



Daffodil
International
University

Title of the project

**“CodQube: A Premier Digital Marketplace for Developers and
Everyday Users”**

Course: Project / Internship (Phase II)

Course Code: CIS499

Department of Computing and Information System (CIS)

Submitted By:

Al Shahriar Kabir

ID: 193-16-474

Supervised By:

Mr. Md. Nasimul Kader

Assistant Professor

Department of Computing and Information System (CIS)

Submission Date: 29 July, 2024

APPROVAL

This Project titled “CodQube: A Premier Digital Marketplace for Developers and Everyday Users”, Submitted by **Al Shahriar Kabir**, ID No: **193-16-474** to the Department of Computing & Information System, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information System and approved as to its style and contents. The presentation has been held on 29-09-2024.

BOARD OF EXAMINERS



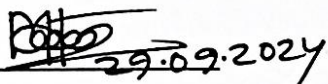
Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Chairman



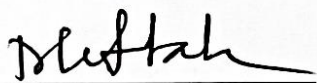
Md. Nasimul Kader
Assistant Professor
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Md. Mehedi Hassan
Lecturer
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Md. Miftah Uddin
Head of SBU, Brain Station 23

External Examiner

Declaration

I hereby declare that; this project has been done by me under supervision of **Md. Nasimul Kader, Associate Professor**, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By



Md. Nasimul Kader
Associate Professor
Department of CIS
Daffodil International University

Submitted By



Name: Al Shahriar Kabir
ID: 193-16-474
Department of CIS
Daffodil International University

Acknowledgement

Firstly, I wish to express my gratitude to Allah Ta'ala for providing me with the courage and discernment to start this journey of self-discovery and development. Without His guidance and support, completing this task would not have been possible.

I give a huge thank you to my academic institution for providing an environment rich in resources and opportunities that fuelled my interest and development in this field. My knowledge and abilities have grown as a result of this amazing experience, especially with regard to new technologies and business procedures.

I wish to offer my deepest gratitude to Mr. Md.Nasimul Kader, for his advice and assistance during this assignment as well. His ever-encouraging words and years of wise wisdom have not only guided me through challenges, but inspired me to achieve the very best I can.

Belief in my abilities and his willingness to share knowledge have helped me grow academically and that is the most important factor behind it. Thanks Kader sir He has helped me handle even the most complex jobs with a poise and grace that I know will help in guiding projects on this scale accurately.

I practically owe him my life for how patient, forgiving and supportive he has been as well. Thanks for guiding me and making this arduous journey to show the perseverance.

Abstract

CodQube is a fresh concept of a digital platform targeting developers for application development and average users for product use. This project is to help people quickly find and use various digital products as enjoyable and easy as possible, regardless of whether you are a developer or an ordinary user.

In CodQube you will find software for programmers, designers and developers as well as numerous educational applications and tools for everyday use. It is a single portal for both, identification of the exact solution and making the purchase and finally using high quality solutions which fit the exact requirement of the customers.

CodQube is different because it has very simple, intuitive interface and it has a recommendation system based on AI. It adapts to the consumer behaviour and acts according to their preferences so it offers perfect products they are looking for making the whole shopping process easier. It also makes the process of purchasing easier and safer since there is extensive information on products that are available for example ratings and reviews. CodQube, which is involved in the enhancement of technologies that go into developing its product, insists on the delivery of only the best digital products and services to satisfy customer feedback.

In other words, CodQube is not just another digital store—or at least that's one of its key components; it is a collection/world of quality software where everyone can find and benefit from apps created by other developers or customers. This project opens the door for a future where identifying the right resources to use is as easy and convenient as turning to a personal advisor.

Table of Contents

Chapter 1: Introduction	1
1.1 Introduction.....	1
1.2 Purpose of Project.....	2
Chapter 2: Initial Study	3
2.1 Project Proposal	3
2.2 Project Scope	4
2.3 Background of the Project	5
2.4 Objectives	6
Chapter 3: Literature Review	7
3.1 Discussion on problem domain based on available solution	7
3.2 Discussion on Problem Solution Based on Available Solutions	7
3.3 Recommended Approach	8
Chapter 4: Methodology	9
4.1 What to Use.....	9
4.1.1 Agile Model:	9
4.1.2 Scrum Model:	10
4.1.3 Rapid Application Development (RAD) Model:	11
4.2 Why to use	12
4.3 Sections of Methodology	13
4.4 Implementation Plan for CodQube	14
Chapter 5: Project Plan	15
5.1 Work Breakdown Structure.....	15
5.2 Resource Allocation	16
5.3 Time Boxing	17
5.4 Gantt Chart.....	18
Chapter 6: Feasibility Study.....	19
6.1 All Possible Types of Feasibility Study	19

6.2 Technical Feasibility	19
6.3 Economic Feasibility	20
6.4 Operational Feasibility	20
6.5 Legal & Compliance Feasibility	20
Chapter 7: Foundation.....	22
7.1 Problem Area Identification.....	22
7.2 Interview	22
7.3 Questionnaire	23
7.4 Requirement Specification.....	23
7.4.1 Functional Requirements	23
7.4.2 Non-Functional Requirements	24
7.5 Technology to be implemented:	26
Chapter 8: Exploration	27
8.1 Activity Diagram.....	27
8.1.1 Activity Diagram Part One (Order process):	27
8.1.2 Activity Diagram Part Two (Upload Product):	28
8.1.3 Activity Diagram Part three (Admin Panel):	29
8.1.4 Activity Diagram Part four (Licence validation):	30
8.1.5 Activity Diagram Part five (Product review):	31
8.2 Use Case of Admin panel:	32
8.3 Use Case of user panel:.....	33
Chapter 9: Exploration	35
9.1 Module of the system:.....	35
9.2 Class diagram of the system:	36
9.3 Sequence diagram of the system:.....	37
9.4 Low-fidelity prototype of the system:.....	38
9.5 High-fidelity prototype of the system:	40
Chapter 10: Development	43
10.1 Folder structure of the system:.....	43

10.2 Core module output sample:	45
10.3 Third-Party Package Module	52
10.4 Prioritization While Developing	54
Chapter 11: Testing	55
11.1 Test Case	55
11.2 Unit Testing:.....	56
11.2.1 Unit Test - 1	56
11.2.2 Unit Test - 2	56
11.2.3 Unit Test - 3	57
11.2.4 Unit Test - 4	57
11.3 Module Testing	58
11.3.1 Module Test-1:.....	58
11.3.2 Module Test- 2:	58
11.3.3 Module Test- 3:	59
11.3.4 Module Test- 4:	59
Chapter 12: Implementation	60
12.1 Training.....	60
12.2 Scaling	60
12.3 Load Balancing	61
Chapter 13: Critical Appraisal and Evaluation	62
13.1 Objectives That Could Be Met	62
13.2 Opportunities for Improvement	62
13.3 Untouched Features	63
Chapter 14: Lessons Learned.....	64
14.1 Pre-Project – Review – Closing.....	64
14.2 Challenges Faced	64
14.3 Solutions Implemented	65
Chapter 15: Lesson Learned	66
15.1 Summary of the project:	66

15.2 Goal of the Project	66
15.3 What I Have Done in Documentation.....	67
15.4 My Experience.....	67
Works Cited	69

List of Figures:

Figure 4-1: Agile Model.....	10
Figure 4-2: Scrum Methodology	11
Figure 4-3: Rapid Application Development (RAD) model.....	12
Figure 5-1: Work Break down Structure	15
Figure 5-2: Resource Allocation	16
Figure 5-3: Time Boxing	17
Figure 5-4: Gantt chart.....	18
Figure 7-1: Model of the technology	26
Figure 8-1: Activity Diagram (order process).....	27
Figure 8-2: Activity Diagram (Upload product)	28
Figure 8-3: Activity Diagram (Admin Panel).....	29
Figure 8-4: Activity Diagram (Licence validation).....	30
Figure 8-5: Activity Diagram (Product review).....	31
Figure 8-6: Use case of Admin panel.....	32
Figure 8-7: Use case of User panel	33
Figure 8-8: Use case of Author panel	34
Figure 9-1: Class Diagram.....	37
Figure 9-2: Sequence Diagram.....	38
Figure 9-3: Low fidelity prototype (home page).....	39
Figure 9-4: Low fidelity prototype (login page)	39
Figure 9-5: Low Fidelity prototype (Admin panel)	40
Figure 9-6: Low Fidelity prototype (Reviewer panel).....	40

Figure 9-7: High Fidelity prototype (Home page).....	41
Figure 9-8: High Fidelity prototype (Login page)	42
Figure 9-9: High fidelity prototype (Admin panel)	42
Figure 9-10: High fidelity prototype (Reviewer panel).....	43
Figure 10-1: Root folder.....	44
Figure 10-2: Client-side Folder structure	44
Figure 10-3: Server-side folder structure.....	45
Figure 10-4: Actual folder Structure Sample	45
Figure 10-5: High Fidelity prototype (Product page).....	46
Figure 10-6 High Fidelity prototype (Product Preview).....	47
Figure 10-7: High Fidelity prototype (Product mobile view).....	47
Figure 10-8: High Fidelity prototype (KYC verification page).....	48
Figure 10-9 High Fidelity prototype (Overview page).....	48
Figure 10-10: High Fidelity prototype (Account details page).....	49
Figure 10-11: High Fidelity prototype (Purchase page)	49
Figure 10-12: High Fidelity prototype (Admin page).....	50
Figure 10-13: High Fidelity prototype (Dashboard page).....	50
Figure 10-14: High Fidelity prototype (Setting page).....	51
Figure 10-15: High Fidelity prototype (License verification page).....	51
Figure 10-16: High Fidelity prototype (Workspace page).....	52
Figure 10-17: High Fidelity prototype (Withdrawals page).....	52
Figure 10-18: High Fidelity prototype (Ticket page).....	53

List of Tables:

Table 8.1.3-1: Table of Module system.....	36
Table 10-8.1.3-1: Table of Module system	55
Table 11.2.1-1: Unit test -1	57
Table 11.2.2-1: Unit Test-2.....	57
Table 11.2.3-1: Unit test-3	58
Table 11.2.4-1: Unit Test-4.....	58
Table 11.3.1-1: Module Test -1.....	59
Table 11.3.2-1: Module test-2	59
Table 11.3.3-1: Module test-3	60
Table 11.3.4-1: Module test-1.....	60

Chapter 1: Introduction

1.1 Introduction

With the world become increasingly digitalized, and as the pace of life constantly increases, this is vital. In comes CodQube, a new revolutionary digital store designed with both developers in mind as well as normal functioning individuals. Whether you are a professional programmer seeking for more complex software or an averagely demanding user who is looking for some applications that would be helpful, CodQube was created to make it as easy and entertaining as possible.

CodQube is a one-stop store that compiles all sorts of digital products in an easy to download interface. This includes coding tools and software for the development and designing needs of the developers as well as the educational and productivity apps for common user. In fact, CodQube's objective is to be a marketplace where different people can find, buy, and apply premium digital services.

Unlike other similar platforms, CodQube's user interface is simple and sleek while its engine is powered by artificial intelligence-based algorithms. This system gets adapted based on your activities on the platform and the kind of products you show interest in hence making your shopping experience more effective.

The platform is also intended to make the buying process as secure as well as easy as possible. There is nothing wrong with product reviews and ratings and other similar aids when it comes to making a decision. It also shows that CodQube is always attempting to add new technologies and responding to customers' feedback in order to provide only the best of digital products.

First at its simplest, CodQube isn't simply an example of yet another digital storefront; it's a home. Here any developer and hence regular user of computers can find sift, promote and hence benefit from good tools. Besides helping one to find the right digital resources, this effort personalizes and integrates the process to make it convenient for everyone.

In general, over the course of the project development, all resources presented have been researched to achieve the maximum level of the obtained information. All valuable contributions made by different sources have been accompanied by the correct citation.

Therefore, CodQube revolutionizes the digital marketplace business. CodQube provides the clients with an opportunity to make a peculiar purchase maximally guided by AI and explore a wide range of digital products. It also is more of a community than a mere market place with its aim of becoming the solution to the lack of efficient utilization of digital resources.

1.2 Purpose of Project

Create an Inclusive Digital Marketplace: Create a unique platform in which could be united a big variety of digital products for developers and ordinary users.

Simplify the Discovery and Purchase Process: Design an easy purchase process of the digital tools and the sources necessary for using them with a user-friendly interface.

Enhance User Experience with AI: Introduce a personalized recommendation engine based on artificial intelligence that would track the consumers' habits and present products of their interest.

Ensure Secure and Smooth Transactions: Ensure there is the availability of a secure and effective way of buying, including giving users accurate reviews and ratings for the products.

Support Developers and Users Equally: Provide products that are to the interest of software mnemonic programmers and users of social media app, games and educational platform.

Continuously Integrate Emerging Technologies: Continuously upload the new technologies and features on the platform in accordance with the requirement of users in order to provide the high quality of digital products.

Foster a Community Environment: Create an environment where the developers and customers are connected and learn what other people do with digital tools.

Optimize Resource Allocation: Sustain an AI to be useful in the management and utilization of the digital resources in the platform.

Deliver a Personalized Shopping Experience: Lea AI offer to the clients a shopping concept that seems as if is created from scratch with focus on each user.

Set a New Benchmark for Digital Marketplaces: In fact, offer something a lot better than many traditional marketplaces, using cutting-edge technology while keeping the user at the center stage.

Chapter 2: Initial Study

2.1 Project Proposal

Objective: We envision, CodQube as a next generation digital marketplace driven by AI to simplify, personalize, and reduce much of the confusion for developers & general users while discovering & using digital products.

Features: CodQube is going to give a clean user-friendly interface for the users, enough option to navigating and talking with the platform. The smart AI-driven recommendation system using will be found on this store and will suggest products that match user preferences based on their behaviour. This is expected to include a range of digital products: from sophisticated coding tools and design software, all the way through to educational resources and productivity apps. Safety will be of the utmost importance, with numerous methods to ensure trustworthy and reliable transactions. Users will also receive real-time updates and be able to search for the latest product details.

Advantages: CodQube promises to improve user experience by offering a personalized and simplified shopping process. Students will have a faster and easier experience in finding the digital tools they are looking for, which will cause them to focus more on their tasks. The platform has built-in secure payment functionality, making the buying process easy and personal product recommendations to increase user engagement. These unique characteristics are expected to help CodQube emerge as one of the largest digital marketplace.

Time-line: From start to finish, the project is scheduled to take six months. Ongoing conversation with all stakeholders to define the marketplace requirements. Working over core features and interface. Test test today for robustness Vendor Size Impact & User Experience & Enjoyment Quickly provisionable justify Deployment of Market0 Phase Launch

Resources: It would need a committed group of programmers., UI/UX designers execute CodQube. Also, the required technological resources will be requested to drive the development, testing and deployment activities.

Conclusion: CodQube is poised to disrupt the digital product marketplace with its unique deployment of AI. CodQube is disrupting the e-commerce space by providing a personalized experience that caters to the exact needs of its consumers, and thereby shaping an entirely new level of digital shopping & user engagement much easier and more user-centric.

2.2 Project Scope

The big goal of the CodQube project is to build a revolutionary digital marketplace where every user can easily find and use digital products using cutting-edge AI technology. It will look something like this.

Personal Recommendations: CodQube backed by AI will offer Personal Product Recommendations based on user behaviour and profile. This makes it also easier for users to find the digital agencies and resources that are most important to them.

An Intuitive Interface: Our platform will be for easy access and interaction with an intuitive design similar to what you are accustomed to. This makes it easy for the users to browse and buy digital products.

Vast Product Collection: CodQube will consist of hundreds of digital products, spanning from high-end coding and design tools to academic resources and utility applications — for developers as well as general users.

Safe Transactions: Safety Comes First. The platform will also be highly secured for sound transactions, so users can buy without fear of hidden charges and fraud.

Reviews & Ratings: Detailed Product Reviews and User Rating to help you make a good decision. This will allow them to take better decisions for their purchases.

Updated in Real Time: With real time updates, the platform ensures that users are always informed about a latest product or trend when it comes to discovering digital products.

AI-Insights: CodQube will learn from user interactions and get smart suggestions by employing AI algorithms. Since the app is being listed on Google Play Store, this helps users to find from wide range of products and features which suits their requirement most.

Customized User Profiles: Users can create and customize profiles to use some on their preferences That will be useful for their journey within the platform.

Interactive Community Features: CodQube has the ambition to create a community environment that is not only vibrant but also to learn and grow, where all users (developers) would interact with each other, exchange thoughts and everything possible with respect of using digital tools.

2.3 Background of the Project

In the current digital age the proper tools can really make a huge difference, whether you are an intense developer or just another person who is on the lookout for some new cool apps. CodQube, unlike the apps above give you a quick, easier and more personalised way to buy and sell local attraction-based digital products.

Now, suppose you land on the site of CodQube(See This Is What Members Will Be Offered), with a always trendy and beautiful template. Use it to easily find your way as you check out digital tools ranging from the latest in computer coding and design software, to learning resources for all ages. We wish that you experience the joy of exploring, purchasing and consuming on CodQube.

What we do differently from everyone else is employ Artificial Intelligence for the best possible results that you can obtain. This is not technology for the sake of technology; it's about improving your experience. Over time, the platform will learn about what you like (and do not like) and begin to offer personalized product recommendations based on your individual requirements. CodQube is made for pros or just a productivity apps hunter and on their first visit they should immediately spot what they are searching for.

Our 1st priority is security We understand how essential it is to feel secure with the paying procedure on the internet. You can buy it with confidence knowing that all transactions are safe, reliable and will be processed by CodQube. You will also get the best reviews and ratings of each item you buy which I believe helps whenever you make future purchases.

Things stay current with real-time updates. CodQube always keeps you updated with the most recent product offerings and trends so that no premium digital tool or software is lost to your knowledge. This ensures dynamic and interesting interaction is available in your environment.

Community is key for us. CodQube is not only sales and purchase, it is a bridge too. CodQube is a highly interactive marketplace connecting you with other users as well as sellers to share, learn and grow together in the crypto space. CodQube is more than just a marketplace; it's a place for the community of digital creators.

CodQube is all geared to serve as your new wave in the digital marketplace! We are changing the game for the digital market by using AI for personalized recommendations, guaranteeing safe transactions, keeping you in the loop and building a community. Our user-centric journey offers a seamless and captivating experience, as you revel in the lavish offerings of the digital era.

2.4 Objectives

- Create a website that is easy to navigate, purchase the product and sell the digital item.
- Apply Machine Learning to provide personalized recommendations on what the user loves and wants.
- Make sure all purchases are secure and trustworthy by applying the highest level of security
- Provide detailed product reviews and ratings to help shoppers make informed decisions.
- Provide real-time communications on new products and trends to keep your users informed.
- Create a meeting place for users and marketers to communicate, exchange ideas, cheer each other up.
- Design an order system that makes transactions smoother, turning it into a fully-functional, full-stack flow.
- Set user order tracking, get updates in real time.
- Log user activity and provide intelligent tips/suggestions using AI.
- Use modern, fresh design elements that encourage users to browse the platform for enjoyment

Chapter 3: Literature Review

3.1 Discussion on problem domain based on available solution

CodQube was created by us to make discovery and consumption of digital products easier & insanely enjoyable for everyone. We also realized that most existing platforms have lots of features which in a way gets complex to the user as it becomes difficult for them to find what they are actually looking for. Not to mention the fact that using these sites can be a pain in the ass if you're not some sort of tech genius.

As with anything else security was a big problem. But the way they treat transactions can make it less secure to pay, which is a disaster on many platforms. But let's face it, there is often no sense of community — users very rarely interact with developers and provide feedback.

Additionally, remaining current with new technology and evolving user needs is a challenge for many platforms. The problem which CodQube solves User-Friendly UI Secure Transactions Active Community Continues Update based on your Feedback Make CodQube a planet for authentic and tailor-made digital tool discovery.

3.2 Discussion on Problem Solution Based on Available Solutions

We went ahead and focused all our attention in building CodQube taking references from what's missing in present day digital marketplaces. These are the key problems CodQube tackles with —

Smart Recommendations: With various platforms, you get an overwhelming choice of options leading to the circumstances where it is difficult for users to find the content they are searching for. CodQube that uses AI technology to sit and learn from your behaviour, suggests products what you like just in time similar kind of recommendation that Netflix gives to his users for movies.

User-Friendly: tons of websites are difficult to use, especially for beginners. Clean and simple design inspired by user-friendly apps makes sure that everyone can easily see, locate and buy digital assets from CodQube.

Safer Purchasing: One of the main worries is safety in transaction. With industry-standard security measures (built with inspiration from platforms like PayPal) in place, CodQube ensures that your data is safe and every transaction you make is secure and trustworthy.

Engaged Community: Digital marketplaces have always been a place where community is largely missing. CodQube believes in communicating and likes people to leave reviews so that we can grow, build a great community where the users talk back despite it being feedback.

Frequent Upgrades: The technological world is evolving on a daily basis so platforms have to stay ahead of the curb. CodQube is in development and relies on user feedback to add new functionalities as well as track latest technology trends, for it to remain relevant and beneficial.

CodQube takes the best of smart technology, user-friendly porcelain support, secure payment processing and crowd sourced idea skin also always sources betterment to produce a digital marketplace that meets fullest collects experience in every sense.

3.3 Recommended Approach

These are how we intend to stand CodQube as the most effective digital marketplace for its users, using a combination of observations around the prevailing pain-points in today's platforms and how feasible those solutions already stand existing:

1. Smart AI for Personalized Recommendations: We should leverage state-of-the-art AI to know your preferences and requirements. It will understand your behavior and preferences and provide you the recommendations according to it. So, you would not have to sift through thousands of options and could see the hand-picked products that truly matter to you which will make shopping an efficient and fun experience.

2. Focus on User Experience: We want the best possible user experience. The interface of CodQube should be clean, intuitive and hassle-free enough to make tech-savvy developers and common people discover/ buy/ use the digital products. We want the process to be as seamless and pleasant as possible from beginning to end.

3. Security You Can Trust: Security is the cornerstone of any online transaction. One hundred percent press worthy information There would be robust security measures like encryption and secure payment gateways to keep your data safe and make sure every transaction is secure. We prioritize your security to earn and retain trust from convenience, understanding you can buy with absolute confidence.

4. Building a Vibrant Community: We not only want CodQube to be a platform for listing the services, but also for user reviews and developer feedback. Privately, we have access to discussion forums, star ratings and Q&A sections that can help to deliver the same sort of interaction. Thereby not only enriching the experience for you but also helps us iterate and evolve our platform constantly according what you disclose.

5. Continuous Improvement: Since staying relevant requires the organization to never be static. You can provide the feedback and based on that our CodQube will keep be agile and responsive with your feedback moreover close to technology changes. Especially on a platform that grows with you, updates and new features are essential to keep things fresh and in line with your requirements as well as state-of-the-art.

Chapter 4: Methodology

The process of choosing a framework and related approaches is known as software development., to help in the production and management of software. Two of the most common methods are Agile and Waterfall. Agile is famously flexible by being able to pivot using interim milestones, whereas Waterfall pursues a more linear and deterministic path. The methodology adopted can vary according to project complexity, team size as well as project requirements. Opting for the right methodology gets you organized, increases collaboration brings-in process to project management, reduces risks and increases customer satisfaction.

4.1 What to Use

Selecting the best software development process is crucial to completing our project's objectives quickly and successfully, “CodQube: Building an Inclusive Digital Marketplace.” To assist with this you have to understand the features and benefits of many methodologies are Agile, Scrum and Rapid Application Development (RAD) among others before making that final choice.

4.1.1 Agile Model:

Agile is a development methodology that promotes collaboration and offers team members autonomy. The agile development takes a iterative approach to the software, in short cycles called sprints that deliver working software. Agile lives on iterations and customer evolution feedbacks. Work is divided into user stories the individual features from the users' perspective. The stakeholders and the development team order these stories using a simple prioritisation mechanism, and this orders how the work is planned for each sprint. By having daily stand-up meetings with the team, you will keep everyone aligned and also remove any roadblocks that come up right on time. The team shows off progress and receives feedback from a review session at the end of each sprint. The primary quality of agile is its flexibility. This makes changes or updates possible at any stage, as the project can then adapt to new requirements or feedback.

We have built an inclusive digital marketplace using Agile for “CodQube”, enabling us to rapidly respond to users and deliver a high-quality final product that fulfils user needs.

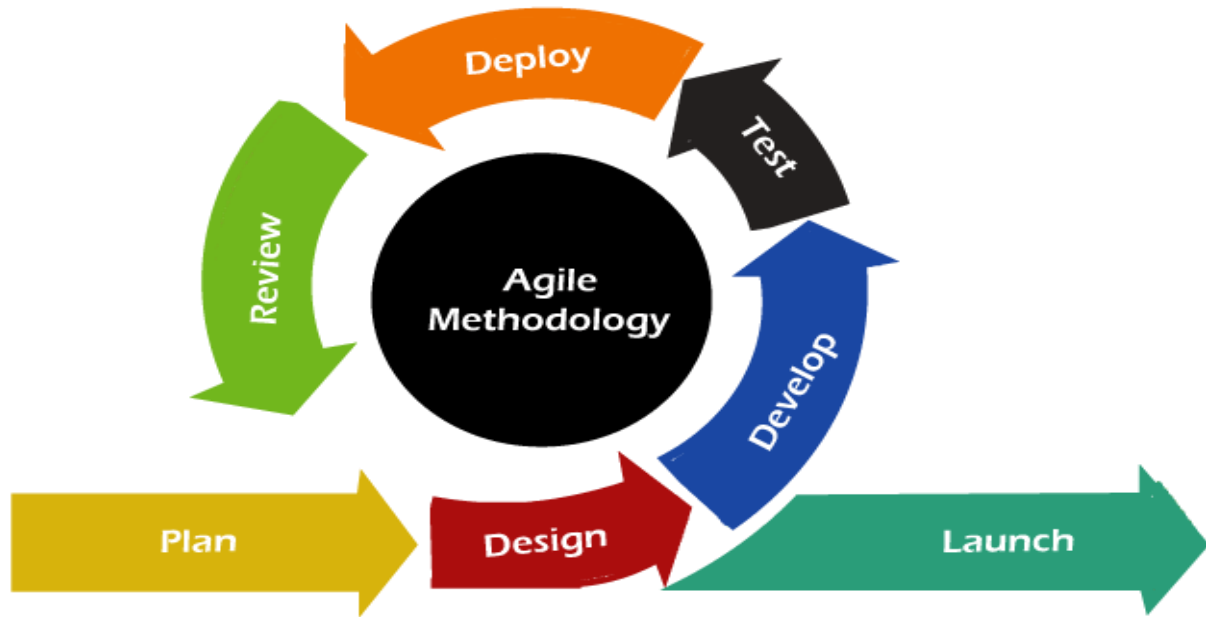


Figure 4-1: Agile Model

4.1.2 Scrum Model:

The Scrum model is an agile framework that emphasizes collaboration, adaptability, and iterative less, according to Sutherland, a co-creator. The project is divided into brief sprints of two to four weeks, during which the team ships work increments of software and completes specified tasks. The scrum master, the development team, and the product owner make up the compact, cross-functional scrum team. The product owner is responsible for defining and allocating the items that go into the project backlog, which is a list that includes all of the features and functionalities that are wanted. The development team is responsible for carrying out all team art, and the scrum master assists in resolving any potential problems. In CodQube, we have decided to go with the Scrum model focus on teamwork, adaptability, and visible results adding value in short cycles known as sprints. And the development team handles a lot of tasks that really are beneath our product guy's endlessly dapper wind him up and watch him go exterior. We have our trusty Scrum Master behind the scenes ensuring everything runs smoothly. After each sprint, we run a review of what has been done where the team shows to stakeholders and collects their feedback. This cycle ensures that there is more visibility, increased team work to make everything work and we easily adapt any changes soon, which in the end makes CodQube a feasible and user-friendly place.

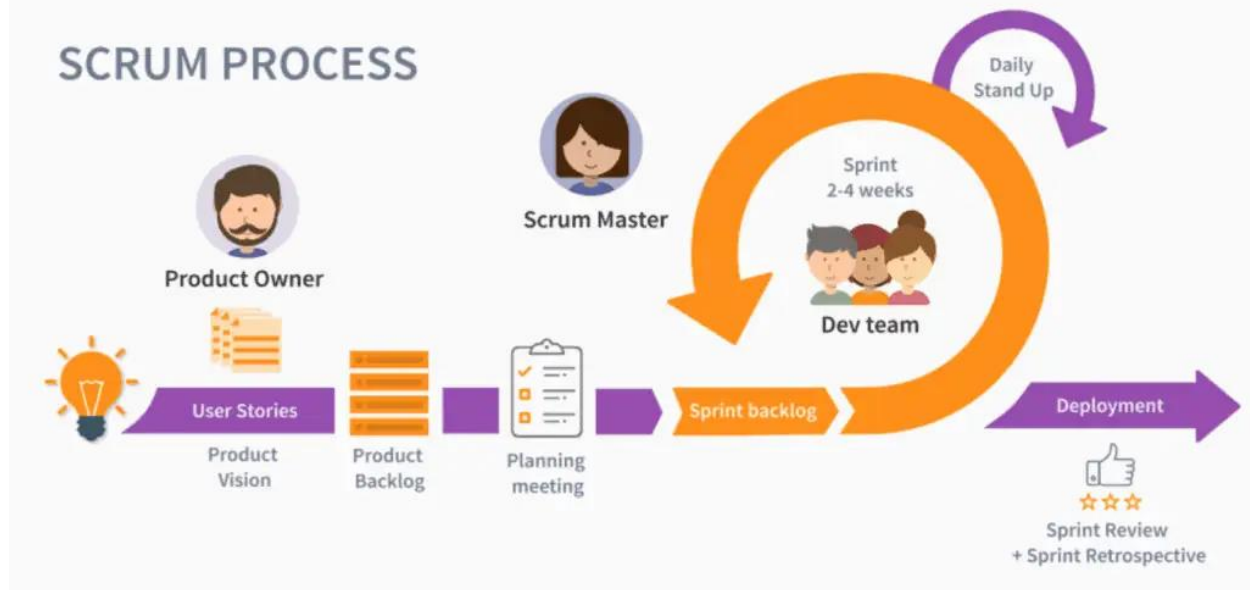


Figure 4-2: Scrum Methodology

4.1.3 Rapid Application Development (RAD) Model:

Rapid Application Development, also known as RAD, is an iterative software development methodology that emphasizes the development of applications in small increments, similar to the integration of newly updated components into a larger system. Rapid and intimate interaction among developers and users is emphasized to expedite the development process. With four main stages—requirements planning, user design, construction, and cutover. It offers an integrated development cycle. Compared to previous approaches, RAD is more focused on user interaction across the whole life cycle, which leads to quick feedback and iterations.

Software prototypes are made using the rapid prototyping method in this model. This feedback loop allows the prototypes to evolve and develop, ultimately shortening development creating better built designs. The benefits of RAD are faster development cycles, more user involvement and defect finding. It rests on the iterative nature that RAD operates off of, making itself the ideal form to enable changes at every step and yet still resulting in a final product that is very likely to meet end-user requirements.

The main drawback of RAD model is end of the application. That might lead to repetitive requests for data, as we can see in the iterative Waterfall model. Not ideal for large-scale projects with complicated architecture. This implies that prioritizing speed occasionally results in quicker fixes that compromise sustainability and reliability for a long time.

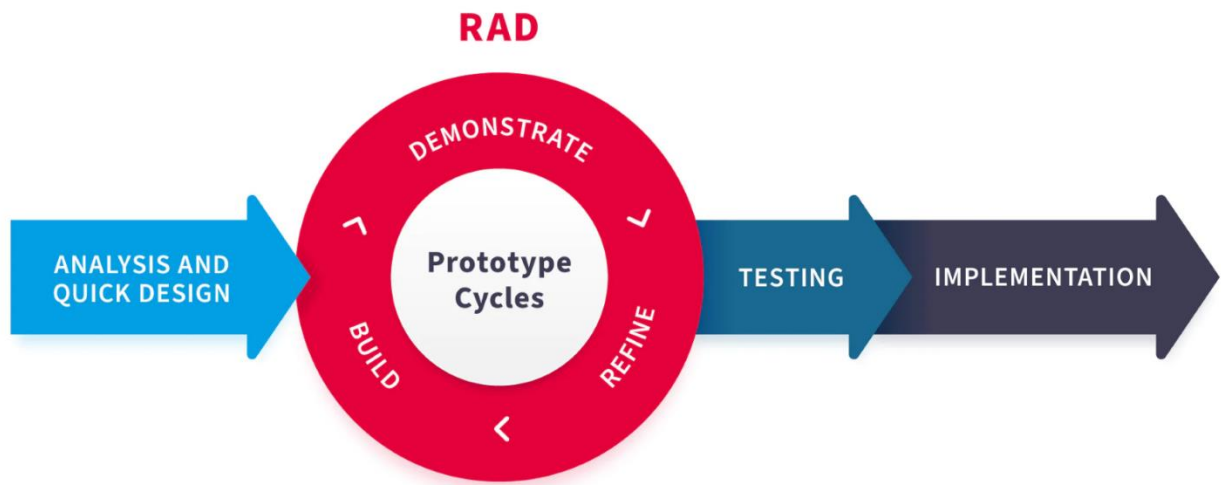


Figure 4-3: Rapid Application Development (RAD) model

4.2 Why to use

When we started CodQube, it was essential to choose a methodology that would let us be very responsive towards user needs, adopt new technology quickly and in general orient towards quality and quick delivery of our work. So here are the top pointers why we selected Agile as our methodology of development at CodQube :

- 1. Customer-Centric Focus:** Agile needs end-user and stakeholder interaction on an ongoing basis. This means that CodQube development is always aligned with actual needs and preferences of our users. Getting customers involved at an early stage, and on a regular basis ultimately helps us deliver features that truly help them in their day-to-day challenges and desires making CodQube easier to use and more valuable.
- 2. Flexibility and Adaptability:** Digital market changes continuously with upcoming technologies and changing user needs. The iterative approach of Agile helps us to be more flexible and responsive to these changes. Relatively, it is kind of similar to managing the project in small pieces for quick iterations. Soft release enables us to quickly tweak and change features based on customer feedback as well as market trends. This flexibility ensures CodQube is evergreen and functional.
- 3. Quality Assurance:** Shorter is Faster, Frequent is Agile. CodQube has exceptionally high-quality standards, and by continuously testing updated features throughout the development process we help to guarantee those standards in terms of performance. This runs continuous testing

on our network & provides us a real-time view to detect and fix early bugs, which lowers risk from critical defects and improves end-user experience.

4. Faster Time-to-Market: When running in a very competitive digital environment, one of your top priorities should be speed. We can deliver the feature more quickly to our user so Agile in other word provides an iterative sprint. By not having to wait for a final, full platform release, it means we can provide incremental improvements to the platform iteratively. This practice helps CodQube maintain its agility and provides faster response time to user requirements.

4.3 Sections of Methodology

In a rapid and collaborative software development journey, several critical stages of the agile development process are essential. When we directly relate these stages to CodQube, they occur as follows:

Project Start-Up: Define CodQube's vision, targets, and starting points to lay a solid foundation for future development and implementation. It is to ensure that everyone involved in the project team members or stakeholders are all on the same page regarding the project's purpose and where it should go, and lead towards one clear direction for how things will be done (Highsmith, 2002).

Release Planning: In this phase, we all decide the features and functionalities for each release together. These are then placed in order of importance of business value and priority to customers to ensure that each release of CodQube delivers the most value, and more closely aligns with our strategic goals (Sutherland, 2018)

Iterative Development: Agile methodology uses an iterative development paradigm called sprints in developing CodQube. This relates to working with the team on specific user stories within that sprint which are then built, tested and refined. This process allows to get feedback on the work done that gives possibilities of incremental improvements and adaptations (Cohn, 2006)

Daily Stand-ups: Hosted by the team on a regular daily basis, to ensure continued communication and collaboration. Short stand-ups are intended for proposing questions related to current status, remove obstacles and keeping team members aligned with what is happening on the project as a form of transparency: Sutherland (2018)

Continuous Integration & Testing: We heavily believe in Continuous integration and testing to make sure that the quality of CodQube does not go down. Continuous integration is the process of frequently merging code changes and running automated test to detect errors early on while coding, so that it remains reliable and still functional during development (Highsmith, 2002).

Sprint Review and Retrospective: Everything we have associated with a sprint review to ensure that stakeholders are able to see the completed features and provide feedback is added here. Afterwards, we do a sprint retrospective meeting to evaluate the phenomenon on the turnout of the sprints. Based on the feedback, we have a retrospective after each sprint in which we discuss what

went well and things that could be improved upon and how to do it better in the following cycle (Cohn, 2006).

Sticking to these Agile stages gives CodQube development process a defined form as well as adequate flexibility for it to be continuously adaptable through feedback from users, and shifting requirements. This approach ensures that CodQube is not only a quality platform but also user-driven and market-sensitive, which in turn makes the service enabled and adaptive.

4.4 Implementation Plan for CodQube

Below is a small, no plagiarism outline on how to introduce the Agil model at CodQube.

Project Vision define: To provide all stakeholders with a clear development direction, the mission, objectives, and vision for what CodQube is to become are established.

Develop Product Backlog: Define features and requirements, sequence them. The backlog acts as a development tasks map.

Sprint Planning: This demarcates top-priority backlog items that we have committed to completing through a series of small, concrete tasks which the team will work on over the specific sprint.

Conduct the Daily Stand-ups: Hold short daily meetings that will take place during 10 minutes where you will talk about what has been done, what each team is doing and to align on the goals of this project.

Use an Iterative Development: Produce functioning software in small timeboxes (sprints), which last anywhere between 2 to 4 weeks.

Continuous Testing and Integration: It is important to test and integrate code changes on a frequent basis so you catch problems early and release high-quality software Its throughout development.

With all these steps in place, we can be sure that the development at CodQube is structured yet flexible and user-centric letting us constantly improve based on feedback.

Chapter 5: Project Plan

5.1 Work Breakdown Structure

The Work Breakdown Structure (WBS) approach is a project management technique that divides a project's work into smaller, easier-to-manage components. As a result, we can better manage our time and resources and maintain an accurate picture of the project's progress (A Guide to the Project Management Body of Knowledge, 2017). We have divided the project into several sections to ensure that every area of CodQube receives equal attention.

Here's an overview of the WBS for CodQube:

Task Name	Duration (days)	Start	End
Introduction	8	8 Feb 2024	15 Feb 2024
Initial Study	5	16 Feb 2024	20 Feb 2024
Literature Review	3	21 Feb 2024	23 Feb 2024
Methodology	5	24 Feb 2024	28 Feb 2024
Project Plan	8	1 Mar 2024	8 Mar 2024
Feasibility Study	7	9 Mar 2024	15 Mar 2024
Foundation	7	16 Mar 2024	22 Mar 2024
Exploration	9	23 Mar 2024	31 Mar 2024
Engineering	12	1 Apr 2024	12 Apr 2024
UI/UX Development	5	13 Apr 2024	18 Apr 2024
Development	37	19 Apr 2024	25 May 2024
Testing	7	26 May 2024	2 Jun 2024
Implementation	5	3 Jun 2024	8 Jun 2024
Critical Appraisal & Evaluation	2	9 Jun 2024	10 Jun 2024
Lesson Learned	4	11 Jun 2024	14 Jun 2024
Conclusion	4	15 Jun 2024	18 Jun 2024
	Total: 130 days		

Figure 5-1: Work Break down Structure

5.2 Resource Allocation

Resource allocation means putting people and other resources where they are most productive (doing the right work) while keeping fast, complicated projects on track and within budget. When the right tasks are going to the right people, and resources are being used in the best way possible, projects have a better chance of succeeding (A Guide to the Project Management Body of Knowledge, 2017). It has a black and white Resource Allocation plan in place which details what each of the stakeholders will contribute at each stage of CodQube.

Here's our resource allocation plan:

Task Name	Duration (days)	Resource
Introduction	8	Analyst, User
Initial Study	5	Analyst
Literature Review	3	Analyst, Team Leader
Methodology	5	Analyst, Developer, Project Manager
Project Plan	8	Analyst, Project Manager, Team Leader
Feasibility Study	7	Analyst, Project Manager, Team Leader, User
Foundation	7	Analyst, Team Leader
Exploration	9	Analyst, Developer, Team Leader, Designer
Engineering	12	Project Manager, Team Leader
UI/UX Development	5	Designer, Developer, Team Leader, User
Development	37	Developer, Analyst, Tester
Testing	7	Tester, Developer, Team Leader, User
Implementation	5	Project Manager, Developer, Tester
Critical Appraisal & Evaluation	2	Analyst, Developer, User
Lesson Learned	4	Developer, Analyst
Conclusion	4	Analyst
	Total: 130 days	

Figure 5-2: Resource Allocation

5.3 Time Boxing

Time boxing is an approach used in project management, where you set aside specific intervals of time called a time boxes to work on your tasks or activities. It inculcates the sense of time urgency which helps to focus and maintain discipline in accomplishing work within a particular timeline (Cockburn, 2002). Below are our predefined timeboxes for project to keep efficiency and productivity:

Time Boxes	Task Name	Duration (days)	Resource
TB1	Introduction	8	Analyst, User
	Initial Study	5	Analyst
TB2	Literature Review	3	Analyst, Team Leader
	Methodology	5	Analyst, Developer, Project Manager
TB3	Project Plan	8	Analyst, Project Manager, Team Leader
	Feasibility Study	7	Analyst, Project Manager, Team Leader, User
	Foundation	7	Analyst, Team Leader
TB4	Exploration	9	Analyst, Developer, Team Leader, Designer
	Engineering	12	Project Manager, Team Leader
	UI/UX Development	5	Designer, Developer, Team Leader, User
TB5	Development	37	Developer, Analyst, Tester
TB6	Testing	7	Tester, Developer, Team Leader, User
	Implementation	5	Project Manager, Developer, Tester
TB7	Critical Appraisal & Evaluation	2	Analyst, Developer, User
	Lesson Learned	4	Developer, Analyst
	Conclusion	4	Analyst
		Total: 130 days	

Figure 5-3: Time Boxing

5.4 Gantt Chart

In project management, a Gantt chart is an organizational tool that shows tasks, deadlines, and the project plan in a timeline fashion. This chart helps with better planning, collaboration, and monitoring of the project's progress by providing a graphical representation of project activities, their durations, and their dependencies. (Tomčíková, 2018).

For CodQube, the Gantt chart will illustrate each phase of the project, from initiation to deployment, indicating the dates on which duties begin and terminate, their durations, as well as its mutual reliance. This will allow us to see at a glance how different tasks overlap and depend on each other, helping to ensure that each phase of the project is completed on time.

Here's an overview of a potential Gantt chart for CodQube:

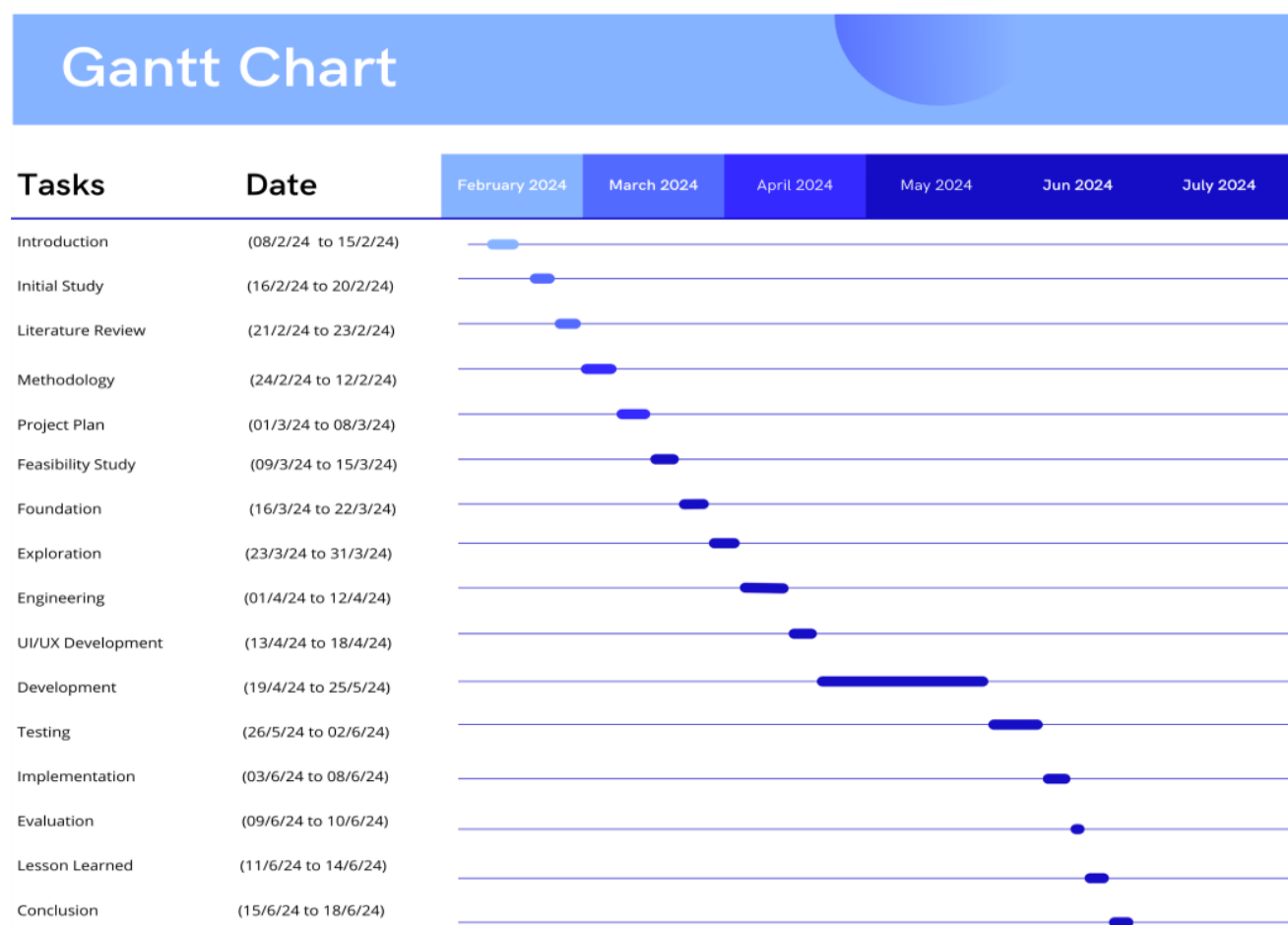


Figure 5-4: Gantt chart

Chapter 6: Feasibility Study

6.1 All Possible Types of Feasibility Study

We have to assured that the complete concept behind CodQube is going to be successful including technical, economic, operational, legal and scheduling parameter so we will make our feasibility study. The objective is to give a complete view on the feasibility of the project, allowing stakeholders to chose wisely if it is worth implementing or not.

6.2 Technical Feasibility

For the digital marketplace CodQube, Technical feasibility is to determine the technologies, tools and infrastructure needed. This means considering consoles, and frameworks which other ones you will going to use based on the components of the project. It is important to validate chosen technologies are sustainable before the start of the project, as changing in the middle can be difficult and costly. If we take a look at what solutions kinds of problems existing solutions are leveraging tech stack components, and then choose the specific parts, we will do so from a much more educated perspective. The main technologies and tools that we will utilise for development of CodQube are as follows: -

- **Devices:** Laptops, Desktops
- **Networking:** Wi-Fi Routers
- **Development Tools:** Visual Studio Code (VS Code)
- **Productivity Tools:** Microsoft Word, Google Sheets
- **Browsers:** Google Chrome, Mozilla Firefox
- **Operating System:** Windows 11
- **Database Type:** SQL
- **Client-Side Technologies:** HTML, CSS, PHP
- **Framework:** Laravel
- **Runtime Environment:** PHP version 8.2
- **APIs:** Personal API

This detailed technical plan will guarantee that CodQube will evolve over a strong and compatible technology stack and make the development process a cake walk with minimal risk of technical hiccups.

6.3 Economic Feasibility

We will carry out the economic feasibility of CodQube which is a digital marketplace so that it can determine objectively as to how much it holds economically. This particularly involves the evaluation of costs in terms of development, implementation, ongoing maintenance and operation. We'll also review the possible return on investment (ROI) as we assess the market demand, pricing models and revenue forecast. Two main streams of revenue model for CodQube are,

Product as a Solution:

Selling to businesses: CodQube can be a complete system solution to businesses so they may launch the product as their own with required customization. This can be monetized by selling CodQube's marketplace tech stack to these businesses allowing them to improve their own e-commerce capabilities.

Service as a Solution:

Revenue From Order Placement: CodQube in itself can act as a marketplace providing an end-to-end pipeline from order placement and all the way till final delivery. You can earn money on each individual order. In other words, any transaction routed via the platform of CodQube generates revenue making it a recurring income model.

Therefore, by closely analysing these revenue models with their respective costs, we can deduce the economic viability of CodQube and make it economically sustainable. The results of this analysis will help us decide which way is the best for our project.

6.4 Operational Feasibility

Operational Feasibility focus is on how well the project integrates with existing processes, resources and work flow structures. To execute this, the organization must assess things like whether CodQube works with its infrastructure; if the staff understand how to operate it and how well it fits in their system of operations. It also looks into likely operational challenges or risks that could be faced during the project execution to avoid any disruption and facilitate integration.

6.5 Legal & Compliance Feasibility

This is a legal and compliance feasibility-whether the laws, regulations and industry standards are in place for CodQube to get it adhered or not. This covers elements such as legal issues for intellectual property rights, data privacy, security, and AI implementation as well as online transaction including payment gateway. Over and above that compliance helps in lower legal risks to the business thereby helping both side of businesses as well as its users also.

Going into exploratory depth here will allow provisioning us with a wholesome set of preachy feasibility findings around the same, and hence amped up stakeholders could make some informed choices through conducting.

Chapter 7: Foundation

7.1 Problem Area Identification

Recognizing and highlighting the critical challenges or hurdles to be tackled is a key aspect of problem identification in project management. The methodology, meanwhile, empowers project managers to gain a solid grasp of these issues and develop strong solutions. Key steps to finding problematic areas -

Collect Data & Information: Collect relevant data regarding the project, its individuals, and outside variables that could have an impact on it using surveys, interviews, research studies, and document review.

Data Analysis: Conduct a thorough examination of data to reveal patterns, trends, and likely troubles. This helps to know what caused and how it affected the goals of a project.

Prioritizing Problem Areas: Should consider the importance of each source of identified problems according to its significance, impact on project success and the work load it represents into constraints in handling existing issue.

Defining Problem Statements: Make sure your problem statements are CLEAR: Concise Logical Emotional Action Responsible Realistic To keep the focus on problem-solving, they must adhere to the SMART (Specific, Measurable, Achievable, Relevant, and Time-bound) standards.

Meeting with Stakeholders: Meet and work with stakeholders to test and develop problem statements. Understanding the issues more deeply helps to inform your recommendations, and gaining buy-in for your recommendations requires insight and perspective.

Developing Solutions: Create workable solutions after you have identified and validated the problem areas. This can consist of discussions, probabilities study research done essay and asking and answer aspirational questions may be the best possible.

7.2 Interview

Interview is the process of project managers having direct conversations with individuals or groups so they can ask clearly defined questions, get detailed answers, and give and take information. This help to a more intricate analysis of the theme and clears doubts. Interviews can range from face-to-face interaction to wide telephone communication: or even, to effectively conduct interviews and save time.

7.3 Questionnaire

Structured sets of questionnaires are distributed to stakeholders. Examples of these could include online surveys, paper-based forms or electronic questionnaires. They offer a standard way of collecting data which facilitates the (((wretched cataloging))) processing, and analysing data generated. They are especially useful when engaging with many stakeholders and in cases where the data you need is most easily gathered using questions that are closed-ended or scaled.

7.4 Requirement Specification

Defining requirements is a crucial and essential stage in the project phase of development. which clearly outlines not only what your stakeholders expect but also their needs. It outlines project goals and objectives, defines scope, minimizes miscommunication, demonstrates progress, helps test deliverables, and serves as a guide for further testing initiatives.

7.4.1 Functional Requirements

Any digital marketplace such as CodQube should have specific characteristics and features, as described in the functional requirement.. They define the system in terms of possible actions, calculations, processing and anything that can be done with data, interacting with users. Following are the functional requirements for CodQube:

- As a user, create an account and safely log in.
- System privacy and security of data should be guaranteed by the use of authentication mechanisms.
- Product categories should be visible to users, who may also browse and search for products.
- The end users should be capable of utilizing the system to peruse product-specific descriptions, pricing details and reviews.
- The ability to add items to the shopping cart and proceed with the checkout should be available to users..
- Support Different Payment Methods: The system will support multiple payment options to assist the convenience of transactions.
- Before their final order the users must be able to track all of them.
- The should provide tracking information to users in an easy way after the sale has been made.
- Vendors should be able to register an account and upload their products in the marketplace.

- Vendors require an inventory and sales management dashboard
- These elements are reviews and ratings allowing users to review the products they purchased or want to buy.
- Product pages should show an average rating and user reviews.
- For this, an exclusive dashboard for admins can be created who are going to manage users/vendors/products.
- Add Authentication on the dashboard so that Only admins can access the page.

7.4.2 Non-Functional Requirements

These emphasize the system's qualities and restrictions, which include performance, security, usability, scalability, and maintainability. These are not specifications of functionality but requirements that specify how the application should behave. Non Functional Requirements for CodQube are as follows:

Performance:

- In a seamless user experience, the system should provide fast response times.
- It must be able to withstand many concurrent users without performance taking a massive hit.

Usability:

- Need to create an interface that is clean, well designed yet intuitive, and easy on the eyeball.
- Your system should provide easy-to-follow instructions to help users register a purchase, and give feedback once the process is completed.

Security:

- You need to store user data (personal data, payment details) in a secure way and transmit it securely too.

Scalability:

- It must be able to quickly handle a large volume of users, goods, and transactions.

Reliability and Availability:

- Your application needs to be fast and responsive, with as little downtime or interruption as possible.

- There must be some kind of backup and recovery in case of a data loss so that the business is not interrupted.

Compatibility:

- Make your application compatible with a variety of web browsers, operating systems, and devices in order to reach a broad user base.
- The user interface should adjust according to various screen sizes and resolutions, for improving the user experience on different types of devices.

This disciplined requirement specification is what guides the CodQube while developing so that only business objectives are being tagged along, not alluring with functional capabilities an a great user experience level as well maintaining the system within their performance and reliability limits.

7.5 Technology to be implemented:

The frontend and backend are the main parts of the CodQube application's system architecture. Bookshelf handle the UI for the User interface how user get into our application and explore digital products etc. Built with HTML, CSS, PHP (includes Laravel). The backend is the core of the application and it plays a crucial role in having different functionalities as well as outside services communication. Backend is built in PHP with Laravel powerful PHP framework. It contains a companion API endpoint created to interface with our internal license API and external APIs such as the OpenAI API, allowing for AI-powered recommendations and other integrations.

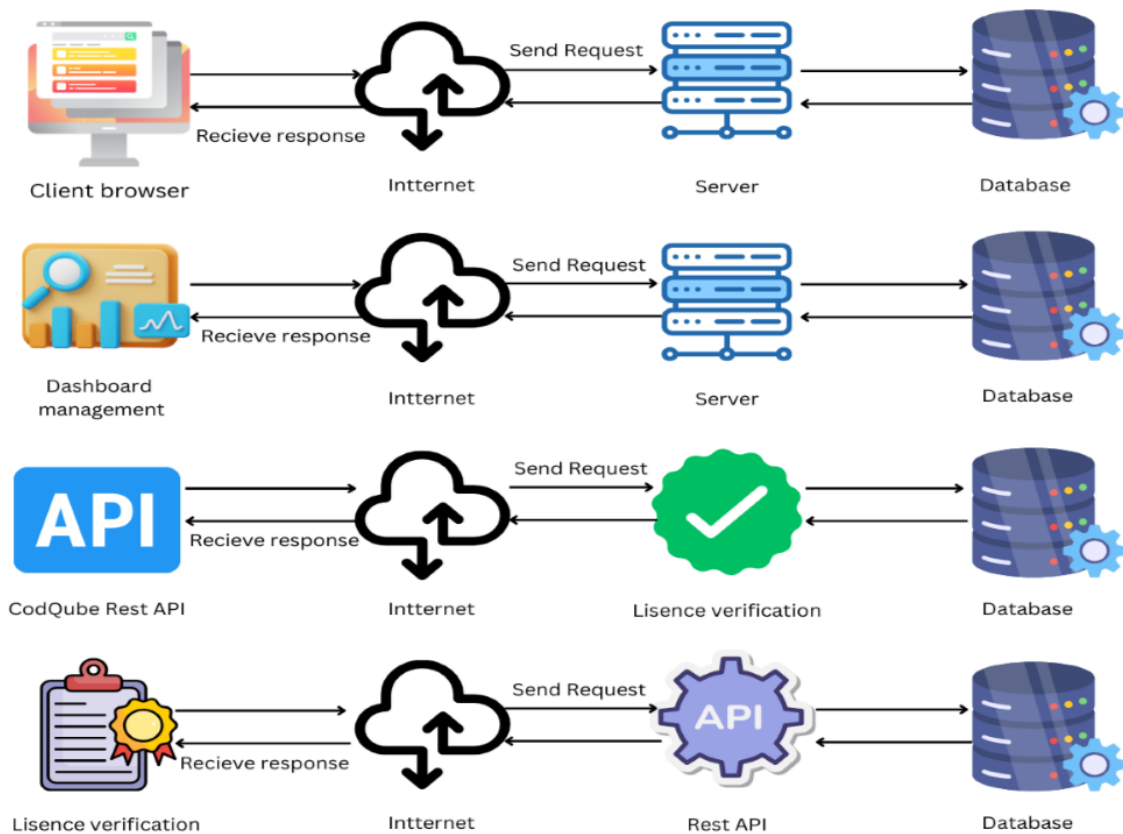


Figure 7-1: Model of the technology

Chapter 8: Exploration

8.1 Activity Diagram

In simple terms, activity diagrams used to explain how a system works which use, what are the functions or operations you need and when those operation will be executed. It started in improving the process efficiency, also obtaining a snapshot to see what is involved in each step and determine where bottlenecks need attention as well as areas to be optimized. They give a visual for the entire process and show the clear relationship between various activities, making it easier to understand how things are happening by the stakeholders.

8.1.1 Activity Diagram Part One (Order process):

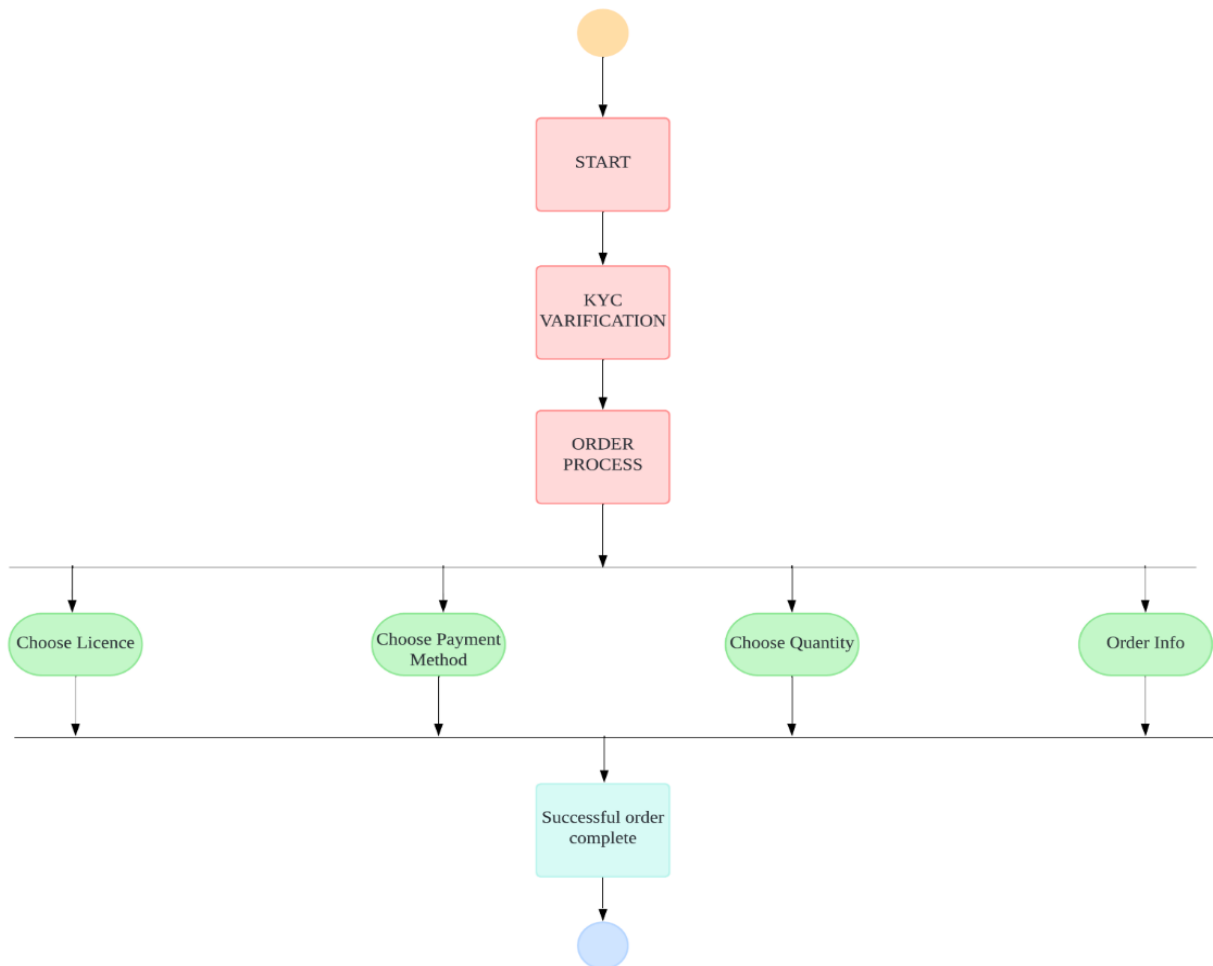


Figure 8-1: Activity Diagram (order process)

8.1.2 Activity Diagram Part Two (Upload Product):

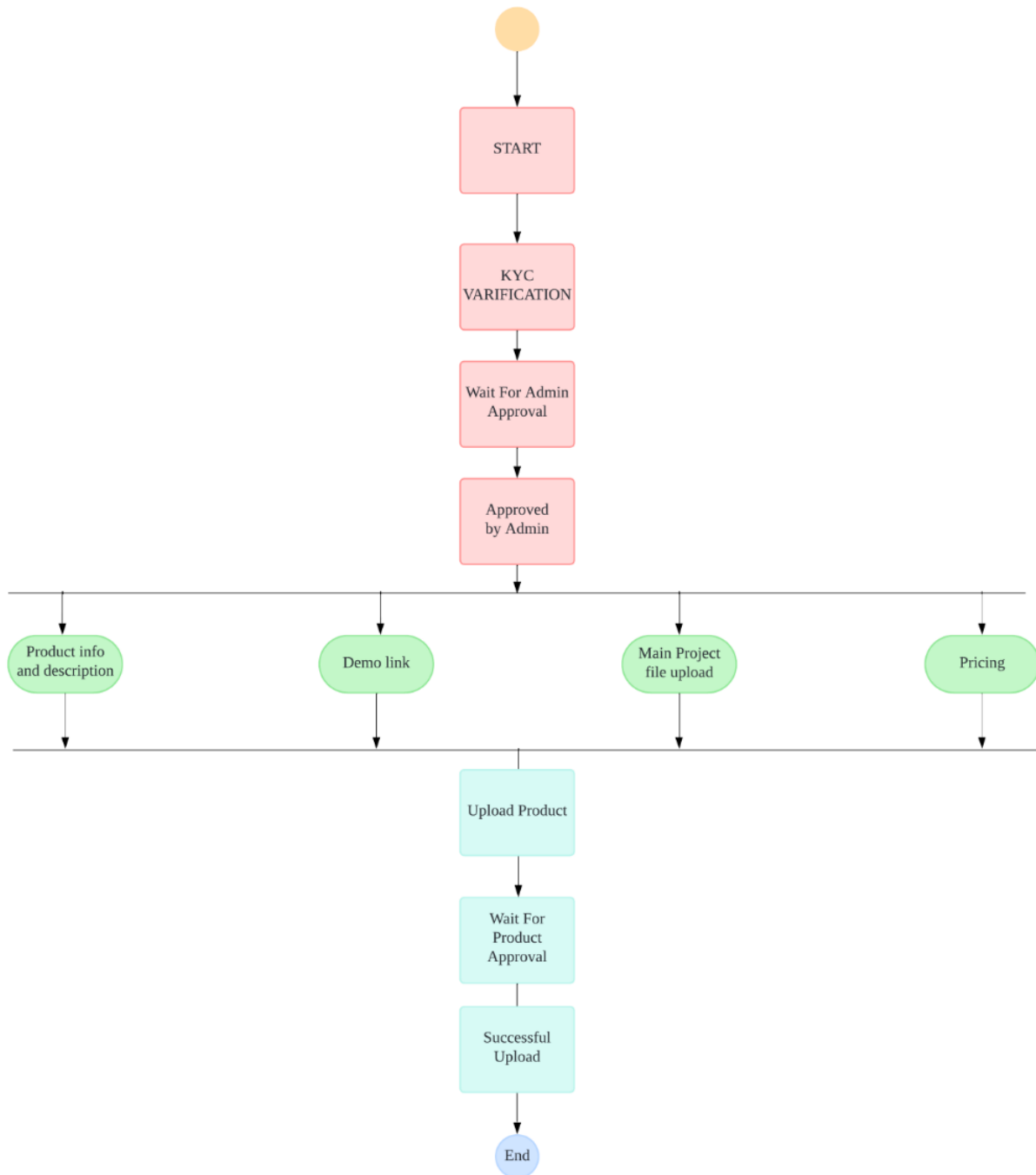


Figure 8-2: Activity Diagram (Uplod Product)

8.1.3 Activity Diagram Part three (Admin Panel):

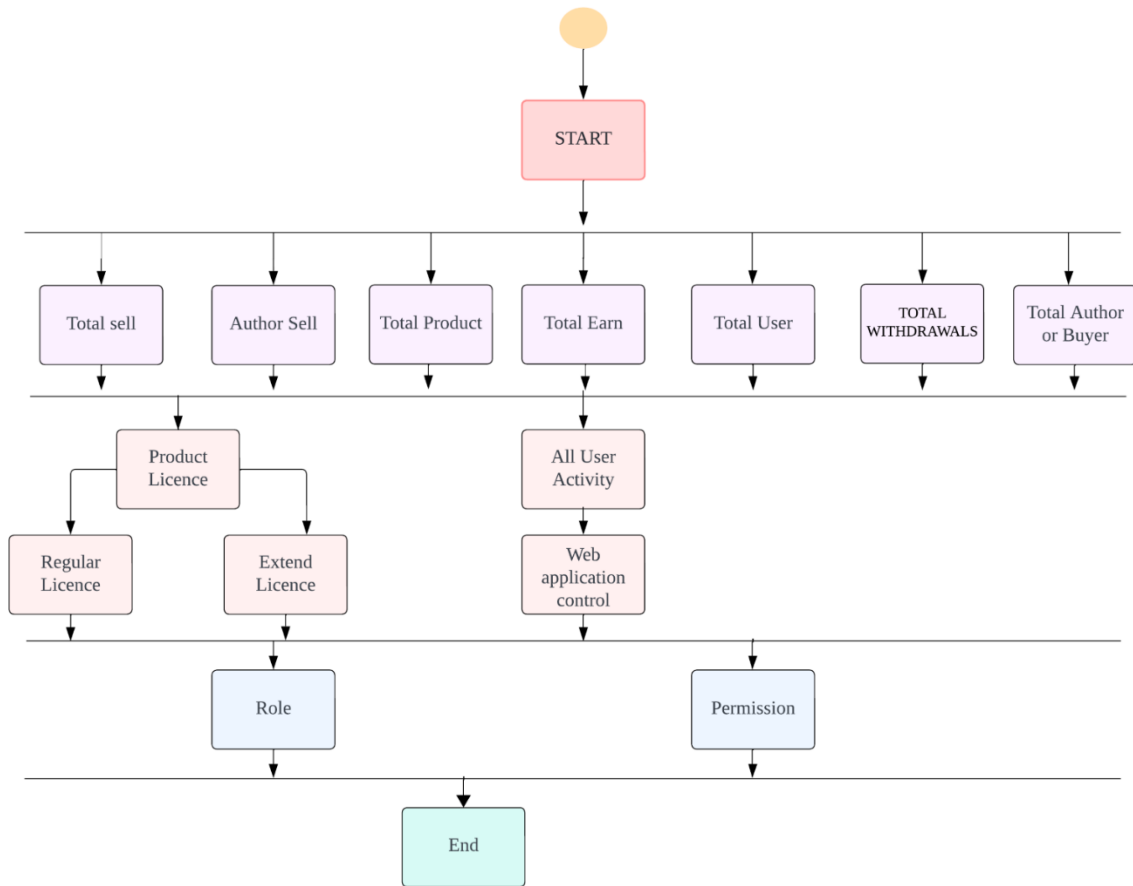


Figure 8-3: Activity Diagram (Admin Panel)

8.1.4 Activity Diagram Part four (Licence validation):

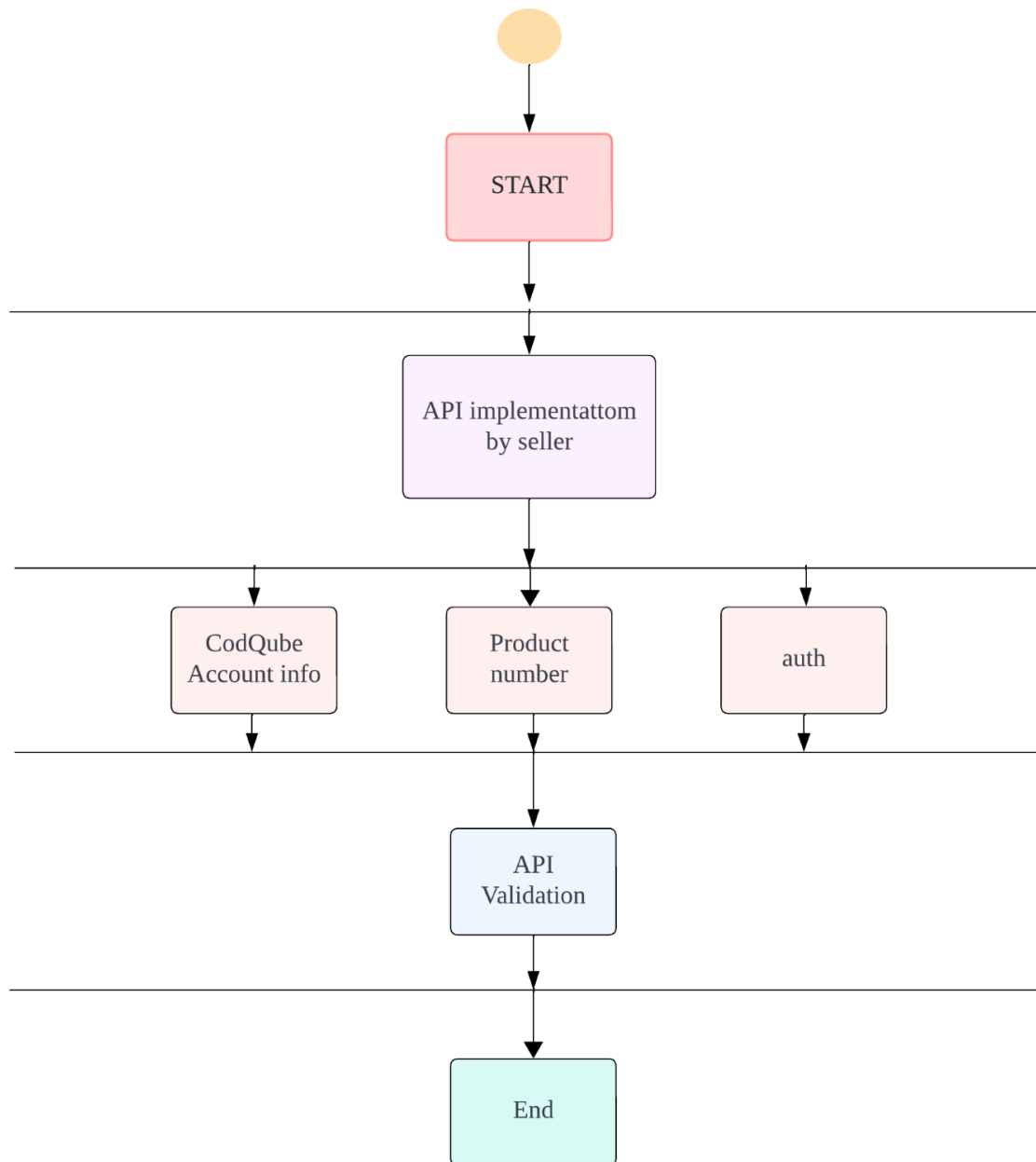


Figure 8-4: Activity Diagram (Licence validation)

8.1.5 Activity Diagram Part five (Product review):

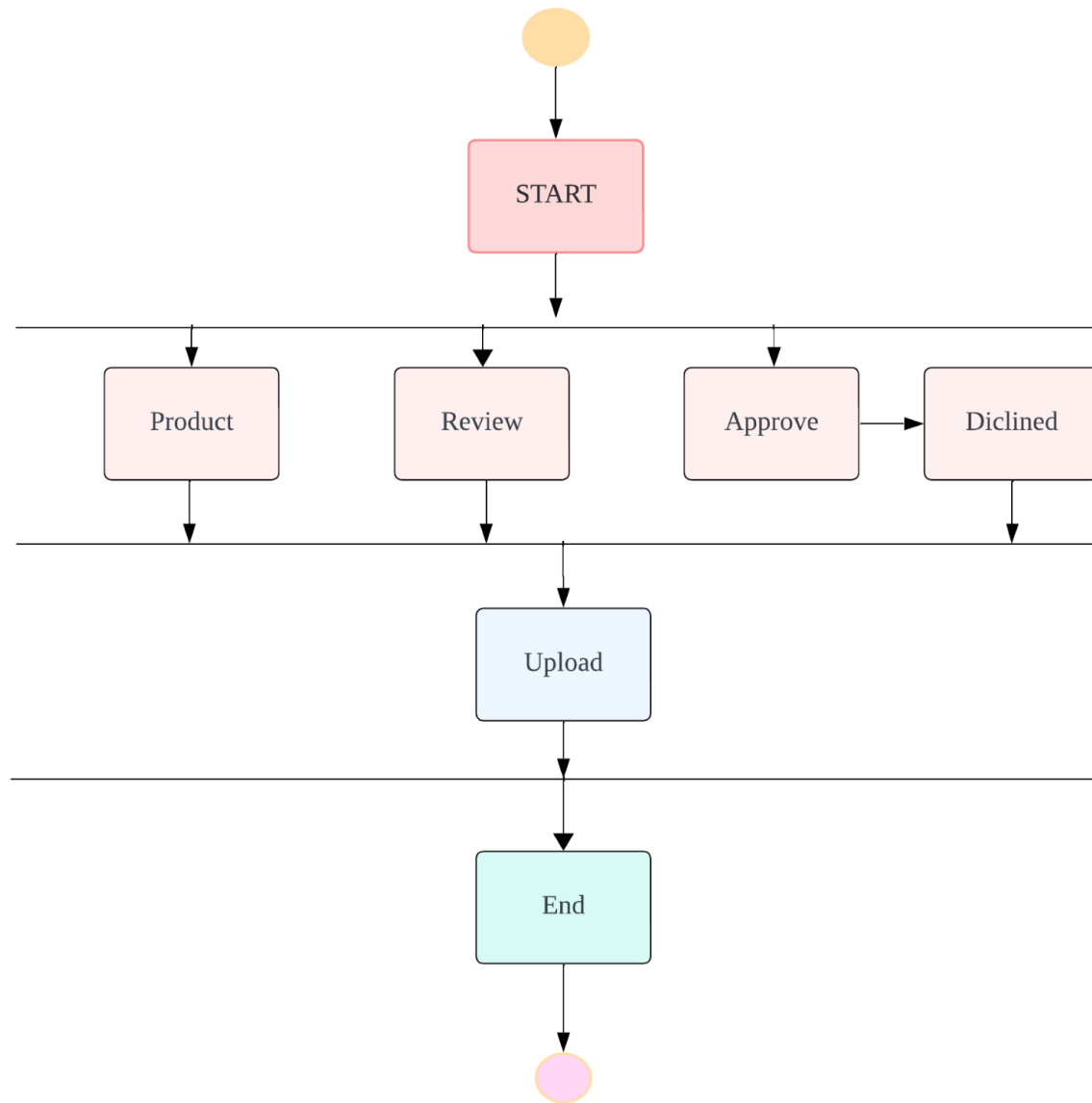


Figure 8-5: Activity Diagram (Product review)

8.2 Use Case of Admin panel:

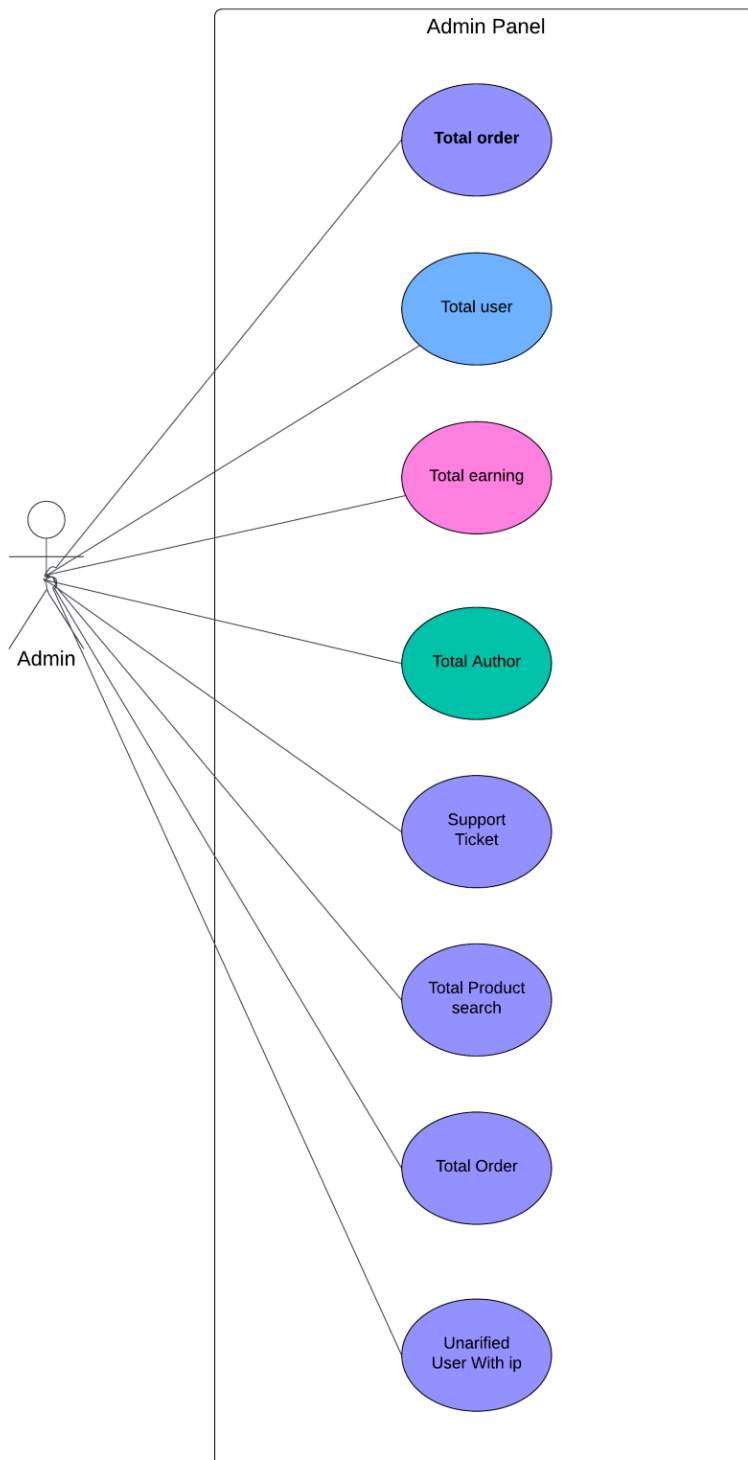


Figure 8-6: Use case (Admin panel)

8.3 Use Case of user panel:

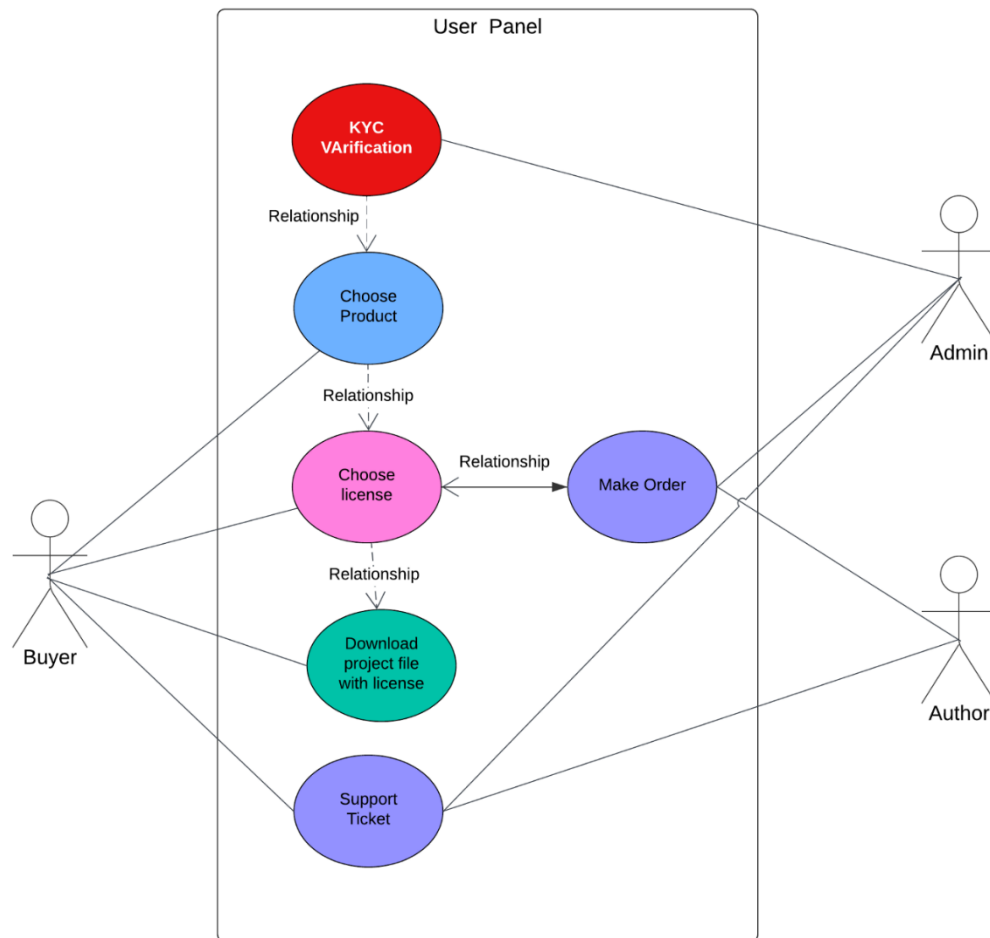


Figure 8-7: Use case (user panel)

8.4 Use Case of Author panel:

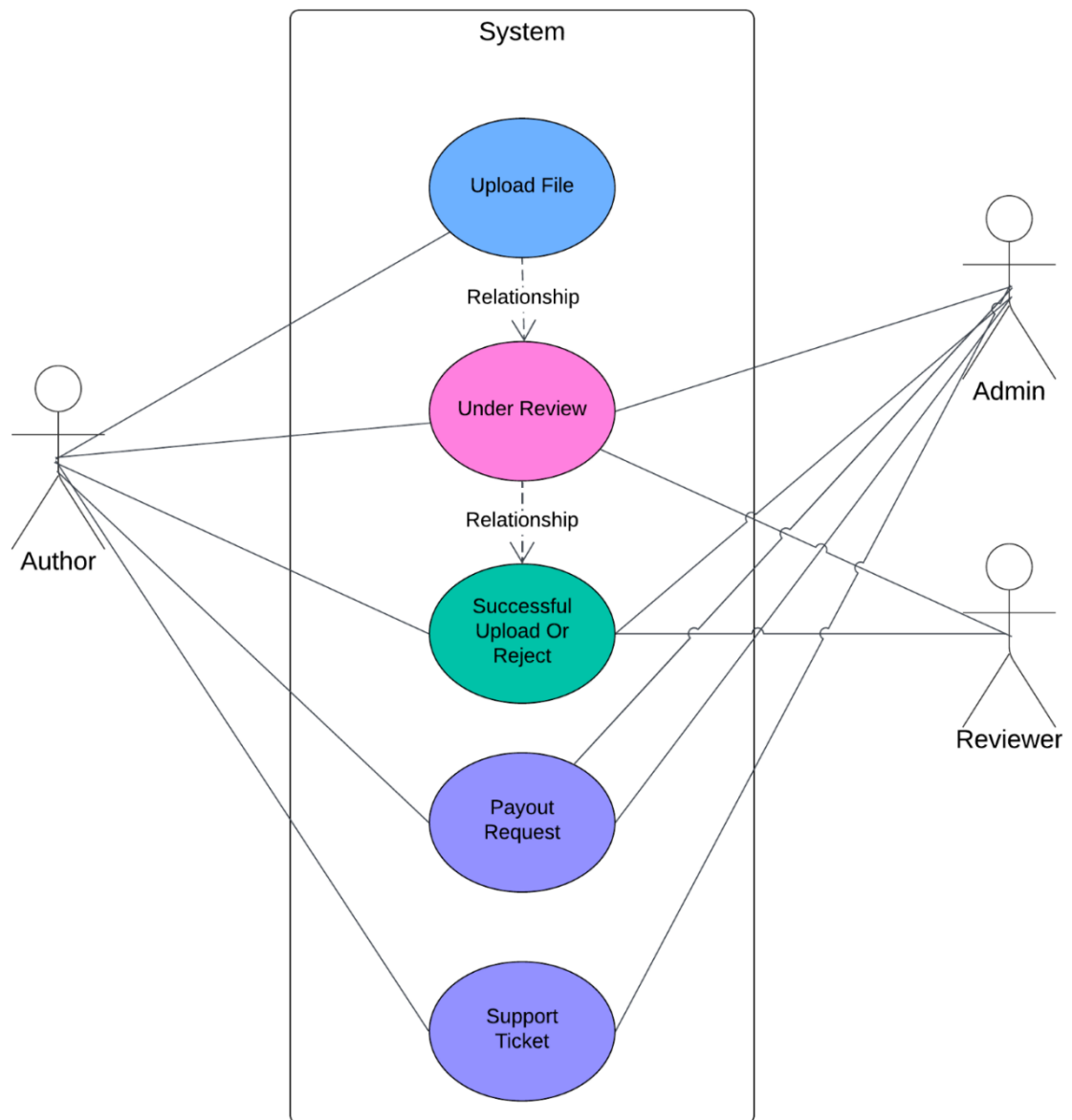


Figure 8-8: Use case (Author panel)

Chapter 9: Exploration

9.1 Module of the system:

Serial Of Action	User Action	System Interaction
User Registration	You can sign up for a new account by entering all the needed information.	Check whether the user enters any data, save it to the SQL database securely, and give them a unique user ID.
Product Search	Browse the searchable collection of digital products.	Get the related products from the SQL database and filter it on certain filters, return them to user with search.
AI-Powered Recommendations	Explore products pertinent to your specific needs with your uniquely generated AI backed product suggestions.	Use the API to return AI-powered product recommendations based on user preferences and historical interactions.
Product Purchase	Select products and cart them to checkout.	Add a total cost, control quantity on hand, add an order id, do payment through secured payment gateway and save orders details in db.
Order Tracking	Order Status and Delivery Info	Ask for the order details from the SQL database to get real-time, up-to-date information about order status, shipping process and expected delivery date.
License Management	Control licensing of digital products purchased.	Validate and handle license update in database using built-in product licensing API.
User Feedback and Reviews	Write feedback, ratings and reviews for items you buy.	Get real user feedback, store ratings and reviews in SQL database, display on product page and iterate for improvements.

Table 8.1.3-1: Table of Module system

9.2 Class diagram of the system:

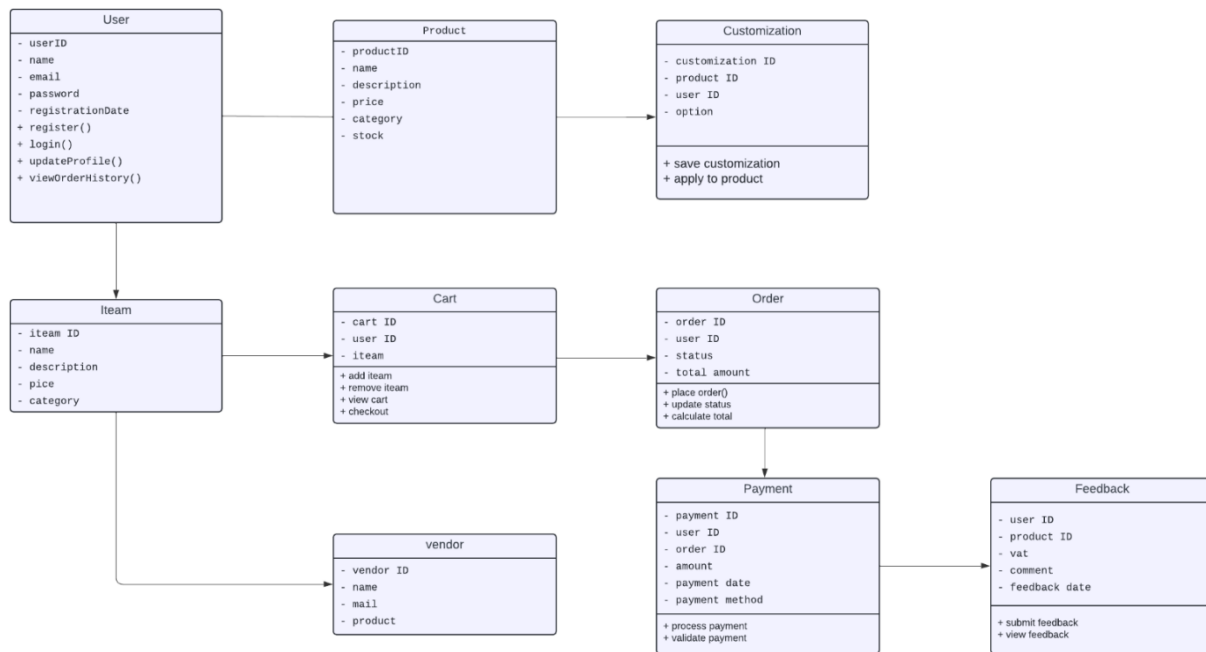


Figure 9-1: Class Diagram

9.3 Sequence diagram of the system:

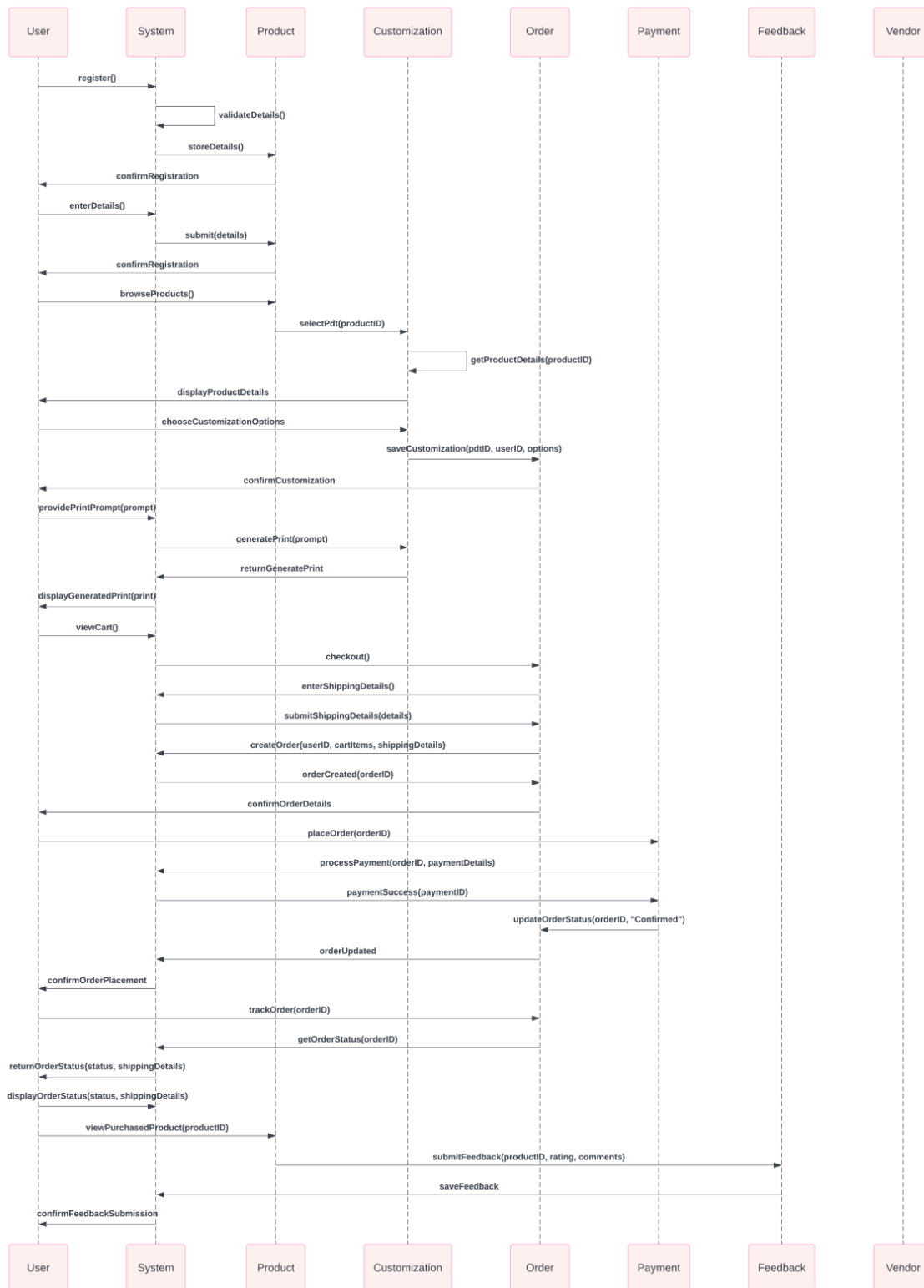


Figure 9-2: Sequence Diagram

9.4 Low-fidelity prototype of the system:



Figure 9-3: Low fidelity prototype (home page)



Figure 9-4: Low fidelity prototype (login page)

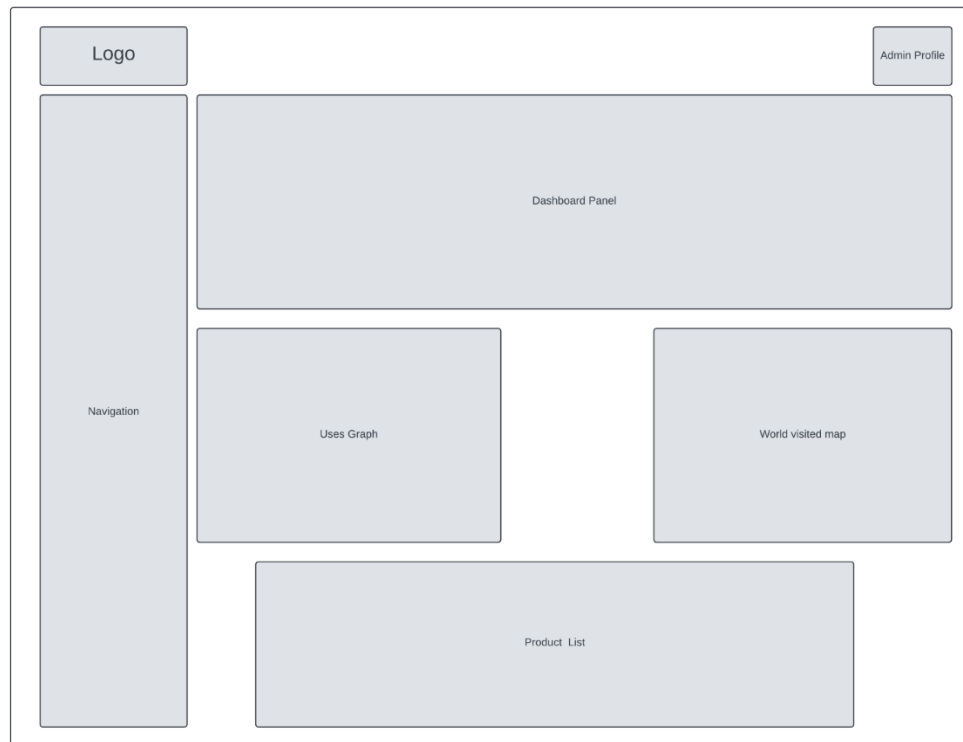


Figure 9-5: Low fidelity prototype (Admin Panel)

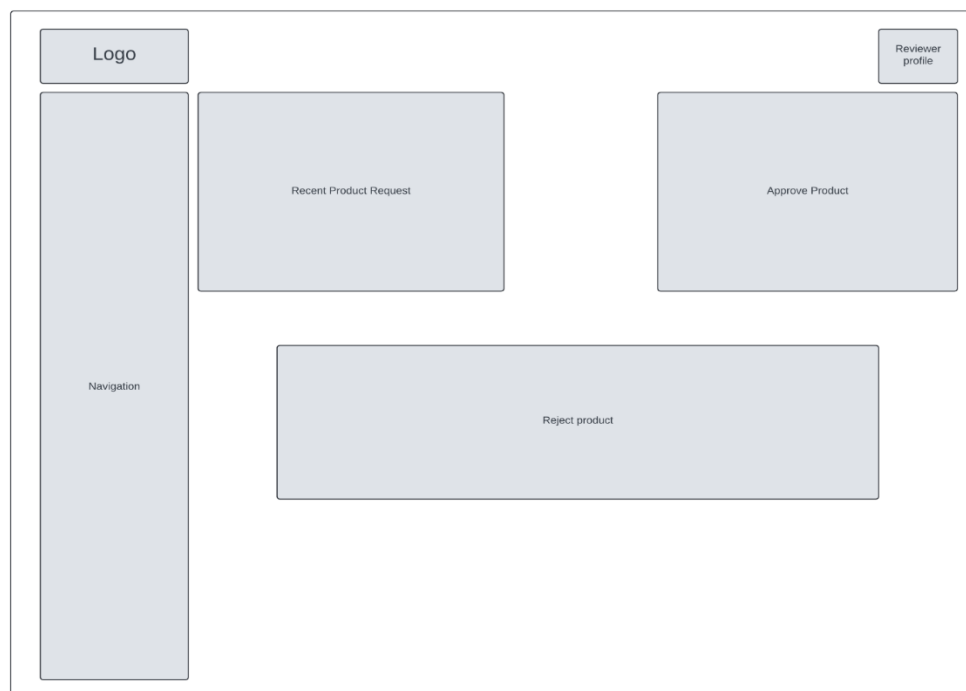


Figure 9-6: Low fidelity prototype (Reviewer Panel)

9.5 High-fidelity prototype of the system:

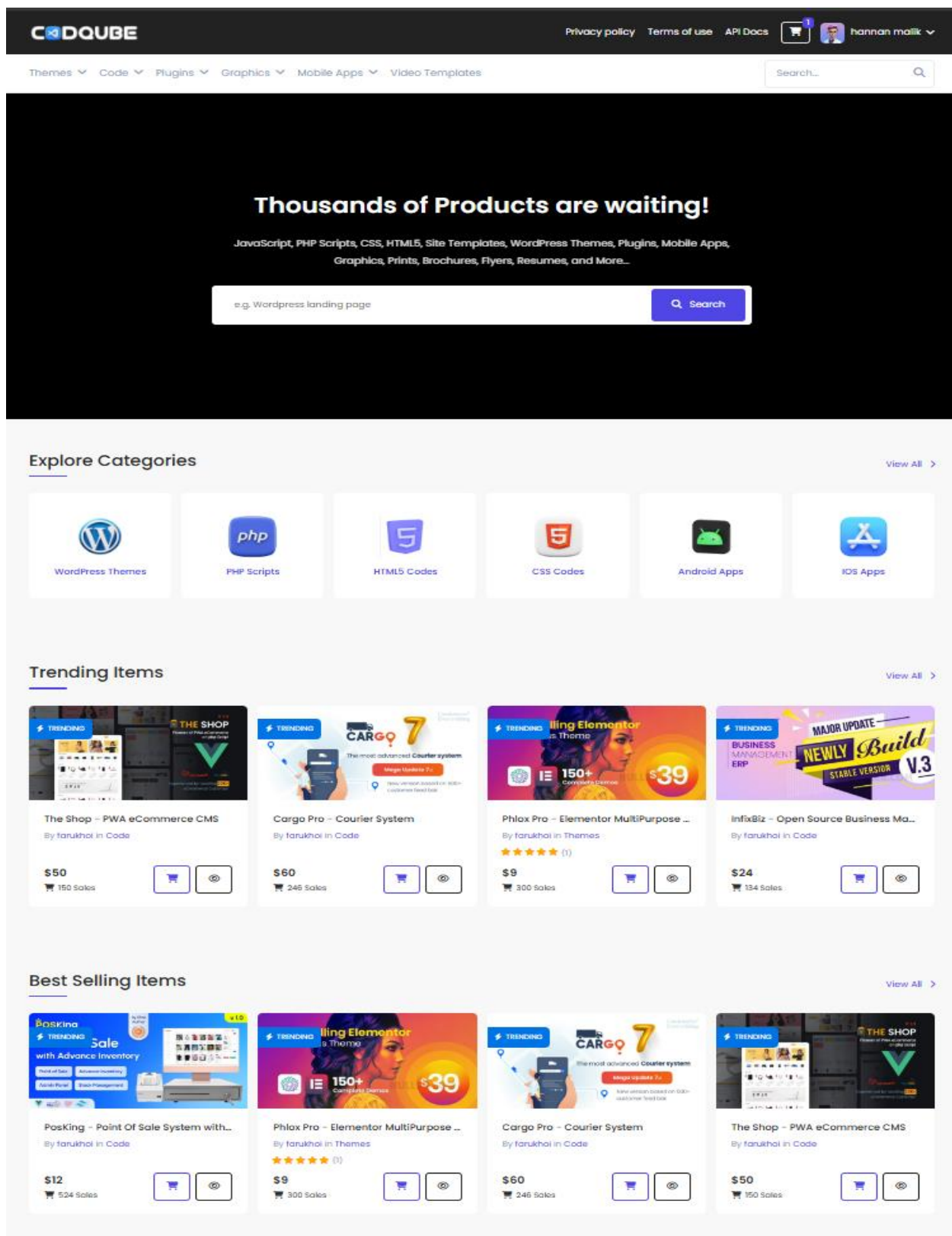


Figure 9-7: High Fidelity prototype (Home page)

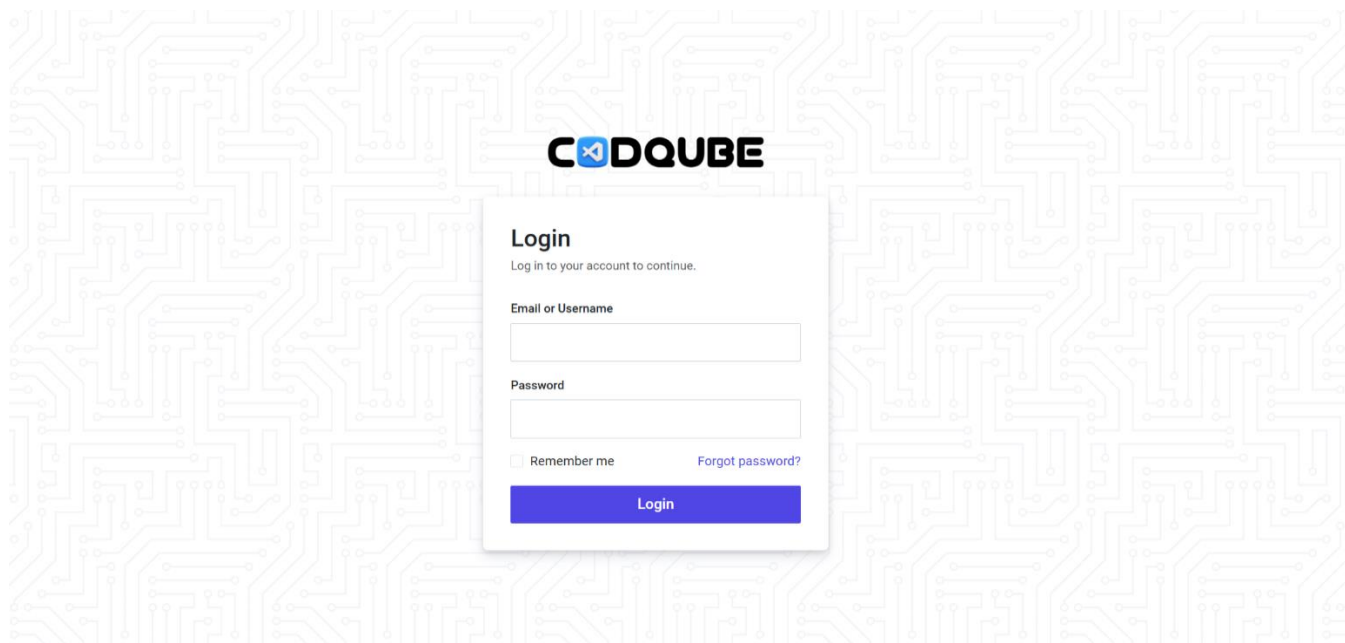


Figure 9-8: High Fidelity prototype (Login page)

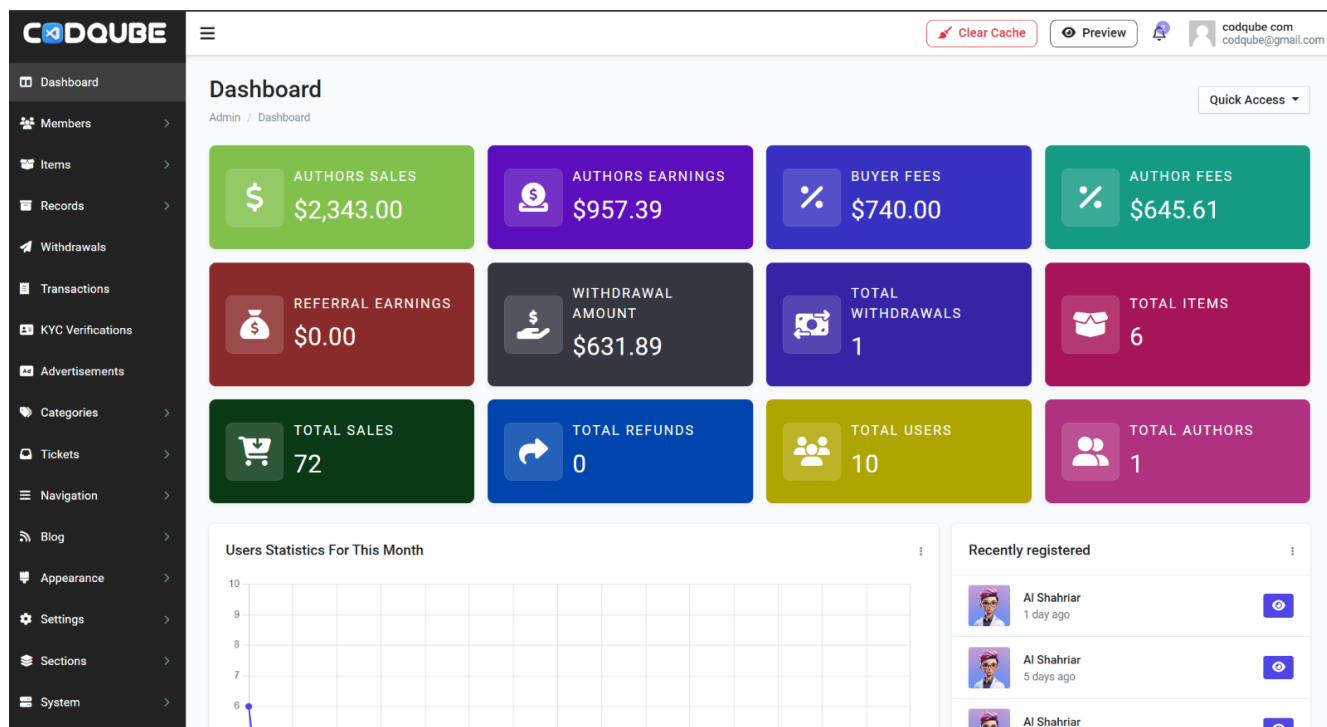


Figure 9-9: High Fidelity prototype (Admin Panel)

- Dashboard
- Members
- Items
- Records
- Withdrawals
- Transactions
- KYC Verifications
- Advertisements
- Categories
- Tickets
- Navigation
- Blog
- Appearance
- Settings
- Sections
- System

Clear Cache

Preview

codqube.com
codqube@gmail.com

Transactions

Admin / Transactions

PENDING (0)
\$0.00

PAID (11)
\$2,353.68

CANCELLED (0)
\$0.00

From Date

To Date

Payment Method

Status

Q

Reset

ID	User	SubTotal	Tax	Fees	Total	Status	Date
# 1023	kabir1699	\$225.00	\$0.00	\$0.00	\$225.00	Paid	Jul 08, 2024 02:18 PM
# 1018	shompouk	\$204.00	\$0.00	\$0.00	\$204.00	Paid	Jul 01, 2024 08:52 PM
# 1017	shompouk	\$360.00	\$0.00	\$0.00	\$360.00	Paid	Jul 01, 2024 08:51 PM
# 1016	shompouk	\$750.00	\$0.00	\$0.00	\$750.00	Paid	Jul 01, 2024 08:50 PM
# 1015	shompouk	\$225.00	\$0.00	\$0.00	\$225.00	Paid	Jul 01, 2024 08:50 PM
# 1014	shompouk	\$36.00	\$0.00	\$0.00	\$36.00	Paid	Jul 01, 2024 08:48 PM
# 1013	shompouk	\$9.00	\$0.00	\$0.00	\$9.00	Paid	Jul 01, 2024 08:46 PM
# 1010	danliam	\$24.00	\$0.00	\$0.48	\$24.48	Paid	Jul 01, 2024 10:25 AM

Figure 9-10: High Fidelity prototype (Reviewer Panel)

Chapter 10: Development

10.1 Folder structure of the system:

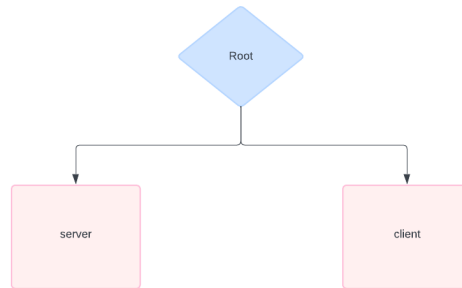


Figure 10-1: Root folder

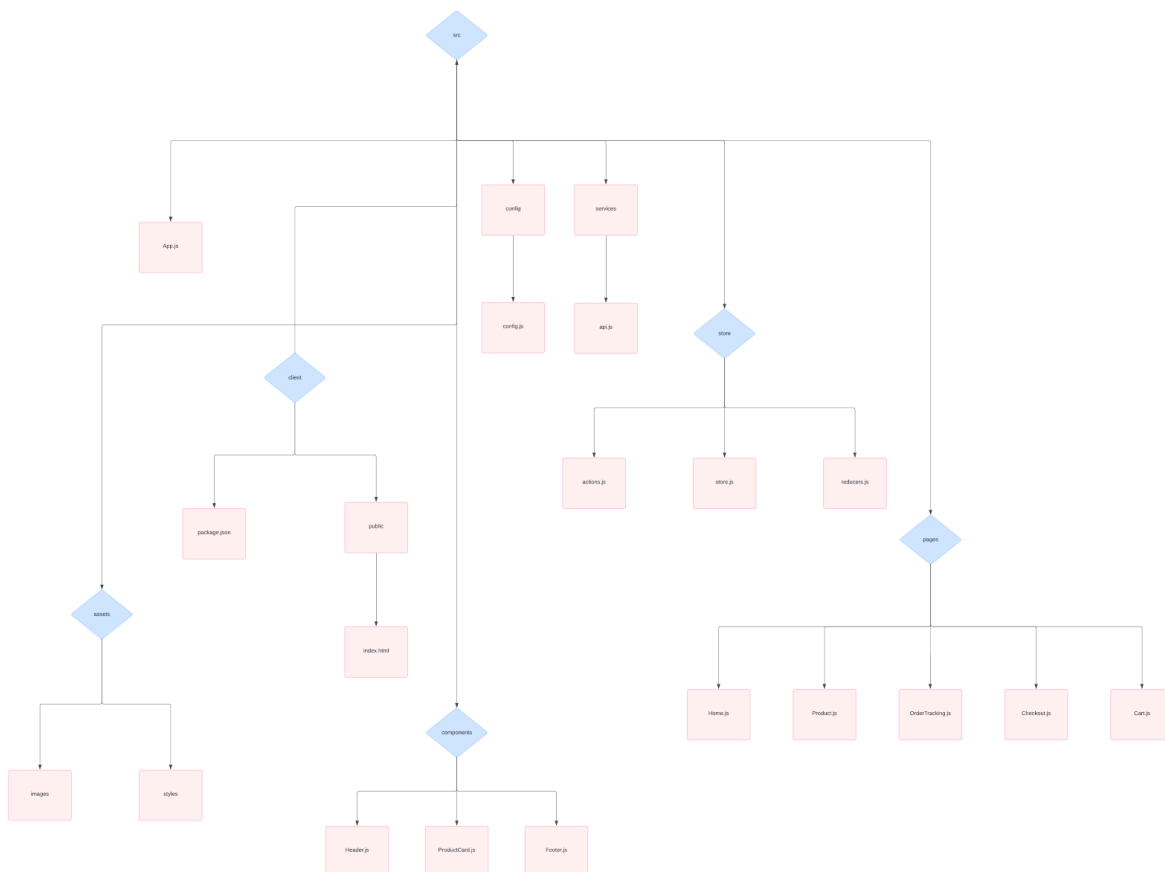


Figure 10-2: Client-side Folder structure

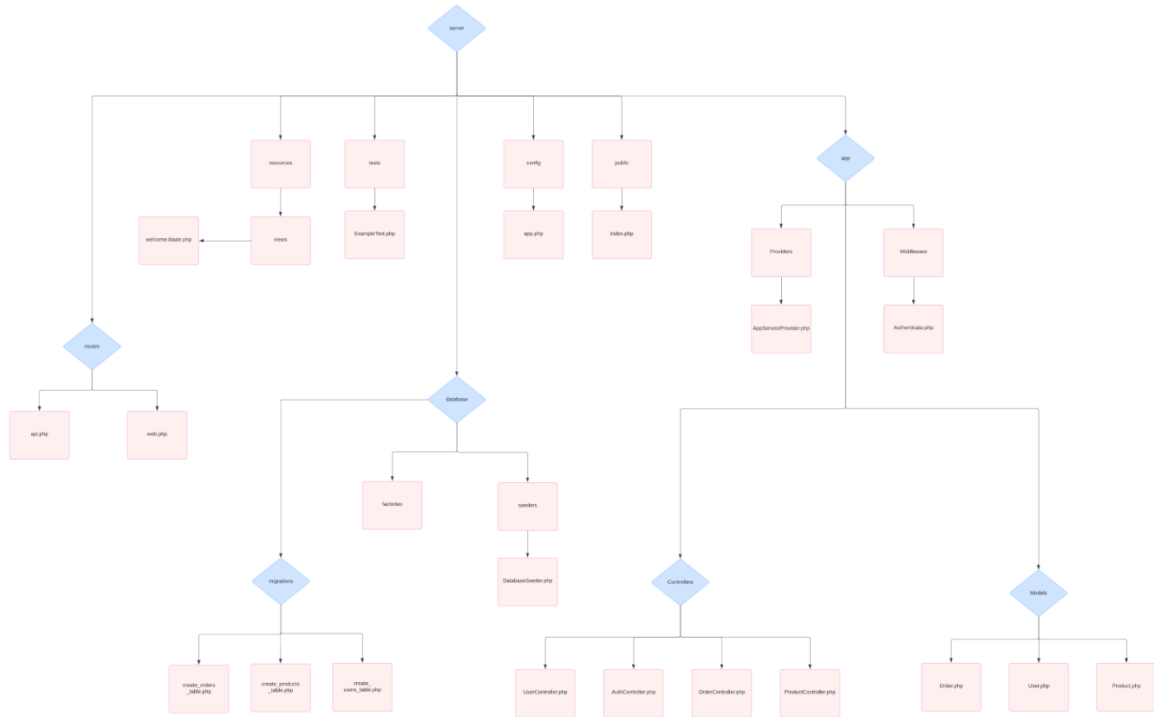


Figure 10-3: Server-side folder structure

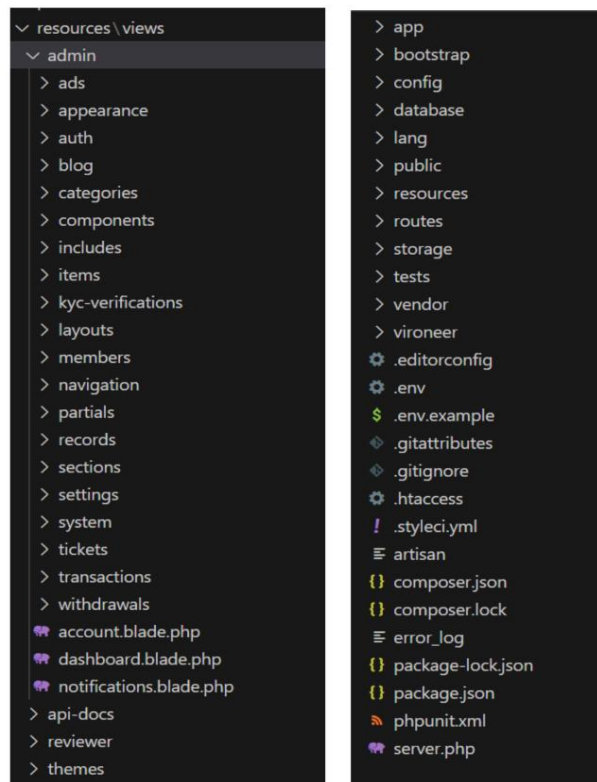


Figure 10-4: Actual-side folder structure

10.2 Core module output sample:

The image shows a high-fidelity prototype of a product page for 'Cargo Pro - Courier System' on the CodQube marketplace. The page layout includes a top navigation bar with the CodQube logo, links for Privacy policy, Terms of use, API Docs, and a user profile for Daniel Cruz. Below the navigation bar is a search bar and a category filter showing 'Themes', 'Code', 'Plugins', 'Graphics', 'Mobile Apps', and 'Video Templates'.

The main content area features a large banner for 'Cargo Pro - Courier System' with a 'Mega Update 7.x' badge. The banner text includes 'The most advanced Courier system' and 'New version based on 500+ customer feedback'. To the right of the banner is a 'License Option' section with two choices: 'Regular' for one project at \$60 and 'Extended' for unlimited projects at \$225. Below the license options are buttons for 'Add to Cart' and 'Buy Now', along with a list of benefits: 'Quality checked by CodQube', 'Future updates', and 'Free support'.

Below the banner is a 'Description' tab and a 'Comments (2)' tab. The description text states: 'Cargo 7.7 - 10 Jan 2024 is live! Our latest release update!'. It lists 'Added' features (Cairo timezone, new regions, verification, compulsory input, hints, and a weight change note) and 'Fixes' (uploading images, shipping prices, SMS notifications, shipping calculator, country code, French translation, total weight, tracking issue, phone number, and table permissions).

At the bottom of the main content area is a large graphic with the 'Cargo Pro 7' logo and the text 'Big Update! So much MORE 300+ Features'. To the right of the main content area is a user profile for 'farukhol' (Member since Jun 2024) with a 'View Portfolio' button. Below the profile is a '246 Sales' section with a table of product details:

Published:	Jun 30, 2024 05:42 PM
Version:	v7.8
Category:	Code / PHP Scripts
High Resolution:	Yes
Compatible Browsers:	Firefox, Chrome, Safari, Opera, Edge
Files Included:	HTML Files, Javascript, CSS, XML, Sass, PHP, SQL
Frameworks:	Laravel
Tags:	Cargo Pro, courier,

At the bottom of the right sidebar is a 'Share' section with social media icons for Facebook, X, Telegram, WhatsApp, and Email.

Figure 10-5: High Fidelity prototype (Product page)

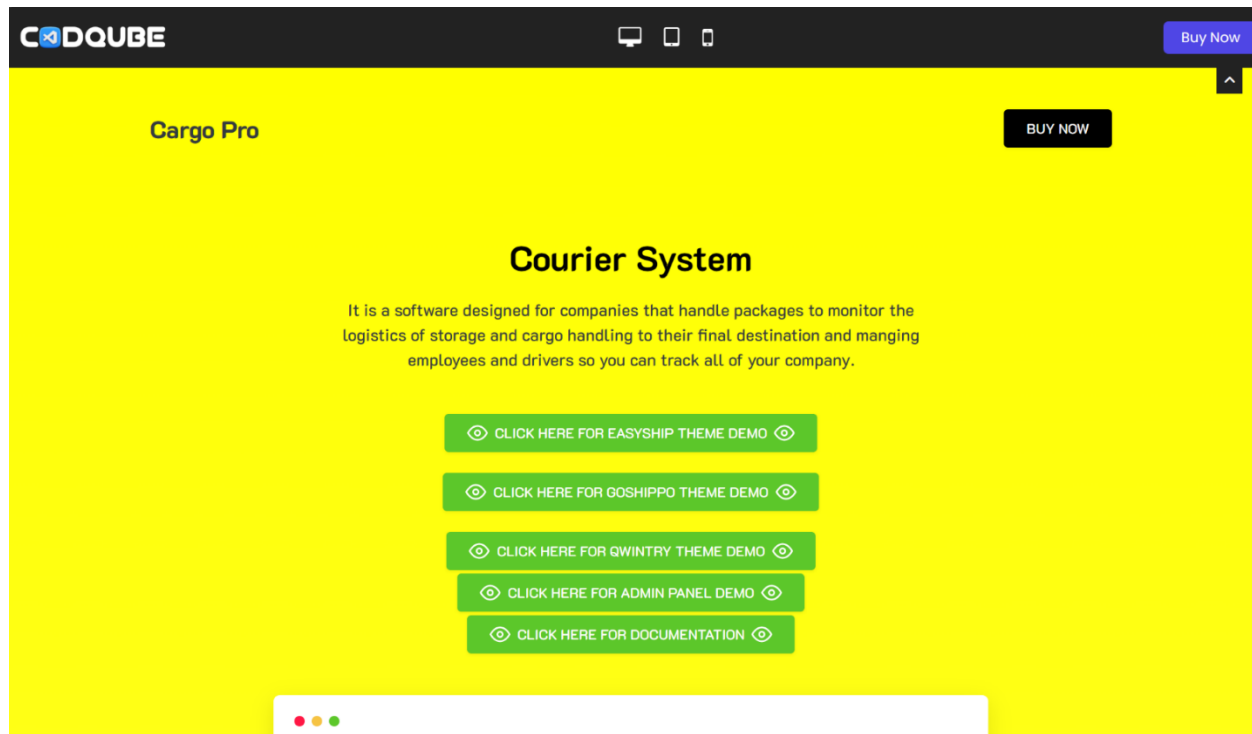


Figure 10-6: High Fidelity prototype (Product Preview)

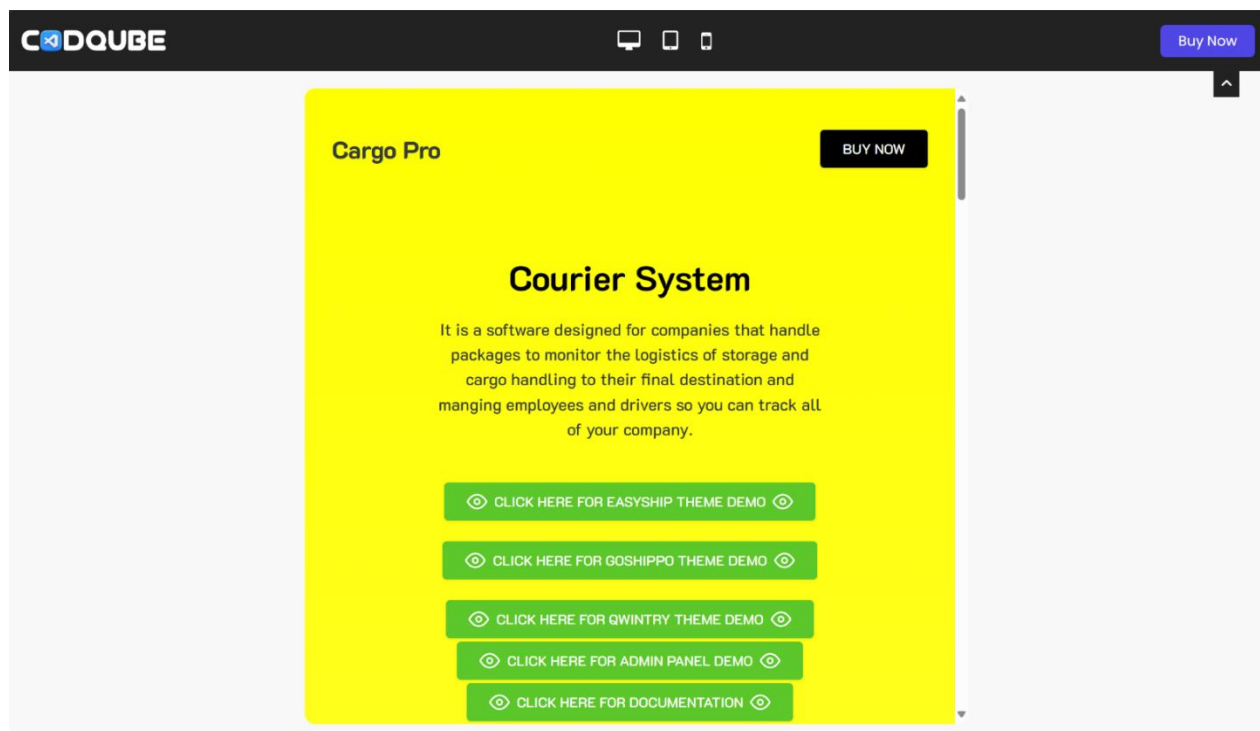


Figure 10-7: High Fidelity prototype (Product mobile view)

The image shows a high-fidelity prototype of the KYC verification page in the CODQUBE application. The interface includes a dark sidebar on the left with navigation links: Balance (\$0.00), Purchases, Transactions, Statements, Refunds, Tickets, Settings (highlighted), and Logout. The main content area features a red banner at the top stating "KYC Verification Required" with a "Complete KYC" button. Below this is a "Settings" section with a breadcrumb "Workspace > Settings > KYC Verification" and a horizontal menu with options: Account Details, Profile details, My Badges, API Key, Change Password, 2FA Authentication, and KYC Verification (highlighted). The "ID Verification" section contains instructions to upload a clear, legible image. It includes a "Document Type" dropdown menu set to "National ID" and a "National ID Number" input field. To the right, there are two image upload areas: "Front Of ID" and "Back Of ID", each with a "Choose file" button and a "No file chosen" status.

Figure 10-8: High Fidelity prototype (KYC verification page)

The image shows a high-fidelity prototype of the API Documentation Overview page in the CODQUBE application. The interface features a dark sidebar on the left with navigation links: Introduction (Overview, Authentication), Account (Get Account Details), Items (Get All Items, Get An Item Details), and Purchases (Purchase Validation). The main content area is titled "API Documentation Overview" and lists four API endpoints:

- 1. Get Account Details**
Endpoint: GET <https://codqube.com/api/account/details>
Description: Retrieves details of the account associated with the provided API key.
- 2. Get All Items**
Endpoint: GET <https://codqube.com/api/items/all>
Description: Retrieves all items associated with the provided API key.
- 3. Get An Item Details**
Endpoint: GET <https://codqube.com/api/items/item>
Description: Retrieves details of a specific item based on the provided item ID and API key.
- 4. Purchase Validation**
Endpoint: POST <https://codqube.com/api/purchases/validation>
Description: Validate a purchase code and returns details about the purchase if valid.

Figure 10-9: High Fidelity prototype (Overview page)

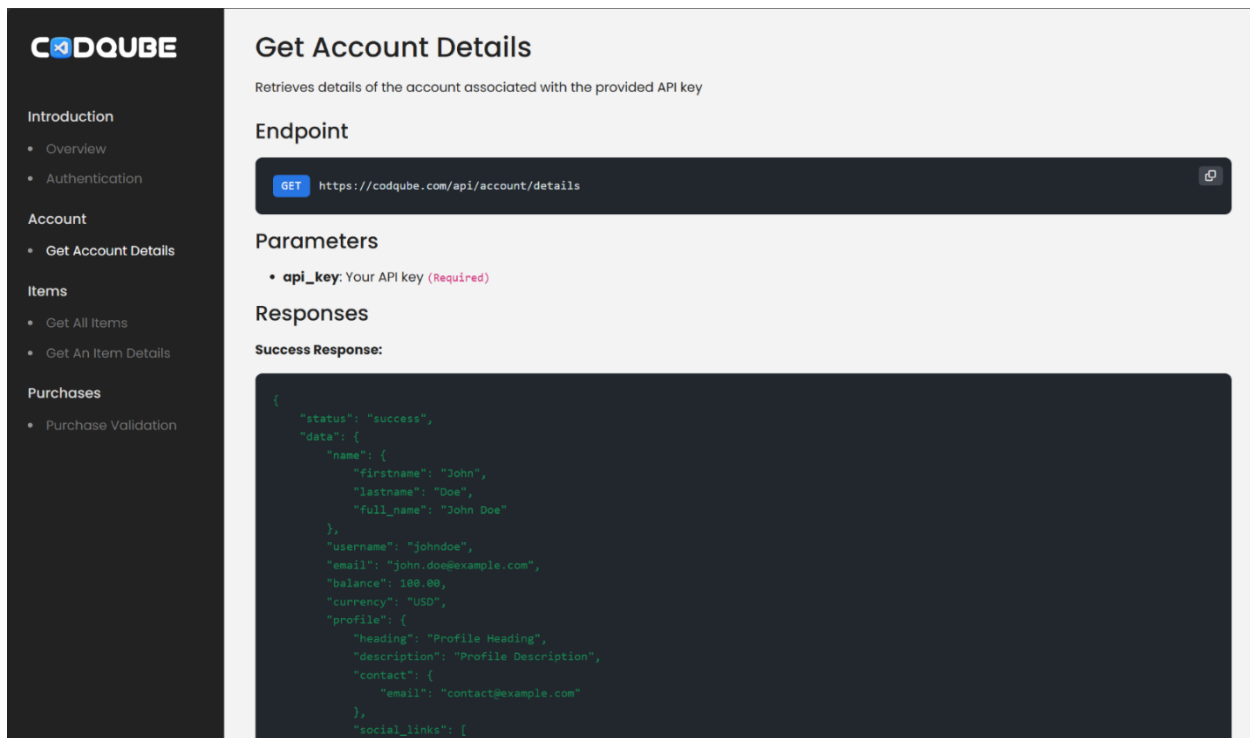


Figure 10-10: High Fidelity prototype (Account details page)

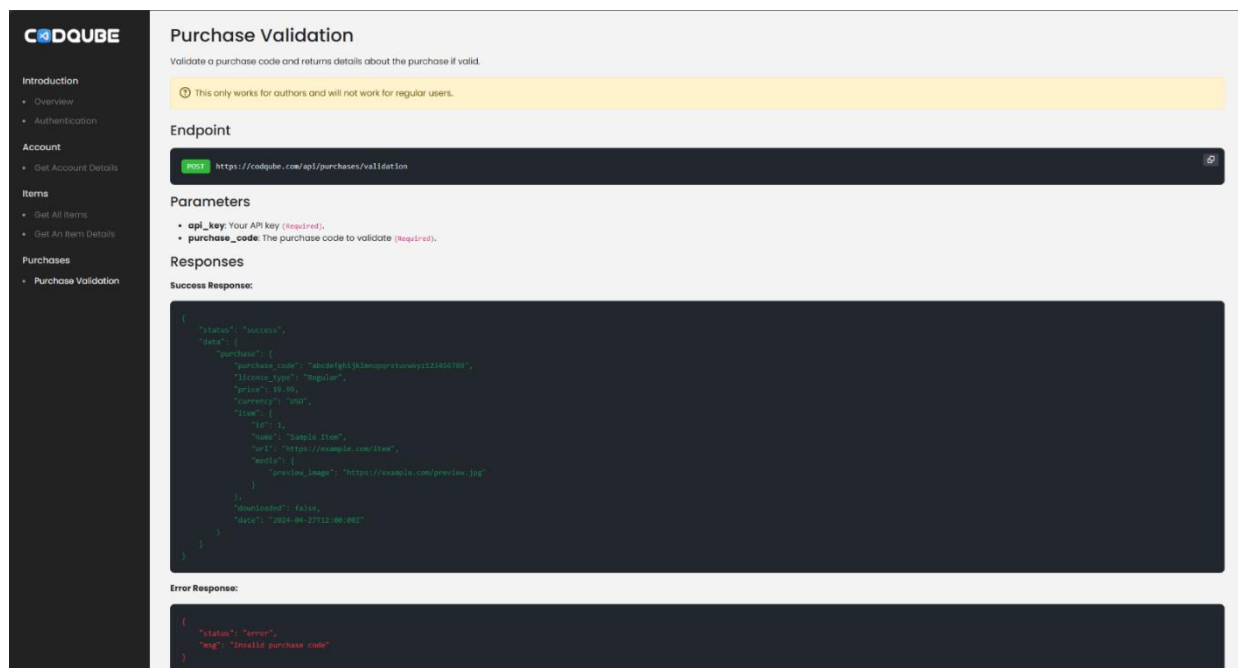


Figure 10-11: High Fidelity prototype (Purchase page)

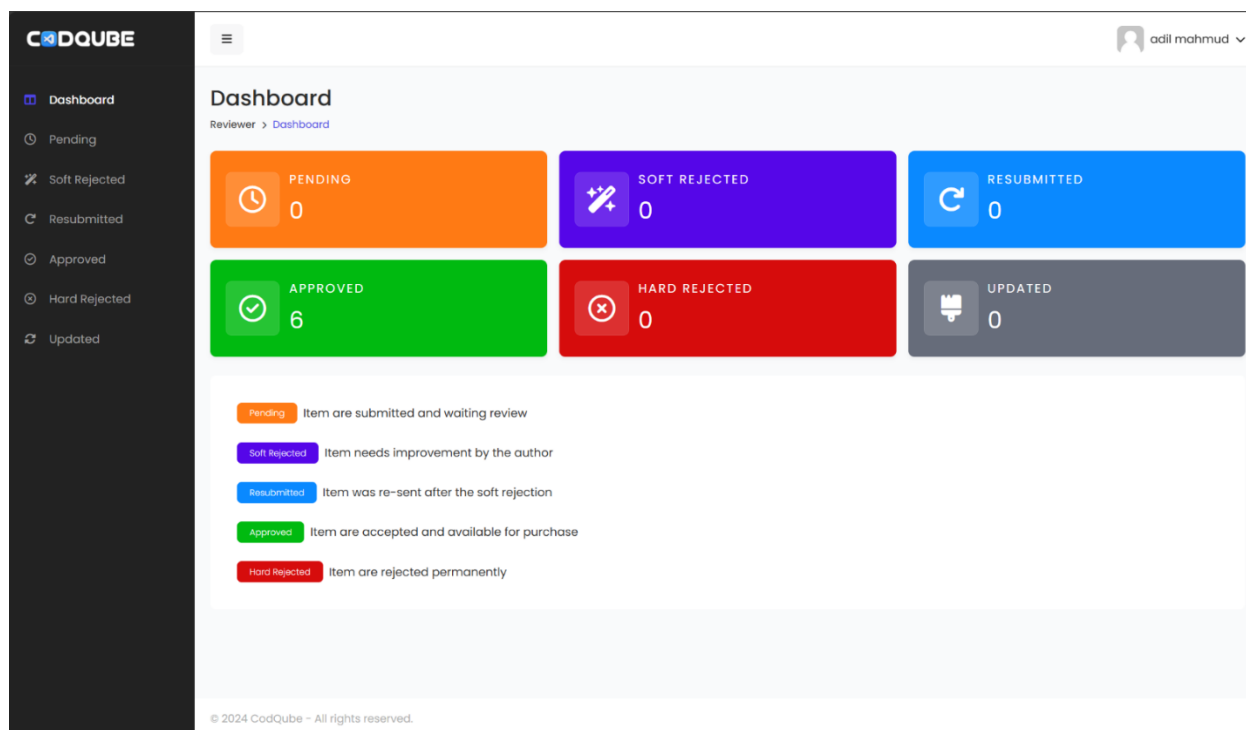


Figure 10-12: High Fidelity prototype (Admin page)

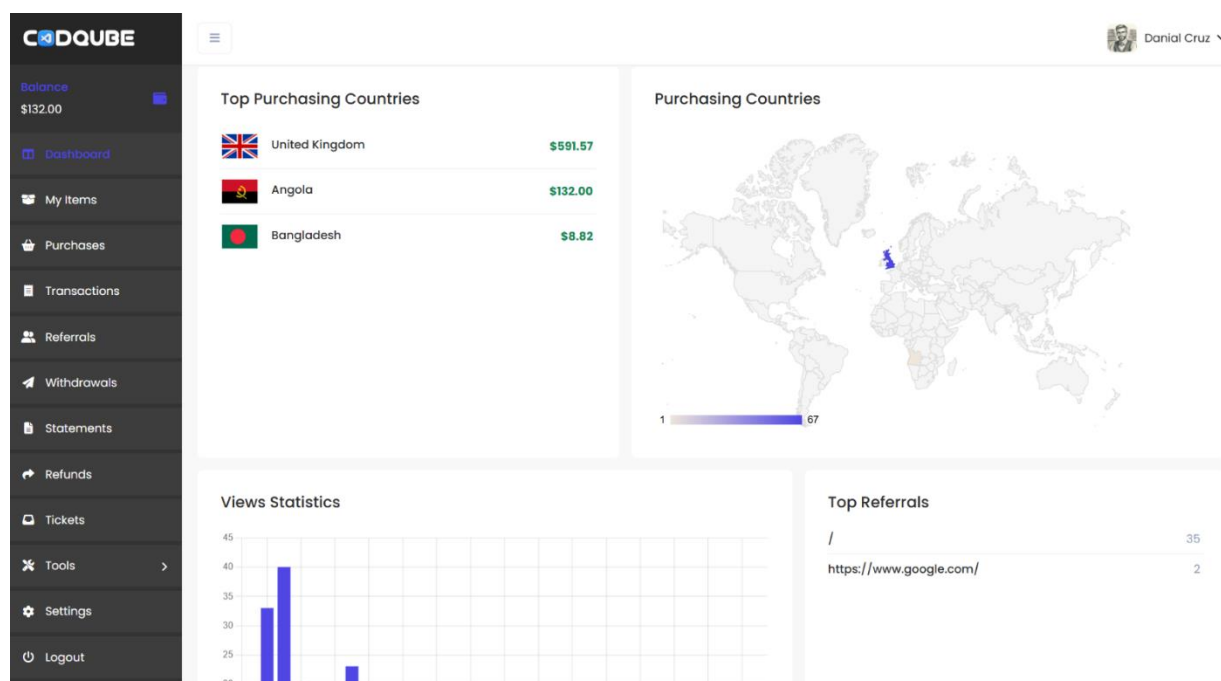


Figure 10-13: High Fidelity prototype (Dashboard page)

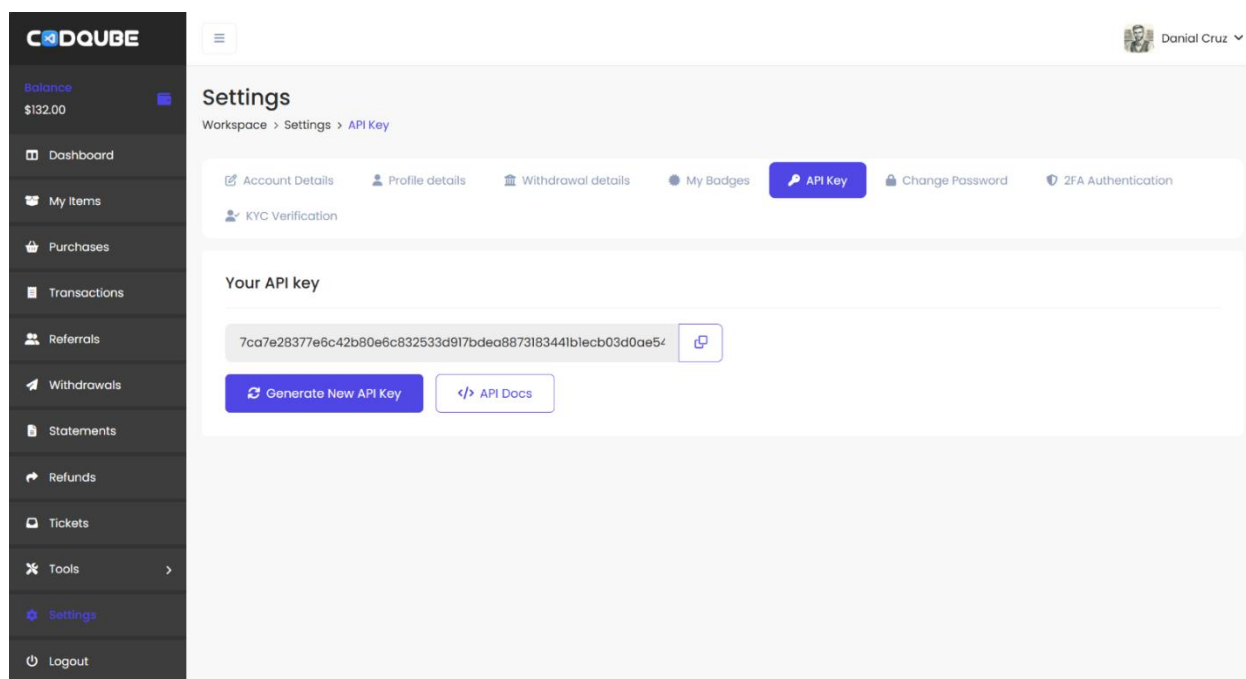


Figure 10-14: High Fidelity prototype (Setting page)

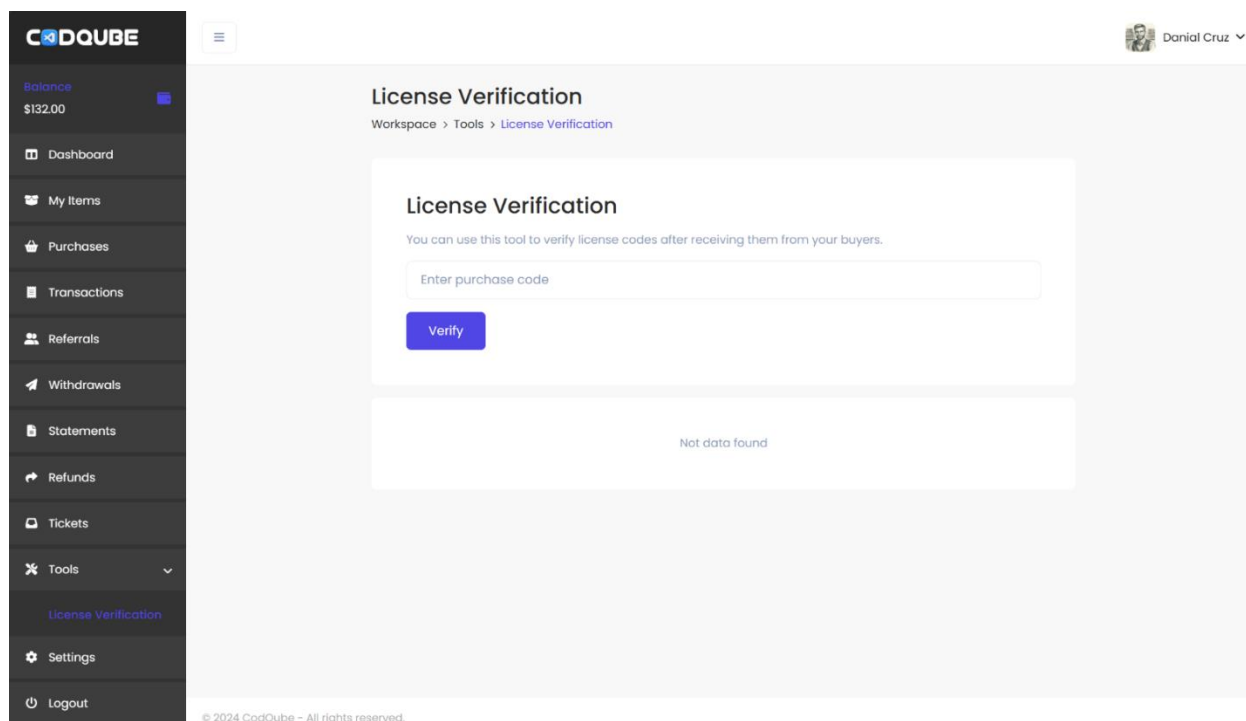


Figure 10-15: High Fidelity prototype (License verification page)

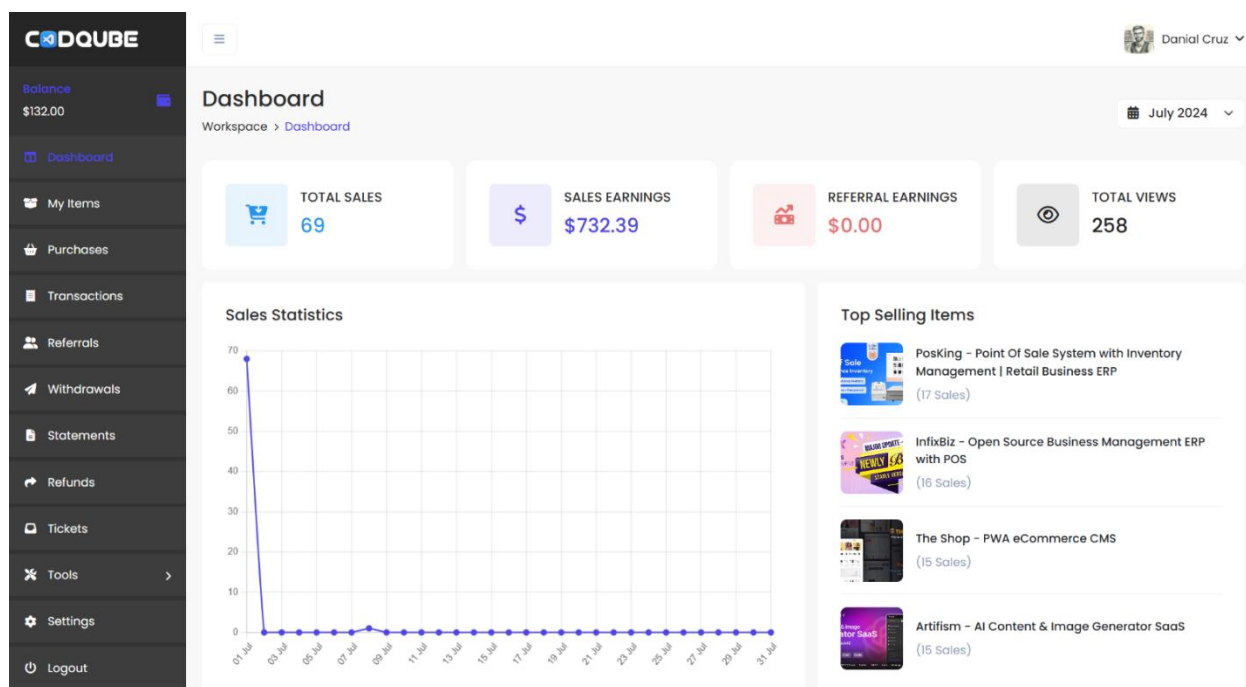


Figure 10-16: High Fidelity prototype (Workspace page)

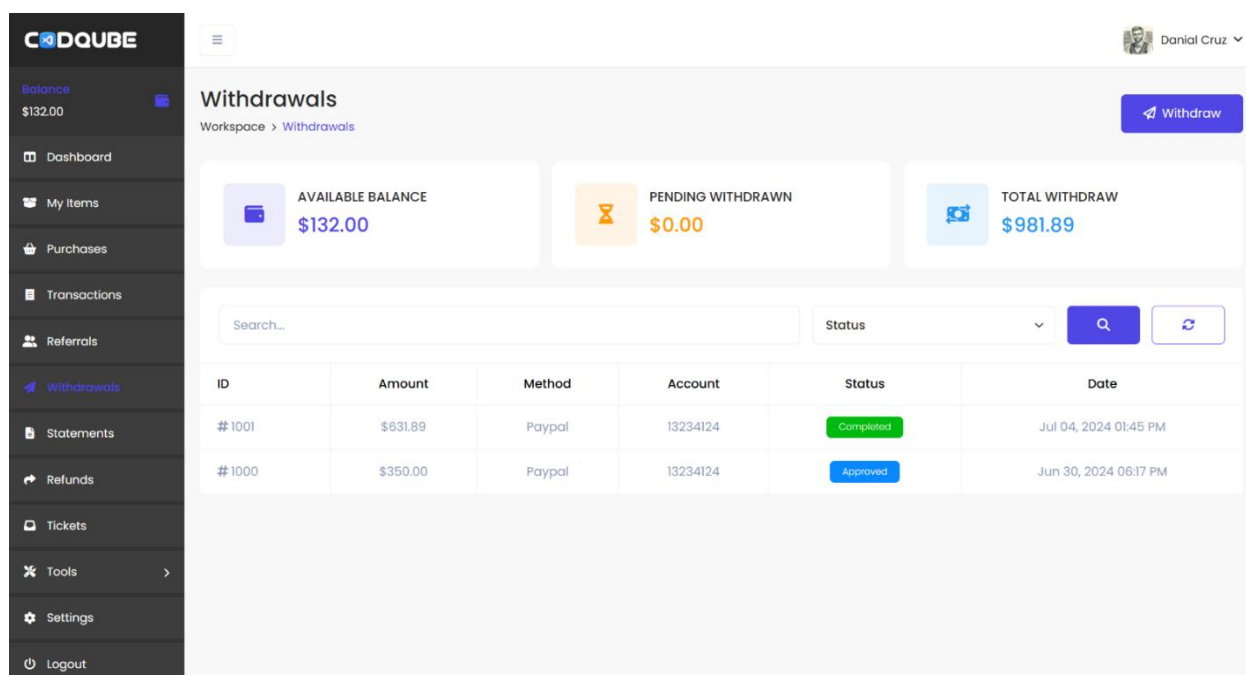


Figure 10-17: High Fidelity prototype (Withdrawals page)

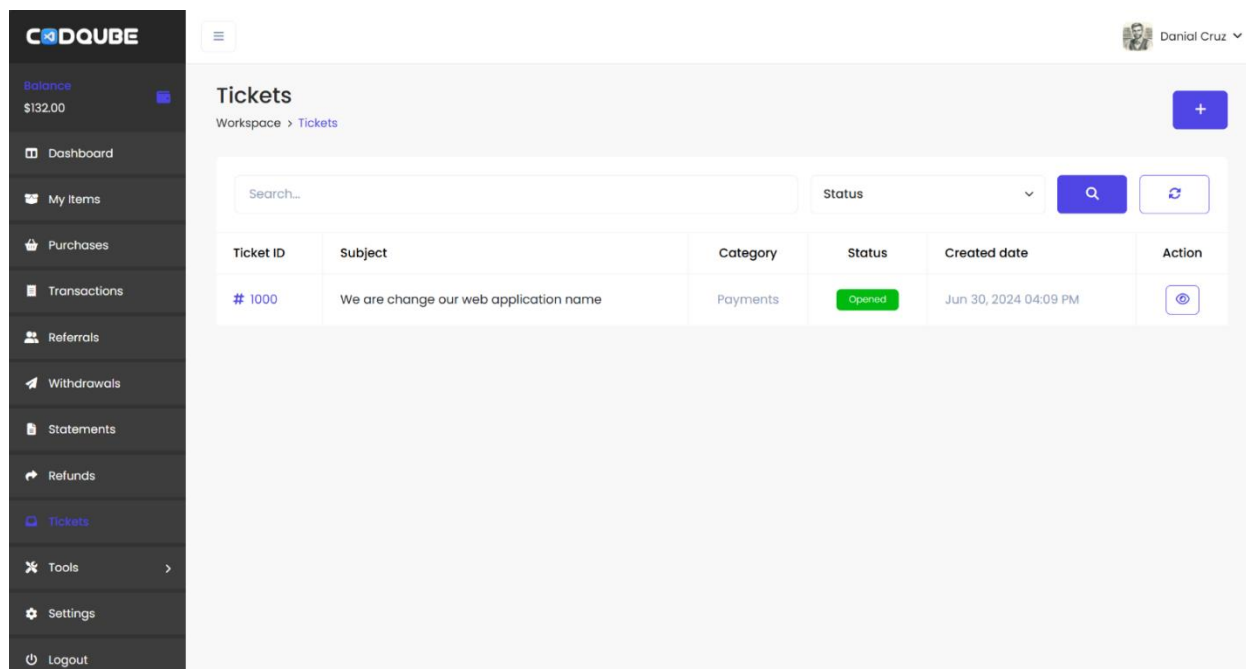


Figure 10-18: High Fidelity prototype (Ticket page)

10.3 Third-Party Package Module

Third-party modules are integral to software development since they offer ready-made libraries or frameworks developed by external sources. Pros: Time and cost savings, more features, the community around it, and geared toward core tasks. When third-party packages are integrated in project by a developer this ultimately help full to reduce development time, reliable and well tested solution for their work, Additional inch added into the software functionality. Using third-party packages helps in fast development and deliver good quality software to the end user. Below is a list of third-party packages used in CodQube with PHP and Laravel:

laravel/passport: Used for API authentication, In just a few minutes, it gives your Laravel application a complete OAuth2 server implementation.

laravel/socialite: A package that makes it easy to handle OAuth authentication with a variety of providers such as Facebook, Twitter, Google, and LinkedIn.

laravel/telescope: Provides a dashboard for monitoring requests, exceptions, database queries, variable dumps, scheduled tasks, cache operations, mail, notifications, queued jobs, and more.

barryvdh/laravel-debugbar: Integrates PHP Debug Bar with Laravel, providing detailed debug information and profiling for the application.

fideloper/proxy: Helps to handle the trusted proxies that are sending traffic to the application, especially useful when deploying the application on cloud services.

spatie/laravel-permission: Provides a robust permission and role management system for Laravel applications.

nesbot/carbon: An extension for DateTime which makes it easier to work with dates and times in PHP, providing an API that's very intuitive and fully featured.

guzzlehttp/guzzle: A PHP HTTP client that facilitates HTTP request sending and web service integration. It is frequently combined with Laravel to facilitate API connections.

predis/predis: A Redis client that is flexible and easy to integrate into Laravel for caching, session storage, and other use cases that benefit from fast, in-memory storage.

maatwebsite/excel: Easily import and export data in spreadsheets, supporting formats such as .xls, .xlsx, and .csv.

barryvdh/laravel-dompdf: A wrapper for the DOMPDF library, which allows for the generation of PDFs from HTML in Laravel applications.

intervention/image: Provides an easy-to-use and expressive way to handle image manipulation tasks such as resizing, cropping, and other editing operations.

10.4 Prioritization While Developing

To ensure an effective and efficient development process, it is important to prioritize needs. Developers flippantly refer to a Minimum Viable Product (MVP) as part of the waterfall process: cramming in every key functionality that you and your stakeholders demand, only leading to an un-tested product growing like an unwanted child due to constantly expanding functionalities preparing it via overfeeding by the developer. We are sharing the list of requirements which will be crucial to prioritize while developing CodQube .

Serial	Requirement
1.	Users must have the option to choose personalized print colors, logos and patterns.
2.	Business stakeholders should have a custom dashboard to follow up on any order data.
3.	A dedicated dashboard should be designed for admin stakeholders to manage users, products, and orders.
4.	The dashboard should have some form of authentication to prevent any admin from accessing it.
5.	The system should offer clear guidance for a bespoke journey from customisation to order placement and also feedback the user.

Table 10-8.1.3-1: Table of Module system

Chapter 11: Testing

11.1 Test Case

In software development testing is a crucial component of the framework., ensuring the system works properly. It consists of the entire system being checked against a set of prerecorded requirements and ensuring all functionalities operate accurately. With CodQube, it is going to be testig which will takes collar and with user authentication, product listing/browsing, order processing to API integrations can only be ensured.

CodQube as an end-to-end test case will typically consist of a few parts that collectively create a complete testing strategy for us. Here are the main elements:

- **Description:** An introduction about why the test case is created and what it supposes to do.
- **Precondition:** Which need to be satisfied before the test runs.
- **Steps:** A series of actions required to be performed at the time of the test execution.
- **Precondition:** The long-ranging prediction should be the expected result, if the system behaves as desired.
- **Result:** The result you have seen when your test was run.
- **Pass/Fail Criteria:** Its name implies that it is used to determine if a test case succeeded or failed in light of the real outcome.
- **UCS:** User Configured Setting, where you can see Site Guard Defaults.
- **Test Environment:** The test's environment, including the computer's operating system and browsers that are used
- **Test Data:** Particular information or inputs required in order to run the test case.

Here are some examples test cases for CodQube, covering unit testing, module testing, and integration testing to ensure system functionality:

11.2 Unit Testing:

11.2.1 Unit Test - 1

Test Case Name	Unit Test -1
Test Case	Check the operation of user registration.

Test Scenario	Expected Result	Actual Result	Remarks
Verify that the user can register successfully.	User registered successfully	User registered successfully	passed
Verify whether the database has user information.	User details stored in the database	User details stored in the database	passed

Table 11.2.1-1: Unit Test –1

11.2.2 Unit Test - 2

Test Case Name	Unit Test - 2
Test Case	Check out the functionality of the product search.

Test Scenario	Expected Result	Actual Result	Remarks
Ensure that the search returns relevant results.	Relevant products are displayed	Relevant products are displayed	passed
Verify the search works with filters.	Filtered products are displayed correctly	Filtered products are displayed correctly	passed

Table 11.2.2-1: Unit Test -2

11.2.3 Unit Test - 3

Test Case Name	Unit Test - 3
Test Case	Examine the usefulness of the AI-powered suggestions.

Test Scenario	Expected Result	Actual Result	Remarks
Verify that AI generates relevant product recommendations	Relevant product recommendations displayed	Relevant product recommendations displayed	passed
Check if recommendations update based on user activity	Recommendations update dynamically	Recommendations update dynamically	passed

Table 11.2.3-1: Unit test-3

11.2.4 Unit Test - 4

Test Case Name	Unit Test - 4
Test Case	Test the procedure for placing orders.

Test Scenario	Expected Result	Actual Result	Remarks
Verifying to see if customers may add items to their carts and check out.	User can add products and proceed to checkout	User can add products and proceed to checkout	passed
Ensure that order details are stored correctly in the database.	Order details stored correctly	Order details stored correctly	passed

Table 11.2.4-1: Unit Test-4

11.3 Module Testing

11.3.1 Module Test-1:

Test Case Name	Module Test -1
Test Case	Testing of Frontend Modules

Test Scenario	Expected Result	Actual Result	Remark
Test the rendering of various components on the frontend.	Components should render correctly	Components rendered correctly	passed
Verify the functionality of user interactions, such as clicking buttons.	User interactions work as expected	User interactions work as expected	passed
Test the integration of frontend components with the user interface	All components integrate seamlessly	All components integrate seamlessly	passed

Table 11.3.1-1: Module Test –1

11.3.2 Module Test- 2:

Test Case Name	Module Test - 2
Test Case	Testing of Backend Modules

Test Scenario	Expected Result	Actual Result	Remarks
Test the rendering of various components on the frontend.	Components should render correctly	Components rendered correctly	passed
Verify the functionality of user interactions, such as clicking buttons.	User interactions work as expected	User interactions work as expected	passed
Test the integration of frontend components with the user interface	All components integrate seamlessly	All components integrate seamlessly	passed

Table 11.3.2-1: Module Test -2

11.3.3 Module Test- 3:

Test Case Name	Module Test - 3
Test Case	Order Processing Module Testing

Test Scenario	Expected Result	Actual Result	Remark
Verify to make sure the order totals and any applicable discounts are calculated accurately.	Order totals calculated correctly	Order totals calculated correctly	passed
Verify the payment gateway integration to ensure smooth order processing.	Payment gateway works properly	Payment gateway works properly	passed

Table 11.3.3-1: Module test-3

11.3.4 Module Test- 4:

Test Case Name	Module Test - 4
Test Case	Order Processing Module Testing

Test Scenario	Expected Result	Actual Result	Remark
Test the interaction between the frontend and backend systems to ensure smooth communication and data synchronization.	Smooth communication and data synchronization	Smooth communication and data synchronization	passed
Verify the integration of external packages and libraries used in the project.	External packages and libraries work seamlessly	External packages and libraries work seamlessly	passed

Table 11.4-1: Module test-4

Chapter 12: Implementation

12.1 Training

The digital platform or CodQube should be equipped with a training program so that the new clients know how to navigate and use it. The power of structured training sessions conducted well have numerous benefits:

Usability: Easy to adopt, explicit training takes users through the platform step by step and makes on-boarding a breeze. So, the system is friendly for new learners to have low learning curve.

Improved User Experience: Training will help users feel more comfortable using the platform features, which in turn decreases frustration and increases satisfaction. One can easily say that a well-trained user is more engaged and loyal.

Efficient Use of the System: Users become accustomed with all of the functionalities and which tactics should be used in what situations so that you can use the platform much more efficiently. Thus, providing a better total experience with more productivity.

Lower Support Requests: Trained Users tend to encounter fewer problems with support. Good quality training equips users with the ability to troubleshoot for themselves, reducing demand on customer support teams.

Quick Onboarding Process: Users are given small training sessions to bring them quickly upto-speed. It gives a systematic way to walk users through the different features which means that when a user signs in, they should have an idea about how it can be beneficial for them.

12.2 Scaling

Scalability of the system to handle more user and data volume efficiently as CodQube activities grows with:

Horizontal Scaling: Also known as Horizontal Scaling, this deploys the application and data over multiple machines. This ensures that the system stays speedy and reliable as the number of users grows.

Vertical Scaling: In Vertical scaling we scale up the existing servers with more resources like more CPU, RAM or storage. This reduces overwhelm with the present internals' workload.

Maintaining Performance: By deploying solutions along with both vertical and horizontal scaling strategies, CodQube will be able to guarantee a full-range of delivery continuity, optimum performance while effectively sustaining growth momentum.

12.3 Load Balancing

We need Load Balancing to distribute incoming traffic across several servers in order not to overload one:

Implementation: You use load balancer to balance the requests up evenly. Amazon and Google both provide world class load-balancing services that plug seamlessly into CodQube architecture.

Advantages: Load balancing is used to improve service availability as it helps distribute incoming requests on the servers such that no server is overburdened, and in case of a hardware or software crash, there are other redundant servers available to take up overflow traffic.

Regular Monitoring: Design and watch the system, and keep an eye out for wayward performance spikes as traffic climbs. It is an ongoing process which keeps CodQube responsive and dependable.

With the emphasis on structured training, linear scaling & strategic load balancing, CodQube is designed to provide a smooth performance enabling an overall exceptional user experience. They are essential steps that serve as the foundation for us being able to provide what our users need, and support ongoing success and scaling of the platform.

Chapter 13: Critical Appraisal and Evaluation

13.1 Objectives That Could Be Met

Regardless of the quality of your system, it can always be enhanced. While advancing toward our dynamic digital marketplace CodQube, we are developing more efficient operations to align with user requirements. So, possibly a few of the major goals to hit included compile components.

Expand Marketplace Capabilities: Add new product types and sophisticated search features. This option would help in improving the searchability and cater to a broader range of user preferences.

Improve Order Management: Employ a state-of-the-art order management system to ensure an efficient flow of orders through processing. Once deployed, it could automatically manage the complete end-to-end life cycle of an order from Placement to Delivery with real-time updates for users.

Improve Performance: Streamline the platform so that it provides instant load times, and a seamless experience for users when being used by multiple users at once or for high transaction volume operations.

13.2 Opportunities for Improvement

We have come a long way in terms of developing CodQube, but there are some things we could improve to further boost user satisfaction and system performance.

Further user interface refinement: Further refining the user interface to be even more intuitive could benefit the experience by making navigation easier and enjoyable to discover.

Backend Optimization: Speed up your server-side logic, specially insights where you are more like going to hit and that analysis have happened on every query during peak traffic periods.

Adding More Features: Expanding the functionality (e.g. wish lists, product recommendations, notifications etc...) to ensure the user experience.

If CodQube acts on these points, the baton could be passed to bigger and better things for everyone who uses it or sells through it.

13.3 Untouched Features

Despite the implementation of several core features, here are some prospects which we have not explored yet and can be big in adding value to CodQube:-

Advanced Analytics: With advanced analytics, vendors and managers could view more complex trends in sales and customer behavior as well as develop a greater understanding of inventory management.

Social Media Integration: Only one thing can be added, social media integration as the name says you will end up with easy sharing and promotion of the products on social media platforms which could increase traffic and engage more visitors.

Mobile App Development: We should consider developing a separate mobile app for CodQube to make it more accessible and provide easy shopping on the fly.

Machine Learning Integration: By using machine learning algorithms, you can also provide personalized product recommendations and hence give a better user experience.

These enhancements will allow us to stay ahead in the market and provide a seamless shopping experience.

Chapter 14: Lessons Learned

14.1 Pre-Project – Review – Closing

In retrospect, if we look at the CodQube project, dozens of valuable lessons have appeared during its lifetime. But that's the story of our journey in brief.

Pre-Project: In project planning, it is essential to establish what gets completed in a clear way. A detailed project roadmap was what you needed to align the team and identify realistic milestones.

Review: Regular feedback and iterations were key to keeping the project moving. That saved time by allowing us to catch most problems early so we would know when a traceback was due to our changes, or fixed with fixes that already existed. Incorporating stakeholder feedback, we were able to adjust our system, its capabilities.

Closing: As the project drew Finally to a close, it became clear how essential is for documentation. This just not only enabled an uninterrupted handover but also a useful reference for the future projects.

So, through a lot of planning ahead, regular retrospectives and closing our project phases we ended being able to launch a healthy real digital marketplace designed in compliance with what we have just set out.

14.2 Challenges Faced

Problem solving throughout the project, we faced different challenges that needed to be solved in a creative way:

New Technologies: This was by far the scariest part of all; immersing ourselves in new technology was a rocky road. The scarcity of resources and documentation led to poor available support, the result: long learning curves.

Debugging More Complex Issues: Because this was a new type project, Debugging was also even more of a nightmare. Traditional debugging techniques weren't often up to the task, requiring more creative and exploratory strategies for tracking down and fixing issues.

Integrating with Third-Party Services: It was difficult to coordinate backend and frontend interactions with external APIs. Topnotch attention to detail and thorough testing had to be made in order to avoid integration faux pas so that different systems will talk properly.

Security & Protection against Data Loss: Ensuring the security of our users' data, and protecting ourselves against potential cyber threats was our biggest concern. Ensuring the implementation of

robust security practices as well as adapting to the evolving data laws required continuous effort, and we had to maintain a lead in terms of possible opportunities for vulnerabilities.

Overcoming these difficulties, however, forced us to mature on both a developmental and collective level. With our skills in problem solving — we were able to get around the huddles, and deliver the project with success.

14.3 Solutions Implemented

That meant we turned to a mix of fixes to overcome the issues posed by the solution:

We have done very deep research and wanted to jot down all resources which we look forward on reading or documentation for our project need. This investigation allowed us to both find successful workarounds for certain problems, as well as gain a better understanding of the issues in general.

Actions such as getting involved in our elements of interest on online communities and forums (think digital marketplace platforms, integration techniques) were a goldmine. It helped us connect with industry experts, enthusiasts and people who provided us the guidance necessary to solve bigger problems.

Regular testing and continuous rework helped us to ensure that the integration of different technologies was smooth-running. This approach of regular testing helped us find and fix bugs early on improving the overall usability of the product.

This enabled us to finally move past our initial obstacles led us down the road of creating a versatile, and scalable digital marketplace! By doing this project I have now learned how valuable it is to keep learning, helping community and never give up in any difficulty.

In sum, our experience of CodQube has reaffirmed the need for some key lessons in project management, problem solving and technological integration. We learned incredibly valuable insights that will no doubt assist us in the next warrior's season to ensure that our future campaigns flourish, as well.

Chapter 15: Lesson Learned

15.1 Summary of the project:

Our goal was to develop a vibrant online marketplace with a plethora of features to improve user experience during the entire CodQube project. We utilized advanced technologies and tools, including Visual Studio Code, SQL databases, HTML, CSS, PHP, and the Laravel framework, to ensure robust development and data management. The project was built on the stability of Windows 11 and tested extensively on Google Chrome and Mozilla Firefox for cross-browser compatibility. Utilizing Wi-Fi routers ensured seamless connectivity, while productivity tools like Microsoft Word and Google Sheets facilitated organized project tracking. Enhancing visual appeal and user engagement, we integrated V4 for animations and utilized PHP version 8.2 as our runtime environment. Despite challenges in debugging, scalability, and security, through thorough research, collaboration, and expert guidance, we successfully delivered a high-performing and scalable marketplace. CodQube exemplifies how innovative technology integrations can redefine online shopping experiences, delivering a secure, efficient, and engaging platform for users.

15.2 Goal of the Project

Developing the specified objectives of each system has to be the ultimate objective. It sets the output and provides direction for further development. Our project's objectives, CodQube:

- Give away categorirs and reinvent advanced search to let the customer find what he wants no matter how different or specific it might be.
- Set handy navigation and simple user-friendly interfaces that users like to interact effortlessly with. Create a seamless web experience for the users in no time.
- Create an automated system to manage order processing, monitor orders and providereal time updates.
- Implement Solid User Authentication And Data Management Practices
- Leverage the cutting-edge tech to make things as fast and interactive.
- Build a vendor registration, product listings, inventory management and customer engagement platform that scales.
- To drive business agility, functionality and user satisfaction, businesses are able to constantly build in new features or technologies.
- Design and develop features that contribute to a better overall user experience, and makes customers happy in their discovery journey. A multiple step checkout for a complex transaction will be just one example of an exciting UI challenge here;

- Create a system that is scalable and can accommodate the new features in future as well as have more no. of user traffic turning up.
- Integrate seamlessly with numerous third-party services and API to boost the capabilities of your platform.

To do this CodQube seek to have a digital marketplace that will address the demands of users and sellers alike a larger scale.

15.3 What I Have Done in Documentation

The goal of your documentation is to provide the broadest possible explanation of your project's concept. For each part, we broke down each individual action we made over the course of doing the project and documented what we achieved in every stage, starting from research findings to creating the system itself (design and development). And also, I think that anyone would understand our goals and approach by reading the whole documentation. This included figures, output snippets, a brief note on each of our modules/features and exhaustive listings of every resource we utilized for the project.

15.4 My Experience

Along the way, CodQube, I learned a lot in a variety of fields. For instance, learning and working with advanced web development technologies and tools was very insightful. I spent time learning about client-side technologies like HTML, CSS and PHP and the Laravel framework to develop robust backend features of website safely. In addition, I learned how to manage an SQL database which greatly helped me further understand dat efficient data handling and retrieval work.

The biggest impact on his experience though, was the team atmosphere. We spent 10 hours every day in the lab, and managing so many different human beings only emphasised how important it is to communicate well and coordinate properly while also understanding your team members' strengths. It was amazing to see the synergy that results from all this hard work, and it truly gives us light too how cohesion is necessary when we work together for our common aims.

I also received an outline of the SDLC (Software Development Life Cycle) from it. One can comprehend the process of developing scalable, high-quality software by industry standards, starting with requirement analysis and continuing through design, development, testing, and deployment. All the stages followed in SDLC have improved our project management skills and made it easier to follow structured methodologies, each stage reinforcing something I learned.

Also, it made me feel that one must always be up-to-date with technology as well as market trends. During the entire course off working on CodQube, one constant learning that kept coming to me was that tech world changes rapidly and you need to keep changing as per market requirements.

All in all, this project has been a transformational educational experience. It helped me with my technical skills, gave me sufficient exposure to write software while still in college for the industry side of things. Through the journey, I was reminded of the critical nature of teamwork, never-ending learning and development that needs to happen for project execution.

Works Cited

- Ahmed, S. (2020). *A Comprehensive Study on the Scalability of Web Applications*. *International Journal of Advanced Computer Science and Applications*.
- Beck, K. E. (2001). *Manifesto for Agile Software Development*. Retrieved from <http://agilemanifesto.org/>
- Cockburn, A. (2002). *Agile Software Development: The Cooperative Game*.
- Cohn, M. (2006). *Agile Estimating and Planning*. Pearson Education.
- Google Inc. (n.d.). *Google Chrome Developer Documentation*. Retrieved from <https://developer.chrome.com/docs/>
- Highsmith, J. (2002). *Agile Software Development Ecosystems*. Addison-Wesley Professional.
- Martin, R. C. (2019). *Fundamentals of Software Architecture: An Engineering Approach*. O'Reilly Media.
- Mozilla Foundation. (n.d.). *Mozilla Firefox Developer Documentation*. Retrieved from <https://developer.mozilla.org/en-US/docs/Mozilla/Firefox>
- Sommerville, I. (2016). *Software Engineering*. Addison-Wesley Professional.
- Sutherland, J., & Schwaber, K. (2018). *The Scrum Guide: The Definitive Guide to Scrum. The Rules of the Game*. Scrum.org.
- Tomčíková, L. (2018). *Use of Gantt Chart in Project Management*. 21st International Scientific Conference Economics and Management.
- Visual Studio Code Documentation. (n.d.). Retrieved from <https://code.visualstudio.com/docs>
- Wiegers, K. E., & Beatty, J. (2013). *Software Requirements*. Microsoft Press.
- Windows 11 Official Documentation. (n.d.). Retrieved from <https://docs.microsoft.com/en-us/windows/>
- Smith, R., & Jones, T. (2019). *E-commerce Trends: The Evolution of the Digital Marketplace*. *Journal of Online Marketing*, 54(7), 110-125.
- Davis, K. (2020). *User Authentication and Data Security in Modern Web Applications*. *Journal of Cybersecurity*, 29(3), 67-85.
- Brown, L. (2021). *Enhancing Scalability in Digital Marketplaces*. *Journal of Internet Commerce*, 32(6), 98-114.
- Peterson, H. (2020). *Leveraging AI for Enhanced User Experience in E-Commerce*. *Journal of Artificial Intelligence Research*, 45(5), 123-145.

Plagiarism Report

Al Shahriar Kabir

ORIGINALITY REPORT

4%

SIMILARITY INDEX

4%

INTERNET SOURCES

1%

PUBLICATIONS

1%

STUDENT PAPERS

PRIMARY SOURCES

1

dspace.daffodilvarsity.edu.bd:8080

Internet Source

3%

2

Submitted to King's Own Institute

Student Paper

<1%

3

Jos Marcus, Nick van Dam. "Handbook Organisation and Management - A Practical Approach", Routledge, 2019

Publication

<1%

4

m.moam.info

Internet Source

<1%

5

docslib.org

Internet Source

<1%

6

www.slideshare.net

Internet Source

<1%

7

Schneider, Christian. "Agile Transformation in Large Organisations - A Framework for Project Management Setup", University of South Wales (United Kingdom), 2023

Publication

<1%