

**Handy Work**

**Presented By:**

1. Abdalhadi Zoheir Abdalhadi Jaber 31901002080
2. Mustafa Kamal Yousef Al-monayer 31901002037
3. Abdalrahman khaled hasan azaizeh 31901001021
4. Mohamad-Hisham Bassam al-Belbeisi 31901001053
5. Ahmad Issam Abu-Baker 31901001071

**Supervisor: Dr.Raed Khalil**

Majors

1. Software Engineering
2. Computer Science

**Al-Balqa' Applied University**

**Salt- Jordan**

**Date: 19/5/2023**

**Abstract**

**Handy Work** is a platform that provides a connection between two end-users, The first one for example is capable of completing carpentry jobs and the other end-user is in need of a person with carpentry talents to fulfill a certain Job.

Therefor a person with a certain talent can apply for jobs in his field of expertise, and a person with certain job request can add a job request to be completed.

When a user finds a Job that he can complete, the system allows him to make an offer with the price he is willing to complete the job with, therefor it also guarantees the other user (AKA Job Lister) to get the job done with the best price possible.

**Handy Work** also provides the users a two-way review system to assure to the user the quality of each user as a job lister and as a job taker.

**Acknowledgement**

**Dear supervisors,**

I am writing to express my deepest gratitude and appreciation for the support and guidance I have received throughout the completion of my graduate project, **Handy Work**. I would like to take this opportunity to acknowledge the invaluable assistance provided by several individuals and entities who have played a significant role in the successful completion of my project.

First and foremost, I would like to extend my sincere thanks to my project supervisor, Dr. Raed Khalil, for their exceptional guidance, unwavering support, and invaluable expertise. Their insightful feedback, constructive criticism, and commitment to excellence have been instrumental in shaping the direction of my project and enhancing its overall quality. Their dedication and availability have been truly remarkable, and I am incredibly grateful for their mentorship throughout this endeavor.

Also , I would like to express my heartfelt appreciation to all the participants who generously dedicated their time and shared their insights for my research study. Their contribution has been invaluable in enriching the data and enhancing the overall quality of my project.

Please convey my sincere gratitude to everyone mentioned above for their invaluable contributions. Without their support and involvement, the successful completion of this project would not have been possible.

Once again, thank you for providing me with the opportunity to undertake this project, and for the support and encouragement throughout. I am truly grateful for all the assistance and guidance I have received.

Yours sincerely,

**Handy Work team**

**DECLARATION**

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Bachelor in (software engineering andcomputer science) is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed By:

Abdalhadi Zoheir Abdalhadi Jaber 31901002080 Mustafa Kamal Yousef Al-monayer 31901002037 Abdalrahman khaled hasan azaizeh 31901001021 Mohamad-Hisham Bassam al-Belbeisi 31901001053 Ahmad Issam Abu-Baker 31901001071

**Date: 15/01/2023**

**Table Of Contents**

**Chapter 1 Introduction**

Preamble page1

Project Motivation page1 Project aim and objectives page1

Project software, hardware and techniques requirements page2

Project expected output page4

Project scheduling page5

**Chapter 2 Background**

**Frontend level page8**

**Backend level page11**

**OS level:**

**Data level: page11**

**Network level: page 13**

**Backend level page15**

**Security level page 18**

**Table Of Contents**

**Chapter 3: Analysis and implementation**

**Analysis methodology page20**

**Implementation and design page23**

**Figures of relationship page 24**

**Backend implementation page27**

**Frontend implementation (sequence diagrams) page 33**

**Design and user scenarios page 37**

**Chapter 4: Conclusion and future work**

**Conclusion page57**

**Limitations page58**

**Future work page59**

OTP Page 59

User type page 60

Maps page 61

**References page63**

**List of figures**

**Figure 1 Software Development Cycle page5**

**Figure 2 android studio logo page8**

**Figure 3 flutter logo page9**

**Figure 4 dart logo page10**

**Figure 5 user-job relation. Page24**

**Figure 6 user- job\_offer relation. Page24**

**Figure 7 user-job\_review relation. Page25**

**Figure 8 job-image\_url relation. Page25**

**Figure 9 job- job\_offer relation. Page26**

**Figure 10 job- job\_review relation. Page26**

**Figure 11 com.grad.handywork package page27**

**Figure 12 com.grad.handywork.aop package page27**

**Figure 13 com.grad.handywork.config package page28**

**Figure 14 com.grad.handywork.controller package page28**

**Figure 15 com.grad.handywork.dto package page29**

**Figure 16 com.grad.handywork.entitiy package page30**

**Figure 17 com.grad.handywork.enumtypes package page30**

**Figure 18 com.grad.handywork.exception package page30**

**Figure 19 com.grad.handywork.mapper package page31**

**Figure 20 com.grad.handywork.repo package page31**

**Figure 21 com.grad.handywork.service package page32**

**Figure 22 com.grad.handywork.util package page32**

**Figure 23 com.grad.handywork.validation package page33**

**Figure 24 Login sequence diagram page33**

**Figure 25 User registration sequence diagram page34**

**Figure 26 Update user profile sequence diagram page34**

**Figure 27 Image upload processing using Base64 Encoding sequence diagram page35**

**Figure 28 User review system sequence diagram page36**

**List Of Tables**

**Table 1.………………………………..Page 11**

**Table 1.2………………………………..Page 16**

**Chapter 1**

**Introduction**

* 1. **preamble:**

Handy work is an app that provides a way for Two End-Users to be able to complete and request Jobs while having a creative way of pricing called offers and also having a two way review system to ensure that no user gets scammed.

* 1. **Project Motivation:**

Our motivation is to create a platform that helps the craftsmen with low reputation to gain popularity in his field of expertise.

* 1. **Project aim and objectives:**

The aim of this project is to develop a new mobile app that provides the users with a service to complete and request jobs while benefiting financially, by using the two way review system we aim to reduce the time taken to find qualified person to take the job compared to finding outside the app.

* 1. **Project software, hardware and techniques requirements.**
* **Software Requirements :**

**Backend:**

1. Java 17 or later
2. MySQL Server 8
3. Telebit.io
4. Any distribution of Linux

**Frontend:**

1. Android OS 6.0 or later

* **Hardware Requirements:**

**Backend:**

* 1. **PC**

**Frontend:**

1. **Mobile devices:** Any mobile device.

* **Techniques Requirements:**

1. Scrum development methodology.
   1. Continuous integration and deployment.

1.2 Test-driven development.

* **Performance and Security Requirements:**

**Maximum response time:** 3 seconds

**System uptime**: 99.99%

**Data encryption:**  AES-256 bit

**Data Hashing:** Bcrypt

* 1. **Projected expected output:**

**Project Software development:**

The projected expected output for a software development project could be to develop a new mobile application that has at least 10,000 downloads within the first month of launch.

**Project Marketing campaign:**

The projected expected output for a marketing campaign project could be to increase app traffic by 20% and generate at least 100 leads within the first quarter of the campaign.

We are expecting the users to benefit from using the app on financial level and on a time consuming level while also we want the low reputation craftsmen to gain more popularity to make their life’s better, lastly we plan to make the app available at any given time .

**Project scheduling:**

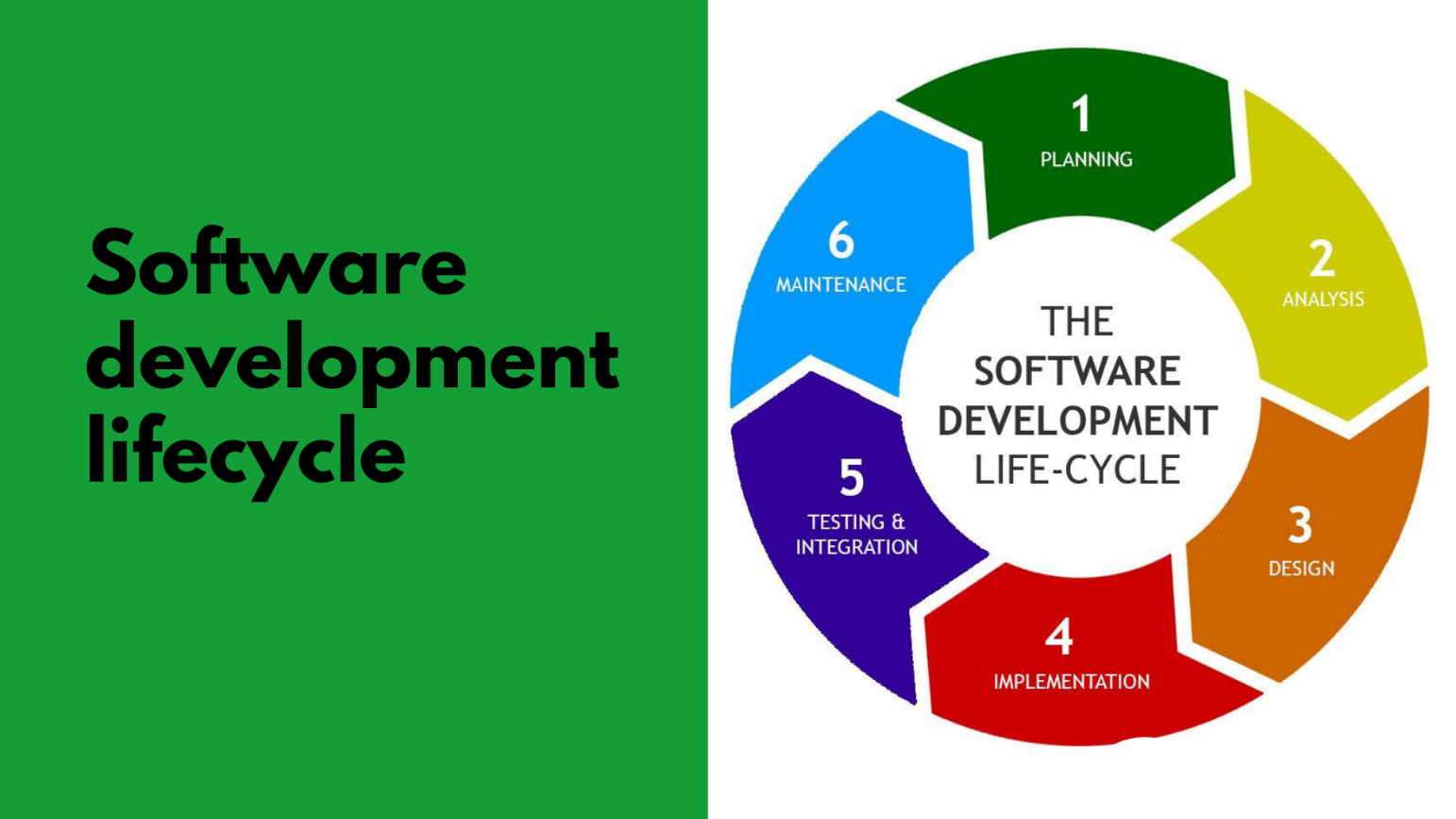


Figure 1 Software Development Cycle

1. **Planning Stage:**

We started working on the system at the planning stage, and exercising All system requirements and specifications are prepared in the analysis stage which is the next stage of the system development life cycle (SDLC).

* 1. **Designing Stage:**

Moving to the design stage system architecture and design are developed based on requirements gathered during the analysis stage.

1. **Implementation Stage:**

During the implementation phase, all the system specifications, needs and requirements are mixed to build the final system, it is important to ensure that the app is designed for user-friendliness and performance. This includes optimizing the app for speed, ensuring that it is responsive, and designing an intuitive user interface. In addition, the app should be tested thoroughly to identify and address any bugs or issues before it is released to the public.

A key consideration during the implementation phase is ensuring that the app is compatible with a wide range of devices and platforms. This involves testing the app on different devices and operating systems to ensure that it works seamlessly across all platforms.

Finally, once the app has been developed, tested, and deployed, it is important to monitor user feedback and make updates as necessary. This may involve releasing bug fixes or introducing new features based on user feedback.

Overall, the implementation phase is a critical stage in the app development process that requires close attention to detail and a commitment to quality and user satisfaction.

Table 1 Project scheduling table

| **Stage Name** | **Start Date** | **End Date** | **Duration (days)** | **Resource Assigned** |
| --- | --- | --- | --- | --- |
| Project Planning | 15/03/2023 | 22/3/2023 | 7 | Project Manager |
| Requirements Gathering | 23/01/2023 | 01/04/2023 | 7 | Business Analyst |
| Design and Prototyping | 02/04/2023 | 04/04/2023 | 3 | Designer |
| Development | 05/4/2023 | 07/05/2023 | 32 | Developer |
| Quality Assurance | 08/05/2023 | 09/05/2023 | 2 | QA Specialist |
| User Acceptance Testing | 10/05/2023 | 15/05/2023 | 6 | Business Analyst |

**Chapter 2**

**Background**

**Introduction:**

The tools we are using in the development of the app on both the front end and the back end are shown below.

**Frontend level**

1. Android Studio:



Figure 2 android studio logo

**We used android studio as our development environment, Android Studio is the official integrated development environment (IDE) for Android application development. It is based on IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools.**

1.2 Flutter **(3.7.8 • channel stable):**

****

Figure 3 flutter logo

**We used flutter development kit to deign the user interface, Flutter is Google's portable UI toolkit for crafting beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.**

* 1. Dart **(Dart 2.19.5):**



Figure 4 Dart logo

**We used dart language as our UI programming language, Dart is a client-optimized language for developing fast apps on any platform. Its goal is to offer the most productive programming language for multi-platform development, paired with a flexible execution runtime platform for app frameworks.**

**The Backend level is on the Next Page**

**Backend levels**

**OS Level:**

**1.Linux Server (Ubuntu 22.10) :**

The backed of the application is running on a Linux machine running Ubuntu 22.10 the reason we choose Linux is that it provides a lot of feasibility especially in the server side there is a lot of great tools for running back-end applications and DBMS (Database Management system) and we are using some great that only run on Linux i.e., Docker.

**2.Docker:**

We are running the back-end jar on a docker container it makes it very portable and can be installed and run in a single command also we are running the database on a docker container it makes it portable with all the data and can also be installed and run in a single command.

**Data Level**

**1.MySQL:**

We are running the back-end jar on a docker container it makes it very portable and can be installed and run in a single command also we are running the database on a docker container it makes it portable with all the data and can also be installed and run in a single command.

**2.ORM (Object Relational Mapper):**

We are using Java Persistence API (JPA) as our ORM JPA is a JavaEE (Called Jakarta After JavaEE Version 10 (x)) specification that persists (Save, Delete, Read and Update) data to the database by using object to entity mapping, it maps your objects to the entities in the database so no longer you need to pre-process any of the data form the database to save it in a object JPA gives you that for free and it handle entity relationships too.

**Network Level**

**1.Telebit:**

Telebit This tool is a web tunnel that doesn't just make it easy for us to deploy the back-end locally but also provides us with a HTTPS tunnel that makes all the requests to the server secure and encrypted this tool is available to UNIX based system which includes Linux.

**2.Java Web servlet:**

web servlets is the hart of web applications they are the module that handle the request and map it to the right method that handle that request and they are the ones which responsible for sending the response back java has it own implementation it's called java web servlet.

**3.Apache Tomcat:**

we need to run the code we wrote in a web server a very famous web server for java is tomcat and its made by the will known organization Apache its considered an industry standard.

**4.Base64 Encoding:**

We are using base64 encoding and decoding, base64 is a binary to text encoding it has 64 characters and uses the '=' character as padding we are encoding the images binaries when the user submit them in the front end

we encode this images to base64 using dart them send them to the back-end as string or list of string(case dependent) then we decode this images to their binaries then upload then to the CDN by cloudinary.

**Back-end Level**

**1.Maven**

We used maven as our builder and dependency managers maven is very reliable when it comes to managing the dependencies and building the application and it makes the source code potable between IDEs.

**2.Spring & Spring boot:**

We built the backend using spring 6 with spring boot 3 spring framework is a very large framework for the Java programming language for example you can build with it android application in short almost anything that can be built with Java you can use spring to build it, here we are using it for the back-end, now spring boot is a bootstrap for spring application it makes building projects much more faster by reducing the boilerplate code and configuration code and files so you can focus on the building the project itself.

**3.Java:**

We used Java 17 (OpenJDK Version) it’s the minimum Java version supported by Spring 6 and it’s the latest Java LTS (Long Term Support) version you must go with LTS versions on production apps so that’s why we used this specific version, the back-end is composed from more than 7000 lines of code (Lines we have written) and 13 packages and 69 java files that include classes, Enums, interfaces and annotations (@interface).

**4.Cloudinary Java SDK :**

We used cloudinary to store our images files so what we do is upload the image to the back-end when the user submit it then upload it to cloudinary from the backend then we save the URL in the database we did this to ensure fast image fetching time and to lower the strain on the back-end.

**5.Project Lombok:**

Project Lombok is a java library its job is to lower boilerplate code because Java is a statically typed language it requires a lot of boilerplate code like setter and getter methods, constructors, to String, equal and hash code so because these methods are generic (Boilerplate Like) project Lombok can save time and space and lower complexity by generating these methods project Lombok saved us almost 4,000 lines of code.

**6.Spring MVC:**

Spring MVC is a web framework with the MVC (Model, View, Controller) design pattern it is the most mature technology for building java web applications and its considered the industry standard.

**7.DTOs (Data To Object):**

On the mention of the MVC pattern it has a big flaw in a restfully API approach it uses the entity (Objects)as the receiving data and sending data it's a big security flaw you don't want to send the whole data some data shouldn't be sent so we use DTOs to converts the DTO to a object when receiving data and object to DTO when sending data so that the user cannot access the entity directly.

**8.Mapstruct:**

Map struct is a code generation library same as Lombok but it has a different use it generate the code the converts the object to a DTO and the DTO to object that's why it is called mapper.

**9.Spring Validation**

it's a framework that make constraints on the objects that the server receive in our case its DTOs and objects that get persisted in the database it has some free implementation for some constraints like max characters and min it also provide the ability to make custom constraints that's how we check for the uniqueness of emails, phone numbers, and usernames we build a custom validator to check if those are unique or not.

**Security Level**

**1.JWT (Json Web Token):**

We are using Json web token as our security holder Json web token are the most implementation for securing web application they are considered an industry standard Json web token are Superior in every way, shape and form they are encrypted and signed with secret keys so it can’t be tampered with any they are state-full and request aware they hold the data of the issuer (the User) so that the server can know who is asking and who

is sending.

**2.HTTPS:**

we are using HTTPS calls everything is encrypted.

**3.Bcrypt:**

we are using bcrypt as our password hashing function if anyone could hack their way to the database the passwords couldn't be known even the dev how is viewing the database cannot know the password because we are using salts with the password salts are random characters that get added to the password so even if the same password is stored twice the hash output will be totally different, bcrypt is the number 1 tool when it gets to hashing functions even tech giants use it.

**4.AOP (Aspect Oriented Programming)**

Aspect Oriented Programming (AOP) is a JavaEE (Jakarta) tool some times it is referred to as interceptors and that's their job they intercept method calls while JWT job is Authentication (Checking if the user is Authenticated and can access the system) Aspects (Interceptors) it's job is Authorization (Checking if the specific user can access a specific resource) some examples we used Aspects for checking if the user is updating his profile or another user profile, checking if the user is posting a job, review, offer in his name or another user profile and so on, so that's what aspects helped us with making sure no body is accessing a resource he does not own.

**Chapter 3**

**Analysis and implementation**

**Analysis and methodology:**

The analysis phase is the first stage in the software development life cycle (SDLC) where a software project is analyzed in detail. During this phase, the primary focus is on gathering and analyzing requirements to determine what the software should do and how it should function.

The analysis phase is essential because it sets the foundation for the entire software development process. It helps to ensure that the software project meets the needs of its intended users and stakeholders, and that it is developed within the available resources.

**The following are some of the key activities that take place during the analysis phase:**

**1.0** Requirements gathering: This involves collecting detailed information about what the software is supposed to do, who will use it, and what the expected outcomes are.

**2.0** Feasibility study: This involves assessing the viability of the proposed software solution. It involves evaluating factors such as technical feasibility, economic feasibility, and operational feasibility.

**3.0** Risk assessment: This involves identifying and analyzing potential risks and issues that may arise during the development process and throughout the software's lifecycle.

**4.0** Defining project scope: This involves identifying the boundaries of the software project and determining what is included and excluded.

**5.0** Prototyping: This involves creating a basic version of the software to test and evaluate its functionality.

The analysis phase is critical for the success of the software project as it helps to ensure that the project meets the requirements and expectations of its stakeholders. It also helps to identify potential issues early in the development process, which can save time and resources later.

## **Analysis Methodology:**

In this project, we chose object-oriented methodology to analyze the system, whole system.

Object oriented analysis is considered as one of the most common techniques to analyze the system requirements. This process or method is based on using objects, O-O analysis commonly used in real-world applications. The main result of using this method is a set of software objects that represent actual system users, things, transactions and events which are allocated in the current system.

The most common language that is used in object-oriented analysis is UML that stands for Unified Modeling Language. It uses a set of different concepts that come from object-oriented programming language there are set of different examples about them such as:

- Classes.

- Inheritance.

- Objects.

- Messages.

- Methods.

The fundamental idea behind an object-oriented (OO) language is object decomposition, breaking the complex software system down into its various objects, combining the data and the functions that operate on the data into a single unit, the object. Objects are discussed and built by modeling real-world instances. A typical OO system consists of a number of cooperating objects, each of which may or may not collaborate with other objects in order to achieve some tasks for the user. Real-world objects display the characteristic of high cohesion; they maintain a single theme or focus, in true, and software objects modeling real world objects.

**Implementation and design:**

1. **implementation:**

Database level:

We used one to many relationships. One-to-many database relationship is a relationship between two tables where one record in a table is linked to multiple records in another table. It allows for efficient data storage and retrieval by avoiding duplication of data. One table is designated as the "one" side and the other as the "many" sides. A foreign key column is added to the "many" tables that references the primary key of the "one" table, linking the two tables together.

Below are the figures of relationships:

A screenshot of a computer

Description automatically generated with low confidence

**Figure 5 user-job relation.**

A screenshot of a computer

Description automatically generated with low confidence

**Figure 6 user- job\_offer relation.**

A screenshot of a computer

Description automatically generated with low confidence

**Figure 7 user-job\_review relation.**

**A screenshot of a computer

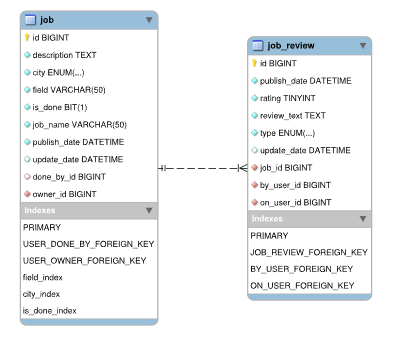
Description automatically generated with medium confidence**

**Figure 8 job-image\_url relation.**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Figure 9 job- job\_offer relation.**

****

**Figure 10 job- job\_review relation.**

**Backend level:**

Java Packages:

1.0



Figure 11 com.grad.handywork package

this package contains the class (HandyworkApplication) with the main method. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork)

2.0

A screen shot of a computer screen

Description automatically generated with low confidence

Figure 12 com.grad.handywork.aop package

this package contains the classes that intercept the method calls throughout the system these classes are divided in to two aspects (Main Focus of the classes) first is security witch checks if the user.

is accessing his own resources and the second is pre-processing to pre-process data and check for data sanity before interning the business logic classes, these classes can live in the service package but

its best to put them in their on classes because you want separation in the code you don't want to bloat the service classes, service classes should contain only business logic. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/aop)

3.0

A screen shot of a computer

Description automatically generated with medium confidence

Figure 13 com.grad.handywork.config package

this package contains the classes that initiate classes as beans a bean is a object that is managed within the spring container rather than using the new keyword you can use the auto wired annotation (@Autowired) this pattern is called dependency injection, rather than creating a certain object every time a method is called you can let spring container manage your object creation and by default every object is managed as singleton this will lower memory consumption and will improve performance and will produce lower latency. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/config)

4.0

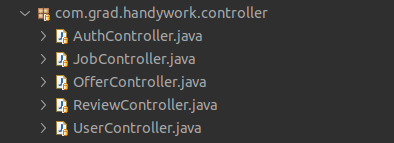


Figure 14 com.grad.handywork.controller package

this package contains the controller classes it uses the mvc design pattern this classes are responsible for the mapping of URL to method this classes act like routers they route the right URL to the right method call and they define the expected input and output this I/O data is defined as classes (because java is a statically typed language).[source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/controller)

5.0

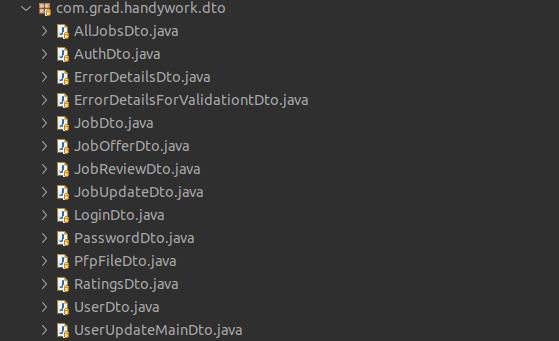


Figure 15 com.grad.handywork.dto package

this package contains the I/O classes the DTO (Data to object) design pattern overcomes a big limitation in the MVC world typically when you are using MVC you will accept and send the data as is,

more specific you will accept and send entities lets say you are sending a user entity the password will be sent, and you don't want to do that with the help of DTOs you can make new classes the will exclude and transform this data so you have full controller of what data you accept and what data you send all this will be done using mappers that will transform an entity to a dto.  
 [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/dto)

6.0

A picture containing text, font, screenshot

Description automatically generated

Figure 16 com.grad.handywork.entitiy package

this package contains the class representation of the database tables and the relation between this tables all thanks to JPA, for example a user can have many jobs as owned so the user class contains List<job> field named jobs and a job can have one user as owner so it has a User field named owner and we can observe this in the done By relation and throughout all of the entities.

[Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/entity)

7.0

A picture containing text, font, screenshot

Description automatically generated

Figure 17 com.grad.handywork.enumtypes package

this package contains the enumeration for the cities and the review types this enums are defined as String. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/enumtypes)

8.0

A picture containing text, screenshot, font

Description automatically generated

Figure 18 com.grad.handywork.exception package

this package contains the runtime exceptions and the handler for this custom made runtime exceptions. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/exception)

9.0

A screen shot of a computer

Description automatically generated with low confidence

Figure 19 com.grad.handywork.mapper package

this package contains the mapper that map/transform entities to DTOs and vice-versa using map struct lib. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/mapper)

10.0

A screen shot of a computer

Description automatically generated with low confidence

Figure 20 com.grad.handywork.repo package

this package contains the JPA interfaces that persist (create, read, update, delete) data into the database. [source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/repo)

11.0

A screen shot of a computer

Description automatically generated with medium confidence

Figure 21 com.grad.handywork.service package

this package contains the service classes which are responsible for all the business logic in the system they calculate, transform, filter...etc data.

[Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/service)

12.0



Figure 22 com.grad.handywork.util package

this package contains the utility classes mainly the utility responsible for image upload. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/util)

13.0

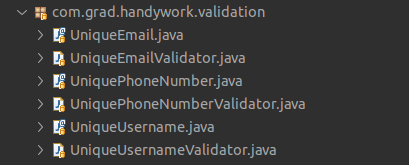


Figure 23 com.grad.handywork.validation package

this package contains the annotations and their handlers to validate the uniqueness for usernames, emails, and phone number. [Source code link](https://github.com/mustafaAlmonayer/handywork-api/tree/main/src/main/java/com/grad/handywork/validation)

**Frontend level:**

1. Below are some of the sequence diagrams explaining the system.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 24 Login sequence diagram

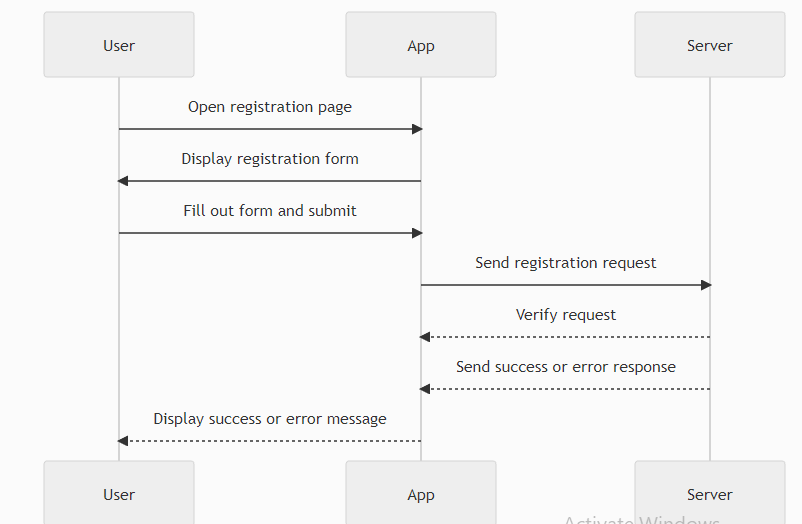


Figure 25 User registration sequence diagramA screenshot of a diagram

Description automatically generated with low confidence

Figure 26 Update user profile sequence diagram

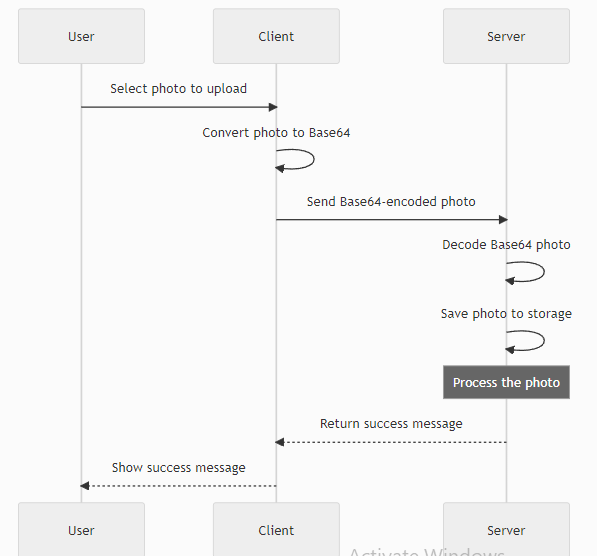


Figure 27 Image upload processing using Base64 Encoding sequence diagram

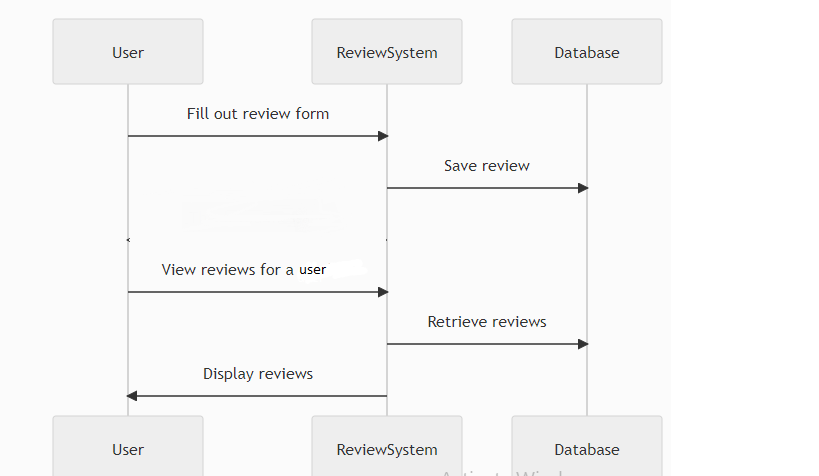


Figure 28 User review system sequence diagram

1. **Design and user scenarios**

**Scenario 1: opening the app.**

A logo for handy work

Description automatically generated with low confidence A screenshot of a login form

Description automatically generated with medium confidence A screenshot of a phone

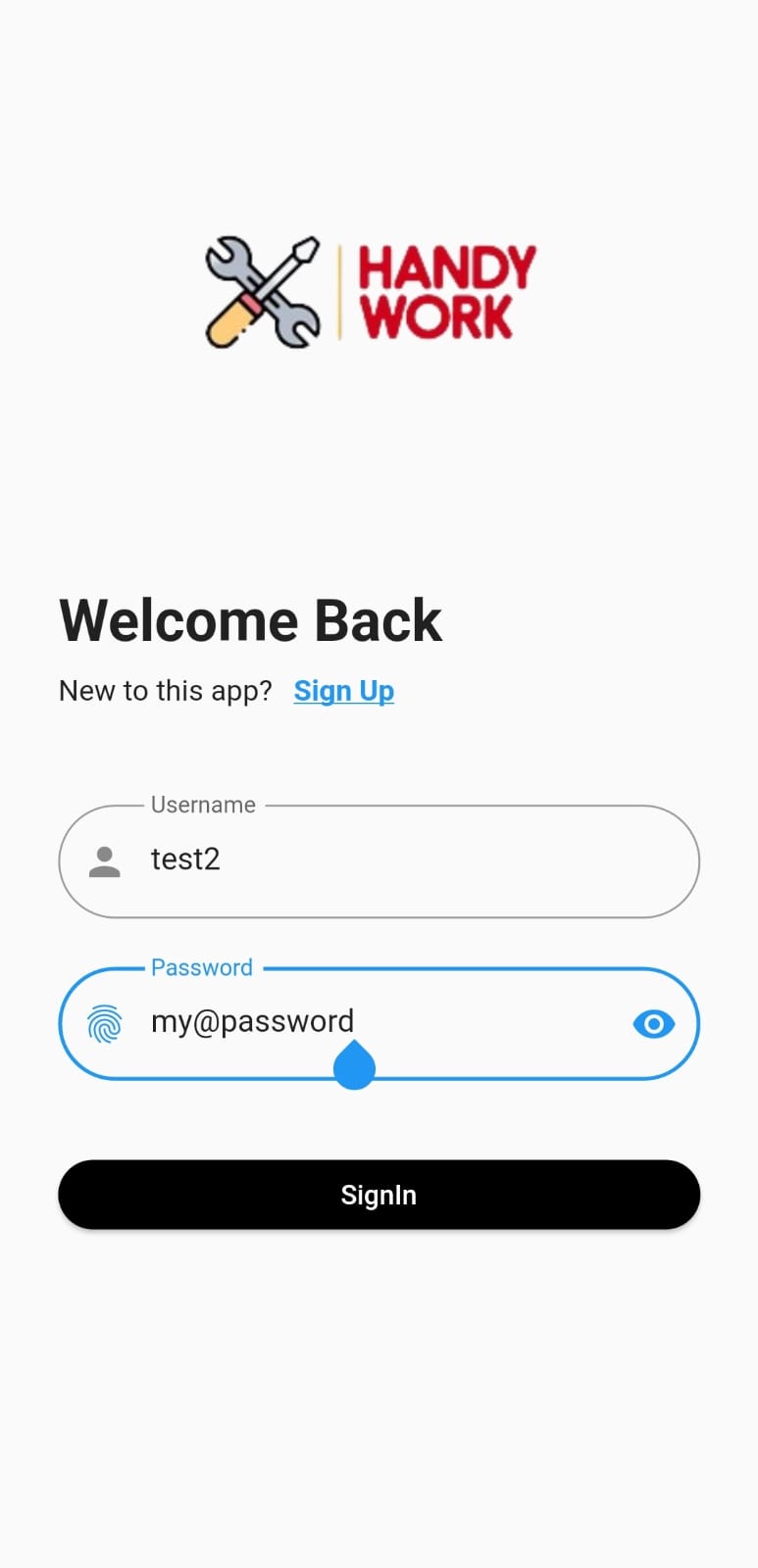
Description automatically generated with medium confidence

1.Splash screen 2. Login screen 3. signup screen

When opening the app a splash screen will showup for a couple of seconds then it will take you to the login screen(labelled as figure 2). In case you don’t have a previous account you tap on the sign up button shown in (figure 2) and it takes you to the signup page and you fill out the fields and you can optionally upload a profile picture,(\*the user name, email and phone number must be unique\*) then you tap on signup at the bottom of (figure 3),In case the data is not unique a toast massage will be shown that it is already used.

By that that you now have an account and can use your username and password to sign in.

**Scenario 2: signing to the app and opening the main screen (Browse Jobs)**

 A screenshot of a phone

Description automatically generated with medium confidence

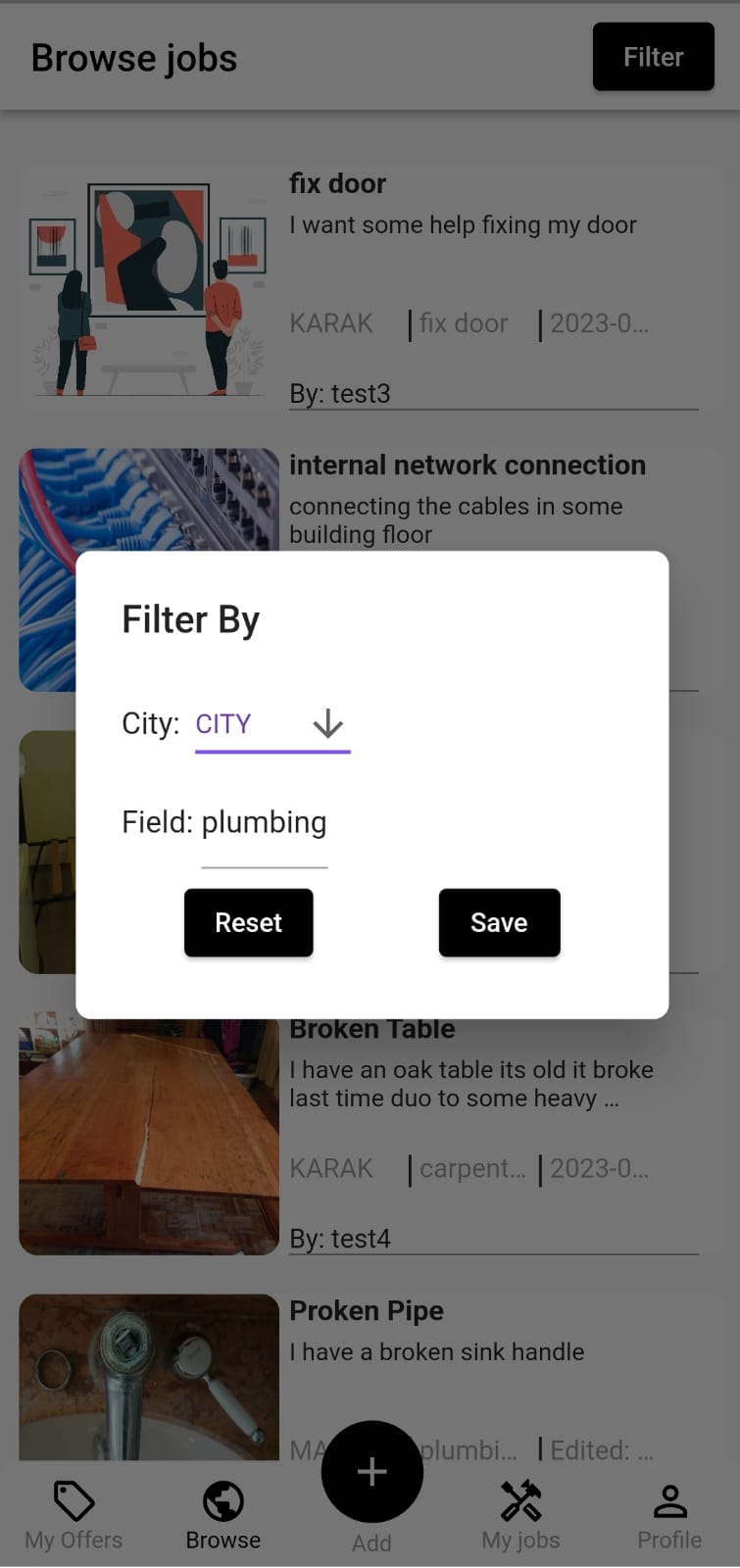
1.Log in screen 2. The browse screen

Now that you have a an account you type your credentials in their fields and tap sign In ,next it will take you to the brows page(labelled as 2)

Which we will discuss in the next scenario.

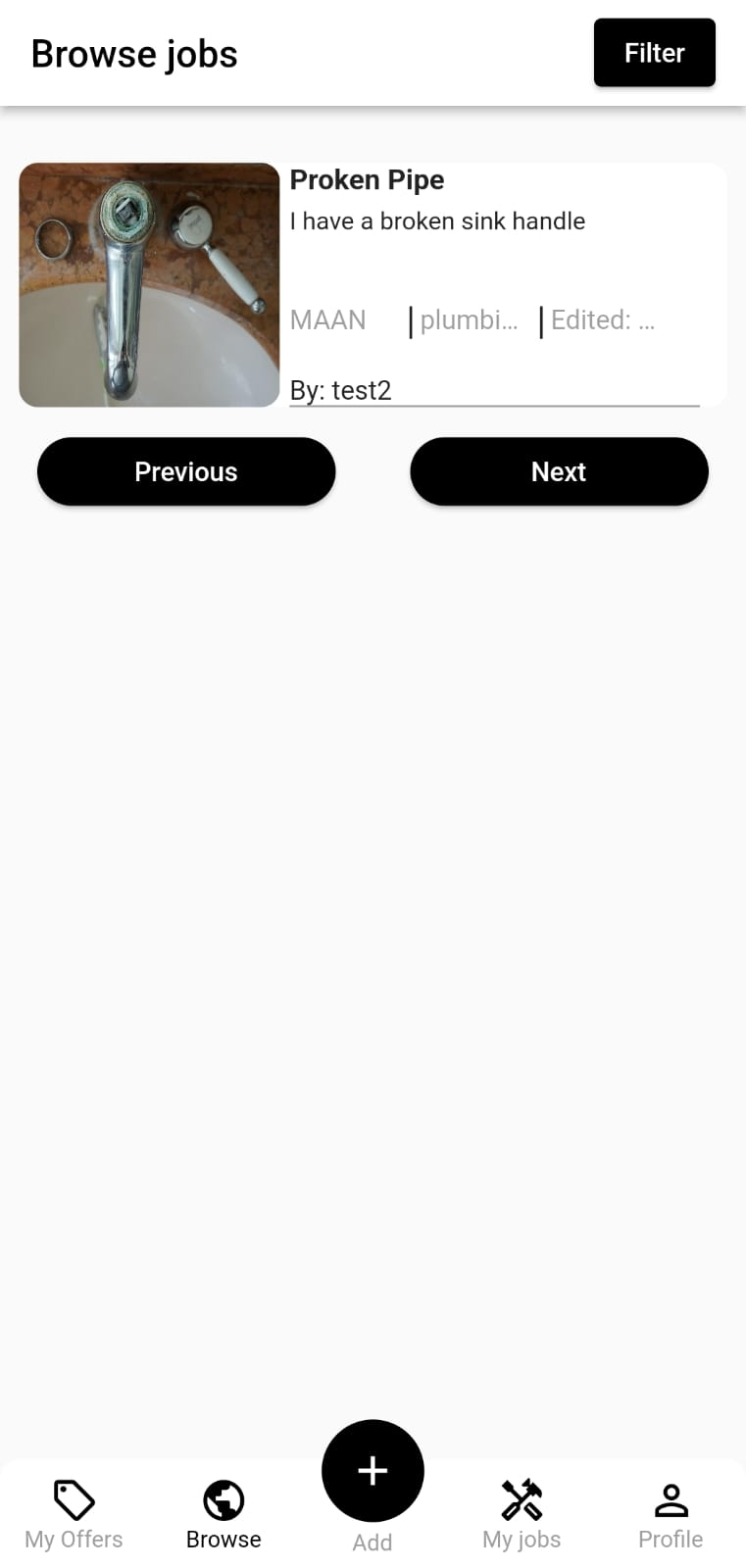
**Scenario 3: browsing Jobs and searching for them based on city and the job field (filtered sorting)**

A screenshot of a phone

Description automatically generated with medium confidence

1. The browse screen 2.filter screen popup

For start the brows screen will show you all the available jobs and you can keep browsing by tapping next and previous, Lets say you are a plumber and you do not want to keep searching job by job to find a job in your field of expertise, so you tap on the filter button shown in the browse screen (labelled as 1),then a popup screen is shown and in it you can filter the showed jobs in the browse segment , as the previous example you are a plumber so you press filter and type plumbing as shown in (figure 2) (\*you can also filter jobs by City\*) then all the jobs in the plumbing field will be shown as the figure below.



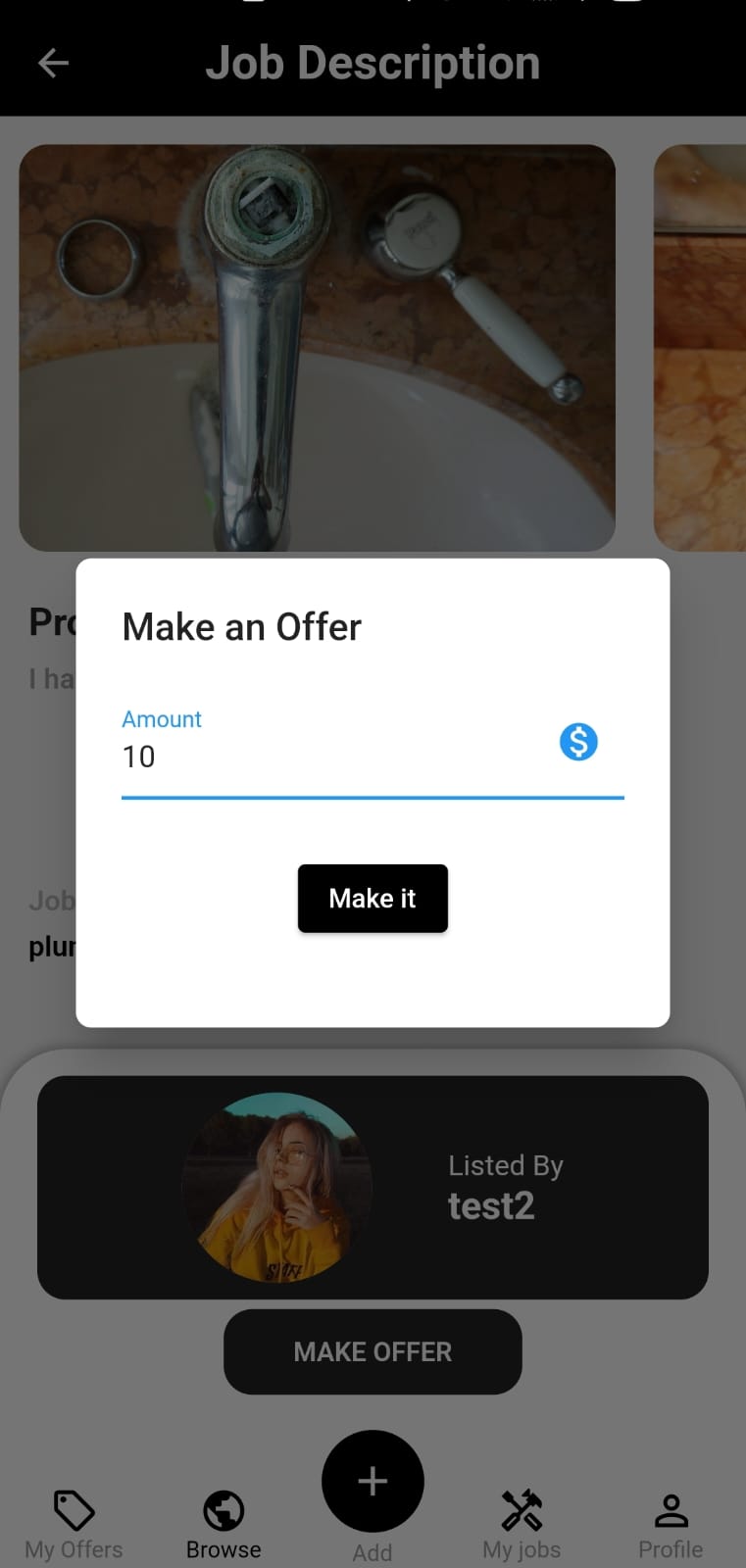
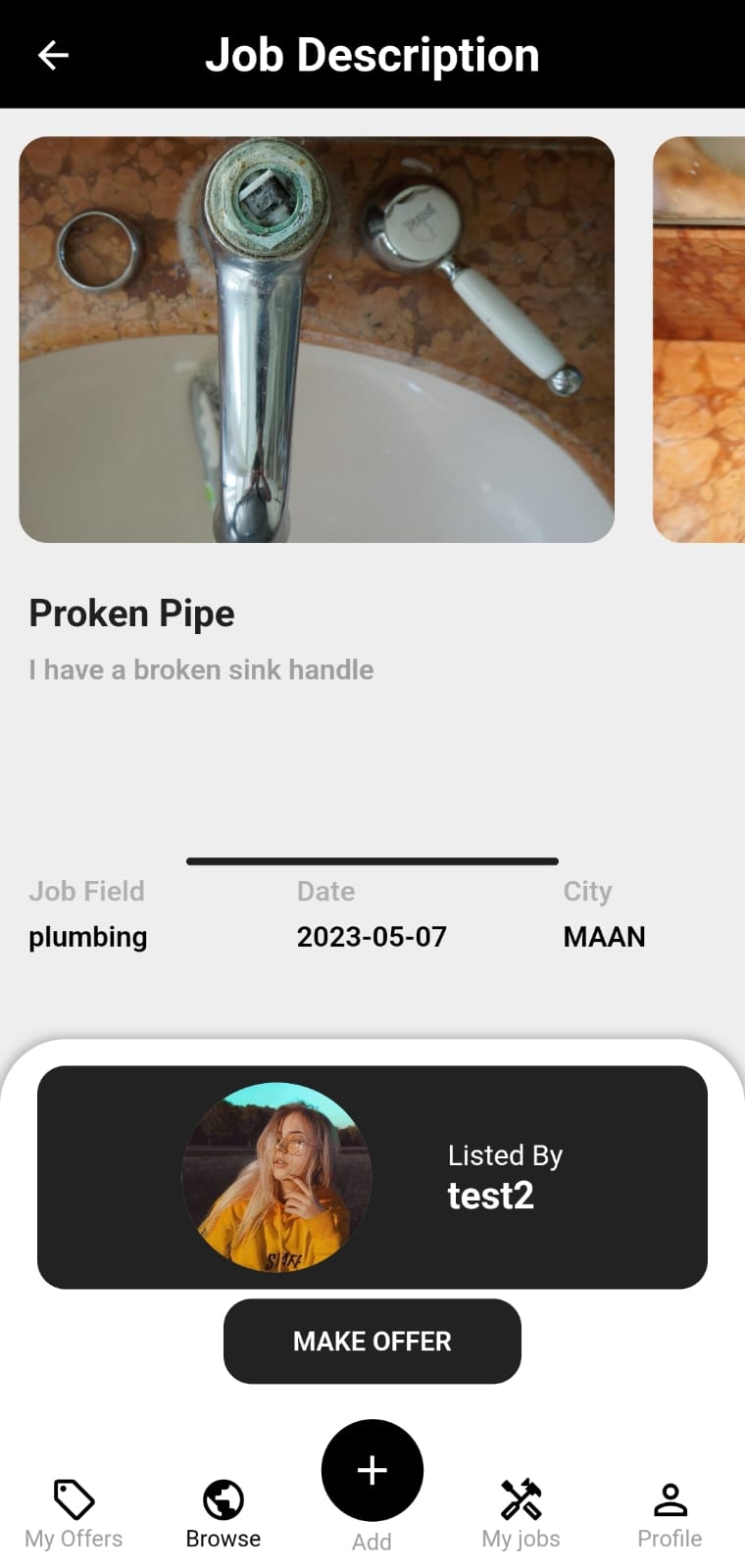
Tap on the job to see the description.

The figure on the left shows the result after using the filter.

3. Filter result

**Scenario 4: finding a job, looking the details, by whom it was listed and making an offer for it.**

For example, you wanted to know more about the job that was just showed after the filtering, you tap on the job, and it will take you to the job description page as shown in the figure below.



Tapping on make offer shows this pop-up screen.

Tap on the listed by to take you to the lister profile

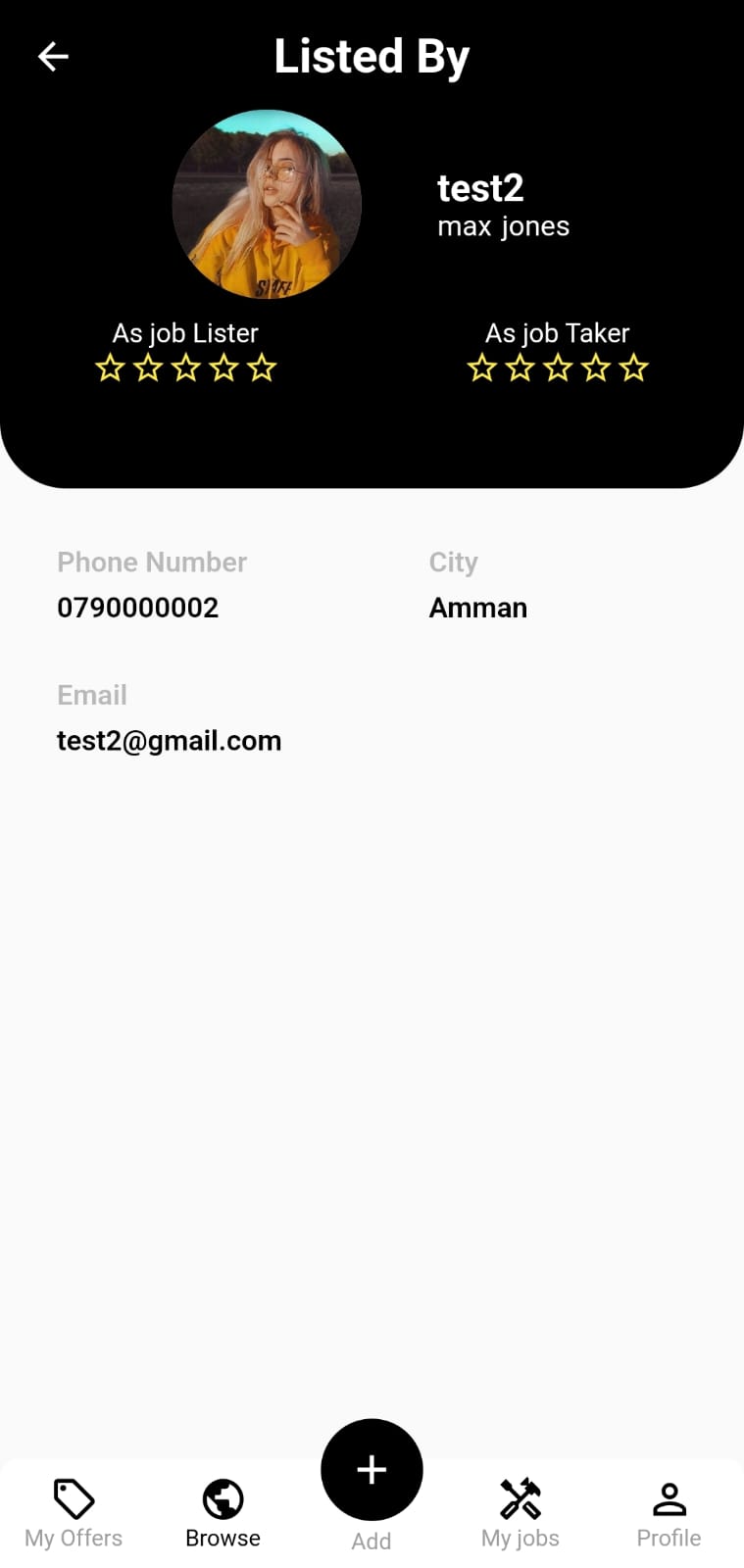


1.Job description 2. Make offer popup.

So you see the job details and find yourself qualified to complete the job so you make an offer with the amount you are wishing to complete the job with.

As for the offer it will appear for the job lister as we will show in another scenario,

But before that lets say you want to check on the user that listed the job, you press on the listed by box to take you to the profile as shown below.



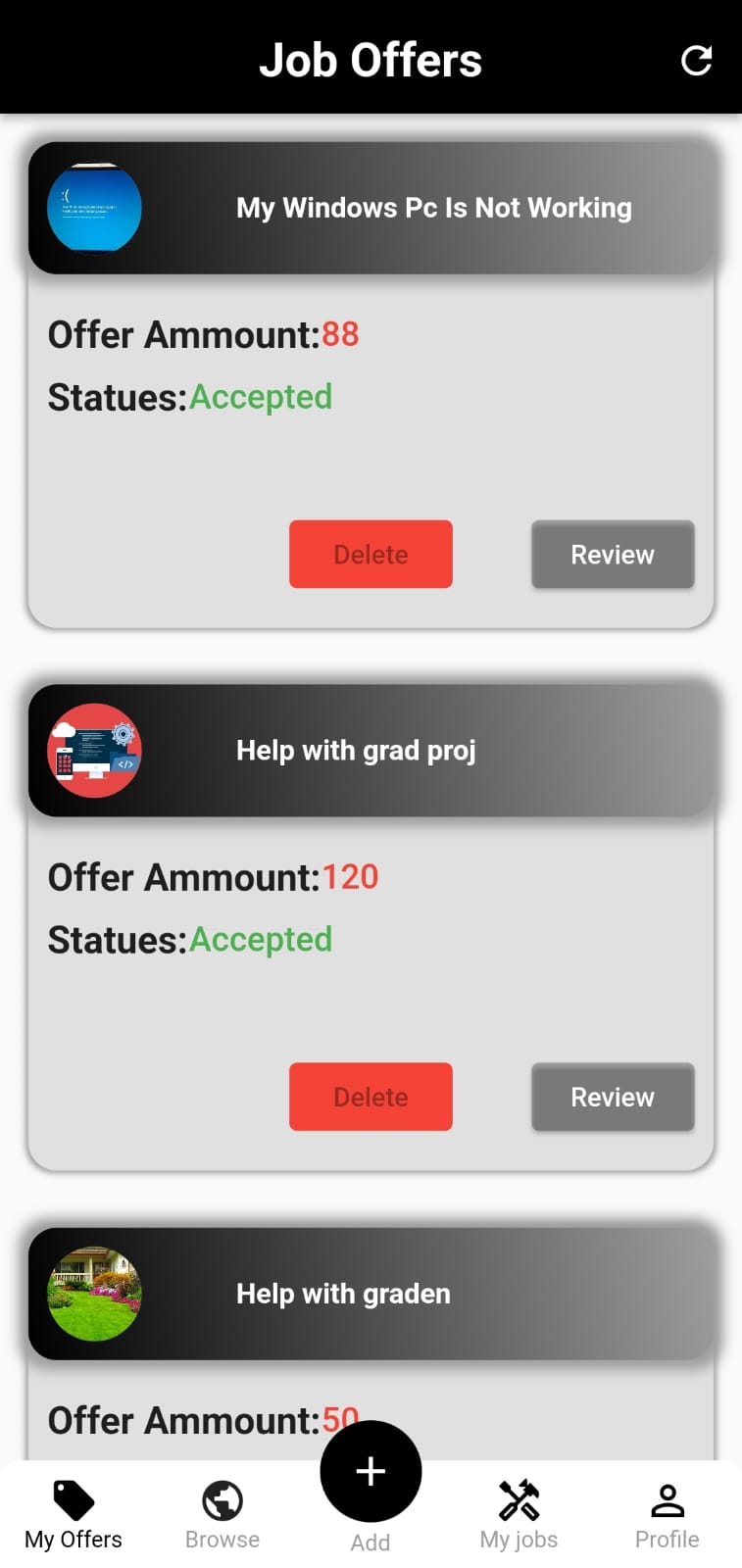
3.job lister profile

The job lister profile shows you some information (email, phone number, the city he is active in, also the username and the first, last name)

Another thing that is visible is the ratings, and as you can see there is a two types of ratings , **the first:** as job taker ,and the **second:** as job lister , will also be shown in another scenario.

**Scenario 5: seeing all the offer that has been made by you to other jobs.**

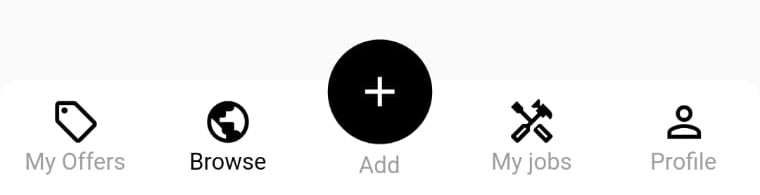
A screenshot of a phone

Description automatically generated with medium confidence

to see the offers that you already made and their status , from the navigation bar at the bottom we press my offers ,then the screen in figure 2 , each offer contains the amount you offered and statues (accepted \ rejected \ pending), also a delete button that will only work if the offer is rejected or pending , also a review button so that the job taker review the job lister (the review could be about the job really matching the details mentioned or paying the approved offer amount ) , also when pressing on the upper bar of the job offer a screen containing a review from the job lister about the job taker(could be regarding his completion of the job or the quality of completing).

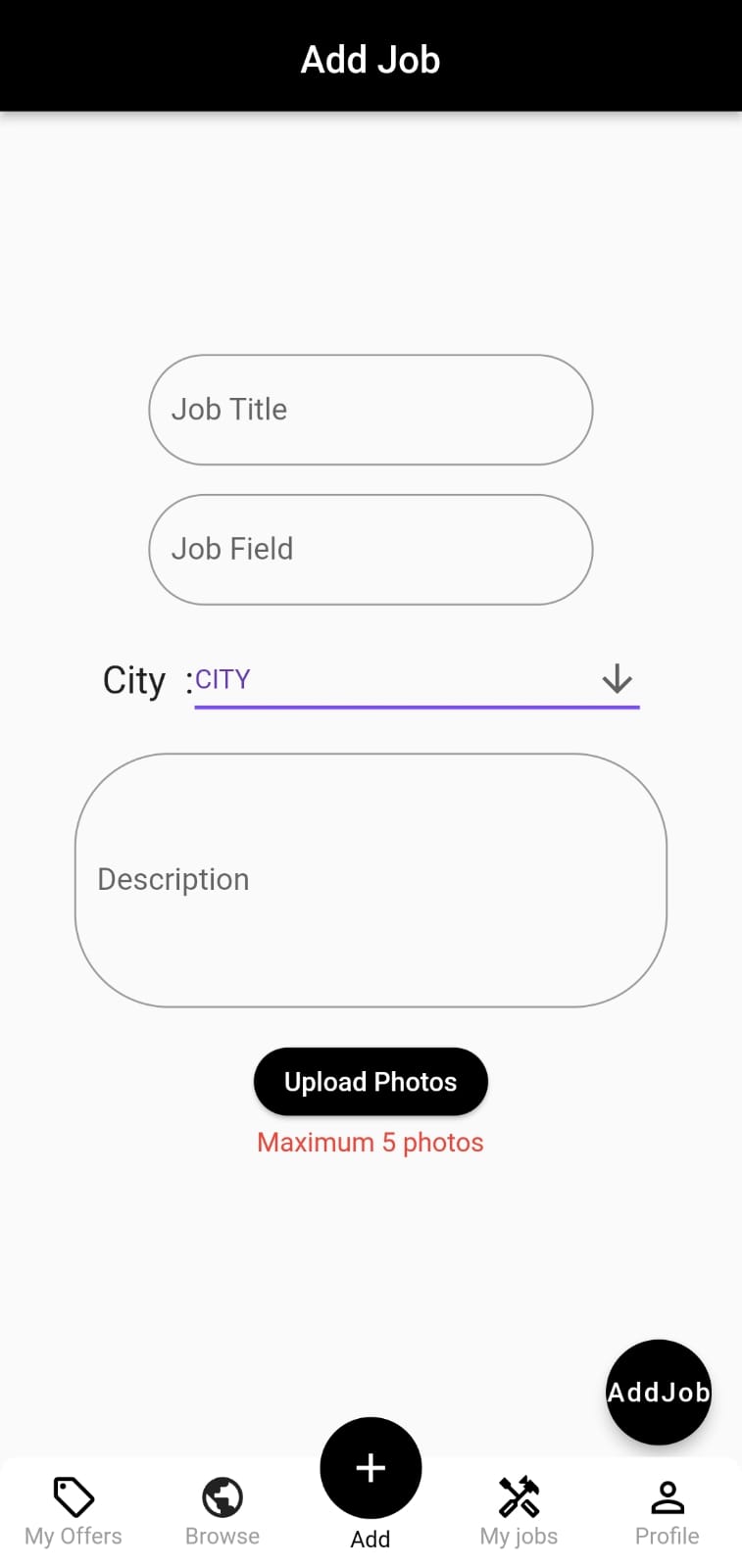
**Scenario 6: Requesting or adding a new job and checking the offers that came for it .**

So if you want to add a job you need to press the **+** in the bottom navigation bar



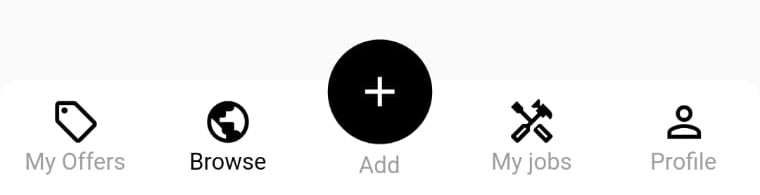
1. Navigation bar

After a tapping a new screen opens as below.

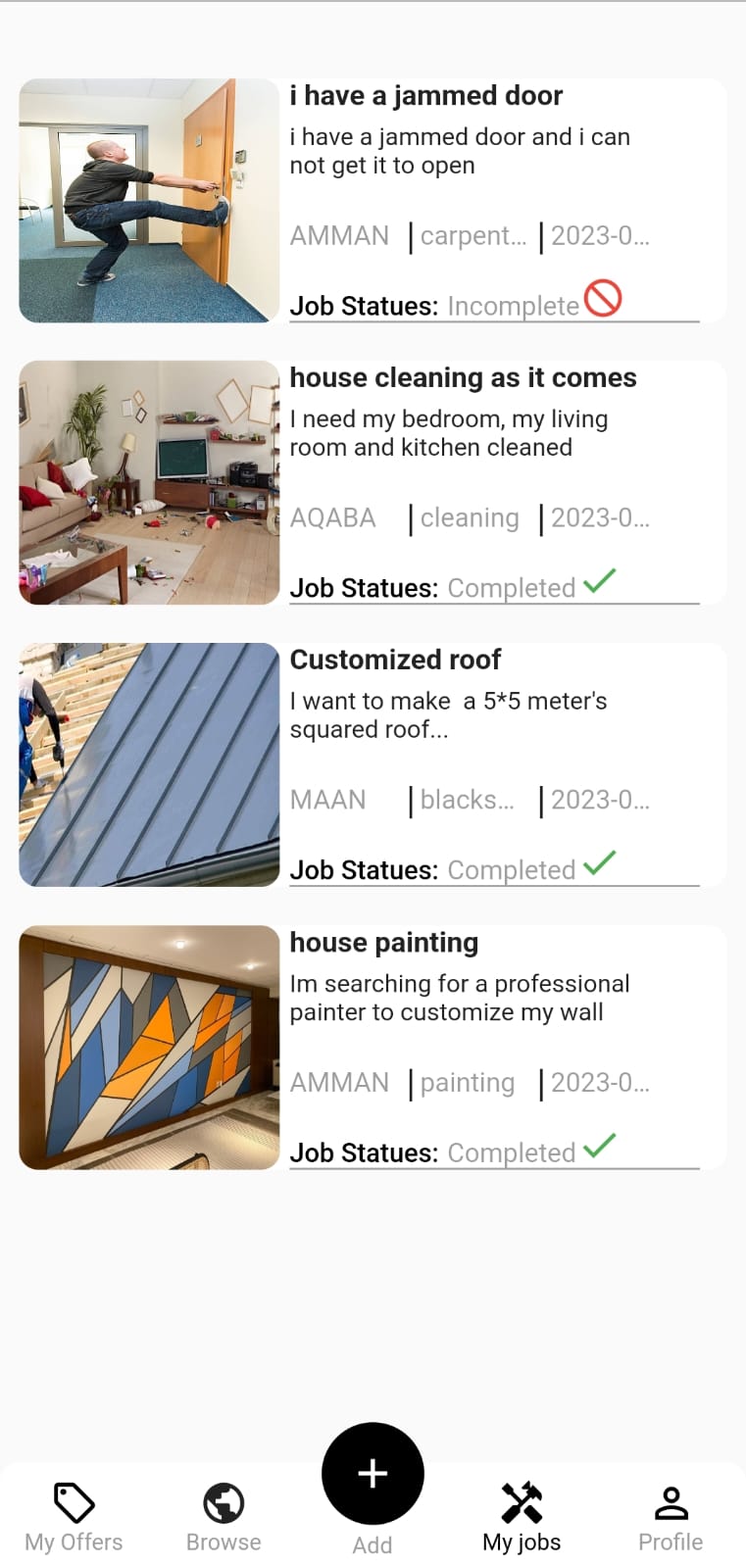


After this screen appears you fill out the required details and you can upload up to 5 photos showing what the job is about. After you fill all the required you press on the add job button at the bottom right of the screen.

So after adding a job request ,you can look all your jobs in the my jobs section in the navigation bar.



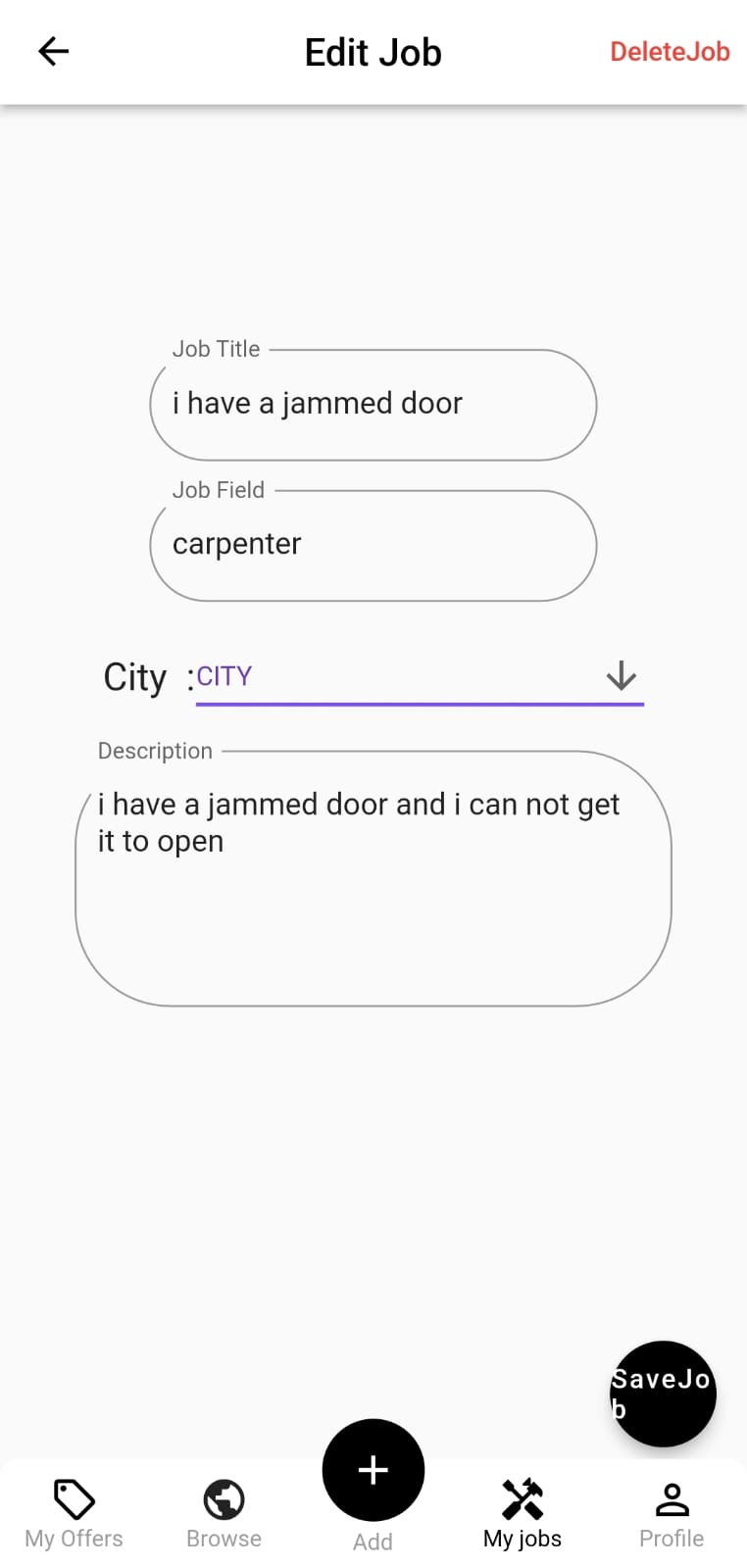
A new screen will open containing the jobs that have been listed by you.



For example you add the jammed door job listed at the top ,now want to check to see if there is any offers that came ,you simply tap on the job and then you tap at offers button at the bottom then you will see the offers as shown in the next page.

Before looking at the offers ,in case you want to adjust the job details you can press the blue circled icon and it will open a screen so you can adjust the details.

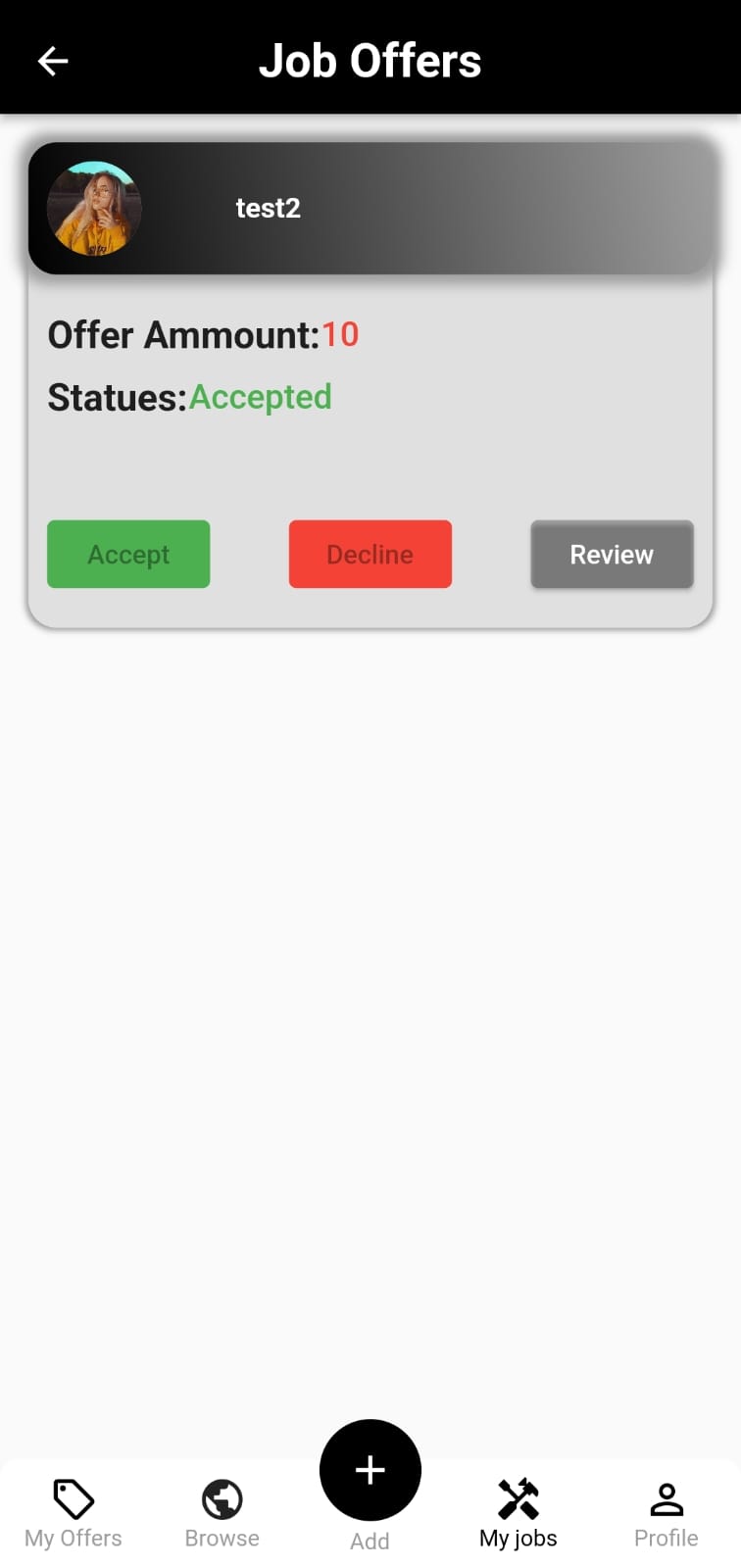
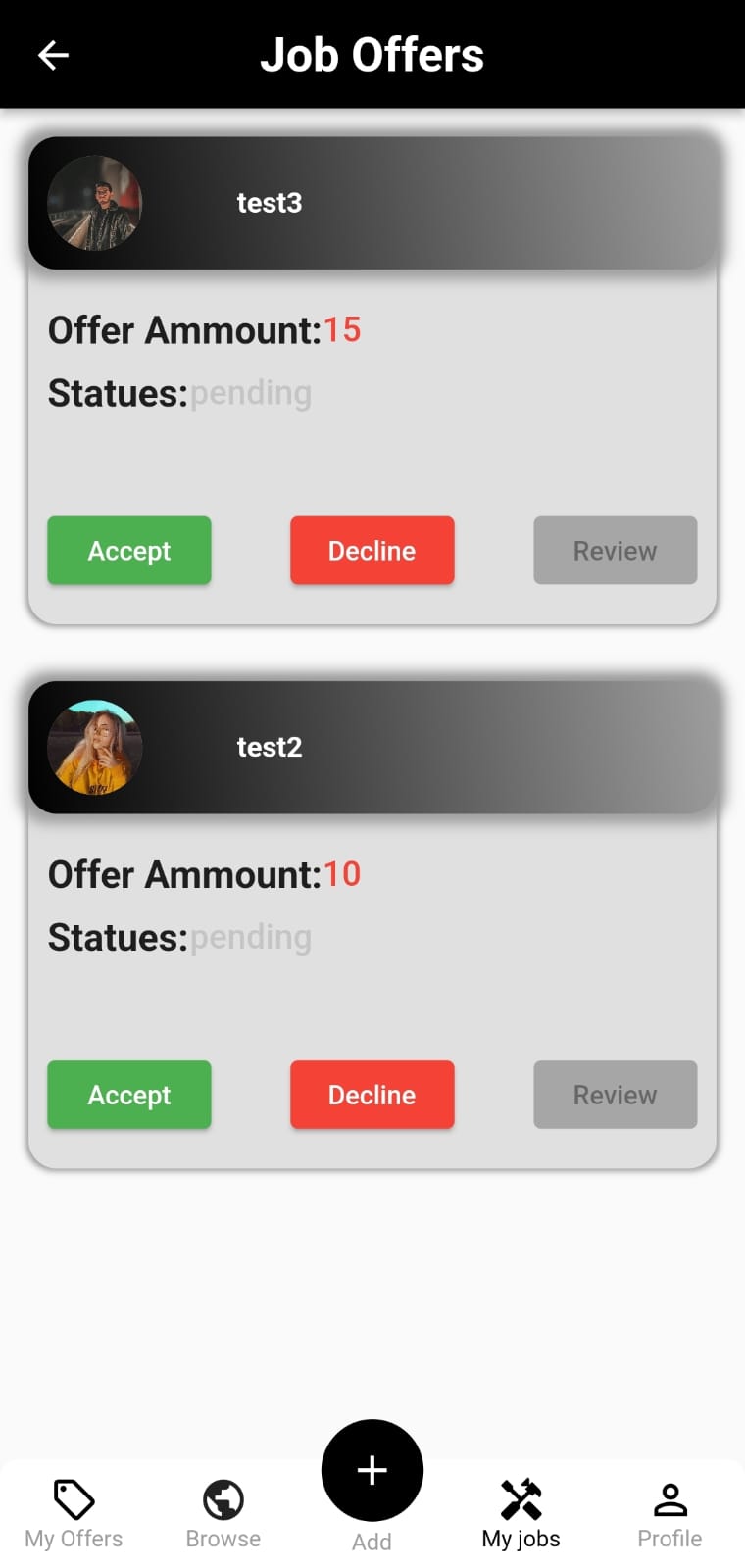
A screenshot of a phone

Description automatically generated with low confidence

1.edit button + offers 2.edit job details

You can also delete the job if you want to .

Now for the offers ,by tapping on the offers button circled in red a screen containing the offers will appear.

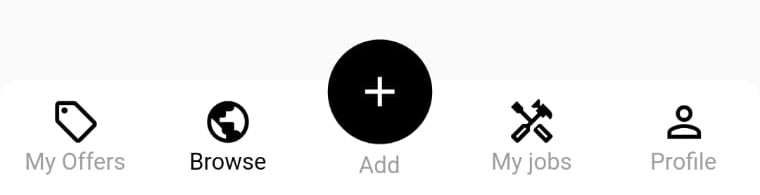


1. Job offers 2.job offers after accepting

Lets say you got to offers from 2 different users and you liked the user called test2 better, you tap on accept and all the other offers will disappear as shown in figure 2,also the offer statues for the person who made the offer will change to accepted.

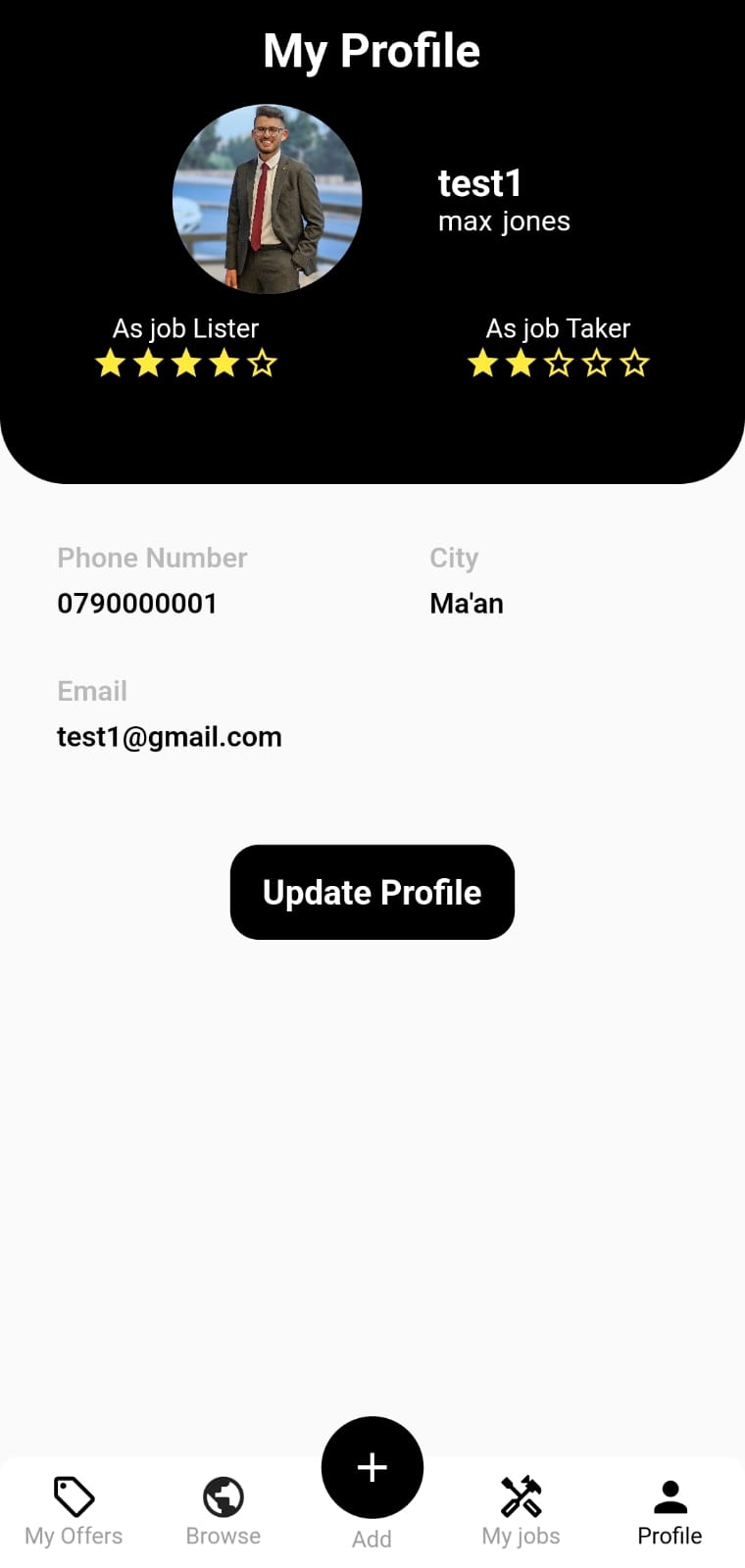
**Scenario 7: in case you want to update your profile .**

By tapping on the profile icon at the right of the navigation bar



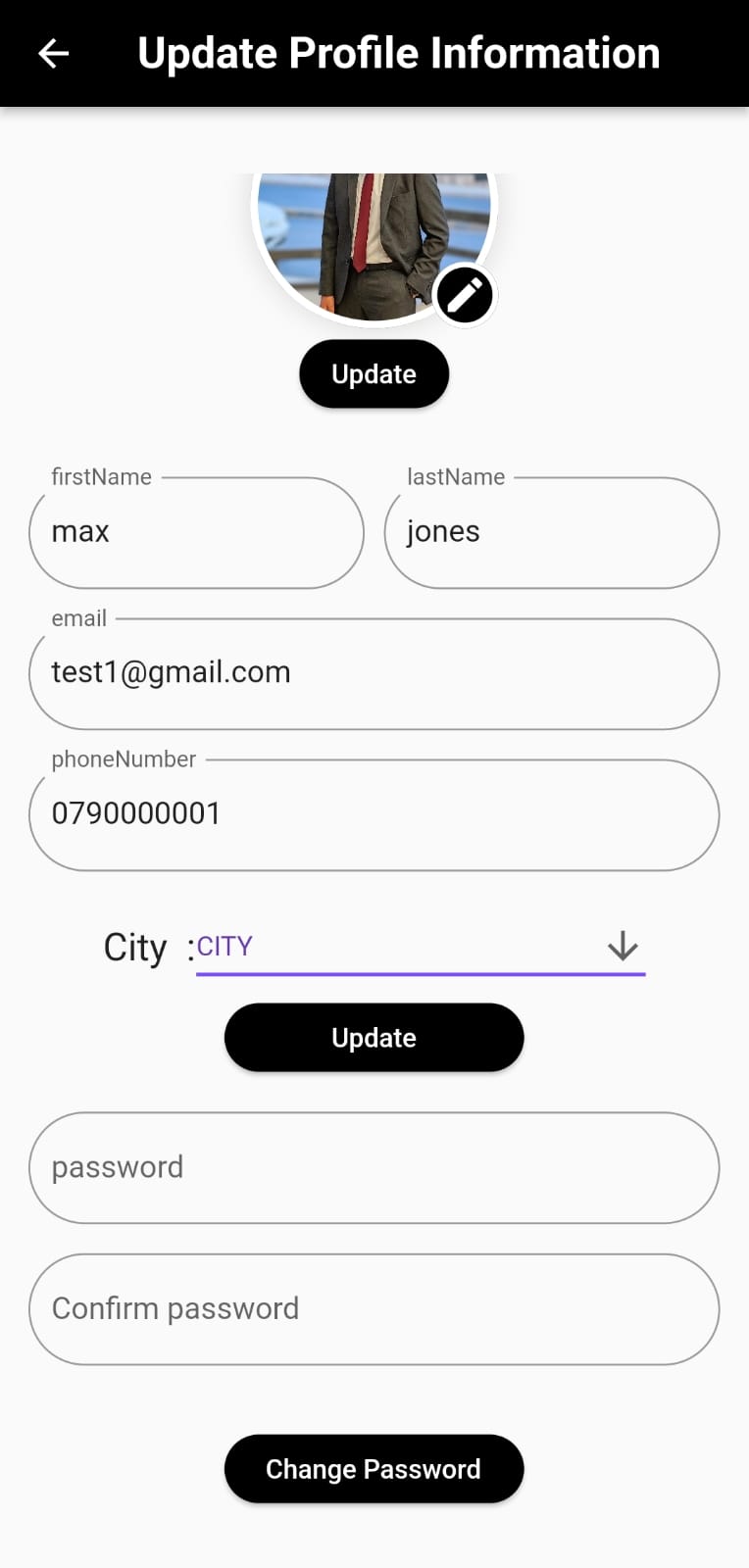


A screen containing all your info will appear like below



In case you want to change your information, you need to tap on update profile button, then a screen containing the fields to change will appear.

A screenshot of a login box

Description automatically generated with low confidence

1.authentecation 2.update profile screen

An authentication screen will appear ,just type in your password and the screen (figure 2 ) appears , change the info you want and leave the rest as it is , you can also change your profile picture so don’t worry about using the one , then just tap update and your new info will be visible on your profile screen.

Also if you want to change the password just type in the new one in both fields (\*should be the exact same in both fields\*) then tap change password.

**Scenario 7: Rating the job lister and the job taker(Two way review system).**

for example you made an offer and it was accepted and you went and completed the job, handy work offers a two way review system, it helps you decide when making an offer for a job or accepting an offer , because a by logic if a user have bad reviews then it is better to stay away and not making an offer to his job or accepting an offer from him.



So, how to know a user’s reviews? You simply open their profile and at the top you will find two types of ratings.

A close-up of a phone number

Description automatically generated with low confidence

Here are the reviews that the user got by completing jobs.

If you want to see them you just tap on the red circled mark.

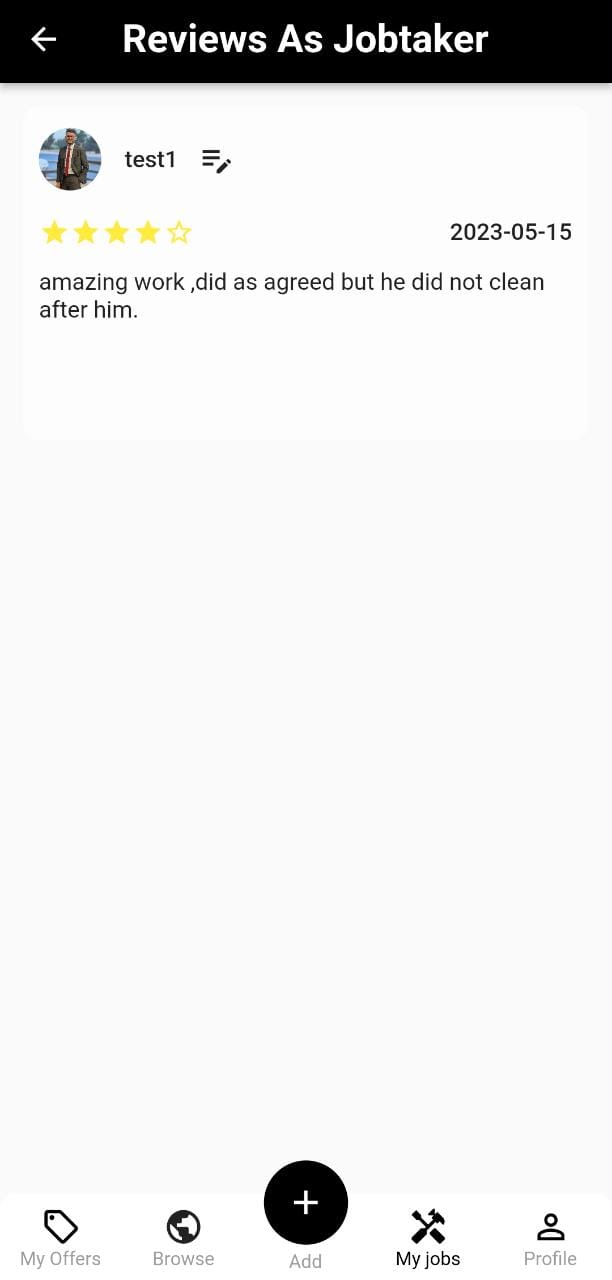
Here are the reviews the user got when listing jobs and as you can see there is no reviews yet.

If you want to see them you just tap on the blue circled mark.



1.offerd by profile

After tapping for example on the (as job taker ),this screen will open

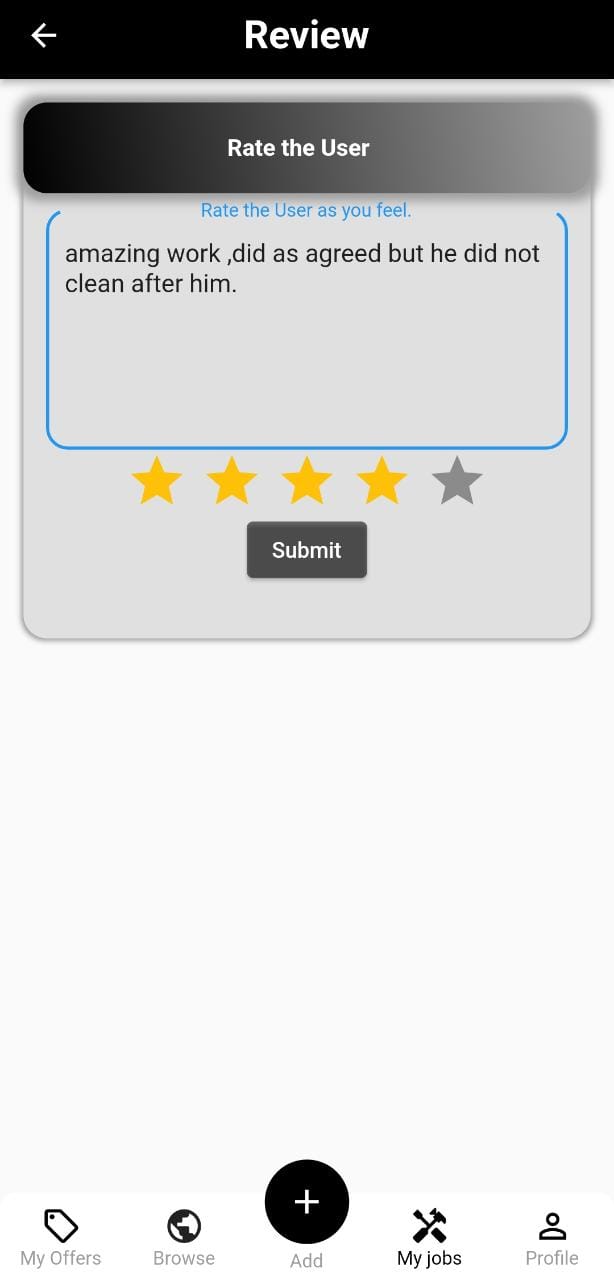
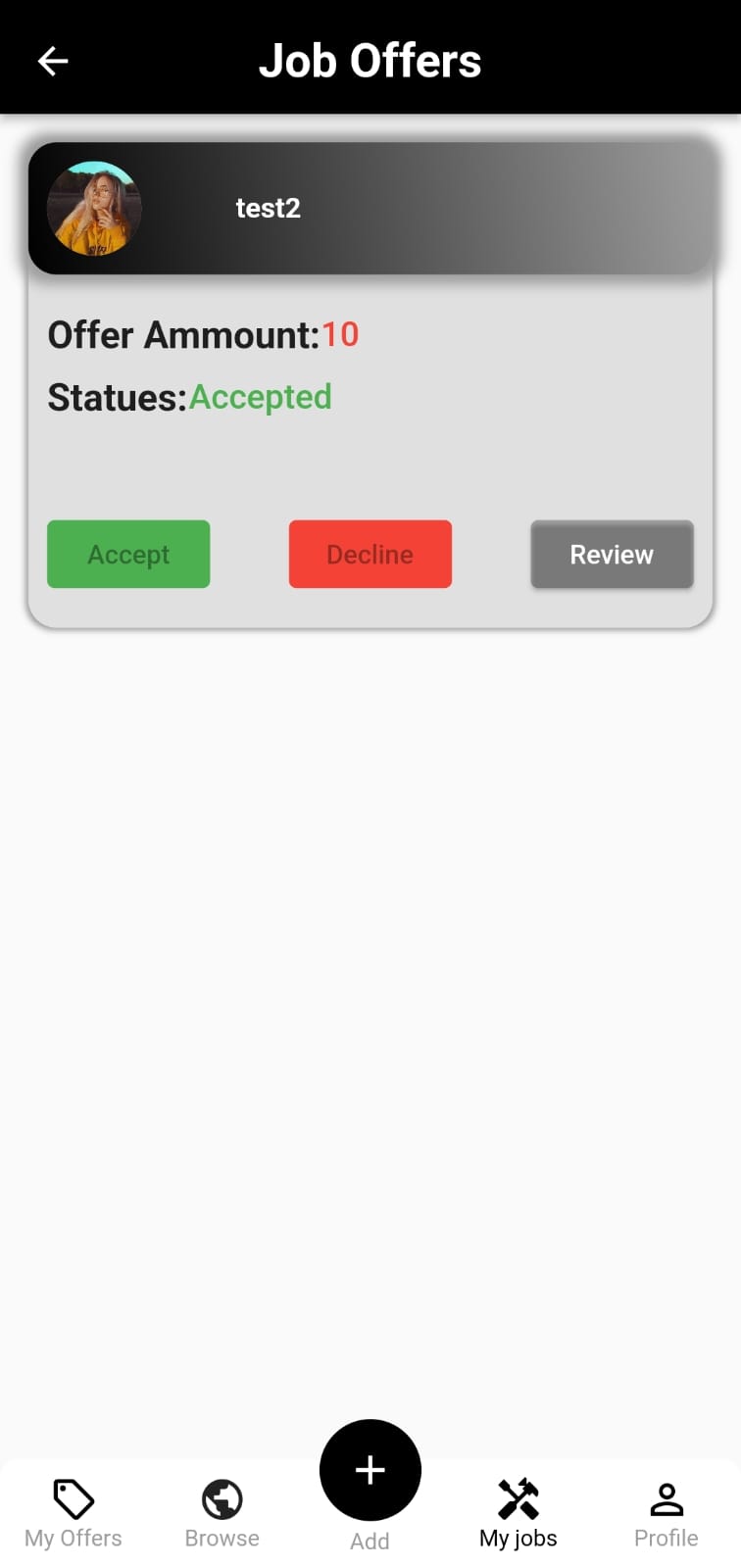
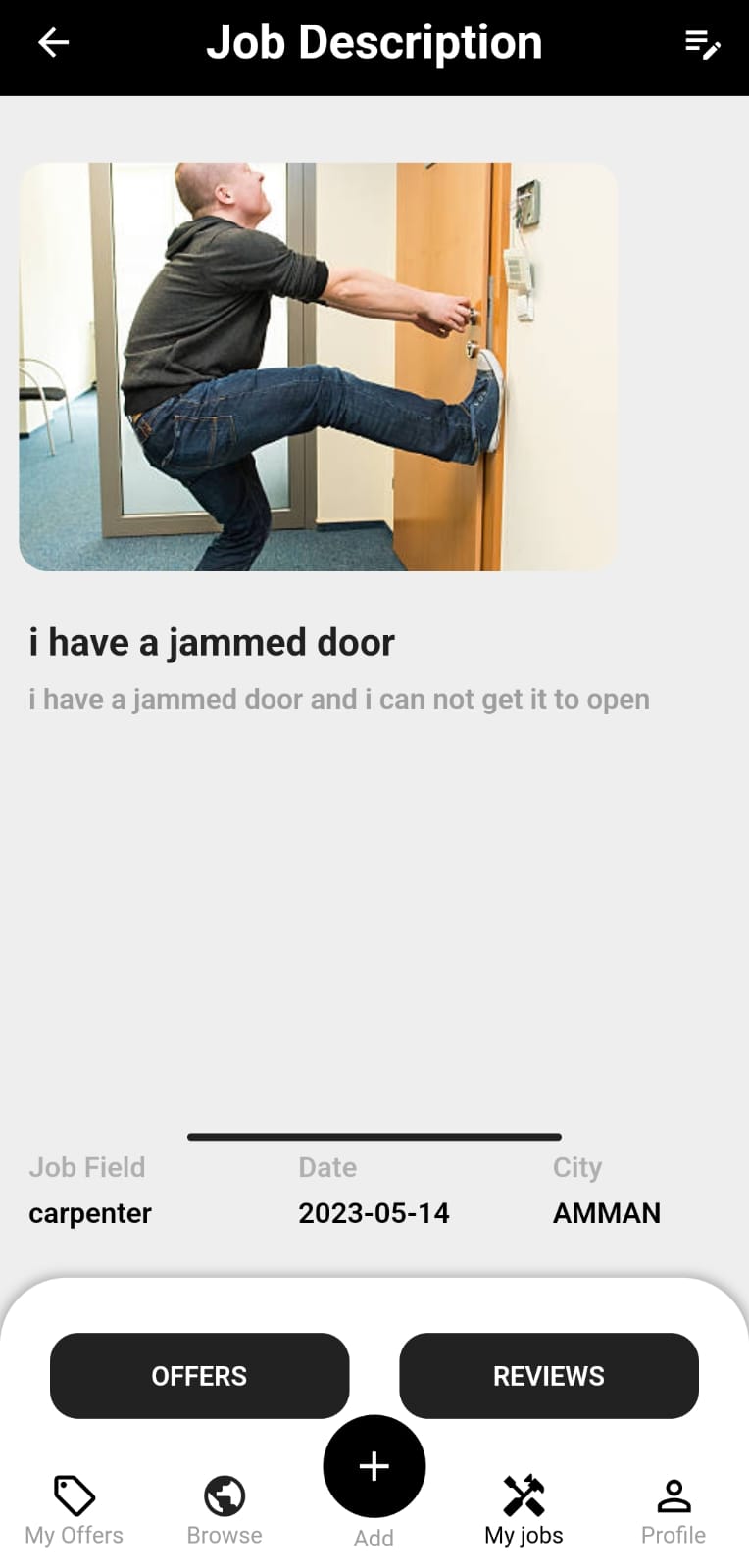


As you can see this user only got 1 review

2.reviews as job taker

So firstly how make a review for a person that completed a job for me .

1.from my jobs , you choose the job and check the offers , then press on the red circled button called review

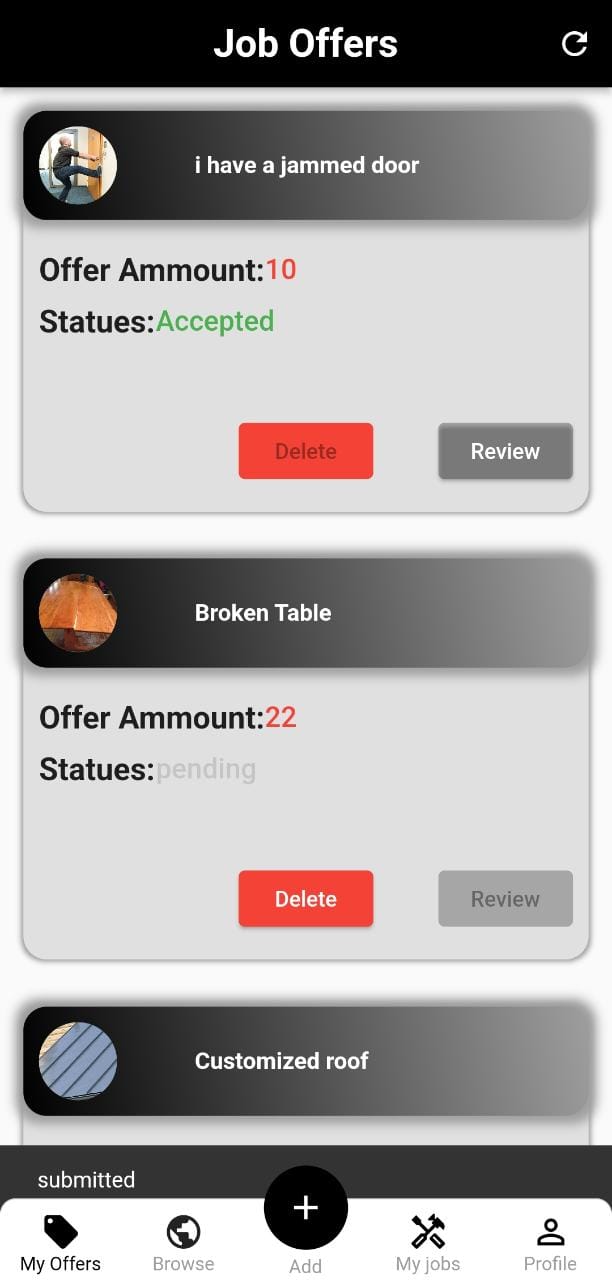


1.Job description 2.Review from job offer 3..submitting a review

After submitting the review will be available at the users profile and the stars count will change according to the average of the reviews the user have.

Secondly, if you want to make a review for the job lister after completing his request.

You tap on my offers and from the offer you made to his job request , press on review .

A screenshot of a phone

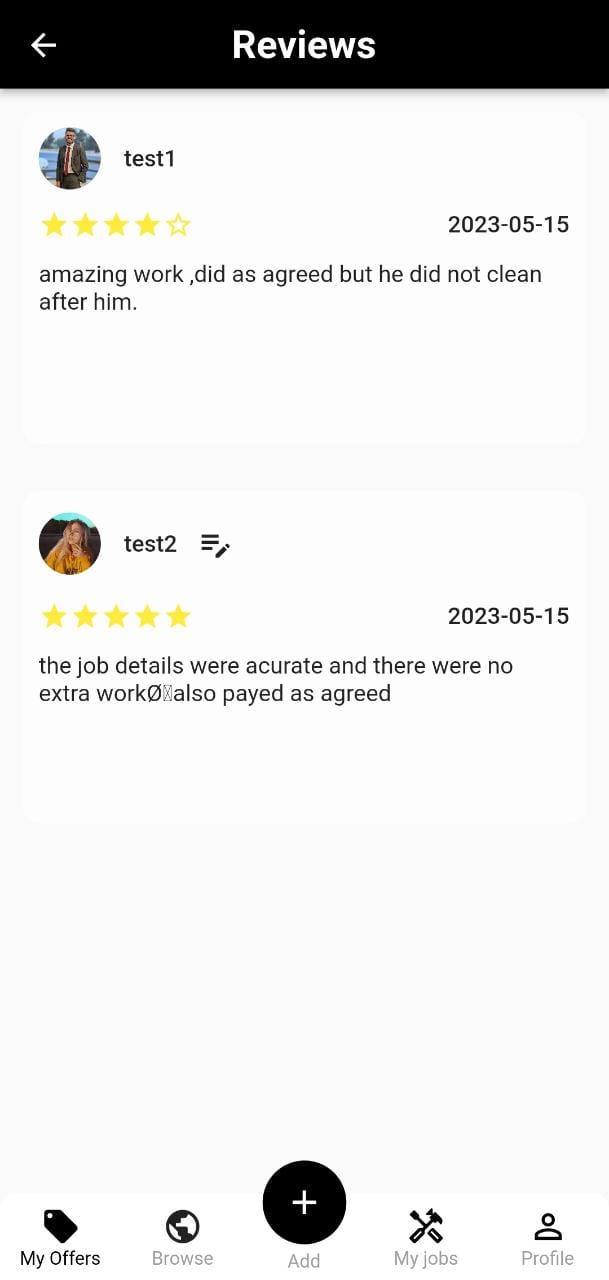
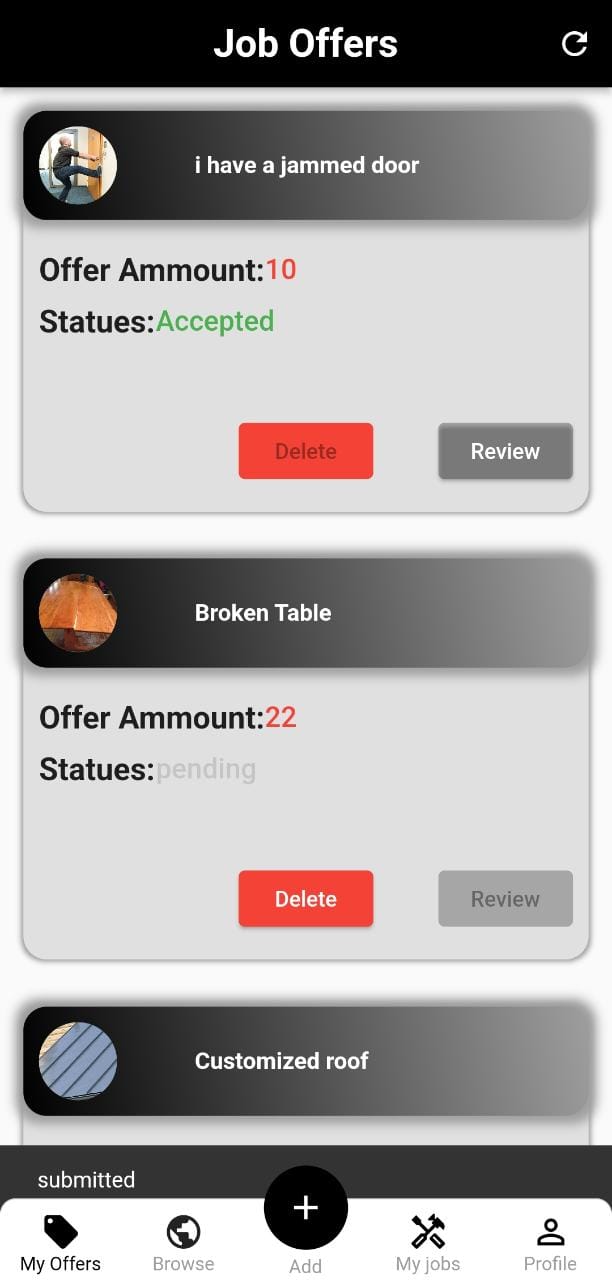
Description automatically generated with low confidence

1.review button 2.submitting a review

After submitting the review will be visible at the job listers profile in the (as job lister reviews )

thirdly ,as the job taker you want to know what the job lister reviewed you for a certain job.

From my offers tap on the marked red bar.



1.My offers 2.both reviews regarding the job

By tapping on the red marked bar a screen containing both reviews

Job takers review and the job lister review.

As you noticed , the blue circled icon in figure 2 lets you edit your review by tapping on it and opens the screen below.

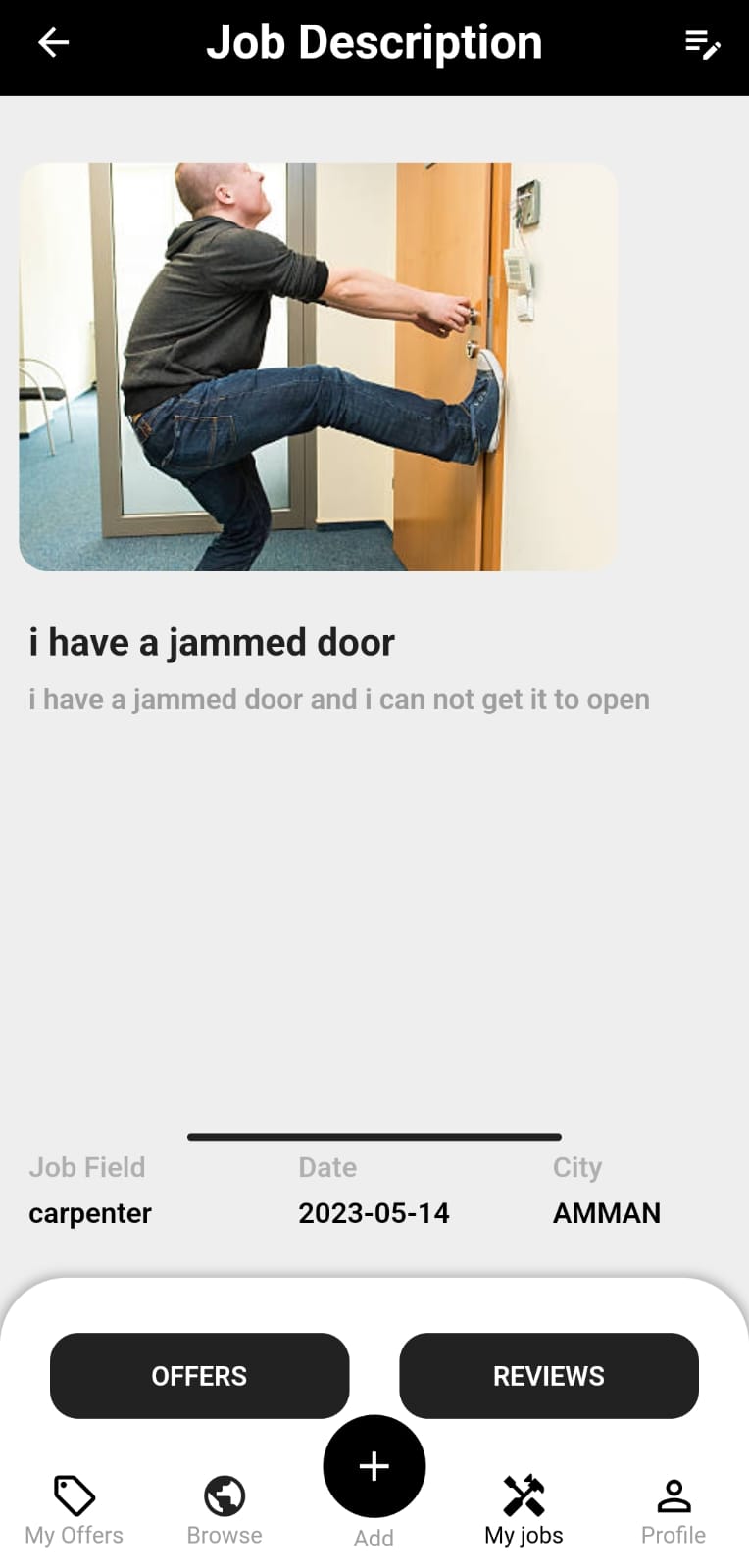
A screenshot of a phone

Description automatically generated with low confidence

Change the written text or the number of stars and submit.

The edited review will be shown instead.

Also if you want to see the reviews regarding a certain job you listed.

A screenshot of a phone

Description automatically generated with medium confidence

By tapping on the reviews button it will also take you to the same page as before and you can also edit your review that you made on the job taker.

**Chapter 4**

**Conclusion:**

**We have built an user friendly app that will help any one with a craft or knowledge in any area reach more people and gain financial benefits ,assuring no scam happens, also benefitting the people in need for craftsmen to complete a certain job requested .**

**Limitations :**

**We have some limitations in our system because we have low hardware capabilities.**

**1.our number of users are limited because we use a pc as our server .**

**2.we can not confirm if the job is really completed before making a review.**

**3.on frontend level the app could be designed and implemented better ,but having to learn designing and implementing using flutter and completing the project at the same time took some group effort.**

**Future work :**

1. **We plan on using OTP**

**OTP stands for "One-Time Password." It is a unique and temporary password or code that is typically used for authentication purposes, especially in online services, apps, or systems that require an additional layer of security.**

**It's important to note that OTP implementation may vary depending on the specific system or application you are using. Some systems may use mobile apps for OTP generation, while others rely on third-party services for OTP delivery.**

**Our reasons to use it are because of a user might forget his password also for email authentication.**

1. **We plan to make the user chose between two types of users :**

**1.Individual :for users who only work by themselves**

**2.organaization : for users with more than one person**

**working for them.**

**3.maps**

**Using maps in your app offers several benefits:**

**1.Enhanced User Experience: Incorporating maps into your app provides users with a visually appealing and interactive experience. Maps allow users to view and navigate geographic locations, providing context and relevance to the information or services your app offers.**

**2.Location-based Services: Maps enable the integration of location-based services, such as real-time navigation, geolocation tracking, or proximity-based recommendations. Users can easily find nearby points of interest, businesses, or events based on their current location.**

**3.Improved Wayfinding and Directions: Maps help users navigate from one location to another efficiently. By providing directions, turn-by-turn navigation, and estimated travel times, your app can assist users in reaching their destinations accurately and conveniently.**

**Overall, incorporating maps into your app enhances usability, provides valuable location-based services, and enriches the overall user experience. It opens up possibilities for a wide range of applications, including travel, navigation, e-commerce, social networking, and more.**

**References:**

1. **https://pub.dev/**
2. <https://developer.android.com/studio>
3. <https://flutter.dev/>
4. <https://dart.dev/>
5. <https://ubuntu.com/>
6. <https://www.docker.com/>
7. <https://www.mysql.com/>
8. <https://en.wikipedia.org/wiki/Object%E2%80%93relational_mapping>
9. <https://telebit.cloud/>
10. <https://en.wikipedia.org/wiki/Jakarta_Servlet>
11. <https://tomcat.apache.org/>
12. <https://en.wikipedia.org/wiki/Base64>
13. <https://www.eclipse.org/>
14. <https://maven.apache.org/>
15. <https://spring.io/>
16. <https://www.java.com/en/>
17. <https://cloudinary.com/>
18. <https://projectlombok.org/>
19. <https://www.baeldung.com/spring-mvc-tutorial>
20. <https://en.wikipedia.org/wiki/Data_transfer_object>
21. <https://mapstruct.org/>
22. <https://www.baeldung.com/spring-boot-bean-validation>
23. <https://jwt.io/>
24. <https://en.wikipedia.org/wiki/HTTPS>
25. <https://en.wikipedia.org/wiki/Bcrypt>
26. <https://en.wikipedia.org/wiki/Aspect-oriented_programming>
27. <https://en.wikipedia.org/wiki/QA/QC>
28. <https://en.wikipedia.org/wiki/SQL>
29. <https://en.wikipedia.org/wiki/Apk_(file_format)>
30. https://en.wikipedia.org/wiki/JAR\_(file\_format)