

Premier University
Department of Computer Science & Engineering



Final Year Project Report
On
Track My Subscription

Submitted by

Mohammed Arafat Ali

ID: 1603110201187

Abdul Karim

ID: 1603110201188

Suchitra Das

ID: 1603110201209

In partial fulfillment of the degree of

Bachelor of Science in Computer Science & Engineering

Under the Guidance of

Mr. Anik Sen

Assistant Professor

Department of Computer Science & Engineering

Premier University, Chittagong

April 2022

Declaration

In partial fulfillment of the requirements for the degree of Bachelor of Science, the project report is submitted to Department of Computer Science and Engineering, Premier University. We hereby declare that the work provided in this paper was completed under the supervision of Mr. Anik Sen, Assistant Professor, Department of Computer Science & Engineering, Premier University and that materials from other researchers' work are mentioned as references. This project report has never been submitted before, in whole or in part, for any degree.

.....

Mohammed Arafat Ali

ID: 1603110201187

.....

Abdul Karim

ID : 1603110201188

.....

Suchitra Das

ID : 1603110201209



CERTIFICATE OF APPROVAL

This project title “Track My Subscription” submitted by Mohammad Arafat Ali (ID: 1603110201187), Abdul Karim (ID: 1603110201188) and Suchitra Das (ID: 1603110201209) has been accepted as satisfactory in fulfillment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering (CSE) as B.Sc. Engineering to be awarded by Premier University, Chittagong.

.....
Prof. Dr. Taufique Sayeed

Chairman

Department of Computer Science and Engineering
Premier University, Chittagong

.....
Mr. Anik Sen

Assistant Professor

Department of Computer Science and Engineering
Premier University Chittagong

ACKNOWLEDGEMENT

First and foremost, when presenting this project, we express our heartfelt gratitude to Almighty for his grace and blessings, which enabled us to successfully finish this project work.

We'd like to take this occasion to thank our esteemed supervisor, Mr. Anik Sen Sir, Assistant Professor, Department of Computer Science & Engineering, Premier University, Chattogram. This project executed successfully under his supervision. His continual assistance and desire to help grasp this project and its manifestation in great depth of knowledge aided us in completing the assignment. Throughout the advancement of our work, his unwavering support, cooperation, inspiration, amazing direction, constructive criticism, and fantastic suggestions helped us make this work considerably better. We are cordially grateful to him and we really feel blessed to have him as our supervisor.

We are also grateful to Prof. Dr. Taufique Sayeed, Chairman of Premier University's Department of Computer Science and Engineering, for his unwavering encouragement and support. Finally, we'd want to express our gratitude to our parents, whose blessings and appreciations aided us in finally finishing this project. We'd also like to convey our gratitude and respect to our parents for their unwavering support, without which this work would not have been possible.

Last but not least, we'd like to express our gratitude to our parents. We appreciate our family's love, as well as their moral and financial support. We owe a debt of gratitude to our friends, classmates, seniors, juniors, and lab assistants, who have provided us with unwavering support and assistance in every way. We are extremely fortunate to have such a kind person as them by our side.

ABSTRACT

“Track My Subscriptions” is like having users own virtual assistant keeping watch on all users recurring payments of purchased subscriptions. Subscribers may use “Track My Subscription” to keep track of their subscription activity and receive notifications and reminders, as well as manage their payment methods, currency rates, and more. Automated reminders and notifications are sent by “Track My Subscription” so that users aren't billed for subscriptions users don't want. It helps users keep track of their monthly spending and see how much money they have remained for the rest of the month. Filtering by folder or payment type might also help consumers quickly find the information they are looking for. Therefore, we have built an interactive and effective website called “Track My Subscriptions”.

Table of Contents

Chapter 1	1
Introduction	1
1.1 Introduction	1
1.2 Objectives	1
1.3 Motivation	1
1.4 Summary	2
Chapter 2	3
Literature Review	3
2.1 Introduction	3
2.2 Review of Existing Systems	3
2.3 Comparison of Our System	4
2.4 Necessity of Methodology	4
2.5 Software Development Life Cycle (SDLC)	4
2.5.1 Feasibility Study	5
2.5.2 Requirement Gathering and Analysis	5
2.5.3 System Specification	5
2.5.4 System Design	5
2.5.5 Program Design and Coding	5
2.5.6 Testing	5
2.6 Software Process Model	6
2.7 Summary	6
Chapter 3	7
Software Requirement Specification	7
3.1 Introduction	7
3.2 Requirement Engineering	7
3.2.1 Requirement Analysis	7
3.2.2 Requirement Specification	9
3.2.3 Requirement Validation	10

3.3 Specific Requirements	11
3.3.1 User Interface	11
3.3.2 Hardware Interface	11
3.3.3 Software Interface	11
3.4 Software Requirement Specification	12
3.4.1 User Requirements	12
3.4.2 System Requirements	12
3.4.3 Software Requirements	12
3.5 Summary	12
Chapter 4	13
Normalization and Data Dictionary	13
4.1 Introduction	13
4.2 Database	13
4.3 Database Management System (DBMS)	13
4.4 Database Normalization	13
4.5 Normalization Rule	13
4.5.1 First Normal Form (1NF)	14
4.5.2 Second Normal Form (2NF)	14
4.5.3 Third Normal Form (3NF)	14
4.5.4 Boyce-Codd Normal Form (BCNF)	15
4.5.5 Fourth Normal Form (4NF)	15
4.6 Data Dictionary	15
4.7 Summary	26
Chapter 5	27
Modeling	27
5.1 Introduction	27
5.2 Use Case Diagram	27
5.3 Activity Diagram	29

5.3.1 Activity Diagrams for User	29
5.3.2 Activity Diagrams of Admin	32
5.4 ER Diagram	36
5.5 Summary	36
Chapter 6	37
Software Implementation	37
6.1 Introduction	37
6.2 Software Implementation	37
6.3 Web Server	39
6.4 Text Editor	40
6.5 Software	40
6.6 Snapshots	41
6.6.1 Product Add	41
6.6.2 Add New Deal Type	41
6.6.3 Subscription Add	42
6.6.4 Add Subscription by User	43
6.6.5 Deal Platform	44
6.6.6 Deal Type	45
6.6.7 Product Category	45
6.6.8 Add New Product Founder	46
6.6.9 User Register	47
6.6.10 User Log In	47
6.6.11 User Dashboard	48
6.6.12 All Subscription In User Dashboard	49
6.6.13 Membership Level	49

6.6.14 User Subscription Add	50
6.6.15 User Product Request	51
6.6.16 User Dashboard After Purchase	51
6.6.17 Purchase Subscription Information	52
6.6.18 Subscription Alert	53
6.6.19 Subscription Bulk Add	54
6.7 Summary	55
Chapter 7	56
Discussion & Conclusion	56
7.1 Introduction	56
7.2 Discussion	56
7.3 Limitation of The System	57
7.4 Future Work	57
7.5 Summary	57
Reference	58

Table of Figures

Figure 1: Software Development Life Cycle	4
Figure 2: Requirement Analysis Process Model	8
Figure 3: Requirement Specification Modeling	9
Figure 4: Non-functional Requirement	10
Figure 5: Use Case Diagram	28
Figure 7: Activity Diagram of Profile	29
Figure 8: Activity Diagram of Subscriptions	30
Figure 9: Activity Diagram of Accomplishment	30
Figure 10: Activity Diagram of Search	31
Figure 11: Activity Diagram of Request	31
Figure 12: Activity Diagram of All Users	32
Figure 13: Activity Diagram of Membership Request for Admin	33
Figure 14: Activity Diagram of Admin Product View	33
Figure 15: Activity Diagram of All Categories	34
Figure 16: Activity Diagram of Membership Request for User	35
Figure 17: Activity Diagram of Subscription for Admin And User	35
Figure 18 : ERD	36

Chapter 1

Introduction

1.1 Introduction

The project “Track My Subscription” is a subscription tracking tool offering notifications, reminders, and payment type management. It gives automated reminders and cautions to avoid unwanted subscriptions. This project is to add the subscriptions user has purchased. It also provides a search feature where users can search subscriptions and also see a list of deals. In this project admin can add membership for users and requesting admin for adding deals. This system is providing facilities to change ownership of subscriptions. It'll send users a reminder exactly when user want it. So users can cancel the payment before it hits their credit card. Users can even send a reminder to someone else.

Objectives

- To add the subscriptions user has purchased.
- To provide a search feature so that users can search subscriptions.
- To see a list of deals on this platform.
- To see a list of founders having their own deals on this platform.
- To provide a feature where admin can add membership for users.
- To request admin for adding deals which purchased by users.
- To change ownership of subscriptions.

1.2 Motivation

There are many tracking subscription application for keeping watch on all recurring payments of purchased subscriptions. But the main motive is to build this application “Track My Subscription” which will give the facilities of purchasing subscriptions and changing ownership of purchasing subscriptions.

1.3 Summary

“Track My Subscription” is a subscription monitoring tool featuring notifications, reminders, and payment type control. It delivers automated reminders and warnings to avoid unnecessary subscriptions. It allows users to track spending month-to-month and estimate remaining budget for the current month. Users may create payment folders and view them all in a calendar. Users can also filter their view by folder or payment type to quickly find the information they need.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

A literature review is a narrative, analytical summary depiction of current material pertaining to a given subject or field of study. A literature review is also not a compilation of paraphrased excerpts from other sources. The fundamental goal of writing a literature review is to convey to the reader the type of knowledge and thoughts that have been established on a topic, as well as the strengths and flaws.

2.2 Review of existing systems

There are some Track Subscriptions systems present. These are –

Truebill

Truebill identifies your subscriptions to help you stop paying for things you no longer need. Your concierge is there when you need them to cancel unwanted subscriptions so you don't have to. [1]

Trim

Trim is a financial management tool that enables users to view recurring charges and manage subscriptions. A free account will give you a personal finance dashboard, spending alerts and subscription management help. For additional fees, you may access bill negotiation, debt payoff and other features. One premium feature, the bill negotiation service, charges 33% of your savings. [2]

Appsumo

A simple Notion template for keeping track of all of your subscriptions in one spot. In today's world where almost everything is accessible on a subscription basis. Whether it is a movie, music, apps, etc. Subscription is charged on a monthly or yearly basis. A person subscribes to a minimum of 10 services as per the latest survey. We are finding it more difficult to keep track of all our subscriptions as a customer because we subscribed on different dates and ends on different dates. [3]

2.3 Comparison of our system

The main objective of this project is to add the subscriptions user has purchased. So that all users can purchase subscriptions in the system. Users can search subscriptions. They can also see the list of deals. In our application it has feature of changing ownership. But existing systems don't have this feature.

2.4 Necessity of Methodology

We should follow a method when developing software or a system. A system or software is a complicated human-made item. The entire complicated process is created in a step-by-step technique known as a methodology for that artifact. As a result, it is critical to create a system that is consistent.

To discuss about it, we need to have a basic knowledge of why it is necessary. We are entirely aware of our goal through studying points. It describes how we will progress successfully, and it will present a methodology that will carry out the processes to construct our proposed system. We need a technique to develop our system, and a Software Life Cycle is a good fit at this time.

2.5 Software Development Life Cycle (SDLC)

A software life cycle model (also known as a process model) is a visual depiction of the software development life cycle. A life cycle model is a diagram as shown in figure 2.1 that depicts all of the steps involved in moving a software product through the stages of its life cycle.



Figure 2.1. Software development life cycle.

It also encapsulates the framework in which these techniques will be implemented. SDLC is a

process that gives us a broad overview and guidelines for creating high-quality software. The "System Design Life Cycle" is another name for this procedure. To make the system dependable for the user, we shall follow the SDLC process.

2.5.1 Feasibility Study

It assists in identifying whether software can meet all criteria as intended.

2.5.2 Requirement Gathering and analysis

The requirement collection and analysis phase of the SDLC is the most significant stage in the development of a system. A meeting was conducted to determine the needs. During this phase, a series of generic questions are asked and answered.

2.5.3 System specification

A System Needs Specification (SRD) is a structured collection of data that encapsulates a system's requirements. Our first task in this phase is to create an entity relationship diagram, which depicts the relationship between entities and their attributes.

2.5.4 System Design

There are two methods for obtaining system design: overall design and detailed design. The overall design determines the system's structural shape. A functional design strategy is called a "structure chart." Detailed design is a procedure that the entire program goes through. Our proposed system could benefit from a progressive refinement.

2.5.5 Program design and coding

We followed the SDLC approach to design and code our solution. This approach necessitates only 10% of the time spent on the job. We used PHP, Laravel, HTML, JavaScript, and CSS as web-based programming languages.

2.5.6. Testing

One of the most important stages in the development of a system is the testing phase. We finished our coding for design purposes in our system. Then we double-checked all of our dynamic fields. As a result, we looked for flaws. There are two ways to implement our proposed system:

- White Box Testing
- Black Box Testing

2.6 Software Process Model

A graphical representation of an object is provided by a software process model. It depicts a software system's activity. We followed the Agile model while designing the program.

Following are the Agile manifesto principles -

- **Individuals and interactions:** We work both independently and together on our project. Individuals and interactions are valued in the Agile methodology.
- **Working software:** To understand public demand, we constructed our system step by step and tested it with our supervisor. Because the greatest way to develop is with a working demo.
- **Customer collaboration:** Our project is entirely centered on the needs of our clients. We can't get the public demand and necessity without consumer engagement. As a result, constant client connection is essential.
- **Responding to change:** We can make them rapidly and continue to develop them using the agile model.

Advantages of Agile Model:

- Customer satisfaction is achieved by the constant distribution of software.
- Customer, testers, and developers communicate on a regular basis.
- During the development phase, requirements can be updated.

2.7 Summary

The existing systems and the Software Development Life Cycle are discussed in this chapter (SDLC).

Chapter 3

SOFTWARE REQUIREMENT SPECIFICATION

3.1 Introduction

A software requirement specification (SRS) is a process that consists of a document that describes how the system should function. Normally shut off near the end of the requirements engineering cycle. To create the system, the functional and non-functional requirements are checked here.

3.2 Requirement Engineering

The user's needs are known to as requirements. To convert these requirements into programming, we must first identify the most important demands and desires of clients. To do so, we must analyze the framework issue area that the customer requires during the precondition building stage. Requirement engineering is the method for achieving this goal.

We examine a set of facts related to the system's aims and objectives during this procedure. It's critical to recognize that a significant portion of requirements engineering is concerned with the stakeholders or parties involved in the process.

.Requirement Engineering process consists of the following main activities:

- Requirement Analysis
- Requirement Specification
- Requirement Validation

3.2.1 Requirement Analysis

Requirement analysis is an important process that gathers all of the necessary information to create a requirement model. To create uniform and unambiguous requirements, we examine, improve, and scrutinize the obtained requirements.

The following methods were used to collect data for analysis:

- Asking peoples

- Observing their operations
- Gather information with some questionnaires

We can represent requirement analysis process as the below as shown in Figure 3.1:

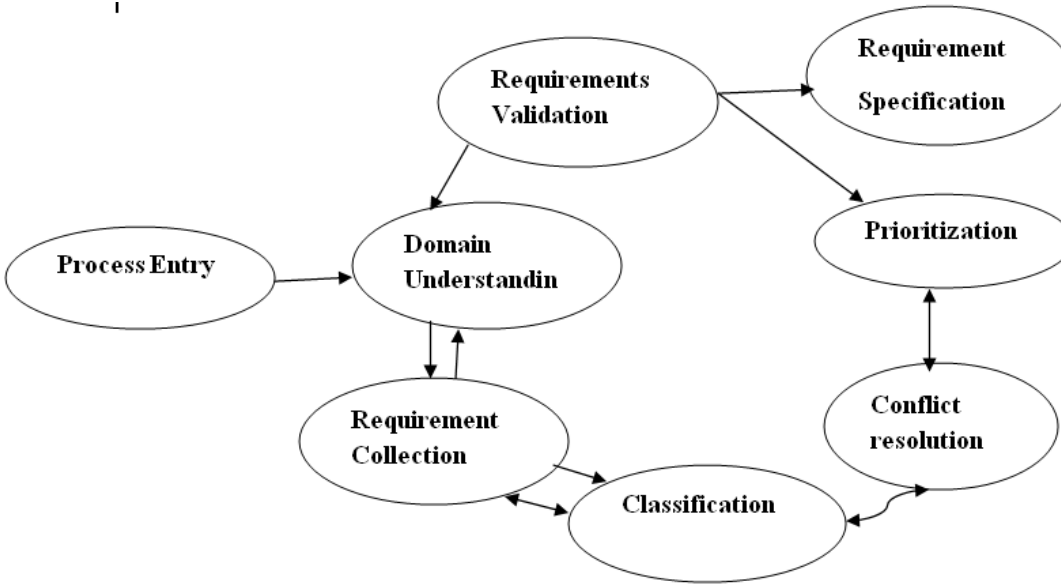


Figure 3.1. Requirement analysis process model.

➤ **Domain Understanding**

This level is all about learning about the sector in which software systems will be used.

➤ **Requirement Collection**

The user's needs and the problem domain concept are covered in this section. Customers are the general public in this initiative.

➤ **Conflict Resolution**

When a system has a large number of subscribers, the requirements will clash. We are acutely aware of conflicts in our system. Our system is mature enough to handle any conflicts that may arise.

➤ Prioritization

Interaction with the user is required for prioritization in order to determine the most important requirements, among others.

➤ Requirement Validation

This phase determines whether they are full and accordance with the subscriber's specific requirements.

➤ Classification

This method divides an unorganized collection of needs into many clusters. We have four categories of user information in our system, which is stored in coherent clusters.

3.2.2 Requirement Specification

The requirement specification is a technique that comprises requirements for constituents, constructions, existence, performance, and so on.

Diagram of Requirement Specification as shown in Fig 3.2:

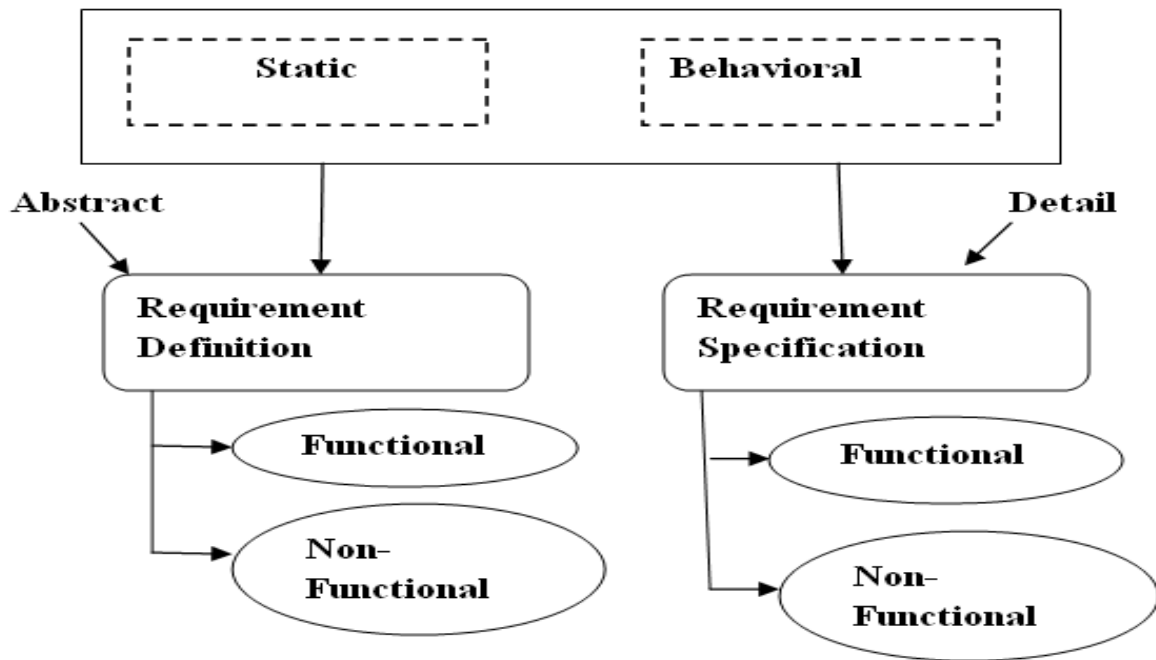


Figure 3.2. Requirement specification modeling.

Non Functional Requirement

A non-functional requirement (NFR) is a set of criteria that can be used to assess a system's performance rather than specific behavior.

Non-functional requirements are shown in figure 3.3:

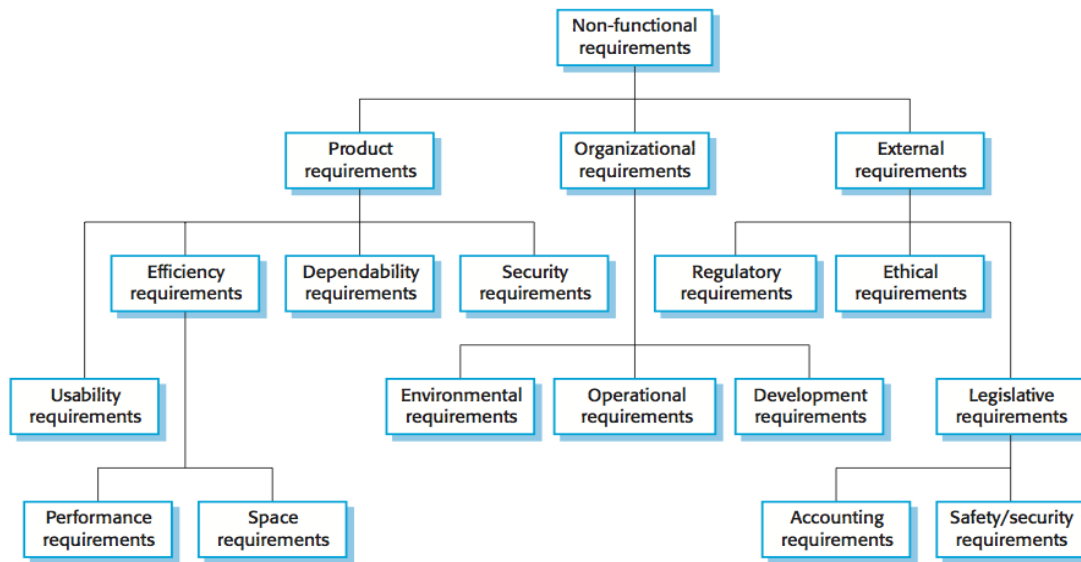


Figure 3.3. Non-functional requirement.

3.2.3 Requirement Validation

Gives the customer an overview of the requirements that have been developed through the system. This is an investigative stage that also identifies the issue. These issues could result in significant rework expenses.

Different types of checks are performed on the requirements:

1. **Validity Check:** Stakeholder-proposed functionality should be in line with what the system needs to do. Our system, for example, has a variety of clients with a variety of requirements.
2. **Consistency Check:** The document's requirements should be separated, which implies they should not clash.

3. **Completeness Check:** This requirement document should include specifications that define all functions and limitations.
4. **Realism Check:** The need must be verified in light of current technology in order to ensure their realization.

3.3 Specific Requirements

3.3.1 User Interface

Because of the system's step-by-step processing, a user-friendly interface was required.

3.3.2 Hardware Interface

Table 3.1. Hardware requirement table.

Operating System	Windows 10, Linux, Unix etc.
Computer/Processor	Intel CORE i5
Memory	At least 512 MB RAM
Web Server	PHP compatible web server
Required memory on the server	2 GB

3.3.3 Software Interface

Table 3.2. Platform and tools table.

Database	MySQL
Web browser	Any PHP supported web browser such as Mozilla Firefox, Google Chrome, etc.
Editor	Visual Studio Code
Programming Language & Technologies	PHP, Wordpress, HTML, CSS, JavaScript, Vue.js, Tailwind, Headless UI.

3.4 Software Requirement Specification

3.4.1 User Requirements

An extension might be anything from a high-level abstract declaration of services to a system limitation.

The requirements are divided into various categories. The following are some examples of required categorizations:

- Diagrams showing the service provided by the system, as well as its operational limits.
- Basically, it was written with customers in mind.

3.4.2 System Requirements

- Structural Documents are documents that lay out the entire description of a system's services.
- Written in the form of a contract between the contractor and the client.

3.4.3 Software Requirements

- A entire set of software data that can be used to create a design or implementation.
- Designed with developers in mind.

3.5 Summary

Software requirement specifications are discussed in this chapter. What software and hardware interfaces are needed to develop this platform, as we mentioned in this chapter.

Chapter 4

Normalization and Data Dictionary

4.1 Introduction

We'll discuss about databases, database management systems (DBMS), database normalization, and data dictionaries in this chapter.

4.2 Database

A database is a collection of data that has been organized in a systematic way. They allow for electronic data storage and manipulation. The main purpose of a database is to store data. A database management system is usually in charge of a database (DBMS). The data, the DBMS, and the applications that go with them are referred to as a database system, which is commonly shortened to just database.

4.3 Database Management System (DBMS)

A database management system (DBMS) is software that allows you to create, define, and manipulate databases. It enables the user to conveniently store, process, and analyze data. A database management system (DBMS) provides us with an interface or tool that allows us to execute various tasks such as building a database, putting data in it, updating data, adding tables in the database, and so on. Some DBMS examples include MySQL, Microsoft Access, SQL Server, FileMaker, Oracle, RDBMS, Clipper, and FoxPro.

4.4 Database Normalization

The process of arranging a database, usually a relational database, is known as database normalization. The goal of this phase is to optimize the storage and retrieval of data in a database. In most cases, normalization means separating existing tables into several ones.

4.5 Normalization Rule

At various phases, normalization rules are used to edit or update bibliographic metadata.

Normalization rules are divided into the following normal forms:

1. First Normal Form (1NF)
2. Second Normal Form (2NF)
3. Third Normal Form (3NF)

4. Boyce-Codd Normal Form (BCNF)
5. Fourth Normal Form (4NF)

4.5.1 First Normal Form (1NF)

The first normal form (1NF) refers to a single table within a relational database system and establishes the fundamental rules for database normalization. Normalization consists of three main steps, each of which builds on the previous. The first normal form is the most basic.

The first normal form states that:

- The table's columns must all be unique
- For each collection of related data, separate tables must be created
- A unique column or concatenated columns called the primary key must be used to identify each table
- There can't be any duplicate rows
- There can't be any duplicate columns
- There are no null values in any row/column intersections
- There are no multi valued fields at row/column intersections

4.5.2 Second Normal Form (2NF)

The second normal form (2NF) is a database normalization normal form. If a relation meets both of the following conditions, it is said to be in the "second normal form."

1. It should be in its original form (First normal Form).
2. It has no non-prime attributes that are functionally dependent on any suitable subset of any relation's candidate key.

4.5.3 Third Normal Form (3NF)

The third normal form, often known as 3NF, is one of several database normalization ideas that also includes the first normal form (1NF) and second normal form (2NF), in which no primary key attribute is transitively dependent on the primary key.

4.5.4 Boyce-Codd Normal Form (BCNF)

The Boyce-Codd Normal Form, also known as 3.5 Normal Form, is an extension of the third normal form. The following two conditions must be met for a table to satisfy the Boyce-Codd Normal Form:

1. It needs to be written in the Third Normal Form
2. A should be a super key for any dependency $A \rightarrow B$

4.5.5 Fourth Normal Form (4NF)

Other than a candidate key, the fourth normal form (4NF) is a level of database normalization in which there are no non-trivial multi valued dependencies. It is based on the Boyce-Codd Normal Form and the first three normal forms (1NF, 2NF, and 3NF) (BCNF). They create a different table in the case of multiple relationships. This protocol is maintained by a database known as Fourth Normal Form.

4.6 Data Dictionary

A data dictionary is a well-structured document in which all data pieces are applicable to the system and have accurate descriptions, allowing users and system analysts to grasp inputs, outputs, and even intermediate calculations with ease.

In wp_deal_platforms table ID is the primary key and name, slug are the index. Other fields are platform_url, platform_logo, created_at, updated_at. This table is used to store data of platforms of Deals on our platform.

Table 4.1. wp_deal_platforms.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
Name	varchar(255)	-	No	None

slug	varchar(255)	-	Yes	Null
platform_url	varchar(255)	-	Yes	Null
platform_logo	varchar(255)	-	Yes	NULL
created_at	timestamp	-	No	Current_timestamp()
updated_at	timestamp	-	Yes	NULL

In wp_deal_types table id is the primary key and name, slug both are Index. Other fields are description, created_at, updated_at. This table is used to store data of types of Deals on our platform.

Table 4.2. wp_deal_types.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
Name	varchar(255)	-	No	None
Slug	varchar(255)	-	Yes	Null
description	varchar(255)	-	Yes	None
created_at	timestamp	-	Yes	NULL
updated_at	timestamp	-	Yes	NULL

In wp_pricing_plans table ID is the primary key and product_id, is foreign key and code, plan_name, deal_price, regular_price are index. Other fields are name, link, created_at, updated_at. This table is used to store data of pricing plans of a deal.

Table 4.3. wp_pricing_plans.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
product_id	bigint(20)	Foreign Key	No	None
code	varchar(255)	-	Yes	NULL
features	varchar(255)	-	No	None
plan_name	varchar(255)	-	No	None
name	varchar(255)	-	Yes	NULL
deal_price	varchar(255)	-	Yes	NULL
screenshot	varchar(255)	-	Yes	NULL
link	varchar(255)	-	Yes	NULL
regular_price	varchar(255)	-	Yes	NULL
created_at	timestamp	-	No	Current_timestamp()
updated_at	timestamp	-	No	NULL

In wp_products, table ID is the primary key and product_founder_id , product_category_id are foreign keys and name, slug are the index. Other fields are description, stacking, max_stack, ltd_base_price ,deal_url, website_url, login_url, logo, plan_name, deal_type, relase_date, end_date, redeem_deadline, fb_group, twitter, youtube, blog, verdict, product_status, featured_image, deal_status, currency, avg_rating, rating_user_count, deal_document, deal_platform_id, created_at, updated_at. This table is the table where we stored our deals as product.

Table 4.4. wp_product.

Field	Type	Index	Null	Default
ID	bigint (20)	Primary Key	No	None
product_founder_id	int (11)	Foreign Key	No	None
Product_category_id	varchar (255)	Foreign Key	No	None
name	varchar (255)	-	No	NULL

slug	varchar (255)	-	Yes	NULL
description	text	-	Yes	NULL
stacking	tinyint (1)	-	Yes	NULL
max_stack	varchar (255)	-	Yes	NULL
ltd_base_price	varchar (255)	-	Yes	NULL
deal_url	varchar (255)	-	Yes	NULL
website_url	varchar (255)	-	Yes	NULL
login_url	varchar (255)	-	Yes	NULL
logo	varchar (255)	-	Yes	NULL
plan_name	varchar (255)	-	Yes	NULL
deal_type	varchar (255)	-	Yes	NULL
release_date	date	-	Yes	NULL

end_date	date	-	Yes	NULL
redeem_deadline	int (11)	-	Yes	NULL
fb_group	varchar (255)	-	Yes	NULL
twitter	varchar (255)	-	Yes	NULL
youtube	varchar (255)	-	Yes	NULL
blog	longtext	-	Yes	NULL
verdict	text	-	Yes	NULL
product_status	varchar (255)	-	Yes	NULL
feature_image	varchar (255)	-	Yes	NULL
deal_status	varchar (255)	-	Yes	NULL
currency	varchar (255)	-	Yes	NULL
avg_rating	float	-	Yes	NULL
rating_user_count	bigint(20)	-	Yes	NULL
deal_document	text	-	Yes	NULL
deal_platform_id	bigint(20)	Foreign Key	Yes	NULL
created_at	timestamp	-	Yes	NULL
updated_at	timestamp	-	Yes	NULL

In wp_product_categories table ID is the primary key and name and slug both are index. Other fields are description, created_at and updated_at. Categories of deals are stored in this table.

Table 4.5. wp_product_categories

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
name	varchar (255)	-	No	None
slug	varchar (255)	-	No	None
description	varchar (255)	-	Yes	NULL
created_at	timestamp	-	Yes	current_timestamp()
updated_at	timestamp	-	Yes	NULL

In wp_product_founders table ID is the primary key. Other fields are avatar, twitter, facebook, website, email, youtube, linkedin, country, bio, created_at and updated_at. Founders info are stored in this table.

Table 4.6.wp_product_founders.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
name	varchar (255)	-	Yes	None
avatar	varchar (255)	-	Yes	NULL

twitter	varchar (255)	-	Yes	NULL
facebook	varchar (255)	-	Yes	NULL
website	varchar (255)	-	Yes	NULL
email	varchar (255)	-	Yes	NULL
youtube	varchar (255)	-	Yes	NULL
linkedin	varchar (255)	-	Yes	NULL
country	varchar (255)	-	Yes	NULL
bio	varchar (255)	-	Yes	NULL
created_at	timestamp	-	No	current_timestamp()
updated_at	timestamp	-	Yes	NULL

In wp_product_rating table ID is the primary key and product_id, user_id are foreign keys. Other fields are review,rating, created_at and udated_at. This table is used to store rating of a deal.

Table 4.7. wp_product_raitng.

Field	Type	Index	Null	Default
-------	------	-------	------	---------

ID	bigint(20)	Primary Key	No	None
product_id	bigint(20)	Foreign Key	No	None
user_id	bigint(20)	Foreign Key	No	None
rating	Float(5.2)	-	Yes	NULL
review	varchar (255)	-	Yes	NULL
created_at	timestamp	-	No	current_timestamp()
updated_at	timestamp	-	No	NULL

In wp_product_with_deal_platforms table, ID is the primary key and product_id, deal_platform_id are the foreign key. Other fields are created_at and updated_at. This table is used to store multiple deals info on single platform.

Table 4.8. wp_product_with_deal_platform.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
product_id	bigint(20)	Foreign Key	No	None
deal_platform_id	bigint(20)	Foreign Key	No	None
created_at	timestamp	-	Yes	NULL
updated_at	timestamp	-	Yes	NULL

In wp_subs table, ID is the primary key and user_id, product_id, pricing_plan_id, subs_tag_id, deal_platform_id, deal_type_id, subs_folder_id, are foreign key . Other fields are migration, name, stacked, price, status, bought_date, refund_date, redeem_date, payment_processor and plan. This table is used to store the data of subscriptions, user has purchased from various deals.

Table 4.9. wp_subs.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
user_id	bigint(20)	Foreign Key	No	None
product_id	bigint(20)	Foreign Key	Yes	NULL
Pricing_plan_id	bigint(20)	Foreign Key	Yes	NULL
subs_tag_id	bigint(20)	Foreign Key	Yes	NULL
deal_platform_id	bigint(20)	Foreign Key	Yes	NULL
deal_type_id	bigint(20)	Foreign Key	Yes	NULL
subs_folder_id	bigint(20)	Foreign Key	Yes	NULL
name	varchar (255)	-	No	NULL
stacked	varchar (255)	-	Yes	NULL
price	varchar (255)	-	Yes	NULL
status	varchar (255)	-	Yes	NULL
bought_date	date	-	Yes	NULL
refund_date	date	-	Yes	NULL

redeem_date	date	-	Yes	NULL
signup_email	varchar (255)	-	Yes	NULL
payment_processor	varchar (255)	-	Yes	NULL
login_id	varchar (255)	-	Yes	NULL
currency	varchar (255)	-	Yes	NULL
invoice	varchar (255)	-	Yes	NULL
cycle	varchar (255)	-	Yes	NULL
folder	varchar (255)	-	Yes	NULL
retail_price	varchar (255)	-	Yes	NULL
listing_price	varchar (255)	-	Yes	NULL
sold_for	varchar (255)	-	Yes	NULL
for_sale	tinyint(1)	-	No	NULL
Notes	varchar (255)	-	Yes	NULL
privacy	varchar (255)	-	Yes	NULL
plan	varchar (255)	-	Yes	NULL
created_at	timestamp	-	No	NULL
updated_at	timestamp	-	Yes	NULL

In wp_subs_folder table, ID is the primary key and user_id is the foreign key. Other fields are name, slug, created_at and updated_at. This table is used to store folder data of subs.

Table 4.10. wp_subs_folder.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
user_id	bigint(20)	Foreign Key	No	None
name	varchar (255)	-	No	None
slug	varchar (255)	-	No	None
created_at	timestamp	-	No	NULL
updated_at	timestamp	-	No	NULL

In wp_subs_reminder table, ID is the primary key and subs_id is the foreign key. Other fields are one_contact, one_note, one_date, two_contact, two_note, two_date, three_contact, three_notes, three_date created_at, updated_at. Here in this table we are storing reminder info of a subscription.

Table 4.11. wp_subs_reminder.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
subs_id	bigint(20)	Foreign Key	No	None
one_contact	varchar (255)	-	Yes	NULL
one_note	varchar (255)	-	Yes	NULL
one_date	date	-	Yes	NULL

two_contact	varchar (255)	-	Yes	NULL
two_note	varchar (255)	-	Yes	NULL
two_date	date	-	Yes	NULL
three_contact	varchar (255)	-	Yes	NULL
three_note	varchar (255)	-	Yes	NULL
three_date	date	-	Yes	NULL
created_at	timestamp	-	Yes	NULL
updated_at	timestamp	-		NULL

In wp_subs_tags table, ID is the primary key and user_id is foreign key. Other fields are name, slug, created_at and updated_at. Subscription tag info is storing in this table.

Table 4.12. wp_subs_tags.

Field	Type	Index	Null	Default
ID	bigint(20)	Primary Key	No	None
user_id	bigint(20)	Foreign Key	No	None
name	varchar (255)	-	No	None
slug	varchar (255)	-	Yes	NULL
created_at	timestamp	-	Yes	NULL
updated_at	timestamp	-	Yes	NULL

4.7 Summary

We discussed about databases, database management systems (DBMS), normalization, and data dictionaries in this chapter.

Chapter 5

Modeling

5.1 Introduction

We'll discuss about our project's use case diagram, activity diagram, and ER diagram in this chapter.

5.2 Use Case Diagram

Use case diagrams are the scenario-based method, which identifies the actions of communication.

Admin

- Admin can see the list of users of this platform. He can approve or disapprove user
- Admin can add and manipulate all the membership functionality. He can approve and disapprove membership update request
- Admin can see, add, update and delete products on this platform
- Admin can see, add, update and delete subscription on this platform
- Admin can see, add, update and delete subscription deal type on this platform
- Admin can view and update his profile
- Admin can see, add, update and delete founders on this platform
- Admin can see, add, update and delete rating
- Admin can see list of membership on this platform
- Admin can see users request of adding products
- Admin can track ownership change of subscription

User

- Users can register and login on this platform
- Users can view and update his profile
- Users can see, update and delete subscription from his profile
- User can buy subscription on this platform
- User can see list of subscription on this platform
- User can search subscription using this platform
- User can request to update the membership to premium after completing the subscription

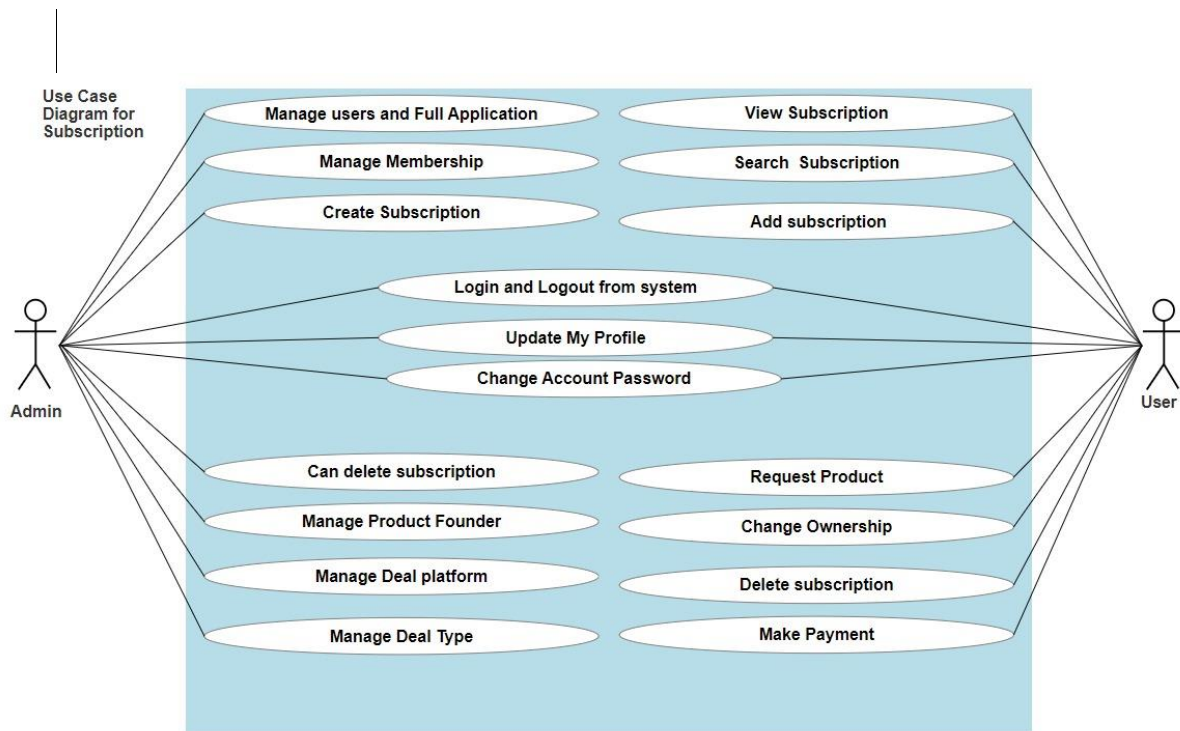


Figure 5.1. Use case diagram of user and admin.

5.3 Activity Diagram

5.3.1 Activity Diagrams for User

On this platform, there are three kinds of users. They are user, Premium, & Admin. All of the users will get a separate profile section. In this profile section, they can see their subscription information, expire date and accomplishment information as well. They will also get search functionality.

Profile

In this platform, every user will get an individual profile section. He can see his profile information and can update information as well like country, region, mobile number as shown in figure 5.2 :

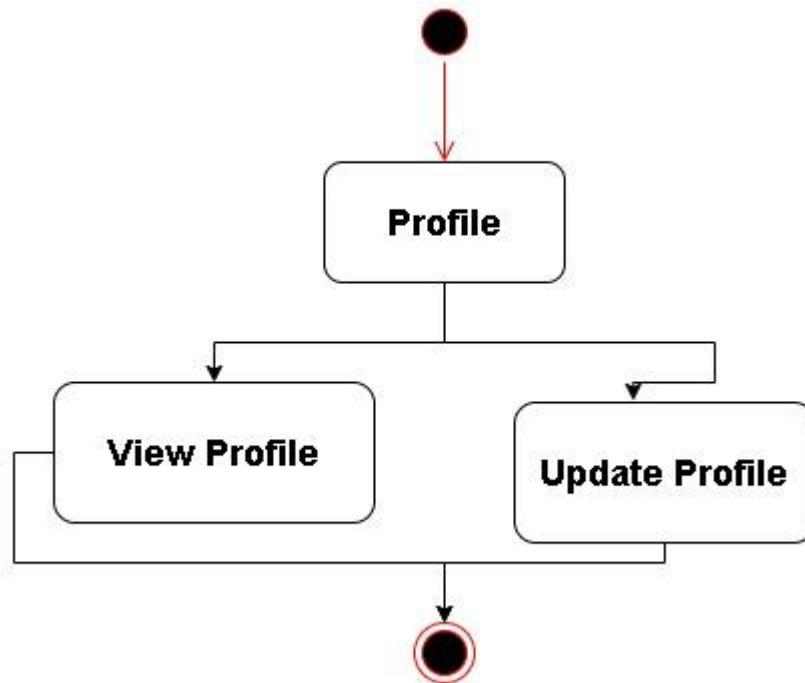


Figure 5.2. Activity diagram of profile.

Subscription

Every user can see his subscription information in his profile. Users can add new subscription information. Users can update or delete their subscription information as well as shown in figure 5.3 :

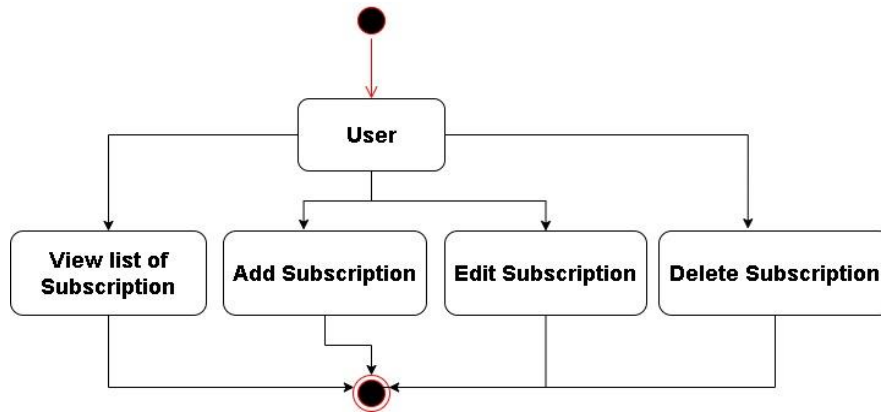


Figure 5.3. Activity diagram of subscription.

Accomplishment

Every user can see his accomplishment information in his profile. Users can add new subscription accomplishment information. User can update or delete his accomplishment information as well as shown in figure 5.4 :

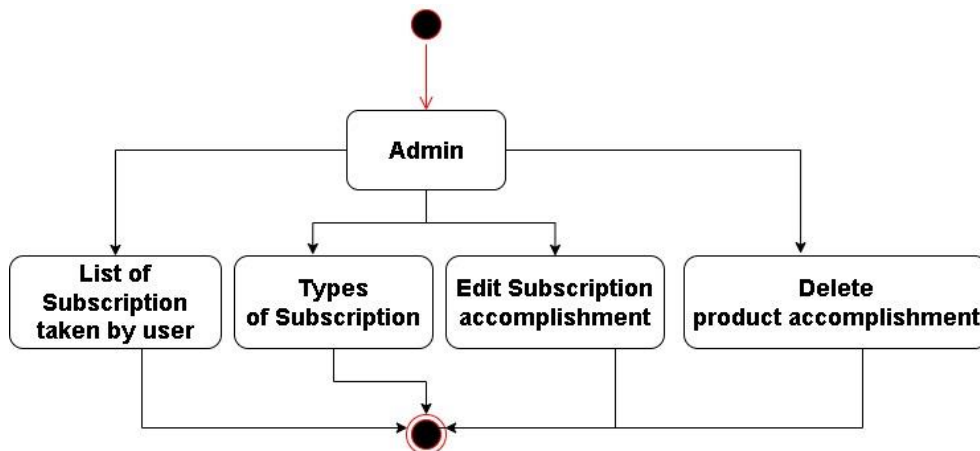


Figure 5.4. Activity diagram of accomplishment.

Search

Every user can search for subscription. They can also filter the search result as well as shown in figure 5.5:

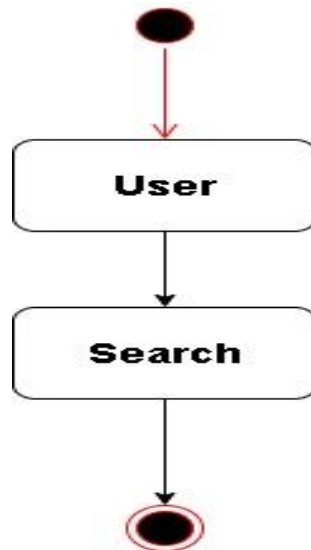


Figure 5.5. Activity diagram of search.

User Product Request

Every user can Request for subscription as shown in figure 5.6:

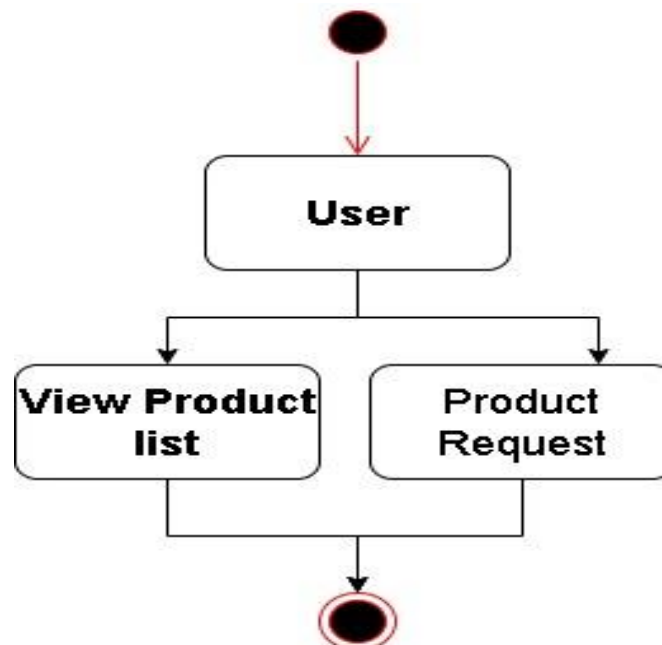


Figure 5.6. Activity diagram of user product request.

5.3.2 Activity Diagrams of Admin

All Admin

Admin can see the list of users in our platform. He can approve or disapprove users by checking their information as shown in 5.7:

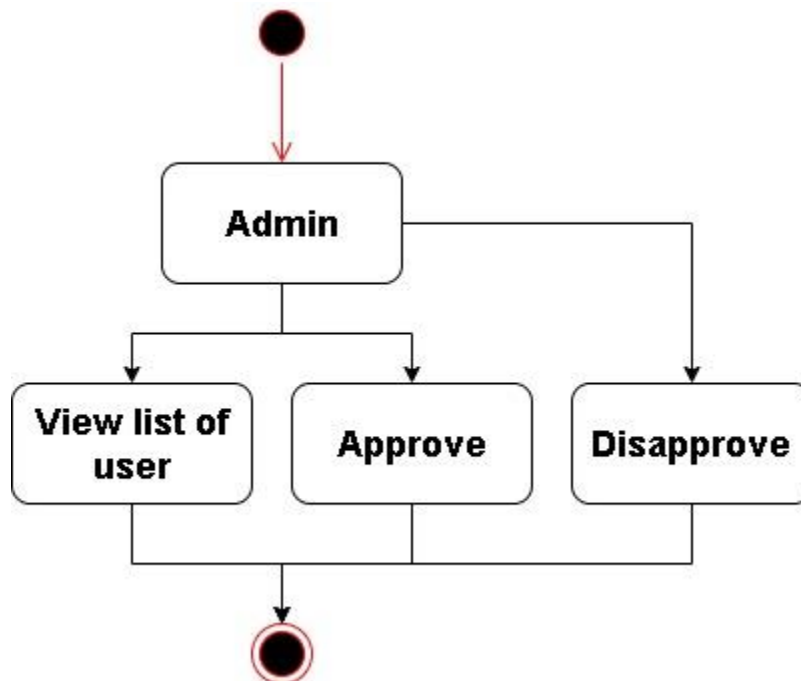


Figure 5.7. Activity diagram of all users.

Membership Request (For admin)

After taking subscription, users can update their membership to Premium by providing necessary information. Again, a premium can update his membership to platinum by providing necessary information. Admin can see all the requests of the users and after checking he can approve membership update requests as shown in 5.8:

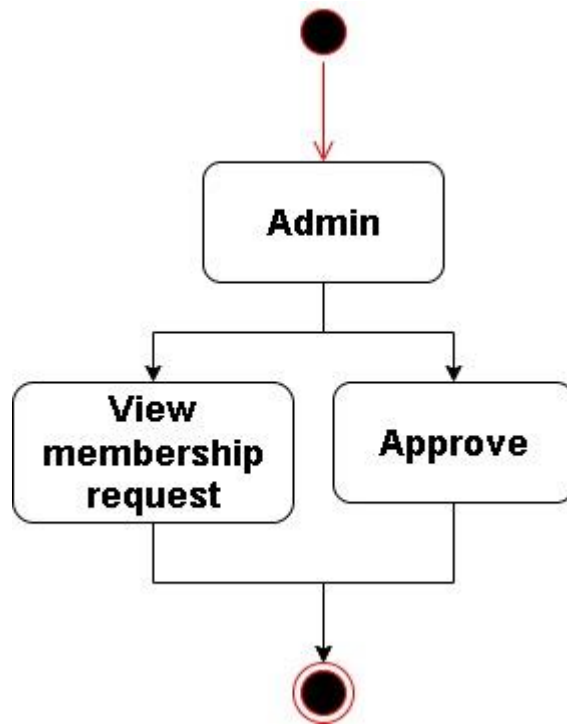


Figure 5.8. Activity diagram of membership request for Admin.

Admin Product View

After Adding products, admin can see all the product list. Admin can edit and delete all the products as shown in 5.9:

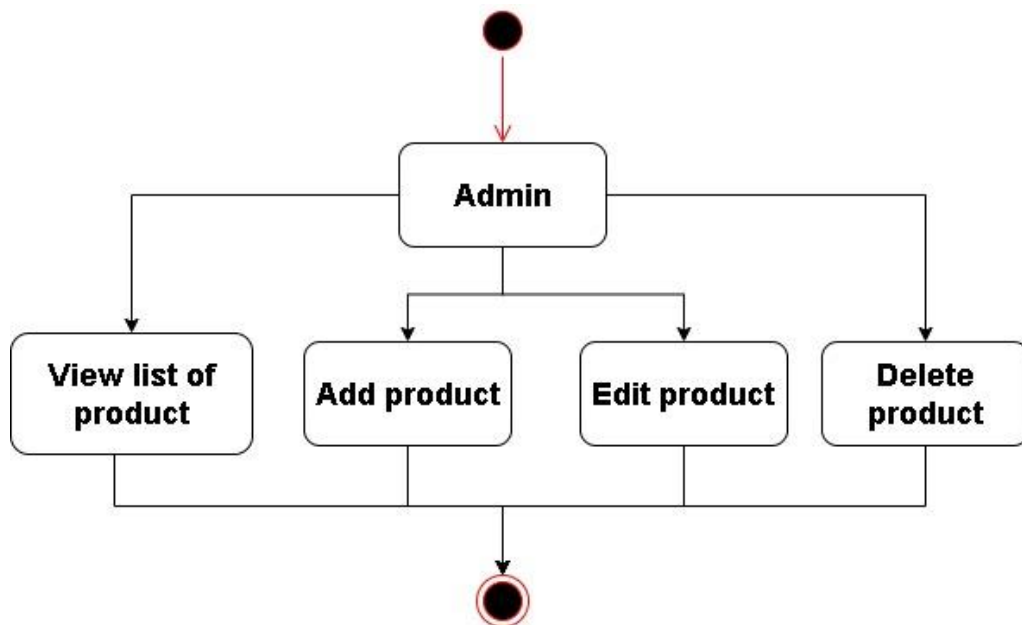


Figure 5.9. Activity diagram of Admin Product view.

Admin Product Category

Admin can see the list of categories in our platform. He can edit or delete category as shown in figure 5.10:

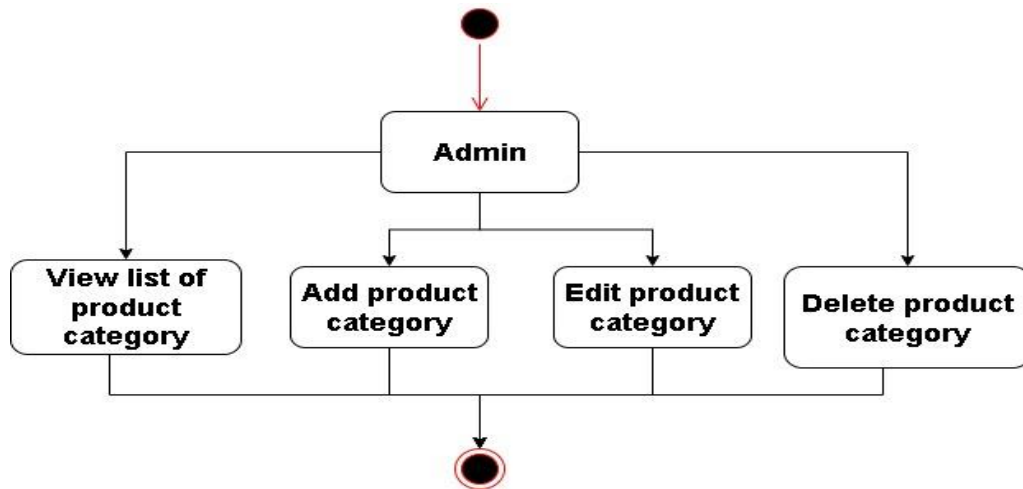


Figure 5.10. Activity diagram of all categories.

Membership Request (For User)

After completing subscription, user can request to update their membership to premium with necessary information. If they provide wrong information accidentally, they can update their request also as shown in figure 5.11:

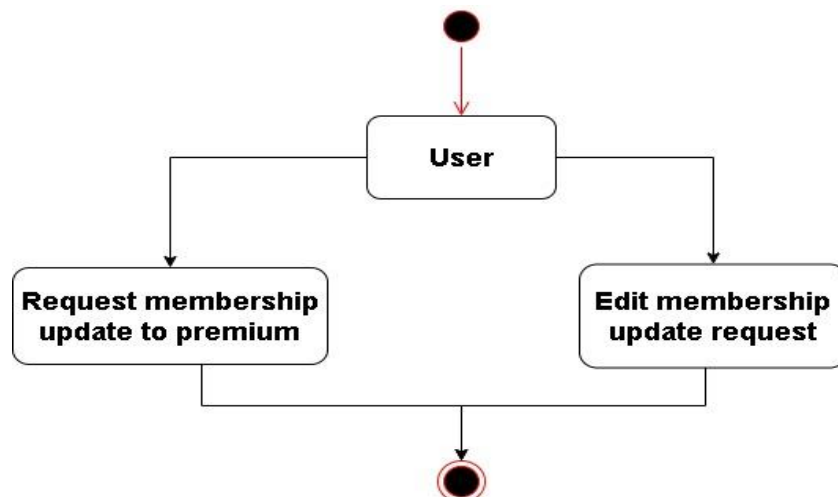


Figure 5.11. Activity diagram of membership request for user.

Admin / User

They can see all the subscriptions that are made by Admin on our platform as shown in figure 5.12:

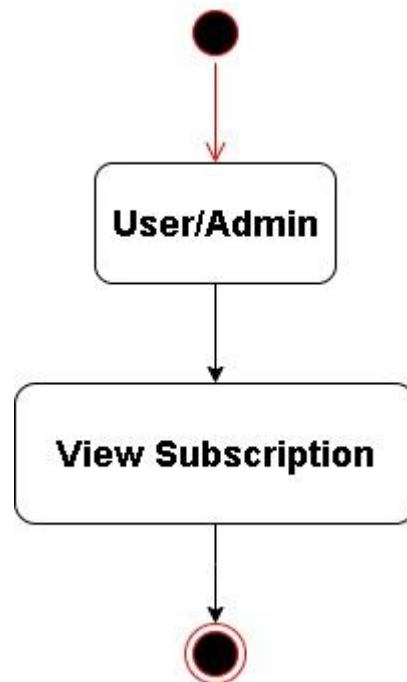


Figure 5.12: Activity diagram of subscription for admin and user.

5.4 ER Diagram

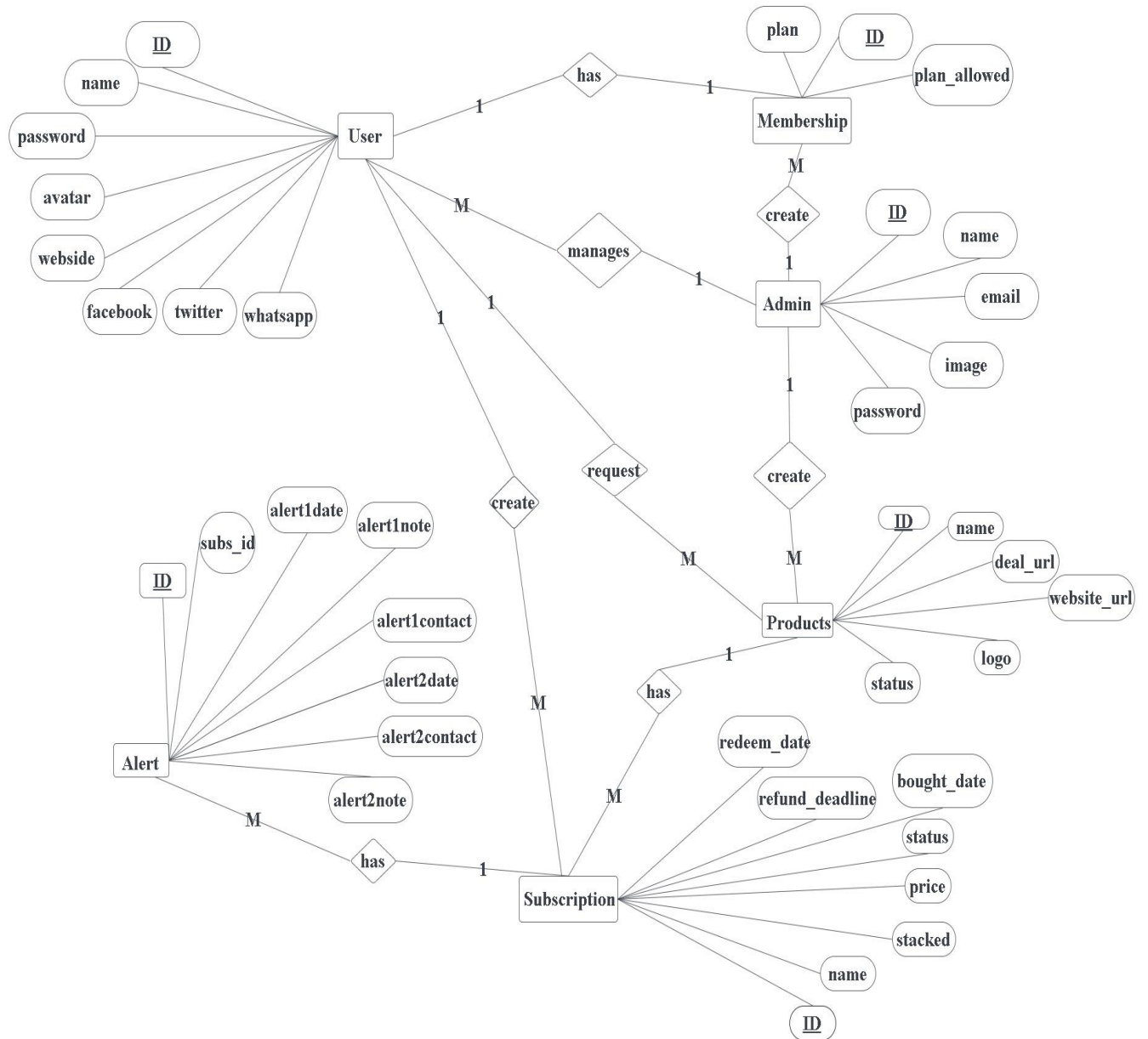


Figure 5.13. ERD.

5.5 Summary

We demonstrated how our product works by utilizing a use case diagram and an activity diagram.

Chapter 6

SOFTWARE IMPLEMENTATION

6.1 Introduction

We searched for all known and available materials that could assist us as we made the decision to conduct the project. Implementing a website is difficult since it entails storing and retrieving a large amount of data. We employed a relational data-based system and an object-oriented database connection to make string and searching more reliable, easier, and faster than ever before. PHP, Wordpress, HTML, CSS, JavaScript, Vue.js, Tailwind, and Headless UI have all made significant contributions to the development of websites. We've employed all of them to ensure the integrity of our project.

6.2 Software Implementation

We implement our software using given below:

1. PHP
2. Wordpress
3. HTML
4. CSS
5. JavaScript
6. Vue.js
7. MySQL
8. Tailwind

PHP

PHP is a server side scripting language. Basically, it is used to develop static websites or dynamic websites or web applications. PHP scripts are illustrated on a server which has PHP installed. This language is very efficient for web development, and can easily be embedded in HTML code. [4]

WordPress

WordPress is free, open source publishing software that can be installed locally on a web server and viewed on a proprietary web site or hosted in the cloud and viewed on the WordPress web site. In 2001, Matt Mullenweg and Mike Little co-founded WordPress, launching it in May 2003 with the help of several core contributing developers. At that time, WordPress was merely a free blogging forum for users. Since then, WordPress has evolved to be a full content management system with widgets, plugins and customizable themes. WordPress is licensed under GPLv2, which lists terms and conditions for copying, modifying and distributing free software.[5]

HTML

HTML stands for Hypertext Markup Language. It allows the user to create and structure sections, paragraphs, heading, links and block quotes for web pages and applications. HTML is not a programming language, meaning it doesn't have the ability to create dynamic functionality. Instead, it makes it possible to organize and format documents, similarly to Microsoft Word. HTML documents are files that end with a .html extension. You can view them using any web browser (such as Google Chrome, Safari or Mozilla Firefox). The browser reads the HTML file and renders its content so that internet users can view it. [6]

CSS

CSS (Cascading Style Sheets) allows you to create great looking web pages, but how does it work under the hood? This article explains what CSS is, with a simple syntax example, and also covers some key terms about the language. CSS can be used for very basic document text styling For example changing the color and size of headings and links. It can be used to create a layout. Besides turning a single column of text into a layout with a main content area and a sidebar for related information. It can even be used for effects such as animation. Have a look at the links in this paragraph for specific examples. [7]

JavaScript

JavaScript is one of the most popular scripting languages. At present, millions of web pages improve their design, validate forms, detect browsers, create cookies and much more. It runs almost all the major browsers such as google chrome, internet explorer firefox, opera.

- It is a lightweight programming language
- JavaScript directly embedded to HTML pages
- Easy to use require no license [8]

Vue.js

Vue (pronounced /vju:/, like view) is a progressive framework for building user interfaces. Unlike other monolithic frameworks, Vue is designed from the ground up to be incrementally adoptable. The core library is focused on the view layer only, and is easy to pick up and integrate with other libraries or existing projects. On the other hand, Vue is also perfectly capable of powering sophisticated Single-Page Applications when used in combination with modern tooling and supporting libraries. [9]

MySQL

MySQL is a freely available open source RElational Database Management System (RDBMS) that uses Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use. [10]

Tailwind

Tailwind CSS is basically a utility-first CSS framework for rapidly building custom user interfaces. It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override. [11]

6.3 Web Server

A web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computer's HTTP clients. [12]

Apache

Apache is the most widely used web server software which is developed and maintained by Apache Software Foundation. It can be customized to meet the needs of many different environments by using extensions and modules. Most of the WordPress hosting providers use Apache for their web server software. [13]

Web Browser

A browser is a software application used to locate, retrieve and display content on the World Wide Web, including web pages, images, video and other files. We used Google Chrome to develop and run the system. Because it is more reliable for databases.[14]

6.4 Text Editor

Visual Studio Code

Visual Studio Code is a lightweight but powerful source code editor. It runs on desktop and is also available for Windows, macOS and Linus. That provides different facilities with build-in support for JavaScript, TypeScript and Node.js etc. [15]

- IntelliSense for programming language
- Efficient command palette
- Highly integrated version control
- Debugging
- Side by side edition on different files

6.5 Software

XAMPP

XAMPP is a free and open source, cross-platform where web server, solution stack package, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP and Perl programming language. It is very easy to use because of its several functionality.

- Installation procedure is very easy
- XAMPP is available for Windows and Linus
- It has several modules
- Offers both a full and a standard version [16]

6.6 Snapshots

6.6.1 Product Add

Admin can easily add product to our system by providing their name, product_founder_id, product_category_id, name, slug and their membership type. After successful form submission, the admin will check the product information. The figure showing the product add option in depicted in Fig.6.1.

The screenshot displays the 'Add new product' form. At the top, there's a notification about a plugin update. The form is divided into several sections: 'Add title' with a text input; 'Add Products' with fields for 'Add Name', 'Add slug', 'Select product category', 'Base Price', 'Website URL', 'Deal Type', 'Redeem deadline', 'YouTube', and 'Product status'; 'Select product founder' with a dropdown; 'Deal URL', 'Logo', 'End date', 'Twitter', and 'Deal Platform'; 'Write Blog' with a rich text editor; 'Enter verdict' with a rich text editor; 'Add Description' with a rich text editor; 'Deal document' with a file upload area; 'Upload featured image' with a file upload area; 'Currency' with a dropdown; 'Rating' with a slider; and 'Enable Stacking?' with a checkbox. The right sidebar contains 'Publish' options (Save Draft, Preview, Publish) and 'Featured image' (Set featured image).

Figure 6.1. Admin Product Add.

6.6.2 Add new Deal Type

To add a new Deal, Admin can easily create new deal type and also edit deal type on our system. Every deal is part of a deal category. The figure showing the add new deal type option in depicted in Fig.6.2.

Add new deal type

Screen Options

There is an update available for: [Meta Box](#).
[Begin updating plugin](#) | [Dismiss this notice](#)

Add title

Deal Type

Name

Slug

Description

WishList Member

Content Protection is disabled for this Post Type.

Enable Content Protection

Publish

Save Draft

Preview

Status: Draft [Edit](#)

Visibility: Public [Edit](#)

Publish immediately [Edit](#)

Publish

Activate Windows

Go to Settings to activate Windows

Figure 6.2. Add new deal type.

6.6.3 Subscription Add

Admin can easily add subscription into our system. All other users are able to see all the subscription. Users are also able to buy or delete their subscription. The figure showing the subscription add option in depicted in Fig.6.3.

Add title

Add Your Subscription

Enter Name

Enter product name

select your product

Select the product

Select your pricing plan

Select the pricing plan

Select your subscription tag

Select the subscription ...

Select your deal platform

Select the deal platform

Select your deal type

Select the deal type

Select your subs folder

Select the subscription ...

Enter Stack

Enter product stack

Enter price

Enter price

Status

Select status

Status

Bought Date

Refund Deadline

Refund Deadline

Redeem Date

Redeem Date

Currency

Select Currency

Payment Processor

Payment Processor

Invoice

+ Add Media

Cycle

Select Cycle

Retail Price

Enter retail price

Author

LoggedIn Admin

Notes

write your notes

Publish

Save Draft

Preview

Status: Draft

Edit

Visibility: Public

Edit

Publish immediately

Edit

Publish

Activate Windows

Go to Settings to activate Windows.

Figure 6.3. Subscription Add.

6.6.4 Add Subscription by User

User may easily add subscription by simply providing product data like product, plan bought, price, status. Notifications are available to users. The figure showing the add subscription by user option in depicted in Fig.6.4.

Add a subscription ×

Get started by filling in the information below to add a new subscription.

Primary details required

Fill all the below details.

Product

Plan you bought

Price

Status

☐ **For Sale**
This subs will shown for sale if checked.

Go to Settings to activate Windows.

Figure 6.4. Subscription Add By User.

6.6.5 Deal Platform

Admin can easily create deal platform and also edit deal platform. All users can see all the deals platform and can buy subscription from that platform. The figure showing the deal platform option in depicted in Fig.6.5.

Deal Platform ^ v ▲

Name

Slug

Platform URL

Platform logo

WishList Member ^ v ▲

Content Protection is disabled for this Post Type.

Publish ^ v ▲

Status: Draft [Edit](#)

Visibility: Public [Edit](#)

[Edit](#)

Figure 6.5. Deal Platform.

6.6.6 Deal Type

Admin can add deal type which subscription user wants to buy. This is a deal category UI where the admin can add a new deal category and can see all the deal types. Every deal is part of a deal category. There are many types of deal categories. The figure showing the deal type option in depicted in Fig.6.6.

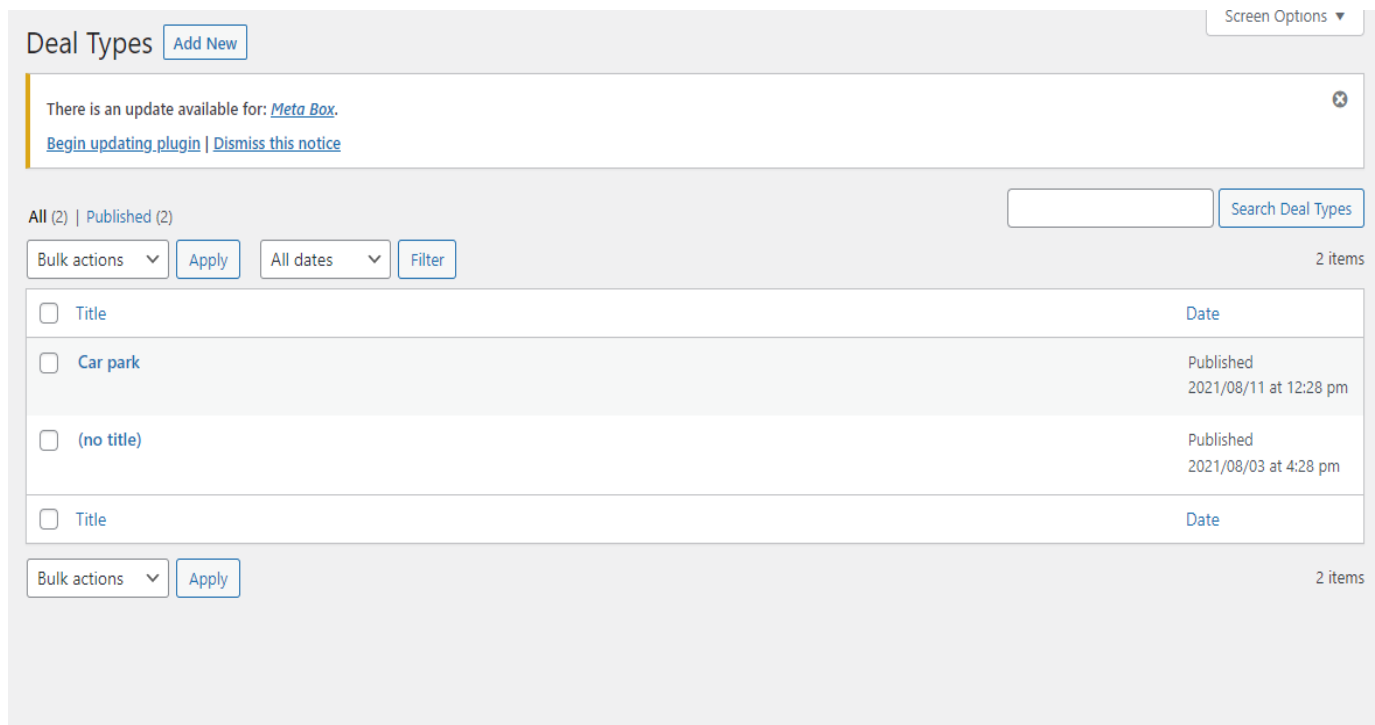


Figure 6.6. Deal Type.

6.6.7 Product Category

Admin can easily add category to our system by providing their name, slug and description. After successful form submission, the admin will check the product category. The figure showing the product category option in depicted in Fig.6.7.

Add new product category

There is an update available for: [Meta Box](#).
[Begin updating plugin](#) | [Dismiss this notice](#)

Add title

Product Category

Name: Enter product category name
 Slug: slug of your product category

Description: Write a description about your product category

WishList Member

Content Protection is disabled for this Post Type. [Enable Content Protection](#)

Publish

Save Draft Preview

Status: Draft [Edit](#)

Visibility: Public [Edit](#)

Publish immediately [Edit](#)

Publish

Activate Windows
Go to Settings to activate Windows.

Figure 6.7. Product Category.

6.6.8 Add New Product Founder

Admin can easily add new product founder to our system by providing their name, email, Facebook, Twitter, country and Avatar. After successful form submission, the admin will check the product founder. This is a deal founder section where the admin can add a new product founder and also see all the founders. So the admin needs to insert the deal founders' information. This is the UI for deals. The figure showing the product founder option is depicted in Fig.6.8.

Add new product founder

There is an update available for: [Meta Box](#).
[Begin updating plugin](#) | [Dismiss this notice](#)

Add title

Product Founders

Name: Enter name
 Email: Founders email
 Facebook: Founders Facebook Account URL
 Twitter: Founders Twitter Account URL
 LinkedIn: Enter LinkedIn URL
 Website: Website URL
 Youtube: Enter youtube
 Country: Enter country
 Bio: Enter bio
 Avatar: [+ Add Media](#)

Publish

Save Draft Preview

Status: Draft [Edit](#)

Visibility: Public [Edit](#)

Publish immediately [Edit](#)

Publish

Activate Windows

Figure 6.8. Product Founder.

6.6.9 User Register

To use the system, users must have an account. To create an account, the users have to put username, email, and password. Username and email must be unique. The confirm password field must be the same as the password to prevent users from accidentally typing a wrong password. The figure showing the registration option in depicted in Fig.6.9.

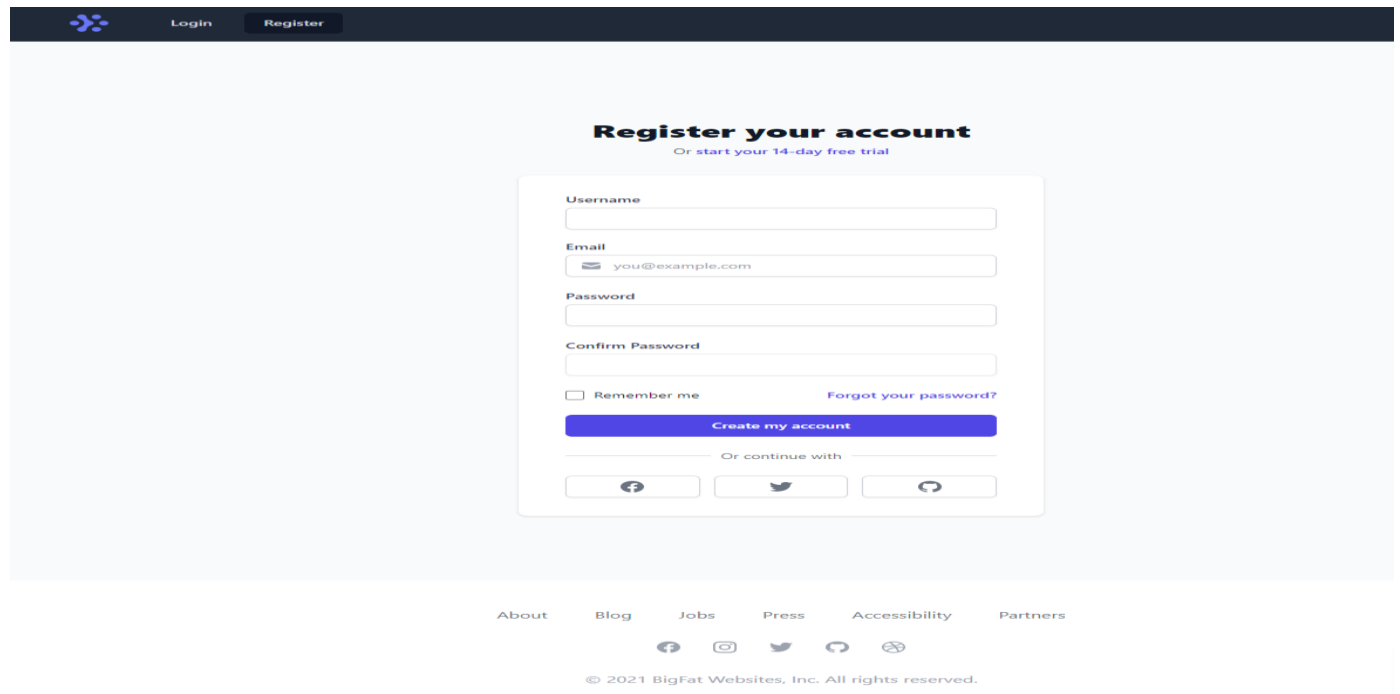
The image shows a web registration form titled "Register your account" with a subtitle "Or start your 14-day free trial". The form is centered on a light gray background. It includes input fields for "Username", "Email" (with a placeholder "you@example.com"), "Password", and "Confirm Password". Below these fields are checkboxes for "Remember me" and a link for "Forgot your password?". A prominent blue button labeled "Create my account" is positioned below the form. Underneath the button, there is a section "Or continue with" followed by three social media icons: Facebook, Twitter, and GitHub. The top of the page features a dark navigation bar with a logo and "Login" and "Register" links. The footer contains a horizontal menu with links: "About", "Blog", "Jobs", "Press", "Accessibility", and "Partners", followed by social media icons and a copyright notice: "© 2021 BigFat Websites, Inc. All rights reserved."

Figure 6.9. User register.

6.6.10 User Login

To use the system, users must have to login into our system. If a user already registered in our system User can login on our system and user can easily sign in. Otherwise user must have created an account. The figure showing the user login option in depicted in Fig.6.10.

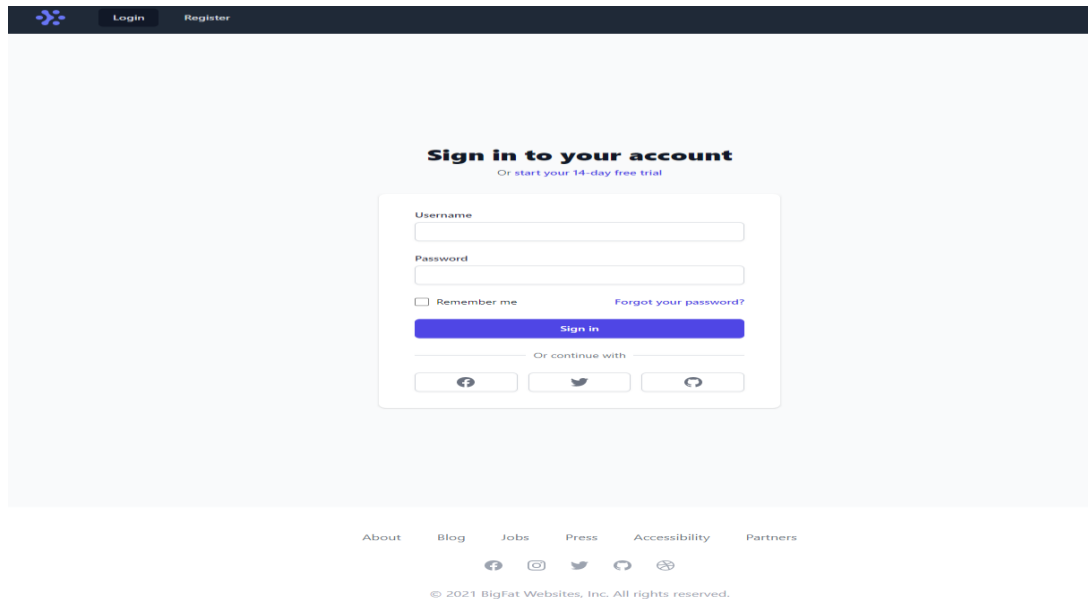


Figure 6.10. User login.

6.6.11 User Dashboard

In user dashboard, user can easily see –

1. Total number of users in our system.
2. Total number of subscriptions.
3. Total number of subscription types.
4. Total number of requests to update membership.
5. Total number of subscriptions sold.
6. Total number of active deals.
7. Total numbers of deals bought.
8. Total amount of money spent.
9. Total numbers of active deals.

The figure showing the user dashboard option in depicted in Fig.6.11.

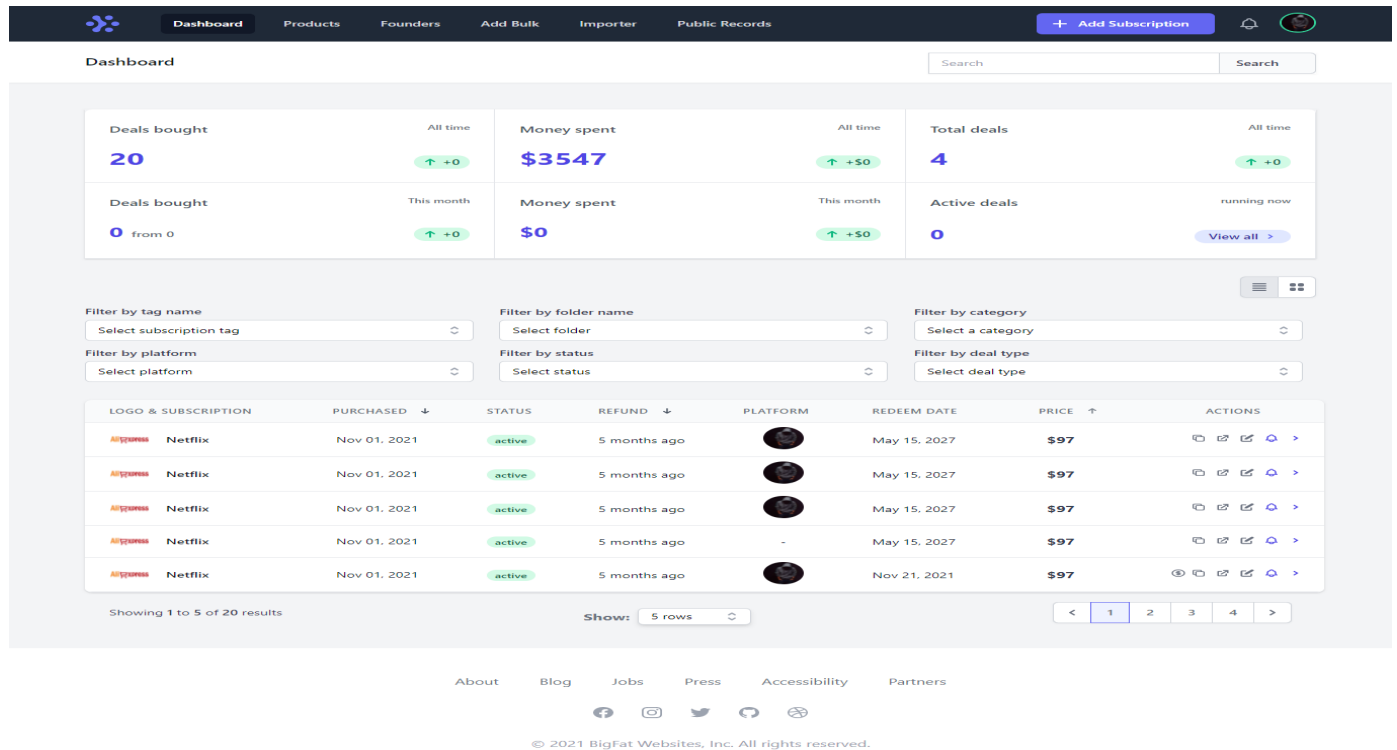


Figure 6.11. User dashboard.

6.6.12 All Subscription in User Dashboard

By clicking into the All-Users tab, User will see a list of subscription with their information. User can filter as well by clicking into the top 3 tab to see folder, category and platform for see specific subscription. The figure showing the all subscription in user dashboard option in depicted in Fig.6.12.

LOGO & SUBSCRIPTION	PURCHASED ↑	STATUS	REFUND ↓	PLATFORM	REDEEM DATE	PRICE ↓
Netflix	Nov 01, 2021	active	5 months ago 🗑		May 15, 2027	\$97
Netflix	Nov 01, 2021	active	5 months ago 🗑		May 15, 2027	\$97
Netflix	Nov 01, 2021	active	5 months ago 🗑		May 15, 2027	\$97
Netflix	Nov 01, 2021	active	5 months ago 🗑	-	May 15, 2027	\$97
Netflix	Nov 01, 2021	active	5 months ago 🗑		Nov 21, 2021	\$97

Showing 1 to 5 of 5 results

Show: 5 rows

< 1 >

Figure 6.12. All subscriptions.

6.6.13 Membership Level

This is a membership user interface where the admin can manage membership requests. There are three types of membership that are available on our platform: basic, pro, and power level. User can get a membership plan on our system. The figure showing the membership level option in depicted in Fig.6.13.

Screen Options ▼

Membership Levels

There is an update available for: [Meta Box](#).

[Begin updating plugin](#) | [Dismiss this notice](#)

Basic level allowed records *

10

Pro level allowed records *

20

Power level allowed records *

30

Basic level allowed access

Allowed ▼

Pro level allowed access

Allowed ▼

Power level allowed access

Allowed ▼

Save Settings

Figure 6.13. Membership Level.

6.6.14 User Subscription Add

Our system allows users to easily add subscriptions. Users must put in the product name, the plan when bought, the sale price, and need to select a status. The figure showing the subscription add option in depicted in Fig.6.14.

REFUND ▼

PLATFORM

5 months ago

M

5 months ago

M

5 months ago

M

5 months ago

-

M

5 months ago

N

Show:

5 rows

g

Jobs

Press

Accessibility

21 BigFat Websites, Inc. All rights reserved.

Add a subscription

Get started by filling in the information below to add a new subscription.

Primary details

required

Fill all the below details.

Product

Find product

Plan you bought

5 stack or Agency plan

Price

\$ 0.00

Status

Select status

☐

For Sale

This subs will shown for sale if checked!

Go to Settings to activate Windows.

Figure 6.14. User Subscription Add.

6.6.15 User Product Request

Users can request products on our system that are not available on the system. The figure showing the user product request option in depicted in Fig.6.15.

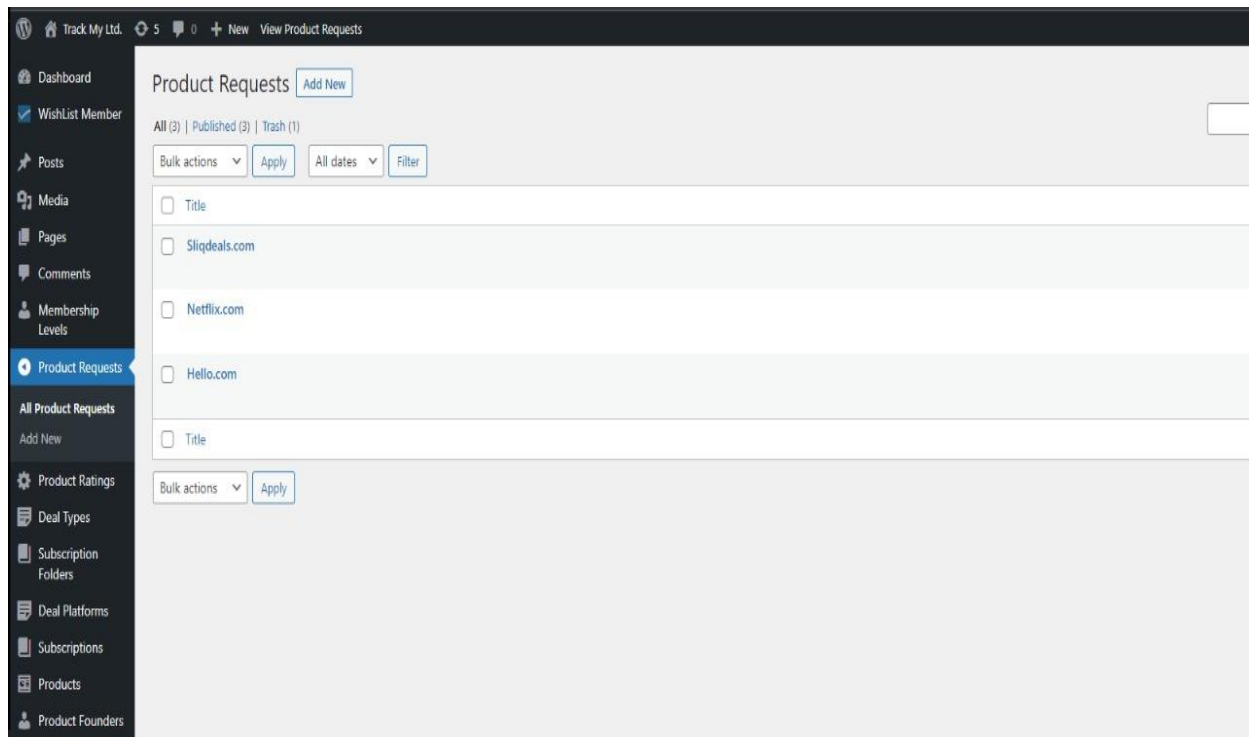


Figure 6.15. Product Request.

6.6.16 User Dashboard after Purchase

User can see updated dashboard on this system when they purchase a subscription. The figure showing the user product request in depicted in Fig.6.16.

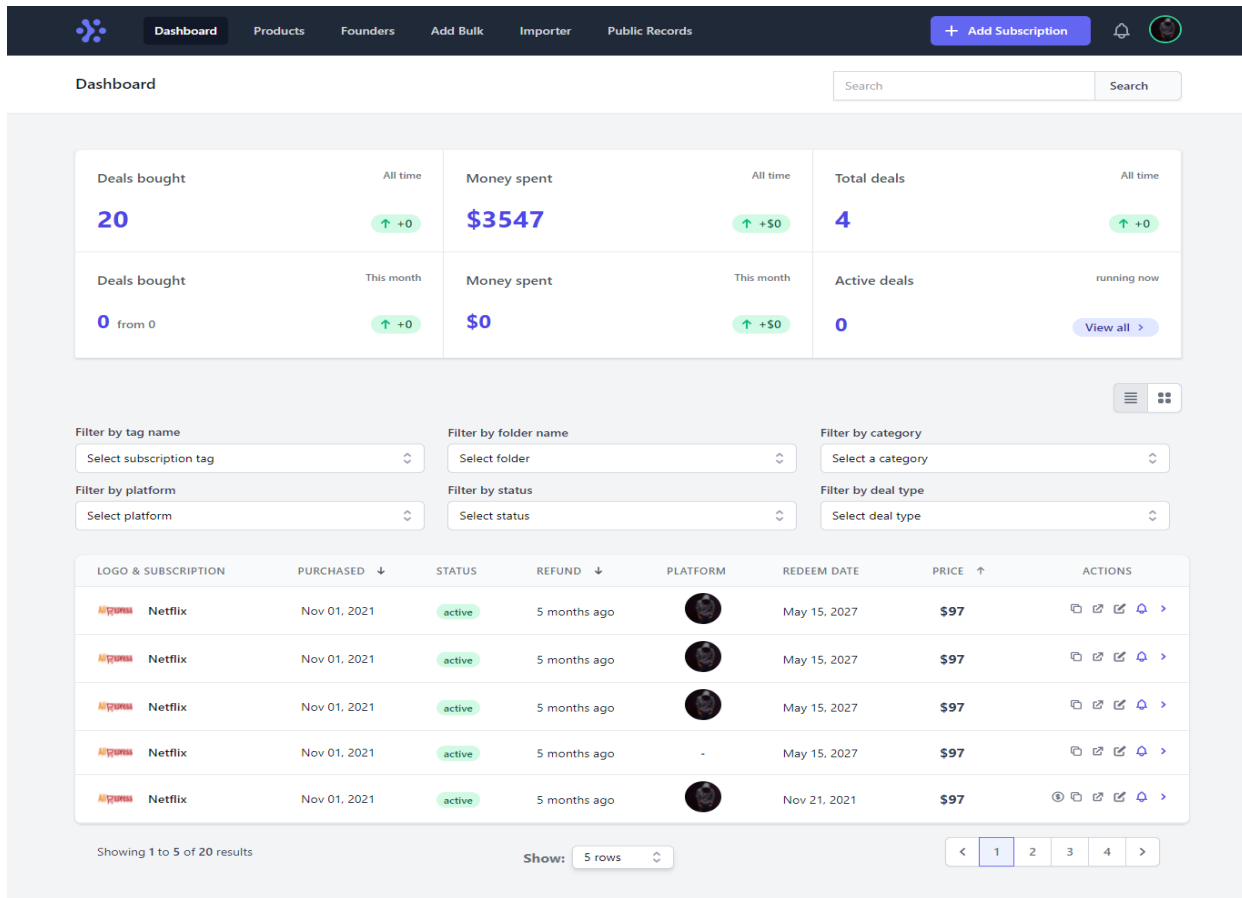


Figure 6.16. Dashboard after purchase.

6.6.17 Purchase Subscription Information

User can see purchase subscription information on this system. User can also see license type, listing price, purchase time, selling price. The figure showing the purchase subscription information option in depicted in Fig.6.17.

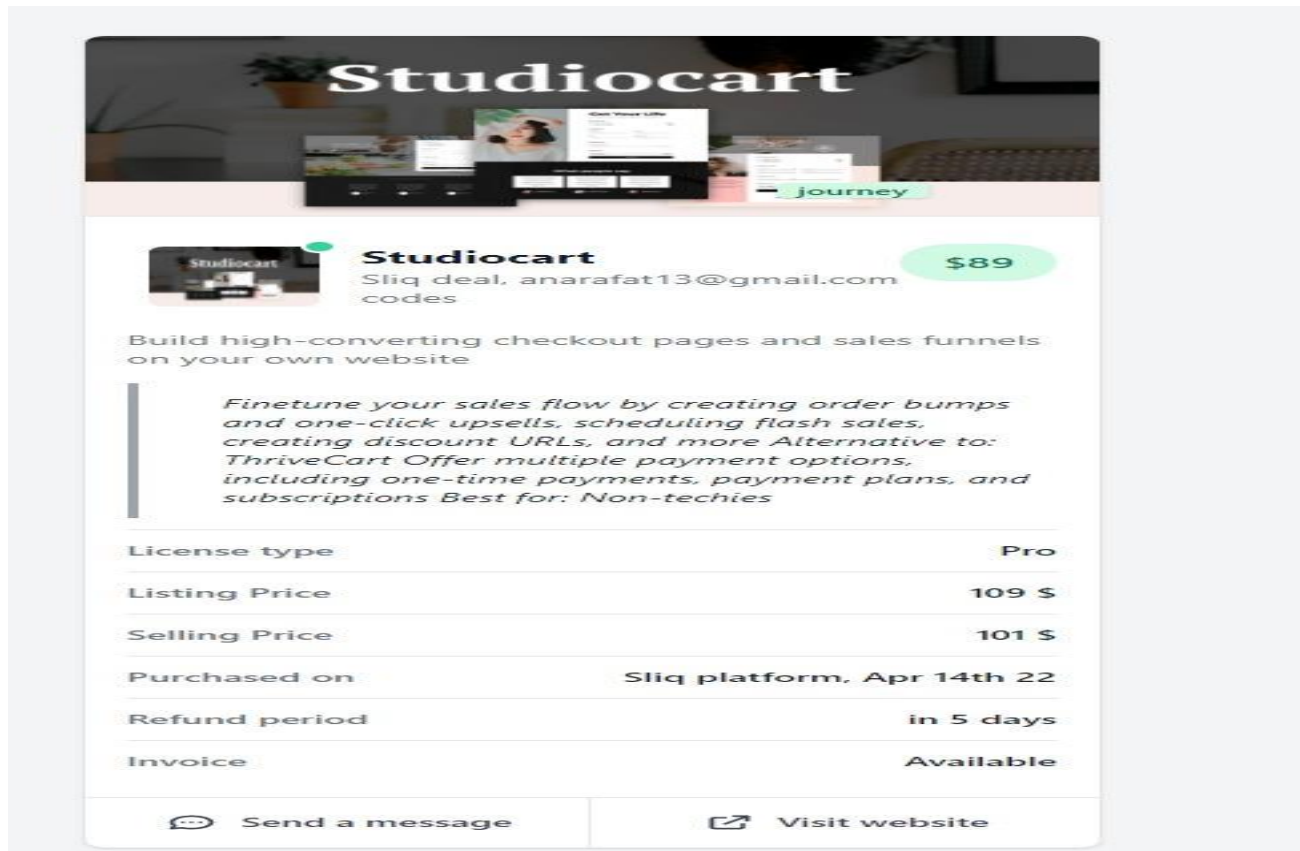


Figure 6.17. Purchase Subscription Information.

6.6.18 Subscription Alert

User will receive notification alert for subscription end period date on this system. Users can set an alert three times whenever they want. They will get notification of their selected time. The figure showing the subscription alert option in depicted in Fig.6.18.

Primary Misc **Alert**

Alerts optional

Add/edit alerts to get email alerts before refund deadlines.

Alert 1 date mm/dd/yyyy

Alert contact 1 you@example.com

Alert 1 Notes

Alert 2 date mm/dd/yyyy

Alert contact 2 you@example.com

Alert 2 Notes

Figure 6.18. Subscription alert.

6.6.19 Subscription Bulk Add

In our system, User can add bulk subscription. User can add subscription by one click. The figure showing the subscription bulk add option in depicted in Fig.6.19.

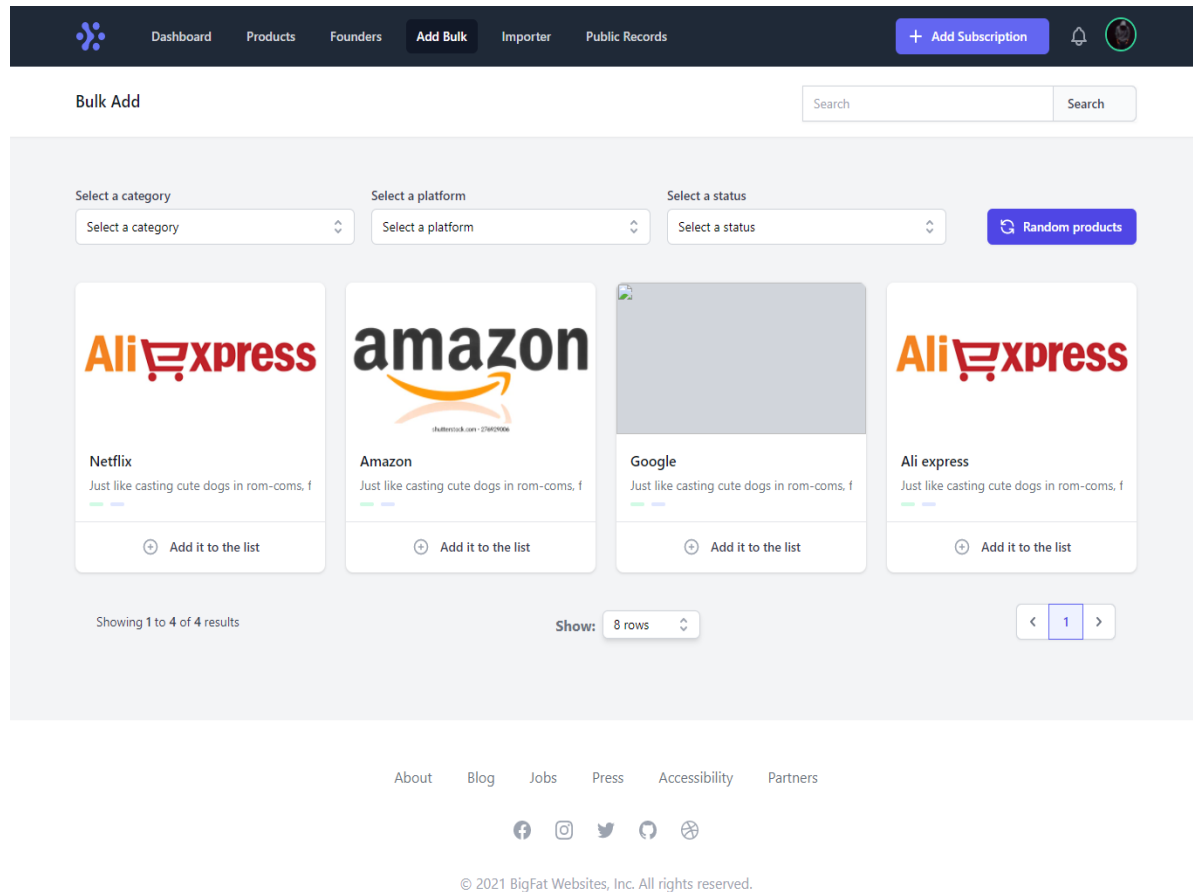


Figure 6.19. Subscription Bulk Add.

6.7 Summary

The implementation of this project is discussed in this chapter. What technology did we utilize to create this website. We also go over each feature with screenshots.

Chapter 7

DISCUSSION AND CONCLUSION

7.1 Introduction

After completing this project, we must state that we did our best to design this system in the most appropriate, useful, and user-friendly manner possible. We achieve the system's expected outcomes. The system is ready to use from anywhere, on any PC, and it can be accessed via the internet from any computer. Unauthorized users are well protected by us. Furthermore, because our system is user-friendly, we can claim that it has the best performance. Despite the fact that our technology has significant limitations, users will benefit from it. We will endeavor to overcome its restrictions in the future.

7.2 Discussion

Our project is broken down into various sections. Introduction, literature study, and methodology, software requirements specification, normalization, and data dictionary, software design, and implementation are the items on the list.

We covered the project's goals, an overview of the current system, and a proposed system in the introduction.

We examined the software development life cycle and which software model is utilized in the system in the literature review and methodology.

We examined the software development life cycle and which software model is utilized in the system in the literature review and methodology.

We can learn about the requirements of a system that are needed by the end-user by looking into software requirements specifications.

We normalized several tables in the normalization and data dictionary section to handle data redundancy by reducing multiple value, partial dependency, and transitive dependency.

We've focused on the ER diagram, activity diagram, and user case diagram in software design.

We described the implementation tools that we used during the construction of our system in Implementation.

7.3 Limitation of the system

Currently our project is for only desktop users. We didn't implement our project as a Single Page Application, so for this reason our page reloads after every action. There is no automatic system to change ownership of a subscription, User have to make it by communicating to each other. There is no panel to add deals by deals founder.

7.4 Future work

1. We will add a mobile app for our project.
2. We will improve the UI.
3. We will make a system to stop recurring payments of any subscription from our system.

7.5 Summary

In this chapter we discussed the limitations of our project and the future plan of our project.

References

- [1] <https://www.truebill.com/feature/manage-subscriptions> [Accessed on: 31 March, 2022].
- [2] <https://money.usnews.com/money/personal-finance/saving-and-budgeting/articles/track-and-manage-subscriptions-with-these-apps> [Accessed on: 31 March, 2022].
- [3] <https://appsumo.com/deals/subscription-tracker-and-management-notion/> [Accessed on: 31 March, 2022].
- [4] <https://www.guru99.com/what-is-php-first-php-program.html> [Accessed on: 1 April, 2022].
- [5] <https://www.techtarget.com/searchcontentmanagement/definition/WordPress> [Accessed on: 1 April, 2022].
- [6] <https://community.codenewbie.org/lanrewaju/100daysofcode-chapter-1-html-and-css-basics-3709#:~:text=What%20is%20HTML%3F,for%20web%20pages%20and%20applications.> [Accessed on: 1 April, 2022].
- [7] https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/What_is_CSS [Accessed on: 1 April, 2022].
- [8] <https://en.wikipedia.org/wiki/JavaScript> [Accessed on: 1 April, 2022].
- [9] <https://v2.vuejs.org/v2/guide/?redirect=true> [Accessed on: 1 April, 2022].
- [10] <https://www.siteground.com/tutorials/php-mysql/mysql/#:~:text=MySQL%20is%20a%20freely%20available,ease%20and%20flexibility%20of%20use> [Accessed on: 1 April, 2022].
- [11] <https://www.geeksforgeeks.org/introduction-to-tailwind-css/#:~:text=Tailwind%20CSS%20is%20basically%20a,have%20to%20fight%20to%20override> [Accessed on: 1 April, 2022].
- [12] <https://www.toppr.com/ask/question/web-server/> [Accessed on: 1 April, 2022].
- [13] <https://www.wpbeginner.com/glossary/apache> [Accessed on: 1 April, 2022].
- [14] <https://www.webopedia.com/definitions/browser/> [Accessed on: 1 April, 2022].
- [15] <https://code.visualstudio.com/docs> [Accessed on: 1 April, 2022].
- [16] <https://en.wikipedia.org/wiki/XAMPP> [Accessed on: 1 April, 2022].