



## **Project Name: It Agency**

**By**

Name: Md:Bokhtiar Alam

Id:171-35-172

## **Supervised by**

Md. Shariful Islam

Lecturer

Department Of Software Engineering

Daffodil International University

A Project Submitted in Partial Fulfillment of the Requirement for the Degree of  
Bachelor of Science in Software Engineering.

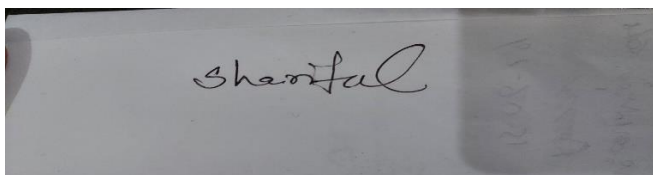
Spring-2021

DEPARTMENT OF SOFTWARE ENGINEERING

DAFFODIL INTERNATIONAL UNIVERSITY

## Approval

This project titled on “It Agency”, submitted by MD. Bokhtiar Alam, ID: 171-35-172 to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

A photograph of a handwritten signature in black ink on a light-colored surface. The signature is written in a cursive style and reads "Shariful".

---

**Mr. Shariful Islam**

**Lecturer (senior scale)**

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

**Internal  
Examiner 1**

---

**Mr. SK. Fazlee Rabby**

**Lecturer**

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

**Internal  
Examiner 2**

---

**Nayeem Hasan**

**Lecturer (senior scale)**

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

**Internal  
Examiner 3**

## DECLARATION

It hereby declares that this project has been done by me under the supervision of Mr. Shariful Islam, Lecturer (senior Scale), Department of Software Engineering, Daffodil International University. It also declares that neither this project nor any part of this has been submitted anywhere else for award of any degree.

Bokhtiar

---

Name: MD. Bokhtiar Alam,  
ID: 171-35-172

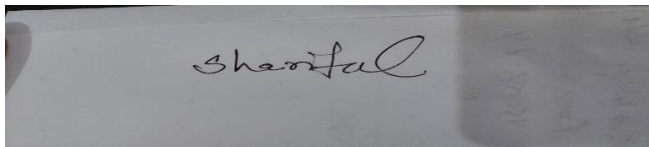
Batch: 22nd

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

Supervised by:

A rectangular box containing a handwritten signature in black ink. The signature is written in a cursive style and appears to read "Shariful".

---

MR. Shariful Islam

Lecturer (senior Scale)

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

## **ACKNOWLEDGEMENT**

I have undertaken this project. Be that as it may, it would not have been conceivable without the kind help of numerous people. I might want to stretch out my earnestness because of every one of them. I am exceptionally obligated to Daffodil International University for their direction and steady supervision by Mr. Shariful Islam and co-supervisor Md. Rajib Mia and in addition for giving necessary information with respect to the venture and additionally for their help in finishing the task. I would like to express my gratitude towards our parents, our batchmate, and members of DIU for their kind cooperation and consolation which helped us in finishing this task. My thanks and thanks likewise go to my supervisor and co-supervisor in building up the energetically bailed us out with their capacities.

## Appendix A

<b>APPROVAL</b>	i
<b>INTERNSHIP DECLARATION</b>	ii
<b>ACKNOWLEDGEMENT</b>	iii
<b>EXECUTIVE SUMMARY</b>	iv
<b>LIST OF TABLES</b>	vii
<b>LIST OF FIGURES</b>	viii

### **CHAPTER 1: INTRODUCTION** 1

- 1.1 Project Overview
- 1.2 Project Purpose
  - 1.2.1 Background
  - 1.2.2 Benefits & Beneficiaries
  - 1.2.3 Goals
- 1.3 Stakeholders
- 1.4 Proposed System Model (block diagram)
- 1.5 Project Schedule
  - 1.5.1 Gantt Chart
  - 1.5.2 Release Plan/Milestone

### **CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION**

- 2.1 Functional Requirements
- 2.2 Data Requirements
- 2.3 Performance Requirements
  - 2.3.1 Speed and Latency Requirements
  - 2.3.2 Precision or Accuracy Requirements
  - 2.3.3 Capacity Requirements
- 2.4 Dependability Requirements
  - 2.4.1 Reliability Requirements
  - 2.4.2 Availability Requirements
  - 2.4.3 Robustness or Fault-Tolerance Requirements
  - 2.4.4 Safety-Critical Requirements

## 2.5 Maintainability and Supportability Requirements

2.5.1 Maintenance Requirements

2.5.2 Supportability Requirements

2.5.3 Adaptability Requirements

2.5.4 Scalability or Extensibility Requirements

## 2.6 Security Requirements

2.6.1 Access Requirements

2.6.2 Integrity Requirements

2.6.3 Privacy Requirements

## 2.7 Usability and Human-Interaction Requirements

2.7.1 Ease of Use Requirements

2.7.2 Personalization and Internationalization Requirements

2.7.3 Understandability and Politeness Requirements

2.7.4 Accessibility Requirements

2.7.5 User Documentation Requirements

2.7.6 Training Requirements

## 2.8 Look and Feel Requirements

2.8.1 Appearance Requirements

2.8.2 Style Requirements

## 2.9 Operational and Environmental Requirements

2.9.1 Expected Physical Environment

2.9.2 Requirements for Interfacing with Adjacent Systems

2.9.3 Projectization Requirements

2.9.4 Release Requirements

## 2.10. Legal Requirements

2.10.1 Compliance Requirements

2.10.2 Standards Requirements

## **CHAPTER 3: SYSTEM ANALYSIS**

3.1 Use Case Diagram

3.2 Use Case Description (for each use case)

3.3 Activity Diagram (for each use case)

3.4 System Sequence Diagram (for each use case)

## **Chapter 4: System Design Specification**

4.1 Class Responsibilities Collaboration (CRC) Cards

4.2 Sequence Diagram (for each use case)

4.3 Class Diagram

4.4 Database Design Diagram

## 4.5 Development Tools & Technology

### 4.5.1 User Interface Technology

4.5.1.1 ASP.NET MVC4 Framework or Laravel Framework

4.5.1.2 jQuery UI or Angular JS or Other jQuery Framework

4.5.1.3 CSS Framework or Twitter Bootstrap

4.5.1.4 Font Awesome or Others

### 4.5.2 Implementation Tools & Platforms

4.5.2.1 Microsoft Visual Studio 2010 or PHP Storm

4.5.2.2 MSSQL Server 2008 or MySQL Community Server 5.5

4.5.2.3 .NET Runtime or Apache HTTP Server

## **CHAPTER 5: SYSTEM TESTING**

### 5.1 Testing Features

5.1.1 Features to be tested

5.1.2 Features not to be tested

### 5.2 Testing Strategies

5.2.1 Test Approach

5.2.2 Pass/Fail Criteria

5.2.3 Suspension and Resumption

5.2.4 Testing Schedule

5.2.5 Traceability Matrix

### 5.3 Testing Environment (hardware/software requirements)

### 5.4 Test Cases

## **CHAPTER 6: USER MANUAL**

6.1 User Manual (type A user)

6.2 User Manual (type B user)

6.3 User  
Manual

6.4 Type  
C user

## **CHAPTER 7: PROJECT SUMMARY**

7.1 GitHub Link

7.2 Critical Evolution

7.3 Limitations

7.4 Obstacles & Achievements

7.5 Future Scope

# List of Tables

<b>Table01: Maintenance requirement.....</b>	<b>5</b>
<b>Table02: Supportability Requirements .....</b>	<b>6</b>
<b>Table03: Adaptability Requirements.....</b>	<b>6</b>
<b>Table04: Scalability or Extensibility Requirements.....</b>	<b>7</b>



## List of Figures

<b>Figure 01: Gantt chart.....</b>	<b>3</b>
<b>Figure 02: Use Case Diagram.....</b>	<b>15</b>
<b>Figure 03: Activity Diagram.....</b>	<b>33</b>
<b>Figure 04: System Sequence Diagram.....</b>	<b>34</b>
<b>Figure 05: Sequence Diagram .....</b>	<b>35</b>
<b>Figure 06: Class Diagram.....</b>	<b>45</b>
<b>Figure 07: Entity-Relationship Diagram.....</b>	<b>46</b>
<b>Figure 08: User Manual.....</b>	<b>54</b>

## CHAPTER 1: INTRODUCTION

### 1.1 Project Overview

IT Agency is an online base website Pre-Order and Selling platform in Bangladesh. Our technology will help to find who wants to create or buy websites. Here are a lot of services where clients and server sites can message for communication. Our service as like [website, apps, seo, digital marketing], It agency have lots of product in website user can buy and ordering then user get support us, as like as a customer when buy the website he doesn't domain hosting publishing we are support her publishing domain hosting, in future face the problem we are support 3 month free service.

### 1.2 Project Purpose

We are living in the generation of technology.

- It Agency anyone can get our services this website
- It Agency website user can buy software/apps purchases this website at lower cost
- This website is user friendly that customer can afford software easily

#### 1.2.1 Background

It agency is an online-based platform, we are living in the generation of technology, this website people get a lot of services there for user get easily her facility, this website has a lot of products like [ web application, apps, SEO, digital marketing], customer can buy the product and admin can panel order confirmation this order.

Users can pre-order this website like [ web application, apps] then fill-up the form product title, product documentation, and project brief description.

customer can hire the freelancers our it agency support then the customer can contact freelancer phone or email,

admin can add the software then the user can purchase this software in this website, and we support 3month customer support when a customer buy the website we can set up domain hosting publish after we are supporting 3month service free

### **1.2.2 Benefits & Beneficiaries**

It agency is an application that attracts more users. This system will make sure that there is no waste of money. Can improve user satisfaction. Providing relevant information. By using this system people can easily get solutions to their problems.

### **1.2.3 Goals**

The objective of the project targets:

- We provide lots of services (Web, Mobile App, Desktop App, etc.).
- The client can communicate with the admin.
- This platform will provide a Pursuing website and we take pre-order.
- Admin can see the client's feedback.
- Admin can be seen product those are sell before

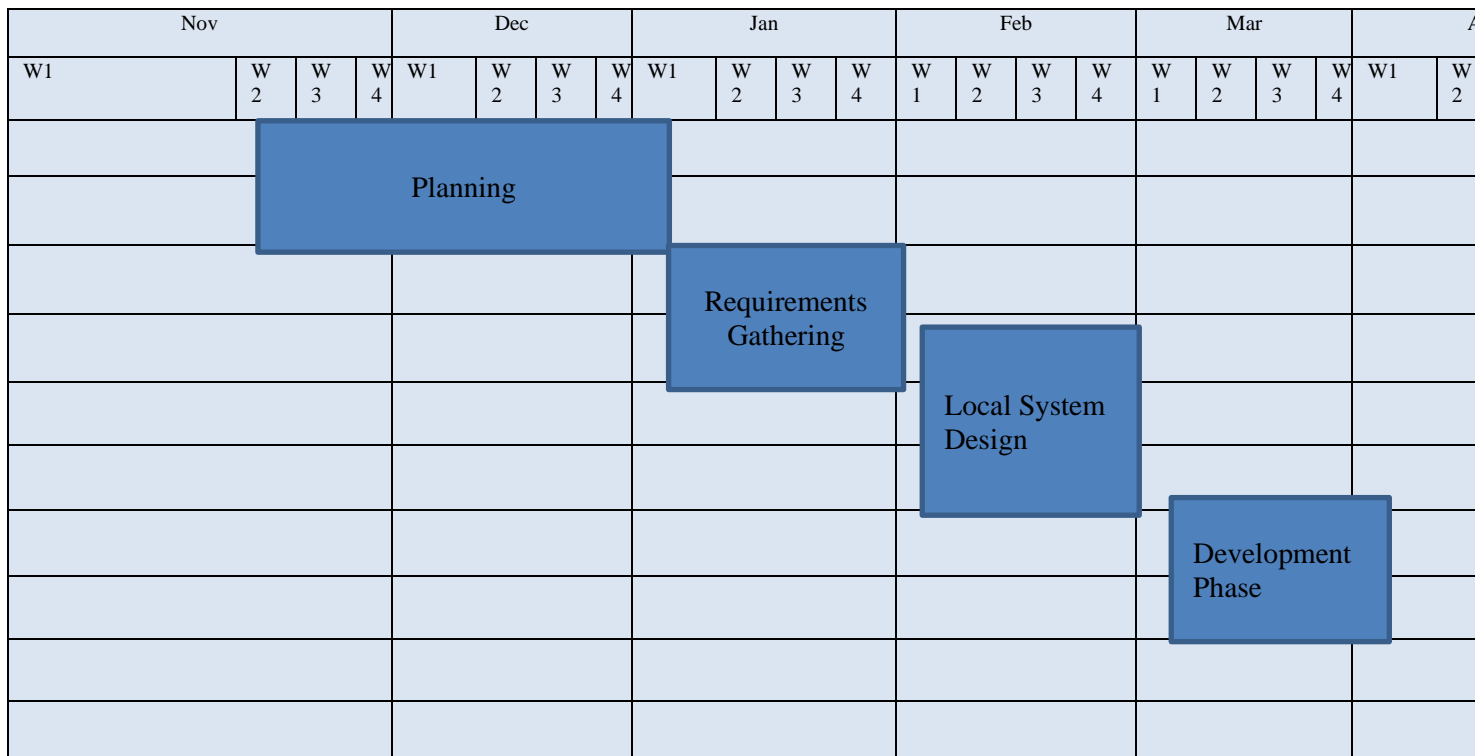
### **1.3 Stakeholders**

Those who are using our web system and also android apps are our stakeholders. The user is stakeholders.

### **1.4 Project Schedule**

Though our project is big, this project will take time. But I am trying to finish the web and mobile application part from the whole system for our Final Defense. So, I used (5) months to make our system. Here's the Gantt Chart. So that I can easily find out when I have finished our work

### 1.4.1 Gantt Chart



### 1.5.2 Release Plan/Milestone

Because our system is not a small project. I can't finish it within a short time. So, I will update it & release the updated version next time. On our next release date, I will try to use an Appointment system & will try to make the system more reliable

Release plan-1: I will try to release our system on July 2021

## CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION

### 2.1 Functional Requirements

Our application is an IT Agency Support. There are produce many services, Example (Web Application, App, SEO, etc) and buy Software, this provides a registration and login for Users. Users can register by giving their necessary details. After successful registration, the user can log in by giving a username and password. Then the user has to fill up their profile. A user can see other user-profiles and also post.

## **2.2 Data Requirements**

Data requirements refer to those data which are needed to build the system model. For my project, I need to focus on some points such as:

- 1) User Information:
  - User Identification
  - Signup/Login
  - Authentication
  - View another user information
  
- 2) Create Service:
  - Create Freelancers
  - Create Product
  - Order Information
  - Modify Whole System,

## **2.3 Performance Requirements**

Performance requirements are one of the most important things for an application. If the performance is well, then the Application will be more useable.

### **2.3.1 Speed and Latency Requirements**

Speed: The system will run at a high speed.

Latency: The latency will be also user-friendly.

### **2.3.2 Precision or Accuracy Requirements**

The precision and the accuracy is for the research purpose however we know And implemented the project. This project is fully functional in localhost access.

### **2.3.3 Capacity Requirements**

User capacity: About 15000 people can access the website at a time. The

capacity will be increased according to the users

## **2.4 Dependability Requirements**

However, this project has dependability because it depends on the frameworks and users.

### **2.4.1 Reliability Requirements**

Reliability is an important non-functional requirement for most software products so a software requirements specification (SRS) should contain a reliability requirement, and most do. A software product will fail under certain conditions, with certain inputs, and given the same inputs and conditions will fail every time until the cause of the failure is corrected. So, I tried to make our software more reliable so that users can easily use our system & get better service from us.

### **2.4.2 Availability Requirements**

This project idea unique but and the numbers of availability of the Software will increase based on the increments of the users. We plan to publicly launch the sites and we capable to manage large datasets

### **2.4.3 Robustness or Fault-Tolerance Requirements**

This project has the tolerance of the least user right now as it is in the localhost so based on the computer requirement and the configuration it can manage the datasets.

### **2.4.4 Safety-Critical Requirements**

Safety and privacy are maintained in the project. In our privacy policy, this is mentioned that no data will be disclosed or shared publicly.

## **2.5 Maintainability and Supportability Requirements**

At least one backup server with the same configuration as in the main server is also recommended for fault tolerance and better performance. Separate storage (with backup) for the database, electronic document, and manuscript is also recommended. Multiple computing nodes with storage are required for high availability and to enhance the performance of the application. Again, after a certain period, the preliminary manuscript files and other files related to that can be deleted manually from the database to increase the performance.

### 2.5.1 Maintenance Requirements

<b>Requirement No.</b>	<b>Requirement</b>
L-1	While login matches the username with user type
L-2	User will get the functionalities of his/her type
L-3	Login time should be stored in the log file

### 2.5.2 Supportability Requirements

<b>Requirement No.</b>	<b>Requirement</b>
A-1	Admin will log in using email & password
A-2	Admin will add Service & Software name
A-3	Admin can add Generic & Category name of the Software.

### 2.5.3 Adaptability Requirements

<b>Requirement No.</b>	<b>Requirement</b>
AI-1	Seller & user both can add their personal information.
AI-2	Users may need to add their information.
AI-3	A seller needs to add their specialty information.

### 2.5.4 Scalability or Extensibility Requirements

<b>Requirement No.</b>	<b>Requirement</b>
CI-1	customer can check their order
CI-2	Admin can check their information & can add edit.
CI-3	Then all update is saving in the database

### 2.6 Security Requirements

Each time there is a security violation. For this reason, when the user will register, the user will receive a mail & have to verify this for login. Otherwise, a user can't log in to the main option.



### **2.6.1 Access Requirements**

To access the website a user must need the credentials to access the website, otherwise, the user only can view the website can't order or purchase the Software the other access.

### **2.6.2 Integrity Requirements**

The operational demand definition method includes the subsequent activities:

1. Identify stakeholders who will or have an interest in our system. So mainly our stakeholders are Sellers & customers. They should have Register & Login for their other activities.
2. Establish measures of effectiveness and suitability, so that users can easily access & can use our system.

A system of dependableness relies upon a stable atmosphere. The look of the environmental system for your information center should make sure that every system will operate faithfully whereas, remaining at intervals the vary of its in operation specifications. Therefore, I attempted to make our system's atmosphere additional reliable so that users will simply access the system with no delay.

### **2.6.3 Privacy Requirements**

Privacy is an important part of the business model or any website. We ensure the privacy and safety of the database that is going to be stored in the Service & Software database. There will be no pirates with the data.

## **2.7 Usability and Human-Interaction Requirements**

It's very easy to use the website. It's very user-friendly so to interact with the website a user doesn't need anything besides the internet connection to access the website. To order any Software or service a user must need the credential username and password.

### **2.7.1 Ease of Use Requirements**

The user interface must be familiar to users, and so may need to follow a single set of rules consistent with those of the operating system, or other mainstream applications. These days, most vendors do follow this good practice, and it is much less an issue than it used to be.

### **2.7.2 Personalization and Internationalization Requirements**

The structure of your e-commerce system, including the software itself, the externally visible properties of the user interface, and the relationships between them.

Consider your new requirements for international markets, finding the balance of what is not in your e-commerce site that needs to be added. Likewise, examine what is in your site's code that needs to be changed to support the markets.

### **2.7.3 Understandability and Politeness Requirements**

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders.

Generally, functional requirements describe system behavior under specific conditions. For instance:

A search feature allows a user to hunt among various invoices if they want to credit an issued invoice.

### **2.7.4 Accessibility Requirements**

Accessibility focuses on how a disabled person accesses or benefits from a site, system, or application. Accessibility is an important part of designing our site and should be considered throughout the development process.

### **2.7.5 User Documentation Requirements**

The user requirement(s) document (URD) or user requirement(s) specification (URS) is a document usually used in software engineering that specifies what the user expects the software to be able to do.

Once the required information is completely gathered it is documented in a URD, which is meant to spell out exactly what the software must do and becomes part of the contractual agreement. A customer cannot demand features not in the URD, while the developer cannot claim the product is ready if it does not meet an item of the URD.

The URD we used as a guide for planning cost, timetables, milestones, testing, etc. The explicit nature of the URD allows customers to show it to various

stakeholders to make sure all necessary features are described.

### **2.7.6 Training Requirements**

Identify and document the types of training required. More than one type of training may be required for a training group. The following suggests training appropriate for:

1. application staff,
2. operations staff,
3. technical staff.
4. Tips and Hints

The project delivery team may not conduct every training course but may advise the customer as to the training required and recommend applicable vendor training courses.

## **2.8 Look and Feel Requirements**

The look and feel requirements describe the intended spirit, the mood, or the style of the product's appearance. These requirements specify the intention of the appearance and are not a detailed design of an interface

### **2.8.1 Appearance Requirements**

Appearance:

A site must be visually appealing, polished, and professional. Remember, it's reflecting your company, your products, and your services. Our website may be the first, and only, impression a potential customer receives of your company.

An attractive site is far more likely to generate a positive impression and keep visitors on your site once they arrive. As businesses large and small continue to populate the web, your challenge is to attract and keep users' attention. Ideas like this are what PR professionals pay attention to keep our businesses successful.

Functionality:

Every component of your site should work quickly and correctly. Broken or poorly constructed components will only leave your visitors frustrated and disillusioned with your company. Across the spectrum, everything should work as expected, including hyperlinks, contact forms, site search, event

registration, and so on.

Error-free copy:

Remember the exposure your website will get. Double-check your facts and figures, as you don't know who may be quoting you tomorrow. Nor do you want to be recognized or remembered for typos, incorrect grammar and punctuation, or misspellings. Spelling mistakes and bad grammar are as unforgivable on a website as they are in other company materials.

Usability:

A critical, but often overlooked component of a successful website is its degree of usability. Your site must be easy to read, navigate, and understand. Some key usability elements include:

Simplicity:

The best way to keep visitors glued to your site is through valuable content, good organization, and attractive design. Keep your site simple and well organized.

Fast-loading pages:

A page should load in 20 seconds or less via dial-up; at more than that, you'll lose more than half of your potential visitors.

Minimal scroll:

This is particularly important on the first page. Create links from the main page to read more about a particular topic. Even the Search Engines will reward you for this behavior.

Consistent layout:

Site layout is extremely important for usability. Use a consistent layout and repeat certain elements throughout the site.

Prominent, logical navigation:

Place your menu items at the top of your site, or above the fold on either side. Limit your menu items to 10 or fewer. Remember, your visitors are in a hurry -- don't make them hunt for information.

Descriptive link text:

Usability testing shows that long link text makes it much easier for visitors to find their way around a site. Long, descriptive link text is favored by Search Engines, too. Backlinks are important to give users a sense of direction and to keep them from feeling lost. Use a site map, and breadcrumbs, if necessary.

Cross-platform/browser compatibility:

Different browsers often have different rules for displaying content. At a minimum, you should test your site in the latest versions of Internet Explorer (currently, versions 8 and 9), as well as Firefox and Safari.

Screen Resolution:

Screen resolution for the typical computer monitor continues to increase. Today, the average web surfer uses a resolution of 1024 x 768 pixels. However, you need to make sure that what looks good in this setting will also work nicely for other resolutions.

Search Engine Optimized (SEO)

There are hundreds of rules and guidelines for effective search engine optimization, and this isn't the place to cover them all. For starters, follow these simple rules:

Include plenty of written content in HTML format. Don't use Flash, JavaScript, or image-only objects for your navigational items.

Use your important keywords frequently and appropriately in your copy.

Minimize the use of tables and use Cascading Style Sheets for layout and positioning; keep your HTML code clutter-free.

Leverage your links -- make them descriptive and use your keywords in the link text. Many Software have been written about Search Engine Optimization, and its scope is too broad to cover here. You can find more information on this important topic on our Search Engine Optimization page.

We've barely scratched the surface of what makes a website most effective. However, by following these simple guidelines, you will be headed in the right direction. Visit our section on website design and development for more information, or send us an e-mail. We'd be happy to answer your questions!

## **2.8.2 Style Requirements:**

We are always looking for ways to improve how they manage the look, feel, and complexity when building a website. This can be difficult when we have

multiple team members and we are two people in our team so we maintain the below guideline for our project

Style guide:

A style guide is a document of code standards that details the various elements and patterns of a site or application. It is a one-stop place to see all visual styles of the site such as headers, links, buttons, color pallets, and any visual language that is used on the site.

## **2.9 Operational and Environmental Requirements:**

The operational demand definition method includes the subsequent activities:

1) Identify stakeholders who will or have an interest in our system. So mainly our stakeholders are sellers & customers. They should have Register & Login for their other activities.

2) Establish a measure of effectiveness and suitability, so that users can easily access & can use our system.

A system of dependableness relies upon a stable atmosphere. The look of the environmental system for your information center should make sure that every system will operate faithfully whereas, remaining at intervals the vary of its in operation specifications. therefore, attempted to form our systems atmosphere additional reliable so that users will simply access the system with no delay.

### **2.9.1 Expected Physical Environment**

Design is fundamentally an innovative process. The methods discussed in this chapter are intended to support the identification and exploration of design alternatives to meet the requirements revealed by analyses of opportunity space and context of use. The methods are not a substitute for creativity or inventiveness. Rather they provide a structure and context in which innovation can take place. We begin with a discussion of the need for and the methods used to establish requirements based on the concept of user-centered design. The types of methods included here are work domain analysis, workload assessment, situation awareness assessment, participatory design; contextual design; physical ergonomics; methods for analyzing and mitigating fatigue, and the use of prototyping, scenarios, persona, and models, and simulations. As with the descriptions in Chapter 6, each type of method is described in terms of uses, shared representations, contributions to the system design phases, and strengths, limitations, and gaps. These methods are grouped under design because their major contributions are made in the design phase; however, it is important to note that they are also used in defining the context of use and in evaluating design outcomes as part of system operation.

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

Motivation:

Requirements for the interfaces to other applications often remain undiscovered until implementation time. We avoid a high degree of rework by discovering these requirements early.

Fit Criterion

For each inter-application interface, specify the following elements:

- The data content
- The physical material contents
- The medium that carries the interface
- The frequency
- The volume

## **2.9.3 Projectization Requirements**

In the projection requirements, we have followed the following guideline for our project work.

Content

Any requirements that are necessary to make the product into a distributable or salable item.

Considerations

Some products have special needs to turn them into salable or usable products. might consider that the product has to be protected such that only paid-up customers can access it. Ask questions of our marketing department to discover unstated assumptions that have been made about the specified environment and the customers' expectations of how long installation will take and how much it will cost.

## **2.9.4 Release Requirements**

To publish a website there are so many checks list need to check the UI design, any kind of error all the functionality is working properly or not. The database connection and stored process all the valid and legal information.

## **2.10. Legal Requirements**

The Act requires you to disclose certain information about the identity of your company on your website. This information doesn't need to be on every page, but it does need to be easily found so it will typically go on our Contact Us page. The footer section of your pages will be useful to both users, and for your search engine optimization:

### **2.10.1 Compliance Requirements**

The terms and conditions agreement is not only for customers but also highly important for protecting your business by making sure customers know their rights and responsibilities.

It helps you avoid customer uncertainty and misunderstandings, so it should be written with absolute clarity about what should be done in any given situation. Consumers should be informed about all the issues that come with shipping and delivery, the return policy, and the privacy policy as well as information about pricing, payment, taxes, and so on.

### **2.10.2 Standards Requirements**

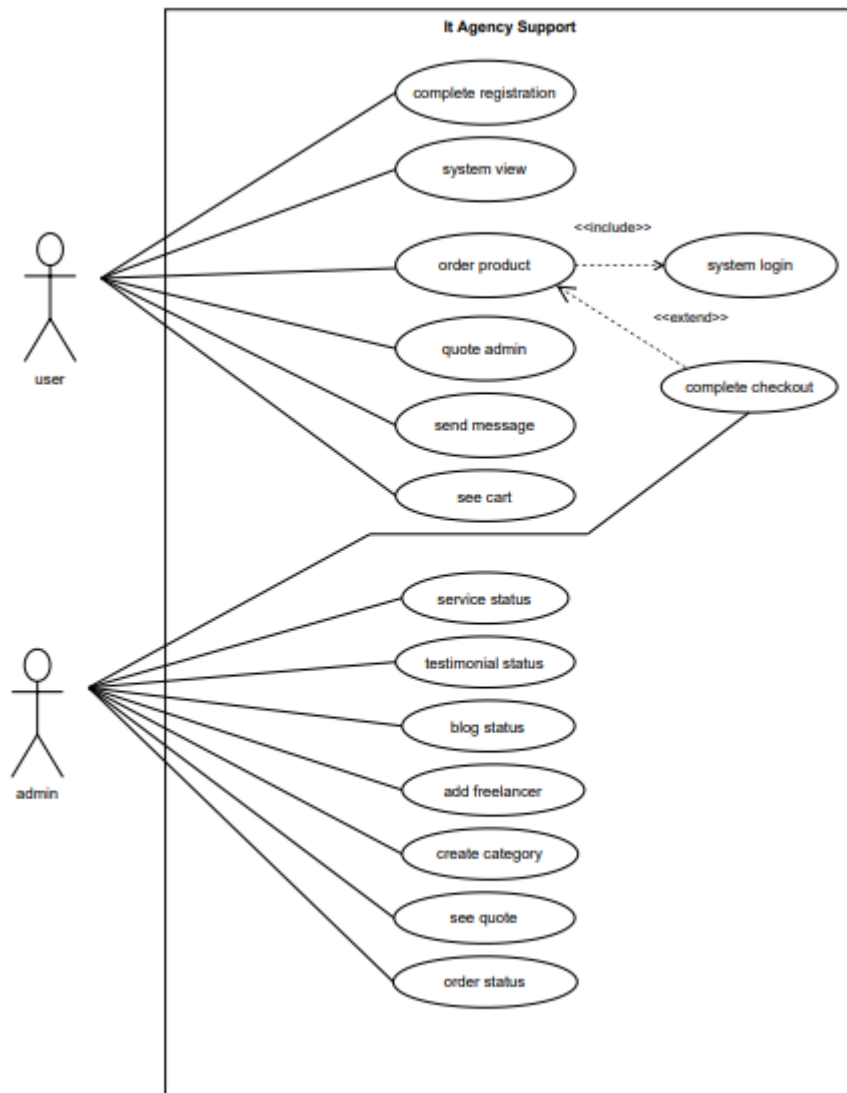
Tech Requirements & Getting Setup The “Right Way”

1. Domain Name
2. Hosting
3. The 2 most common types of hosting you will come across are
4. Website Platform
5. Website Speed
6. SSL Security
7. SEO Friendly
8. Tracking and Analytics

## **CHAPTER 3: SYSTEM ANALYSIS**

### **3.1 Use Case Diagram**





### 3.2 Use Case Description (for each use case)

Use Case Name:	Complete registration										
Scenario:	Complete a registration.										
Triggering Event:	Customer wants to login into the system										
Brief Description:	A new customer needs to first register into the system before performing any transaction.										
Actors:	Customer										
Related Use Case:	none										
Stakeholders:	<table border="1"> <tr> <td>Use Case Name:</td> <td>Complete registration</td> </tr> <tr> <td>Scenario:</td> <td>Complete a registration.</td> </tr> <tr> <td>Triggering Event:</td> <td>Customer wants to login into the system</td> </tr> <tr> <td>Brief Description:</td> <td>A new customer needs to first register into the system before performing any transaction.</td> </tr> <tr> <td>Actors:</td> <td>Customer</td> </tr> </table> System management	Use Case Name:	Complete registration	Scenario:	Complete a registration.	Triggering Event:	Customer wants to login into the system	Brief Description:	A new customer needs to first register into the system before performing any transaction.	Actors:	Customer
Use Case Name:	Complete registration										
Scenario:	Complete a registration.										
Triggering Event:	Customer wants to login into the system										
Brief Description:	A new customer needs to first register into the system before performing any transaction.										
Actors:	Customer										
Preconditions:	An unregistered customer										
Post conditions:	Registered customer										

	<ol style="list-style-type: none"> <li>1. The customer clicks the register button on the Home Page.</li> <li>2. The system displays the register page.</li> <li>3. The customer enters all of the required information.</li> <li>4. The customer clicks the send button.</li> </ol>	<ol style="list-style-type: none"> <li>4.1. The system checks that all of the required information were entered. <ol style="list-style-type: none"> <li>a. If yes, the system update the customer's record in the customer and account tables in the database.</li> <li>b. System displays OK message.</li> </ol> </li> </ol>
Exception Conditions:	1.if customer don't fill up any block of the form ,then customer can not complete registration	

Use Case Name:	System login	
Scenario:	Customer have to login to perform a transaction	
Triggering Event:	When customer want to do a transaction	
Brief Description:	A customer needs to log-in into the system before performing any transaction.	
Actors:	Customer	
Related Use Case:	none	
Stakeholders:	System management	
Preconditions:	A registered user	
Post conditions:	Successful login	
Flow Of Events:	Actor	System

	<ol style="list-style-type: none"> <li>1. The customer clicks the log-in button on the home page.</li> <li>2. The customer enters his/her user ID and password.</li> <li>3. The customer clicks the OK button.</li> <li>4. customer is an authorized user; the system displays the Personal Home Page to the customer</li> </ol>	<ol style="list-style-type: none"> <li>1.1. The system displays the Log-in Page</li> <li>3.1. The system validates the log-in information against the account table in the database.</li> <li>4.1 The system displays the personal home page to the customer.</li> </ol>
Exception Conditions:	1.if customer cannot login , then the customer cannot perform any transaction	

Use Case Name:	Order Products	
Scenario:	A customer can order product to purchase.	
Triggering Event:	After selecting the Product	
Brief Description:	When user add to cart button click then cart store, and cart details menu show	
Actors:	Customer	
Related Use Case:	System login, checkout, product, cart, order	
Stakeholders:	System management	
Preconditions:	User have logged-in.	
Post conditions:	User make a choice for product	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. The customer enters the keyword for a product and clicks the search button on the personal home Page.</li> <li>2. The customer chooses the desired products and clicks add to cart button on the web page.</li> </ol>	<ol style="list-style-type: none"> <li>1.1. The system displays the matching products on the web Page.</li> <li>2.1. The system adds the products into the customer's order table in the database.</li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If user do not order product then user cannot get Software.</li> </ol>	



Use Case Name:	Complete Checkout	
Scenario:	A customer can purchase the Software in his/her shopping cart	
Triggering Event:	When a customer want to buy a Software	
Brief Description:	User can checkout the Software when checkout success then user can see her order status, After ordering the required Software, a customer can pay the money	
Actors:	Customer	
Related Use Case:	none	
Stakeholders:	System management ,	
Preconditions:	The user have logged in and has at least have one Software in the cart	
Post conditions:	Customer checked out the Software	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. The customer clicks the checkout button on the web page.</li> <li>2. The customer checks the order list for any inconsistency. If nothing found, customer clicks the proceed button.</li> <li>3. The customer enters the relevant payment method information and clicks the ok button.</li> <li>4. The customer checks that all information is correct and then check out the Software.</li> </ol>	<ol style="list-style-type: none"> <li>1.1. The system displays the products in the order table of the customer on the web Page.</li> <li>2.1. The system displays the Invoice page.</li> <li>3.1 The system checks that the payment method is valid. Then, the system displays the delivery details page.</li> <li>3.2 The system checks that the payment system is valid. Then, the system displays the delivery details page.</li> <li>4.1 The system will display the check-out information for confirmation.</li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If Customer don't login to the system, then the customer cannot check out the Software.</li> <li>2. If customer don't give information about payment method then the customer cannot check out the Software.</li> <li>3. If customer don't give information about delivery location then the customer cannot check out the Software.</li> </ol>	

Use Case Name:	System View	
Scenario:	A customer can Service, Testimonials, Portfolios, Freelancers, blog, Category	
Triggering Event:	Customer can all menus button access in frontend	
Brief Description:	<p>A Customer Can see the service and get the service when he/she contact us it agency supports,</p> <p>A Customer Can see the Testimonials it agency supports,</p> <p>A Customer Can see the portfolios, portfolios means it agency support before all project show here</p> <p>A Customer Can see the Freelancers and customer can communication it agency support freelancer</p> <p>A Customer Can see the blog and details description and choice category</p>	
Actors:	Customer	
Related Use Case:	System login, System view	
Stakeholders:	System management	
Preconditions:	The user have logged-in	
Post conditions:	The Customer Can see the all system view project	
Flow Of Events:	Actor	System



	<ol style="list-style-type: none"> <li>1. The Customer can see the service when click the Services menus</li> <li>2. when Customer click the testimonial menus</li> <li>3. When Customer Can portfolios menus click</li> <li>4. All freelancer manus when click the customer</li> <li>5. Blog menus click the customer t</li> </ol>	<ol style="list-style-type: none"> <li>1.1 The system displays the Service it agency support</li> <li>2.1 The System displays the all testimonials it agency support</li> <li>3.1. The system displays all portfolios show it agency support</li> <li>4.1 The System displays all freelancer show the system in it agency support</li> <li>5.1 The system display all blog with category show in the it agency support</li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If a user do not login to the system then user cannot sell the Software.</li> <li>2. If a user do not input correct about the products he going to sell then he cannot sell the Software.</li> </ol>	

Use Case Name:	Service Status
Scenario:	Admin want to control a it Agency service status
Triggering Event:	It occurs when admin want to manage a service
Brief Description:	Admin can edit their service like adding new service, remove a service and update service status.
Actors:	Admin
Related Use Case:	System login
Stakeholders:	Admin management
Preconditions:	Admin check a service
Post conditions:	Admin updated the status of the service

Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. Admin enter the system after login</li> <li>2. Admin add a new service.</li> <li>3. Admin approve a service from a user after checking the service.</li> <li>4. Admin remove a service.</li> <li>5. Admin update the status of a service which is unavailable.</li> </ol>	<ol style="list-style-type: none"> <li>2.1. System added a new service</li> <li>4.1. System remove the service.</li> <li>5.1. System update the service status</li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If admin do not login then admin cannot edit a service status</li> <li>2. If user give wrong information about then admin will not approve the service</li> </ol>	

Use Case Name:	testimonials Status
Scenario:	Admin want to control a it Agency testimonials status
Triggering Event:	It occurs when admin want to manage a testimonial
Brief Description:	Admin can edit their testimonials like adding new testimonials, remove a testimonials and update testimonials status.
Actors:	Admin
Related Use Case:	System login
Stakeholders:	Admin management

Preconditions:	Admin check a testimonials	
Post conditions:	Admin updated the status of the testimonials	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. Admin enter the system after login</li> <li>2. Admin add a new testimonials.</li> <li>3. Admin remove a testimonials.</li> <li>4. Admin update the status of a testimonials which is unavailable.</li> </ol>	<ol style="list-style-type: none"> <li>2.1 System added a new testimonials</li> <li>4.1 System show the testimonials. <ol style="list-style-type: none"> <li>a. System update the testimonials status</li> </ol> </li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>5. If admin do not login then admin cannot edit a testimonials status</li> <li>6. If user give wrong information about then admin will not approve the testimonials</li> </ol>	

Use Case Name:	blogs Status
Scenario:	Admin want to control a it Agency blogs status
Triggering Event:	It occurs when admin want to manage a blogs
Brief Description:	Admin can edit their blogs like adding new blogs, remove a blogs and update blogs status.

Actors:	Admin	
Related Use Case:	System login	
Stakeholders:	Admin management	
Preconditions:	Admin check a blogs	
Post conditions:	Admin updated the status of the blogs	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. Admin enter the system after login</li> <li>2. Admin add a new blogs.</li> <li>3. Admin remove a blogs.</li> <li>4. Admin update the status of a blogs which is unavailable.</li> </ol>	<ol style="list-style-type: none"> <li>2.1 System added a new blogs</li> <li>4.2 System show the blogs. <ol style="list-style-type: none"> <li>a. System update the blogs status</li> <li>b. System remove the blogs</li> </ol> </li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>5. If admin do not login then admin cannot edit a testimonials status</li> <li>6. If user give wrong information about then admin will not approve the testimonials</li> </ol>	

Use Case Name:	Add Freelancer
----------------	----------------

Scenario:	Admin want to control a it Agency Freelancer	
Triggering Event:	It occurs when admin want to manage a Freelancer	
Brief Description:	Admin can edit their Freelancer like adding new Freelancer, remove a Freelancer and update Freelancer status.	
Actors:	Admin	
Related Use Case:	System login	
Stakeholders:	Admin management	
Preconditions:	Admin check a Freelancer	
Post conditions:	Admin updated the status of the Freelancer	
Flow Of Events:	Actor	System
	<ul style="list-style-type: none"> <li>7. Admin enter the system after login</li> <li>8. Admin add a new Freelancer.</li> <li>9. Admin remove a Freelancer.</li> <li>10. Admin update the status of a Freelancer which is unavailable.</li> </ul>	<ul style="list-style-type: none"> <li>2.1 System added a new Freelancer</li> <li>4.3 System show the Freelancer. <ul style="list-style-type: none"> <li>a. System update the Freelancer status</li> <li>b. System remove the Freelancer</li> </ul> </li> </ul>
Exception Conditions:	<ul style="list-style-type: none"> <li>11. If admin do not login then admin cannot edit a Freelancer status</li> <li>12. If user give wrong information about then admin will not approve the Freelancer</li> </ul>	

Use Case Name:	Create Category	
Scenario:	Admin want to control a it Agency Category	
Triggering Event:	It occurs when admin want to manage a Category	
Brief Description:	Admin can edit their Category like adding new Category, remove a Category and update Category status.	
Actors:	Admin	
Related Use Case:	System login	
Stakeholders:	Admin management	
Preconditions:	Admin check a Category	
Post conditions:	Admin updated the status of the Category	
Flow Of Events:	Actor	System

	<ul style="list-style-type: none"> <li>13. Admin enter the system after login</li> <li>14. Admin add a new Category.</li> <li>15. Admin remove a Category.</li> <li>16. Admin update the status of a Category which is unavailable.</li> </ul>	<ul style="list-style-type: none"> <li>2.1 System added a new Category</li> <li>4.4 System show the Category. <ul style="list-style-type: none"> <li>a. System update the Category status</li> <li>b. System remove the Category</li> </ul> </li> </ul>
Exception Conditions:	<ul style="list-style-type: none"> <li>1. If admin do not login then admin cannot edit a Category status</li> <li>2. If user give wrong information about then admin will not approve the Category</li> </ul>	

Use Case Name:	See Cart
Scenario:	A customer can Add to cart his/her cart.
Triggering Event:	Customer puts his/her cart
Brief Description:	A Customer can add product cart and show all carts item list
Actors:	Customer
Related Use Case:	Log-in
Stakeholders:	System management

Preconditions:	The user have logged-in	
Post conditions:	The order table has been updated.	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. The customer add to cart button then cart add</li> <li>2. The customer clicks the send button on the webpage.</li> <li>3. The customer checks the cart list</li> </ol>	<ol style="list-style-type: none"> <li>1.2 The system displays the Sell used Software web page.</li> <li>3.2. The system displays a all cart list</li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If a user do not login to the system then user cannot cart add.</li> </ol>	

Scenario:	Order Status
Triggering Event:	Occurs when customer order a Software
Brief Description:	When a user order to buy a Software admin will check whether the Software is available or not if the Software is available then admin gives approval to sell the Software.



Actors:	Admin , user	
Related Use Case:	System login	
Stakeholders:	Admin management, System management	
Preconditions:	Admin checking a the ordered Software	
Post conditions:	Admin approved the ordered Software	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. A customer login to the system and order a Software.</li> <li>2. Admin check the availability of the Software.</li> <li>3. If the Software is available then admin can approved the order to sell the Software.</li> <li>4. User will get the Software</li> </ol>	
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If the Software is not available then admin cannot approved the order of the Software.</li> </ol>	

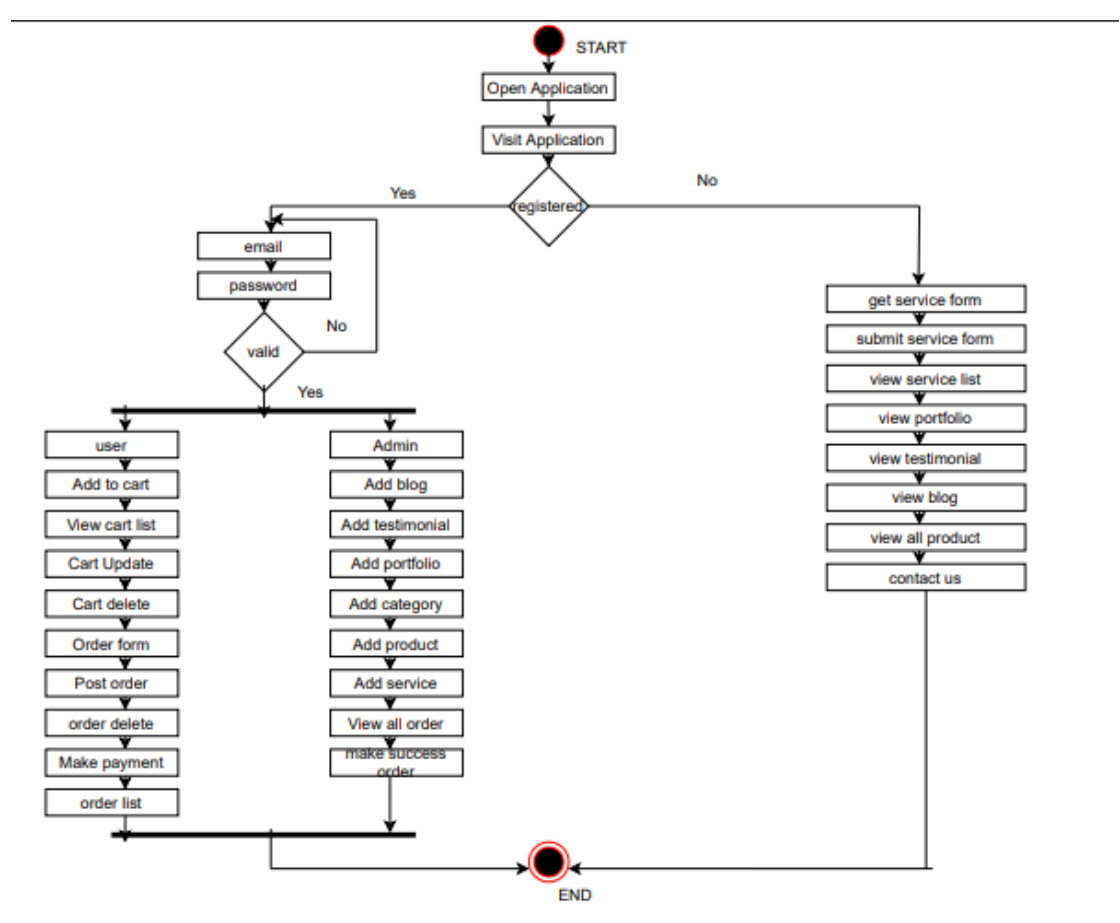
Use Case Name:	Quotes Status	
Scenario:	User send Quotes to admin	
Triggering Event:	Occurs when user send Quotes to admin	
Brief Description:	User can directly send email to admin to share Quotes details information's.	
Actors:	User , admin	
Related Use Case:	none	
Stakeholders:	Admin management	
Preconditions:	User send email to admin	
Post conditions:	Admin received email from user and replied to user	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> <li>1. User send an email to admin to reply Quotes.</li> <li>2. Admin received the system</li> <li>3. Admin feedback the quotes</li> </ol>	<ol style="list-style-type: none"> <li>1.1 Quotes replay</li> <li>3.1 problem solved</li> </ol>
Exception Conditions:	<ol style="list-style-type: none"> <li>1. If user don't send email to admin then admin cannot receive the System.</li> </ol>	



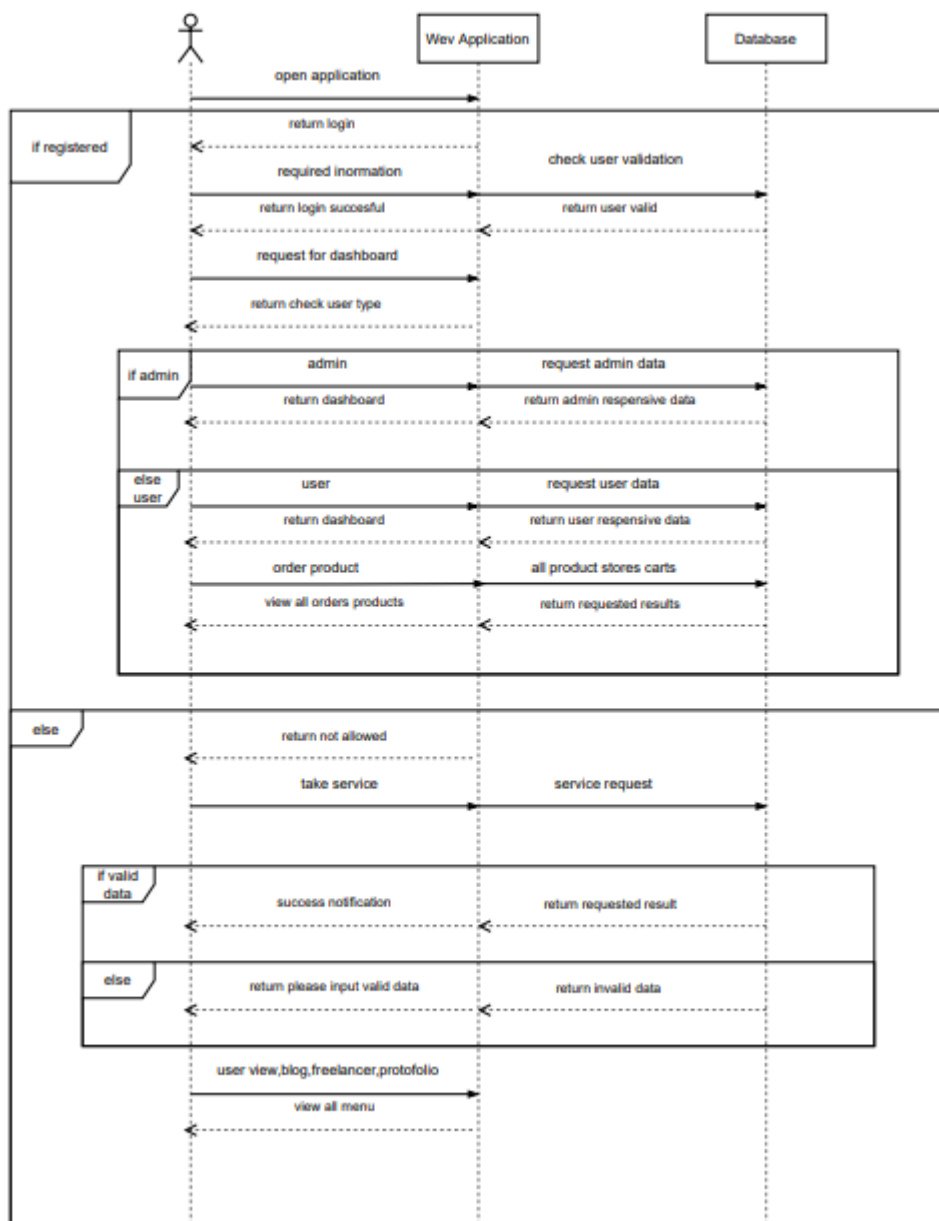
Use Case Name:	Check message	
Scenario:	User send message to admin	
Triggering Event:	Occurs when user send message to admin	
Brief Description:	User can directly send email to admin to share any problem about system.	
Actors:	User , admin	
Related Use Case:	none	
Stakeholders:	Admin management	
Preconditions:	User send email to admin	
Post conditions:	Admin received email from user and replied to user	
Flow Of Events:	Actor	System
	<ul style="list-style-type: none"> <li>4. User send an email to admin to solve a problem about system.</li> <li>5. Admin received the email</li> <li>6. Admin solved the problem</li> <li>7. Admin replied to the user about solving the problem</li> </ul>	<ul style="list-style-type: none"> <li>1.1 problem created</li>   <li>3.1 problem solved</li> </ul>

Exception Conditions:	2. If user don't send email to admin then admin cannot receive the email.
-----------------------	---

### 3.1 Activity Diagram (for each use case)



### 3.2 System Sequence Diagram (for each use case)



## CHAPTER 4: SYSTEM DESIGN SPECIFICATION

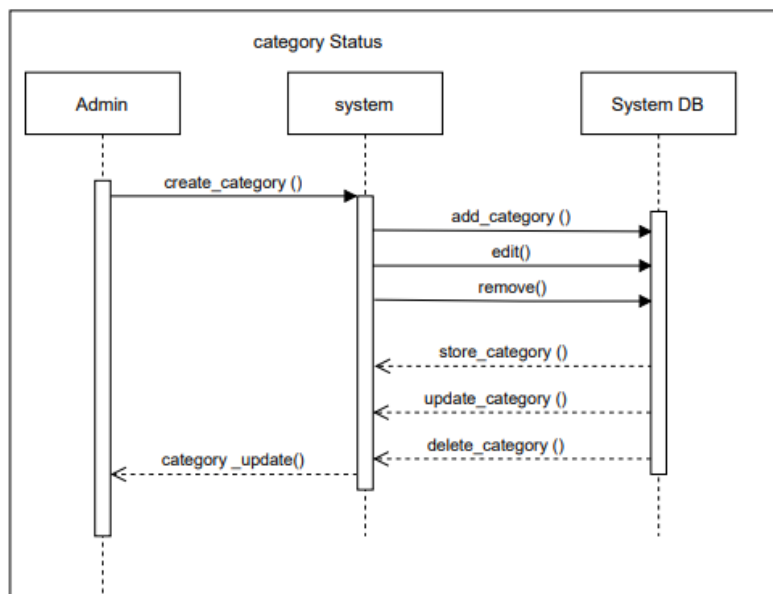
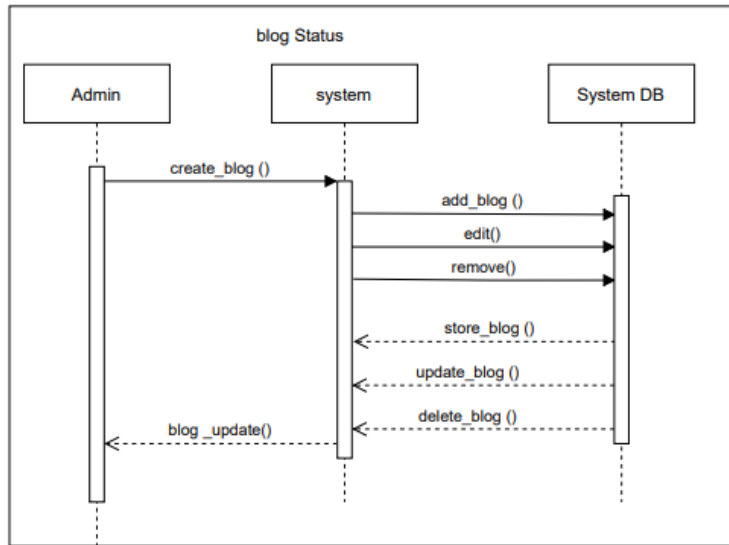
### 4.1 Class Responsibilities Collaboration (CRC) Cards

Although CRC cards were originally introduced as a technique for teaching object-oriented concepts, they have also been successfully used as a full-fledged modeling technique. My experience is that CRC models are an incredibly effective tool for conceptual modeling as well as for detailed design. CRC cards feature prominently in eXtreme Programming (XP) (Beck 2000) as a design technique. My focus here is on applying CRC cards for conceptual modeling with your stakeholders.

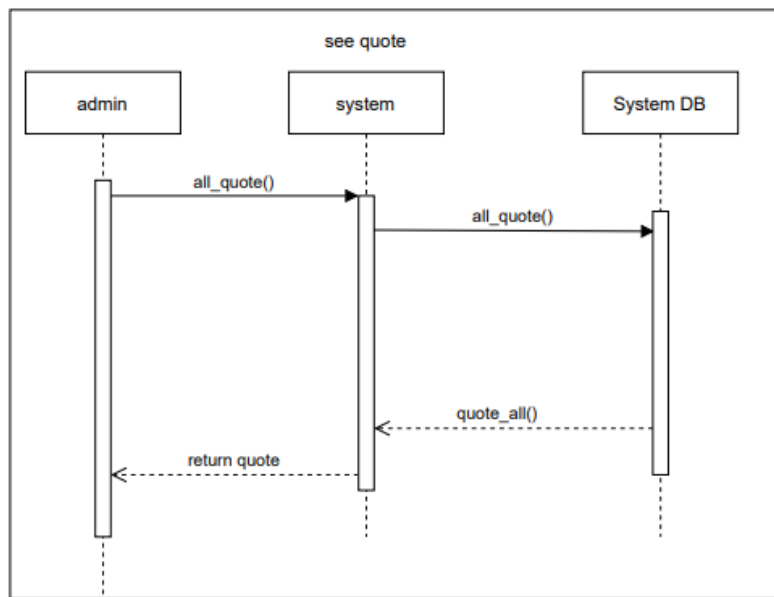
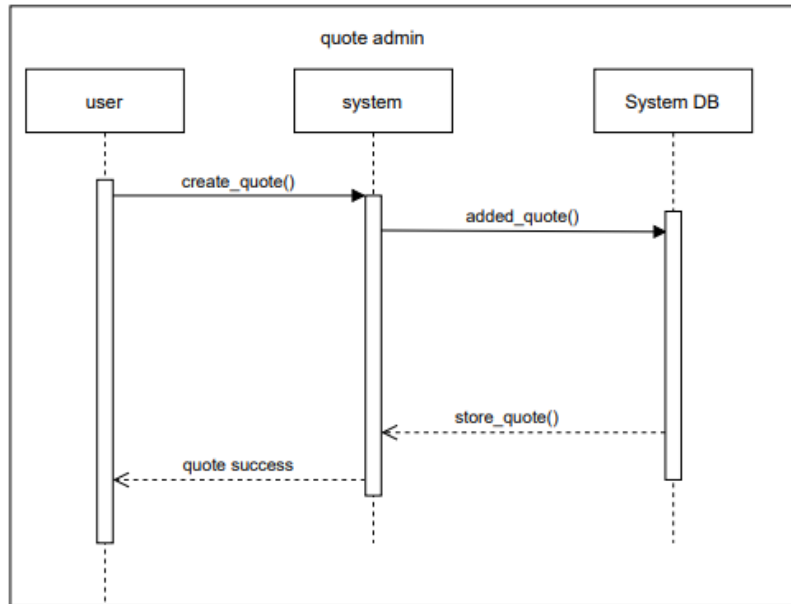
A class represents a collection of similar objects. An object is a person, place, thing, event, or concept that is relevant to the system at hand. For example, in a university system, classes would represent students, tenured professors, and seminars. The name of the class appears across the top of a CRC card and is typically a singular noun or singular noun phrases, such as Student, Professor, and Seminar. You use singular names because each class represents a generalized version of a singular object.

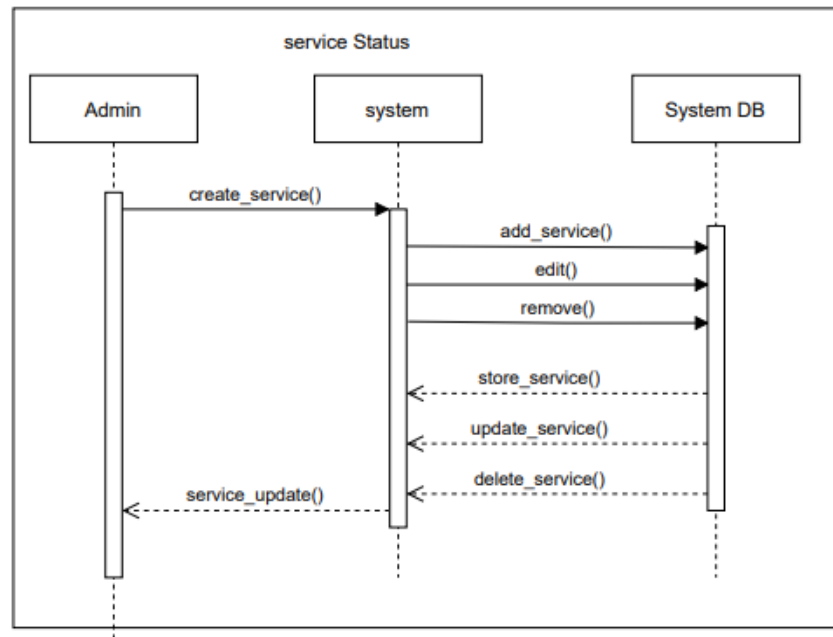
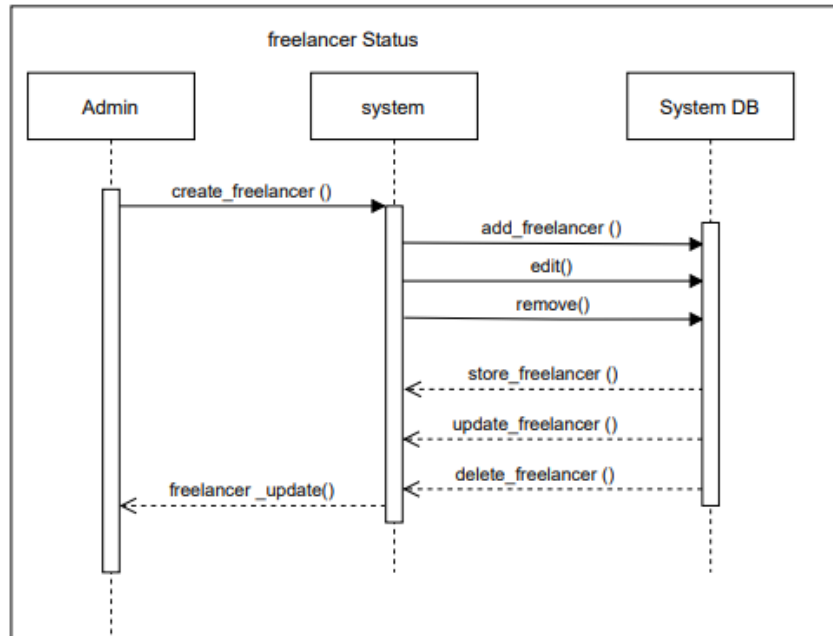
## **4.2 Sequence Diagram (for each use case)**

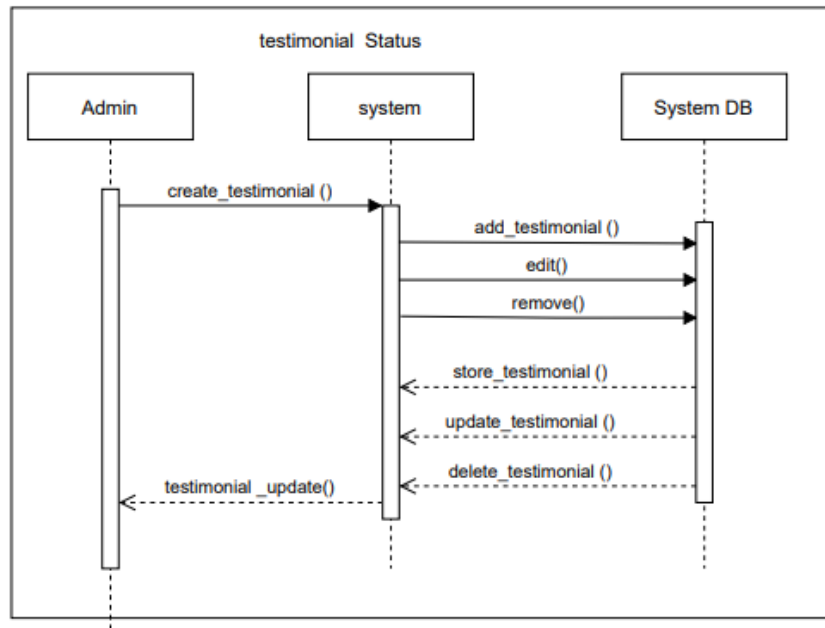
### **1. Admin**



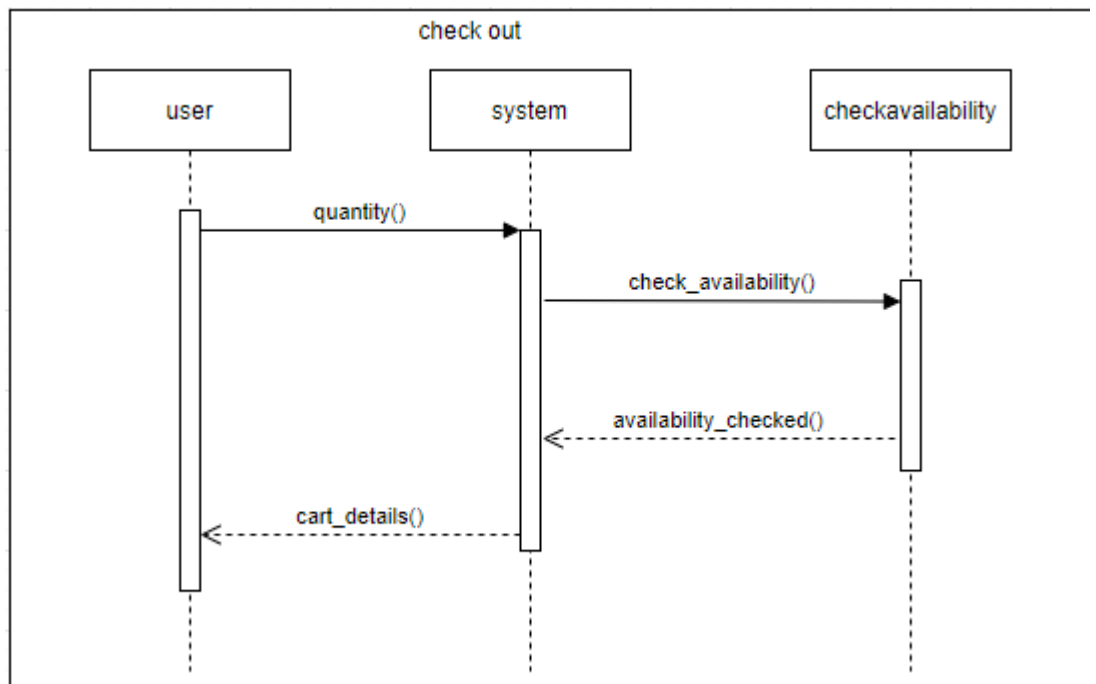


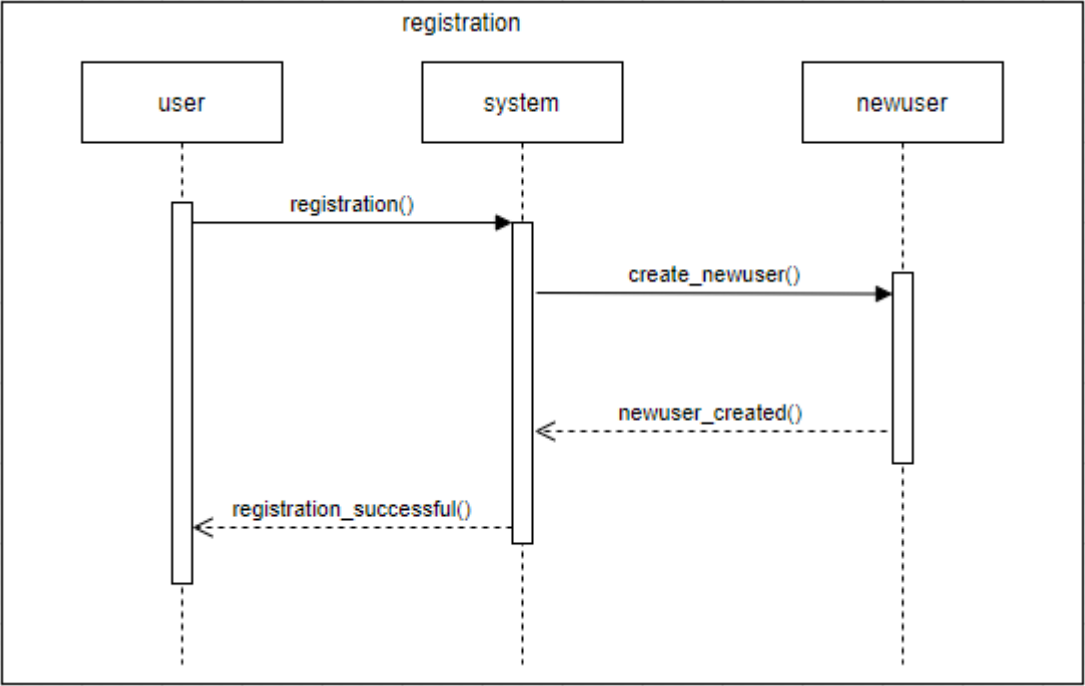
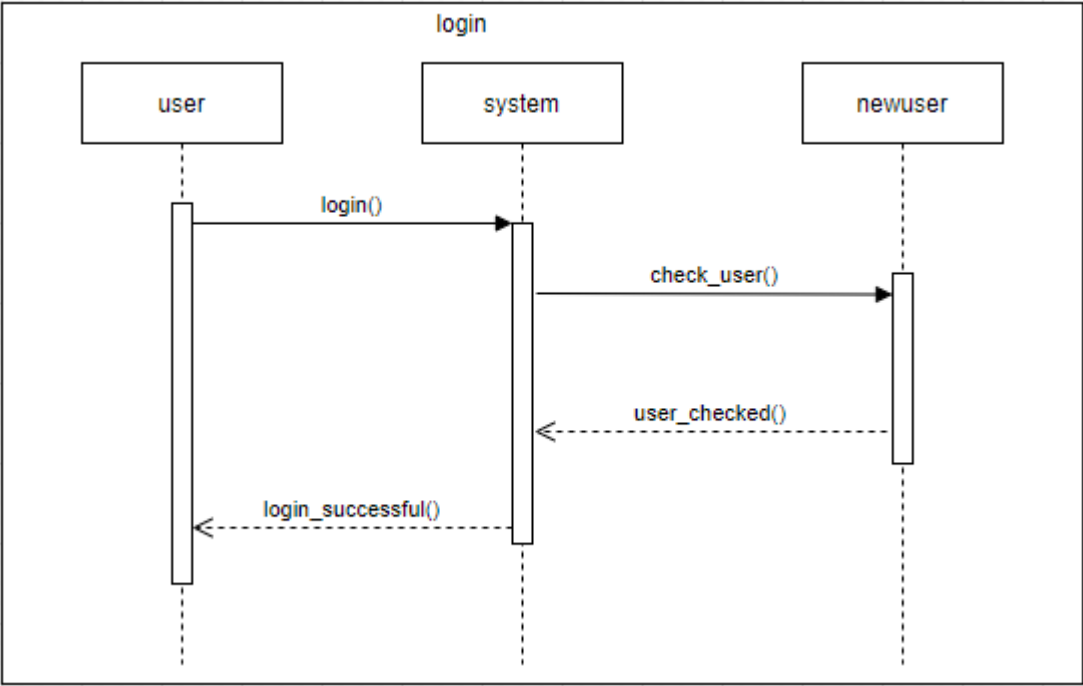


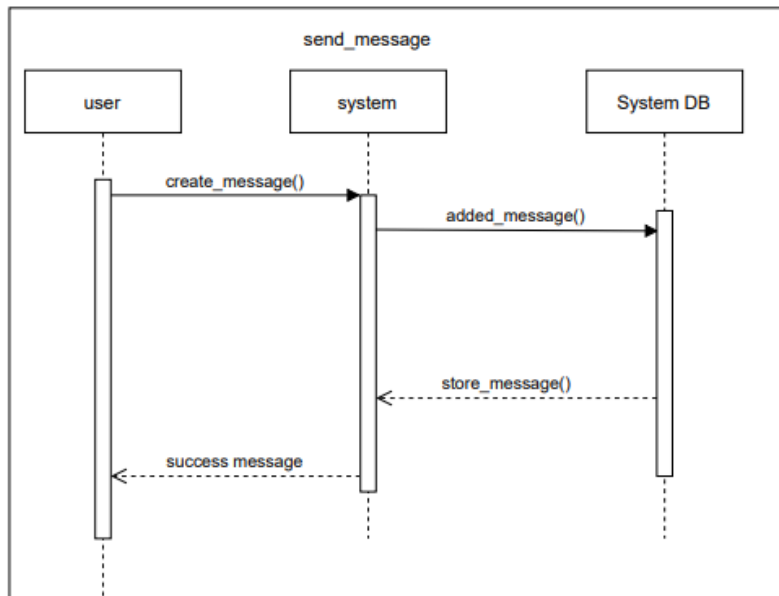
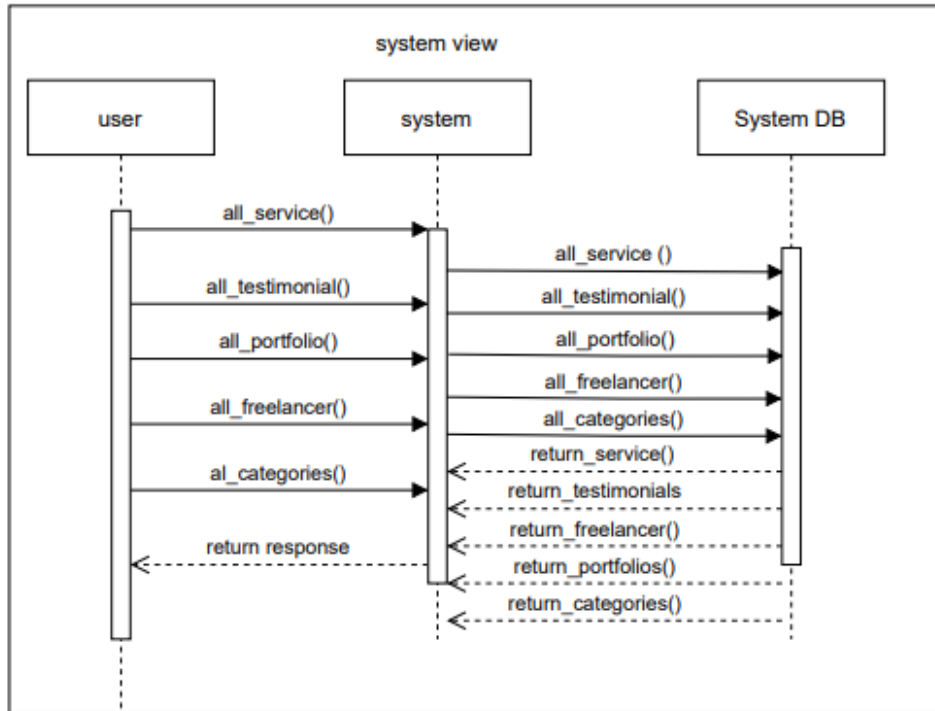




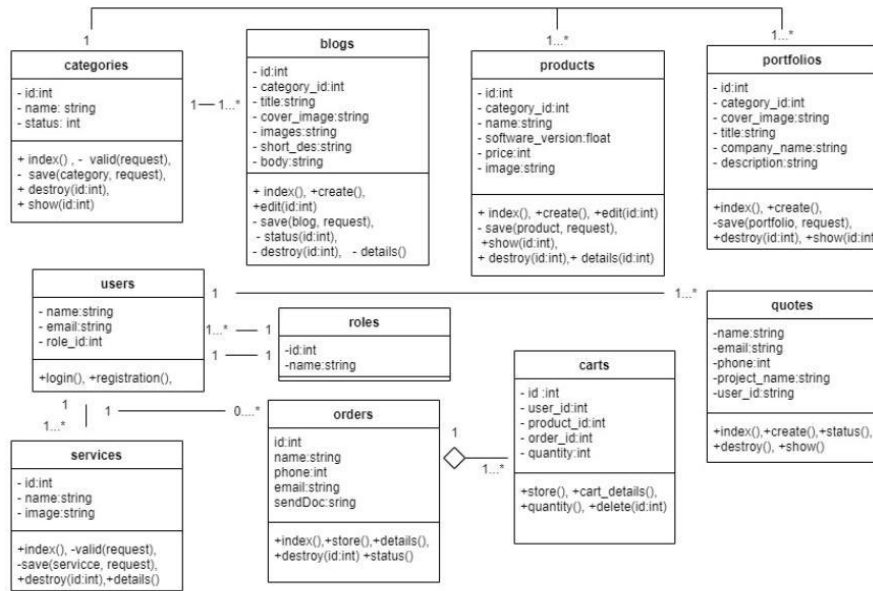
## 2. User



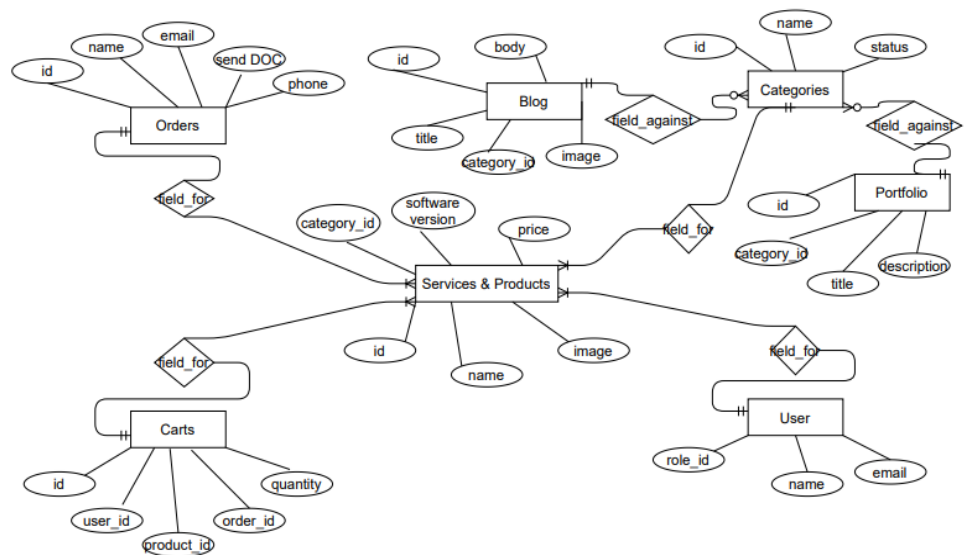




### 4.3 Class Diagram



### 4.4 Database Design Diagram



### 4.6 Development Tools & Technology:

The web development tools that help the developer to ease the workflow of development are:

1. JavaScript Libraries
2. Front-end Frameworks
3. Databases
4. Programming Languages
5. Icons
6. JavaScript Libraries
7. PHP Framework

JavaScript happens to be one of the most popular programming languages for the developer community. The library contains a set of already prepared libraries – that helps in easy access to web development. Some of the famous Javascript libraries are:

1. Front-end Frameworks
2. Php Frameworks

The front-end frameworks constitute folders and files. JavaScript, CSS, and HTML and Bootstrap are some of the examples to be cited.

#### Programming Languages

The languages form the backbone of web development tools. The popular ones used are PHP framework (Laravel), JavaScript and many more.

#### **4.5.1 User Interface Technology:**

User experience marks the top priority for almost every business in the market today. As a beginner in the developer world, you would have come across frequently that even the brief that you were given for your project, mentioned the simplicity of its user interface that it should feature. It doesn't matter how complex are the functions and background operations happening alongside. What the users see and get to experience has to be a seamless and smooth execution. Take any one of them, and you would notice that the platform is powerful yet simple with a brilliant user interface. And that is because of the many versatile frameworks working behind the curtains.

However, there is no denying that the ever-increasing demand of the customers and the market, in general, requires a constant need to put out high-quality

functionality and usability. As this responsibility piles up on the developer, many developers find themselves stuck and confused about which framework to pick among so many.

#### **4.5.1.1 ASP.NET MVC4 Framework or PHP Framework**

Laravel is a web application framework with expressive, elegant syntax. We believe development must be an enjoyable, creative experience to be truly fulfilling. Laravel attempts to take the pain out of development by easing common tasks used in the majority of web projects, such as authentication, routing, sessions, and caching.

Laravel aims to make the development process a pleasing one for the developer without sacrificing application functionality. Happy developers make the best code. To this end, we've attempted to combine the very best of what we have seen in other web frameworks,

#### **4.5.1.3 CSS Framework or Bootstrap:**

Cascade Style Sheet version 3.0 is used in this project. Because of this framework and the bootstrap style the site is responsive. Anyone can use any smart tools to access the website. Bootstrap has consistently been one of the biggest CSS Frameworks there is. It is often a de facto framework, especially for Web Developers. After several years of work, in early 2018, the official 4.0 version was released! Bootstrap 4.0 was a significant update.

#### **4.5.1.4 Font Awesome or Others:**

Font Awesome is used in this project. To make a beautiful website there are various front styles needed font awesome is a database of the font where we all collect the font and linked in our project work. Get vector icons and social logos on your website with Font Awesome, the web's most popular icon set, and toolkit.

Font Awesome is a font and icon toolkit based on CSS and LESS. It was made by Dave Gandy for use with Twitter Bootstrap and later was incorporated into the BootstrapCDN. Font Awesome has a 20% market share among those websites which use third-party Font Scripts on their platform, ranking it second place after Google Fonts.

#### **4.5.2 Implementation Tools & Platforms**

To implement this project various things needed to be done

1. Research Analysis
2. Notepad++/ Bracket
3. PHP framework (Laravel)



4. Microsoft Visual Studio 2010

5. Database

And so many things that are required to developed a professional eCommerce site.

#### **4.5.2.1 Microsoft Visual Studio 2010 or PHP Storm**

Both of the tools are user-friendly however I am confident and used to VMS 2010. It was very easy to figure out the problem and solve it. Also, it helps us to work on the project as a live collaboration.

#### **4.5.2.2 MSSQL Server 2008 or MySQL Community Server 5.5**

For the database, we have used MySQL Community Server 5.5 as we have taught in our university. MySQL is very user-friendly and very easy to use. All the database queries are run on MySQL server.

#### **4.5.2.3 .NET Runtime or Apache HTTP Server**

Apache HTTP Server for the database connection otherwise no performance can be done from the admin to user interface. In the design phase, the login form for user admin is designed to log in and access the database connection is needed. Apache HTTP Server is used for the database connection.

## **CHAPTER 5: SYSTEM TESTING**

### **5.1 Testing Features**

#### **5.1.1 Features to be tested**

- 1) First, I need to test Registration & Login Features.
- 2) Then I need to test the part of the order.
- 3) Then I can test the feature of Approve of the order part.
- 4) I can also test the View order.
- 5) Seller features can be tested. (Like Seller information, Seller Education, Seller Specialty).

#### **5.1.2 Features not to be tested**

- 1) I need not test the feature of the Home Page.

2) I need not test about page, contact page features.

## **5.2 Testing Strategies:**

A testing strategy is an overview that describes the testing approach of the code development cycle. it's created to tell project managers, testers, developers regarding some key problems with the testing method. check methods describe however the merchandise risks of the stakeholder's square measure satisfied at the test level, that varieties of testing square measure to be performed, and that entry and exit criteria apply. they're created supported development style documents. System style documents square measure primarily used, and infrequently abstract style documents could also be stated.

### **5.2.1 Test Approach:**

Test approach is the test strategy implementation of a project, which defines how testing would be carried out. Test approach has two techniques: Proactive - An approach in which the test design process is initiated as early as possible to find and fix the defects before the build is created. This approach is applied in our project.

### **5.2.2 Pass/Fail Criteria:**

The exact pass/fail criteria for load and stress testing are to be determined by ITS. However, in a general sense, if the application performs to an acceptable degree after it has been put under a certain capacity, then the test item will pass. On the contrary, the test item will fail if the application underperforms. Our system is tested with all the test cases and it passed the requirement.

### **5.2.3 Suspension and Resumption:**

It is important to understand that if a defect is detected at a point after which the testing shall resume proves to be of no use, then applying the resources on testing will be futile. One needs to specify the reason for stopping the test activities and define the acceptable level of defects that allows the testing process to surpass those defects. Few defects may arise in resuming the test activity. Those are the hidden defects that were somewhere ignored earlier.:

### **5.2.4 Testing Schedule**

A Test Plan is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. Test Plan helps us determine the effort needed to validate the quality of the application under test. The test plan serves us as a blueprint to conduct software testing activities as a defined process, which is

minutely monitored and controlled by the test manager.

### 5.2.5 Traceability Matrix

The main agenda of every tester should be to understand the client's requirements and make sure that the output product should be defect-free. To achieve this goal, every QA should understand the requirement thoroughly and create positive and negative test cases.

This would mean that the software requirements provided by the client have to be further split into different scenarios and further to test cases. Each of these cases has to be executed individually. A simple way is to trace the requirement with its corresponding test scenarios and test cases. This merely is termed as 'Requirement Traceability Matrix.'

The traceability matrix is typically a worksheet that contains the requirements with its all-possible test scenarios and cases and their current state, i.e. if they have been passed or failed. This would help the testing team to understand the level of testing activities done for the specific product.

1. Requirement Traceability Matrix
2. Requirement ID
3. Requirement Type and Description

Test Cases with Status

### 5.3 Testing Environment (hardware/software requirements)

Network

- Documentation required like reference documents/configuration guides/installation guides/ user manuals

Test Case ID:	Test Name:	Test Case Objective:	Pre-Requisite:	Steps to Perform:	Expected Result:	Output:	Status : Pass/Fail.
---------------	------------	----------------------	----------------	-------------------	------------------	---------	---------------------

001	Sign up with empty fields	To verify the none of the fields remain empty	Displaying sign up form	1. Submit a form with several empty fields. 2. Click the signup button	Error message showing indicating empty fields	Error	pass
002	Sign up with completing fields	Completing the fields with data	Displaying sign up form	1. Submit the form by completing all empty fields. 2. Click the signup button	Sign up successful	Input data in the wrong field	fail
003	Sign up with corrected data	To input the correct data incorrect field	Displaying sign up form	1. Submit the form by completing all correct data incorrect fields. 2. Click the signup button	Sign up successful	Sign up successful	pass
004	Login with empty fields	To verify the none of the fields remain empty	Displaying login form	1. Submit the form with several empty fields. 2. Click the login button	Error message showing indicating empty fields	Error	pass
005	login with completing fields	Completing the fields with data	Displaying login form	1. Submit the form by completing all empty fields. 2. Click the login button	Login successful	The user is not registered	fail
006	login with corrected data	To input the correct data incorrect field	Displaying login form	1. Submit the form by completing all correct data incorrect fields. 2. Click the sign-up button	Login successful	Login successful	pass
007	Order Software by guest	Checking order criteria	Displaying Software	1.guest select Software 2.guest added it to the cart	You need to login	You need to login	pass

Tests are limited to what can be tested and what not should be tested.

Following people are involved in test environment setup

- System Admins,
- Developers
- Testers
- Sometimes users or techies with an affinity for testing

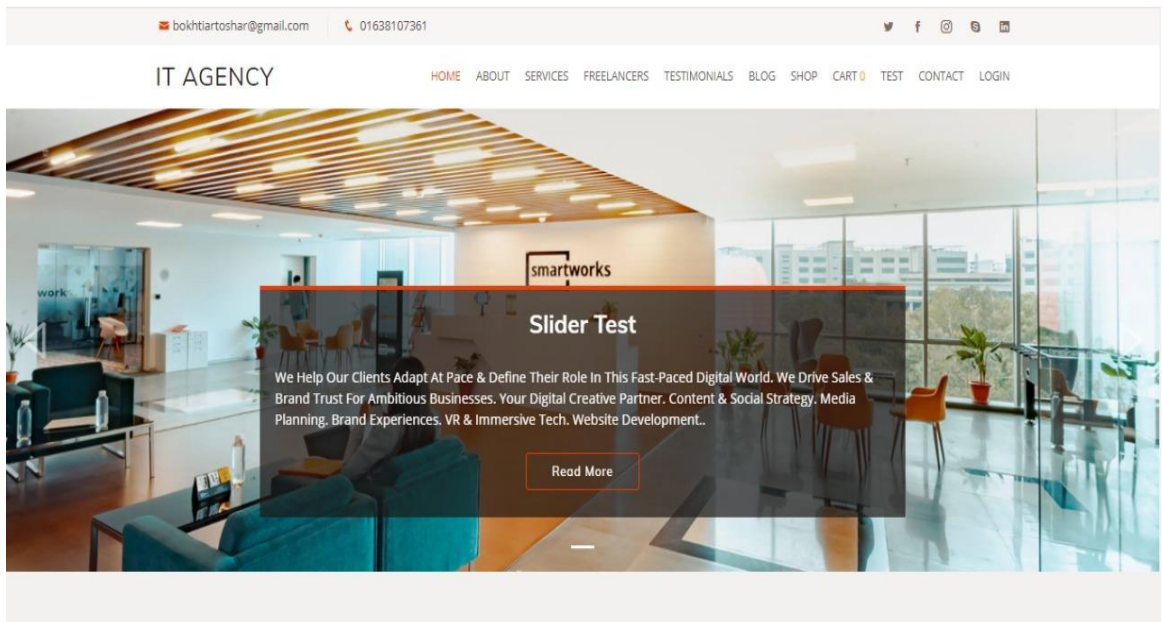
## 5.4 Test Cases

Test Case ID:	Test case name	Test Case Objective:	Pre-Requirement:	Steps to Perform:	Expected Result:	Output:	Status: Pass/Fail.
008	Order Software by user	Checking order criteria	Displaying Software	1.user select Software 2.user added it to the cart	Software added to the cart	Software added to the cart	pass
009	Check Out	Check out method	Displaying payment	1.user can use a payment method to check out the Software	Check out successful	Check out successful	pass
010	Resell Software by user	Resell Software criteria	Displaying Software	1.user can upload their Software 2.user can set their price	Software uploaded	Software uploaded	pass
011	Resell Software by guest	Resell Software criteria	Displaying Software	1.user can upload their Software 2.user can set their price	Software uploaded	You need to login	fail
012	Software status by admin	Add and remove a Software	Displaying Software	1. admin add a new Software in the system 2. admin remove a Software in the system	Software updated	Software updated	pass
013	Software status by admin	Admin approve a Software from the user	Displaying Software	1. user upload a Software 2. admin approve the Software	Software updated	Software updated	pass
014	Order approved by admin	Admin approve sell order	Display order	1. user orders a Software. 2. admin approve the order	Order approved	Order approved	pass



## CHAPTER 6: USER MANUAL

### 6.1 User Manual Home Page)



### 6.2 User Manual (user)

Indexing	product Title	Price	Image	Quantity	Action
1	Inventory Management for Perfex CRM Inventory Management for Perfex CRM	60000		1	<a href="#">Remove Item</a>
2	MLM Matrix Plan	100000		2	<a href="#">Remove Item</a>
3	Stocky - Ultimate Inventory Management System with Pos Stocky - Ultimate Inventory Management System	15796		2	<a href="#">Remove Item</a>

Your Choice Software Please Checkout...

Indexing	product Title	Price	Image	Quantity	Action
1	Inventory Management for Perfex CRM Inventory Management for Perfex CRM	60000		<input type="text" value="1"/> Submit	Remove Item

[Continue Shopping](#)

Total Taka: 60000 TK

[Checkout Shopping](#)

- Categories
- Inventory
- pos
- E-commerce
- MLM



**Inventory Management for Perfex CRM Inventory Management for Perfex CRM**

Price: 60000 Taka

[Visit Website](#) [ADD TO CART](#)

by GreenTech\_Solutions in Add Ons  
Software Version: PHP 7.x - 8.x

Software Framework: Laravel

Software languse: JavaScript JS  
HTM LCSSS PHP



**Stocky - Ultimate Inventory Management System with Pos Stocky - Ultimate Inventory Management System**

Price: 7898 Taka

[Visit Website](#) [ADD TO CART](#)

by ui-lib in Project Management T  
Software Version: PHP 7.x - 8.x

Software Framework: Laravel



### Website Creation Sites

Admin 2 months ago

A web developer is a programmer who specializes in, or is specifically engaged in,

[Read More](#)

#### Search

#### Categories

- Website (0)
- Apps Development (1)

#### Recent Posts


-  Website Creation Sites  
2 months ago
-  apps development  
2 months ago



bokhtiar\_toshar@gmail.com 01638107361


IT AGENCY HOME ABOUT SERVICES FREELANCERS TESTIMONIALS BLOG SHOP CART 0 TEST CONTACT LOGIN

Testimonials [Home / Testimonials](#)



**bokhtiar toshar**  
WebTech Group's Backend Developer

“ Develop amazing web and app experiences with tips and advice from the experts at Web.Dev. Keep your users engaged with the latest web development capabilities! ”




**istiake ahmed novel**  
CEO WebTech group

“ A web developer is a programmer who specializes in, or is specifically engaged in, the development of World Wide Web applications using a client-server model ”

bokhtiar\_toshar@gmail.com 01638107361


IT AGENCY HOME ABOUT SERVICES FREELANCERS TESTIMONIALS BLOG SHOP CART 0 TEST CONTACT LOGIN

Services [Home / Services](#)




**WEB DEVELOPMENT**

It's Never Too Late to Learn a New Skill! Learn to Code and Join Our 45+ Million Users. Enjoy Extra Quizzes & Projects and Exclusive Content.




**APPS DEVELOPMENT**

Build Native Apps on IOS and Android From a Single Codebase. Download and Build Today.



**DIGITAL MARKETING**

Experience modern teaching methods and lectures that will equip you with business skills.




**GAME DEVELOPMENT**

Game Development is the art of creating games and describes the design, development and release of a game

bokhtiar\_toshar@gmail.com 01638107361

IT AGENCY HOME ABOUT SERVICES FREELANCERS TESTIMONIALS BLOG SHOP CART 0 TEST CONTACT LOGIN

About [Home / About](#)



## WebTech Group's

WebTech is essentially about bringing together the development and the operations teams in a software development firm. Both teams collaborate to eliminate silos and strive for quicker innovation adoption.

## Some of our recent works

APPS WEB SEO GAMING WEBSITE APPS DEVELOPMENT



## We've created more than 200 websites this year!

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

REQUEST A QUOTE

## Check out our services



### WEB DEVELOPMENT

It's Never Too Late to Learn a New Skill! Learn to Code and Join Our 45+ Million Users. Enjoy Extra Quizzes & Projects and Exclusive Content.



### APPS DEVELOPMENT

Build Native Apps on iOS and Android From a Single Codebase. Download and Build Today.



### DIGITAL MARKETING

Experience modern teaching methods and lectures that will equip you with business skills.

## 6.4 Admin panel

The screenshot displays the admin panel for 'TECH88'. On the left is a sidebar menu with options: Dashboard, Services, Category, Portfolio, Testimonial, Blog, Slider, Contact, Freelancers, Quote, Client, About, and Product Category. The main content area is titled 'blog Form' and includes a '+ Create Blog' button. Below this is a table with columns: INDEX, TILE, COVER IMAGE, SHORT DESCRIPTION, STATUS, and ACTION. Two blog entries are visible:

INDEX	TILE	COVER IMAGE	SHORT DESCRIPTION	STATUS	ACTION
1	Website Creation Sites		A web developer is a programmer who specializes in, or is specifically engaged in,	Inactive	Active
2	apps development		aplicaciones for use on mobile devices such as tablets, smartphones and smart relojes	Inactive	Active

At the bottom of the page, there is a footer with the text: © 2021 It Agency Support.

TECH88 QUALITY PARTS FOR QUALITY SERVICES

Search

admin

### category Form













+ category List

category Name

Select Position

Portfolio

Save New Category

INDEX	CATEGORY NAME	POSITION	STATUS	ACTION
1	apps	portfolio	Active	Inactive  
2	web	portfolio	Inactive	Active  
3	SEO	portfolio	Inactive	Active  
4	Gaming	portfolio	Inactive	Active  
5	Website	blog	Inactive	Active  
6	Apps Development	blog	Inactive	Active  
Index	Category Name	Position	Status	Action

© 2021 IT Agency Support





TECH88 QUALITY PARTS FOR QUALITY SERVICES

Search

admin

### Service List

+ Service List

INDEX	SERVICE NAME	IMAGE	DESCRIPTION
1	Web Development		It's Never Too Late to Learn a New Skill! Learn to Code and Join Our 45+ Million Users. Enjoy Extra Quizzes & Projects and Exclusive C
2	Apps Development		Build Native Apps on iOS and Android From a Single Codebase. Download and Build Today.
3	Digital Marketing		Experience modern teaching methods and lectures that will equip you with business skills.
4	Game Development		Game Development is the art of creating games and describes the design, development and release of a game
Index	Service Name	Image	Description

TECH88 QUALITY PARTS FOR QUALITY SERVICES

Search

admin

Default Home Dashboards Default New

TOTAL SALES

**\$350,897**

Last 30 days

NEW USERS

**2,356**

↑ 3.48% Since last month

SALES

**924**

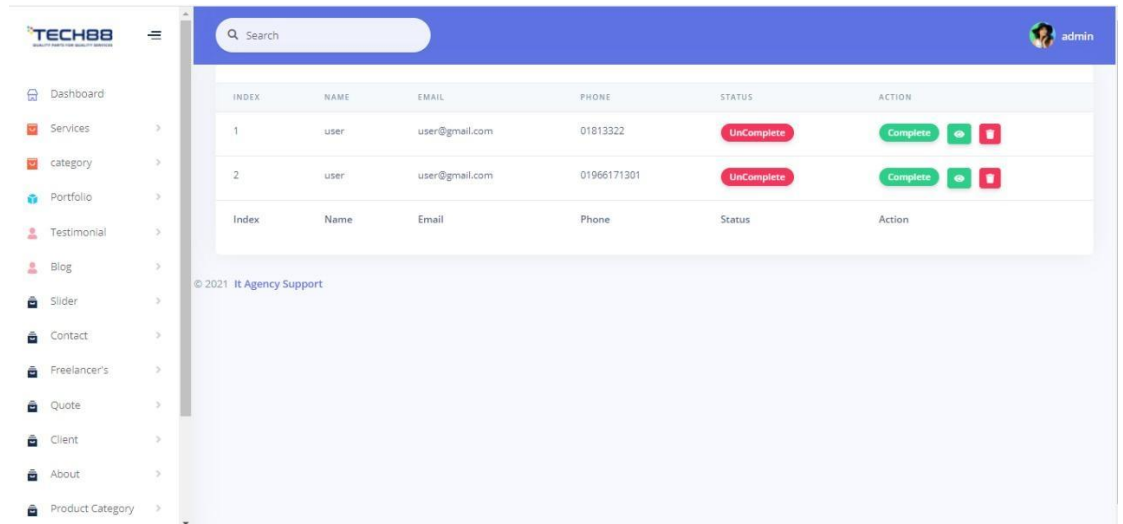
↑ 3.48% Since last month

PERFORMANCE

**49,65%**

↑ 3.48% Since last month

© 2021 IT Agency Support



## CHAPTER 7: PROJECT SUMMARY

### 7.1 GitHub Link

<https://github.com/Bokhtiar/It-Agency-Web-Application-using-laravel-modules-packge-in-repository-petten--Full-Project-2021>

### 7.2 Critical Evolution

There was some critical situation for the evolution of this project main was the database connection and then the responsiveness of all the pages and lastly the checkout process.

### 7.3 Limitations

This project will hold a limited number of information and data regarding software, customers and dates when customers buy the software. If the customer does not have a registration, then they never issue the software to that particular one. Create distinct product users based on their roles and permissions. Authenticate users at their login. Provide the list of software the users will buy.

### 7.4 Future Scope

In the future join the employee panel, I will add the role permission admin can set the menu employee which menu gets. I will add apps to this website that already create an API so in the future I can do it.

