HelpLah Milestone 3

Motivation

Today, when someone is looking for various providers of maintenance services such as cleaning services or a plumber or air-conditioning servicing, they have 2 alternatives to go about this.

The first of which is to ask friends and families for any recommendations. The second is to scour through the yellow pages or Facebook groups and forums to look for any providers. The problem with the first option is the lack of choice. You just have to trust that the providers your friends and families engage with are reliable and charge a fair price. The problem with the latter is that even though you may often be able to find more choices, it is difficult for a user to judge which is good and which is bad due to the lack of other customer's feedback. It is also time-consuming to have to individually call each of these providers to enquire about their pricing and availability. Also, you are sure that everyone uses these services from time to time, so why isn't there an easier way to go about this?

About our application

For our orbital project, we have created an android app using Android Studio that serves as a centralised platform that allows providers of maintenance services in Singapore to advertise their services and for customers to view them. Our app also contains a hiring system where customers can send a job request to a provider for a job on a certain date and time. We also provide a chat functionality for customers to talk to businesses and a review system for customers to leave their feedback for other potential customers to view.

What's new

Since we have already completed all the features we set out to create in milestone 1 and 2, we only added a few minor features for milestone 3. We created a basic push notification system using Firebase Cloud Messaging so that users can view and click and be alerted to new notifications even when not using the app. We also added a few minor extended features for our job request and listing feature which you can view below. You can also view our milestone 2 and 3 videos to find out more about our app's features.

We also did some more unit testing and user testing for milestone 3 and managed to uncover some bugs that we have described below.

We have also created some documentation and charts for our app so that future developers can more easily manage and add new features into the app. You can view this documentation below.

Core Features

Listings Search

This feature which is only available to consumer users, will allow users to look for businesses based on the different service categories. Upon logging in, users will be brought to the page as shown in figure 1.1 below. This page displays the different services that users can search for in the app. Currently only 12 services are provided but more can be added in the future.

Upon clicking a category, users are then brought to a page which displays all the available businesses that currently provide the service (Figure 1.2) . In that page, users can view the businesses availability, starting price and its review score. The availability figure shows how quickly a user can schedule a job request. For instance, a business with an availability of 4 hours means that they can accept a job request with a timing in the next 4 hours. This feature can be useful in an emergency. For instance, if I'm locked out of my house due to a faulty lock, I can use this feature to look for a locksmith to come down to my house as soon as possible.

Users also have the options to filter and sort the results as seen in figure 1.3. They can filter based on availability and also sort by score, price and popularity (Number of reviews).

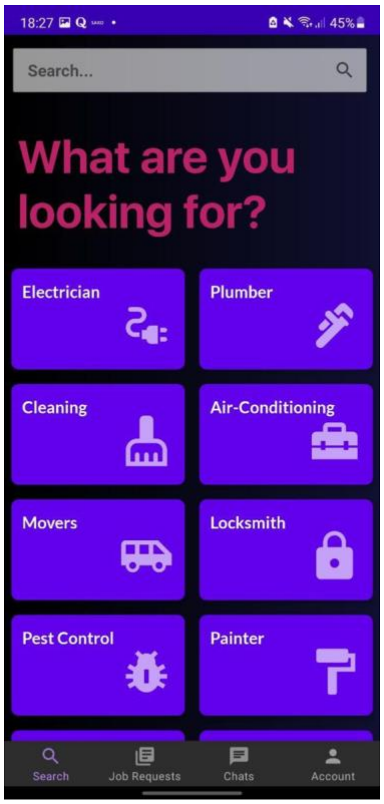
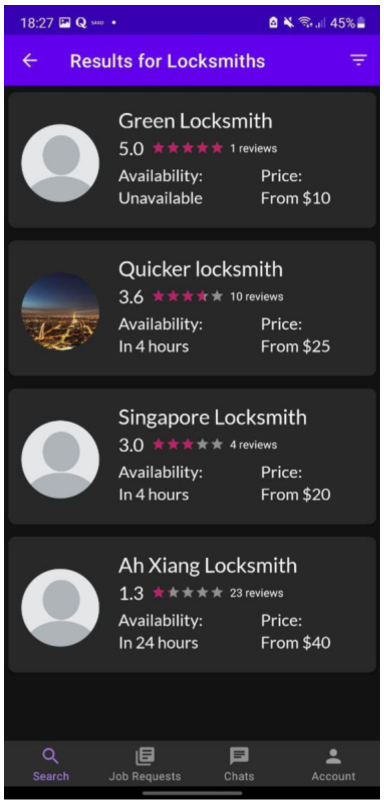


Figure 1.1



Results for Locksmiths

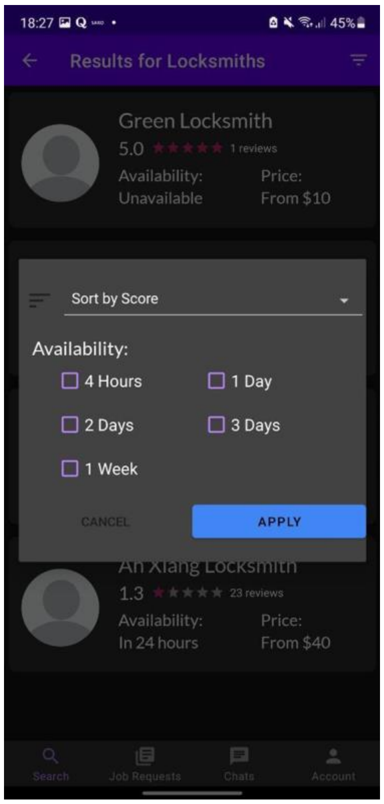
Green Locksmith
5.0 ★★★★★ 1 reviews
Availability: Unavailable
Price: From $10

Quicker locksmith
3.6 ★★★★★ 10 reviews
Availability: In 4 hours
Price: From $25

Singapore Locksmith
3.0 ★★★★★ 4 reviews
Availability: In 4 hours
Price: From $20

Ah Xiang Locksmith
1.3 ★★★★★ 23 reviews
Availability: In 24 hours
Price: From $40

Figure 1.2



Results for Locksmiths

Green Locksmith
5.0 ★★★★★ 1 reviews
Availability: Unavailable
Price: From $10

Sort by Score

Availability:
☐ 4 Hours ☐ 1 Day
☐ 2 Days ☐ 3 Days
☐ 1 Week

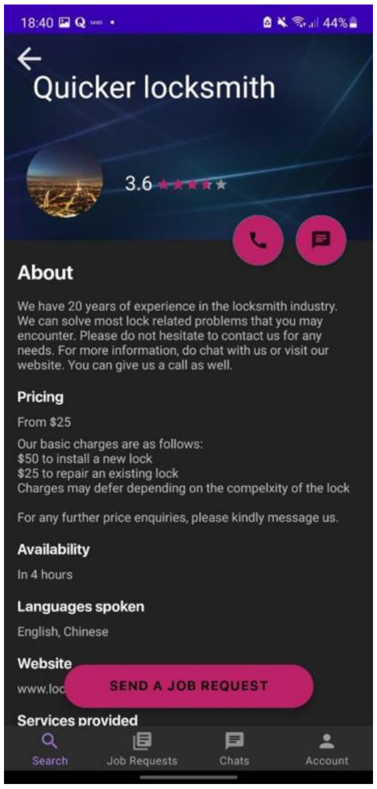
Ah Xiang Locksmith
1.3 ★★★★★ 23 reviews
Availability: In 24 hours
Price: From $40

Figure 1.3

Upon clicking a business listing, they are brought to a page (Figure 1.4) which shows a more in depth description about the selected listing. They can view a description of the listing, a pricing note written by listing which further breaks down the pricing charges, their website (if available), and also the languages spoken by the business. They also have three action buttons on this page. A user can chat with the business through the app's chat function, call the business or send a job request to the business.

Upon scrolling down they can see some of the reviews that have been written by other users who have engaged this business before (Figure 1.5 and 1.6). If the business has replied to the reviews, the user can also see these replies.

In figure 1.6, users can also filter the review results based on the review score. Also, since the app allows a business to provide multiple services, the review also adds a filter which allows users to see reviews only applicable to them. For example, if a user is looking for locksmiths, then the users can filter the reviews to only see reviews from users who have engaged this company as locksmiths. Other reviews from users who have engaged this business for other services are not shown.



18:40 44%

Quicker locksmith

3.6

About

We have 20 years of experience in the locksmith industry. We can solve most lock related problems that you may encounter. Please do not hesitate to contact us for any needs. For more information, do chat with us or visit our website. You can give us a call as well.

Pricing

From $25

Our basic charges are as follows:
$50 to install a new lock
$25 to repair an existing lock
Charges may defer depending on the compelxity of the lock

For any further price enquiries, please kindly message us.

Availability

In 4 hours

Languages spoken

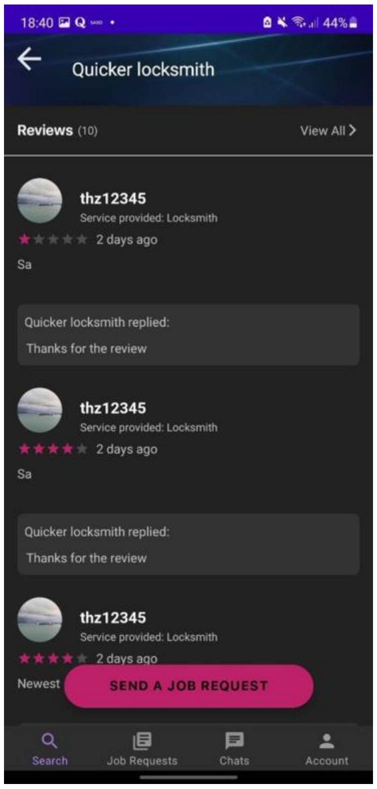
English, Chinese

Website

www.loc

Services provided

Figure 1.4



18:40 44%

Quicker locksmith

Reviews (10) View All

thz12345
Service provided: Locksmith
2 days ago
Sa

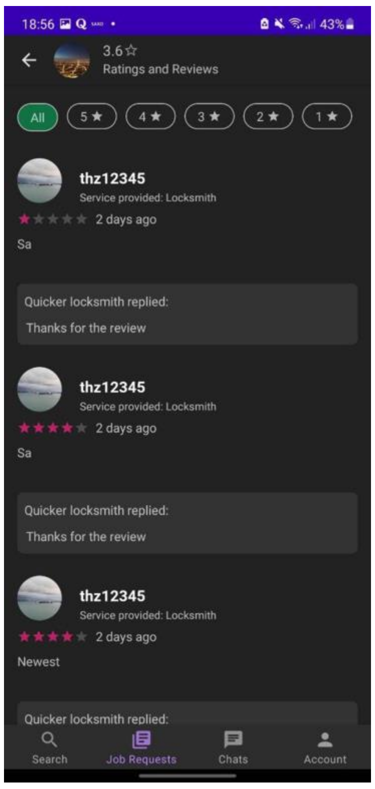
Quicker locksmith replied:
Thanks for the review

thz12345
Service provided: Locksmith
2 days ago
Sa

Quicker locksmith replied:
Thanks for the review

thz12345
Service provided: Locksmith
2 days ago
Newest

Figure 1.5



18:56 43%

3.6☆
Ratings and Reviews

thz12345
Service provided: Locksmith
2 days ago
Sa

Quicker locksmith replied:
Thanks for the review

thz12345
Service provided: Locksmith
2 days ago
Sa

Quicker locksmith replied:
Thanks for the review

thz12345
Service provided: Locksmith
2 days ago
Newest

Quicker locksmith replied:

Figure 1.6

For milestone 3, we added a search functionality in our home page, as seen in figure 1.7, for consumer users to search for a particular listing without having to select a category and manually finding the listing. This will allow a user to easily search for a listing they have in mind. This function is done by combining the cloud Firestore data with Algolia so as to enable text search.

When users search for a particular listing, all the listings which match the search phrase are presented to them as seen in figure 1.8. The number of results and time taken to carry out the search is also displayed.

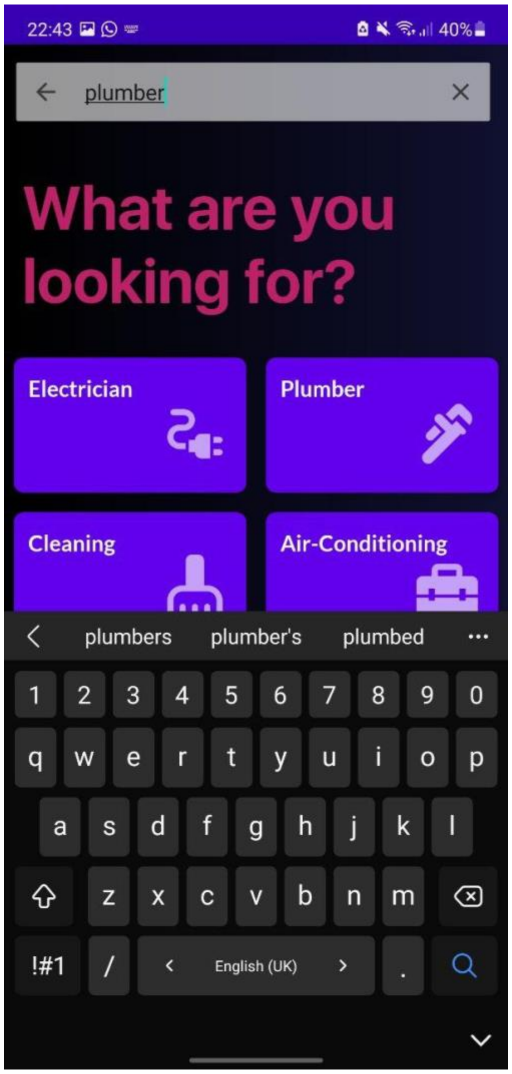
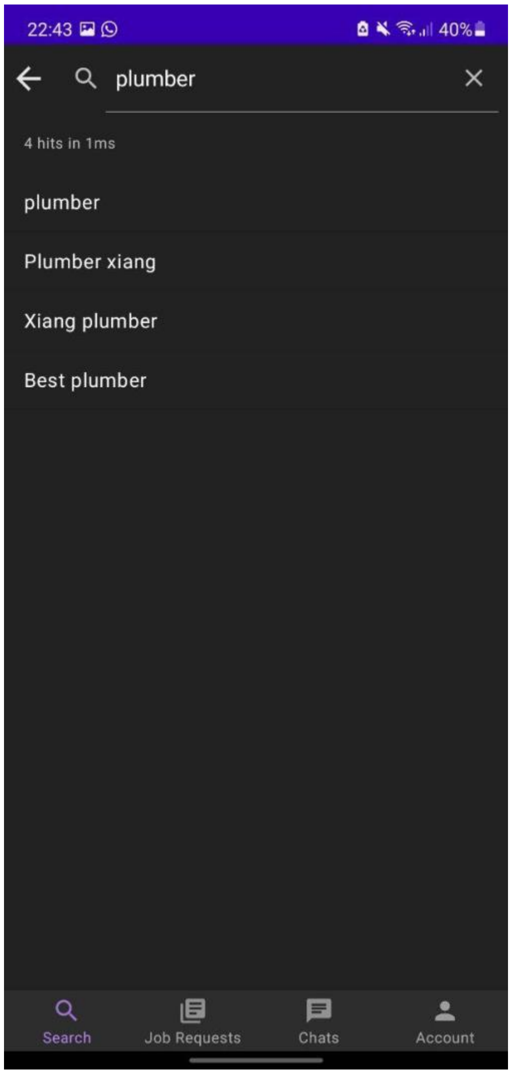


Figure 1.7



4 hits in 1ms

plumber

Plumber xiang

Xiang plumber

Best plumber

Figure 1.8

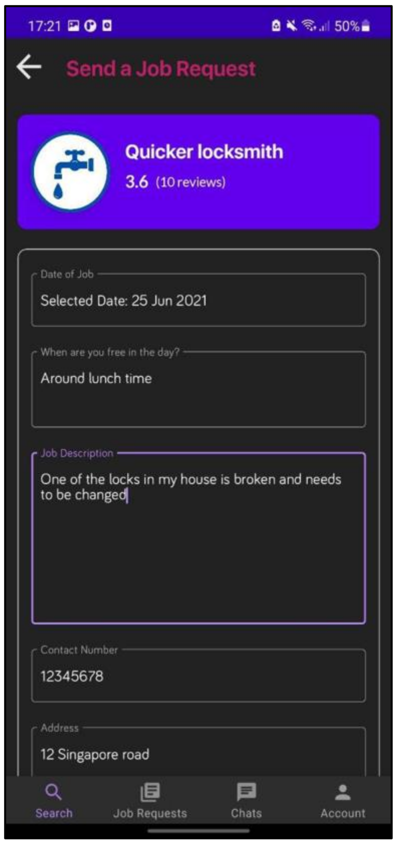
<u>Job Requests</u>

After selecting a business that they are interested in, a user can then send a job request to the business. In order to send a job request, a user has to fill out certain information about the job request as seen in Figure 2.1 below.

Information required to send a job request:

- A job date
- A job description
- A timing note such as what time of the day they are free at.
- Their personal details such as their contact and address. (These details will be auto filled if the users has set these details during the registration process)

The job request then gets uploaded to the app's Firestore database. A user can then view the job request he sent in the job requests tab of the app (Figure 2.2). He can then expand it as seen in Figure 2.3 in order to view more information about the job request. In the expanded view, there are also options presented for the user to click. In this case, the user can edit the job request, cancel it or chat with the recipient of the job request.



Send a Job Request

Quicker locksmith
3.6 (10 reviews)

Date of Job
Selected Date: 25 Jun 2021

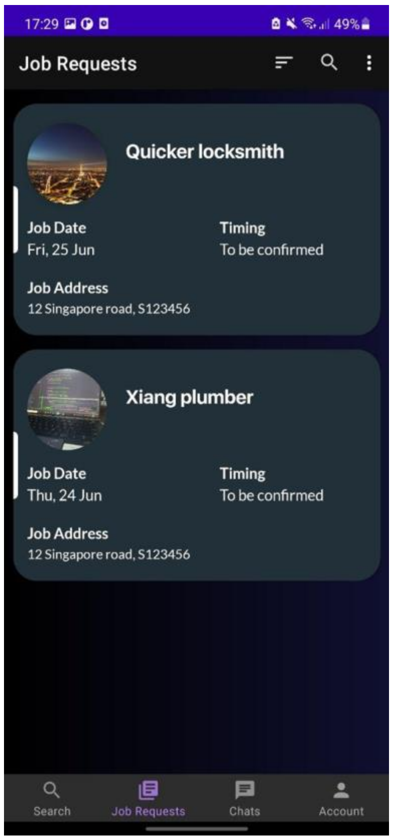
When are you free in the day?
Around lunch time

Job Description
One of the locks in my house is broken and needs to be changed

Contact Number
12345678

Address
12 Singapore road

Figure 2.1

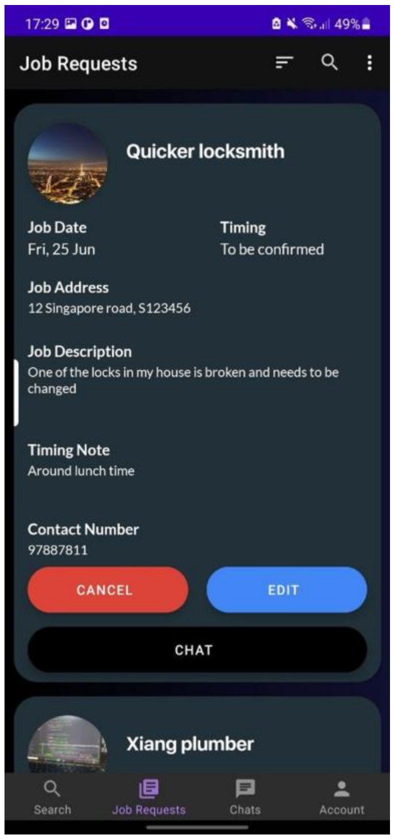


Job Requests

Quicker locksmith
Job Date: Fri, 25 Jun
Timing: To be confirmed
Job Address: 12 Singapore road, S123456

Xiang plumber
Job Date: Thu, 24 Jun
Timing: To be confirmed
Job Address: 12 Singapore road, S123456

Figure 2.2



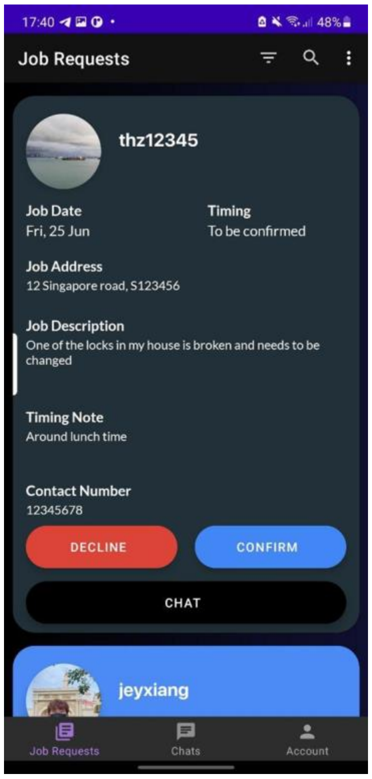
Job Requests

Quicker locksmith
Job Date: Fri, 25 Jun
Timing: To be confirmed
Job Address: 12 Singapore road, S123456
Job Description: One of the locks in my house is broken and needs to be changed
Timing Note: Around lunch time
Contact Number: 97887811

Xiang plumber

Figure 2.3

In the business interface of the app, business users also have a similar function where business users can view job requests sent by users (Figure 2.4) . Upon receiving a job request, a business user can then view its details and then decide to decline or confirm the job request. If they choose to confirm a job request, they can then confirm a timing based on the timing note as seen in figure 2.5. Once the job request has been confirmed, the status of the job request changes. As you can see in Figure 2.6, the colour of the card as well as the action items also changes as well.



thz12345

Job Date
Fri, 25 Jun

Timing
To be confirmed

Job Address
12 Singapore road, S123456

Job Description
One of the locks in my house is broken and needs to be changed

Timing Note
Around lunch time

Contact Number
12345678

jeyxiang

Figure 2.4

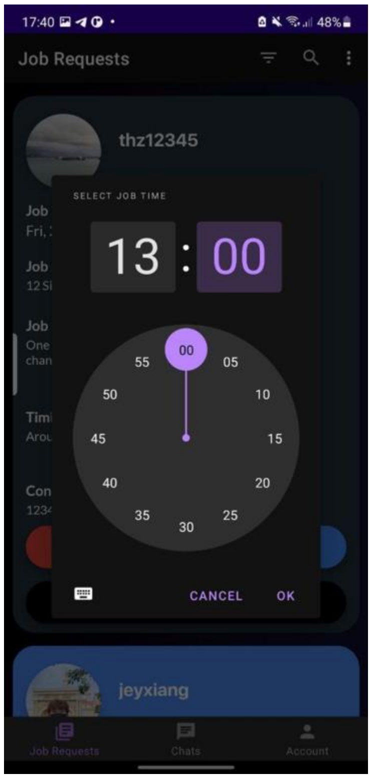
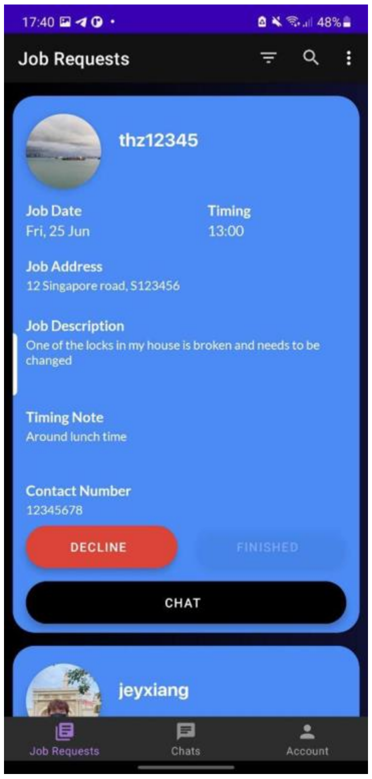


Figure 2.5



thz12345

Job Date
Fri, 25 Jun

Timing
13:00

Job Address
12 Singapore road, S123456

Job Description
One of the locks in my house is broken and needs to be changed

Timing Note
Around lunch time

Contact Number
12345678

jeyxiang

Figure 2.6

Each job request will have a status attached to it and based on the status, the colour of the job request card for both users and business will be different. The status of the job request also affects the action item available to the user. As you can see in figure 1.4 and figure 1.6, after confirming the job request, the action item changes from “Confirm” to “Finished”, which the business can click to mark the job request as finished. In this case, the action item is greyed out because the date of the job request has not passed and thus it cannot be marked as finished yet. The full details of the status and action items can be found in the table below.

Job request status	**Color of card**	**Action for consumer users**	**Action for business users**
Pending	Dark greyish blue (Figure 2.4)	Edit	Confirm
Confirmed	Light blue (Figure 2.6)	Edit (Option will be greyed out if the date of job is within 1 day)	Mark as finished (Option will only be available if the date of job has passed)
Finished	Green	Leave a review	None
Cancelled	Red	None	None

Some additional details about the action items:

- If a request has been confirmed and a user decided to edit it, the status of the request will revert to pending.
- If an option is greyed out and the user clicks on it, a message will be displayed on the screen which explains why the option cannot be chosen. For example, if a business tries to mark a job request as finished before the date of the job, a message "Can only mark request as finished after date of job has passed" will be shown.

When a job request is marked as finished, the user will have the option to leave a review for the business. Since reviews can only be left after a job request has been marked as finished, this lowers the possibility of any fake reviews or trolls in our app.

When users click the review action item they are brought to a page as seen in figure 2.7. They can then give a rating with the rating bar and then write a short comment about the business.

When the user submits their review, it triggers a firebase cloud function which updates the review score of the business and the number of reviews.

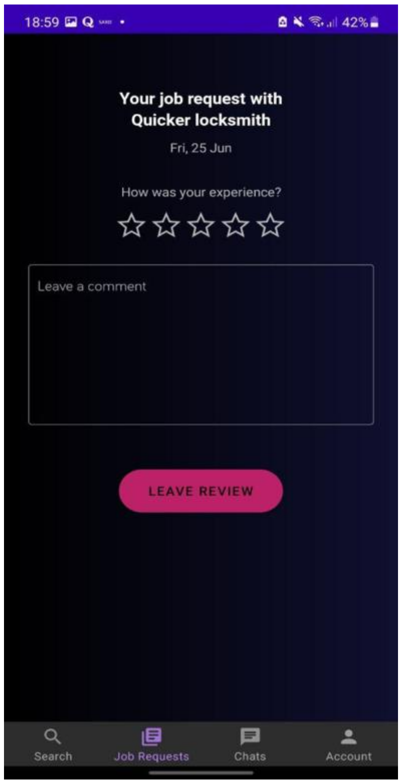
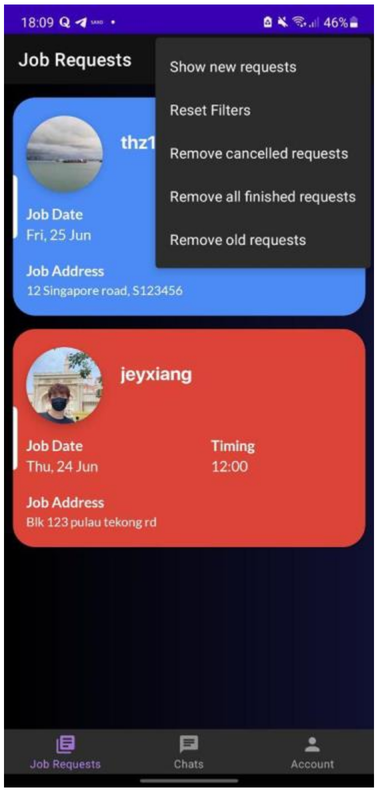


Figure 2.7

The job request of the app also provides users with additional functionalities as seen in figure 2.7. They can delete cancelled requests so that they no longer see it or delete requests that are older than a week. (The requests are not actually deleted from the database. They are just marked as removed so that the user no longer sees it) They can also long click individual requests to delete it individually as seen in figure 2.8.



thz1

Job Date
Fri, 25 Jun

Job Address
12 Singapore road, S123456

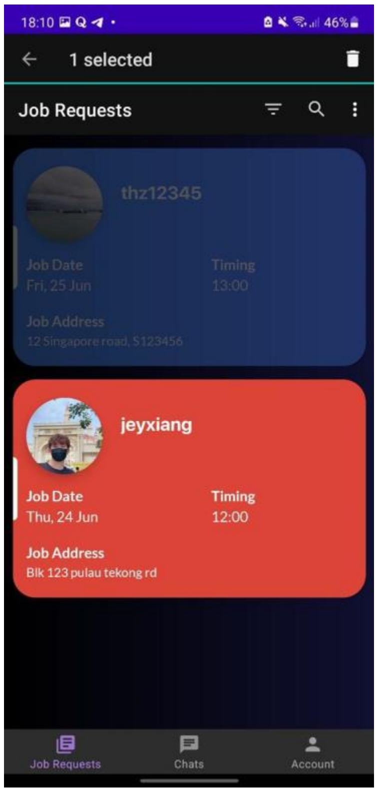
jeyxiang

Job Date
Thu, 24 Jun

Timing
12:00

Job Address
Blk 123 pulau tekong rd

Figure 2.8



thz12345

Job Date
Fri, 25 Jun

Timing
13:00

Job Address
12 Singapore road, S123456

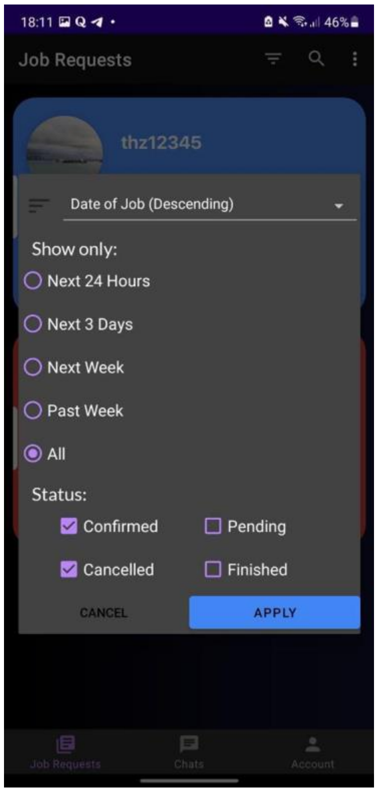
jeyxiang

Job Date
Thu, 24 Jun

Timing
12:00

Job Address
Blk 123 pulau tekong rd

Figure 2.9



Date of Job (Descending)

Show only:

Next 24 Hours

Next 3 Days

Next Week

Past Week

[✓ All]

Status:

[✓ Confirmed] Pending

[✓ Cancelled] Finished

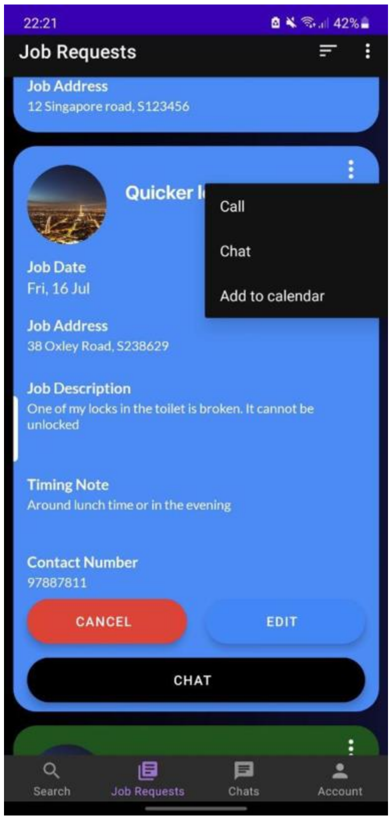
Figure 2.10

As business users may receive many job requests, they also have better filter options as seen in figure 2.9 in order for them to more easily manage their job requests. For instance, they can sort the job requests based on their created date or they can view job requests that are in the next 3 days or only view those whose status is pending. Furthermore, the notification system which will be elaborated down below will allow business users to more easily manage the job requests they received.

As for users, their filter options are more limited. They can only sort the requests by date created or the date of job. The rationale behind this is that users will tend to only make at most a few requests per month and thus will not have many requests to manage. Thus, by simplifying the sort options, we can make a more simplified and easy to use user experience.

For milestone 3, we created an overflow options menu for each job request so as to add more functionality to this feature. The overflow menu contains 3 options as seen in figure 2.11. The first two options are call and chat. Call will call the other party of the job request using the phone's calling app. The chat function will open a new chat with the other party using the app's chat functionality.

The third options differ according to whether the user is a business or consumer user. For consumer users, they have the option to add the job request on their phone's calendar. The user is given a choice of what calendars they want to use if their phone contains different calendar apps (figure 2.12). The details of the job request such as the timing and date will be automatically filled as seen in figure 2.13. For business users, they have the option to open the job's location on google maps in their phone for easy navigation (figure 2.14 and 2.15).



Job Address
12 Singapore road, S123456
Quicker l
Job Date
Fri, 16 Jul
Job Address
38 Oxley Road, S238629
Job Description
One of my locks in the toilet is broken. It cannot be unlocked
Timing Note
Around lunch time or in the evening
Contact Number
97887811

Figure 2.11

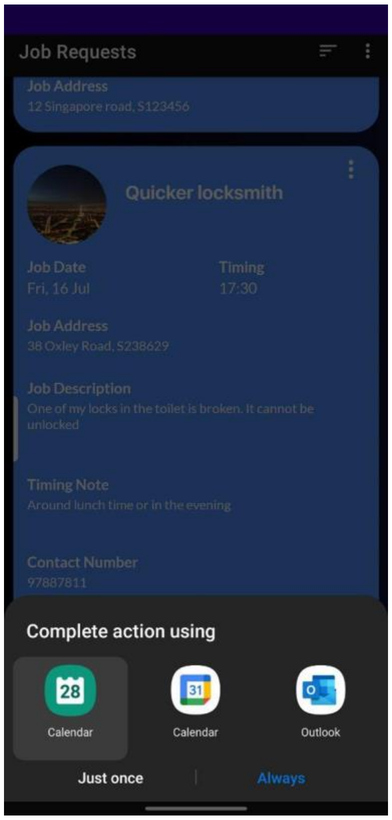
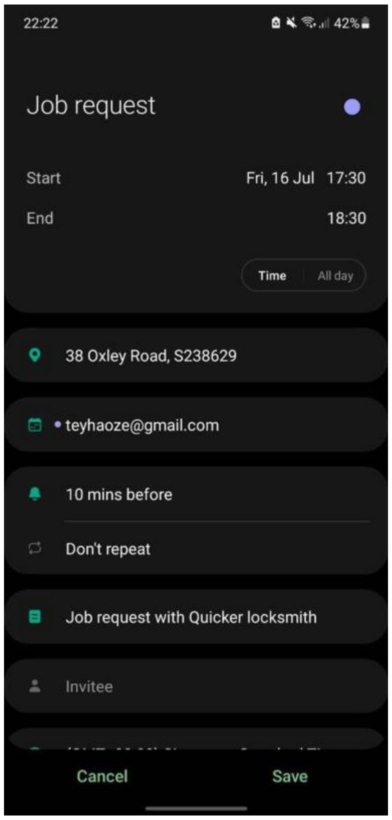
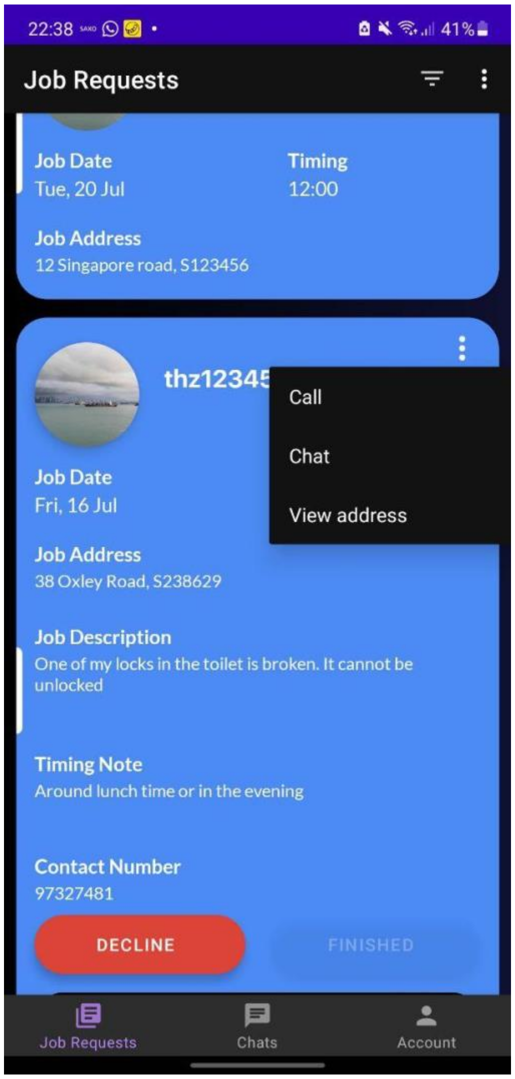


Figure 2.12



Job request
Start Fri, 16 Jul 17:30
End 18:30
Time All day
38 Oxley Road, S238629
teyhaoze@gmail.com
10 mins before
Don't repeat
Job request with Quicker locksmith
Invitee

Figure 2.13



22:38 41%

Job Requests

Job Date
Tue, 20 Jul

Timing
12:00

Job Address
12 Singapore road, S123456

thz12345

Job Date
Fri, 16 Jul

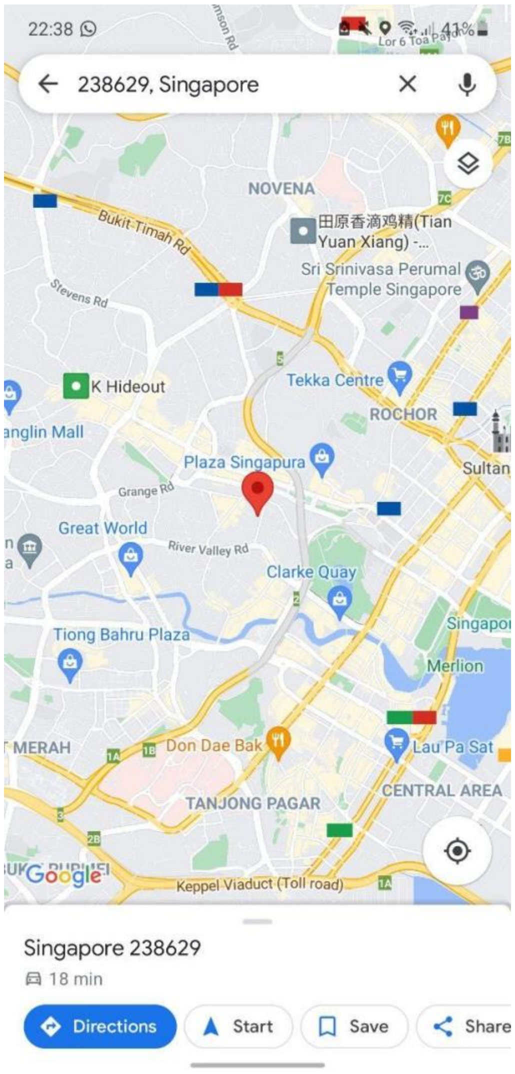
Job Address
38 Oxley Road, S238629

Job Description
One of my locks in the toilet is broken. It cannot be unlocked

Timing Note
Around lunch time or in the evening

Contact Number
97327481

Figure 2.14



22:38 41%

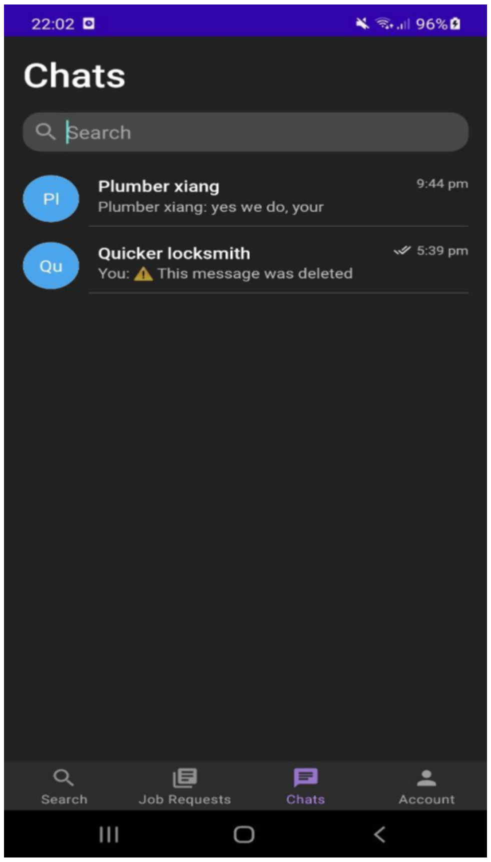
238629, Singapore

Singapore 238629
18 min

Figure 2.15

Chat Page

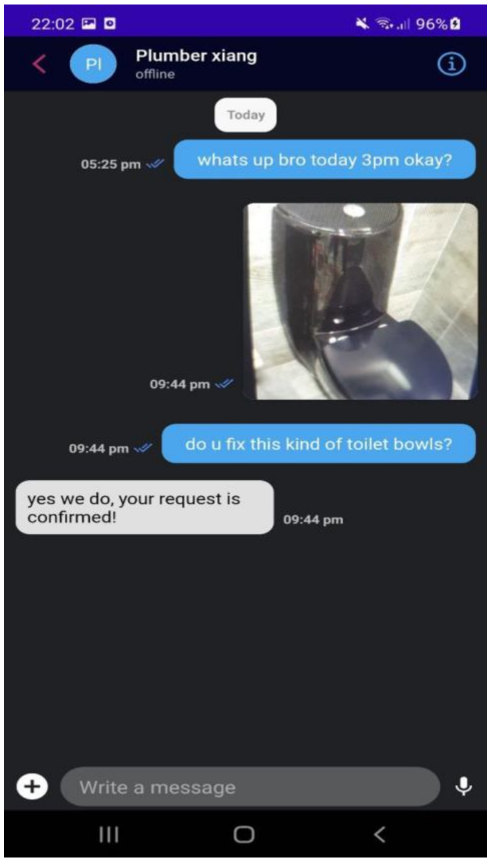
The Chat page is a core feature which allows direct communication between consumers and businesses. The chat feature allows consumers to inquire about prices, availability, type of services provided etc. (See figure 3.2) . The chat function is similar to other chat applications out there, and it includes an online/offline status bar, a double tick to indicate if a message is read or not, the ability to delete messages and also send images and locations. Hence, this feature will enhance user experience when it comes to seeking the service that they yearned for.



Plumber xiang 9:44 pm
Plumber xiang: yes we do, your

Quicker locksmith 5:39 pm
You: ⚠ This message was deleted

Figure 3.1



05:25 pm whats up bro today 3pm okay?

09:44 pm

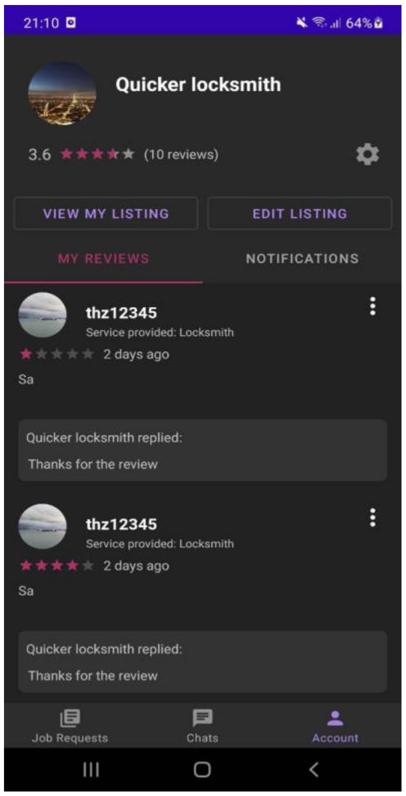
09:44 pm do u fix this kind of toilet bowls?

yes we do, your request is confirmed! 09:44 pm

Figure 3.2

Accounts Page

Both businesses and consumers will have an account page with a slight difference. Businesses can view the reviews that they received from customers that they served, as well as the replies that they give (Figure 4.1). A notification bar also allows them to view any customers' request (Figure 4.2), and upon clicking it will direct them to a request page (Figure 1.4) . An additional feature that exists on the business account interface is the "view my listings option" , which will redirect the user to a listing page as shown on Figure 4.3 . This will allow businesses to check if the information displayed on their listing is accurate and up to date. Another option "Edit listings" will redirect them to the Business listings page (Figure 6.1)


21:10 64%

Quicker locksmith

3.6 ★★★★★ (10 reviews)

VIEW MY LISTING | EDIT LISTING

MY REVIEWS | NOTIFICATIONS

thz12345
Service provided: Locksmith
★★★★★ 2 days ago
Sa

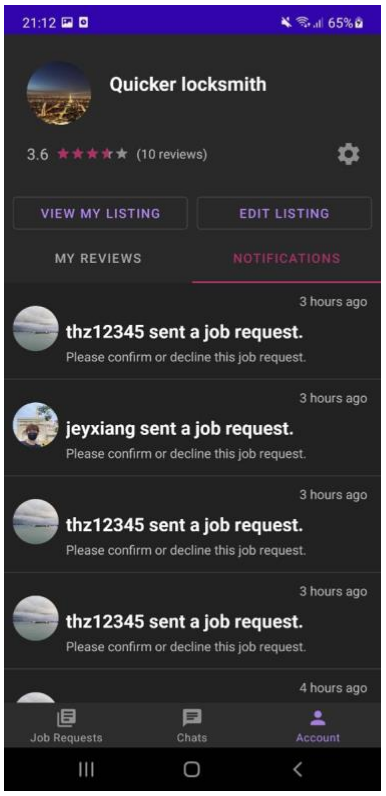
Quicker locksmith replied:
Thanks for the review

thz12345
Service provided: Locksmith
★★★★★ 2 days ago
Sa

Quicker locksmith replied:
Thanks for the review

Job Requests | Chats | Account

Figure 4.1


21:12 65%

Quicker locksmith

3.6 ★★★★★ (10 reviews)

VIEW MY LISTING | EDIT LISTING

MY REVIEWS | NOTIFICATIONS

3 hours ago
thz12345 sent a job request.
Please confirm or decline this job request.

3 hours ago
jeyxiang sent a job request.
Please confirm or decline this job request.

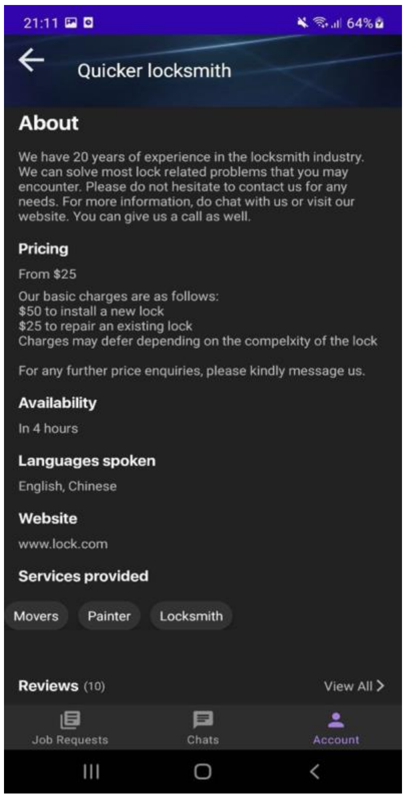
3 hours ago
thz12345 sent a job request.
Please confirm or decline this job request.

3 hours ago
thz12345 sent a job request.
Please confirm or decline this job request.

4 hours ago

Job Requests | Chats | Account

Figure 4.2


21:11 64%

Quicker locksmith

About

We have 20 years of experience in the locksmith industry. We can solve most lock related problems that you may encounter. Please do not hesitate to contact us for any needs. For more information, do chat with us or visit our website. You can give us a call as well.

Pricing

From $25

Our basic charges are as follows:
$50 to install a new lock
$25 to repair an existing lock
Charges may defer depending on the compelxity of the lock

For any further price enquiries, please kindly message us.

Availability

In 4 hours

Languages spoken

English, Chinese

Website

www.lock.com

Services provided

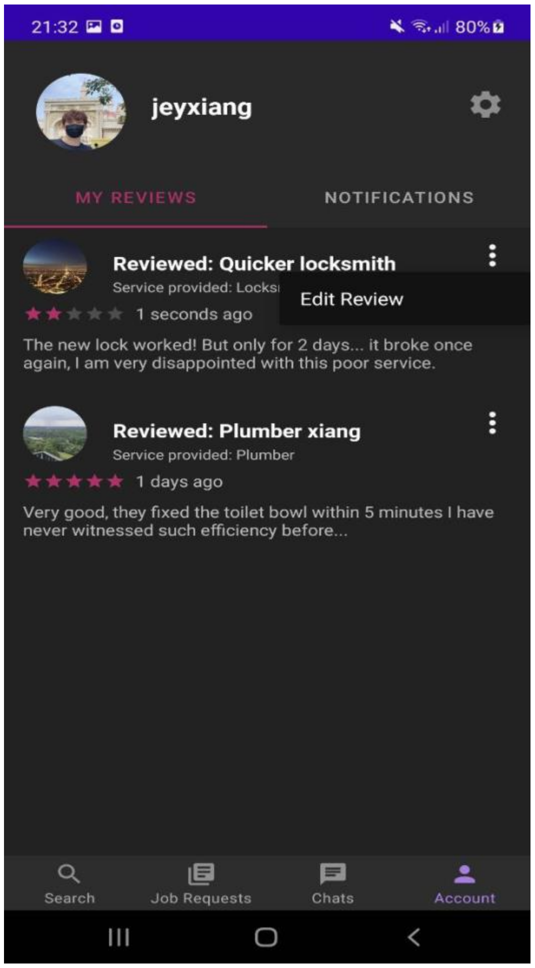
Movers | Painter | Locksmith

Reviews (10) View All >

Job Requests | Chats | Account

Figure 4.3

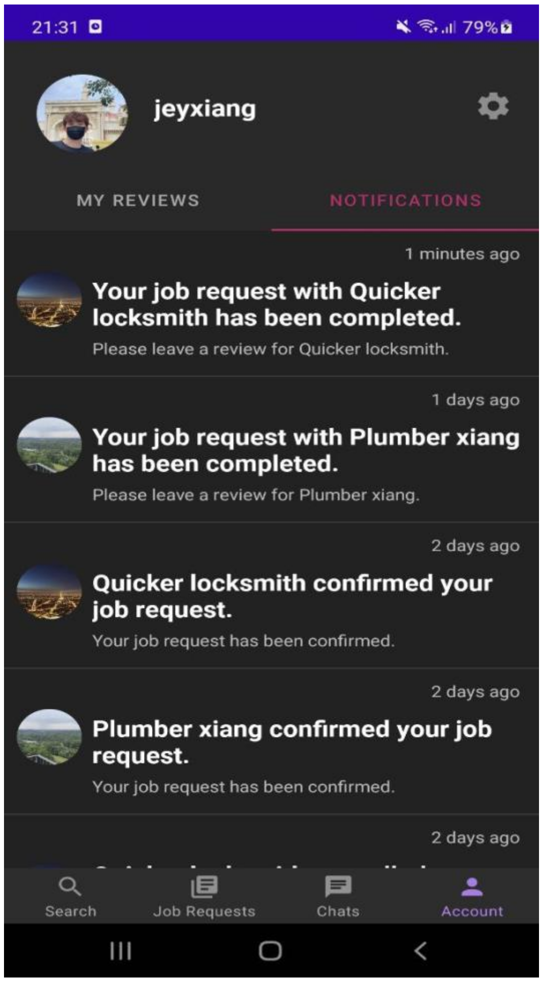
As for the consumers, their account page consists of only the review and notification bars. Users can see their past reviews and edit their reviews accordingly. (Figure 4.1) The notification panel will notify them if their job request is accepted or rejected. (Figure 4.2) . The setting button on the top right of the account page will redirect the user to the settings page (See next page).



Reviewed: Quicker locksmith
Service provided: Locksı
★★★★★ 1 seconds ago
The new lock worked! But only for 2 days... it broke once again, I am very disappointed with this poor service.

Reviewed: Plumber xiang
Service provided: Plumber
★★★★★ 1 days ago
Very good, they fixed the toilet bowl within 5 minutes I have never witnessed such efficiency before...

Figure 4.4



1 minutes ago
Your job request with Quicker locksmith has been completed.
Please leave a review for Quicker locksmith.

1 days ago
Your job request with Plumber xiang has been completed.
Please leave a review for Plumber xiang.

2 days ago
Quicker locksmith confirmed your job request.
Your job request has been confirmed.

2 days ago
Plumber xiang confirmed your job request.
Your job request has been confirmed.

2 days ago

Figure 4.5

Settings Page

Just like any other applications out there, the Settings page (Figure 5.1) enables both businesses and users to edit their account details, password, contact administrators as well as to write in complaints. Users can also set their profile pictures accordingly. The log out button will also be placed on this page.

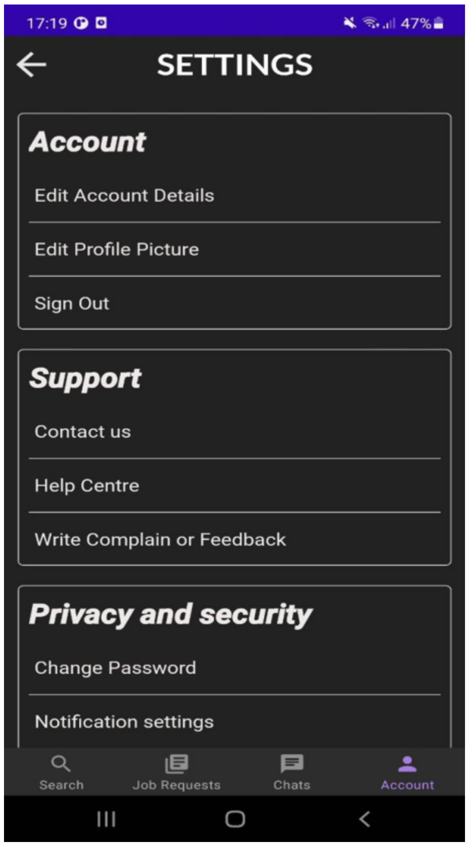


Figure 5.1

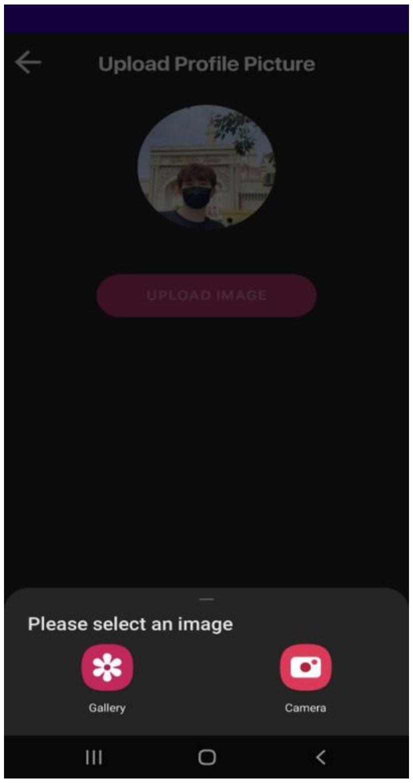
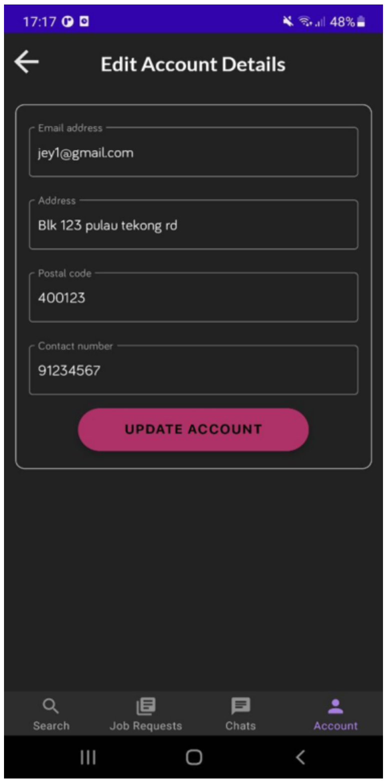


Figure 5.2



Email address: jey1@gmail.com

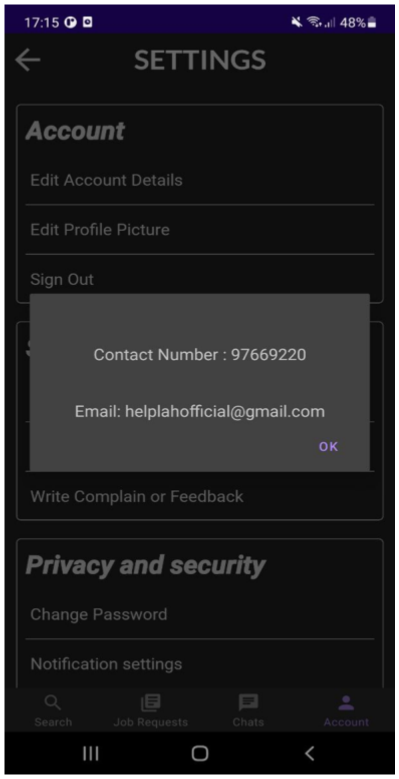
Address: Blk 123 pulau tekong rd

Postal code: 400123

Contact number: 91234567

UPDATE ACCOUNT

Figure 5.3



Contact Number : 97669220

Email: helplahofficial@gmail.com

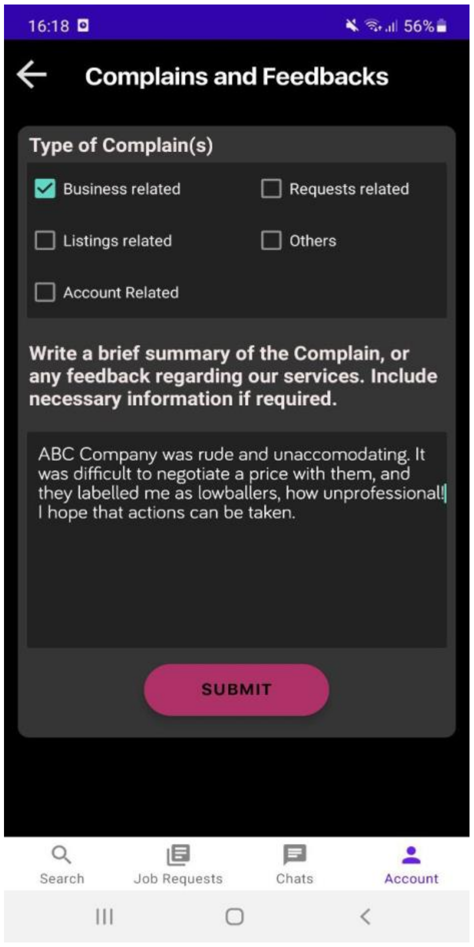
OK

Figure 5.4

As shown in Figure 5.2, users can upload their profile picture from their gallery or their camera, and the photo will be saved in firebase storage and reflected in the application. Before selecting any image, the upload image will be greyed and disabled. This is to prevent users from pressing the upload button without selecting any image.

Users can edit their Email address, address, and contact details as shown in Figure 5.3. Account details will be updated into the firestore database, and upon successfully updating, a message will be displayed to tell the users if the update was successful. Before the Email can be changed, a check will take place to see if the email is valid and usable.

Upon pressing the “contact us” option, an alert dialog will pop out (Figure 5.4) , displaying the Contact number and Email of the administrators.

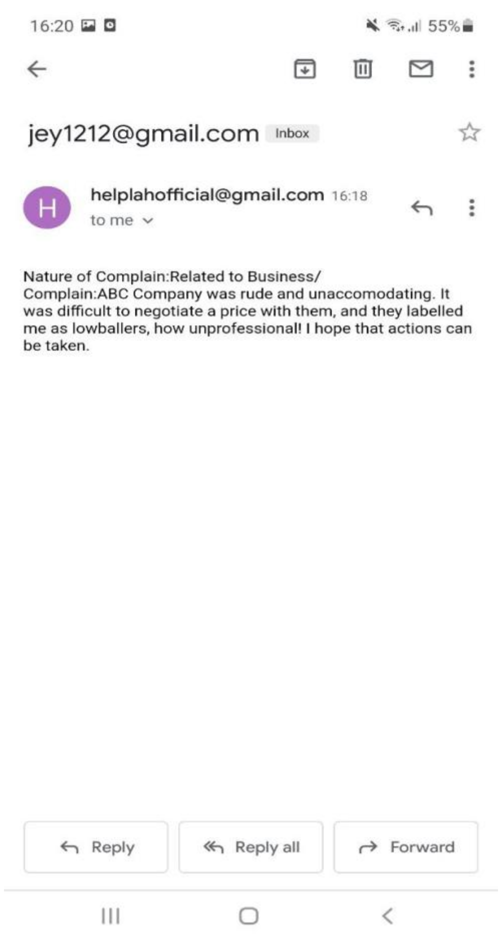


[✓ Business related] Requests related
Listings related Others
Account Related

Write a brief summary of the Complain, or any feedback regarding our services. Include necessary information if required.

ABC Company was rude and unaccomodating. It was difficult to negotiate a price with them, and they labelled me as lowballers, how unprofessional! I hope that actions can be taken.

Figure 5.5



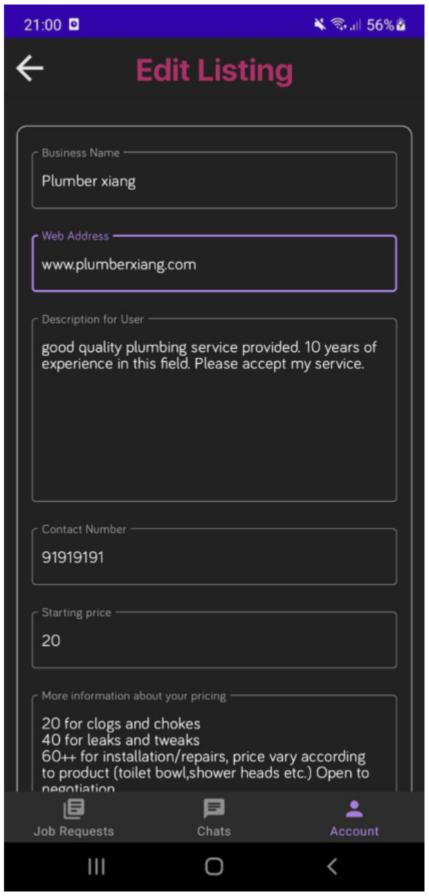
Nature of Complain:Related to Business/
Complain:ABC Company was rude and unaccomodating. It was difficult to negotiate a price with them, and they labelled me as lowballers, how unprofessional! I hope that actions can be taken.

Figure 5.6

Users can also write in complaints and select their nature of complaint, as shown in Figure 5.5. Upon submitting, any complaints/feedback will be directed to our administrative email, and the user's email will be taken down as the subject for us to contact them for further enquiries. This allows us to actively keep track of any complaints and take the necessary actions. The email will be formatted as shown in Figure 5.6.

Business Listing Page

This feature, which can only be found under the business interface, allows businesses to edit their listings. For instance, they can change their website address, their business description, and adjust their starting prices accordingly. A pricing note also allows businesses to customise their pricings to their preference (Figure 6.1) . They can also customise their availability dates, set their preferred language and select their choice of service that they want to provide (Figure 6.2). Once they have updated their changes, the changes will be reflected in the database.



Business Name: Plumber xiang

Web Address: www.plumberxiang.com

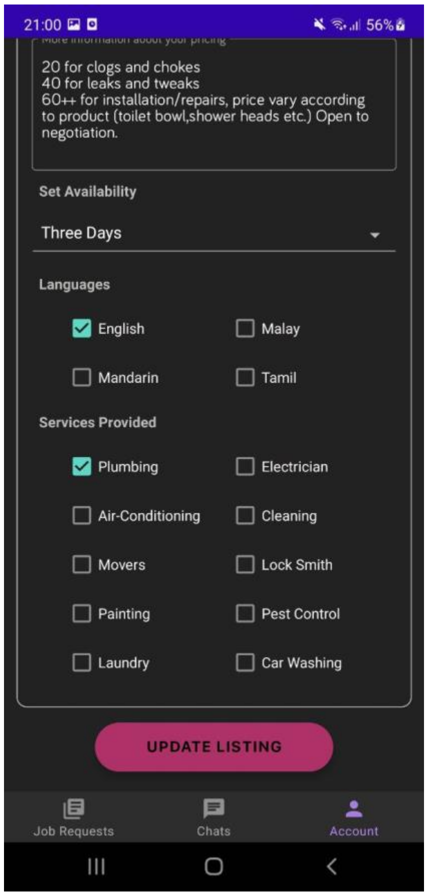
Description for User: good quality plumbing service provided. 10 years of experience in this field. Please accept my service.

Contact Number: 91919191

Starting price: 20

More information about your pricing: 20 for clogs and chokes
40 for leaks and tweaks
60++ for installation/repairs, price vary according to product (toilet bowl,shower heads etc.) Open to negotiation

Figure 6.1



More information about your pricing: 20 for clogs and chokes
40 for leaks and tweaks
60++ for installation/repairs, price vary according to product (toilet bowl,shower heads etc.) Open to negotiation.

Set Availability: Three Days

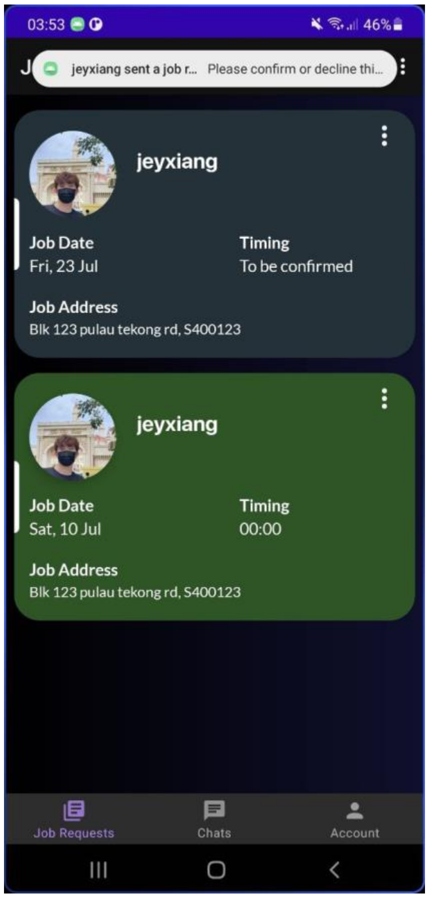
Languages: [✓ English] Malay Mandarin Tamil

Services Provided: [✓ Plumbing] Electrician Air-Conditioning Cleaning Movers Lock Smith Painting Pest Control Laundry Car Washing

Figure 6.2

Push Notification

Push Notification is an additional feature to enhance user experience. We will be using Firebase Cloud Messaging for our push notifications. From figure 7.1, businesses can receive notifications when they receive job requests. They will also receive notifications when users cancel requests, or leave reviews. Likewise, users can also receive notifications if businesses accept their job requests or cancel them (Figure 7.2). On clicking of the notification, both users and business will be redirected to their respective notification page (Figure 4.5). In addition, users can also receive chat notifications, as seen in figure 7.3, and upon clicking of the notification, they will be redirected to the chat page. (Figure 3.1)


03:53 46%

jeyxiang sent a job r... Please confirm or decline thi...

jeyxiang

Job Date Fri, 23 Jul

Timing To be confirmed

Job Address Blk 123 pulau tekong rd, S400123

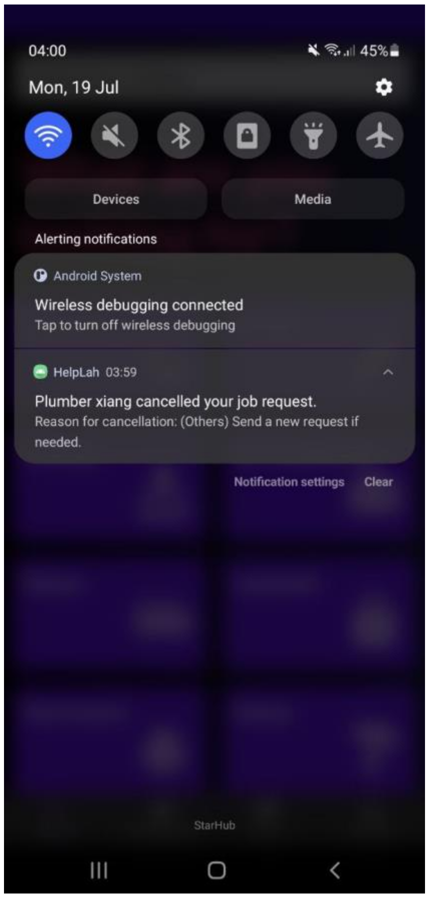
jeyxiang

Job Date Sat, 10 Jul

Timing 00:00

Job Address Blk 123 pulau tekong rd, S400123

Figure 7.1


04:00 45%

Mon, 19 Jul

Alerting notifications

Android System

Wireless debugging connected

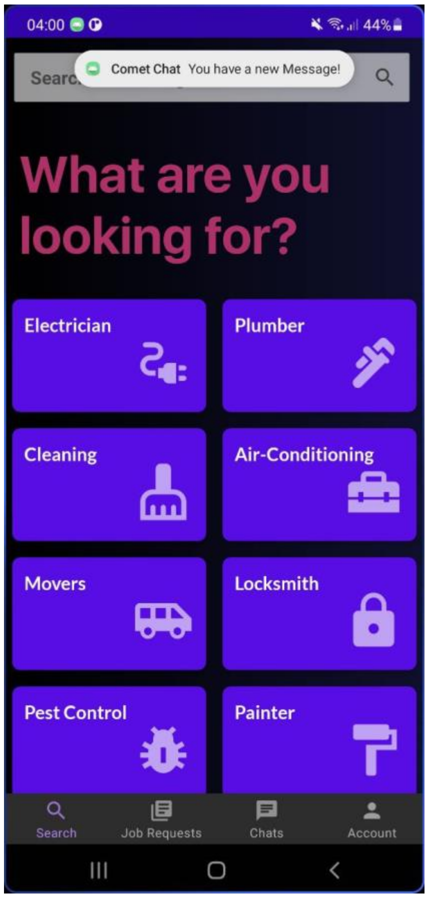
Tap to turn off wireless debugging

HelpLah 03:59

Plumber xiang cancelled your job request.

Reason for cancellation: (Others) Send a new request if needed.

Figure 7.2


04:00 44%

Comet Chat You have a new Message!

Figure 7.3

Password Recovery

A “Forgot password” option can be found on the main login page (Figure 8.1). Upon clicking that option, users will be directed to the page as shown on figure 8.2. Users can enter their email registered with their account and retrieve an authentication code. Afterwards, an email will be sent to that email if it can be found in the database. (Figure 8.3)

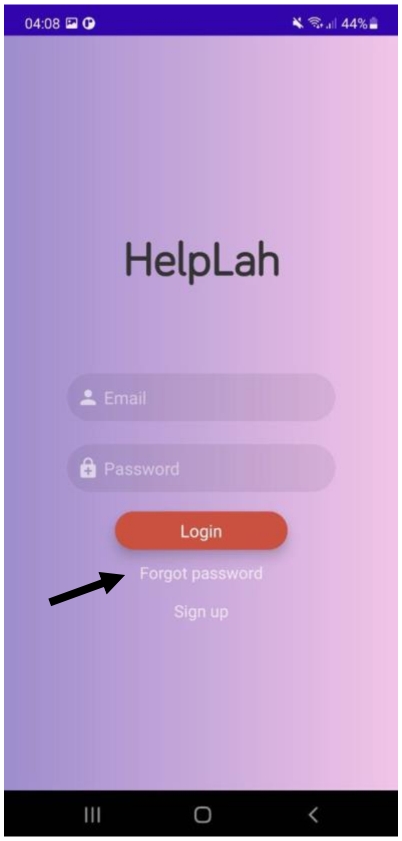
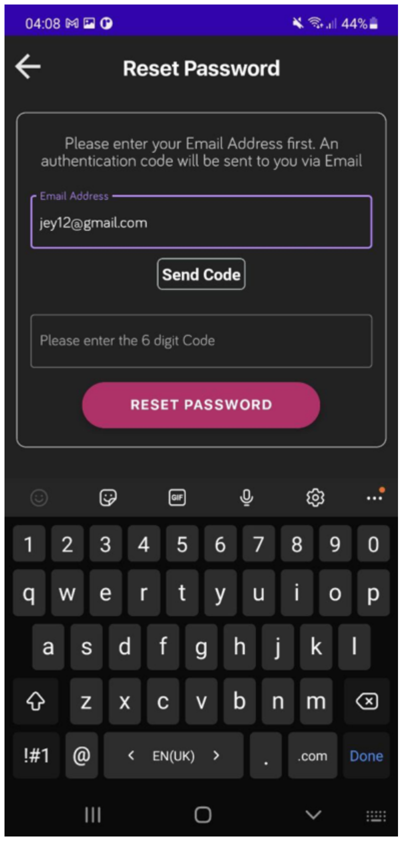


Figure 8.1

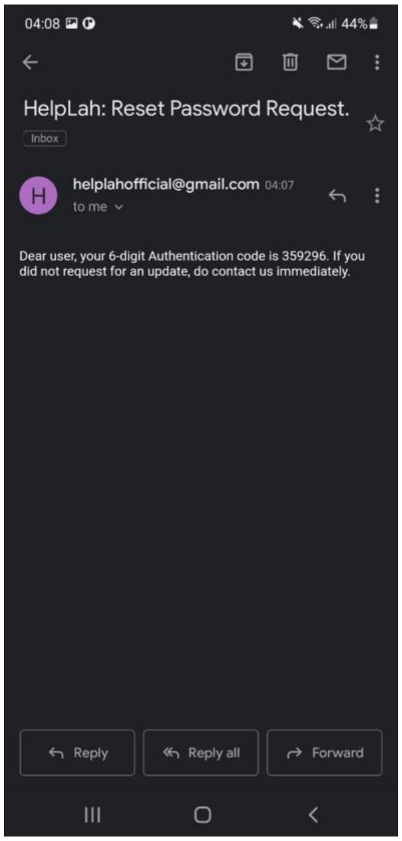


Please enter your Email Address first. An authentication code will be sent to you via Email

Email Address: jey12@gmail.com

Please enter the 6 digit Code

Figure 8.2



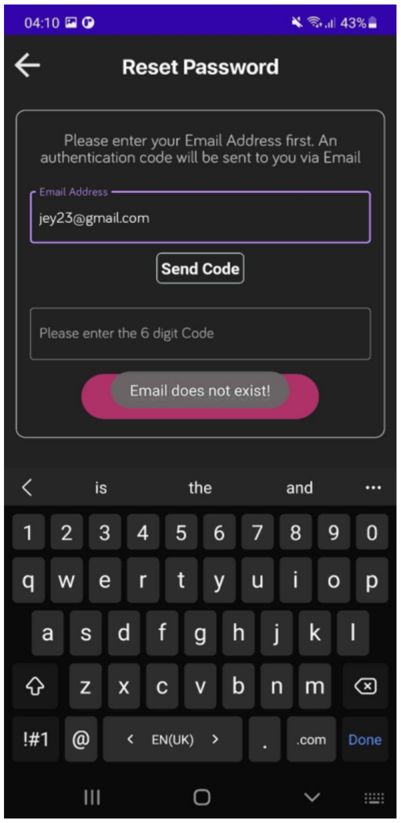
HelpLah: Reset Password Request.

helplahofficial@gmail.com 04:07
to me

Dear user, your 6-digit Authentication code is 359296. If you did not request for an update, do contact us immediately.

Figure 8.3

A wrong email will trigger a message as shown on Figure 8.4. After a correct code has been entered (Figure 8.5) , the user will receive another email which will provide them a link to change their password. (Figure 8.6)



04:10 43%

Reset Password

Please enter your Email Address first. An authentication code will be sent to you via Email

Email Address

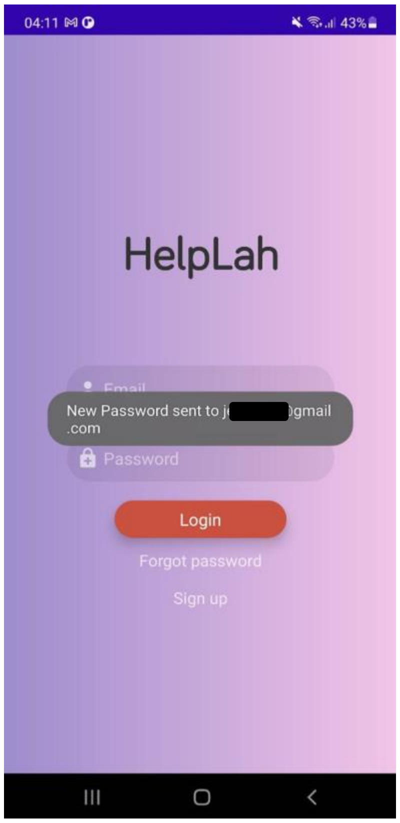
jey23@gmail.com

Send Code

Please enter the 6 digit Code

Email does not exist!

Figure 8.4



04:11 43%

HelpLah

Email

New Password sent to j[redacted]gmail .com

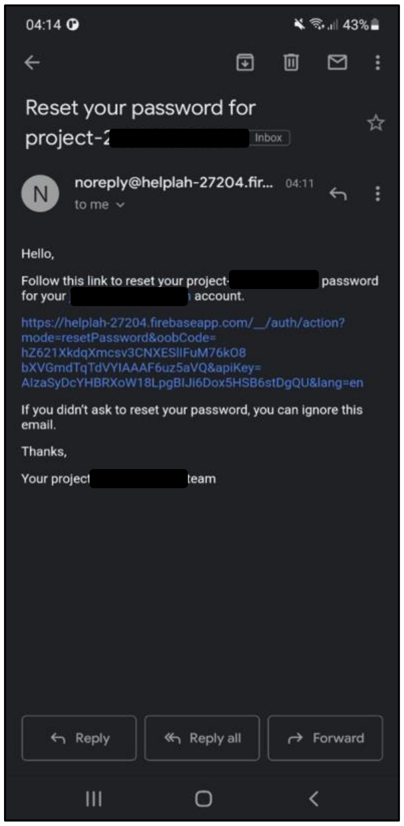
Password

Login

Forgot password

Sign up

Figure 8.5



04:14 43%

Reset your password for project-2[redacted] Inbox

noreply@helplah-27204.fir... 04:11
to me

Hello,

Follow this link to reset your project-[redacted] password for your [redacted] account.

https://helplah-27204.firebaseapp.com/__/auth/action?mode=resetPassword&oobCode=hZ621XkdqXmcsv3CNXESlIFuM76kO8bXVGmdTqTdVYIAAAF6uz5aVQ&apiKey=AIzaSyDcYHBRXoW18LpgBIJi6Dox5HSB6stDgQU&lang=en

If you didn't ask to reset your password, you can ignore this email.

Thanks,

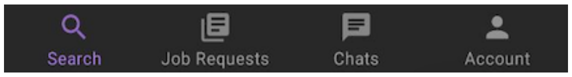
Your project[redacted]team

Figure 8.6

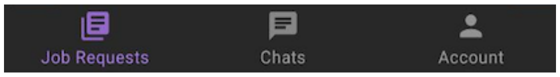
How the app works

Frontend
Since our app will be used by 2 different groups of users, customers and businesses, we decided to create two different interfaces to serve both consumers and businesses separately in the same app. Both Consumers and Businesses will share different flows of the program based on their different needs. When a user logs in, the app will check if it is a customer and business and then redirect the user to the correct home page accordingly.

The consumer user and business user interface share a common general design in that they are both controlled by a bottom navigation bar that takes them to the different components of the app. However, there is some difference between the 2 navigation bars. For users, the navigation bar consists of 4 components (search page, job requests, chat and account), whereas the business users consists of only 3 with the search page left out.



Navigation bar for customer users



Navigation bar for business users

We will be using a single activity architecture for our application using Google's Navigation Component and following Google's Material design for our theme. This means that the business and customer interface will each only have one main activity, which hosts the different fragments that contain all the UI related items. Navigation between the different fragments is handled by the Navigation Component which uses each main activity as a host for navigation and swaps individual fragments into that host as users navigate through the app.

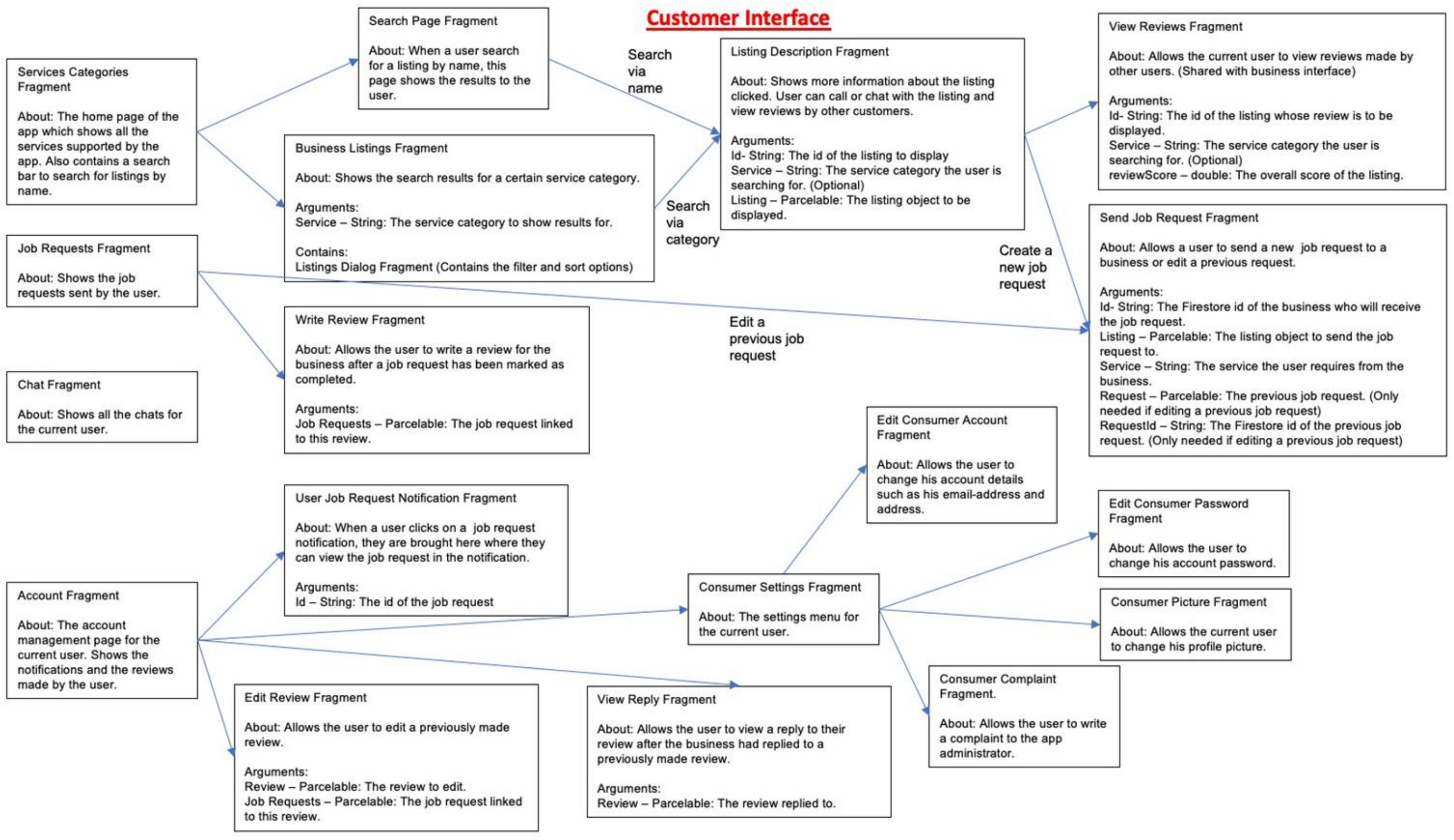
The Navigation Component helps us to simplify the navigation flow in our application. Instead of having to manage the navigation logic and lifecycle of each fragment or activity individually, all of these are handled by the Navigation Component. This allows us to very easily modify the navigation flow of the application using the navigation graph. Furthermore, such a design choice also provides more seamless transitions and less loading time between different UI layouts.

The Navigation Component also allows us to make the bottom navigation bar more useful. The Navigation Component maintains a back stack of fragments visited. This means that the user can switch between the different tabs in the bottom navigation bar and come back to where they left off. For example, if a business user is looking at his job requests on the job request tab and receives a message, he can change to the chats tab to view the new message and then return to the job request tab which will retain the same configuration and layout as when he left without having to recreate the fragment. Without using the Navigation Component, implementing such a feature would require a lot more code and make the codebase harder to maintain.

We have inserted 2 diagrams below to show the navigation flow for both the consumer and business interface of our app. The diagrams will allow future developers to more easily modify the navigational flow of our app. What each fragment does and the arguments needed to initialize it is listed in the flowchart. More details on each fragment can be found in the Javadoc comment in the codebase.

To have a clearer view of these flow charts, you can click the 2 links below: (Will be inserted later)

- Customer user interface: Customer navigation interface.jpeg
- Business user interface: Business navigation interface.jpeg



Services Categories Fragment

About: The home page of the app which shows all the services supported by the app. Also contains a search bar to search for listings by name.

Search Page Fragment

About: When a user search for a listing by name, this page shows the results to the user.

Business Listings Fragment

About: Shows the search results for a certain service category.

Arguments:
Service – String: The service category to show results for.

Contains:
Listings Dialog Fragment (Contains the filter and sort options)

Listing Description Fragment

About: Shows more information about the listing clicked. User can call or chat with the listing and view reviews by other customers.

Arguments:
Id- String: The id of the listing to display
Service – String: The service category the user is searching for. (Optional)
Listing – Parcelable: The listing object to be displayed.

View Reviews Fragment

About: Allows the current user to view reviews made by other users. (Shared with business interface)

Arguments:
Id- String: The id of the listing whose review is to be displayed.
Service – String: The service category the user is searching for. (Optional)
reviewScore – double: The overall score of the listing.

Send Job Request Fragment

About: Allows a user to send a new job request to a business or edit a previous request.

Arguments:
Id- String: The Firestore id of the business who will receive the job request.
Listing – Parcelable: The listing object to send the job request to.
Service – String: The service the user requires from the business.
Request – Parcelable: The previous job request. (Only needed if editing a previous job request)
RequestId – String: The Firestore id of the previous job request. (Only needed if editing a previous job request)

Job Requests Fragment

About: Shows the job requests sent by the user.

Write Review Fragment

About: Allows the user to write a review for the business after a job request has been marked as completed.

Arguments:
Job Requests – Parcelable: The job request linked to this review.

Chat Fragment

About: Shows all the chats for the current user.

Edit Consumer Account Fragment

About: Allows the user to change his account details such as his email-address and address.

User Job Request Notification Fragment

About: When a user clicks on a job request notification, they are brought here where they can view the job request in the notification.

Arguments:
Id – String: The id of the job request

Edit Consumer Password Fragment

About: Allows the user to change his account password.

Account Fragment

About: The account management page for the current user. Shows the notifications and the reviews made by the user.

Consumer Settings Fragment

About: The settings menu for the current user.

Consumer Picture Fragment

About: Allows the current user to change his profile picture.

Consumer Complaint Fragment.

About: Allows the user to write a complaint to the app administrator.

Edit Review Fragment

About: Allows the user to edit a previously made review.

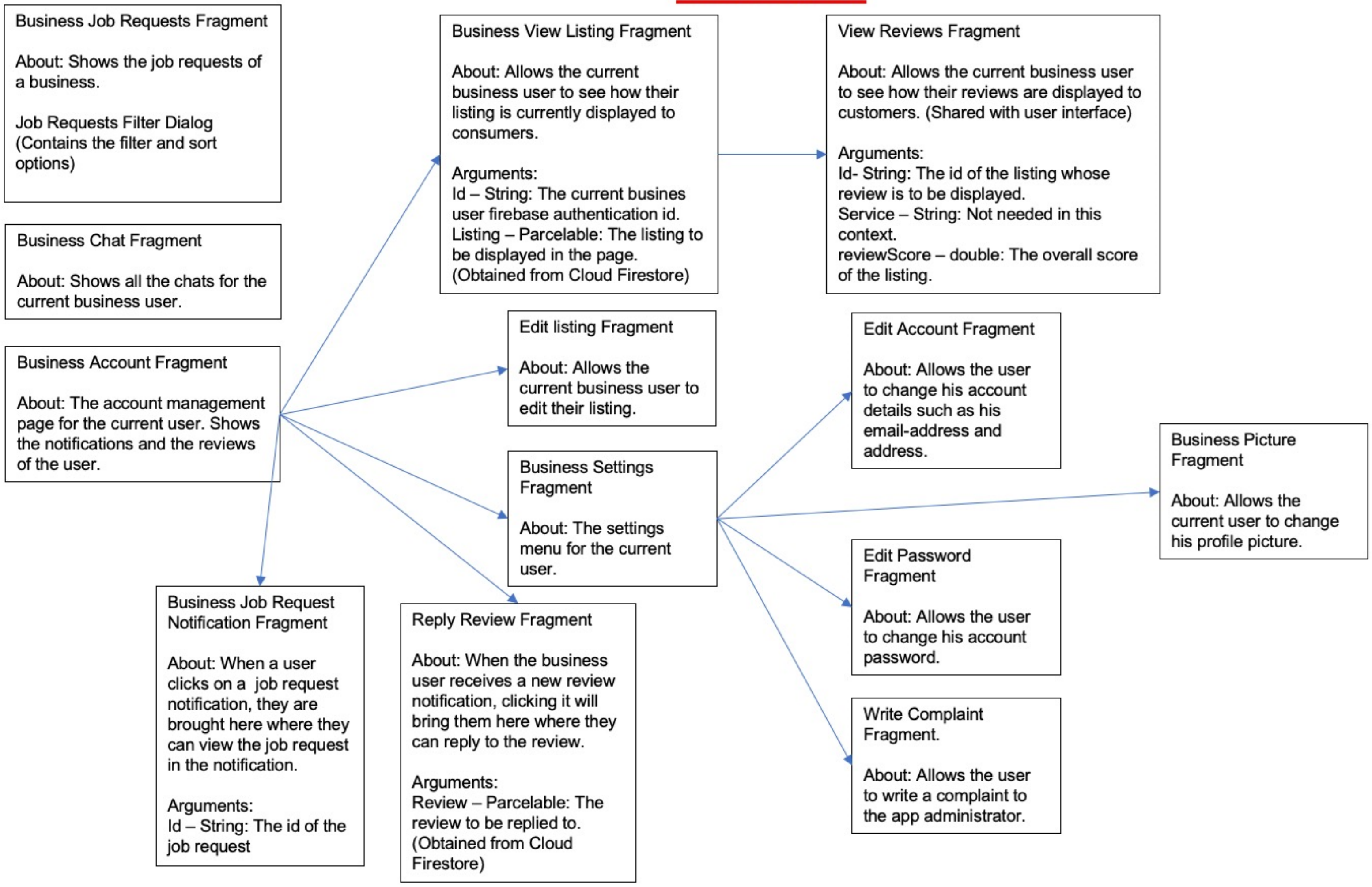
Arguments:
Review – Parcelable: The review to edit.
Job Requests – Parcelable: The job request linked to this review.

View Reply Fragment

About: Allows the user to view a reply to their review after the business had replied to a previously made review.

Arguments:
Review – Parcelable: The review replied to.

Business Interface



Business Job Requests Fragment

About: Shows the job requests of a business.

Job Requests Filter Dialog (Contains the filter and sort options)

Business Chat Fragment

About: Shows all the chats for the current business user.

Business Account Fragment

About: The account management page for the current user. Shows the notifications and the reviews of the user.

Business View Listing Fragment

About: Allows the current business user to see how their listing is currently displayed to consumers.

Arguments:
Id – String: The current busines user firebase authentication id.
Listing – Parcelable: The listing to be displayed in the page. (Obtained from Cloud Firestore)

View Reviews Fragment

About: Allows the current business user to see how their reviews are displayed to customers. (Shared with user interface)

Arguments:
Id- String: The id of the listing whose review is to be displayed.
Service – String: Not needed in this context.
reviewScore – double: The overall score of the listing.

Edit listing Fragment

About: Allows the current business user to edit their listing.

Edit Account Fragment

About: Allows the user to change his account details such as his email-address and address.

Business Settings Fragment

About: The settings menu for the current user.

Business Picture Fragment

About: Allows the current user to change his profile picture.

Edit Password Fragment

About: Allows the user to change his account password.

Write Complaint Fragment.

About: Allows the user to write a complaint to the app administrator.

Business Job Request Notification Fragment

About: When a user clicks on a job request notification, they are brought here where they can view the job request in the notification.

Arguments:
Id – String: The id of the job request

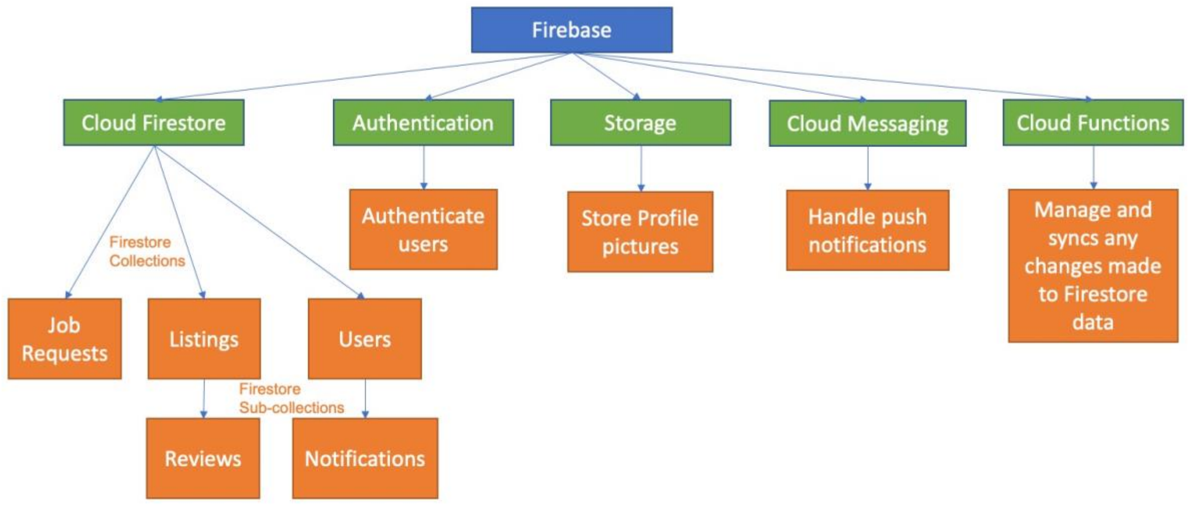
Reply Review Fragment

About: When the business user receives a new review notification, clicking it will bring them here where they can reply to the review.

Arguments:
Review – Parcelable: The review to be replied to. (Obtained from Cloud Firestore)

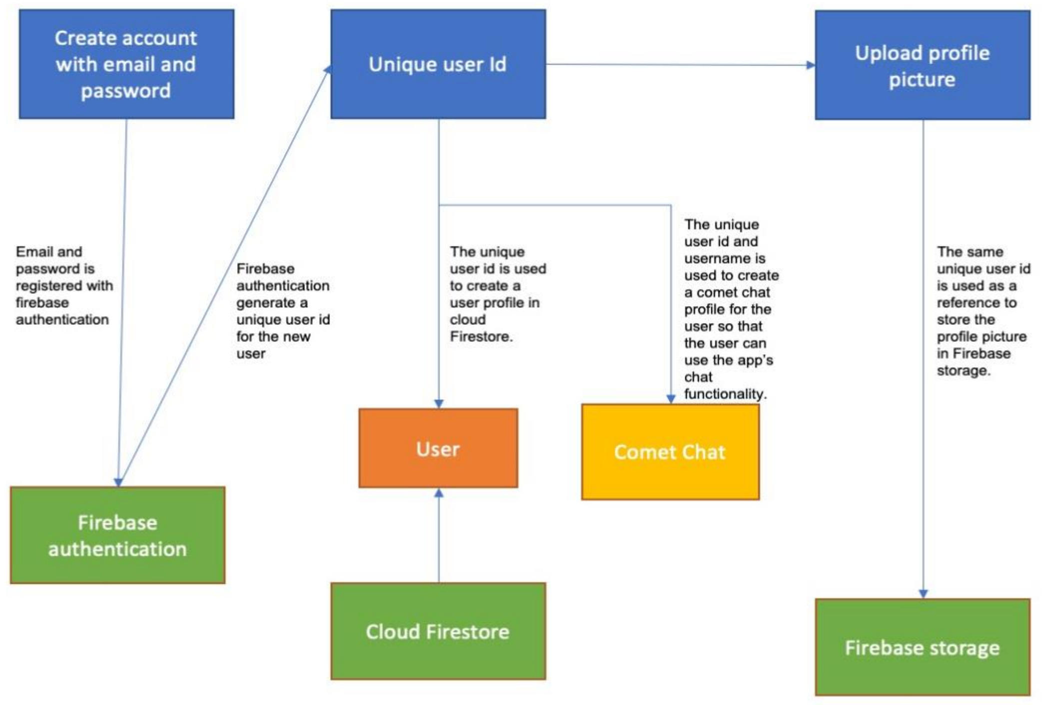
Backend

For our backend, we will be using Google Firebase and CometChat which will handle our chat features. The diagram below shows all the Firebase features that we will be using as well as all the collections and sub-collections used in Cloud Firestore. For our char function, we will be using the CometChat API.

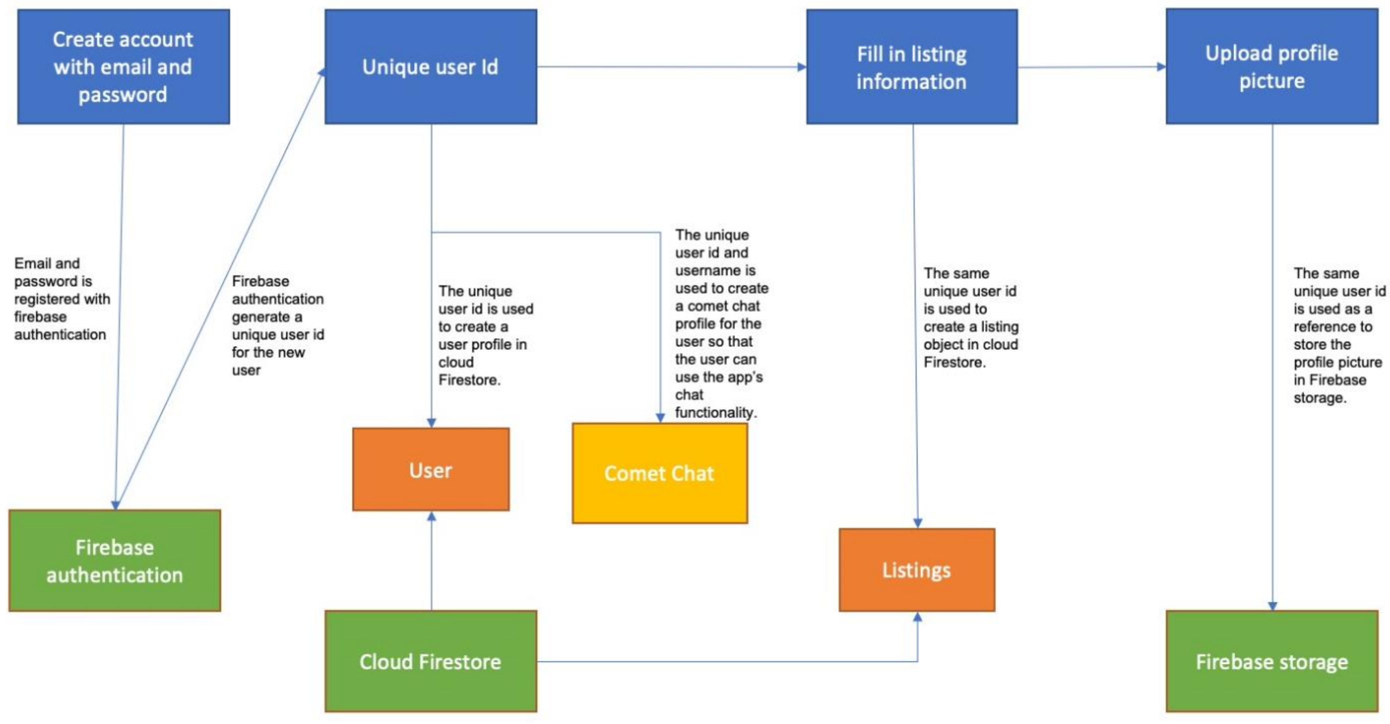


The 2 diagram below shows the process when a user signs up and how it relates to our backend.

Customer User Sign-up Process

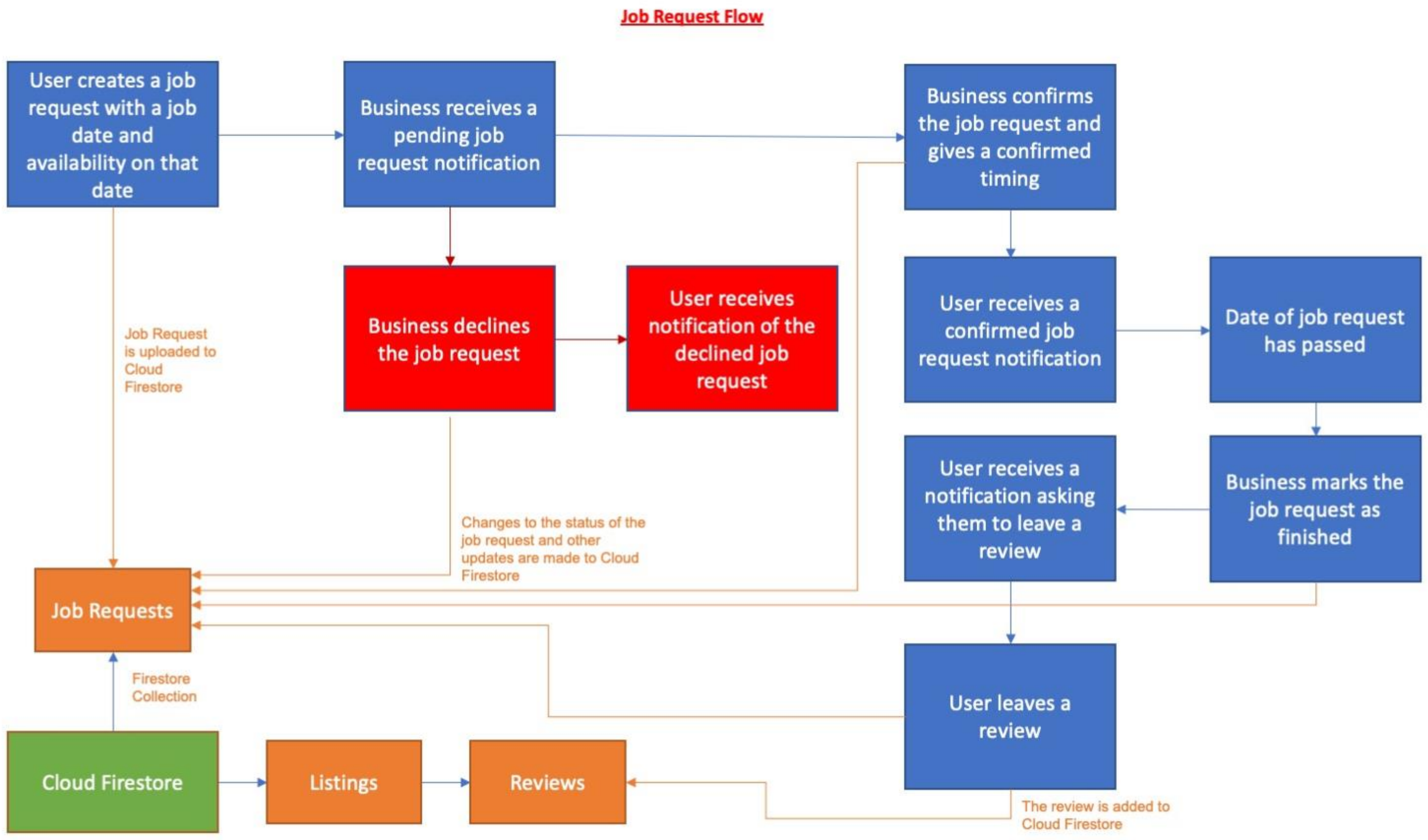

Create account with email and password
Unique user Id
Upload profile picture
Email and password is registered with firebase authentication
Firebase authentication generate a unique user id for the new user
The unique user id is used to create a user profile in cloud Firestore.
The unique user id and username is used to create a comet chat profile for the user so that the user can use the app's chat functionality.
The same unique user id is used as a reference to store the profile picture in Firebase storage.
User
Comet Chat
Firebase authentication
Cloud Firestore
Firebase storage

Business User Sign-up Process


Create account with email and password
Unique user Id
Fill in listing information
Upload profile picture
Email and password is registered with firebase authentication
Firebase authentication generate a unique user id for the new user
The unique user id is used to create a user profile in cloud Firestore.
The unique user id and username is used to create a comet chat profile for the user so that the user can use the app's chat functionality.
The same unique user id is used to create a listing object in cloud Firestore.
The same unique user id is used as a reference to store the profile picture in Firebase storage.
User
Comet Chat
Listings
Firebase authentication
Cloud Firestore
Firebase storage

We have also inserted another flowchart to show how our job request function works and how the information is updated in our backend.

This link contains a clearer view on the flowchart: Job request flow.jpeg


Job Request Flow
User creates a job request with a job date and availability on that date
Business receives a pending job request notification
Business confirms the job request and gives a confirmed timing
Business declines the job request
User receives notification of the declined job request
User receives a confirmed job request notification
Date of job request has passed
Job Request is uploaded to Cloud Firestore
User receives a notification asking them to leave a review
Business marks the job request as finished
Changes to the status of the job request and other updates are made to Cloud Firestore
Job Requests
Firestore Collection
User leaves a review
Cloud Firestore
Listings
Reviews
The review is added to Cloud Firestore

<u>Difficulties faced</u>

Chat app

One of the major difficulties we faced in developing our application was building the chat functionality. Initially, we wanted to build a chat function from scratch using the Firebase Firestore database as our backend. We managed to find some examples on the internet and thought it was feasible and thus we wanted to try it out. This approach initially presented a major problem which is that Firestore charges per document read. Thus, simply storing the user messages in individual documents and loading all of them at once when a user enters a chat is not feasible. Not only would it consume a lot of the phone’s resources, it would also incur a lot of document reads especially if a user has sent many messages in the past.

The way to overcome this was to use pagination which means that when a user opens a chat, only the initial few messages will be loaded. The rest of the messages will only be loaded when the user scrolls up. However, using pagination also presented its own set of problems. Mainly, it was hard to get the real time update a user would expect from a chat app. New messages would only be loaded if a user refreshes the page. This is of course not feasible as well. We tried for a few days to combine the pagination function with real time updates but the end result was extremely buggy and didn't always work. Sometimes new messages would appear and sometimes it would not.

Eventually, we realised that it would be much harder than we thought to make a modern looking chat app with functionalities such as the ability to send images and locations. We thus decided it would be too time consuming to build a chat app from scratch and such a task should probably be done as a new and separate project. We looked for some solutions on the internet and found that we could use a chat API instead for our app. We researched the different chat APIs and eventually we settled on using CometChat which allowed us to very easily build the chat functions that you see above. It provided us with all the functionalities we wanted from our chat function and also was easy to integrate with our database.

Of course, there are some downsides to using a chat API which is the cost. Currently, we are on a free plan which only allows a maximum of 25 users. If the app was to be released, using a this chat API would add to our costs.

Database

Our app uses Google's Firestore database as a backend database. Firestore is a noSQL, non-relational database. As a result, a lot of the data in firestore is denormalized and there are alot of duplications in the database to ensure fast queries. Hence, when a user changes a data we have to ensure that the changes are synced at all required places.

In order to ensure that, we used Firebase's cloud functions functionality to implement a function that triggers whenever a change in a user or business detail is made. For instance, if a business decides to change his name, the cloud functions ensure that all job requests and reviews with the businesses name in it are also updated.

Testing

For testing, we managed to carry out user testing and some unit testing. For our user testing, we installed the app in our phones and would constantly test out the different functions and features in order to ensure that they work as intended. We also checked any changes with our database, to ensure that any updates and changes to our database are made correctly. For our unit testing, we tested on certain cases to see if our app still works as intended. For instance, we tested to see if there were certain fields that were null when they should not be and what would happen if the user inserts an incorrect value. Some of the bugs that we encountered during the testing process are listed below.

- Initially, when we designed the job requests page, every time a user deleted a job request, the job request itself would be deleted from the database. However, this ended up breaking our notification system. For instance, when a user cancels and deletes a job request, the business would receive a notification that the job request has been cancelled. However, when he clicks on the notification which would lead to the job request in question, the app would crash since the request has been deleted from the database and could not be found. After discovering this, we then modified the job request functionality such that when a user deletes a job request, it is only marked as delete but it still exists in our database.
- When a user edited a job request, what is supposed to happen is that the old request data is passed to the new request along with the edits. However on testing and inspecting the database, we realised that we missed out on some lines in the code which resulted in the new request not being correctly updated.
- As mentioned in the features above, each job request has an action item associated with it. However, on using the app, we realised some of the actions were different from what was expected. We then inspected the code and realised that we missed out on an if else clause in our code.
- We also managed to encounter a bug in our edit listing page. In our business listing model, the listing's phone number is designed to be stored as an integer. However, in our edit listing page, whenever a business updates its phone number, the new number will be stored as a string instead. Hence, when loading a data, the app would crash since a String cannot be converted to an integer.
- Whenever a user filters a query in the search listings page and goes to another component with the bottom navigation bar and then returns to the search listing page, the filter options would be reset. This would result in a bad user experience. We

managed to fix this by using a viewModel which stores the data in such a way that it can survive configuration changes.

- A user can view a business profile either through a search page or by clicking the profile picture in the job request page. In our unit testing, we discovered that when the user views the business profile via the latter method and submits a job request from there, the job request will have a null id.
- We found that by using the multi select function in our job requests page, users were able to delete confirmed job requests even when the date of job has not passed. This was not an intended feature since users can accidentally delete a confirmed job request. Thus, we created checks to ensure that confirmed job requests are not accidentally deleted.
- In our unit testing, we discovered that users can enter a postal code that does not fit into Singapore's postal code format of 6 numbers, which may potentially cause a bug in the future. Thus, we made changes to the code to ensure that users have to submit a valid postal code.
- For our push notifications system, although the firebase Cloud Messaging token ID is deleted from our database when a user signs out, the server can still push notifications to the device. We found out that when a device is registered as a token, it will still be saved in the firebase cloud. This will cause the device to receive notifications even if a user has signed out of their account. Hence, we found a way to completely remove the registered token from the cloud whenever a user signed out from their account.
- When testing our push notification system, we realised that when users signed back in to their account, they will receive all the notifications that they received beforehand. This could cause unnecessary discomfort to users. Thus, upon debugging, we realised that a line of code that is used to obtain the necessary information from the database is triggered every time a user signs in. This is because the database will be updated when a new device token is detected. Hence, we made use of a different way to obtain the data from the database.
- When we used multiple devices to test the push notifications system and pushed multiple notifications to users at one go, we realised that only one notification is spawned, and the latest notification will override the previous one. As such, we decided to give each notification a unique ID so that each notification will be pushed separately.