# new logo.jpg

# New Generation University College

**DEPARTMENT OF COMPUTER SCIENCE**

**TITLE:-Web-based Nekemte City Residential Id Card Management System**

Web-based Nekemte City Residential ID Card Management System" has been prepared and submitted by the following students

1. Fuad Getu
2. Samuel Ayana
3. Gelata Darje
4. Kebena Busa
5. Garoma Gonfa

Submitted to department of computer science new generation university college for the requirement of

the Bachelor of Science degree in Computer Science.

Advisor:-Mr..Eshetu Tasfa

Nekemte,Ethiopia

# Acknowledgements

I would like to express my gratitude to our advisor for his guidance, support and his continuous enthusiasm

And encouragement throughout the project. We are also very grate full and extend our sincere thanks to the

Principals and staff members of the department of CS their cooperation by sharing the load that we was

Teaching to make we have time to work on this project and throughout our study.

Finally many thanks to friends, who have helped and given our suggestions, supports and

Corrections throughout the project

**Abstract**

The rapid urbanization of nekemte city in Ethiopia’s oromia region has exposed inefficiencies in it’s manual residential id card management system. Characterized by data inaccuracies bureaucratic delays. And security vulnerabilities.

This project proposes a web-based residential id card management system to digitize the lifecycle of id card issuance, renewal verifications and tracking .by replacing paper base processes with automated workflows, the system aims to enhance transparency, reduce administrative bottlenecks, and improve service delivery for residents and municipal authority

# List of Acronyms

**API**: Application Programming Interface

**CI**: Continuous Integration

**CRUD**: Create, Read, Update, Delete

**CSS**: Cascading Style Sheets

**DB**: Database

**HTML**: Hypertext Markup Language

**HTTP**:HypertextTransferProtocol

**HTTPS**:HypertextTransferProtocolSecure

**ID**: Identification

**IDE**: Integrated Development Environment

**JS**: JavaScript

**JSON**: Java Script Object Notation

**MFA**: Multi-Factor Authentication

**MVC**: Model-View-Controller

**ORM**: Object-Relational Mapping

**PHP**:PHP: Hyper text Preprocessor

**RBAC**: Role-Based Access Control

**SIEM**: Security Information and Event Management

**SQL**: Structured Query Language

**UAT**: User Acceptance Testing

**UI**: User Interface

**WCAG**: Web Content Accessibility Guidelines

[Figure 1UseCase Diagram 13](#_bookmark37)

[Figure2 Sequence Diagram 16](#_bookmark39)

[Figure3 Resident Registration Sequence Diagram 17](#_bookmark40)

[Figure4Resident Login Sequence Diagram 18](#_bookmark41)

[Figure5Moderator Review and Approval Sequence Diagram 19](#_bookmark42)

[Figure6Admin User Management Sequence Diagram 20](#_bookmark43)

[Figure7 Resident ID Card Application Process 22](#_bookmark45)

[Figure8 Moderator ID Card Approval Work flow 23](#_bookmark46)

[Figure9 Admin User Account Management Process 24](#_bookmark47)

[Figure10 Analysis Level Class Diagram 25](file://localhost/C:/Users/Sadik%20AbduRahman/Desktop/CRICMS.docx%23_Toc163405669)

[Figure11Deployment Diagram 27](#_bookmark50)

[Figure12 Class Type Architecture 33](#_bookmark62)

[Figure13 State Chart for Admin Dashboard 35](#_bookmark66)

[Figure14State Chart for Moderator Actions 36](#_bookmark68)

[Figure15 State Chart for Resident Activities 37](#_bookmark70)

[Figure16 Collaboration Diagram for User's Login 38](#_bookmark73)

[Figure17CollaborationDiagram for ID Card Approval Process 39](#_bookmark75)

[Figure18 Component Modeling 40](#_bookmark77)

[Figure19Deployment Diagram 41](#_bookmark79)

[Figure20 Persistence Modeling 42](#_bookmark81)

Contents

[APPROVALSHEET i](#_bookmark0)

[List of Acronyms ii](#_bookmark1)

[CHAPTER1.Introduction 1](#_bookmark2)

* 1. [Background of the organization 1](#_bookmark6)
     1. [Vision of the Project 1](#_bookmark4)
     2. [Mission of the Project 1](#_bookmark5)
  2. [Statement of Problem 2](#_bookmark8)
  3. [Objectives 2](#_bookmark9)
     1. [General Objective 2](#_bookmark10)
     2. [Specific Objectives 2](#_bookmark11)
  4. [Feasibility Analysis 3](#_bookmark12)
  5. [Scope and Significance of the Project 3](#_bookmark13)
  6. [Target and Beneficiaries of the System 4](#_bookmark14)
  7. [Methodology 4](#_bookmark15)
     1. [Data Source 4](#_bookmark16)
     2. [Fact Finding Techniques 4](#_bookmark17)
     3. [Systems Analysis and Design 5](#_bookmark18)
     4. [Development Tools 5](#_bookmark19)
     5. [Testing Procedures 5](#_bookmark20)
     6. [Implementation 5](#_bookmark21)
     7. [Limitations of the Project 6](#_bookmark22)
     8. [Risks, Assumptions ,and Constraints 6](#_bookmark23)
  8. [Time Plan 7](#_bookmark24)
  9. [Budget Plan 7](#_bookmark25)

[Chapter2.ExistingSystem 8](#_bookmark26)

* 1. [Introduction of Existing System 8](#_bookmark27)
  2. [Requirements of the Proposed System 8](#_bookmark28)
     1. [Functional Requirements 8](#_bookmark29)
     2. [Non-functional Requirements 9](#_bookmark30)
     3. [User Interface 10](#_bookmark31)
     4. [Backup and Recovery 10](#_bookmark32)
     5. [Documentation 11](#_bookmark33)

[Chapter3.SystemAnalysis 12](#_bookmark34)

* 1. [System Requirement Specifications(SRS) 12](#_bookmark35)
     1. [Use Case Diagrams 12](#_bookmark36)
     2. [Sequence Diagrams 16](#_bookmark38)
     3. [Activity Diagrams 21](#_bookmark44)
     4. [Analysis Level Class Diagram 25](#_bookmark48)
  2. [Deployment Diagram 27](#_bookmark49)

[Deployment Nodes 28](#_bookmark51)

[Artifacts 28](#_bookmark52)

[Dependencies: 28](#_bookmark53)

[Database: 28](#_bookmark54)

[Deployment 28](#_bookmark55)

* 1. [User Interface Prototyping 29](#_bookmark56)
  2. [Supplementary Specifications 29](#_bookmark57)
  3. [Waterfall Model Diagram 30](#_bookmark58)

[Chapter4.SystemDesign 31](#_bookmark59)

* 1. [Introduction 31](#_bookmark60)
  2. [Class Type Architecture 32](#_bookmark61)
  3. [Class Modeling 34](#_bookmark63)
  4. [State Chart Diagrams 35](#_bookmark64)
     1. [State Chart for Admin Dashboard 35](#_bookmark65)
     2. [State Chart for Moderator Actions 36](#_bookmark67)
     3. [State Chart for Resident Activities 37](#_bookmark69)
  5. [Collaboration Modeling 38](#_bookmark71)
     1. [Collaboration Diagram for User's Login 38](#_bookmark72)
     2. [Collaboration Diagram for ID Card Approval Process 39](#_bookmark74)
  6. [Component Modeling 40](#_bookmark76)
  7. [Deployment Modeling 41](#_bookmark78)
  8. [Persistence Modeling 42](#_bookmark80)
  9. [User Interface Design 43](#_bookmark82)

[Chapter5.ImplementationandTesting 44](#_bookmark83)

* 1. [Introduction 44](#_bookmark84)
  2. [Objectives 44](#_bookmark85)
  3. [Constraints on Implementation 44](#_bookmark86)
  4. [Testing 45](#_bookmark87)
     1. [Testing by Requirements 45](#_bookmark88)
     2. [Unit Testing 46](#_bookmark89)
     3. [Integration Testing 46](#_bookmark90)
  5. [Error Handling 47](#_bookmark91)
  6. [Sample Pages and Source Codes 48](#_bookmark92)

[Chapter6.ConclusionandRecommendation 51](#_bookmark93)

* 1. [Conclusion 51](#_bookmark94)
  2. [Recommendation 51](#_bookmark95)
  3. [Installation 52](#_bookmark96)

# CHAPTER 1

# Introduction organization

## Background of the city of Nekemte, situated in the Oromia region of Ethiopia, has been facing challenges in the management to Resident ID card system .The existing manual and paper-based processes have led to issues such as data inaccuracy, lengthy application procedures, and difficulty in maintaining reliable resident records. To address these problems, the city authorities have identified the need for a comprehensive digital solution that can revolutionize the ID card management system.

### 1.1.1. Vision of the Project

ThevisionoftheWeb-basedNekemteCityResidentIDCardManagementSystemistoprovide a centralized and efficient platform for managing the identification cards of Nekemte city residents. By digitizing the ID card management process, the system aims to improve the accessibility, accuracy, and security of resident information, ultimately enhancing the overall administrative and service delivery capabilities of the city.

### 1.1.2. Mission of the Project

The mission of the Web-based Nekemte City Resident ID Card Management System is to develop a user-friendly and secure web-based application that streamlines the process of ID card registration, issuance, and management. The system will enable residents to easily apply for and access their ID cards , while also providing administrative tools for efficient oversight and control by the city authorities

## 1.2. Problem statement

The existing manual ID card management system in Nekemte city faces several challenges, including:

* Inefficient record-keeping and data management
* Lengthyandinconvenientapplicationandissuanceprocessesforresidents
* Lack of centralized over sight and control over the ID card system
* Difficulties in maintaining up-to-date and accurate resident information
* In creased risk of errors ,fraud, and unauthorized access to resident data

## 1.3. Objectives

## 1.3.1.General Objective

The primary objective of the Web-based Nekemte City Resident ID Card Management System is to develop a comprehensive digital solution that streamlines the ID card management process, enhances data integrity, and improves the overall efficiency of the city's resident identification system.

### 1.3.2.Specific Objectives

* + - 1. Digitize the resident ID card application and issuance process to improve accessibility and convenience for the city's residents.
      2. Implement a centralized database system to store and manager resident information, ensuring accurate and up-to-date records.
      3. Provideadministrativetoolsanddashboardsforcityauthoritiestoefficientlyoverseeand manage the ID card system.
      4. Enhance the security of the resident ID card system by implementing robust access controls and data protection measures.
      5. Integrate the ID card system with other city services and e-government initiatives to facilitate seamless access to public services for Nekemte residents.

## 1.4.Feasibility Analysis

A comprehensive feasibility analysis was conducted to assess the technical, operational, and financial viability of the Web-based Nekemte City Resident ID Card Management System project. The analysis considered the following aspects:

1. **Technical Feasibility**: The project team evaluated the available technology, hardware, and software resources to ensure the technical feasibility of the system. The system architecture, development tools ,and infrastructure requirements were carefully examined to ensure the project's technical viability.
2. **Operational Feasibility**: The team assessed the operational requirements, such as user training, system maintenance, and administrative support, to ensure the smooth implementation and ongoing operation of the ID card management system.
3. **Financial Feasibility**: The financial analysis considered the initial investment, operational costs, and potential cost savings or revenue generation through the implementationofthedigitalIDcardsystem.Theprojectwasfoundtobefinancially viable and cost-effective in the long run.

The feasibility analysis concluded that the Web-based Nekemte City Resident ID Card ManagementSystemisaviableandsustainableprojectthatcansignificantlyimprovethecity's resident identification and service delivery capabilities.

## 1.5. Scope and Significance of the Project

The scope of the Web-based Nekemte City Resident ID Card Management System encompasses the following key components:

1. Resident ID card registration and application management
2. Resident ID card issuance and distribution
3. Centralized data base management of resident information
4. Administrative tools and dashboards for city authorities
5. Integration with other city services and e-government initiatives

The successful implementation of this project will have significant impact on the city of Nekemte and its residents, including:

* + Improved accessibility and convenience for residents in applying for and accessing their ID cards
  + Enhanced data integrity and reliability through the use of a centralized digital system
  + IncreasedefficiencyandstreamlinedprocessesforcityauthoritiesinmanagingtheID card system
  + Strengthened security and data protection measures to safe guard resident information
  + Facilitatedintegrationwithothercityservicesande-governmentinitiativesforseamless service delivery

## 1.6. Target and Beneficiaries of the System

The primary target and beneficiaries of the Web-based Nekemte City Resident ID Card Management System are:

1. **Nekemte City Residents**: The system will directly benefit the residents of Nekemte by providing them with a user-friendly and accessible platform to apply for, receive, and manage their ID cards.
2. **Nekemte City Authorities**: The city authorities, including the mayor's office, administrative departments, and municipal services, will benefit from the improved efficiency, data management, and oversight capabilities offered by the digital ID card system.
3. **Other City Stakeholders**: The integration of the ID card system with other city services and e-government initiatives will indirectly benefit various stakeholders, such as local businesses, service providers, and community organizations, by facilitating seamless access to public services.

## 1.7.Methodology

### 1.7.1 Data Source

The project team gathered data from various sources, including:

* Interviews with city of facials and residents
* Review of existing ID card management processes and records
* Analysis of similar digital ID card systems implemented in other cities
* Consultation with industry experts and technology providers

### 1.7.2Fact Finding Techniques

Theteamemployedthefollowingfact-findingtechniquestogatherthenecessaryinformationfor the project:

* Interviews: Conducted in-depth interviews with city authorities, resident representatives, and other stakeholders to understand the current challenges and requirements.
* Observation: Observed the existing ID card management processes and procedures to identify pain points and opportunities for improvement.
* Document Review: Reviewed existing documents, policies, and regulations related to the city's ID card management system.
* Brainstorming: Organized collaborative brains to riming sessions to generate ideas and solutions for the proposed system.

### 1.8. Systems Analysis and Design

Theprojectteamfollowedastructuredsystemsanalysisanddesignmethodology,which included the following key steps:

* Requirement Gathering and Analysis
* Use Case Modeling
* System Architecture Design
* User Inter face Prototyping
* Data base Design
* Application Development

### 1.9. Development Tools

TheWeb-basedNekemteCityResidentIDCardManagementSystemwasdevelopedusingthe following technology stack:

* Front-end:HTML5,CSS3,JavaScript,Bootstrap
* Back-end: PHP, My SQL
* Frame work: Larval
* Development Tools: Visual Studio Code, GitHub
* Project Management: Google Docs

### 1.10. Testing Procedures

The project team implemented a comprehensive testing strategy, including:

* Unit Testing: Individual components and modules were tested to en sure their functionality and correctness.
* Integration Testing: The system was tested as a whole, verifying the integration and interaction between different components.
* User Acceptance Testing: The system was tested by city officials and resident representativestoensureitmeetsthespecifiedrequirementsanduserexpectations.
* Security Testing: The system was tested for vulnerabilities and security threats to ensure the protection of resident data and system integrity.

### 1.11. Implementation

The implementation of the Web-based Nekemte City Resident ID Card Management System involved the following key steps:

* + - 1. Deployment of the system infrastructure, including the web server, data base server, and network configuration.
      2. Migrationofexistingresidentdatafromthepreviousmanualsystemtothenewdigital platform.
      3. Training and on boarding of city officials, moderators, and residents on the use of the new system.
      4. Gradual rollout and phased deployment of the system to ensure smooth transition.
      5. Continuous monitoring, maintenance, and optimization of the system based on user feedback and operational requirements.

### 1.12. Limitations of the Project

TheWeb-basedNekemteCityResidentIDCardManagementSystemprojecthasthefollowing limitations:

* Dependenceonreliableinternetconnectivityandinfrastructurefortheweb-based application
* Potentialchallengesintheinitialdatamigrationfromthepreviousmanualsystem
* Requirementforongoingusertrainingandtechnicalsupporttoensureeffectivesystem utilization
* Limitedintegrationwithothercityservicesandsystemsbeyondtheinitialscopeofthe project

### 1.13. Risks, Assumptions, and Constraints

The project team identified and addressed the following risks, assumptions, and constraints:

#### 1.13.1. Risks:

* Resistance to change from city officials and residents
* Technological failures or system downtime
* Data security and privacy breaches
* Budget and resource constraints

#### 1’14.Assumptions:

* AvailabilityofreliableinternetconnectivityandinfrastructureinNekemte
* Commitmentandsupportfromcityauthoritiesforthesuccessfulimplementationofthe system
* Willingness of residents toad opt and utilize the new digital ID card system

#### 1.15.Constraints:

* Limitedbudgetandfinancialresourcesfortheinitialimplementationandongoing maintenance
* Time constraints for the project development and rollout
* Existing legacy systems and integration challenges

The project team developed mitigation strategies and contingency plans to address the asterisks, assumptions, and constraints throughout the project lifecycle.

* 1. **1.16.Time Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Duration | Start Date | End Date |
| **Proposal** | 2 weeks | April 03 | April 17 |
| **System requirement analysis** | 4 weeks | May 05 | May 14 |
| **System & object design** | 6 weeks | June 18 | June 26 |
| **Implementation** | 8 weeks | May 3 | May 20 |
| **Testing** | 4 weeks | June 9 | June 30 |
| **Documentation project submission** | 6 weeks | July 20 | July 30 |
| **Maintenance and Support** | Ongoing |  | - |

* 1. **1.17.Budget Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Name | Quantity | Unit price in birr | Total |
| 1 | Purchase of flash disk  (8GB) | 1 | 500.00 | 500.00 birr |
| 2 | Purchase of CD RW | 1 | 200.00 | 200.00 birr |
| 3 | desk top  personal pc | 1  1 | 27.000  42.00 | 27.00birr  42.00birr |
| 4 | Miscellaneous expense | - | - | 200.00birr |
| 6 | Total estimate cost | - | - | 69.900birr |

### 1.9.1 Team configuration

In our project, we have five members and each of us have specified work. The following are the types of tasks and as well as the responsibility each of us have been assigned for.

|  |  |  |  |
| --- | --- | --- | --- |
| Project title | Web based nekemte city residential id card management system | | |
|  | Name | Id No | Responsibility |
| Samuel ayana | 35243 | Manager |
| Garoma gonfa | 35 | Analysis the system |
| Geleta daraja | 35200 | Designer |
| Fuad getu | 35193 | Programmer |
| Kebena busa | 35217 | Tester |

# Chapter2.

## 2.1. Introduction of Existing System

The existing ID card management system in Nekemte city is primarily a manual and paper-based process. Residents are required to visit the city administration office in person to apply for their ID cards, submit supporting documents, and wait for the cards to be processed and issued. The process involves the following steps:

1. Resident submits paper-based application form, along with required documentation (such as photos, proof of identity, and residence).
2. City of facials manually review the application and supporting documents.
3. If the application is approved the resident's information is recorded in a physical registry book.
4. The ID card is then printed and physically handed over to the resident.
5. ResidentinformationandIDcarddetailsarestoredinphysicalfilesandcabinets,making it challenging to maintain accurate and up-to-date records.

This existing system faces several challenges including:

* + Lengthy and inconvenient application and issuance processes
  + In efficient data management and record-keeping
  + Lack of centralized over sight and control
  + Increased risk of errors ,fraud ,and unauthorized access store indent data

## 

## 2.2. Requirements of the Proposed System

### 2.2.1 Functional Requirements

The proposed Web-based Nekemte City Resident ID Card Management System should provide the following key functional requirements:

#### Resident Registration and ID card Application:

* + - * + Residents should be able to register and apply for their ID cards online through a user-friendly web interface.
        + Thesystemshouldallowresidentstosubmitrequireddocumentationandpersonal information digitally.
        + The system should provide real-time status up dates on the application process.

#### ID card Approval and Issuance:

* + - * + Cityadministratorsandmoderatorsshouldbeabletoreview,approve,andreject resident ID card applications.

The system should facilitate the digital issuance and distribution of

ID cards to approved residents

The system should generate and store unique ID card numbers for each resident

.

#### Resident Information Management:

* + - * + The system should maintain a centralized data base of resident information, including personal details, ID card data, and application history.
        + The system should provide tools for city authorities to manage and update resident records as needed.
        + The system should allow residents to view and update their personal information.

#### Reporting and Analytics:

* + - * + The system should generate comprehensive ere ports on ID card applications, issuance, and resident demographics.
        + Thesystemshouldprovideanalyticaltoolsanddashboardstohelpcityauthorities monitor and manage the ID card system effectively.

#### Integration with Other City Services:

* + - * + The system should be designed to integrate with other city services and e- government initiatives, enabling residents to a access range of public services using their ID cards.
        + Thesystemshouldprovidesecureauthenticationandauthorizationmechanisms for integration with other city systems.

### 2.2.2 Non-functional Requirements

The proposed Web-based Nekemte City Resident ID Card Management System should also meet the following non-functional requirements:

#### Usability:

* + - * + The system should have a user-friendly and intuitive interface ,catering to the needs of both residents and city authorities.
        + The system should provide clear and concise instructions and guidance to users.

#### Security and Data Privacy:

* + - * + Thesystemshouldimplementrobustsecuritymeasurestoprotectresidentdata and prevent unauthorized access.
        + The system should comply with relevant data protection regulations and guidelines.

#### Scalability and Performance:

* + - * + The system should be designed to handle a growing umber of resident registrations and ID card applications.
        + Thesystemshouldprovidefastandresponsiveperformance,evenwithalarge user base.

#### Availability and Reliability:

* + - * + The system should be available and accessible o users at all times, with minimal downtime.
        + The system should have mechanism sin place to en sure data backup, recovery, and business continuity.

#### Maintainability and Flexibility:

* + - * + The system should be easy t maintain, with clear documentation and modular design.

The system should be adaptable to future changes in requirements or technology

### 2.3. User Interface

The proposed Web-based Nekemte City Resident ID Card Management System should provide a user-friendly and visually appealing interface with the following key features:

* + - 1. **Responsive and Mobile-friendly Design**: The system should be designed to be accessible and optimized for use on various devices, including desktops, tablets ,and smart phones.
      2. **Clear and Intuitive Navigation**: The user interface should have a well-structured navigation system, making it easy for users to access and perform their desired actions.
      3. **Consistent Branding and Aesthetics**: The system should incorporate the city' branding, including colors, logos, and visual elements, to create a cohesive and recognizable identity.
      4. **Accessibility and Inclusivity**: The interface should be designed with accessibility guideline sin mind, ensuring that he system can be used by individuals with diverse needs and abilities.
      5. **Interactive and Engaging Elements**: The interface should in corporate interactive elements, such as forms, dashboards, and data visualizations, to enhance the user experience and facilitate efficient task completion.
      6. **Seamless User Workflows**: The interface should be designed to guide users through the various functionalities of the system, ensuring a smooth and streamlined experience.
      7. **Multilingual Support**: The system should provide the ability to switch between multiple languages, catering to the diverse linguistic needs of Nekemte residents.

### 

### 2.4.Backup and Recovery

The proposed system should implement robust backup and recovery mechanisms to ensure the protection and availability of resident data. This includes:

* + - 1. **Regular Data Backups**: The system should perform regular, automated backups of the entire database and files to ensure data integrity and resilience.
      2. **Secure Off –site Storage**: Backup data should be stored securely in an off-site location, such as a cloud-based storage service, to mitigate the risk of local disasters or system failures.
      3. **Disaster Recovery Plan**: The system should have a well-documented disaster recovery planthatoutlinestheproceduresandresponsibilitiesforrestoringthesystemanddatain the event of a major incident.
      4. **Periodic Recovery Testing**: The backup and recovery processes should be periodically tested to ensure the reliability and effectiveness of the system's disaster recovery capabilities

# Chapter3.SystemAnalysis

## 3.1. System Requirement Specifications (SRS)

### 3.1.1. Use Case Diagrams

The Web-based Nekemte City Resident ID Card Management System has the following key use case diagrams:

#### Use Case Diagram for Resident Interaction

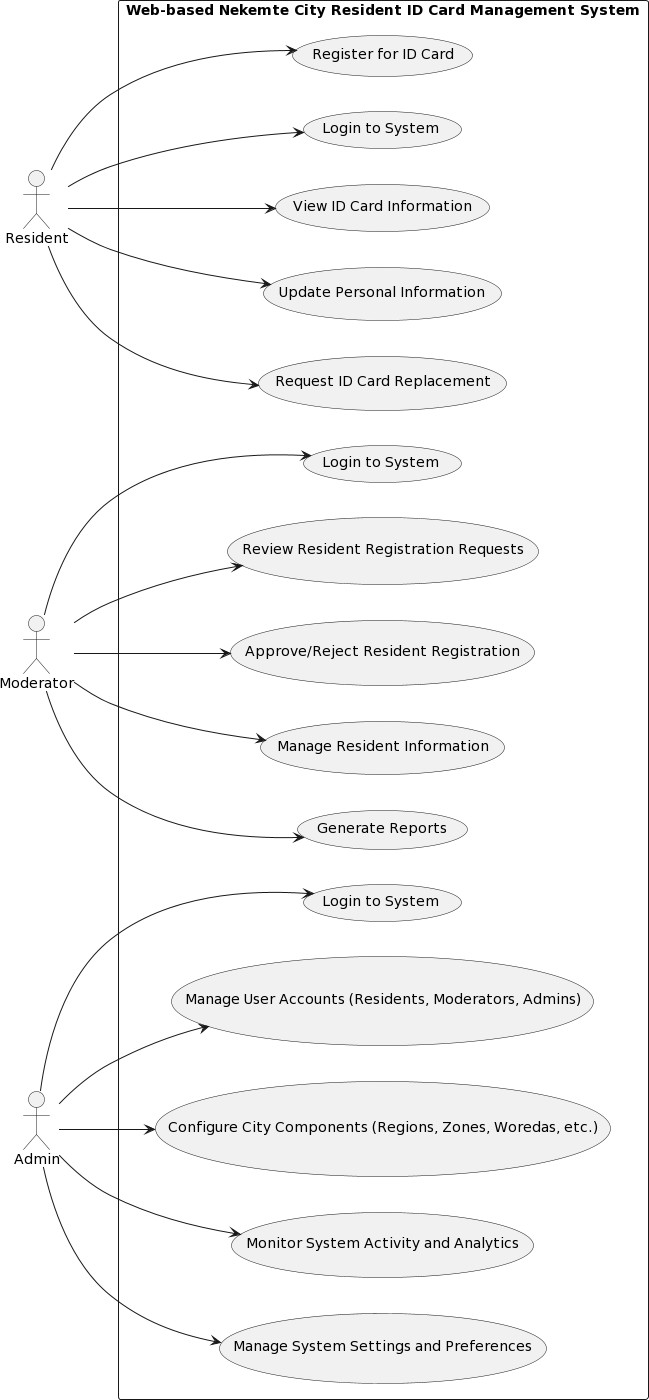
* + Register for ID Card
  + Login to System
  + View ID Card Information
  + Update Personal Information
  + Request ID Card Replacement

#### Use Case Diagram for Moderator Interaction

* + Login to System
  + Review Resident Registration Requests
  + Approve/Reject Resident Registration
  + Manage Resident Information
  + Generate Reports

#### Use Case Diagram for Admin Interaction

* + Login to System
  + Manage User Accounts(Residents ,Moderators A dmins)
  + Configure City Components(Regions, Zones Woredas ,etc.)
  + Monitor System Activity and Analytics
  + Manage System Settings and Preferences



*Figure1UseCaseDiagram*

#### 3.1.1.1.Use Case Documentation

The detailed use case documentation for the key use cases is provided below:

#### 3.1.2. Use Case: Register for ID Card

* + - * + Actor: Resident
        + Description: A resident registers for a new ID card by providing the personal information and uploading required documents.
        + Preconditions: The resident must have a valid email address and access to the internet.
        + Main Flow:

The resident navigates to the "Register for ID Card "page.

The resident enters their personal information, including name, date of birth, address, and contact details.

The resident up loads the required documents, such as a photo ,proof of identity, and proof of residence.

The resident reviews the information and submits the registration form.

The system generates a unique ID card number and displays a confirmation message.

* + - * + Alternate Flows:

If the resident provides in valid or incomplete information, the system displays an error message and prompts the resident to correct the issues.

If the document upload fails, the system displays an error message and allows the resident to try uploading the documents again.

* + - * + Post-conditions:Theresident'sregistrationrequestissavedinthesystem,andthe application is forwarded to the moderator for review and approval.

#### 3.1.2.Use Case: Approve Resident Registration

* + - * + Actor: Moderator
        + Description: A moderator reviews and approves or rejects a resident's ID card registration request.
        + Preconditions: The moderator has logged in to the system and has the necessary permissions to manage resident registrations.
        + Main Flow:

The moderator navigates to the" Manage Resident Registrations "page.

Themoderatorreviewstheresident'ssubmittedinformationandsupporting documents.

The moderator either approves or rejects the registration request.

If approved, the system generates an ID card number and updates there

indent’s status.

If rejected, the moderate or provides areas on for the rejection, and the system notifies the resident.

* + - * + Post-conditions: The resident's registration status is up dated in the system, and the resident is notified of the outcome.

#### 3.1.3.Use Case: Manage User Accounts

* + - * + Actor: Admin
        + Description: An admin manages the user a counts for residents , moderators ,and other adman’s.
        + Preconditions: The admin has logged into the system and has the necessary permissions to manage user accounts.
        + Main Flow:

The admin navigates to the "Manage User Accounts "page.

The admin can view a list of all user accounts.

The admin can create new sera accounts, edit existing ones ,or deactivate/delete accounts as needed.

The admin can assign or modify user roles and permissions.

The admin can reset user passwords or unlock locked accounts.

* + - * + Alternate Flows:

If the admin encounters any issues or discrepancies with the user accounts, they can investigate and take appropriate actions.

The admin can also generate reports on user account activities and management.

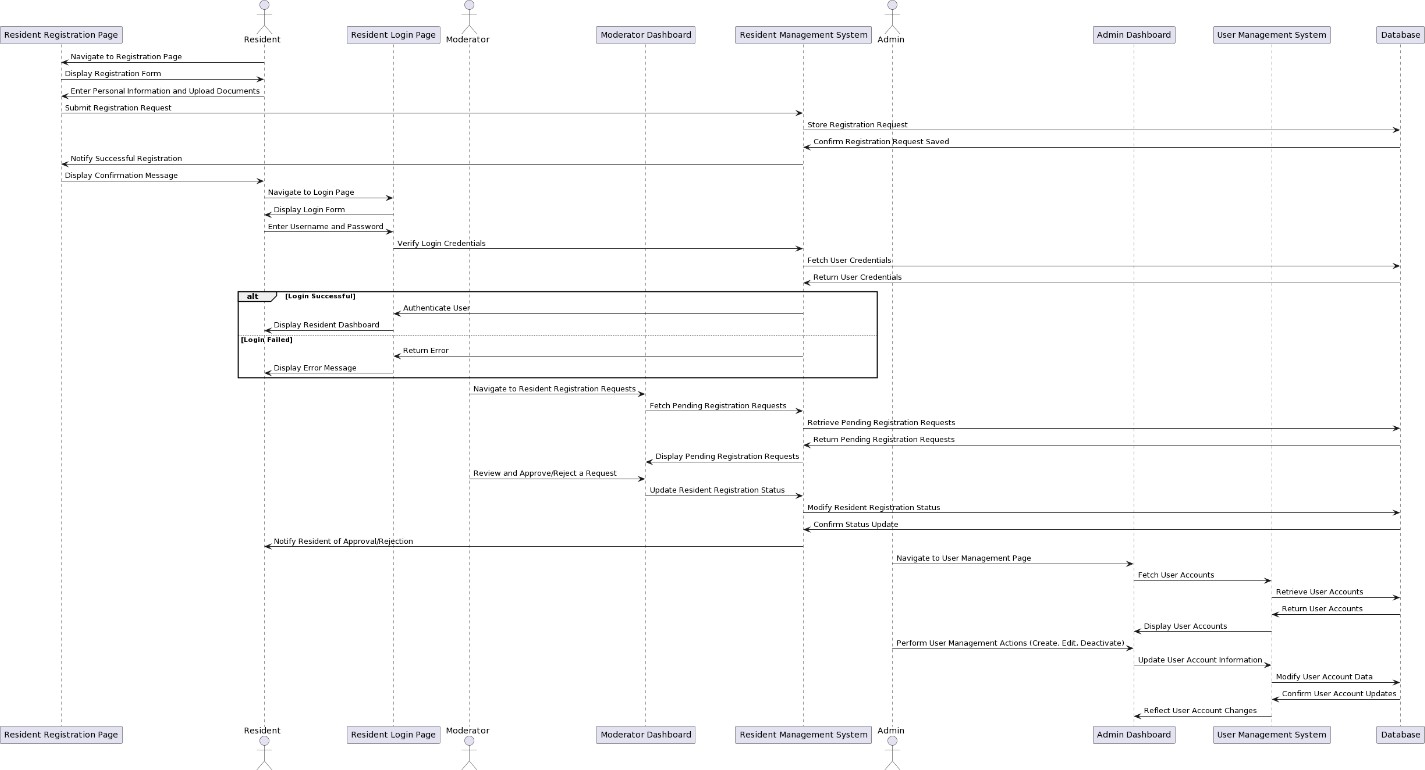
* + - * + Post-conditions: The user a account information is updated in the system, and the changes are reflected for the affected users.

### 3.1.3.1 Sequence Diagrams

The sequence diagrams for the Web-based Nekemte City Resident ID Card Management System include:

1. Resident Registration Sequence Diagram
2. Resident Login Sequence Diagram
3. Moderator Review and Approval Sequence Diagram
4. Admin User Management Sequence Diagram

These sequence diagrams illustrate the interactions between the various actors (residents, moderators,admins)andthesystemcomponents,depictingtheflowofactionsandresponses during the execution of the key use cases.

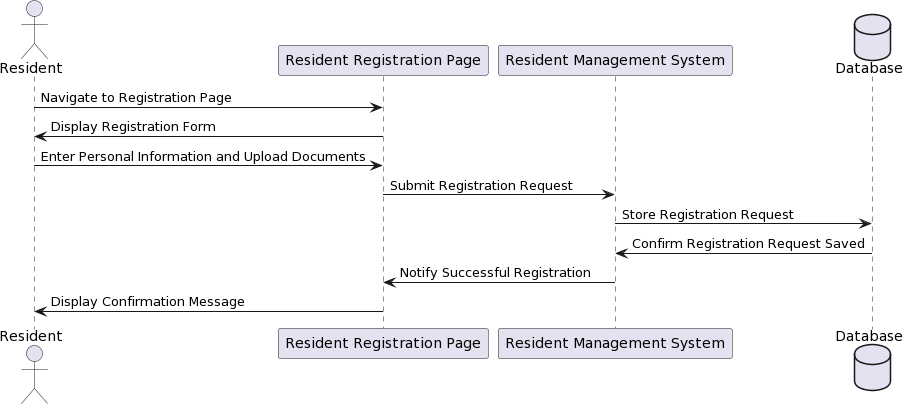


*Figure2SequenceDiagram*

HerearethedetailedsequencediagramsforthekeyusecasesoftheWeb-basedNekemteCity Resident ID Card Management System:

#### Resident Registration Sequence Diagram:

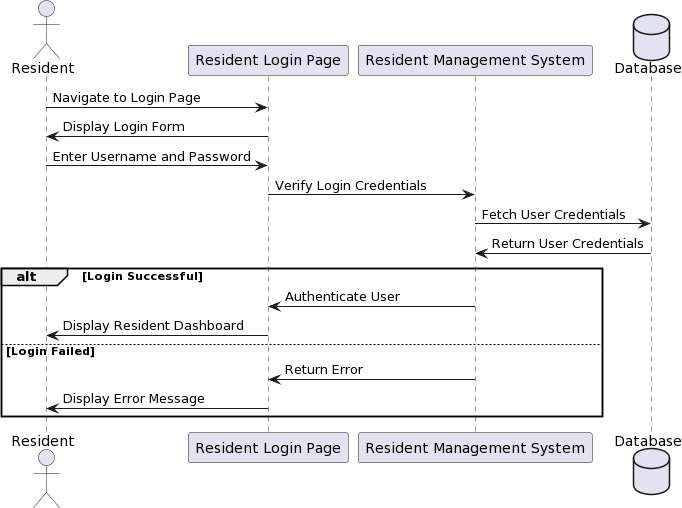
* The resident navigates to the registration page and is presented with are gist ration form.
* The resident enters their personal information and uploads the required documents.
* TheresidentregistrationpagesendstheregistrationrequesttotheResidentManagement System.
* The Resident Management System stores the registration request in the database.
* The data base confirms the successful storage of the registration request.
* TheResidentManagementSystemnotifiestheregistrationpageaboutthesuccessful registration.
* The registration page displays a confirmation message to the resident.



*Figure3ResidentRegistrationSequenceDiagram*

#### Resident Login Sequence Diagram:

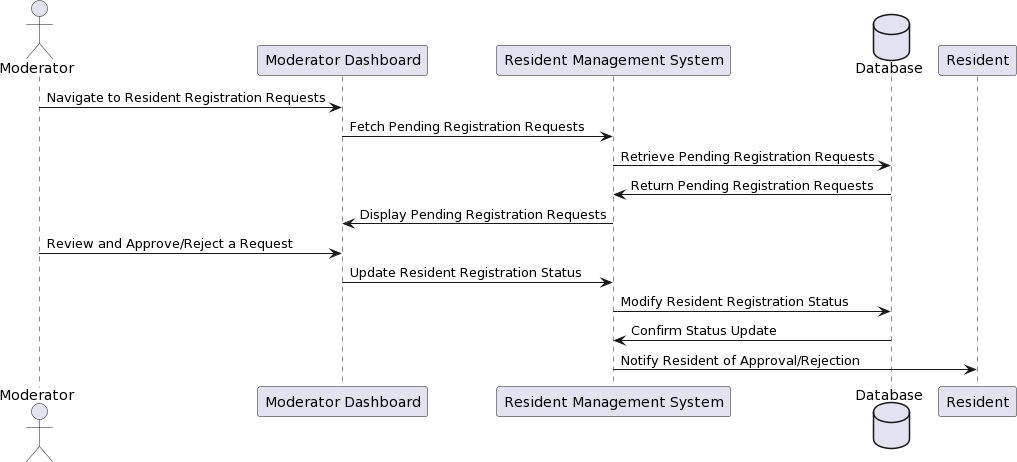
* The resident navigates to the login page and is presented with a login form.
* The resident enters their username and password.
* TheloginpagesendsthelogincredentialstotheResidentManagementSystemfor verification.
* The Resident Management System fetches the user credentials from the database.
* The data base returns the user credentials to the Resident Management System.
* The Resident Management System authenticates the user and notifies the login page.
* If the login is successful, the login page displays the resident dashboard.
* If the login fails, the login page displays an error message.



*Figure4ResidentLoginSequenceDiagram*

#### Moderator Review and Approval Sequence Diagram:

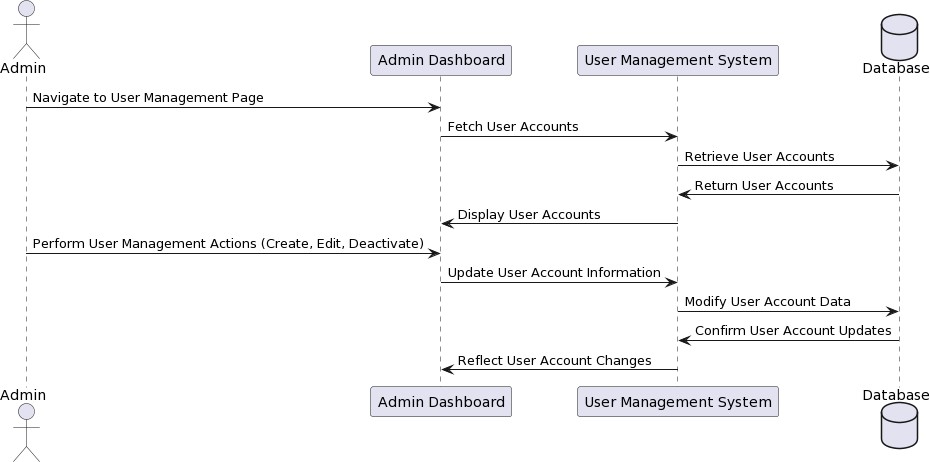
* The moderator navigates to the Resident Registration Requests page on the Moderator Dashboard.
* The Moderator Dashboard requests the Resident Management System to fetch the pending registration requests.
* TheResidentManagementSystemretrievesthependingregistrationrequestsfromthe database.
* ThedatabasereturnsthependingregistrationrequeststotheResidentManagement System.
* TheResidentManagementSystemsendsthependingregistrationrequeststothe Moderator Dashboard.
* The moderator reviews the registration requests and decides to approve or reject them.
* TheModeratorDashboardupdatestheresidentregistrationstatusthroughtheResident Management System.
* TheResidentManagementSystemmodifiestheresidentregistrationstatusinthe database.
* The data base confirms the status update.
* TheResidentManagementSystemnotifiestheresidentabouttheapprovalorrejectionof their registration.



*Figure5ModeratorReviewandApprovalSequenceDiagram*

#### Admin User Management Sequence Diagram:

* The admin navigates to the User Management page on the Admin Dashboard.
* The Admin Dashboard requests the User Management System to fetch the user accounts.
* The User Management System retrieves the user accounts from the database.
* The data base returns the user accounts to the User Management System.
* The User Management Systems ends the user account information to the Admin Dashboard.
* The admin performs various user management actions(create, edit, deactivate)on the Admin Dashboard.
* The Admin Dashboard sends the user account updates to the User Management System.
* The User Management System modifies the user account data in the database.
* The data base confirms the user account updates.
* TheUserManagementSystemreflectstheupdateduseraccountinformationonthe Admin Dashboard.



*Figure6AdminUserManagementSequenceDiagram*

These sequence diagrams provide a detailed overview of the interactions between the actors (residents, moderators,admins)andthevariouscomponentsoftheWeb-basedNekemteCity Resident ID Card Management System during the execution of the key use cases.

### 3.1.3.2. Activity Diagrams

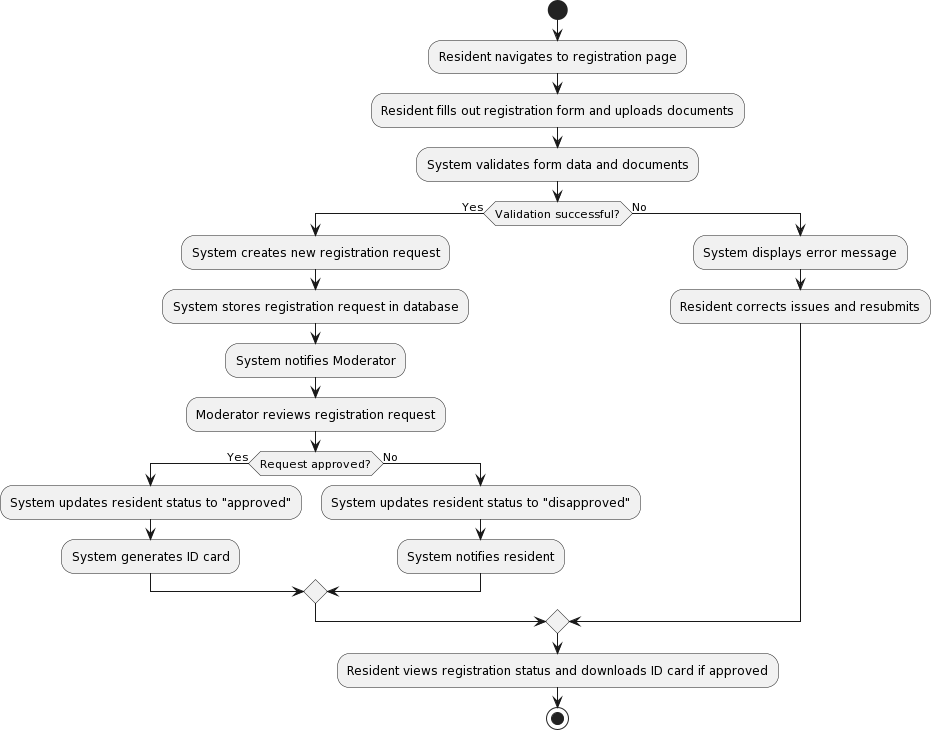
TheactivitydiagramsfortheWeb-basedNekemteCityResidentIDCardManagementSystem include:

1. Resident ID Card Application Process
2. Moderator ID Card Approval Workflow
3. Admin User Account Management Process

Theseactivitydiagramsprovideavisualrepresentationofthestep-by-stepactivitiesanddecision pointsinvolvedintheexecutionofthekeysystemprocesses,highlightingtheflowofcontroland the responsibilities of the different actors.

#### 3.1.3.3.Resident ID Card Application Process:

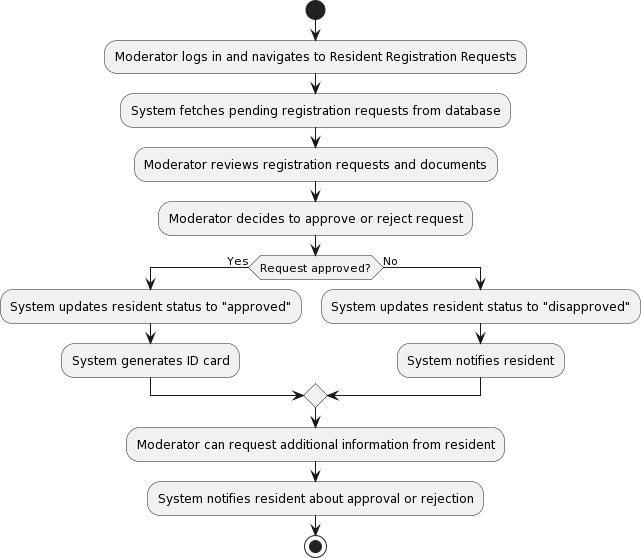
* + - * 1. The resident navigates to the registration page.
        2. Theresidentfillsouttheregistrationformwiththeirpersonalinformationanduploads the required documents.
        3. The system validates the registration form data and the uploaded documents.
        4. If the validation is successful, the system creates new resident registration request and stores it in the database.
        5. If the validation fails, the system displays an error message to the resident, and the resident can correct the issues and resubmit the registration form.
        6. The system not iffiest the moderator about the new resident registration request.
        7. The moderator reviews the registration request and either reapproves or rejects it.
        8. If the request is approved, the system updates the resident's status to" approved" and generates the ID card.
        9. If the request is rejected, the system updates the resident's status to "disapproved "and notifies the resident.
        10. TheresidentcanviewthestatusoftheirregistrationanddownloadtheIDcardif approved.



*Figure7ResidentIDCardApplication Process*

#### 3.2. Moderator ID Card Approval Work flow:

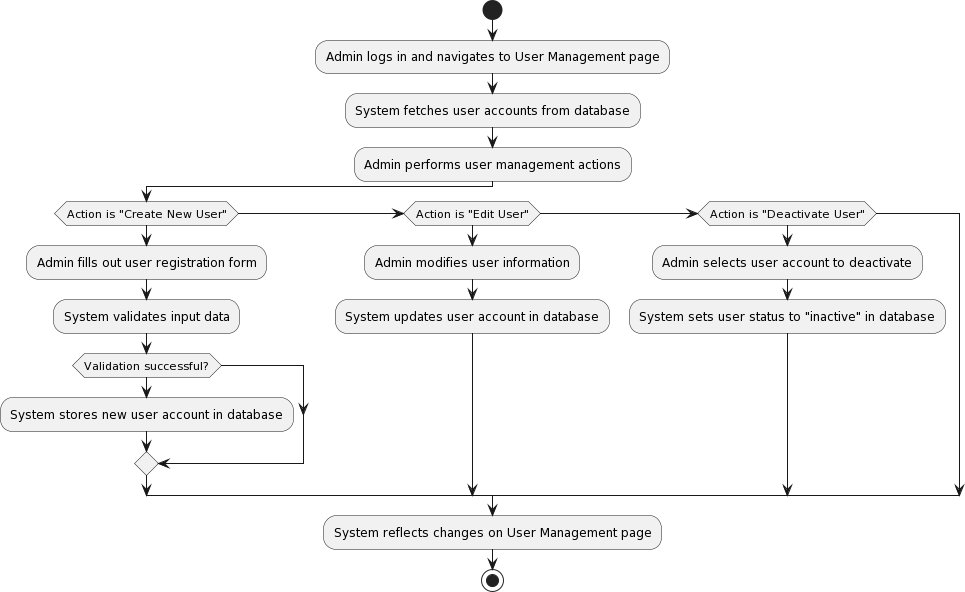
* + - * 1. The moderator logs into the system and navigates to the Resident Registration Requests page.
        2. Thesystemfetchesthelistofpendingresidentregistrationrequestsfromthedatabase.
        3. The moderator reviews each registration request and the associate documents.
        4. The moderator decides to either approve or reject the registration request.
        5. Iftherequestisapproved,thesystemupdatestheresident'sstatusto"approved"and generates the ID card.
        6. If the request is rejected, the system updates the resident's status to "disapproved "and notifies the resident.
        7. Themoderatorcanalsorequestadditionalinformationordocumentsfromtheresidentif needed.
        8. Thesystemnotifiestheresidentabouttheapprovalorrejectionoftheirregistration request.



*Figure8ModeratorIDCardApprovalWorkflow*

#### 3.3. Admin User Account Management Process:

* + - * 1. The admin logs in to the system and navigates to the User Management page.
        2. The system fetches the list of user accounts (admin, moderators, and residents) from the database.
        3. The admin can perform various user management actions, such as creating a new user account, editing an existing user account, or deactivating a user account.
        4. For creating a new user account, the admin fills out the user registration form and the system validates the input data.
        5. If the validation is successful, the system stores the new user account in the database.
        6. For editing an existing user account ,the admin modifies the user in formation and the system updates the account details in the database.
        7. For deactivating a user account, the admin selects the account and the system sets the user's status to "inactive" in the database.
        8. The system reflects the changes made by the admin on the User Management page.



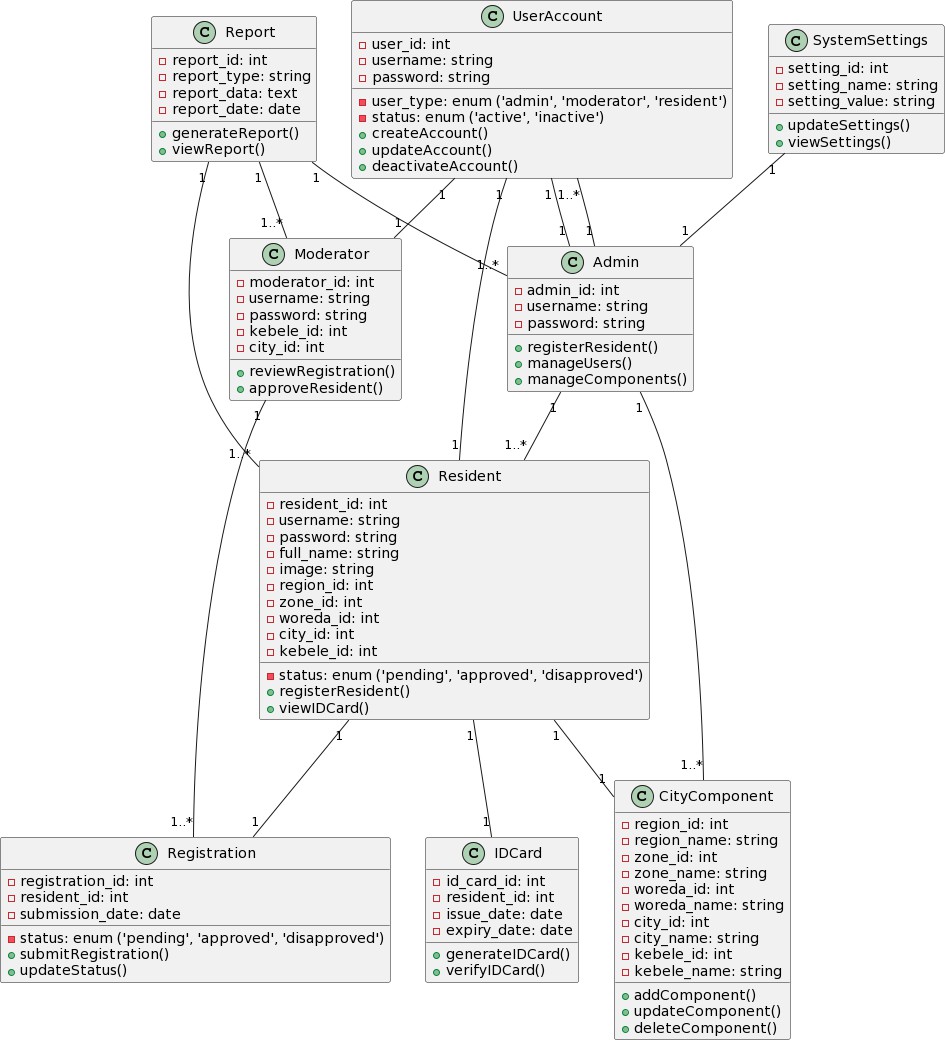
*Figure9AdminUserAccountManagement Process*

These activity diagrams provide a detailed visualization of the key processes involved in the Web-based Nekemte City Resident ID Card Management System, helping to understand the flow of activities and decision points for the resident ID card application, moderator approval, and admin user management functionalities.

### 3. 4.Analysis Level Class Diagram

The analysis-level class diagram for the Web-based Nekemte City Resident ID Card ManagementSystemdepictsthemainclassesandtheirrelationships.Thekeyclassesinclude:

* Resident
* Moderator
* Admin
* ID Card
* Registration
* City Component(Region ,Zone ,Woreda, City, Kebele)
* User Account
* Report
* System Settings



*Figure10AnalysisLevelClassDiagram*

#### Resident Class:

Represents a city resident with attributes like resident\_ id, username, password, full \_name, image,andlocation-relatedIDs(region\_id,zone\_id,woreda\_id,city\_id,kebele\_id).Italsohasa status attribute to track the resident's registration status.

#### Moderator Class:

Represents a moderator with attributes like moderator \_id ,username ,password ,kebele \_id, and city \_id. Moderators can review and approve resident registrations.

#### Admin Class:

Represents an admin with attributes like admin \_id, username, and password. A dmins can performfunctionslikeregisteringresidents,managingusers,andmanagingcitycomponents.

#### ID Card Class:

RepresentsanIDcardissuedtoaresident,withattributeslikeid\_card\_id,resident\_id, issue\_ date, and expiry\_ date.

#### Registration Class:

Represents are sident's registration request, with attributes like registration\_ id, resident \_id, submission \_date, and status.

#### City Component Class:

Represents the various administrative components of the city, including region ,zone ,woreda, city, and kebele , with their respective IDs and names.

#### User Account Class:

Represents a user account, with attributes like user \_id ,username, password, user \_type, and status.

#### Report Class:

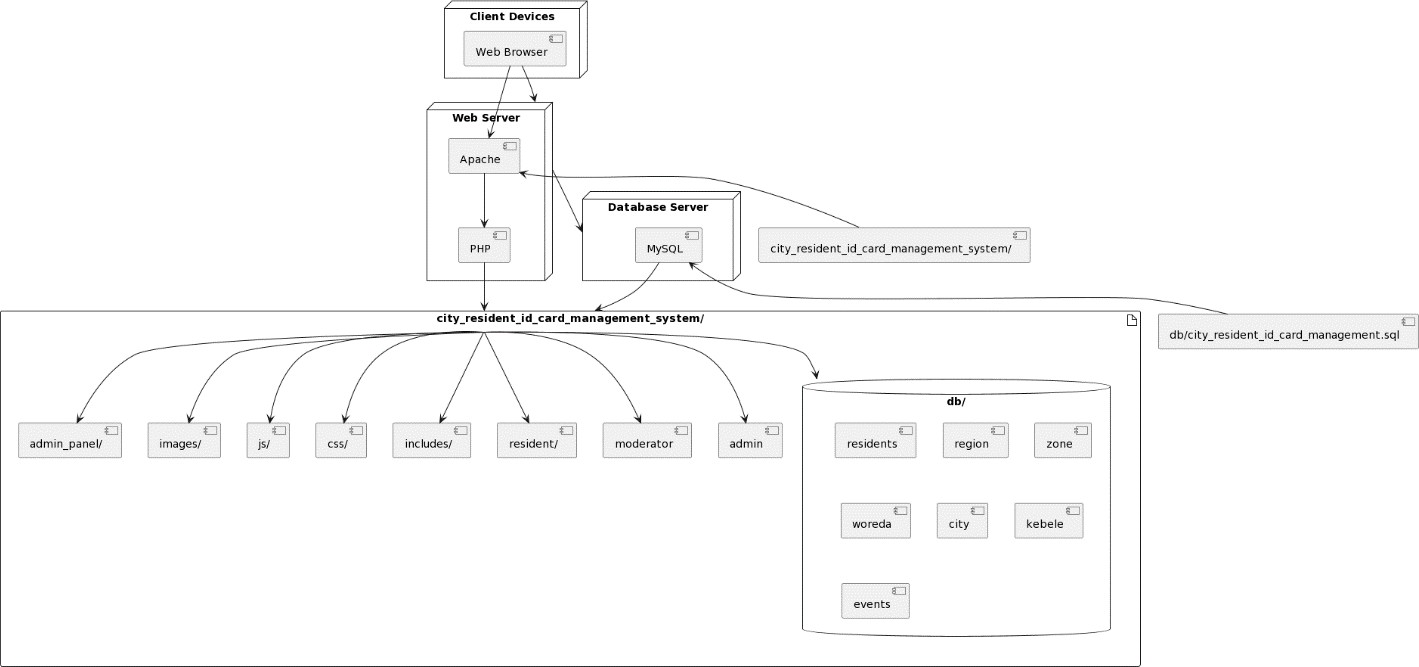
Represents are port generated for residents, moderators, or admits, with attributes like report \_id, report \_type, report\_ data, and report \_date.

#### System Settings Class:

Representsthesystemsettingsthatcanbeupdatedbytheadmin,withattributes like setting\_id, setting \_name, and setting\_ value.

## 3.5. Deployment Diagram

The Deployment Diagram show cases the architecture of the City Resident ID Card Management System, delineating the distribution of components across various nodes and their interdependencies.



*Figure11DeploymentDiagram*

### Deployment Nodes:

The diagram out lines three primary deployment nodes:

* **Webserver**: This node hosts the web application and encompasses components such as Apache and PHP.
* **Data base Server**: Represents the server housing My SQL ,the data base management system utilized by the system.
* **Client Devices**: Devices used by end-users to access the system ,typically web browsers.

### Artifacts:

Components of the project are illustrated as artifacts within the "city\_resident\_id\_card\_management\_system/"directory,includingadmin/,moderator/,resident/, etc., each denoting specific modules or functionalities.

### Dependencies:

Dependenciesbetweencomponentsaredepictedthrougharrowsindicatingtheflowofcontrolor data:

* The web server node relies on Apache and PHP for processing web requests.
* Client devices interact with the web server through web browsers.
* The data base server node hosts My SQL, the data base management system employed by the system.
* Various dependencies exist among project components, such as PHP depending on Apache and components relying on the project root.

### Database:

A separated depiction of the "city \_resident \_id\_ card \_management" data bases how cases its tables, such as admin, moderator, residents, etc., housing critical data for user management, administrative divisions, resident information, and event tracking.

### Deployment:

Arrows indicating deployments how case artifacts being deployed to their respective nodes:

* The web server node hosts the project directory.
* The data base server node houses the data base schema file (city\_resident\_id\_card\_management.sql).
* Client devices communicate with the web server ,which subsequently interacts with the database server for data retrieval and storage.

## 3.6. User Interface Prototyping

The project team conducted extensive user inter face prototyping to ensure the design of the Web- based Nekemte City Resident ID Card Management System meets the needs and expectations of the users. The prototyping process involved the following steps:

1. **Wireframing**:Low-fidelitywireframesketcheswerecreatedtorepresentthebasiclayout and structure of the system's pages and user interfaces.
2. **Interactive Prototyping**: The wireframes were then transformed into interactive prototypes using tools like Figment or Adobe XD, allowing for the simulation of user interactions and navigation flows.
3. **User Feedback and Iteration**: The interactive prototypes were presented to city officials, resident representatives, and other stake holders for feedback and evaluation. Based on the feedback, the prototypes were iteratively refined and improved.
4. **Visual Design**: The finalized user interface prototypes were then transformed into high- fidelity visual designs, in corpora ting the city's branding ,color schemes, typography, and other visual elements.

The user interface proto typing process ensured that the final system design would be intuitive, user-friendly, and aligned with the needs and expectations of the Nekemte city residents and authorities.

## 3.7. Supplementary Specifications

In addition to the functional and non-functional requirements, the Web-based Nekemte City ResidentIDCardManagementSystemalsoincludesthefollowingsupplementaryspecifications:

#### Security and Data Privacy:

* + Thesystemshouldimplementrole-basedaccesscontrolstoensurethatuserscan only access the functionalities and data relevant to their assigned roles.
  + All sensitive data ,such as resident personal information and ID card details, should be encrypted both at rest and in transit.
  + Thesystemshouldcomplywithrelevantdataprotectionregulations,suchasthe General Data Protection Regulation (GDPR) or the Ethiopian Data Protection Proclamation.

#### Accessibility and Internationalization:

* + The system should be designed to be accessible or users with diverse abilities, including those with visual, auditory, or motor impairments.
  + Thesystemshouldsupportmultiplelanguageoptions,allowingresidentsto interact with the system in their preferred languages.

#### Performance and Scalability:

* + The system should be able to hand lea large and growing number of resident registrations and ID card applications without significant performance degradation.
  + Thesystemshouldbedesignedwithscalablearchitectureandinfrastructureto accommodate future expansion and increased user load.

#### Disaster Recovery and Business Continuity:

* + The system should have comprehensive backup and recovery mechanisms to ensuredataintegrityandavailabilityintheeventofsystemfailuresornatural disasters.
  + Thesystemshouldhaveawell-defineddisasterrecoveryplanthatoutlinesthe procedures and responsibilities for restoring the system and data in case of emergency situations.

#### Integration and Interoperability:

* + Thesystemshouldbedesignedwithopenandwell-documentedAPIstofacilitate integration with other city services and e-government initiatives.
  + Thesystemshouldadheretoindustry-standardprotocolsanddataformatsto ensure seamless interoperability with external systems.

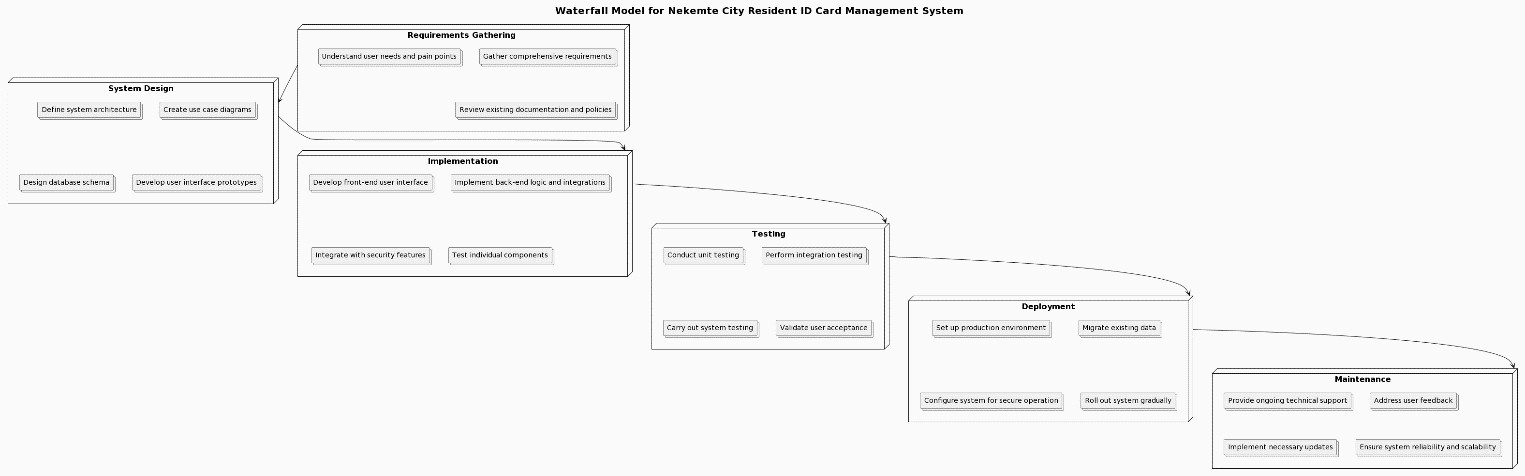
#### Maintenance and Support:

* + Thesystemshouldbedesignedwithamodularandmaintainablearchitecture, making it easy to update, upgrade, and troubleshoot.
  + Comprehensive documentation, including user manuals and technical guides, shouldbeprovidedtosupporttheongoingmaintenanceandadministrationofthe system.

## 

## 3.8. Waterfall Model Diagram

The development of the Nekemte City Resident ID Card Management System followed the classic Waterfall model, as depicted in the diagram below:



# Chapter4.System Design

## 4.1. Introduction

The system design for the Web-based Nekemte City Resident ID Card Management System focuses on creating a robust and scalable architecture that addresses the functional and non- functional requirements identified in the previous section. The design process involved the following key activities:

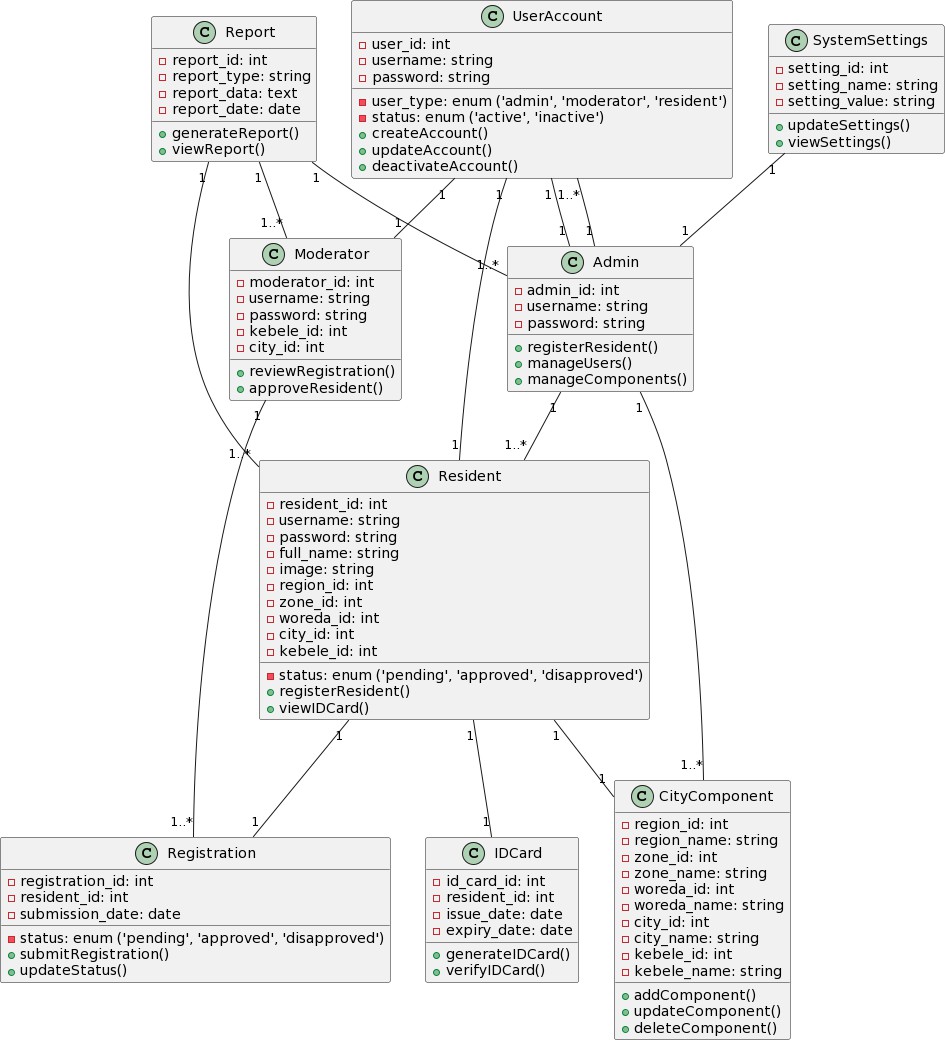
1. **Class Type Architecture**: Defining the high- level class types and their relationships to establish the overall system structure.
2. **Class Modeling**: Specifying the detailed attributes and methods for each class to capture the system's core functionality.
3. **State Chart Diagrams**: Modeling the dynamic behavior of the system, including the different states and transitions for key components.
4. **Collaboration Modeling**: Defining the interactions and collaborations between the various actors and system components.
5. **Component Modeling: Designing** the modular components of the system and their dependencies.
6. **Deployment Modeling**: Determining the physical deployment architecture and infrastructure requirements.
7. **Persistence Modeling**: Designing the data storage and retrieval mechanisms for resident and system-related information.
8. **User Interface Design**: Crafting the visual and interactive aspects of the system's user interfaces.

The following sections provide a detailed overview of the system design for the Web-based Nekemte City Resident ID Card Management System.

## 4.2. Class Type Architecture

The class type architecture for the Web- based Nekemte City Resident ID Card Management System consists of the following key classes:

1. **Resident**: Represents the city residents who apply for and manage their ID cards.
2. **Moderator**: Represents the city of facials responsible for reviewing and approving resident ID card applications.
3. **Admin**: Represents the system administrators who manage the overall functionality and configuration of the system.
4. **ID Card:** Represents the digital ID cards issued to residents, including the biunique identification numbers and associated information.
5. **Registration:** Represents the process of residents applying for and being issued ID cards.
6. **City Component**: Represents the various admin I stratify visions and geographical entities within the city, such as regions, zones, woredas, cities, and kebeles.
7. **User Account**: Represents the user accounts for residents, moderators and administrators, including their login credentials and permissions.
8. **Report**: Represents the various reports and analytics generated by the system for monitoring and decision-making.
9. **System Settings**: Represents the configurable settings and preferences for the overall system operation.



*Figure12ClassTypeArchitecture*

These class types and their relationships form the foundation of the system's object-oriented design,enablingtheimplementationoftherequiredfunctionalitiesandensuringmaintainability and extensibility.

## 4.3. Class Modeling

The detailed class modeling for the Web-based Nekemte City Resident ID CardManagement System includes the specification of the attributes and methods for each of the key classes identified in the class type architecture.

For example, the Resident class might have the following attributes and methods:

#### Attributes:

* id: Unique identifier for the resident
* first Name: First name of the resident
* last Name :Last name of the resident
* date Of Birth: Date of birth of the resident
* gender: Gender of the resident
* address: Address of the resident
* phone Number: Contact phone number of the resident
* email: Email address of the resident
* registration Date: Date of ID card registration
* id Card Number :Unique ID card number issued to the resident
* id Card Status: Status of the ID card(e.g. ,pending ,approved ,rejected)

#### Methods:

* register For ID Card():Allows the resident to submit an ID card application
* update Personal Info():Allows the resident to update their personal information
* view ID Card Details():Allows the resident to view their ID card information
* requestIDCardReplacement():AllowstheresidenttorequestareplacementfortheirID card

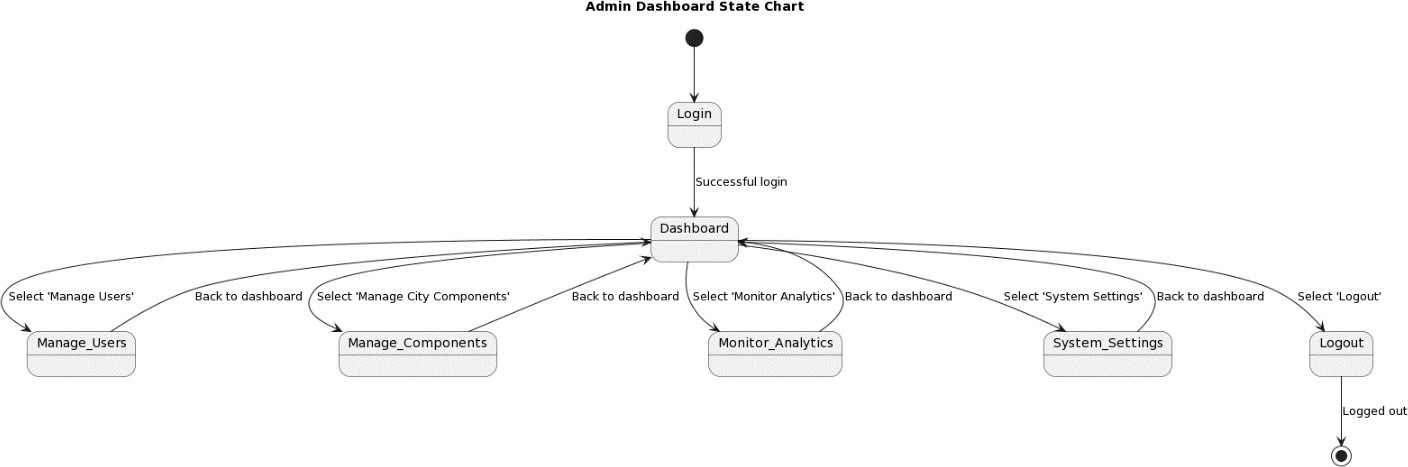
Similar class modeling exercises were conducted for the other key classes, such as Moderator, Admin, ID Card, and Registration ,to capture the specific attributes and methods required for the system's functionality.

## 4.4. State Chart Diagrams

### 4.4.1State Chart for Admin Dashboard

The state chart diagram for the admin dashboard illustrates the different states and transitions that an admin user can experience while interacting with the system. The key states include:

* + - 1. **Login**: The admin user logs in to the system.
      2. **Dashboard**: The admin user accesses the main dashboard, where they can perform various administrative tasks.
      3. **Manage Users**: The admin user can manage user accounts, including residents, moderators, and other admits.
      4. **Manage City Components: The** admin user can configure and manage the city's administrative components, such as regions, zones, wored as, cities, and kebeles.
      5. **Monitor Analytics**: The admin user can view system analytics and reports.
      6. **System Settings: The** admin user can access and modify the system's configuration settings.
      7. **Logout**: The admin user logs out of the system.



*Figure13StateChartforAdminDashboard*

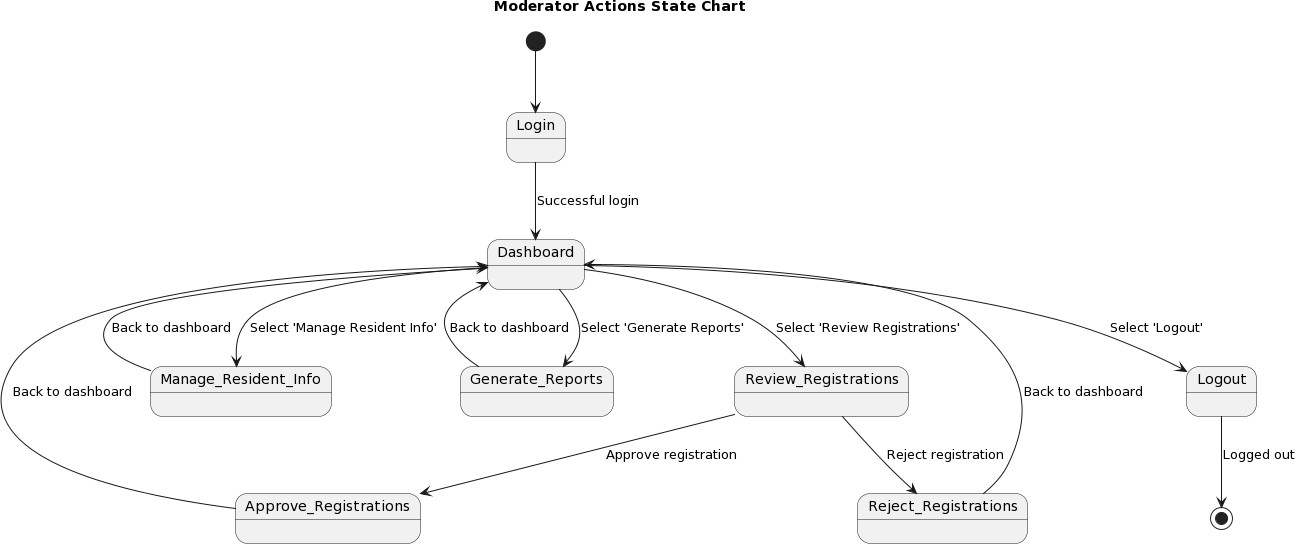
The state chart diagram captures the allowed transitions between the sestets, as well as the events and conditions that trigger the state changes.

### 4.4.2. State Chart for Moderator Actions

The state chart diagram for the moderator actions how cases the different states and transitions involved in the moderator's interactions with the system. The key states include:

* + - 1. **Login:** The moderator user logs into the system.
      2. **Dashboard**: The moderator user accesses the main dash board, where they can perform their assigned tasks.
      3. **Review Registrations**: The moderator user can review and process spending resident ID card registration requests.
      4. **Approve Registrations**: The moderator user can approve valid resident ID card registration requests.
      5. **Reject Registrations:** The moderator user can reject invalid or in complete resident ID card registration requests.
      6. **Manage Resident Info:** The moderator user can view and update resident information as needed.
      7. **Generate Reports**: The moderator user can generate reports on the ID card registration and approval process.
      8. **Logout**: The moderator user log out of the system.

The state chart diagram illustrates the allowed transitions between these states, as well as the events and conditions that trigger the state changes.

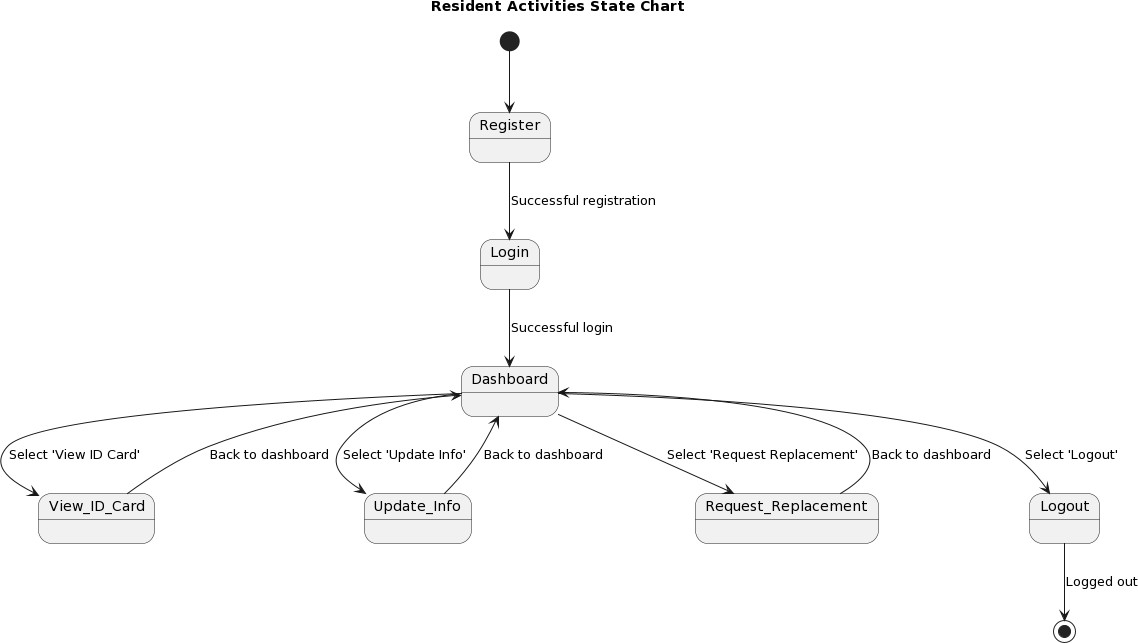


*Figure14StateChartforModeratorActions*

### 4.4.3. State Chart for Resident Activities

The state chart diagram for the resident activities outlines the different states and transitions that a resident user can experience while interacting with the system. The key states include:

* + - 1. **Register**: The resident user registers for a new ID card by providing the impersonal information and uploading the required documents.
      2. **Login**: The resident user logs into the system to access their ID card-related functionalities.
      3. **Dashboard**: The resident user accesses the min dash board, where they can view their ID card information and status.
      4. **View ID card**: The resident user can view the details of their issued ID card.
      5. **Update Info:** The resident user can update the impersonal information, such as dress or contact details.
      6. **Request Replacement**: The resident user can request are placement for their ID card, if needed.
      7. **Logout**: The resident user logs out of the system.



*Figure15StateChartforResident Activities*

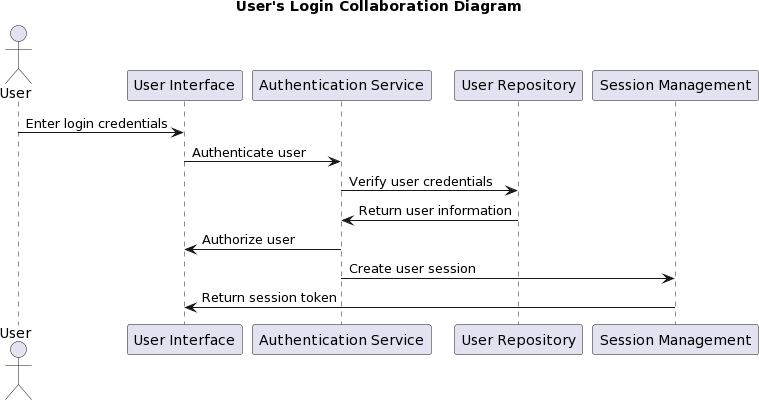
The state chart diagram captures the allowed transitions between the sestets, as well as the events and conditions that trigger the state changes.

## 4.5. Collaboration Modeling

### 4.5.1. Collaboration Diagram for User's Login

The collaboration diagram for the user's login process illustrates the interactions between the variouscomponentsandactorsinvolvedintheloginfunctionality.Thekeycomponentsinclude:

* + - 1. **User Inter face**: The web-based user inter face where the user enters their login credentials.
      2. **Authentication Service**: The service responsible for verifying the user's login credentials and authorizing access to the system.
      3. **User Repository**: The data store that manages the user account information, including usernames and passwords.
      4. **Session Management**: The componentthathandlesthecreation,maintenance,and termination of user sessions.



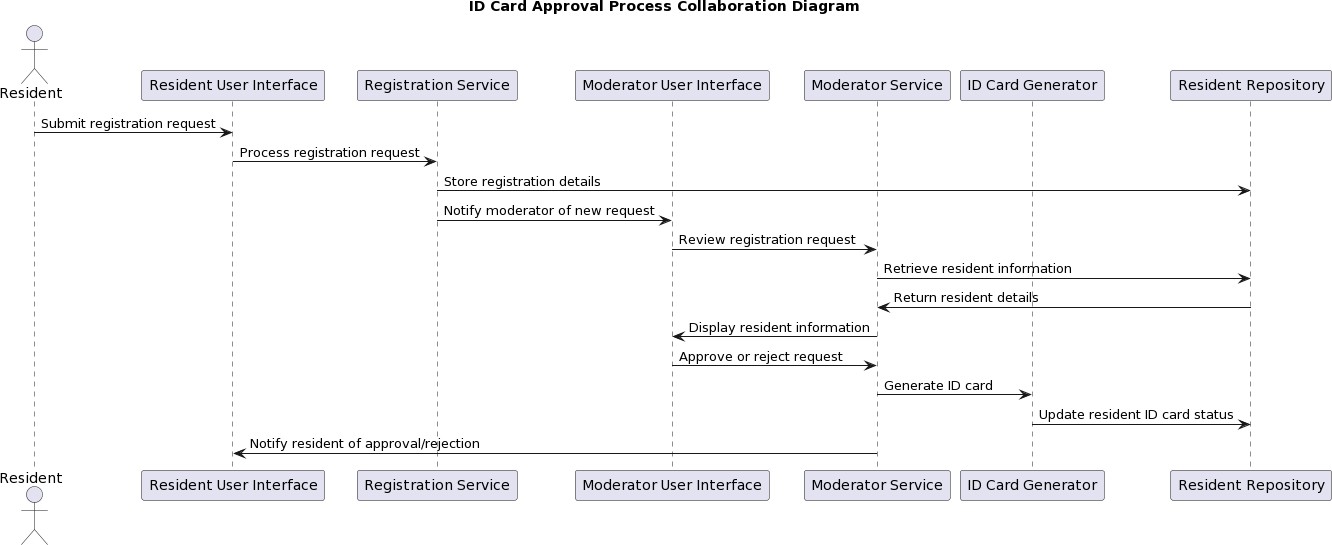
*Figure16CollaborationDiagramforUser's Login*

The collaboration diagram depicts the sequence of messages and method calls between these components, demonstrating how the login process is executed, from the user's input to the successful authentication and session establishment.

### 4.5.2Collaboration Diagram for ID card Approval Process

The collaboration diagram for the ID card approval process illustrates the interactions between thevariouscomponentsinvolvedintheresidentregistrationandIDcardissuanceworkflow.The key components include:

* + - 1. **Resident User Interface**: The web interface where residents can submit their ID card registration requests.
      2. **Registration Service**: The service responsible for receiving, validating, and processing the resident registration requests.
      3. **Moderator User Interface**: The web interface where moderators can review and approve or reject the resident registration requests.
      4. **Moderator Service**: The service that handles the moderator’s actions, such a viewing, approving, or rejecting the registration requests.
      5. **ID Card Generator**: The component that generates the unique ID card numbers and updates the resident's ID card status.
      6. **Resident Repository**: The data store that manages the resident information, including their registration details and ID card status.



*Figure17CollaborationDiagramforIDCardApprovalProcess*

The collaboration diagram depicts the sequence of messages and method calls between these components, demonstrating how the resident ID card approval process is executed, from the resident's submission of the registration request to the final issuance of the ID card.

## 4.6. Component Modeling

The component modeling for the Web-based Nekemte City Resident ID Card Management System involves the identification and design of the key software components that makeup the overall system. The main components include:

#### User Interface Components:

* + Resident-facing components (e.g. ,registration, dashboard, ID card view)
  + Moderator-facingcomponents(e.g.,registrationreview,approval,resident management)
  + Admin-facingcomponents(e.g.,usermanagement,citycomponentconfiguration, analytics)

#### Business Logic Components:

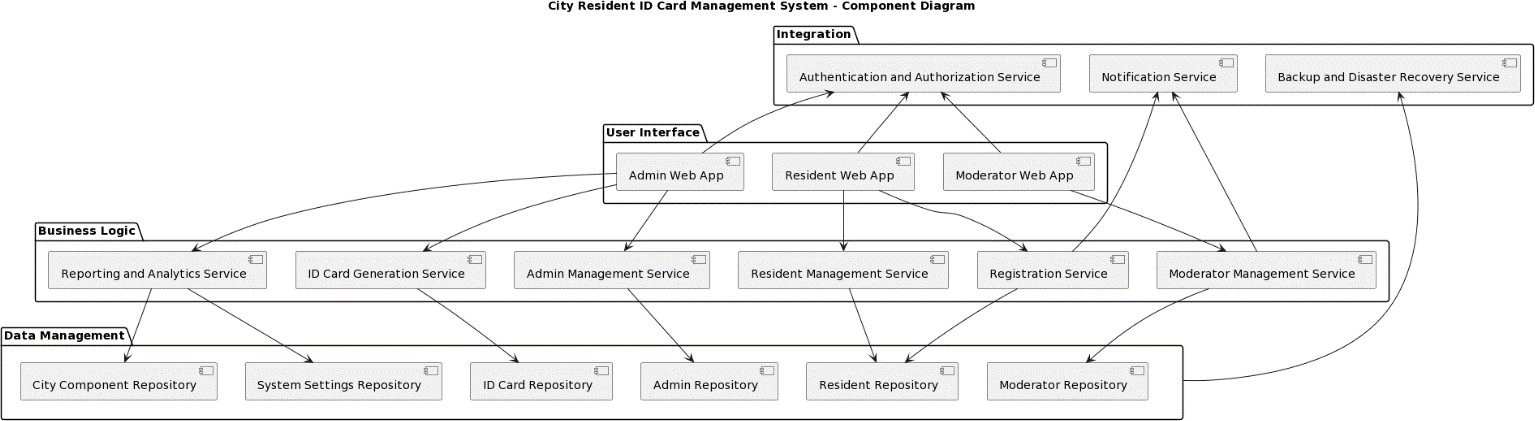
* + Registration Service
  + Resident Management Service
  + Moderator Management Service
  + Admin Management Service
  + ID Card Generation Service
  + Reporting and Analytics Service

#### Data Management Components:

* + Resident Repository
  + Moderator Repository
  + Admin Repository
  + ID Card Repository
  + City Component Repository
  + System Settings Repository

#### Integration Components:

* + Authentication and Authorization Service
  + Notification Service(email, SMS)
  + Backup and Disaster Recovery Service



*Figure18Component Modeling*

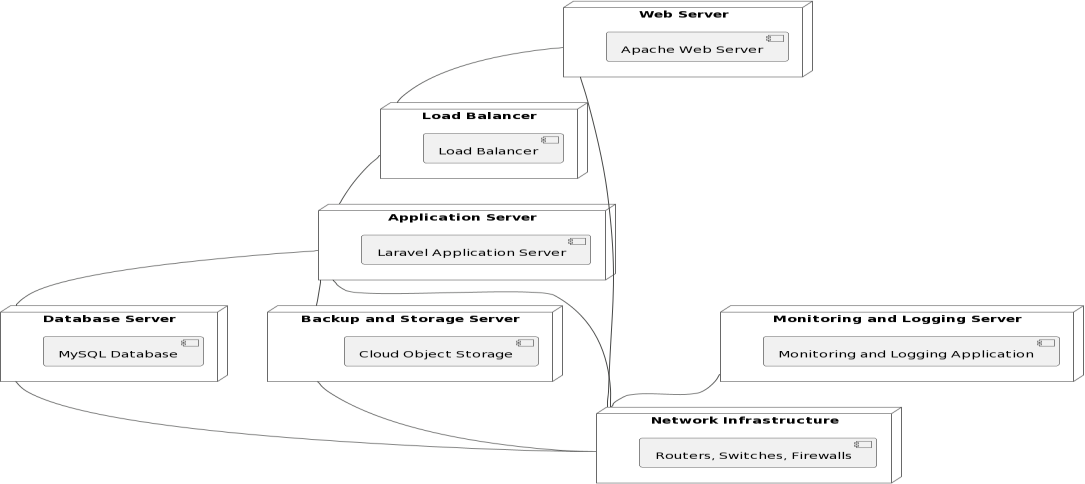
These components and their inter dependencies are designed to ensure modular and scalable architecture, facilitating maintainability, testability, and future extensibility of the system.

## 4.7. Deployment Modeling

The deployment modeling for the Web- based Nekemte City Resident ID Card Management System defines the physical infrastructure and deployment architecture required to host and operate the system. The key deployment components include:

1. **Web Server**: A web server(e.g., Apache, Nginx)that hosts the system's front-end application and serves the user interface to the clients.
2. **Application Server**: An application server(e.g. ,Larval, Spring Boot)that runs the system's business logic and handles the backend processing.
3. **Data base Server**: A data base server(e.g., My SQL ,Postgre SQL)that stores and manages the system's data, including resident information, ID card details, and other related data.
4. **Load Balancer**: A load balancer that distributes the incoming user traffic across multiple web and application server instances, ensuring high availability and scalability.
5. **Backup and Storage Server**: A backup and storage server (e.g., cloud-based object storage) that maintains regular backups of the system's data and serves as the secure off- site storage for disaster recovery purposes.
6. **Network Infrastructure**: The network infrastructure, including routers, switches, and firewalls, that provide secure connectivity and access control for the system's components.
7. **Monitoring and Logging Server**: A server that collects and analyzes system logs, performance metrics, and security events to ensure the overall heal the and operational integrity of the deployment.

The deployment model is designed to be scalable, fault-tolerant, and secure, aligning with the non-functional requirements of the system, such as availability ,reliability ,and data protection.



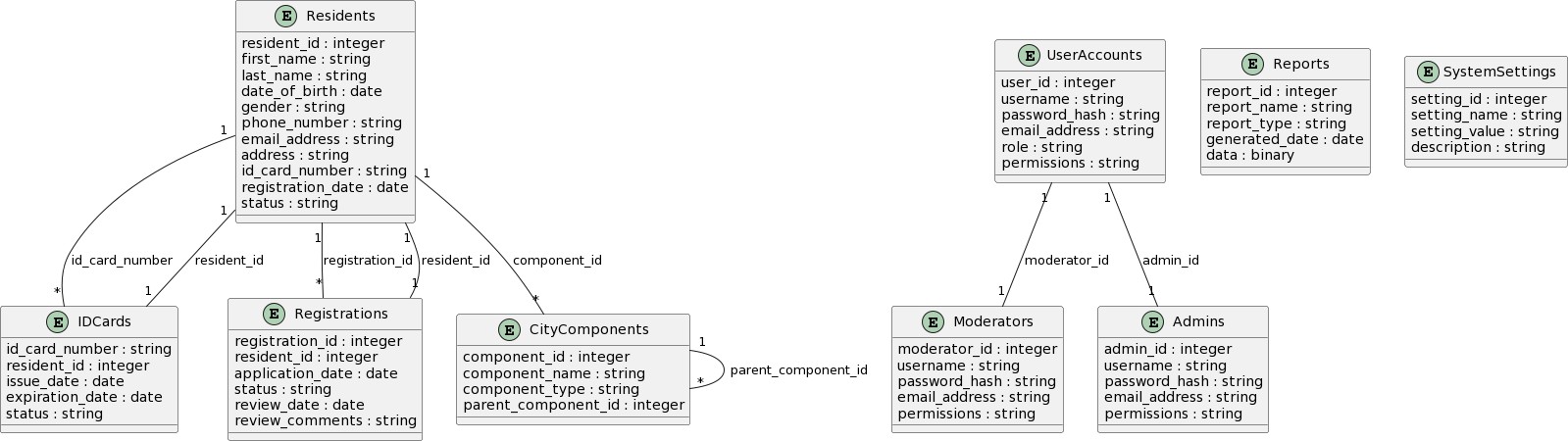
*Figure19DeploymentDiagram*

## 4.8. Persistence Modeling

The persistence modeling for the Web-based Nekemte City Resident ID Card Management Systemfocusesonthedesignofthedatabaseschemaanddatastoragemechanismsrequiredto persist the system's data. The key entities and their relationships include:

1. **Residents**: Stores the personal information and ID card details of the registered residents.
2. **Moderators**: Stores the login credentials and permissions for the moderate or users.
3. **Admin:** Stores the login credentials and permissions for the administrative users.
4. **ID Cards**: Stores the unique ID card numbers and associated resident information.
5. **Registrations**: Stores the details of the resident ID card registration requests, including their status (pending, approved, rejected).
6. **City Components**: Stores the hierarchical structure of the city's administrative visions (regions, zones, woredas, cities, kebeles).
7. **User Accounts**: Stores the user account information, including login credentials, roles, and permissions.
8. **Reports**: Stores the generated reports and analytics data related to the ID card management system.
9. **System Settings** :Stores the configure able settings and preferences for the overall system operation.

The data base schema is designed to ensure data integrity, maintain referential relationships ,and facilitate efficient querying and retrieval of the necessary information to support the system's functionalities.



*Figure20PersistenceModeling*

## 4.9. User Interface Design

The user interface design for the Web-based Nekemte City Resident ID Card Management Systemfocusesoncreatingavisuallyappealing,intuitive,anduser-friendlyexperienceforthe various user roles (residents, moderators, and administrators). The key elements of the user interface design include:

1. **Responsive and Mobile-friendly Layout**: The user interface is designed to be responsive end optimized for various screen sizes and devices, ensuring a consistent and accessible experience for users.
2. **Consistent Branding and Visual Identity**: The user interface incorporates the city's branding,includingcolors,logos,andtypography,tocreateacohesiveand recognizable visual identity.
3. **ClearandIntuitiveNavigation**:Theuserinterfacefeaturesawell-structurednavigation system, with intuitive menus and clear pathways for users to access the desired functionalities.
4. **AccessibilityFeatures**:Theuserinterfaceisdesignedwithaccessibilityguidelinesin mind, providing features such as high-contrast modes, screen reader support, and keyboard-only navigation to accommodate users with diverse needs.
5. **InformativeDashboardandVisualizations**:Theuserinterfaceincludesinformative dashboards and data visualizations, such as charts and graphs, to help users quickly understand the system's status and key metrics.
6. **SeamlessUserWorkflows**:Theuserinterfaceisdesignedtoguideusersthroughthe various tasks and processes, such as ID card registration, approval, and management, ensuring a smooth and efficient user experience.
7. **Responsive Feedback and Notifications**: The user interface provides clear and timely feedbacktousers,includingnotifications,errormessages,andsuccessindicators,tokeep them informed about the system's status and actions.
8. **MultilingualSupport**:Theuserinterfacesupportsmultiplelanguageoptions,allowing residents to interact with the system in their preferred languages.

Theuserinterfacedesignisaniterativeprocess,withcontinuousfeedbackandrefinementfrom the city authorities, resident representatives, and other stakeholders to ensure the system meets their needs and expectations.

# Chapter5.ImplementationandTesting

## 5.1. Introduction

The implementation and testing phase of the Web-based Nekemte City Resident ID Card Management System involved the translation of the system design into a working software application, as well as the comprehensive testing of the system's functionalities and performance. This section outlines the key activities and considerations during this phase of the project.

## 5.2. Objectives

The primary objectives of the implementation and testing phase were to:

1. Develop the system's software components according to the design specifications.
2. Ensure the system's functionalities and performance meet the stated requirements.
3. Identifyandaddressanyissuesordefectsinthesystemduringthedevelopmentand testing process.
4. Prepare the system for deployment and hand over to the city authorities.

## 5.3. Constraints on Implementation

The implementation of the Web-based Nekemte City Resident ID Card Management System faced the following key constraints:

1. **Limited Technical Expertise**: The city authorities had limited in-house technical expertisetosupporttheongoingmaintenanceandadministrationofthesystem, requiring a focus on creating comprehensive documentation and training.
2. **Tight Project Timeline**: The project had a fixed timeline for development and deployment, which required careful planning and efficient resource allocation to meet the deadlines.
3. **Budget Constraints**: The available budget for the project was limited, necessitating the selection of cost-effective technologies and infrastructure solutions.
4. **Integration with Legacy Systems**: The need to integrate the ID card management systemwiththecity'sexistinglegacysystemsanddatabasesposedchallengesintermsof data migration and system interoperability.

Theprojectteamdevelopedmitigationstrategiesandcontingencyplanstoaddressthese constraints and ensure the successful implementation and deployment of the system.

## 5.4. Testing

### 5.4.1. Testing by Requirements

The testing process for the Web-based Nekemte City Resident ID Card Management System was primarily driven by the identified functional and non-functional requirements. The key testing activities included:

#### Functional Testing:

* + - * + Validation of user registration, login, and access control mechanisms
        + Verification of resident ID card application, approval, and issuance workflows
        + Evaluation of resident information management and update functionalities
        + Testing of administrative features, such as user management and city component configuration

#### Non-functional Testing:

* + - * + Usability testing to ensure the system's inter faces and work flows meet user expectations
        + Performance testing to assess the system's scalability and responsiveness under various load conditions
        + Security testing to identify and mitigate potential vulnerabilities and threats
        + Accessibilitytestingtoensurethesystem'scompliancewithrelevantguidelines and standards

#### Integration Testing:

* + - * + Verification of the system's integration with external data sources and legacy systems
        + Evaluation of the data exchange, transformation ,and synchronization processes

#### Acceptance Testing:

* + - * + Involvement of city authorities and resident representatives to validate the system's compliance with the specified requirements
        + Confirmation of the system's readiness for deployment and handover

The testing activities were documented, and the results were thoroughly analyzed to ensure the final system meets the needs and expectations of the city authorities and residents.

### 5.4.2. Unit Testing

The project team conducted extensive unit testing to ensure the individual components and modules of the Web-based Nekemte City Resident ID Card Management System function as expected. The unit testing process involved the following key steps:

* + - 1. **Test Case Development**: The team created comprehensive test cases for each unit of the system, covering the various functionalities, inputs, and expected outputs.
      2. **Test Automation**: The team utilized automated testing frame works ,such as PHP Unit for the backend components and Selenium for the front-end user interface, to streamline the execution of unit tests.
      3. **Continuous Integration**: The unit tests were integrated into the project's continuous integration (CI)pipeline, ensuring that any code changes were automatically tested before being merged into the main codebase.
      4. **Test Coverage Monitoring**: The team closely monitored the test coverage metrics to maintainahighlevelofcodecoverageandidentifyareasthatrequiredadditionaltesting.
      5. **Defect Tracking and Resolution**: Any issues or defects is covered during the unit testing process were logged, prioritized, and addressed by the development team.

### 5.4.3. Integration Testing

In addition to the unit testing, the project team also conducted comprehensive integration testing to verify the proper functioning of the system's components and their interactions. The key steps in the integration testing process were:

* + - 1. **Test Environment Setup**: The team created a dedicated integration testing environment that closely mirrored the production deployment architecture, including the web server, application server, and database server.
      2. **Test Case Development**: The team developed integration test cases that focused on the interactions between the various components of the system, such as the user interface, business logic, and data persistence layers.
      3. **Incremental Integration Testing**: The team adopted an incremental approach to integration testing, starting with the core system components and gradually incorporating additional modules and interfaces.
      4. **End-to-End Scenario Testing**: The team designed and executed end-to-end test scenarios that simulated the complete user journeys, such as resident registration, ID card approval, and report generation.
      5. **Performance and Scalability Testing**: The team conducted load and stress stets to assess the system's performance and scalability under various user load conditions, ensuring the system can handle the expected traffic and usage patterns.
      6. **Defect Tracking and Resolution**: Any issues or defects identified during the integration testing process were logged, prioritized, and addressed by the development team.

The successful completion of the integration testing provided confidence in the system's overall functionality, reliability, and readiness for deployment.

## 5.5. Error Handling

The Web-based Nekemte City Resident ID Card Management System in corporatesrobusterror handling mechanisms to ensure the system's stability and resilience. The key error handling strategies include:

1. **Centralized Exception Handling**: The system utilizes centralized exception handling framework that captures and logs all runtime errors and exceptions, providing a single point of control for error management.
2. **Structured Error Messages**: The system displays clear and user-friendly error messages to the end-users, providing them with relevant information about the issue and guidance on how to resolve it.
3. **Graceful Degradation**: In the event of critical system failures or errors, the system is designed to gracefully degrade, maintaining essential functionalities and preventing data loss or corruption.
4. **Logging and Monitoring**: The system integrates with a comprehensive logging and monitoring solution ,which tracks and analyzes system events, performance metrics, and error patterns to support troubleshooting and proactive issue resolution.
5. **Failover and Redundancy**: The system's infrastructure is designed with failover mechanismsandredundantcomponentstoensurehighavailabilityandminimizethe impact of hardware or service failures.
6. **Data Validation and Sanitization**: The system implements robust data validation and sanitization techniques to prevent and mitigate potential security threats, such as SQL injection or cross-site scripting (XSS) attacks.

These error handling strategies help to ensure the overall reliability and maintainability of the Web-based Nekemte City Resident ID Card Management System, enabling the city authorities to provide a stable and secure platform for resident ID card management.

## 5.6. Sample Pages and Source Codes

The implementation of the Web-based Nekemte City Resident ID Card Management System resultedinthedevelopmentofvarioussoftwarecomponentsanduserinterfaceelements.The following are sample pages and source code excerpts from the system:

**Resident Registration Page**:

<formaction="/register"method="post">

<divclass="form-group">

<labelfor="firstName">FirstName</label>

<inputtype="text"class="form-control"id="firstName"name="firstName"required>

</div>

<divclass="form-group">

<labelfor="lastName">LastName</label>

<inputtype="text"class="form-control"id="lastName"name="lastName"required>

</div>

<!--Additionalformfieldsforresidentinformation -->

<buttontype="submit"class="btnbtn-primary">Register</button>

</form>

**Moderator Approval Page**:

<?php

//Fetchthependingresidentregistrationrequests

$pendingRequests=$registrationService->getPendingRequests();

?>

<tableclass="table">

<thead>

<tr>

<th>ResidentName</th>

<th>ApplicationDate</th>

<th>Status</th>

<th>Actions</th>

</tr>

</thead>

<tbody>

<?phpforeach($pendingRequestsas$request):?>

<tr>

<td><?=$request->firstName.''.$request->lastName?></td>

<td><?=$request->registrationDate?></td>

<td>Pending</td>

<td>

<ahref="/approve/<?=$request->id?>"class="btnbtn-success">Approve</a>

<ahref="/reject/<?=$request->id?>"class="btnbtn-danger">Reject</a>

</td>

</tr>

<?phpendforeach;?>

</tbody>

</table>

**Admin User Management Page**:

<?php

//Fetchthelistofuser accounts

$users=$userService->getAllUsers();

?>

<tableclass="table">

<thead>

<tr>

<th>Username</th>

<th>Role</th>

<th>Status</th>

<th>Actions</th>

</tr>

</thead>

<tbody>

<?phpforeach($usersas$user):?>

<tr>

<td><?=$user->username?></td>

<td><?=$user->role?></td>

<td><?=$user->status?></td>

<td>

<ahref="/edit/<?=$user->id?>"class="btnbtn-primary">Edit</a>

<ahref="/delete/<?=$user->id?>"class="btnbtn-danger">Delete</a>

</td>

</tr>

<?phpendforeach;?>

</tbody>

</table>

**ID card Generation and Printing**:

<?php

//Fetchtheapprovedresidentregistrationrequest

$resident=$registrationService->getApprovedResident($id);

//GenerateauniqueIDcardnumber

$idCardNumber=$idCardService->generateIDCardNumber();

//Updatetheresident'sIDcardinformation

$resident->idCardNumber=$idCardNumber;

$resident->idCardStatus='Issued';

$registrationService->updateResident($resident);

//RendertheIDcardtemplateand print

$idCardTemplate=$this->render('id-card.html.twig',[ 'resident' => $resident,

'idCardNumber'=>$idCardNumber

]);

//OutputtheIDcardtemplateforprinting echo $idCardTemplate;

?>

**Reporting and Analytics**:

<?php

//FetchtheIDcardregistrationandissuance statistics

$registrationStats=$reportingService->getRegistrationStats();

$issuanceStats=$reportingService->getIssuanceStats();

?>

<div class="row">

<divclass="col-md-6">

<h2>RegistrationStatistics</h2>

<canvasid="registrationChart"></canvas>

</div>

<divclass="col-md-6">

<h2>IDCardIssuance Statistics</h2>

<canvasid="issuanceChart"></canvas>

</div>

</div>

<script>

// Render the charts using a charting library (e.g., Chart.js) newChart(document.getElementById('registrationChart'),{ type: 'bar',

data: {

labels:<?=json\_encode(array\_keys($registrationStats))?>, datasets: [{

label:'Registrations',

data:<?=json\_encode(array\_values($registrationStats))?>

}]

}

});

newChart(document.getElementById('issuanceChart'),{ type: 'line',

data: {

labels:<?=json\_encode(array\_keys($issuanceStats))?>, datasets: [{

label:'IDCards Issued',

data:<?=json\_encode(array\_values($issuanceStats))?>

}]

}

});

</script>

These sample pages and source code excerpts providing limps in to the implementation of 1the Web-based Nekemte City Resident ID Card Management System, showcasing the various functionalities and user interface elements developed during the project.

# Chapter6.ConclusionandRecommendation

## 6.1. Conclusion

The Web-based Nekemte City Resident ID Card Management System has been successfully developed and implemented, addressing the key challenges faced by the city's existing manual ID card management system. The system provides a centralized and efficient platform for managing the identification cards of Nekemte city residents, improving the accessibility, accuracy, and security of resident information.

Through the implementation of this project, the city of Nekemte has achieved the following key benefits:

* + 1. Streamlined resident ID card application and issuance process, enhancing the convenience and experience for the city's residents.
    2. Improveddataintegrityandreliabilitythroughtheuseofacentralizeddigitaldatabase, reducing the risk of errors and inconsistencies.
    3. IncreasedefficiencyandoversightforcityauthoritiesinmanagingtheIDcardsystem, enabling better decision-making and service delivery.
    4. Strengthenedsecuritymeasurestoprotectresidentdataandpreventunauthorizedaccess or misuse of the ID card system.
    5. Facilitatedintegrationwithothercityservicesande-governmentinitiatives,allowing for seamless access to public services for Nekemte residents.

The successful deployment of the Web-based Nekemte City Resident ID Card Management System has set the foundation or further digital transformation and improved public service delivery within the city.

## 6.2. Recommendation

To ensure the long-terms sustainability and continued success of the Web-based Nekemte City Resident ID Card Management System, the following recommendations are made:

* + 1. **Ongoing Maintenance and Support**: The city authorities should establish a dedicated IT support team to provide regular maintenance, updates, and technical assistance for the system.Thiswillensurethesystem'sreliabilityandresponsivenesstothechangingneeds of the city and its residents.
    2. **Continuous User Training and Engagement**: The city should invest incomprehensive user training programs for residents, moderators, and administrators to ensure effective adoption and utilization of the system. Additionally, regular feedback and engagement with users will help identify areas for improvement and enhance the overall user experience.
    3. **Integration with Other City Services**: The city should continue to explore and implementintegrationbetweentheIDcardmanagementsystemandothercityservices, such as e-government portals, municipal service delivery, and social we lf are programs. This will further enhance the system's value and utility for the residents.
    4. **Expansion and Scalability**: As the city's population grows the ID card management systemshouldbedesignedtoscaleandaccommodatetheincreasingnumberofresidents and ID card applications. The city should plan for future infrastructure and system upgrades to ensure the system's ability to handle the growing demand.
    5. **Disaster Recovery and Business Continuity**: The city should regularly review and update the system's disaster recovery and business continuity plans to ensure the availability and resilience of the ID card management system, even in the event of unforeseen disruptions or emergencies.

1. **Continuous Improvement and Innovation**: The city should establish a process for regularly reviewing the system's performance, gathering user feed back, and identifying opportunities for improvement and innovation. This will help keep the ID card management system relevant and responsive to the evolving needs of the city and its residents.

By following these recommendations, the city of Nekemte can ensure the long-term success and sustainabilityoftheWeb-basedNekemteCityResidentIDCardManagementSystem, ultimately enhancing the overall quality of public services and the well-being of its residents.

## 6.3. Installation

The Web-based Nekemte City Resident ID Card Management System can be installed and deployed using the following steps:

* + 1. **Server Provisioning**: Set up the required server infrastructure, including the web server, application server, and database server, as per the deployment model outlined in the system design.
    2. **Data base Setup**: Create the data base schema and populate it with the initial data using the provided SQL scripts located in the db/ directory.
    3. **Application Deployment**: Deploy the web-based application files to the web server, ensuring that the necessary dependencies and configurations are in place.
    4. **Configuration and Integration**: Customize the system settings, integrate with any external systems or services, and configure the access controls and user permissions as required.
    5. **Testing and Validation**: Conduct thorough testing of the deployed system, including functional, performance, and security tests, to ensure the system is functioning as intended and meeting the specified requirements.
    6. **Monitoring and Maintenance**: Setup the necessary monitoring and logging solutions to track the system's performance, security, and any issues that may arise. Establish maintenance and support plan to address any future updates, up grades, or troubleshooting needs.

# 6.4References

1.Sommerville .I(2015). Software Engineering (10th Ed.) Person.

Used For System Design Methodologies And Requirements Analysis

2.Laravel Documentation(2023).

Larval Frame Work. [Https://Laravel.Com/Docs](Https://laravel.com/docs)

Guided The Implementation Of Mvc Architecture And Crud Operations

3.Msql Official Documentation (2023). Myasql 8.0 References Manual.

[Https://Dev.Mysql.Com/Doc](https://dev.mysql.com/doc)

4.W3shools. (2023)Html5,Css3.And Javascript Tutorials

[Https://Www.W3schools.Com](https://www.w3schools.com)

5.General Data Protection Regulations

(GDPR).(2018).EU Data Protection

Rules. Https://Gdpr-Info.Eu

Referenced For Data Privacy And Security Compliance