

Software Requirements Specification

0. Preface

The following SRS document is based on the project of Contact Manager. It starts with the introduction of project specifying its context to for what it is used for, problem specification and scope listing the functionalities of the web application. Then structure of document is summarised. Then there is description about the workflow, use-cases and the design. Functional, interface and performance requirements are also mentioned. It also specifies the constraints of our design and how we will work on non-functional requirements. Then at last their budget presentation and schedule and at last there additional materials.

1. Introduction

1.a Context

The following project is about web based application named as '*Contact Manager*'. While building this project we are following software development guidelines. We have selected a model of software development life cycle and according to that we are proceeding. Project is scheduled and all the timelines are decided for each component of the project.

1.b Problem Specification

It generally happens that our contact list in our phones keep on increasing and we are not able to manage them properly. Sometimes our sim card losses and we loss all our contacts with that sim card. Most of us do not keep backup for contacts. Or we want a place where we could get all the details of a particular contact be it email, phone number, photo and description of the entity. For all this we have come up with an app Contact Manager so that next when users will use this app they will remain stress free about their contacts.

1.c Scope of the document

The documents list all the requirements be it functional or non functional of this web based app. It ensures to cover the solution of all the aspects that were discussed in Problem specification. Scope of the app is to cover the following:

- Contact Manager helps users to keep it's contacts safe and at the same time they are easy to use and manage. User has to enter id and password to login. First user has to create a new account by signing up, if he/she is a new user.

- User can go to the particular contact and at that place he/she will get all the info about the contact be it user's phone number,email,profession,and details.Each contact has phone links(when clicked in mobile app calls the contact) and mail links(when clicked ,the user can directly write mail to the contact).
- User has no need to save each and every contact in this app.
- The contacts can be easily imported and exported from the phone.
- It allows you to update or delete your contacts whenever user wants.
- The app is compatible and can run on various system such as android,web,desktop,mac os etc,.

1.d Summary/overview of the document and its structure

The document ensures that the reader gets a full idea about the project and clarity on the use cases,requirements,workflow and design.It ensures that user gets answers of all such questions. Be it why to use this app?Where to use?What are its features?How to use?The document also attaches the schedule and approximated budget.We have planned the project in such a way such that client gets fully satisfied with the product.Requirements are kept flexible and are dependent on client.These could be changed with client's consent.The document covers each and every aspect in detail such that any new employee if joins the team, he/she is able to understand the ongoing project and could start the work as soon as possible.

2.General Description

2.a.Workflow

In the workflow we will explain that what are the series of steps that would be taken to complete the project.

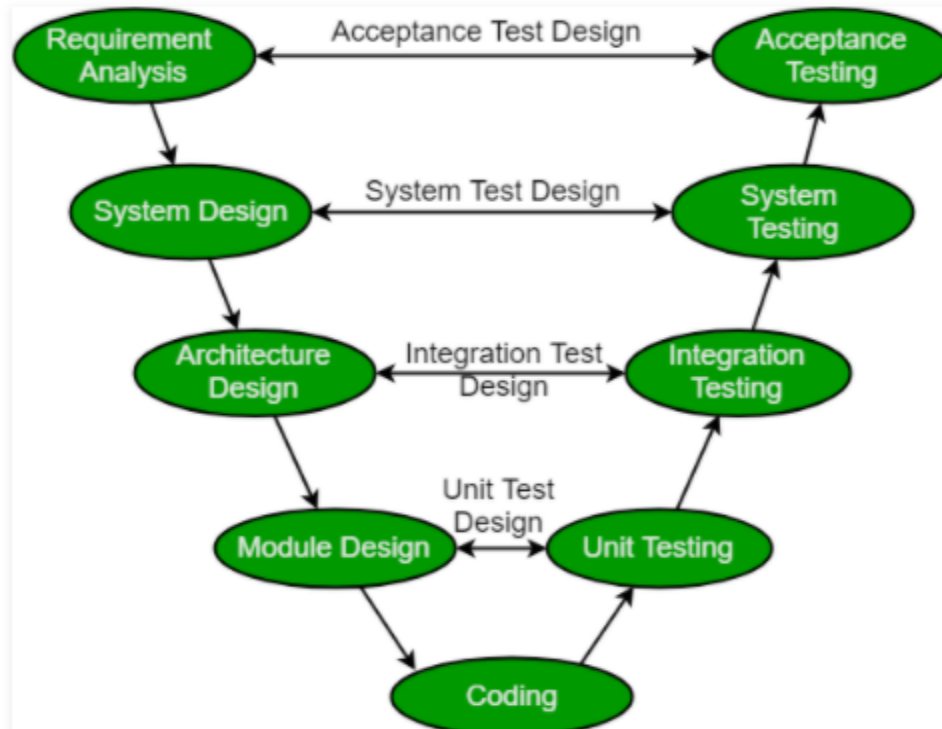
Our project is based on the **V model of sdlc**.

The project building starts with the requirements gathering that are listed further in this document.Then comes system design in which preplanning of the software is done and basic approach is discussed.Programming language is chosen and framework is decided.The software for coding is installed.Then architectural design is specified.Different modules are formulated and each module is assigned a task.After working on high level design we move to Module design also referred as low level design.Different tasks are distributed among developers.Internal design of modules is discussed and then the Coding phase starts.After the implementation unit testing of all the modules is performed.

Modules are integrated and integration testing is performed. Then we move to system testing in which we check the entire system functionality and the communication of system under development with external systems.

And the final phase is acceptance testing which involves the testing of the product in user environment.

The process workflow diagram is given below:



2.b Use cases/User scenarios

The different use cases which will be covered by this app are:

- Makes easier to search the contacts using the search bar that will be installed in the app.
- Let's user to find the email of the contact and redirects the user directly to the gmail page where he/she could write the e-mail to that contact.
- User login functionality so that only the user have access to his/her contacts. User have to enter the id and password for login into his/her account.
- Contacts can be imported/exported from the app so that user does not have to enter each and every contact on it's own.
- User can also find details of the contact at the same place.
- Has a strong dynamic back-end so that any problem faced at the user end could be solved quickly and easily.

2.c Overall description of the software product

This web based application is decided to made using django a python framework. There will be both front end and back end. The app will have dynamic back end. The product runs on web, android, desktop and almost all the platforms. The app will have login functionality, search bar, contact which will include the person's phone number, email, profession, person's photo and other details. The app will also contain user profile which will include all the related information which is required from the user end. User can any time change it's password as and when required. The user has to enter it's id and password each time he/she uses the app. User can import contacts from the phone to save the contacts into the app. If by chance user has lost his/her sim card then he/she could also export the contacts from the app. The app has appealing UI and ensures it's quality. It would be maintained after the deployment and user will get regular updates of the app which will be add on of new features.

3. Functional Requirements

The requirements which are listed by the clients are mentioned below:

- While creating the account it should give suggestions for the strong password. And for the existing user, if by chance user has entered wrong password or id it should display "*wrong id or password*".
- This app should save the contacts of the user with each contact having details of email, profile photo, phone number, profession and other related info.
- A search bar to navigate to a particular contact.
- It should redirect the user to the user's compose mail when the user clicks on the mail of the contact.
- User's profile should contain all the information about the user and he/she should have access to change this as and when required.
- User should be able to import/export the contacts from the phone to web app and vice versa.

4. Interface Requirements

The user can interact with the software on multiple platforms. These could be smartphones, web, desktop etc,. This app is mainly a web based app so it's interface is designed in such a way that it's web look is given the first priority. However while app designing it would be kept in mind that it maintains it's quality on other platforms as well. On the web it would click-based and keyboard whereas it would be touch-based on the smartphones.

Display requirements from the user side are:

- The background should be white or any other light color.
- At the home page itself there should be the list of all the contact in rectangular boxes.
- Each contact's name and photo should only appear at the home page.
- All contacts at the home page should have fine spacing between them.
- Other details are shown after the user clicks on the particular contact.
- After clicking on a contact ,a new page should appear displaying only one contact's information.
- At one of the sides it should have options which include 'Home' , 'User profile' , 'help'.
- At the end of the page it could display the company's website,email,phone number etc.,.

5.Performance Requirements

Here user expectations with the app would be listed.These are:

- **Start time**-The application should display the opened document within 10s after it is started.
- **Edit Response Time**-The application should display updated values within 1s after user triggers the edit operation.
- **Smooth Scrolling**-While a user scrolls the requirements table, the application should not display scrolling jerks longer than 100ms.
- **Document Size**-The application shall allow users to open documents up to 10000 objects with total size up to 50MB.
- **Logical database requirements**-The application shall encrypt the persistent application data.

6.Design Constraints

- **Programming language to be used**- HTML,Javascript and CSS will be used for developing the front end specifically.Rest all is implemented using django framework of python.
- **IDE**- Any ide which supports python,html,javascript and css could be used.For this case one could use vs code.
- **Execution environment**-It includes operating systems such as android,mac os and systems such as web or desktop on which this application could run.
- **User interface needs**-This would include all the interface requirements such as light background,contact size,dropdown menu on the side etc.,.

- **Performance expectation**-It would include all the performance requirements such as small size,smooth scrolling,high capacity etc,.

7.Non-functional requirements

- **Compatibility**-Should be able to run on various platforms such as android,mac os etc,.And it should run smoothly on a device along with other applications.
- **Usability**-Should have high learnability,accessible to diverse population,should have appealing UI,helps user from avoid making any error and appropriate recognizability.
- **Reliability**-All requirements should be match with the software,should be available 24-7 hours and should be able to recover from the software failure.
- **Security**-The data should be highly confidential and outside users should not be able to access the internal data.It should be non-repudiated,accountable and authentic to the user.

7.a Measures

We will use the following metrics to relatively measure the above mentioned quality attributes.

- **Compatibility test:**

Devices including laptops,pc,smartphones	YES
Operating systems including Windows,Android and mac OS	YES
Browsers including google chrome,internet explorer,Mozilla Firefox etc,.	YES
Could run on all types of ISPs	YES

Therefore the device has 100% compatibility.

- **Usability test:**

Any demo or guidelines	No
------------------------	----

Available in different languages	No
Font size and UI appealing to user	Yes
Easy to operate and Control	Yes
Error protection like it shows incorrect password entered when user has entered wrong password.	Yes
Appropriate recognizability	Yes

Therefore the device has 66.67%

- **Reliability test:**

Does the software meets all requirements	Yes
Does the software is available 99.99% of the time	Yes
Does the software has ability to tolerate any defect or error if it comes	No
Will the software be able to recover if any error comes or site crashes	Yes

Therefore the device has 75% reliability.

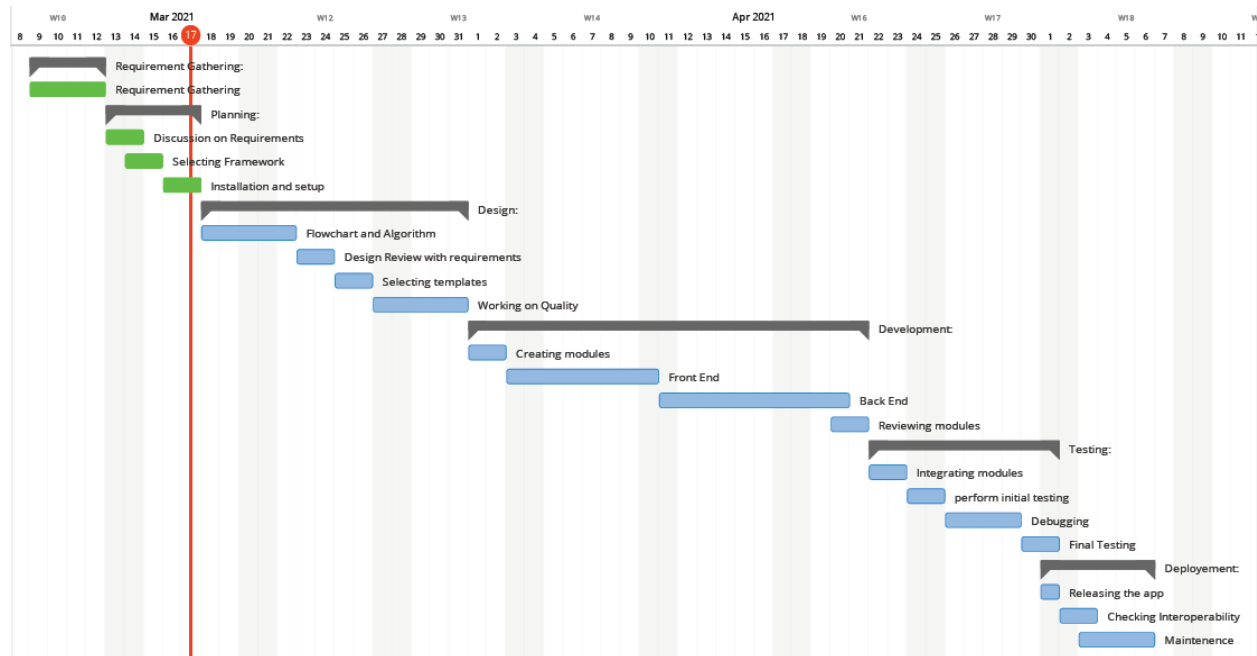
- **Security test:**

Does users have access to account of other users	No
Is the software non-repudiated	Yes(because it has dynamic backend)
Accountability of user	Yes(because of dynamic backend)
Authenticity	Yes(because of dynamic backend)

Therefore the device is 100% secure.

8.Schedule and budget estimates

Following is the proposed gantt chart with the proper schedule showing the starting and completion of each phase.



The following table calculate the approximated budget as follows:

Employees	Number of employees	Salary of 1 employee for 2 months duration
Software developers	2	2 lakhs
Graphic designers	2	2 lakhs
Testing team	1	1 lakh
Debugging team	1	1 lakh
Senior Member and other executives	2	2 lakh
Other requirements	-	2 lakh
Total=		16 lakhs

Therefore the total budget of the software is 16 lakhs.

