Title: Smart Courier Service

Program Name: Software Engineering

Session: Session (2021 - 2025)

Authors Name and Roll Number: Submitted By

Sallah Udin 2021-SE-18

Supervisor:

Engr. Daniyal Nazir

Department Name: Department of Software Engineering

University of Azad Jammu And Kashmir

Approval Sheet

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

DEDICATION

This project could never have been possible without the support of my family. My whole success is dedicated to my parents. I thank them for the spirit of hard work, courage and determination throughout my College days till today and also to our respected teachers.

Sallah Udin

ACKNOWLEDGEMENT

Special thanks to our project supervisor Engr. Daniyal Nazir whose support and guidance played an important role in our achievements. We are also thankful to our FYP coordinator Engr. Ahmad Khawaja who supported us and also offered us help to make this project successful. Their guidance and advice, both mentally and technically has been of great importance to the final outcome of this final thesis. We sincerely appreciate their time and efforts.

PREFACE

This project thesis concerns with the development of **"Smart Courier Services".** This Project thesis covers the various phases in the development of this project, which are as follow:

CHAPTER 1. INTRODUCTION

This chapter describes a brief introduction about **"Smart Courier Services"** and also about the tools used for the development of this project.

CHAPTER 2. EXISTING SYSTEM

This chapter describes the background (Existing system) and also limitation of **"Existing system"**.

CHAPTER 3. PROPOSED SYSTEM

This chapter describes the proposed **"Smart Courier Services"** and its features.

CHAPTER 4. SYSTEM DESIGN

This chapter describes the system's design of **"Smart Courier Services"** design phase (architecture) and also about the user interface of the system.

CHAPTER 5. SYSTEM TESTING

This chapter describes the testing phase of the proposed **"Smart Courier Services"**.

CHAPTER 6. CONCLUSION & FUTURE WORK

This chapter describes the conclusion and future work of the proposed **"Smart Courier Services"**.

ABSTRACT

This project highlight on the development of "Smart Courier Services," a peer-to-peer, on-demand delivery platform for revolutionize traditional courier systems. The platform connects customers with couriers using geolocation technology, real-time tracking, and smart algorithms.Customers can request deliveries, track packages in real-time, and receive delivery notifications, while couriers can accept or decline requests based on their availability.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1. Introduction

The pick and drop item delivery Courier Service system is used to pick a product from one place and drop it to another place within the city in a specific time. This system can be mainly used when you have forgotten your product (Example: File and you're not able to back and pick it up. This system will do this for you).In this system, you have to enter your pick up point and delivery point. The main advantage of this Courier Service system over the courier system is you can decide the time limit and assign job to any agent. Based on the time you will be charged the lesser is time more you will be charged.Moving from one apartment or house to the other can be quite stressful process especially if the individual(s) does not have the resources (vehicle) to carry out such action. This difficulty also occurs when sending a parcel or goods to a particular location. The system will also provide to its users an unlimited scope when it comes to package delivery, they will be able to send packages without stepping out of their house where door to door delivery of this service takes place in real-time regardless of the distance and how little or big an item is, the system is sure to meet the needs of its users in such horizon.

1.2. Purpose

The new system is designed to solve problems affecting the manual system in use. !t is designed to be computerized thereby relieving both the customers and staff from much stress as experienced in the manual system. This system will do the analyzing and storing of information either automatically or interactively. SCS is a major improvement on the post office as it delivers better services in a very convenient manner and eliminates the time wasted in standing in long queues waiting to buy stamps and other postal products. This system enables a registered user to buy postal products online and have them sent to his mail box without having to visit the post office. !t also enables him to buy goods online and have them delivered to his residence address via courier service. This makes the prospect of using this service more appealing as it saves time and money, plus it is a very fast way of purchasing goods online.

1.3. Scope of the Project

The scope of this project is to design, develop, and implement a peer-to-peer on-demand delivery platform that enables anyone to become a courier. The platform will have the following features:
The Smart Courier Service aims to address these challenges by revolutionizing the traditional courier industry. By leveraging a peer-to-peer approach, our platform enables anyone to become a courier, providing a flexible, efficient, and cost-effective solution for on-demand deliveries. This innovative approach has the potential to transform the courier industry, providing customers with faster, more reliable, and more personalized delivery experiences.

1. User-friendly mobile application.
2. Geolocation-based courier matching and real-time tracking
3. On-demand delivery request and acceptance system

4. Secure payment processing and transaction management
5. Rating and review system for couriers and customers
6. Real-time analytic and performance monitoring

1.4. Definitions, Acronyms, Abbreviations

Table 1 Abbreviations

ABBREVIATION	DEFINITION
FYP	Final year project
PC	Personal Computer
SCS	Smart Courier Service
App	Android Application
OS	Operating System

DB	Database
OO	Object Oriented
API	Application Programming Interface
USER /SENDER	Any person that owns a package and wants to send it from one place to another and publishes his job on the application

1.3. Use Case and Usage Scenario

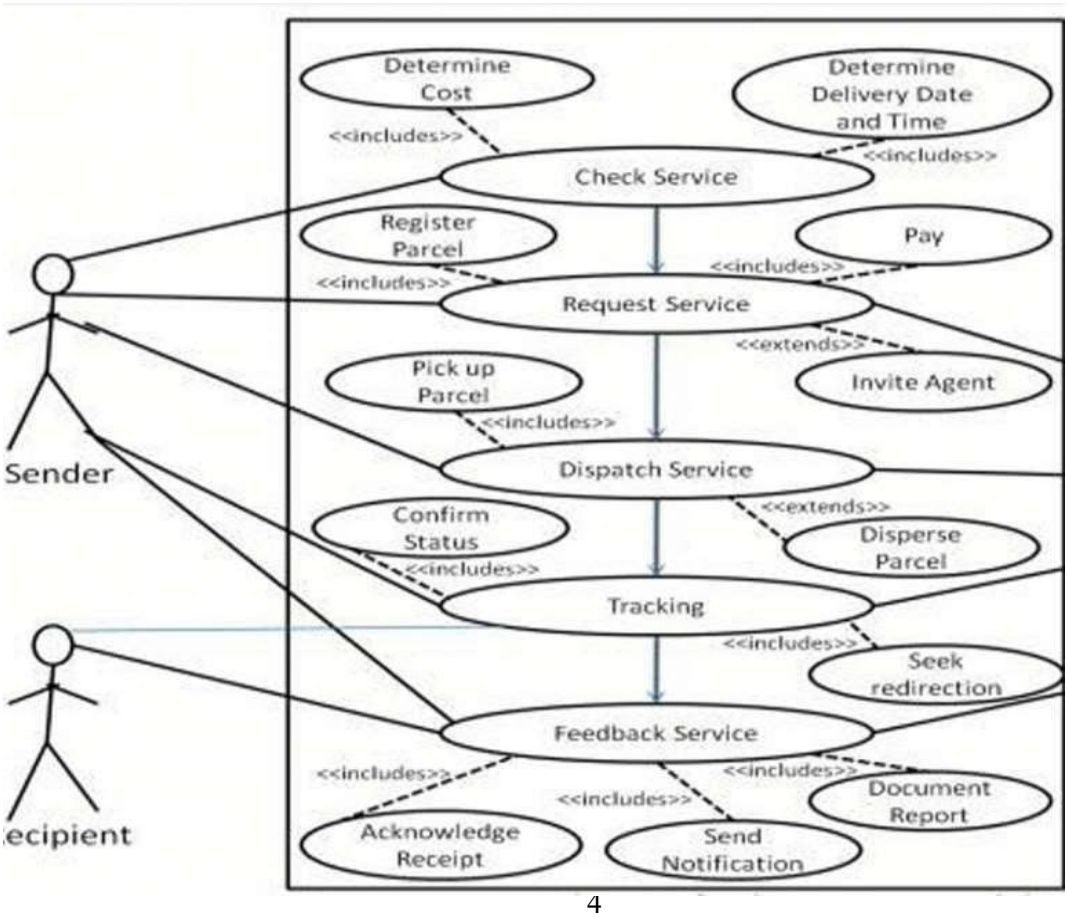


Figure 1 Use Case Diagram

1.5.1. Usage Scenarios

Table 2 Usage Scenario for Signup

<table>
<tr><td>• Use Case Title</td><td>• Sign-up</td></tr>
<tr><td>• Use Case ID</td><td>• 1</td></tr>
<tr><td>• Requirement ID</td><td>• 2.3.1</td></tr>
<tr><td colspan="2">• Description: This use case is for creating an account in Smart Courier Services</td></tr>
<tr><td colspan="2">• Pre-Condition
• User must provide unique username i.e. First Name, Last Name, Email, Password, Phone number and profile picture.
• Application should be hosted on an online server so everyone can access it.</td></tr>
<tr><td>• Task Sequence</td><td>• Task Execution</td></tr>
<tr><td>• Click on Sign Up button</td><td></td></tr>
<tr><td>• Server provide sign up page.</td><td></td></tr>
<tr><td>• Users provide all the related information for sign up to create new account</td><td>• Error message if any field is empty</td></tr>
<tr><td colspan="2">• Post Condition: User account is created successfully</td></tr>
</table>

Table 3 Usage Scenario for Login

• Use Case Title	• Login
• Use Case ID	• 2
• Requirement ID	• 2.3.2
• Description: It enable the user log in the system for posting a job.	
• Pre-Condition • User must have an account in the **Smart Courier Services** app. • Database should be working online. • Application should be working online.	
• Task Sequence	• Task Execution

• User open app and click on login.	
• User must provide correct username and password for login.	• Nil
• System check entries in database	• If username and password does not exist in the database than it will generate an error that account doesn't not exist.
• If profile exist than it will take user to another activity for ads job.	• Your profile was not found
• Post Condition: • User successfully login to the system • User can now access the feature of this app	
• Unresolved Issues:	
• Authority: Administration and System	

Figure 2 Use Case

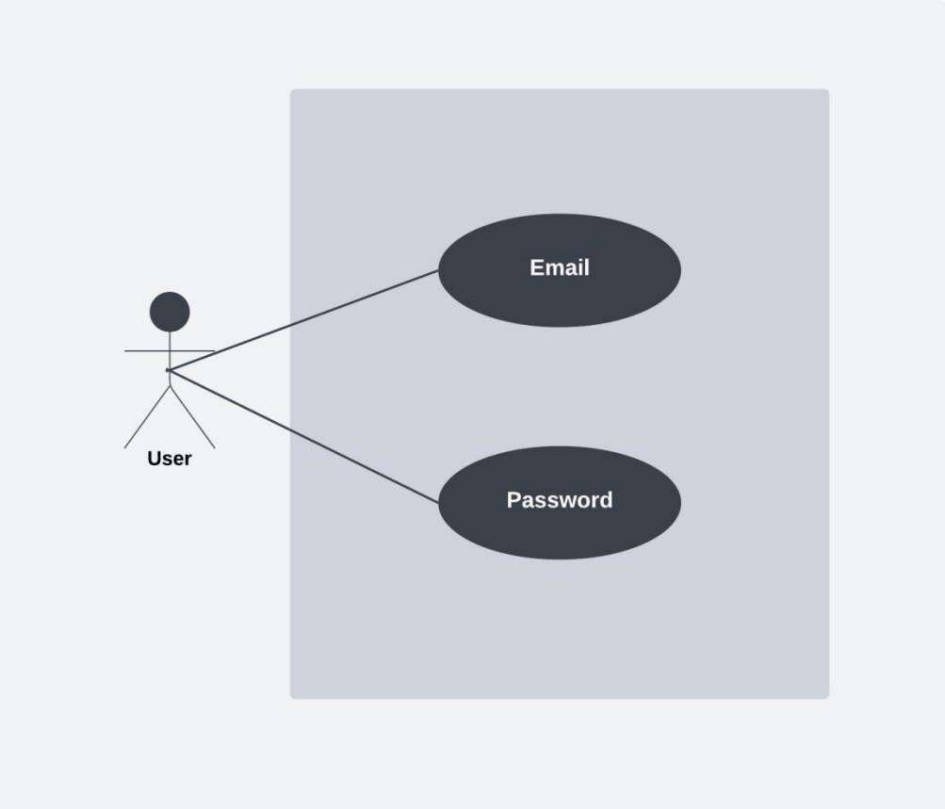


Diagram for Login

Table 4 Usage Scenario for Post Ad

• Use Case Title	• Post Ad
• Use Case ID	• 3
• Requirement ID	• 2.3.3
• Description: This use case is for publishing ad's in the app.	

• Pre-Condition: • User must be login to the system. • System database should be working condition	
• Task Sequence	• Task Execution
• When user logged into the system the first view is of submitting forms for publishing the ad.	• Invalid username and password
• The user can see the form and interact with it.	• Nil
• The user can submit the form after fill each blank.	• Cannot submit or publish ads.
• Post Conditions: User can perform the following function. • Create • Update • Delete	
• Unresolved Issue:	
• Authority: Administration and System	

Table 5 Usage Scenario for My Ads

• Use Case Title	• My ads
• Use Case ID	• 4
• Requirement ID	• 2.3.4

<table>
<tr><td colspan="2">• Description: The purpose of this use case is to maintain the history of the specific user.</td></tr>
<tr><td colspan="2">• Pre-Condition
• User must be logged in to the system
• Application should be connected to the internet.</td></tr>
<tr><td>• Task Sequence</td><td>• Task Execution</td></tr>
<tr><td>• User must be sign into the system and click on My ads.</td><td>• Error account not found.</td></tr>
<tr><td>• User can perform the functionality like (CRUD)</td><td>• Internet connection is lost.</td></tr>
<tr><td colspan="2">• Post Condition:
• User can view his/her ads.
• User can check the information of how much ads are published in the past.</td></tr>
<tr><td>• Unresolved Issue:</td><td></td></tr>
<tr><td>• Authority: Administration and System</td><td></td></tr>
<tr><td>• Modification history: 1.0 Authors:
• Description:</td><td></td></tr>
</table>

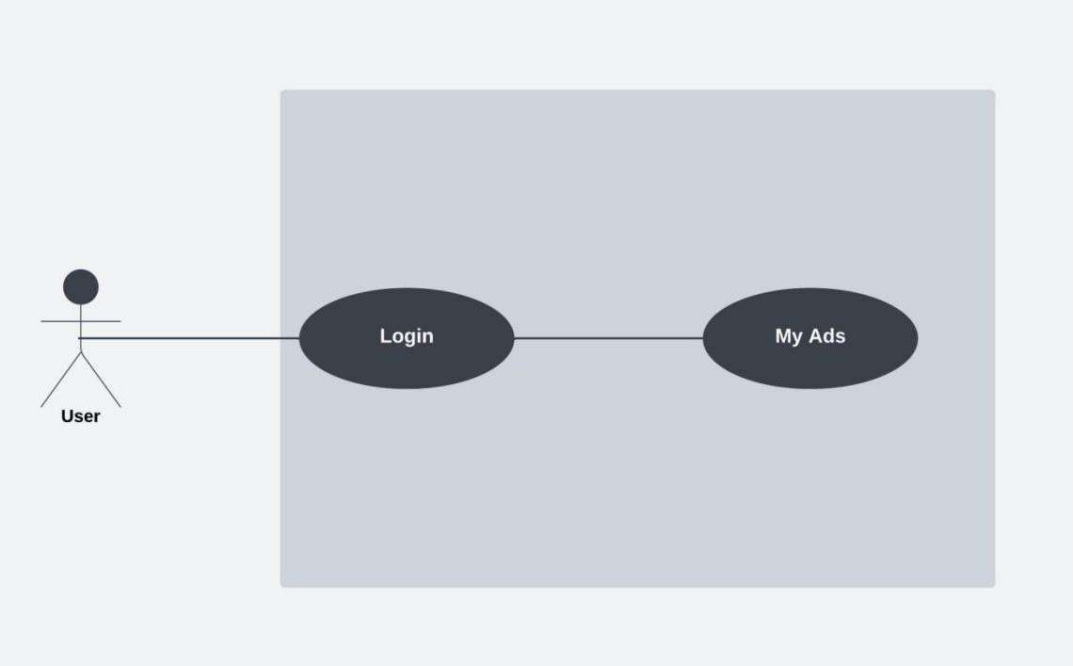

Login
My Ads
User

Table 6 Usage Scenario for Search Jobs

• Use Case Title	• Search
• Use Case ID	• 5
• Requirement ID	• 2.3.5
• Description: This use case is for user to search his related jobs.	
• Pre-Condition • System must be connected to the internet. For search required .	
• Task Sequence	• Task Execution
• 1. User can open the app and search his jobs .	• Internet error
• 2. User can search-jobs either by name .	• No match found
• 3. System search the ad from database and show result.	• Nil.
• Post Condition: User is able to search .	
• Unresolved Issue:	
• Authority: User.	

Table 7 Usage Scenario for checking information of Jobs

• Use Case Title	• Check jobsr.
• Use Case ID	• 6
• Requirement ID	• 2.3.6
• Description: The use case is for checking Jobs info.	
• Pre-Condition • User must find his Jobs first. • Database should be available in online mode.	
• Task Sequence	• Task Execution
• 1. User can search the ad and click on it.	• Internet error.
• 2. System can show detail of specific Jobs.	• Resource not found.
• Post Condition: • User can see the overall b information and contact of the person. • User can see the seller contact.	

1.6. WBS and Gantt Chart

1.6.1. Work Breakdown Str

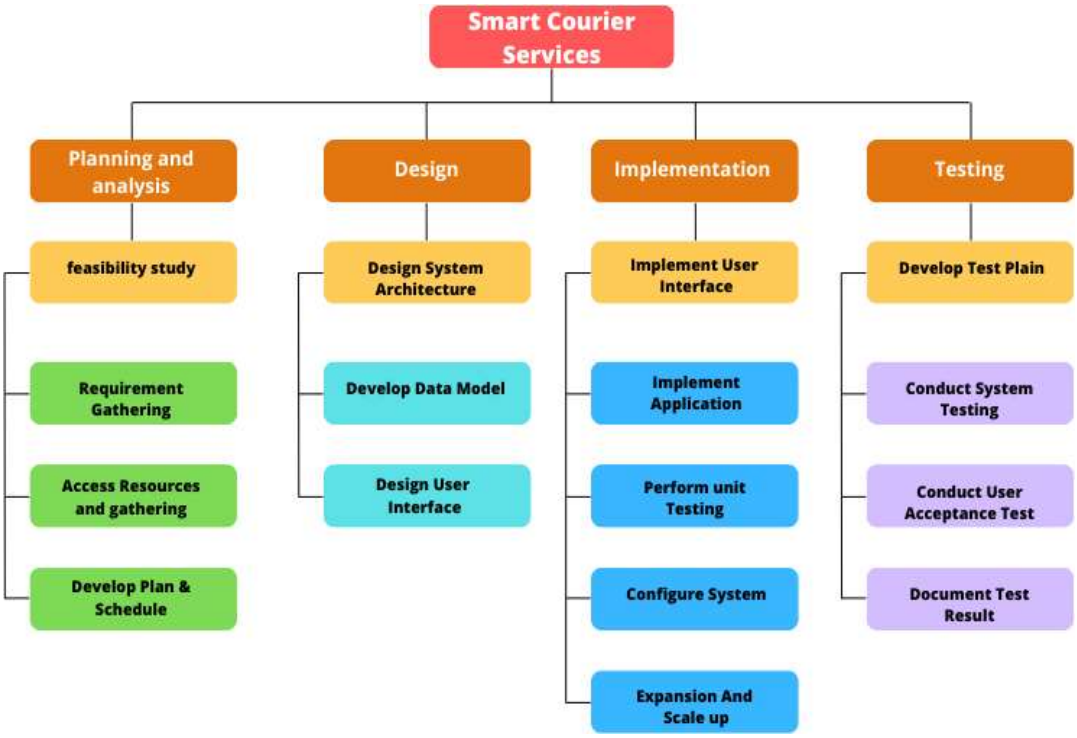
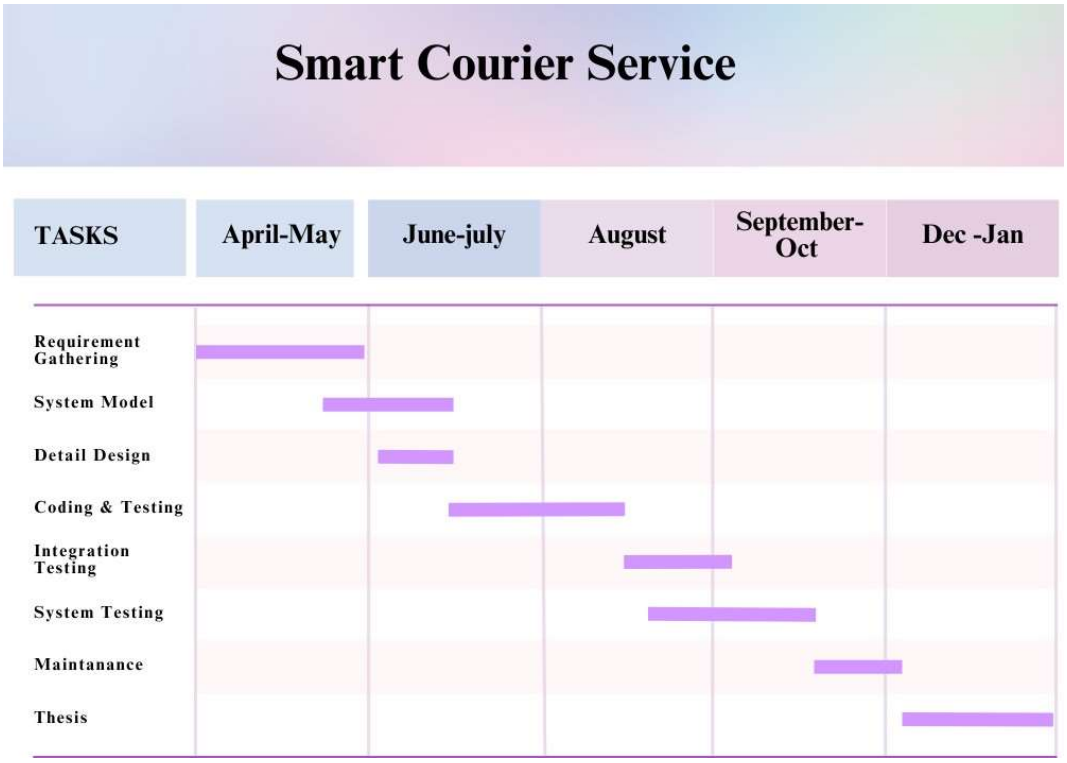


Figure 4 Work Breakdown Structure

1.6.2. Gantt Chart



1.6.3.

Figure 5 Gantt Chart

CHAPTER 2

EXISTING SYSTEM

2.1. Existing System

TCS application is an item delivery software developed by the courier company in Pakistan. It was released long time ago. The application provides courier services across cities in Pakistan while Leopard courier is an on-demand delivery application which provides instant couriers services in popular cities in Pakistan. The application provides top-notch service to over 250,000 users in the aforementioned country. The existing Courier Service System, if you want to deliver a product from one place to another place in a city. We have to look for a courier system, but the drawback of the courier system is that you cannot decide time limit and major concern is the weight of the product to be delivered. If we need to deliver product immediately then we cannot use courier system nowadays. If not courier system, then we have to deliver the product by ourselves and invest our time, this becomes a tedious and hectic task.Consumer surveys including the latest study by Citizens Advice reveal that as many as two thirds of consumers are likely to have experienced at least one problem with their online shopping parcel delivery in the past year. The most common problem is not a late or lost parcel, as one might think, but the problem of the no-one-at-home slip.

2.2. Literature Review

Existing systems don't have to spend a lot of money, or any money, on advertising if you don't want to. There are many free ways you can advertise. Try out Craigslist or Back-page.com. It's an easy and quick way to post an ad. And don't forget about networking, simply getting out there and talking up your business to prospective customers. Try making a web site or blog. There are many free and easy ways to do it. On your web site or blog, talk about your business, your prices and what you have to offer.

You need to be organized. An organized courier is a successful courier. As you start getting more and more jobs, you need to be sure to have your day planner or other scheduling method nearby at all times. Keep track of your jobs, and keep track of your customers. You never want to get a call wondering why you aren't where you need to be. You don't ever want to say, "Oops." Keep a detailed schedule, and this shouldn't be a problem. Keeping record of parcels in a courier service company and their delivery information is carried out manually. This litters the office with much paper documents. Most often records are misplaced and when a client comes to collect his parcel.

2.3. Limitations of Existing System

2.3.1. Inflexibility in Delivery

Existing courier system do not allow users to set specific time limits for package delivery.

2.3.2. Weight and Size

Most traditional systems impose significant limitations on the weight and size of items that can be shipped, reducing their practicality for certain types of deliveries.

2.3.3. High Costs

Traditional courier systems are associated with high operational costs, which are often passed on to customers, making the services less affordable.

CHAPTER 3

PROPOSED SYSTEM

3.1. Introduction

The new system is designed to solve problems affecting the manual system in use. !t is designed to be computerized thereby relieving both the customers and staff from much stress as experienced in the manual system. This system will do the analyzing and storing of information either automatically or interactively.

3.2. Modules of System

The system comprises the set of modules which are listed below**:**

3.2.1. Authentication

They must have to login to the system for work.

3.2.2. Category

There are different categories of delivery jobs.

3.2.3. All Jobs

When user clicks on the all jobs option the overall jobs are shown to user, he just has to scroll up or down to search for them.

3.2.4. Delivers

The parcels we delivers from one place to another

3.2.5. My Ads

User can check and see the jobs we has published in the past and can perform the functionality of creating and deletion.

3.2.6. My Account

A user and agent will be able to create accounts all users can modify their profile information

3.3. Functional Requirements

We have designed the complete architecture of the SCS android application which shows the entities involved in application and their access roles. It is an Android application with following features:

- A user and agent will be able to create accounts
- All users can modify their profile information
- Both the user and agent can rate each other in other to gain reputation for doing fair business
- The user can create a new job to be displayed when agents search for jobs in view jobs menu
- When an agent needs to find a package for a job, he can use a search form which asks for destination, origin, package size, type and date/time
- Rate of delivery/Return delivery according to delivery type, weight and zones will be shown to booking customer at the time of booking, after that customer can confirm
- Status of delivery will be updated regularly after successful delivery by courier.
- Administrator will confirm all ty^{2}p^{2}es of user registration after necessary versification and will also manage all types of user
- For regular delivery booking, delivery agent will take payment while

3.4. Non- Functional Requirements

3.4.1. Usability

An interface should be easy to learn how to use and easy to remember how to use. The latter pertains especially to devices that require infrequent use.

3.4.2. Maintainability

3.4.3. Reliability

Reliability is usually defined as the probability that a product will operate without failure for a specified number of uses (transactions) or for a specified period of time. To be truly testable a requirement for software reliability should be stated as a forecast and the test results should indicate the confidence level associated with the forecast that the product will meet the requirements. Reliability measures for the application are as follow:

3.5. Feasibility

- Goals are easy to accomplish quickly and with few or no user errors intuitiveness: The interface is easy to learn and navigate; buttons, headings, and help/error messages are simple to understand low perceived workload: The interface appears easy to use, rather than intimidating, demanding and frustrating

3.5.1. Economic Feasibility

The purpose of economic feasibility is used to determine that whether the proposed systemwill provide the positive outcomes to the organization or not. It typically involves the costbenefits and analysis of the examined project whether it is possible to be implemented.Once the requirements of new system are defined and given solution to the problem is selected, its cost benefits and analysis of each alternative can be determine and the solution to the problem is selected and to be ensure that the project we are going to build is economically feasible.

Our system is an android application so it is easily available to its users and can be free to use for everyone. The system is going to be built in android studio using dart so no expensive or payable tools can be used to complete the project, so it is economically feasible to be built.

3.5.2. Operational Feasibility

It ensures that how the system is operationally feasible to solve its problem. Operational feasibility is mainly concern with issues whether the system is going to work properly after its development or implementation. It ensures that the system can work properly after its implementation. As the application ensures usability, maintainability and reliability so it is operationally feasible.

3.5.3. Technical Feasibility

It is based on the outline design of the system requirements to be ensure that our organization is

technically feasible to build this project or not. The purpose of technical feasibility is to ensure that our staff or our organization is technically feasible to implement the project and understanding of the present resources of organization and the expected needs of the proposed system. It is the evaluation of software and the hardware resources to meet the purpose system. Now a days everyone educated or uneducated both are aware of the use of the system applications so they can easily use the proposed system and on the other hand system cannot take extra technical requirements and not need new skills to implement the project so the project is technically feasible too. Our application is easy to use by every user so it cannot need the users to learn any skills and users can be easily aware of the system operations and understand it easily.

3.6. Chosen Methodology

In this project, we have used design science research methodology with AGILE approach. The process includes six steps:

- Problem identification and motivation.
- Definition of the objectives for a solution.
- Design and development.
- Demonstration.
- Evaluation.
- Communication.

3.6.1. Design Science Research Methodology

DSRM can help identify and implement solutions in challenges .

DSRM can help improve knowledge through the creation .

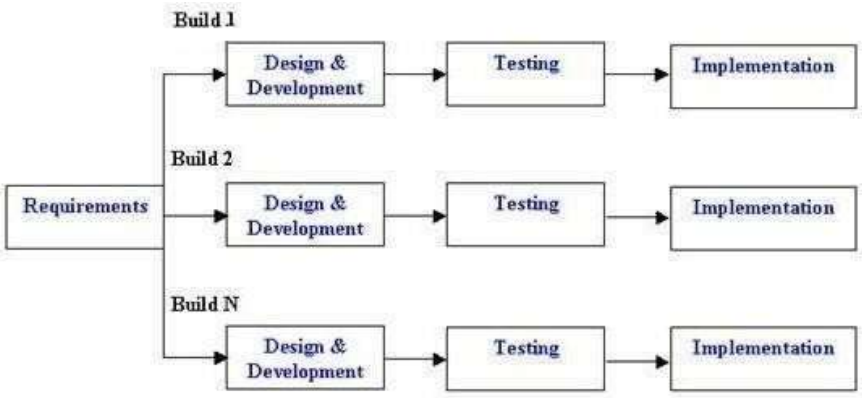


Figure 6 Incremental Life Cycle Model

3.6.2. Advantages of design Science Research

- Agile approach is best for software products due to following reasons:
- Visibility of project details
- Increased team efficiency
- Ability to adapt to change
- Ability to scale
- People before processes and tools
- Working software over documentation
- Collaboration over negotiation
- Change management over project planning

3.6.3. Reason for Chosen Methodology

The main reason why we choose this method is because we have followed the same steps in our research. First we have identified the problem area, and then we have defined the objectives for the solution of the problem with studying the related technologies. After that we have designed the application. We have also developed a prototype of the design and then we have demonstrated our solutions We have tested the application in lab environment. The last step of the DSRM is the future work of this research.

3.7. Project Structure

Working as a team reduces time and cost efforts on the project. We Ahmad Bin Asmat & Sardar Hamza worked together to develop this app and complete this document.The team structure we

adopted is the Democratic team structure. It suffers from less manpower turnover. Democratic team structure suits to less understood problems since a group of developers can invent a better solution than a single individual. It encourages programming as we can share and review one another work. We work by sharing ideas, problems and their solutions we divided tasks and completed them.

CHAPTER 4

SYSTEM DESIGN

4.1. Introduction

The smart courier service design and implementation are constrained by corporate policies,
regulatory requirements, hardware limitations, and specific technologies and tools. Compliance with
data privacy and security standards, transportation regulations, and e-commerce platform integration is mandatory. Additionally, developers must adhere to industry-standard design
patterns, programming languages, and communication protocols

4.2. Purpose

The objective of this document is to define the detail design of the system in sufficient depth to make it possible to continue developing software while having a clear grasp of what is to be built and how it is expected to be built. This document's purpose is to allow for software development to continue. This paper contains all of the relevant information regarding the software and the system that will be constructed.

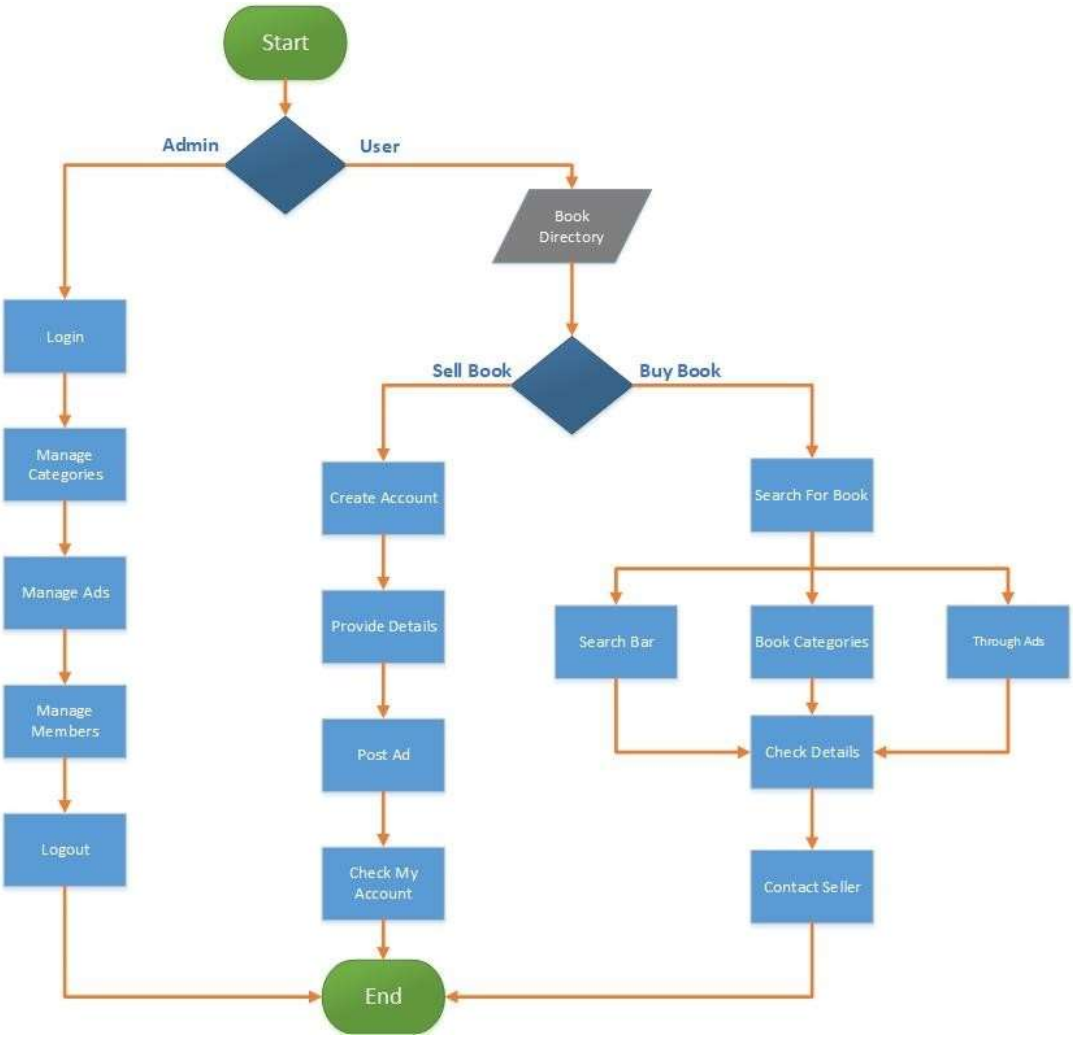


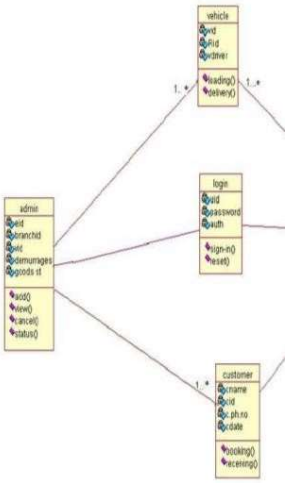
Figure 7 Flowchart

4.3. Scope

The software design specification is for a base level system which will work as a proof of concept for the use of building the system that provides base level functionality to show the feasibility and the advantages to the people through this system. For this particular design document, the main focus is on the people to help them in emergency situations.

4.4. Class Diagram

Class diagram is a static diagram and it is used to model the static view of a system. The static view describes the vocabulary of the system. A class diagram is a type of static structure diagram in Unified Modelling Language (UML) that represents the structure of a system or software application in terms of classes, their attributes, methods, and the relationships between them. It provides a visual representation of the classes, their properties, and the associations or dependencies among them.



4.5. Activity Diagram

Activity diagram illustrate the work-flow and activities within a specific use case or process. We created activity diagrams for our key processes like Sign-up, Login, Post jobs. These diagrams help visualize the steps involved and decision points within each process e.g. in case of sign-up the user has to fill out a form and is the input is valid then the account will be created otherwise if the input is not valid then the system will show an error and user will be redirected to sign-up form.

4.5.1. Activity Diagram (Sign-up)

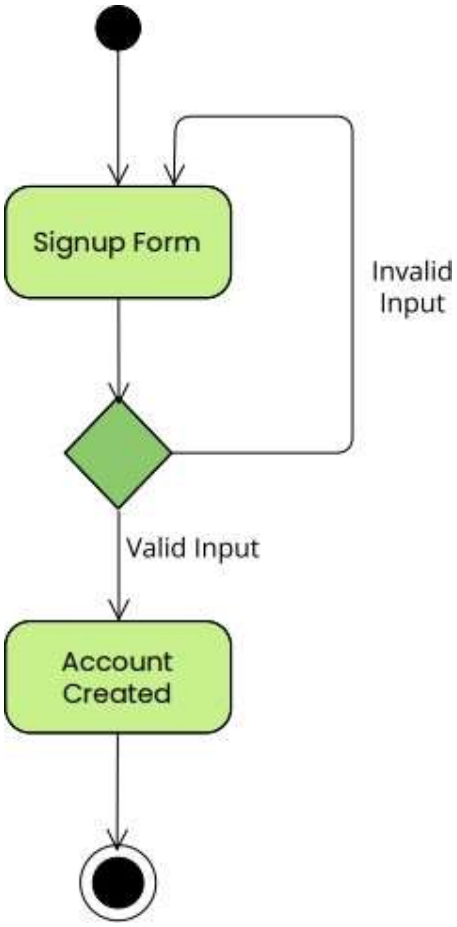


Figure 9 Activity Diagram for Sign-up

4.5.2. Activity Diagram (Login)

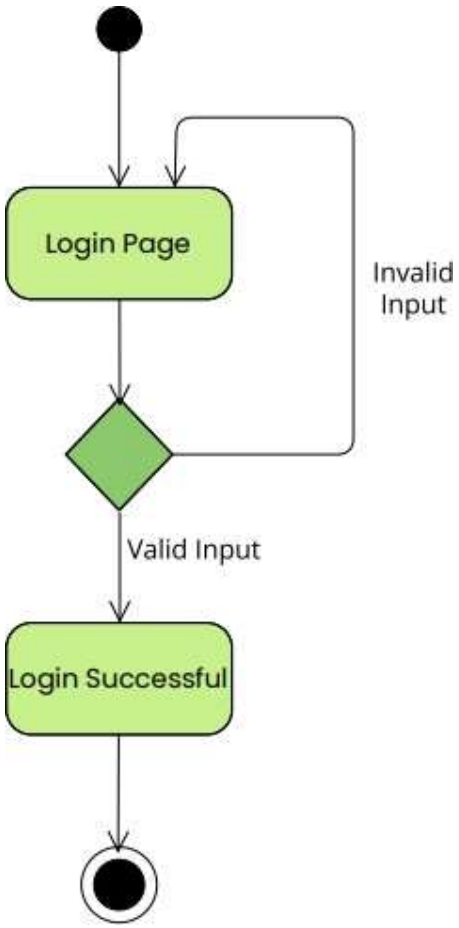


Figure 10 Activity Diagram for Login

4.5.3. Activity Diagram (Post Jobs)

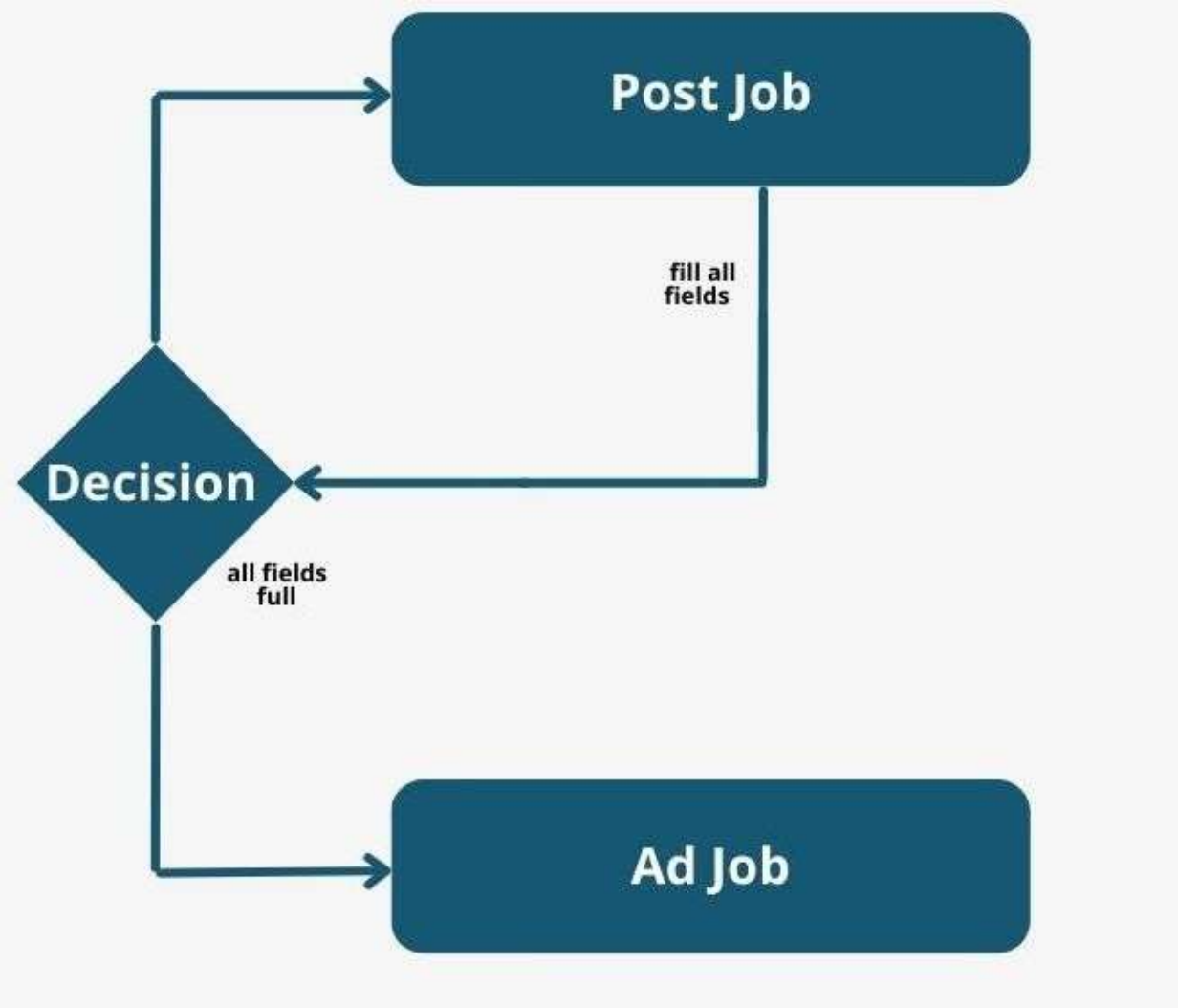


Figure 11 Activity Diagram for Post Job

4.6. Data Flow Diagram

Data flow diagrams provides an overview of our mobile application's interactions with external entities. In case of our application level 0 DFD represents the overview of our application.

In case of level 1 DFD we dive deep into the system and illustrates Data Flow through different components which include sign-up, login, search , contact, post jobs, my ads and my account.

When we go on step further on level 2 DFD it shows the data flow between different components and we can gain insights into how data is processed, shared, and stored within

our system.

4.6.1. Level 0

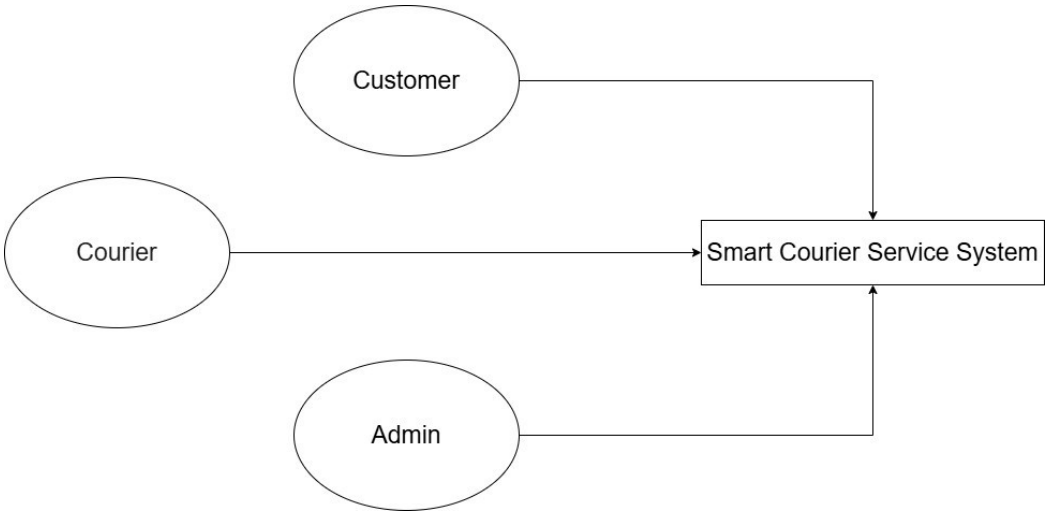


Figure 12 DFD Level 0

4.6.2. Level 1

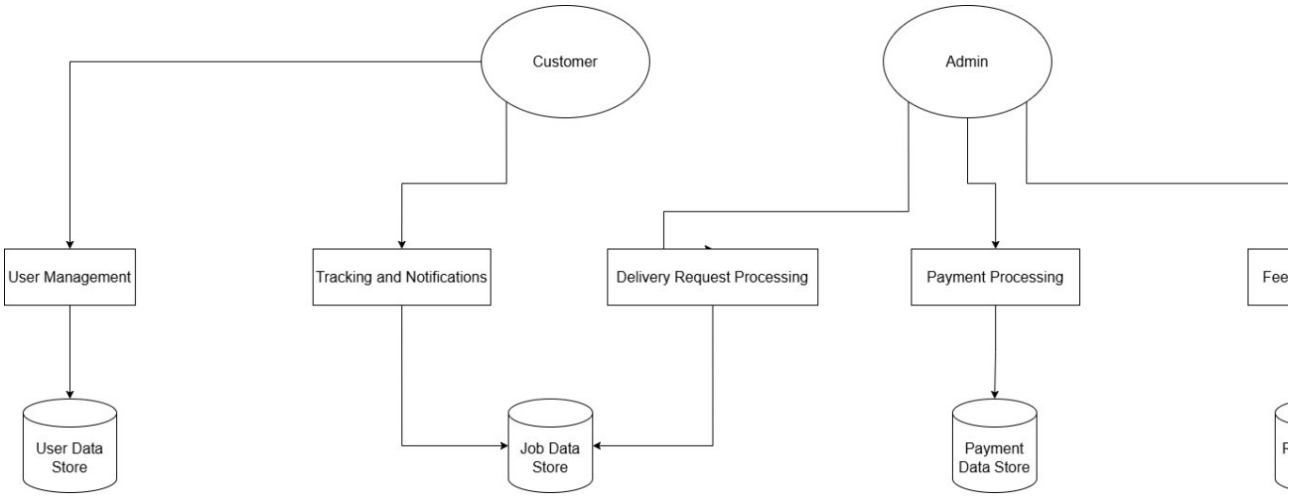


Figure 13 DFD Level 1

4.6.3. Level 2

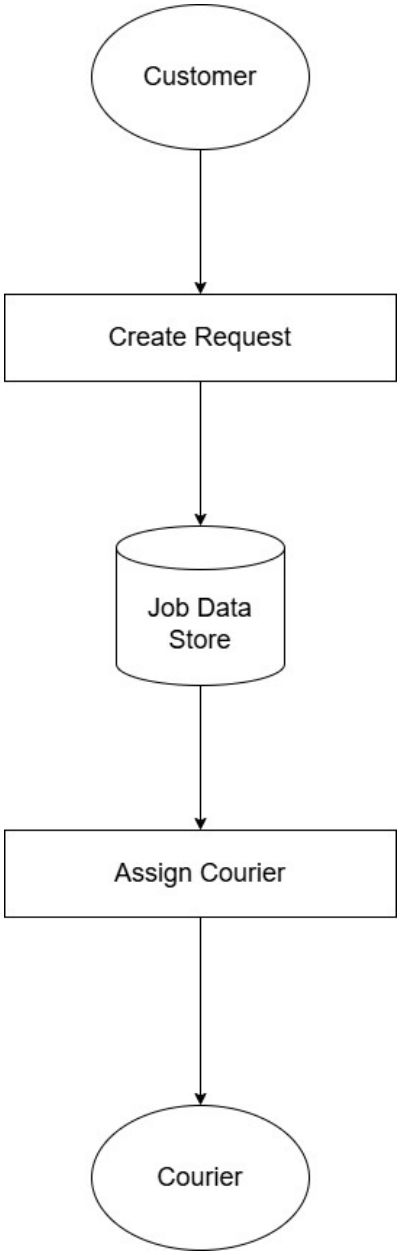


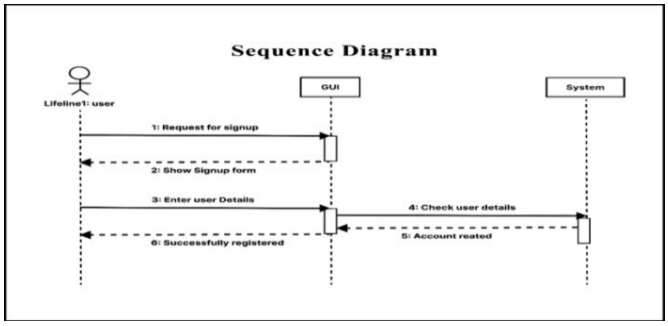
Figure 14 DFD Level 2

4.7. Sequence Diagram

Sequence diagram displays the interactions and messages exchanged between components over time. For our android application, we created sequence diagrams for common user journeys, such as user and admin. These diagrams show the flow of interactions between the user and the system e.g. in case of how the user will login into the system and post or search jobs.

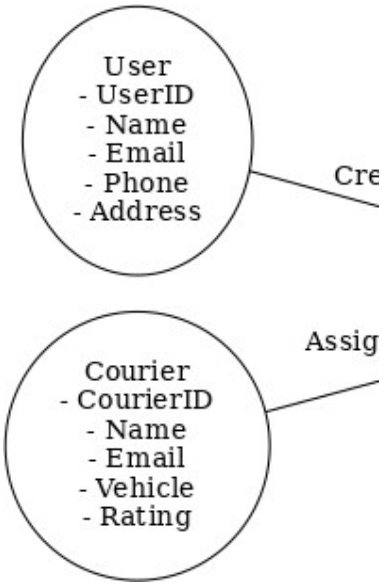
4.7.1. Sequence Diagram

Figure 15 Sequence Diagram



4.8. ERD

An Entity Relationship Diagram, also known as ERD, is a visual representation used to model the data structure and relationships within a database system. For our android application, an ERD will help us design the database schema and illustrate how various entities are related to one another.



4.9. Graphical User Interface (GUI)

4.9.1. Post Ad Screen

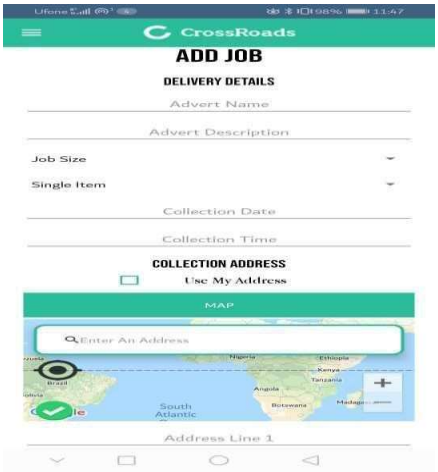


Figure 23 Post jobs Screen

4.9.2. My Ads Screen

CHAPTER 5

SYSTEM TESTING

5.1. Testing Introduction

Once the system has been successfully developed, testing has to be performed to ensure that the system working as intended. This is also to check that the system meets the requirements stated earlier. Besides that, system testing will help in finding the errors that may be hidden from the user. There are few types of testing which includes the unit testing, functional testing and integration testing. The testing must be completed before it is being deploy for user to use.

5.2. Purpose of Testing

- To check the usability of the system.
- To check the reliability of the system.
- For to find the errors in the system.
- For to ensure that the system is build according to the requirements.
- To check the performance of the system.

5.3. Objectives of Testing

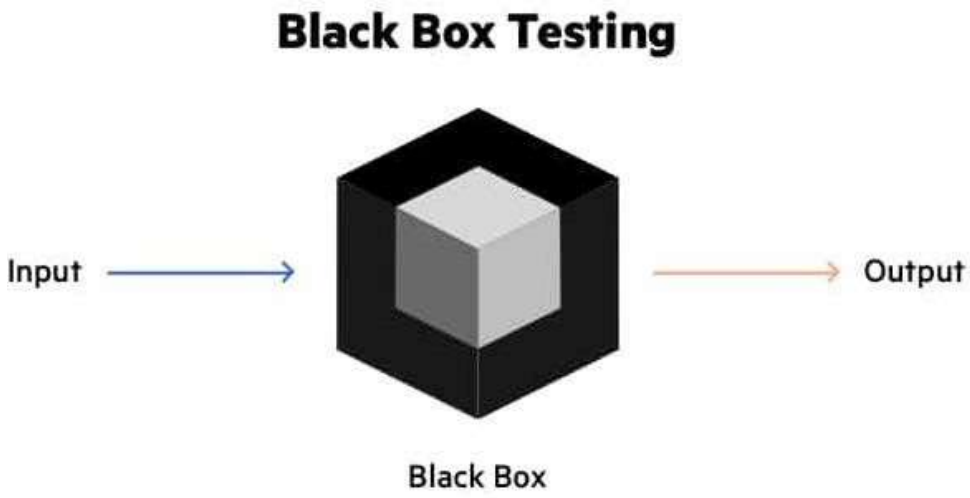
- Software testing has many goals and objectives, but the major objectives of software testing are as follows:
- Finding defects in the software system to be ensure that the software is created correctly, and no bugs left in the development of the system.
- To make sure that the end results meet the business and user requirements.
- To prevent defects.
- To make sure that the software meets the SRS document of the system.
- To gain confidence of the customer by providing the quality of the system.

5.4. Testing Techniques

Software testing techniques help to design better test cases. Since exhaustive testing is not possible; Manual Testing Techniques help reduce the number of test cases to be executed while increasing test coverage. They help identify test conditions that are otherwise difficult to recognize.

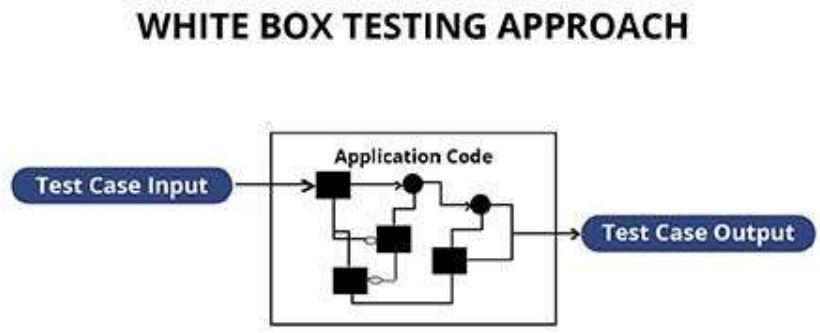
5.4.1. Black Box Testing

Black box testing is usually done by giving some inputs to the system and check the output of the system if the given inputs match with the result then over testing can be completed successfully and we don't have to check the internal logics of the system which are done in white box testing. Black box testing takes an external perspective of the test object to drive test cases. These tests are usually functional, but it can be functional and non-functional both. In black testing we don't need to have the proper knowledge of the internal functions of the system we just give inputs to the system and check the outputs of the system.



5.4.2. White Box Testing

In white box testing technique the tester is aware of the internal workings of the product, has access to its source code, and is conducted by making sure that all internal operations are performed according to the specifications. This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one clearly sees.



5.5. Black Box Test Cases

5.5.1. Black box testing for Name

Interface Name: First Name

ECP_ID: 1

Valid: {A-Z, a-z}

Invalid: {0-9, @, #, $, /}

Table 8 Black box testing for First Name

Test case id	***Input***	***ECP***	***Expected Output***	***Actual Output***
1	*Ahmad*	*{A-Z, a-z}*	*Accept the name*	*Valid*
2	*Ahmad123*	*{0-9, @, #, $, /}*	*Reject the name*	*Invalid*

Interface Name: Last Name

ECP_ID: 2

Valid: {A-Z, a-z}

Invalid: {0-9, @, #, $, /}

Table 9 Black box testing for Last Name

Test case id	***Input***	***ECP***	***Expected Output***	***Actual Output***
1	*Hamza*	*{A-Z, a-z}*	*Accept the name*	*Valid*
2	*Hamza123*	*{0-9, @, #, $, /}*	*Reject the name*	*Invalid*

5.5.2. Black box testing for Password

Interface Name: Password

ECP_ID: 3

Valid: {8>=password<=21}

Invalid: {8<password>21}

Table 10 Black box testing for Password

Test case id	*Input*	*ECP*	*Expected Output*	*Actual Output*
1	*khan12345*	*{8>=password<=21}*	*Accept the password*	*Valid*
2	*123khan*	*{8<password>21}*	*Reject the password*	*Invalid*
3	*123#. 22*	*{8<password>21}*	*Reject the password*	*Invalid password*

5.5.3. Black box testing for Email

Interface Name: Email

ECP_ID: 4

Valid: {a-z, @, 1-9, domain}

Invalid: {A-Z, £, &, #, $, /, missing domain}

Table 11 Black box testing for Email

Test case id	*Input*	*ECP*	*Expected Output*	*Actual Output*
1	*abc1@gmail.com*	*{a-z, @,1-9, domain*	*Accept the email*	*Valid*
2	*123AB@khan.co m*	*{A-Z, £, &, /, missing domain}*	*Reject the email*	*Invalid*

5.5.4. Black Box Testing for Phone Number

Interface Name: Phone Number

ECP_ID: 5

Valid: {0-9=11}

Invalid: {11>0-9<11}

Table 12 Black box testing for Phone Number

Test case id	*Input*	*ECP*	*Expected Output*	*Actual Output*
1	*03185552192*	*{0-9=11}*	*Accept the number*	*Valid*
2	*03185552182*	*{11>0-9<11}*	*Reject the number*	*Invalid*
3	*03151313999124*	*{11>0-9<11}*	*Reject the number*	*Invalid*

5.6. White box Test Cases

5.6.1. White Box Test Case Using Multiple Path Coverage Code

```
back.setOnClickListener(new
    View.OnClickListener() { @Override
public void
    onClick(View v) {
    finish();
    Intent intent = new Intent(forget_password.this, login_with_pin.class);
    start Activity(intent);
    }
    });
reset.setOnClickListener(new
    View.OnClickListener() { @Override
public void on Click(View v) {
if (editText.getText().toString().isEmpty()) {
```

```
        Toast.make Text(forget_password.this, "Enter Valid Email Address",
        Toast.LENGTH_SHORT).show();
}
else {
        auth.sendPasswordResetEmail(editText.getText().toString())
        .addOnCompleteListener(new OnCompleteListener<Void>() {
        @Override
public void on Complete(@Non Null Task<Void> task) {
        textViewResult.set Visibility(View.VISIBLE);
if (task.is Successful()) {
        textViewResult.setText("Password Reset Instructions has been sent to " +
        editText.getText().toString() + ", Kindly check your mail and complete the
        process.");
        editText.setText("");
} else {
        textViewResult.setText("Unable to send password reset instructions! please check
        your email and try again.");
}
```

5.6.2. White Box Test Cases

Table 13 White Box Test Cases

Id	Condition	Input	Output
1	Enter email	Valid	Email Sent
2	Enter email	Invalid	Not Registered
3	Enter password	Valid	Login
4	Enter password	Invalid	Error

5.6.3. Results

Table 14 Test Cases Path Coverage

Test case Id.	Path Covered
1	ab
2	ah
3	abcdeg
4	abcfg

CHAPTER 6

CONCLUSION & FUTURE WORK

6.1. Conclusion

Efficient logistic System is one of the leading factors associated with boosting trades in countries with thriving economies. Pakistan one of the leading economies in Asia with a very active manufacturing sector largely has logistic systems at the core of its economic activities. Distribution networks like courier services enable goods and services to be sent from the provider to the final consumer. Recent research shows that the public have expressed average satisfaction with the current system and there is an obvious need to improve upon current method of distributing goods and services via courier services. This research paper studies the possibilities of implementing a system on Android Based Smartphones to ease and make access to courier services more convenient and efficient. The researchers designed questionnaire using Scale Systems and distributed them to Courier Service Users. Analysis of respondents' feedback shows the obvious need for the implementation of courier service mobile application based on Android Operating System platform.

6.2. Future Work

In terms of a further research, the author suggests that an application could be designed to improve the information flow between SCS and the consumers. Since most of the respondents in this study were relatively young, they are probably using smart phones and would, therefore, be willing to use an application designed to shipment monitoring and communication purposes. A more detailed contents and further design of the application would be worth investigating.

REFERENCES

1. Lyda M.S. Lau, Jayne Curson, Richard Drew, Peter Dew and Christine Leigh, (1999), Use Of VSP Resource Rooms to Support Group Work in a Learning Environment, ACM 99, pp-2.

2. Hideyuki Nakanishi, Chikara Yoshida, Toshikazu Nishmora and TuruIshada, (1996), FreeWalk: Supporting Casual Meetings in a Network, pp 308-314 http://www.acm.org/pubs/articles/proceedings/cscw/240080/p308-nakanishi.pdf

3. Ali Behforooz& Frederick J.Hudson, (1996), Software Engineering Fundamentals, Oxford University Press. Chapter 8, pp255-235.

4. Page Author, Page Title, http://www.bt.com/bttj/archive.htm, Last date accessed.

[10] Testing Techniques https://www.geeksforgeeks.org/differences-between-black-box-testing-vswhite-box-testing/