Know the language’ to stimulate the feature.*
2. *Select a valid .mp3 file from computer to be given as input from the Select file option.*
3. *Click on ‘Done’ and wait for the output to be processed.*

4.2.3 Functional Requirements

REQ-1: *The user must input a ‘valid’ .mp3 file without any discrepancies in the file type, format and extension it possesses. If any error is found in the input file given, the application will not be able to give any valid output and will give an appropriate error.*

REQ-2: *It is also required that the application is installed with all necessary libraries and modules. If not installed properly, the feature may not work and may not show any error message.*

REQ-3: *Internet Connection (TBD)*

## Playing the song

4.3.1 Description and Priority

This function will allow user to listen to the song while the output is being processed. The song can be muted/unmuted based on user preference. *The priority of this feature is medium in terms of benefit (rated 4 out of 10), medium in terms of cost (rated 5 out of 10) and medium in terms of risk (rated 5 out of 10).*

4.3.2 Stimulus/Response Sequences

1. *After uploading the file and clicking on ‘Done’ button, the song starts playing for a few seconds.*
2. *User can mute the song by clicking on the mute icon and unmute by clicking on it again.*
3. *User can adjust volume from keyboard or use system functionality.*

4.3.3 Functional Requirements

REQ-1: *The user must input a ‘valid’ .mp3 file without any discrepancies in the file type, format and extension it possesses. If any error is found in the input file given, the application will not be able to play the song and will give an appropriate error.*

REQ-2: *The time period of song must be more than 30 seconds. If not, an error will be generated.*

REQ-3: Appropriate output device for music to be played.

# Other Nonfunctional Requirements

## Performance Requirements

*The software will must use a reliable, effective ,efficient and an error proof algorithm. The processing time must be kept to a minimum but it should not neglect the correctness of the output.*

*There should be no loss of data and under optimum operating conditions, the program should not crash.*

## Safety Requirements

*The format of the file to be analyzed should be “.mp3” for the effective functioning of the program.*

*If the format of the file is unsupported , it may lead to the corruption of the file or crashing of the software itself.*

*In case the software takes too long to analyze the file or program does not respond , a manual shutdown will interrupt the process and shut down the application.*

## Security Requirements

*The product is open to all users. No authentication of the user will be required to access the functions.*

## Software Quality Attributes

***Availability****: It should be available at all times after installation in the desktop and works without an*

*internet connection.*

***Maintainability****: It should be easy to maintain and operate. No additional software is required for its functioning except the music player.*

***Portability****: Easily portable across Linux and MacOS platforms. Windows will be soon supported.*

***Reliability****: The event of failure of the software should be very rare. It should have a very high success rate in terms of execution. Probability of correctness may vary due to some circumstances.*

***Reusability****: User can run the software as many times as he/she wishes to identify songs. Reusability of the code for further development in terms of language support, artist recognition etc. is also possible.*

***Testability****: Since this software uses neural networks , testing is an essential part to ensure the output is correct. Testability of the software should be high and hence finding faults would be easier.*

***Usability****: Easy to use and learn the functions , very basic and user friendly interface.*

## Business Rules

*NA*

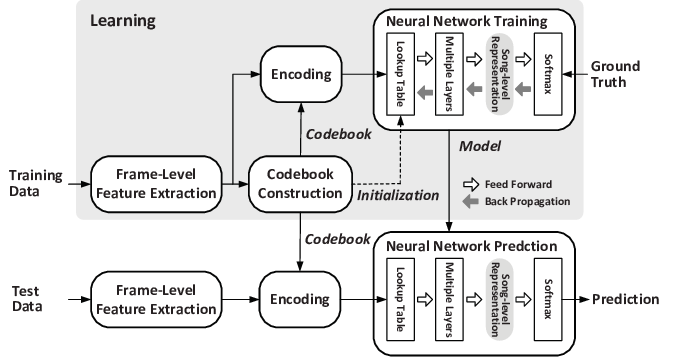
# Other Requirements

*All requirements have been mentioned above.*

**Appendix A: Glossary**

|  |  |
| --- | --- |
| *Genre* | *A* ***music genre*** *is a conventional category that identifies some pieces of music as belonging to a shared tradition or set of conventions. Example : Jazz , Rock, Hip-Hop etc.* |
| *Datasets* | *a collection of related sets of information that is composed of separate elements but can be manipulated as a unit by a computer.* |
| *Neural Networks* | *A neural network is a series of algorithms that attempts to identify underlying relationships in a set of data by using a process that mimics the way the human brain operates. Neural networks have the ability to adapt to changing input so the network produces the best possible result without the need to redesign the output criteria.* |
| *Sampling Rate* | *The* ***rate*** *at which samples of an analog signal are taken in order to be converted into digital form.* |
| *Bag Of Words* | *The* ***bag-of-words*** *model is a simplifying representation used in natural language processing and information retrieval (IR). In this model, a text (such as a sentence or a document) is represented as the* ***bag*** *(multiset) of its* ***words****, disregarding grammar and even* ***word*** *order but keeping multiplicity.* |
| *Deep Learning* | ***Deep learning*** *is a subset of machine learning in AI that has networks which are capable of learning unsupervised from data that is unstructured or unlabeled.* |

**Appendix B: Analysis Models**



*The flow of data and the exact process of analysis of the song has also been mentioned in section 2.2(Product functions).*

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*

***References:***

[*http://deepsound.io/music\_genre\_recognition.html*](http://deepsound.io/music_genre_recognition.html) *(for training of neural networks, production of spectrograms)*

*Posted by Piotr Kozakowski & Bartosz Michalak on Wed 26 October 2016*

[*https://medium.com/towards-data-science/how-to-build-a-simple-song-recommender-296fcbc8c85*](https://medium.com/towards-data-science/how-to-build-a-simple-song-recommender-296fcbc8c85) *(For classification of genres)*

*Posted by Eric Le on 24 April 2017*

[*https://labrosa.ee.columbia.edu/millionsong/blog/11-4-11-musixmatch-dataset-connecting-lyrics*](https://labrosa.ee.columbia.edu/millionsong/blog/11-4-11-musixmatch-dataset-connecting-lyrics)

*(Musixmatch dataset)*

*Posted by millionsong on Mon, 04 November 2011*

[*http://neuroph.sourceforge.net/tutorials/MusicClassification/music\_classification\_by\_genre\_using\_neural\_networks.html*](http://neuroph.sourceforge.net/tutorials/MusicClassification/music_classification_by_genre_using_neural_networks.html) *( Idea and Concept of the project)*

*Posted by by Marina Jeremić, Faculty of Organizational Sciences, University of Belgrade*

[*https://pdfs.semanticscholar.org/9143/9bd130b608cf5012a84cdc517e636a57221f.pdf*](https://pdfs.semanticscholar.org/9143/9bd130b608cf5012a84cdc517e636a57221f.pdf)

*(Research paper on Convolutional neural networks)*

*Posted by Thomas Lidy ,Vienna University of Technology and Alexander Schindler,*

*Austrian Institute of Technology.*

<https://chatbotslife.com/finding-the-genre-of-a-song-with-deep-learning-da8f59a61194>

*( Finding genre of the song using Deep Learning)*

*Posted by Julien Despois ,Nov 21, 2016.*

[*https://courses.engr.illinois.edu/ece544na/fa2014/Tao\_Feng.pdf*](https://courses.engr.illinois.edu/ece544na/fa2014/Tao_Feng.pdf) *(Deep learning for music genre classification)*

*Posted by Tao Feng ,University of Illinois*

[*https://github.com/mlachmish/MusicGenreClassification*](https://github.com/mlachmish/MusicGenreClassification) *( Deep learning and sound processing)*