uOttawa

Walk-In Clinic App Final Report

InterCrew

Gianluca Coletti, Ahmed Sorour, Yuta Lee, Saffat Aziz,
Phillip Brule
SEG2105
Dr. Andrew Forward

Introduction 3
Assumptions 3
UML Diagrams 4
Deliverable 1 Diagram 4
Deliverable 2 Diagram 4
Deliverable 3 Diagram 5
Final UML Class Diagram 5
Member Roles 6
Screenshots 7
Lessons Learned 14

Introduction

The InterCrew WalkIn Clinic app is an app that allows for patients to search for clinics, book an appointment, and rate the clinics they go to. This report outlines the process in which we designed and created this app. This includes UML diagrams, our team member roles, screenshots, and the lessons we learned through the process.

The purpose of this project was to create a walk-in clinic app with the functionality to create users, clinics, and services. This project developed many different software engineering skills through the creation of the app. These skills included design, backend development, frontend development, and the ability to work effectively in a group.

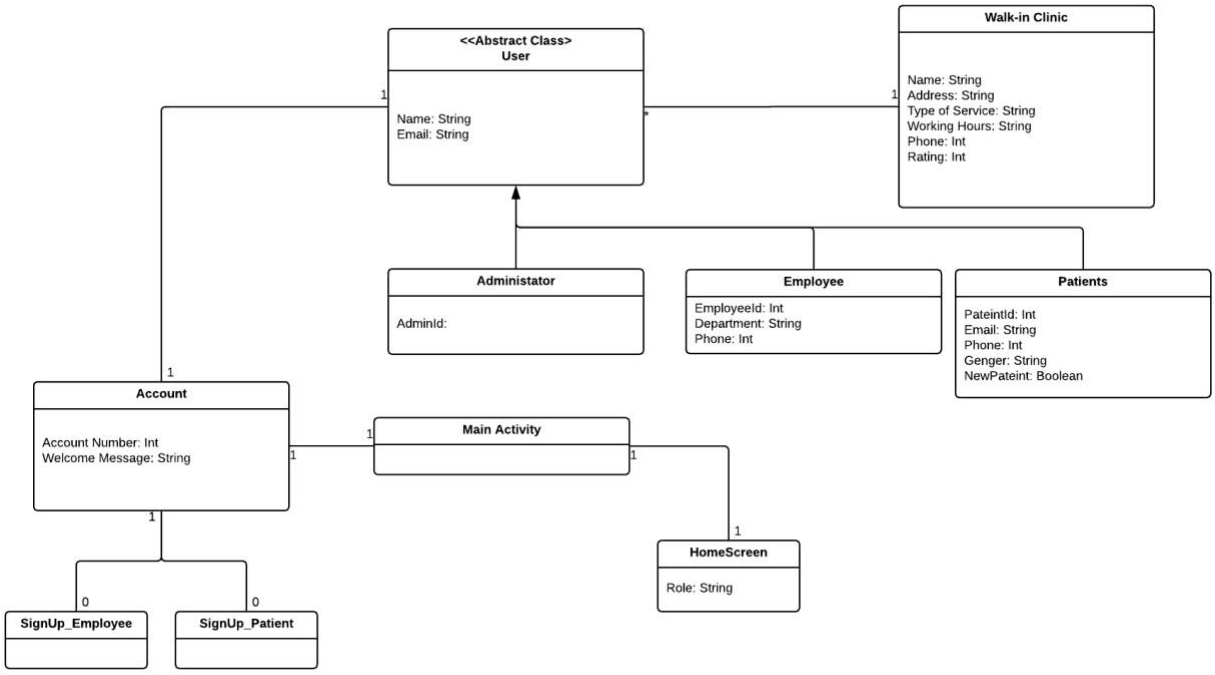
Assumptions

Below is a list of some simple assumption we made that our group thought made sense in the context of building an app for walk-in clinics.

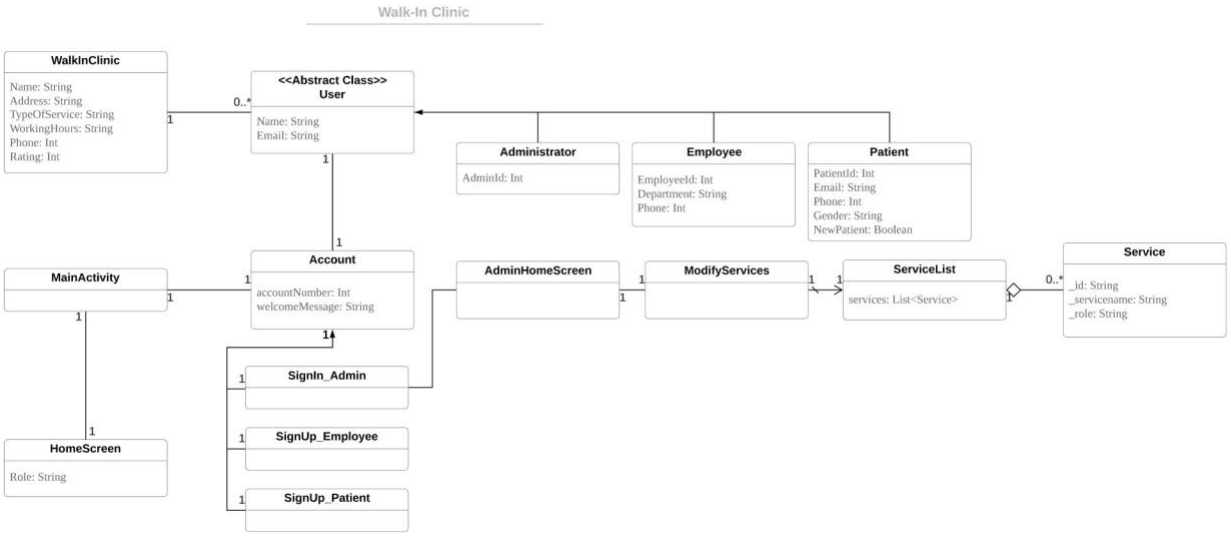
- A patient is only able to make one appointment per clinic
- A patient is only able to book an appointment for the current day
- An employee can be signed in without having a clinic
- An employee can only be associated with one clinic

UML Diagrams

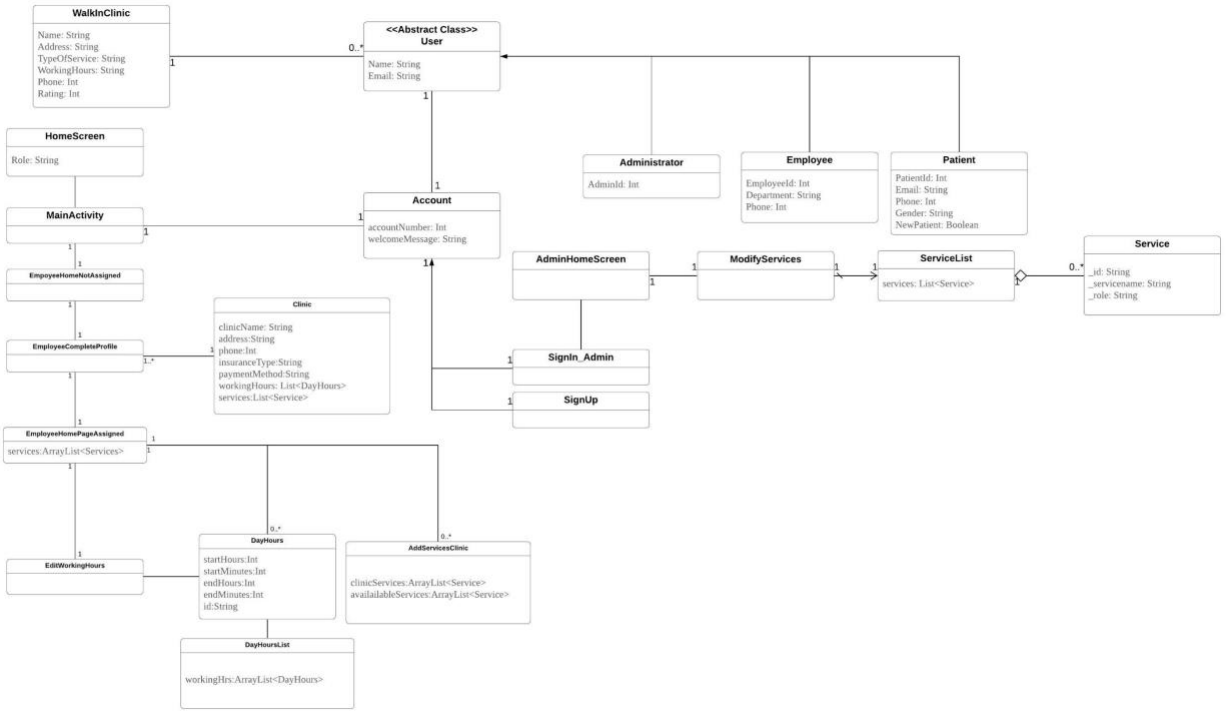
Deliverable 1 Diagram



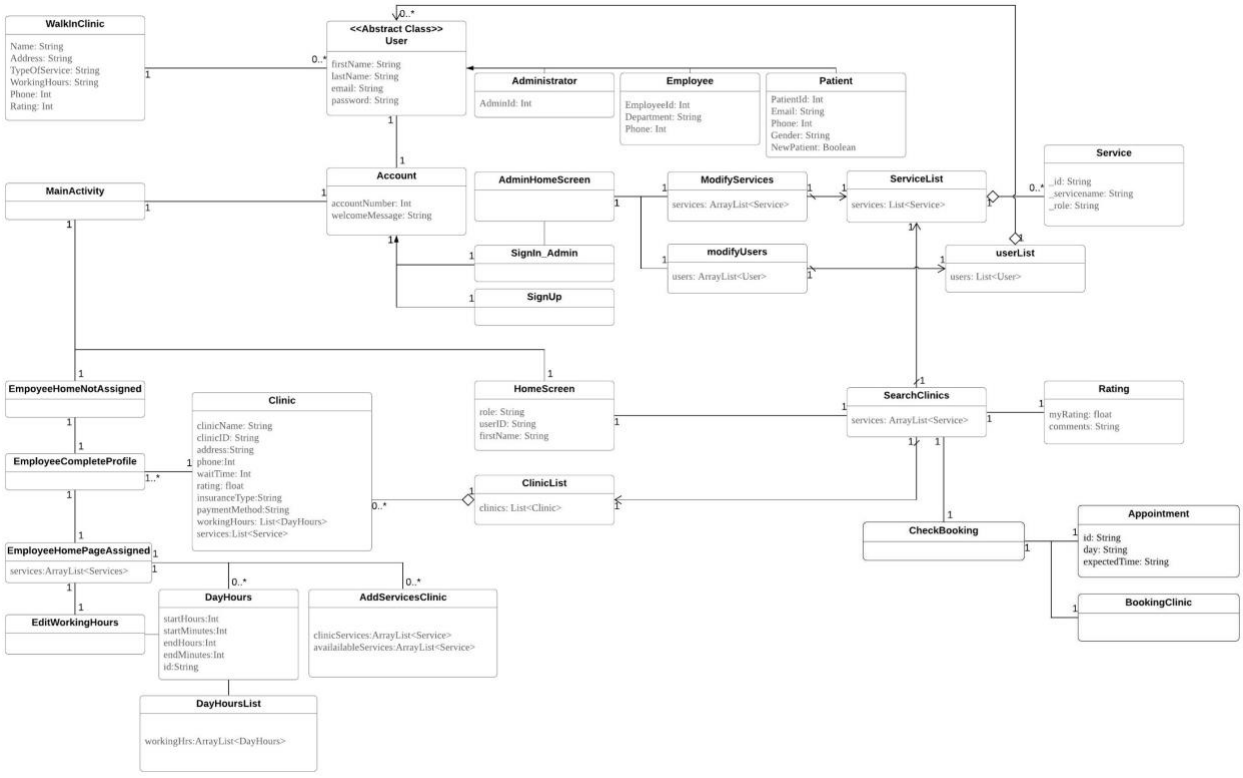
Deliverable 2 Diagram



Deliverable 3 Diagram



Final UML Class Diagram



Member Roles

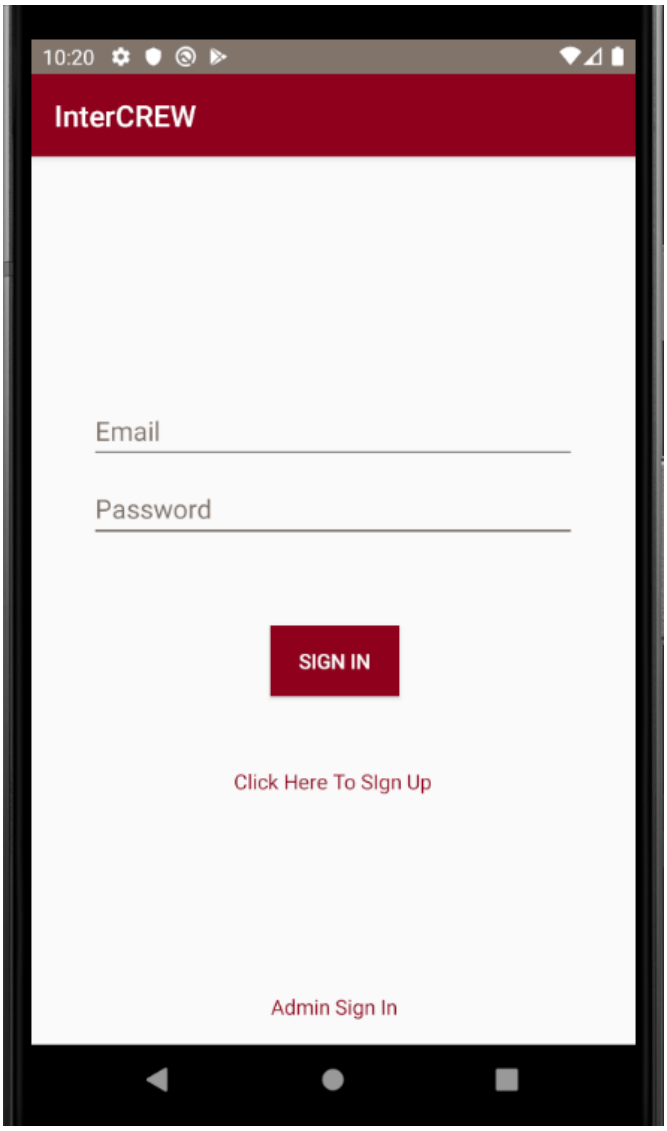
In order to complete the required tasks for each deliverable our project, we had a meeting within a week of competing the previous deliverable. This meeting was either in person or a group call. Over the course of our project Sorour became the group leader usually dividing up the work needed to be done on each deliverable. The group would then come to consensus on what tasks each member would take.

	Sorour	**Saffat**	**Gianluca**	**Yuta**	**Phillip**
Dev. 1	-Sign UP Employee and Patient -Login detection for user type	-UML -Login Screen -Field Validations	-Home Page -App design	-Classes	-Home Page
Dev. 2	-Database -Admin add and Delete available services -Fix structure	-Unit Testing -Debugging -Field Validation	-CircleCI integration -Unit Test cases	-UML -Clinic class -Modify Services	-Delete Service
Dev. 3	-Database -Adding and	-Clinic Class -Complete	-Testing -Clinic	-Home Screen	-Testing

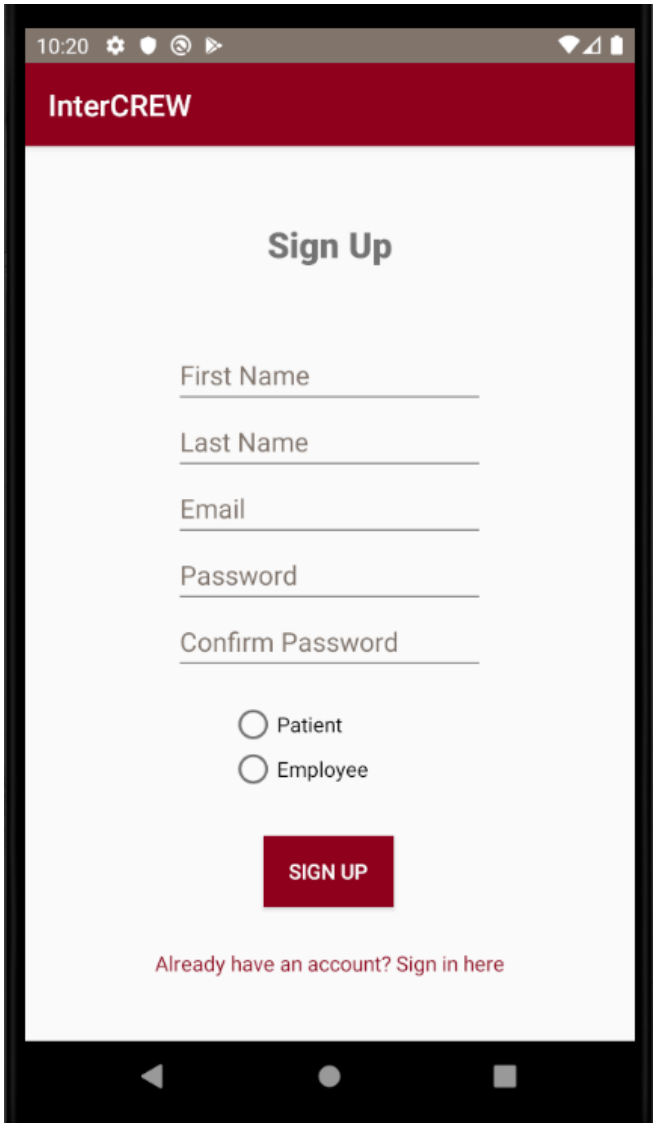
	Deleting Services to clinics -Bug fixes	Profile Page -UML -Field Validations	Working Hours -Complete Profile	Employee	
Dev. 4	-Database Back end work -Searching for clinics -Booking Appointments -Designed search list	-Rating Page -UI design -Debugging -Field Validation	-Booking -CircleCI -App Design	-Booking -UML	-Report- -Testing

Screenshots

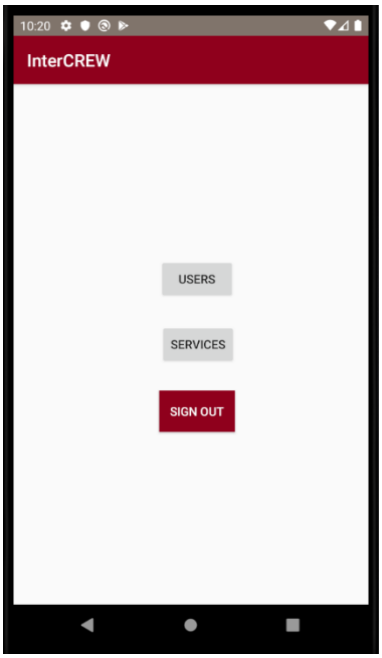
Home/Log-In Screen



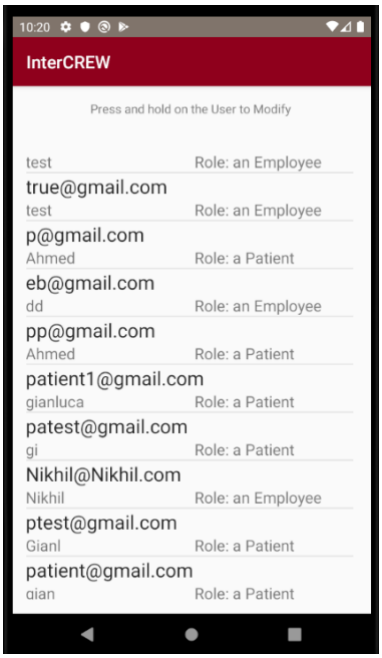
Sign-Up



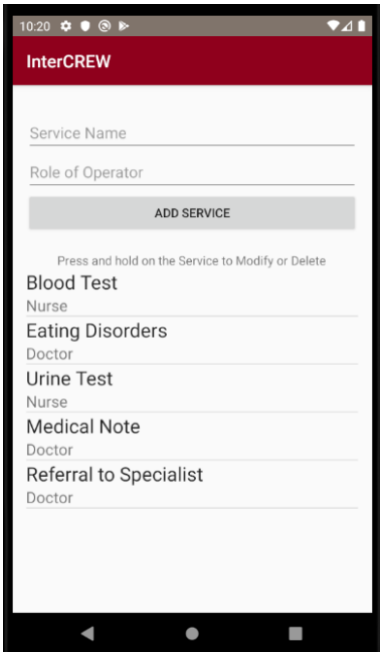
Admin Home Screen



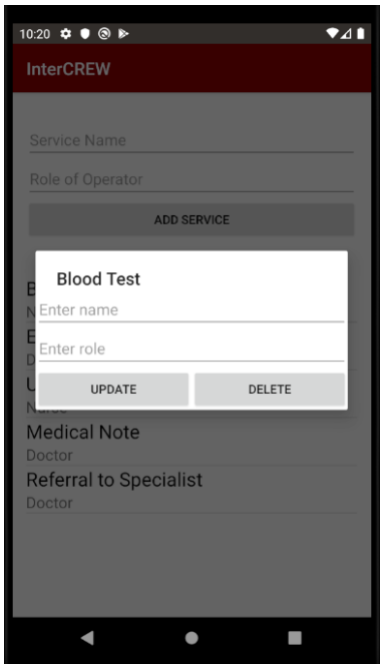
Modify Users



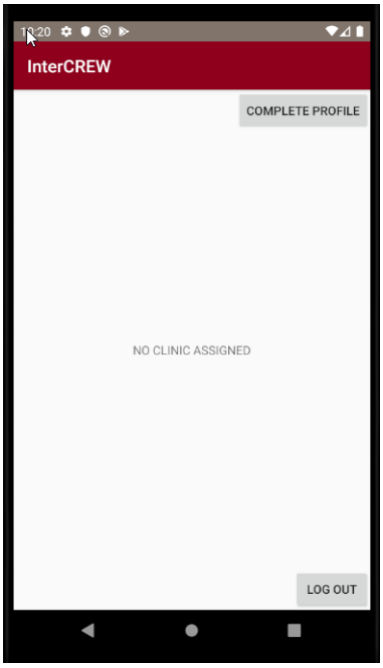
Modify Services Activity

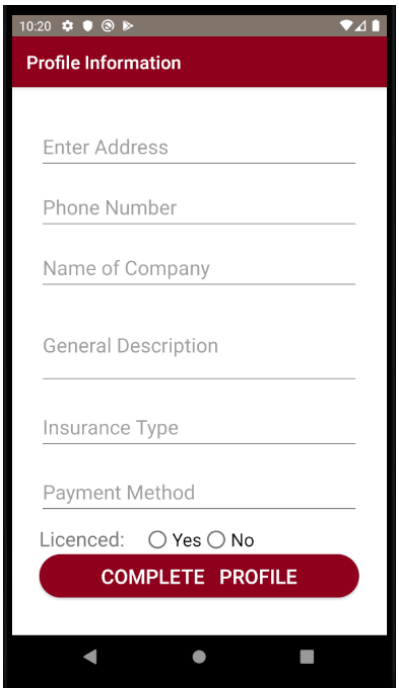


Update or Delete Dialog

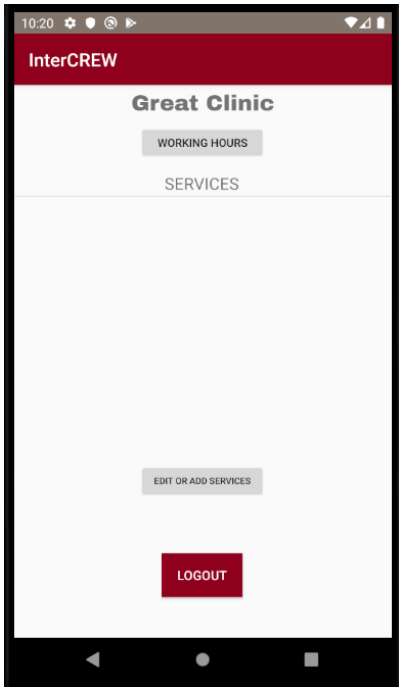


Employee HomeScreen
Complete Profile Information

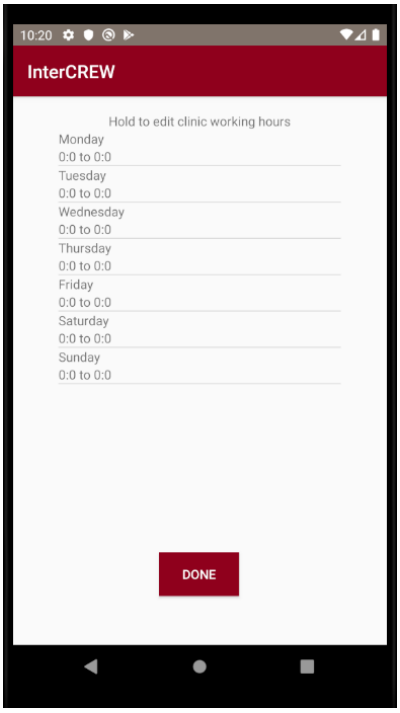




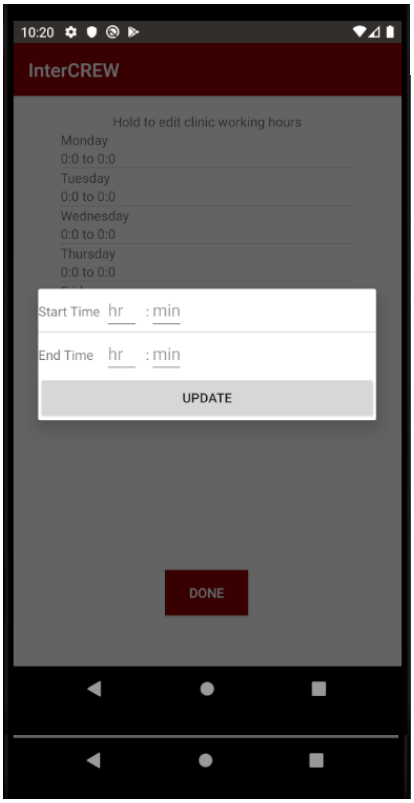
Home Screen Employee with Clinic



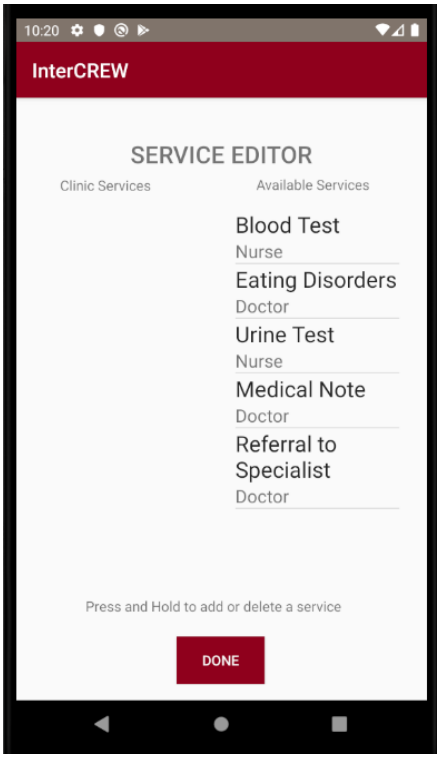
Edit Clinic Working Hours



Edit Working Hours Dialog

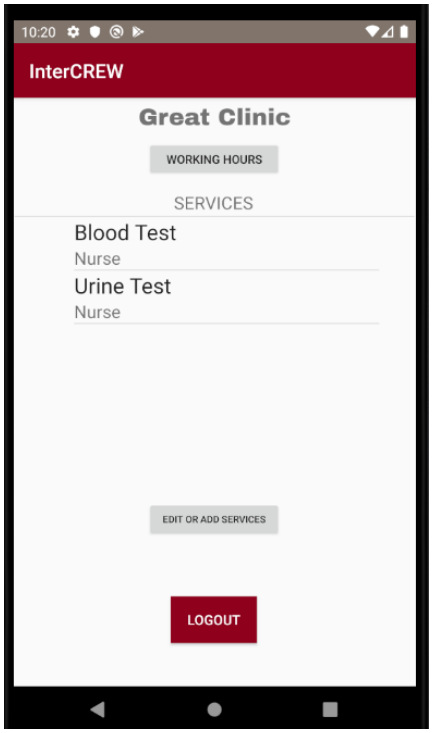


Service Editor Activity

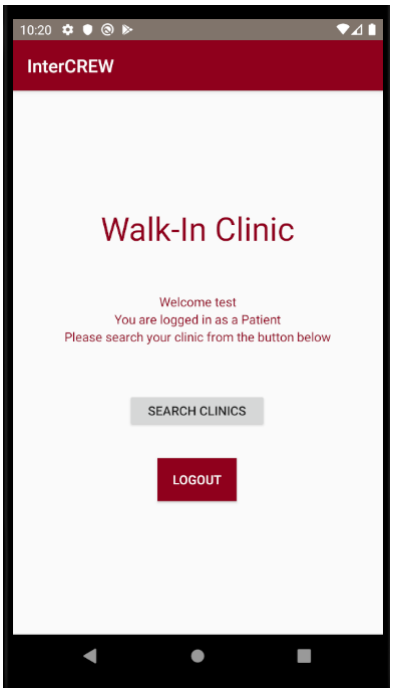


Add Service Dialog Box

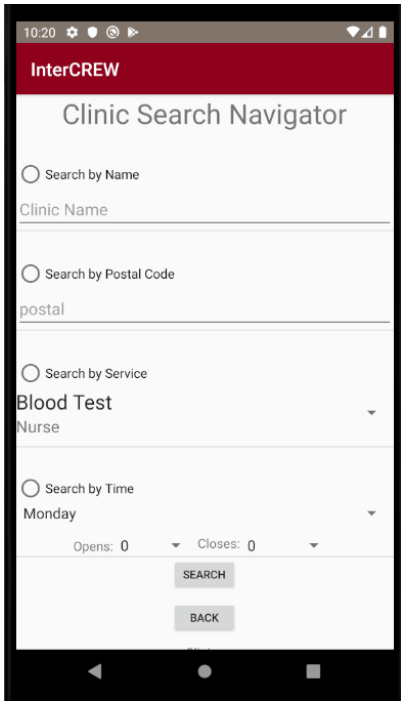
Employee HomePage With Services



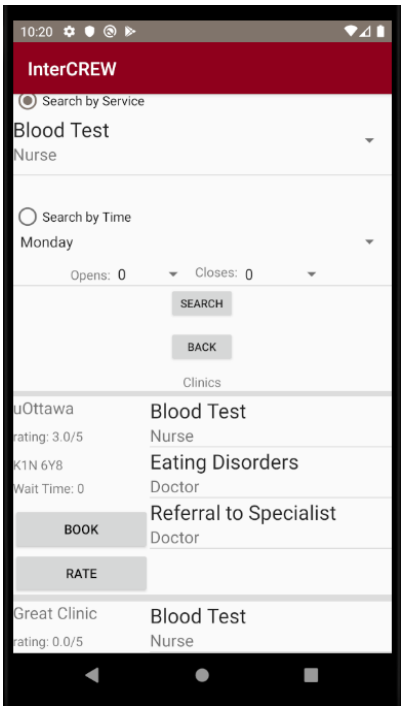
Patient Home Screen



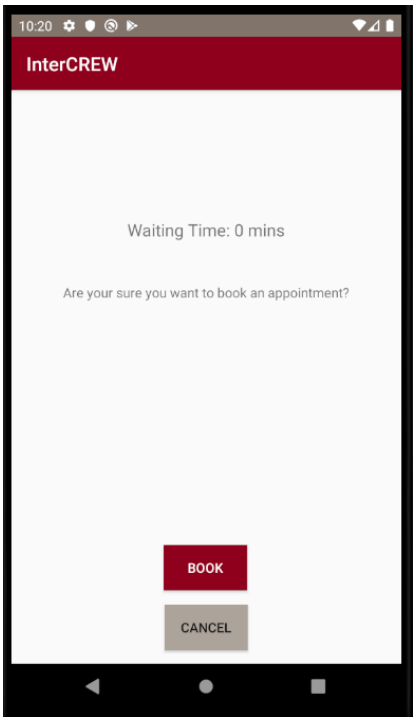
Search Clinics Page



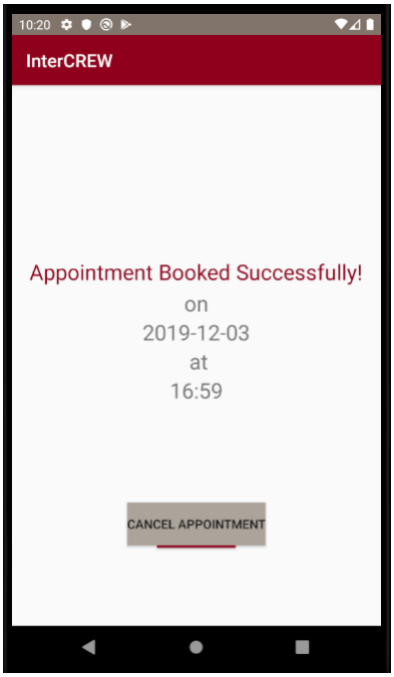
Clinic Search with Clinics



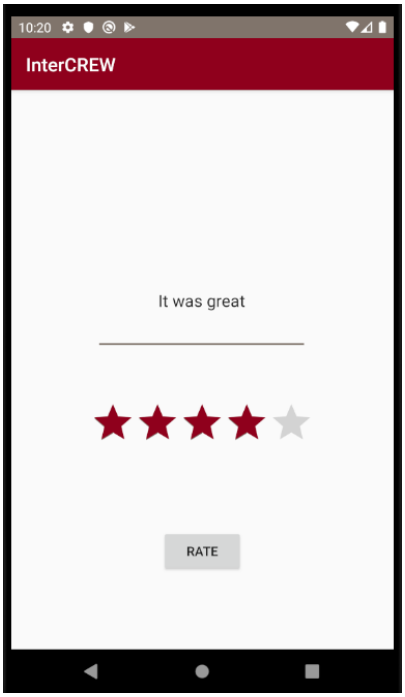
Book Appointment Screen



Appointment Booked



Rate Clinic Page



Lessons Learned

Over the course of this project we ran into many obstacles. These obstacles included, commits that caused errors making it hard for other group members to understand what is going wrong, learning new software and skills, and trying to conceptualize the best way to do a task.

The first skill that we had to learn was how to use GitHub. GitHub was essential for us to be able to work on the app together. Learning to create branches and merge conflicts was very important to learn early on in the development. As a group we decided that it was important that whenever a member pushes anything, the code must run without any errors.

A lot of the work in our app was working with a database. For our project we used Firebase for authentication and storing our user, clinic, and service data. We had to learn how to set up the database, store data into the database from user input and retrieve data from the database.

In order to get the work done for each deliverable we had to divide up the work in a way that members would be able to get their part done when they had time to. Because parts of the app sometimes relied on other parts this was something that had to be carefully done. Another aspect to the group work was accountability and having deadlines before the submission date to ensure everything was done.

Throughout the project we used a few different ways of communicating and having our meetings. We met up in person, and had calls over discord, skype, and used WhatsApp for instant messaging. We learned that having in-person meetings yielded the best results in terms of brainstorming and having the whole team on the same page for the objectives for our next deliverable. The online calls were effective for updates and were convenient as we could have impromptu meetings whenever needed, but were not

as ideal as the in person meetings for brainstorming. WhatsApp was something we used from the very beginning constantly to keep in communication and stay on track with what was needed to be done.

We learned that testing was something that helped our app run they way we expect it to. Using multiple types of testing and how to use them helped to insure our logic, and user interface worked the way we expect it to. We quickly saw the usefulness tests brought to ensure our app met our requirements.

We discovered a lot about what AndroidStudio offers and how to develop android apps. Through AndroidStudio we learned about activities, how to design an effective UI, and use different dependencies to make our app better.

This group project provided our whole group with introductory into developing and designing apps, and working with a team. We learned a whole set of new valuable skills we can keep moving forward.