Chapter I: Introduction

1.1 Introduction of Project

Razza, an Online Citizen Information Processing Solution, is a SaaS (Software as a Service) based online solution fulfilling all the aspects of a complete citizen information searching service.

Razza is a web-based solution that enables an authorized individual or law & order maintaining organization to effectively search for all the necessary details of an individual in an all-in-one platform.

Razza is based on the concept of NagarikApp, a portal by Government of Nepal, but enhances the functionality of searching of an individual data by the means of any other data. For example, we can search for all other details of the person based on his/her registered phone number, passport number, citizenship number, voter id card number or any other officially recognized unique access point.

1.2 Problem Statement

The lack of effective systems for collecting, storing, and managing the personal information of individuals within a country or region. This can lead to a range of issues, including identity theft, fraud, and a lack of access to essential services. (Clough, 2015)

According to, Ram et al. (2017), One of the main problems associated with underdeveloped government in recording personal data is the lack of reliable identification systems. Without proper identification, individuals may be unable to access basic services such as healthcare, education, and banking, or may be vulnerable to fraud and identity theft. (Christensen, 2012)

For all these problems, the aim of my project is to solve these all problems with a high security system in which government can keep the data of citizens within a single system for a reliable identification and for the ease of the use/find of each and every data being connected with the government. (Higgins & Green, 2008)

Overall, the proposed project will have a significant impact on the day-to-day operations of government, improving the government ability to deliver quick and reliable service to the citizens by reducing the workload on both government agencies and government.

1.3 Objectives

The specific objectives are mentioned below:

- **Efficient Data Collection**: The system should be designed to collect and manage data from citizens in an efficient manner. This includes ensuring that the data collected is accurate, complete, and up-to-date.
- **Improved Decision Making**: The system should provide decision-makers with accurate and timely information that can help them make better decisions. This may include information about citizens' needs, preferences, and behaviors.
- **Enhanced Service Delivery**: The system should help improve the quality and efficiency of service delivery to citizens. This may include faster processing times, improved access to services, and better communication with citizens.
- **Improved Accountability**: The system should help improve accountability by making it easier to track and monitor the performance of government agencies and service providers.
- **Protection of Citizens' Privacy**: The system should be designed to protect citizens' privacy and ensure that their data is only used for the purposes for which it was collected.

Overall, the objective of Razza is to help government agencies and service providers better understand citizens' needs and preferences, and to improve the quality and efficiency of service delivery to citizens.

1.4 Scope & Limitations

In today's world, unavailability of reliable identification system has led to various of disadvantages as mentioned in the problem statement. So, this project will be of great help to banking industry, insurance industry and law & order management systems. (Executive, 1991)

However, there are various limitations to this project itself. The first one being the required number of data to achieve the prementioned goal. And preventing the misuse of such classified information is also a real pain. (Andersson & Kjellgren, 2017)

Chapter II: Requirement Analysis

2.1 Functional Requirement

Functional requirements are defined by the function or component required of an application.

- Providing the online interface for searching of citizen's data.
- Decreasing the risk of identification frauds.
- Decrease time required to access individual data.
- To make the system more secure.

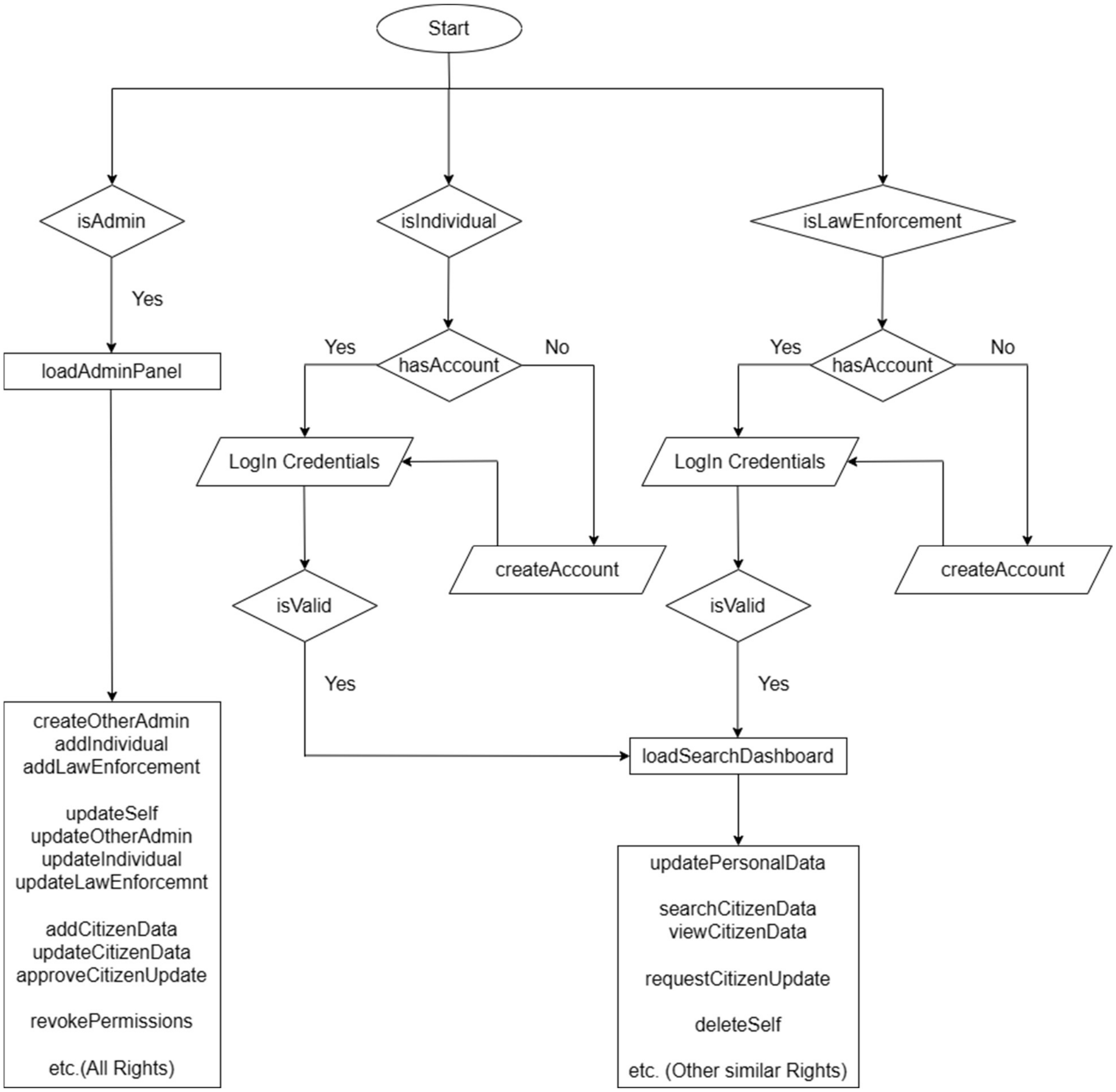


Fig: Use Case Diagram of Razza

2.2 Non–Functional Requirements

- The overall performance of the system must be above 20 Mbps.
- The system must be secured by a strong password.
- The UI design must be consistent throughout all the pages.
- The service must be fully responsive to work on all devices and browsers.

2.3 Feasibility Study

A) Technical Feasibility

- **Infrastructure and Platform:** Cloud Platform like Digital Ocean, Linode as a server.
- **Data Storage:** Postgres, Sqlite3 or any other relational database as a storage medium.
- **Software Development:** Nuxt3 with Tailwind as Frontend & Django as Backend.
- **Data Integration:** Backend will be exposed as API under *server-url/api/v2/*.
- **Scalability:** Hosting on the cloud platform means its highly scalable.

B) Operational Feasibility

- **User Acceptance:** Feedbacks from individuals will evaluate its acceptance & adoption.
- **Workflow Integration:** It integrates well with the previous technology i.e., Nagarik App.
- **Training & Support:** User Documentation & Training sessions must do the trick.

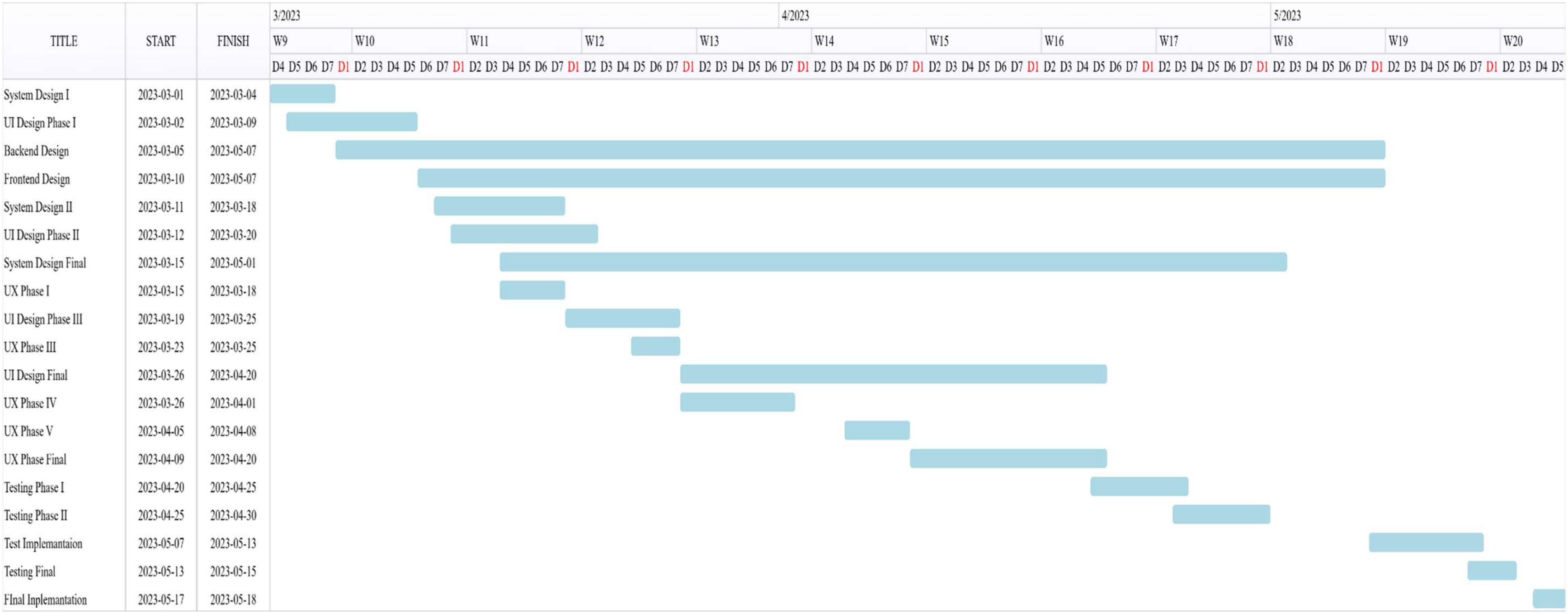
C) Economic Feasibility

- **Cost-Benefit Analysis:** On conducting a comprehensive cost-benefit analysis, Razza is forecasted to be a big hit on the market.
- **Return on Investment (ROI):** Based on the pool data, Razza has 3000% ROI.
- **Revenue Generation:** Razza can be stated as public company and IPO model used as the revenue generation model.

D) Legal Feasibility

- **Data Protection & Privacy Laws:** Razza is GDPR and other all privacy laws compliant service.
- **Intellectual Property Rights:** Every data is owned collectively by the individual and Government of Nepal and possess the right make the data confidential on the case of a written order from the Justice Department of Nepal or other foreign entities.
- **Compliance with Government Regulations:** Razza is completely in alliance with Government of Nepal and complies with all the laws responsible.
- **Jurisdictional Considerations:** Razza is a national project but it follows all the jurisdictional laws to encounter any problems that may occur in the future.

E) Schedule / Timeline Feasibility



TITLE	START	FINISH
System Design I	2023-03-01	2023-03-04
UI Design Phase I	2023-03-02	2023-03-09
Backend Design	2023-03-05	2023-05-07
Frontend Design	2023-03-10	2023-05-07
System Design II	2023-03-11	2023-03-18
UI Design Phase II	2023-03-12	2023-03-20
System Design Final	2023-03-15	2023-05-01
UX Phase I	2023-03-15	2023-03-18
UI Design Phase III	2023-03-19	2023-03-25
UX Phase III	2023-03-23	2023-03-25
UI Design Final	2023-03-26	2023-04-20
UX Phase IV	2023-03-26	2023-04-01
UX Phase V	2023-04-05	2023-04-08
UX Phase Final	2023-04-09	2023-04-20
Testing Phase I	2023-04-20	2023-04-25
Testing Phase II	2023-04-25	2023-04-30
Test Implemantaion	2023-05-07	2023-05-13
Testing Final	2023-05-13	2023-05-15
FInal Inplemantation	2023-05-17	2023-05-18

3/2023
4/2023
5/2023
W9
W10
W11
W12
W13
W14
W15
W16
W17
W18
W19
W20
D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5 D6 D7 D1 D2 D3 D4 D5

Fig: Gantt Chart

Chapter III: System Design

3.1 Class Diagram

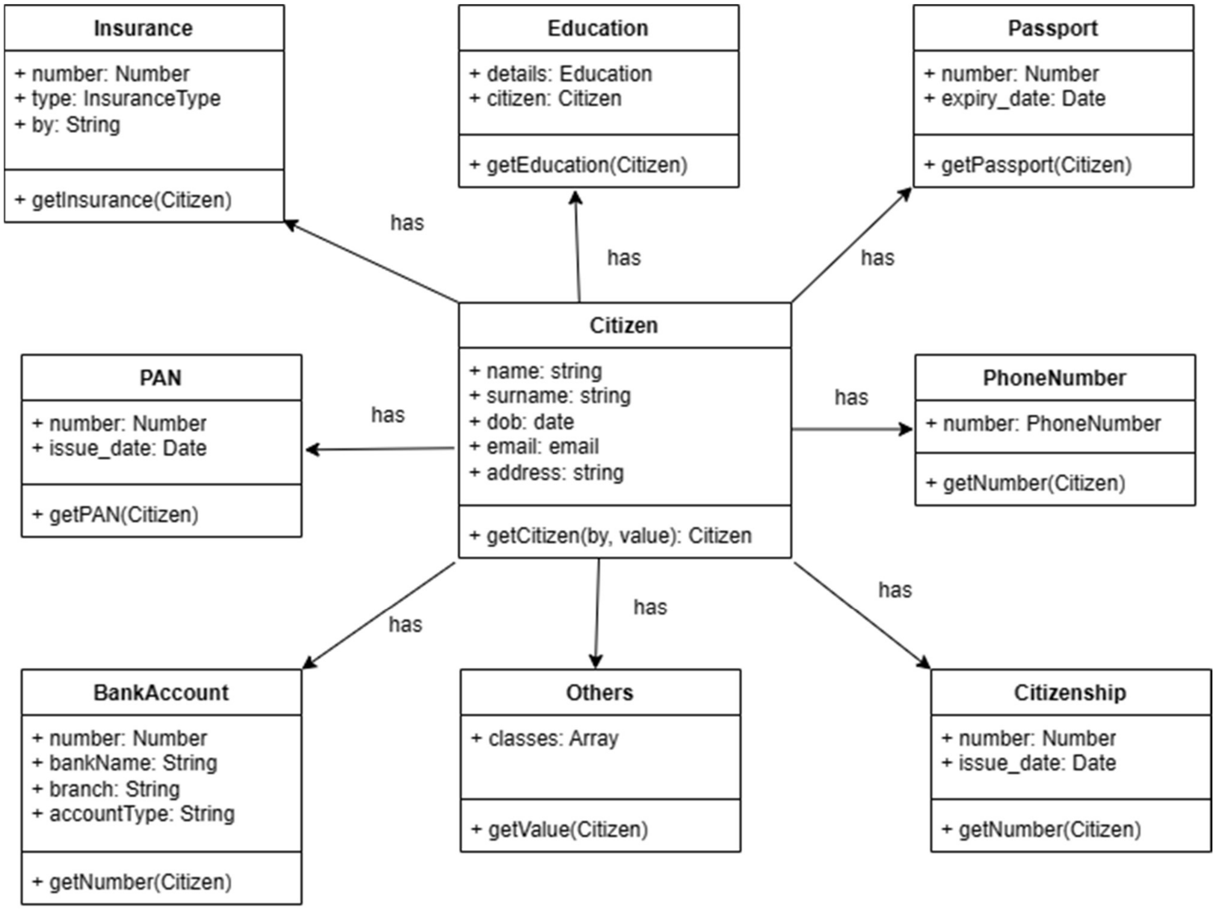


Fig: Class Diagram of Razza

3.2 Sequence Diagram

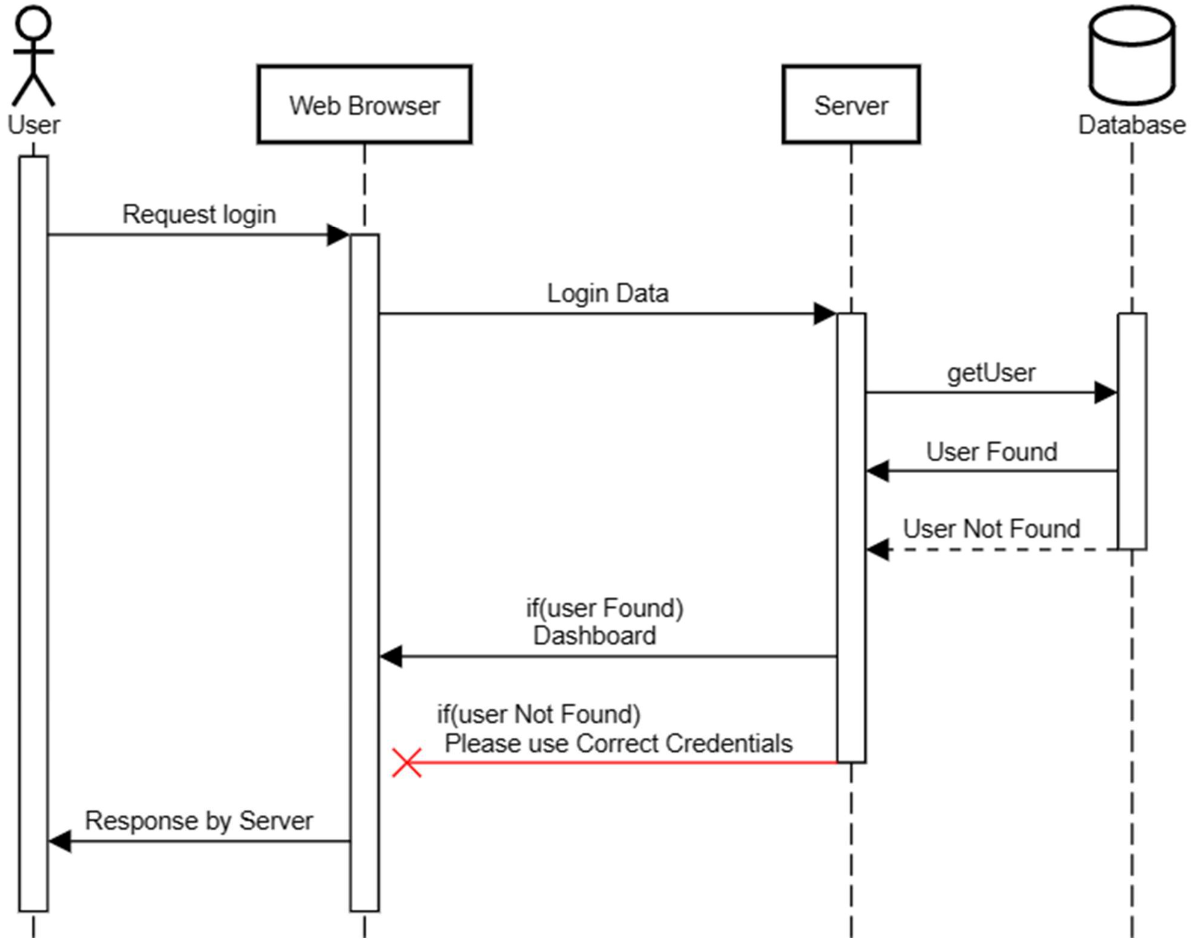


Fig: Sequence Diagram of Razza for login user flow

3.3 Activity Diagram

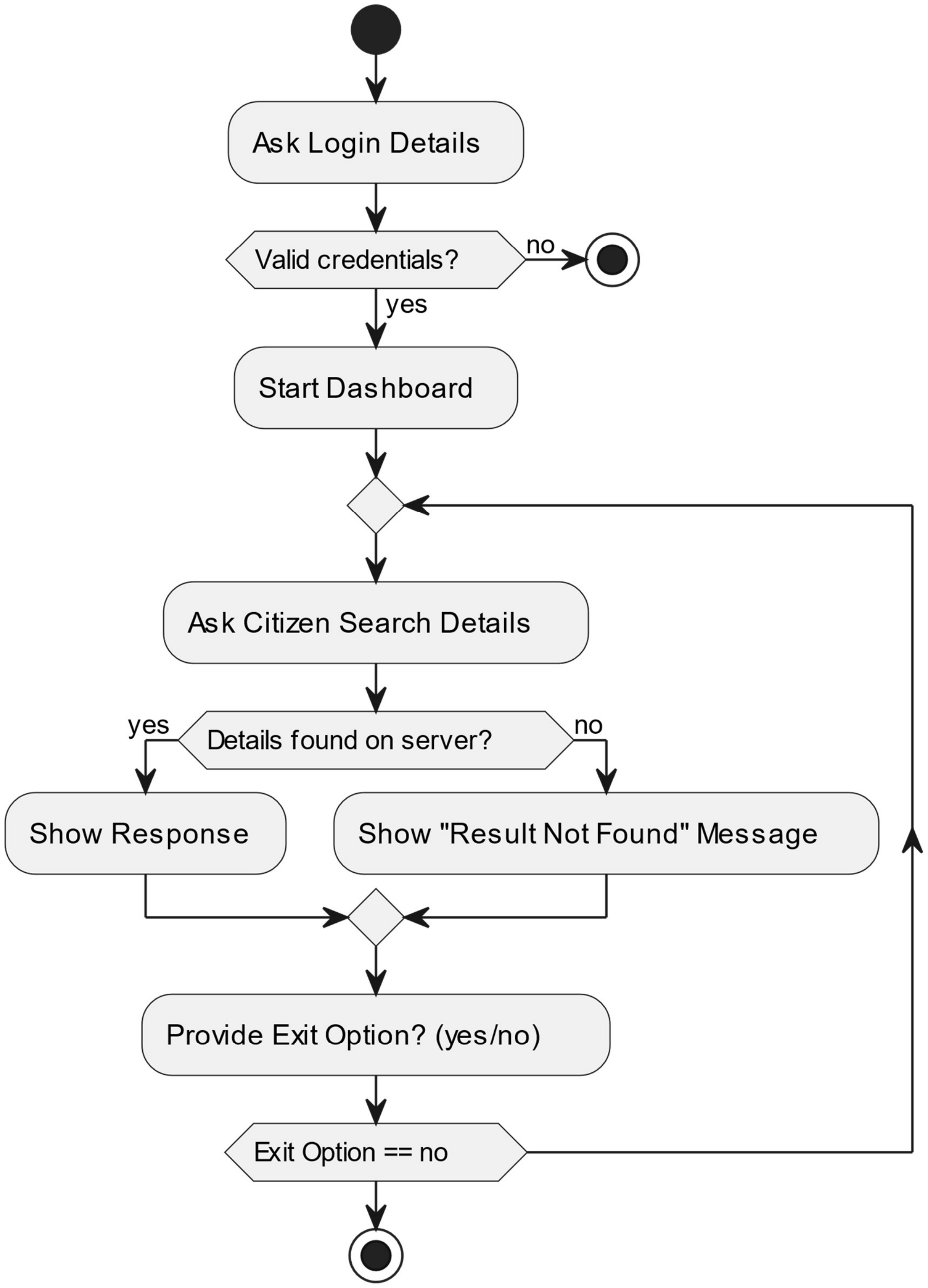


Fig: Activity Diagram of Razza for citizen search details

3.4 State Diagram

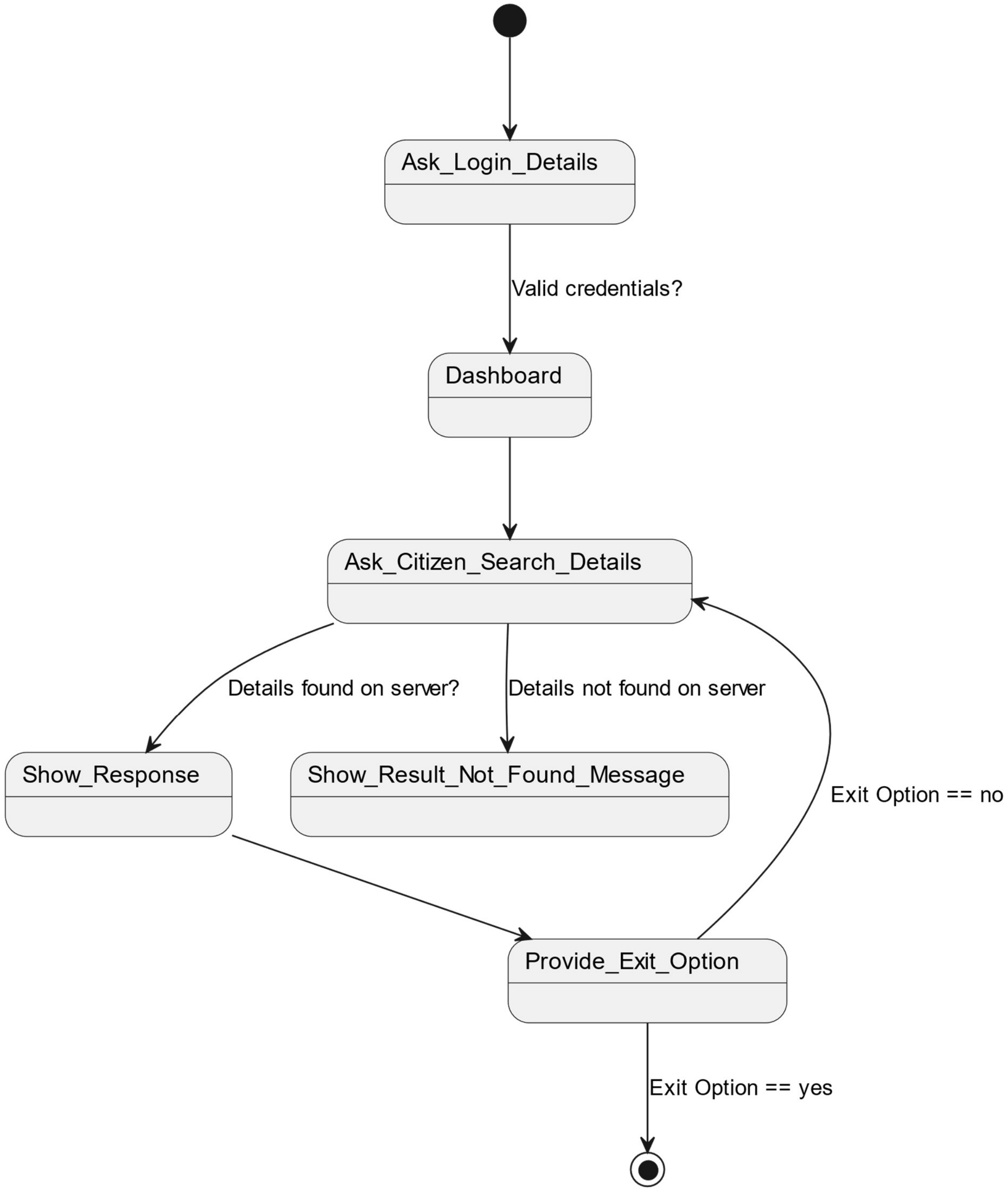


Fig: State Diagram for same use case as Activity Diagram

3.5 Schema Diagram

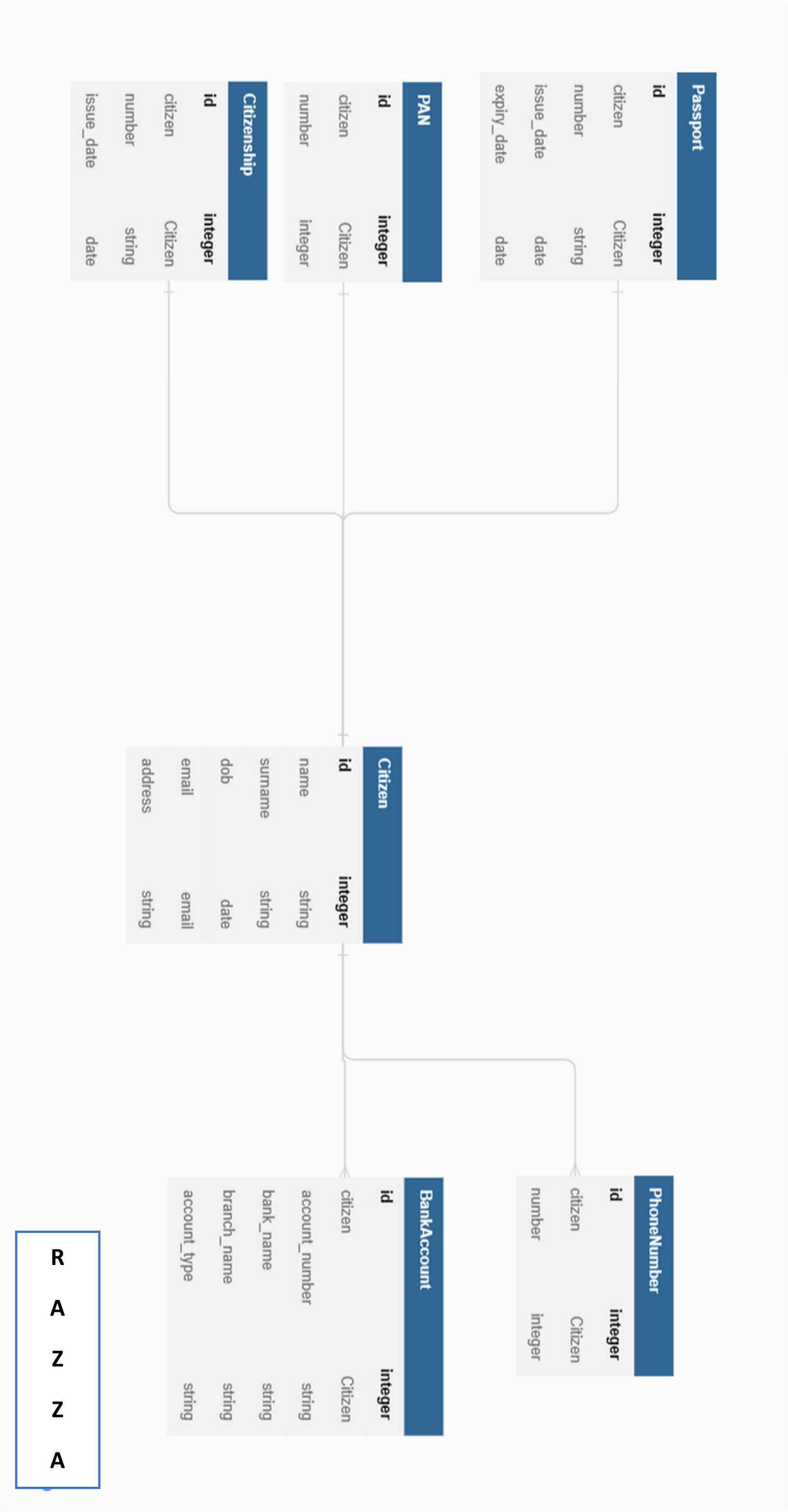


Fig: Schema Diagram of Razza

Chapter IV: Implementation & Testing

4.1 Implementation Methodology

Incremental Model is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation and testing phases. Every subsequent release of the module adds function to the previous release. The process continues until the complete system achieved.

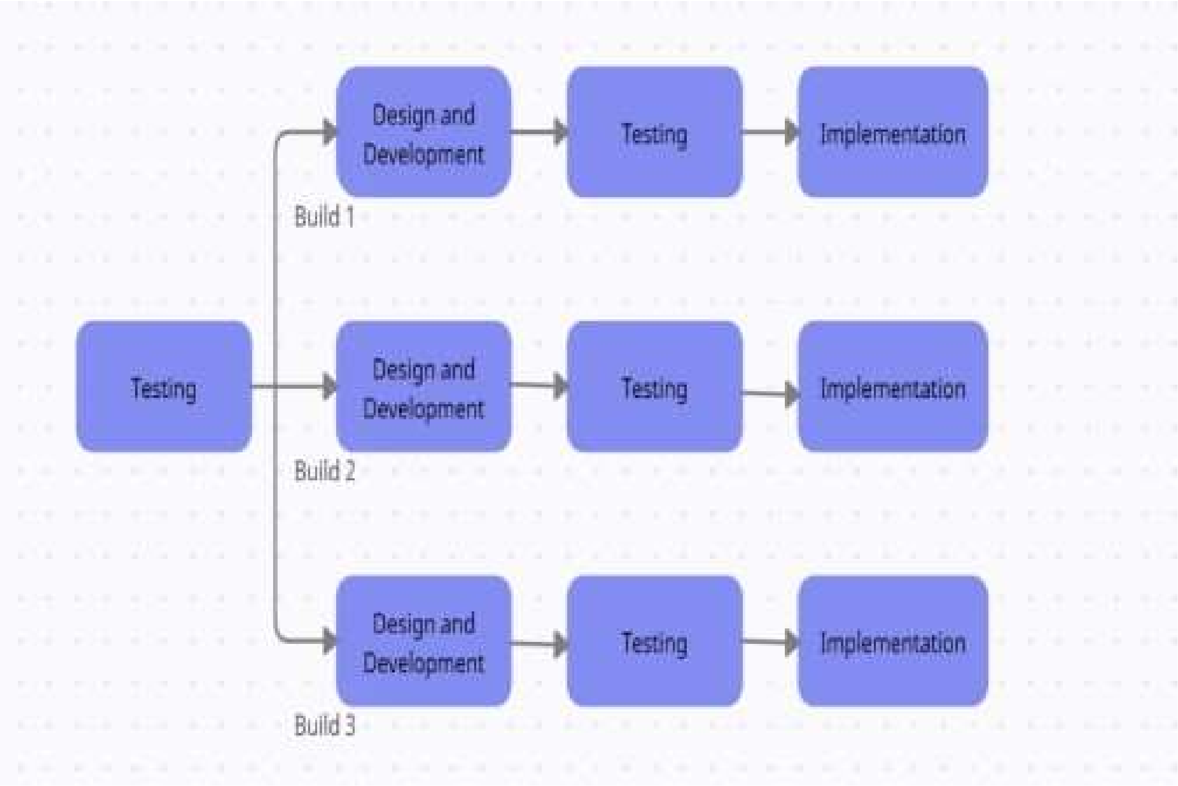


Fig: Software Process Model (i.e., Incremental Model)

4.2 Tools Used

While designing the **backend**, Django, a python-based framework seemed to be the ideal choice for our application. Since, Django provides a pre-built admin panel, it reduced the development time by around 50%.

For **frontend**, there were a really great number of choices since I was thinking of using a JavaScript based framework. At last, I stumbled upon Nuxt 3, a production ready SSR compliant JavaScript framework based on Vue 3. And for styling the pages, Tailwind CSS was incomparable to any other. Nuxt Icons and other Nuxt community-based tools are used was used.

And SQLite 3 was an ideal choice for the **database**, since it doesn't need any server and is highly compatible with Django, out of the box.

- **Frontend:** Nuxt 3 & Tailwind CSS
- **Backend:** Django
- **Database:** SQLite 3

In addition to these, various other technologies like **Django Rest Framework, Pinia Store,** etc. were also used in the technological stack.

4.3 Testing

Testing is an essential part of the software development process that ensures the quality, reliability, and functionality of a system like Razza. We used various testing mechanisms to ensure the stability of Razza service. Some of the testing mechanisms along with their documented test plans, test cases and test results are stated in Chapter V.

Chapter V: Test Results

Test Plan:

Objective: To verify the functionality and performance of key features in Razza.

1. Test Scope:
   - User registration and authentication
   - Data collection and management
   - Search functionality based on different access points
   - Integration with external databases or APIs
   - Performance under expected load conditions
2. Test Environment:
   - Operating System: Windows 10
   - Browser: Microsoft Edge
   - Test Framework: Jest for frontend, Django testing framework for backend
   - Database: SQLite 3
   - Performance Testing Tool: Apache JMeter
3. Test Strategy:
   - Unit testing for individual components and modules
   - Integration testing to ensure smooth communication between frontend and backend
   - End-to-End testing to validate critical user journeys
   - Performance testing to evaluate system response under load conditions
4. Test Schedule:
   - As per Gantt Chart

Test Cases:

Test Case 1: User Registration

- Description: Verify that a user can successfully register in Razza.
- Steps:
  a) Enter valid user information.
  b) Submit the registration form.
  c) Check if the user's information is correctly stored in the database.
- Expected Result: User registration is successful, and user data is stored correctly.

Test Case 2: User Authentication (Login)

- Description: Verify that a user can successfully login in Razza.
- Steps:
  a) Enter valid user credentials.
  b) Submit the login form.
  c) Check if the user's information is successfully logged in & redirected to dashboard.
- Expected Result: User login is successful, and user is redirected to dashboard.

Test Case 3: Search by Full Name

- Description: Verify that all details for certain individual is displayed correctly.
- Steps:
  a) Enter valid search query.
  b) Initiate the search.
  c) Verify that the correct details are displayed.
- Expected Result: Details with associated search query is displayed correctly.

Test Case 4: Search by Phone Number

- Description: Verify that all details for certain individual is displayed correctly.
- Steps:
  a) Enter valid search query.
  b) Initiate the search.
  c) Verify that the correct details are displayed.
- Expected Result: Details with associated search query is displayed correctly.

Test Case 5: Search by Citizenship Number

- Description: Verify that all details for certain individual is displayed correctly.
- Steps:
  - a) Enter valid search query.
  - b) Initiate the search.
  - c) Verify that the correct details are displayed.
- Expected Result: Details with associated search query is displayed correctly.

Test Case 6: Search by Bank A/c Number

- Description: Verify that all details for certain individual is displayed correctly.
- Steps:
  - a) Enter valid search query.
  - b) Initiate the search.
  - c) Verify that the correct details is displayed.
- Expected Result: Details with associated search query is displayed correctly.

Test Case 7: Integration with external database (e.g., PostgreSQL)

- Description: Verify the integration with an external database to fetch correct information.
- Steps:
  - a) Perform a search using a valid access point.
  - b) Verify the individual details are displayed correctly.
- Expected Result: The system integrates with an external database seamlessly.

Test Results:

Test Case 1: User Registration

- **Result**: Pass
- **Comment**: User registration is successful, and user data is stored correctly.

Test Case 2: User Authentication (Login)

- **Result**: Pass
- **Comment**: User login is successful, and user is redirected to dashboard.

Test Case 3: Search by Full Name

- **Result**: Pass
- **Comment**: Details with associated search query is displayed correctly.

Test Case 4: Search by Phone Number

- **Result**: Pass
- **Comment**: Details with associated search query is displayed correctly.

Test Case 5: Search by Citizenship Number

- **Result**: Pass
- **Comment**: Details with associated search query is displayed correctly.

Test Case 6: Search by Bank A/c Number

- **Result**: Pass
- **Comment**: Details with associated search query is displayed correctly.

Test Case 7: Integration with external database (e.g., PostgreSQL)

- **Result**: Pass
- **Comment**: The system integrates with an external database seamlessly.

Chapter VI: Conclusion

5.1 Conclusion

In conclusion, Razza is a comprehensive and efficient Online Citizen Information Processing Solution that aims to fulfill the needs of government agencies and service providers in searching and managing citizen information. By leveraging technologies such as Django and Django Rest Framework for the backend and Nuxt.js 3 for the frontend, Razza offers a robust and user-friendly platform.

Throughout its implementation, Razza demonstrates technical feasibility through the effective use of frameworks and tools that enable efficient data collection, improved decision-making, enhanced service delivery, improved accountability, and the protection of citizens' privacy. The use of Django and Django Rest Framework ensures a secure and scalable backend infrastructure, while Nuxt.js 3 provides a modern and responsive user interface.

Overall, Razza demonstrates great potential in facilitating efficient citizen information processing and management. Its successful implementation can bring about significant benefits for government agencies, service providers, and citizens alike, ultimately improving the quality of services, decision-making, and accountability in the public sector.

5.2 Future Enhancement

In the future, there are several potential enhancements that can be considered for Razza to further improve its functionality and meet evolving user needs. Some of these enhancements include:

1. **Advanced Search Capabilities**: Expand the search functionality of Razza to allow for more advanced search options, such as searching based on multiple criteria, complex queries, and fuzzy matching algorithms. This would provide users with more flexibility and accuracy when searching for citizen information.
2. **Integration with External Systems**: Enhance Razza's capabilities by integrating with other external systems, databases, or APIs. This integration would enable seamless access to additional sources of data, such as financial records, healthcare information, or educational records, providing a more comprehensive view of an individual's profile.
3. **Data Analytics and Insights**: Incorporate data analytics capabilities within Razza to derive insights and trends from the collected citizen information. By leveraging data visualization tools and advanced analytics techniques, Razza can help government agencies and service providers make data-driven decisions, identify patterns, and optimize service delivery.
4. **Mobile Application**: Develop a mobile application companion for Razza, allowing users to access the platform from their smartphones or tablets. This would provide convenience and flexibility for users who require on-the-go access to citizen information or need to perform tasks remotely.
5. **Enhanced Security Features**: Continuously enhance the security measures of Razza to ensure the protection of citizen data. Implement advanced encryption methods, two-factor authentication, and regular security audits to maintain the integrity and confidentiality of the system.
6. **Integration with Online Payment Systems**: Integrate online payment systems within Razza to enable convenient and secure payment of fees, fines, or other transactions related to citizen services. This would streamline the payment process, reducing administrative burdens and improving overall user experience.
7. **Collaboration and Communication Features**: Incorporate features that facilitate collaboration and communication between government agencies, service providers, and citizens.
8. **Continuous Improvement and User Feedback**: Establish a feedback mechanism within Razza to gather user suggestions, identify pain points, and prioritize future enhancements. Regularly update and improve the system based on user feedback and changing requirements, ensuring that Razza remains relevant and valuable over time.

References

1. Clough, J. (2015). Towards a common identity? The harmonization of identity theft laws. *Journal of Financial Crime*, *22*(4), 492–512.
2. Ram, N., Brinberg, M., Pincus, A. L., & Conroy, D. E. (2017). The Questionable Ecological Validity of Ecological Momentary Assessment: Considerations for Design and Analysis. *Research in Human Development*, *14*(3), 253–270.
3. Christensen, J. N. (2012). The hidden trillions: Secrecy, corruption, and the offshore interface. *Crime Law and Social Change*, *57*(3), 325–343.
4. Higgins, J. P. T., & Green, S. (2008). *Cochrane Handbook for Systematic Reviews of Interventions*. Wiley.
5. Executive, G. B. H. a. S. (1991). *Successful Health & Safety Management*.
6. Andersson, M., & Kjellgren, A. (2017). The slippery slope of flubromazolam: Experiences of a novel psychoactive benzodiazepine as discussed on a Swedish online forum. *The Slippery Slope of Flubromazolam:*