SEG2105 Introduction to Software Engineering- Fall 2021
Android Project: Byblos Mobile Application

Charbel Habchi 300166056
Eshaan Kunchur 300176301
Rabih Daoud 300164191
Jeyason Jeyaparan 300165084
Sabina Solomon 300174303

Group 60
December 8, 2021
Professor Miguel Garzon

Introduction

The purpose of this report is to provide the screenshots of the Byblos application which is an application used by auto rental/moving services branches to facilitate customer service requests. The application allows customers to select a branch and fill out a service request form which can later be approved or denied by the branch receiving the form. They may also give the branch a rating and a comment. The admin creates the services available to the branches to select from and indicates which form fields are required. The branch functionality also includes filling out a branch profile.

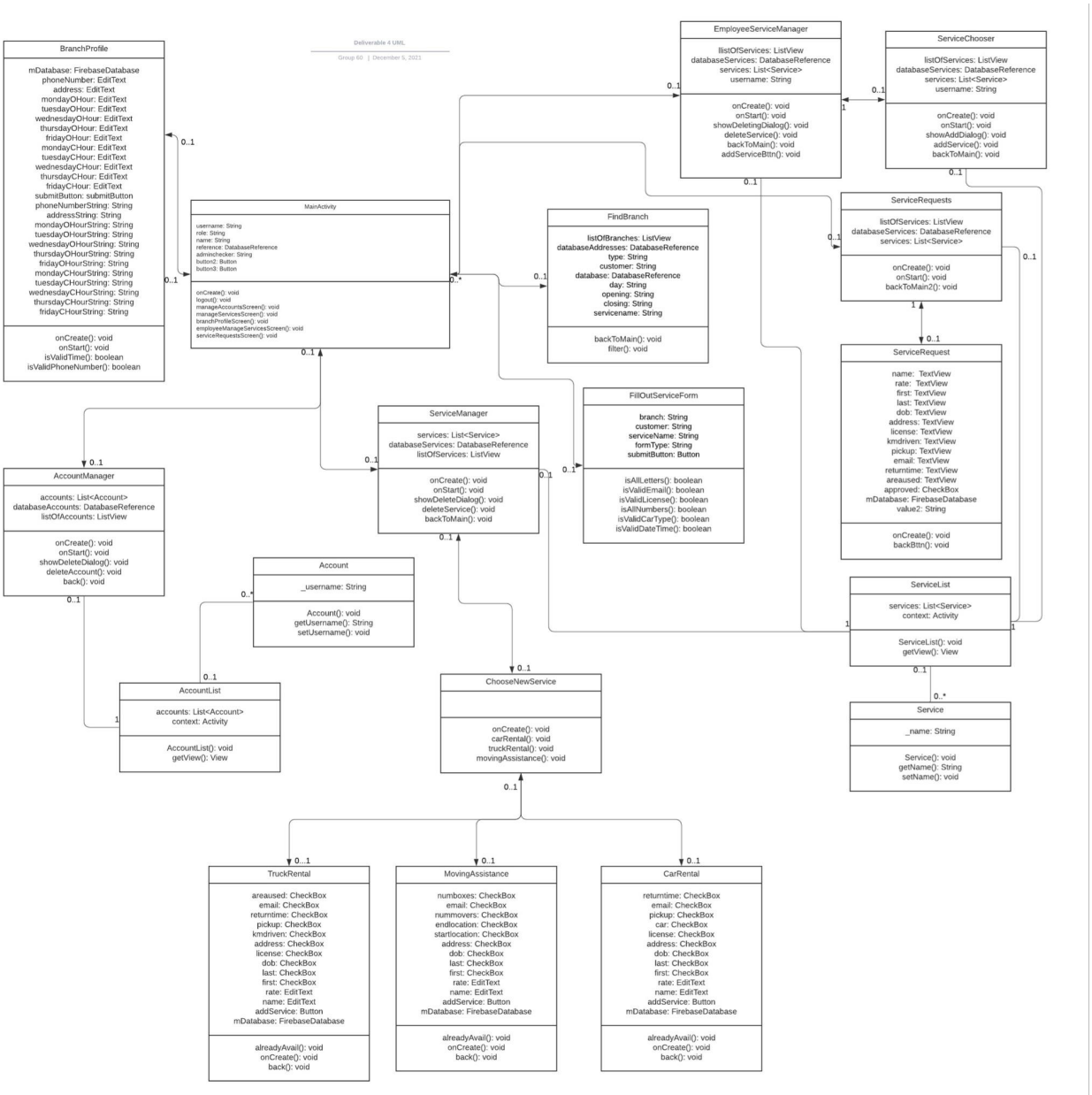
The project developed software engineering skills through application development. These skills include group work, design, Firebase, and Android development. The report ends with a lessons learned section.

Contributions

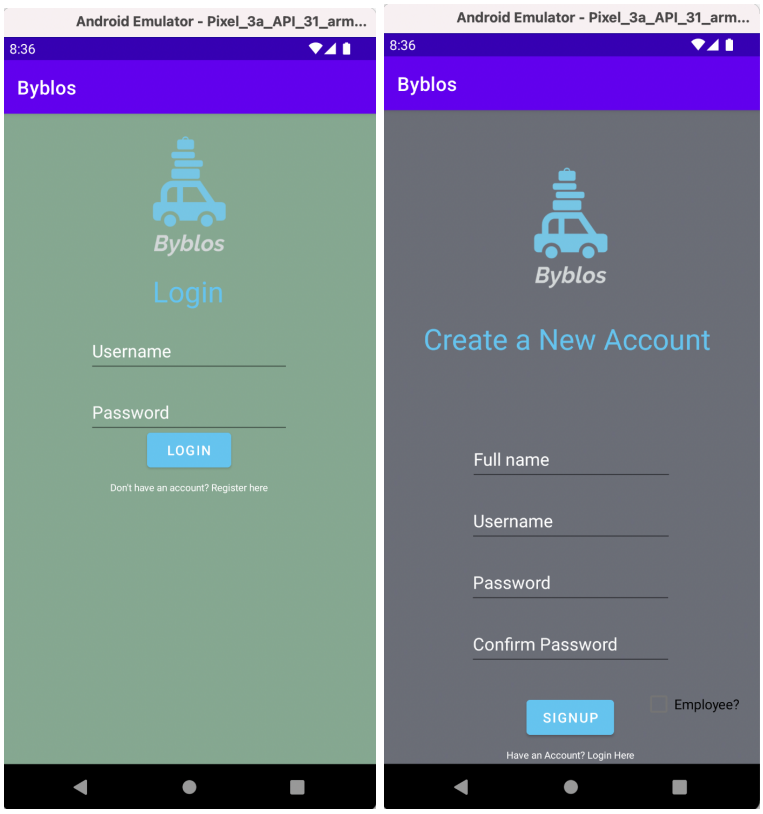
Name	**Deliverable 1**	**Deliverable 2**	**Deliverable 3**	**Deliverable 4**
Charbel Habchi	-contributed to the UML diagram -Set up firebase -Creation of employee/customer account types -Welcome page displaying name and username	-UML diagram -created all 3 services which can be offered by the branches -removing services ability -editing services ability -removing accounts ability -field validation	-contributed to the UML diagram -hooked up branch profile to automatically updating firebase -branch ability to add services -branch ability to delete services -ability to display service requests and approve/deny them	-UML diagram -ability to filter through branches by address/services available/working hours -branch rating -hooked up customer service form to firebase to automatically save form and display on branch side -contributed to the final report
Eshaan Kunchur	-Contributed to UML diagram - Helped design the flow of user interactions with the application - Worked on	-Contributed to UML diagram - Helped with field validation - Helped plan user interaction	- Created Branch Profile field validation system for different field types (working hours, phone	- “Lessons Learned” portion of the report - Revised the UML

	Branch Employee code		number, address, etc.)	
Rabih Daoud	-created the logo and the UI for the login page -contributed to UML diagram	-Contributed to UML diagram -Aided in the planning of user engagement	-Aided in the planning of user engagement -Created the manage services page	-Created 10 unit test cases -Aided in the final report
Jeyason Jeyaparan	-Contributed to the UML diagram -Aided in the design of the UI for the login page	-Contributed to the UML diagram -Assisted with the manage services functionality and UI of the project	-Contributed to the UML diagram -Assisted with the functionality of the manage services page -Assisted with the functionality of the buttons of the manage services page	-Contributed to the UML diagram -Assisted with the development of the fillOutServices form -Aided in the final report
Sabina Solomon	-Contributed to UML diagram -Participated in UI design -Bug fixes and field validation	-Helped design user interface/ user interaction - Implemented ChooseNewService -Contributed to the UML diagram	-Helped with Branch Profile and created the UI -Participated in planning user interaction -Updated the functionality and UI of MainActivity for employees/ branches	-Created and implement user service form that adapts to all different types of services -Created field validation for the service form -Developed user ability to select the which service form they want to fill out for the particular branch -Aided in the final report

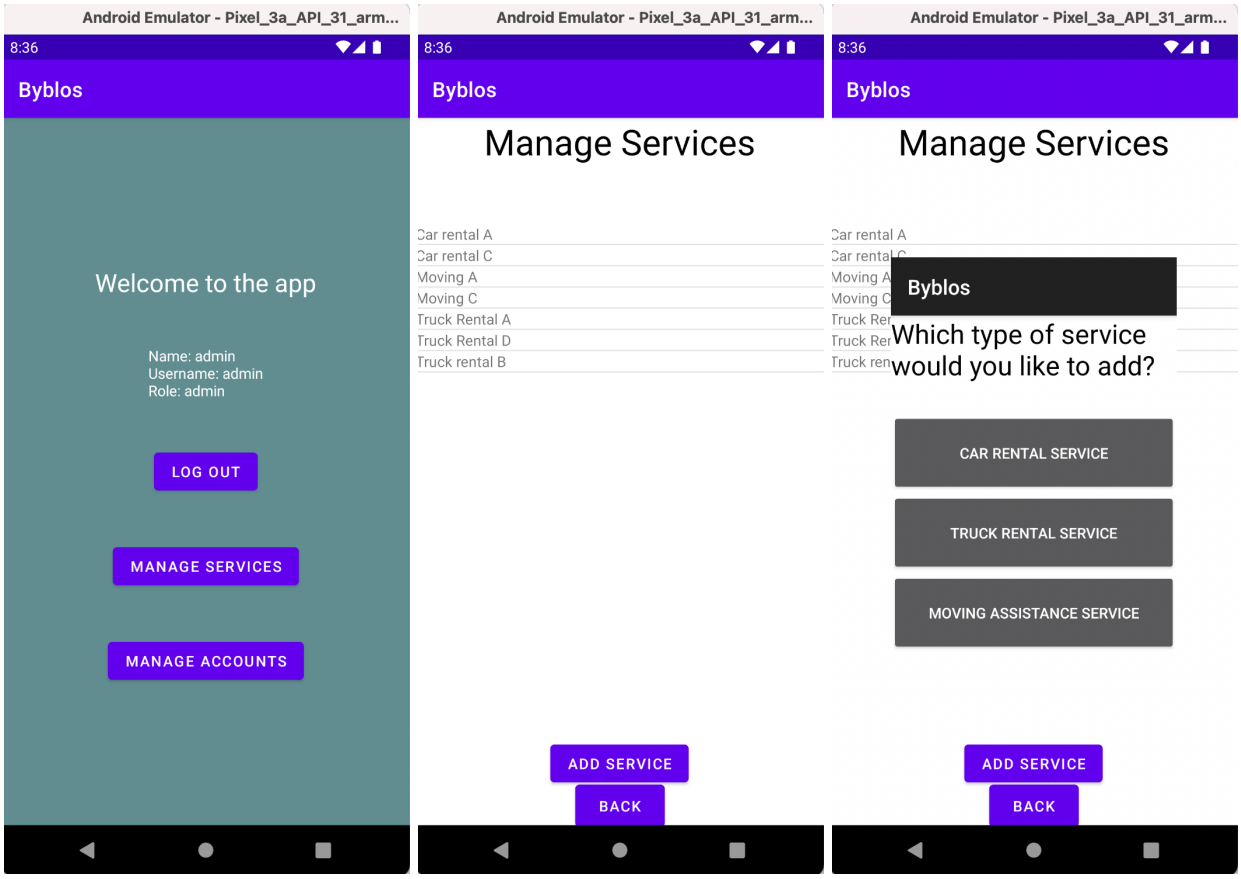
UML Diagram


Deliverable 4 UML
Group 60 | December 5, 2021
BranchProfile
mDatabase: FirebaseDatabase
phoneNumber: EditText
address: EditText
mondayOHour: EditText
tuesdayOHour: EditText
wednesdayOHour: EditText
thursdayOHour: EditText
fridayOHour: EditText
mondayCHour: EditText
tuesdayCHour: EditText
wednesdayCHour: EditText
thursdayCHour: EditText
fridayCHour: EditText
submitButton: submitButton
phoneNumberString: String
addressString: String
mondayOHourString: String
tuesdayOHourString: String
wednesdayOHourString: String
thursdayOHourString: String
fridayOHourString: String
mondayCHourString: String
tuesdayCHourString: String
wednesdayCHourString: String
thursdayCHourString: String
fridayCHourString: String
onCreate(): void
onStart(): void
isValidTime(): boolean
isValidPhoneNumber(): boolean
EmployeeServiceManager
llistOfServices: ListView
databaseServices: DatabaseReference
services: List<Service>
username: String
onCreate(): void
onStart(): void
showDeletingDialog(): void
deleteService(): void
backToMain(): void
addServiceBttn(): void
ServiceChooser
listOfServices: ListView
databaseServices: DatabaseReference
services: List<Service>
username: String
onCreate(): void
onStart(): void
showAddDialog(): void
addService(): void
backToMain(): void
MainActivity
username: String
role: String
name: String
reference: DatabaseReference
adminchecker: String
button2: Button
button3: Button
onCreate(): void
logout(): void
manageAccountsScreen(): void
manageServicesScreen(): void
branchProfileScreen(): void
employeeManageServicesScreen(): void
serviceRequestsScreen(): void
FindBranch
listOfBranches: ListView
databaseAddresses: DatabaseReference
type: String
customer: String
database: DatabaseReference
day: String
opening: String
closing: String
servicename: String
backToMain(): void
filter(): void
ServiceRequests
listOfServices: ListView
databaseServices: DatabaseReference
services: List<Service>
onCreate(): void
onStart(): void
backToMain2(): void
ServiceRequest
name: TextView
rate: TextView
first: TextView
last: TextView
dob: TextView
address: TextView
license: TextView
kmdriven: TextView
pickup: TextView
email: TextView
returntime: TextView
areaused: TextView
approved: CheckBox
mDatabase: FirebaseDatabase
value2: String
onCreate(): void
backBttn(): void
FillOutServiceForm
branch: String
customer: String
serviceName: String
formType: String
submitButton: Button
isAllLetters(): boolean
isValidEmail(): boolean
isValidLicense(): boolean
isAllNumbers(): boolean
isValidCarType(): boolean
isValidDateTime(): boolean
ServiceManager
services: List<Service>
databaseServices: DatabaseReference
listOfServices: ListView
onCreate(): void
onStart(): void
showDeleteDialog(): void
deleteService(): void
backToMain(): void
AccountManager
accounts: List<Account>
databaseAccounts: DatabaseReference
listOfAccounts: ListView
onCreate(): void
onStart(): void
showDeleteDialog(): void
deleteAccount(): void
back(): void
Account
_username: String
Account(): void
getUsername(): String
setUsername(): void
ServiceList
services: List<Service>
context: Activity
ServiceList(): void
getView(): View
Service
_name: String
Service(): void
getName(): String
setName(): void
AccountList
accounts: List<Account>
context: Activity
AccountList(): void
getView(): View
ChooseNewService
onCreate(): void
carRental(): void
truckRental(): void
movingAssistance(): void
TruckRental
areaused: CheckBox
email: CheckBox
returntime: CheckBox
pickup: CheckBox
kmdriven: CheckBox
address: CheckBox
license: CheckBox
dob: CheckBox
last: CheckBox
first: CheckBox
rate: EditText
name: EditText
addService: Button
mDatabase: FirebaseDatabase
alreadyAvail(): void
onCreate(): void
back(): void
MovingAssistance
numboxes: CheckBox
email: CheckBox
nummovers: CheckBox
endlocation: CheckBox
startlocation: CheckBox
address: CheckBox
dob: CheckBox
last: CheckBox
first: CheckBox
rate: EditText
name: EditText
addService: Button
mDatabase: FirebaseDatabase
alreadyAvail(): void
onCreate(): void
back(): void
CarRental
returntime: CheckBox
email: CheckBox
pickup: CheckBox
car: CheckBox
license: CheckBox
address: CheckBox
dob: CheckBox
last: CheckBox
first: CheckBox
rate: EditText
name: EditText
addService: Button
mDatabase: FirebaseDatabase
alreadyAvail(): void
onCreate(): void
back(): void

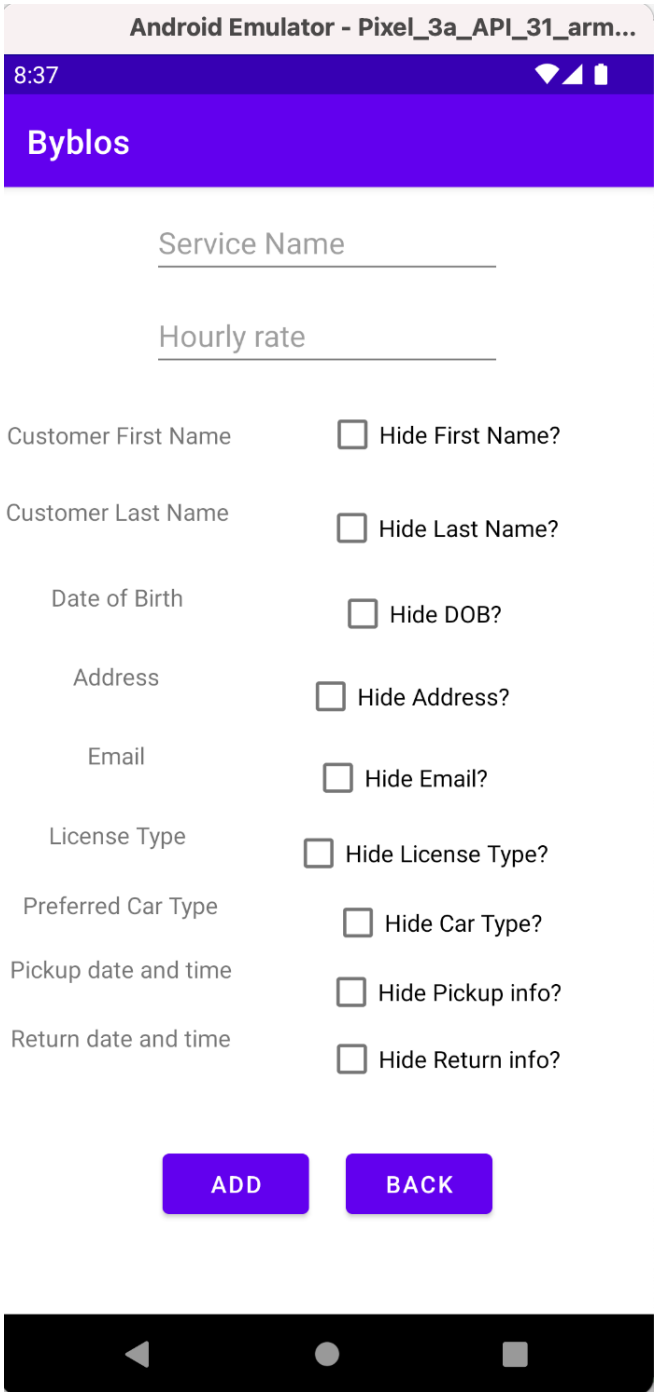
Screenshots



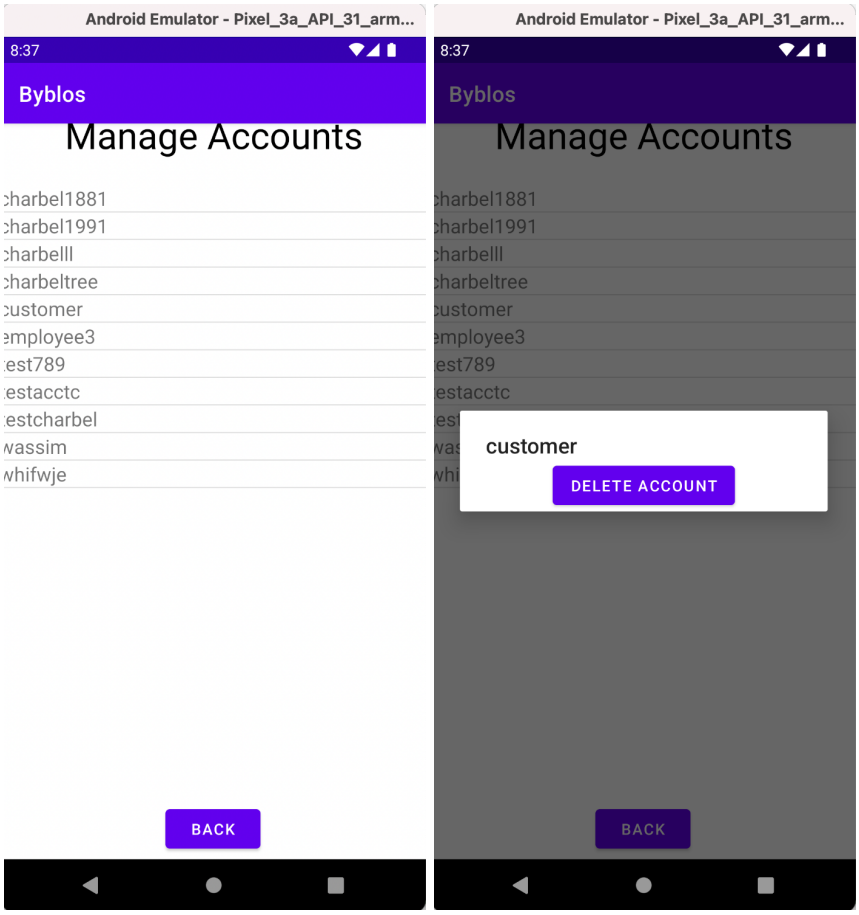
The login and register screens



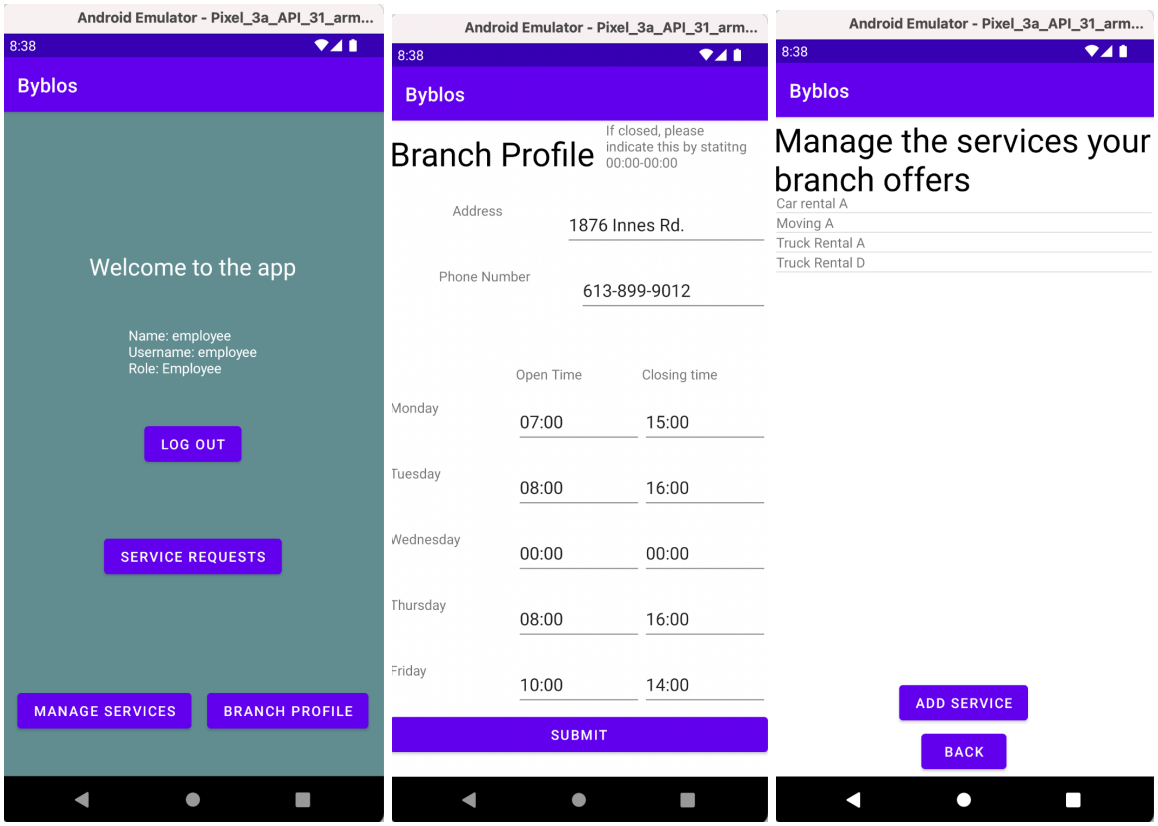
Admin functionality



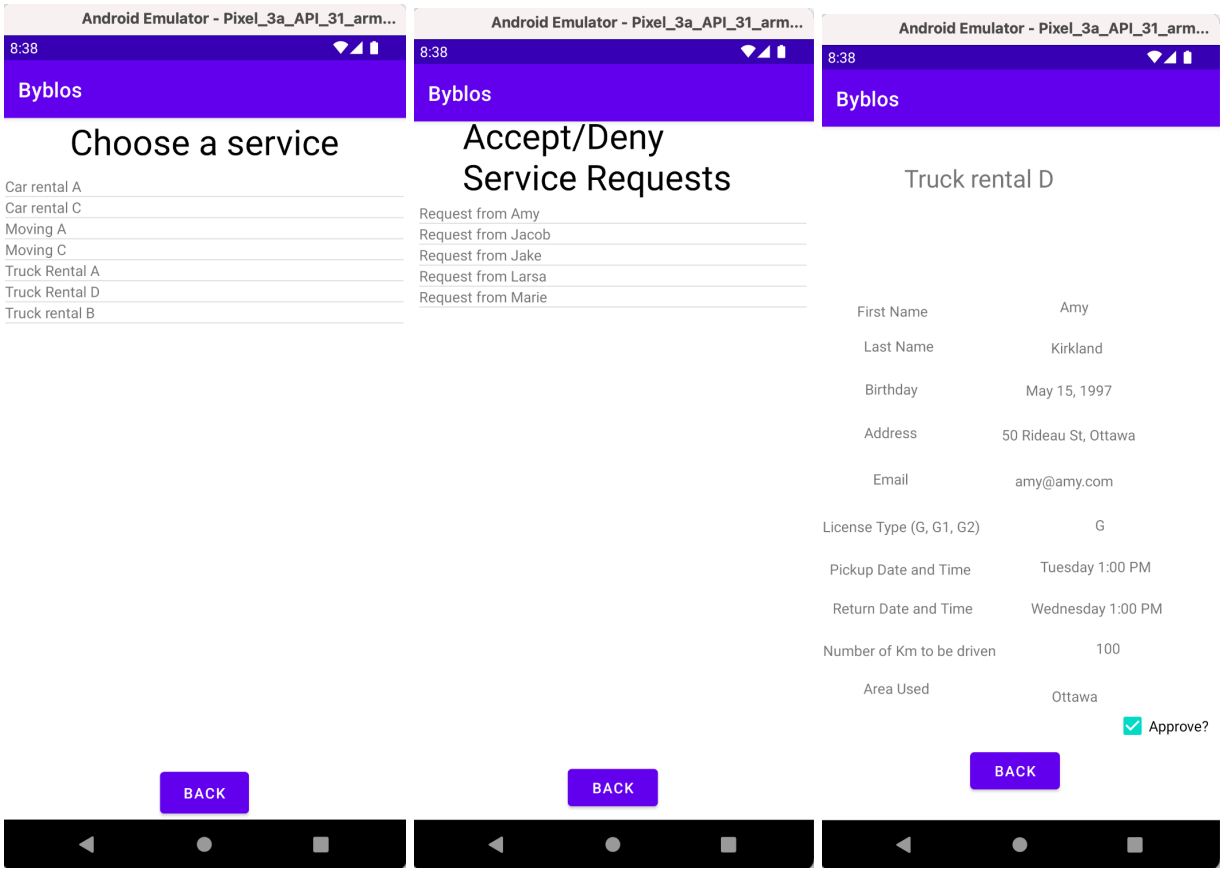
Admin creating a service form



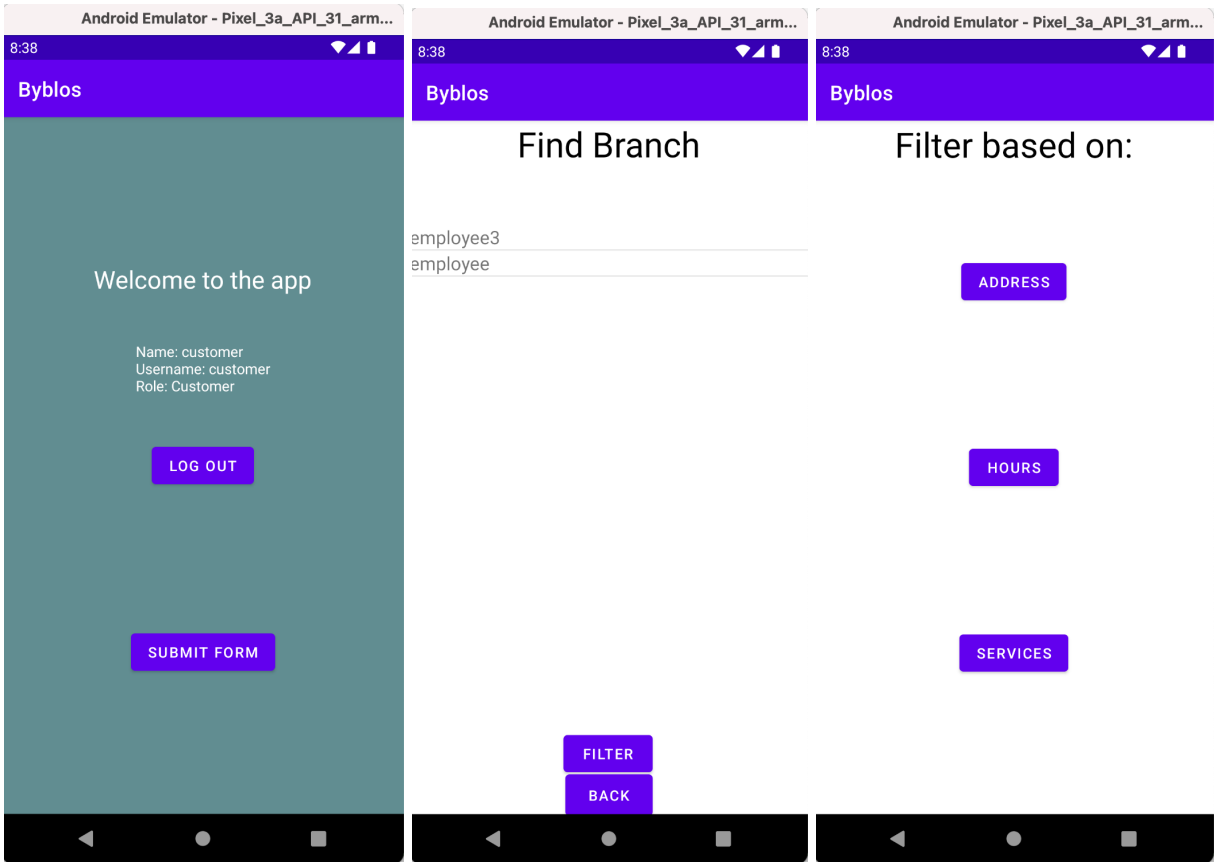
Admin managing/deleting account functionality



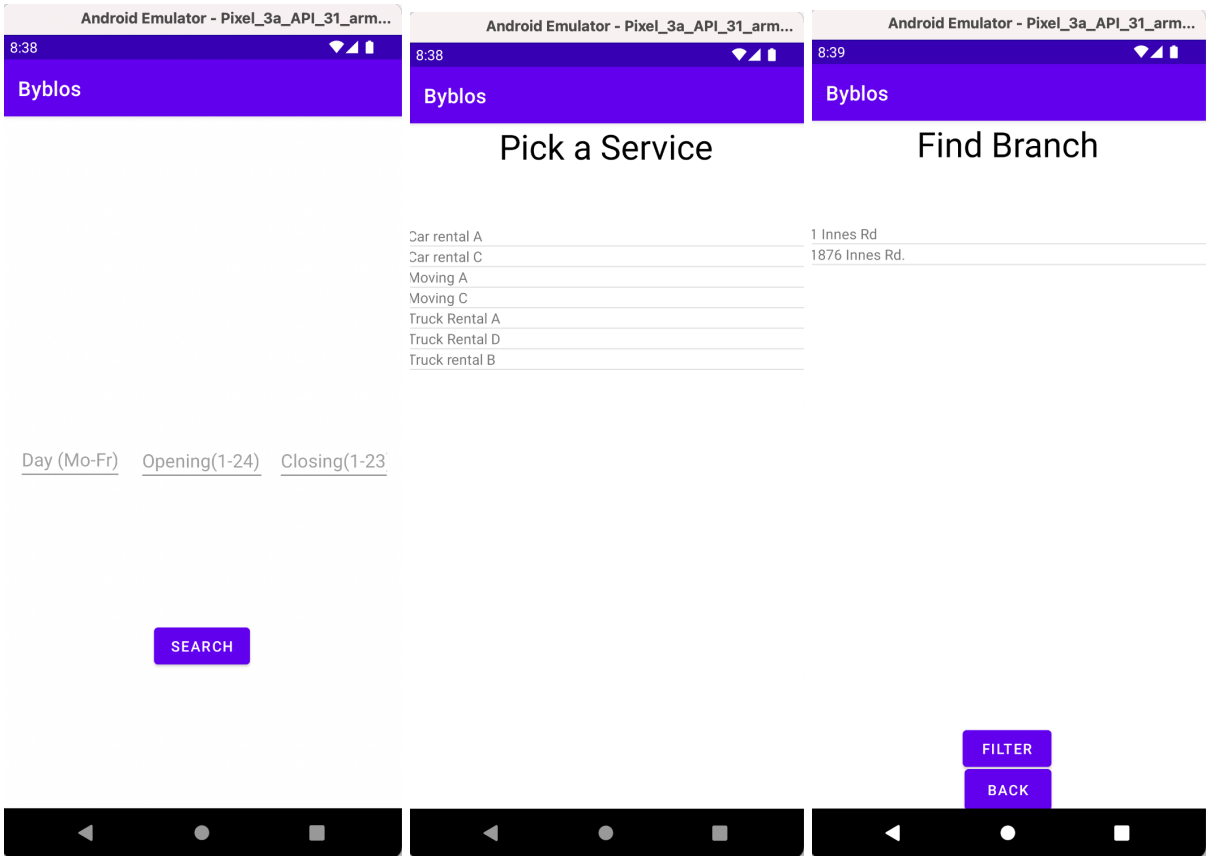
Employee functionality



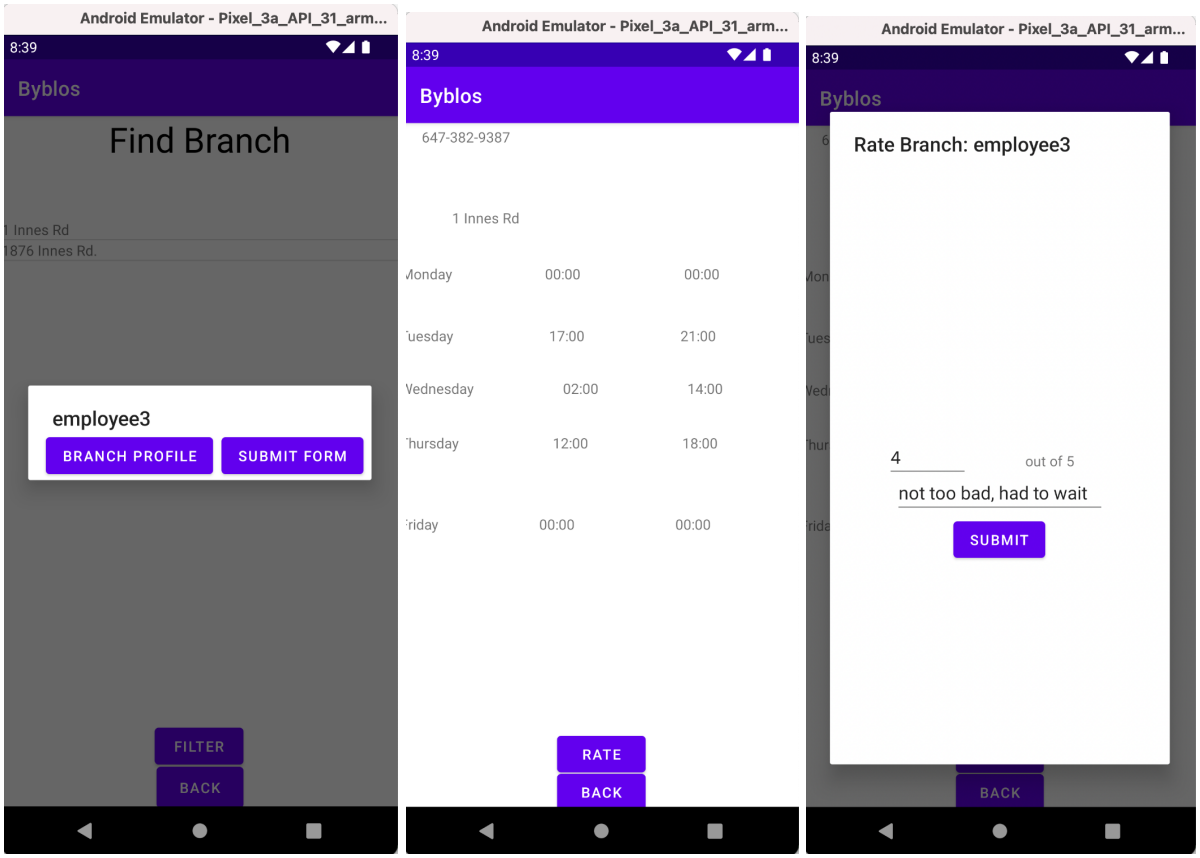
Employee choosing a new service to add/accepting or denying service requests



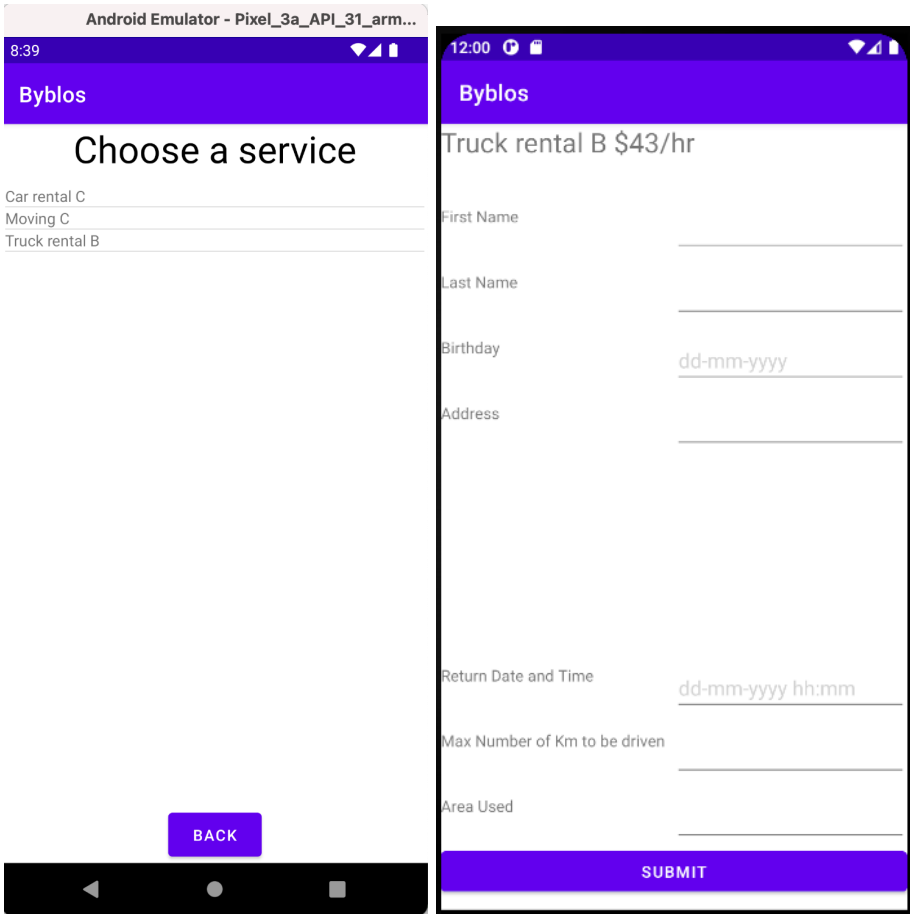
Customer functionality, finding branches and filtering them



Customer filtering by hours/services/branch addresses



Customer viewing branch profile and giving it a rating



Customer choosing a service offered by the branch and filling out a form

Lessons learned

Throughout the course of the semester, our group was able to apply many of the concepts we learned in class to our project. Furthermore, we gained competence in Android application development, and became much more adept at handling version control through Git. Although we are pleased with our team's performance and the quality of our final product, there are a few important lessons that we all took from this experience.

In the early stages of developing our mobile application, our group quickly noticed that our lack of Android development experience was hindering our progress. Initially, our team struggled to implement some of the functionalities and designs that we aimed for our app to include. In this stage of development, we learned the importance of gathering information prior to planning and implementing. After spending a few days getting accustomed to the nuances of Android (through tutorials and proof-of-concept programs) our team felt much more comfortable moving forward.

Another lesson that our team learned from this experience was the importance of working on smaller components first, and then combining them into larger modules. The task of implementing a fully-functional auto rental/moving services application that leverages a Backend-as-a-Service tool for data storage seemed daunting at first. However, as we began to focus on the smaller parts of the puzzle first through a bottom-up design approach, we were able to get tasks done and slowly build up our product through each component.

Another very important lesson we learned was the importance of planning extra time for our tasks. We learned that it is difficult to place a tight time expectation for a task to get done, as we would often encounter technical difficulties that would consume extra hours. Furthermore, with Android and Firebase being new technologies to all of our group members, errors were much more likely to occur, and debugging them would be a lengthy process as well. Therefore, we put our best efforts towards being proactive with deliverables and starting them early on in the week they were released.

Overall, our team enjoyed collaborating on this project. Nearly all software being developed in the industry relies on flexible, well-rounded teams, and we believe that this experience helped us gain an understanding for why certain industry practices are followed in group software development.