# CliniConnect

# Team1

Benjamin Chrysler
Mcduff luke j
Shweta parihar
Chaitanya sri Patluri

# Project Goal and Objectives (revised)

**Overall goal:** This application is to improve contact between patient (in particular, low income demographic) and there clinic which they attend.

**Objectives:**

We are planning to create a mobile application that will do the following:

- Patient can fill the form prior to clinic appointment, which in turn reduce the patient and doctor waiting time and will also improve the wait time in busy inner city clinics.
- Patient can fill the exit satisfaction form as per his comfort.
- Patient will be able to get his lab order for bloodwork etc.… electronically, which can be scanned.
- Patient will get remainder notification of his appointment.
- Patient who are required to monitor their blood pressure and blood sugar daily or weekly can keep track of their readings using this application.
- A map that will show from whatever location opened, the directions to the clinic, this would benefits patients who may require rides from friends or family members unfamiliar with the location of the clinic.
- Administrator will be able to administer the application using admin page.

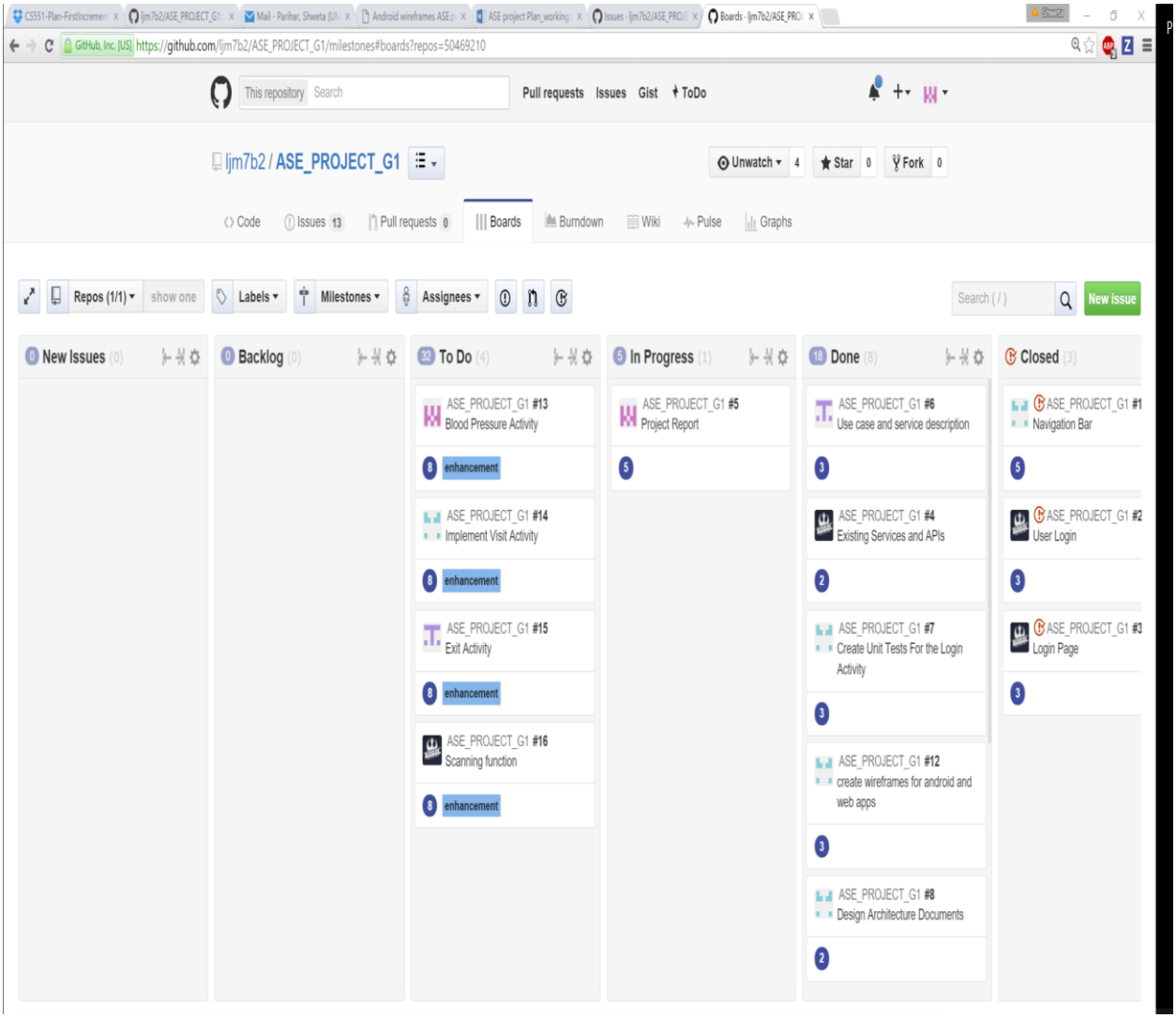
**Features**:

- New patient can register in the registration page of the application.
- Already registered patient can login to the application using the login page.
- Prior clinic visit form page with the exact replica of the actual physical form.
- Exit satisfaction form page.
- Bloodwork details page with the facility to be scanned by pathology labs.
- Reminder icon on the status bar for the upcoming appointment.
- Blood pressure page for updating and keeping track of daily or weekly blood pressure readings.
- Blood sugar page for updating and keeping track for blood sugar readings.
- A map that will show from whatever location opened, the directions to the clinic, this would benefits patients who may require rides from friends or family members unfamiliar with the location of the clinic.
- Administrator can add new patient to the system, send blood work notification to the patient.

**Significance:** While some patient-to-clinic apps exist, they are typically patient portals which can contain an overwhelmingly large amount of data and features. By streamlining a few key important services we can help increase patient health.

# Project Plan

**Zenhub screenshot**




**Schedule for the four different increments.**

**Stories (Issues):** This first increment lays the foundation of the user application side of our service. We have a User Login Page in which users can input their credentials to access the App. Also, we have a drawer style navigation bar that will connect to the other activities that we will implement in future increments. Both of these are on the user side of the software, that will be accessed through android devices.

**Service Design: The** service design at this stage, is to create a base application that a User can log into after visiting the clinic in the first place. Users will only be able to login once they have visited the clinic. After, the service is meant to make interactions between the User and the clinic to be more immediate and more transparent. The Unit testing progressed by individually isolating the login from the navigation bar. Within the navigation bar, the widgets for the phone and the email connections were also tested by themselves before being connected with the navigation bar.

**Service Implementation:** The service at this point is only being implemented in xml and java. Since our service is currently concentrated on the User side, we will only be using xml and java for this. The service is currently tested on a variety of devices and device emulators.

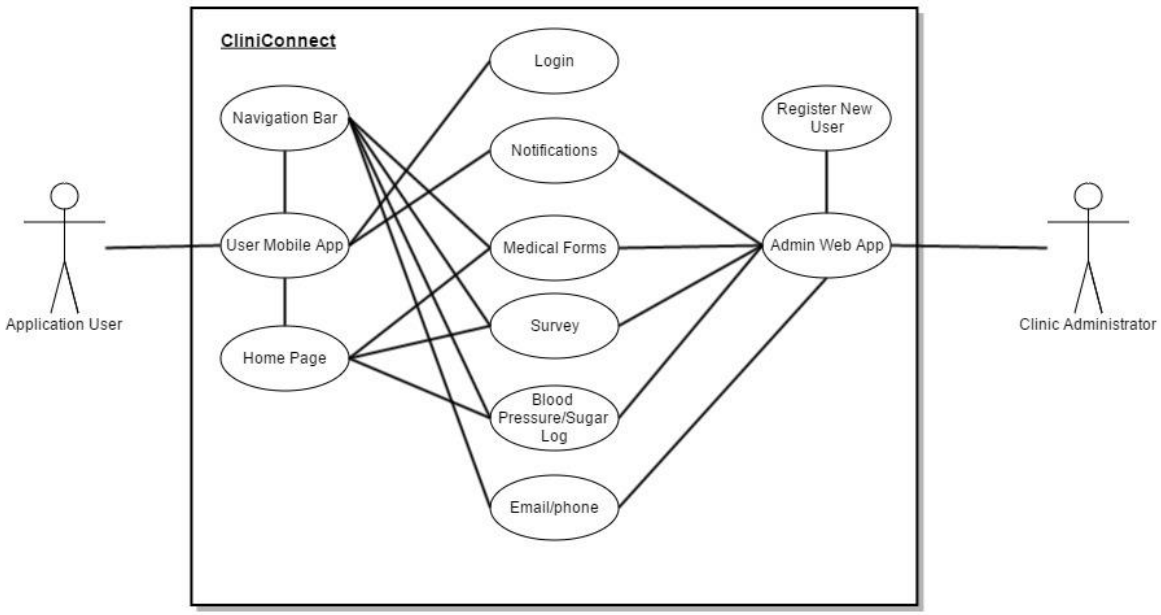
**Project Timelines, Members, Task Responsibility:**

The work division has been made in Zenhub please check our board for more detail.

https://github.com/ljm7b2/ASE_PROJECT_G1/milestones#boards?repos=50469210

**User Stories:** When the User wants to initially access the application the user will enter their email and predetermined password to access the application. When the User wants to Navigate to any other page the User can use the navigation bar to redirect to another page. When the user wants to contact the clinic the User can use the navigation bar to access communication links, calling and emailing that will put the user in touch with the clinic.

**Use Case**

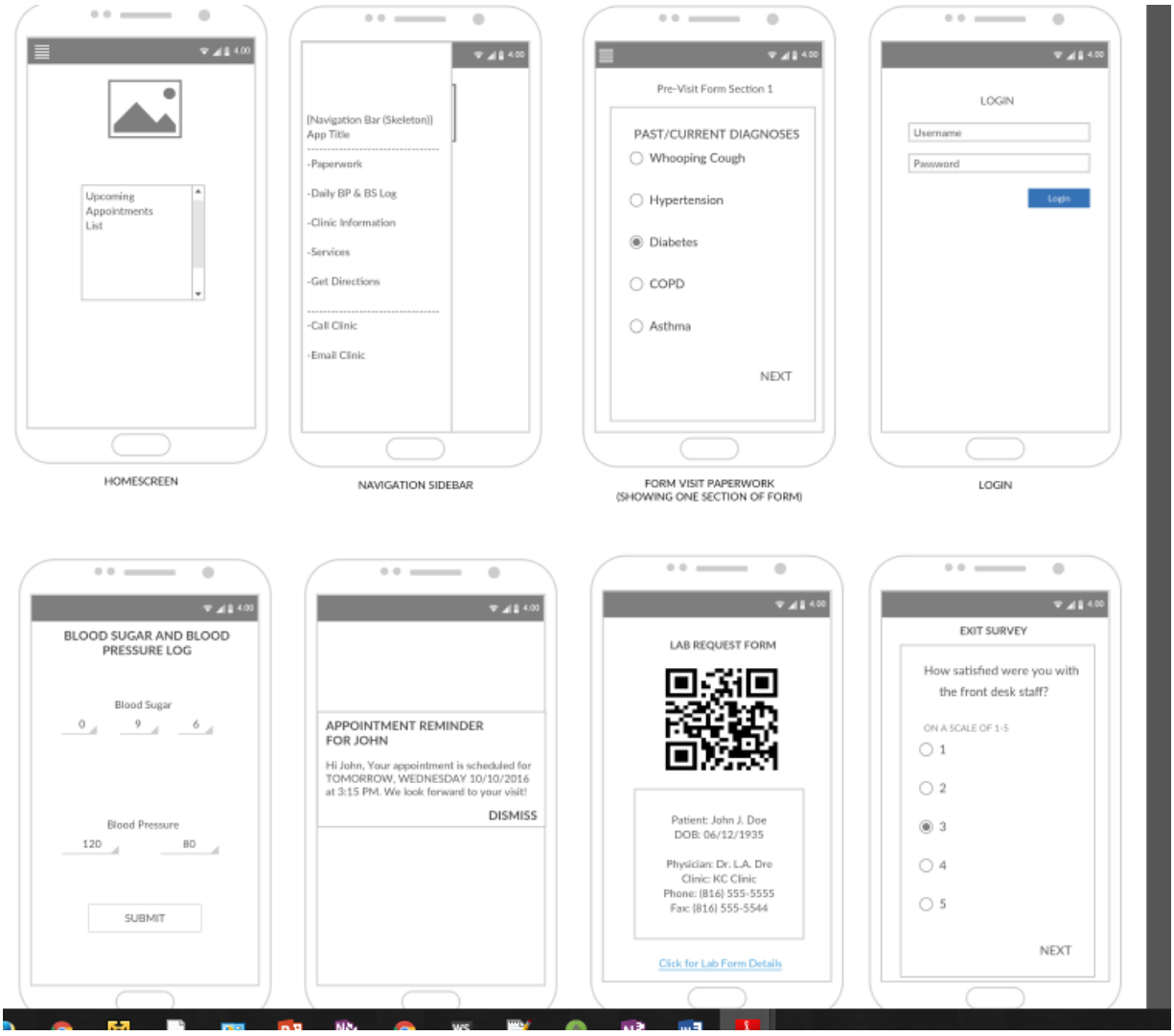


**Service description:** CliniConnect is an application that helps patients handle their paperwork in an organized and timely fashion. The application is intended for those who either do not have the time to come early to an appointment to fill out paperwork, have a tendency to miss/forget appointments and even those who need more constant interaction with their medical clinic.

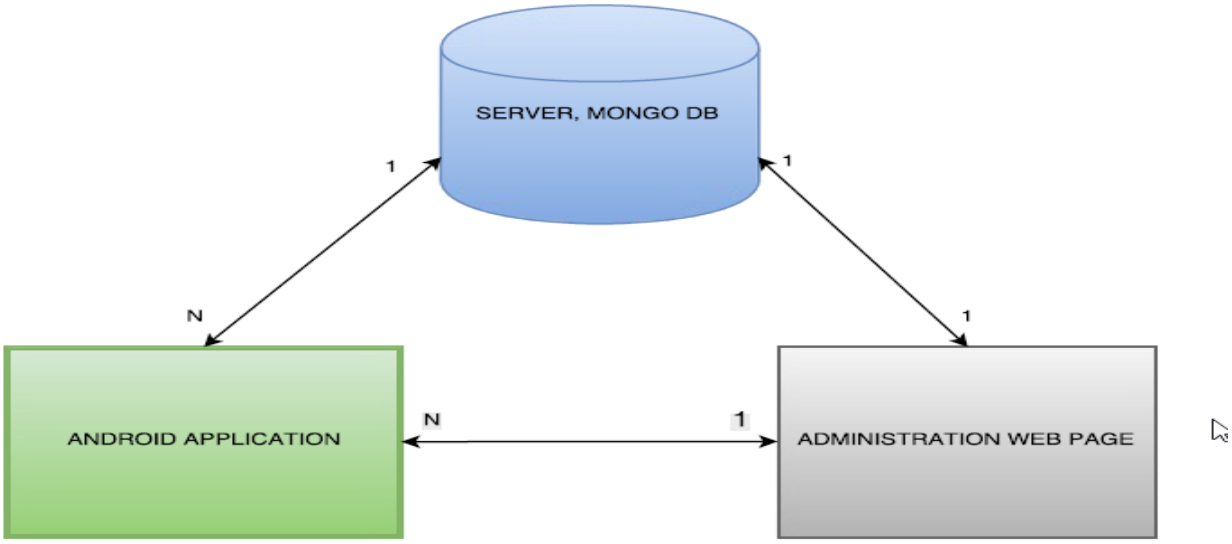
# First Increment Report

In this first increment of "**CliniConnect"** we have designed the overall structure and flow of the application using wireframes and UML diagrams. In the initial face we have begun with implementing the login activity and navigation sidebar.

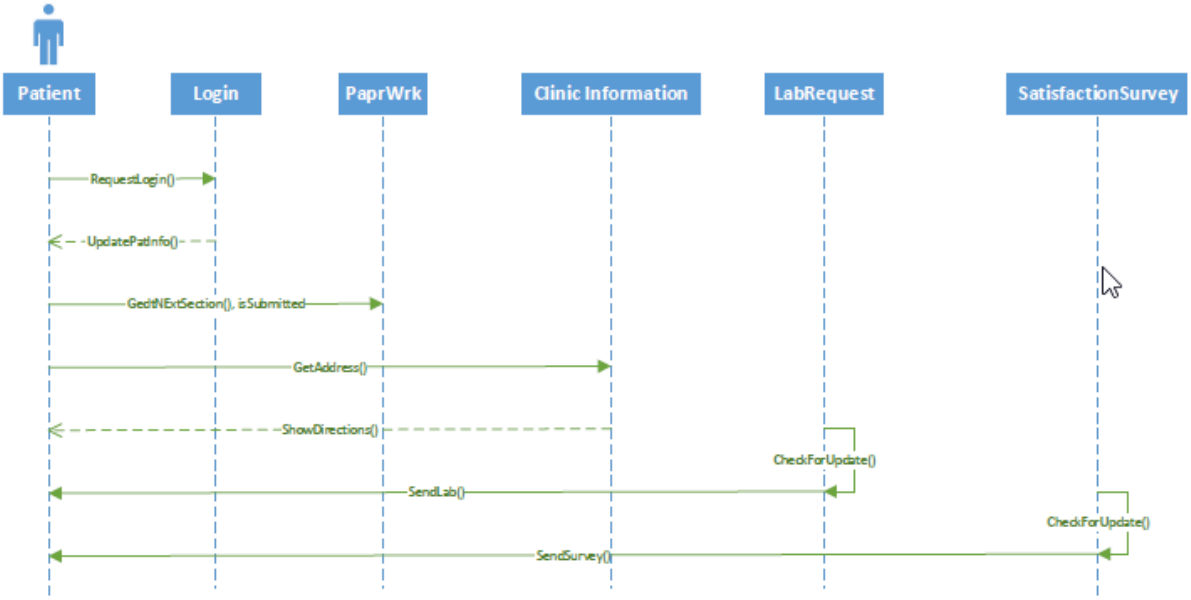
## Detail Design: Wireframes



# ARCHITECTURE STACK

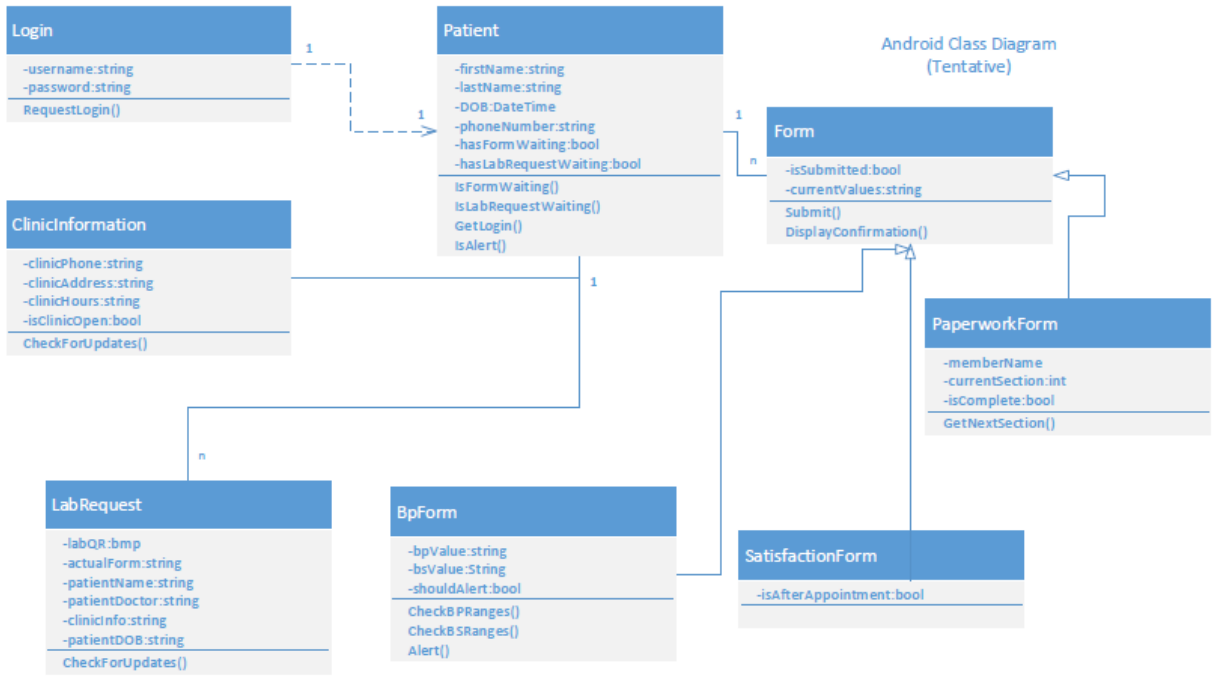



## Sequence diagram




ANDROID SEQUENCE DIAGRAM (TENTATIVE)

## Class diagram




## Used existing Services/API

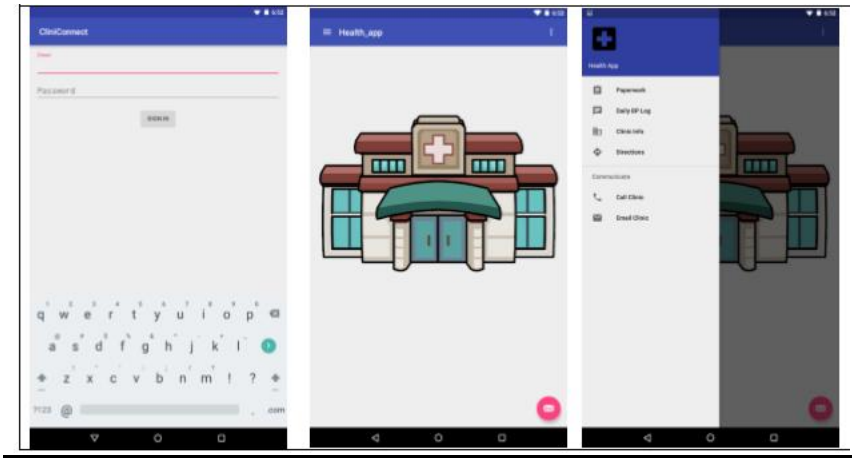
**Google Maps API**: Used to get directions from the user's current location to the clinic in Google Maps.

**Widgets:**

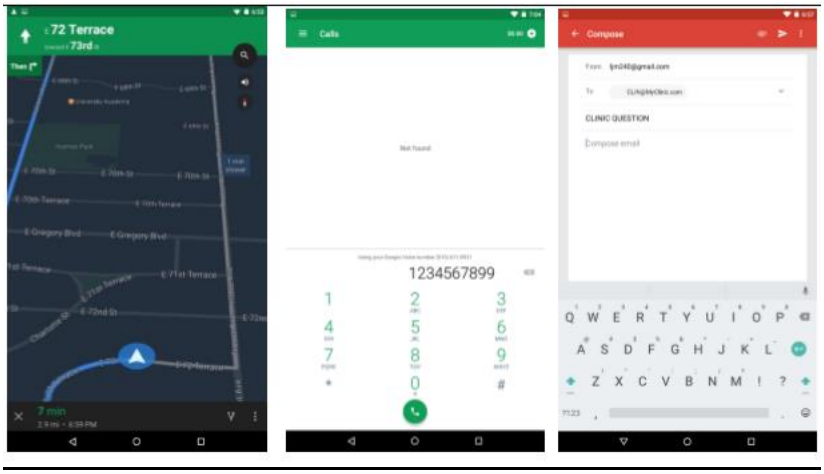
- **Phone Widget**: When selected, the user can call the clinic using existing phone services on the phone like Google Hangouts Dialer or the Phone Dialer.
- **Email Widget**: When Selected, the user can email the clinic using existing email applications on the phone like E-mail, GM

**Implementation and Deployment:**

We have implemented Login screen, home screen and Navigation drawer below are the screenshots.



When the user select “Direction” tab on navigation drawer, google map is opened providing direction from user’s current location to a hard coded address (clinic address)
When user select “Call Clinic” or “Email Clinic” phone/email client is opened with hard coded (Clinic phone number/email address)



**Test Cases using Junit:**

```
@RunWith(AndroidJUnit4.class)
@LargeTest
public class LoginActivityTest {
    private String email;
    private String badEmail;
    private String password;
    private String badPassword;
    @Rule
    public ActivityTestRule<LoginActivity> rule =
            new ActivityTestRule<LoginActivity>(LoginActivity.class);
    @Before
    public void initValidString(){
        email = "admin@admin.com";
        badEmail = "cat";
        password = "admin";
        badPassword = "a";
    }
    @Test
    public void testSuccessLogin() throws Exception{
        Intents.init();
        onView(withId(R.id.email))
                .perform(typeText(email));
        onView(withId(R.id.password))
                .perform(typeText(password), closeSoftKeyboard());
        onView(withId(R.id.email_sign_in_button)).perform(click());
        rule.launchActivity(new Intent());
        intended(hasComponent(LoginActivity.class.getName()));
        intended(hasComponent(MainActivity.class.getName()), times(1));
        Intents.release();
    }
    @Test
    public void testFailedLoginBadPassword() throws Exception{
        Intents.init();
        onView(withId(R.id.email))
                .perform(typeText(email));
        onView(withId(R.id.password))
                .perform(typeText(badPassword), closeSoftKeyboard());
        onView(withId(R.id.email_sign_in_button)).perform(click());
        rule.launchActivity(new Intent());
        intended(hasComponent(LoginActivity.class.getName()));
        intended(hasComponent(MainActivity.class.getName()), times(0));
        Intents.release();
    }
    @Test
    public void testFailedLoginBadEmail() throws Exception{
        Intents.init();
        onView(withId(R.id.email))
                .perform(typeText(badEmail));
        onView(withId(R.id.password))
                .perform(typeText(password), closeSoftKeyboard());
        onView(withId(R.id.email_sign_in_button)).perform(click());
        rule.launchActivity(new Intent());
        intended(hasComponent(LoginActivity.class.getName()));
        intended(hasComponent(MainActivity.class.getName()), times(0));
        Intents.release();
    }
    @Test
    public void testFailedLoginBadEmailAndBadPassword() throws Exception{
        Intents.init();
```

```
        onView(withId(R.id.email))
                .perform(typeText(badEmail));
        onView(withId(R.id.password))
                .perform(typeText(badPassword), closeSoftKeyboard());
        onView(withId(R.id.email_sign_in_button)).perform(click());
        rule.launchActivity(new Intent());
        intended(hasComponent(LoginActivity.class.getName()));
        intended(hasComponent(MainActivity.class.getName()), times(0));
        Intents.release();
    }
}
```

**Implementation status report**

**Work completed:**
Login screen, home screen and Navigation drawer with direction service

**Responsibility (Task, Person)**
The timeline for this increment spanned for only two weeks, since our original plan was rejected based on similarity to another project. Every member of the group was involved in the brainstorming sessions that led to the format for the service, CliniConnect, we have mapped out. As we found the deadline fast approaching we divided up the tasks. On Implementation and testing, Sri and Luke put in several hours of coding. Luke also contributed the class UML design and sequence design. Shweta is on the task of the Report and connecting all of the pieces contributed by our team members. Ben is the team lead, and worked on team management, increment scheduling and Project Planning.

**Work completed and work to be completed:**
Everything has been updated in zenhub please visit our zenhub link below.

https://github.com/ljm7b2/ASE_PROJECT_G1#boards?repos=50469210

**Bibliography**


- Wireframes design http://creately.com/
- Testing Information http://developer.android.com/tools/testing-support-library/index.html
- Android Studio http://developer.android.com/sdk/index.html
- Google Maps API https://developers.google.com/maps/


**Github: https://github.com/ljm7b2/ASE_PROJECT_G1**