**THE SUPERIOR COLLEGE LAHORE**

|  |
| --- |
| chk_captcha |

**Faculty of Computer Science & IT**

**Department of Software Engineering**

**Final Year Project**

**PROJECT REPORT (Part-1)**

**Detecting Influential Nodes Using Multilayered Social Network Analysis (SNA)**

Project ID: **[write ID here Issued by FYP Manager]**

**Project Team**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student Name** | **Student ID** | **Program** | **Contact Number** | **Email Address** |
| Zainab Akhter | Bcsm-s17-047 | BSCS | 03058650804 | Zainab.akhter661@gmail.com |
| Husnain Ashraf | Bcsm-s17-061 | BSCS | 03486618536 | ali.husnain5827@gmail.com |
| Fahad Rajput | Bcsm-s17-021 | BSCS | 03164254105 | Fahadrajput115@gmail.com |
|  |  |  |  |  |
|  |  |  |  |  |

**Project Supervisor: Jr. Lec Muhammad Kaleem Ullah**

**Designation: Jr. Lec**

**Project Report**

**Detecting Influential Nodes Using Multilayered Social Network Analysis (SNA)**

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Author(s)** | **Version** | **Date** | **Notes** | **Supervisor’s Signature** |
|  | 1.0 |  | <Original Draft> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  | <Changes Based on Feedback From Faculty> |  |
|  |  |  | <Added Project Plan> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**APPROVAL**

|  |  |
| --- | --- |
| **Project Supervisor** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |

|  |  |
| --- | --- |
| **Project Manager** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

|  |  |
| --- | --- |
| **Head of the Department** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Dedication

*This work is dedicated to my . . . . . .*

# Acknowledgements

I am really thankful to my supervisor who has . . . . . . . . . .

# Executive Summary

[12 pt, Calibri, Justified]

[*An executive summary summarizes a longer report or proposal or a group of related reports in such a way that readers can rapidly become acquainted with a large body of material without having to read it all.* *This section summarizes the overall document, and should include the important highlights from the document. It should be concise. It is NOT an introduction, index or table of contents, it is a summary. The Executive Summary should not make any reference to other parts of the document. You have to write one page to let reader understand an overview of the project.]*

# Table of Contents

[Dedication iv](#_Toc505638416)

[Acknowledgements v](#_Toc505638417)

[Executive Summary vi](#_Toc505638418)

[Table of Contents vii](#_Toc505638419)

[List of Figures ix](#_Toc505638420)

[List of Tables x](#_Toc505638421)

[Chapter 1 1](#_Toc505638422)

[Introduction 1](#_Toc505638423)

[1.1. Background 2](#_Toc505638424)

[1.2. Motivations and Challenges 2](#_Toc505638425)

[1.3. Goals and Objectives 2](#_Toc505638426)

[1.4. Literature Review/Existing Solutions 2](#_Toc505638427)

[1.5. Gap Analysis 2](#_Toc505638428)

[1.6. Proposed Solution 2](#_Toc505638429)

[1.7. Project Plan 3](#_Toc505638430)

[1.7.1. Work Breakdown Structure 3](#_Toc505638431)

[1.7.2. Roles & Responsibility Matrix 3](#_Toc505638432)

[1.7.3. Gantt Chart 3](#_Toc505638433)

[1.8. Report Outline 3](#_Toc505638434)

[Chapter 2 4](#_Toc505638435)

[Software Requirement Specifications 4](#_Toc505638436)

[2.1. Introduction 5](#_Toc505638438)

[2.1.1. Purpose 5](#_Toc505638439)

[2.1.2. Document Conventions 5](#_Toc505638440)

[2.1.3. Intended Audience and Reading Suggestions 5](#_Toc505638441)

[2.1.4. Product Scope 5](#_Toc505638442)

[2.1.5. References 6](#_Toc505638443)

[2.2. Overall Description 6](#_Toc505638444)

[2.2.1. Product Perspective 6](#_Toc505638445)

[2.2.2. Product Functions 6](#_Toc505638446)

[2.2.3. User Classes and Characteristics 6](#_Toc505638447)

[2.2.4. Operating Environment 7](#_Toc505638448)

[2.2.5. Design and Implementation Constraints 7](#_Toc505638449)

[2.2.6. User Documentation 7](#_Toc505638450)

[2.2.7. Assumptions and Dependencies 7](#_Toc505638451)

[2.3. External Interface Requirements 8](#_Toc505638452)

[2.3.1. User Interfaces 8](#_Toc505638453)

[2.3.2. Hardware Interfaces 8](#_Toc505638454)

[2.3.3. Software Interfaces 8](#_Toc505638455)

[2.3.4. Communications Interfaces 9](#_Toc505638456)

[2.4. System Features 9](#_Toc505638457)

[2.4.1. System Feature 1 9](#_Toc505638458)

[2.4.1.1. Description and Priority 9](#_Toc505638459)

[2.4.1.2. Stimulus/Response Sequences 9](#_Toc505638460)

[2.4.1.3. Functional Requirements 9](#_Toc505638461)

[2.4.2. System Feature 2 10](#_Toc505638462)

[2.4.2.1. Description and Priority 10](#_Toc505638463)

[2.4.2.2. Stimulus/Response Sequences 10](#_Toc505638464)

[2.4.2.3. Functional Requirements 10](#_Toc505638465)

[2.4.3. System Feature 3 (and so on) 11](#_Toc505638466)

[2.5. Other Nonfunctional Requirements 11](#_Toc505638467)

[2.5.1. Performance Requirements 11](#_Toc505638468)

[2.5.2. Safety Requirements 11](#_Toc505638469)

[2.5.3. Security Requirements 12](#_Toc505638470)

[2.5.4. Software Quality Attributes 12](#_Toc505638471)

[2.5.5. Business Rules 12](#_Toc505638472)

[2.6. Other Requirements 12](#_Toc505638473)

[Chapter 3 13](#_Toc505638474)

[Use Case Analysis 13](#_Toc505638475)

[3.1. Use Case Model 14](#_Toc505638476)

[3.2. Fully Dressed Use Cases 14](#_Toc505638477)

[Chapter 4 15](#_Toc505638478)

[System Design 15](#_Toc505638479)

[4.1. Architecture Diagram 16](#_Toc505638480)

[4.2. Domain Model 16](#_Toc505638481)

[4.3. Entity Relationship Diagram with data dictionary 16](#_Toc505638482)

[4.4. Class Diagram 17](#_Toc505638483)

[4.5. Sequence / Collaboration Diagram 17](#_Toc505638484)

[4.6. Operation contracts 17](#_Toc505638485)

[4.7. Activity Diagram 18](#_Toc505638486)

[4.8. State Transition Diagram 18](#_Toc505638487)

[4.9. Component Diagram 18](#_Toc505638488)

[4.10. Deployment Diagram 19](#_Toc505638489)

[4.11. Data Flow diagram [only if structured approach is used - Level 0 and 1] 19](#_Toc505638490)

[Chapter 5 20](#_Toc505638491)

[Implementation 20](#_Toc505638492)

[5.1. Important Flow Control/Pseudo codes 21](#_Toc505638493)

[5.2. Components, Libraries, Web Services and stubs 21](#_Toc505638494)

[5.3. Deployment Environment 21](#_Toc505638495)

[5.4. Tools and Techniques 22](#_Toc505638496)

[5.5. Best Practices / Coding Standards 22](#_Toc505638497)

[5.6. Version Control 22](#_Toc505638498)

[Appendices 23](#_Toc505638499)

[Appendix A: Information / Promotional Material 24](#_Toc505638500)

[Reference and Bibliography 27](#_Toc505638502)

[Index 29](#_Toc505638503)

# List of Figures

1.1 Caption of first figure of first chapter 6

1.2 Caption of second figure of first chapter 7

2.1 Caption of first figure of second chapter 14

2.2 Caption of second figure of second chapter 22

2.3 Caption of third figure of second chapter 26

5.1 Caption of first figure of fifth chapter 49

5.2 Caption of second figure of fifth chapter 49

# List of Tables

1.1 label of first table of first chapter 6

1.2 label of second table of first chapter 7

2.1 label of first table of second chapter 14

2.2 label of second table of second chapter 22

2.3 label of third table of second chapter 26

5.1 label of first table of fifth chapter 49

5.2 label of second table of fifth chapter 49

# Chapter 1

# Introduction

**Chapter 1:** Introduction

This chapter is all about basics and pre inputs of our project. In this chapter we will discuss about our project’s important aspects these are background of the project, motivations, challenges that we will face through out the processing state, our main goal and objectives, we will discuss deeply about the existing solutions, Gap Analysis that tells us to get from our current state to our desired state then we will discuss the proposed solution our methodology that will tell us what are benefits of our methodology who will get the real benefit through it solution to the problem and way to implement it, through the Project plan, WBS, RACI and Gantt we will dicuss the Performance, Responsibilities and Roles of the project team. In a nutshell we will give the Project Report.

## Background

Like many other fields, Online Business Community have attracted huge amount of research,

especially from Information Technologies and networks due to various reasons. First of all Online Businesses have climbed up the graph since the recent years because of the increasing number of digital buyers across the globe. The digital buyers had increased from 1.32 Billion to 1.92 Billion in just few years i.e. 2014 to 2019 respectively. According to researchers the amount is expected to rise to $2.14 Billion at the end of 2021. Companies nowadays are using the social network analysis tools and techniques to enhance their business positions. Analyzing the social networks provide the massive amount of useful information for the business. The applications of Social Network Analysis with respect to Business include Recommender System, Churn Prediction, Target Marketing and many more. Because of this, there is a room for effective research in order to contribute in the area of online business communities. In the recent years, the business communities have started using the SNA methodologies to enhance their business by detecting the key Players, developing the marketing campaigns, detecting behaviors of their customers, predicting the customer’s churns, developing recommender systems etc.

## Motivations and Challenges

Thing which motivates us is that fact, people who are hardworking and doing amazing work but they don’t know how to spread their little business and they don’t know how to make more money by their business. The major Challenge here is to make people able to use our methodology, some people do not have much exposure to smartphones/Laptops so it is a challenge for us to make people use our methodology and make them believe that it is for their benefit. We have to market our methodology wisely to get most out of it. By using our this our users can enhance their business by detecting the key Players, developing the marketing campaigns, detecting behaviors of their customers, predicting the customer’s churns, developing recommender systems etc.

## Goals and Objectives

The aim of this project is to perform social network analysis on multi layered networks to detect Influential nodes which can help marketing community in improving their businesses. Our ambition is to connect our society’s skill full people to the bigger market and outer world, we have so many people in our country who have skills but they cannot get best out of it. Our goal is to provide a better life to those who actually do work but did not get most of the credit. Those people need to be in limelight. They cannot afford a fancy shop or marketing plan to sell their products on a broader scale so we are focusing on that part of our society to strengthen them.

**Objectives**

* To gather social network datasets
* To design a generic feature vector for cross network analysis
* To develop classification techniques for analysis (decision tree, random forest, id3, naive bayes Algorithms)
* To extract influential nodes using social network analysis

## Literature Review/Existing Solutions

A number of researchers had worked on the social network analysis for different kind of the

applications. Kimura Metal presents a cascaded method for detection of most key players

in online social networks. Authors suggested different algorithms and techniques that extract

network parameters which play a very vital role in detecting the key players in social network.

Based on their statistical analysis we can say, their proposed algorithm is highly recommended

than the greedy solution of the Leicht M et al and Newman community structure algorithm.

Salvatore Catanese et al describes the friendship relation between users and how to access

data of Facebook users. For achieving the purpose they used the web crawlers to access data

directly from the website. By using gathered data and information the author constructed a sub

group graph that represents the anonymous relationship among a sub group of significant users. An adhoc privacy complaint crawlers is studied to extract data from Facebook. Rejection sampling methods and Breadth-first-search (BFS) are used to minimize the biasness and for

visualizing the structural characteristics of different samples that consist of huge no. of key

players. Authors developed a visualizing tool for analysis of qualitative and quantitative

characteristics of social media. To achieve the results they improved the efficiency of existing

online social media analysis (OSN) techniques and adopted existing techniques methods and

algorithms. Pasquale De Moeetal describes that how to analyze the behaviors of new users

to predict whether the two nodes could be considered as a similar one. Author proposed a

framework where the estimation can be performed to verify and check the similarities of two

users that is based on visualizing the different activities that includes social events in which users are fully involved and on the basis of information of social relation i.e. common groups of users and friends. Leidys del Carmen measured the clustering and association rules with famous CRISP. DM method to analyze the behaviors of the customers of the fashion industry in Instagram social network, which provided the industry with the handful of important information regarding their products and their trends and likeness among their customers.Numerous related research discussed the presence of influential nodes in a particular social networks (e.g. Facebook, Twitter, Micro blog). Cha et al extracted Twitter data and did analysis to find the influence of Twitter users by comparing the network metrics values of Clustering Coefficient, Degree, Eigenvector Centrality and retweets. Rossi, Vazirgiannis and Malliaros proposed a framework that visualize the complex social networks and detect influential nodes in a network. Authors proposed a famous technique i.e. K-truss composition method that helps in visualizing and analyzing the social networks and detection of Influential nodes.

## Gap Analysis

In today’s modern world, everyone is competing to achieve their target, but there is a gap between their target and achievement. A Gap Analysis helps a project team to understand what they want to be, the gap between where they are now and where they want to be and hence, what steps should be taken to close the gap.

**Current State**

In our project we are at the initial state of the performance we are doing deeply research about the whole project this is our current state where we are now.

**Gap Analysis( How do we do that?/ What do we need to do to get there? )**

Our Proposed schema is to detect the Key Players with the graphic matrix by using python language then apply the classification methods for the evaluation and detection of the key Player our schema is divided into 4 phases.

1. Social Network Selection (Dataset)

2. Pre-Processing of the Data

3. Visualization and Feature Selection

4. Classification

**Desired State (Where we want to go?)**

Finally we will reach to the final state that is our Desired State. The aim of this project is to perform social network analysis on multi layered networks to detect Influential nodes which can help marketing community in improving their businesses.

## Proposed Solution

Our Proposed schema is to detect the Key Players with the graphic matrix by using python language then apply the classification methods for the evaluation and

detection of the key Player our schema is divided into 4 phases.

1. Social Network Selection (Dataset)

2. Pre-Processing of the Data

3. Visualization and Feature Selection

4. Classification

## Project Plan

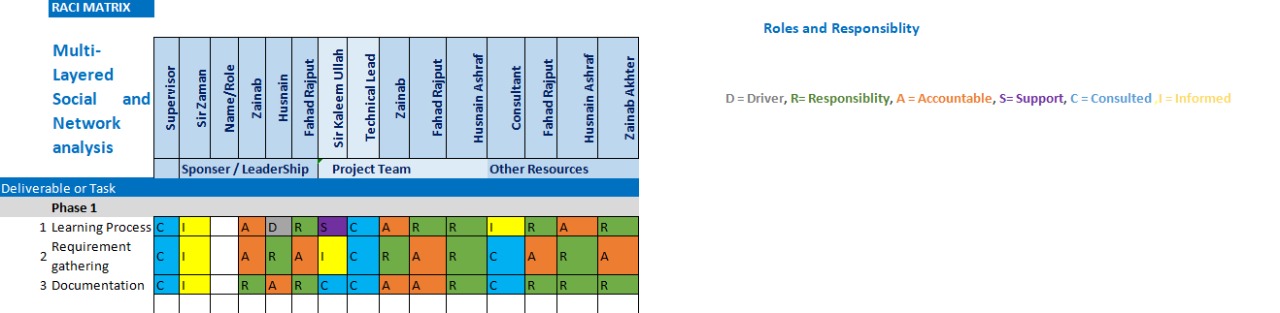
Data Gathering is the first phase then our next phase is Data Preprocessing (Selection of AI algorithms) then we will perform Training and Testing.

## Work Breakdown Structure

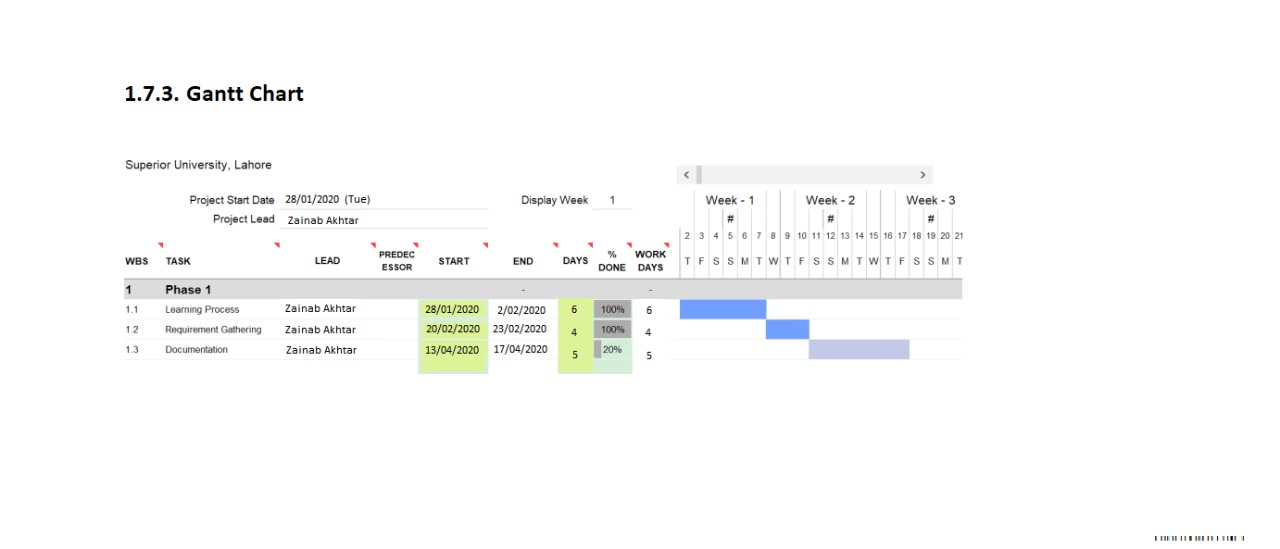
We have Break down all the processes involved in the development of our project to understand the project in a better way and to develop the project on these guidelines, after all the processes we will combine them to form our functional result.



## Roles & Responsibility Matrix



## Gantt Chart



## Report Outline

Documentation is currently in progress.

# Chapter 2

# Software Requirement Specifications

**Chapter 2:** Software Requirement Specifications



## Introduction

## Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>

## 

## Document Conventions

<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>

## 

## Intended Audience and Reading Suggestions

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

## Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>

## References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

## Overall Description

## Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

## Product Functions

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>

## User Classes and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

## Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

## Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).>

## User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

## Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

## External Interface Requirements

## 

## User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

## 

## Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

## Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

## Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

## System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## System Feature 1

<Don’t really say “System Feature 1.” State the feature name in just a few words.>

## Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

## Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

## Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-SF1-1: <Write your requirement here>

REQ-SF1-2:

REQ-SF1-3:

## System Feature 2

<Don’t really say “System Feature 1.” State the feature name in just a few words.>

## Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

## Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

## Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-SF2-1:

REQ-SF2-2:

REQ-SF2-3:

## System Feature 3 (and so on)

## Other Nonfunctional Requirements

## Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

## Safety Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>

## Security Requirements

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

## Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

## Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

# Chapter 3

# Use Case Analysis

**Chapter 3:** System Analysis

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

[*Between 4 to 8 lines describe what is this chapter all about*]

## Use Case Model

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Fully Dressed Use Cases

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

# Chapter 4

# System Design

**Chapter 4:** System Design

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

[*Between 4 to 8 lines describe what is this chapter all about*]

## Architecture Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Domain Model

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Entity Relationship Diagram with data dictionary

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Class Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Sequence / Collaboration Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Operation contracts

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Activity Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## State Transition Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Component Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Deployment Diagram

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Data Flow diagram [*only if structured approach is used - Level 0 and 1*]

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

# Chapter 5

# Implementation

**Chapter 5:** Implementation

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

[*Between 4 to 8 lines describe what is this chapter all about*]

## Important Flow Control/Pseudo codes

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Components, Libraries, Web Services and stubs

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Deployment Environment

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Tools and Techniques

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Best Practices / Coding Standards

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

## Version Control

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

# Appendices

# Appendix A: Information / Promotional Material

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

[*Between 4 to 8 lines describe what is this appendix all about*]

* 1. **Broacher**

* 1. **Flyer**
  2. **Standee**
  3. **Banner**
  4. **First Level heading [16 pt, Calibri, Bold, Left aligned]**

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

* + 1. **Second level heading [14 pt, Calibri, Bold, Left aligned]**

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

* + - 1. **Third level heading [12 pt, Calibri, Bold, Left aligned]**

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

# Appendix [no.]: Appendix Title

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

[*Between 4 to 8 lines describe what is this chapter all about*]

* 1. **First Level heading [16 pt, Calibri, Bold, Left aligned]**

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

* + 1. **Second level heading [14 pt, Calibri, Bold, Left aligned]**

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

* + - 1. **Third level heading [12 pt, Calibri, Bold, Left aligned]**

[Paragraph Text 12 pt, Calibri, 1.5 Line Spacing, Justified]

# Reference and Bibliography

**Reference and Bibliography**

[1] M. Sher, M. Rehman, “*Title of the Paper*” Conference name/Journal Name, Edition, Volume, Issue, ISBN/ISSN, PP, Publisher/City-Country, Year.

[2] ……

# Index

**Index**

**[A]**

**[B]**

**[C]**