CliniConnect

Group 1

Benjamin Chrysler (ID:08)

Shweta Parihar (ID:42)

Luke, McDuff(ID:27)

Sri Patluri (ID:44)

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 reminder 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.

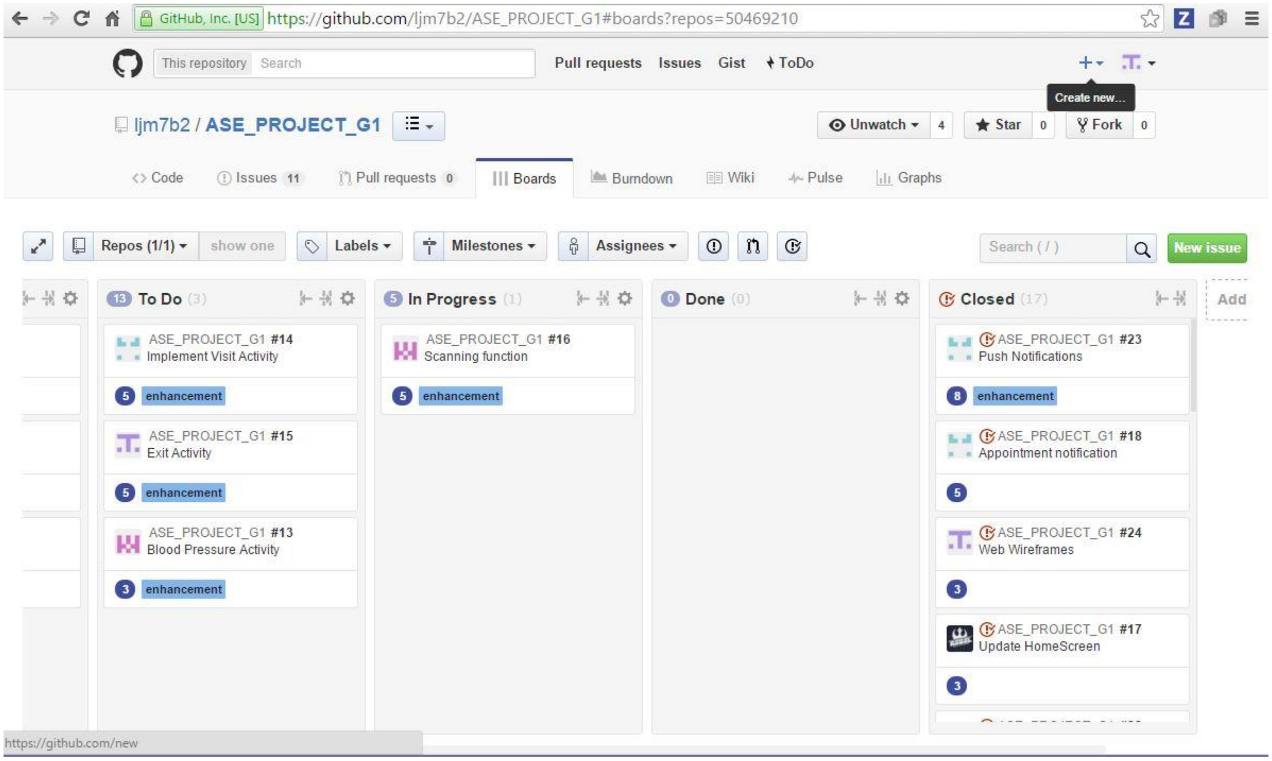
Features:

- New patient can be registered in the registration page of the web 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


GitHub, Inc. [US] https://github.com/ljm7b2/ASE_PROJECT_G1#boards?repos=50469210
This repository Search
Pull requests Issues Gist ToDo
Create new...
ljm7b2 / ASE_PROJECT_G1
Unwatch 4 Star 0 Fork 0
Code Issues 11 Pull requests 0 Boards Burndown Wiki Pulse Graphs
Repos (1/1) show one Labels Milestones Assignees
Search (/) New issue
13 To Do (3)
ASE_PROJECT_G1 #14 Implement Visit Activity
5 enhancement
ASE_PROJECT_G1 #15 Exit Activity
5 enhancement
ASE_PROJECT_G1 #13 Blood Pressure Activity
3 enhancement
5 In Progress (1)
ASE_PROJECT_G1 #16 Scanning function
5 enhancement
0 Done (0)
Closed (17)
ASE_PROJECT_G1 #23 Push Notifications
8 enhancement
ASE_PROJECT_G1 #18 Appointment notification
5
ASE_PROJECT_G1 #24 Web Wireframes
3
ASE_PROJECT_G1 #17 Update HomeScreen
3
Add
https://github.com/new

Schedule for the four different increments.

Stories (Issues): This second increment will improve our android application and will lay the foundation of the administrator side of our service. We have introduced push notifications in which both administrators and users participate. Users will receive a notification through the app on their device whenever the administrator sends a notification through the admin web site. Also, we have implemented a base page layout for all web pages. Additionally, the user application can receive QR codes that contain information about the needed lab work. Also, the database has been created to store user name, password and identification certification.

Service Design: The service design at this stage, starts with the creation of a base layout for the admin site. Admin will be able to create new users and send notifications directly to the android application. This code displays the information users need for their lab tests, instead of having to carry along extra paperwork from place to place. Additionally, we included push notifications that send appointment reminders from the administrators. These notifications may also include short messages from the administrators.

Service Implementation: The service now includes the implementation of the administration side which is comprised of html, css and javascript. The QR code creation and display on the user app side, the push notification sending and receiving, as well as the home page and shell of the user site have all been implemented. Both the administration page and user application sides have been implemented. Our database information has been implemented on mlab.

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 present their lab code they can access it through the web application through the navigation bar. When the User has an upcoming appointment, the Administration can send an appointment push notification to the User's app. When the Admin wants to Navigate to any other page the User can use the navigation bar to redirect to another page.

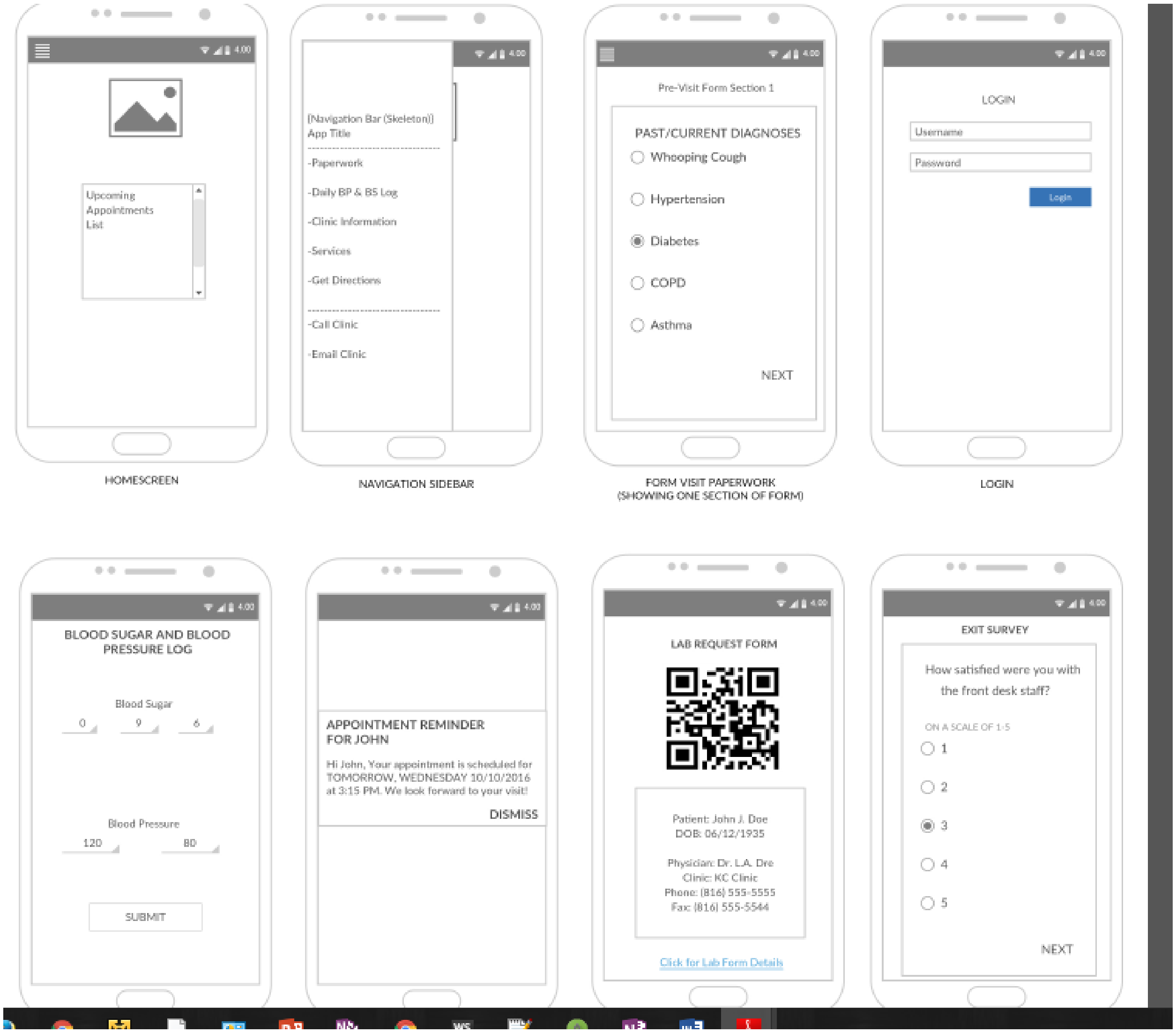
Use Case

Service description: CliniConnect is an service 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. The service includes an administration portal that serves as to connect the administration to the user. This service allows administrators to keep their patients properly up to date.

Second Increment Report

In this second increment of "**CliniConnect**" we have designed the overall structure and flow of the application and administration using wireframes and UML diagrams. In this phase we have implemented the shell of the administration portal. Included in this increment, are the ability to update the lab forms from the administration portal to the user app. Also, we have created the ability to have administrators send personalized notifications to the user's device. When clicking on the notification the User is sent back to the application.

Detail Design: Wireframes



Upcoming Appointments List

HOMESCREEN

(Navigation Bar (Skeleton))
App Title

-Paperwork

-Daily BP & BS Log

-Clinic Information

-Services

-Get Directions

-Call Clinic

-Email Clinic

NAVIGATION SIDEBAR

Pre-Visit Form Section 1

PAST/CURRENT DIAGNOSES

○ Whooping Cough

○ Hypertension

● Diabetes

○ COPD

○ Asthma

NEXT

FORM VISIT PAPERWORK
(SHOWING ONE SECTION OF FORM)

LOGIN

Username

Password

Login

LOGIN

BLOOD SUGAR AND BLOOD PRESSURE LOG

Blood Sugar

0 9 6

Blood Pressure

120 80

SUBMIT

APPOINTMENT REMINDER FOR JOHN

Hi John, Your appointment is scheduled for TOMORROW, WEDNESDAY 10/10/2016 at 3:15 PM. We look forward to your visit!

DISMISS

LAB REQUEST FORM

Patient: John J. Doe
DOB: 06/12/1935

Physician: Dr. L.A. Dre
Clinic: KC Clinic
Phone: (816) 555-5555
Fax: (816) 555-5544

Click for Lab Form Details

EXIT SURVEY

How satisfied were you with the front desk staff?

ON A SCALE OF 1-5

○ 1

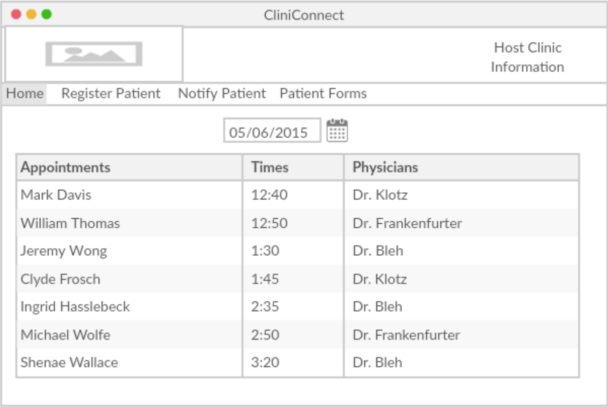
○ 2

● 3

○ 4

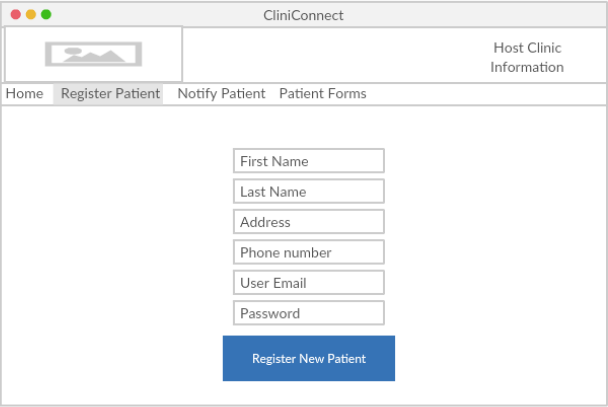
○ 5

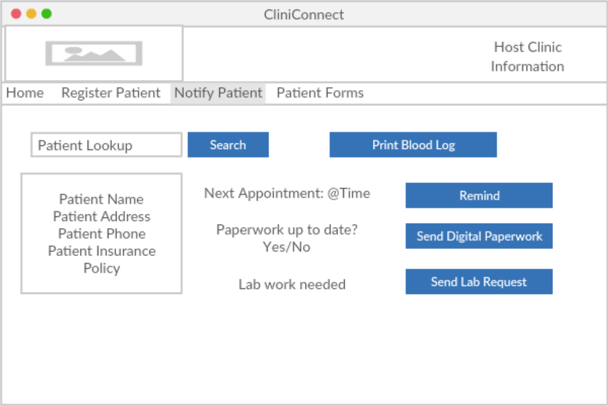
NEXT



05/06/2015

Appointments	Times	Physicians
Mark Davis	12:40	Dr. Klotz
William Thomas	12:50	Dr. Frankenfurter
Jeremy Wong	1:30	Dr. Bleh
Clyde Frosch	1:45	Dr. Klotz
Ingrid Hasslebeck	2:35	Dr. Bleh
Michael Wolfe	2:50	Dr. Frankenfurter
Shenae Wallace	3:20	Dr. Bleh



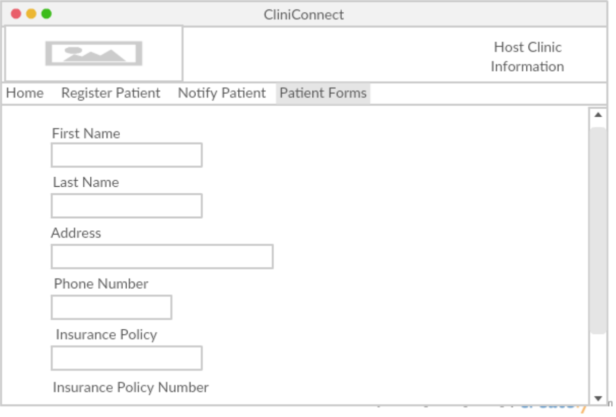


Patient Name
Patient Address
Patient Phone
Patient Insurance Policy

Next Appointment: @Time

Paperwork up to date? Yes/No

Lab work needed



First Name

Last Name

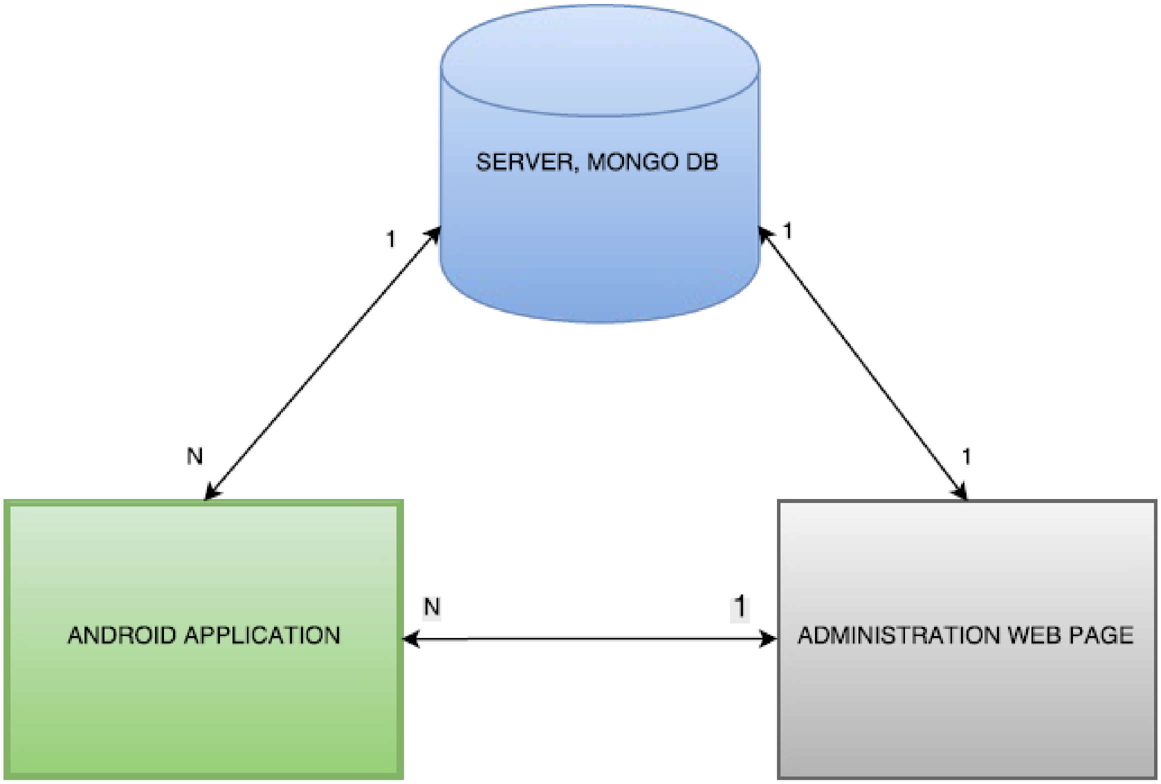
Address

Phone Number

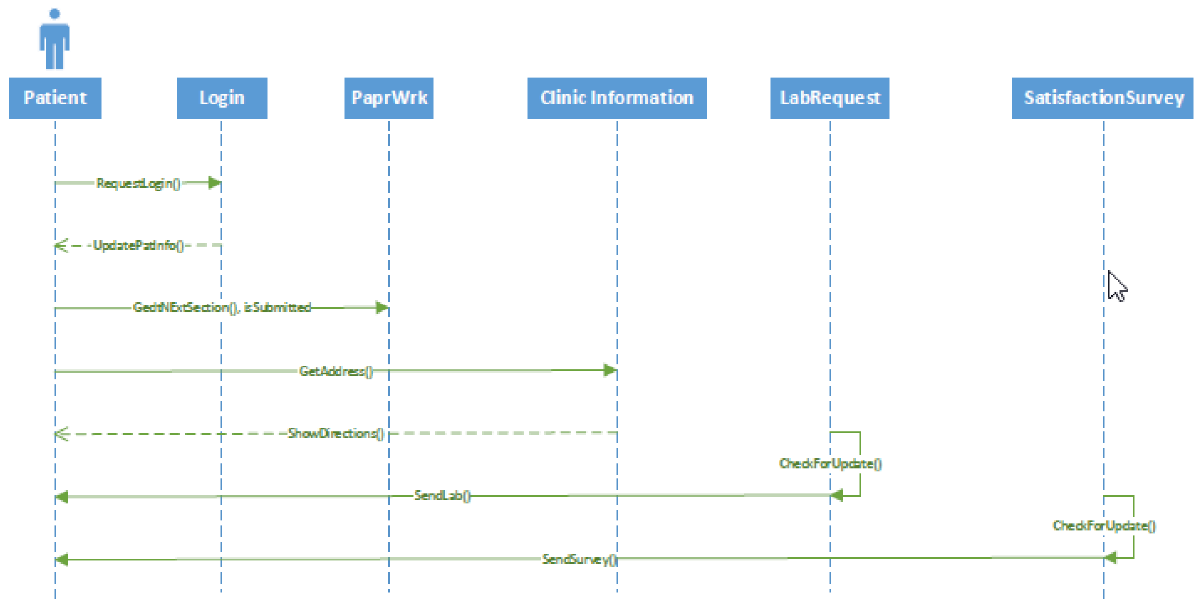
Insurance Policy

Insurance Policy Number

ARCHITECTURE STACK



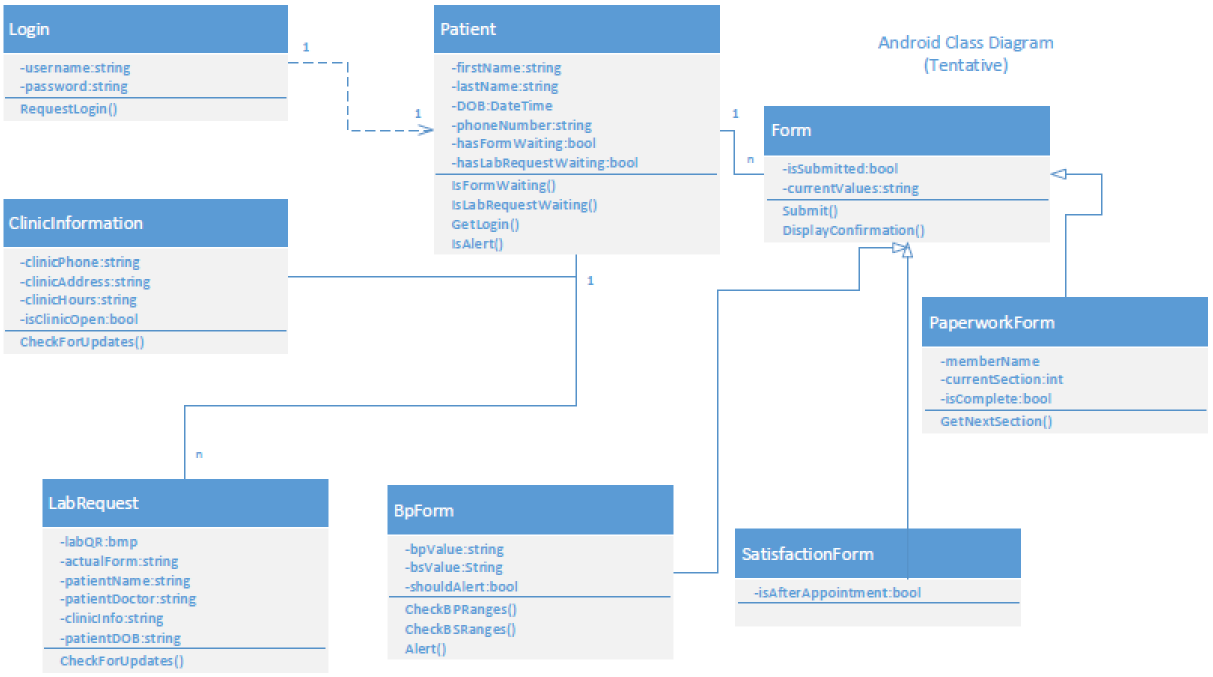
Sequence diagram



ANDROID SEQUENCE DIAGRAM (TENTATIVE)

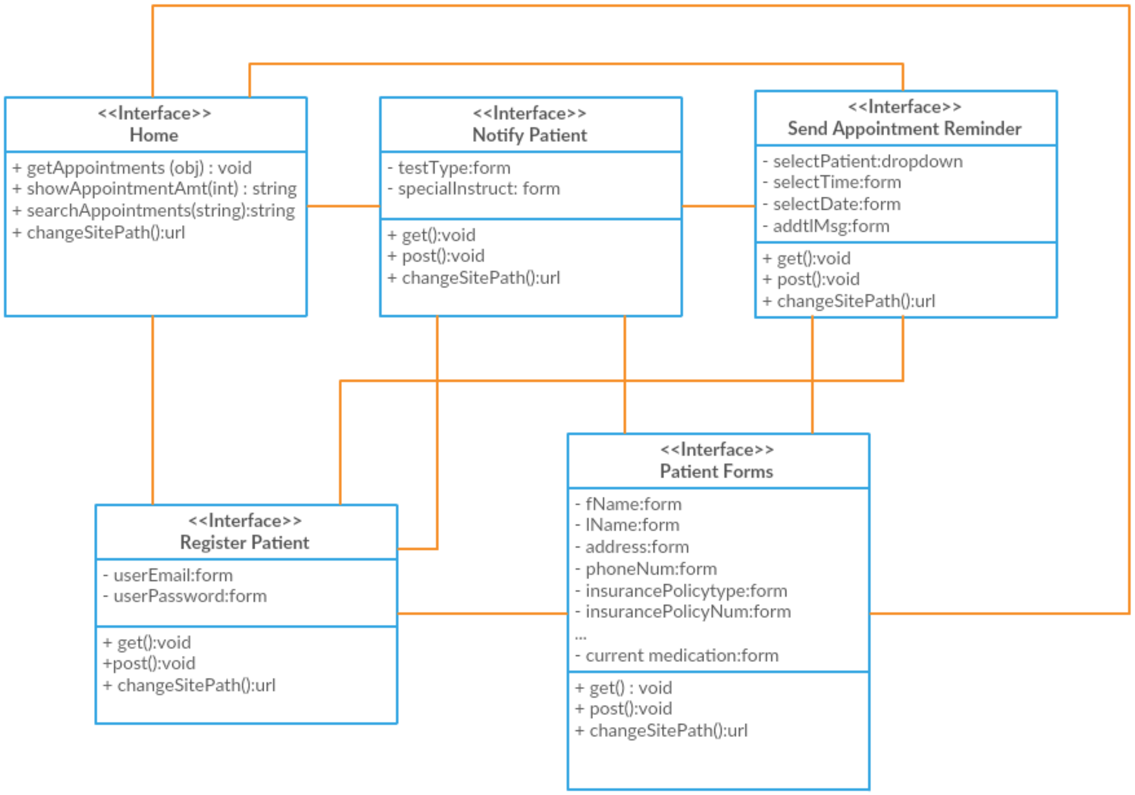
Class diagram

Android



Android Class Diagram
(Tentative)

Web



<u>Used existing Services/API</u>

APIs:

- **Google Maps:** Used to get directions from the user's current location to the clinic in Google Maps.
- **ZXing (Zebra Crossing):** Used to convert lab request information into QRcode.

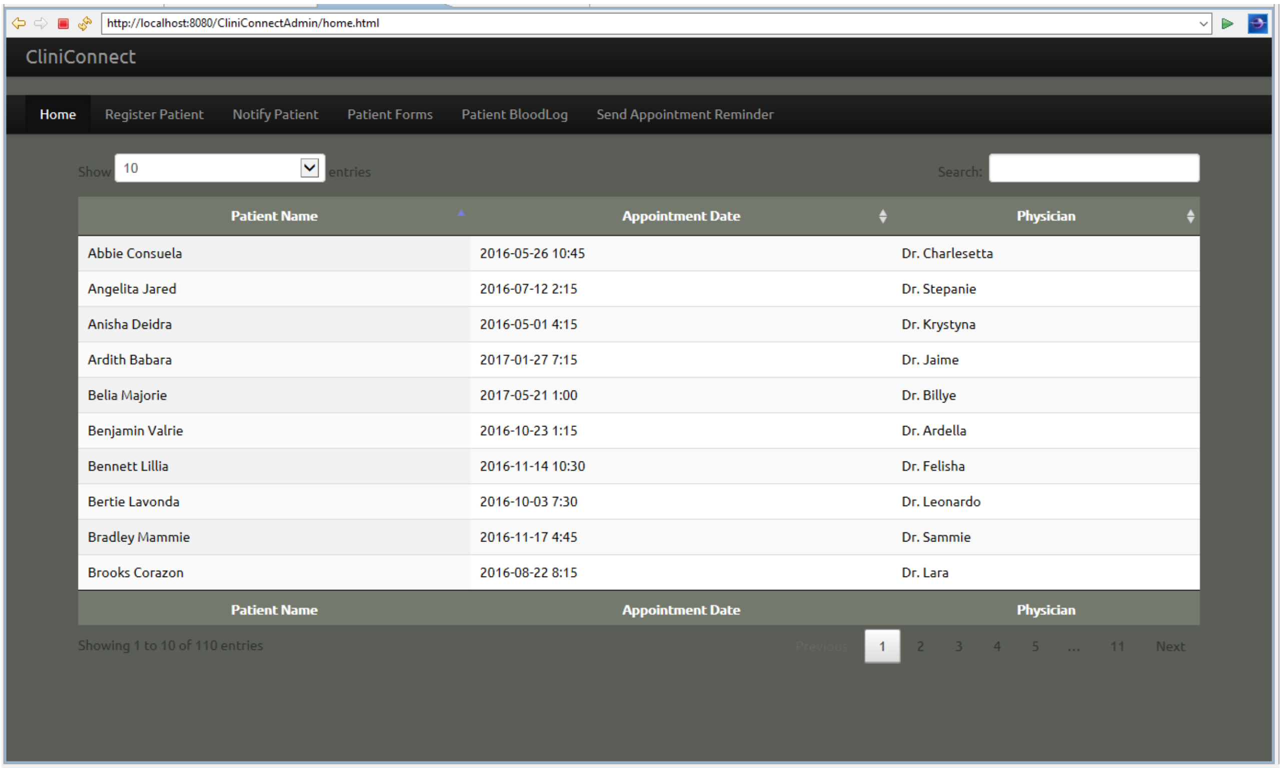
Widgets:

- **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
- **ImageView Widget:** Used to display the QR code after it had been processed by ZXing.
- **TextView Widget:** Used to display information about the information embedded in the QR code.

Implementation and Deployment:

We have implemented Home screen, Navigation bar, Notification sending and receiving, QRcode sending and receiving, Labs form pages and also created a generic logo for the application. Below are the screenshots.

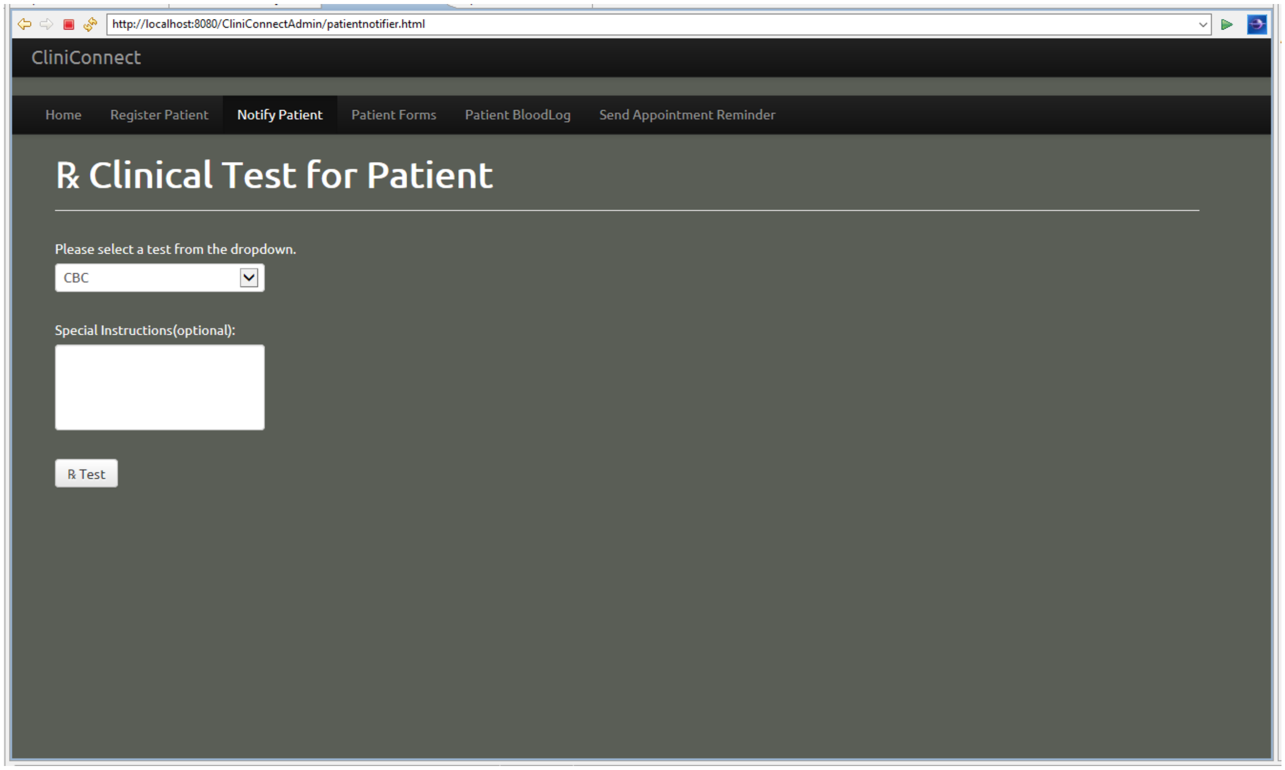
Admin Web Page Home:



Patient Name	Appointment Date	Physician
Abbie Consuela	2016-05-26 10:45	Dr. Charlesetta
Angelita Jared	2016-07-12 2:15	Dr. Stepanie
Anisha Deidra	2016-05-01 4:15	Dr. Krystyna
Ardith Babara	2017-01-27 7:15	Dr. Jaime
Belia Majorie	2017-05-21 1:00	Dr. Billye
Benjamin Valrie	2016-10-23 1:15	Dr. Ardella
Bennett Lillia	2016-11-14 10:30	Dr. Felisha
Bertie Lavonda	2016-10-03 7:30	Dr. Leonardo
Bradley Mammie	2016-11-17 4:45	Dr. Sammie
Brooks Corazon	2016-08-22 8:15	Dr. Lara
Patient Name	Appointment Date	Physician

Showing 1 to 10 of 110 entries

Admin Web Page QR requester



http://localhost:8080/CliniConnectAdmin/patientnotifier.html
CliniConnect
Home
Register Patient
Notify Patient
Patient Forms
Patient BloodLog
Send Appointment Reminder

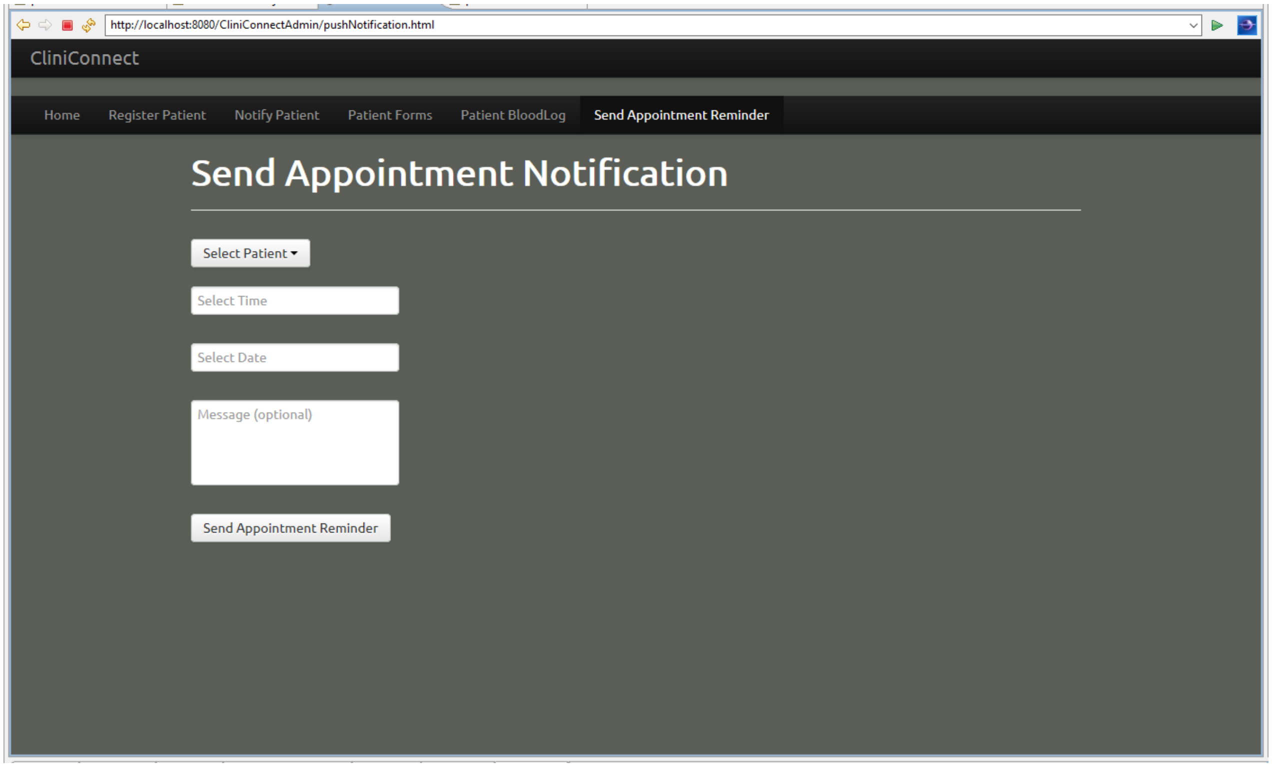
℞ Clinical Test for Patient

Please select a test from the dropdown.

CBC

Special Instructions(optional):

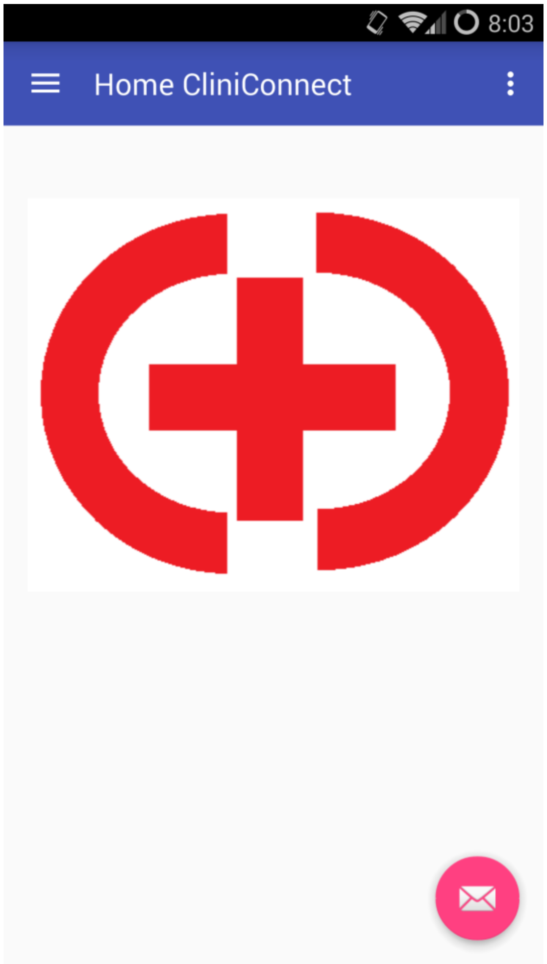
Admin Web Page Appointment Sending



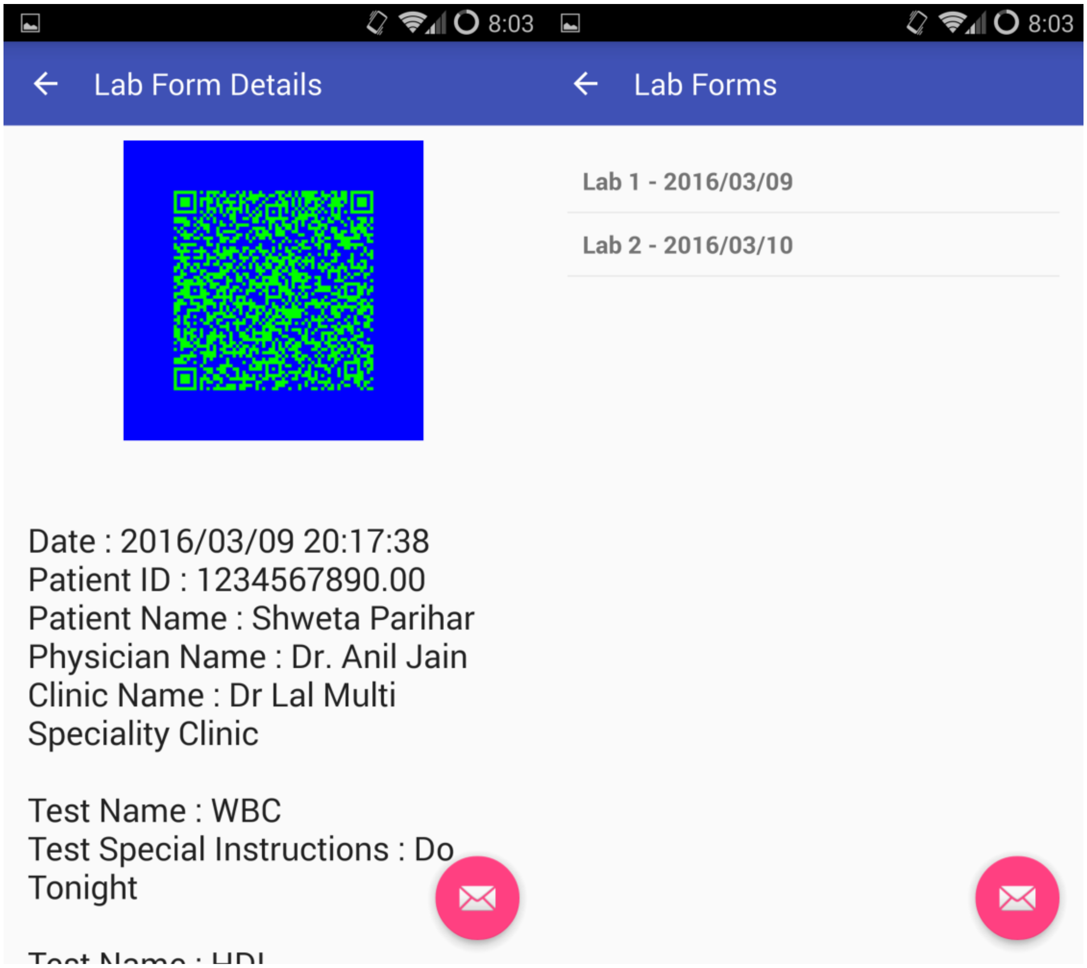
http://localhost:8080/CliniConnectAdmin/pushNotification.html

Send Appointment Notification

Mobile Home page with logo



Mobile Lab Pages (1st image with QR code)



Lab Form Details

Date : 2016/03/09 20:17:38
Patient ID : 1234567890.00
Patient Name : Shweta Parihar
Physician Name : Dr. Anil Jain
Clinic Name : Dr Lal Multi Speciality Clinic

Test Name : WBC
Test Special Instructions : Do Tonight

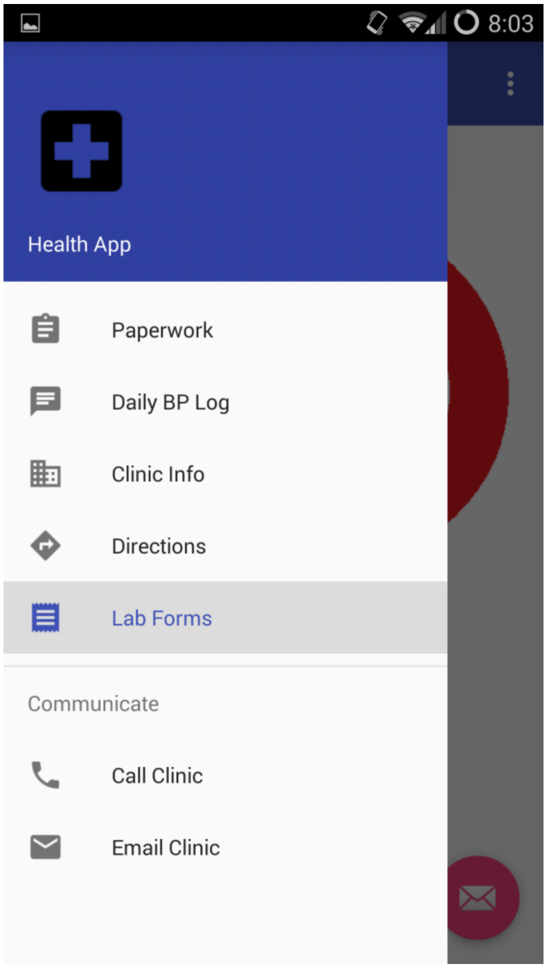
Test Name : HDL

Lab Forms

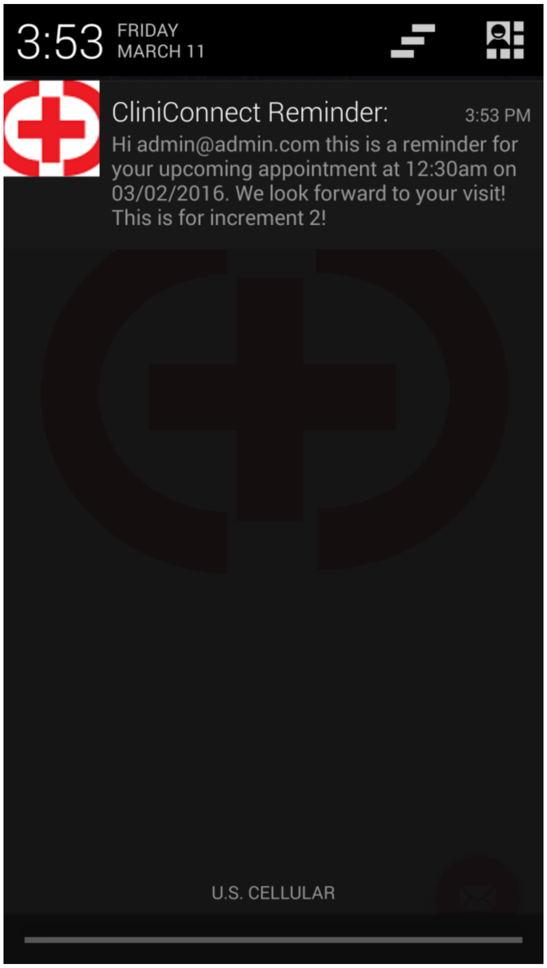
Lab 1 - 2016/03/09

Lab 2 - 2016/03/10

Mobile Application Navigation Bar Updated with Lab Forms



Mobile Application Push Notification

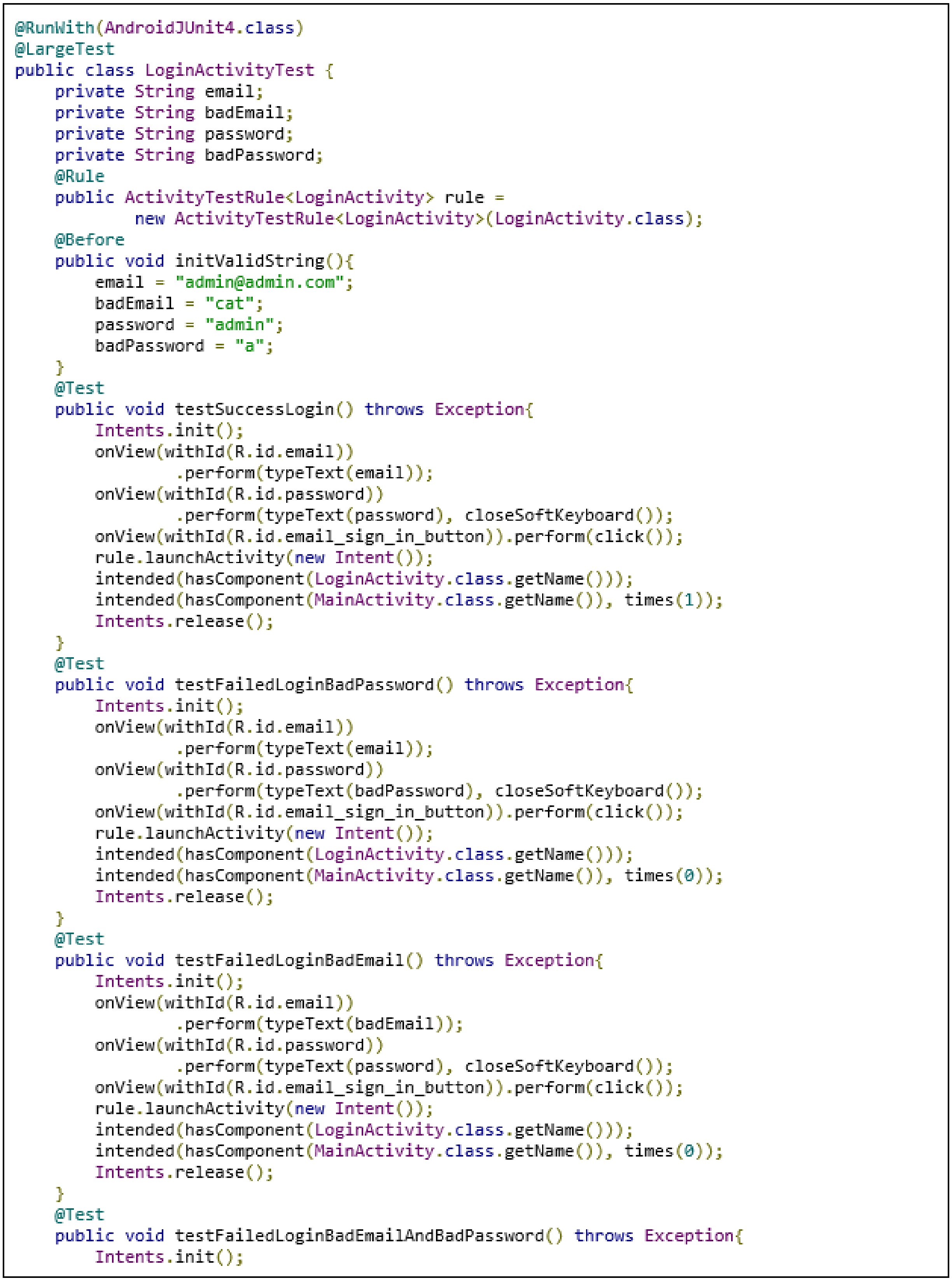


CliniConnect Reminder: 3:53 PM

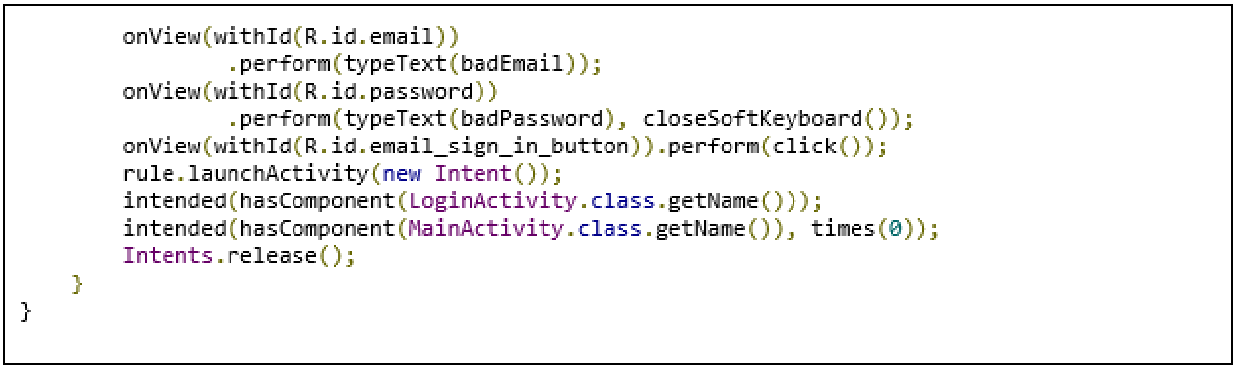
Hi admin@admin.com this is a reminder for your upcoming appointment at 12:30am on 03/02/2016. We look forward to your visit! This is for increment 2!

Unit Test Cases:

```
@RunWith(AndroidJUnit4.class)
@LargeTest
public class QRLabTest {



    @Rule
    public ActivityTestRule<ClinicalTest> rule =
            new ActivityTestRule<>(ClinicalTest.class);


    @Test
    public void DateOpensQRTest() throws Exception{
        Intents.init();
        SystemClock.sleep(5000);
        onView(withId(R.id.clinical_test_date_list)).perform(click());

        rule.launchActivity(new Intent());
        SystemClock.sleep(5000);
        intended(hasComponent(ClinicalTest.class.getName()));
        intended(hasComponent(QRCodeAndDetailsActivity.class.getName()), times(1));
        Intents.release();
    }
}
```



```
@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:

The work completed in this increment was, web page design, QR code (both user and administration sides), Push notifications (both user and administration sides), home page for web, and database setup were completed.

Responsibility (Task, Person)

In this increment, work was divided into tasks to use time efficiently. The task of the QR codes went to Shweta, implementing the user end result as well as the administrator's back end request. Sri handled the Implementation of the home web pages and the general shared layout for each web. Ben, designed the web pages, created the logo for the application and wrote the report. Luke implemented the database setup, as well as implemented the push notifications on both administrator and user ends.

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