Technical Design Document

# Project Name : Ebug Tracker Web App

## Team Name : J\_FSD\_5

### 

### Amulya Bendi (SPOC - I)

### Shubham Kapoor (SPOC - II)

### Shreya Dubey

### Soumya Sathputhe

### Anirudh Bhardwaj

### Uma Shankar Bhagat

# INDEX

## Abstract

## Introduction

## Software Requirements Specifications

## Functional Requirements

## Flow Charts

## Sequence Diagram

## Use Case Diagram

## Class Diagram

# ABSTRACT

For many years Bug Tracking mechanism was employed only in some of the large software development houses. Most of the others never bothered with bug tracking at all and instead simply relied on shared lists and email to monitor the status of defects

EBug Tracking System is a web-based application that is designed to help quality assurance and programmers keep track of reported software bugs in their work in the form of tickets. Tickets will be assigned to a person with a ticketid, flag, description, project name and its details.

Nowadays, when projects are so extensive, defects or bugs have existed as a problem in the system and they are normally inevitable in software development. A bug could be in either a program’s source code or its design or anything that is vulnerable.

# Introduction

EBug Tracking System is a web-based application that is designed to help quality assurance and programmers keep track of reported software bugs in their work..

Bugs will be assigned to a person with a ticketid, flag, description, project name.

Tickets can be submitted to the admin with an attachment for the bug

detailed report.

Admin can maintain staff, projects, bug categories, bug priorities, status of bug etc.

# Software Requirement Specification

### MINIMUM HARDWARE REQUIREMENTS

#### **Processor** :An Intel Pentium 4 processor or later that's SSE3 capable

#### **Operating System**

#### **For Mac** : OS X El Capitan 10.11 or later

#### **For Linux** : 64-bit Ubuntu 14.04+, Debian 8+, openSUSE 13.3+, or Fedora Linux 24+

#### **For Android** : Android Lollipop 5.0

#### **Storage**

#### **Hard-Disk** : 256GB Recommended

#### **RAM**: 4Gb Recommended

#### 

### **Technology Stack For App Development**

### FrontEnd Development Stack

### Angular

### HTML/CSS

* + BOOTSTRAP

### BackEnd Development Stack

### Spring-Boot

### Hibernate for ORM

### mySql/Oracle for database

### Testing using Junit Mockito

### Other

### IDE for Backend : Eclipse/ STS

### IDE for FrontEnd : vsCode

# Functional Requirements

## Admin Module

* This module has the entire access to all other modules, It can login to the app and can enter the details of staff, project, view bugs send from the customers.The admin can also also assign work to staffs, view bug case flow status details, send messages to customers using this bug tracking application

## Employee Module

* Employee can login to the site using username and password. Then he/she can view the all bugs assigned to them and able to check tickets based on critical level and status. He can directly give solution message to customers or he/she can assign the bugs to other staffs if the bug is related to them.

## Customer Module

* Customer can view bugs case flow details with which he/she is involved. The customer registers in to the applications and login to the site using username and password.

Whenever a bug is raised from his software, he sends the bug details to the administrator with print screen of the bug generated. He/she may see the bug case flow details and bug status along with remedy details at any time using the ticket number generated during new bug entry.

# 

# Module Division

* Admin Module - shubham Kapoor and Uma Shankar

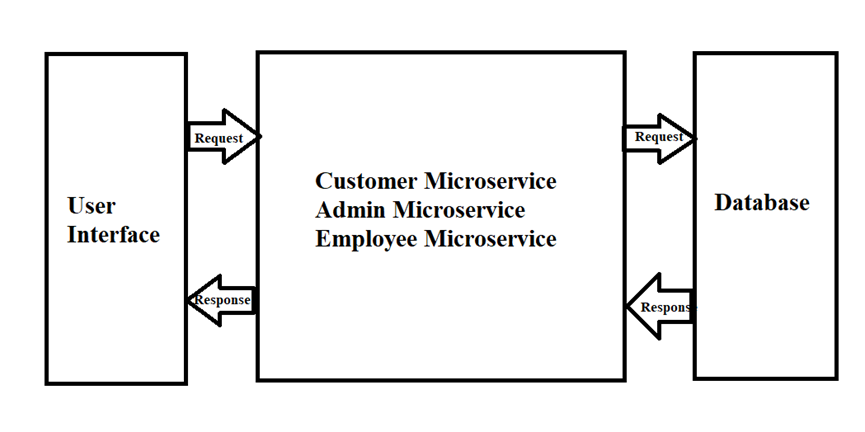
1. Able to add Employee
2. Add project
3. Assign tasks (Bugs ticket) to Employee
4. Able to send messages to Customer
5. Able to update Critical level of the bug

* Employee Module - Shreya Dubey and Anirudh Bhardwaj

1. Resolve the bug and send solution to the customer
2. Update the critical level of the bug
3. Assign bug to other employee if its not related to him

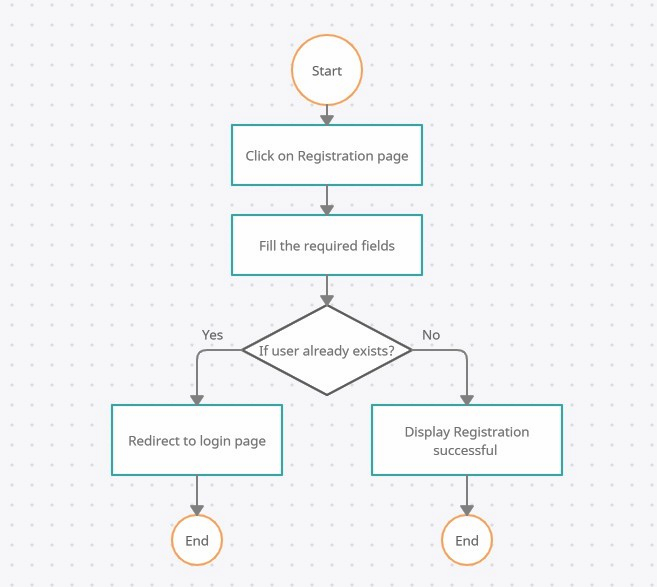
* Customer Module - Amulya and Soumya

1. Register and login to the application
2. Raise the ticket for the bug
3. View ticket details using ticket id

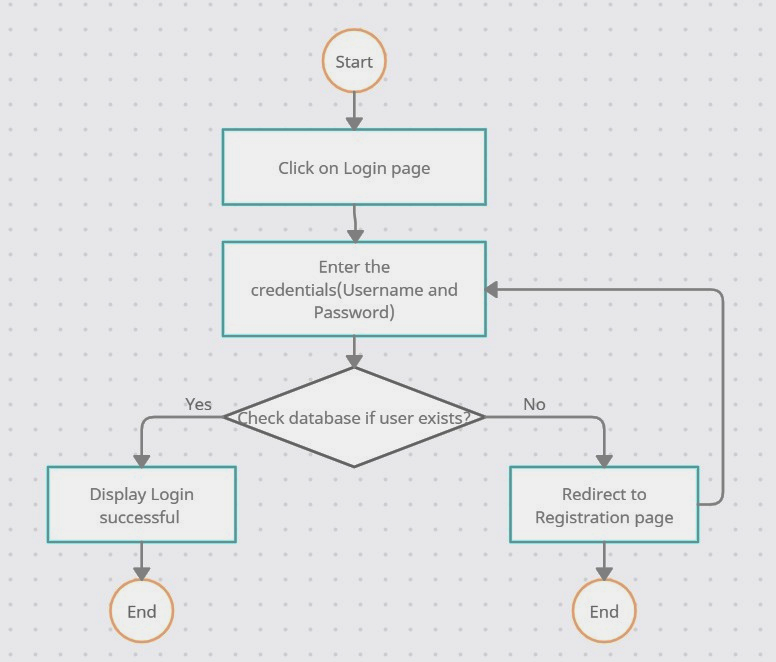
Architectural Diagram

# Flow Charts

## Login and Registration Of Actor

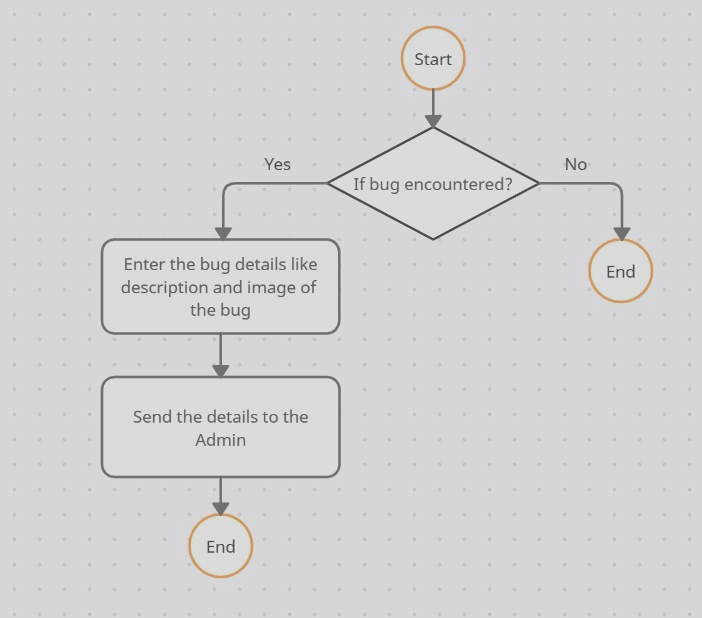


Registration

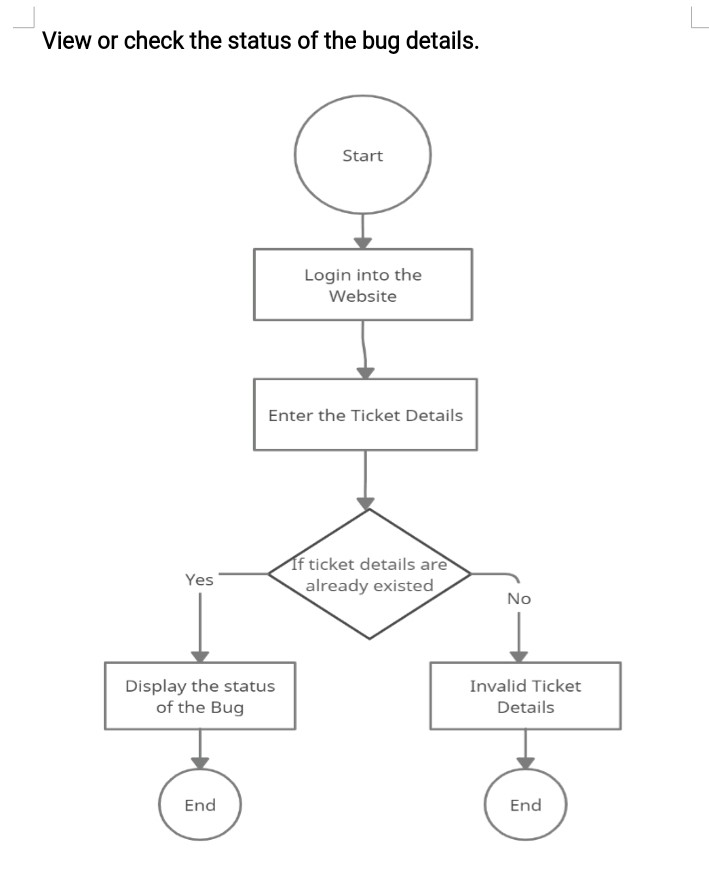


Login

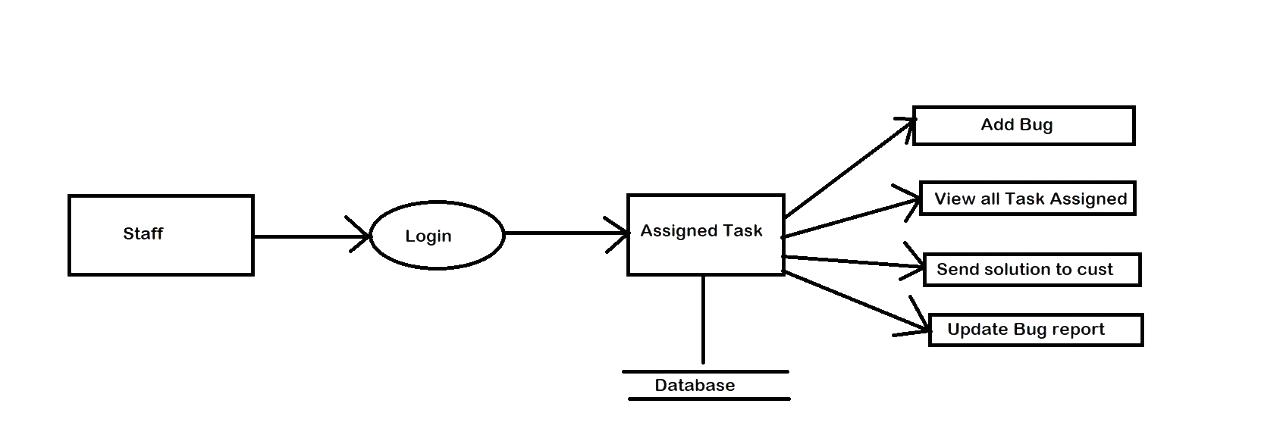
## Flowchart of Customer



Customer Sending details of Bug to Admin

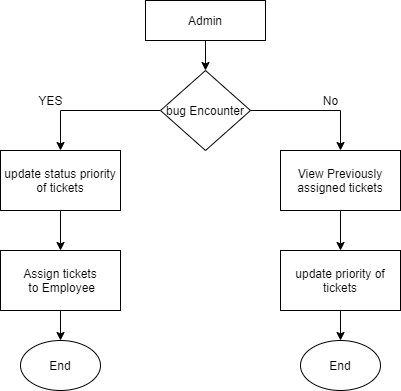


## Flowchart of Employee



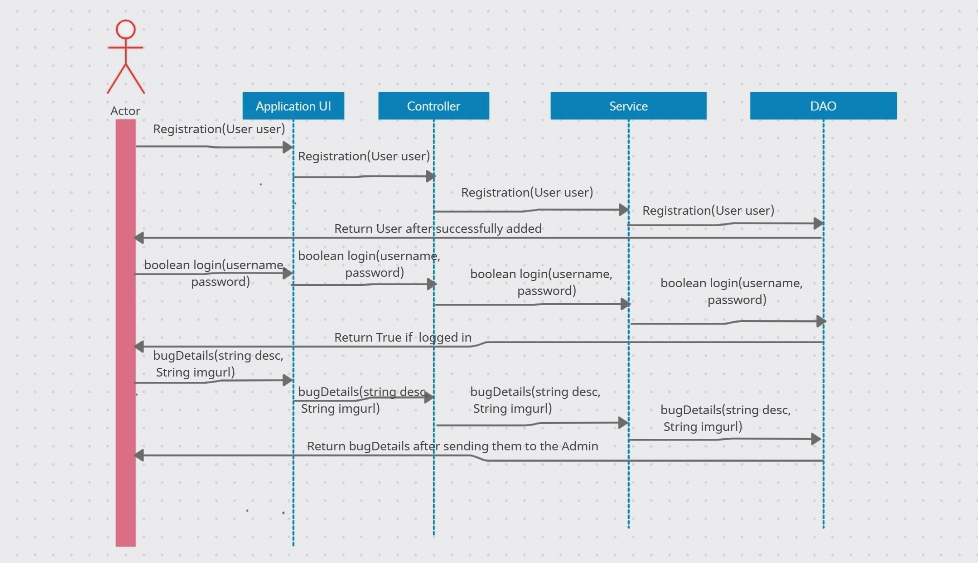
Employee/Staff

## ADMIN FLOW CHART

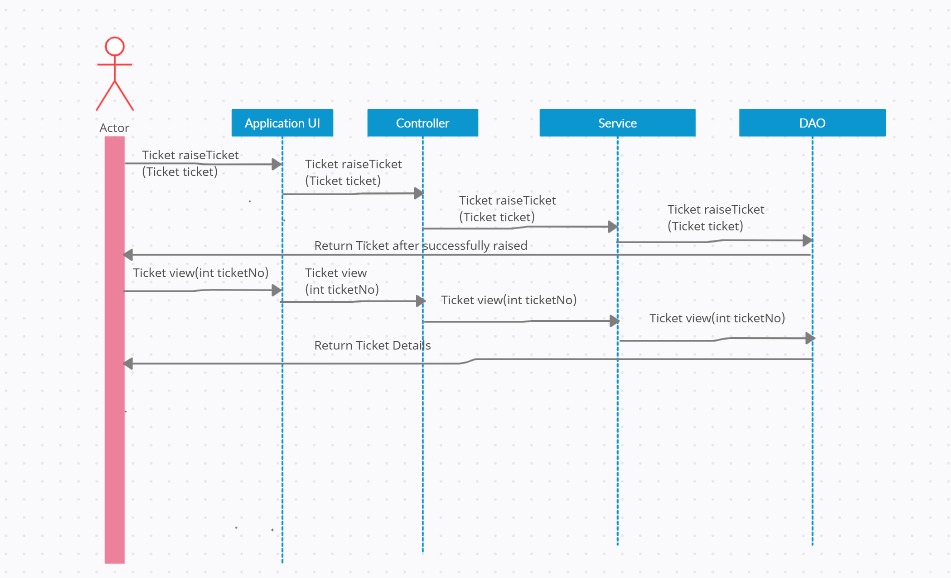


# 

# Sequence Diagram

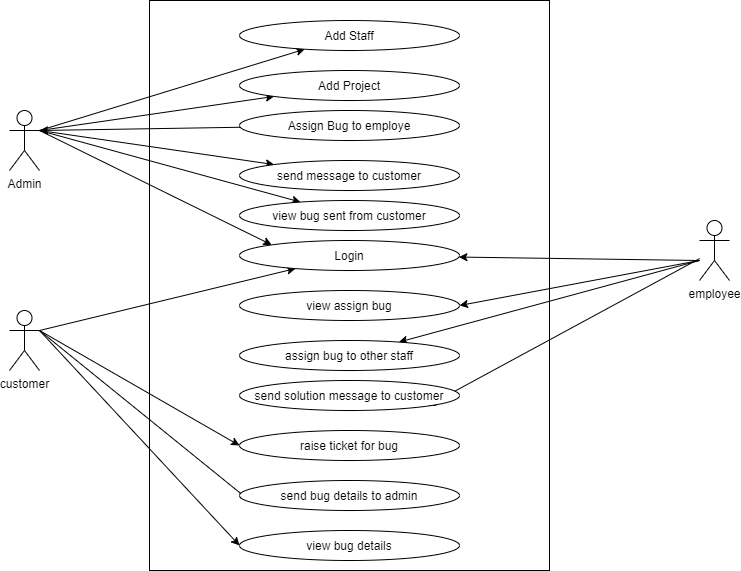


Login and Registration Functionality

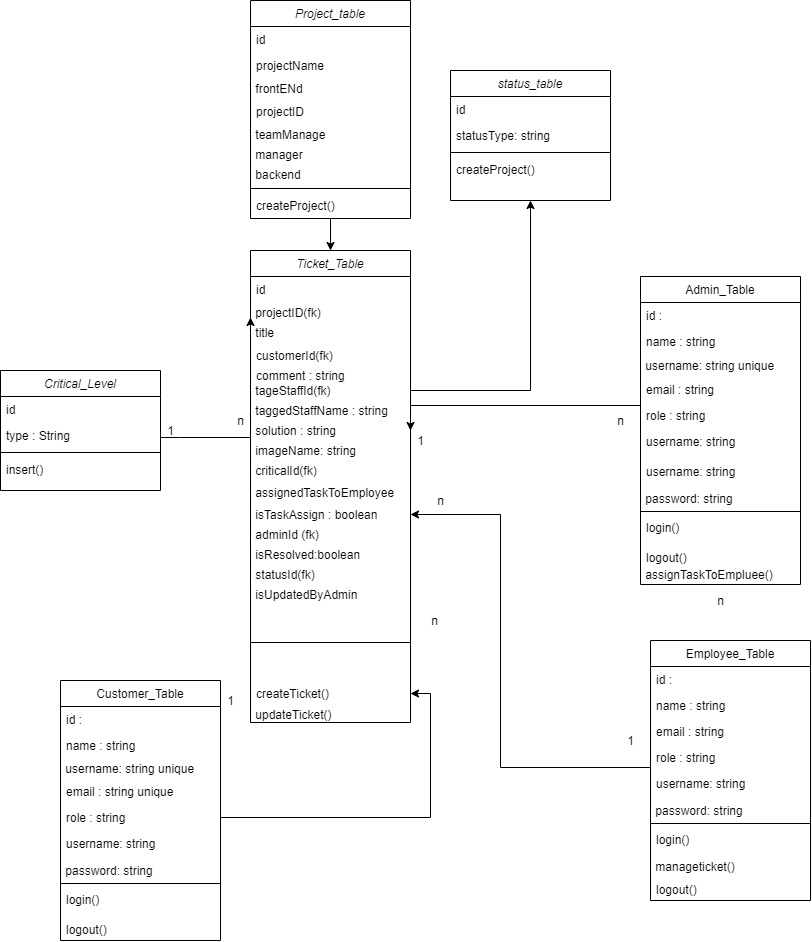


Customer Creating Tickets

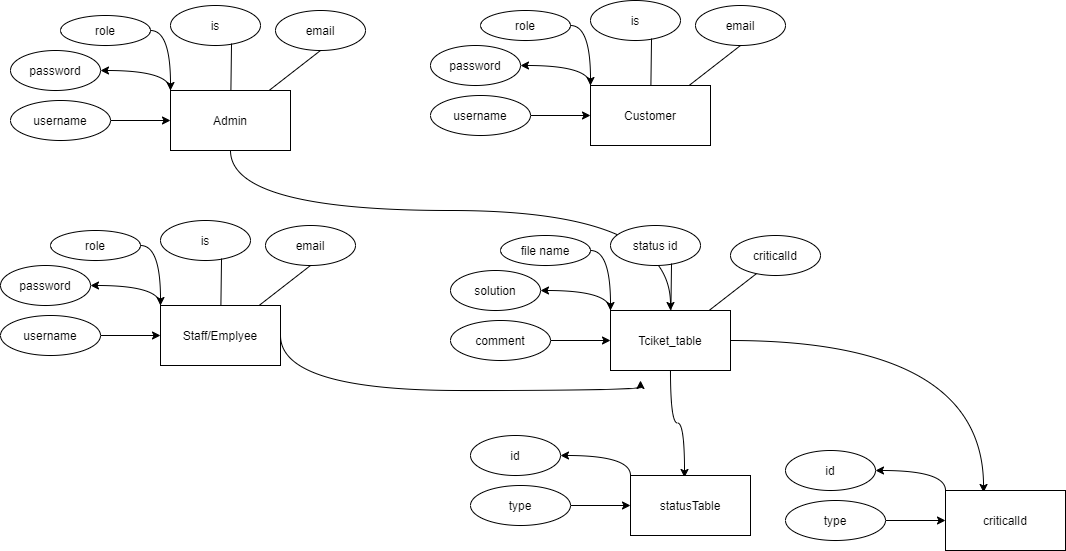
# Use Case Diagram



# Class Diagram



# ER DIAGRAM



# UI WireFrames

