**Software Requirements Specification**

**for**

Expense Tracker

**Rk University**

**06/03/2024**

**Table of Contents**

**Table of Contents** [**ii**](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.23ckvvd)

**Revision History** [**ii**](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.30j0zll)

**1. Introduction** [**1**](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.ihv636)

1.1 Purpose [1](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.32hioqz)

1.2 Document Conventions [1](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.1hmsyys)

1.3 Intended Audience and Reading Suggestions [1](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.41mghml)

1.4 Project Scope [1](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.2grqrue)

1.5 References [1](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.vx1227)

**2. Overall Description** [**2**](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.3fwokq0)

2.1 Product Perspective [2](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.1v1yuxt)

2.2 Product Features [2](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.4f1mdlm)

2.3 User Classes and Characteristics [2](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.2u6wntf)

2.4 Operating Environment [2](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.19c6y18)

2.5 Design and Implementation Constraints [2](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.3tbugp1)

2.6 User Documentation [2](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.28h4qwu)

2.7 Assumptions and Dependencies [3](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.nmf14n)

**3. System Features** [**3**](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.37m2jsg)

3.1 System Feature 1 [3](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.1mrcu09)

3.2 System Feature 2 (and so on) [4](https://docs.google.com/document/d/1l2gh1ZA1VggfwiJY9eLX3g6oZnwHKzMgQ1xvjYJ3mCM/edit#heading=h.46r0co2)

**4. External Interface Requirements 5**

4.1 User Interfaces 5

4.2 Hardware Interfaces 5

4.3 Software Interfaces 6

4.4 Communications Interfaces 6

**5. Other Nonfunctional Requirements 7**

5.1 Performance Requirements 7

5.2 Safety Requirements 7

5.3 Security Requirements 7

5.4 Software Quality Attributes 7

**6. Other Requirements 8**

**Appendix A: Glossary 9**

**Appendix B: Analysis Models 10-11**

**Appendix C: Issues List 11**

# **Introduction**

## **Purpose**

*This document outlines the requirements for the development of an Expense Tracker App for the web platform using the Jsp. The app is designed to help users efficiently manage and track their expenses.*

## **Intended Audience and Reading Suggestions**

*The document is intended for developers, testers, project managers, and stakeholders involved in the Expense Tracker App project. Reading this document in its entirety is recommended for a comprehensive understanding of project requirements.*

## **Project Scope**

*The scope of the project is to design and develop a web app exclusively for users. The app enables users to record, categorize, and analyze their expenses. It offers features such as user registration, budget tracking, and expense reporting.*

# **Overall Description**

## **Product Perspective**

*The Expense Tracker is a standalone web application exclusively for the web platform. It interacts with a backend system for user authentication and data storage.*

## **Product Features**

*The app includes features such as user registration, expense entry, budget tracking, and reporting, all tailored for web users.*

## **User Classes and Characteristics**

*The app caters to individual web users who want to track their personal expenses. Users range from individuals seeking basic expense tracking to those with more complex financial management needs.*

## **Design and Implementation Constraints**

*The app is built using the jsp lib and interfaces with a backend service for data storage. It must adhere to jsp design and coding conventions.*

## **User Documentation**

*User documentation will be provided within the app to assist website users in understanding its features and functionality.*

## **Assumptions and Dependencies**

*The app assumes that users have access to compatible devices and an internet connection for certain features, such as data synchronization.*

# **System Features**

*Certainly, let's outline the system features for the Expense Tracker App along with their descriptions and priority levels:*

## **System Feature 1**

**Description and Priority:** This feature allows users to create accounts and securely log in to the app. It is of **High priority** as it is fundamental to user engagement and data security.

**Stimulus/Response Sequences:**

1. User opens the app.
2. User selects the "Sign Up" option.
3. User enters their email and creates a password.
4. User submits the registration form.
5. The system verifies the information and sends a confirmation email.
6. User receives the email, clicks the confirmation link, and is redirected to the app.
7. User logs in with the registered email and password.

**Functional Requirements:**

1. **REQ-1:** Users can register using a valid email address and a secure password.
2. **REQ-2:** Passwords must meet security criteria (e.g., minimum length, special characters).
3. **REQ-3:** The system sends a confirmation email to the user's registered email address.
4. **REQ-4:** Users can log in using their email and password.
5. **REQ-5:** Users receive an error message for invalid login attempts.
6. **REQ-6:** Users can request a password reset if they forget their password

### System Feature 2: Dashboard

**Description and Priority:** The Dashboard provides users with an overview of their financial situation and is of **High priority** as it serves as the entry point to the app.

**Stimulus/Response Sequences:**

1. User logs in to the app.
2. The system loads the Dashboard.
3. The Dashboard displays the user's total expenses, budget progress, and quick access to other features.

**Functional Requirements:**

* **REQ-7:** The Dashboard shows the total expenses for the current period.
* **REQ-8:** Users can see the progress of their budgets with visual indicators.

### System Feature 3: Expense Entry

**Description and Priority:** This feature allows users to manually enter their expenses and is of **High priority** as it's a core functionality of the app.

**Stimulus/Response Sequences:**

1. User selects the "Add Expense" option.
2. User enters the expense details (amount, date, category, notes etc).
3. User submits the expense entry.

**Functional Requirements:**

* **REQ-10:** Users can add expense details, including the expense amount, date, and category.
* **REQ-11:** Users can add optional notes and attach receipts/images.
* **REQ-12:** The system validates expense data and stores it in the cloud database.
* **REQ-13:** Users receive confirmation of the successful expense entry.

These are the first three system features of the Expense Tracker App, along with their descriptions, priority levels, stimulus/response sequences, and functional requirements. Additional system features, such as Expense Categories, Budget Tracking, and Expense History, will be documented in a similar manner to provide a comprehensive view of the app's functionality.

## *External Interface Requirements*

### *User Interfaces****Logical Characteristics:*** *The user interface for the Expense Tracker App will follow the M, ensuring a consistent and intuitive user experience. Key elements and characteristics include:*

### 

### *Software Interfaces*

***Connections with Other Software Components:*** *The Expense Tracker App integrates with various software components:*

* ***Database:*** *Utilize mySql for local data storage on the web device.*
* ***Browser:*** *The app is designed exclusively for browsers.*
* ***jsp Lib:*** *Leverage the jsp for building the user interface and managing app logic.*
* ***Third-Party Libraries:*** *Integrate relevant jsp packages and libraries for enhanced functionality (e.g., charting libraries for reports).*

***Data Items and Messages:***

* *Incoming: User data, expense details, and user authentication requests.*
* *Outgoing: Expense records, budget updates, and synchronization data.*

***Services and Communications:***

* *Utilize RESTful APIs for communication with the backend server for user registration and synchronization.*
* *Implement data sharing via secure HTTPS connections for data privacy.*

***Data Sharing Mechanism:*** *Data sharing will occur through well-defined APIs, allowing the app to retrieve and update user data securely. Data sharing will be in JSON format over HTTPS connections.*

### *Communications Interfaces*

***Communication Requirements:*** *The Expense Tracker App will require the following communication functionalities:*

* ***Email:*** *Allow users to send feedback or contact support via email.*
* ***Web Browser:*** *Open links within the app using the device's default web browser.*
* ***Network Server Communications:*** *Communicate with the backend server for user authentication and data synchronization via RESTful APIs.*
* ***Data Transfer Rates:*** *Ensure efficient data transfer with considerations for low-bandwidth scenarios.*
* ***Security:*** *Implement TLS/SSL encryption for secure data transmission.*
* ***Synchronization:*** *Implement data synchronization mechanisms to keep local and server data consistent.*
* ***Error Handling:*** *Define error codes and responses for network-related issues.*

***Communication Standards:***

* *Email communication will use standard email protocols.*
* *Web links will utilize default web browser.*
* *RESTful APIs will adhere to industry-standard HTTP/HTTPS protocols.*

***Communication Security:*** *All data transmission will be encrypted using HTTPS to ensure data security and privacy. Error handling for communication issues will provide informative user feedback.*

*These External Interface Requirements outline how the Expense Tracker App will interact with users, hardware components, and external software services to provide a seamless and secure expense tracking experience. Detailed UI specifications and API documentation should be provided separately to ensure a clear understanding of the app's behavior and functionality.*

# **Other Nonfunctional Requirements**

### *Performance Requirements*

1. ***Response Time:***
   * *The app should respond to user interactions (e.g., button clicks, data retrieval) within 1 second under normal network conditions to ensure a smooth user experience.*
2. ***Data Synchronization:***
   * *Data synchronization between the web app and the backend server should be completed within 12 seconds for an average dataset size to prevent delays in updating expenses and budgets.*
3. ***Scalability:***
   * *The system should be able to handle a growing number of users and data records. Performance should not degrade significantly as the user base and data volume increase.*

### *Safety Requirements*

1. ***Data Privacy:***
   * *The web app must adhere to data privacy regulations (e.g., GDPR) to ensure user data privacy. Personal financial data should not be shared with third parties without user consent.*
2. ***Backup and Data Recovery:***
   * *Regular automatic backups of user data should be performed to prevent data loss. Users should have the ability to recover their data in case of device loss or failure.*
3. ***Prevent Data Corruption:***
   * *Implement safeguards to prevent data corruption, particularly during data synchronization between the mobile app and the server.*

### *Security Requirements*

1. ***User Authentication:***
   * *Users must authenticate securely using email and password. Passwords should be securely hashed and stored.*
2. ***Data Encryption:***
   * *All data transmissions, including user authentication and synchronization, should be encrypted using TLS/SSL to prevent data interception.*
3. ***Authorization:***
   * *Implement role-based access control to ensure that users can only access and modify their own data.*
4. ***Secure Session Management:***
   * *Implement secure session management to prevent session hijacking and ensure that sessions automatically expire after a reasonable idle period.*

### *Software Quality Attributes*

1. ***Usability:***
   * *The web app should be user-friendly and intuitive. Users should be able to perform common tasks, such as adding expenses and viewing reports, with minimal effort.*
2. ***Reliability:***
   * *The web app should operate reliably without frequent crashes or errors. It should have a low rate of unexpected app closures.*
3. ***Maintainability:***
   * *The codebase should be well-organized and maintainable to facilitate future updates, bug fixes, and enhancements.*
4. ***Portability:***
   * *The web app should run smoothly on a wide range of web, ensuring compatibility with different screen sizes and resolutions.*
5. ***Robustness:***
   * *The app should gracefully handle unexpected scenarios, such as network interruptions or server unavailability, without crashing.*
6. ***Interoperability:***
   * *The app should be designed to work seamlessly with different versions of browsers.*

*These nonfunctional requirements ensure that the Expense Tracker web App not only performs well but also prioritizes user safety, data security, and overall software quality. They guide the development process and help in setting clear expectations for the app's behavior and performance.*

### *Other Requirements*

***Database Requirements:***

* *The app shall use localhost database for database for storing user expense data.*
* *Data in the database should be structured to efficiently support expense history and reporting features.*

***Internationalization Requirements:***

* *The app shall support multiple languages to cater to a diverse user base.*
* *All user-facing text and labels should be translatable and displayed based on the user's language preference.*

***Legal Requirements:***

* *The app must comply with all relevant data privacy and protection regulations, such as GDPR or local data protection laws.*
* *Any third-party libraries or services used in the app must comply with licensing and legal requirements.*

***Reuse Objectives:***

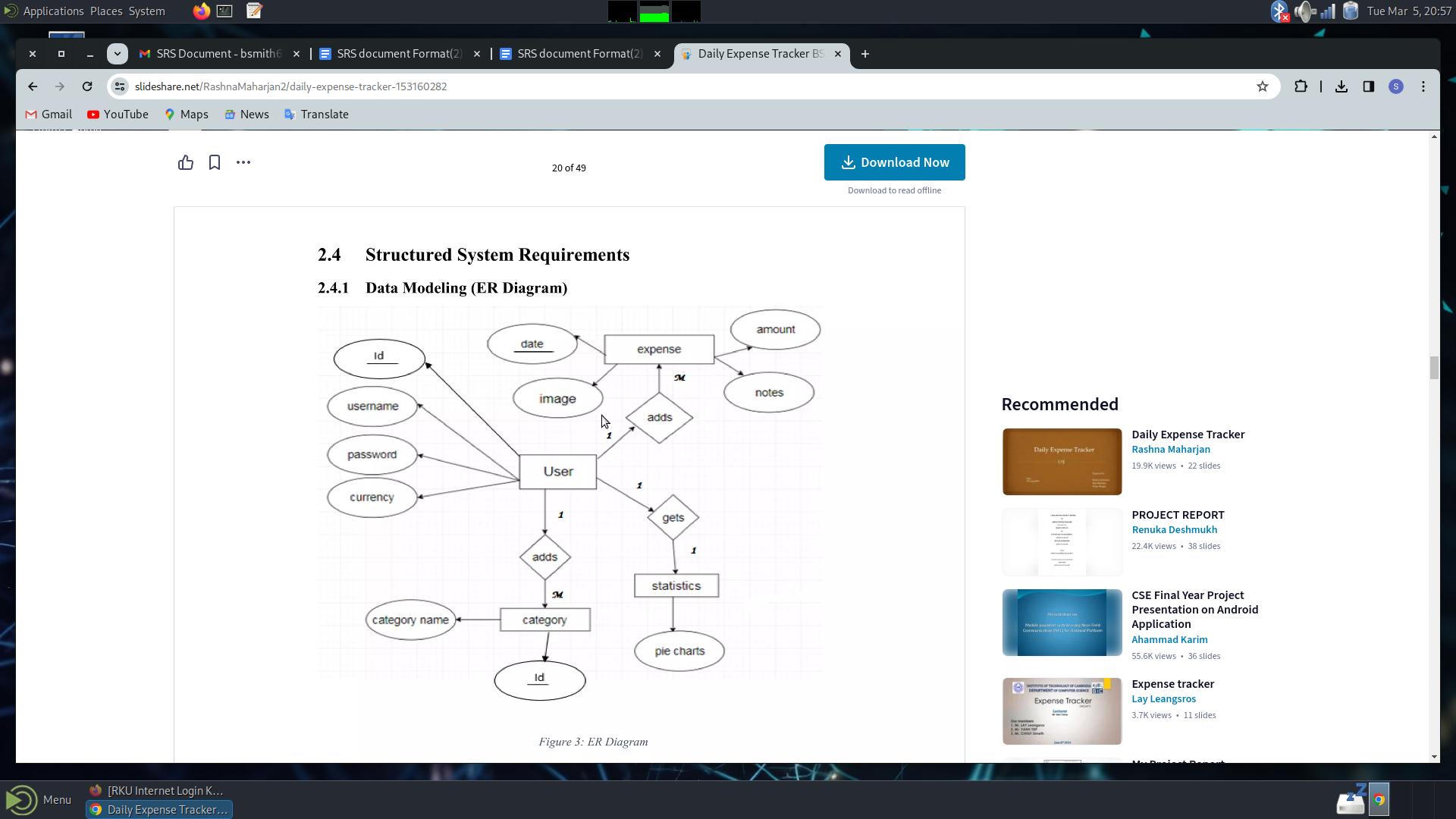
* *Code components and UI elements should be designed for reusability to facilitate the development of future projects or features.*
* *Commonly used functions and utilities should be organized into reusable modules.*

### *Appendix A: Glossary*

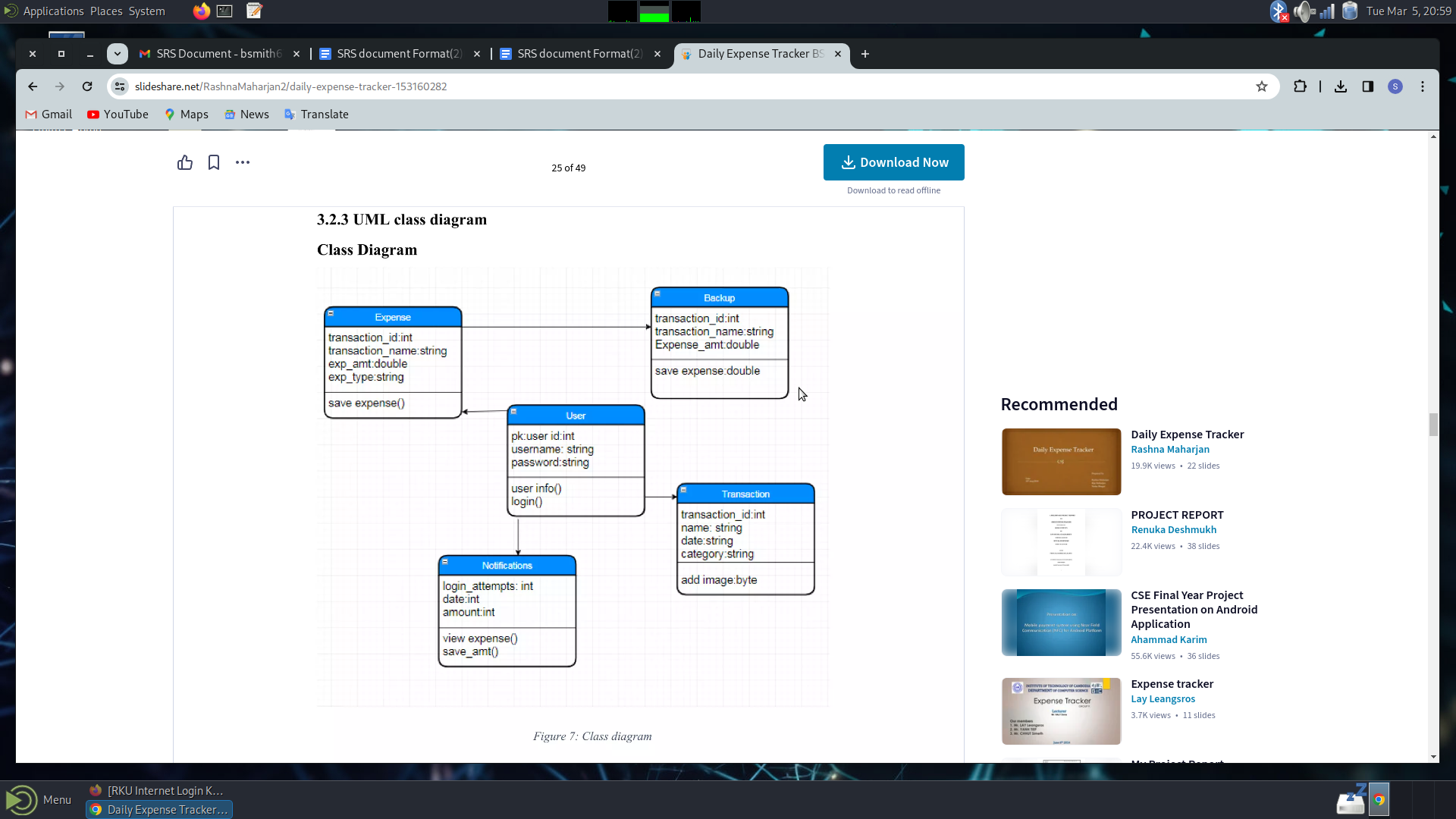
* ***API:*** *Application Programming Interface*
* ***GDPR:*** *General Data Protection Regulation*
* ***HTTPS:*** *Hypertext Transfer Protocol Secure*
* ***JSON:*** *JavaScript Object Notation*
* ***RESTful:*** *Representational State Transfer (Web services architecture)*
* ***TLS/SSL:*** *Transport Layer Security/Secure Sockets Layer*
* ***UI:*** *User Interface*
* ***URL:*** *Uniform Resource Locator*

**Appendix B: Analysis Models**

**1.Data Modeling(ER-Diagram):**

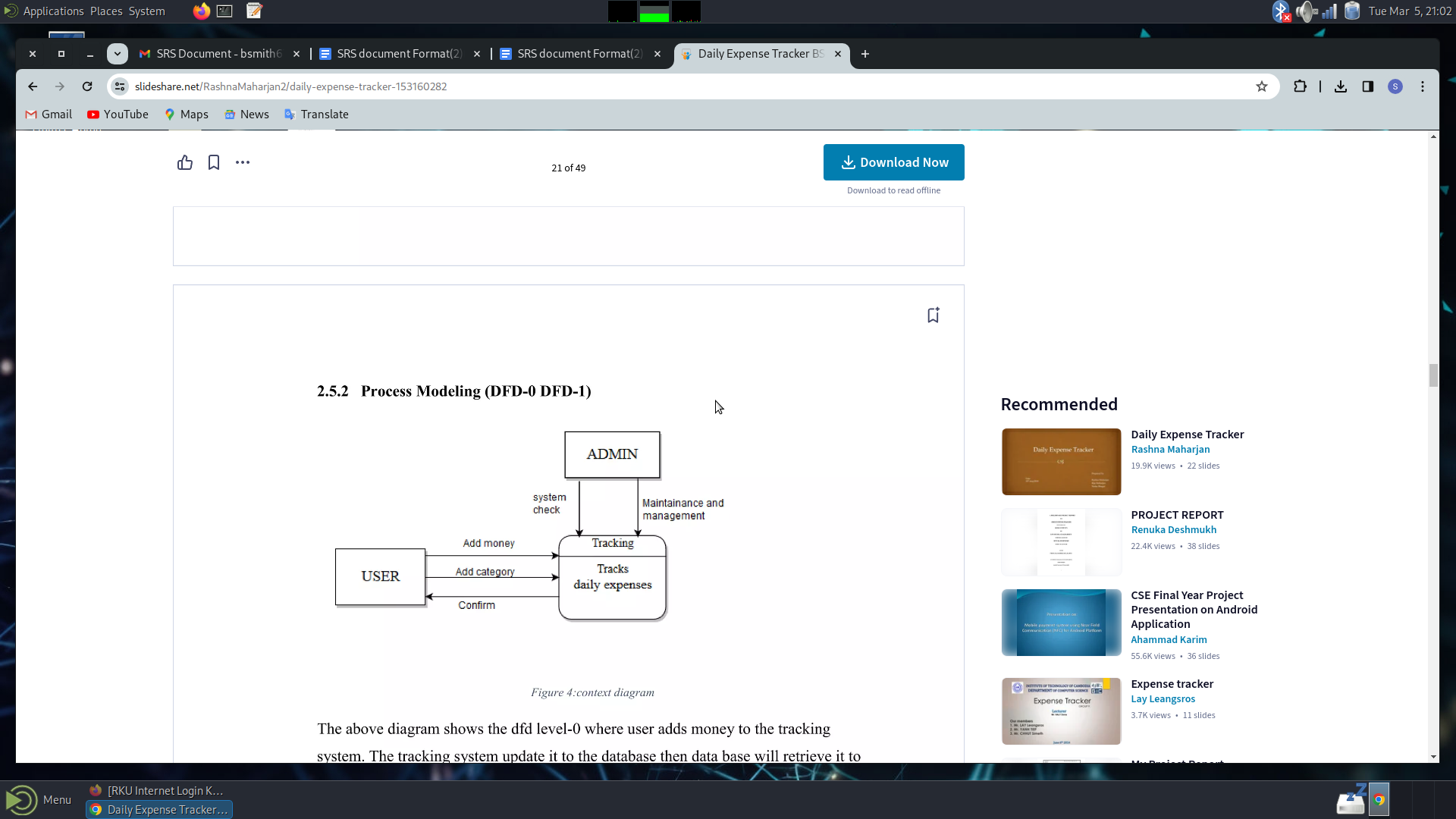
****

### *2.class Diagram:*

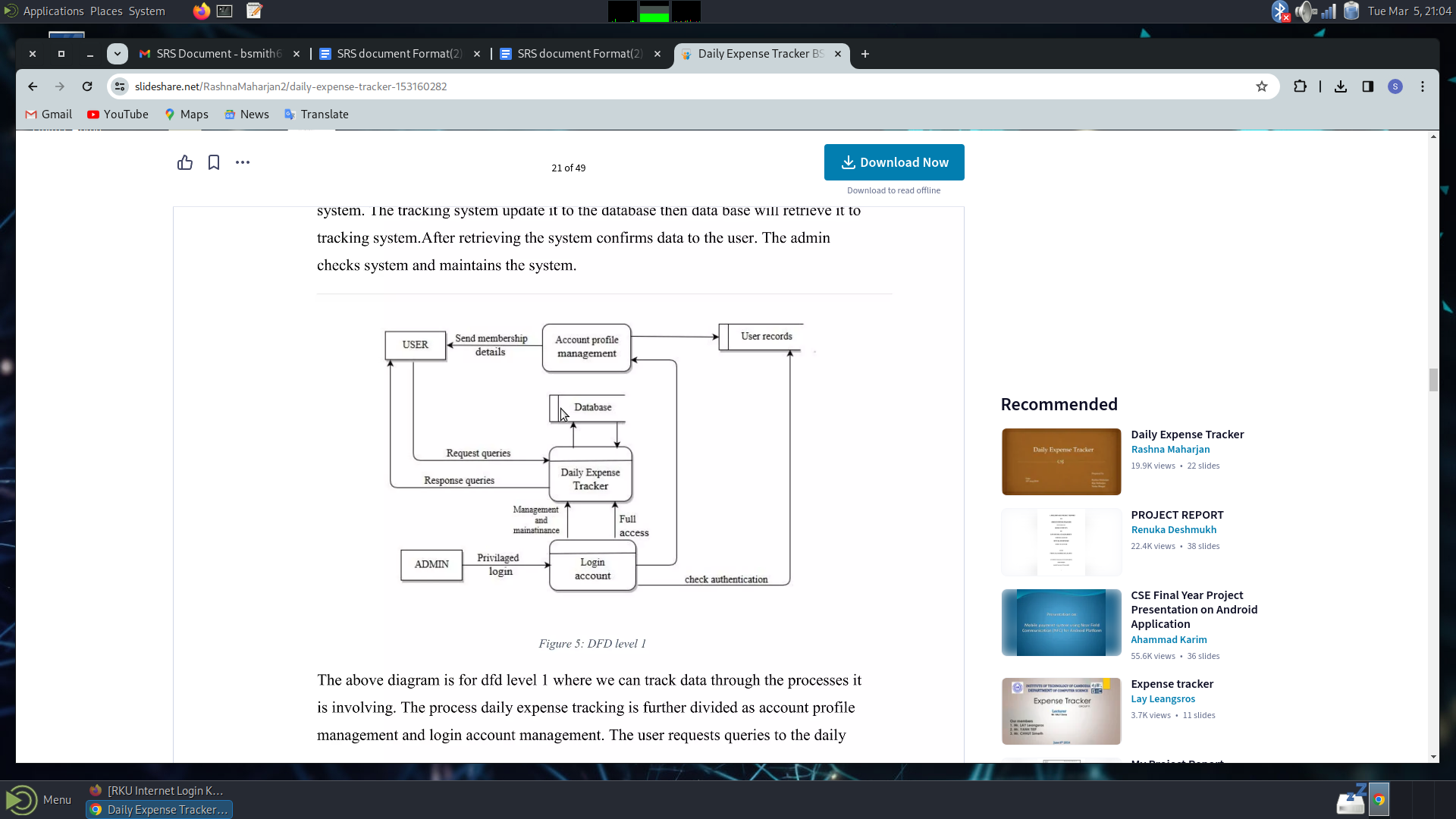


**3.Process Modelling(DFD-0 and DFD-1):**

**DFD-level 0**

****

**DFD:level 1**

****

### *Appendix C: Issues List*

*There is no issue in the project till now.*