



Project Title

**“Order place and maintain system”**

Submitted By

**Ahmed Musa**

152-35-1141

Supervised By

**Ms. TapusheRabayaToma**

Lecturer

Department of Software Engineering  
Faculty of Science & Information Technology  
Daffodil International University

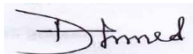
A project submitted in partial fulfillment of the requirement for the degree of  
Bachelor of Science in Software Engineering

**Department of Software Engineering**  
**DAFFODIL INTERNATIONAL UNIVERSITY**

Spring– 2019

## Letter of Approval

This project report “**Order place and maintain system**” prepared by **Ahmed Musa (ID: 152-35-1141)** is approved in Partial Fulfillment of the requirement for the Degree of BACHELOR OF SOFTWARE ENGINEERING. The said student have completed his project work under my supervision. During the working period I found him sincere, hardworking and enthusiastic.



---

**Ahmed Musa**

ID: 152-35-1141

Batch : 17<sup>th</sup>

Department of Software Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Supervised By**



---

**Ms. Tapushe Rabaya Toma**

lecturer

Department of software engineering,  
Daffodil International University.

## **Acknowledgement**

First of all, I am very much blessed as I have successfully reached towards the final semester. And so this is why I am pleased to my almighty. From the very beginning of my university life, I have learned a lot about software engineering as well as core computer science related knowledge from my course teachers. Moreover, they teach us ethics, morality and politeness.

Besides, I am so much thankful to my parents that, I was always supported by them. They always prioritize my opinion and inspired me as well.

I am also so many grateful to my supervisor **Ms. TapusheRabayaToma** for allowing me to work on this project. She always supports me to make this projects successful. Specially, I am very thankful to my brother **Mutasimbillah** who always appreciate my professionalism.

## **Dedication**

I would like to dedicate this project to my respectable Elder brother and one of friends, my instructor, my Honorable teachers who are always dear and near to me and without whose patience, care, understanding, unsparing support, affection and most of all deepest love it would not have been possible to come up to this position.

## **Abstract**

This project is aimed to developing an online order place. The entire project has been developed keeping in view of the distributed client server computing technology, in mind. The project entitled “Order place and maintain system” is a web application. It is developed usingLaravel(PHP Framework), used Bootstrap 4 for frontend and MySQL. This project has tried to incorporate all the advanced features of PHP such as to fulfill the aim of the project. The application has been designed such that it encompasses the managerial as well as public functionalities. Among the managerial functionalities are the task performed by the various offices. The public features are exercised by the users and outsiders according to the permissions level availed to them.

## Table of Contents

Chapter: 01 .....	1
Introduction.....	1
1.1 Project Overview .....	2
1.2 Project Purpose .....	2
1.3 Stakeholders.....	3
1.4 Project Schedule.....	4
Chapter: 02.....	6
Software Requirement Specification .....	6
2.1 Functional Requirements .....	7
2.2 Data Requirements .....	12
2.3 Performance Requirements .....	12
2.4 Dependability Requirements .....	13
2.5 Maintainability & Supportability Requirements.....	14
2.6 Security Requirements.....	15
2.7 Usability and Human-Interaction Requirements .....	16
2.8 Look and Feel Requirements .....	17
2.9 Operational and Environmental Requirements.....	17
2.10 Legal Requirements .....	18
Chapter: 03.....	19
Requirement Analysis.....	19
3.1 Use Case Diagram.....	20
3.2 Activity Diagram .....	26
3.3 Sequence Diagrams.....	32
Chapter: 04.....	37
System Design Specification .....	37
4.1 Development tools and technology .....	37
4.2 Class Diagram .....	40
4.3 Database Design Diagram .....	40
Chapter: 05.....	41

System Test.....	41
5.1 Testing Features .....	42
5.2 Testing Strategy.....	42
5.3 Testing Schedule .....	44
5.4 Testing Environment .....	45
5.6 Test Cases.....	45
Chapter: 06.....	48
User Manual.....	48
6.1 Website.....	49
6.2 Customer login page .....	49
6.3 Customers landing page .....	50
6.4 Making order by photo ID.....	50
6.5 Making new order by uploading photo.....	51
6.6 Order confirmation .....	52
6.7 Selecting payment gateway .....	52
6.8 Paying by bKash .....	53
6.9 Paying by Card .....	53
6.10 Customer orders history .....	54
6.11 Customer saved order.....	55
6.12 Focus point/Studio location .....	55
6.13 Sending message to admin.....	56
6.14 Customers profile .....	56
6.15 Customers logout.....	57
6.16 Admin login .....	57
6.17 Admins Dashboard.....	58
6.18 All orders.....	58
6.19 Photo gallery .....	59
6.20 Pending task .....	59
6.21 Undelivered orders.....	60
6.22 Photo size and price settings .....	60
6.23 Customers observing .....	61
6.24 Messages .....	61

Chapter: 07 .....	62
Conclusion .....	62
7.1 GitHub Link .....	63
7.2 Project Summary .....	63
7.3 Limitations.....	63
7.4 Obstacles and Achievements.....	63
7.5 References .....	64



## List of figures

Figure 1.0: Gantt Chart .....	4
Figure 3.1: Use Case diagram for “Order place and maintain system” .....	21
Figure 3.2: Activity diagram for Registration .....	26
Figure 3.3: Activity diagram for Login .....	27
Figure 3.4: Activity diagram for Print Stored Photo .....	27
Figure 3.5: Activity diagram for Print Non-Stored Photo .....	28
Figure 3.6: Activity diagram for Save order .....	28
Figure 3.7: Activity diagram for Payment .....	29
Figure 3.8: Activity diagram for Exact Map .....	30
Figure 3.9: Activity diagram for Search Order .....	30
Figure 3.10: Activity diagram for Pending Task .....	31
Figure 3.11: Activity diagram for Undelivered Orders .....	31
Figure 3.12: Activity diagram for Customer Observing .....	32
Figure 3.13: Sequence Diagram for Print Stored Photo .....	32
Figure 3.14: Sequence Diagram for Print Non-Stored Photo .....	33
Figure 3.15: Sequence Diagram for Save order .....	33
Figure 3.16: Sequence Diagram for Payment .....	34
Figure 3.17: Sequence Diagram for Exact Map .....	35
Figure 3.18: Sequence Diagram for Pending Task .....	35
Figure 3.19: Sequence Diagram for Undelivered Orders .....	36
Figure 3.20: Sequence Diagram for Customer Serving .....	36
Figure 4.0: Class Diagram .....	40
Figure 4.1: Database Diagram .....	40
Figure 6.1: Website .....	49
Figure 6.2: Customer login page .....	50
Figure 6.3: Customers landing page .....	50
Figure 6.4: Making new order by photo id .....	51
Figure 6.5: Making new order by uploading photo .....	52
Figure 6.6: Confirm order .....	52
Figure 6.7: Selecting payment method .....	53
Figure 6.8: Paying by bKash .....	53
Figure 6.9: Paying by card .....	54
Figure 6.10: Orders history and status .....	54
Figure 6.11: Saved order .....	55
Figure 6.12: Focus Point .....	55
Figure 6.13: Sending message to admin .....	56
Figure 6.14: Customers profile .....	56
Figure 6.15: Customers logout .....	57
Figure 6.16: Admin login page .....	57
Figure 6.17: Admins dashboard .....	58

Figure 6.18: All orders.....	58
Figure 6.19: Photo gallery .....	59
Figure 6.20: Pending task .....	59
Figure 6.21: Undelivered Orders .....	60
Figure 6.22: Photo size and price settings.....	60
Figure 6.23: Customer observing .....	61
Figure 6.24: Messages from customers .....	61

# **Chapter: 01**

## **Introduction**

## **1.1 Project Overview**

We are living in a generation where technologies are like magic. Here, whatever you want to do you can do that. It just matter of little bit time and trick of hands. Our job is make things weird to ordinary, darkest to brightest, Inflexible to flexible, and here is my project to achieve our goal. I am working for a Photography company named 'Focus Digital Color Lab' which is prints photos for customers for last more than ten years.

## **1.2 Project Purpose**

Until this time the company take orders manually. Like, customer come to them → for place an order → pay cash by hand → take a receipt → come again next day for collecting their photos. On the other hand employee takes orders in a ledger book, they have to write all information about orders. I can see that here's the problem. By this old process it takes too much time to accepting a order, takes money and space for the ledger book and stuffs. So I offered them to do the same thing automatically.

### **1.2.1 Background**

I went to print my photo from "Focus Digital Color Lab", Suddenly I realize that the way they prints customers photo this is too much old and time consuming process and also expensive. So, I decided to offer them a paperless way where customers can give orders by their own, and the manager can see customers orders.

### **1.2.2 Beneficiaries & Benefits**

My application would be beneficial for some point of view. Now, I am mentioning those below:

#### **Beneficiaries**

- Studio owner
- Customer

#### **Benefits**

- A great paperless way to maintaining orders
- Look after orders per week, per months, per year
- Customers increasing by online marketing
- Business growth
- Customers knows what is the condition of his/her order
- Easy communication with customers

- Getting customers information by their registration process
- No negotiation with service cost
- Knowing customers information
- Influencing design for customers
- Customers don't need to think about photo printing sizes
- Online payment system
- Customers can change their info at anytime
- Storing previous orders of customer
- Facility to paying later for unpaid orders
- Reducing customers time
- Quick customer support

I have mentioned some benefits as well as beneficiaries. So, I think this application is very much helpful for customer and the studio owner.

### 1.2.3 Goals

The main goals of this project is to develop an application based on web. As more than 90% of people are uses internet on their mobile device or on their Computer at home, office, university etc. So my target is to develop a system for easy going life.

### 1.3 Stakeholders

- Studio owner
- Customers
- Bkash
  
- **Studio owner:** Studio owner have an individual panel of this application named Admin Panel. Admin who accepts orders from customer and maintain those until delivered to the customer.
  
- **Customer:** Customers have an individual panel of this application named Customer panel. By this panel customer can register their user account and can give orders from it and also can see what is the condition of his/her order so that customer can collect his/her photos from the studio.
  
- **Bkash:** Bkash is the only payment gateway for this application. I build an API for completing transactional process of customers payment. So, the system have to send request to bkash cluster server and bkash cluster server has to be responded.

## 1.4 Project Schedule

I need to prepare a scheduling plan to complete the project on time. It also refers to make communication with what task need to get done within timeframe.

### 1.4.1 Gantt Chart

Gantt chart is mainly a production control tools. It remained us to complete our assigned tasks within a certain period of time. For developing software, it is mostly used. Now I will show a Gantt chart for my project.

Activities		W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
<b>Planning</b>	Ideas																
	Problem definition																
	Proposal planning																
<b>Requirements</b>	Requirement specification																
	Requirement analysis																
<b>QA – 1</b>	Quality assurance																
<b>System design</b>	Sketching																
	Design specification																
	Database design																
<b>Implementation-1</b>	Crud operation																
<b>QA – 2</b>	Test cases																
<b>Implementation-2</b>	Photo uploading																
<b>Testing</b>	Unit testing																
	Blackbox testing																
<b>Delivery</b>	Software release																
<b>Scheduled time</b>																	
<b>Buffered time</b>																	

Figure 1.0: Gantt Chart

## 1.4.2 Release Plan or Milestone

The release plan or milestones are given below:

<b>Activities</b>	<b>Duration in week</b>	<b>Total week</b>
<b>Brainstorming</b>	Week 1	1
<b>Problem identification</b>	Week 1, Week 2	2
<b>Requirement specification</b>	Week 2	1
<b>Requirement analysis</b>	Week 2	1
<b>Sketching</b>	Week 4	1
<b>Design specification</b>	Week 4	1
<b>Database design</b>	Week 5	1
<b>Customer experience</b>	Week 5, Week 6, Week 7, Week 8	4
<b>Quality assurance</b>	Week 3,	1
<b>Test case</b>	Week 3, Week 7, Week 8, Week 9	4
<b>Getting feedback</b>	Week 10, Week 11, Week 12, Week 13	4
<b>Unit testing</b>	Week 11, Week 12	2
<b>Black-box testing</b>	Week 13, Week 14, Week 15	3
<b>Software release</b>	Week 16	1

# **Chapter: 02**

## **Software Requirement Specification**



## 2.1 Functional Requirements

Functional requirements refer to the functions which are mandatory to the system. Functional requirements must be able to perform on the software system. Every system must have some functional requirements. Now, I am going to mention functional requirements associating with my project.

### 2.1.1 Photo studio based website

<b>Requirements 1</b>	<b>Photo studio based website</b>
<b>Description</b>	A Studio website, where people can know about studio, product and service charge.
<b>Stakeholders</b>	Customer

### 2.1.2 Registration process

<b>Requirements 2</b>	<b>Registration process</b>
<b>Description</b>	Studio customers have to sign up for the first time to place an order for service.
<b>Stakeholders</b>	Customer

### 2.1.3 Login process

<b>Requirements 3</b>	<b>Login process</b>
<b>Description</b>	A Studio website, where people can know about studio, product and service charge.
<b>Stakeholders</b>	Customer

### 2.1.4 Print stored photo

<b>Requirements 4</b>	<b>Print stored photo</b>
<b>Description</b>	A copy of a photo has stored in studios database with a Photo Id.
<b>Stakeholders</b>	Customer

### 2.1.5 Print non stored photo

<b>Requirements 5</b>	<b>Print non stored photo</b>
<b>Description</b>	Photos are uploaded by customers with different sizes.
<b>Stakeholders</b>	Customer

### 2.1.6 Photo id for print

<b>Requirements 6</b>	<b>Photo id for print</b>
<b>Description</b>	A specific photo id for stored photo provided by studio.
<b>Stakeholders</b>	Customer

### 2.1.7 Gallery photo

<b>Requirements 7</b>	<b>Gallery photo</b>
<b>Description</b>	Soft copy of photos customers wants to print.
<b>Stakeholders</b>	Customer

### 2.1.8 Photo size for print

<b>Requirements 8</b>	<b>Photo size for print</b>
<b>Description</b>	Have to select which sizes are want a customer to print.
<b>Stakeholders</b>	Customer

### 2.1.9 Quantity for print

<b>Requirements 9</b>	<b>Quantity for print</b>
<b>Description</b>	Have to select how many photos customers want to print.
<b>Stakeholders</b>	Customer

### 2.1.10 Confirm order

<b>Requirements 10</b>	<b>Confirm order</b>
<b>Description</b>	Check all the arrangement before place a order.
<b>Stakeholders</b>	Customer

### 2.1.11 Save order

<b>Requirements 11</b>	<b>Save order</b>
<b>Description</b>	Customers can save their orders to pay later.
<b>Stakeholders</b>	Customer

### 2.1.12 Orders history

<b>Requirements 12</b>	<b>Orders history</b>
<b>Description</b>	Customers can view their orders history.
<b>Stakeholders</b>	Customer

### 2.1.13 Orders status

<b>Requirements 13</b>	<b>Orders status</b>
<b>Description</b>	Customers can watch their order status.
<b>Stakeholders</b>	Customer

### 2.1.14 Payment

<b>Requirements 14</b>	<b>Payment</b>
<b>Description</b>	After completing all the arrangements customers have to pay the service bill.
<b>Stakeholders</b>	Customer, Bkash

### 2.1.15 Support Message

<b>Requirements 15</b>	<b>Support Message</b>
<b>Description</b>	Customer will get support if they needed, its link one-to-one offline chat.
<b>Stakeholders</b>	Customer

### 2.1.16 Exact Map

<b>Requirements 16</b>	<b>Exact Map</b>
<b>Description</b>	Customers will get exact location of that studio.
<b>Stakeholders</b>	Customer

### 2.1.17 Admin Login

<b>Requirements 17</b>	<b>Admin Login</b>
<b>Description</b>	Admin can login with special account to monitor and maintain orders.
<b>Stakeholders</b>	Studio Owner

### 2.1.18 Orders

<b>Requirements 18</b>	<b>Orders</b>
<b>Description</b>	A page of all orders.
<b>Stakeholders</b>	Studio Owner

### 2.1.19 Search Orders

<b>Requirements 19</b>	<b>Search Orders</b>
<b>Description</b>	Admin can search orders by photo id.
<b>Stakeholders</b>	Studio Owner

### 2.1.20 Orders Statistics

<b>Requirements 20</b>	<b>Orders Statistics</b>
<b>Description</b>	A graphical view of orders by current month and current year.
<b>Stakeholders</b>	Studio Owner

### 2.1.21 Pending Task

<b>Requirements 21</b>	<b>Pending Task</b>
<b>Description</b>	Those orders which is reached but employees didn't work on it.
<b>Stakeholders</b>	Studio Owner

### 2.1.22 Undelivered Orders

<b>Requirements 22</b>	<b>Undelivered Orders</b>
<b>Description</b>	Those orders which is done, but customers didn't collect it.
<b>Stakeholders</b>	Studio Owner

### 2.1.23 Customers Observing

<b>Requirements 23</b>	<b>Customers Observing</b>
<b>Description</b>	Watching customers list and active or inactive them.
<b>Stakeholders</b>	Studio Owner

### 2.1.24 Support Reply

<b>Requirements 24</b>	<b>Support Reply</b>
<b>Description</b>	Admin will reply of customer messages.
<b>Stakeholders</b>	Studio Owner

### 2.1.25 Set size and price list

<b>Requirements 25</b>	<b>Set size and price list</b>
<b>Description</b>	Admin will set what will be the price for specific size.
<b>Stakeholders</b>	Studio Owner

## 2.2 Data Requirements

To defining data requirements for my application almost all of data would be loaded from remote users. And for that purpose I need to focus on some major points. Such as:

- Types of entity of the system
- Route data locations
- Capacity and resources of the data requirements
- Data source sequence
- Data availability schedules
- Quantity of data
- Availability of data

## 2.3 Performance Requirements

It is very important to maintain performance of any software system. To ensure performance, I need to maintain some steps. Now, I will explain some perspective by which I am going to enhance the performance of my project.

### 2.3.1 Speed & Latency Requirements

Speed and latency requirements must be ensured while retrieving data from the cloud server.

<b>SLR-1</b>	Search result must be faster.
<b>Description</b>	When traffic police search for a vehicle, then the search result must show within seconds.
<b>Stakeholders</b>	Studio Owner

### 2.3.2 Precision & Accuracy Requirements

Results that is to be shown to the end user is need to be accurate. Because, wrong information might be ruined the whole business process.

<b>PAR-1</b>	Search result must be accurate
<b>Description</b>	When studio owner search for a order, then the search result must be according to the input value given by Studio owner.
<b>Stakeholders</b>	Studio Owner

### 2.3.3 Capacity Requirements

The developed system by me must be capable to handle user data, provide accurate information, handling database, manage http request etc.

<b>CR-1</b>	The system will handle thousands of data.
<b>Description</b>	The system need to handle data thousands of data every moment.
<b>Stakeholders</b>	Studio Owner

### 2.4 Dependability Requirements

The term dependability is measured based on four dimensions. Such as:

1. Availability
2. Reliability
3. Safety
4. Security

If I want to say that my application system is dependable then it must fulfill the four dimensions. But there are other tasks. Like there is no way to make mistakes or my system should have the ability to detect and then remove errors. Besides that, it is also very important to limit the damage which might be caused by system failure.

#### 2.4.1 Reliability & Availability Requirements

Now, I will mention requirements which is related to reliability and availability.

<b>PAR-1</b>	The system must be available on 24 X 7
<b>Description</b>	<ul style="list-style-type: none"><li>● Our system must be available all day long, every day in a week</li><li>● The system must be updated regularly</li><li>● System must be malware free</li></ul>

<b>Stakeholders</b>	Studio Owner, Customer
---------------------	------------------------

### 2.4.2 Robustness or Fault-Tolerance Requirements

To ensure robustness and fault-tolerance facilities to the end users, it is urgent to ensure 0% crash. Moreover, it must show accurate results.

<b>RFT-1</b>	The system handles all user access without system errors
<b>Description</b>	Thousands of user might hit our application system at a time. All their requests must be handled without any fault.
<b>Stakeholders</b>	N/A

### 2.4.3 Safety-Critical Requirements

There are only safety-critical requirements in my project, and that is payment transaction ID.

### 2.5 Maintainability & Supportability Requirements

It is very important to provide after service or support to the end users.

#### 2.5.1 Maintainability Requirements

<b>MR-1</b>	System helps to update user profile
<b>Description</b>	It is very important to update user profile.
<b>Stakeholders</b>	Studio Owner, Customer

#### 2.5.2 Supportability Requirements

Supportability requirements may have related to some extends. Like:

- Testability
- Extensibility
- Adaptability
- Maintainability
- Compatibility
- Configurability
- Serviceability
- Install ability

My application meets all of the above requirements related to supportability.



### 2.5.3 Adaptability Requirements

There are no adaptability requirements in our system software.

## 2.6 Security Requirements

Making software security as a requirement is very important. Software security requirements should be its functional requirement. Software security enforces security of an application system. Functionality related to software security can either be directly tested or observed. Some security related requirements is given below:

- Signing in a customer
- Signing in a admin
- Get access according to logged in user
- Upload photos to the server without having any issue as a customer
- Maintaining all orders without having any issue as a admin
- Signing out as a customer and admin
- Handling encrypted passwords

While accessing to the system, each and every features must provide a central authentication mechanism. There is also a process to prevent entering into the system by ensuring hashed password for the unauthenticated users.

### 2.6.1 Access Requirements

For accessing to our application system, there remains some authentication and authorization techniques. And every features of my system will provide it. Now I will provide an explanation below.

<b>AR-1</b>	Application provides security mechanism.
<b>Description</b>	Every features is designed in such a way that it only give access to the authorized and authenticated users.
<b>Stakeholders</b>	Studio Owner, Customer

### 2.6.2 Integrity Requirements

Integrity requirements refers to a security system which ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to the malicious modification or accidental destruction. For that reason, I will store system user passwords as encrypted format which is impossible to decrypt. It is also called hashed password.

### 2.6.3 Privacy Requirements

It is very important to ensure privacy of the system users. Privacy requirements enhances to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database should be protected.

## 2.7 Usability and Human-Interaction Requirements

The main target of developing any system is to make the system user friendly and easy to usable for the end users.

### 2.7.1 Ease of Use Requirements

My application is easy to use and also easily understandable.

<b>EUR-1</b>	Application must be usable for the end users.
<b>Description</b>	This apps is enough usable to the traffic police by which they can operate this system easily.
<b>Stakeholders</b>	Studio Owner, Customer

### 2.7.2 Personalization and Internationalization Requirements

There are not any personalization and internationalization requirements to our system. This maiden version of our application is only be operated by Bangladesh.

### 2.7.3 Understandability and Politeness Requirements

It is already said that the application which we are going to develop, is understandable enough. The system provides hints to users whether any error occurred or wrong. By reading those errors users can be able to operate the system easily.

### 2.7.4 Accessibility Requirements

There are no specific accessibility requirements associated to our system yet.

### 2.7.5 User Documentation Requirements

Documentation are mainly two types. One is internal documentation which is generally written by the application engineers. It is prepared to make development life cycle easier for the system engineers or system analysts.

<b>UDR-1</b>	The system engineer documentation
<b>Description</b>	To develop my application named Automatic order place and

	maintaining system, firstly I have make a system analysis team as well as documentation team.
<b>Stakeholders</b>	System analysts or software developers.

## 2.7.6 Training Requirements

Training requirements involved in after service of any application. It is very necessary to properly train up end users to the system so that they would be capable to operate easily. After launching the full package to the market, firstly I provide training to the end users like, customer, studio owner.

## 2.8 Look and Feel Requirements

Look and feel requirements mainly refers how the system will look like and how the user interface or graphical user interface of my system will display to the user.

### 2.8.1 Appearance Requirements

General customer and studio owner must know which input fields are required and which are not. For that reason, I will use labels for all input fields. Input fields might be text type, radio, checkbox, spinner etc.

<b>AR-1</b>	Labels of mandatory fields must be bold.
<b>Description</b>	The mandatory field's label must be bold and all input fields must have placeholder to make it easier for the users.
<b>Stakeholders</b>	System analysts or software developers.

## 2.9 Operational and Environmental Requirements

Operational and environmental requirement refers to the capabilities, performance measurements, process, measurements of effectiveness, measurements of performance, measures of sustainability, measurements of technical performances etc.

### 2.9.1 Expected Physical Requirements

There are no expected physical requirements in our system.

## **2.9.2 Requirements for Interfacing with Adjacent Systems**

There are no requirements for interfacing with adjacent system for my project.

## **2.9.3 Release Requirements**

There are no specific release requirements in our system.

## **2.10 Legal Requirements**

Legal requirements normally refer to the terms and conditions or privacy policy of any organizations. The terms and condition of my application is that, no third party software or person are allowed to engage to use this system data for their business purpose.

### **2.10.1 Compliance Requirements**

There are no specific compliance requirements for our system.

### **2.10.2 Standards Requirements**

There are no specific standards requirements for our system.

# **Chapter: 03**

## **Requirement Analysis**

### **3.1 Use Case Diagram**

I have use case diagram. And there are two actors. Each actor plays different role. And those are already indicated to this use case diagram. This diagram will clarify our system in brief.

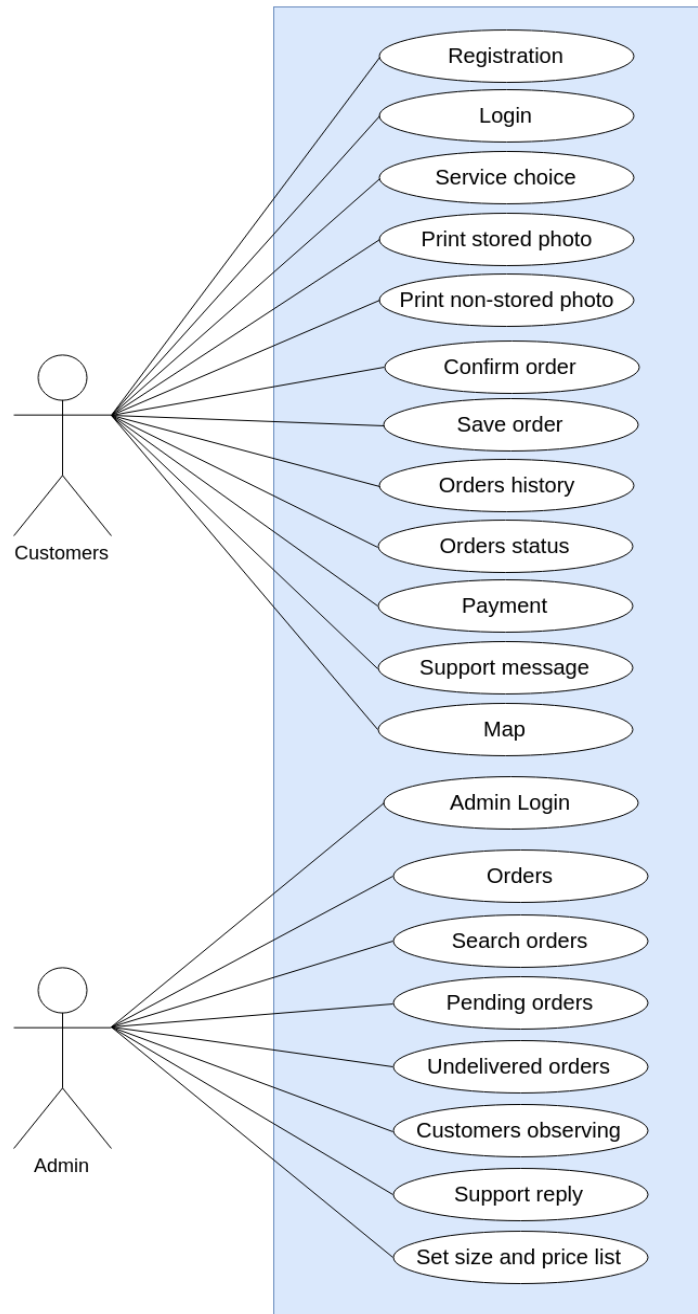


Figure 3.1: Use Case diagram for “Order place and maintain system”

### 3.1.1 Registration

<b>Use Case Title</b>	Customer will register
<b>Goal</b>	Customer save their information to the server
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>➤ User can be anyone</li> <li>➤ User type must be customer</li> </ul>
<b>Success End Condition</b>	Customer successfully insert related data
<b>Failure End Condition</b>	Customer cannot be able to save data
<b>Primary Actors:</b> <b>Secondary Actors:</b>	Customer N/A
<b>Trigger</b>	Customer data visible to admin
<b>Description / Main Success Scenario</b>	Customer will add his own information to the system for registration approval
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

### 3.1.2 Login

### 3.1.3 Print stored photo

<b>Use Case Title</b>	Print stored photo
<b>Goal</b>	Customer can give order by photo id
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>➤ User have to be registered</li> <li>➤ User must have a photo id</li> </ul>
<b>Success End Condition</b>	Customer give order successfully
<b>Failure End Condition</b>	Customer cannot be able to give order by photo id
<b>Primary Actors:</b> <b>Secondary Actors:</b>	Customer N/A
<b>Trigger</b>	Order data visible to admin
<b>Description / Main Success Scenario</b>	Customer will give a order by photo id which is provided by photo studio
<b>Alternative Flows</b>	N/A



### 3.1.4 Print non-stored photo

<b>Use Case Title</b>	Print non-stored photo
<b>Goal</b>	Customer will order for photo which is not-stored
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>➤ User have to be registered</li> <li>➤ User must have to upload photo</li> </ul>
<b>Success End Condition</b>	Customer successfully give his/her order
<b>Failure End Condition</b>	Customer cannot be able to give order
<b>Primary Actors: Secondary Actors:</b>	Customer N/A
<b>Trigger</b>	Order data visible to admin
<b>Description / Main Success Scenario</b>	Customer will give order by their uploaded photo which is not stored in studios database
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

### 3.1.5 Payment

<b>Use Case Title</b>	Payment
<b>Goal</b>	Customer will pay for order
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>➤ User have to be registered</li> <li>➤ User must have to pay for order</li> </ul>
<b>Success End Condition</b>	Customer will pay successfully
<b>Failure End Condition</b>	Customer cannot be able to pay
<b>Primary Actors: Secondary Actors:</b>	Customer N/A
<b>Trigger</b>	Transaction data visible to admin
<b>Description / Main Success Scenario</b>	Customer will give payment which is estimated by photo studio
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

### 3.1.6 Search order

<b>Use Case Title</b>	Admin search order
<b>Goal</b>	Admin can search customer orders
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>➤ User have to be admin</li> <li>➤ User must have to give photo id or trx id</li> </ul>
<b>Success End Condition</b>	Admin will found orders successfully
<b>Failure End Condition</b>	Data not found
<b>Primary Actors: Secondary Actors:</b>	Admin N/A
<b>Trigger</b>	Orders data visible to admin
<b>Description / Main Success Scenario</b>	Admin will input users photo id or transaction id by this process admin will find a orders information
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

### 3.1.7 Pending Task

<b>Use Case Title</b>	Pending task
<b>Goal</b>	Admin can see which orders are pending
<b>Preconditions</b>	User have to be admin
<b>Success End Condition</b>	Admin will found pending orders successfully
<b>Failure End Condition</b>	Data not found
<b>Primary Actors: Secondary Actors:</b>	Admin N/A
<b>Trigger</b>	Pending orders data visible to admin
<b>Description / Main Success Scenario</b>	Admin will see users pending orders by this process
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

### 3.1.8 Undelivered orders

<b>Use Case Title</b>	Undelivered orders
<b>Goal</b>	Admin can see which orders are ready
<b>Preconditions</b>	User have to be admin
<b>Success End Condition</b>	Admin will found undelivered orders successfully
<b>Failure End Condition</b>	Data not found
<b>Primary Actors: Secondary Actors:</b>	Admin N/A
<b>Trigger</b>	Undelivered orders data visible to admin
<b>Description / Main Success Scenario</b>	Admin will see users ready but not delivered orders by this process
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

### 3.1.9 Customer observing

<b>Use Case Title</b>	Customer observing
<b>Goal</b>	Admin will see customers information
<b>Preconditions</b>	User have to be admin
<b>Success End Condition</b>	Admin will found customer information successfully
<b>Failure End Condition</b>	Data not found
<b>Primary Actors: Secondary Actors:</b>	Admin N/A
<b>Trigger</b>	Customer data visible to admin
<b>Description / Main Success Scenario</b>	Admin will found customer information and can active/inactive a user
<b>Alternative Flows</b>	N/A
<b>Quality Requirements</b>	N/A

## 3.2 Activity Diagram

I have prepared some activity diagram according to my use case. These activity diagrams are properly referring the flow of the individual conditions of the project.

### 3.2.1 Registration

User can register in this system by providing required information.

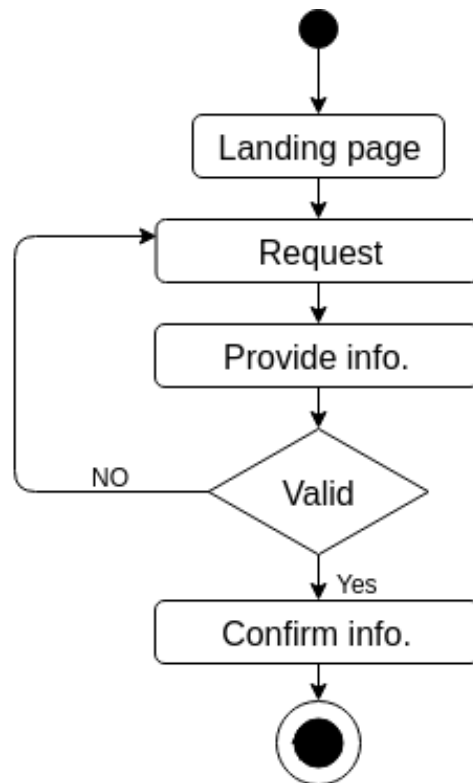


Figure 3.2: Activity diagram for Registration

### 3.2.2 Login

Registered user can login into the system so that they can access all the services.

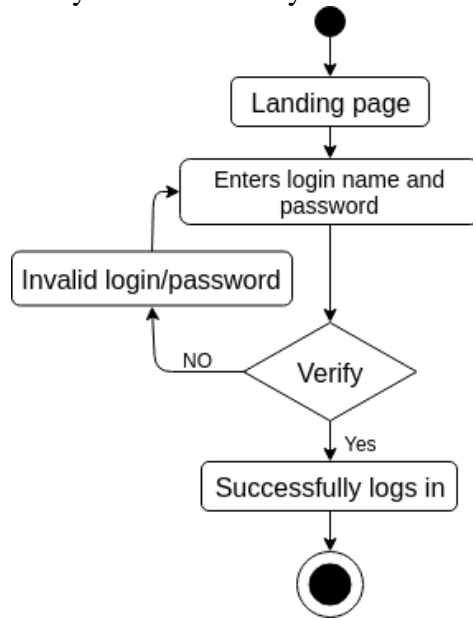


Figure 3.3: Activity diagram for Login

### 3.2.3 Print Stored Photo

Registered user can give order by photo id. Which is provided by studio owner.

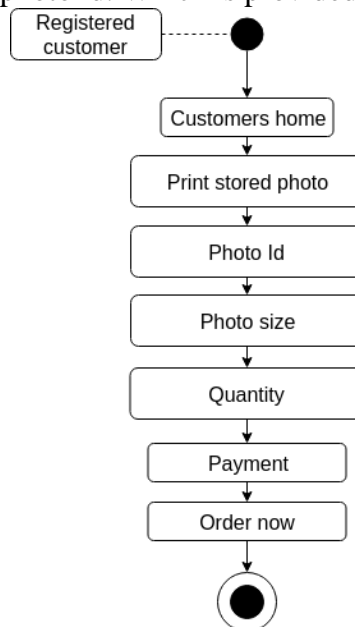


Figure 3.4: Activity diagram for Print Stored Photo

### 3.2.4 Print Non-Stored Photo

Registered user can give order by uploading photo to the system

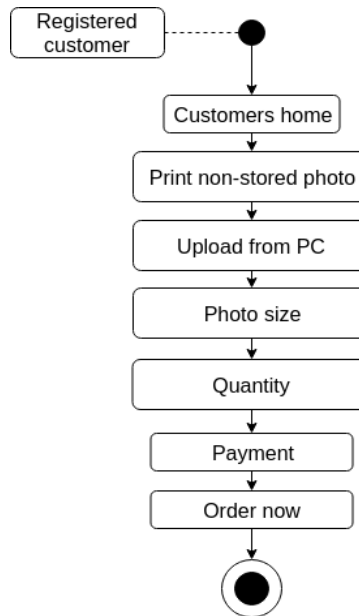


Figure 3.5: Activity diagram for Print Non-Stored Photo

### 3.2.5 Save Order

Registered user can save their order so that they can pay later to give a complete order.

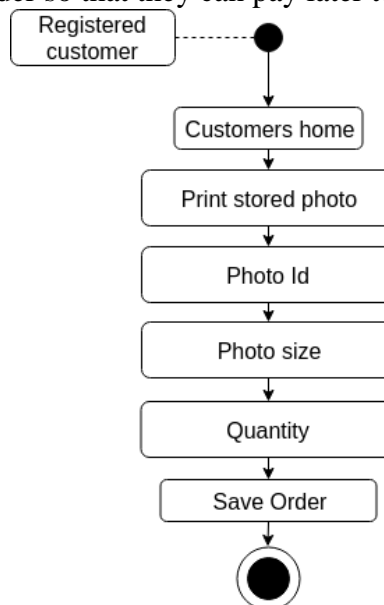


Figure 3.6: Activity diagram for Save order

### 3.2.6 Payment

When a customer will give a order they have to pay for it. This is the way how payment will work

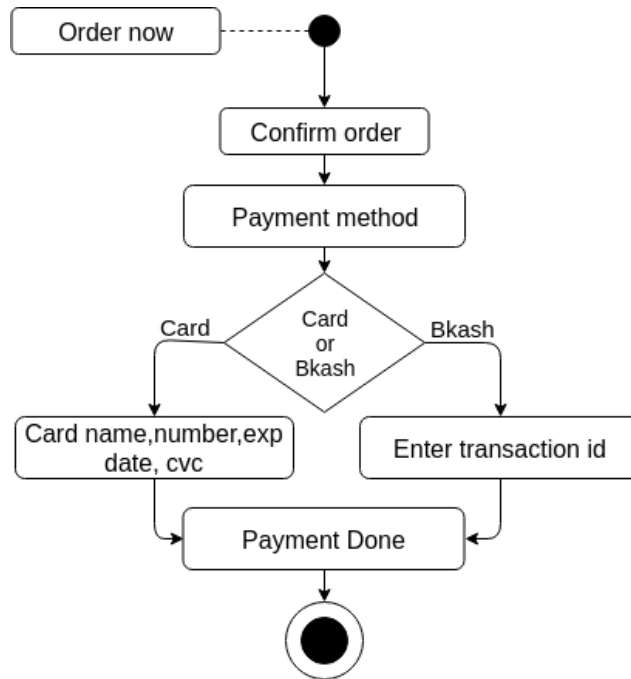


Figure 3.7: Activity diagram for Payment

### 3.2.7 Exact Map

Exact map/location of that studio

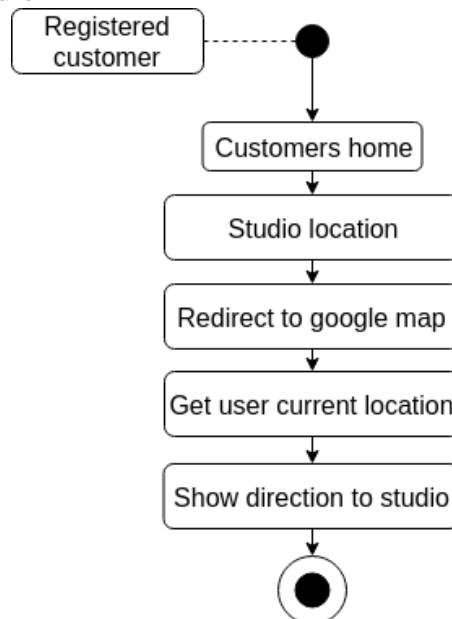


Figure 3.8: Activity diagram for Exact Map

### 3.2.8 Search Order

This function is for admin by this admin can search for a order by Photo id or transaction id

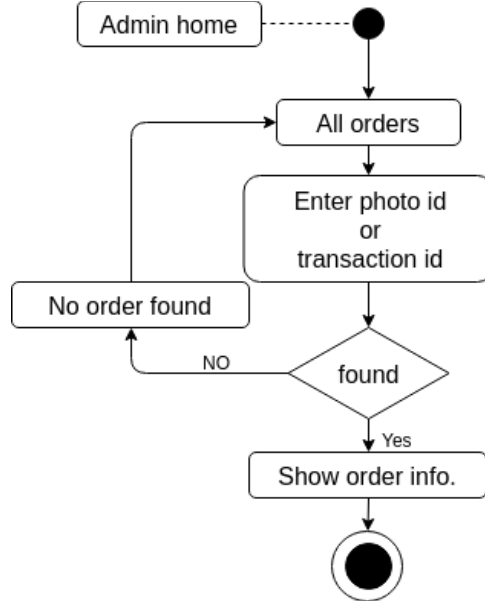


Figure 3.9: Activity diagram for Search Order

### 3.2.9 Pending Task

This function is for admin by this admin can see how many orders are pending to develop

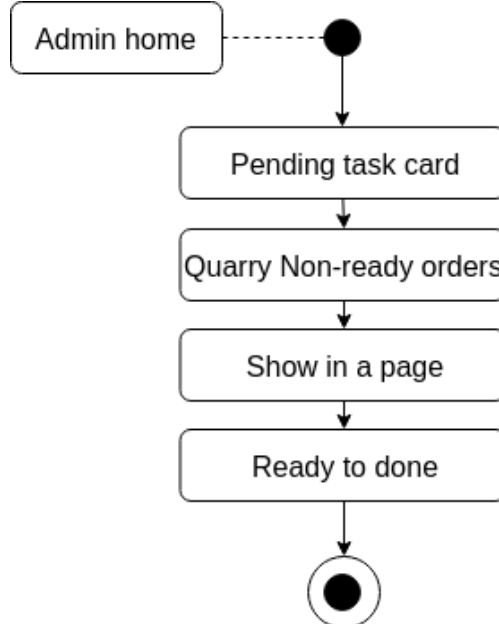




Figure 3.10: Activity diagram for Pending Task

### 3.2.10 Undelivered Orders

This function is for admin by this admin can see how many orders are ready but not delivered yet

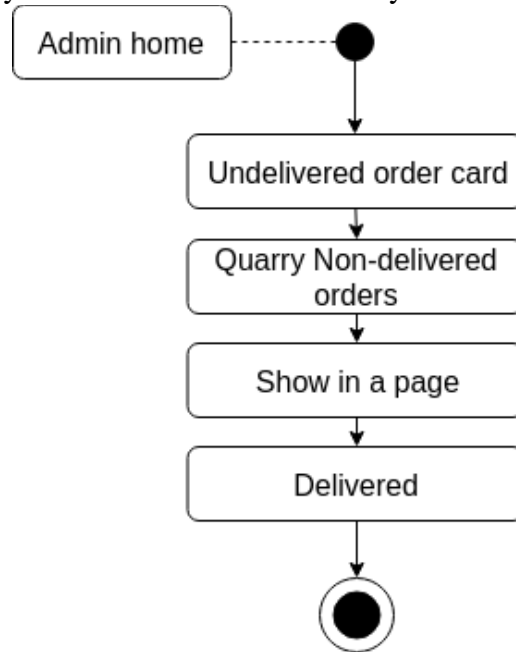


Figure 3.11: Activity diagram for Undelivered Orders

### 3.2.11 Customer Observing

This function is for admin by this admin can see Customers information and active/Deactive a customer

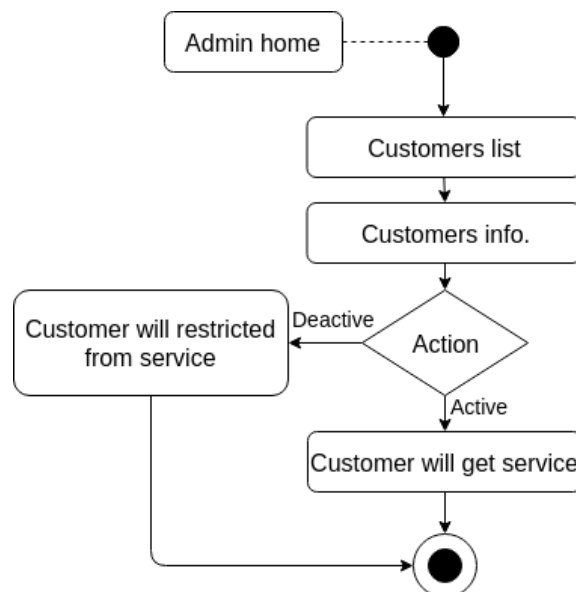


Figure 3.12: Activity diagram for Customer Observing

### 3.3 Sequence Diagrams

Mainly sequence diagrams understand us how the data will be followed in any application. Now I am going to show some sequence diagrams.

#### 3.3.1 Print Stored Photo

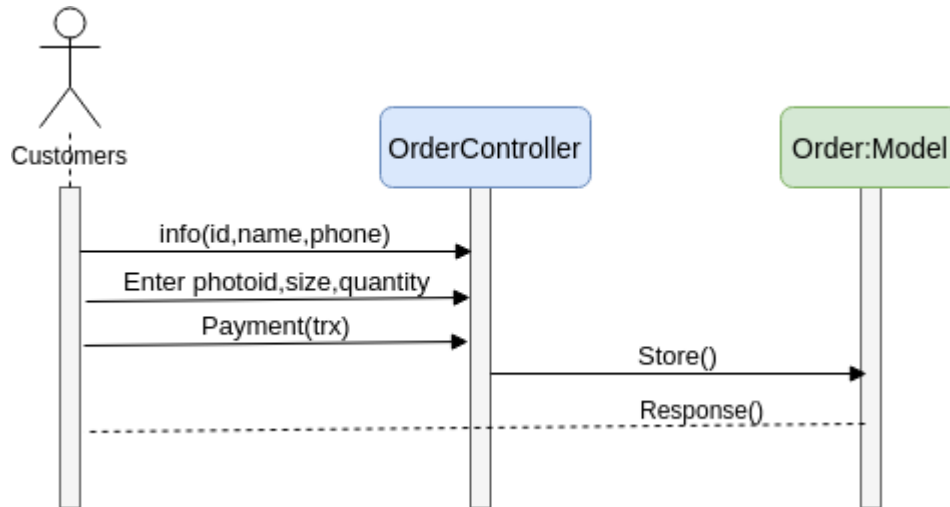


Figure 3.13: Sequence Diagram for Print Stored Photo

#### 3.3.2 Print Non-Stored Photo

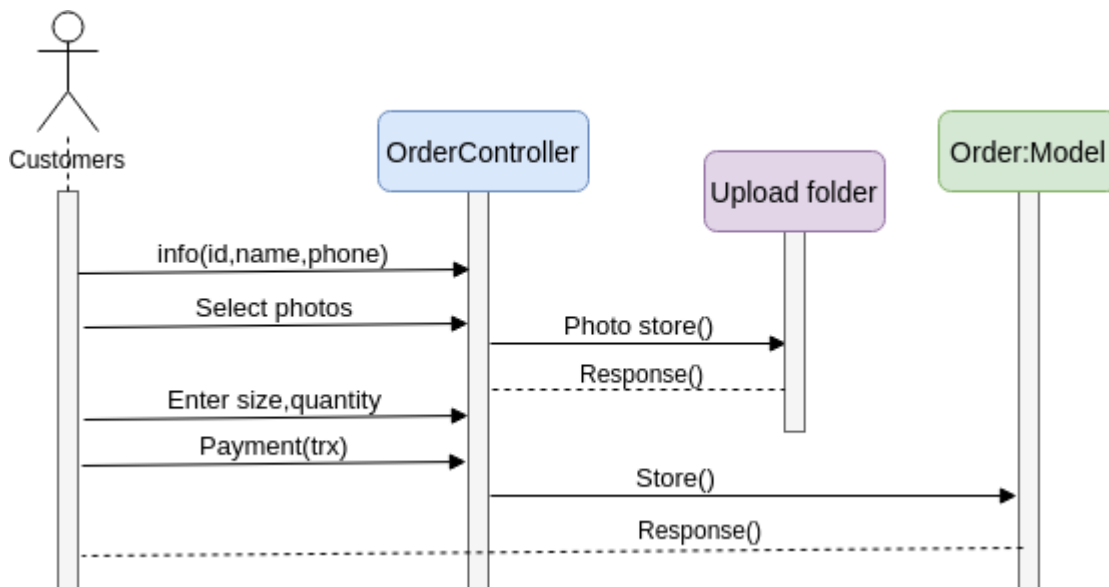


Figure 3.14: Sequence Diagram for Print Non-Stored Photo

### 3.3.3 Save Order

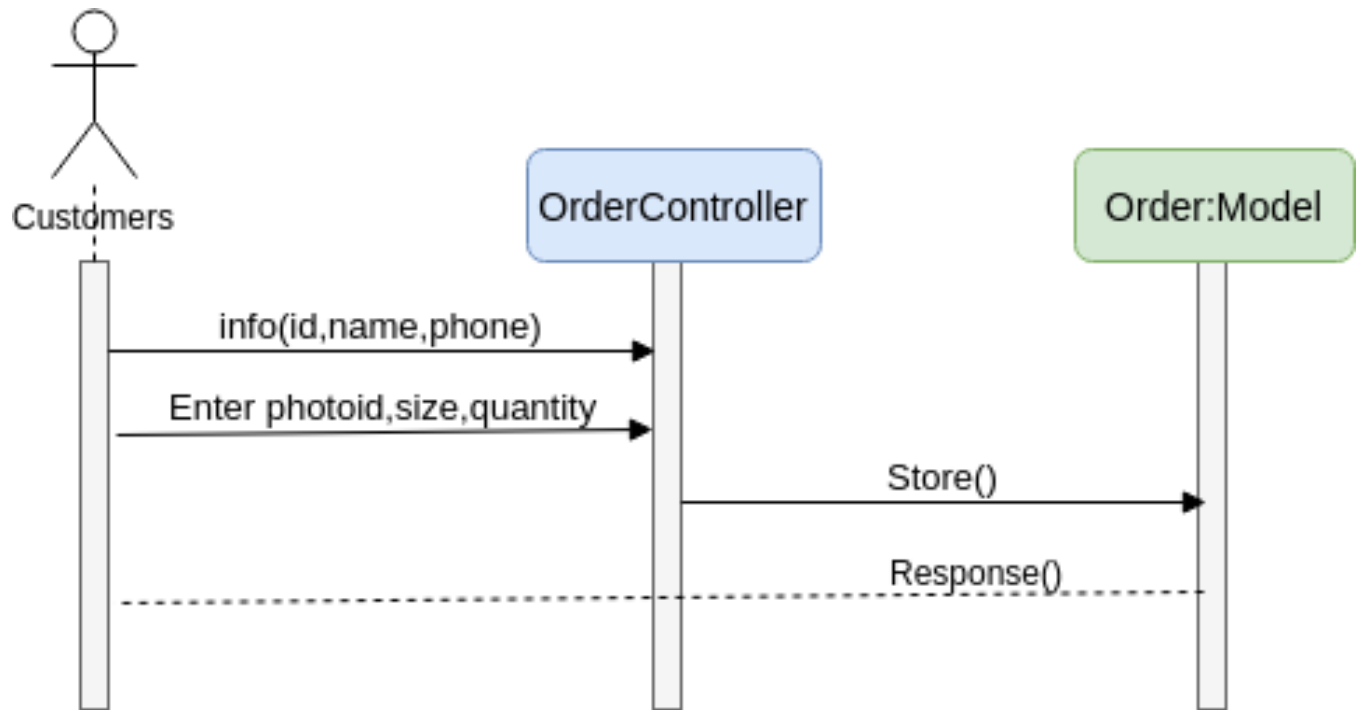


Figure 3.15: Sequence Diagram for Save order

### 3.3.4 Payment

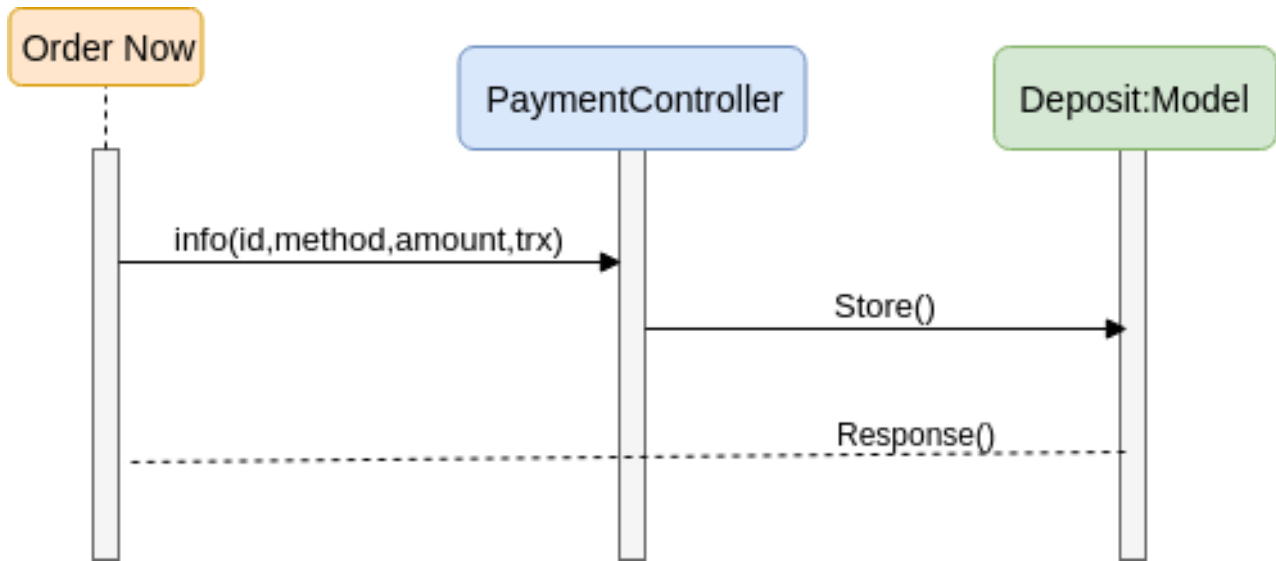


Figure 3.16: Sequence Diagram for Payment

### 3.3.5 Exact Map

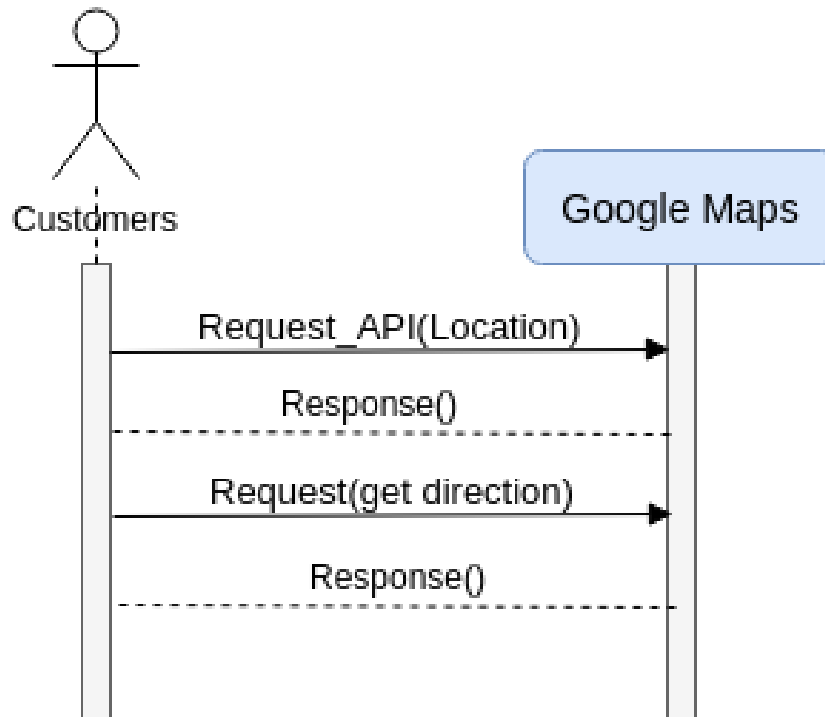


Figure 3.17: Sequence Diagram for Exact Map

### 3.3.6 Pending Task

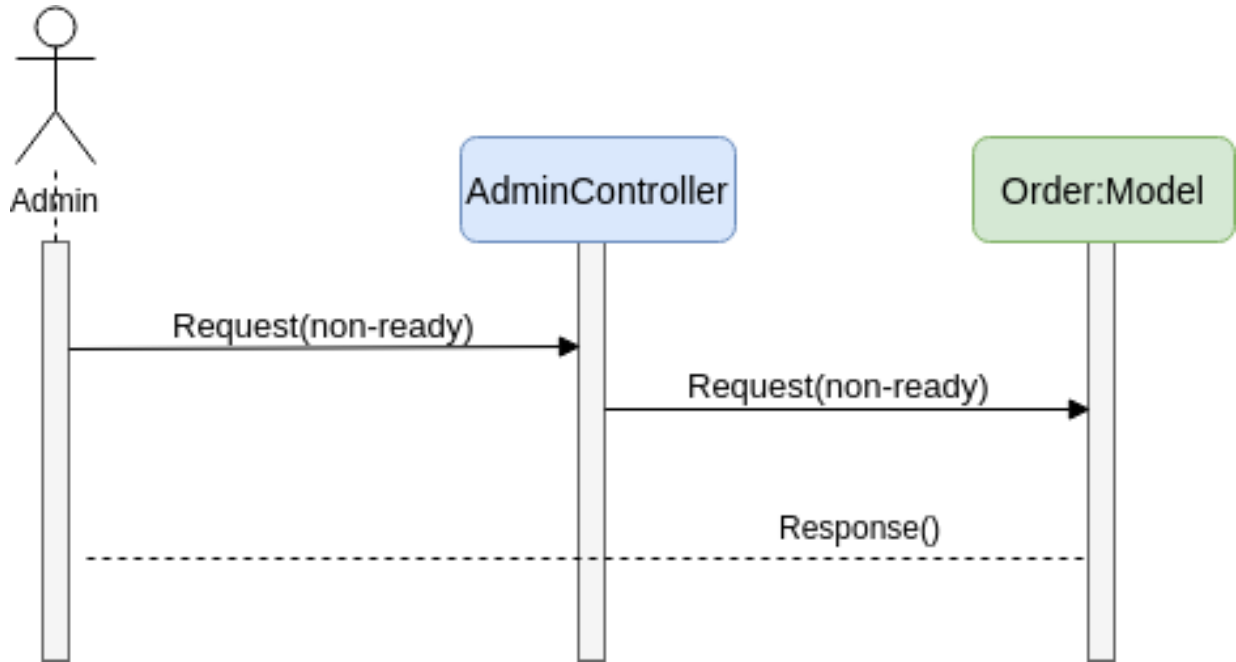


Figure 3.18: Sequence Diagram for Pending Task

### 3.3.7 Undelivered Orders

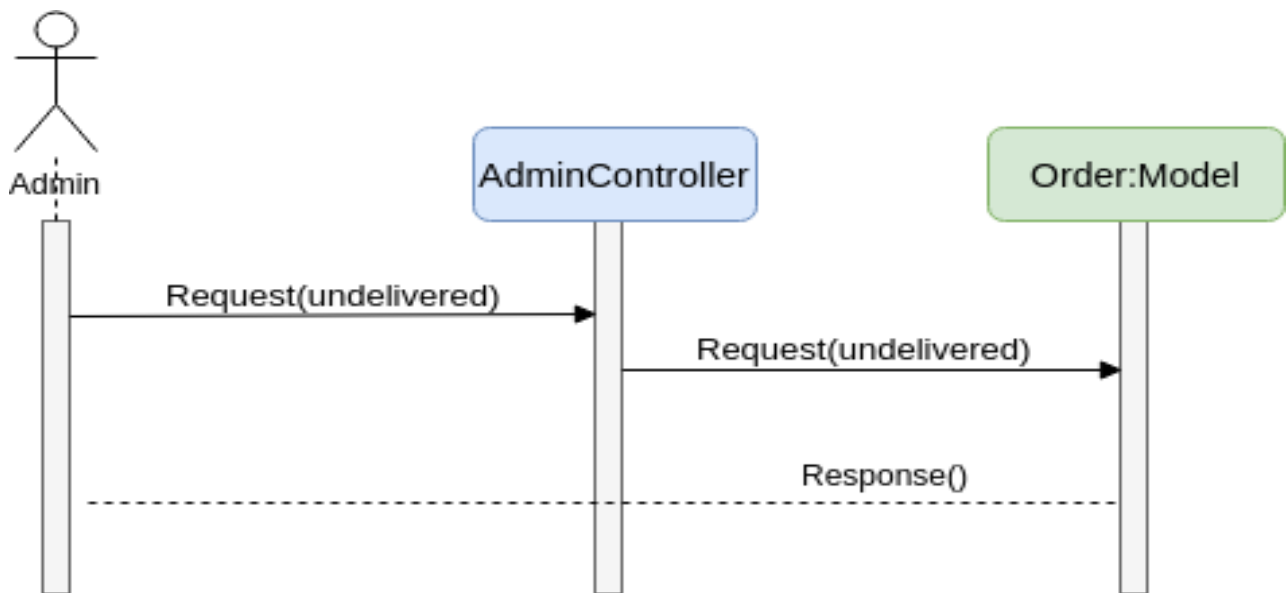


Figure 3.19: Sequence Diagram for Undelivered Orders

### 3.3.8 Customer Serving

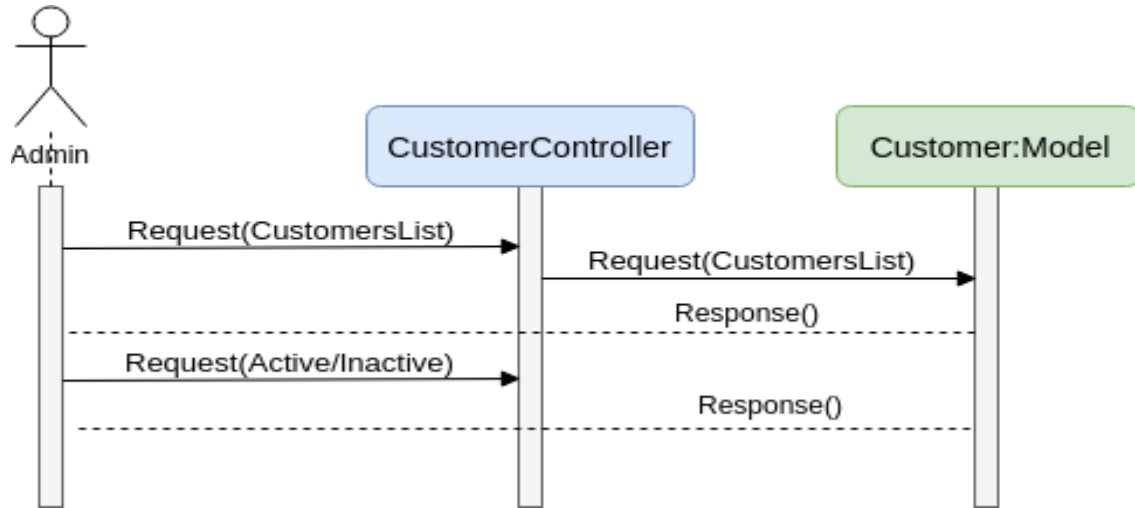


Figure 3.20: Sequence Diagram for Customer Serving

# Chapter: 04

## System Design Specification

### **4.1 Development tools and technology**

Software development tools are used to develop software. There are different kinds of tools for software developers to make the process of software development very smooth. But now I will mention some of tools that I am using to develop my project.

#### **4.1.1 User Interface Technology**

First of all, after landing any application, interface of that software is appeared to the user. So, the importance of user interface is very high. For success of any software application, a good looking user interface plays a vital role. User interface includes using good image, graphics, typography, stylesheets, scripting etc.

#### **4.1.1.1 CSS framework or Twitter Bootstrap**

After completing markup, cascading style sheets are designed. It generally explains how HTML elements will display. There are three ways to write stylesheet. One internal CSS, another is external CSS. And last one is inline CSS. But most of time external CSS is used. Because, by using external CSS, all CSS data can be kept in different files. But nowadays CSS framework seems to be very popular. In our application, I have used Twitter Bootstrap framework. This framework is developed mainly for CSS and JavaScript. It is a free and open source front-end framework. This framework is highly used for designing website and web application also. This framework supports almost every browser. Like Google Chrome, Mozilla Firefox, Opera, Safari, Internet Explorer etc. Bootstrap also provides media query which features responsive layout for different devices with different screen size. Bootstrap provides a set of some files which contains stylesheets which gives basic definition.

Bootstrap also provides some JavaScript components also. There are some built in components like JQuery UI. By using Bootstrap framework, we get both CSS and JavaScript facilities with a single platform. But before start designing an application interface with Bootstrap, one may have some basic knowledge about this framework. It will increase the efficiency.

#### **4.1.1.2 JQuery UI**

JQuery UI is a name of library which uses JavaScript as core programming language. It simplifies codes of a programming language named JavaScript. Generally, it interacts with Graphical User Interface (GUI). It also provides visual effects with animation. It can also override cascading style sheets. It also provides AJAX functionality which stands for Asynchronous JavaScript and XML. It is also a subset of JavaScript. By using AJAX no page reloading is required. It is also compatible with any browser like Google Chrome, Mozilla Firefox, Opera, Safari, Internet Explorer etc.

#### **4.1.1.3 Programming Language**

For developing any application system minimum one programming language is essential. In my application, there are two different programming language is used. One is for front-end side. And another is for server side. The front-end language that I have used to my application is JavaScript. And the server side programming language name is PHP which stands for Hypertext Preprocessor. Both of them are open source general purpose scripting language.



## **4.1.2 Implemented tools and platform**

As I have said before, there are some tools and technologies that need to be used for developing software. It is very important to determine which tools and platforms are the best match of my requirements. After making a proper decision, one need to start using them.

### **4.1.2.1 Integrated Development Environment**

IDE stands for Integrated Development Environment. Programmers write code on IDE. After that IDE provide the feature to execute the source code. For developing my web application, I have used two IDE. Because my project has two versions. One is for web version, and anotheris for mobile users. To develop my web version, I have used phpStromes which is powered by JetBrainscompany. It is a commercial IDE for cross platform environment. It is able to suggest code to the programmers also.

### **4.1.2.2 Web Server**

We have used apache server. It is a free and open source software to use. It can be used on cross platform. It supports a wide range of features and most of them are already implemented as compiled modules. This module can extend the main features or core functionality.

### **4.1.2.3 Database Server**

For developing our whole project, we have followed Relational Database Management System or RDBMS. And we find that MySql provides the feature of RDBMS. So we should not have any issue to use MySql database. It is also very easy to use. It can also ensure the security, scalability, high performance and many things.

## 4.2 Class Diagram

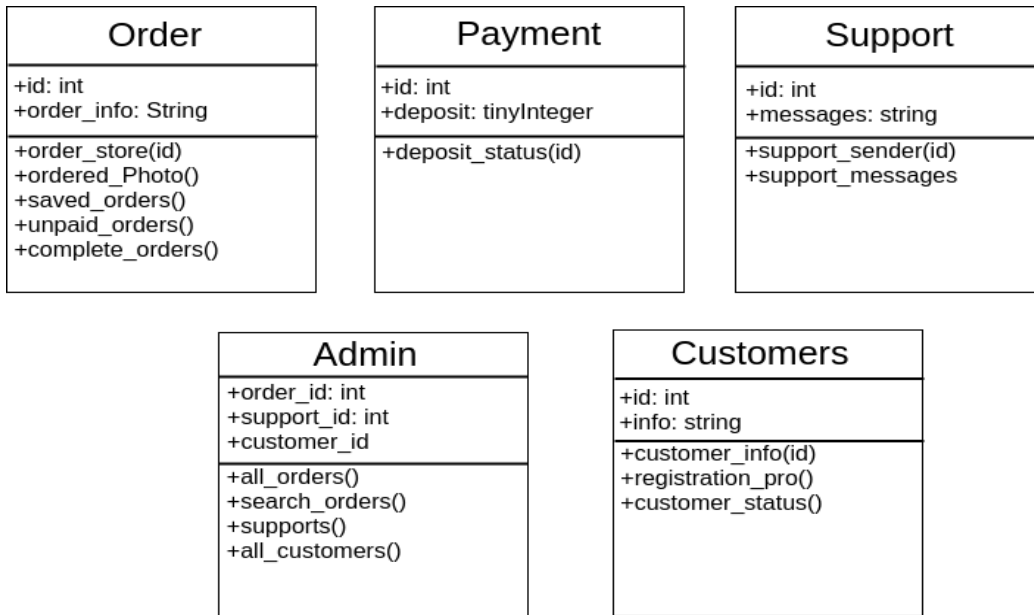


Figure 4.0: Class Diagram

## 4.3 Database Design Diagram

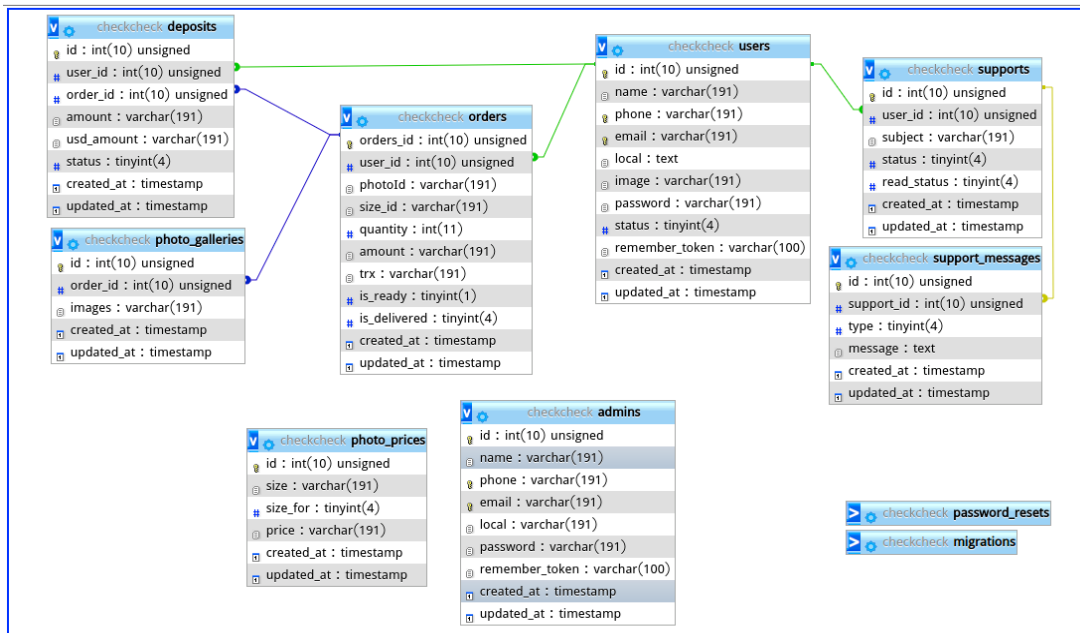


Figure 4.1: Database Diagram

# **Chapter: 05**

## **System Test**

## 5.1 Testing Features

Feature testing can be considered as making change to add or modify the new functionality to the existing project. To test the features and functionality, a new test set is to be written for testing purpose. Almost every feature and functionality have different characteristics. Those are designed to make the application more useful, intuitive, reliable, secured, scalable, effective and efficient.

### 5.1.1 Features to be tested

Features	Priority	Description
Registration	1	To become a customer they need to be registered first.
Login	1	User must be authenticated by login
Logout	1	Session must be destroyed after logout
Giving photo order by photo id	3	Order data must be inserted properly with photo id
Giving order by uploading photo	3	Photo upload must have to be uploaded properly
Saving a order	2	Order data must be inserted properly
Order history	2	Retrieve a users orders history properly
Searching a order by admin	3	Admin should found the orders data properly
Watching pending orders	3	Admin should found which orders are pending
Watching undelivered orders	2	Admin should found which orders are not delivered yet
Customer observing	1	Admin should watch customers information perfectly
<b>Technological Features</b>		
Database	1	Database will be used at almost every operation. So this is why, this part must be controlled tightly.

Here,

1 = Low Priority,

2 = Medium Priority,

3 = High Priority

## 5.2 Testing Strategy

Testing strategy is to be considered as a general purpose of testing process. Testing object, testing function methods, total available resources are included to the testing strategy. It is also the indicator of test levels that are to be performed on the whole software development life cycle. Those strategies that are prepared by the quality assurance team should be reviewed by the developers of the application. After that it should be reviewed by the test team leads.

Different kinds of testing strategies can be performed according to the type of application system that need to be tested.

## 5.2.1 Test approach

To complete the test process, testers must take some approach. There are mainly two test approach.

- **Automation testing:** Automation testing is a name of testing technique by which test engineers prepare some scripts according to test plan and after that they use suitable tools to perform testing of the software. Nowadays, almost every software company follow the approach of automation testing.
- **Manual testing:** Manual testing is also a name of technique of testing by searching out the bugs or vulnerability in an application. In this process, test engineers manually test and execute the test cases without having any automation tools.

### 5.2.1.1 Black Box Testing

Black box testing is a software testing method by which test engineers test the software without having knowledge of the internal architecture of the application that need to be tested. It is also known as behavioral testing. Black box testing can be both functional or non-functional. It ignores the internal mechanism of a system. We have decided to perform the equivalent class partitioning and Boundary value analysis techniques to implement.

### 5.2.1.2 Equivalent Class Partitioning

Equivalent class partitioning is also known as equivalent partitioning. In this technique input data is to be divided into groups. Those group of data are expected to perform as similar behavior. Each groups works as same as other groups also.

The main advantage of following the approach of equivalent class partitioning is to reduce the total number of test cases from infinite to finite. Another advantage is that it can be applied to all testing levels also.

For legal input value output will produce a meaning data. But for illegal input value, output will not produce meaningful data.

### 5.2.1.3 Boundary Value Analysis

Boundary value analysis is a name of software testing by which input data can be included within a boundary range. Following this strategy, test engineers firstly define the boundary value. After that they take a specific boundary and then write the test cases that will be go through the application for testing purpose. After completing those tasks, testers use test cases to test applications software.

### 5.2.1.4 White Box Testing

White box testing is also a name of testing approach which is also known as clear box testing, glass box testing, open box testing, transparent box testing, code based testing or structural testing. It is opposite to the black box testing. In black box testing, the internal architecture or algorithms is not known to the testers whether the whole software architecture is known to the testers while white box testing.

Testers can also predict the output of every test cases for white box testing. White box testing can be classified into some levels. Such as:

- Unit Testing
- Integration Testing
- System Testing

The main advantage of white box testing is that testing is more throughout and the testing can be started from the very beginning stage.

### 5.2.2 Pass / Fail Criteria

Pass or fail criteria will be set by the test engineers. They will prepare the pass / fail criteria on the basis of which input data are worked and which are not works well. Those data that are worked well will considered as pass criteria. And rest of input data will be considered as fail criteria.

Now I will give the pass / fail criteria below.

- System crash will not be considered as pass case.
- If any criteria pass 100% times, then it will be considered as pass criteria only.
- If data can't be displayed to the application properly, then it is also to be considered as fail criteria.

## 5.3 Testing Schedule

Test Phase	Time
Testing plan create	1 week
Test specification	2 week
Unit testing	During development time
Component test	1 week
Test Phase	Time
Integration testing	1 week
Validating use cases	1 week
Testing user interfaces	1 week
Load testing	1 week

Performance testing	1 week
Release to production	1 week

## 5.4 Testing Environment

Testing environment means to prepare the environment with hardware and software so that test engineers can be able to execute test cases as required. Besides hardware and software usage, network configuration might be needed to execute test plans.

For making the environment for testing, some key area need to set up. Those are:

- Test data
- Database server
- Client's operating system
- Front end running environment
- Browser
- System and application
- Network
- Hardware with server operating system
- Documentation is also required. Like: user manuals, installation guides, configuration guides, documents etc.

## 5.6 Test Cases

A test case refers to some rules and regulations or conditions by which it can be determined whether a system can be able to meet the works or requirements under test cases properly. We know, there is a chance to have some fault or break holes in any application. This is very common scenario. And those issues are solved usually by software testing approaches. But if we don't care of those issues, then the full application development might be ruined. So proper testing must be needed. For testing my application, I have prepared some test cases. Now, I am going to provide them.

### 5.5.1 Log In

<b>Test case #1</b>		<b>Test case name:</b> Log in			
<b>System:</b> Automatic order place...		<b>Subsystem:</b> N/A			
<b>Designed by:</b> Ahmed Musa		<b>Designed date:</b> 10-April-2019			
<b>Executed by:</b>		<b>Executed date:</b>			
<b>Short description:</b> The registered users need to login to the application, and before that my application will check the authentication and authorization.					
<b>Pre-conditions:</b>					
<ul style="list-style-type: none"> <li>• Users are always redirected to the login page whether they have not authenticated by application while entering into the landingpage.</li> <li>• Assume that, the Email is 'ahmusa@live.com' and password is 'bangladesh'</li> </ul>					
Step	Email	Password	Expected result	Pass/Fail	Comment
1	Ahmu	dsgs	Invalid		
2	ahmusa@live.com		Password can't be blank		
3		Bangla	Email can't be blank		
4	ahmusa@live.com	bangladesh	Successfully login to the application and redirected to the landingpage		
<b>Post-conditions:</b> Users including Customer and Admin will successfully login to the application.					

### 5.5.2 Input field is required

<b>Test case #2</b>		<b>Test case name:</b> Input field is required in all pages			
<b>System:</b> Automatic order place and ....		<b>Subsystem:</b> N/A			
<b>Designed by:</b> Ahmed Musa		<b>Designed date:</b> 10-April-2019			
<b>Executed by:</b>		<b>Executed date:</b>			
<b>Short description:</b> Users customers and admin need to fill up all that fields which as marked as required.					
<b>Pre-conditions:</b>					
<ul style="list-style-type: none"> <li>• Users need to login first to be authenticated by the application program.</li> <li>• Every required field must be filled up by that user.</li> </ul>					
Step	Action	Response	Pass/Fail	Comment	
1	All required fields are not filled yet.	Fields must not be empty.			



2	All input field is filled up by the user.	Application will save those information.		
<b>Post-conditions:</b> Every information is saved to the database of the application program properly.				

### 5.5.3 Making an order

<b>Test case #3</b>		<b>Test case name: Making a order</b>		
<b>System:</b> Automatic order place and ...		<b>Subsystem:</b> N/A		
<b>Designed by:</b> Ahmed Musa		<b>Designed date:</b> 10-April-2019		
<b>Executed by:</b>		<b>Executed date:</b>		
<b>Short description:</b> A registered customer will give a order by a photo id which is provided by studio.				
<b>Pre-conditions:</b>				
<ul style="list-style-type: none"> <li>● User must be authenticated by application.</li> <li>● User type must be Customer.</li> <li>● User must have a photo Id</li> </ul>				
<b>Step</b>	<b>Action</b>	<b>Response</b>	<b>Pass/Fail</b>	<b>Comment</b>
1	All data is not provided	Application push customer to provide all required data to the application.		
2	Data provided properly	Data saved successfully for completing a order		
<b>Post-conditions:</b> After providing order data to the application, it will be pending for develop.				

# **Chapter: 06**

## **User Manual**

## 6.1 Website

First of all a user will get a website to explore studios information and services then here they will get a customer panel named “Login As Customer”

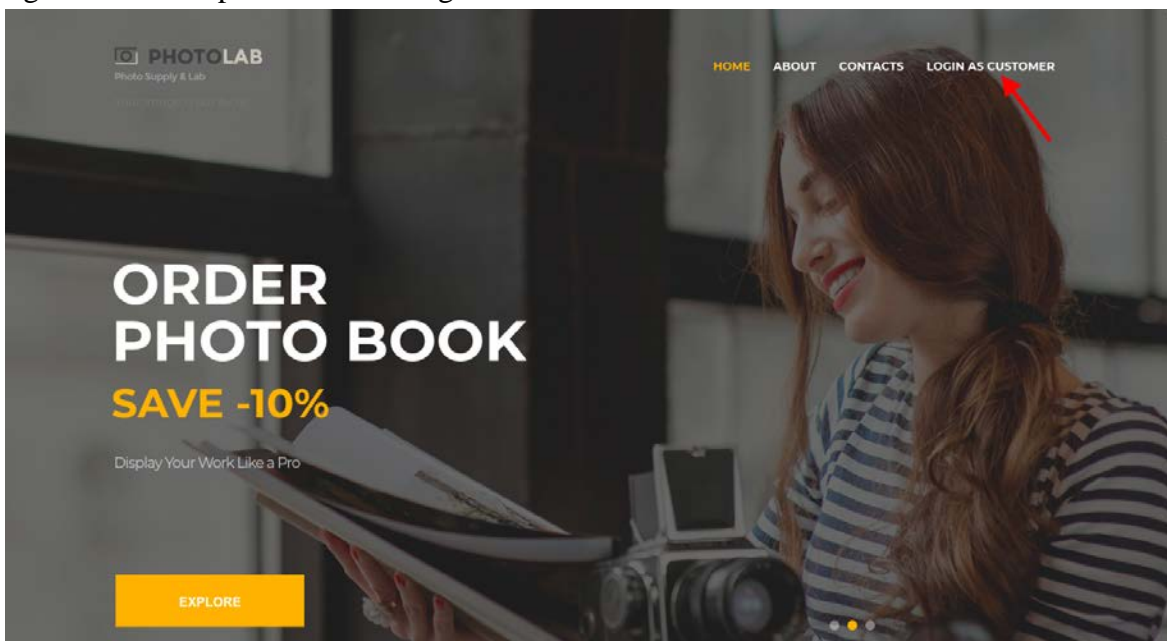


Figure 6.1: Website

## 6.2 Customer login page

Customers need to login in the application to get service from the studio.

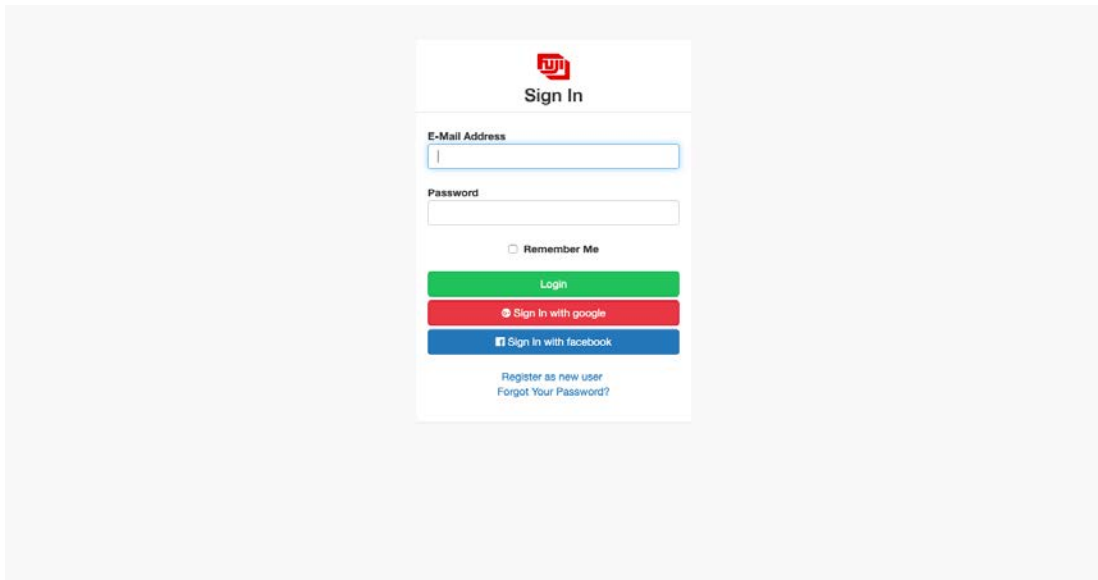


Figure 6.2: Customer login page

### 6.3 Customers landing page

Customer will be able see this page when they authorised by the system authentication

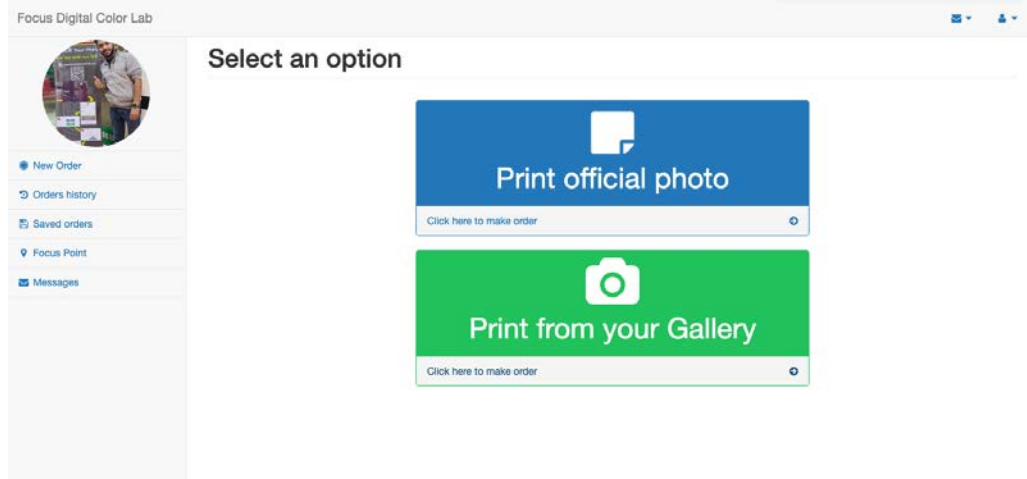


Figure 6.3: Customers landing page

### 6.4 Making order by photo ID

Customers will make order clicking by “Print official photo” by their photo id, which is provided by photo studio.

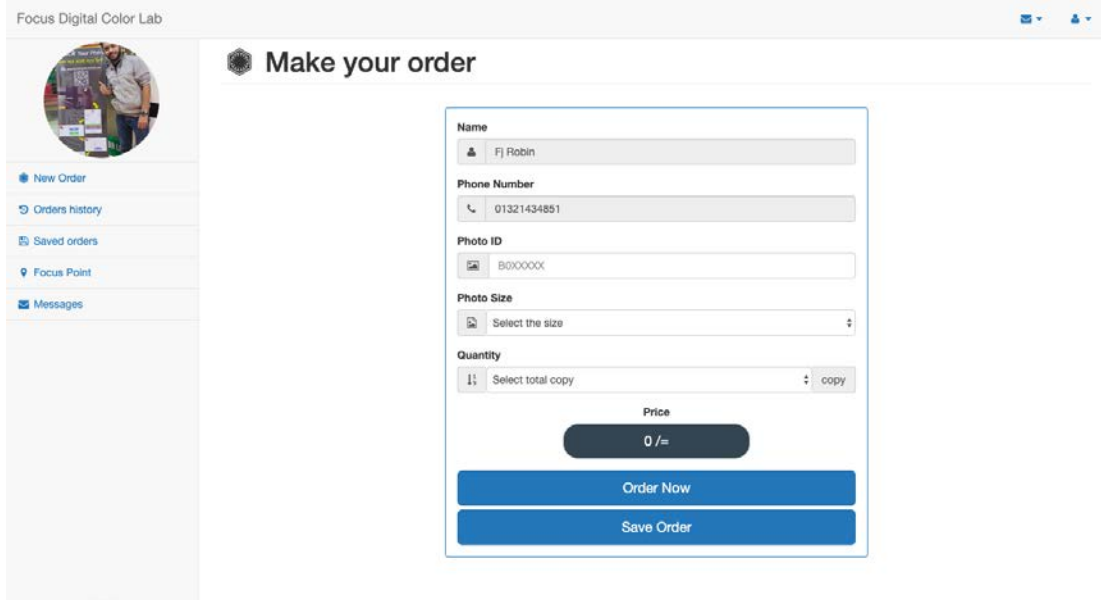


Figure 6.4: Making new order by photo id

## 6.5 Making new order by uploading photo

Customers will make order clicking by “Print from gallery” by uploading their photo, Clicking “+” icon customers will be able to upload maximum ten photos.

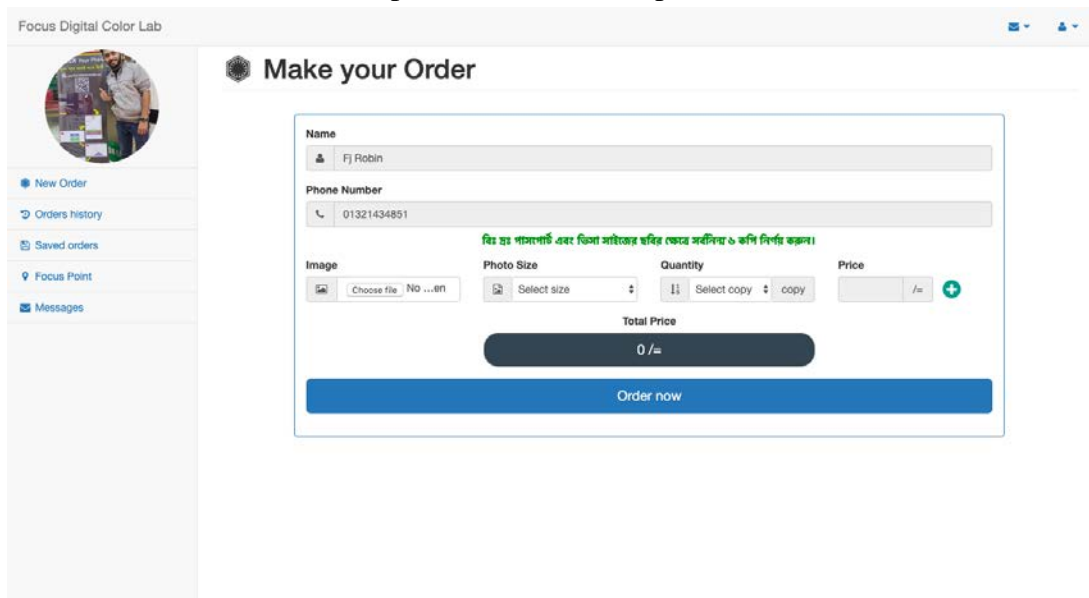


Figure 6.5: Making new order by uploading photo

## 6.6 Order confirmation

Customers will be able to confirm his/her orders before payment.

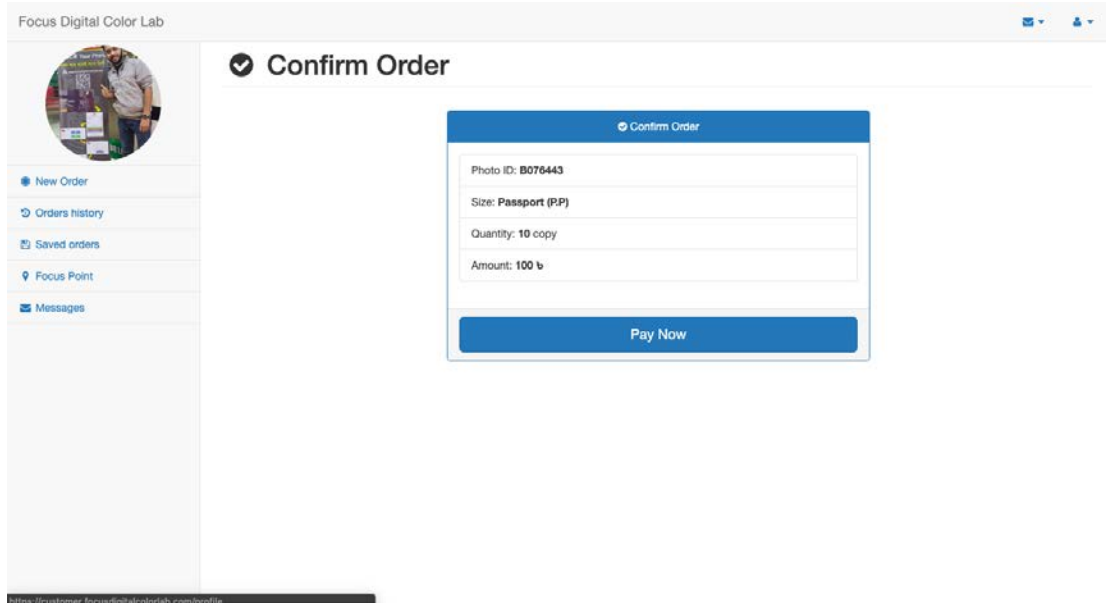


Figure 6.6: Confirm order

## 6.7 Selecting payment gateway

After confirmation by clicking one of them, customer will be able to pay by that payment method

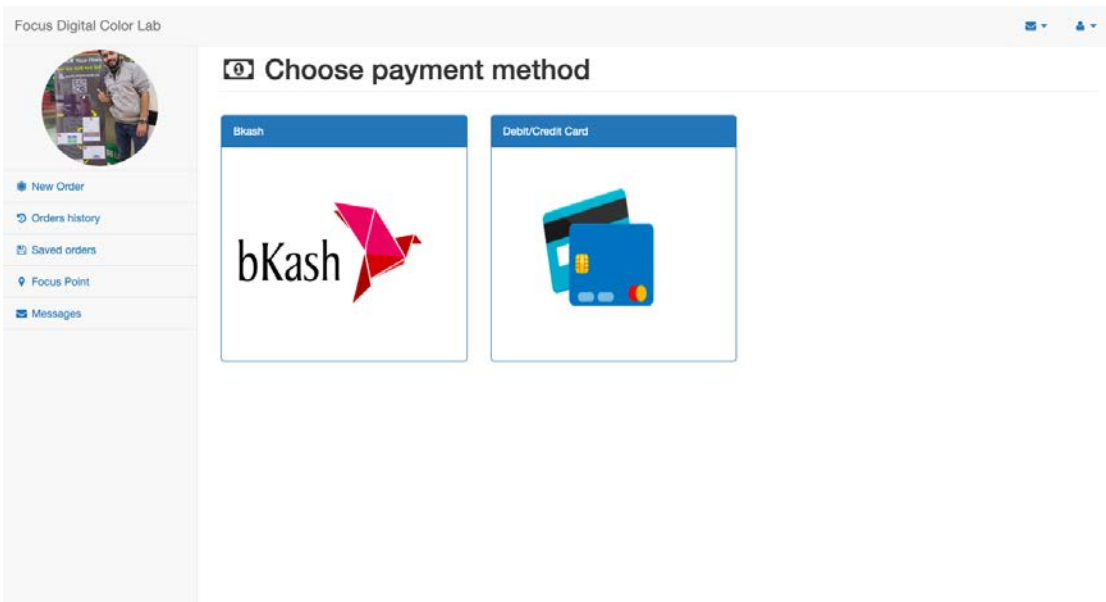


Figure 6.7: Selecting payment method

## 6.8 Paying by bKash

After selecting bkash as payment method, customer will be able to complete their order by putting transaction id here

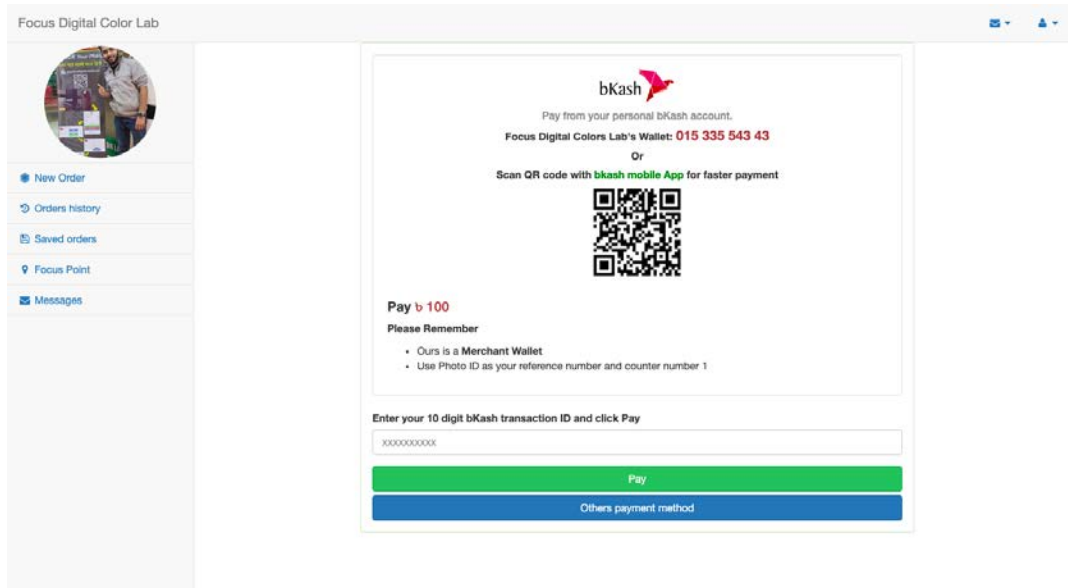


Figure 6.8: Paying by bKash

## 6.9 Paying by Card

After selecting card as payment method, customer will be able to complete their order by putting cards information here

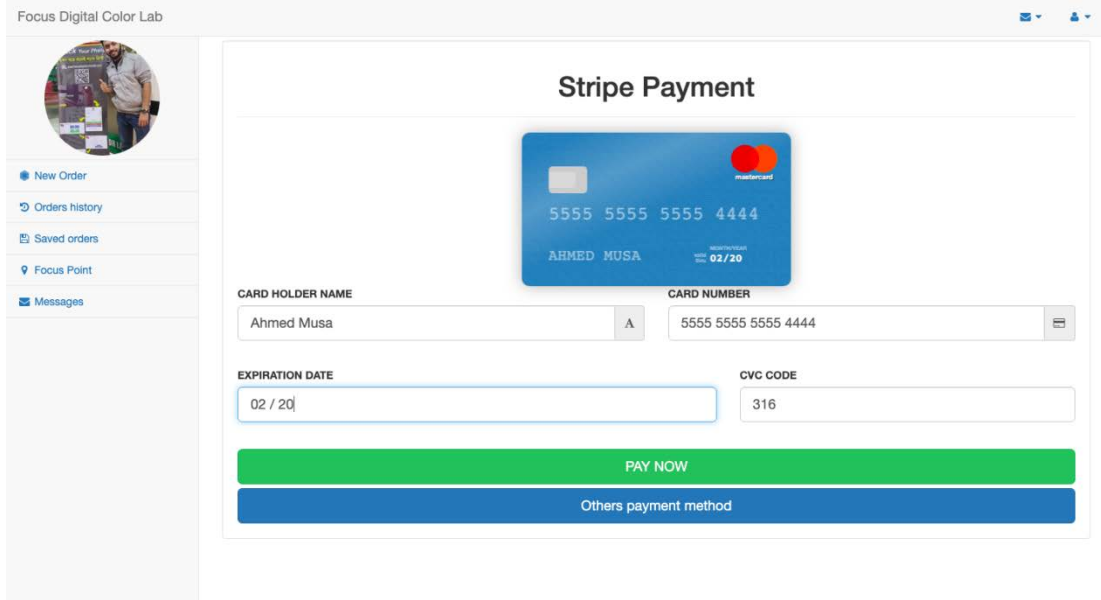


Figure 6.9: Paying by card

## 6.10 Customer orders history

By clicking “Orders history” from side menu customer will be able to see their order history and orders status as well

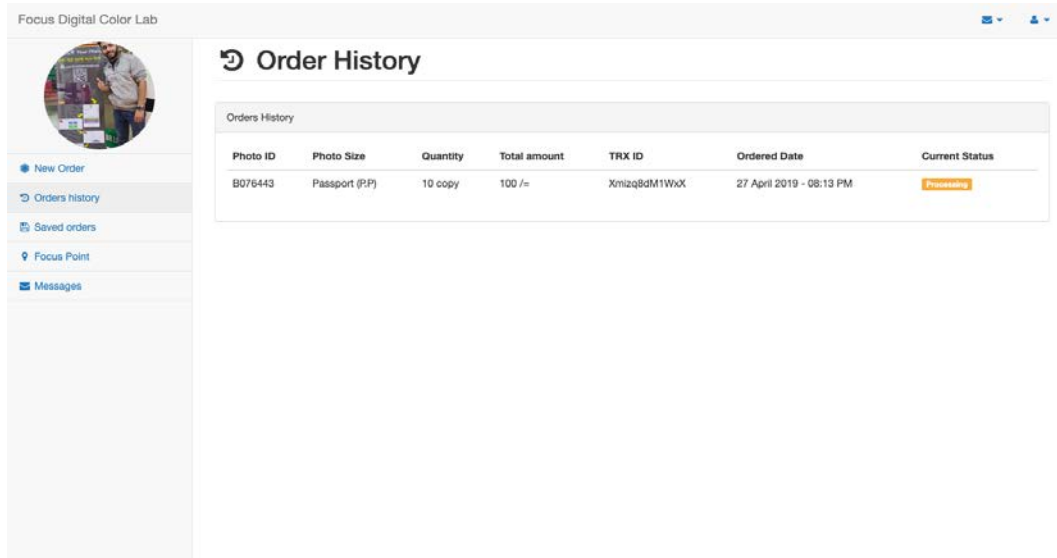


Figure 6.10: Orders history and status



## 6.11 Customer saved order

By clicking “Saved orders” from side menu customer will be able to see their saved orders and able to pay from here or can delete that order

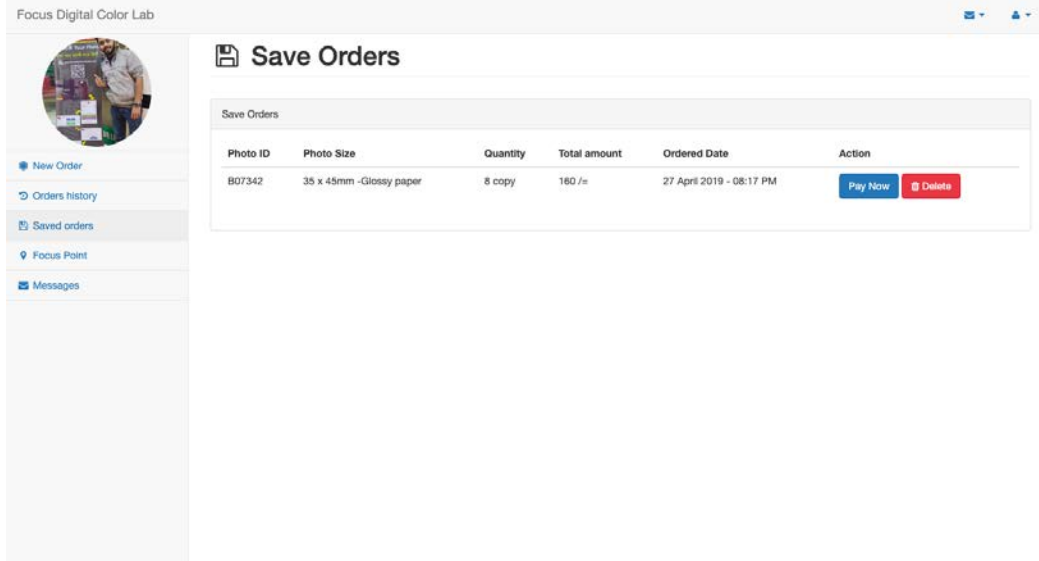


Figure 6.11: Saved order

## 6.12 Focus point/Studio location

By clicking “Focus point” from side menu customer will be able to see the studio location over the google map

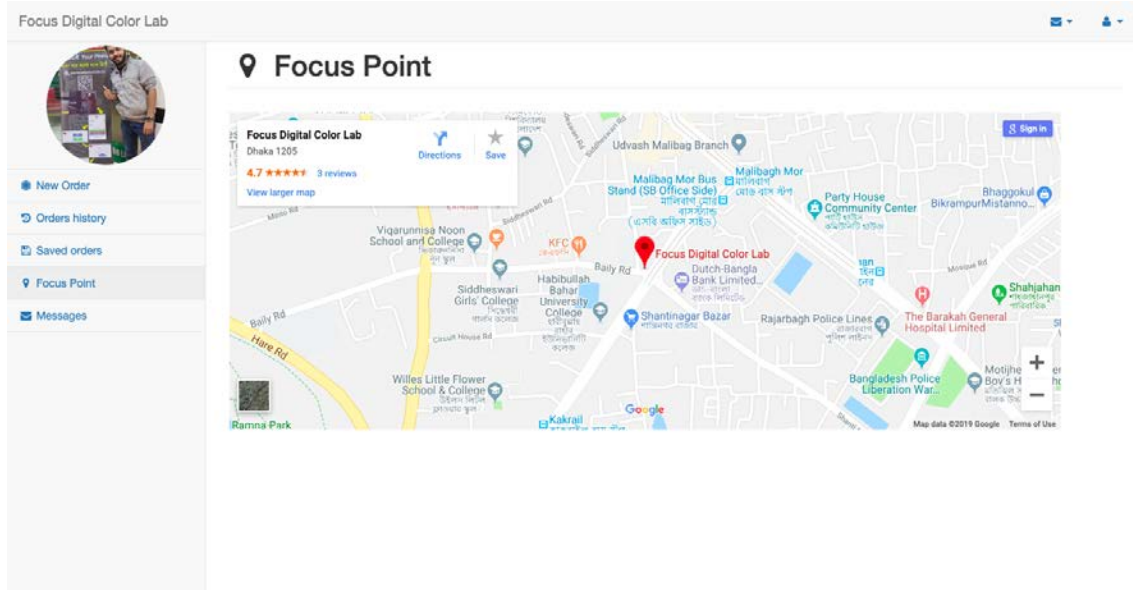
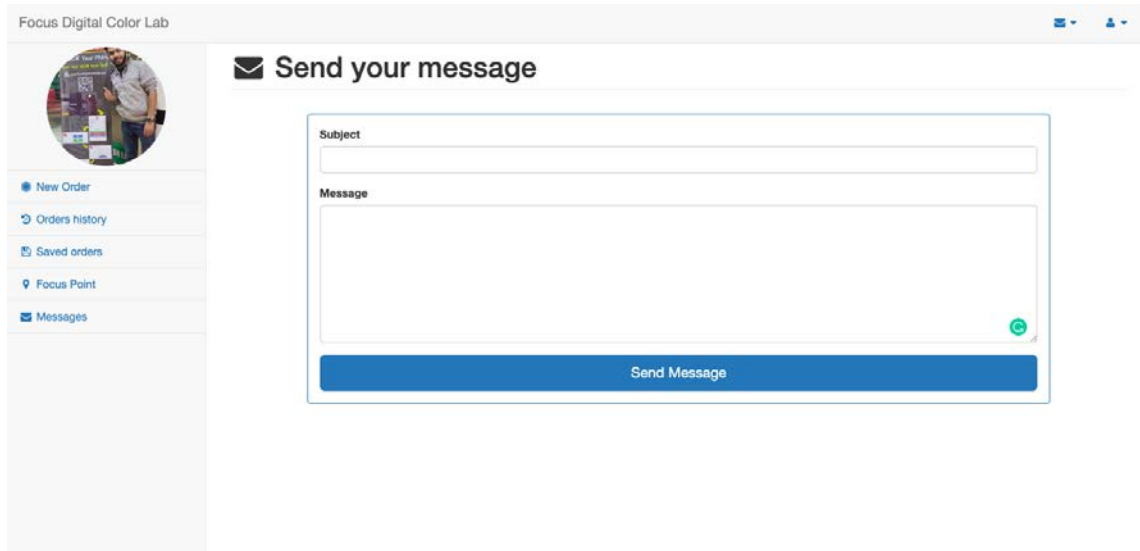


Figure 6.12: Focus Point

## 6.13 Sending message to admin

Here customer will be able to send any message/complain/feedback to admin, just have to give a message subject then full content

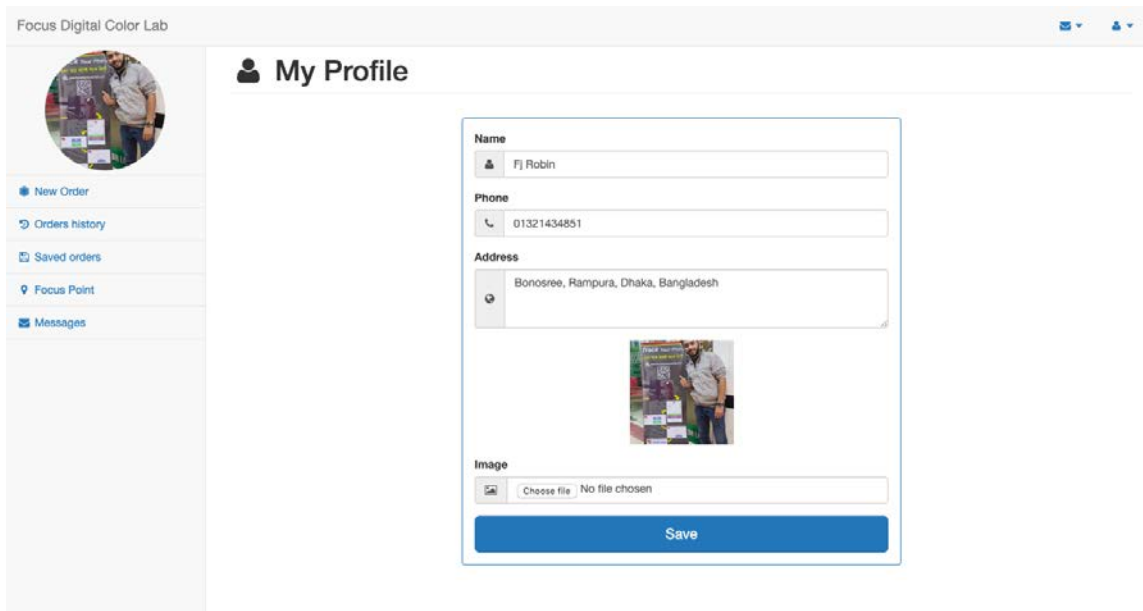


The screenshot shows the 'Send your message' form in the Focus Digital Color Lab user interface. The form is titled 'Send your message' and is located on the right side of the page. It features a 'Subject' input field and a larger 'Message' text area. A green circular icon with a plus sign is visible in the bottom right corner of the message area. Below the message area is a blue 'Send Message' button. On the left side of the page, there is a sidebar menu with a profile picture and several navigation options: 'New Order', 'Orders history', 'Saved orders', 'Focus Point', and 'Messages'. The top of the page displays the text 'Focus Digital Color Lab' and some utility icons.

Figure 6.13: Sending message to admin

## 6.14 Customers profile

By clicking profile photo from side menu customer will be able to edit his/her profile photo, phone number, address



The screenshot shows the 'My Profile' form in the Focus Digital Color Lab user interface. The form is titled 'My Profile' and is located on the right side of the page. It features several input fields: 'Name' (with the value 'FJ Robin'), 'Phone' (with the value '01321434851'), and 'Address' (with the value 'Bonosree, Rampura, Dhaka, Bangladesh'). Below the address field is a profile picture of a man in a white shirt. At the bottom of the form is an 'Image' section with a 'Choose file' button and the text 'No file chosen'. A blue 'Save' button is located at the bottom of the form. On the left side of the page, there is a sidebar menu with a profile picture and several navigation options: 'New Order', 'Orders history', 'Saved orders', 'Focus Point', and 'Messages'. The top of the page displays the text 'Focus Digital Color Lab' and some utility icons.

Figure 6.14: Customers profile

## 6.15 Customers logout

Here, customers will be able to logout from system by clicking “Logout” from top menu

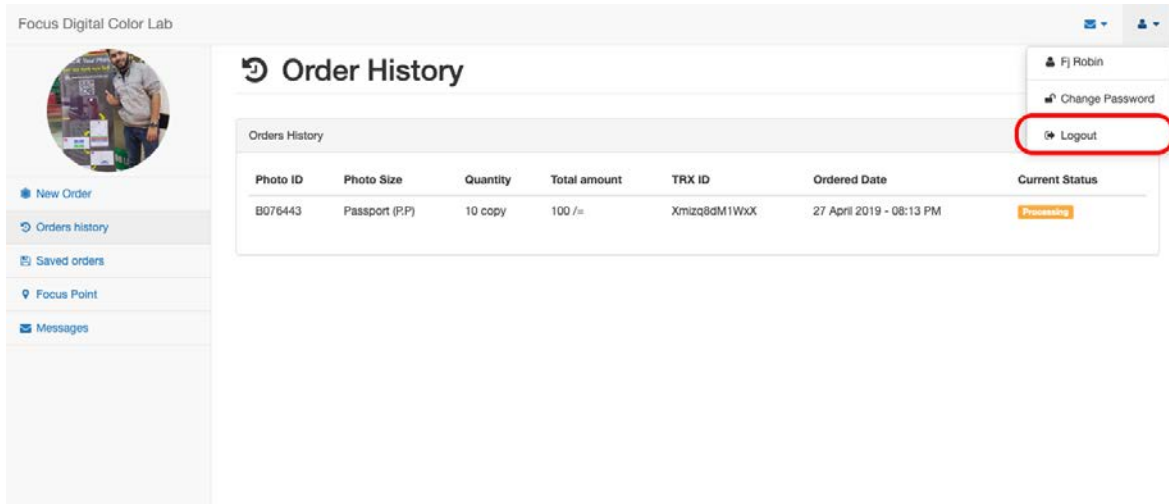


Figure 6.15: Customers logout

## 6.16 Admin login

Admin have to login into system to maintain orders, website/admin by this url admin will get login page

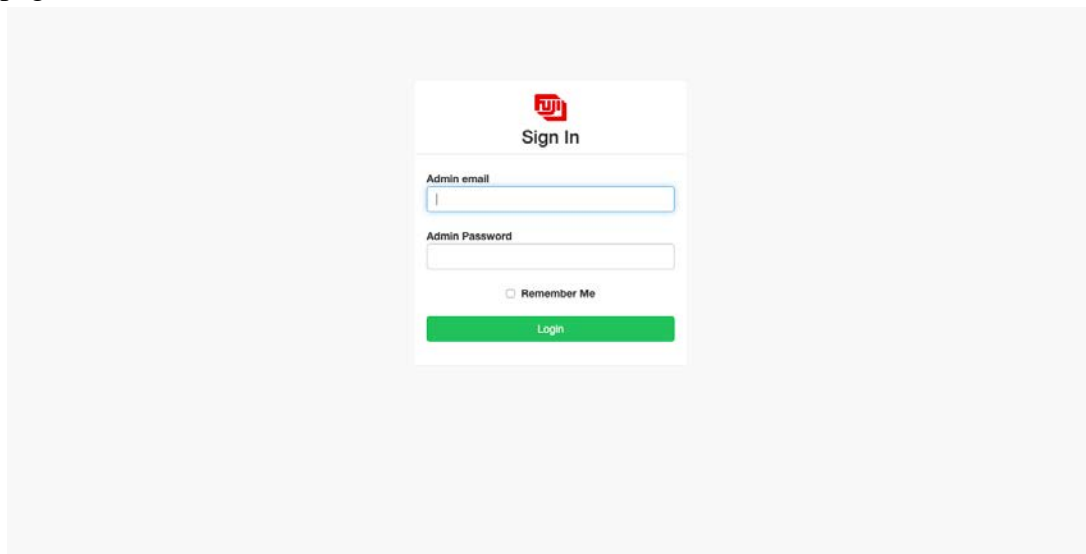


Figure 6.16: Admin login page

## 6.17 Admins Dashboard

When admin successfully login into system he/she will be able see a complete view of full maintaining system

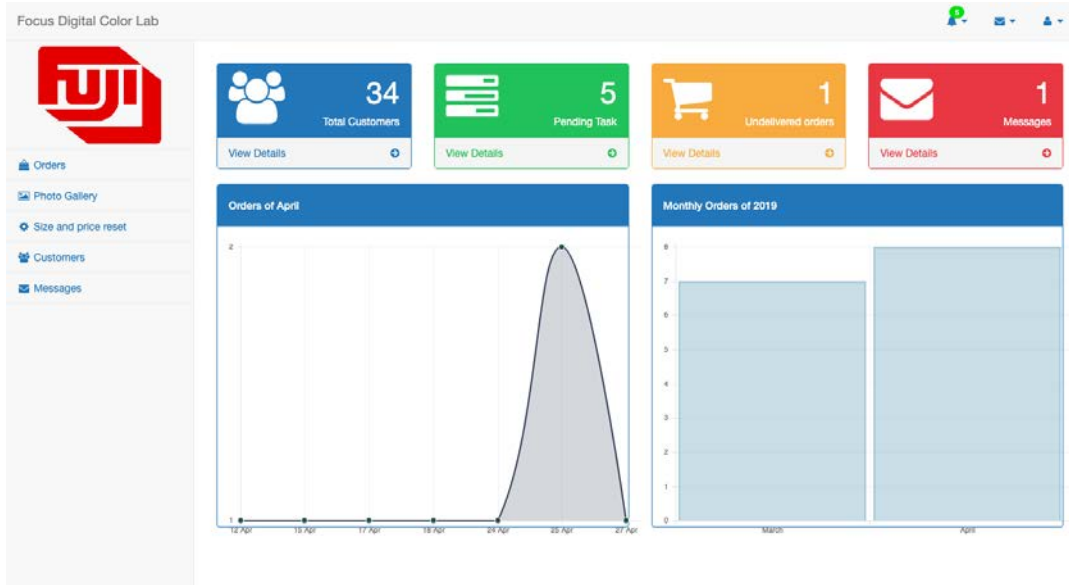


Figure 6.17: Admins dashboard

## 6.18 All orders

By clicking “Orders” from side menu bar admin will be able see all orders condition

Name	Phone	Photo ID	Photo Size	Quantity	Total amount	TRX ID	Ordered	Status
Fj Robin	01321434851	B076443	Passport (PP)	10 copy	100 /=	Xmizq8dM1WwX	27 April 2019 - 08:13 PM	Mark as ready
Robin 007	01303256410	88iUNao	Multiple	Multiple	160 /=	Cash Payment	25 April 2019 - 07:12 PM	Delivered
Robin 007	01303256410	B011115	35 x 45mm -Matte paper	18 copy	360 /=	Cash Payment	25 April 2019 - 07:10 PM	Delivered
Robin 007	01303256410	6WUXdN	Multiple	Multiple	60 /=	Cash Payment	24 April 2019 - 10:22 PM	Mark as delivered
Robin 007	01303256410	B098765	35 x 45mm -Glossy paper	8 copy	160 /=	ZDfReqqB83DG	18 April 2019 - 06:12 PM	Delivered
Atik Faysal	01835897172	B0999777	40 x 50mm -Matte paper	17 copy	340 /=	Cash Payment	17 April 2019 - 07:31 PM	Delivered
Atik Faysal	01835897172	B0999777	Passport & Stamp	17 copy	170 /=	ySbq0A4KniSz	15 April 2019 - 08:13 PM	Mark as delivered
Sazal Islam	01744542955	B028300	Passport & Stamp	10 copy	100 /=	6DC34GGQUB	12 April 2019 - 06:21 PM	Mark as delivered
Ashiqur Pavel		B65Rj7g	Multiple	Multiple	120 /=	Cash Payment	12 March 2019 - 01:27 AM	Delivered
Robin 007	01303256410	B074645	35 x 50mm -Glossy paper	16 copy	160 /=	Cash Payment	09 March 2019 - 03:51 AM	Delivered

Figure 6.18: All orders

## 6.19 Photo gallery

By clicking “Photo Gallery” from side menu bar admin will be able see all orders which is non-stored photo/Uploaded photo

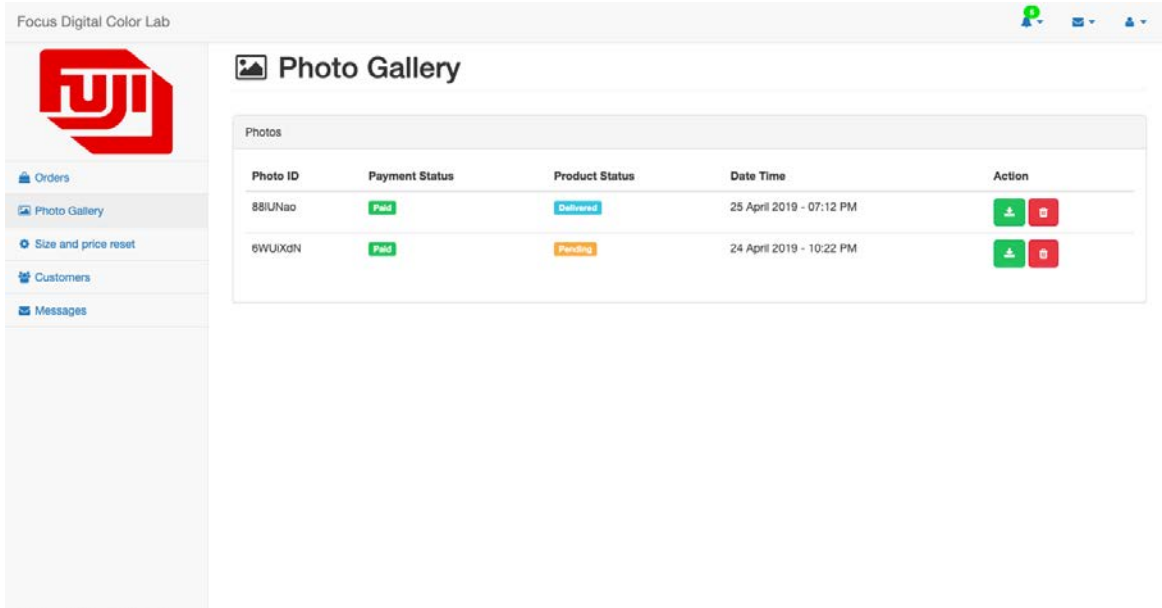
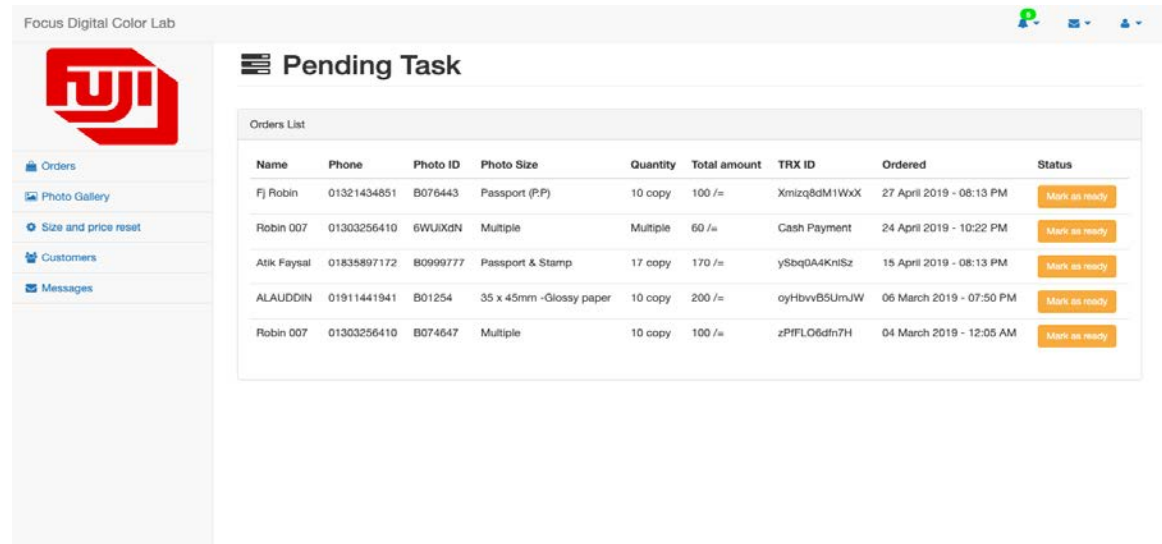


Photo ID	Payment Status	Product Status	Date Time	Action
88IUxNao	Paid	Delivered	25 April 2019 - 07:12 PM	<a href="#">Download</a> <a href="#">Delete</a>
6WUjXdN	Paid	Pending	24 April 2019 - 10:22 PM	<a href="#">Download</a> <a href="#">Delete</a>

Figure 6.19: Photo gallery

## 6.20 Pending task

By clicking “Pending Task” from dashboard card admin will be able see all orders which is incomplete or undeveloped orders



Name	Phone	Photo ID	Photo Size	Quantity	Total amount	TRX ID	Ordered	Status
FJ Robin	01321434851	B076443	Passport (P/P)	10 copy	100 /=-	Xmizq8dM1WxX	27 April 2019 - 08:13 PM	<a href="#">Mark as ready</a>
Robin 007	01303256410	6WUjXdN	Multiple	Multiple	60 /=-	Cash Payment	24 April 2019 - 10:22 PM	<a href="#">Mark as ready</a>
Atik Faysal	01835897172	B0999777	Passport & Stamp	17 copy	170 /=-	yStqQA4KntSz	15 April 2019 - 08:13 PM	<a href="#">Mark as ready</a>
ALAUDDIN	01911441941	B01254	35 x 45mm -Glossy paper	10 copy	200 /=-	oyHbvVBSUmJW	06 March 2019 - 07:50 PM	<a href="#">Mark as ready</a>
Robin 007	01303256410	B074647	Multiple	10 copy	100 /=-	zPFfLO6dfn7H	04 March 2019 - 12:05 AM	<a href="#">Mark as ready</a>

Figure 6.20: Pending task

## 6.21 Undelivered orders

By clicking “Undelivered orders” from dashboard card admin will be able see all orders which is completed but customer didn’t collect it yet

Name	Phone	Photo ID	Photo Size	Quantity	Total amount	TRX ID	Ordered	Status
Sazal Islam	01744542955	B028300	Passport & Stamp	10 copy	100 /=-	6DC34GGQUB	12 April 2019 - 06:21 PM	<a href="#">Mark as delivered</a>

Figure 6.21: Undelivered Orders

## 6.22 Photo size and price settings

By clicking “Size and price reset” from side menu bar admin will be able reset different type of sizes and prices for them

size	price	Size and Price for	Actions
Passport & Stamp	10 /=-	For Official Photo	<a href="#">Edit</a> <a href="#">Delete</a>
Passport & Stamp	10 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
Passport (P.P)	10 /=-	For Official Photo	<a href="#">Edit</a> <a href="#">Delete</a>
Passport (P.P)	10 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
Others	100 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
B2 (3.5" x 2.5")	10 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
8R (8" x 10")	150 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
8L (8" x 12")	180 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
6R (6" x 8")	70 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>
6L (6" x 9")	80 /=-	For Gallery Photo	<a href="#">Edit</a> <a href="#">Delete</a>

Figure 6.22: Photo size and price settings

## 6.23 Customers observing

By clicking “Customers” from side menu bar admin will be able see all of users and can active/deactivated them

Name	Phone	Email	Local Address	Registered by	Action
Farzeen Faiz	01842140768	farzeen@gmail.com	banasree, rampura		Deactive
Mutasim Billah		mutasim.ewu@gmail.com	Banasree, Dhaka		Deactive
Faiz Ullah	01713140765	faiz_ullah4@yahoo.com	Dhaka bangladesh		Deactive
Robin 007	01303256410	fjrobin007@gmail.com	Dhaka	google	Deactive
Fj Robin	01521434851	fjrobinprof@gmail.com	Bonosree rampura	google	Deactive
mutasim billah	01719469346	mullobodh@gmail.com	Uttara	google	Deactive
ah musa	01817140765	ahmusa16@gmail.com	Dhaka, Bangladesh	google	Deactive
Toisfor Rahman Eem	01872332485	eemxpress@gmail.com	Rampura	google	Deactive
MD Rasei		rasetoolboy1998@gmail.com		facebook	Deactive
Abdullah Mamun		ahmed_mamun99@yahoo.co.uk		facebook	Deactive

Figure 6.23: Customer observing

## 6.24 Messages

By clicking “Messages” from side menu bar admin will be able see all messages from customers

User Name	Email	Phone	Subject	Status	Date Time	Action
fasdfasd	t514504@nwytg.net	01541214447	Error For Order Please Solved this issue	Staff replied	09 April 2019 - 03:38 PM	View
Krishanee RS	krishanee.rs@gmail.com		Pic	Staff replied	07 March 2019 - 05:28 AM	View

Figure 6.24: Messages from customers

# **Chapter: 07**

## **Conclusion**



## 7.1 GitHub Link

<https://github.com/FjRobin/pFDLC>

## 7.2 Project Summary

I have started to develop this project from August. From the very starting of development of this project, it demands hard working, patients, persistency to meet the requirements of stakeholders. After that I have proposed the design. And then started to work.

Database plays a vital role for any application software. And so this is why, I have designed the database diagram having tables with proper relationship. After that, I have prepared the user interface and take their approval to continue to the next part. It is to be said that, the interface of my application is very simple and easy to understand. After completing that, I have started to write the core functionality of the project.

Developing project is not end of all tasks actually. There are some other important tasks to perform. And that is testing. It is also known as quality assurance also. Almost at every software company there are a quality assurance team. Their main responsibility is to find the loopholes or vulnerability of software. If there any bug remains before handover to the stakeholders, there is a change to ruin the whole project. So testing plan is very important. And after developing the project, I have assured the quality of this project.

## 7.3 Limitations

For developing this project, I have faced some limitations. Now I will describe those in brief.

- **Limited photo uploading:** In my application there is a option where customers can upload their photo to print but that is have a limitation that customer will be able to upload only less than ten photos
- **Execution time:** So that this system will upload photos the system to more time than regular execution time.

## 7.4 Obstacles and Achievements

I believe that if there are not any obstacles to develop a project, then there doesn't have any challenges. Because we know, challenge give us the opportunity to prove ourselves. Obstacles, challenges and achievements are like a path to the success.

Before starting this project, I didn't know the actual flow of software development life cycle. By developing this project, I have learnt to know how to have a row requirement from clients. After that I have learnt system analysis, database design and many things. My supervisor helps me a lot from the very beginning of the development of this project.

## 7.5 References

I have gained some knowledge from some platforms. Obviously I will mention those references. For making my project successful those resources help me a lot. Not I will mention the names below.

- [www.google.com](http://www.google.com)
- [www.youtube.com](http://www.youtube.com)
- [www.github.com](http://www.github.com)
- [www.wikipedia.com](http://www.wikipedia.com)
- [www.php.net](http://www.php.net)
- [www.mysql.com](http://www.mysql.com)
- [www.mysql.com/products/workbench](http://www.mysql.com/products/workbench)
- [www.themeforest.net](http://www.themeforest.net)
- [www.getbootstrap.com](http://www.getbootstrap.com)
- [www.w3schools.com](http://www.w3schools.com)
- [www.jquery.com](http://www.jquery.com)
- [www.sublimetext.com/3](http://www.sublimetext.com/3)
- [www.apachefriends.org](http://www.apachefriends.org)
- [www.stakeoverflow.com](http://www.stakeoverflow.com)
- <https://www.jetbrains.com/>