**Understanding In-app Ads and Detecting Hidden Attacks through the mobile App -Web Interface.**

Contents

[I. Project Details 3](#_Toc97220598)

[II. Structure of the system 4](#_Toc97220599)

[1. System Design 4](#_Toc97220600)

[2. Modules 5](#_Toc97220601)

[2.1 Admin Module: 5](#_Toc97220602)

[2.2 User Module: 6](#_Toc97220603)

[2.3 Android User Module: 6](#_Toc97220604)

[3. Components 7](#_Toc97220605)

[3.1 Sign-Up: 7](#_Toc97220606)

[3.2 Login Validation: 7](#_Toc97220607)

[3.3 Dashboard Display: 7](#_Toc97220608)

[III. Requirement Specification 9](#_Toc97220617)

[1. Functional Requirements 9](#_Toc97220618)

[1.1 Components Functional Requirement with respect to each module 9](#_Toc97220619)

[1.2 Database Requirement 13](#_Toc97220620)

[2. Non-Functional Requirements 14](#_Toc97220621)

[2.1 Security Requirements: 14](#_Toc97220622)

[2.2 Privacy Requirements: 14](#_Toc97220623)

[2.3 Scalability and performance Requirements: 14](#_Toc97220624)

[2.4 Usability Requirement: 14](#_Toc97220625)

[2.5 Compatibility Requirement: 14](#_Toc97220626)

[3. Interfaces (user, hardware, software, and/or communication) 14](#_Toc97220627)

[3.1 Hardware Interface Requirements: 14](#_Toc97220628)

[3.2 Software Interfaces Requirements: 15](#_Toc97220629)

[3.2 User Interface Requirement: 15](#_Toc97220630)

[3.3 Communication Interface Requirement: 20](#_Toc97220631)

[IV. Project Implementing Plan 20](#_Toc97220632)

[1. Development Phase -1: 20](#_Toc97220633)

[1.1 Developing Database tables (Login Tables). 20](#_Toc97220634)

[1.2 Developing the application layer for the webpage. 20](#_Toc97220635)

[1.3 Developing Webpages for all Modules. 20](#_Toc97220636)

[1.4 Integrating Database with the Front-End applications. 20](#_Toc97220637)

[2. Development Phase-2: 20](#_Toc97220638)

[2.1 Develop back-end code for User, Admin and Android-userModules. 20](#_Toc97220639)

[2.2 Integrate Components in the User, Admin and Android-user Modules with the Front-End Applications and the Database. 20](#_Toc97220640)

[2.3 Perform Initial testing. 21](#_Toc97220641)

[3. Development Phase-3: 21](#_Toc97220642)

[3.1 Performing integrated system testing 21](#_Toc97220643)

[3.2 Preparing test cases and identifying Bugs. 21](#_Toc97220644)

[3.3 Fixing the identified bugs. 21](#_Toc97220645)

[3.4 Re-testing until all the requirements are met. 21](#_Toc97220646)

[V. Member contribution table 22](#_Toc97220647)

Table of Figures

[Figure 1: High Level Design of the Bug Tracking System 3](#_Toc97154878)

[Figure 2: System Structure Diagram of the Bug Tracking System. 4](#_Toc97154879)

# Project Details

**Project Title:**  **Understanding In-app Ads and Detecting Hidden Attacks through the mobile App -Web Interface.**

**Group Name:** Team Resolve

**Group Members:**

1. Navyasri Arekatla
2. Lokesh Bachu
3. Sushma Bobbishetty
4. Naveen Bolgom
5. Sathvik Cheekati
6. Vamshi Teja Goud Puli
7. Shiva Teja Reddy Yammanuru

# **Structure of the system**

## System Design

The Users access the In-app Adds system to perform each and every task which is completely handled by Admin privileged accounts.

Graphical user interface

Description automatically generated

Figure : High Level Design of the In-App Add control System.

Below is the complete System Structure Diagram of the In-App Add Control System, that is explained in detail in the “2. Modules” and “3. Components” sections under “II. Structure of the system” heading of the document.

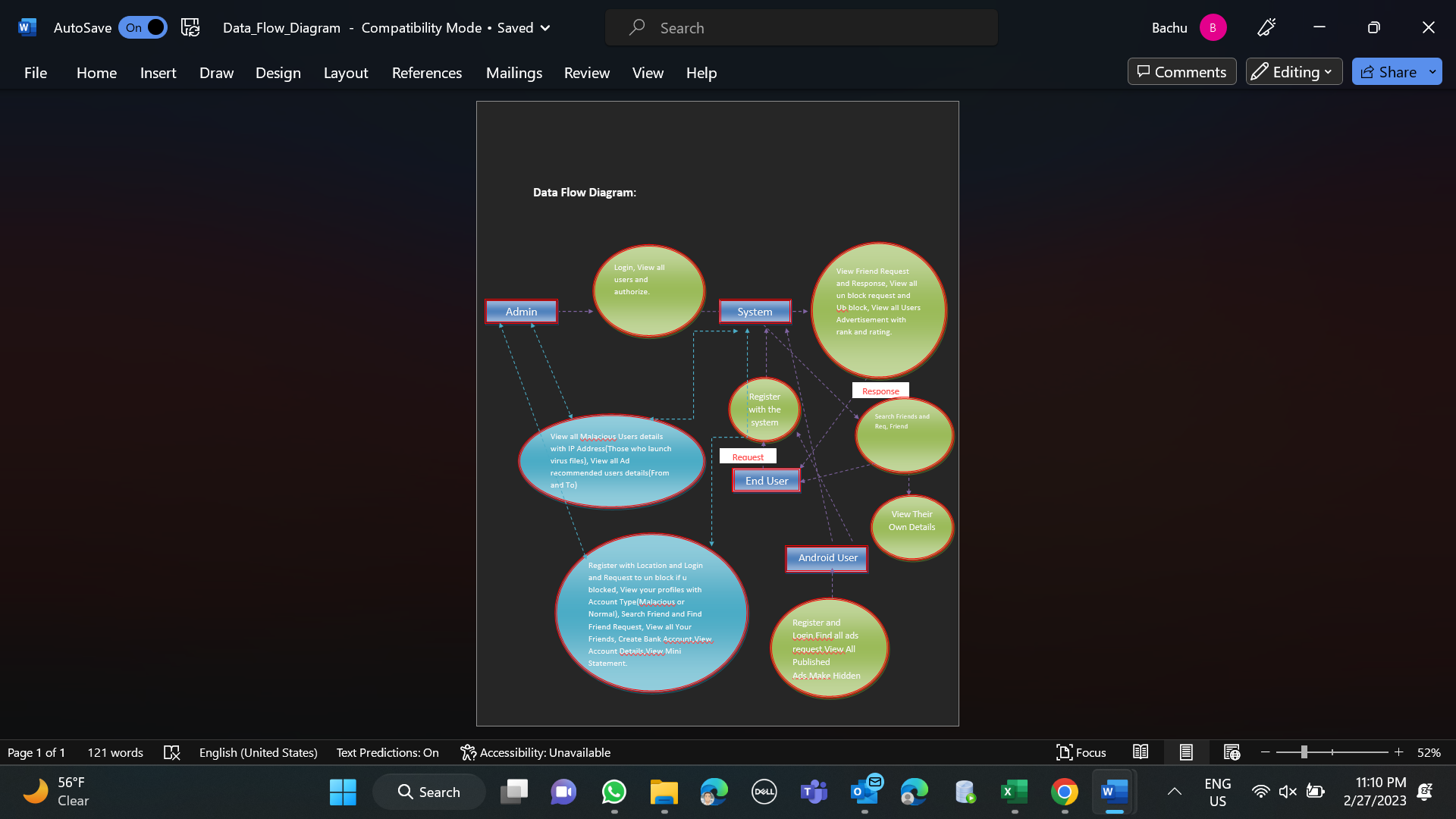


Figure : System Structure Diagram of the In-App Add Control System.

## Modules

The overall system structure is described with respect to each module. Below are the three Main Modules:

### Admin Module:

1. In this module, admin has to login with valid username and password.

2.After login successful he can do some operations such View all users and authorize.

3.View Friend Request and Response, View all un block request and Ub block, View all Users Advertisement with rank and rating.

4.View all Malicious Users details with IP Address(Those who launch virus files),View all Ad recommended users details(From and To).

5.View ad request users and Calculate number of months that u want to advertise and assign cost like for one month --1000,2 months---2000 etc..,

6.View phishing malicious users(those who try to ad without no. Balance), View all bocked users with IP add,Date and Time,View no.of malicious and no.of normal users in Chart.

7.View all ad ranks in Chart.

### User Module:

1. In this module, there are n numbers of users are present.
2. User should register before doing some.
3. After registration successful he can login by using valid user name and password.
4. After Login successful, he will do some operations like Register with Location and Login and Request to un block if u blocked.
5. View your profiles with Account Type(Malicious or Normal),Search Friend and Find Friend Request.
6. View all Your Friends, Create Bank Account.
7. View Account Details, View Mini Statement.
8. Enter your from to To Date and send request to admin to launch Advertisement, Add Advertisement Details(Advertisement category, Ad name, Ad desc(attach file),Ad Date and Time,Compay name, company est. Year).
9. View all your ad details with rank and rating.
10. View all your friends ad details and recommend to other friends.

### Android User Module:

1. In this module, there are n numbers of android users are present.
2. User should register before doing some.
3. After registration successful he can login by using valid user name and password.
4. After Login successful, he will do some operations like searching advertised products, attacking data, viewing all ad products.

## Components

The In-App Add System consists of below components:

### Sign-Up:

Users should sign up with the email id and select their role (Customer/Manager/Employee).

### Login Validation:

Users should type the URL and enters his/her login credentials, validation of credentials is done as a part of this component.

### Dashboard Display:

Displays all the functions the user is currently working on.

# Requirement Specification

## Functional Requirements

### Components Functional Requirement with respect to each module

Initially whenever the employee/Manager/Customer joins the organization, they are given user id and password.

#### Android Module Requirements:

##### Sign-Up:

Customers can sign up with their organizations email id with Role as Customer.

##### Login Validation:

Customer should type the URL and enters his/her login credentials. Validation of credentials is done.

##### Dashboard Display:

Dashboards display all the tickets created by the user with a button “Create Bug” to create new Bug/Ticket.

#### User Module Requirement:

##### Sign-Up:

Manager can sign up with their organizations email id with role as Manager.

##### Login Validation:

Manager should type the URL and enters his/her login credentials. Validation of credentials is done.

##### Dashboard Display:

Dashboard display all the functions assigned to the user for the client.

#### Admin Module Requirement:

##### Login Validation:

Employee should type the URL and enters his/her login credentials. Validation of credentials is done.

##### Dashboard Display:

Display all the Functions of admin currently working on.

### 1.2 Database Requirement

#### 1.2.1 Database Requirement:

Database will have **records of all the request and accepts** created by the user and admin.

Database contains of different tables related to each functionality and the different user.

Tables:

* Account
* Admin
* Advertisement
* AdvRequest
* Evidence
* Hidden\_attack.
* Login
* Malicious
* Recommend\_To\_Freind
* Req\_res
* Request
* Transaction
* User

#### Login Database Requirement:

Login Database will have below fields:

1. Username: Username will be given to each User, Admin and Android-user by the organization
2. Password: Password will be given to each User, Admin and Android-user by the organization.
3. Role: Role User, Admin and Android-user will be given by the admin.

## Non-Functional Requirements

### Security Requirements:

We will be using JavaScript functions for User Authentication and securing user data. Even during DOS attacks, the software must be available and behave consistently. The integrity of the customer account information must be ensured by the software. Any browser that the server cannot authenticate should not be served a restricted web page.

### Privacy Requirements:

The application doesn’t show any personal information of users apart from the email id that will be used while registering. We will be using Java Script for Authenticating the login credentials.

### Scalability and performance Requirements:

All the employees in the organization will be working on the In-App Add tracking system, as a part of their daily job. Hence the web page needs to be highly responsive and should perform at a greater speed, we will be using Apache Tomcat server that is highly scalable.

### Usability Requirement:

Usability plays an important role as this application works as a bridge between the user interface should be well-designed. We will be using Java programming for making web pages user friendly. For the users to learn quickly and understandably, we'll take them on a tour of the options and actions available to them.

### Compatibility Requirement:

Our website should be compatible in working in the following browsers.

1. Chrome Browser Version 98.0.0.0 or above
2. Microsoft Edge Version 90 or above
3. Safari Browser Version 13.1.2 or above

## Interfaces (user, hardware, software, and/or communication)

### Hardware Interface Requirements:

This application would need a browser installed on **Laptop / PC**.

### Software Interfaces Requirements:

1. **HTML**: We will be using HTML for website Designing.
2. **JavaScript**: We will be using JavaScript for making webpages interactive and to connect to the database and server. Also, for email verification, Registration and Login Java Script is used.
3. **Java:** PHP is the language that will be used for developing the web application.
4. **SQL:** We will be using SQL for accessing the database.
5. **Eclipse:** We will be using Eclipse as Integrated Development Environment for the Project
6. **Apache Tomcat Server:** Using the Tomcat server we are going to deploy our project in it.

### User Interface Requirement:

#### Sign-Up Interface:

In Sign-Up interface, Users should sign in with the username and password.

Once the sign-up is done admin gets the notification to allow the user In.

#### Login Interface:

In login interface, Users should type his/her login credentials, validation of credentials is done as a part of this component.

#### Dashboard Interface:

Dashboard interface displays all the functionalities a user can perform.

* Web page will display some options such as “friend request” button in Customer View.
* Web page will display a request accept button in admin View.
* All functions accessed by admin are shown for admin page.

#### Add Creation:

Add Create interface is only available for the Users, to create a new add.

#### Add View/Edit Option:

Bug View/Edit Interface will be same for every role but authorities to edit few fields will vary with the user.

#### Project Report:

This interface is only available for the Manager, to track all the bugs and download report based on the filters available.

# Project Implementing Plan

Below are the three development phases for developing the project:

## 1. Development Phase -1:

### Developing Database tables .

**Requirements:**

We will be using Oracle for Database creation.

### Developing the application layer for the webpage.

**Requirements:**

We will be using JAVASCRIPT for Application layer development.

### Developing Webpages for all Modules.

**Requirements:**

We will be using HTML, AJAX and JQUERY for the development of Webpages.

### Integrating Database with the Front-End applications.

**Requirements:**

We will be using HTML, AJAX, JQUERY and JAVASCRIPT for integration, also for login and registration validation.

## 2. Development Phase-2:

### Develop back-end code for User, Admin and Android-user Requirements:

We will be using Java and JAVASCRIPT for coding each component with respect to User, Admin and Android-user Roles and authorities.

### Integrate Components in the User, Admin and Android-user Modules with the Front-End Applications and the Database.

**Requirements:**

We will be using Eclipse as Integrated development Environment.

### Perform Initial testing.

**Requirements:**

The application will be deployed through Apache Tomcat server, once done few test cases can be prepared and tested as a part of initial testing.

## 3. Development Phase-3:

### Performing integrated system testing

**Requirements:**

Test cases can be prepared to test the complete code and must be documented in the Microsoft Word.

### Preparing test cases and identifying Bugs.

**Requirements:**

Performing repeated testing and resolving bugs and documenting the results by making use of MS Office.

### Fixing the identified bugs.

**Requirements:**

PHP and JavaScript can be used in fixing the bugs in the back end.

### Re-testing until all the requirements are met.

**Requirements:**

XAMPP server is used to deploy the final application.

# Member contribution table

|  |  |  |  |
| --- | --- | --- | --- |
| **Member Name** | **Contribution Description** | **Overall Contribution (%)** | **Note (If Applicable)** |
| **Navyasri Arekatla** | The overall structure of the system, including diagram. Also defines user Interfaces. | 14 |  |
| **Lokesh Bachu** | Project implementation Plan  Phase-1, Non-functional Requirement specification. | 15 |  |
| **Sushma Bobbishetty** | Project implementation Plan  Phase-2. | 15 |  |
| **Naveen Bolgom** | Project implementation Plan  Phase-3. | 14 |  |
| **Sathvik Cheekati** | Deliverable-2 document and member contribution table. | 14 |  |
| **Vamshi Teja Goud Puli** | Interfaces requirement (software, and communication). | 14 |  |
| **Shiva Teja Reddy Yammanuru** | Functional requirement specification. | 14 |  |

# VI MEETING Minutes:

We conducted total of 4 meetings where we discussed about the deliverable 2 agendas and the other following:

**Exchanging updates on specifications.**

**Brief discussion on questions raised in workshop.**

**Final documentation on deliverable 2**

**VII Note: This is considered as a note mentioning the deliverable 2 file is completed.**