Test plans (Final sprint)

(These are the tests for the final report only, the testing report for the other sprints can be found on the website or the wiki page).

Common Test Case:

At no time should the web site display a Django error frame or blank screen.

Automation tests are found in sample/tests.py file. These are tests the server's response to POSTs, regarding changes to doctors, field workers, patients and cases.

To run tests, type "python manage.py test sample"

User Story #8: doctor can view patient scans

Task 1: Modify case page (and maybe patient page as well) to be able to view uploaded images.

The first 4 steps are for creating a case if there isn't one.

1) Login as a worker (User: worker1, Password: worker)
2) Click "Add new patient" to create a new patient
3) Click "Add new case" to create a new case.
4) Click on "Browse" to browse an image from your computer, click submit to upload the image. Finish creating the case.
5) Login as a doctor (Username: doctor1, Password: doctor).
6) Click on a case, and then click on a link inside the image sub-field.
7) Opening this link should display a patient image.

User Story #11: doctor can add comments on patient's case

Task 1: Design patient case page

Automated tests for this task are found in tests.py

Task 2: Complete the frontend and backend of "Add new case" page, so that it updates the patient page, and the field worker page.

Follow the same 5 steps from US #8. After logging in as a doctor, the case should show up in the cases table. Similarly, the worker page should also be updated.

Task 3: Make a "Add comments section", which updates the comment section, and sorts by latest comment.

Follow the same 5 steps from US #8.

6) Click on a case, and then click on "Reply"
7) Enter some comment, and upload an image from your computer, then click submit. The submitted comment as well as a preview of the image should be at the top (if there are

multiple comments on the case).

8) Logout from doctor, and login as worker. Add a comment and upload an image, and once again, the submitted comment and the image should be at the top.

Task 4: Doctor’s can lock or unlock a case

Follow the same 5 steps from US #8 to create a case.

6) Click on the case, and select “Locked (me)” from the Lock/Unlock this case drop down

7) Click on “Dashboard” in the top left, and you will see the name appear under the “Reviewer” column of the case table.

8) Unlock the case, and go to the dashboard, and you will see the name disappear under “Reviewer”

Task 5: Doctor’s can Open/Close cases

Follow the same 6 steps from the previous task to create a case and Lock the case.

7) When you lock the case, the string Case status: Open, should change to a drop down.

8) Click on “Closed” to close the case.

9) Click on “Dashboard” in the top left, and you will see the case is Closed in the Status column

User Story #12: doctor can add sub commenting a patient’s case

Task 1: Add a sub comment button, that will add “update”, or link, a comment made by the doctor, but not edit it.

Follow steps 1-8 from US #11 Task 4 to login as a doctor and add a comment.

1) Next to every comment, there should be a “Reply” button, click one of them to add a sub-comment. Also, upload a picture from the computer.

2) The sub-comment should show up indented at the bottom of the comment, as well as a preview of the picture.

User Story #14: doctors edit the priority of the patient.

Task 1: Include the ability for doctors to change priority, and update the respective pages.

Automated tests for this task are found in tests.py

User Story #15: doctors and field workers have search bar in home page

Task 1: Make the search box on the doctor’s/field workers page

Nothing to test here.

Task 2: Search the database for the searched term.

1) Login as a doctor

2) Search for a name, which will take filter the table. This table should display all the names from the search term. You should be able to click on a name to be redirected to that person’s page.

3) Search something that you know for sure isn’t on any of the pages. The case table should be empty.

Task 3: Filter the table

Same as previous task

Task 4: Ability to sort the results.

Same as previous task.

Click on any of the top bars to sort it by ascending/descending order. Click on it again to toggle sort.

User Story #20: upload images from the app (field worker)

Task 1: Send data from the “Add new case” page to the server

1) Send a picture via the Android phone app.
2) Check if the the picture is uploaded to the server or not.

User Story #22: assigning the priority to the patients(field worker)

Task 1: Ability to modify priority. Update respective pages/tables.

Automated tests for this task are found in tests.py

User Story #23: android app for field workers

Task 1: Connect the Django server.

1) Login the worker account on the app
2) Check if there is a correct message responded form the server.

Task 2: Create an app interface for field works to add new patient.

1) Run the Android app emulator, wait for about 10mins to load the app.
2) Check if the interface layout is properly or not.

Task 3:Build an app interface for patient’s page.

1) Run the Android app emulator.
2) Search a patient, then check if the correct list of patient loaded from the server.
3) Click on one patient from the patient table
4) Check if the patient information loaded correctly with the server data.

Task 4: Build an app interface for case page

1) Run the Android app emulator.
2) Search a patient (existing one), then 3) click on the patient from the patient table
3) Check if the correct list of cases loaded from the server.
4) Add a new case
5) Check if the case shows up on the patient’s page.

User Story #24: privilege to add comments to each patient(field worker).

Task 1: Update the “Add comments section” so there is a clear (color) distinction between comments added by field workers and doctors.

Follow the steps from US #11 Task 4 to add a case, and add a comment.

Adding a comment as a doctor should make the font blue, while adding a comment as a worker should make the font green.