Chapter 1

INTRODUCTION

The Android Based Furniture Shopping Application is an android application for a particular company that allows users to check out for various furniture available in the store and can even purchase them online. Android Based Furniture Shopping Application contains a wide variety of products with high quality and benefit like:-

- There are no national and international barriers.
- Here the consumers will be in a demanding position and suppliers will not be in command position.
- There are enormous employment opportunities in this system.

The application provides a categorized list of furniture products of different styles and models like sofa, wall drop, Tv stand, bedside, coffee table, dining table, bed set, dressing table, chair etc. The company is planning to add the latest models to the catalog at best prices and offers making it for any online shoppers looking for furniture. Users have to login using the same username and password used at the time of registration into the application for browsing each product and can then add them into their cart. After selecting the products users can make secure online payment via credit card or cash on delivery. Along with buying the items the application included an additional feature called customization. The clients can give request to the dealer or admin with some particular design or model with their suitable measurements. The clients can buy the accepted customized products through any online payment mode. Thus the android based furniture shopping application project brings an entire furniture shop online and makes it easy for both buyer and seller to make furniture deals. Sql serves as a backend to store furniture lists and inventory data. Thus the Android based furniture shopping application project brings an entire furniture shop online and makes it easy for both buyer and seller to make furniture deals.

Chapter 2

PROOF OF CONCEPT

2.1 EXISTING SYSTEM

In the existing system everything is done manually. That is the client need to go to the shop directly and buy the products and need to pay direct to the dealer. In this the user's needs to search items physically shop by shop if there is more than one shop for the particular company it is not feasible for customer. So we need to be converted into automated system to reach out among large number of peoples easily. The Existing system have drawbacks like

- The existing system works manually
- Less availability of products
- Low proper coordination
- Human effort is maximum
- Time consuming
- Difficult to find products
- Cost consuming
- Need to search items shop by shop which is not feasible
- Not accurate
- Difficult to reach distant user
- All modes of payment are not available

2.2 PROPOSED SYSTEM

The proposed system is to automate the existing system. The proposed system provided advance features on the interface for better functioning and efficiency of the users. The admin interface gets all the features to control and manage whole system, and can manage the accounts of the customers. The customer gets the option to view the already available furniture's and can by them online. Customer can request customized build furniture option in which is subdivided into category according to the size, design, type, etc. the system estimates the price and show it to the customer's interface to give him an idea and then the customer can pay the charges online through the system. After confirming the order by the customer the products can be tracked. The products have been delivered to the address of the customer given at the time of booking. The customers can give feedback and rating about the product after the delivery of the product.

The most important functions of this system are:

- User interface
- Faster access to products
- Easy and flexible to use
- Easy to access
- Efficient traceability
- Availability of previous data for future reference
- Save time and money
- Large data can be accessed in small time

2.3 SYSTEM SPECIFICATIONS

A software requirement specification (SRS), a requirements specification for a software system, is a complete description of the behaviour of a system to be developed and may include a set of use cases that describe interactions the users will have with the software. The SRS is a technical document which provides a framework for the software development process. It provides an overview of the software or application including what it should do and what its parameters are, how it will interact with its environment and the end users, and its hardware and software requirements. The SRS should be able to be read and understood by all the parties involved in the production of the software, including clients, designers, coders, testers and end users. The functional requirements are concerned with the way the software behaves in response to the client's or business needs, and should be readily accessible to client or end user. In addition, it also contains non-functional requirements. Non-functional requirements impose constraints on the design or implementation (such as performance engineering requirements, quality standards, or design constraints) The software requirements specification document enlists all necessary requirements that are required for the project development. To derive the requirements, we need to have clear and thorough understanding of the products to be developed.

2.4 SOFTWARE SPECIFICATIONS

The selection of software is very important in the existence and proper working of any software. When selecting software, the size and capacity requirements are also important. Below are some of the software that is required for the system. Software specification is a formal report, which acts as a representation of software that enables the customers to review whether it is according to the requirements. Also, it comprises user requirements for a system as well as detailed specifications of the system requirements. The software specification is a specific software product, program, or set of applications that perform particular functions in a specific environment. It serves several goals depending on who is writing it. First, the software specification could be written by the client of a system. Second, the it could be written by a developer of the system. The two methods create entirely various situations and establish different purposes for the document altogether. The first case, software specification is used to define the needs and expectation of the users. The second case, it is written for various purposes and serves as a contract document between customer and developer. The software specification used in this project includes:

Platform	Android Studio
Programming Language	Java, MYSQL, PHP
RDBMS	MYSQL
Web server	XAMPP

Table No:1 Software specifications

2.5 HARDWARE SPECIFICATIONS

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. I require much different software to make the application which is in making to work efficiently. It is very important to select the appropriate software so that the software works properly. The hardware specification used in the project are:

Main Processor	Intel core i3 or above
RAM	8 GB or above
Keyboard	Standard 108 keys
Mouse	3D Optical mouse
Monitor	15’ Standard
Hard disk	10 GB of available disk space minimum or above

Table No: 2 Hardware specifications

Chapter 3

IMPLEMENTATION

The implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented. Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system can be implemented only after testing is doneand is found to be working to specifications. The implementation stage is a systems project. The implementation stage involves following tasks:

- Test system with sample data
- Detection and correction of errors
- Make necessary changes in the system
- Check with the existing system
- Installation of hardware and software utilities
- Training and involvement of user personnel

3.1 TOOLS AND PLATFORM

PLATFORM: ANDROID

Android is a Linux based operating system designed primarily for touch screen mobile devices such as smart phones and tablet computers. Android is open source and Google releases the code under the Apache License This open source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. Additionally, Android has a large community of developers writing applications that extend the functionality of devices, written primarily in a customized version of the Java programming language. Android is an open source mobile operating system that combines and builds up on parts of many different open source projects. Android’s kernel is based on the Linux kernel and has further architecture changes by Google outside the typical Linux kernel development cycle. Android does not have a native Window System nor does it support the full set of standard GNU libraries, and this makes it difficult to port existing Linux applications or libraries to Android. Android’s user interface is based on direct manipulation using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching and reverse pinching to manipulate on screen objects. Android devices boot to the home screen, the primary navigation and information point on the device, which is similar to the desktop found on PCs. Android home screens are typically made up of applications and widgets applications launch the app, whereas widgets display live, auto updating content such as theweather forecast, the user’s email inbox, or a news ticker directly on the home screen.

FRONT END: JAVA

Java is a general purpose computer programming language that is concurrent, class based, object oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers write once,run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to byte code that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2016, Java is one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling, a Canadian, at Sun Micro systems (whichhas since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Micro systems Java platform. The language derives much of its original features from Small talk, with syntax similar to C and C++, but it has fewer low-level facilities than either of them.

The latest version is Java 11, released on September 25, 2018. Java 11 is a currently supported long term support (LTS) version (“Oracle Customers will receive Oracle Premier Support”). Oracle released for the legacy. Java 8 LTS public update which is free for commercial use in January 2019. Oracle will still support Java 8 with public updates for personal use up to at least December 2020. Oracle is highly recommends that you uninstall older versions of java because of serious risks due to un-resolved security issues. Since Java 9 is no longer supported, Oracle advises its users to immediately transition.

BACKEND: MYSQL, PHP

ABOUT MYSQL

MySQL is an open source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. It is free and open source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.

MySQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database driven web applications, including Drupal, Joomla, phpBB, and WordPress. It is also used by many popular websites including Facebook, Flickr, MediaWiki, Twitter and YouTube etc.

MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home-brewed lexical analyzer. MySQL works on many system platforms, including AIX, BSDi, FreeBSD, HP- UX, eComStation, i5/OS, IRIX, Linux, MACOS, Microsoft Windows,NetBSD,

Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Oracle Solaris, Symbian, SunOS, SCO Open Server, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists. The MySQL server software itself and the client libraries use dual-licensing distribution. They are offered under GPL version 2, or a proprietary license. Support can be obtained from the official manual. Free support additionally is available in different IRC channels and forums. Oracle offers paid support via its MySQL Enterprise products. They differ in the scope of services and in price. Additionally, a number of third party organisations exist to provide support and services, including MariaDB and Percona.

MySQL has received positive reviews, and reviewers noticed it "performs extremely well in the average case" and that the "developer interfaces are there, and the documentation (not to mention feedback in the real world via Web sites and the like) is very good". It has also been tested to be a "fast, stable and true multi-user, multi-threaded sql database server".

MySQL was created by a Swedish company, MySQL AB, founded by David Axmark, Allan Larsson and Michael "Monty" Widenius. Original development of MySQL by Widenius and Axmark began in 1994. The first version of MySQL appeared on 23 May 1995. It was initially created for personal usage from MySQL based on the low-level language ISAM, which the creators considered too slow and inflexible. They created a new SQL interface, while keeping the same API as MySQL. By keeping the API consistent with the MySQL system, many developers were able to use MySQL instead of the (proprietarily licensed) MySQL antecedent. The OLAP Services feature available in SQL Server version 8.0 is now called SQL Server 2008 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component.

The Repository component available in SQL Server version 8.0 is now called Microsoft SQL Server 2008 Meta Data Services. References to the component now use the term Meta Data Services. The term Repository is used only in reference to the repository engine with in Meta Data Services to Java 11. Oracle extended support for Java 6 ended in December 2018.

ABOUT PHP

The term PHP is an acronym for PHP: Hypertext Preprocessor. PHP is a server-side scripting language designed specifically for web development. PHP is one of the most widely used server-side scripting language for web development. Popular websites like Facebook, Yahoo, Wikipedia etc. are developed using PHP.

PHP is so popular because it's very simple to learn, code and deploy on server, hence it has been the first choice for beginners since decades. PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge. The PHP language evolved without a written formal specification or standard until 2014, with the original implementation acting as the de facto standard which other implementations aimed to follow. Since 2014, work has gone on to create a formal PHP specification.

As of September 2019, over 60% of sites on the web using PHP are still on discontinued "EOLed" version 5.6 or older versions prior to 7.1 are no longer officially supported by The PHP Development Team, but security support is provided by third parties, such as Debian.

FEATURES OF PHP

PHP development began in 1994 when Rasmus Lerdorf wrote several Common Gateway Interface (CGI) programs in C, which he used to maintain his personal homepage. He extended them to work with web forms and to communicate with databases, and called this implementation "Personal Home Page/Forms Interpreter" or PHP/FI. PHP/FI could be used to build simple, dynamic web applications. To accelerate bug reporting and improve the code, Lerdorf initially announced the release of PHP/FI as "Personal Home Page Tools (PHP Tools) version 1.0 " on the Usenet discussion group compute to the infosystems. www.authoring.cgi on June 8, 1995. This release already had the basic functionality that PHP has today. This included Perl-like variables, form handling, and the ability to embed HTML. The syntax resembled that of Perl, but was simpler, more limited and less consistent features are:

- Simple
- Simplicity
- Efficiency
- Platform independent
- Case sensitive
- Faster
- Interpreted

Chapter 4
SYSTEM DESIGN

System design is an interactive process through which requirement are transmitted to a “blue print” for constructing the software initial the blue print depicts a holistic view of software that is design is represented at a high-level abstraction a level that can be directly traced to specific data, functional and behavioural requirement. System design is the solution to the creation of a new system. This is the important aspect made up of several steps. System design is the process of developing specifications for a candidate system that meet the criteria established in the system analysis. Major step in system design is the preparation of the input forms and output reports in a form applicable to the users.

The main objective of system design is to use the package easily by a computer operator. System design is the creative act of invention, developing new inputs, a database, off line files, method, procedures, and output for processing business to meet an organization objective. System design built information gathered during the system analysis. As design interaction occur subsequent refinement leads to design representation at much lower level of abstraction. System design is a creative art of inventing and developing input, data bases, off line files, method and procedures, for processing data to get meaning full output that satisfy the organization objectives. Through the design phase consideration to the human factor, that is inputs to the users will have on the system.

4.1 INPUT DESIGN

Input design is the process of converting user-oriented input to a based format. Inaccurate input data are the most common cause of errors in data processing. Errors entered by data entry operators can be controlled by input design. The goal of designing input data is to make data entry as easy, logical, and free from errors. When we approach input data design we design the data source documents that capture the data and then select the media used to enter them into computer.

User-friendly screen format can reduce the burden on end users, who are design of source document. Source document is the form in which the data can initially capture. The next step is the design of the document layout. In the layout organizes the document by placing information, where it will be noticed and establishes the appropriate sequence of items. User interface design is very important for any application. The interface design describes how the software communicates within itself, to system that interpreted with it and with humans who use it. The input design is the process of converting the user-oriented inputs into the computer based format. Input design is a part of overall system design, which requires very careful attention. If data going into the system is correct, then the processing and output will magnify these errors. Thus, the designer has several clear objectives in the different stages of input design:

1. To produce a cost-effective method of input.
2. To achieve the highest possible level of accuracy.
3. To ensure that input is acceptable to and understand by the user.

In accurate input data is most common cause of data processing errors.If poor input design particularly where operators must enter data from source documents permits bad data to enter a computer system, the outputs produced are of little value. The input design process was initiated in the study phase were, as a part of the feasibility study:

1. Input data were found to be available for establishing and maintaining master and transaction files and for creating output records
2. The most suitable types of input media for either off-line or on-line devices were selected after a study of alternative data capture techniques

4.2 OUTPUT DESIGN

The data is fed into the system using simple inactive forms. The forms have been supplied with messages so that the user can enter data without facing any difficulty. This data is validated where verity requires in the project. This ensures that only the correct data have been incorporated into system. The goal of designing input data is to make the automation as easy and free from errors as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirements such as user friendlines, consistent format and interactive dialogue forgiving the right messages forthe user at the right time are also considered for development for this project. Computer output is the most important and direct source of information to the user decision-making.

In the output design it is determine how the implementation is to be played for immediate need and also the hard copy output. A major form of input is a hard copy from Printers, CRT screen display are the examples for providing computer based output. The output design associated with the system includesthe various reports of the table generations and query executions. A quality outputis one, which meets the requirements of end user and presents the in- formation clearly. In any system result of processing are communicated to the user and to the other system through outputs. In the output design it is determined how the in-formation is to be displayed for immediate need. It is the most important and directsource in formation to the user. Efficient and intelligent output design improves the system's relationships with the user and helps in decision making.

The objective of the output design is to convey the information of all the past activities, current status and to emphasis important events. The output generally refers to the results and information that is generated from the system. Outputs from computers are required primarily to communicate the results of processing to the users. Output design is one of the most important features of the information system. The logical design of an information system is analogous to an engineering blue print of an automobile. It shows the major features and how they are related to one another. The outputs, inputs and databases are designed are in this phase. At the beginning of the output design various types of outputs suchas external, internal, operational, and interactive and turnaround are defined. Then the format, content, location, frequency, volume and sequence of the outputs are specified. The content of the output must be defined in detail. The system analysis has two specific objectives at this stage.

- ➢ To interpret and communicate the results of the computer part of a systemto the users in a form, which they can understand, and which meets their requirements.
- ➢ To communicate the output design specifications to programmers in a way in which it is unambiguous, comprehensive and capable of being translatedinto a programming language.

4.3 DATABASE DESIGN

The Database design is the process of producing a detailed data model of a database. The logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributes or each entry. The term database design can be used to describe many different parts of the design of an overall data base system. Principally, and most correctly, it can be though to fast the logical design of the base data structures used to store the data. In the relational model, these are the tables and views. In an object data base, the entities and relationships map directly to object classes and named relationships. However, the term data base design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall data base applications within the database management system.

4.4 USECASE DESIGN

A use case diagram is used to represent the dynamic behaviour of a system. It encapsulates the system’s functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system. The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system’s requirement, which includes both internal as well as external influences.It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system. It is essential to analyse the whole system before starting with drawing a use case diagram, and then the system’s functionalities are found. And once every single functionality is identified, they are then transformed into the use cases to be used in the use case diagram. After that, we will enlist the actors that will interact withthe system. The actors are the person or a thing that invokes the functionality ofa system. It may be a system or a private entity, such that it requires an entity to be pertinent to the functionalities of the system to which it is going to interact. Once both the actors and use cases are enlisted, the relation between the actor and use case or system is inspected. It identifies the no of times an actor communicates with the system. Basically, an actor can interact multiple times with a use case or system at a particular instance of time. These are the purposes of a use case diagram:-

- It gathers the system’s needs.
- It depicts the external view of the system.
- It represents the interaction between the actors.

4.5 USECASE DIAGRAM

USECASE DIAGRAM OF ADMIN:

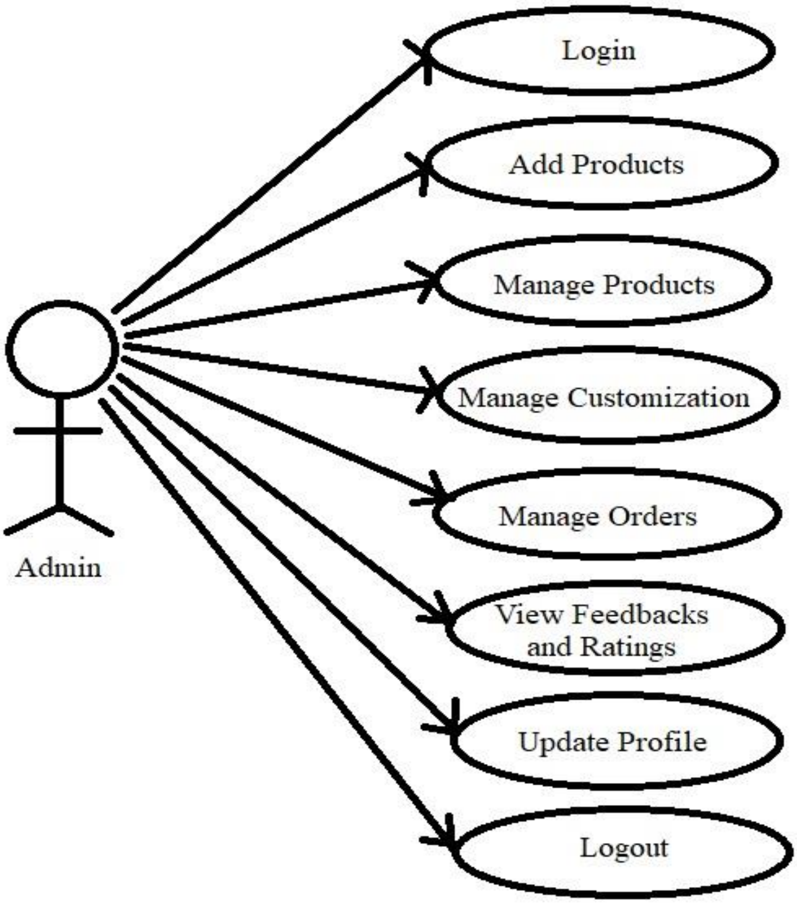


Figure No: 1   Use case diagram od admin

USECASE DIAGRAM OF USER:

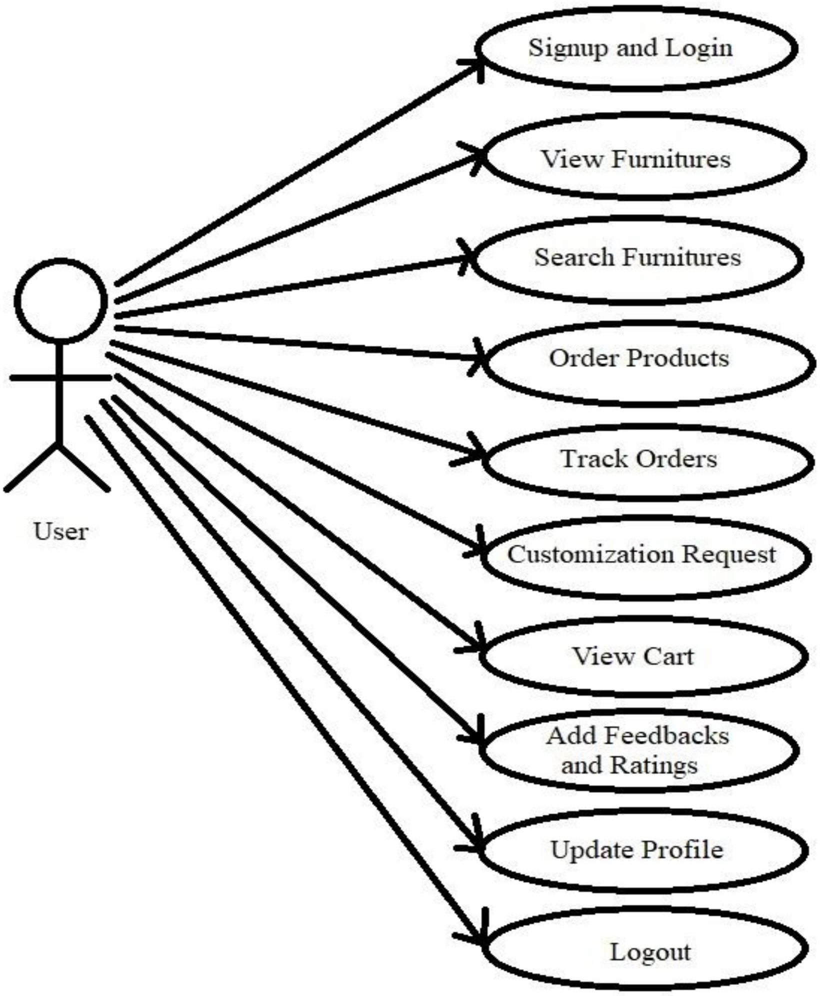


Figure No: 2 Use case diagram for user

Chapter 5

SYSTEM DEVELPOMENT

5.1 MODULES

5.1.1 REGISTRATION AND LOGIN

Admin will login into the application. Admin can manage all the activities. User can create an account and login to the application using the username and password given at the time of registration.

5.1.2 CART MANAGEMENT

Manages the user’s cart and manages the items ordered by users.

5.1.3 PRODUCT MANAGEMENT

Admin can add new items based on the category’s**.**

5.1.4 CUSTOMIZATION MANAGEMENT

Manages the customization of product requested by the user and the user can order the item if the admin accept the request.

5.1.5 WISHLIST MANAGEMENT

Can mark the items liked by the user as wishlist.

5.1.6 RATING AND FEEDBACK MANAGEMENT

The user can give feedback and ratings. Admin can view the feedbacks and ratings.

5.1.7 ACCOUNT MANAGEMENT

Admin and user can manage there account by editing username, phone number etc.

5.1.8 PAYMENT MANAGEMENT

Manages the payment done by the user while purchasing the product through online mode.

Chapter 6
SYSTEM IMPLIMENTATION

6.1 TESTING

Testing is an important stage in the software development life cycle. System testing is a critical element of a software quality assurance and representsthe ultimate review of specification, design and coding.

Importance of software testing and its implication with software quality cannot be over emphasized. Testing is one-way developers can validate the quality of a software product and verify that it fully meets the specification. During testing, the system is tested with a set of cases and checked whether the input of the program is performing as it is expected. The system tested and reviewed toensure that the entire user requirement has being satisfied.

Testing was done throughout the system development at various stages since it is always a good practice to test the system at many different levels at various intervals that is sub systems, program modules as work progress and finally the system as a whole. If this is not done, then the poorly tested system can fail after installation. Testing is a very tedious and time consuming job. For a test to be successful the tester should try and make the program file. Each test is designed with the intention of finding errors in the way system will process it. Though testing of a program doesn’t guarantee the reliability of the system, it is done to assure that the system runs errors free. The Testing process begins by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The Process verifies that the application meets the requirements specified inthe system requirements document and is bug free.

At the End of each testing day, the summary of completed and failed tests is prepared. And the Application is redeveloped and retested until every itemis resolved. All the changes and retesting are tracked through spread sheets. Applications are not allowed to the launch until all identified problem are fixed. Finally,a report is prepared at the end of testing to show exactly what was tested and to listthe final outcomes. The software testing methodology is applied in four distinct phases:

- Unit Testing
- Integration Testing
- User Acceptance Testing
- Output Testing

6.1.1 UNIT TESTING

Developers typically do unit testing in order to trace out bugs in each module of the code. Unit testing is done in parallel with coding. It includes testing each function and procedures. Unit testing is also called as module testing. In module testing each module are tested for any possible logical error. They are also tested for specification to see if they are working as per the program should do and they are tested under various conditions. Each module is being tested thoroughly in order to discover pitfalls. Specification testing examines the specification what the program should do and how it should perform under various conditions. The testing will be done by entering data into different tables using forms. The data with less validation will be tested first. Whenever an error is encountered, an informative error message will be displayed which informs user about the type of error. After the completion of form testing the program will betested. The unit testing is done to identify

- The image entries are in the correct format.
- No duplicate entries are present.
- To check whether it provide the required result.

6.1.2 INTEGRATION TESTING

Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together (” big bang”). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). The task of the integration test is to check that components or software applications, interacts without error. Therefore, testing the data flow between 2 modules is integration testing.

6.1.3 USER ACCEPTANCE TESTING

User acceptance testing is done by the user to check whether the project has met the requirement that has been mentioned at the beginning of the project. Flood alert is tested by the user by inputting values and the result generated is also validated. If the accuracy is as expected user approve the system.

6.1.4 OUTPUT TESTING

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it does not produce the required output in the specific format. The output generated or displayed by the system under consideration is tested asking the users about the format required by them. Here, the output is considered in two ways, one is on the screen and other is printed format. The output format on the screen is found to be correct as the format designed according to the user needs. For the hard copy also, the output comes out as specified by the user. Hence output testing does not result in any connection in the system.

6.2 VALIDATION CHECK

The validation phase reveals the failures and the buds in the developed system. It will become to known about the practical difficulties the system faces when the operated in the true environment. Validation is the process of ensuring that user input is clean, correct, and useful. Typical validation tasks are:

- Has the user filled in all required fields?
- Has the user entered a valid email?
- Has the user entered text in a numeric field?

Form validation normally used to occur at the server, after the client had entered all the necessary data and then pressed the submit button. If the data entered by a client was incorrect or was simply missing, the server would have to send all the data back to the client and request that the form be resubmitted with correct information. This was really a lengthy process which used to put a lot of burden on the server. Most often, the purpose of validation is to ensure correct useinput. Validation can be defined by many different methods and deployed in many different ways. Server side validation is performed by a web server, after input has been sent to the server. Client side validation is performed by a web browser, before input is sent to a web server.

Chapter 7

SYSTEM MAINTENANCE

System maintenance is a going activity, which covers a wide variety of activities including, removing program and design errors, updating documentation and test data and updating user support system maintenance is a catchall termused to describe various forms of computer or server maintenance required to keep a computer system running properly, it can describe network maintenance which could mean that servers are being physical repaired, replaced or mode. For the purpose of convenience, maintenance may be categorized into three classes they are:

7.1 CORRECTIVE MAINTENANCE

This type of maintenance implies removing errors in a program,which might have kept in the system due to faulty design or wrong assumption. Corrective maintenance is a set of technical tasks designed to repair equipment failures as there is a need to fix them or even replace them. This type of maintenance is used to correct errors in the equipment which needs to be serviced to restore its initial function. The corrective maintenance model is always necessary and can be applied to equipment with a low-level of criticality whose failures do not interfere with the company’s productivity.

7.2 ADAPTIVE MAINTENANCE

In adaptive maintenance program functions are changed to enable the information system to satisfy the information needs of the user. Adaptive maintenance refers to the enforcement of changes in the monitoring, use or other operational details of a metallic structure or object to prevent corrosion from spreading from one part of the metal where it is already present to another. adaptive maintenance involves upgrading corrosion monitoring software systems to adjust the parameters the software recognizes as corrosion-inducing properties. The better the software is at reading such properties, the better the degree of corrosion prevention in a given application. Adaptive maintenance consists of changing, often self-regulating software that monitors changes in an external environment. The term environment in this context refers to the conditions that influence corrosion.

7.3 PERFECTIVE MAINTENANCE

In perfective maintenance means adding new programs or modifyingthe existing programs to enhance the performance of the information system. Thistype of maintenance under taken to respond to user addition needs which may bedue to the changes within or outside of the organization. This maintenance strategy is carried out on a regular basis, and this means that the equipment is inspected even if there are no signs of failure. This way, any equipment failure is avoided as much as possible to make sure the proper functioning and safety of the assets. Preparing preventive maintenance plans for equipment and facilities means preventing breakdowns and reducing the probability of equipment failures that interfere with the companies’ performance. This maintenance routine is programmed using maintenance plans, which allows the responsible department to control operations and to know in advance which parts or resources are required to make sure a suitable preventive operation.

Chapter 8

CONCLUSION

The project entitled “Android Based Furniture Shopping Application” was completed successfully. The android application has been developed with much care and free of errors and it is efficient. The purpose of this android application for purchasing items available on the shop. Its exclusively developed for shopping furniture with a wide variety of latest models. This application allows users to browse the item, add items to cart and order the products for shipment by paying online. The user can mark the products liked as wish list. Along with this there is a feature called customization where user can request the owner for a particular product and can even buy them if the request is accepted by the admin. The Android Based Furniture Shopping Application is designed for people to shop online rather than searching for furniture by visiting the shop.

This project helped to understand about the development phases of a project and software development life cycle. This project helped to gain valuable information and practical knowledge on designing android application and management of database using mysql. The project was tested with different features of project. The project has been developed for implementing on a particular company for selling various kinds of products in the store. There is a scope for further development in the project to a great extend.

Chapter 9

FUTURE SCOPE

The Android Based Furniture Shopping Application can extend its services using GUI and augmentation on apps for apple and android users, this leads to an increase in the user base. It may also lead to features like sending automatic emails to registered users when there is any clearance or discount sales and price drop on any item. The users could subscribe for price alerts which would enable them to receive messages when price for products fall below a particular level. Work on enhancing the user interface by adding more user interactive features.

Chapter 10

CODING

9.1 SOURCE CODE

Registration.java

```
package com.example.Furniture.User;


import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;


import androidx.appcompat.app.AppCompatActivity;


import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import com.example.Furniture.Config;
import com.example.Furniture.OptionActivity;
import com.example.Furniture.R;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.HashMap;
import java.util.Map;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Registraction extends AppCompatActivity {

    EditText UserName,MobileNumber,Emailid,PassWord,location;
    Button register;
    String url= Config.baseURL+"registerer.php";
    String status,error,type;

    String U,M,E,P,locs;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registraction);
        UserName=findViewById(R.id.userreg);
        MobileNumber=findViewById(R.id.phonereg);
        Emailid=findViewById(R.id.emailreg);
        PassWord=findViewById(R.id.passreg);
        register=findViewById(R.id.buttonreg);
        location=findViewById(R.id.location);

        Intent intent=getIntent();
        type=intent.getStringExtra("User");
        register.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                register();
            }
        });
    }
    private void register(){

        U=UserName.getText().toString();
        M=MobileNumber.getText().toString();
        E=Emailid.getText().toString();
        P=PassWord.getText().toString();
        locs=location.getText().toString();

        if (P.isEmpty() || P.length() < 6) {
            PassWord.setError("Password cannot be less than 6 characters!");
        }
        if (TextUtils.isEmpty(U)){
            UserName.setError("Please enter username");
            UserName.requestFocus();
            return;
        }
        else if(TextUtils.isEmpty(M)){
            MobileNumber.setError("Please type your mobile number");
            MobileNumber.requestFocus();
            return;
        }
        else if(!isPhoneValid(M))
        {
          MobileNumber.setError("Invalid number");
          MobileNumber.requestFocus();
          return;
        }
        else if (TextUtils.isEmpty(E)){
          Emailid.setError("Please enter your email id");
          Emailid.requestFocus();
          return;
        }
        else if(!isEmailValid(E)){
          Emailid.setError("Invalid email");
          Emailid.requestFocus();
          return;
        }
        else if (TextUtils.isEmpty(locs)){
          location.setError("Please type your location");
          location.requestFocus();
          return;
        }
        else if (TextUtils.isEmpty(P)){
          PassWord.setError("Please type your password");
          PassWord.requestFocus();
          return;
        }
        StringRequest str=new StringRequest(Request.Method.POST, url, new
    Response.Listener<String>() {
          @Override
```

```
            public void onResponse(String response) {


               // Toast.makeText(Registraction.this, response, Toast.LENGTH_SHORT).show();


                try {
                    JSONObject jsnb=new JSONObject(response);
                    status=jsnb.getString("StatusID");
                    error=jsnb.getString("Error");


                } catch (JSONException e) {
                    e.printStackTrace();
                }


                if ("0".equals(status))
                {
                    Toast.makeText(Registraction.this,error,Toast.LENGTH_SHORT).show();
                }
                else{
                    Toast.makeText(Registraction.this,"Registration
successfull",Toast.LENGTH_SHORT).show();
                    Intent i=new Intent(Registraction.this, OptionActivity.class);
                    startActivity(i);
                }


            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
```

```
Toast.makeText(Registraction.this,error.toString(),Toast.LENGTH_SHORT).show();


            }
        })
        {
            @Override
            protected Map<String, String> getParams() {
                Map<String, String> params = new HashMap<>();
                params.put("username", U);
                params.put("Mobile_number", M);
                params.put("Email_id", E);
                params.put("password", P);
                params.put("location", locs);
                params.put("type", type);
                return params;
            }
        };
        RequestQueue rq= Volley.newRequestQueue(this);
        rq.add(str);
    }
    public static boolean isPhoneValid(String s) {
        Pattern p = Pattern.compile("(0/91)?[6-9][0-9]{9}");
        Matcher m = p.matcher(s);
        return (m.find() && m.group().equals(s));
    }
     public static boolean isEmailValid(String email) {
        String emailRegex = "^[a-zA-Z0-9_+&*-]+(?:\\."+
                "[a-zA-Z0-9_+&*-]+)*@" +
            "(?:[a-zA-Z0-9-]+\\.)+[a-z" +
            "A-Z]{2,7}$";


        Pattern pat = Pattern.compile(emailRegex);
        return pat.matcher(email).matches();
    }
    public boolean isValidPassword(final String password) {
        Pattern pattern;
        Matcher matcher;


        final String PASSWORD_PATTERN = ("^" +
            //"(?=.*[0-9])" +         //at least 1 digit
            //"(?=.*[a-z])" +         //at least 1 lower case letter
            //"(?=.*[A-Z])" +         //at least 1 upper case letter
            "(?=.*[a-zA-Z])" +      //any letter
            "(?=.*[@#$%^&+=])" +    //at least 1 special character
            "(?=\\S+$)" +           //no white spaces
            ".{4,}" +               //at least 4 characters
            "$");
        pattern = Pattern.compile(PASSWORD_PATTERN);
        matcher = pattern.matcher(password);
        return matcher.matches();
    }
}
```

Registration.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
```

```
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".User.Registraction">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="590dp"
    android:layout_marginTop="30dp"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="left"
        android:layout_marginLeft="10dp"
        android:text="Create account"
        android:textColor="@color/brown"
        android:textSize="30sp"
        android:textStyle="bold" />

    <!--        <androidx.cardview.widget.CardView-->
    <!--            android:layout_width="match_parent"-->
    <!--            android:layout_height="500dp"-->
    <!--            android:layout_margin="30dp"-->
    <!--            android:padding="6dp"-->
    <!--            android:paddingStart="10dp"-->
    <!--            android:paddingEnd="10dp"-->
    <!--            app:cardCornerRadius="20dp"-->
    <!--            app:cardElevation="6dp">-->

    <LinearLayout
        android:layout_width="match_parent"
```

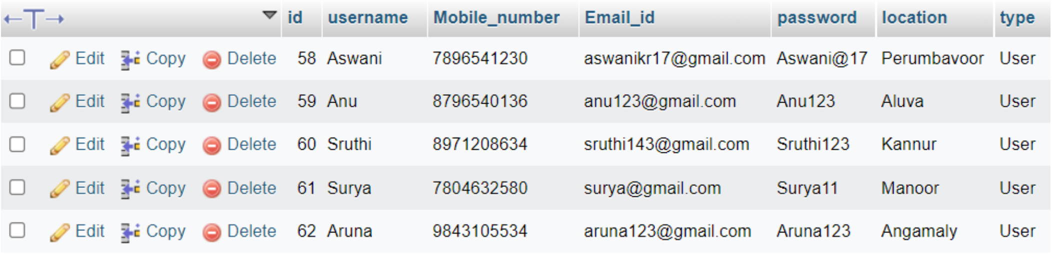
```
android:layout_height="wrap_content"
android:layout_margin="21dp"
android:layout_marginTop="20dp"
android:orientation="vertical">

<com.google.android.material.textfield.TextInputLayout
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp">

    <EditText
        android:id="@+id/userreg"
        android:layout_width="match_parent"
        android:layout_height="45dp"
        android:hint="Name" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp">

    <EditText
        android:id="@+id/phonereg"
        android:layout_width="match_parent"
        android:layout_height="45dp"
        android:hint="Mobile number"
        android:inputType="phone" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
```

```
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp">

        <EditText
            android:id="@+id/emailreg"
            android:layout_width="match_parent"
            android:layout_height="45dp"
            android:hint="Email" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp">

        <EditText
            android:id="@+id/location"
            android:layout_width="match_parent"
            android:layout_height="45dp"
            android:hint="Location" />
    </com.google.android.material.textfield.TextInputLayout>

     <com.google.android.material.textfield.TextInputLayout
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox.Dense"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        app:passwordToggleEnabled="true">

        <EditText
```

```
                android:id="@+id/passreg"
                android:layout_width="match_parent"
                android:layout_height="45dp"
                android:hint="Set Password"
                android:inputType="textPassword" />
        </com.google.android.material.textfield.TextInputLayout>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:text="! atleast 6 characters"
            android:visibility="gone"
            android:textColor="@color/black" />
    </LinearLayout>
    <!--        </androidx.cardview.widget.CardView>-->

    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/buttonreg"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="30dp"
        android:layout_marginEnd="30dp"
        android:textColor="@color/white"
        android:background="@drawable/bg"
         android:text="SIGN UP" />
</LinearLayout>

</RelativeLayout>
```

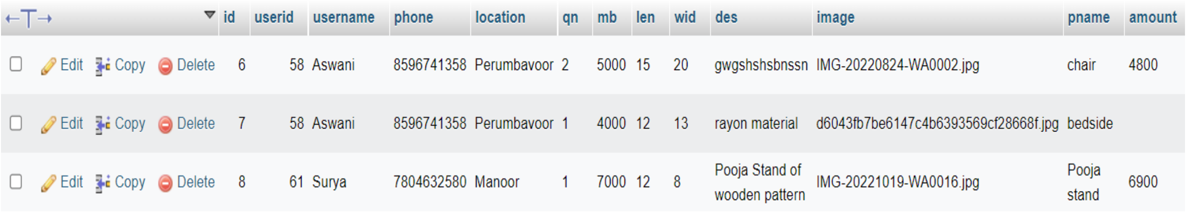
9.2 DATABASE

Registration table



←T→	id	username	Mobile_number	Email_id	password	location	type
□ Edit Copy Delete	58	Aswani	7896541230	aswanikr17@gmail.com	Aswani@17	Perumbavoor	User
□ Edit Copy Delete	59	Anu	8796540136	anu123@gmail.com	Anu123	Aluva	User
□ Edit Copy Delete	60	Sruthi	8971208634	sruthi143@gmail.com	Sruthi123	Kannur	User
□ Edit Copy Delete	61	Surya	7804632580	surya@gmail.com	Surya11	Manoor	User
□ Edit Copy Delete	62	Aruna	9843105534	aruna123@gmail.com	Aruna123	Angamaly	User

Customization table

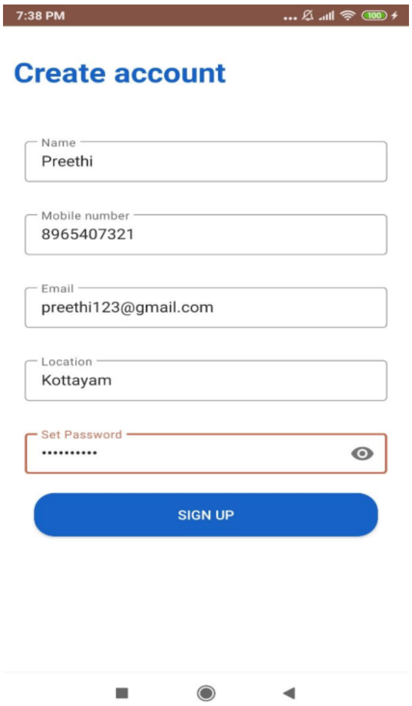


←T→	id	userid	username	phone	location	qn	mb	len	wid	des	image	pname	amount
□ Edit Copy Delete	6	58	Aswani	8596741358	Perumbavoor	2	5000	15	20	gwgshshsbnssn	IMG-20220824-WA0002.jpg	chair	4800
□ Edit Copy Delete	7	58	Aswani	8596741358	Perumbavoor	1	4000	12	13	rayon material	d6043fb7be6147c4b6393569cf28668f.jpg	bedside	
□ Edit Copy Delete	8	61	Surya	7804632580	Manoor	1	7000	12	8	Pooja Stand of wooden pattern	IMG-20221019-WA0016.jpg	Pooja stand	6900

9.3 SCREENSHOTS

USER SIDE

Registration page:



7:38 PM

Create account

Name: Preethi

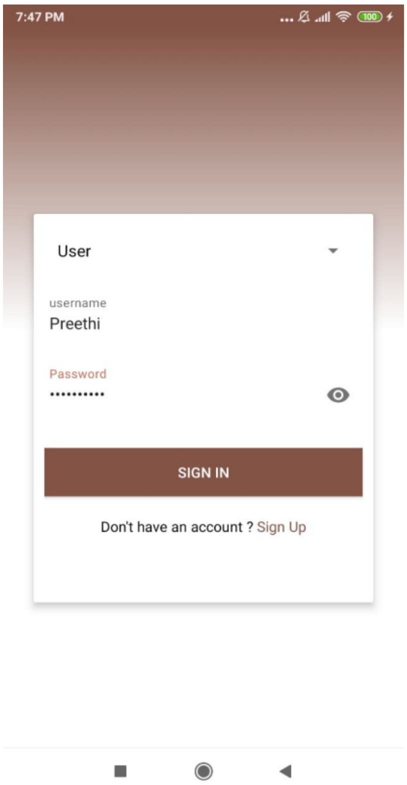
Mobile number: 8965407321

Email: preethi123@gmail.com

Location: Kottayam

Set Password: ••••••••••

Login page:



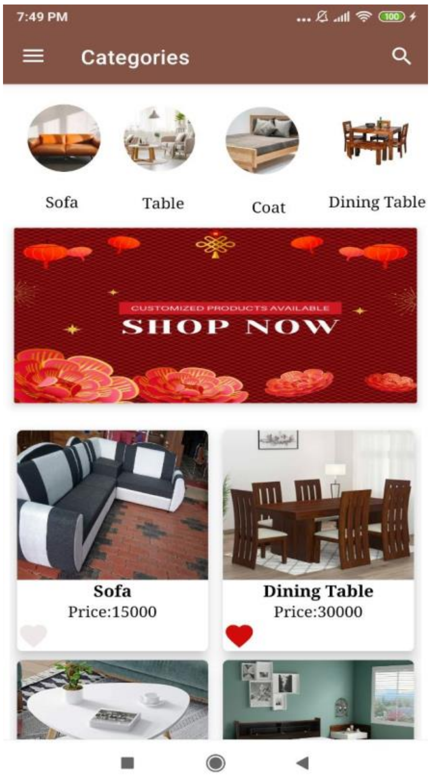
7:47 PM

User

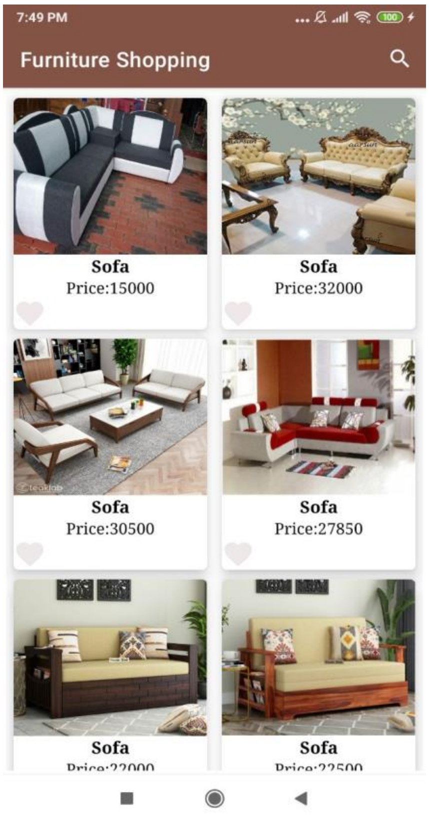
username: Preethi

Password: ••••••••••

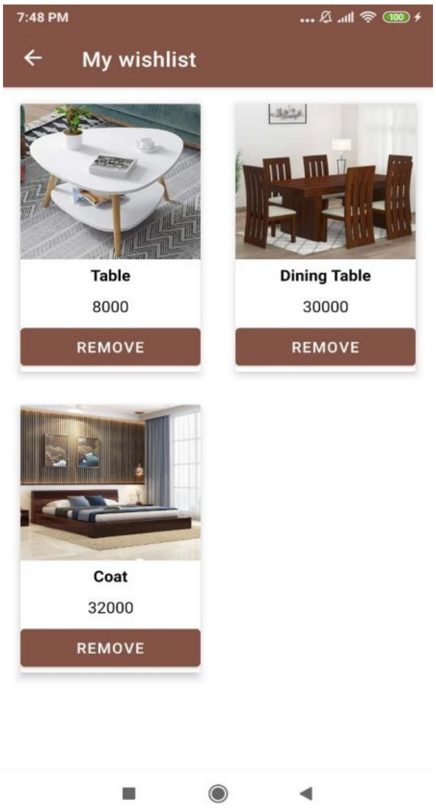
Home page:



Category wise search:



Wish list:



7:48 PM

My wishlist

Table

8000

REMOVE

Dining Table

30000

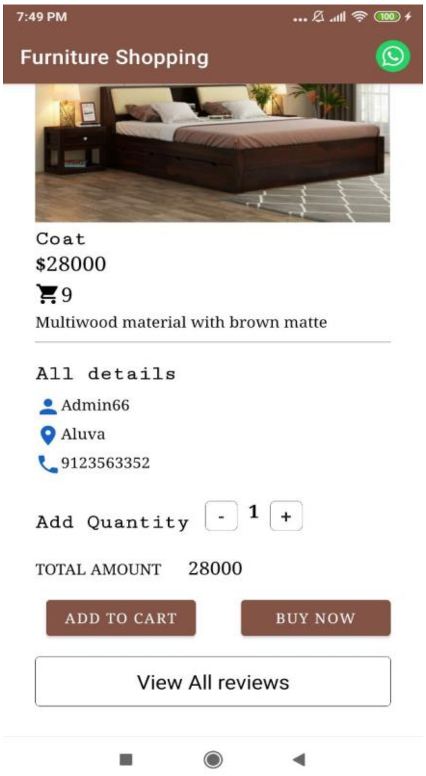
REMOVE

Coat

32000

REMOVE

Product view:



7:49 PM

Furniture Shopping

Coat

$28000

9

Multiwood material with brown matte

All details

Admin66

Aluva

9123563352

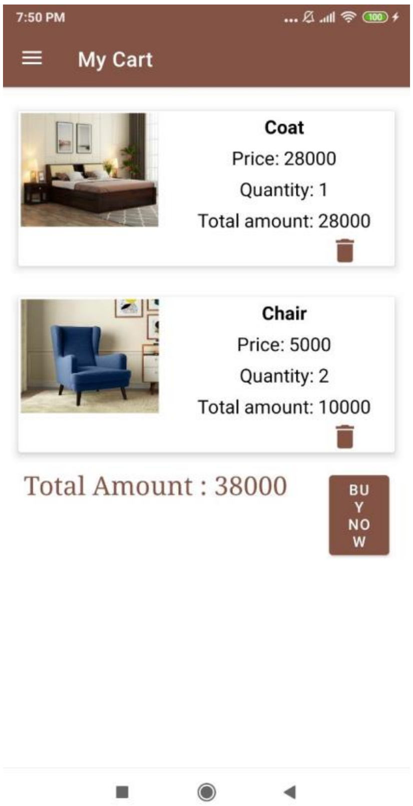
Add Quantity - 1 +

TOTAL AMOUNT 28000

ADD TO CART BUY NOW

View All reviews

Cart view page:



Coat

Price: 28000

Quantity: 1

Total amount: 28000

Chair

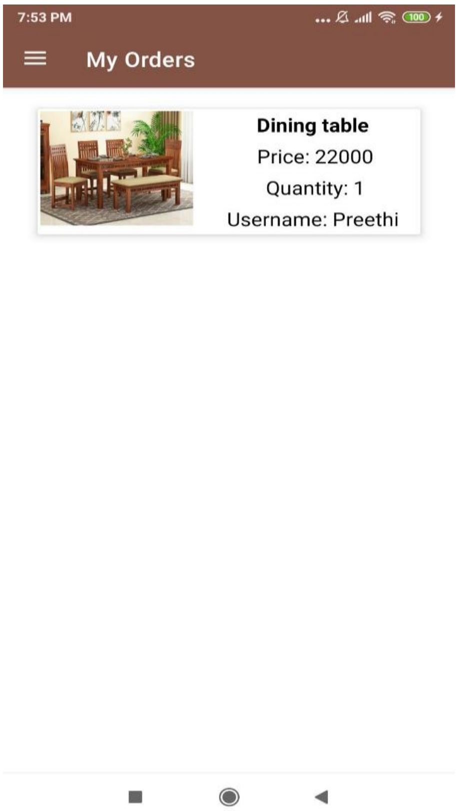
Price: 5000

Quantity: 2

Total amount: 10000

Total Amount : 38000

Order view page:



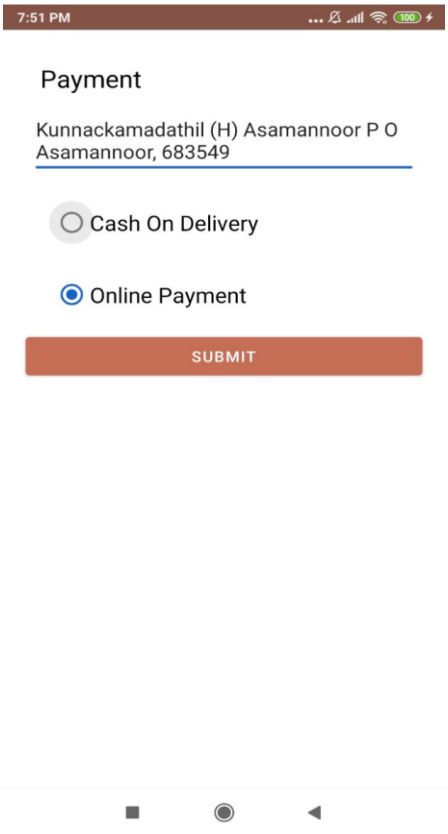
Dining table

Price: 22000

Quantity: 1

Username: Preethi

Payment page:

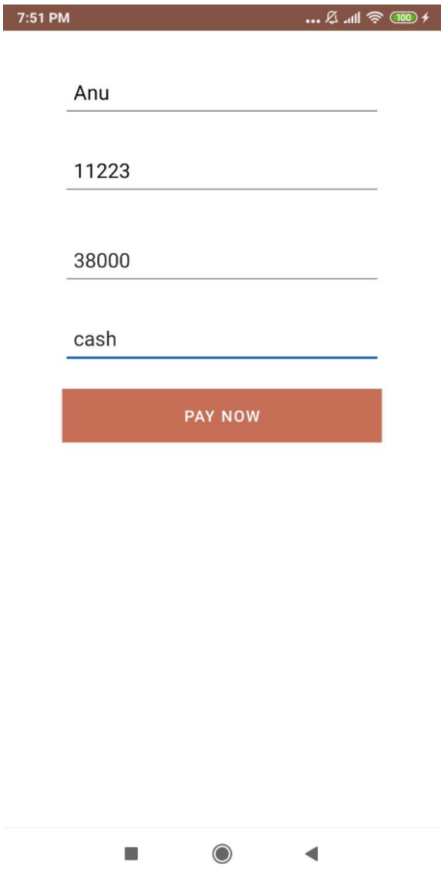


Payment

Kunnackamadathil (H) Asamannoor P O Asamannoor, 683549

○ Cash On Delivery

● Online Payment



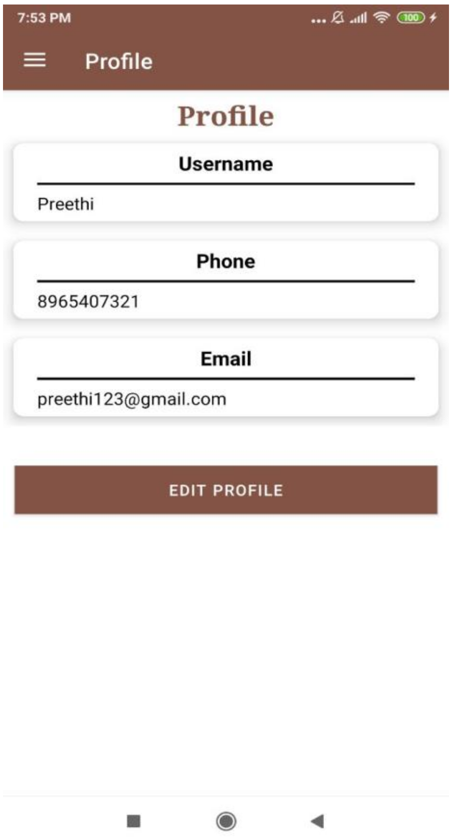
Anu

11223

38000

cash

Profile page:



7:53 PM

Profile

Profile

Username

Preethi

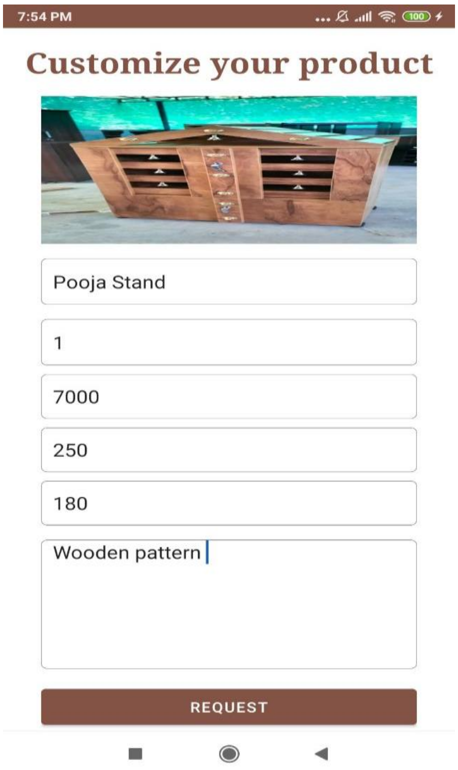
Phone

8965407321

Email

preethi123@gmail.com

Customization request page:



7:54 PM

Customize your product

Pooja Stand

1

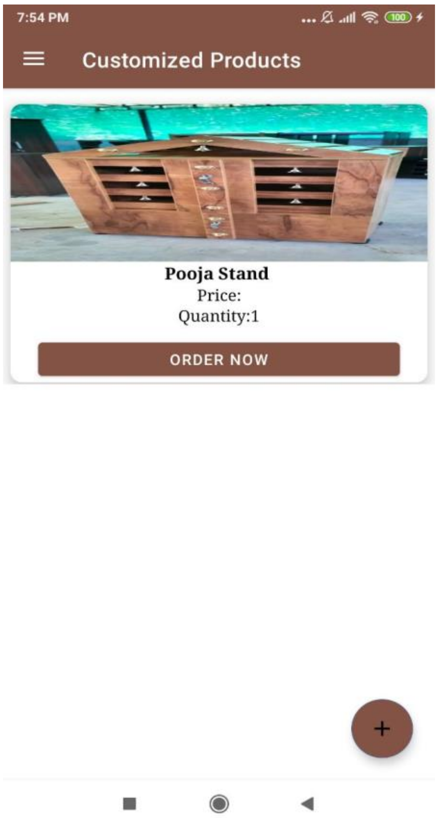
7000

250

180

Wooden pattern

Customization order page:

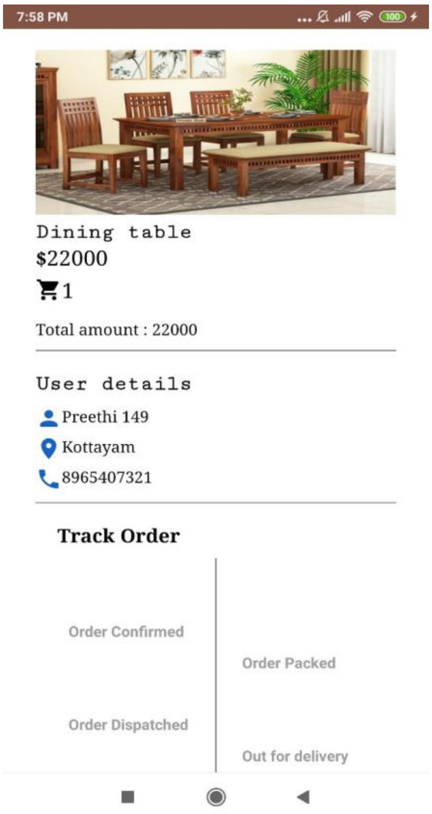


7:54 PM

Customized Products

Pooja Stand
Price:
Quantity:1

Order tracking page:



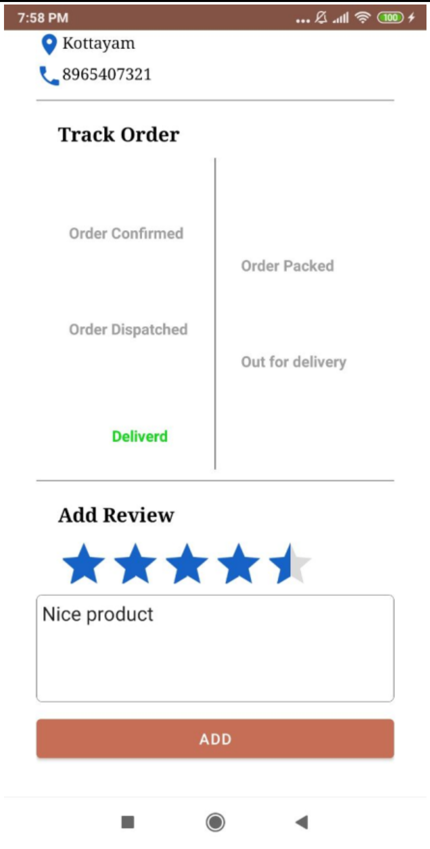
7:58 PM

Dining table
$22000
1

Total amount : 22000

User details
Preethi 149
Kottayam
8965407321

Track Order



📍 Kottayam

☎ 8965407321

Track Order

Order Confirmed

Order Packed

Order Dispatched

Out for delivery

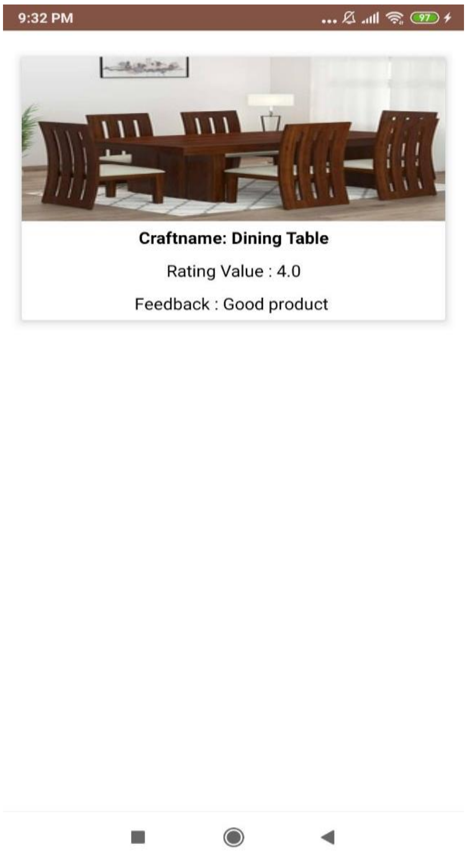
Deliverd

Add Review

★★★★☆

Nice product

Feedback page:



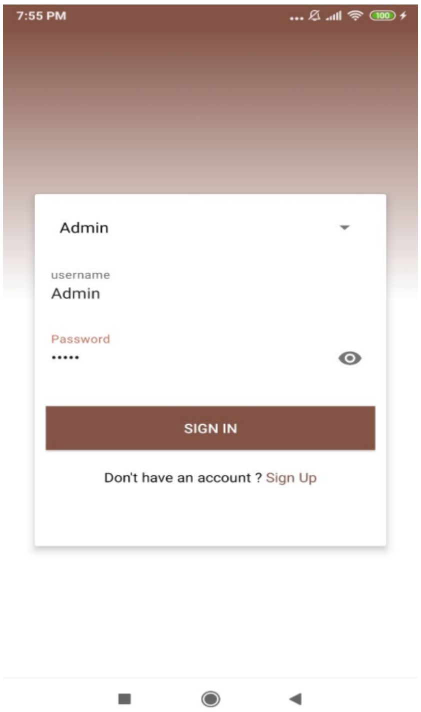
Craftname: Dining Table

Rating Value : 4.0

Feedback : Good product

ADMIN SIDE

Login page:

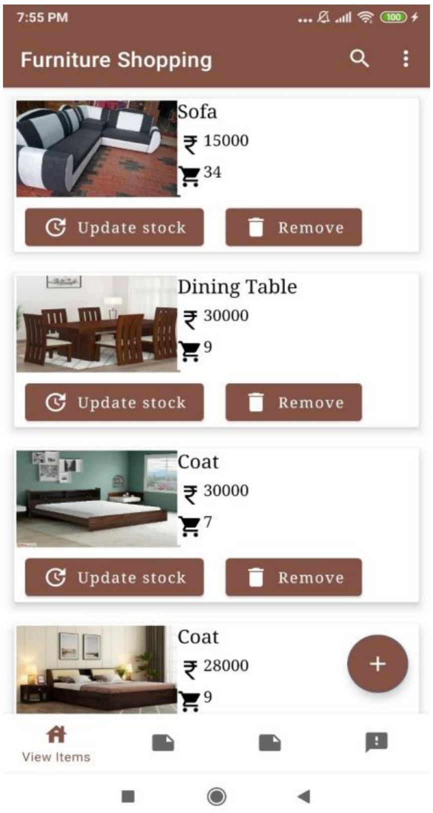


Admin

username
Admin

Password
•••••

Adding new item page:



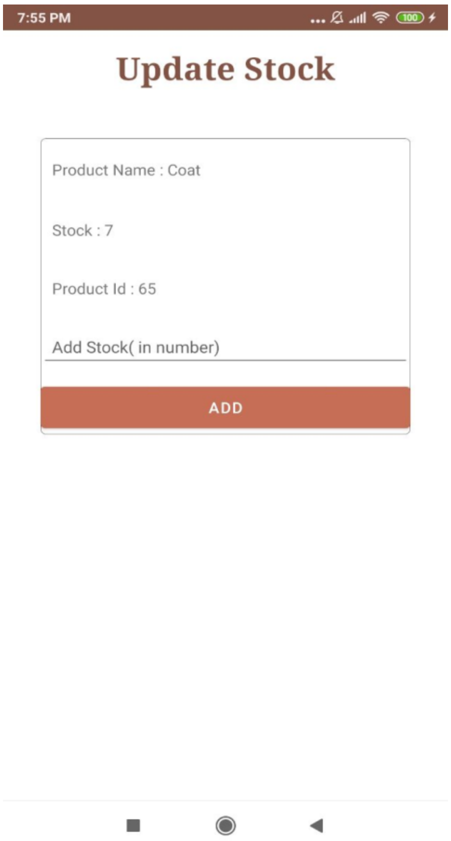
Sofa
₹ 15000
34

Dining Table
₹ 30000
9

Coat
₹ 30000
7

Coat
₹ 28000
9

Stock update page:



Update Stock

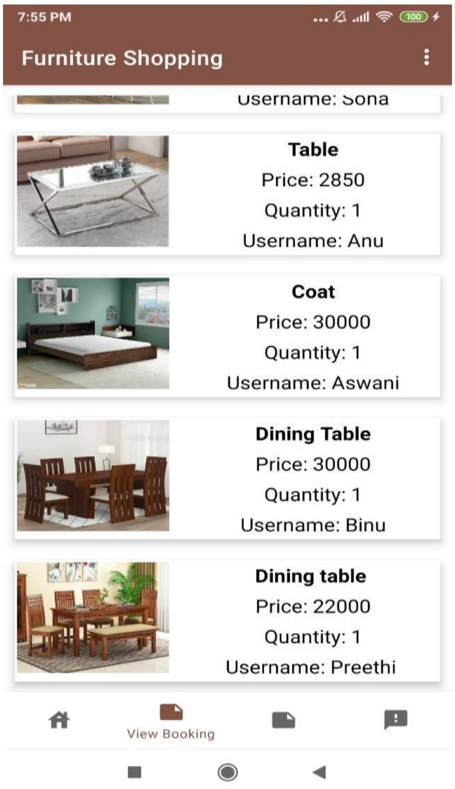
Product Name : Coat

Stock : 7

Product Id : 65

Add Stock(in number)

Booking view page:



Username: Sona

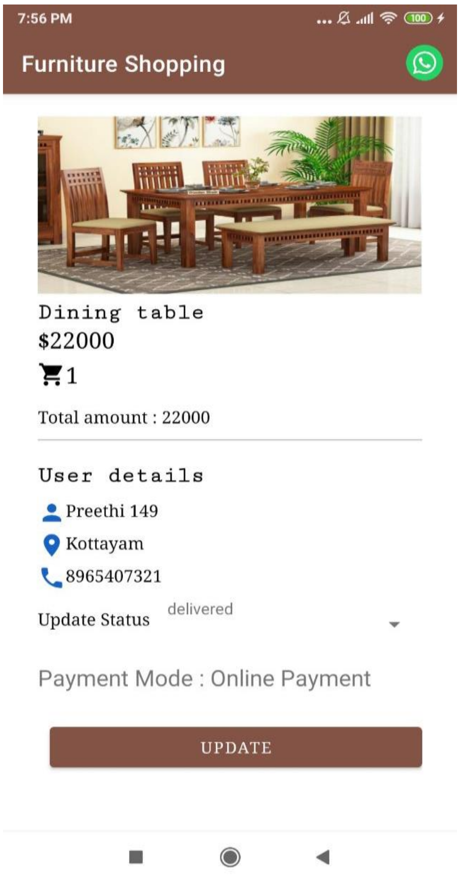
Table
Price: 2850
Quantity: 1
Username: Anu

Coat
Price: 30000
Quantity: 1
Username: Aswani

Dining Table
Price: 30000
Quantity: 1
Username: Binu

Dining table
Price: 22000
Quantity: 1
Username: Preethi

Updating order status page:



Dining table
$22000
1

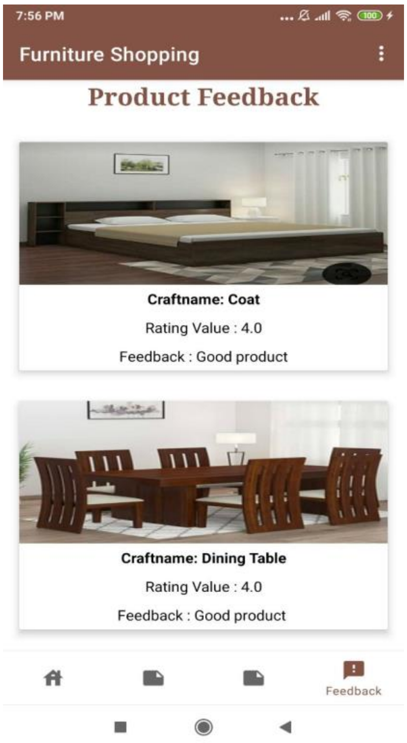
Total amount : 22000

User details
Preethi 149
Kottayam
8965407321

Update Status delivered

Payment Mode : Online Payment

Feedback viewing page:

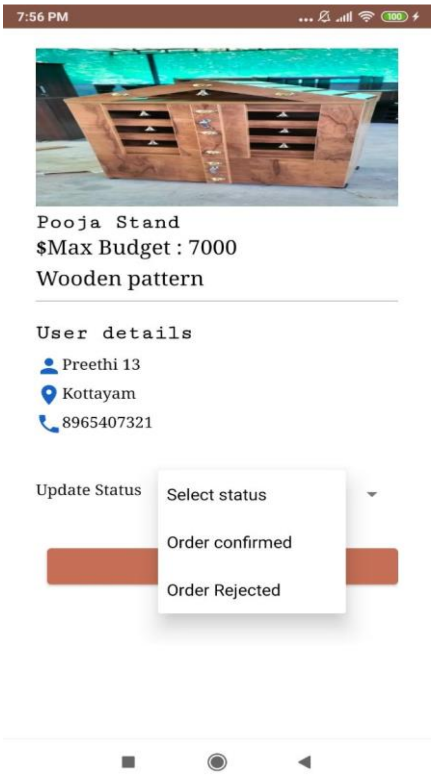


Product Feedback

Craftname: Coat
Rating Value : 4.0
Feedback : Good product

Craftname: Dining Table
Rating Value : 4.0
Feedback : Good product

Updating customization status:



7:56 PM

Pooja Stand

$Max Budget : 7000

Wooden pattern

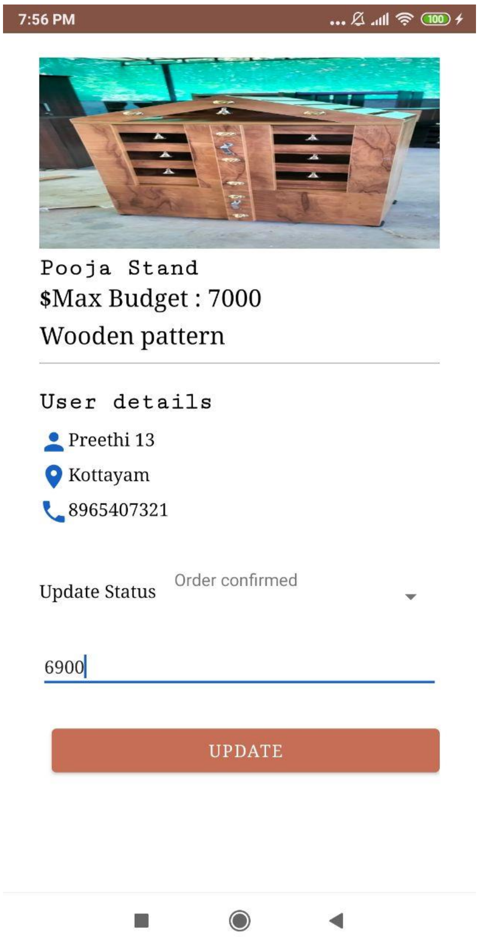
User details

Preethi 13

Kottayam

8965407321

Update Status



7:56 PM

Pooja Stand

$Max Budget : 7000

Wooden pattern

User details

Preethi 13

Kottayam

8965407321

Update Status: Order confirmed

6900

Chapter 11

BIBLIOGRAPHY

[1] Nowacki, Paweł & Woda, Marek. (2020). Capabilities of ARCore and ARKit Platforms for AR/VR Applications.

[2] Akshay Singh, Sakshi Sharma and Shashwat Singh. Article: Android Application Development using Android Studio and PHP Framework. IJCA Proceedings on Recent Trends in Future Prospective in Engineering and Management Technology RTFEM 2016(1):5-8, July 2016.

[3] Google Developers. 2021. Sceneform overview | Sceneform (1.15.0) | Google Developers. [online] Available at: https://developers.google.com/sceneform/develop [Accessed 11 May 2021].

[4] JavaScript Enlightenment,Cody Lindley-First Edition, based onJavaScript 1.5, ECMA-262, Edition

[5]Mc GrawHill's , Java : The complete reference 7thEdition, Herbert Schildit

[6] Complete CSS Guide ,Maxine Sherrin and John Allsopp-O'ReillyMedia; September 2012

[7] https://ieeexplore.ieee.org/document/9211184

https://ieeexplore.ieee.org/document/9391825

https://nextbigtechnology.com/online-furniture-shop-app-development/

https://nevonprojects.com/ebug-tracker-bug-tracking-system-project/

https://nevonprojects.com/ebug-tracker-bug-tracking-system-project/