

# **REMIND US**

## **A PROJECT REPORT**

*in partial fulfillment for the award of the degree of*

## **BACHELOR OF TECHNOLOGY**

**IN**

## **COMPUTER SCIENCE ENGINEERING**

*Under the Guidance of*

**PALLABI SAHA**

*Project Carried Out At*



**Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)**

**SDF Building, Module#132,Ground Floor , Salt Lake City, CP Block,Sector-5**

**Kolkata - 700 091**

*Submitted By*

**KOUSTAV GHOSH**



**TECHNO INDIA SALT LAKE (MAIN)**

**KOLKATA**

In association with



*(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)*

1. Title of the Project: Remind Us

2. Project Members: 1. Imanul Anam Siddique 2. Sk Hosanur Rahaman 3. Warisa Nusrat 4. Koustav Ghosh 5. Sandipan Biswas.

Name and Address of the Guide: Miss. Pallabi Saha

Trainer (ANDROID)

Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)

Module 132, SDF Building, Sector 5, Salt Lake, Kolkata - 64

3. Project Version Control History

Version	Primary Authors	Description of version	Date Completed
Final	1. Imanul Anam Siddique 2. Sk Hosanur Rahaman 3. Warisa Nusrat 4. Koustav Ghosh 5. Sandipan Biswas.	Project Report	

1.

4.

2.

5.

3.

Signature of Approval

Signatures of Team Members

Date:

Date:

**For Office Use Only**

Miss. Pallabi Saha  
Project Proposal Evaluator

# PROJECT RESPONSIBILITY FORM

## REMIND US

### Group No 1

SI No.	Name	Responsibility
1.	Imanul Anam Siddique	Project Leader and Coding
2.	Sk Hosanur Rahaman	Code modifying,Co leader,Designing
3.	Warisa Nusrat	Designing and Code testing
4.	Koustav Ghosh	Coding and documentation
5.	Sandipan Biswas	Coding,Designing,Testing

Each group member must participate in project development and developing the ideas for the required elements. Individual group members will be responsible for completing tasks which help to finalize the project and the performance. All group members must be assigned a task.

NAME OF THE STUDENTS	SIGNATURE
1. Imanul Anam Siddique	
2. Sk Hosanur Rahaman	
3. Warisa Nusrat	
4. Koustav Ghosh	
5. Sandipan Biswas	

Date:

## DECLARATION

I hereby declare that the project work being presented in the project proposal entitled "REMIND US" in partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE ENGINEERING at ARDENT COMPUTECH PVT. LTD, SALT LAKE, KOLKATA, WEST BENGAL is an authentic work carried out under the guidance of Miss. PALLABI SAHA. The matter embodied in this project work has not been submitted elsewhere for the award of any degree of our knowledge and belief.

Date:

Name of the student: Koustav Ghosh

Signature of the student:



**Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)**

**SDF Building, Module#132,Ground Floor , Salt Lake City, CP Block,Sector-5**

**Kolkata - 700 091**

## **CERTIFICATE**

This is to certify that this proposal of minor project entitled “REMIND US” is a record of bona fide work, carried out by 1. Imanul Anam Siddique 2. Sk Hosanur Rahaman 3. Warisa Nusrat 4. Koustav Ghosh 5. Sandipan Biswas. under my guidance at ARDENT COMPUTECH PVT LTD. In my opinion, the report in its present form is in partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE ENGINEERING and as per regulations of the ARDENT®. To the best of my knowledge, the results embodied in this report, are original in nature and worthy of incorporation in the present version of the report.

## **Guide / Supervisor**

---

**Ms. Pallabi Saha**

Trainer(ANDROID)

Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)

Module 132, SDF Building, Sector 5, Salt Lake, Kolkata - 64

## **ACKNOWLEDGEMENT**

I would like to express our gratitude to Pallabi Saha ,my project advisor for providing me an opportunity to work on this project, which gave me a great exposure to develop the skills towards Android Application Development. I thank her for her generous advice and encouragement which helped us to complete the project successfully.

Lastly, I would like to thank the entire faculty and staff of our Departments.

**Koustav Ghosh**

**(Techno India Saltlake Main**

**B.Tech in CSE 2<sup>nd</sup> Year,3<sup>rd</sup> semester )**

## TABLE OF CONTENTS

S.NO.	NAME OF THE TOPIC	PAGE NO
1.	<b>INTRODUCTION</b>	9
1.1	Objective	10
1.2	Scope	10
2.	<b>System Analysis</b>	11
2.1	Identification of Need	11
2.2	Feasibility Study	12
2.3	Work flow	12
2.4	Functional requirement	13
2.5	Non Functional requirement	14
2.6	Hardware and software requirement	14
3.	<b>System design</b>	15
3.1	Data flow diagram	16
3.2	Entity-relationship diagram	18
3.3	Database design	19
3.4	User interface design	21
	<b>Snapshots</b>	27
4.	<b>Implementation and testing</b>	28
4.1	introduction	28
4.2	Objective of testing	29
4.3	Test cases	30-33
5.	<b>Software system attributes</b>	34
5.1	Database security measures	34
5.2	System security measures	34
5.3	limitations	34
6.	<b>conclusion</b>	35
7.	<b>Future scope and further enhancements</b>	36
8.	<b>References</b>	36

A large, light orange oval with a subtle gradient and a thin dark outline, centered on the page.

**REMIND US**



## §1. INTRODUCTION

There are bunch of schedule reminder android applications available today on play store providing static notification to user before a specified interval of time. The static notifications will not be proven to be useful when travel time for user is considerably large with respect to reminder set for fixed time. Also this could cause delays in arriving at event location or even missing an event.

**“REMIND US”** is a solution to the above problem. This is a location based reminder application which dynamically notifies user so that they can be on or before time for an event. The application extracts event details from Google calendar. The event details are assumed to be pre-entered by user in Google calendar, and then application synchronizes the data with Google calendar. It will run in background to access the user’s current location and calculate travel time with respect to mode of travel selected by user.

The application allows user to edit/modify each schedule to set the importance as well as mode of travel. Also user can do profiling for location access and reminder settings. Based on the provided details and policies set by user the application will intelligently notify user about the scheduled event. The notification sent to user also includes the travel information to start navigation.

## **1.1 OBJECTIVES**

REMIND US is a type of app that helps to manage time. The goal of this project is to create an Android App to help people manage events and important tasks like taking their medicine, taking their food, attending in any ceremony in proper time. In order to better define the behavior of this system we devised a requirement further clarifying what functionality this system is required to fulfill, such as functionality and constraints. Furthermore, we specified that this system has to be implemented using an SQLite database.

## **1.2 SCOPE**

The project is defined by the boundaries of the selection process and our narrowing of the project definition. Project selection was completed by meeting together and each person suggesting an idea. After discussion of the upsides and downsides of the various ideas, our team came to a unified agreement on a reminder system. Review of the project goals, deliverables, tasks to complete, their associated costs and deadlines, further narrowed our project to a precise application idea.

## **§2. SYSTEM ANALYSIS**

### **2.1 IDENTIFICATION OF NEED**

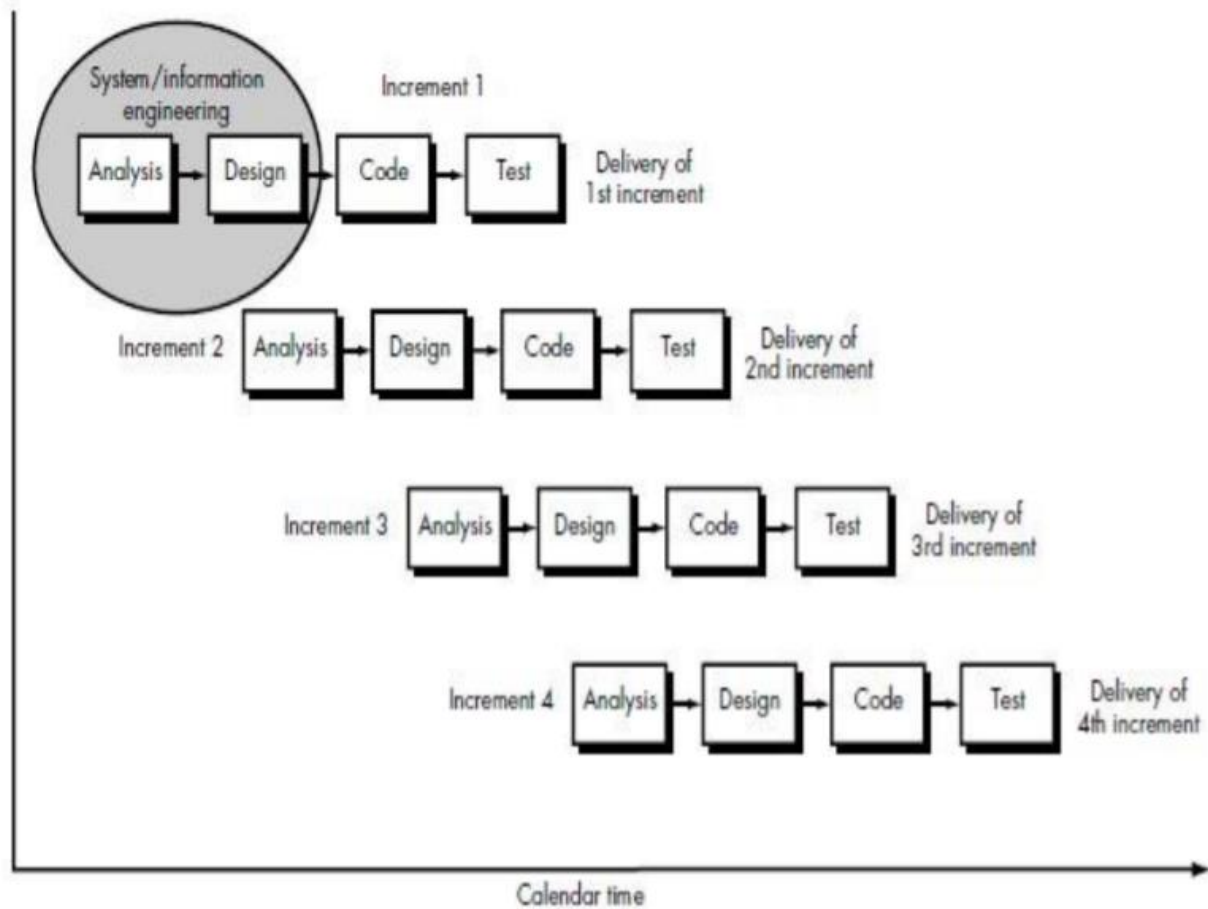
System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organization are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action. A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

### **2.2 FEASIBILITY STUDY**

Feasibility study is made to see if the project on completion will serve the purpose the organization for the amount of work. Effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development. The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and operational feasibilities.

## 2.3 WORK FLOW

This Document plays a vital role in the development life cycle (SDLC) as it describes the complete requirement of the system. It is meant for use by the developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process. INCREMENTAL MODEL was being chosen .



The developer is responsible for:-

- Developing the system, which meets the SRS and solving all the requirements of the system?
- Demonstrating the system and installing the system at client's location after the acceptance testing is successful.
- Submitting the required user manual describing the system interfaces to work on it and also the documents of the system.
- Conducting any user training that might be needed for using the system.
- Maintaining the system for a period of one year after installation.

## 2.4 FUNCTIONAL REQUIREMENTS

### Modules:

The modules used in this software are as follows:

- **Create event**-user can create an event by clicking on the floating button
- **Set name & description of the event**-user can save the event by giving particular name & description
- **Set date & time**-user can set the required date & time for the accuracy of getting notifications
- **Set event**-then finally user can set the event easily by clicking on save event button

## 2.5 NON-FUNCTIONAL REQUIREMENTS

- **Usability Requirement:** The system shall allow the users to access the system from any browsers, no special training is required. The system user friendly and the system is written in simple English.
- **Availability Requirement:** The system is available 100% for the user and is used by 24 hours a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.
- **Accuracy:** The system should accurately provide real time information taking into consideration various issues. The system shall provide 100% access reliability.

- **Performance Requirement:** The information is refreshed at regular intervals depending upon whether some updates have occurred or not. The system shall respond the member in less than 2 seconds.
- **Security Requirement:** System will use a secured database and the system will have different users and each user has different types of constraints.
- **Reliability Requirement:** The system has to be 100% reliable due to the importance of data and the damages that can be caused by incorrect data. The system will run 7 days a week and 24 hours a day.

## 2.6 HARDWARE and SOFTWARE REQUIREMENTS

### HARDWARE REQUIREMENTS

- Computer that has a 1.6GHz or faster processor
- 8 GB RAM and above is required to make android studio work efficiently.
- HDD 20 GB Hard Disk Space and Above Hardware Requirements5400 RPM hard disk drive
- DirectX 9 capable video card running at 1024 x 768 or higher-resolution display
- DVD-ROM Drive

### SOFTWARE REQUIREMENTS

- WINDOWS 7 and above
- Android Studio
- Notepad++
- Online Server

## §3. SYSTEM DESIGN

### 3.1 DATA FLOW DIAGRAM

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. A structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc. Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

#### DATA FLOW DIAGRAM SYMBOLS



Source or Destination of Data



Data Flow



Process



Storage

## Steps to Construct Data Flow Diagram

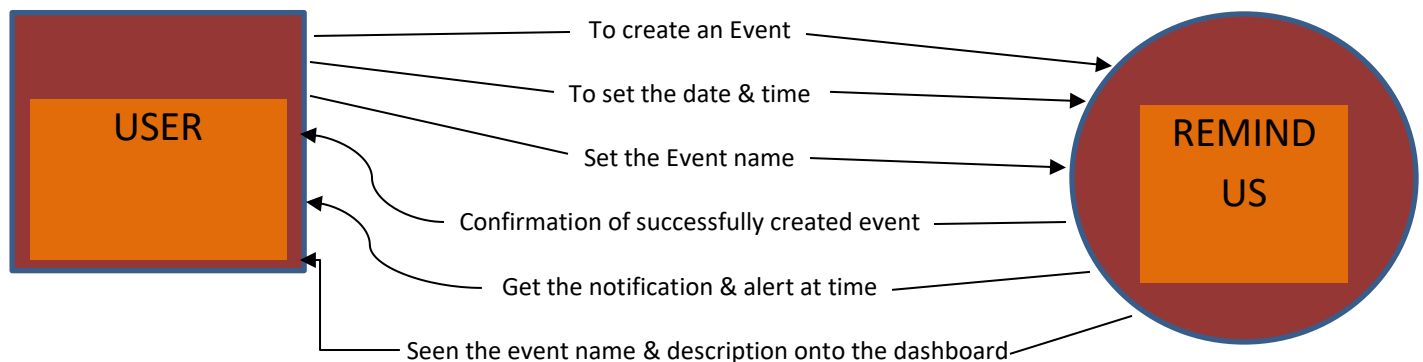
Four Steps are generally used to construct a DFD.

- Process should be named and referred for easy reference. Each name should be representative of the reference.
- The destination of flow is from top to bottom and from left to right.
- When a process is distributed into lower level details they are numbered.
- The names of data stores, sources and destinations are written in capital letters.

### Rules for constructing a Data Flow Diagram-

- Arrows should not cross each other.
- Squares, Circles, Files must bear a name.
- Decomposed data flow squares and circles can have same names.
- Draw all data flow around the outside of the diagram.

## DATA FLOW DIAGRAM





A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. A structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

- Arrows should not cross each other.
- Squares, Circles, Files must bear a name.
- Decomposed data flow squares and circles can have same names.
- Draw all data flow around the outside of the diagram.

## ENTITY RELATIONSHIP DIAGRAM

In software engineering, An **entity–Relationship model (Er model)** is a data model for Describing The Data Or Information Aspects Of A Business Domain Or Its Process Requirements, In An Abstract Way That Lends Itself To Ultimately Being Implemented In a database such as a relational database. The main components of er models are entities (Things) And the relationships that can exist among them. However, Variants Of The Idea Existed Previously, And Have Been Devised Subsequently Such As Super Type And Subtype Data Entities And Commonality relationships.

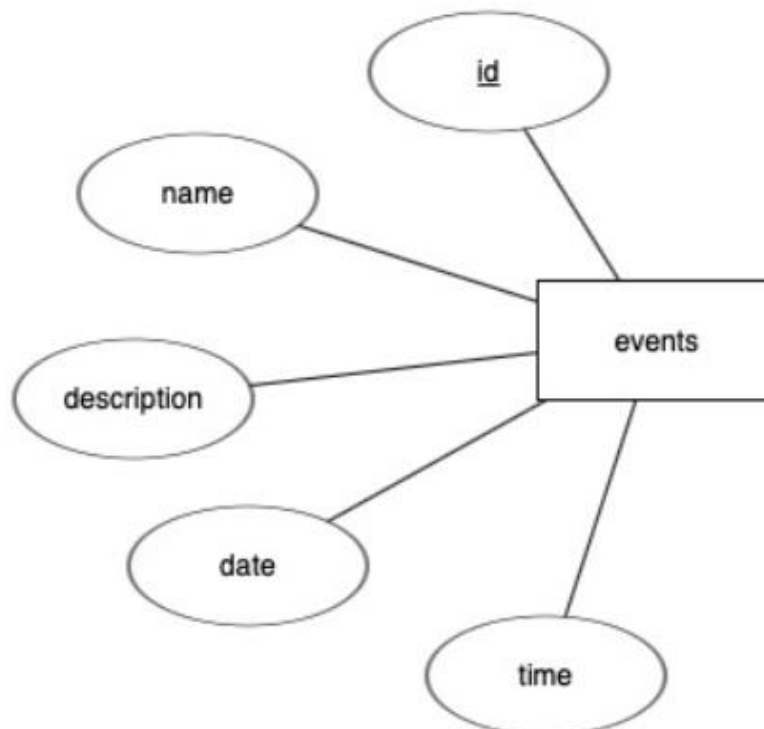
An Entity–Relationship Model Is A Systematic Way Of Describing And Defining A Business Process. The Process Is Modeled As Components (*Entities*) That Are Linked With Each Other By *relationships* that Express The Dependencies And Requirements Between them, Such as: *One building maybe divided into zero or more apartments, but one apartment can only be located in one building.* Entities may have various properties (*attributes*) that characterize them. Diagrams

created to represent these entities, attributes, and relationships graphically are called entity–relationship diagrams.

An ER model is typically implemented as a database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables point to indexes in other tables; such pointers represent the relationships.

The three schema approach to software engineering uses three levels of ER models that may be developed. An entity may be defined as a thing capable of an independent existence that can be uniquely identified. An entity is an abstraction from the complexities of a domain. When we speak of an entity, we normally speak of some aspect of the real world that can be distinguished from other aspects of the real world.

### E-R diagram:-



**Figure: Entity-Relationship Diagram**

## DATABASE DESIGN

A database is an organized mechanism that has capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is two level processes. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called information Level design and it is taken independent of any individual DBMS.

In the following snapshots we display the way we have used SQL Server as the backend RDBMS for our project and the various entities that have been used along with their table definition and table data.

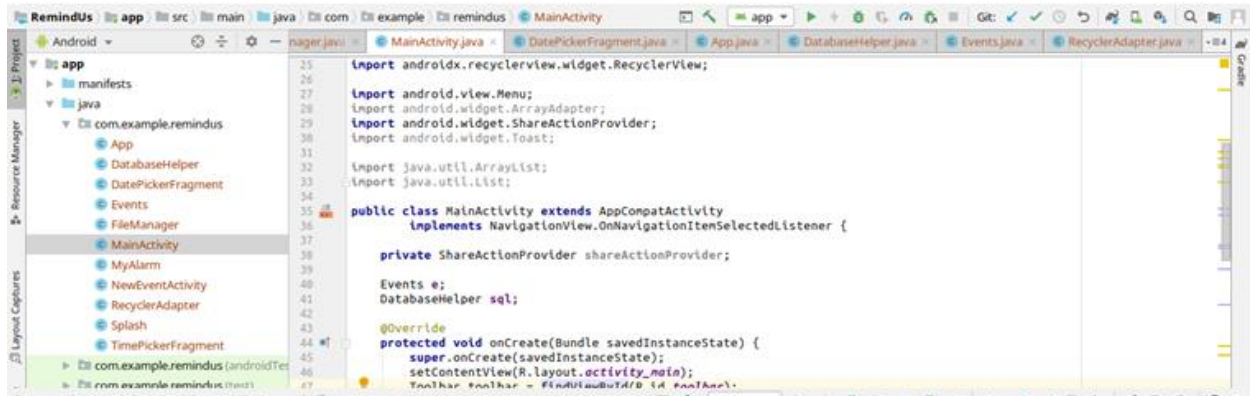


Fig:MainActivity.java

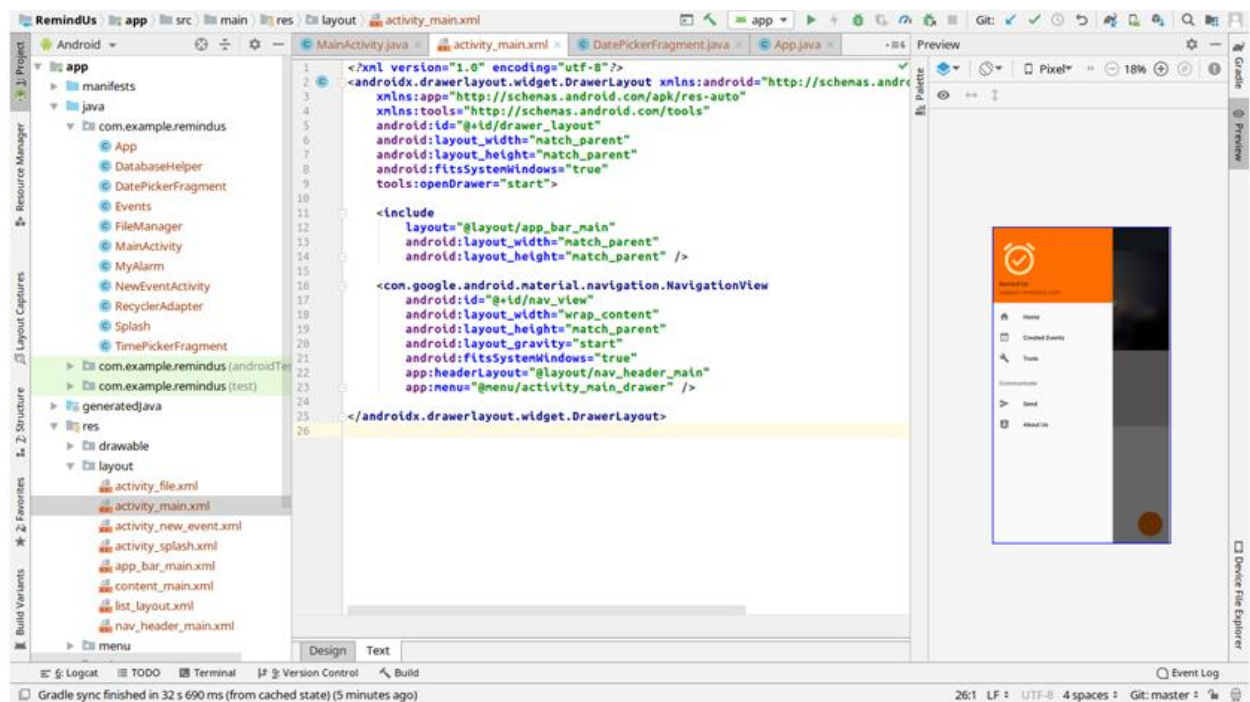


Fig:activity\_main.xml

### 3.4 USER INTERFACE DESIGN

**User interface design (UID)** or **user interface engineering** is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design). Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. Graphic design and typography are utilized to support its usability, influencing how the user performs certain interactions and improving the aesthetic appeal of the design; design aesthetics may enhance or detract from the ability of users to use the functions of the interface. The design process must balance technical functionality and visual elements (e.g., mental model) to create a system that is not only operational but also usable and adaptable to changing user needs. Interface design is involved in a wide range of projects from computer systems, to cars, to commercial planes; all of these projects involve much of the same basic human interactions yet also require some unique skills and knowledge. As a result, designers tend to specialize in certain types of projects and have skills centered on their expertise, whether that be software design, user research, web design, or industrial design.

## §4. Code snippets from the implementation

### ➤ MainActivity.java:

```
package com.example.remindus;

import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.DialogFragment;

import android.app.Activity;
import android.app.DatePickerDialog;
import android.app.TimePickerDialog;
import android.content.Context;
import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ScrollView;
import android.widget.TextView;
import android.widget.TimePicker;

import org.w3c.dom.Text;

import java.util.ArrayList;
import java.util.Date;
import java.util.List;

import de.mateware.snacky.Snacky;

public class NewEventActivity extends AppCompatActivity implements
    TimePickerDialog.OnTimeSetListener, DatePickerDialog.OnDateSetListener
{
    EditText name, desc;
    TextView tvTime, tvDate;
    Button btnTime, btnDate, btnSave;
    String date, time;
    DatabaseHelper db;
```

```

Activity activity;
Context context;View myview;
ScrollView scrollView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_new_event);
        context=getApplicationContext();
        scrollView=findViewById(R.id.scview);

        btnDate = findViewById(R.id.btnDate);
        btnTime = findViewById(R.id.btnTime);
        name = findViewById(R.id.name);
        desc = findViewById(R.id.desc);
        btnSave = findViewById(R.id.btnSave);

        btnSave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                db = new DatabaseHelper(getApplicationContext());

db.addUser(name.getText().toString(),desc.getText().toString(),time,date);

                Snacky.builder().setView(scrollView).setBackgroundColor(Color.parseColor("#FF6C02")).setText("Event saved successfully").setActionText("Ok").setActionClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View view) {
                        startActivity(new Intent(NewEventActivity.this,MainActivity.class));
                    }
                }).setDuration(Snacky.LENGTH_INDEFINITE).success().show();

            }
        });

        btnDate.setOnClickListener(new View.OnClickListener() {
            @Override

```

```

        public void onClick(View view) {

            DialogFragment datePicker = new DatePickerFragment();
            datePicker.show(getSupportFragmentManager(), "date
picker");

        }
    });

    btnTime.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {

            DialogFragment timePicker = new TimePickerFragment();
            timePicker.show(getSupportFragmentManager(), "time
picker");

        }
    });

    @Override
    public void onTimeSet(TimePicker timePicker, int i, int i1) {

        tvTime = findViewById(R.id.tvTime);

        if (i<10 && i1<10) {
            if (i <= 12)
                tvTime.setText("0" + i + ":" + "0" + i1 + " AM");
            else
                tvTime.setText("0" + (i - 12) + ":" + "0" + i1 + "
PM");
        }
        else if (i<10 && i1>10) {
            if (i<=12)
                tvTime.setText("0" + i + ":" + i1 + " AM");
            else
                tvTime.setText("0" + (i-12) + ":" + i1 + " PM");
        }
        else if (i>10 && i1<10) {
            if (i<=12)
                tvTime.setText(i + ":" + "0" + i1 + " AM");
            else
                tvTime.setText((i-12) + ":" + "0" + i1 + " PM");
        }
        else if (i>10 && i1>10){

```



```

        if (i<=12)
            tvTime.setText(i + ":" + i1 + " AM");
        else
            tvTime.setText((i-12) + ":" + i1 + " PM");
    }
    time = tvTime.getText().toString();

}

@Override
public void onDateSet(DatePicker datePicker, int i, int i1, int
i2) {

    i1++;
    tvDate = findViewById(R.id.tvDate);

    if (i1<10 && i2<10)
        tvDate.setText("0" + i2 + "/" + "0" + i1 + "/" + i);
    else if (i1<10 && i2>10)
        tvDate.setText( i2 + "/" + "0" + i1 + "/" + i);
    else if (i1>10 && i2<10)
        tvDate.setText("0" + i2 + "/" + i1 + "/" + i);
    else if (i1>10 && i2>10)
        tvDate.setText( i2 + "/" + i1 + "/" + i);

    date = tvDate.getText().toString();

}

List<String> data = new ArrayList<>();
}

```

## ➤ Activity\_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<androidx.drawerlayout.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:id="@+id/drawer_layout"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:fitsSystemWindows="true"

    tools:openDrawer="start">

<include

    layout="@layout/app_bar_main"

    android:layout_width="match_parent"

    android:layout_height="match_parent" />

<com.google.android.material.navigation.NavigationView

    android:id="@+id/nav_view"

    android:layout_width="wrap_content"

    android:layout_height="match_parent"

    android:layout_gravity="start"

    android:fitsSystemWindows="true"

    app:headerLayout="@layout/nav_header_main"

    app:menu="@menu/activity_main_drawer" />
```

## SNAPSHOTS OF APPLICATION

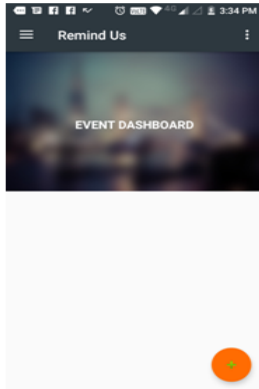


Fig:Dashboard Activity

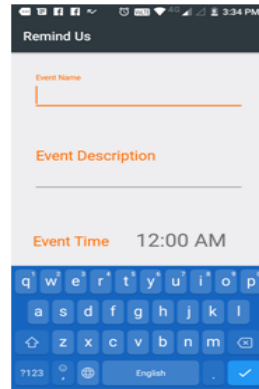


Fig:Activity New Event

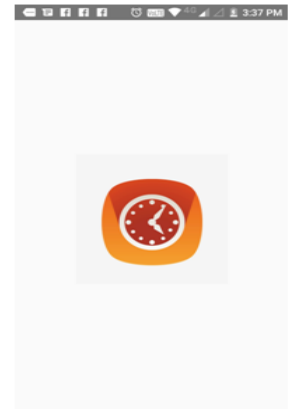


Fig:Activity Splash

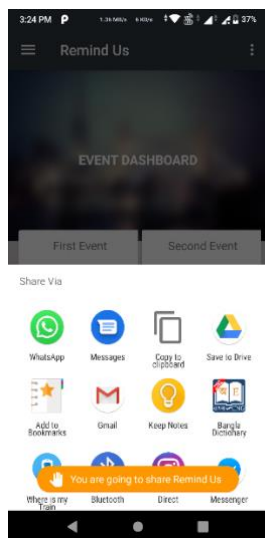


Fig.-share activity

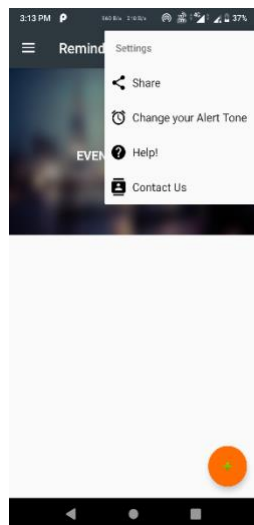


Fig.-settings activity

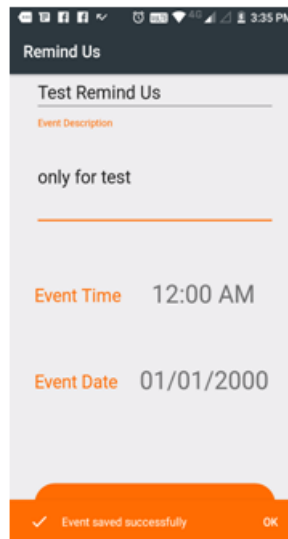


Fig.-Event create activity

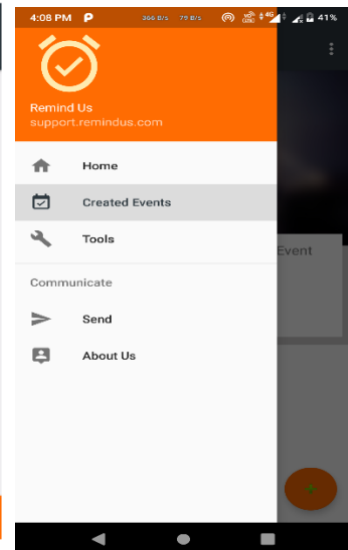


Fig.-Nav Bar activi

# **IMPLEMENTATION AND TESTING**

## **INTRODUCTION:**

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

The implementation is the final and important phase. It involves user-training, system testing in order to ensure successful running of the proposed system. The user tests the system and changes are made according to their needs. The testing involves the testing of the developed system using various kinds of data. While testing, errors are noted and correctness is the mode.

## OBJECTIVES OF TESTING

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. Our user interface to utilize these functions is designed to be user-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability.

### Process Overview

The following represents the overall flow of the testing process:

1. Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.
2. Identify which particular test(s) will be used to test each module.
3. Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.
4. Identify the expected results for each test. Successful unit testing is required before the unit is eligible for component integration/system testing.
5. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.

## TEST CASES

A test case is a document that describe an input, action, or event and expected response, to determine if a feature of an application is working correctly. A test case should contain particular such as test case identifier, test condition, input data Requirement expected results. The process of developing test cases can help find problems in the requirement or design of an application, since it requires completely thinking through the operation of the application.

### **TESTING STEPS:-**

#### **UNIT TESTING**

Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.

#### **INTEGRATION TESTING**

Data can be lost across an interface. One module can have an adverse effect on another, subfunctions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. The objective is to take unit tested module and build program structure.

## **VALIDATION**

At the culmination of the integration testing, Software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

Tested By:	Sk Hosanur Rahaman
Test Type:	Unit Testing
Test Case Number:	1
Test Case Name:	Notification Alert Tone
Test Case Description:	The user should set the Event name,Description & then have to click save button,otherwise don't show anything on the Remainder Event Dashboard
Item(s) to be tested	
1	Show the created event name & description is updated or not
Specifications Input	<b>Expected output/result</b>
1) Tap on the save button	2) Successfully saved & updated the name & description of Event within the Dashboard 3) Not updated anything on the Dashboard or saved an empty one

<b>Tested By:</b>	<b>Imanul Anam Siddique</b>	
<b>Test Type:</b>	Unit Testing	
<b>Test Case Number:</b>	2	
<b>Test Case Name:</b>	Alarm Activity	
<b>Test Case Description:</b>	The user should set the Event by saving required time & date. Then click on ok option & then will get the Remainder alert perfectly.	
<b>Item(s) to be tested</b>		
<b>1</b>	The Remainder Alert tone on that particular date & time plays.	
<b>Specifications Input</b>	<b>Expected output/result</b>	
<b>1)</b> Tap on the ok option & wait for certain required time & date	4) Successfully saved the event & getting alert tone in time 5) Not updated or may be updated but not getting any Alarm	



Tested By:	Sandipan Biswas
Test Type:	Unit Testing
Test Case Number:	3
Test Case Name:	Pop Up Notification
Test Case Description:	The user should set the Event by saving required time & date. Then have to click on ok option & then will show pop up notification on the notification bar
Item(s) to be tested	
1	Notifications on the notification bar shows in a proper way or nothing
Specifications Input	Expected output/result
2) Tap on the ok option & wait for certain required time & date	6) Successfully saved the event & get a notification regarding the Remainder

## SOFTWARE SYSTEM

### ATTRIBUTES

#### ❖ DATABASE SECURITY

System security measure is meant to be provided to make your system reliable & secured from unauthorized

User may create threats to the system. So you should follow some security measures. We have used security

Levels in database level at system level.

#### ❖ SYSTEM SECURITY

If we talk about the system security in our proposed system we have implemented with the help of maintain

The session throughout the systems use.

#### ❖ LIMITATIONS

- People who are not familiar with computer or with android can't access this application.
- If the time & date not setted perfectly according to the requirments, user don't get any alert

## 6. CONCLUSION

This project has been appreciated by all the users in the organization. It is easy to use, since it uses the GUI provided in the user dialog. User friendly screens are provided. The usage of software increases the efficiency, decreases the effort. It has been efficiently employed as a Site management mechanism. It has been thoroughly tested and implemented.

The project “**REMIND US**” is the ideal place for every cooking fan who will get all the ingredients and recipe about the dish. It will provide user the benefit of cooking just with a click sitting in his home.

The software collects the every information of the ingredients and recipe. The software provides a reliable platform for keeping all cooking information. For this kind of online recipe app, the special software must be installed on the server which host the site, or on a secure server which receives all sensitive data.

## FUTURE SCOPE AND FURTHER ENHANCEMENTS

In future, we would like to keep working on this project & make new additions to provide users with more advanced & reliable features & more detailed information. We have set our sights on the following additions in the future:-

1. Addition of Sticky Note option, where user can set their important notes as usual. May use as note reminder.

2. Addition of group such as, friends, relatives, colleagues, etc. Where anyone from that group can set an event & that can be affected to all the other members who ever wanted to do the same.

3. Through sticky notes, anyone can send a note to the suspected group or a particular member of that group.

## 8. REFERENCES

1. <https://www.youtube.com>
2. <https://www.google.com>

A large, horizontally-oriented oval with a gradient of orange colors, from a lighter shade at the top to a darker shade at the bottom. It has a thin black outline.

THANK YOU

