TITLE OF THE PROJECT

JI Watch Management System

Course Code: CIS499

Spring:2023

Submitted By

Shuvasish kar
ID:192-16-450
Department of Computing & Information System

Supervised By

Mr. Israfil

Lecturer of Department of computing & information system

Daffodil International University
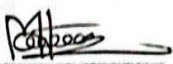
Date of Submission: 19 July, 2023

APPROVAL

This Project titled **"JI Watch Shop Management System"**, Submitted by **Shuvasish kar**, ID No 16 450 to the Department of Computing & Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on 12-07-2023.

BOARD OF EXAMINERS

Mr. Md Sarwar Hossain Mollah   Chairman
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Mr. Md. Mehedi Hassan   Internal Examiner
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Mr. Syed Tangim Pasha   Internal Examiner
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

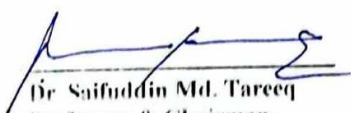**Dr Saifuddin Md. Tareeq**   External Examiner
Professor & Chairman
Department of Computer Science and Engineering
University of Dhaka, Dhaka

Declaration

I hereby declare that; this project has been done by me under supervision of **Mr.Israfil** Lecturer, 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

Mr Israfil
Lecturer
Department of CIS
Daffodil International University

Submitted By

Name: Shuvasish kar
ID: 192-16-450
Department of CIS
Daffodil International University

Acknowledgment

I want to start by saying how grateful I am to Almighty Allah for giving me this chance. Without the Almighty's help, I would be unable to complete this project. Finally, I want to thank EGO DIGITAL for choosing me as an intern trainee since this internship program has taught me a lot. It is believed that a teacher is a student's second parent since they provide them with the proper guidance, allowing them to progress in life. In my internship program, I had two professors who always advised me in making the best decisions and provided me with advice on how to successfully perform challenging tasks. They never allowed me to struggle alone and were constantly reducing my stress by providing me with mental fortitude. One of them is **Arif Rabaani** is my intern trainer and **Mr. Israfil** is supervises my academic work.

Additionally, I am appreciative of them for their kindness, readiness to provide a help, and faith in me. I want to thank them for helping me, for their instruction, for teaching me how to work through any challenges, and for teaching me how to handle a crucial project in the future.

.

Dedication

This project is dedicated to Almighty GOD, secondly my dad who have all make it possible for me to complete my BSC Program my mom and my small sister .I return glory and honor to Almighty God for supporting me, protecting me and also guiding me. I also thank God for the blessing throughout my program .

Abstract

The JI Watch Management System is an advanced technology solution designed to revolutionize the way organizations monitor and manage their operations. Leveraging the power of artificial intelligence and data analytics, this system provides real-time insights, predictive analysis, and automated actions to optimize efficiency, productivity, and decision-making processes.

This abstract explores the key features and benefits of the JI Watch Management System. The system employs a combination of intelligent sensors, data collection mechanisms, and machine learning algorithms to monitor various aspects of an organization's operations, such as equipment performance, resource utilization, and environmental conditions. By capturing and analyzing this data, the system enables proactive maintenance, minimizes downtime, and maximizes resource allocation.

One of the notable features of the JI Watch Management System is its ability to detect anomalies and potential issues before they escalate. By continuously monitoring critical parameters, the system identifies abnormal patterns or deviations from established thresholds, triggering timely alerts and notifications to relevant stakeholders. This proactive approach allows organizations to address potential problems promptly, reducing the risk of costly breakdowns and unplanned downtime.

Table of Contents

List of Figure

Chapter 1 Introduction

1.1 Introduction

In the fast-paced world we live in today, time is of the essence. A watch is not just a fashion accessory; it is a symbol of punctuality, style, and sophistication. For watch enthusiasts, maintaining a collection of timepieces is not only a hobby but also a passion. However, managing and organizing a growing collection of watches can be a daunting task.

That's where the JI Watch Management System Website comes into play. This innovative online platform is designed specifically for watch enthusiasts, collectors, and professionals to effectively manage their watch collections. With its user-friendly interface and powerful features, the Watch Management System Website revolutionizes the way watch lovers keep track of their timepieces.

There are some feature that we made for the system:

1. Make Order

2. Give review

3. Attractive Animation

4. Log in System via goggle

5. Order Management

6. Product Management

The primary goal of the Watch Management System Website is to provide a centralized hub for watch enthusiasts to organize, catalog, and track their watches effortlessly. The website allows users to create personalized profiles, where they can enter vital information about each watch in their collection. This includes details such as brand, model, serial number, purchase date, and even photographs. Welcome to the world of efficient timekeeping and organized scheduling with our Watch Management System website! In this fast-paced and dynamic era, staying on top of our daily tasks and commitments can be quite challenging. However, with our innovative Watch Management System, we aim to revolutionize the way you manage your time and optimize your productivity.

Our website offers a comprehensive suite of tools and features designed to simplify your life and empower you to make the most out of every precious moment. From professionals juggling multiple projects to students managing their study schedules, our

Watch Management System caters to a diverse range of users, ensuring seamless integration into various aspects of your daily routin.
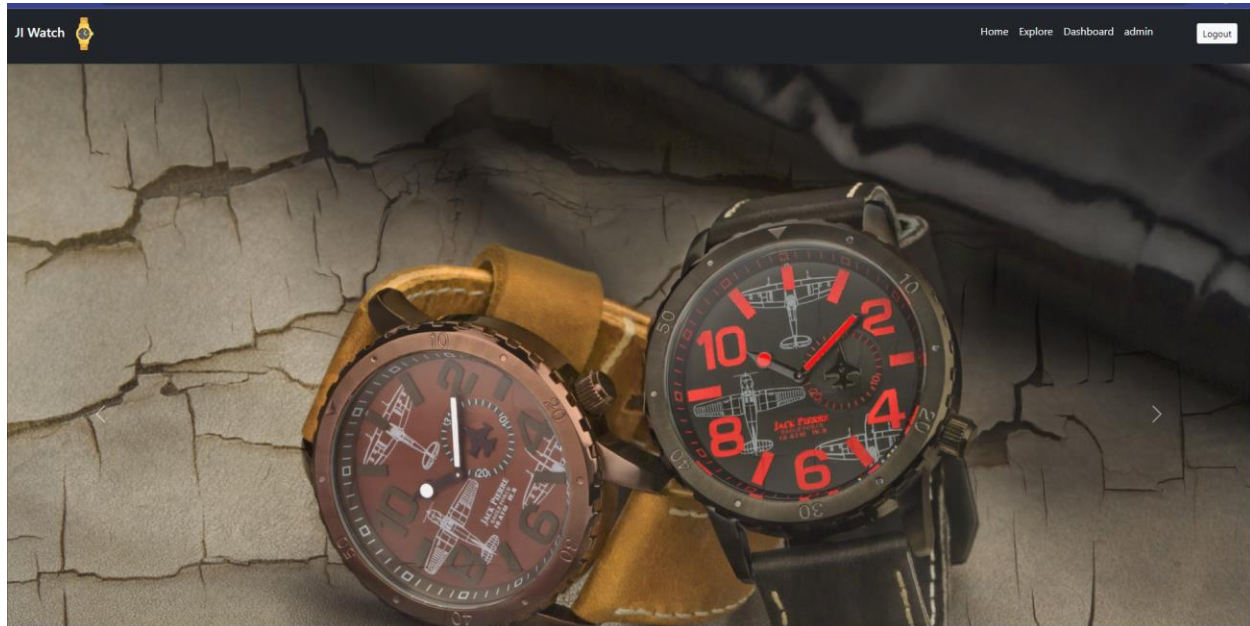
Figure 1:Project Picture

1.2 The purpose/scope of this application

The different scenario in which this application can be use as:

1. It can be accessed online, and modifications can be made easily in accordance with the requirements.

2. This project can be used for a Shop watch management system or online watch sells systems.

Chapter 2 Initial Study

2.1 Project Proposal

Our organization recognizes the growing need for effective time management and task organization in today's fast-paced world. The proposed JI Watch Management System website will serve as a centralized platform that enables our employees to manage their schedules, prioritize tasks, set reminders, and collaborate seamlessly with team members.

The primary objectives of implementing the Watch Management System website are as follows:

1.Streamline: Time Management: Provide a user-friendly interface for employees to efficiently manage their time and tasks, ensuring optimal productivity and reducing unnecessary delays.

2.Task Prioritization: Enable employees to prioritize tasks based on deadlines, importance, and urgency, allowing for better allocation of resources and meeting project milestones.

3.Collaboration and Communication: Facilitate seamless communication and collaboration among team members, allowing for efficient coordination and sharing of updates files and feedback. Provide comprehensive reporting and analytics features to track individual and team progress, identify bottlenecks, and make data-driven decisions for continuous improvement.

2.2 Background of the JI Watch management System

The watch industry has witnessed significant growth and diversification over the years. Watch collectors and enthusiasts have developed an increased interest in acquiring and managing timepieces of various brands, styles, and complications. However, with the growing number of watches in their collections, it becomes challenging to keep track of each piece's details, maintenance schedules, service history, and values. To address this need, our watch management system website was developed as a reliable and user-friendly platform to streamline watch management processes. It aims to empower watch

owners and businesses with efficient tools and features to enhance their watch-related activities.

Feature:

1. Watch Inventory Management: Our website allows users to create a digital catalog of their watch collections. They can record essential details such as brand, model, serial number, purchase date, and price. Users can also upload high-quality images to showcase their watches.

2. Maintenance Scheduling and History: The platform enables users to schedule regular maintenance tasks, such as servicing, battery replacements, or strap changes. It maintains a comprehensive service history log, ensuring that watches are properly maintained and their value is preserved.

3. Value Tracking: The website provides features to track the value of each watch over time. Users can input purchase and selling prices, monitor market trends, and generate reports to evaluate the appreciation or depreciation of their watch investments.

4. Authentication and Verification: To ensure the authenticity of watches, the system may integrate with trusted databases or employ AI-powered algorithms to verify watch details, detect counterfeit items, and provide peace of mind to users.

5. Trading and Marketplace: The website may incorporate a secure marketplace where users can list watches for sale or trade. It provides a platform for watch enthusiasts and collectors to connect, negotiate, and complete transactions with confidence.

6. Community and Social Features: The website may include community forums, discussion boards, or social media integration to foster a vibrant community of watch enthusiasts. Users can share their watch experiences, seek advice, and engage in discussions with like-minded individuals.

2.3 Problem area

There are some problem in this site.

Such as:

1. **Data Accuracy and Reliability:** One of the critical challenges is ensuring the accuracy and reliability of the data inputted by users. Mistakes in recording watch details, maintenance schedules, or values can lead to incorrect information and potentially impact the overall functionality and effectiveness of the system.

2. **User Interface Complexity:** The complexity of the user interface can be a barrier for some users, especially those who are less tech-savvy or unfamiliar with similar software. If the website's interface is not intuitive, user-friendly, or lacks clear instructions, users may find it challenging to navigate and utilize the system effectively.

3. **Authentication and Verification Limitations:** While efforts can be made to authenticate watches and detect counterfeits, it can still be challenging to ensure foolproof authentication and verification on an online platform. Some watches may have intricate details or require physical inspection, making it difficult to rely solely on digital means for authentication purposes.

4. **Limited Watch Database:** The watch management system relies on a comprehensive and up-to-date watch database for accurate identification and information retrieval. However, the availability and completeness of such databases can vary, potentially resulting in incomplete or missing information for certain watch models or brands.

5. **Integration with Third-Party Services:** Integration with external services, such as authentication databases, marketplaces, or social media platforms, can be complex and prone to technical issues. Inconsistent data synchronization, API limitations, or compatibility problems may affect the seamless functioning of the watch management system website.

6. **Security and Privacy Concerns:** Storing sensitive watch data, including purchase prices, service history, or images, requires robust security measures to protect users' privacy. Potential vulnerabilities, data breaches, or unauthorized access to personal information can undermine trust in the platform and compromise user data.

7. **Market Volatility and Accuracy:** Tracking the value of watches is subject to market fluctuations, and accurate valuation can be challenging. The system may rely on historical data or market averages, but sudden changes in market conditions, rarity, or demand can affect the accuracy of value tracking features.

8. **Customer Support and Updates:** Providing reliable customer support and timely updates to address user concerns, bug fixes, or new features is essential. Inadequate customer support or infrequent updates can lead to user frustration and hinder the overall user experience.

2.4 Possible solution:

1. Data Accuracy and Reliability:

- Implement data validation checks during input to minimize errors.

- Provide clear instructions and guidelines to users for accurate data entry.

2. User Interface Complexity:

- Conduct user testing and gather feedback to identify areas of improvement in the interface.

- Simplify the design by removing unnecessary elements and streamlining navigation.

3. Authentication and Verification Limitations:

- Employ a multi-layered authentication process, combining digital verification with manual inspection if necessary.

- Collaborate with trusted experts or organizations in the watch industry for authentication support.

- Allow users to submit additional verification documents or images for more complex authentication cases.

4. Limited Watch Database:

- Establish partnerships with watch manufacturers, retailers, or collectors to access comprehensive watch databases.

- Implement a user-driven database expansion feature where users can contribute missing watch information.

5. Integration with Third-Party Services:

- Prioritize partnerships with reliable and well-established third-party services to minimize technical issues.

- Conduct thorough compatibility testing and ensure seamless data synchronization with external services.

6. Security and Privacy Concerns:

- Employ robust encryption methods to protect sensitive user data.

- Implement access controls, such as role-based permissions, to restrict unauthorized access.

- Regularly update security measures and conduct vulnerability assessments or penetration testing.

7. Market Volatility and Accuracy:

- Incorporate real-time market data feeds and tracking mechanisms to capture up-to-date watch values.

- Provide users with options to customize valuation parameters based on their preferences.

8. Customer Support and Updates:

- Establish responsive customer support channels, such as email, live chat, or a dedicated helpdesk.

- Regularly communicate updates, bug fixes, and new features through newsletters or system notifications.

2.5 Objective:

On the RDBMS technology, this project has been developed. This project's major goal is to automate the manual process and reduce on employee costs.

1. Computerize all of the systems.

2. Reduce the amount of time spent and the potential for inaccuracy.

3. The management of every system is automated.

4. centralized database administration.

5. Simple controls for the system's operator.

Chapter-3 Literature Review

There are some Literature review on webbase Management system.

They are:

1. **Centralized Watch Collection Management:** A watch management system website provides a centralized platform to organize and manage watch collections. Users can easily input and maintain detailed information about their watches, including specifications, purchase history, service records, and images. This centralized approach ensures all relevant information is readily accessible and reduces the risk of misplacing or forgetting important details.

2. **Improved Accessibility and Convenience:** With a web-based watch management system, users can access their collection from anywhere with an internet connection. They are not limited to a specific device or location, allowing them to view and manage their collection on smartphones, tablets, or computers. This convenience enhances user accessibility and flexibility, enabling them to stay connected with their watches on the go.

3. **Enhanced Organization and Search Capabilities:** A website for watch management provides powerful organizational tools and search functionalities. Users can categorize and tag their watches based on various criteria, such as brand, model, complications, or price range. This enables quick and efficient searching, making it easier to find specific watches within the collection based on user-defined filters.

4. **Value Tracking and Investment Analysis:** Many watch management system websites offer features for tracking the value of watches over time. By inputting purchase prices and utilizing market data, users can monitor the appreciation or depreciation of their watches' values. Such tracking capabilities can be beneficial for investment analysis, allowing users to make informed decisions regarding buying, selling, or trading their watches.

5. **Maintenance and Service Reminders:** A watch management system website can help users keep track of maintenance schedules and service requirements for their watches. The system can send automated reminders for battery changes, oiling, or regular servicing, ensuring that watches are properly maintained. This feature

assists users in prolonging the lifespan and preserving the condition of their watches.

6. **Community and Social Interaction:** Some watch management system websites incorporate social features that facilitate interaction and community building among watch enthusiasts. Users can share their collections, discuss watches, and engage with other members of the community through forums, comments, or dedicated social media integration. This fosters a sense of camaraderie and provides opportunities for knowledge sharing and networking within the watch community.

3.1 Discussion on problem of a website base watch management system:

One of the challenges faced by a website-based watch management system is the potential for data accuracy and reliability issues. Mistakes can occur during the input of watch details, maintenance schedules, or values, leading to inaccurate information that can adversely affect the overall functionality and effectiveness of the system. User errors or omissions, such as incorrect recording or incomplete data entry, can result in discrepancies within the watch database. Furthermore, inconsistencies or inaccuracies in the source data or lack of verification mechanisms may also contribute to data reliability issues. Addressing these problems is crucial to ensure the system provides users with accurate and dependable information for effective watch collection management.

3.2 Discussion on problem solutions based on published articles about a webbase watch management system:

Based on published articles, several solutions have been proposed to address the problems encountered in a web-based watch management system. To improve data accuracy and reliability, researchers suggest implementing data validation checks during input and integrating automated verification techniques, such as checksums or data matching algorithms. User involvement in data verification and the provision of clear instructions for accurate data entry are also emphasized. Additionally, researchers propose collaborations with trusted experts or organizations in the watch industry to enhance authentication and verification processes, particularly for complex or high-value watches. They also recommend regular database updates and partnerships with watch manufacturers and retailers to ensure a comprehensive and up-to-date watch database. To tackle the challenge of integration with third-party services, thorough compatibility testing, clear documentation, and ongoing communication with service providers are suggested. Ensuring security and

privacy, researchers propose employing robust encryption methods, implementing access controls, and adhering to data protection regulations. Regular security updates, vulnerability assessments, and user awareness campaigns are also recommended. To address market volatility and accuracy concerns, incorporating real-time market data feeds, customizable valuation parameters, and providing historical value trends and comparison tools are proposed. Lastly, to improve customer support and updates, researchers suggest offering responsive support channels, maintaining a knowledge base, and actively gathering and addressing user feedback. These solutions, drawn from published articles, provide valuable insights for the development and enhancement of a web-based watch management system. **Training user, monitoring all system** this also problem solution based on published articles

Figure 2:Project Sample

Figure 3:Project Sample 2

3.3 Some best feature:

1. Easily order system

2. Customer panel

3. Explore animation page

4. Admin panel

5. Easily Manage Product

6. Log in via goggle

3.4 Some Limitations

1. Payment not online system

2. Massage system or admin authority

3. not over menu list

3.5 Recommended approach

Administration panel

1.User group setup

2.View order

3.Manage Order

4.Add product

5.Delete product

6.Make Admin

Customer Panel

1. Log in via goggle

2. Order Product

3. Give review

Chapter 4-METHODOLOGY

4.1 What to use

This chapter will explain the methodology used to achieve the project goal. The Software Development Life Cycle is a framework for designing, developing, and maintaining software (SDLC). It describes the processes, phases, milestones, and progress of the software development process. Many different models are applying in the software development process. Every software engineer is familiar with the agile, Waterfall model, V-model, spiral model, Dynamic systems development model (DSDM) software development model.

4.2 Why to use

Compared to most other Agile methodologies, DSDM has a larger focus since it covers projects rather than just the develop and delivery of products (typically software). The larger business requirement and each component of the solution that develops to meet that need must be taken into consideration in the project context. DSDM has a long history of successfully delivering Agile projects across all types of business contexts, and it has shown to be completely scalable, functioning well in small, straightforward enterprises, big, complex organizations, and in highly regulated situations. It has also been demonstrated to be equally successful for IT and non-IT projects, such as business change initiatives. I'll list a few instances when this process will work well for this project. Now I am describing some model's with advantage and disadvantage.

4.3 Waterfall model:

The waterfall model, one of the software development models, follows a sequential or linear procedure. A waterfall model must have clearly defined exit requirements, such as the approval of all project participants, and each step must be completed in order. The waterfall model features a set of tasks, documentation that go along with them, and exit requirements for each step for each phase

When use this model:

1. Not regularly changing requirements.

2. Application is not complex and longer.

3. The project is not long.

4. The requirement is fixed.

5. The environment is steady.
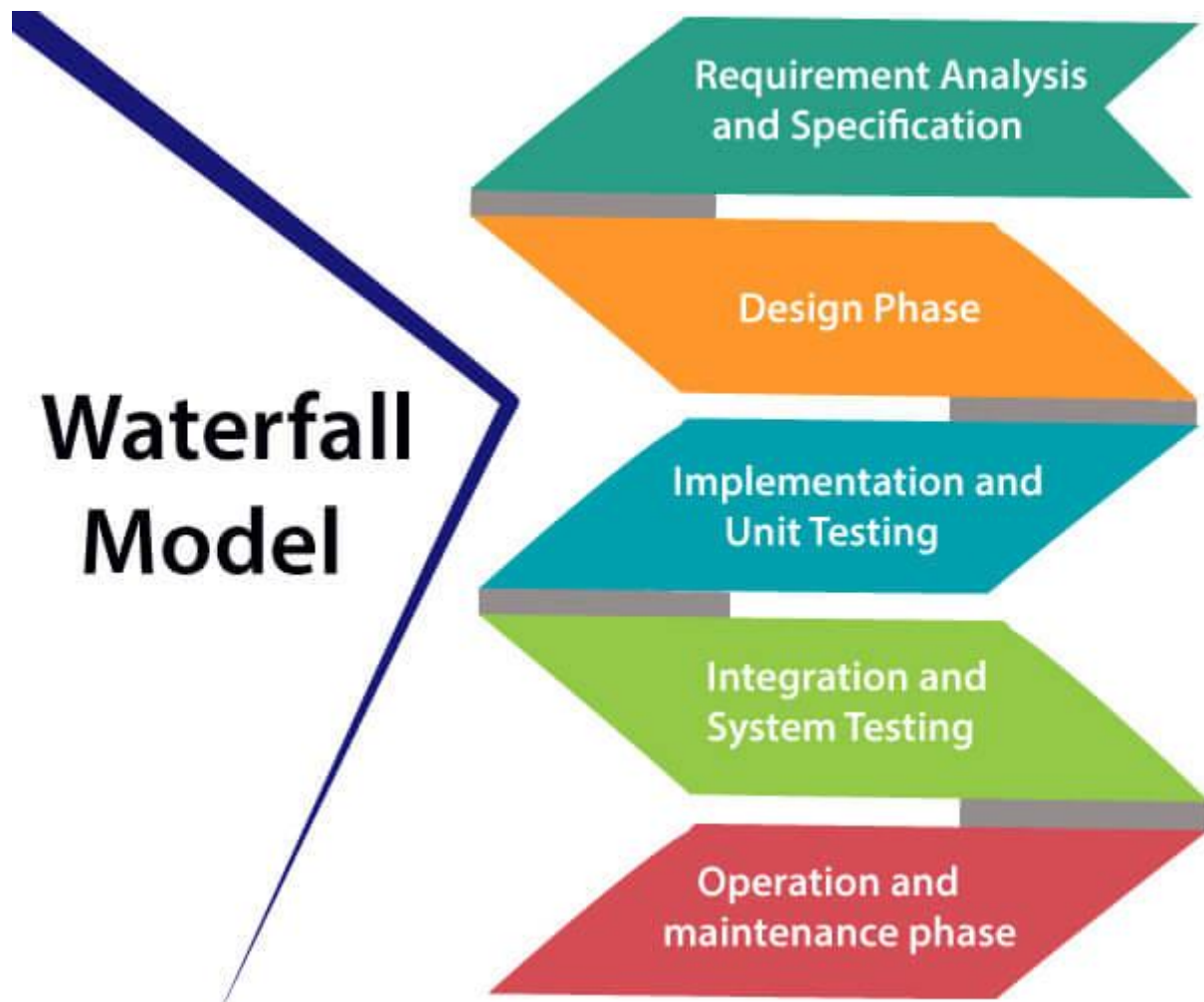
Figure 4:Waterfall model

4.4 Advantage of Waterfall model:

1. Each stage of development must be finished before moving on to the next.

2. Perfect for smaller projects with clearly specified criteria.

3. Before finishing each stage, they should do quality assurance testing.Such as verification and validation.

4. Every stage of the software development cycle involves full documentation.

5. Project is primarily dependent on the project team, with minimal client involvement.

6. Any modifications to software are made while it is being developed.

4.5 Disadvantage of Waterfall model:

1. Error may only be corrected during the phase.

2. For complicated projects where requirements change frequently, it is not acceptable.

3. The testing phase of the development process occurs relatively late.

4. The majority of developers' and testers' time is spent on documentation.

5. It is not possible to integrate client feedback into the ongoing development process.

6. Small modifications or faults that occur in the finished product might result in a lot of issues

4.6 V model

The V-model is a software development and testing methodology that provides a structured and systematic approach to the development and verification of software systems. It is called the V-model because of its distinctive V-shaped diagram, which illustrates the relationship between various development and testing phases.

The V-model is often considered an extension of the traditional waterfall model, as it emphasizes the importance of testing throughout the software development life cycle. It emphasizes the need for early involvement of testing activities and considers testing as an integral part of the entire development process.

4.7 When use this model:

It is often used for small and medium-sized projects where the client's needs are clear.

The V-model provides a clear and structured approach to software development and testing. It outlines the various phases and their corresponding testing activities, ensuring that the project progresses in a systematic manner. This helps in better planning, tracking, and control of the project.

The diagram of v model is below:

V- Model

Developer's life Cycle

Tester's Life Cycle

Verfication Phase

Validation Phase

Business req. Specification

System Req. Specification

High level Design

Low level Design

Coding

Acceptance Testing

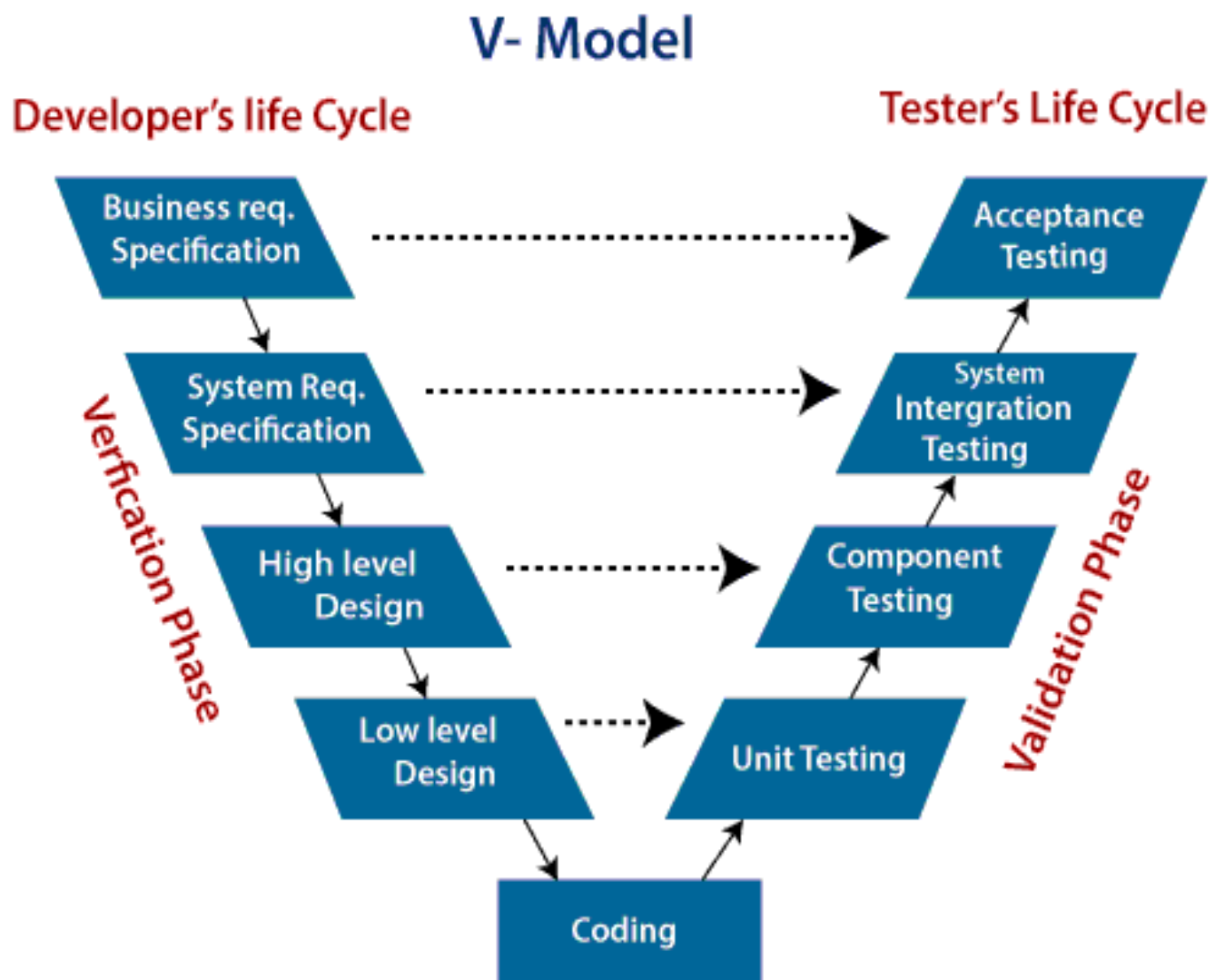System Intergration Testing

Component Testing

Unit Testing

Figure 5:V-Model

4.8 Advantage of V model:

1. It is simple and straightforward to use.

2. There have been significant modifications to detect issues early.

3. Test planning and design, as well as other test activities, are performed prior to coding in this step.

4. Small projects with simple criteria were appropriate for it.

5. It is a very disciplined approach for software development.

6. Each testing phase is directly linked to a specific development phase, ensuring that the testing activities are aligned with the requirements and design specifications.

4.9 Disadvantage of V model:

1. High and unpredictability risk exists.

2. Complex projects don't work well with this approach.

3. After finishing the test phase, it is challenging to return to the initial stage.

4. It is an unsuitable model for long-term projects.

5. The V-model's sequential nature limits the opportunities for iteration and feedback. Once a phase is completed, it is difficult to go back and make changes without disrupting the entire project schedule.

6. The V-model's sequential nature limits the opportunities for iteration and feedback. Once a phase is completed, it is difficult to go back and make changes without disrupting the entire project schedule.

4.10 Dynamic systems development model (DSDM)

The Dynamic Systems Development Model (DSDM) is an agile project delivery framework that provides a structured approach to software development and delivery. It is an iterative and incremental model that focuses on delivering high-quality systems within the constraints of time, cost, and quality. DSDM promotes collaboration, flexibility, and user involvement throughout the development process.

Dynamic Systems Development Model Diagram

Project Phases in DSDM Atern

Pre-Project

Feasibility

Foundations

Exploration

Incremental Deployment

Post-Project

Engineering

©Daffodil International University

Figure 6:DSDM Diagram
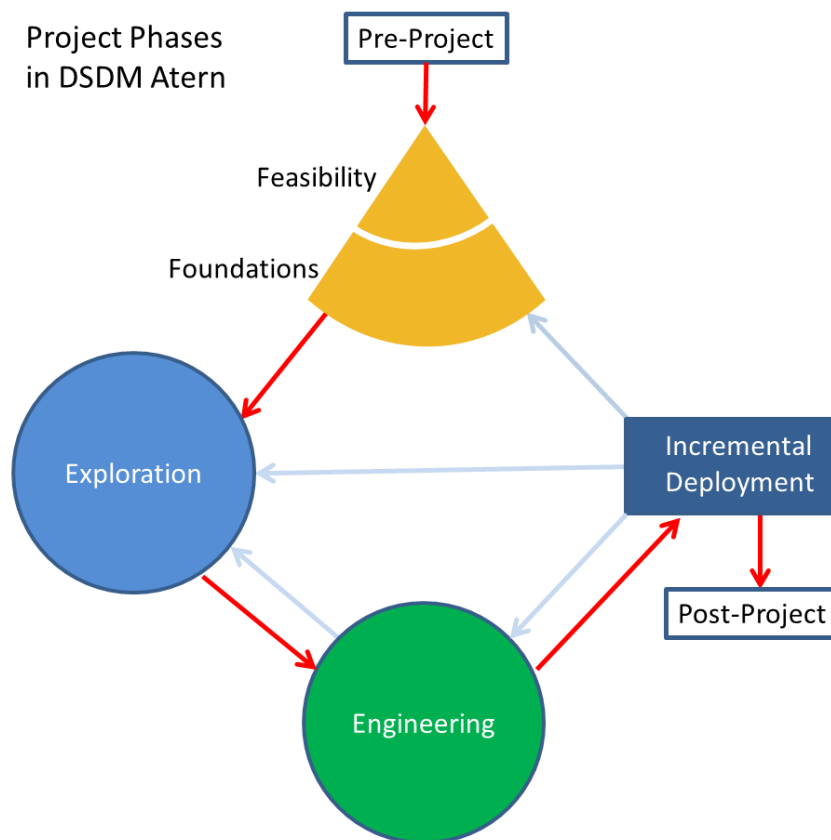
4.11 Advantage of Dynamic Systems Development Model :

1. Projects are completed on schedule while maintaining flexibility.

2. The organization as a whole can quickly understand the progress.

3. The DSDM model's basic form, business cases, ensures that initiatives provided have genuine commercial value. Disadvantage of Dynamic systems development model (DSDM)

4.12 Disadvantage of Dynamic Systems Development Model :

1. Usually, management expenses may be rather costly.

2. Installation fees may be too expensive for small enterprises.

 3. This is not appropriate for small organizations due to high management overhead and expensive deployment.

4. DSDM has a tendency to limit developer innovation. Even if more elegant methods are available, projects are likely to be performed exactly as requested.

4.13 Apply this methodology for my watch management system:

Applying the Dynamic Systems Development Model (DSDM) methodology for a watch management system website would involve the following steps:

1. Feasibility Study: Conduct a feasibility study to assess the viability and suitability of the watch management system website project. Identify the business objectives, scope, and potential risks. Evaluate technical and financial feasibility.

2. Business Study: Engage with stakeholders, including watch manufacturers, distributors, and users, to understand their needs, requirements, and expectations. Identify the key functionalities, user roles, and system constraints for the website.

3. Functional Model Iteration: Define the functional model by identifying the core features and functionalities required for the watch management system website. Prioritize the features based on business value and feasibility. Develop an initial functional prototype to gather feedback from stakeholders.

4. Design and Build Iteration: Based on the feedback received, refine and enhance the functional prototype. Collaborate with the development team to design and build the website iteratively, ensuring that the user interface is intuitive, visually appealing, and aligned with user requirements.

5. User Review: Engage users and stakeholders in regular reviews and demonstrations of the website's progress. Gather feedback and incorporate it into subsequent iterations. Conduct usability testing to validate the user experience and make necessary improvements.

6. Incremental Delivery: Plan for incremental delivery of the watch management system website by dividing the development into manageable iterations or time boxes. Each iteration should deliver a usable and valuable increment of the website.

7. Time boxing: Fix the time available for each iteration and adjust the scope and features based on the time constraints. Prioritize the most critical functionalities for each iteration to ensure that the website meets the desired timelines.

8. Continuous Integration and Testing: Implement continuous integration practices to ensure that code changes are integrated regularly and seamlessly. Conduct rigorous testing at each iteration, including functional testing, performance testing, and security testing, to ensure the quality and reliability of the website.

Chapter 5 Project Methodology

5.1 Sections of methodology

Pre-Project Phase: Several proposals are taken into consideration throughout the pre-project phase, with one being chosen in the end. At this stage, the project's finance requirements are estimated as well as the project's viability. Project lifecycle Phase: The objective of this phase is to manage the project as described in the Pre-Project phase. This phase has been subdivided into the following sub-phases:

1. Feasibility study: Instead than working on the project itself, the goal of these sessions is to develop concepts for project management. The solutions to such queries are going to be used direct the systems develop, assess the costs, and modify it to the DSDM model in order to provide a feasibility report.

2. Study of the business: The analysis of the business is divided into two sections: The business's flexibility is examined in the first report that is only done if system is created by DSDM method. I will try to select the aspects of the project and requirements of users. A selection of criteria will be created at the end of the study & these criteria is prioritized in terms of relevance for the development of the other criteria.

3. Iteration of the functional model: This iteration of the functional model collects and converts requirements stated in former rounds into functional models. Functional prototypes of the requirements are developed in order for the user may see how a system will work. This satisfies the first principle. Various groups of end users assess and validate the efficacy of functional prototypes.

4. Iteration of Design and Development: The purpose of Design & Build on Iteration phase is to transform the functional models developed in the earlier phase into finished, user-satisfactory products. In order to do this, design prototypes will be created and tested with users as in the previous stage.

5. Implementation: After the prototypes evaluated by the target audience, implementation step includes training for the prospective users.

Post-Project Phase: It is tested to ensure that everything works as it should during the post-project phase. Maintenance and software updates are required at this period

5.2 Implementation plans

This is the stage of the project where the completed application is released for usage by the general users. The new system should be made available for usage as soon as a fault with it is found and addressed. The settings, techniques, and release criteria are chosen in this section. If everything goes according, the new system is then tested and implemented.

Chapter 6 Planning

6.1 Project Plan

The execution and monitoring phases of a project are described by a group of formal documentation known as a project plan. In addition to addressing the project's overall scope, cost, and timetable, the strategy also includes risk management, resource management, and communication. Before we start any task on the actual execution the project, we must complete the project planning process to verify that risks and challenges are reduced and the project proceeds well. In these project management papers, each and every facet of your project is explained clearly. The project manager and his team as well as all other participants will benefit from project planning. Planning will enable the accomplishment of necessary tasks such as goal-setting, risk mitigation, deadline avoidance, and delivery of the desired product, technical assistance, or solution. This section will generally describe project planning, including WBS, time boxing, and a Gantt chart.

6.2 The Work Breakdown Structure of my project is:

The Work Breakdown Structure (WBS) is a hierarchical framework that organizes the tasks of a project. A project's structure is "broken down" into more manageable, smaller parts by the WBS. Depending on the requirements of the project, tasks may also be further subdivided into subtasks at each delivery. Work breakdown structure is a component of project lifecycle management that breaks down big, complicated projects into tiny, easily managed components that may be outsourced to certain people or teams.

Gantt chart :

In project management, the Gantt chart will benefit from a challenging project's complexity in order to successfully complete it. The Gantt chart will assist with scheduling and planning, making it simple to finish all tasks of all sizes. A horizontal bar indicating the amount of work that will be completed at each stage will be displayed as a result of various types of scheduling, as well as deadlines for each step of the system, start and completion dates of each assignment.
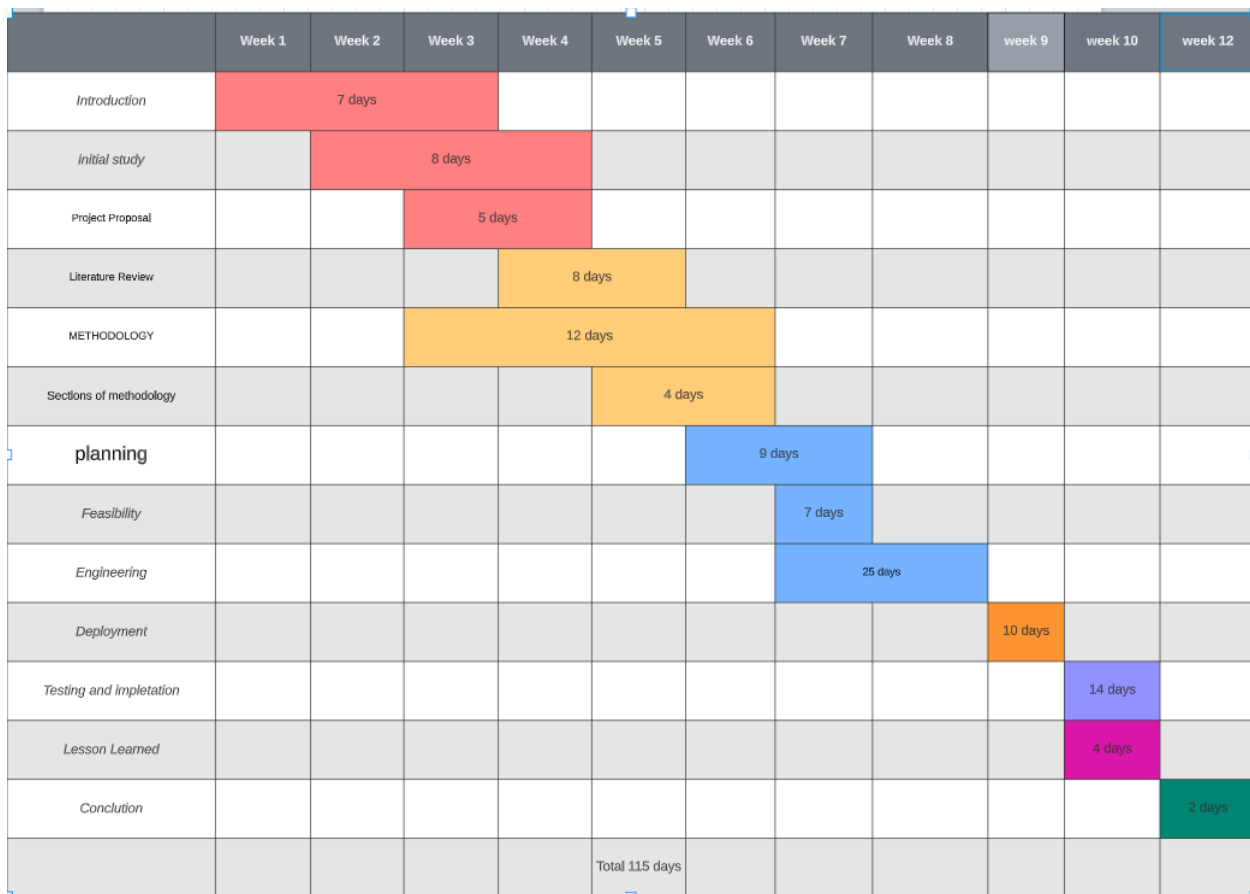
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	week 9	week 10	week 12
Introduction		7 days									
initial study			8 days								
Project Proposal			5 days								
Literature Review				8 days							
METHODOLOGY				12 days							
Sections of methodology					4 days						
planning						9 days					
Feasibility							7 days				
Engineering							25 days				
Deployment									10 days		
Testing and impletation										14 days	
Lesson Learned										4 days	
Conclution											2 days
				Total 115 days							

Figure 7:Gantt Chart

Chapter 7 Feasibility

For a project to be completed effectively, all pertinent factors, including technical, scheduling, legal, and economic considerations, must be taken into account, and the feasibility analysis gauges the likelihood of this accomplishment. One of the numerous factors that define a project's feasibility, including cost and return on investment is whether it will succeed in selling the product to clients or if it will not.

7.1 All Possible Types of Feasibility

Operational feasibility:

The users will find utilizing this project to be quite comfortable. Users of this web-based program can access it from any location with an internet connection. It is more affordable and simple to use.It will reduce customer pain and save time. . This is doable with just a computer and an internet connection. All departments will be interconnected, allowing for the development of interaction. The project has several overall operational benefits.

Technical Feasibility:

Hardware	Software	Database
Wifi router,Desktop	Xaamp, MsWord, Goggle sheet, Goggle crome, windows 11 pro,Php , Asp.net	My sql

Technology use:

From client side	Our server side
HTML,css,javascript,jquery	Asp.net,mongodibi

DSDM:

Dynamic System Development Method (DSDM) is vendor-independent, spans the whole project lifecycle, and offers best practice guidelines for on-time, in-budget project delivery. Its scalability has been demonstrated to handle projects of all sizes and for any business sector. DSDM agile foundation and vendor-independent system make it an excellent fit on this project.

Chapter 8 Foundation

8.1 The Problem Area Identification

Most software is developed in response to a particular situation. In order to create software that is usable, it is essential to identify the specific situation. Making an accurate solution is impossible without understanding the exact solution. Identification of the issue areas is crucial for system development because of this. Users provide information to the software organization in this area. The user's information will be more useful for developing the solution since he or she will use the solution and is aware of the problem for which they will utilize it. The problem can be found in a number of ways. Two of them are mentioned, and this solution uses both of them:

Interview

The interview is a procedure where two people or more converse directly about a specific issue or potential solutions. It's one of the greatest methods for identifying requirements. It can happen face to face with student, teacher, staff etc.

Observation

Applying observational techniques, actual behaviors are observed and then graded. But choosing what to look for and how to determine what defines a certain behavior is a significant difficulty the researcher faces. It is important to specify the behaviors that attract the researcher. so that the observer is aware of what to watch for and measure; ultimately, these may be added up to provide a score. There are many different kinds of observations, each with its own advantages and disadvantages, including naturalistic and controlled observation, covert and overt observation, participant and non-participant observation.i meet their shop to observe what their need is.and then I try to understand their requirements.

Questionnaires:

The questionnaire is another method of collecting data. It is a particular type of interview known as written type interview. Interviewers use this technique must have a list of questions to ask users in order to gather information.

Specific problem area identification and description:

This is much hard to attract the customer in this website.so we need to carefull about customer attractive and very easy to the customers because most of customer in our country cant much smart. For this we keep this site very simple.

8.2 Overall Requirement List

Functional Requirements:

1.Admin can easily make another admin

2. Admin can add a product or delete product.

3.Cusstomer can easily make order, give review or delete them.

4.Customer can log in with via goggle.

5.Customer review shown in home page with slide.

6.Admin or customer information sees in their own profile.

Non-Functional Requirements:

1. User friendly system.

2. Highly secure because there will be access to a lot of student and instructor data.

3. Secure private data and prevent others from editing it.

4. here need to be reports and summary accessible.

8.3 Which Technology to be implemented

The project done with web based . A web-based application is a type of software that utilizes the internet and web browsers to execute specific operations. Web applications uses client-side languages like JavaScript and HTML and server-side languages like PHP and ASP to store and retrieve data, respectively.

The benefit of web based project:

Web applications performed a range of devices and OS, provided the browser is appropriate. All users have access to the same version, therefore there are no incompatibilities. Due to its adoption, there is a decline in the usage of obtained illegally software paid-based online programs.
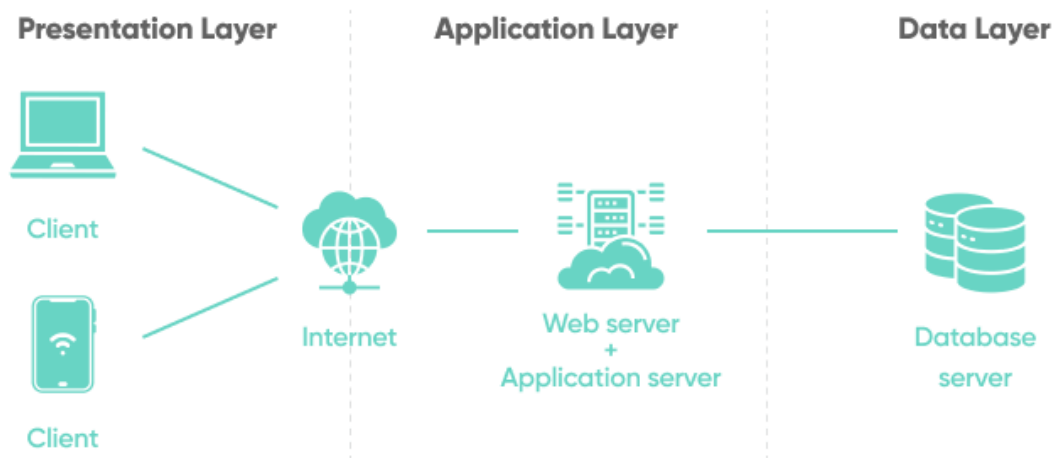
Figure 8:web based diagram

8.4 Recommendations and Justifications:

In order to develop the website of Ji watch management system, I provided security, usability, comfort, and efficiency a lot of consideration. Because of this, selecting an approach is necessary. I used HTML, CSS5, bootstrap, and JavaScript to make the project design excellent, be straightforward, and be simple to use. I made the website responsive and easy to use.

Chapter 9 Exploration

9.1 Old system to buy watch :
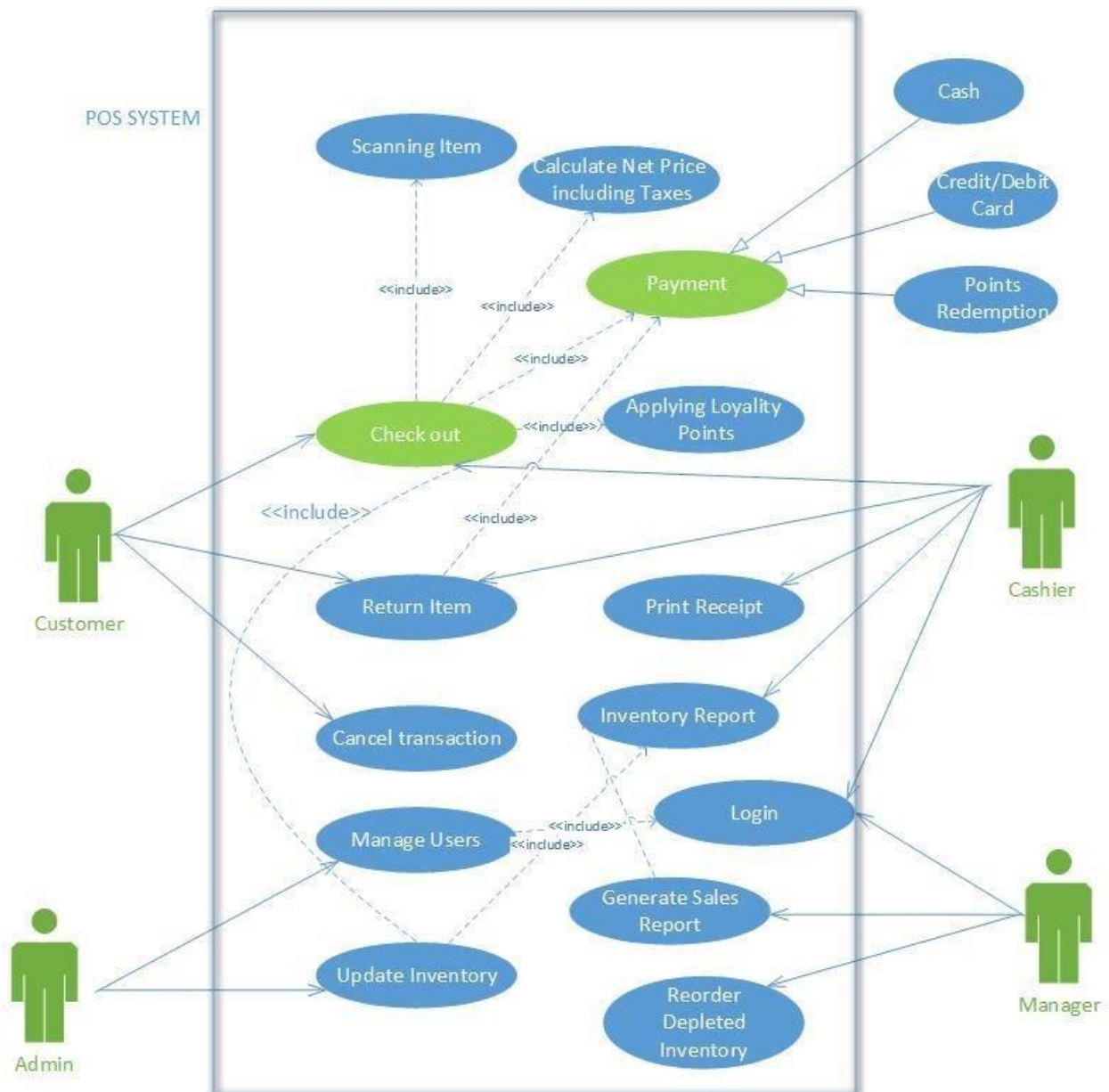
Figure 9:Old system watch sell

9.2 New System Customer activity program:
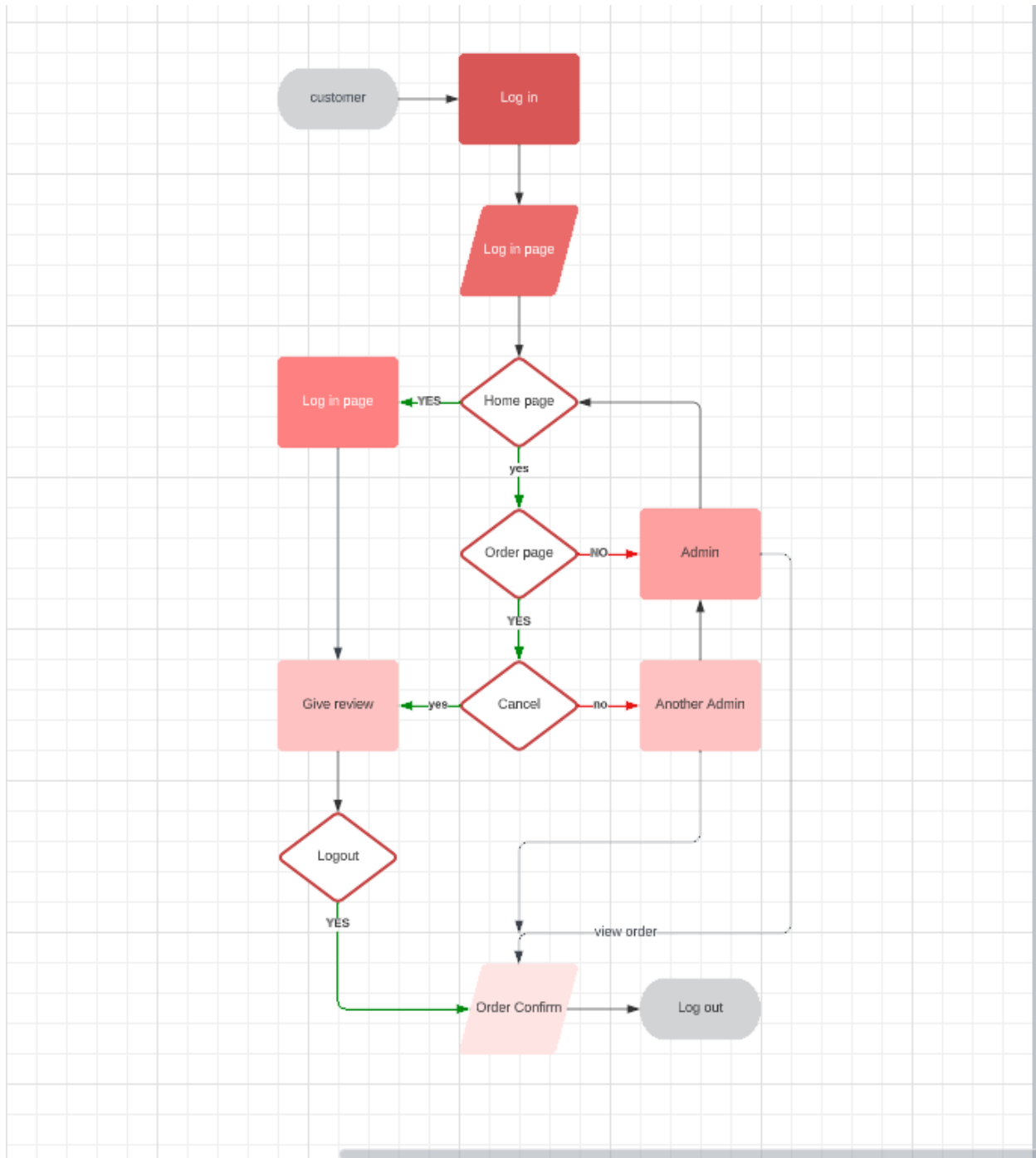
Figure 10:Customer activity

9.3 Admin activity program:
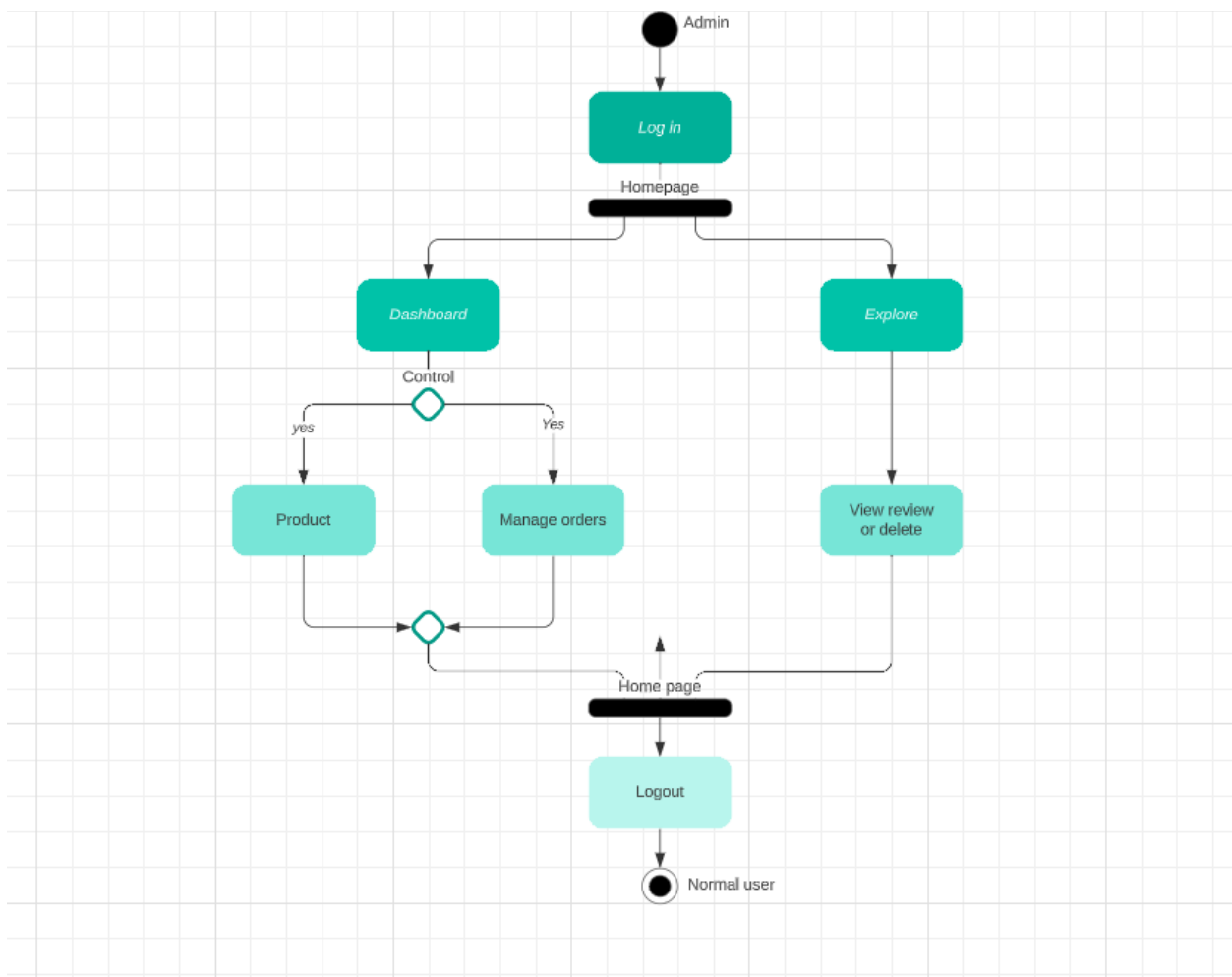
Figure 11:Admin activity

9.4 Full system use case Diagram:
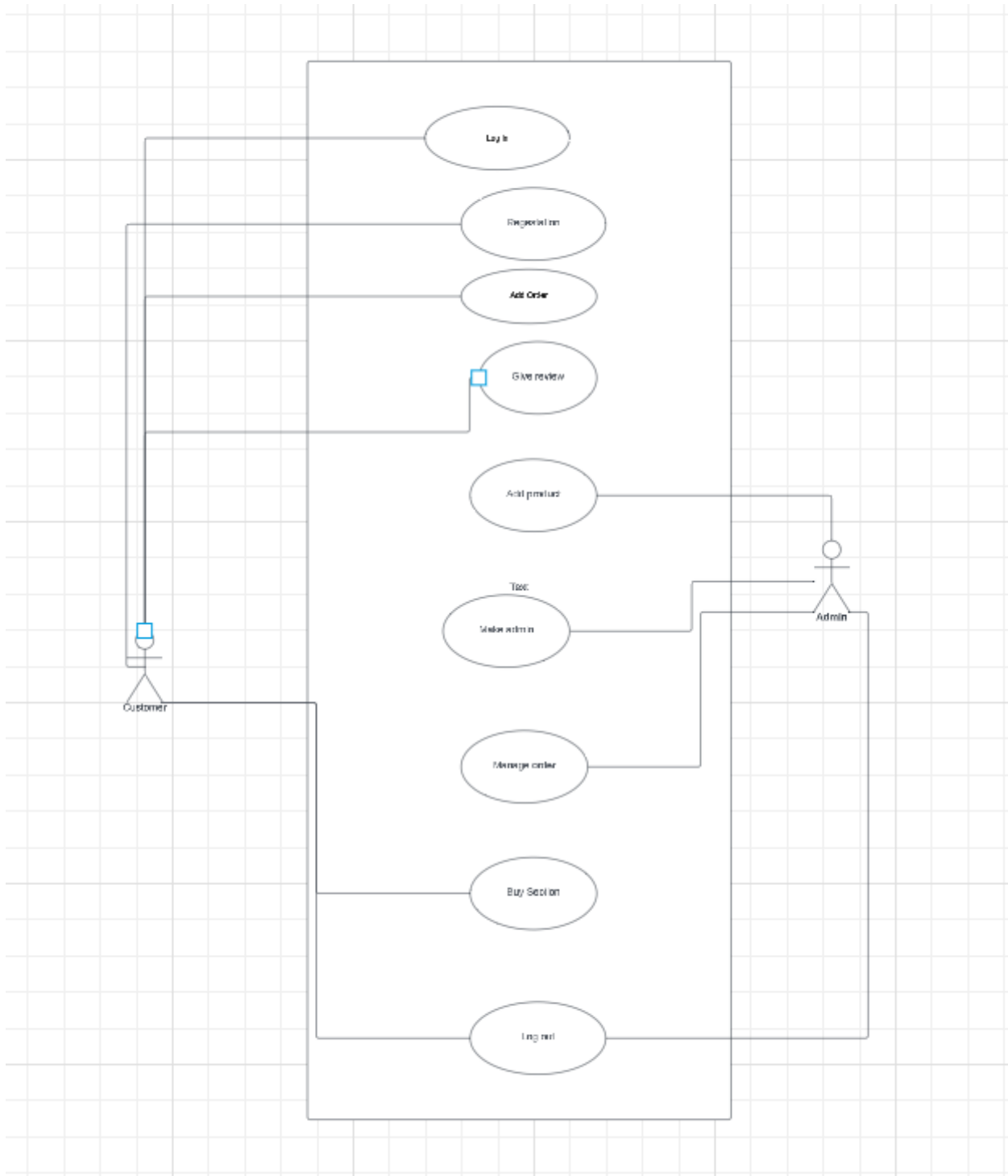
Figure 12:full system use case diagram

9.5 Some Feature of Ji watch management system:
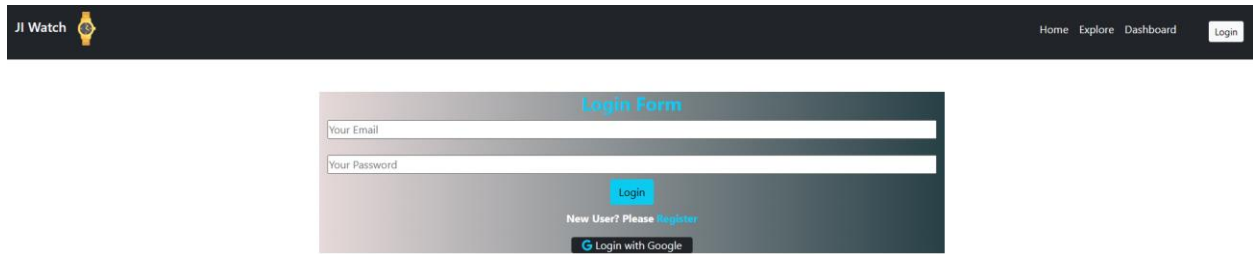
Log In page

Figure 13:Home page

Home page

THE ART OF BEING UNIQUE !!

Our Best Watch Product
⟲

Our Loveable Customer
⟲

Professional Team Members

Our Best Watch Product

Lange & Sohne

Description: Arguably one of the most popular models produced by the manufacturer, the Lange I is a timepiece that's as close to perfection as you're going to get.

Price:$1500

Buy now

Armin Strom

Description: Like a window to heaven, the partially skeletonized dial offers a glimpse into the architecture and artistry of the ARM09 movement within.

Price:$2500

Buy now

MontaBlanc

Description: As rare as beluga caviar, as revered as a Rolls Royce, this timepiece is about as superlative as it gets. What's exceptionally unique and attractive about this offering is that despite its complicated architecture.
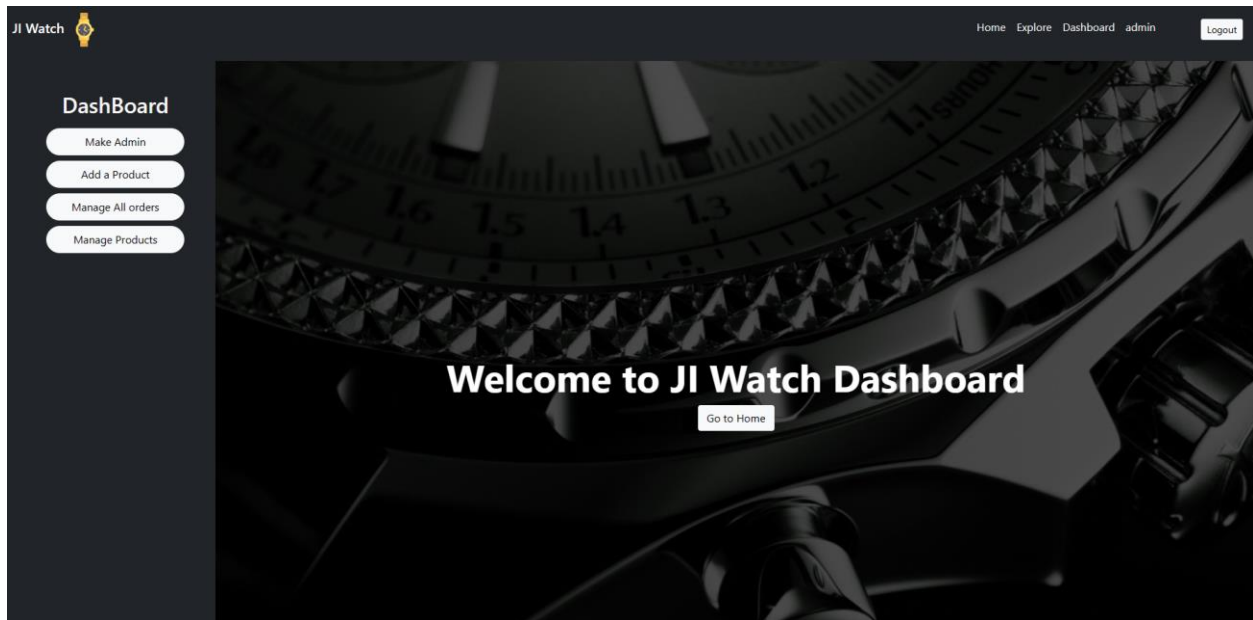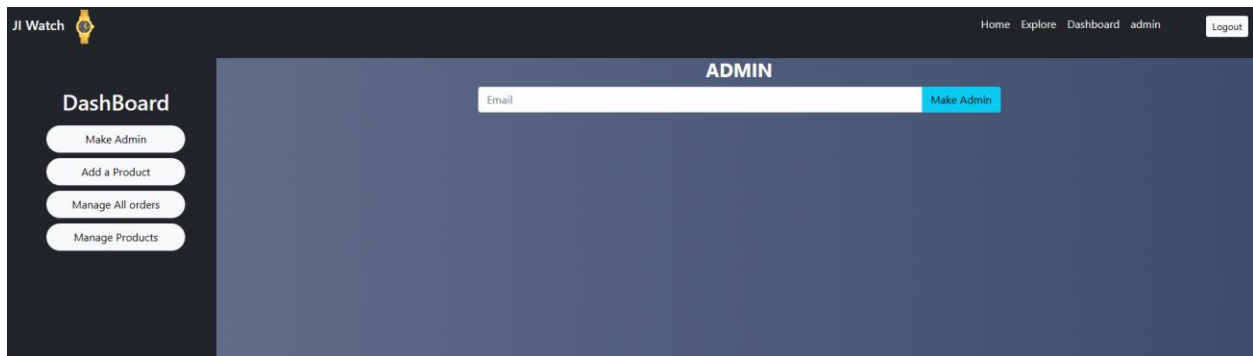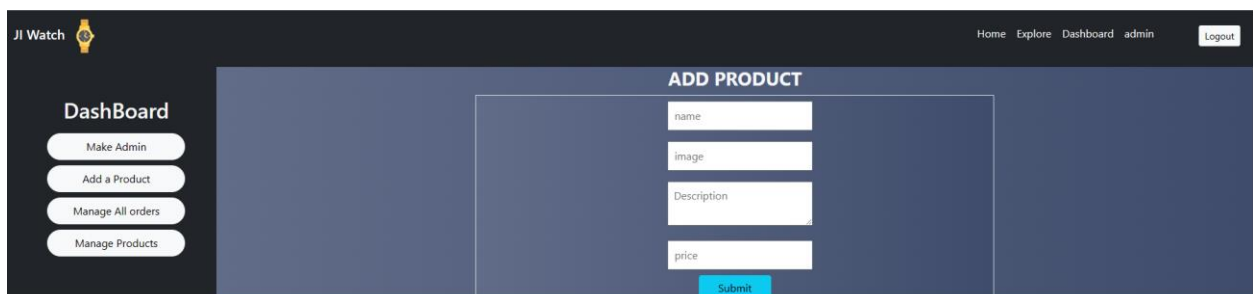
Price:$3500

Buy now

Figure 14:Explore Page

Figure 15:Order Page

Figure 16:Admin Dashboard Picture

Make admin

Add product

Figure 17:Order Manage Pic

Figure 18:Product Manage Pic

Regestation Page(you can log in via goggle)

Customer Dash board

Review System

Chapter 10 Deployment/Development

10.1 Core Module Coding Samples:

> 📁 .firebase
> 📁 public
∨ 📂 src
  > 📁 Components
  > 📁 Contexts/AuthProvider
  > 📁 Firebase
  > 📁 hooks
  > 📁 images
    📄 App.css
    📄 App.js
    📄 App.test.js
    📄 index.css
    📄 index.js
    📄 logo.svg
    📄 reportWebVitals.js
    📄 setupTests.js
  📄 .firebaserc
  📄 .gitignore
  📄 README.md
  📄 firebase.json
  📄 package-lock.json
  📄 package.json
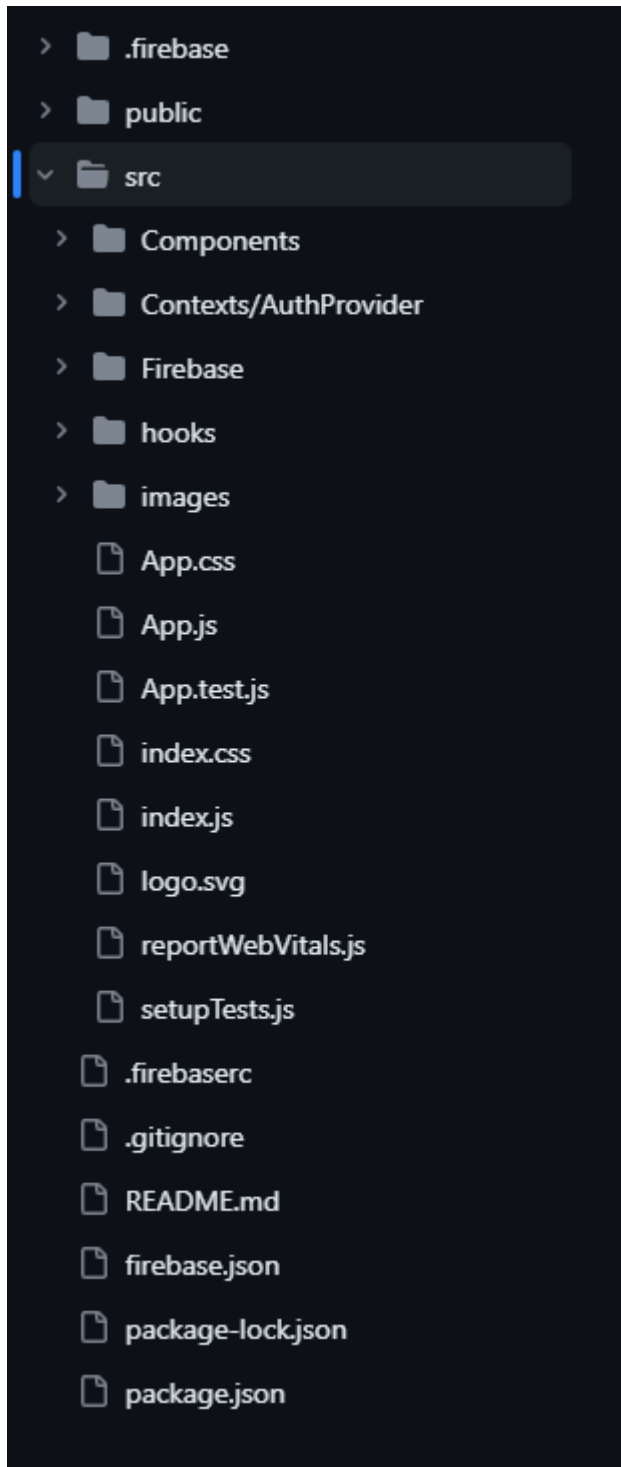
Figure 19:Core module Sample

Login Module

```
1    import React, { useState } from 'react';
2    import { Link, useLocation, useHistory } from 'react-router-dom';
3    import useAuth from '../../hooks/useAuth';
4
5
6 ∨  const Login = () => {
7        const [loginData, setLoginData] = useState({});
8        const { user, loginUser, isLoading, signInWithGoogle, authError } = useAuth();
9
10       const location = useLocation();
11       const history = useHistory()
12
13 ∨     const handleOnChange = e => {
14           const field = e.target.name;
15           const value = e.target.value;
16           const newLoginData = { ...loginData };
17           newLoginData[field] = value;
18           setLoginData(newLoginData);
19       }
20
21       const handleLoginSubmit = e => {
22           loginUser(loginData.email, loginData.password, location, history);
23           e.preventDefault();
24       }
25
26       const handleGoogleSignIn = () => {
27           signInWithGoogle(location, history)
28       }
29
30       return (
31           <div className="container mt-4 bg-dark w-50" style={{ background: `linear-gradient(to right, #e6dada, #274046)` }}>
32               {isLoading && <div className="spinner-border text-info text-center" role="status">
33                   <span className="visually-hidden">Loading...</span>
34               </div>}
35               <h3 className="mt-5 text-center text-info fw-bolder ">Login Form</h3>
36               <form onSubmit={handleLoginSubmit}>
37                   <input
38                       style={{ width: '100%' }}
39                       label="Your Email"
40                       name="email"
41                       placeholder="Your Email"
42                       onBlur={handleOnChange}
43                   />
```

Figure 20:log inmodule

```
                        onBlur={handleOnChange}
                    />
                    <br />
                    <input className="mt-4"
                        style={{ width: '100%' }}
                        label="Your Password"
                        type="password"
                        name="password"
                        placeholder="Your Password"
                        onBlur={handleOnChange} />
                    <br />
                    <button className="btn btn-info mt-2 mb-2" type="submit">Login</button>

                    {user?.email && <div className="alert alert-success" role="alert">
                        User created successfully!
                    </div>}
                    {authError && <div className="alert alert-danger" role="alert">
                        {authError}
                    </div>}

                    <p className="text-white fw-bolder">New User? Please <Link className="text-decoration-none text-info" to="/register">Register</Link></p>
                </form>

                <button className="btn-dark border-0 px-3  rounded text-nowrap" onClick={handleGoogleSignIn}><i className="fab fa-google text-info" ></i> Login with Google
                </button>

            </div>
        );
};

export default Login;
```

```
1     import React, { useState } from "react";
2     import "./MakeAdmin.css";
3
4 ∨   const MakeAdmin = () => {
5       const [email, setEmail] = useState("");
6       const [success, setSuccess] = useState(false);
7
8       const handleOnBlur = (e) => {
9         setEmail(e.target.value);
10      };
11
12 ∨    const handleAdminSubmit = (e) => {
13        const user = { email };
14        fetch("https://watch-server.onrender.com/users/admin", {
15          method: "PUT",
16          headers: {
17            "content-type": "application/json",
18          },
19          body: JSON.stringify(user),
20        })
21          .then((res) => res.json())
22          .then((data) => {
23            if (data.modifiedCount) {
24              console.log(data);
25              setSuccess(true);
26            }
27          });
28
29        e.preventDefault();
30      };
31      return (
32        <div
33          className="adu"
34          style={{ background: `linear-gradient(to right, #606c88, #3f4c6b)` }}
35        >
36          <h3 className="text-white fw-bolder ">ADMIN</h3>
37          <form onSubmit={handleAdminSubmit}>
38            <div class="input-group mb-3 mx-auto" style={{ width: "50%" }}>
39              <input
40                className="form-control "
41                label="Email"
42                type="email"
```

Figure 21:Makeadmin Core module

Manage Order Code Sample:

```
1      import React, { useEffect, useState } from "react";
2
3 ∨    const ManageAllOrders = () => {
4        const [status, setStatus] = useState([]);
5        const [manageOrders, setManageOrders] = useState([]);
6        useEffect(() => {
7          fetch("https://watch-server.onrender.com/manageAllOrders")
8            .then((res) => res.json())
9            .then((data) => setManageOrders(data));
10       }, []);
11 ∨     const handleStatus = (_id, pd) => {
12         pd.status = "shipped";
13
14         fetch(`https://watch-server.onrender.com/manageAllOrders/${_id}`, {
15           method: "PUT",
16           headers: { "content-type": "application/json" },
17           body: JSON.stringify(pd),
18         })
19           .then((res) => res.json())
20           .then((data) => {
21             if (data?.modifiedCount) {
22               setStatus(false);
23             }
24           });
25       };
26
27       return (
28         <div style={{ background: `linear-gradient(to right, #606c88, #3f4c6b)` }}>
29           <div className="mb-4 " style={{ width: "100%" }}>
30             <div>
31               <table className="table-borderless text-white w-100">
32                 <thead>
33                   <tr className="border">
34                     <th scope="col">Name</th>
35                     <th scope="col">Adress</th>
36                     <th scope="col">Action</th>
37                   </tr>
38                 </thead>
39
40                 {manageOrders?.map((pd) => (
41                   <tbody key={pd._id}>
42                     <tr>
```

Figure 22:Make Order sample

Manage Product Code sample:

```
1      import React, { useEffect, useState } from "react";
2
3 ∨    const ManageProducts = () => {
4        const [manageProducts, setManageProducts] = useState([]);
5        useEffect(() => {
6          fetch("https://watch-server.onrender.com/manageProducts")
7            .then((res) => res.json())
8            .then((data) => setManageProducts(data));
9        }, []);
10
11 ∨     const handleRemove = (id) => {
12         const url = `https://watch-server.onrender.com/manageProducts/${id}`;
13         const isReady = window.confirm(
14           "are you sure you want to delete this order?"
15         );
16         if (isReady) {
17           fetch(url, {
18             method: "DELETE",
19           })
20             .then((res) => res.json())
21             .then((data) => {
22               console.log(data);
23               if (data.deletedCount) {
24                 alert("Order deleted succssfully");
25                 const remaining = manageProducts.filter(
26                   (order) => order._id !== id
27                 );
28                 setManageProducts(remaining);
29               }
30             });
31         }
32       };
33
34       return (
35         <div style={{ background: `linear-gradient(to right, #606c88, #3f4c6b)` }}>
36           <div>
37             <table className="table table-borderless text-white">
38               <thead>
39                 <tr className="border">
40                   <th scope="col">Name</th>
41                   <th scope="col">Price</th>
42                   <th scope="col">Action</th>
43                 </tr>
```

Figure 23:Manage Order COde Sample

```
45
46              {manageProducts?.map((pd) => (
47                <tbody key={pd._id}>
48                  <tr>
49                    <td>{pd?.name}</td>
50                    <td>{pd?.price}</td>
51
52                    <td>
53                      <button onClick={() => handleRemove(pd._id)} className="btn">
54                        <i className="fas fa-trash text-danger"></i>
55                      </button>
56                    </td>
57                  </tr>
58                </tbody>
59            ))}
60          </table>
61        </div>
62      </div>
63    );
64  };
65
66  export default ManageProducts;
```

My Order Code sample:

```
1       import React, { useEffect, useState } from "react";
2       import useAuth from "../../hooks/useAuth";
3       import "./MyOrders.css";
4       //added
5       const MyOrders = () => {
6         const { user } = useAuth();
7         const [myOrder, setMyOrder] = useState([]);
8         useEffect(() => {
9           const url = `https://watch-server.onrender.com/orders?email=${user.email}`;
10          fetch(url)
11            .then((res) => res.json())
12            .then((data) => setMyOrder(data));
13        }, []);
14
15        const handleDelete = (id) => {
16          const url = `https://watch-server.onrender.com/orders/${id}`;
17          const isReady = window.confirm(
18            "are you sure you want to delete this order?"
19          );
20          if (isReady) {
21            fetch(url, {
22              method: "DELETE",
23            })
24              .then((res) => res.json())
25              .then((data) => {
26                console.log(data);
27                if (data.deletedCount) {
28                  alert("Order deleted succssfully");
29                  const remaining = myOrder.filter((order) => order._id !== id);
30                  setMyOrder(remaining);
31                }
32              });
33          }
34        };
35
36        return (
37          <div
```

Figure 24:my Order Code sample

Register code sample:

```
1      import React, { useState } from 'react';
2      import { Link, useHistory } from 'react-router-dom';
3      import useAuth from '../../hooks/useAuth';
4
5 ∨    const Register = () => {
6          const [loginData, setLoginData] = useState({});
7
8          const history = useHistory();
9          const { user, registerUser, isLoading, authError } = useAuth();
10
11 ∨       const handleOnBlur = e => {
12             const field = e.target.name;
13             const value = e.target.value;
14             const newLoginData = { ...loginData };
15             newLoginData[field] = value;
16             setLoginData(newLoginData);
17         }
18
19 ∨       const handleLoginSubmit = e => {
20             if (loginData.password !== loginData.password2) {
21                 alert('Your password did not match')
22                 return
23             }
24             registerUser(loginData.email, loginData.password, loginData.name, history);
25             e.preventDefault();
26         }
27         return (
28             <div className="container mt-4 bg-dark w-50" style={{ background: `linear-gradient(to right, #4b6cb7, #182848)` }}>
29                 <h3 className="mt-5 text-center text-info fw-bolder ">Register Form</h3>
30                 {
31                     !isLoading && <form onSubmit={handleLoginSubmit}>
32                         <input className="mb-4"
33                             style={{ width: '100%' }}
34                             label="Your Name"
35                             name="name"
36                             type="text"
37                             placeholder="Your Name"
38                             onBlur={handleOnBlur}
39                             required
40                         />
41                         <br />
42                         <input
43                             style={{ width: '100%' }}
```

Figure 25:regester code sample

```
44                    label="Your Email"
45                    name="email"
46                    type="email"
47                    placeholder="Your Email"
48                    onBlur={handleOnBlur}
49                    required
50                 />
51                 <br />
52                 <input className="mt-4"
53                    style={{ width: '100%' }}
54                    label="Your Password"
55                    type="password"
56                    name="password"
57                    placeholder="Your Password"
58                    onBlur={handleOnBlur} />
59                 <br />
60
61                 <input className="mt-4"
62                    style={{ width: '100%' }}
63                    label="ReType Your Password"
64                    type="password"
65                    name="password2"
66                    placeholder="Re-type Password"
67                    onBlur={handleOnBlur} />
68                 <br />
69                 <button className="btn btn-info mt-2 mb-2" type="submit">Register</button>
70
71
72                 <p className="text-white fw-bolder">Already have a account? Please <Link className="text-decoration-none text-info" to="/login">login</Link></p>
73
74              </form>}
75         {isLoading && <div className="spinner-border text-info" role="status">
76            <span className="visually-hidden">Loading...</span>
77         </div>}
78         {user?.email && <div className="alert alert-success" role="alert">
79            User created successFully!
80         </div>}
81         {authError && <div className="alert alert-danger" role="alert">
82            {authError}
83         </div>}
84
```

```
74              </form>}
75         {isLoading && <div className="spinner-border text-info" role="status">
76            <span className="visually-hidden">Loading...</span>
77         </div>}
78         {user?.email && <div className="alert alert-success" role="alert">
79            User created successFully!
80         </div>}
81         {authError && <div className="alert alert-danger" role="alert">
82            {authError}
83         </div>}
84
85
86      </div>
87   );
88 };
89
90 export default Register;
```

10.2 Possible problem breakdown

To perform a large task properly at one time is pretty difficult. It will also take a lot of time. It will be simpler to complete the task and will take less time overall if it can be divided into manageable chunks. The task will be carried out effectively and professionally.

Some problem are:

1. Redundant data can occur.

2. normal error handling

3. Inadequate requirement analysis.

4. Insufficient testing

5. Insufficient support from users

To avoid these possible problems, I have taken some steps like:

1. Data redundancy was avoided when creating the data structure.

2. Included every type of error handling.

3. Developed a requirement analysis and validated it with actual users.

4. Several techniques were tested.

5. Real users were surveyed on user approval.

Chapter 11 – Implementation

Training

Software is developed to ensure working less painful and to automate as much as possible. A good system must be user-friendly so that users can operate it without difficulty. However, even if the system is user friendly, certain users could still find it difficult to utilize it due to a lack of IT experience. Any large system should include a training phase for this reason. It's very easy to use my system. But in the school where I'll be selling it, I'll also be doing a training phase. In order for them to train others in the school, I need to train a group of school representatives. In a real demo server area, I will show them how the entire system functions and is usable. I will test their ability to perform that flawlessly following my presentation. The training step can be effectively completed when you have taught them.

User	Training	Time	Comment
Admin	1.User group setup 2.Create product 3.manage order 4.Make order	2 hours	Admin can do all of this,Create update or delete
Customer	1.Make order 2.Give review 3.cancel order 4. rate wacth	1 hours	Customer can easily do this.

Load balance:

A system's optimization against the impact of its users is referred to as load balancing. The number of users hit represents the number of concurrent users and the duration of the system. This is referred to as load equalization and load balancing. It allows the system to continue operating normally by distributing the demand over different server.

Chapter 12 – Lesson Learned

12.1 What I have learned:

The JI watch management system is a web based project.Since I was in Ego Digital intern I contributed to this project as a designer and developer. Working on this project taught me a lot of things that will be very useful to me in the future. I have first learnt how to work with software architecture that has been designed by others. While I was working on the design, I discovered several fresh, previously unknown design methods. I have studied them and incorporated them into the method I designed. such as how to make a design more user-friendly and how to create a design using templates. I have prepared for the system's development when the design is finished. I have researched them and implemented them into the method I designed. such as how to make a design more user-friendly and how to create a design using templates. I have prepared for the system's development when the design is finished. Designing a database is a prerequisite before building any system. A system cannot be finished if an appropriate database architecture is not implemented. I have experience working with modest databases that were required for a minor project. However, I have gained experience with a large database and several data tables in this system.

12.2The Problems I Have Faced:

While working on this project, you may encounter various challenges. Here are some common problems faced during web development:

1. **Compatibility issues:** Different web browsers and devices may interpret and display websites differently. Ensuring cross-browser and cross-device compatibility can be a complex task.

2. **Responsive design:** Designing a website that adapts seamlessly to different screen sizes and resolutions requires careful planning and implementation.

3. **Performance optimization:** Users expect websites to load quickly and be responsive. Optimizing code, images, and other assets to reduce load times can be a significant challenge.

4. **Security vulnerabilities:** Web applications are susceptible to various security threats, such as cross-site scripting (XSS), SQL injection, and cross-site request forgery (CSRF). Implementing robust security measures and staying updated with the latest security practices is crucial.

5. **Scalability:** As the user base grows, the website should be able to handle increased traffic and data without experiencing performance issues. Scalability planning from the start can help prevent problems in the future.

6. **Content management**: If the website involves dynamic content, implementing a user-friendly content management system (CMS) can be challenging. Balancing ease of use for content creators with the flexibility and power required for the website can be a delicate task.

And the main problem is client satisfy.Its very hard to satisfy the client requirement.

12.3 What solutions occurred:

Every issue has a solution that may be used to resolve it. I have created a solution for each of the problems I stated above and finished the system. I created a solution first and foremost for utilizing software architecture created by others. As I worked on the design, I came across a number of novel and unexpected core concepts. When creating this system, I acquired them and included them. For instance, how to use templates to construct a design and how to improve usability. I have prepared for the system's development after the design is finished. Before any system can be created, a database must be built.

Chapter 13 – Conclusion

13.1 Summary of the project:

A watch management website is an online platform designed to assist watch enthusiasts and collectors in organizing, tracking, and managing their watch collections. The website serves as a centralized hub where users can store and access information about their timepieces, enabling them to keep a comprehensive record of their watches in one convenient location.

The core functionality of the watch management website revolves around providing users with a user-friendly interface to add, edit, and view details of their watches. Users can create individual profiles and catalogs for each watch, including information such as brand, model, serial number, purchase date, and price. They can also upload high-resolution images to showcase the watches in their collection.

13.2 Goal of the project:

There are some goals of my project.

They are:

1. Computerize all of the systems.

2. Reduce the amount of time spent and the potential for inaccuracy.

3. The management of every system is automated.

4. centralized database administration.

5. Simple controls for the system's operator.

6. No paperwork is necessary

13.3 Success of the project:

I have develop this whole system for a watch sells website.A watch shop can easily use this.there are some future work on this if the client need this.This is my success that I complete this project for the Ji shop.they are very Happy with this project.There are some future work that I worked for.its their payment identification method foe the customer and admin.
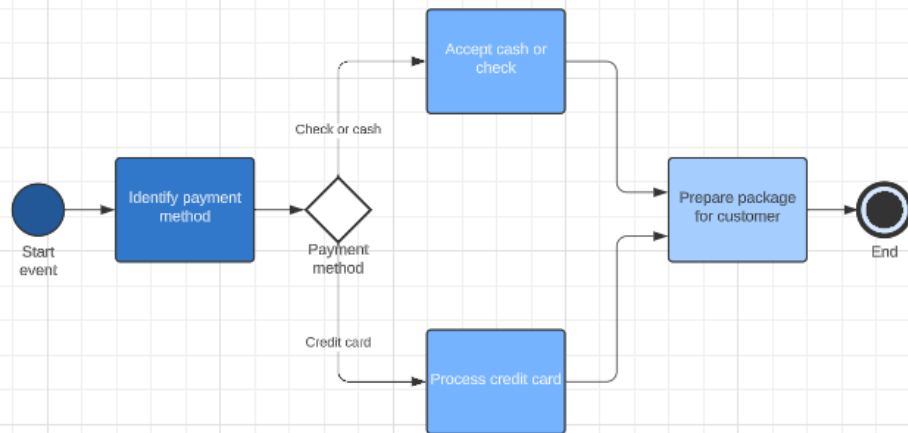
The future work diagram is below:

Figure 26:Future plan

13.4 What I have done in the documentation (stages / activities / plans)

Since this documentation is for an academic project, I had to first complete it according to the structure for academic projects. In this documentation, I have detailed everything I did for the project. I started by writing an introduction to the system in the documentation, then I began the preliminary research for this project and gradually completed all the necessary processes for this system. All of the tables in this documentation were made using Microsoft Word, and any figures or diagrams were made using a third party program.

13.5 what Experience I have gather:

I learned a lot while working on this project, even though I was inexperienced with all the technology it required. I had to re-learn these ideas as well as how to implement them and finish the project fully. Now I know how to collaborate on software architecture designed by others. I discovered some new design approaches that I was not familiar with while I was still designing. I studied them and incorporated them into the system while I was creating it. such as how to use templates to create designs and how to make designs more user-friendly.Thanks my supervisor **Mr. Israfil sir** And my trainer **Md arif Rabbani** being with me and always helps me to go forward.I am very lucky that I got this kind hearted man.

Intern

By Shuvasish kar

Submission Date:10 july 2023

Submission Id: 2129001640

File name: 192-16-450

Word Count: 6147

Character Count: 43265

Plagiarism Report

Intern

ORIGINALITY REPORT

18%	14%	2%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	8%
2	Submitted to Daffodil International University Student Paper	4%
3	Submitted to University of North Texas Student Paper	1%
4	Submitted to University of Greenwich Student Paper	1%
5	Submitted to NCC Education Student Paper	1%
6	Submitted to The Robert Gordon University Student Paper	1%
7	Submitted to Oxford Brookes University Student Paper	<1%
8	Submitted to Buckinghamshire Chilterns University College Student Paper	<1%
9	innovateinsights.net Internet Source	<1%

10	s3-ap-southeast-2.amazonaws.com Internet Source	<1%
11	Submitted to Sheffield Hallam University Student Paper	<1%
12	www.slideshare.net Internet Source	<1%
13	www.buzzle.com Internet Source	<1%