Acknowledgement

During the project,we took help and guidance from a respected person,who deserves our deepest gratitude. As the completion of this project gave us much pleasure,we would like to express our special thanks of gratitude to Dr Swathi jn, our software engineering course faculty who gave me the golden opportunity to do this wonderful project of BugVITa, which also helped us in doing a lot of Research and we came to know about so many new things I am really thankful to her for her patience,insightful comments,enthusiasm,practical,helpful information and unceasing ideas that have helped us tremendously at all times in the completion of this project. Without her support and guidance ,this project would not have been possible.We could have not imagined having a better faculty in our study. We would also like to expand our gratitude to all those who have directly and indirectly guided us in the realization of the project.

Name	Reg.No	Email
Aashish Sharma	19BCE0971	aashish.sharma2019@vitstudent.ac.in
Priyanshu Gaikwad	19BCE0550	priyanshu.gaikwad2019@vitstudent.ac.in
Srikanth Balakrishna	19BCE0158	srikanth.balakrishna2019@vitstudent.ac.in

BugVITa repository:

https://github.com/srikanthbalakrishna/BugVITa

Executive Summary

Bugs are a common occurrence in newly released software and if unresolved can lead to serious consequences in softwares that use sensitive information or may not even function the way they should in a specific scenario. Though every software development process goes through a meticulous process to eliminate bugs sometimes it is not possible to fix all the bugs or even foresee them during the time of development thus new updates of softwares are usually introduced after it's first release.

Though companies have their own bug bounty programs these companies are very large and can afford to pay for such bounties but for a small budding group of developers it is not possible though their product has higher chance of having bugs and thus makes it very tough to maintain a communication between users and development team. BugVITa aims to provide a platform for these developers so that they can connect with the users and can get feedback from them regarding any bugs in their products.

Table of Contents

List of figures:

List of tables :

Abbreviations:

Definition, Acronym, or Abbreviation	**Description**
Bug	In the computer world, a bug is an error in a software program. It may cause a program to crash or show undesired events that results in the problem suffered by the users when they use the software.
BTS	Bug Tracking System
SRS	Software Requirements Specification.
UC	Use case
DFD	Data flow diagram
STD	State Transition diagram
ERD	Entity Relationship Diagram

1.INTRODUCTION

Bug and issue tracking systems are often implemented as a part of integrated project. Some bug trackers are designed to be used with the distributed revision control software. These distributed bug trackers allow bug reports to be conveniently read, added to the database or updated while a developer is offline. Recently, commercial bug tracking system have also begun to integrate with distributed revision control. All type of bug tracking systems are conventionally viewed as a distinct type of software, so we will develop a bug/issue tracking software for all types of bugs.

So, in this document we will provide information regarding BugVITa's system attributes,requirements,standards etc.

1.1.Objective:

This document describes the software requirements and specification for the system in the user and system level, detailed functional requirement are mentioned in the document. The requirement will be illustrated and presented with the help of diagrams which are used to show complicated interactions.

This document also provides a description of any project dependencies that need to be explicitly expressed. Along with the requirements descriptions, it is also the purpose of this document to describe any performance requirements that need to be met. If there are any standards that need to be considered when developing the software are also listed.

Lastly, the purpose of this document is to communicate the system attributes of the BugVITa software. These system attributes include reliability, availability, scalability, maintainability, and portability.

1.2.Motivation:

This project was created to help developers better collaborate with each other and also get the end users involved in finding and fixing bugs so that they could have a much better and a more fluid experience, this also serves as a bug log in a way as any common bugs that might occur in different applications can be easily resolved as there is history of all the bugs present.

This will enable the software development team to better manage and be quick to resolve bugs that occur during use of any said product.Thus making it a CASE Tool and a tool that is used in development of other softwares.

1.3.Background:

It is within the scope of the Software Requirements Specification to describe the specific system requirements of BugVITa.

Bug and issue tracking systems are often implemented as a part of integrated projects. Some bug trackers are designed to be used with the distributed revision control software. These distributed bug trackers allow bug reports to be conveniently read, added to the database or updated while a developer is offline. Recently, commercial bug tracking systems have also begun to integrate with distributed revision control. All types of bug tracking systems are conventionally viewed as a distinct type of software, so we will develop bug/issue tracking software for all types of bugs.

So, in this document we will provide information regarding BugVITa's system attributes,requirements,standards etc.

2.PROJECT DESCRIPTION AND GOALS

It is a simple tool that provides a list of errors and bugs encountered by the developers during the development process. It will provide a summary of each bug that is reported,along with details such as the description of the bug,who has logged the bug, etc. It will use a database to store this list. The application will retrieve this list each time a user opens up the application so they are always working with the latest list of bugs/errors. It includes basic features like adding, deletion and searching for bugs, refresh bug list. Each bug report will have basic attributes like it's name, bug description,name of the person who generated the bug, estimated time to fix it, etc. Furthermore, we aim to create an application that we ourselves can use in our projects in the future.

Our bug tracking system has a particular range of modules so following an evolutionary model would allow us to implement these features one by one in a modular method.

3.TECHNICAL SPECIFICATION

- Node

  The basic backend of the website is being built using NodeJS. This will facilitate user signup and login.
  The project utilizes the guidelines advised and used on the nodeJS's Official Website
  - https://nodejs.org/en/docs/

- MongoDB

  MongoDB is being used as the database in our project.

- ReactJS

  The entire frontend of the website is based on React JS to make it more dynamic and modern.

The following is a table of the requirements that the system SHALL meet

ID	Shall Requirement
	System shall allow all users to 'login' in order to work on a project.
	All users shall be able to add a bug report by filling all the necessary details required.
	All users shall have the ability to search for a bug from the database by name
	All users shall be able to search for a bug that created within a particular time frame ,i.e, between Dates X and Y(e.g: Between 3rd and 5th January)
	All users shall be able to search for a bug using any particular attribute of the bug (e.g: Search all bugs that have Status=resolved)
	The author of a certain bug shall be allowed to modify its details.
	The author of a bug shall be allowed to delete it as well
	The admin/developer shall be able to add a bug report by filling all the necessary details required.
	The admin/developer shall be able to modify any bug.

	The admin/developer shall be able to mark any bug as resolved.
	The admin/developer shall be able to mark a bug as duplicate. The duplicate bug will then redirect to the original bug.

Constraints

ID	***Shall Requirement***
1	The user interface shall also be user friendly so that everyone can use it.
2	Protection at Admin level,system shall try and have security for unauthorized access of data by using a password for database access.
3	Content of the Bug Report shall not be restricted by the system.
4	Honesty of the Bug reporter(author) shall not be the responsibility of the system.
5	Tester shall only be able to add,view,search a bug as well as modify only limited part of his own reports only while the bug is Open.
6	Administrator should be able to link a particular registered user to a desired product/project.

7	It shall not be the responsibility of the system to deal with other aspects of project management such as version control,etc :

User characteristics

User	**Description**
Tester	The tester is a user whose job is solely to use the software made by the developers and find bugs in the application. The tester will report the bug by simply adding a bug to the bug list by filling up it's name, description and other such attributes. The tester can be thought of as the general user in our case who has access to only the basic functionalities of bug reporting.
Developer	The developer is a user who has made the software or one who has been assigned to locate and fix the generated bugs and issues. A developer can also report bugs. Primarily, his job is to work on the bugs and coordinate with his team to make his software better.
Administrator	The admin can be thought of as the Super-User, the admin creates the project and is the leader in the development process of the application. The admin has complete control of the system and all privileges. The admin creates the project and adds 'users' linked to the project. Login credentials are generated and a user can login to start working on the project.

Analyst	The analyst is a part of the development team whose job is to assign a particular bug a developer if needed.The Analyst also has to peer review fixed bugs and clear them. An analyst has the same privileges as a developer with the addition of the ability to add/modify the 'assignee' for a particular bug

4.DESIGN APPROACH AND DETAILS

The project uses various design paradigms from various documentations of React and Node

4.1.Design approach, Materials & Methods:

4.1.1 Architecture Model

BugVITa has been decomposed into the following modules.

- Bug Repository : This module collects data from the user to be used for logging Bugs and issues for a particular Product(project) and also includes User data.
- User Profile : This module is responsible for displaying each User's profile details. It is also responsible for editing profile details like "password"
- Bug Add and Modify: Responsible to update the Bug repository by adding new Bug Reports or by modifying existing Bug reports.
- Bug search: Enables user to search the bug repository by providing search parameters relevant to a particular Bug report object.
- Bug repository viewer: Fetches the list of all bugs pertaining to a particular product from the repository.
- User Login and Registration:
    - Login : Facilitates user login by validating inputted user credentials against the corresponding credentials in the User Database
    - User registration: Enables new users to sign-up by creating their own password and adding their credentials to the User Database

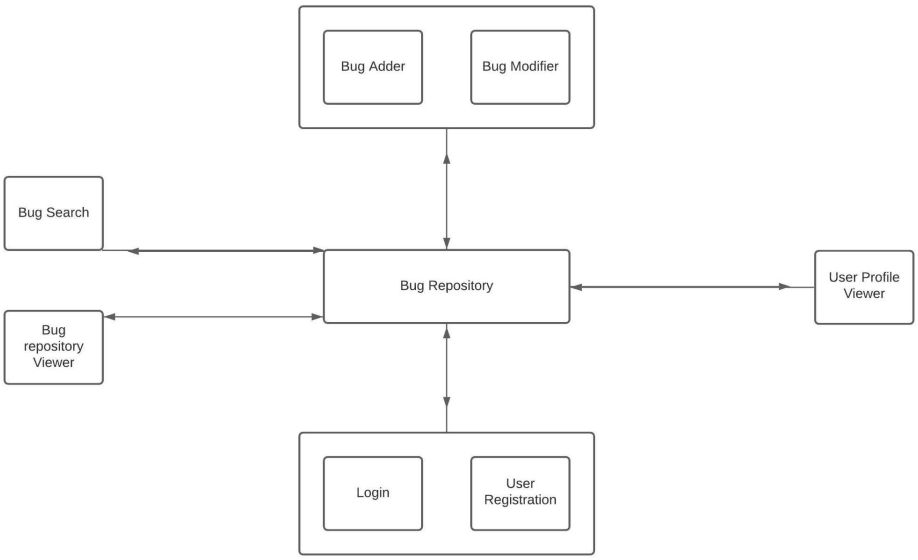


Figure 1 Architecture Diagram

4.1.2 Use Case Diagram

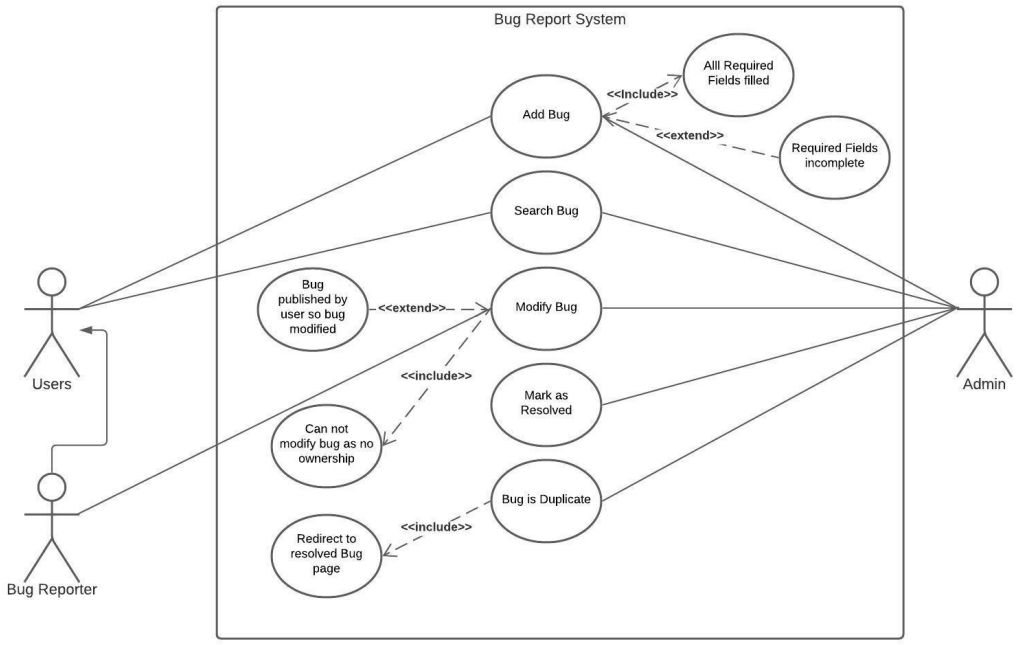


Figure 2 Use Case Diagram

4.1.3 ER Diagram

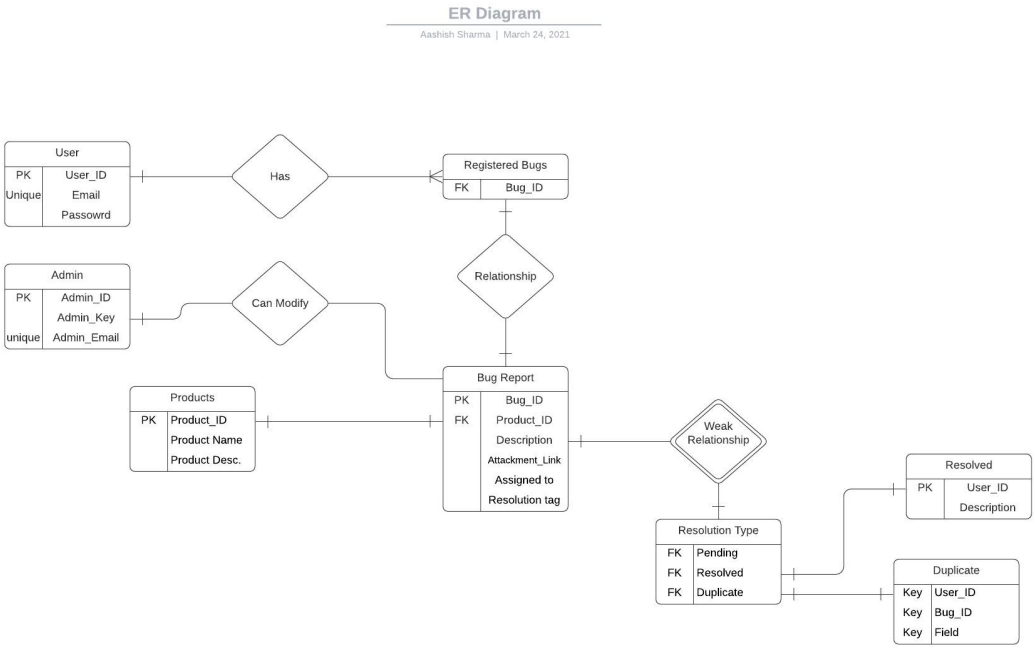


Figure 3 ER Diagram

4.1.4 Data Flow Diagram

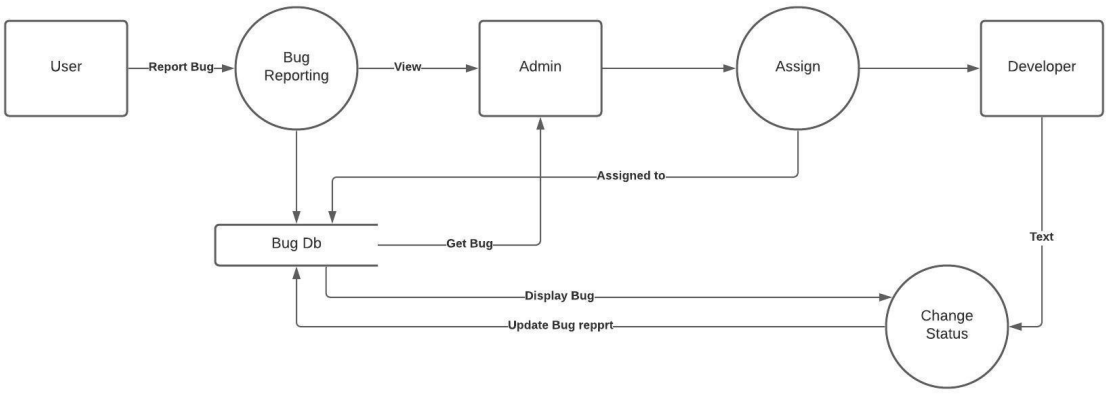


Figure 4 Data Flow diagram

4.1.5 State Transition Diagram

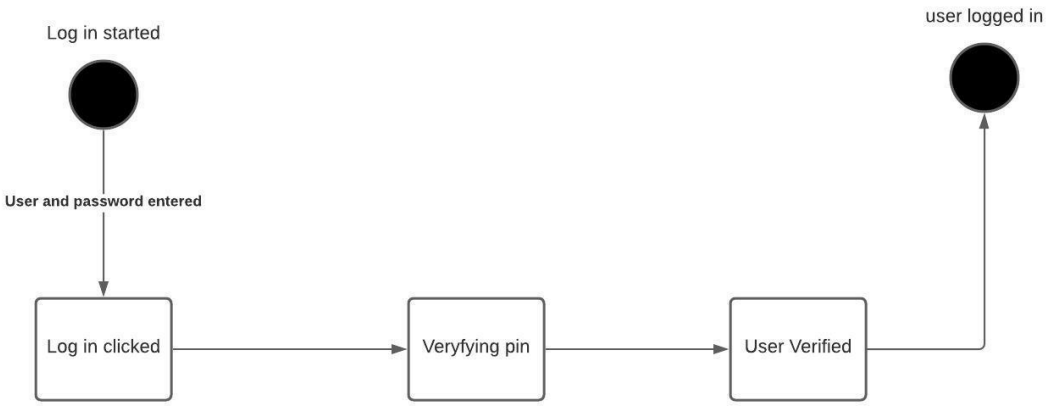


Figure 5a State Transition Diagram for Login

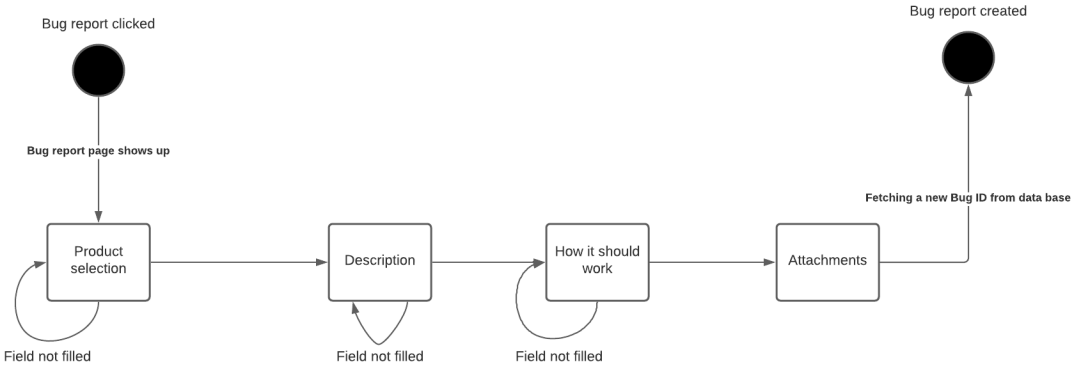


Figure 5b State Transition Diagram for Bug report

4.1.6 Class Diagram

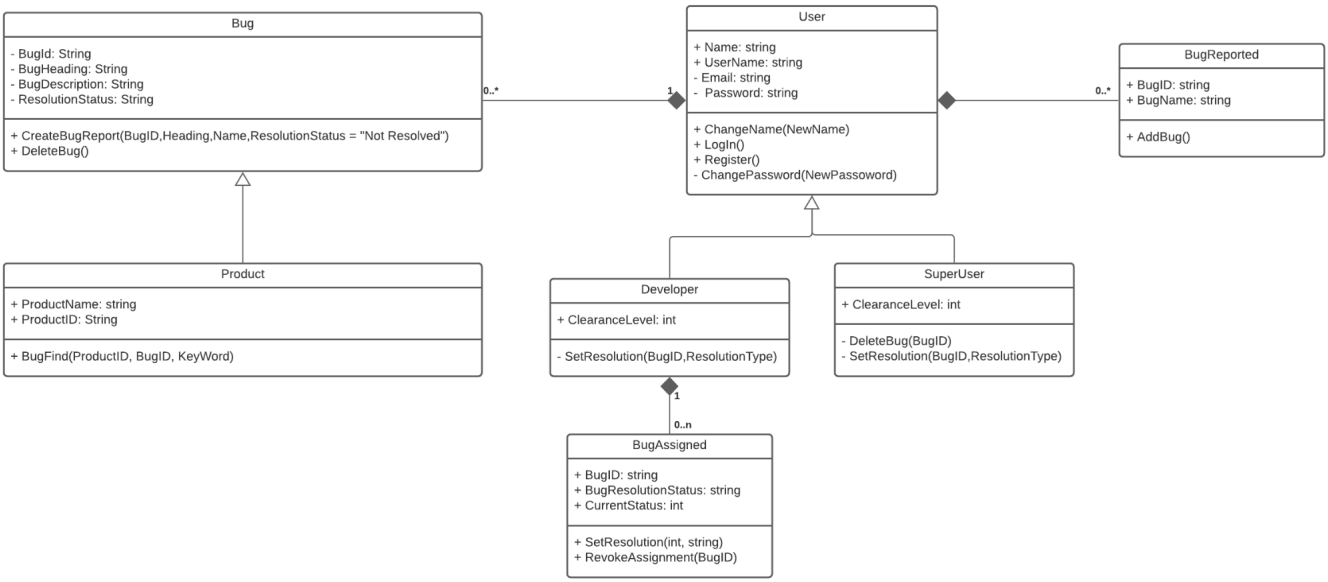


Figure 6 Class Diagram for BugVITa

4.1.7 Swimlane Activity Diagram

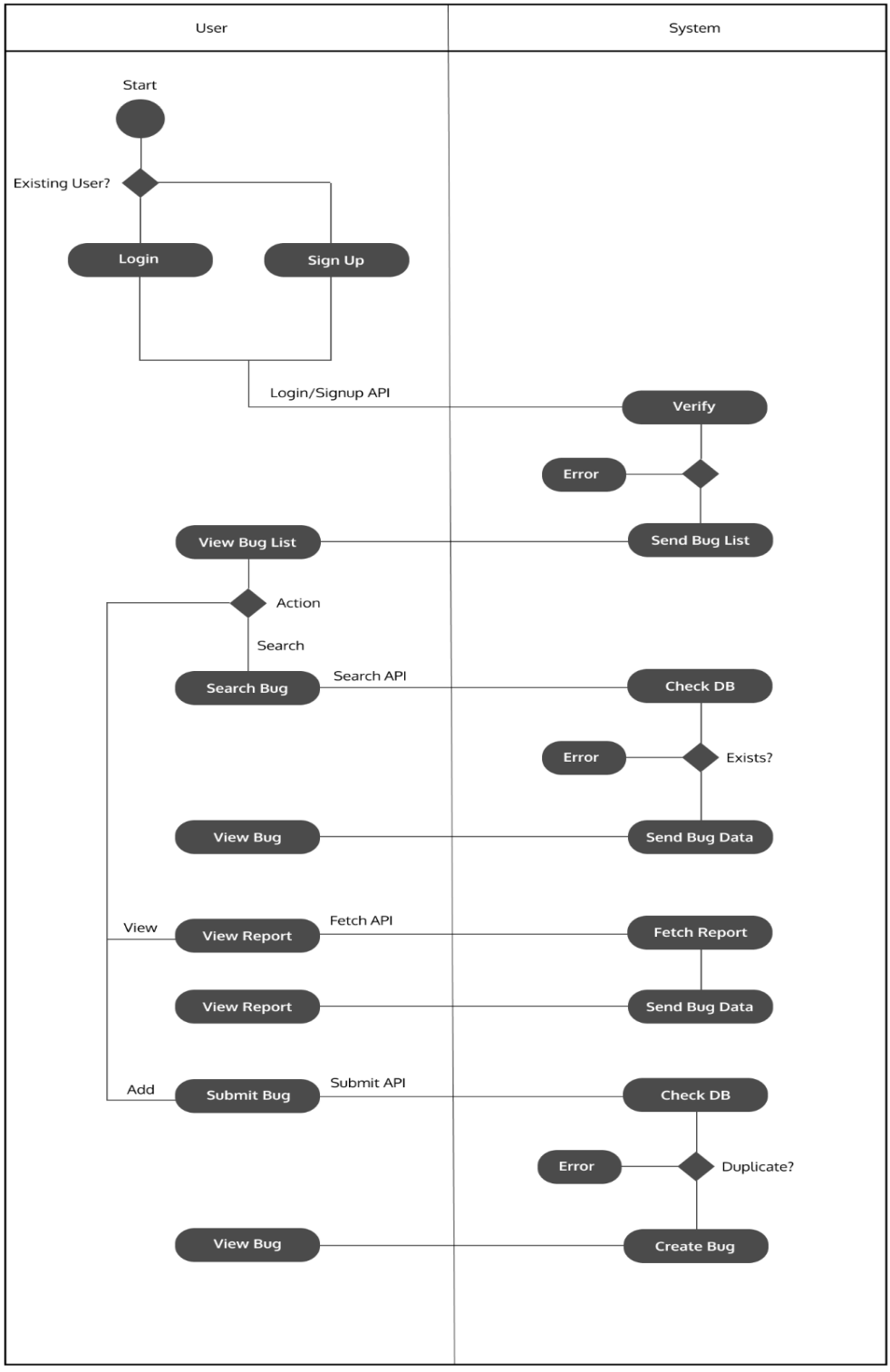


Figure 7 Swimlane Activity Diagram for BugVITa

4.1.8 Sequence Collaboration Diagram

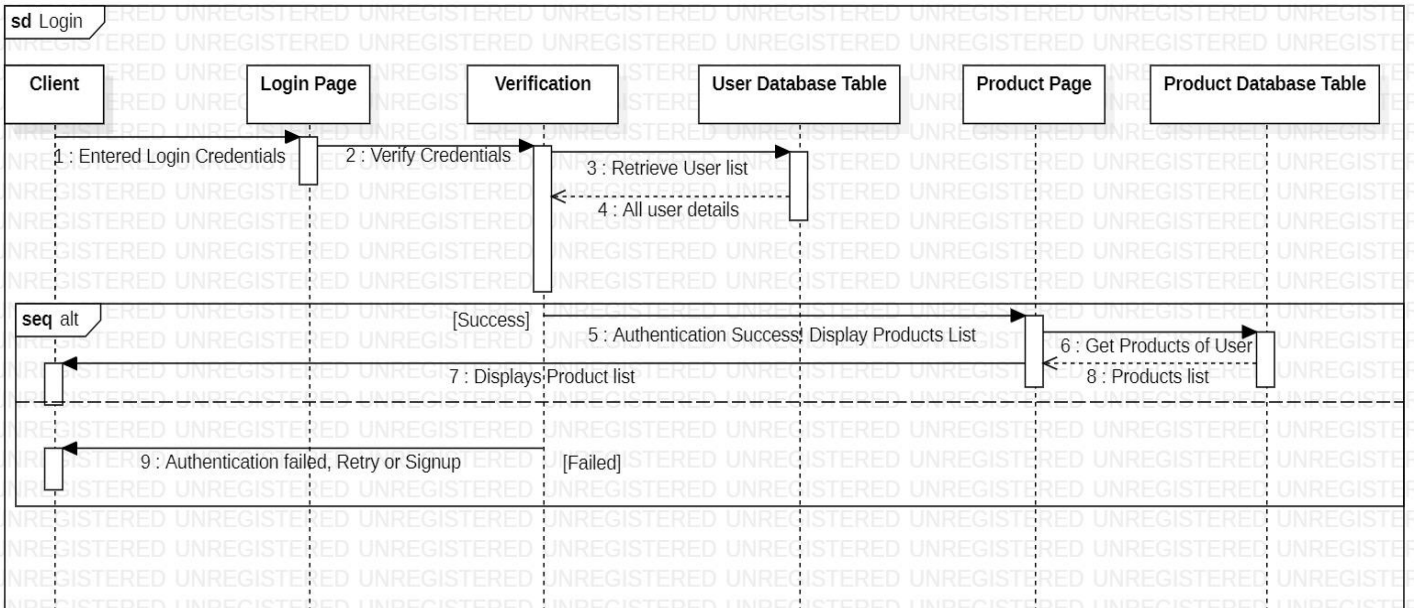


Figure 8a Sequence Diagram for Login

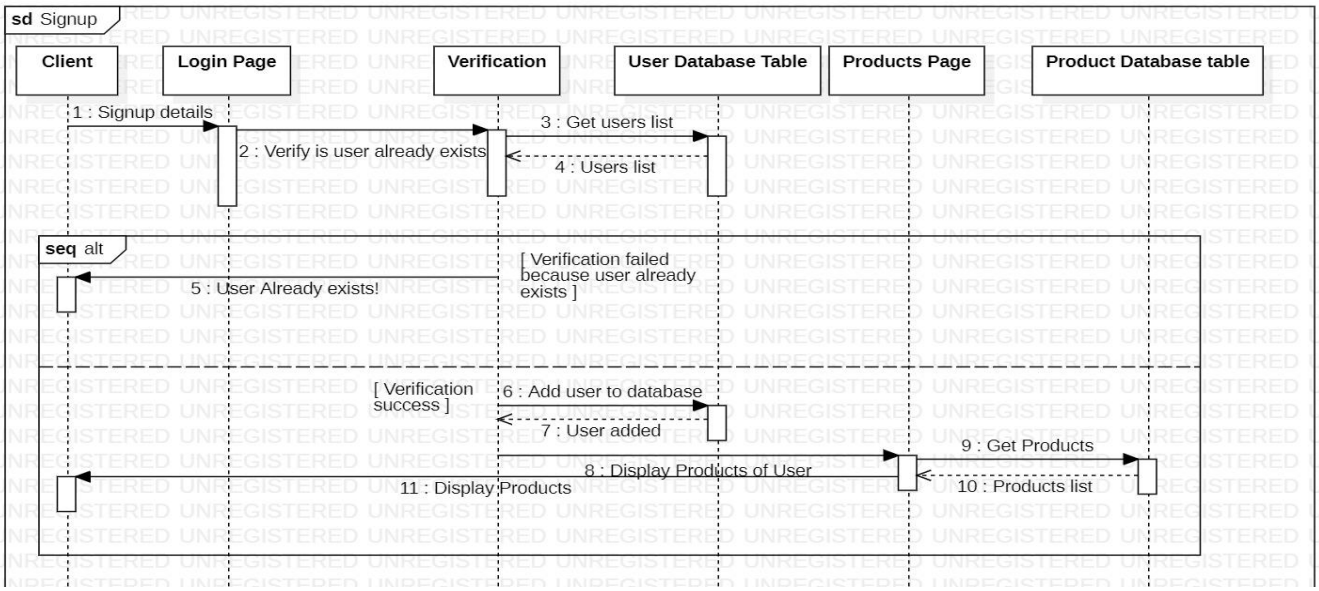


Figure 8b Sequence Diagram for SignUp

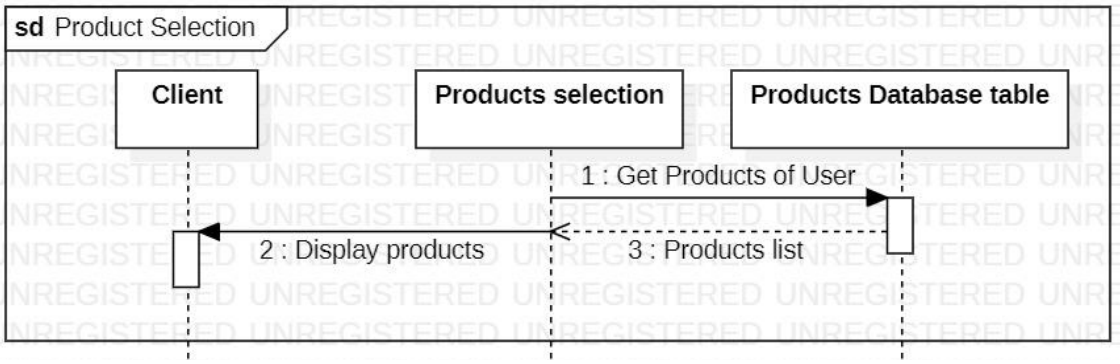


Figure 8b Sequence Diagram for Product Selection

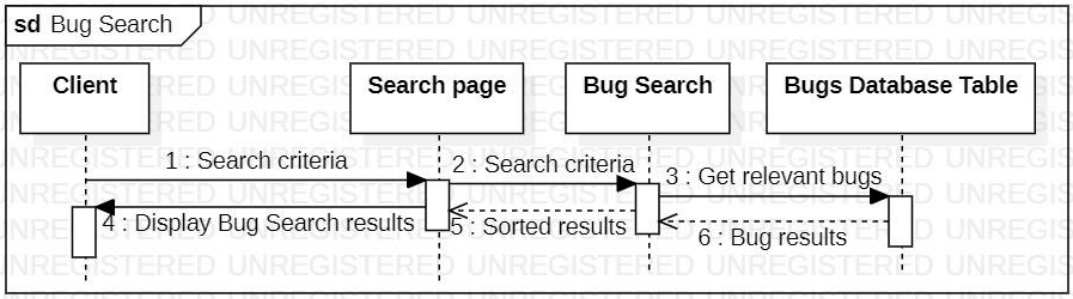


Figure 8b Sequence Diagram for Bug Search

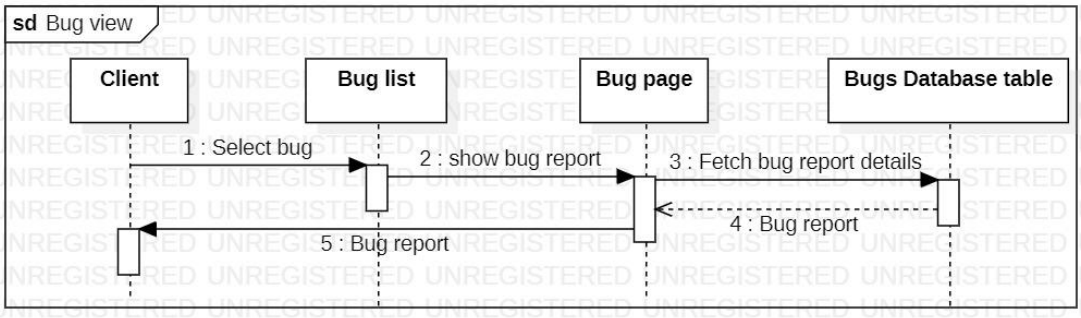


Figure 8b Sequence Diagram for Bug View

4.2.Codes and Standards:

We comply with the following standards in the project

React Documentation Design principle	https://reactjs.org/docs/design-principles.html#:~:text=The%20key%20feature%20of%20React,rippling%20changes%20throughout%20the%20codebase.
GUI Guide	https://material.io/design

5.SCHEDULE, TASKS AND MILESTONES:

The Schedule, Tasks and Milestones of the project are as follows

5.1 Gantt Chart and Activity Network (process)

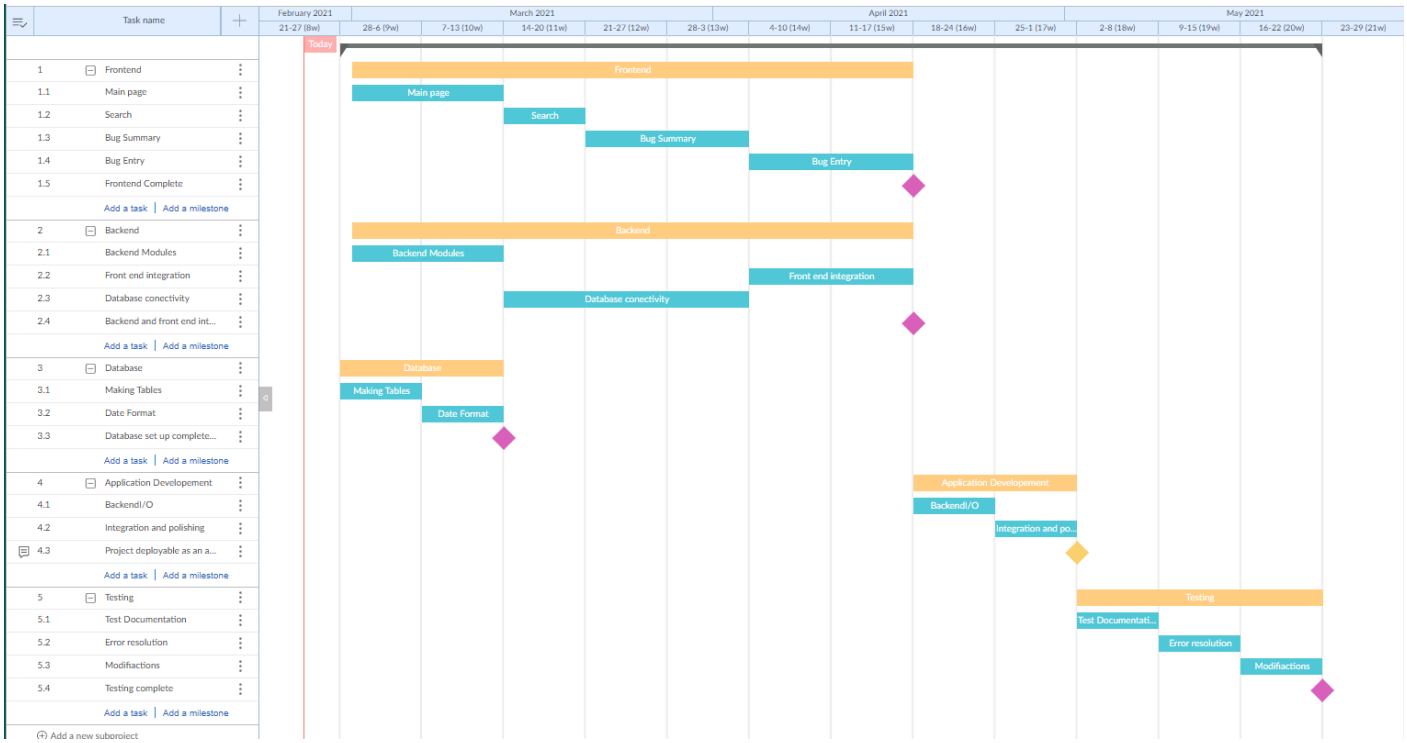


Figure 9 Gantt Chart for process

Task	Label	Predecessor	Staff Required	Duration
Design	A	-	1	5
Frontend	B	A	2	10
Backend	C	A	2	10
Database	D	C	2	10
Application Dev	E	ABCD	3	15
Testing	F	ABCDE	3	5

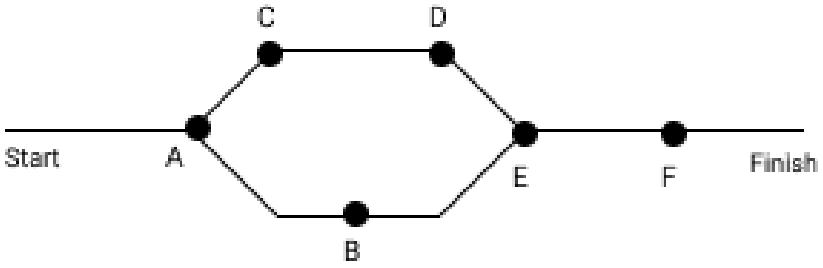
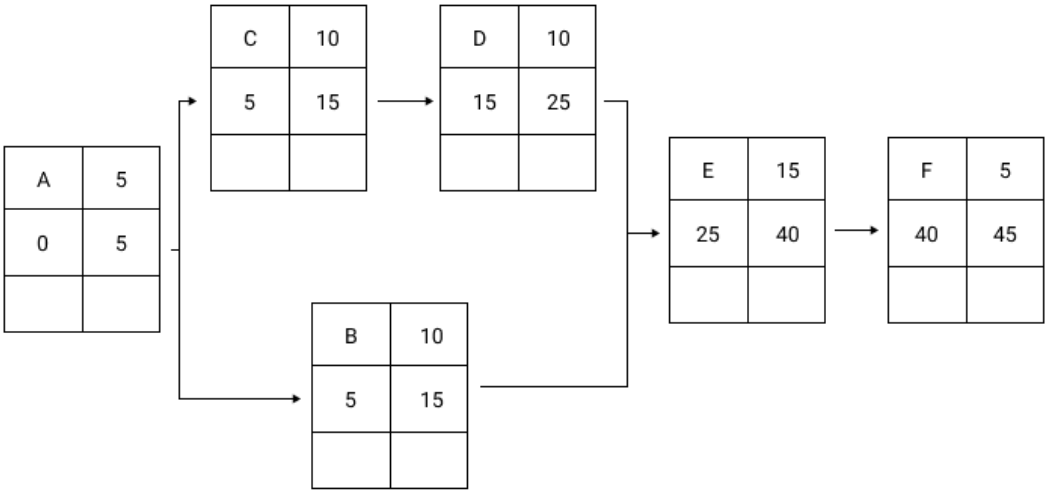
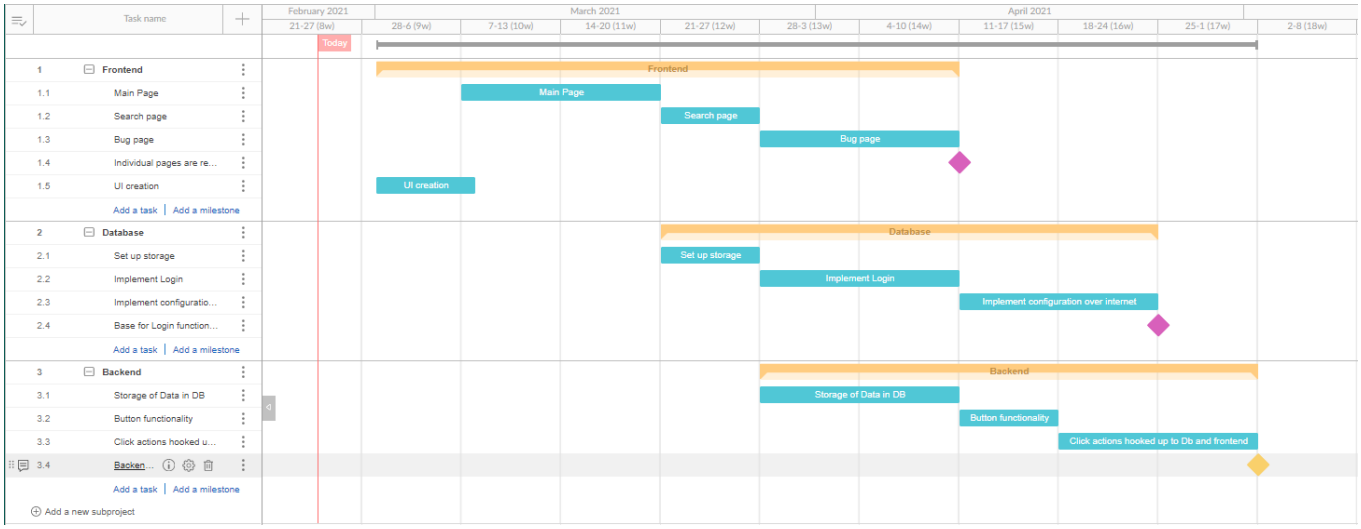


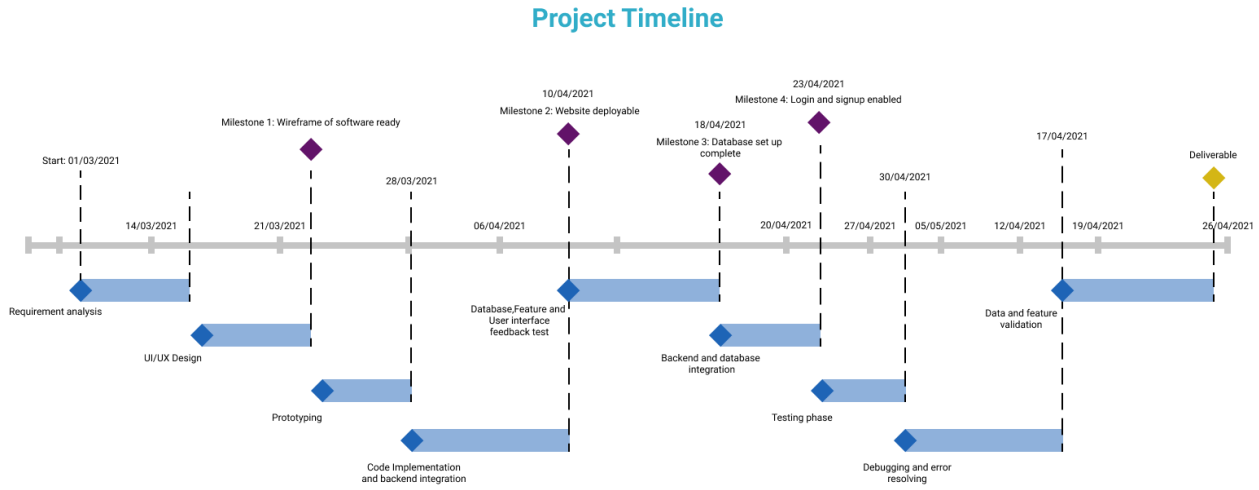
Figure 10 Activity Network



5.2 Gantt Chart and Activity Network (product)

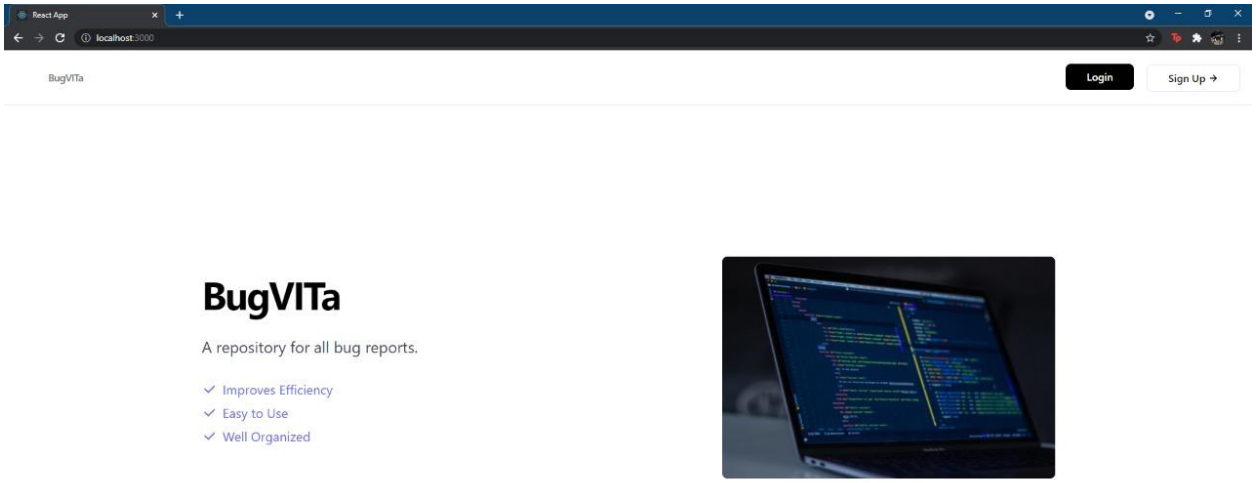


Gantt Chart for product developement

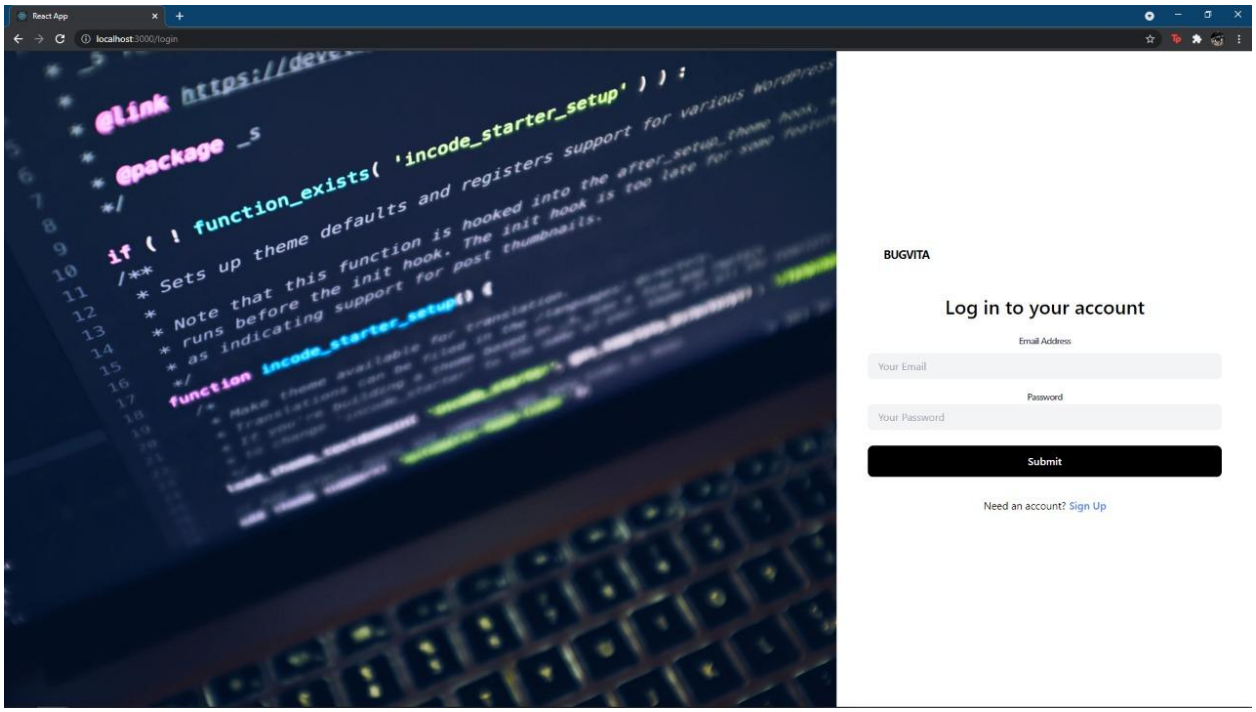


Project Timeline

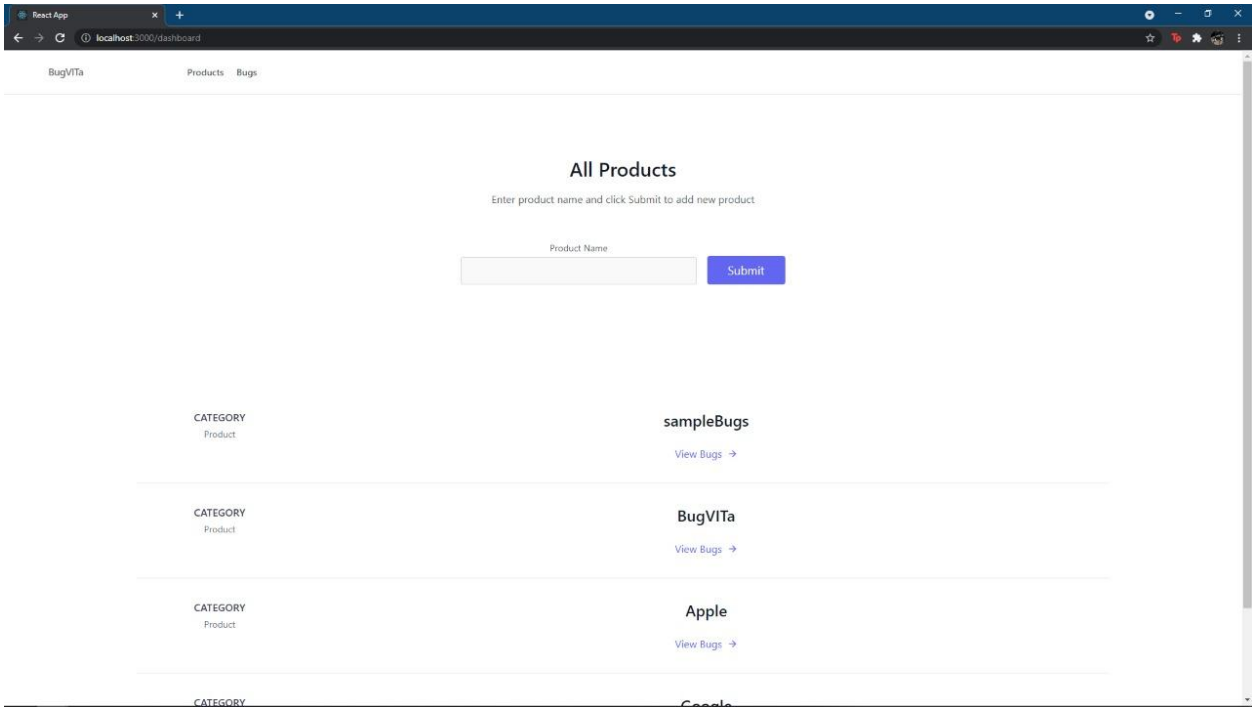
6.PROJECT DEMONSTRATION



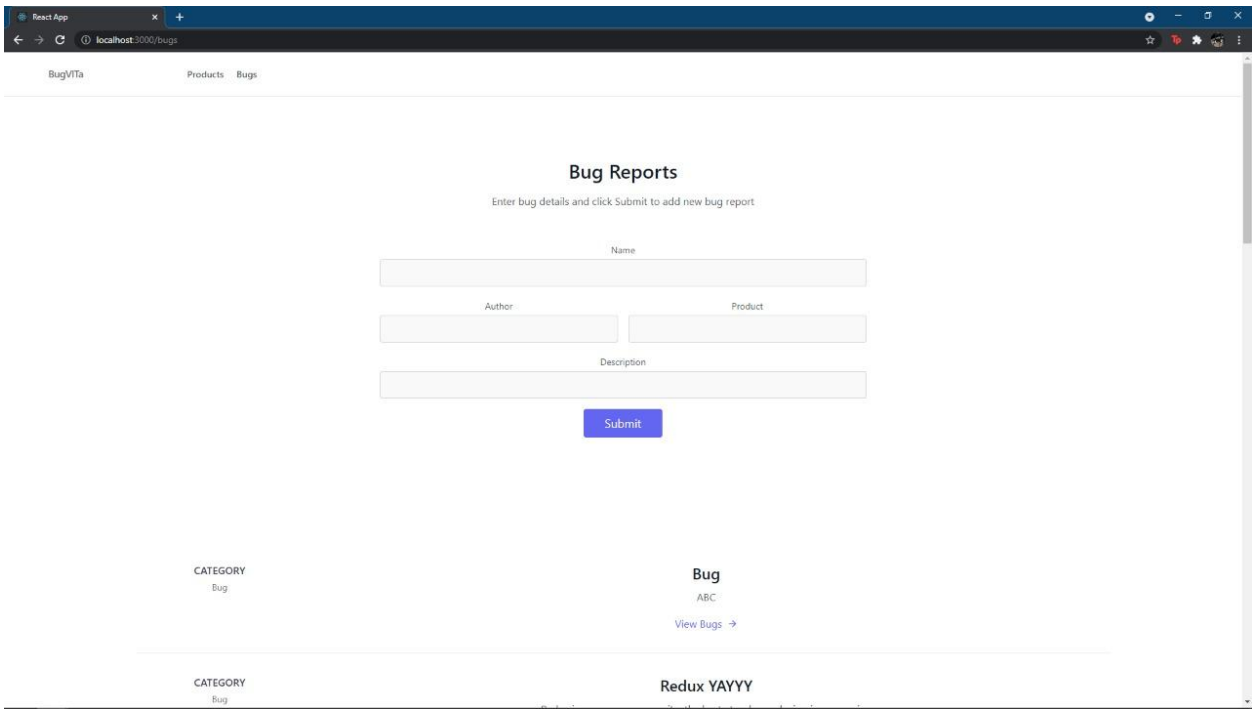
Index Page



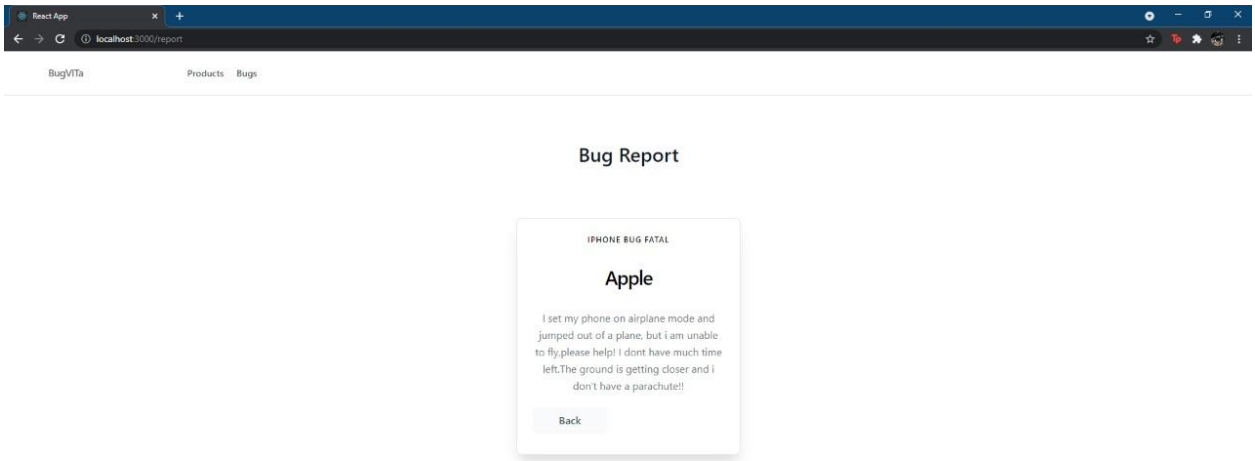
Log in Page



Product Page



Bug View Page



Bug Description

7.RESULT & DISCUSSION

The past 3 months have been very enlightening to get to know the workflow of creating a software right from ideation to a fully fletched working project and we have learned a lot throughout the journey of making this bug tracking and reporting system

The product can help developers in the following ways:

- Create a common platform for sharing bugs.
- Help developers fix more serious and redundant bugs.
- Create a community of developers for healthy discussions and sharing insights

We have been able to successfully integrate:

- Login Page using React and Node js
- Product View Page
- Add a bug Page
- Search Bug option
- Bug description