Title of the project

"Prison Management System"

Course: **Project (Phase II)**

Course Code: **CIS499**

Department of Computing and Information System (CIS)

Submitted By:

Md Sakib

ID: **182-16-328**

Supervised By:

Sarwar Hossain Mollah

Department head of CIS

APPROVAL

This Project titled **"Prison Management System."**, Submitted by **Md. Sakib**, ID No: **182-16-328** to the Department of Computing and 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 on13-01-2025.

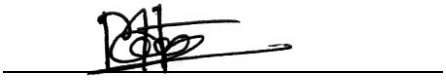
BOARD OF EXAMINERS

Md Sarwar Hossain Mollah **Chairman**

Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

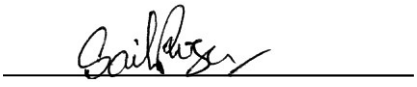
Md.Nasimul Kader **Internal Examiner**

Assistant Professor
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University



Md. Mehedi Hassan **Internal Examiner**

Lecturer (Senior Scale)
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University



Ahmed Saif Reza **External Examiner**

Managing Director & Chief Technology Officer
Medico Bio Limited

Declaration

I hereby declare that; this Internship project has been done by me under supervision of **Md Sarwar Hossain Mollah** 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 Sarwar Hossain Mollah
Associate Professor and Head
Department of CIS
Daffodil International University

Submitted By

Name: Md, Sakib
ID: 182-16-328
Department of CIS
Daffodil International University

ABSTRACT

In any nation Prison is one of the active and vital areas. In every jail in Bangladesh, daily numerous transaction and procedure is occurred which is extremely tough to manage. The supervisor (authority) confronts a lot of challenges since, they have not any suitable management system for jail. They experience plenty of challenges to monitor the prisoner and to manage the complete facility. Not only the authorities but also the prisoner and the visitors also suffer a lot of challenges for missing of the management system. The major purpose of my project is to design a jail management system by which the authorities may administer the prison conveniently. The government of Bangladesh, especially the authority of Bangladesh jail will be a lot of advantages. The prisoner and the visitor will also confront

fewer problems. The financial transaction will be simpler and quicker. Every prisoner's data will be easy to follow and their every move in the jail will be tracked. By a click, the authorities may discover any information about any prisoner and any facts about the jail. So, it will be simpler, more adequate and less time demanding. Prison Management System is meant to simplify and improve the administrative procedures inside correctional institutions. The project focuses on establishing a consolidated and secure platform to handle important operational features such as prisoner data, personnel information, visitor logs, and facility administration.

The system employs modern database design to enable efficient data storage and retrieval, boosting decision-making and decreasing human work. Features like role-based access, automatic reporting, and real-time monitoring increase transparency and security inside the facility.

By automating common processes, the system decreases human error, enhances resource management, and assures compliance with legal and institutional requirements. This project intends to contribute to a better structured and successful jail administration while ensuring the security and integrity of sensitive data.

Table of Contents

Chapter 1 – Introduction

1.1. The system developed:

In our country Prison is one of the dynamic and crucial industries. The authorities encounters a lot of challenges because, they have not any adequate management structure for prison. They have a hard time keeping track of prisoners and maintaining the facility in excellent shape. Due to the lack of a management structure, not only do the authorities face various issues, but so do the convicts and visitors. The fundamental purpose of my project is to build a system for jail administration that will support the smooth operation of the prison. The government of Bangladesh, notably the authority of Bangladesh prison will be a lot of benefits. The prisoner and the visitor will also have less troubles. With this technique, it will be much easier to maintain tabs on each convict and their locations within the facility. Any information about the prison or inmate's record is available to the authorities with the click of a button. So it will be easier, more sufficient and less time consuming

1.2. Justification for the method or used framework:

I have used Laravel to develop the system . Laravel, being a PHP framework, provides some appealing benefits for constructing a strong, scalable, and secure online application like a Prison Management System. Here's why Laravel is an amazing alternative . First comes security in jail . Prison administration systems handle sensitive data, such as prisoner information, personnel details, and visitor logs. Laravel has built-in security features including hashed passwords (via bcrypt), CSRF protection, and SQL injection prevention. It also supports role-based access control (RBAC) may be created quickly to enable safe access to data and features. Another important benefit of laravel is its scalability .
Laravel's modular design and support for caching methods (like Redis and Memcached) make it very scalable. As the institution expands and its data increases, the program may adjust without severe performance decrease. That's why I used Laravel language to develop my system.

1.3. The solution that emerged:

I have developed prison management system. By using the system, the admin (the authority of prison) can manage and monitor every step of everyone in prison easily. The staff members and guard can easily manage the system. So, the whole prison will be simple and easy to manage and monitor the prison.

1.4. The main perpous of the project:

The major purpose and objectives of my project is to simplify the management system of a jail so that, the management can simply handle the full prison management system of a prison. In every jail in Bangladesh, daily numerous transaction and procedure is occurred which is extremely tough to manage. The supervisor (authority) confronts

a lot of difficulty since, they have not any suitable management system for jail. They experience plenty of challenges to monitor the prisoner and to manage the complete facility. Not only the authorities but also the prisoner and the visitors also suffer a lot of challenges for missing of the management system. The major purpose of my project is to design a jail management system by which the authorities may administer the prison conveniently. The government of a nation, especially the authority of jail will be a lot of advantages.

1.5. A short overview of the remaining chapters:

The system is a web-based system. User can utilize the system effortlessly. By the system a staff will record prisoner, visitors, and police. Police officer have also some involvement on the system. Police can read the remarks, check prisoner details, check jail info, may move prisoner. Like every system there is an admin in my system. Admin may check user details, police officer info and admin info. Admin may check prisoner information, court details and can add jail also. The diagrams (use case, activity, sequence), the

testing (unit testing, integration testing) of the system, the structure, scope of the project, strength and weakness of my project will be discussed further in the documentation below.

Chapter 2 – Initial Study

• Project Proposal:

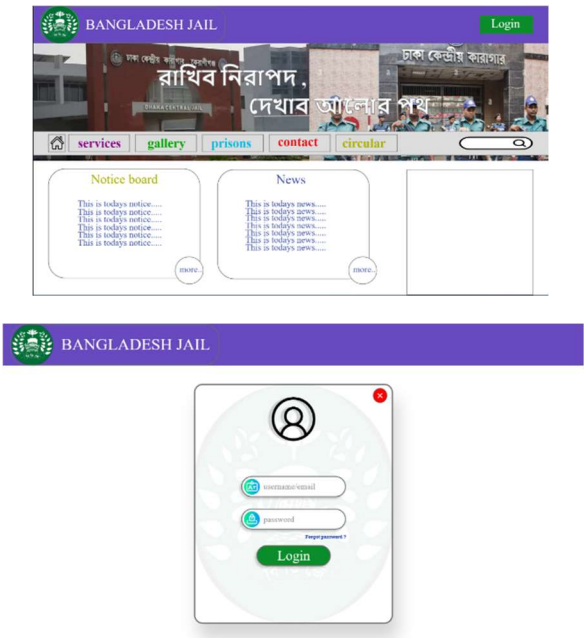
1. Initial Conception:

a) Brief Discussion of the concept:

In our country Prison is one of the busy and important sectors. The authority faces a lot of difficulties because, they have not any proper management system for prison. They feel lot of difficulties to monitor the prisoner and to maintain the whole prison. Not only the authority but also the prisoner and the visitors also face a lot of difficulties for lacking of the management system. The main goal of my project is to create a prison management system by which the authority can manage the prison easily. The government of Bangladesh, specifically the authority of Bangladesh prison will be a lot of benefited. The prisoner and the visitor will also face less difficulties. Every prisoner's details will be easier to track and their every step in the prison will be monitored. By a click, the authority can find any information of any prisoner and any details about the prison. So it will be easier, more sufficient and less time consuming

b) Proof of Concept:

Prototyping: Here provide prototype so that assumption when complete this website how looks like:



2. Initial researc:

a) Market :

Now a days Prison one of the main sectors in Bangladesh. In a jail, a great deal of activity and transactions take place daily. But there is no proper management system for jail management system in Bangladesh. If the jail cannot handle correctly, there would be a great loss for our nation. So there require a proper efficient management system for jail. So, given the current state of affairs, my proposal is very beneficial and feasible.

b) Comparative analysis:

There is no effective and user-friendly jail administration system in Bangladesh. They have a method where they manage the stuffs solely .The authorities employ manual method for maintaining the records of jail and to manage them. But with my system, there is no manual system. By a click of a button they may monitor or control every prisoner, guards and the complete jail also.

3. Feasibility Study

a) Operational Feasibility :

The operational feasibility is the one that will be utilized successfully once it has been created. And the integrated jail controlling and monitoring system is important project to be install and practice the system with the organization. And it is straightforward to run once implemented and we claim this project is operationally viable.

b) Technical Feasibility :

The project should be created so that the essential functionality and performance are realized. The project is created inside newest technologies. Through the technology may become outdated after some period of time, owing to the fact that is simple to apply the system to new technology. The system has been designed using PHP the project is technically viable for development. To operate the system, the system has to be installed or set up. Then printer need to be linked with the system for printing reports and invoicing. Domain and hosting required to purchase to live the system.

c) Economic Feasibility:

The system is allowed to execute numerous actions and it turns the manual system into computerize thus according to the existing system there are so many charges. So, the suggested method is lucrative than the present system, it has low cost to be implemented. Also, there are resources are already accessible, it offers an indicator if the system is economically viable for growth. The jail administration system is entirely web-based technology. So, it is a lucrative project to the user. It will minimize personnel therefore it will be beneficial.

d) Market research analysis based on the feasibility factors :

Prison management systems relate to the method of keeping of records of inmates together with the cases that have been lodged against those convicts. These systems store extensive information about the convicts alongside diverse biometric features, such as thumb imprints, facial expressions, pictures, iris recognition, and others. These days, numerous firms are offering an offer of jail management systems linked with various advance modules, such as medical facility management, case management, workflow control, reporting, scheduling, and demographic data collecting. Growing relevance of these solutions are projected to play vital function in the growth of the worldwide prison management systems market throughout the timeframe of forecast, from 2020 to 2030.

4. Foundations

a) Agree on defined goals/objectives of the project

The major purpose of my project is to establish a jail management system by which the authorities can control the prisons conveniently via one system. Another objective is maintaining the records digitally and securely . The government of Bangladesh, especially the authority of Bangladesh jail will be a lot of advantages. The prisoner and the visitor will also have fewer problems.

b) High-level requirements :

a. Authenticate users.
b. Manage user information.
c. Manage Prison list
d. Show Staff list
e. Manage staff schedule.
f. Transfer staffs
g. Prisoners' category manipulation
h. Manage Prisoner's info.
i. Manage prisoner's tasks
j. Assign prisoner's daily task.
k. Supervise prisoner's daily task
l. Transfer prisoner.
m. Online visitor appointment/request
n. Manage visitor request
o. Different permission for different users
p. Reports for various types of users
q. Manage Jail inventory

c) Prioritize the features/requirements according to MoSCoW

Functional Requirement No	Functional Requirement	Priority
1	User authentication	Must Have
2	Manage user information.	Must Have
3	Prison list	Must Have
4	Staff list	Must Have
5	Manage staffs schedule.	Must Have
6	Transfer staffs	Must Have
7	Prisoners list	Must Have

8	Prisoner's info	Must Have
9	Assign prisoners daily task	Must Have
10	Transfer prisoner	Must Have
11	Different permission for different users	Must Have
12	Manage visitors	Must Have
13	Manage prison store	Must Have
14	Online admission process	Could Have

Chapter 3 -Work Breakdown Structure

a) Gantt chart:

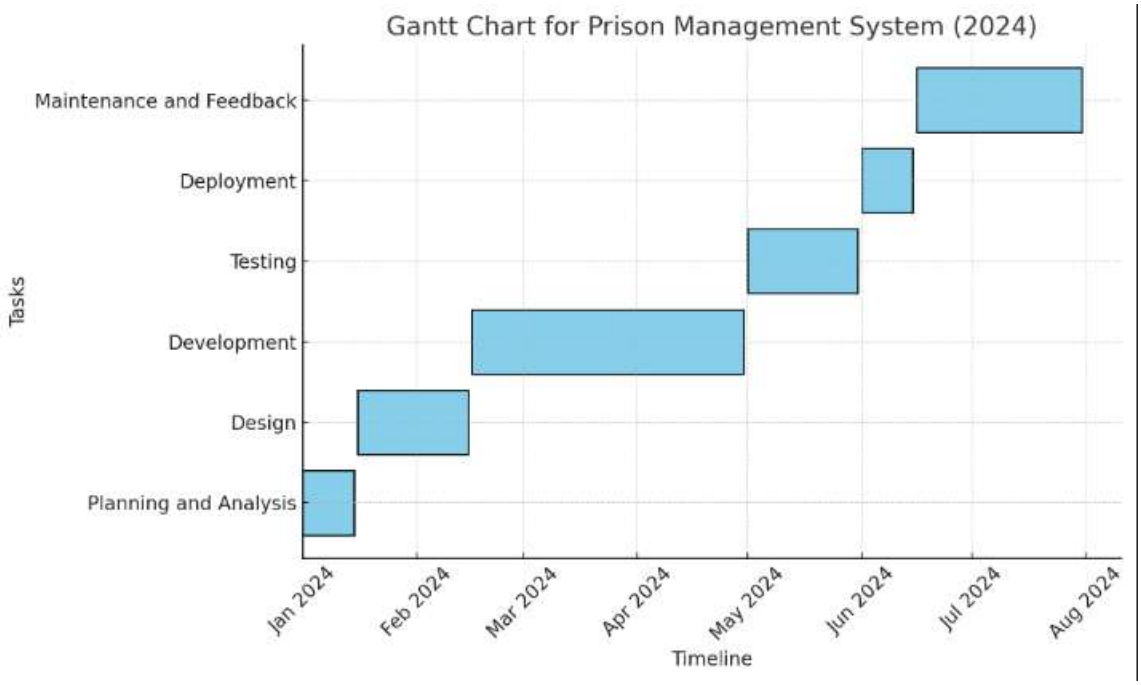


Figure: Gantt chart

b) Identification of key milestones with dates:

Identification of key milestones with dates are given bellow:

Figure: Key milestone

c) Outline of key activities:

Firstly, I have to make DFD (Data Flow Diagram) diagram for full system. By the DFD diagram, I will make Use Case diagram. Use Case diagram will allow me to figure out the user and the purpose of the project. Then I have to develop the ERD (Entity Relationship Diagram). ERD will enable me to develop database. I have to build database via MySQL. Then I have to design my application. I have to create my project using PHP Laravel ,HTML,CSS, and JavaScript. Then I need to tackle the development sector. I need to utilize PHP framework Laravel, MySQL to construct my Project. Then documented it on Microsoft Word.

• Background of the project:

I have designed ‘’Prison Management System” to simplify the management system of the Prison. There is no well-structured management system in our nation. The authorities experiences plenty of problems to monitor the prisoner and to manage the full facility. Not only the authorities but also the prisoner and the visitors also suffer a lot of challenges for missing of the management system. The major purpose of my project is to design a jail management system by which the authorities may administer the prison conveniently.

2. Aims:

The major purpose and objectives of my project is to simplify the management system of a jail so that, the management can simply handle the full prison management system of a prison. In every jail in Bangladesh, daily numerous transaction and procedure is occurred which is extremely tough to manage. The supervisor confronts a lot of challenges since, they have not any suitable management system for jail. They experience plenty of challenges to monitor the prisoner and to manage the complete facility. Not only the authorities but also the prisoner and the visitors also suffer a lot of challenges for missing of the management system. The primary purpose of my project is to establish a jail

management mechanism by which the authorities may control the jail simply. The government of a nation, especially the authority of jail will be a lot of advantages.

3. Problem Area:

There is no well-structured prison management system in our nation. The authorities experiences plenty of problems to monitor the prisoner and to manage the full institution. In some places they still use traditional registry book to maintain data and other information . They have a government portal but they use that limitedly. Due to the absence of a management framework, not only do the authorities have several challenges, but so do the inmates and visitors.

4. Possible solution:

I have built jail Prison managemnnt system. By utilizing the technology, the admin (the authority of jail) may oversee and monitor every move of everyone in prison effortlessly. The cops and prison guards can simply handle the system because of the user friendliness. So, the complete jail will be simple and straightforward to handle and monitor the prison.

Chapter 4 – Literature Review

Literature review is one of the most critical aspect to produce progress of the system. A literature study aids with learning in depth relating this certain zone. During the development phase, literature aids with identifying the basically equivalent activity within the comparable zone. Its assistance with deciding which highlights need to be in the proposed method.

1. Discussion on problem domain based on published articles.

People encounters many difficulty when preparing an event. It is a massive undertaking for a single individual to prepare an event. Without a doubt, some of the time everyday troubles may arrive approaching us. Unexpected incidents, seller connected concerns and tensions are might cause big challenge. Below are some problems that users may encounter:

- Not enough money for the occation
- No regard for the things that can't be measured
- Choosing a site
- Weather that isn't flexible

2. Discussion on problem solutions based on published articles.

The suggested framework entitled "Prison Management System" will built for the government people of Bangladesh. The objective of the project is to abridge the management system of a jail. In Bangladesh, it is exceedingly tough to control the jail .

system reason of big transaction and procedure. The authority confronts various issues with oversight. It is considerably challenging monitor and maintain the full jail. Visitor also suffers various troubles with the system. Within this system, not only the authorities the government will also be highly benefitted.
In this system, basically Admin and Direnctor is the responsible person who may alter and arrange the full system.
Here, the system will be based on web. Therefore, that person may effortlessly procedure to the system. Thru this system, staff may register prisoner, visitor and keep track of the inventory system of prison goods . There are also some part for the police officer. Police may verify information about the prisoner.
Also, can view comments. Police may also move a prisoner using the system. Like other system, my system also includes an admin component. Here admin may examine user, admin and police information. An admin may access a prisoner and court data. In addition, admin may manipulate necessary info such ass adding new cells of prison building into the database or update status of other

users accounts if needed . The director or jail super can assign task to both staffs and prisoners daily task . He can also authenticate visitor request applied by visotors . The guards or staff can report prisoners daily task complision and manage approved visitor and jail inventory . Visitors can apply for visit through web and they will get a token of a QR code which can be used as gate pass or filled with necessary information . Guards can scan the QR code to assist the visitors . The jail super can also monitor the given task by him to the staffs or prisoners .

3. Comparison of three/four leading solutions

After a big study, I did not discover any system like this yet. Therefore, it is rather tough for me to draw a comparison. However, my system has several unique characteristics.
Underneath I will describe the characteristics.

- **Best features**

These are the best aspects of the systems.

- ❖ Provide a welcoming management atmosphere
- ❖ Organize the jail in a methodical manner
- ❖ create a paperless environment

- **Limitations**

- ❖ There is no transaction in the system.
- ❖ The system is non-responsive, so it is quite impossible to use without computer.

4. Recommended approach

Essential features:

We have to define some essential features .These are the essential features of my project which are given bellow

Admin :

- Login
- Home
- Profile
- Dashboard
- Search
- Authenticate user
- Add Jail super/Director
- Jail super/Director list
- Guard / staff list
- Add guard/staff
- Prisoner registration
- Prisoner transfer
- Staff register
- Staff transfer
- Add new cell/prison
- Manage visitor applicatoipn
- Court Info
- Create Task for prisoners
- Assign staff duty
- Task report
- Manipulate Inventory
- Logout

Director/Jail Super:

- Login
- Manage profile
- Search
- Ward list
- Create new ward
- Cell list
- Create new cell

- Assign staff duty
- Assign prisoner duty
- Assign prisoner to cell
- Manipulate visitor application
- Task list
- Task report
- Monitor task report
- Check inventory status

Guard:

- Login
- Manage profile
- Search
- Check visitor list
- Check visitor application approval
- Print Visitor pass
- Check report list
- Create report
- Evaluate prisoners task
- Check prisoners task report
- Submit inventory report

Visitor:

- Apply for visit
- Check approval through email

Chapter 5 – Methodology

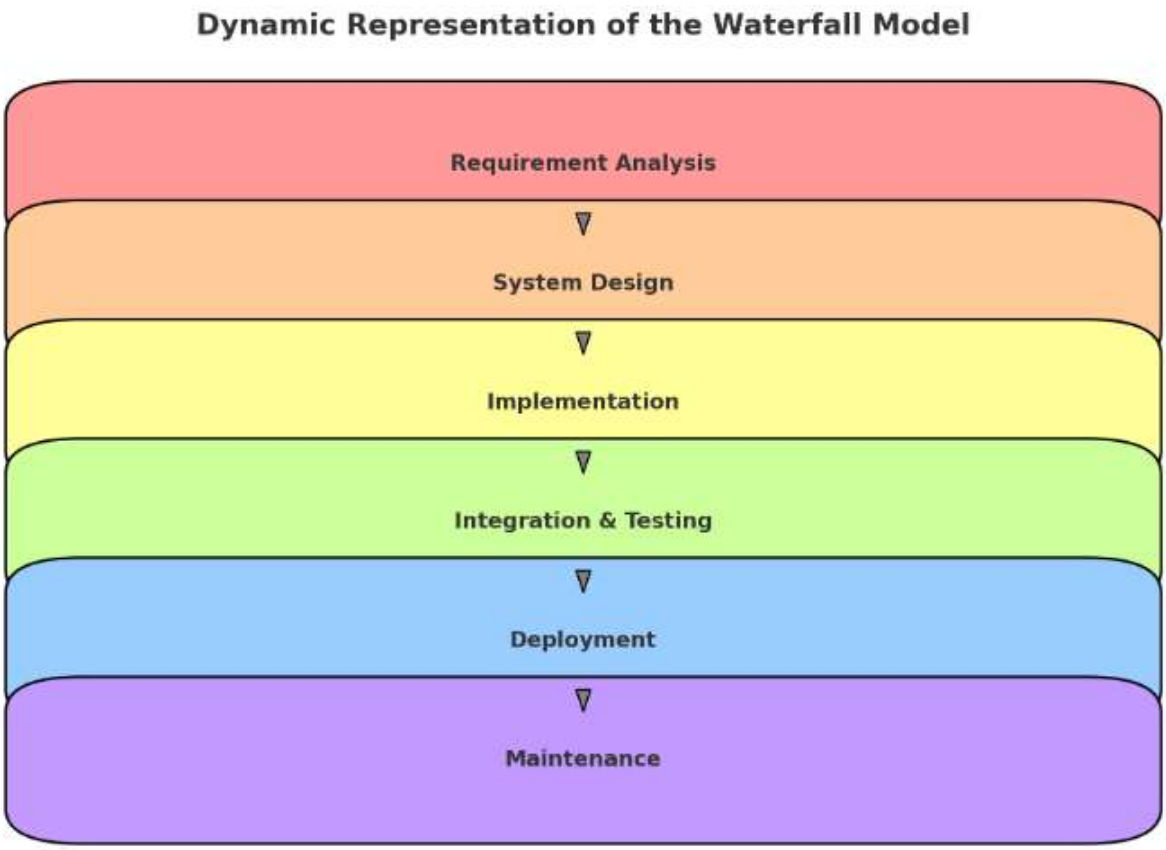
Software development approaches are fundamental to the software development process. Many software development organizations use a variety of approaches to create certified software. When developing software, engineers encounter several challenges. The goal of using the approaches is to create software that is typically successful and of high quality. The system has been very helpful to designers in developing the data structure. Every method has advantages and disadvantages.It does not include any viewpoint. It integrates actual software development framework.

- ## What to use

There are several software development approaches these days. I'm describing them right now.

5.1. Waterfall model:

The Waterfall Model was the main Procedure Model to be presented. It is likewise referred to as a straight sequential life cycle model. It is really basic to grasp and operate. In a waterfall model, each step must be accomplished before the subsequent stage can start and there is no covering in the phases. The Waterfall model is the most timely SDLC technique that was applied for software development. The waterfall Model displays the software development process in a straight sequential stream. This signifies any step in the development method begins merely if the prior stage is done. In this waterfall model, the phases don't cover. (Anon., 2020)



5.1.1. Pros of Waterfall Model:

- IT permits departmentalization and management control.
- It is straightforward and quick to use.
- Because of the model's rigidity each phase has distinct deliverables and a review procedure it is simple to maintain.
- Each phase is processed and finished separately.
- It performs effectively on smaller projects with well defined needs.
- A product may go through the development process like a vehicle in a car wash and, in theory, be delivered on time if a timetable with deadlines for each step of development is established. (Anon., undated)

5.1.2. Cons of Waterfall Model:

- There is little opportunity for introspection or modification.
- It is quite difficult to go back and make changes to an application that was poorly considered at the idea stage after it has entered the testing phase.
- It takes till the end of the life cycle to generate functional software
- A great deal of uncertainty and danger.
- This paradigm isn't suitable for intricate, object-oriented programs.
- A subpar paradigm for lengthy, continuous tasks.
- Unsuitable for projects with needs that are subject to moderate to significant change. (Anon., n.d.)

5.2. Agile Methodology:

Agile methodology is a kind of venture the board process, fundamentally used for software innovation, where requests and arrangements advance via the collaborative exertion of self-sorting out and cross-utilitarian groups and their customers. Coming from the traits and principles of the Agile Manifesto, it was produced as a response to the insufficiencies of traditional advancement procedures, for example, the

Waterfall approach. The software industry is an extremely significant sector because of the manner that software is something that might be continually upgraded. This means engineers need to consistently upgrade and develop their goods to remain on the game and the straight, consecutive methodology of the Waterfall approach just wasn't cutting it.
(Anon., 2020)

5.2.1. Pros of Agile Methodology:

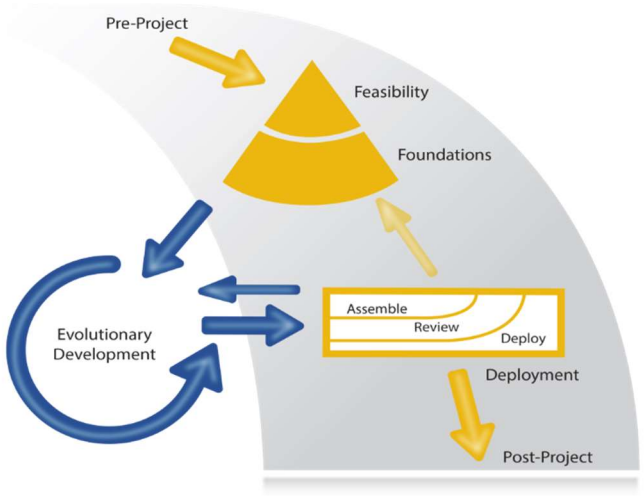
- The timely and consistent provision of beneficial software results in satisfied customers.
- Interactions and people are prioritized above procedures and equipment. Testers, developers, and customers communicate with one another all the time.
- Functional software is regularly released (in weeks as opposed to months).
- The most effective way to communicate is in person.
- Close, everyday collaboration between developers and businesspeople.
- Constant focus on quality design and technical proficiency.
- Consistent adjustment to evolving conditions.
- We embrace even last-minute required modifications. (2020, Anon.)

5.2.2. Cons of Agile Methodology:

- Some software deliverables, particularly the larger ones, make it difficult to estimate the amount of work needed at the start of the software development life cycle.
- Essential design and documentation are not given enough attention.
- If the client representative is unclear about the desired objective, the project may quickly veer off course.
- Senior programmers are the only ones who can make the kinds of choices needed throughout the development process. Therefore, it cannot accommodate novice programmers unless paired with more seasoned resources. (2020, Anon.)

5.3. Dynamic Systems Development Method (DSDM):

Building and maintaining systems is made more structured by the Dynamic Systems Development method (DSDM), a partner degree dexterous code development strategy. A modified version of the humanism guideline serves as the basis for the DSDM rationale. Usually, 80% of an application is communicated in 20% of the time that the whole application (100%) would be desired. The 80% principle, which states that each augmentation basically requires enough labor to stimulate development to the following addition, is maintained in each cycle of the iterative DSDM coding approach. Usually, the remaining details are completed later once several business requirements have been identified or modifications are referred to and appropriate. (2020, Anon.)



[Source: www.agilebusiness.org]

5.3.1. Pros of DSDM:

- Offers a method that is independent of technique.
- Adherent to strict schedule and budget constraints; flexible in terms of need evolution
- Stakeholders are included in the development process.
- Testing is so important that every project team is required to include at least one tester.
- Business value is recognized and anticipated to be the top priority delivery since it was designed from the bottom up by businesspeople.
- Uses a particular methodology to assess each requirement's significance for iteration.
- From the outset, it informs stakeholders that not all criteria will be included in the final product. (2020, Anon.)

5.3.2. Cons of DSDM:

- Involves progressive requirement development
- Focus on RAD may result in less robust code
- Demands complete adherence to the DSDM process
- Needs substantial user involvement
- Needs a proficient development team in both the technical and business domains
- Is likely the largest project compared in this survey. A consortium controls access to the material, and costs may be assessed just to view the reference material.
- It also expects ongoing user engagement and specifies several artifacts and work outputs for every project phase
- it also contains more documentation. (2020, Anon.)

- **Why to use**

I started my endeavor for academic purposes.No specific structure is appropriate for my work since it lacks a commercial rationale. I decided to expand my business using DSDM and Water Fall.

The reason of choosing these two methodologies:

- Following the need analysis, an investigation will follow each operation. Highlights and usefulness may be identified by looking. Here, the waterfall model will perform very well.
- The extra time won't be counted for this activity with the help of the waterfall model. Here, the waterfall model acquires a key role.
- The consumers' interest will be addressed throughout the preliminary inquiry.
- It's really difficult to include all of the clients' needs, which the SDLC supports.
- The MoSCoW priority will help place the highlights in accordance with the DSDM Atern.
- Their tasks will be divided into little parts in accordance with the DSDM Atern.

- At that point, the working time will determine which time boxes the little parts will be placed in.
- The iterative development of the DSDM will occur throughout the development process as the inadequate portion may be modified.
- Following the completion of each component, a model will be constructed. This will help with work that is affirming.
- Every time, quality is tested while working.
- In order to complete the project, the DSDM atern combines all of the labor.

• Sections of methodology

I have selected DSDM methodology. It has following phases:

i. Initial Research:

The main stage of technique is the introductory examination. In any event, this system phase is crucial for concentration since the basic space cannot function properly.

ii. Requirement Elicitation:

The customer has provided a number of requirements in order to fully understand the project. Elicitation takes more time than normal data collection.

iii. Analyzing requirements and setting priorities:

There are more than a hundred criteria for any kind of large programming system. In any event, it is often impossible to complete all of the requirements. This step helps to organize the requirements from the previous step. Setting priorities helps with recognizing which requirements are often necessary to complete the activity. The MosCow priority will set following all of the needs. Last but not least, time boxing helps with planning.

iv. Exploration:

Exploration helps identify the essentials and location in a potent arrangement. It is challenging to specify which arrangements will be cautious and which demands are arranged. Precise yield is this stage's guiding concept. Following this stage, the item will be delivered to design in advance to ensure that there are no specialized problems.

v. Engineering:

This phase is used to advance the underlying solution in order to complete the operational foundation for widespread and expanding use.Non-practical needs are established here in definite creation at this level.

vi. Deployment:

The customer will use this item as their last use in this location. The things are mostly here for transportation. This item will be used by the end user. If end users find any problems or defects with the product, it will be submitted to experts, and if it is found to be in good condition, the customer will use it.

vii. Post project:

The last step of the framework arrangement's construction will take place after the venture, at which point the customers will be able to justify the framework's benefits.

• Implementation plans:

Choice of programming language:

In this project i take use of PHP framework Laravel , it's an online artificial language simultaneously is a scripting language that is interpreted at runtime. I'm mistreatment PHP in my project in my project as a consequence of LARAVEL operates in many platform like UNIX, Unix etc. and it's simple and run with efficiency on the server. The main cause of maltreatment LARAVEL of it dead while not aggregation. Machine-readable text Pre-processor (LARAVEL) code will inserted into hypertext mark-up language code and it may utilize in mix with diverse web page management system. In my project hypertext mark-up language is additional extra benefit for misusing LARAVEL and that i am exhibiting the result or output to utilize HTML.
I'm modifying hypertext mark-up language tags by misusing LARAVEL. I control the details of my project by using LARAVEL.
It creates an easy technique for functioning hand to hand using data server e.g. MySQL. I design my project dynamic as I will. I take information via abusing hypertext mark-up language and insert this data into the info through the LARAVEL. Collect data from web site and transmit it via the server with evaluating the correctness of the info abusing LARAVEL. i take use of LARAVEL for causation mail to the user. I write the user information and simple will realize the todays date and time by misuse LARAVEL.
During this project I'm developing a separate section of the web site for its member. I construct the login and registration with validation by misusing LARAVEL. User will add any member, create, write, remove, edit sections at intervals the info using the LARAVEL. User will ban or prevent people to view particular sites using this strategy.

During this project LARAVEL is handles all forms and its collect information and store it to a file and it will airt to user on various pages.

- **System cutover from the development architecture to the implementation architecture:**

1) Development of Architecture:

Resources:

I have done study on the resources I will need to conduct the project before I begin. To store the data for the jail administration system, I need a SQL server. In order to operate SQL Server, I require XAMPP. After that, I need software that allowed me to use the PHP programming language. I have to install php latest version and composer latest version alongside Laravel . For coding ive used VS Code .

Activities:

I initially made a Gantt chart to outline the time timeline in order to begin the job. I want to investigate the project's necessity. Both nonfunctional and practical demands have existed. The functionality and features of my project have since been finished. After that, I looked at the project's risk management and found the ways to prevent it. Following that, I really created the diagrams, including the use case, category, activity, and sequence diagrams, among others. After that, I really began the project's event. to the project's progress. There were a lot of drawbacks while I was working on the project. But I was able to correctly answer them with the help of Google, my devoted pal, and my esteemed instructor.

Output:

After developing the system, I had the Prison Management System's outputs, which I had to verify to see whether it was working properly. As a result, I have project results after development.

Outcomes:

I created the timeline for completing the jail administration system. To get the results, I have to complete every step. I have the Prison Management System as the result of finishing all the processes.

- **Data migration from the development architecture and/or existing systems to the implementation architecture:**

Training:

It must understand how to utilize the system correctly in order to get the actual outputs of the system. Therefore, in order to operate the jail management system, the management must have the necessary training. since the majority of individuals are not used with using digital systems as opposed to manual ones.
The administrator must get training on how to log in to the system. Admin is a crucial position. Therefore, no one is able to log on to the system. A certain system exists. The administrator must be aware

that. The administrator must next learn how to manage and modify information, as well as how to transfer jail officers, check prison details, check case details, check transfer information, check offer details, check new prison details, check court information, and check reports. Therefore, the administrator must have enough training in order to understand how to operate the jail administration system.
The policeman has the ability to log in to the system as well. Therefore, police officers must be trained to operate the system. A police officer can look up information on courts, transfers, and prisons.
They are also able to search for information. Information on visitors, inmates, officers, etc., may also be checked. Therefore, the officer must have flawless training to do their duties responsibly in order to operate the system effectively. For police personnel to effectively utilize the jail administration system, thorough training is also essential.
Additionally, a staff actor will play a police officer. They are able to utilize the system as well. Both the visitor's and the police officer's data must be registered. They must thus get flawless training. If not, they will enter incorrect data into the system. Therefore, personnel and users must be trained for effective work.

Chapter 6 – Planning

• Project Plan

This section will cover the whole progression process. Although this project is a big undertaking, the advancement method will be straightforward if the complete improvement process is divided into smaller parts. These little pieces of progress will be arranged according to certain models in order to construct the whole framework.

❖ Management Plan / Work Breakdown Structure (WBS)

The project team works hard to complete the whole project on time. One method for breaking down a structure is to use a "deliverable situated various leveled disintegration of the work to be executed by the project group." It helps identify the likelihood and extent of transitioning into more manageable chunks that will be easier for a project team to handle. In the work breakdown structure, each level of work will be described in detail. My work breakdown structure approach has been put to good use in areas where the project will function more efficiently. The job breakdown structure will operate better with the administration framework. The following is the job breakdown structure:

No.	Task Name	Start Date	End Date	Duration (Day)
1	Introduction	01/02/2024	01/02/2024	1
2	Initial study	02/02/2024	04/02/2024	3
3	Literature Review	05/02/2024	08/02/2024	4
4	Methodology	09/02/2024	14/02/2024	6
5	Planning	15/02/2024	17/02/2024	3
6	Feasibility Study	18/02/2024	20/02/2024	3
7	Foundation	21/02/2024	23/02/2024	3
8	Exploration	24/02/2024	28/02/2024	5

9	Engineering phase	29/02/2024	02/03/2024	5
10	Deployment phase	03/03/2024	03/04/2024	30
11	Testing	04/04/2024	13/04/2024	10
12	Implementation	14/04/2024	20/04/2024	7
13	Critical appraisal	21/04/2024	30/04/2024	10

❖ Time Duration / Time Boxing

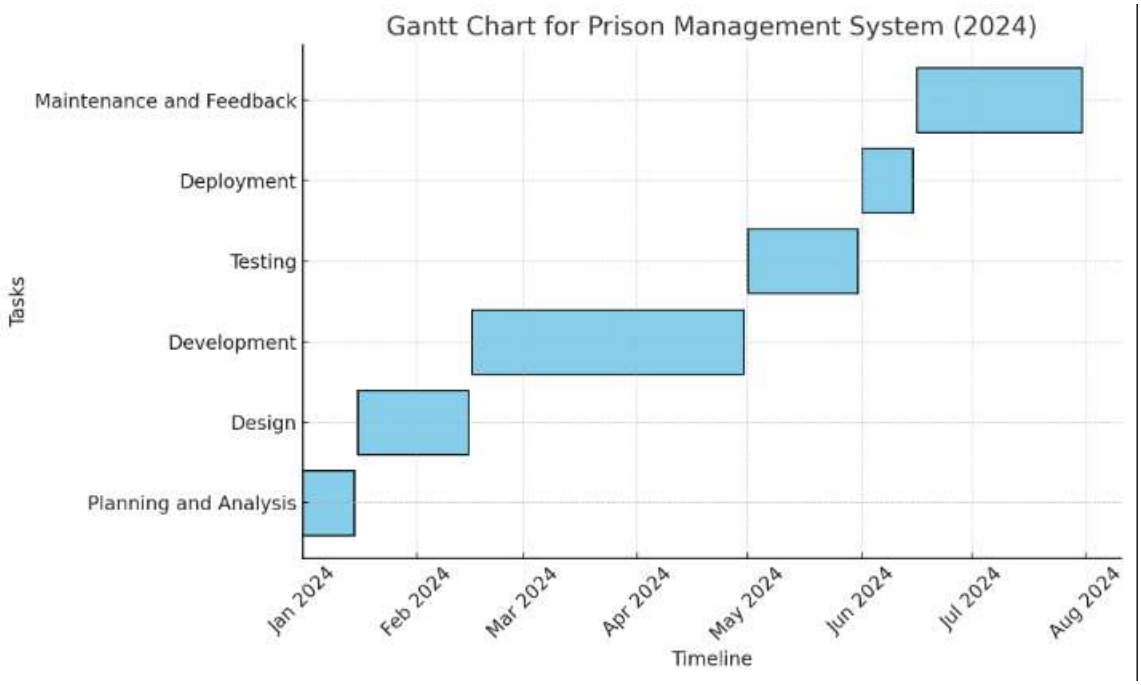
When errands are divided into time-constrained tasks, time allocation helps ensure that the project is completed on schedule.Setting aside a suitable amount of time to complete a job is crucial.

Activities		W 1	W 2	W 3	W 4	W 5	W 6	W 7	W 8	W 9	W 10	W 11	W 12	W 13	W 14	W 15	W 16	W 17
Planning	Title																	
	Proposal Title																	
	Proposal planning																	
Requirements	Requirement Finalizing																	
	Requirement analysis																	
System Prototype	Start Prototyping																	
System design	Design Specification																	
	Interface design																	
	Database design																	
Development	Development System modules																	
	Integrate System Modules																	
Quality Testing	Test cases																	
Testing	Unit testing																	
	Black-box testing																	

Resolve Issues	Resolve issues found																

❖ Gantt Chart

My project's Gantt chart is shown below:



• Test Plan

There are several functions in a project.It will be tested for accuracy and approval.Knowing if the framework is functioning or not is crucial. By identifying the level of risk, the test plan ensures that the frameworks are accurate.

❖ Testing against time boxes

There is a delivery for every box. With the intention of requiring testing after each time box. It will provide a clear understanding of the outcome. For this reason, testing is necessary prior to conveyance. The following is the experiment against the time boxes:

i. Unit testing:

Unit testing is a kind of programming testing in which individual product units or segments are tested. Existence is necessary to ensure that each product unit operates according to plan. The smallest testable component of any product is called a unit. It typically has a single yield and one or a few information sources. A unit in procedural programming might be a single program, piece of labor, method, etc. The smallest unit in object-situated programming is a technique, which might belong to an inferred/youngster class, conceptual class, or base/super class. Some people see an application's modules as a single entity. This is discouraging since the module is likely to have several separate parts. To facilitate unit testing, stubs, drivers, counterfeit/counterfeit goods, and unit testing frameworks are used. (2020, Anon.)

ii. Integration testing:

Integration testing is defined as a kind of testing in which programming components are arranged logically and tested collectively. A typical programming project consists of many product modules that have been created by different developers. Finding surrenders in the relationship between these product components upon integration is the driving force for this level of testing (Anon., 2020).

iii. Module Testing:

Module testing is essentially focused on evaluating individual programming modules or subprograms rather than testing the whole programming application at once. Because it is much easier to identify, understand, and correct flaws at the module level rather at the application level, module testing is useful and often recommended in programming design. Up till now, the main section taught us the definition of module testing. (2020, Anon.)

iv. Acceptance testing:

Acceptance testing is a technique used to determine if the product structure satisfies the requirements. The main goal of this test is to determine if the framework complies with business requirements and whether it has fulfilled the models required for delivery to end users. (2020, Anon.)

v. Performance Testing:

Performance testing evaluates a product program's speed, response time, dependability, asset utilization, and adaptability when it is doing its typical remaining tasks. The goal of performance testing is to eliminate performance bottlenecks in the device or product, not to find real-world defects.

vi. Security Testing:

A kind of software testing known as security testing identifies weaknesses, risks, and opportunities in a product application and prevents malicious attacks by gatecrashers. The purpose of security tests is to identify any potential escape route and flaws in the product architecture that might result in lost information, revenue, or reputation due to the organization's representatives or untouchables. (2020, Anon.)

vii. Test Case:

A test case is a collection of actions carried out to evaluate a certain aspect or functionality of your product application. Test steps, test information, preconditions, and postconditions created for an explicit test scenario to verify any need are all included in a test case. A testing architect may use the specific elements or conditions included in the test case to compare anticipated and actual results and determine whether a product item is meeting the client's requirements.

viii. User acceptance test plan:

Before transferring the product application to the creation state, the end user or customer does a kind of testing known as User Acceptance Testing (UAT) to validate or recognize the product framework. Following the completion of practical, joining, and framework testing, UAT is conducted during the final testing phase.
UAT is mostly used to approve the business stream from beginning to end. It doesn't focus on system testing, spelling mistakes, or cosmetic mistakes. A distinct testing environment with creation-like information layout is used for user acceptance testing. At least two end users will participate in this kind of discovery testing included. User Acceptance Testing is UAT in its fullest form. (2020, Anon.)

Chapter 7 – Feasibility

Feasibility studies are conducted when the framework should be operationally and practically achievable and validated. It helps determine if a project is viable under new circumstances. Different types of feasibility studies, such as financial, operational, and specialized feasibility, will be shown separately in this section.

• All possible type of feasibility

i. Economic Feasibility:

According to the existing system, there are a lot of expenses since the system allows for the completion of several tasks and transforms the manual system into a computerized one. Therefore, the suggested system has a lower implementation cost and is more lucrative than the present one. Additionally, the fact that resources are currently accessible indicates that the system may be developed inexpensively. (2019, Anon.) The whole jail administration system is web-based. Therefore, the project is beneficial for the user. It will gain from the reduction of staff.

ii. Technical Feasibility:

The project should be designed to accomplish the required performance and functionality. The project is being created using the newest technologies available. Although the technology may eventually become outdated, since it is feasible to apply the new technology to the system. Laravel was used to construct the system, making the project technically possible. (2019, Anon.)
The system must be installed or configured in order to function. The printer must then be linked to the system in order to produce invoices and reports. To operate the system, a domain and hosting must be purchased.

iii. Operational Feasibility:

The one that will be utilized successfully after it has been produced is the operational feasibility. Additionally, it is beneficial to develop and practice the integrated jail management and monitoring system with the organization. We conclude that this idea is operationally viable since it is simple to run if put into practice. (2019, Anon.)

• Cost Benefit Analysis

This statistics assessment clearly shows how amazing the present company is performing. Every day, a new piece of work is completed online. Every task is requested online. This framework will be profitable for the user. Cost and benefit estimates for the suggested framework are provided below:

Items	Cost (tk)
Hardware	17000
Software	27500
Server	13000
Development	240000
Management	37000
Others	18000
Total	**398300**

• DSDM – good or not for this project

For this project, I believe DSDM is helpful. Because the DSDM structure approach makes it possible to maintain the project cycle in a viable manner. Following the concentration of each point from the standpoint of the project's viability. It is generally accepted that the DSDM approach is worthwhile and that consumers would benefit from it from a variety of perspectives. Some of the reasons why this is appropriate are listed below:

- ➢ Offers a method that is independent of technique.
- ➢ Adherent to strict schedule and budget constraints
- ➢ Flexible in terms of need evolution
- ➢ Stakeholders are included in the development process.
- ➢ Testing is so important that every project team is required to include at least one tester.
- ➢ Business value is recognized and anticipated to be the top priority delivery since it was designed from the bottom up by businesspeople.
- ➢ Uses a particular methodology to assess each requirement's significance for iteration.
- ➢ From the outset, it informs stakeholders that not all criteria will be included in the final product. (2020, Anon.)

Chapter 8 – Foundation

Overall Requirement List

➢ List of functional requirements:

A project's functional requirements are the kind that must be taken into account throughout the whole undertaking. The developer-understood operations are described as a means of automating the current system. services the system need to provide. This functional requirement outlines the actual functioning of our system or what it does. The following features are offered by the system. It allows inmates' information to be added to the system. Inmates and other recorded data are viewed by the system in accordance with his or her privilege. The technology makes it possible to update data on inmates. Register the family members of the inmate and have the inmate called at a certain time. Create the report the stakeholder requests, including the name, date, and time.

- ➢ Alert audible notifications to inmates, law enforcement, and others when inmates have court appearances.
- ➢ The Prison Management System must have to do authentication to login the system.
- ➢ The Prison Management System should have a transaction system for whole transaction of money in prison.
- ➢ The Prison Management System could have a visitor login platform to show the status of prisoner of their relatives.
- ➢ The Prison Management System won't have the installation process because this project is a web based project

➢ List of system-wide non-functional requirements:

Security:

Our system must be highly secure in order to protect prisoner data and other sensitive information. To achieve this, we employ a number of security measures, such as cryptographic technology and SSH. Additionally, our system has a username and password for all users to interact with, which helps to prevent unauthorized user access. Thus the system is safer.

Performance:

Electric power issues and connection loss may have an impact on its functioning. Our system requires additional equipment to meet this criteria in order to create an interruption in such a scenario.

Accuracy:

The system is prone to inaccuracy since it is operated by humans. Advanced technology and a skilled programmer are required to increase its accuracy.Although we now lack the resources to make the system 100% correct, we want to increase its accuracy over time by gaining experience and information from many sources and using new technological tools that will enable us to accomplish our goals.

Reliability:

When we enter an uncertain input sequence, the system may provide unexpected results. Reliability, on the other hand, means that the system should function around the clock, every day of the week. Therefore, our system is dependable when seen from this angle as much as feasible.

Graphic user interface (GUI):

The system has an extremely user-friendly interface design that has been improved by recently released tools to make using the system more enjoyable for users. It also supports local languages to facilitate simple communication inside the system.
The project is web-based and portable. It must thus be portable.

➢ What Technology to be implemented (Client/Web/Standalone):

Several types of innovations are employed to operate the framework efficiently in this cutting-edge era. These developments are used for a wide range of purposes.

Client/server applications:

Among the innovations are web applications and client/server applications. to maintain a record of routine interactions inside a certain organization. Server or client application is used. It is very beneficial to any firm. Clients have been selected by this application. Additionally, anyone may use this program on many PCs at the same time. This program operates quickly and has few security risks.

Web-based application:

Web applications use the framework via web programs and are dependent on PCs. Additionally, this program is prepared to execute at the same time on many computers.
• No further programming is needed for this application to make use of the framework.
• It can operate quickly and has no customer restrictions.

• This program uses a URL drive.

Recommendations and Arguments:

The following are some of the reasons why the provided venture uses web applications:

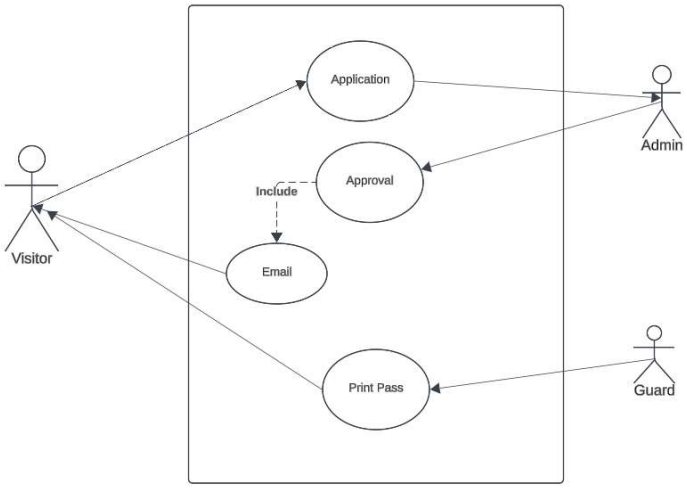
• Since the internet is now widely used, it is envisaged that web programs would make use of this architecture.

• Since the framework has no client constraints, it may be used by innumerable people and is not anticipated to be used by any other programs.

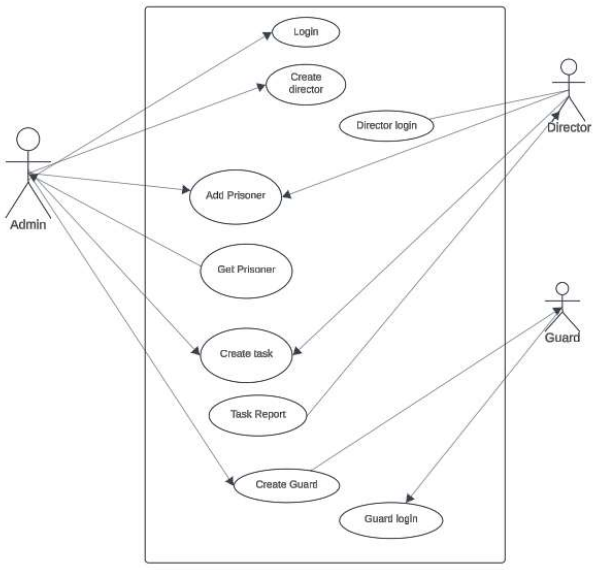
• It is quite easy to use and unrestricted. Operating this framework won't provide any challenges.

Chapter 9 – Engineering

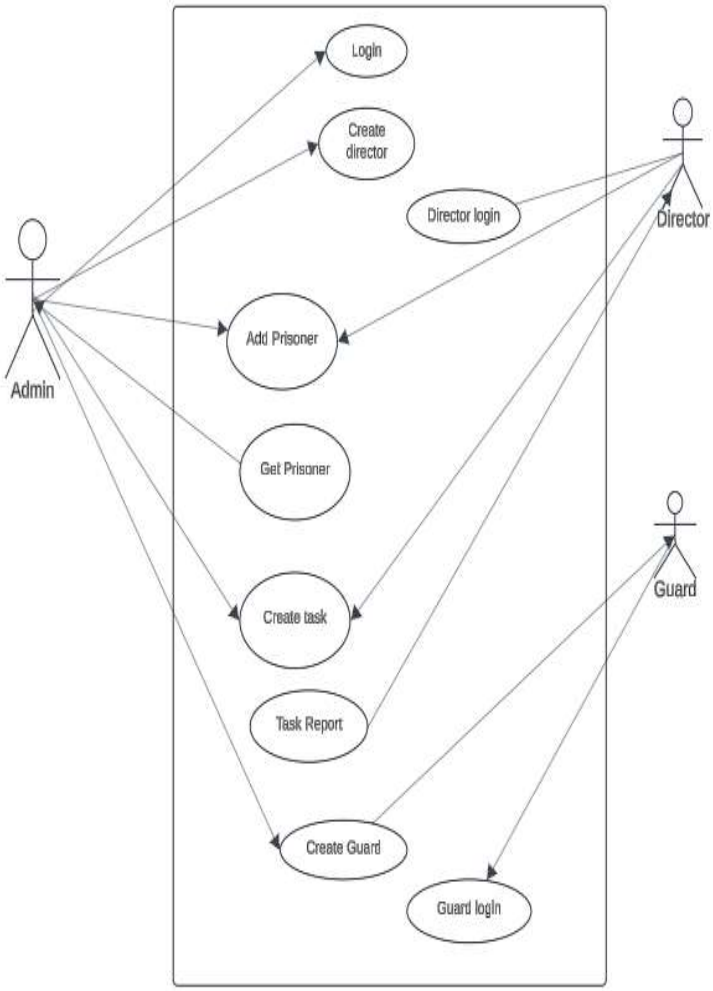
- **Use Case**



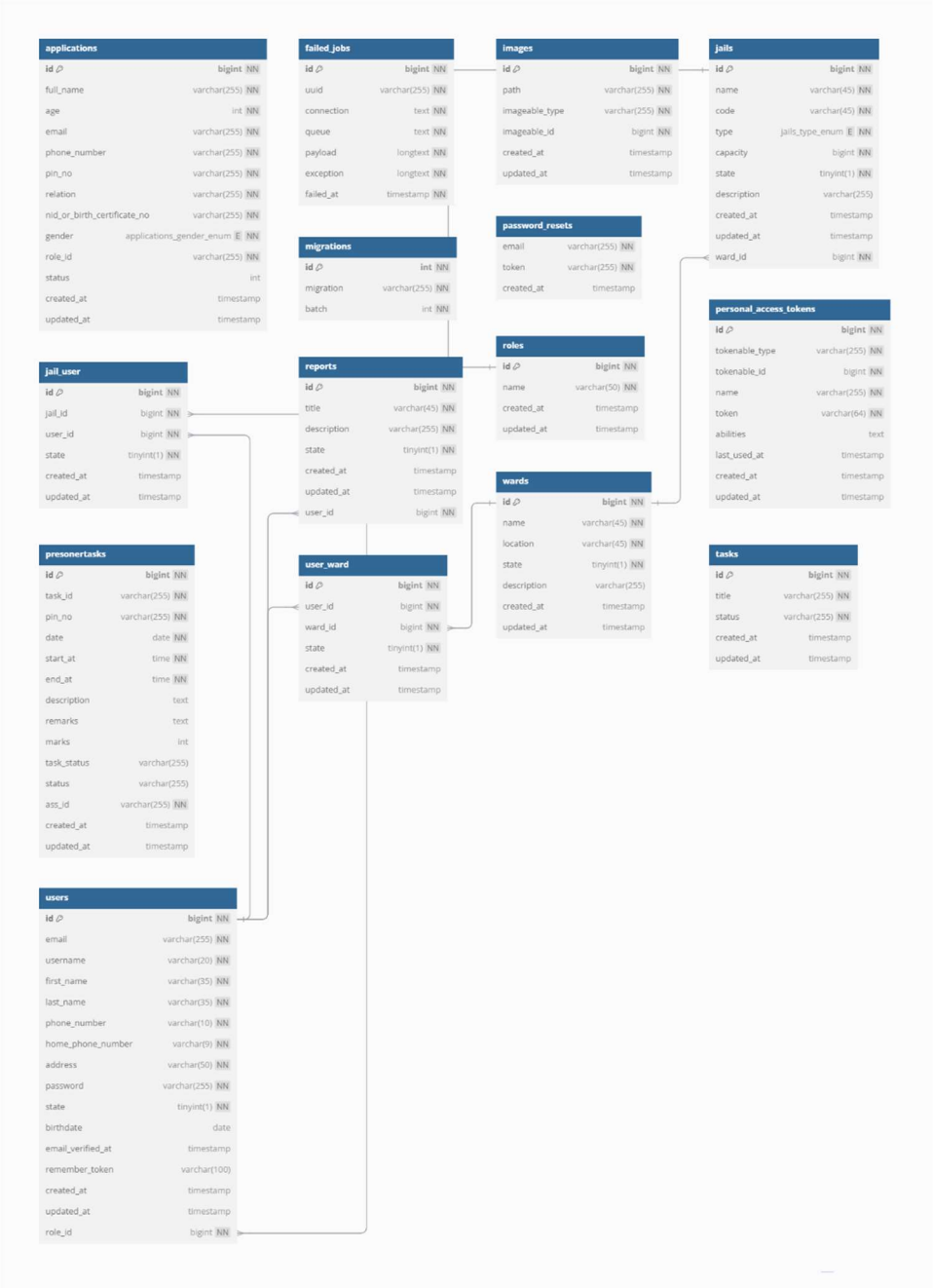
Visitor Application



Admin pannel



- **ERD Diagram**


applications
id bigint NN
full_name varchar(255) NN
age int NN
email varchar(255) NN
phone_number varchar(255) NN
pin_no varchar(255) NN
relation varchar(255) NN
nid_or_birth_certificate_no varchar(255) NN
gender applications_gender_enum E NN
role_id varchar(255) NN
status int
created_at timestamp
updated_at timestamp
failed_jobs
id bigint NN
uuid varchar(255) NN
connection text NN
queue text NN
payload longtext NN
exception longtext NN
failed_at timestamp NN
images
id bigint NN
path varchar(255) NN
imageable_type varchar(255) NN
imageable_id bigint NN
created_at timestamp
updated_at timestamp
jails
id bigint NN
name varchar(45) NN
code varchar(45) NN
type jails_type_enum E NN
capacity bigint NN
state tinyint(1) NN
description varchar(255)
created_at timestamp
updated_at timestamp
ward_id bigint NN
migrations
id int NN
migration varchar(255) NN
batch int NN
password_resets
email varchar(255) NN
token varchar(255) NN
created_at timestamp
personal_access_tokens
id bigint NN
tokenable_type varchar(255) NN
tokenable_id bigint NN
name varchar(255) NN
token varchar(64) NN
abilities text
last_used_at timestamp
created_at timestamp
updated_at timestamp
jail_user
id bigint NN
jail_id bigint NN
user_id bigint NN
state tinyint(1) NN
created_at timestamp
updated_at timestamp
reports
id bigint NN
title varchar(45) NN
description varchar(255) NN
state tinyint(1) NN
created_at timestamp
updated_at timestamp
user_id bigint NN
roles
id bigint NN
name varchar(50) NN
created_at timestamp
updated_at timestamp
wards
id bigint NN
name varchar(45) NN
location varchar(45) NN
state tinyint(1) NN
description varchar(255)
created_at timestamp
updated_at timestamp
presonertasks
id bigint NN
task_id varchar(255) NN
pin_no varchar(255) NN
date date NN
start_at time NN
end_at time NN
description text
remarks text
marks int
task_status varchar(255)
status varchar(255)
ass_id varchar(255) NN
created_at timestamp
updated_at timestamp
user_ward
id bigint NN
user_id bigint NN
ward_id bigint NN
state tinyint(1) NN
created_at timestamp
updated_at timestamp
tasks
id bigint NN
title varchar(255) NN
status varchar(255) NN
created_at timestamp
updated_at timestamp
users
id bigint NN
email varchar(255) NN
username varchar(20) NN
first_name varchar(35) NN
last_name varchar(35) NN
phone_number varchar(10) NN
home_phone_number varchar(9) NN
address varchar(50) NN
password varchar(255) NN
state tinyint(1) NN
birthdate date
email_verified_at timestamp
remember_token varchar(100)
created_at timestamp
updated_at timestamp
role_id bigint NN

Chapter 10 – Deployment / Development

• Core Module Coding Samples

Add Director:

```blade
<x-slot name="title">{{ __("Create a new director") }}</x-slot>

<x-slot name="description">
    {{ __("You can register a new director.") }}
</x-slot>

<x-slot name="form">
    <form method="POST" action="{{ route('director.store') }}" class="grid grid-cols-6 gap-6">
        @csrf

        <!--First name-->
        <div class="col-span-6 sm:col-span-3">
            <x-label for="first_name" :value="__('First name')"/>

            <x-input id="first_name"
                     class="block mt-2 w-full"
                     type="text"
                     name="first_name"
                     :value="old('first_name')"
                     placeholder="Enter the first name"
                     maxlength="35"
                     required/>

            <x-input-error for="first_name" class="mt-2"/>
        </div>

        <!--Last name-->
        <div class="col-span-6 sm:col-span-3">
            <x-label for="last_name" :value="__('Last name')"/>

            <x-input id="last_name"
                     class="block mt-2 w-full"
                     type="text"
                     name="last_name"
                     :value="old('last_name')"
                     placeholder="Enter the last name"
                     maxlength="35"
                     required/>

            <x-input-error for="last_name" class="mt-2"/>
        </div>

        <!--Username-->
        <div class="col-span-6 sm:col-span-3">
            <x-label for="username" :value="__('Username')"/>

            <x-input id="username"
                     class="block mt-2 w-full"
                     type="text"
                     name="username"
                     :value="old('username')"
                     placeholder="Enter the username"
                     maxlength="20"
                     required/>
```

```php
/*Director creation form*/
public function create(): view
{
    return view('dashboard.director.create');
}

/*Save the director's record*/
public function store(UserInformationRequest $request): RedirectResponse
{
    $validated = $request->validated();

    $password_generated = PasswordHelper::generatePassword();

    $director_role = Role::where('name', 'director')->first();

    $director = $director_role->users()->create([
        'first_name' => $validated['first_name'],
        'last_name' => $validated['last_name'],
        'username' => $validated['username'],
        'email' => $validated['email'],
        'birthdate' => DateHelper::verifyDateFormat($validated['birthdate']),
        'phone_number' => $validated['phone_number'],
        'home_phone_number' => $validated['home_phone_number'],
        'address' => $validated['address'],
        'password' => Hash::make($password_generated),
    ]);

    $director->image()->create([
        'path' => $director->generateAvatarUrl(),
    ]);

    $director->notify(
        new RegisteredUserNotification(
            $director->getFullName(),
            $director_role->name,
            $password_generated
        )
    );

    return back()->with('status', 'Director created successfully');
}
```

```php
public function edit(User $user): View
{
    return view('dashboard.director.update', [
        'director' => $user
    ]);
}

/*Save the update of the director's record*/
public function update(UpdateUserInformationRequest $request, User $user): RedirectResponse
{
    $validated = $request->validated();

    $old_email = $user->email;

    $director = $user;
    $director->first_name = $validated['first_name'];
    $director->last_name = $validated['last_name'];
    $director->username = $validated['username'];
    $director->email = $validated['email'];
    $director->birthdate = DateHelper::verifyDateFormat($validated['birthdate']);
    $director->phone_number = $validated['phone_number'];
    $director->home_phone_number = $validated['home_phone_number'];
    $director->address = $validated['address'];
    $director->save();

    $director->updateUIAvatar($director->generateAvatarUrl());

    $this->verifyEmailChange($director, $old_email);

    return back()->with('status', 'Director updated successfully');
}
```

Add Guard :

```php
public function create(): view
{
    return view('dashboard.guard.create');
}

/*Save the guard's record*/
public function store(UserInformationRequest $request): RedirectResponse
{
    $validated = $request->validated();

    $password_generated = PasswordHelper::generatePassword();

    $guard_role = Role::where('name', 'guard')->first();

    $guard = $guard_role->users()->create([
        'first_name' => $validated['first_name'],
        'last_name' => $validated['last_name'],
        'username' => $validated['username'],
        'email' => $validated['email'],
        'birthdate' => DateHelper::verifyDateFormat($validated['birthdate']),
        'phone_number' => $validated['phone_number'],
        'home_phone_number' => $validated['home_phone_number'],
        'address' => $validated['address'],
        'password' => Hash::make($password_generated),
    ]);

    $guard->image()->create([
        'path' => $guard->generateAvatarUrl(),
    ]);

    $guard->notify(
        new RegisteredUserNotification(
            $guard->getFullName(),
            $guard_role->name,
            $password_generated
        )
    );

    return back()->with('status', 'Guard created successfully');
}
```

Guard List :

```php
public function index(): View
{
    $guard_role = Role::where('name', 'guard')->first();

    $guards = $guard_role->users();

    if (request('search')) {
        $guards = $guards->where('username', 'like', '%' . request('search') . '%');
    }

    $guards = $guards->orderBy('first_name', 'asc')
        ->orderBy('last_name', 'asc')
        ->paginate();

    return view('dashboard.guard.index', [
        'guards' => $guards,
    ]);
}
```

Assign Guard :

```php
private int $allowed_number_of_guards_per_ward = 2;

public function __construct()
{
    $this->middleware('can:manage-assignment');
    $this->middleware('active.user')->except('index');
    $this->middleware('verify.user.role:guard')->except('index');
}

public function index(): View
{
    $guard_role = Role::where('name', 'guard')->first();

    $guards = $guard_role->users();

    if (request('search')) {
        $guards = $guards->where('username', 'like', '%' . request('search') . '%');
    }

    $guards = $guards
        ->orderBy('first_name', 'asc')
        ->orderBy('last_name', 'asc')
        ->paginate();

    $wards = Ward::orderBy('name', 'asc')
        ->cursor()->filter(function ($ward) {
        return $this->allowed_number_of_guards_per_ward > $ward->users->count() && $ward->state;
    });

    return view('dashboard.assignment.guards-wards', [
        'guards' => $guards,
        'wards' => $wards->all()
    ]);
}

public function update(GuardWardRequest $request, User $user): RedirectResponse
{
    $validated = $request->validated();
    $guard = $user;

    if ($this->verifyItIsTheSameWard($guard->wards->first(), $validated['ward'])) {
        return back()->with([
            'status' => 'The guard is already in that ward.',
            'color' => 'yellow'
        ]);
    }

    //First, all user's ward relations are deactivated
    $guard_wards_id = $guard->wards->modelKeys();
    $guard->wards()->syncWithPivotValues($guard_wards_id, ['state' => false]);

    //Second, a new assigment is saved between user and jail
    $guard->wards()->sync($validated['ward']);

    return back()->with('status', 'Assignment updated successfully');
```

Add prisoner :

```php
public function create(): view
{
    return view('dashboard.prisoner.create');
}

/*Save the prisoner's record*/
public function store(UserInformationRequest $request): RedirectResponse
{
    $validated = $request->validated();

    $password_generated = PasswordHelper::generatePassword();

    $prisoner_role = Role::where('name', 'prisoner')->first();

    $prisoner = $prisoner_role->users()->create([
        'first_name' => $validated['first_name'],
        'last_name' => $validated['last_name'],
        'username' => $validated['username'],
        'email' => $validated['email'],
        'birthdate' => DateHelper::verifyDateFormat($validated['birthdate']),
        'phone_number' => $validated['phone_number'],
        'home_phone_number' => $validated['home_phone_number'],
        'address' => $validated['address'],
        'password' => Hash::make($password_generated),
    ]);

    $prisoner->image()->create([
        'path' => $prisoner->generateAvatarUrl(),
    ]);

    return back()->with('status', 'Prisoner created successfully');
}
```

```blade
<x-slot name="form">
    <form method="POST" action="{{ route('prisoner.store') }}" class="grid grid-cols-6 gap-6">
        @csrf

        <!--First name-->
        <div class="col-span-6 sm:col-span-3">
            <x-label for="first_name" :value="__('First name')"/>

            <x-input id="first_name"
                     class="block mt-2 w-full"
                     type="text"
                     name="first_name"
                     :value="old('first_name')"
                     placeholder="Enter the first name"
                     maxlength="35"
                     required/>

            <x-input-error for="first_name" class="mt-2"/>
        </div>

        <!--Last name-->
        <div class="col-span-6 sm:col-span-3">
            <x-label for="last_name" :value="__('Last name')"/>

            <x-input id="last_name"
                     class="block mt-2 w-full"
                     type="text"
                     name="last_name"
                     :value="old('last_name')"
                     placeholder="Enter the last name"
                     maxlength="35"
                     required/>

            <x-input-error for="last_name" class="mt-2"/>
        </div>
```

• Possible problems break down

There are a few key features in the system, and they are big enough to make. For this reason, it is anticipated that the whole undertaking job will be divided into little tasks so that the highlighted work can be completed even more smoothly. Below is a possible breakdown of this suggested structure.

Database design and development:

- Attribute selection,
- Relationship building
- Database implementation

The Front-end User Interface:

- Choosing keywords
- Designing pages
- Making navigation

Role-Based Login System:

- Creating the user interface
- implementing role-based login
- Testing

Chapter 11 – Testing

In order to find and fix bugs and make sure the system works as intended, testing is a crucial part of software development. It entails checking that every feature functions as intended and certifying the system to predetermined standards.
Testing will be essential to our project's success in order to guarantee that elements like order processing, AI-generated print, logo addition, and color modification operate as intended. Additionally, it will include verifying how well the frontend, backend, and external APIs integrate.
Usually, a test case is made up of many parts that work together to provide a thorough testing strategy. These elements consist of:

Description:

A succinct synopsis of the test case's goal and purpose.
Precondition:

The initial setup or circumstances needed before the test procedures are carried out.
Steps:

A series of operations or exchanges that must be carried out during the test's execution.
Expected Result:

- The expected behavior or result that, in the event that the system operates as intended, should be seen.
- The observed behavior or outcome during the test case's execution is known as the "actual result."
- The criteria used to decide whether the test case passed or failed in light of the actual outcome are known as the pass/fail criteria.
- The particular setup or environment in which the test is carried out, including specifics like the operating system and browser.

Test Data:

Any particular information or inputs required to carry out the test scenario.
because the test case documentation is brief and easy to understand. All of our test cases will simply provide the test scenario, stages, anticipated and actual results, and pass/fail criteria. To make sure the system works and has all the features, we have conducted unit, module, and integration testing using various test scenarios.

Chapter 12 – Implementation

•Training:

It must understand how to utilize the system correctly in order to get the actual outputs of the system. Therefore, in order to operate the jail management system, the management must have the necessary training. since the majority of individuals are not used with using digital systems as opposed to manual ones.
The administrator must get training on how to log in to the system. Admin is a crucial position. Therefore, no one is able to log on to the system. A certain system exists. The admin must be aware of it. The administrator must next learn how to manage and modify information, as well as how to transfer jail officers, check prison details, check case details, check transfer information, check offer details, check new prison details, check court information, and check reports. Therefore, the administrator must have enough training in order to understand how to operate the jail administration system.
The policeman has the ability to log in to the system as well. Therefore, police officers must be trained to operate the system. A police officer can look up information on courts, transfers, and prisons.
They are also able to search for information. Information on visitors, inmates, officers, etc., may also be checked. Therefore, the officer must have flawless training to do their duties responsibly in order to operate the system effectively. For police personnel to effectively utilize the jail administration system, thorough training is also essential.
Additionally, a staff actor will play a police officer. They are able to utilize the system as well. Both the visitor's and the police officer's data must be registered. They must thus get flawless training. If not, they will enter incorrect data into the system. Therefore, personnel and users must be trained for effective work.

• Big Bang (parallel implementation plan, no pilot)

Big Bang is a technique for updating the existing or outdated framework by switching to a new electronic framework and updating the PCs to incorporate all the changes into result. In any case, it would completely change the work procedures and lower the costs. It usually lasts one or two days.

• Scaling

Scaling is a method for assessing and transferring the materials to the records that are made up of certain rubrics. Another way to put it is that scaling is the process of determining which noble chemicals are present in a range; it is a consistent arrangement of the quantities to which the substances are given.

- ## Load balancing:

This technique limits the system loads. The system hits allude to the load most of the time. Furthermore, hits really refer to the number of customers who are being incorporated into the system at the same moment. A reference to load balancing is made here. Through the load balancer, the load balancing is managed. The load balancer isolates the advancements into the servers. In the unlikely event that it does not isolate the loads into different servers, the system as a whole will be slower due to the transit to several customers at once. I haven't explained how to isolate the servers since this is a training endeavor. Nevertheless, I have already provided the main concepts.

Chapter 13 – Conclusions

Academic goals:

- The system has been put into place using a suitable approach, and the whole project has been documented in accordance with academic standards.
- The system feasibility area is examined.
- The database is properly constructed.

The system is assessed via several implementations and tests.

Individual goals:

The system is progressing well.
Creating the academic documentation, creating the necessary reports, and creating an error-free system.

They are explaining the following:

- **Goal:** Put the system into place using a certain, suitable technique. The DSDM atern philosophy has been chosen for this project. The main steps and techniques of the DSDM approach have been clearly explained in the philosophy section. Here, I am gradually focusing on the DSDM accomplishment rate as well as the problems that arose throughout the improvement period. I've also looked at how to overcome these kinds of problems that we can encounter.
- **Success-related factors:**
  Delivering an optimum output under a time constraint has always been a challenge in academic projects. When DSDM is used at the very beginning of a project, it greatly aids in the dissection of the project's requirements, which may then be arranged using MoSCoW prioritizing. The progressively solid development of DSDM greatly helps in breaking down the requirements into manageable chunks, which in turn makes it possible to complete the project quickly. DSDM provides an iterative component that has been used in the iterative creation and testing of this project.
- **Obstacles encountered:**
  Even if using DSDMs might make this project easier to manage and produce in a typically early foreseeable manner, it is quite challenging to meet the strict time limit when just one person is attempting to do it.
- **Measures done to overcome**:

  Additional working hours have been donated in order to successfully complete this project under a strict time restriction.Creating project documentation in accordance with academic norms. A project's quality is highly dependent on how it has been reported, or, to put it another way, the quality of the documentation. Therefore, we must develop appropriate project documentation with a clear concept and path in order to get successful outcome.

➢ **Success-related factors:**
Finding a method to provide useful and understandable documentation is crucial. Because of this, the data related to this project has been meticulously collected and carried out in a highly organized manner. In order to enhance the structure of the record, product-related queries have been conducted on various documentations, and when data is used from any source, a suitable reference approach has been completed.

➢ **Challenges:**
This documentation was divided into sixteen sections with a vast amount of territory to explore. Its goal was to precisely understand and extract exact facts.
Steps done to get over these issues: The heading function in Microsoft Word 2016 has been really helpful.

➢ **Obstacles encountered:**
Since relating the money-saving advantage inquiry to this framework is one of its most challenging components, we must take into account and examine each piece of equipment that will be used in conjunction with the framework's linking, contrasting, and evaluating of advantages.
The steps made to address this problem include identifying a possible way to compile the equipment's cost from various online sources.
The goal is to properly design the database.
It has been determined that MySQL 2008 will organize this planned endeavor based on the unique evidence of the significant number of qualities and linkages discovered.
Success-related factors:
Every component of the database functions flawlessly after this process is completed.
Obstacles encountered:
Since accurate information is essential for advancement, we should typically be aware of how to apply information types in order to have a precise information type in any database.
Steps taken to go past:
It was easy to overcome this problem since the information kinds for this planned venture database were carefully considered. Varchar information type was also taken into account to reduce the challenges.
Assess the system using various implementation and testing techniques.
Success-related factors: As the enterprise moves forward, testing has been conducted for each smaller element completed.

was put to the test and advanced to the next improvement step. This process made it easier to handle problems from the start and move on to the next phase.
Challenges: It was quite difficult to handle any outcome estimate prior to the testing being completed.

182-16-328

ORIGINALITY REPORT

14%	3%	0%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to NCC Education Student Paper	6%
2	Submitted to Daffodil International University Student Paper	2%
3	www.transparencymarketresearch.com Internet Source	1%
4	Submitted to BPP College of Professional Studies Limited Student Paper	1%
5	Submitted to University of Wales Institute, Cardiff Student Paper	1%
6	Submitted to Wentworth Institute Student Paper	1%
7	Submitted to Gitam University Student Paper	<1%
8	Submitted to AUT University Student Paper	<1%
9	Submitted to Khurasan University	

	Student Paper	<1%
10	Submitted to University of North Texas Student Paper	<1%
11	123dok.com Internet Source	<1%
12	Submitted to Arab Open University Student Paper	<1%
13	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<1%
14	Submitted to Western International College (WINC London) Student Paper	<1%
15	Submitted to EPOKA University Student Paper	<1%
16	Submitted to Pathfinder Enterprises Student Paper	<1%
17	Submitted to Bradford College, West Yorkshire Student Paper	<1%
18	Submitted to National School of Business Management NSBM, Sri Lanka Student Paper	<1%
19	Submitted to University of Greenwich Student Paper	<1%