**IN2901 – Software Development Project**

**Project Proposal**

**Level 2**

**Expense Tracker System**

TODO



Faculty of Information Technology University of Moratuwa

2023

IN2901 – Software Development Project **Contents**

[1. Introduction 1](#_Toc11506)

[2. Problem in brief 1](#_Toc11507)

[2.1. Problem Statement: 1](#_Toc11508)

[2.2. Background: 1](#_Toc11509)

[3. Aim and Objectives 2](#_Toc11510)

[3.1. Aim 2](#_Toc11511)

[3.2. Objectives 2](#_Toc11512)

[4. Proposed Solution 2](#_Toc11513)

[4.1. Nature of the Solution: 3](#_Toc11514)

[4.2. Technologies to be Adapted: 3](#_Toc11515)

[4.3. Feasibility of Implementation: 3](#_Toc11516)

[4.4. Capability to Solve the Problem: 3](#_Toc11517)

[4.5. Availability of Resources: 3](#_Toc11518)

5. Timeline 5

References

Signatures of the Group Members

Supervisors’ Declaration

II

# 1. Introduction

Expense tracking is a crucial part of financial management, helping you understand your spending habits and make informed decisions. Common methods for tracking expenses include manual methods like pen and paper or spreadsheet software, as well as digital solutions such as mobile apps, online platforms, and dedicated software. Key elements to record include the date, description, category, amount, payment method, and receipts.

By consistently tracking your expenses, you can gain insights into where your money goes, allowing you to create a budget, cut unnecessary expenses, and work towards your financial goals. Whether you prefer traditional methods or modern digital tools, finding a tracking method that suits your lifestyle is essential for effective financial management...

# 2. Problem in brief

## 1.1. Problem Statement:

The problem with expense tracking lies in incomplete data, forgotten transactions, and inconsistent categorization. Manual methods are time-consuming, while relying solely on bank statements lacks detail. Privacy concerns and the risk of overspending through mobile apps further complicate the process.

1.2. Background:

Expense tracking is vital for financial management. However, it can be challenging due to various factors. Manual tracking with pen and paper or spreadsheets often results in incomplete records and forgotten expenses. Bank statements offer convenience but lack transaction details and categorization.

Additionally, the increasing use of mobile apps raises privacy concerns, as these apps may access sensitive financial information. Furthermore, some individuals find it easier to overspend when using digital tracking methods. Balancing convenience with accuracy and privacy is essential for effective expense tracking.

# 3. Aim and Objectives

3.1. Aim: The aim of this project is to develop an expense tracking system using ASP.NET, React, and MS SQL for effective financial management.

3.2. Objectives:

- Explore and analyze technologies, including ASP.NET, React, and MS SQL, suitable for building an expense tracker.

- Design and develop a user-friendly and secure expense tracking system that allows users to manage their finances efficiently.

-Evaluate the proposed solution's usability, accuracy, and security to ensure it meets the project's goals and user expectations.

-Prepare comprehensive documentation that includes system architecture, user guides, and technical specifications for future reference and maintenance.

# 4. Proposed Solution

The proposed solution is to use digital expense tracking apps, which streamline expense monitoring with features like automatic categorization, customizable categories, and budget alerts. Regularly reviewing and reconciling transactions within the app ensures completeness and accuracy. Prioritizing apps with strong security measures safeguards your financial data and privacy. Additionally, enabling regular data backups prevents data loss and allows easy access for reference or device changes. This solution simplifies expense tracking, enhances data accuracy, protects privacy, and provides better financial control.

## 4.1.Nature of the Solution:

4.1.1. Digital Expense Tracking Apps: Utilize reputable expense tracking apps like Mint, YNAB, or PocketGuard. These apps offer automatic categorization, receipt scanning, and bank account synchronization, ensuring accurate and convenient expense monitoring.

4.1.2. Regular Check-Ins: Set aside time regularly, such as weekly or monthly, to review and reconcile your expenses within the app. This habit helps prevent forgotten transactions and ensures completeness.

4.1.3. Customized Categories: Personalize expense categories in the app to match your spending habits precisely. This reduces inconsistencies in categorization and provides a clearer overview of your financial behavior.

4. 1.4 Privacy Protection :Prioritize apps with strong security measures, including encryption and multi-factor authentication, to safeguard your financial data. Read privacy policies carefully and grant app permissions selectively.

4.1.5. Budget Alerts: Enable budget alerts within the app to receive notifications when you approach or exceed spending limits in specific categories. This helps control overspending.

4.1.6. Regular Backups: Ensure your expense tracking data is regularly backed up to prevent data loss and enable easy access in case of app issues or device changes.

By implementing these solutions, you can streamline expense tracking, maintain data accuracy, protect your privacy, and gain better control over your finances. Remember to choose an app that aligns with your preferences and needs to maximize the effectiveness of your expense tracking efforts.

## 4.2.Technologies to be Adapted:

4.2.1. Front-end Technologies: React

4.2.2. Backend Technologies: ASP .net

4.2.4. Database: MS SQL

## 4.3. Feasibility of Implementation:

4.3.1. ASP.NET:

4.3.1.1. Mature Framework: ASP.NET is a well-established and mature framework developed by Microsoft. It offers robust features for building scalable and secure web applications.

4.3.1.2. Security: ASP.NET provides built-in security mechanisms and libraries to protect against common web application vulnerabilities, making it a reliable choice for managing sensitive financial data.

4.3.1.3. Scalability: ASP.NET applications can easily scale both vertically and horizontally, ensuring that your expense tracker can handle increased user loads as it grows.

4.3.2 React:

4.3.2.1. Rich User Interface: React is known for creating dynamic and responsive user interfaces. This is essential for providing users with an interactive and enjoyable experience while tracking expenses.

4.3.2.2. Modularity: React's component-based architecture promotes code reusability and maintainability, making it easier to develop and extend your application over time.

4.3.2.3. Community and Resources: React has a large and active community, which means you can find ample documentation, libraries, and resources to help with development.

4.3.3. MS SQL (Microsoft SQL Server):

4.3.3.1. Reliability: MS SQL is a highly reliable relational database management system. It is used in many mission-critical applications, including financial systems.

4.3.3.2. Data Integrity: MS SQL offers robust data integrity features, ensuring that your financial data remains consistent and accurate.

4.3.3.3. Scalability: MS SQL can scale both vertically and horizontally to accommodate increasing data loads.

## 4.4.Capability to Solve the Problem:

4.4.1. Technical Proficiency: The development team's expertise in ASP.NET, React, and MS SQL is essential. Skilled developers with knowledge of these technologies can efficiently design, build, and maintain the expense tracker.

4.4.2. Project Planning: Effective project planning, including defining requirements, timelines, and milestones, is crucial for problem-solving. A well-structured project plan helps manage complexity and ensures that the solution aligns with the problem statement.

4.4.3. Problem Analysis: Understanding the specific requirements and constraints of the expense tracker project is vital. Clear problem analysis enables the team to identify potential challenges and devise effective solutions.

4.4.4. Communication: Effective communication among team members, stakeholders, and end-users is key to resolving issues promptly. It ensures that everyone involved in the project is on the same page and can collaborate effectively.

4.4.5. Resource Management: Proper allocation of resources, including human resources, time, and budget, is essential for successful problem-solving. Adequate resources enable the team to address challenges as they arise.

4.4.6. Testing and Quality Assurance: Rigorous testing and quality assurance processes help identify and rectify issues early in the development lifecycle. This ensures that the final solution is reliable and free from critical defects.

4.4.7. Flexibility and Adaptability: Problem-solving often involves adapting to changing requirements or unexpected challenges. A team's ability to adjust the project plan and technology stack as needed is essential for success.

4.4.8. Security and Compliance: When handling financial data, ensuring security and compliance with relevant regulations is paramount. The capability to implement robust security measures and adhere to compliance standards is critical.

4.4.9. Documentation: Comprehensive documentation of the solution's architecture, codebase, and deployment procedures aids in problem-solving and maintenance, as it provides a clear reference for the development team.

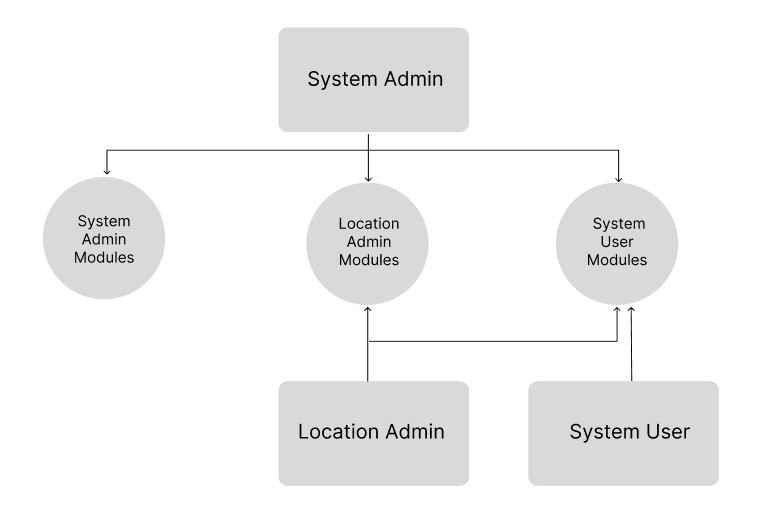
4.4.10. User Feedback: Listening to user feedback and incorporating user preferences into the solution can enhance problem-solving capabilities. A user-centric approach helps address usability issues and improve the overall experience.

4.4.11. Continuous Learning: Staying updated with the latest developments in ASP.NET, React, MS SQL, and best practices in software development is crucial. Continuous learning enables the team to adapt and solve evolving challenges effectively.

## 4.5. Availability of Resources:

The availability of resources is a critical factor in the successful execution of any project. Resources encompass a wide range of elements, including financial, human, technological, and physical assets. Having adequate resources ensures that a project can be completed on time and within budget while meeting its objectives. This availability extends to financial funding, skilled personnel, necessary equipment and technology, and physical infrastructure. Effective resource allocation and management are essential to optimize efficiency and productivity, allowing organizations to deliver high-quality outcomes and achieve their strategic goals. Therefore, careful planning and allocation of resources are fundamental aspects of project management and organizational success.

Figure:



**TimeLine**

**1st Week**

Project Initiation

Project Planning

Study and Development

Testing and Quality Assurance

Project Closure

* Define project objectives, scope, and deliverables.
* Formulate the project team and assign roles and

Requirements Gathering

responsibilities.

* Conduct an initial project kickoff meeting with

the client.  -Collaborate with the client to gather detailed

requirements for the web-based booking system.

-Document user stories, use cases, and functional

requirements.

* Develop a detailed project plan

including tasks, dependencies, and timelines.

Design and Architecture

* Identify and allocate necessary resources.

- Design the user interface (UI) and user

experience (UX) of the web-based system.

- Define the system's architecture and database structure.

- Create wireframes and prototypes for client review.

- Learn about the Technologies relevant to the

project.

* + Begin the development phase, implementing

the front- end and back-end of the system.

* + Develop real-time updates functionality and

user authentication.

* Conduct regular code reviews and testing. - Conduct comprehensive testing, including unit testing,

integration testing, and user acceptance testing (UAT).

Deployment and User Training

- Identify and rectify any bugs or issues.

* Deploy the web-based system to a production - Ensure data security and user data

environment. protection measures are in place.

* Conduct user acceptance testing to validate the

Documentation and Final Review

system’s functionality.

- Prepare comprehensive documentation, including user

manuals and maintenance guides.

* Obtain final approval from Inova IT - Conduct a final review of the system with Inova IT

system. system to ensure all requirements are met.

* Hand over the system, documentation, and any - Address any last-minute feedback or adjustments.

relevant assets.

* Evaluate the project's overall success and areas

for improvement.

### References

1. ReactJS documentation [(https://react.dev/blog/2023/03/16/introducing-react-dev)](https://react.dev/blog/2023/03/16/introducing-react-dev)
2. Material UI documentation ([https://mui.com/material-ui/getting-started/)](https://mui.com/material-ui/getting-started/)
3. NodeJS documentation ([https://nodejs.org/en/docs)](https://nodejs.org/en/docs)
4. Create Rest API with Node.js and Express (https://blog.postman.com/how-to-create-a-rest-apiwith-node-js-and-express/)
5. Version control with GitHub [(https://docs.github.com/en)](https://docs.github.com/en)
6. Oracle Database documentation ([https://docs.oracle.com/en/database/oracle/oracledatabase/index.html)](https://docs.oracle.com/en/database/oracle/oracle-database/index.html)
7. Oracle best practices guidance ([https://www.oracle.com/applications/modern-best-practice/)](https://www.oracle.com/applications/modern-best-practice/)
8. UI / UX fundamentals [(https://medium.com/@nho\_rah/ui-ux-fundamentals-6f8feea0f6ba](https://medium.com/@nho_rah/ui-ux-fundamentals-6f8feea0f6ba) )
9. UI /UX principles and best practices ([https://yatilabs.medium.com/the-ultimate-guide-on-ui-uxdesign-its-principles-best-practices-cf03061c2d9b)](https://yatilabs.medium.com/the-ultimate-guide-on-ui-ux-design-its-principles-best-practices-cf03061c2d9b)
10. Connect oracle dB with node.js application

[(https://www.oracle.com/database/technologies/appdev/quickstartnodeonprem.html)](https://www.oracle.com/database/technologies/appdev/quickstartnodeonprem.html)

1. Node.js Oracle Crud example [(Node.js Oracle CRUD Example: Database Connection & SQL](https://www.techiediaries.com/node-oracle-database-crud/)

[Queries | Techiediaries)](https://www.techiediaries.com/node-oracle-database-crud/)

1. WebSocket introduction ([https://www.geeksforgeeks.org/what-is-web-socket-and-how-it-isdifferent-from-the-http/)](https://www.geeksforgeeks.org/what-is-web-socket-and-how-it-is-different-from-the-http/)
2. How web socket works (YouTube - [https://www.youtube.com/watch?v=pnj3Jbho5Ck)](https://www.youtube.com/watch?v=pnj3Jbho5Ck)
3. Socket.io documentation ([https://socket.io/docs/v3)](https://socket.io/docs/v3)
4. Socket.IO with Nodejs + Express medium article ([Socket.IO with NodeJS + Express. Nowadays, most web developers want to… | by Sude Kılıç | Koçfinans Tech | Medium)](https://medium.com/kocfinanstech/socket-io-with-node-js-express-5cc75aa67cae)

### Signatures of the Group Members

|  |  |  |
| --- | --- | --- |
| **Index No** | **Name** | **Signature** |
| 214092B | Jayatissa P.B.N. |  |
| 214135L | Munasinghe M.A.K.L. |  |
| 214122U | Madhushan H.M.N. |  |
| 214028L | Bandara H.M.A.P. |  |
| 214104R | Kaumadi G.S.B.D.D. |  |

### Supervisors’ Declaration

I hereby declare that I have checked this project, and, in my opinion, this project is adequate in terms of scope and quality.

1. Name of Supervisor : Mrs. Chandimali M.N.

Designation : Lecture, Department of Information Technology

Date :  **/**   **/** 2023

Signature :

Any further comments :

1. Name of Supervisor : Dr.(Ms) Ganegoda G.U.

Designation : Senior Lecture, Department of Interdisciplinary Studies

Date : **/**   **/** 2023

Signature :

Any further comments :