Specification for next sprint (Sprint 2)

We chose to allocate the following user stories to the final sprint, because these user stories have a lower priority than the ones in sprint 2. We pushed some user stories from sprint 2 to final sprint because we didn't manage to complete them. The user stories that were planned for the final sprint depend on the user stories from Sprint 2.

We can define success as how well the user stories are implemented. Not only must it be possible to do what the user stories say can be done, but should contain as few bugs as possible and be written from code that is as bug free and well-written as possible. We will know we have succeeded at --- when all test cases related to the user stories pass.

We realize that we have a lot of user stories to do for this sprint, which is why we have made some user stories optional, since we will not have time to implement them, and also these user stories are not needed and they were only meant to be implemented if we have enough time. Even after this, there are still a lot of user stories that are needed. Although it seems like a lot of work, we have already completed half or most of the work for the user stories pushed back, so we should not have trouble completing them. A strategy we will follow for this sprint is that we will focus on completing user stories one by one, rather than trying to do them all at once, and only managing to get them halfway done.
If user stories take less time than expected, we will work on some of the optional user stories, and if user stories take more time than expected, we will only focus on getting the basic functionality to work first.

User Story #8, #11, #12, #13, #15, #20 are basic functionalities for doctor to be able to manage patients' cases, and the implementations of User Story #9, #10, #17 are relying on them.
User Story #20, #23, #24 are the basic functionalities for field worker to be able to manage new/existing patients' cases.

User Story #8: doctor can view patient scans
Justification: This is important part of our project, and this is medium priority. This is simple to implement and it will take time as expected. We haven't encountered any abnormalities with django. We have pushed this user story back since we first need to figure out how to handle image uploads, and then we can focus on viewing the images. We need to finish part of User Story #20 to provide framework for #8.

User Story #11: doctor can add comments on patient's case
Justification: This is high priority so this need to be done very fast. This will take time as expected and it shouldn't cause any abnormalities. We have pushed this user story back since we only managed to finish about half of the work.

User Story #12: doctor can add sub commenting a patient's case
Justification:This is a high priority for the project so that is why it is needed to be done in this

sprint. We need to consider about other doctors being able to edit the comment as well. Therefore it might take more time than usual. order for each comment added by users, and sort them by adding time, it would take more time than we expect since it requires more than just adding data. We have pushed this user story back since we only managed to finish about half of the work, and this user story depends on #11.

User Story #13: a link to the doctor's profile that contains a history of their comments.
Justification: This is fundamental requirement and feature we should have on the webpage, and it would take the time as we expect since there is no much difficulty on implementing it. We have pushed this user story back since this user story depends on implementing #11 and #12.

User Story #15: doctors/workers have search bar in home page
Justification: This is similar to user story #3. This is medium priority but it is very important. We have pushed this user story back since we only managed to finish about half of the work.

User Story #20: upload images(field worker)
Justification: This is important feature and high priority in our whole project. We have difficulty on uploading an image and linking it to the database table, and it should take more time than we expect. We have pushed this user story back since we haven't completely figured out the android-app user story #23, and how we would send SMS/MMS to the server.

User Story #23: android app for field workers
Justification: this is a high priority for this project. and this needed to be done so that we can test out the android quickly. it probably might take a lot of time since we don't have experience implementing android apps. However, we might also think to make an app that has the same interface for the website. we already have the interface for the android but we haven't fully tested it out yet. We have pushed this user story back since we didn't finish all the tasks for this user story.

User Story #24: privilege to add comments to each patient(field worker).
Justification: Similar to #11, this is high priority so this need to be done very fast once we finish #11. This will take time as expected and it shouldn't cause any abnormalities. We have pushed this user story back since it is very similar to #11 and #12, and depends on them.

User Story #10: doctors add the annotations to patient images and scans.
Justification: This is a medium-high priority, but since it relies on #8 and #20, we decided to do it for final sprint. This user story will be one of the most difficult ones, and we will have to spend a lot of time on this.

User Story #17: search for cases involving the specified organ system.
Justification: This is low priority, and it relies on #15. In fact, we might have to do very little work for this based on the way we designed #15.

Optional user stories

User Story #9: doctors zoom function is required on the images.
Justification: This is low priority, since this feature will very rarely be used or needed, and it relies on #8 and #20. This should not take that much time to complete.

User Story #16: doctors can link two patients.
Justification: This user story is optional because 99% of the time there will not be a mix up of patients, this is why field workers have a search box, so they can search for a patient before creating a new one. Also, since the chances of a mix up happening are small, a worker can contact an admin, who has the functionality to fix up this problem easily.

User Story #18: Help Button for doctors
Justification: This user story is optional because the user interface is very easy and simple to learn, everything is straight forward. Therefore, doctors don't need a help button.

User Story #25: provide contact information on home page
Justification: This user story is optional because this is not a functionality of the project. It is something that would be nice to have if people are visiting the website to find out more information.

User Story #26: provide a map showing the regions that covered for the project.
Justification: This user story is optional because it is also not a functionality of the project. It is something that would be nice to have to "show off" where the app is being used.