Chapter 1: Introduction

1.1 Introduction

As digital tools become more integrated into everyday systems, even traditional practices are enhanced with simple yet effective web solutions. One such tradition is the purchase of coins during Dhanteras, an auspicious festival where buying precious metal symbolizes good fortune and prosperity. To manage the overwhelming demand and long queues typically seen during this period, a more organized approach is essential.

The project 'Coin Registration Portal' is a web-based token generation system developed to facilitate the booking of Dhanteras coins distributed by the Nepal Mint Division. The portal consists of a basic form where users can submit their personal details and, upon completion, receive a digital token. This token must then be presented at the Nepal Mint Division, the sole authorized outlet, to collect the booked coins. The system does not include multiple outlet options or time slot selection — it is designed solely to generate and manage tokens for coin collection. By introducing this simple online booking mechanism, the portal helps reduce crowding, streamline the distribution process, and ensure a more organized and convenient experience for users during the festive season.

1.2 Problem Statement

During the Dhanteras festival, purchasing commemorative coins is a widely followed tradition, symbolizing prosperity and good fortune. However, the current method of acquiring these special edition coins, which requires customers to visit the Nepal Mint Division in person, presents several challenges. Due to the high demand for these coins during this time, people often face long queues, leading to an inefficient and time-consuming experience. This creates unnecessary frustration for customers, who must wait for extended periods, and also poses logistical difficulties for the Nepal Mint Division in managing the crowd and ensuring fair distribution.

Furthermore, the lack of a structured booking system leads to uncertainties about availability, as there is no guaranteed way for individuals to secure their coins ahead of time. This results in missed opportunities for potential buyers and a lack of organization, which could affect the overall customer satisfaction and the efficiency of the distribution process.

To address these challenges, we propose the development of an online token-based booking system. The goal of this system is to streamline the coin booking process by allowing users to submit their information via a simple web form and generate a token that they can later present at the Nepal Mint Division to collect their coins. While the actual transaction and coin collection remain offline, the token system will introduce an organized mechanism to ensure that only those who have pre-booked their coins are eligible for collection, reducing crowd

congestion and waiting times. This solution will not only improve the experience for buyers but also provide the Nepal Mint Division with an efficient way to manage coin distribution during a busy and high-demand period. Ultimately, the system aims to provide users with a smoother, more predictable experience, making the Dhanteras coin acquisition process hassle-free and more enjoyable.

1.3 Objectives

The objective of the project can be stated as:

- To develop a user-friendly web portal that allows customers to easily submit their information and book tokens for purchasing coins during the Dhanteras festival.
- To streamline the booking process by providing a simple form for customers to generate tokens, ensuring a hassle-free experience and reducing the need for long queues at the Nepal Mint Division.
- To offer a secure and organized system for managing token reservations, ensuring that only pre-booked customers are eligible to collect their coins, thus maintaining fairness and efficiency in the distribution process.
- To enhance customer experience by reducing wait times and providing clarity on token availability and collection procedures, ensuring a smoother and more predictable festival coin-buying experience.
- To create a reliable platform that ensures the accurate processing of user information and token generation, supporting a seamless connection between the online booking system and physical coin collection at the Nepal Mint Division.

1.4 Scope and Limitation

1.4.1 Scope

The proposed system is designed to provide a simple and efficient platform for users to book tokens for purchasing commemorative coins during the Dhanteras festival. As a web-based application, it will be accessible to anyone with a device that supports a browser and an internet connection.

The Online Coin Booking Portal will allow users to easily fill out a form to generate a token for coin collection at the Nepal Mint Division. The form will request basic personal details such as name, contact information, and the number of coins the user wishes to reserve. Users will receive a unique token, which can be presented at the Nepal Mint Division during the festival to collect the coins. The system will not include time slot selection, but it will ensure that only those with a valid token can collect coins, ensuring organized and fair distribution.

The portal will provide a secure and seamless experience for users, offering an efficient solution to reduce the time spent waiting in line and managing the crowds during the festival. The platform will not handle financial transactions directly, as the payment and collection process will occur at the physical outlet. However, the system will focus on accurate token

generation, user data processing, and efficient management of token reservations.
In summary, the platform aims to simplify the token booking process, reduce crowd congestion, and provide an organized and transparent method for purchasing coins at the Nepal Mint Division. By offering a simple and accessible web-based interface, the system will enhance the overall experience for both customers and the organization involved.

1.4.2 Limitations

The different limitations of this project are:

- Internet dependency: Since the system is web-based, users without access to a stable internet connection may face difficulty in accessing and using the portal to book their tokens.
- Limited outlet: The token system is designed specifically for coin collection at the Nepal Mint Division, limiting the scope of distribution to a single outlet and preventing users from choosing alternate locations.
- No time slot selection: The system does not include an option for users to select specific time slots for token collection, which may still result in crowded situations at the Nepal Mint Division, particularly during peak hours.
- No payment integration: The platform focuses solely on the booking of tokens and does not handle any payment processing, meaning users must pay for their coins offline at the Nepal Mint Division, which may be less convenient for some users.
- Limited user feedback system: The system does not have a feature for users to provide feedback or rate the process of token booking, which could help improve the user experience in future iterations of the system.

1.5 Report Organization

The report has been prepared following the guidelines provided by Tribhuvan University. The report is separated into different chapters. Each chapter consists of various sub chapters with its content. The preliminary section of the report consists of Title Page, Acknowledgement, Abstract and Table of Contents.
The main report is divided into 5 chapters, which include:

1. Chapter 1: Introduction
   It includes the general overview of the system and the project. It includes the Problem Statement, Objectives, Scope/Limitations and the Development Methodology for the project and the system being developed.

2. Chapter 2: Requirement Analysis and Feasibility Study
   It includes the study of the current scenario/environment the system will be deployed into. It includes the study of the current trends, preferences of people, the existing systems, areas of improvement among others. It includes the requirement and feasibility analysis of the system that can be generated through the studies presented in the previous two chapters. It

will also include the Flowchart, ER and DFD for the system which specifies the workflow, entities, attributes and their relationships.

3. Chapter 3: System Design
   It includes the design of the database, forms and interface of the system. It also includes the implementation details of the selected methodology, and the details of the algorithm used.

4. Chapter 4: Implementation and Testing
   It includes the details of the different design and development tools used and the implementation details of the modules presented in the form of code snippets of functions, classes. It also includes the testing of the system with different test cases as per the requirement.

5. Chapter 5: Conclusion and Recommendations
   It includes the summary of the system and the project. It also includes the possibilities/aspects which the system can implement in the future.

The final part of the report consists of References and Appendices. The references are listed in accordance with the IEEE referencing standards and the Appendices includes the screenshots of the system and the major source code snippets.

Chapter 2: Requirement Analysis and Feasibility Analysis

2.1 Literature Review

The integration of e-governance has transformed how public services are delivered, particularly by making them more accessible, transparent, and efficient. Online platforms in various sectors have significantly reduced the need for physical visits, improving user experience by automating services and reducing administrative overhead. This shift has proven particularly useful in managing high-demand services, such as the Dhanteras commemorative coin distribution, where traditional methods often lead to long queues and inefficiencies.

In the context of e-governance, digital service delivery plays a central role by offering users an easy, online platform to complete transactions without the need to visit government offices. The Online Coin Booking Portal follows this model by allowing users to reserve tokens online for coin collection at the Nepal Mint Division. This digital solution aims to streamline the entire process, reducing crowding and ensuring an organized and fair distribution of coins during the busy festival period.

Research shows that successful e-governance platforms focus on a few key factors: ease of use, security, and efficiency. A user-friendly interface is essential for ensuring that all citizens, regardless of digital literacy, can easily navigate the system. Additionally, secure data handling and transparent processes are vital to maintaining public trust in government-run systems. The Online Coin Booking Portal will incorporate these principles, ensuring that user data is protected through encryption and that the booking process is both simple and transparent.

Crowd management is another significant advantage of adopting an online booking system in the context of e-governance. By providing users with the ability to pre-book their coins, the portal minimizes the uncertainty and chaos often associated with in-person queuing. This approach ensures a more organized, fair, and predictable coin collection process, benefiting both the Nepal Mint Division and the users.

Moreover, the platform's focus on digital accessibility ensures that it can be easily accessed by a broad range of users, including those with limited digital skills or access to advanced technology. The integration of simple forms, clear instructions, and mobile compatibility allows more people to participate in the coin booking process, enhancing inclusivity.

In conclusion, the Online Coin Booking Portal draws on the principles of e-governance to offer a streamlined, secure, and efficient solution for coin distribution during Dhanteras. By prioritizing user-friendliness, security, and effective crowd management, the platform aims to improve the overall experience for both users and the Nepal Mint Division.

Study of Existing System

1. Nepal Government's Online Passport System

The Nepal Government's Online Passport System is an e-governance platform that allows citizens to book appointments for passport applications using a token-based reservation system. This system streamlines the process by allowing users to reserve a time slot for submitting their passport applications, reducing long waiting times and physical visits.
Good Features:

- Token-based booking: Citizens can reserve time slots for passport services, which helps avoid overcrowding.
- Real-time updates: Users receive notifications about their appointments and any changes.
- E-governance integration: The system is part of Nepal's effort to digitize public services, making it more accessible for citizens.
- Secure data management: Sensitive personal information is handled securely.
  This system's token reservation process can be adapted to your coin booking platform, where users can book tokens for limited-edition coin collections.

2. Ravi Coins (Limited Edition Coin Booking)

Ravi Coins is a platform where users can book limited-edition commemorative coins during special events. The system utilizes a reservation-based model to ensure customers can secure the coins before they are sold out, making it very similar to your coin booking portal.
Good Features:

- Exclusive coin reservations: Users can book limited-edition coins during special time frames.
- Secure payment: The platform ensures secure transactions for booking and purchasing coins.
- Easy navigation: The interface is simple and user-friendly, allowing customers to quickly reserve and purchase coins.
- Product details: Detailed descriptions and high-quality images help users make informed decisions.

This system's approach to limited edition coin reservations aligns closely with the functionality you plan to offer on your website.

3. Singapore's e-Appointment System for Public Services

Singapore's e-Appointment System allows users to book appointments for a variety of public services, including passport applications and driving license renewals. It uses a token-based system to manage these reservations and avoid overcrowding at government offices.
Good Features:

- Online token booking: Citizens can reserve time slots for government services online.
- Token confirmation: After booking, users receive a token number, which serves as confirmation of their appointment.
- E-governance integration: The system is part of Singapore's Smart Nation initiative to improve the delivery of government services.
  This system's use of token reservations for appointment management is a strong parallel to how users will reserve tokens for coin collection in your system.

2.2 Requirement Analysis

i. Functional Requirements

End-user
The customer should be able to perform the following activities in the system:

- View Coin Information: Users can view details of the available coin(s) for pre-booking, including the name, image, and description.

- Fill Pre-Booking Form: Users can enter their name, contact information, Citizenship id, pictures of the front and back of the citizenship card and select the desired quantity of coins to pre-book.

- Submit Booking Request: Upon completing the form, users can submit their request to pre-book coins.

- Receive Booking Token: After submission, the system generates a unique booking token (reference number) and displays it on-screen for the user to note. This token serves as proof of their booking.

Admin
The admin is responsible for managing the coin pre-booking platform and should be able to perform the following activities:

- View Submitted Bookings: Admin can access a list of all pre-booking form submissions, including user details such as name, contact number, quantity of coins booked, and uploaded citizenship card.

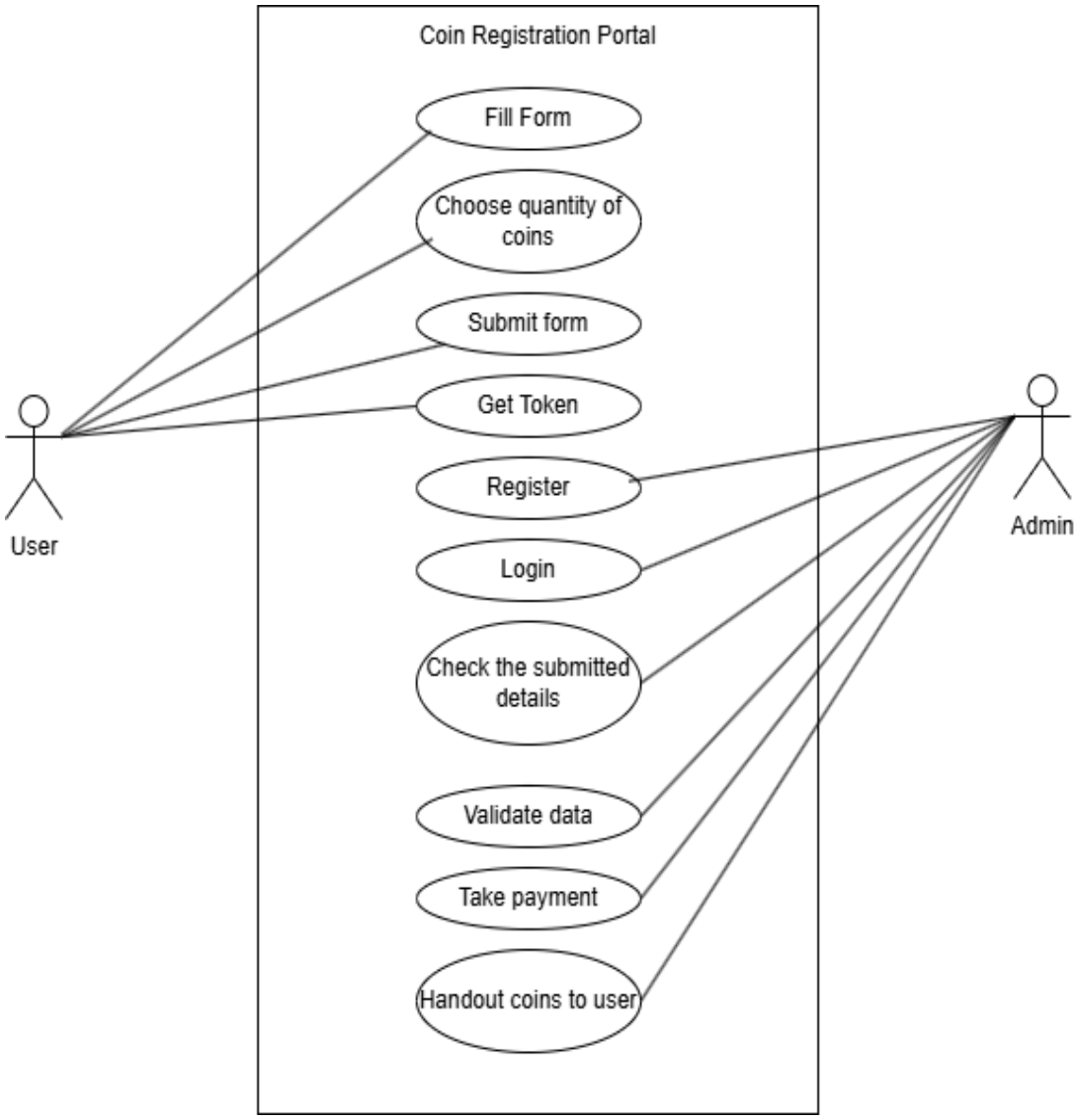
- Verify Citizenship Card: When users arrive in person, the admin can manually check the submitted citizenship card against the original ID presented by the user.

- Mark Booking as Completed: After verifying the user's identity and details, the admin can update the booking status to "Completed" to indicate that the token has been fulfilled.

- Register as Admin: Admin can register themselves on the platform by providing required details such as name, contact information, and a secure password to access the admin dashboard.

Our system allows two user privileges, one is the admin, and the other is the customer. The different activities that can be performed by the users in our system have been mentioned in the points above. And, the Use Case diagram below presents the same in a graphical format.

Use case diagram for Coin Registration Portal



Hardware Requirement

The hardware requirement for the project includes a laptop with support for HTML5, CSS, and other Implementation tools required for the project. The minimum system requirements for the laptop include: i3 8th Gen., 8 GB Ram, 512 SSD, Windows 11. As the final goal of the system is to be a demo for solving the problem related to governance, it is built and run in the developer system using local host. For industrialization, it can be hosted in Azure, AWS and other platforms.

Software Requirement

The software requirements for the project include implementation and development tools like Visual Studio, Asp .NET core, HTML5, CSS, JavaScript, MS SQL all of which are open-source programs so can be utilized in the project without any additional costs.

Draw.io is used to prepare the necessary diagrams for the project which is open source, so no additional cost is required.

ii. Non-Functional Requirements

- **Usability:**
  The website should be easy to navigate, with clear instructions for users to complete coin pre-booking.
- **Performance:**
  The website should be optimized for fast loading times, ideally under 3 seconds, once hosted.
- **Security:**
  Ensure all user data is securely handled, using HTTPS for encrypted communication, once hosted.
- **Availability:**
  Aim for high availability (99.9%) after hosting, with minimal downtime for updates or maintenance.
- **Responsiveness:**
  The design should be adaptable to different screen sizes and devices (desktop, mobile, tablet) after deployment.

2.3 Feasibility Analysis

A feasibility study, also known as feasibility analysis, is an analysis of the viability of an idea. It describes a preliminary study undertaken to determine and document a project's viability. The results of this analysis are used in making the decision whether to proceed with the project or not. In short, a feasibility analysis evaluates the project's potential for success, following feasibility analysis was performed prior to working on the project:

2.3.1 Technical Feasibility

- **Technology Stack**: The website uses simple web technologies such as HTML, CSS, and JavaScript, which are commonly used for basic websites. The system does not require advanced back-end development since it is a simple pre-booking platform without complex payment or authentication systems.

- **Scalability**: Given the simplicity of the website, scaling is not a major concern initially. However, it is important to choose a hosting provider that allows easy upgrading of resources (e.g., more storage or bandwidth) if user traffic increases.

2.3.2 Operational Feasibility

- **Ease of Use**: The website is simple and intuitive, requiring no advanced technical skills from users. The form for pre-booking coins is straightforward, minimizing user friction.

- **Maintenance**: The website requires minimal maintenance. Since the design is simple, updates and changes are easily implemented as needed. Basic monitoring for uptime and performance is required.

2.3.3 Economic Feasibility

- **Development Costs**: The development costs for a simple website are low, especially if pre-built frameworks or a content management system (CMS) are used. The primary costs are hosting, domain registration, and any potential future updates.

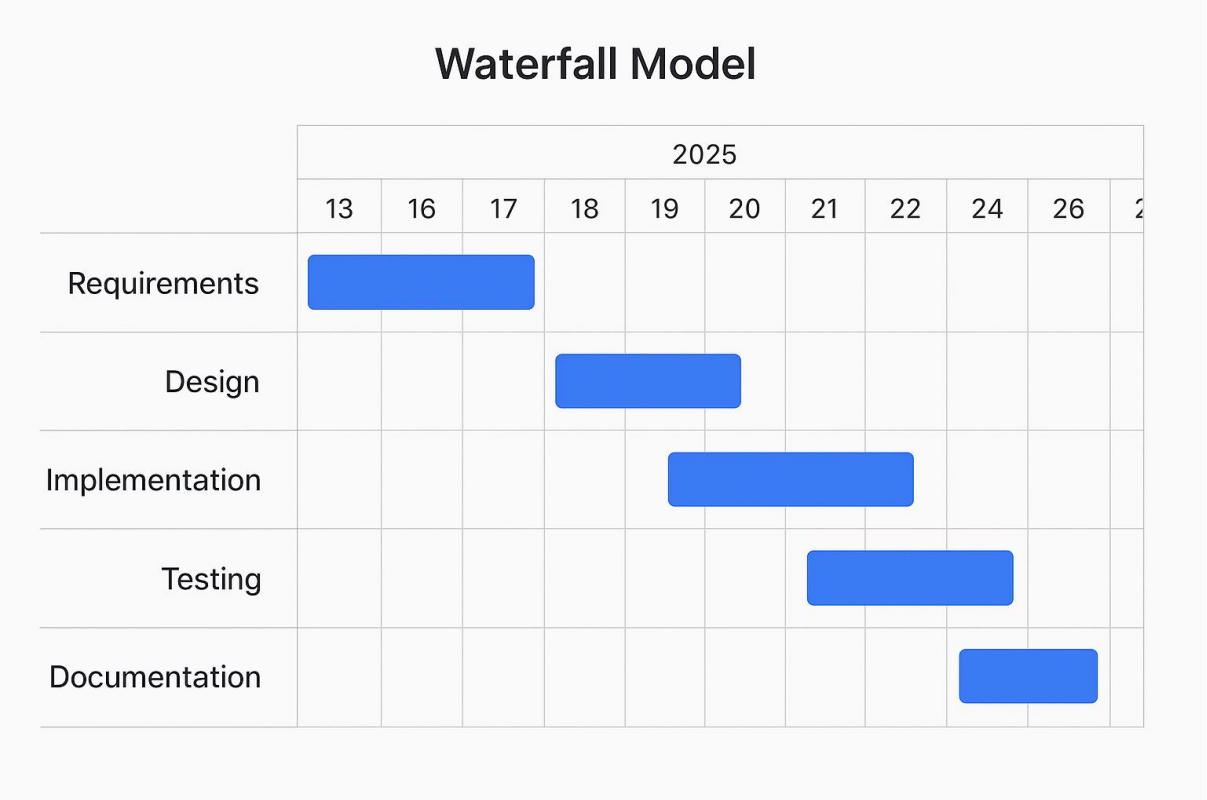
- **Hosting Costs**: Shared hosting services or entry-level cloud hosting are sufficient for this website. Monthly hosting costs are minimal and budget friendly.

2.3.4 Scheduling Feasibility (Gantt Chart)

The project expands over a period of 15 days with tasks being divided along with duration in the time frame. For the system to be delivered in the specified time frame we must make sure deadlines are met and tasks are executed accordingly in the specified duration period.

Table 1 Gantt Chart

Phase	Start Date	End Date	Duration
Requirement Analysis	Jan 13, 2025	Jan 14, 2025	2 days
System Design	Jan 15, 2025	Jan 16, 2025	2 days
Implementation	Jan 17, 2025	Jan 23, 2025	7 days
Testing	Jan 24, 2025	Jan 25, 2025	2 days
Documentation	Jan 26, 2025	Jan 27, 2025	2 days

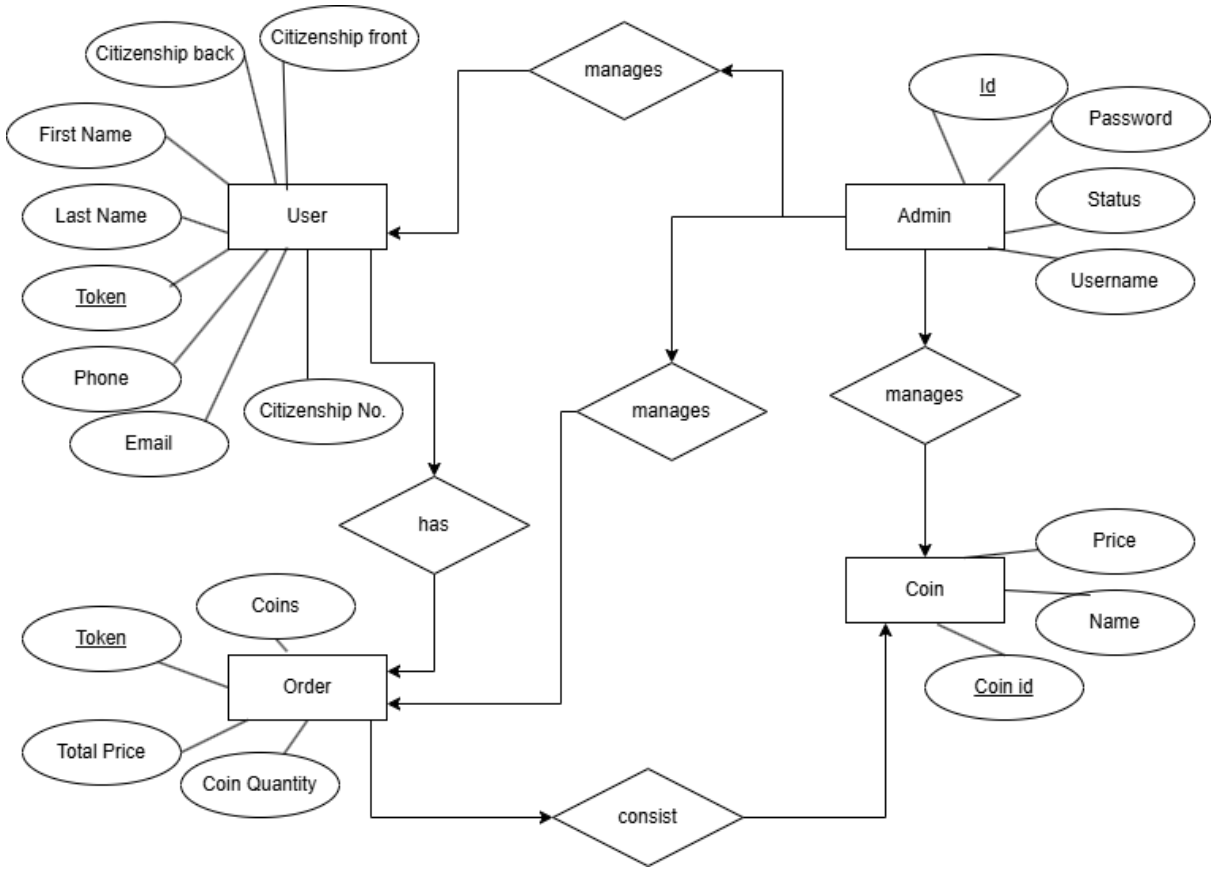


2.4 Structuring System Requirements

The system requirements are organized based on two distinct user roles: End-user (Customer) and Admin. The customer focuses on browsing and pre-booking coins, while the admin manages the platform, including handling customer requests and managing the website content. These requirements ensure smooth user experience for customers and efficient administration for the website operators.

2.4.1 Data Modeling

- **ER Diagram**

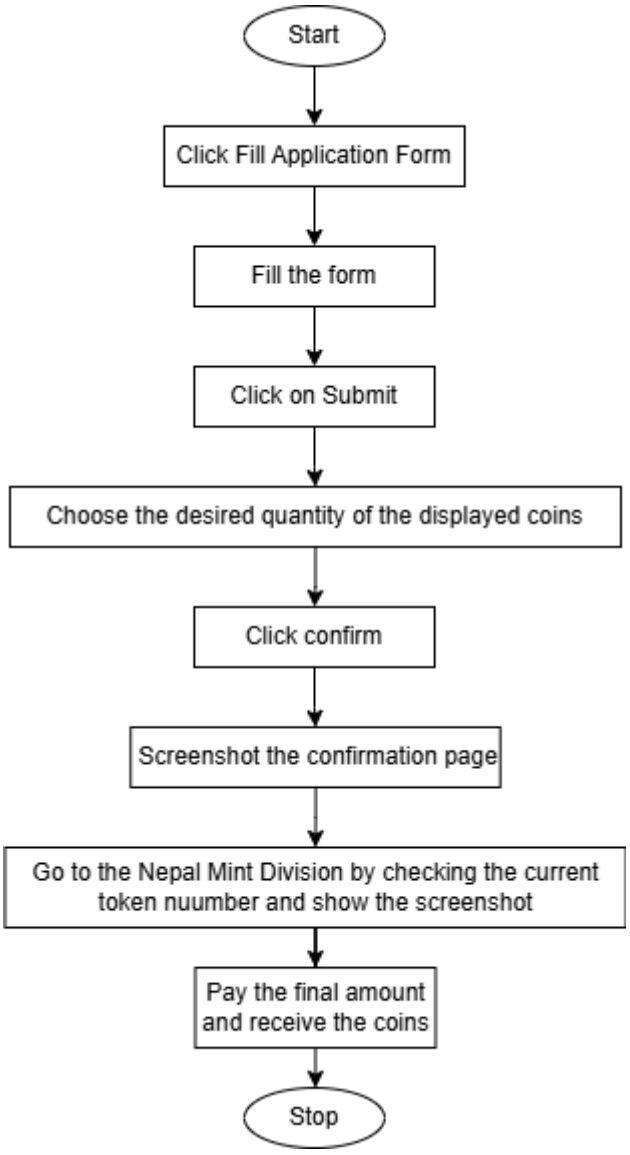


2.4.2 Process Modeling

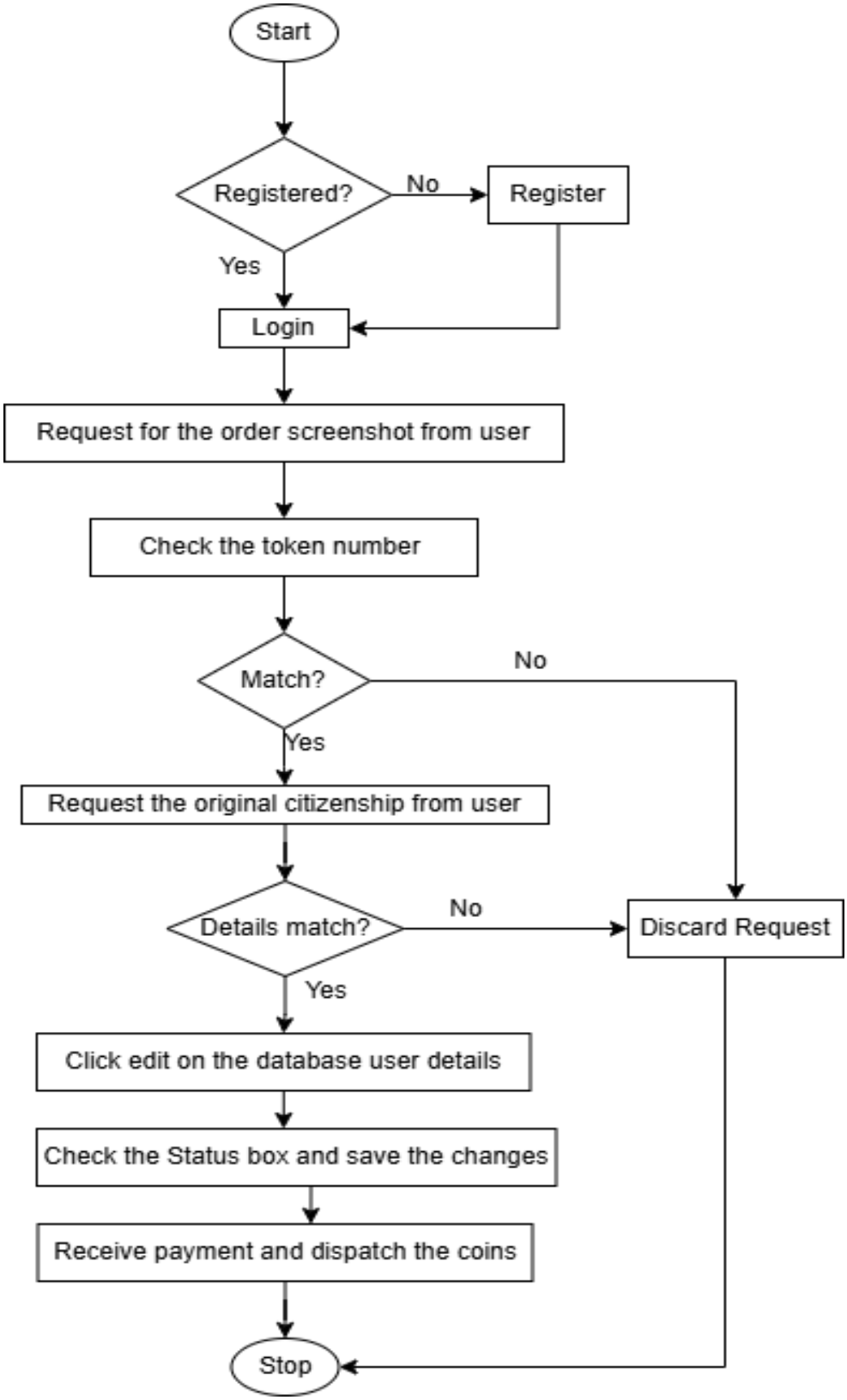
- **Flowchart**

  A flowchart is a diagram that visually represents a process, system, or algorithm. It uses symbols to denote different types of actions or steps, and arrows to show the flow or sequence between them.

Flowchart for Customer



Flowchart for Admin

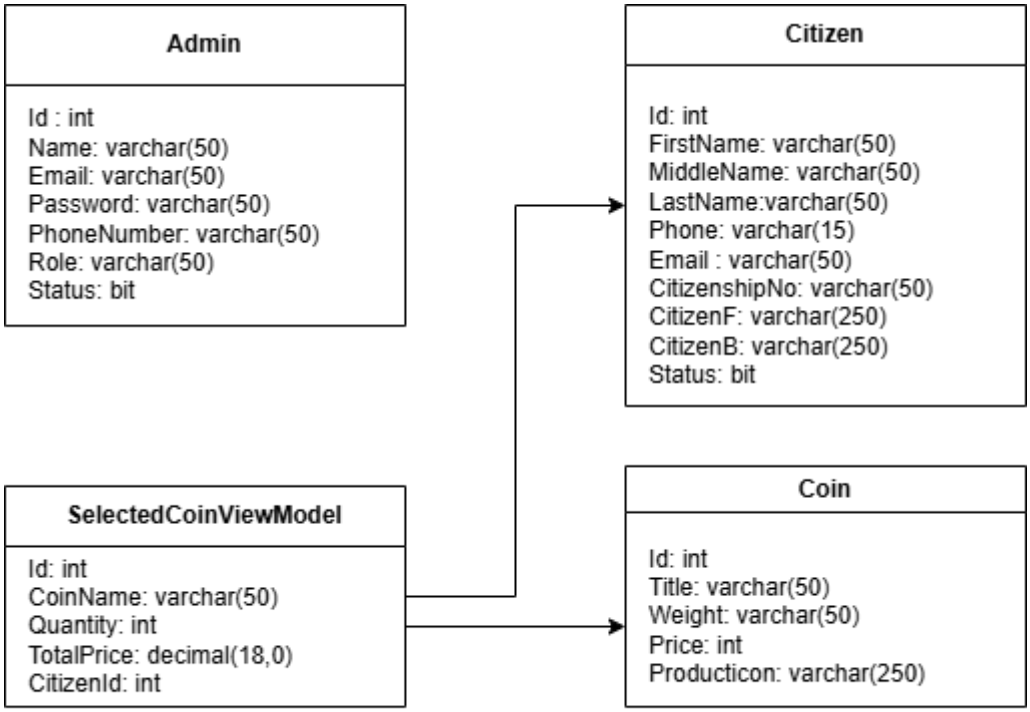


Chapter 3: System Design

The system design of Coin Registration Portal is crucial for ensuring scalability, performance, and maintainability. This chapter outlines the key aspects of Coin Registration Portal architecture, database design, and user interface design.

3.1 Database Design

Our MS SQL database schema is designed to efficiently store and retrieve data for Coin Registration Portal. The schema includes tables for citizens, coin, selected coin view model and admin. Relationships between tables are established using foreign keys to maintain data integrity and ensure efficient querying. Database indexes are utilized to optimize query performance, especially for frequently accessed columns such as titles, names and IDs. Normalization techniques are applied to reduce redundancy and improve data consistency. Here is a simplified overview of the database schema:



3.2 System Architecture

3.2.1 Client-Side (Frontend)

- Users
  - Fill and submit the form
  - Choose quantity of each coin
  - Take a screenshot of the confirmation page for token and selected coins detail

- Admin
  - Manages the citizen database using the admin portal

- Register and login to gain access to administrative tasks
- Check the original citizenship to the photos uploaded by the users in the form for confirmation

- Frontend Technologies
  - HTML, CSS (Bootstrap)
  - JavaScript (for dynamic interactions)
  - Razor Pages
  - Static files (CSS, JS, Images)

3.2.2 Backend (ASP .NET Core MVC Web App)

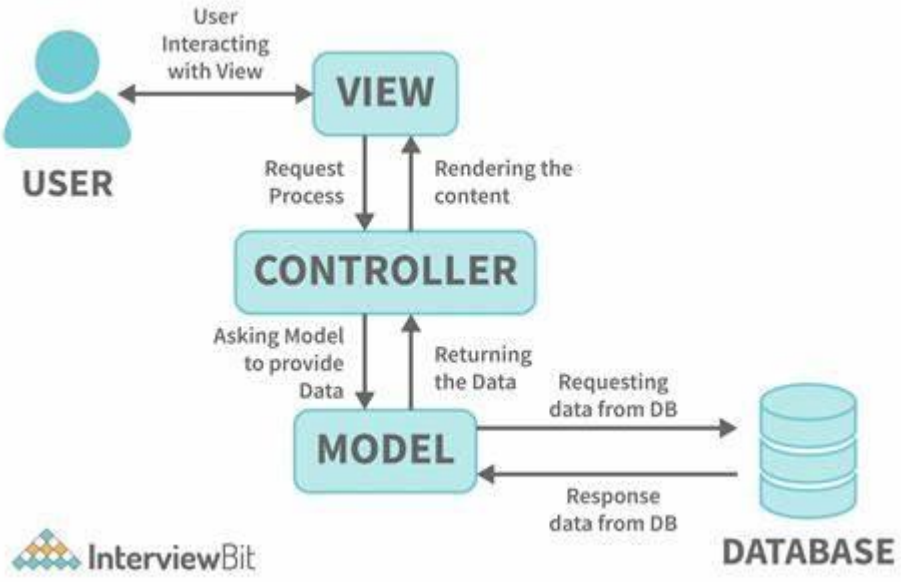
- ASP .NET Core Modules

  - Routing (Maps HTTP requests to controllers and actions)
  - MVC (Separates application into Model, View, and Controller for structured development)
  - Dependency Injection (Injects services and dependencies into controllers and other components)
  - Authentication & Authorization (Verifies user identity and checks access permissions)
  - Middleware (Handles requests and responses in the pipeline)
  - Logging (Captures application logs like errors, warnings, and information)

3.2.3 Database (MS SQL)

- Citizen Table: Stores citizens information who have submitted form
- Coin Table: Stores coin with their name, weight and price
- Admin Table: Stores registered admin users information
- Selected Coin Table: Stores order details including name, quantity and total price.

System Architecture Diagram:



3.3 User Interface Design

The Coin Registration Portal offers a simple, user-friendly interface that prioritizes functionality, ease of use, and a seamless user experience. Built using ASP.NET Core MVC for the backend, the website ensures smooth interactions while keeping the design clean and effective for all users.

Key Design Principles:

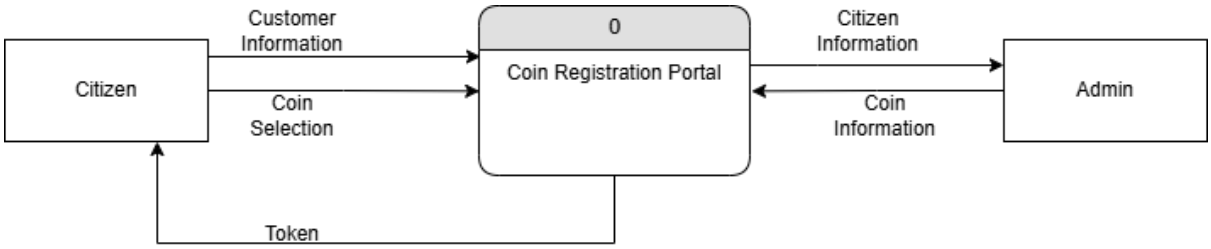
- **Responsive Layout**: The website adjusts fluidly to different screen sizes (desktops, tablets, smartphones), ensuring an optimal viewing experience on all devices.
- **Intuitive Navigation**: A clear and straightforward layout with easy-to-use forms, a search feature, and organized categories for booking coins makes navigation intuitive for all users.
- **Visually Simple and Clean**: A minimalist design with a consistent color palette, simple fonts, and plenty of whitespace ensures that the site is visually appealing without being overwhelming.
- **Consistent Design Language**: Consistent use of typography, colors, and design elements ensures a cohesive look, making the site feel professional and easy to navigate.
- **Accessibility Considerations**: The site is designed with accessibility in mind, featuring proper color contrast, alt text for images, and support for keyboard navigation, ensuring an inclusive experience for all users.

This design approach ensures that Coin Registration Portal is a functional, fast, and engaging platform, providing users with a seamless experience as they pre-book coins and explore the site.

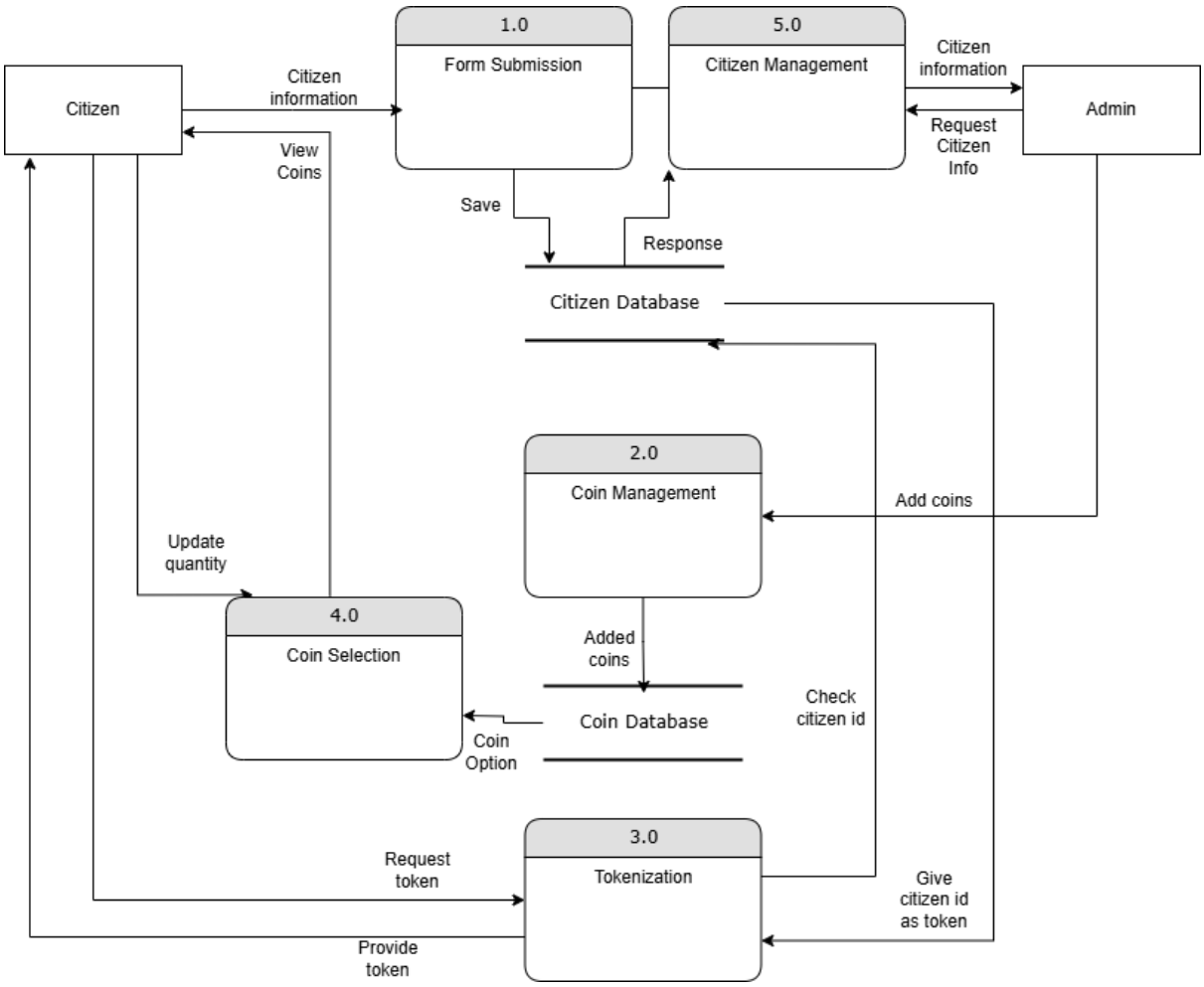
3.4 Process Design

- **DFD (Data Flow Diagram)**

Data Flow Diagram (DFD) is a graphical representation that depicts how data moves through a system. It focuses on the flow of information between processes, data stores, and external entities rather than detailing program logic or control flow. DFDs are beneficial in analyzing the functionality of a system and in documenting or designing system interactions.



Level 0 DFD


1.0
Form Submission
5.0
Citizen Management
Citizen
Citizen information
Citizen information
Admin
Request Citizen Info
View Coins
Save
Response
Citizen Database
2.0
Coin Management
Add coins
Update quantity
4.0
Coin Selection
Added coins
Coin Database
Check citizen id
Coin Option
3.0
Tokenization
Request token
Give citizen id as token
Provide token

Level 1 DFD

Chapter 4: Implementation and Testing

4.1 Implementation

This chapter details the implementation and testing of Coin Registration Portal core components, highlighting key features and technologies used in the front end, backend, and database layers.

4.1.1 Frontend

The front end of Coin Registration Portal is built using ASP.NET Core MVC with Razor Views, offering a simple, form-based user interface. The focus is on ensuring a smooth and straightforward user experience with minimal distractions.

- Template-Based Architecture: The UI is structured using Razor Views, allowing for a clean and reusable design. This modular approach simplifies maintenance and updates.
- Responsive Design: CSS and Bootstrap are utilized to ensure that the website adapts fluidly to various screen sizes, providing an optimal experience on desktops, tablets, and smartphones.
- Navigation & User Interaction:
  - The website features a simple and organized layout, with clear forms for admin registration, login, and basic functionality.
  - Interactive elements like form validation and error messages ensure a seamless user experience for administrators.
- Form Handling: ASP.NET Core MVC's built-in form handling capabilities are used for admin registration and login, providing secure and efficient user interactions.
- Lazy Loading: Basic content such as images or forms is loaded asynchronously, optimizing performance for better user experience.

4.1.2 Backend

The backend of **[Your Website Name]** is built using **ASP.NET Core MVC**, with a focus on simplicity and functionality for managing basic admin tasks and form submissions.

- Authentication: ASP.NET Core Identity is used to manage admin registration, login, and session management, ensuring secure access to the system for administrators.
- Business Logic:
  - Admin Users can register, log in, and manage their accounts.
  - The website doesn't include advanced features like user reviews, bookmarks, or other complex interactions, keeping the focus on simple form-based functionalities.
- Payment Integration:
  - Cash on Delivery (COD) is the only available payment method for users. There are no integrated payment gateways, and all transactions are handled offline at the time of delivery.
- Security Measures:
  - CSRF protection, input validation, and anti-forgery tokens are implemented to ensure a secure system.
  - The application ensures protection against common threats like SQL injection through manual SQL query handling and prepared statements.

4.1.3 Database

The backend uses MS SQL Server for data storage and management, with manual management of database schemas and queries via MS SQL Management Studio.

- Schema Design:
  - The database includes basic tables such as Admin, Citizen, Coin, SelectedCoin to handle essential operations.
  - Data is managed through direct SQL queries and stored procedures, ensuring efficient handling of data without using an ORM.
- Efficient Queries: Raw SQL queries are used to retrieve, insert, and update data efficiently, optimizing database interactions.
- Database Schema Changes: Any schema changes, such as table additions or modifications, are handled manually through MS SQL Management Studio and SQL scripts.

This architecture ensures that Coin Registration Portal is secure, simple, and reliable, providing administrators with an efficient, easy-to-use platform for handling basic tasks like user registration and payment processing.

4.1.4 Code Snippets

This action is triggered when the user submits the form and saves the information in database.

```
[HttpPost]
[ValidateAntiForgeryToken]
[AllowAnonymous]
0 references
public async Task<IActionResult> Create([Bind("Id,FirstName,LastName,MiddleName,Phone,Email,CitizenshipNo," +
    "CitizenF,CitizenB,SelectedViewModels")] Citizen citizen, IFormFile CitizenF, IFormFile CitizenB)
{
    if (ModelState.IsValid)
    {
        // Check if Citizenship Number already exists
        var existingCitizen = await _context.Citizen
            .FirstOrDefaultAsync(c => c.CitizenshipNo == citizen.CitizenshipNo);

        if (existingCitizen != null)
        {
            // Add an error message if the Citizenship Number already exists
            ModelState.AddModelError("CitizenshipNo", "A citizen with this Citizenship Number already exists.");
            return View(citizen);
        }

        // Save Citizenship F and B images
        Guid guid1 = Guid.NewGuid();
        string fileEx1 = CitizenF.FileName.Substring(CitizenF.FileName.LastIndexOf('.'));
        string path1 = Environment.CurrentDirectory + "/wwwroot/CitizenshipImages/" + guid1 + fileEx1;
        FileStream fileStream1 = new FileStream(path1, FileMode.Create);
        await CitizenF.CopyToAsync(fileStream1);
        citizen.CitizenF = guid1 + fileEx1;

        Guid guid2 = Guid.NewGuid();
        string fileEx2 = CitizenB.FileName.Substring(CitizenB.FileName.LastIndexOf('.'));
        string path2 = Environment.CurrentDirectory + "/wwwroot/CitizenshipImages/" + guid2 + fileEx2;
        FileStream fileStream2 = new FileStream(path2, FileMode.Create);
        await CitizenB.CopyToAsync(fileStream2);
        citizen.CitizenB = guid2 + fileEx2;

        // Add the new citizen to the database
        _context.Add(citizen);
        await _context.SaveChangesAsync();
        if (citizen.SelectedCoinViewModels != null)
        {
            foreach (var coin in citizen.SelectedCoinViewModels)
            {
                var selectedCoin = new SelectedCoinViewModel
```

```
                {
                    CoinName = coin.CoinName,
                    Quantity = coin.Quantity,
                    TotalPrice = coin.TotalPrice,
                    CitizenId = citizen.Id  // Link the selected coin to the citizen
                };

                _context.Add(selectedCoin);
            }
            await _context.SaveChangesAsync();
        }


        TempData["CitizenId"] = citizen.Id;
        return RedirectToAction("CoinDashboard", "Coin");
    }
    return View(citizen);
}
```

- This action is triggered when the admin verifies the citizenship card from the users and must check the Status box in the user database after confirmation.

```
// POST: Citizen/Edit/5
// To protect from overposting attacks, enable the specific properties you want to bind to.
// For more details, see http://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
0 references
public async Task<IActionResult> Edit(int id, [Bind("Id,FirstName,LastName,MiddleName,Phone" +
    ",Email,CitizenshipNo,CitizenF,CitizenB,Status")] Citizen citizen)
{
    if (id != citizen.Id)
    {
        return NotFound();
    }

    if (ModelState.IsValid)
    {
        try
        {

            _context.Update(citizen);
            await _context.SaveChangesAsync();
        }
        catch (DbUpdateConcurrencyException)
        {
            if (!CitizenExists(citizen.Id))
            {
                return NotFound();
            }
            else
            {
                throw;
            }
        }
        return RedirectToAction(nameof(Index));
    }
    return View(citizen);
}
```

4.2 Testing

Testing is a critical phase in the development of Coin Registration Portal to ensure that the website is functional, secure, and performs efficiently. This chapter outlines the testing strategies used for the front end, backend, database, and basic functionality.

4.2.1 Unit Testing

- **Frontend**
  - Template Rendering: Razor Views are tested to verify correct page rendering, layout consistency, and responsiveness across different screen sizes.
  - Form Validation: The forms used for admin registration, login, and coin bookings are tested to ensure proper input validation and error handling.
  - Session Management: Sessions are tested to ensure proper data persistence across user interactions, especially during admin login and handling booking information.

- **Backend**
  - Business Logic: Core functionalities, including admin registration, coin booking, and order processing, are tested for accuracy, edge cases, and expected behavior.
  - Authentication: The admin login functionality is tested for secure login, session management, and correct access control.
  - Form Handling: Validation of booking forms and user input are tested to ensure proper handling of form data, including error messages and data persistence.

- **Database**
  - Schema Validation: The database schema and changes (if any) are tested to ensure smooth updates and proper relationship management between tables like Admin, Bookings, and Payments.
  - Data Integrity: Data constraints such as foreign keys, required fields, and unique attributes are tested to prevent any corruption or mismatches in the data.
  - Direct SQL Queries: Since no ORM is used, raw SQL queries are tested for efficiency and correctness, ensuring that all database operations are optimized and correctly executed.

4.2.2 Integration Testing

- Frontend-Backend Communication: End-to-end tests validate data consistency between Razor Views (frontend) and controller actions (backend), ensuring data is passed and displayed correctly.
- User Flows: Key workflows such as admin registration, admin login, coin booking, and order processing are tested to ensure seamless functionality and flow from one step to another.
- Order Processing: The entire process of order creation, order status updates, and order management is tested for correctness, ensuring that all order details are properly stored and updated.

4.2.3 Performance Testing

- Response Time Benchmarking: The performance of controller actions and SQL queries is measured to ensure they execute quickly and efficiently.
- Lazy Loading Evaluation: The impact of lazy loading on coin images and page load

times are evaluated to ensure performance is not compromised as images are loaded asynchronously.

4.2.4 Security Testing

- Authentication Security: The admin login system is tested for vulnerabilities such as session hijacking, brute-force attacks, and ensuring that admin privileges are appropriately secured.
- Input Validation: Forms and input fields are tested for security vulnerabilities, including SQL injection, Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF), ensuring that all user inputs are sanitized and validated.
- Authorization Checks: The system is tested to ensure that role-based access control is working properly, preventing unauthorized users from accessing admin sections or performing restricted actions.
- Database Security: Security measures related to MSSQL are reviewed, ensuring proper user roles and permissions for the database, and that stored procedures are secure from SQL injections.

4.2.5 User Acceptance Testing (UAT)

Before deployment, real users (admin staff and possible testers) perform User Acceptance Testing (UAT) to evaluate the website's usability, functionality, and overall experience.

- User Feedback: Feedback from the testing phase is collected to refine the platform, ensuring that it meets the expectations of the administrators, and the simplicity required for the user experience.
- Ease of Use: The overall user experience is assessed to ensure that navigating through the admin registration, coin bookings, and order management processes is intuitive and straightforward.

This structured testing strategy ensures that Coin Registration System is functional, secure, and optimized for both administrators and users, providing a smooth, reliable platform for booking coins and managing admin-related tasks.

Chapter 5: Conclusion and Recommendations

5.1 Conclusion

Coin Registration Portal is a streamlined and efficient platform designed to provide users with simple yet effective experience for booking coins. By integrating straightforward navigation, secure processes, and an easy-to-use interface, the platform ensures a hassle-free experience for users and administrators alike.

The development process involved careful planning, implementation, and testing to ensure that the system meets functional requirements while delivering an intuitive user interface. Built using ASP.NET Core MVC and MSSQL, the platform offers a reliable backend solution and efficient data management. With Cash on Delivery (COD) as the only payment method, the system guarantees secure transactions and an easy checkout experience for users.

As Coin Registration Portal continues to grow, future enhancements such as the integration of additional payment gateways, user personalization features (such as booking history and preferences), and expanded admin functionalities will enhance the platform's usability and overall performance. These improvements will further streamline user interactions, ensuring that the platform remains secure, user-friendly, and optimized for future needs.
With a commitment to providing a seamless experience for both users and administrators, Coin Registration Portal is poised for long-term success. It offers a reliable, simple, and effective solution for booking coins, connecting users to the services they need with ease and efficiency.

5.2 Future Recommendations

Coin Registration Portal has been designed with future growth and scalability in mind. As the platform evolves, several key features and improvements can be implemented to enhance user engagement, broaden accessibility, and improve overall user experience.

5.2.1 Expanded Payment Gateway Integration

Currently, Cash on Delivery (COD) is the only payment option. Future developments will include integrating additional payment gateways to offer users a wider variety of payment methods. This could include options like PayPal, Stripe, and mobile payment systems, allowing users from various regions to transact seamlessly.

5.2.2 Mobile Application

To cater to users on the go, the development of a mobile application for both iOS and Android will offer an optimized browsing and booking experience. The app will streamline the booking process, allowing users to easily book coins and manage their preferences from their smartphones.

5.2.3 Admin Dashboard Enhancements

Future improvements to the admin dashboard can include enhanced reporting tools, user management, and more detailed insights into transactions and bookings. These features will help admins manage the platform more efficiently and provide better oversight of operations.

5.2.4 User Notifications and Alerts

A feature allowing email or SMS notifications will be added to keep users informed about their bookings, order statuses, and important updates. This will improve user engagement and communication between the platform and its users.

As Coin Registration Portal continues to develop, these enhancements will ensure that the platform remains relevant, user-friendly, and scalable, creating a more interactive and seamless experience for users and administrators alike.

5.3 References

The references section includes various sources used during the development, research, and implementation of Coin Registration Portal. Below are some key references that contributed to the platform's development:

Technical Documentation & Official Resources

- ASP.NET Core Documentation: Official documentation for ASP.NET Core framework used for backend development.
  - https://learn.microsoft.com/en-us/aspnet/core/
- Microsoft SQL Server Documentation: Official documentation for Microsoft SQL Server, the database management system used for data storage.
  - https://docs.microsoft.com/en-us/sql/sql-server/
- ASP.NET Core MVC Documentation: Guide to understanding and implementing the MVC pattern within ASP.NET Core.
  - https://learn.microsoft.com/en-us/aspnet/core/mvc/
- Cash on Delivery (COD) Integration: No specific API used, but documentation and guidelines on implementing COD payment methods.
  - Custom solution based on business requirements.

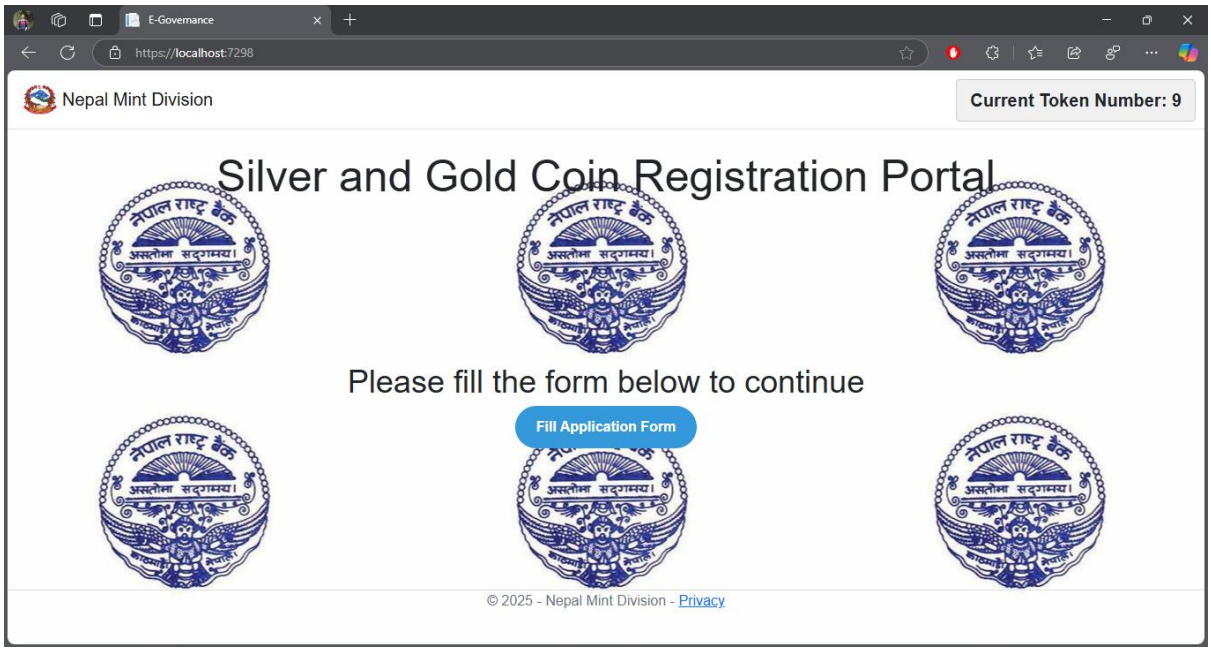
Web Technologies & Design Principles

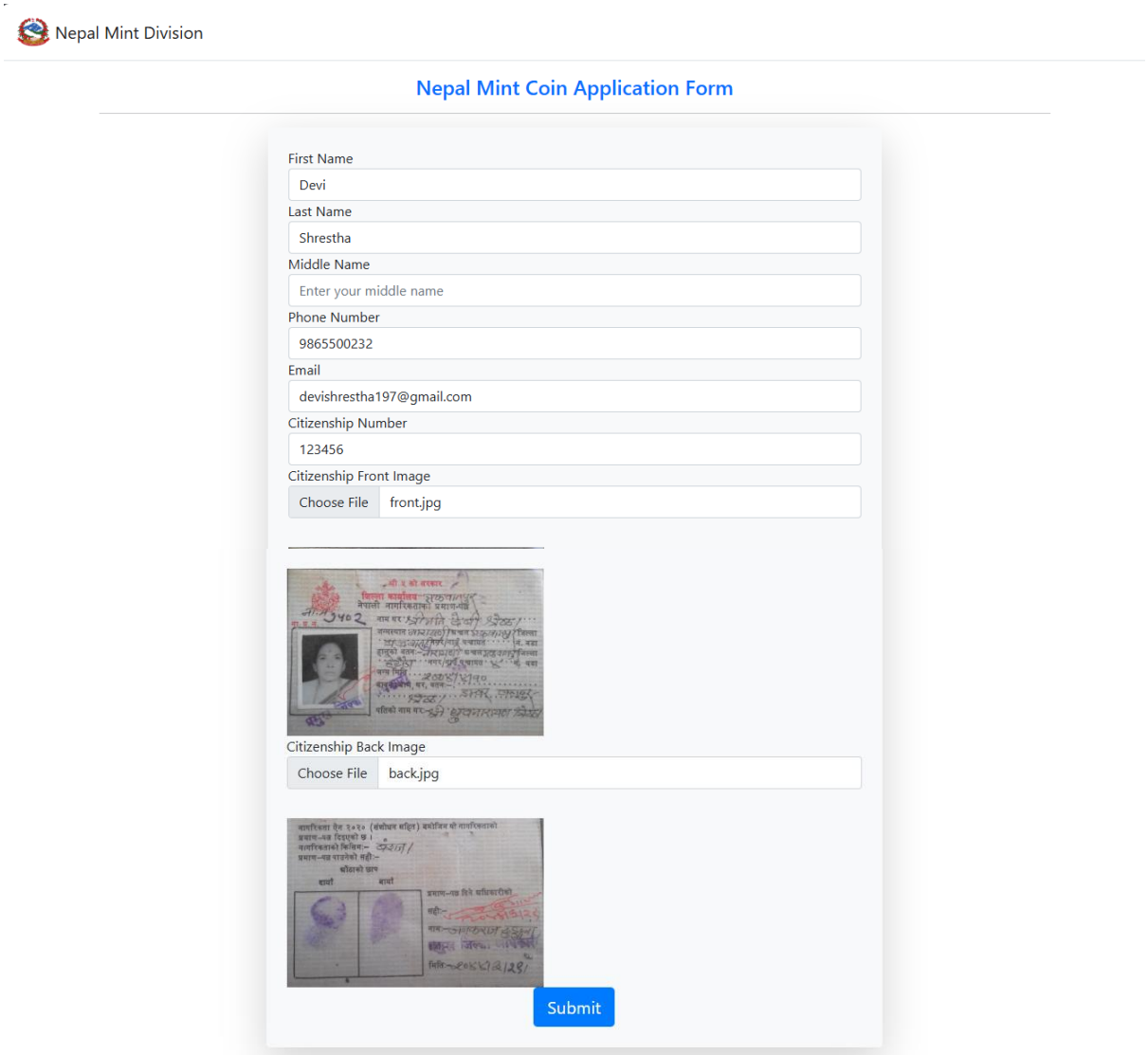
- HTML, CSS, and Bootstrap Documentation: Resources used to design the frontend interface of the website.
  - https://getbootstrap.com/docs/
- W3C Web Accessibility Guidelines (WCAG): Ensuring the website is accessible to all users.
  - https://www.w3.org/WAI/standards-guidelines/wcag/

Security & Best Practices

- ASP.NET Core Security Best Practices: Guidelines to secure the web application from common vulnerabilities.
  - https://learn.microsoft.com/en-us/aspnet/core/security/
- OWASP Top Ten Security Risks: Reference for securing web applications against common security threats.
  - https://owasp.org/www-project-top-ten/

Appendix A: Screenshots





Nepal Mint Division

Limited Purchase Offer: 5 Silver Coins per Type & 1 Gold Coin

SEBON(2500)

Price: Rs. 5500

Weight: 25 gms g

Quantity: 0

Satya Mohan Joshi(1000)

Price: Rs. 2200

Weight: 10 gms g

Quantity: 0

Marwadi Sewa Samiti

Price: Rs. 4360

Weight: 18 gms g

Quantity: 0

Guru Nanak(2500)

Price: Rs. 5500

Weight: 25 gms g

Quantity: 0

Guru Nanak(1000)

Price: Rs. 2200

Weight: 10 gms g

Quantity: 0

NRB Golden Jubilee

Price: Rs. 8400

Weight: 35 gms g

Quantity: 0

Ganesh

Price: Rs. 2200

Weight: 10 gms g

Quantity: 0

Laxmi

Price: Rs. 2200

Weight: 10 gms g

Quantity: 0

Gajalaxmi

Price: Rs. 2200

Weight: 10 gms g

Quantity: 0

Cow Medallion

Price: Rs. 2200

Weight: 10 gms g

Quantity: 2

Cow Medallion

Price: Rs. 5500

Weight: 25 gms g

Quantity: 3

Satya Mohan Joshi(2500)

Price: Rs. 5500

Weight: 25 gms g

Quantity: 5

Gold

Price: Rs. 25550

Weight: 2.5 gms g

Quantity: 1

Confirm

Nepal Mint Division

Your Token Number: 10

Order Confirmation

Coin Name	Quantity	Total Price (Rs.)
Cow Medallion	2	4400
Cow Medallion	3	16500
Satya Mohan Joshi(2500)	5	27500
Gold	1	25550

Grand Total: Rs. 73950

Points to remember:

- Please note your token number and bring original citizenship card when arriving to the central office.
- Reach to the office prior to 10 token numbers.

Nepal Mint Division Login Register

Login Admin

Email

admin@admin.com

Password

••••••••••

Login

© 2025 - Nepal Mint Division - Privacy

Nepal Mint Division Login Register Logout

Manage Admin

id	Name	Email	Password	PhoneNumber	Role	Status	
1	Ishan Shrestha	admin@admin.com	admin1234#	9865500232	Developer	☑	Edit | Details | Delete
2	Admin1	admin1@admin.com	Admin11234#	65455654	Normal	☑	Edit | Details | Delete
3	Admin2	admin2@admin.com	Admin21234#	98654211	Normal	☑	Edit | Details | Delete

© 2025 - Nepal Mint Division - Privacy

Nepal Mint Division Login Register

Register Admin

Name

Email

Password

Confirm Password

PhoneNumber

Register

Nepal Mint Division Logout

Citizen Database

Create

Id	FirstName	LastName	MiddleName	Phone	Email	CitizenshipNo	CitizenF	CitizenB	Status	
8	Ishan	Shrestha		9865500232	ishanrlshrestha@gmail.com	21-4547-12			☑	Edit | Details | Delete
9	Ishan	Shrestha		9865500232	ishanrlshrestha@gmail.com	2521			☑	Edit | Details | Delete
10	Devi	Shrestha		9865500232	devishrestha197@gmail.com	123456			☐	Edit | Details | Delete

Nepal Mint Division

Edit

Citizen

FirstName

Devi

LastName

Shrestha

MiddleName

Phone

9865500232

Email

devishrestha197@gmail.com

CitizenshipNo

123456

CitizenF

5bea353a-1071-4a40-9ec2-2eb25999c21f.jpg

CitizenB

e3b67e43-a6bb-4d03-bf70-156099ea974d.jpg

☑ Status

Save

Back to List