



INTRODUCTION TO PROGRAMMING 1

GROUP 10

DESIGN AND DEVELOPMENT OF AN INTEGRATED EXPENSE TRACKER WEB BASED SYSTEM: IN CASE OF UNIVERSITY STUDENTS

Group Members	ID
1. Lalisa Tamene	ETS0847/16
2. Leoul Zerihun	ETS0852/16
3. Meklit Yemane	ETS0925/16
4. Nebiyu Yalemgeta	ETS1116/16
5. Negasi Berihu	ETS01117/16
6. Robel Wondwossen	ETS1187/16

Submitted To: Mr. Jerusalem

Submission Date: 21/1/2026

Table of Contents

1.Overview.....	1
1.2 Context.....	1
1.3 Motivation.....	2
2. Overall Description.....	3
2.1 Problem Statement.....	3
2.2 Objective.....	3
2.3 Significance.....	4
2.4 Beneficiaries.....	5
3. Software Requirements Specification (SRS).....	6
3.1 Functional Requirement.....	6
3.2 Non-Functional Requirement.....	7
3.4 System Architecture.....	6
3.5 Assumptions/Dependencies.....	7
3.6. Existing System Overview.....	7
3.7 Usecase Diagram.....	8
3.7.1 Use Case Diagram of Existing System.....	8
3.8 Proposed System.....	9
3.8.1 Use Case Diagram of Proposed System.....	10

1. Introduction

1.1 Overview

This document outlines a project, an integrated expense tracker designed for university students. Built with HTML, CSS, and JavaScript, it emphasizes simplicity, modularity, and future scalability. Sections are structured for clarity, with a focus on student financial challenges. The purpose of the integrated spending tracker is to enable students to efficiently manage their own finances. The method helps university students deal with common issues including low income, erratic spending habits, and the need for financial discipline by offering simple tools for budgeting, tracking expenses, and setting savings goals.

Through modern web tools, access across devices becomes possible. Where screens differ, layout flexibility maintains function. Updates proceed without disruption due to structured components. Simplicity follows from separation of parts

A straightforward dashboard meets students with up-to-date snapshots of their spending targets, while delivering alerts without delay. Clarity shapes every part of the design, reducing mental effort so handling funds feels more manageable. Tracking milestones, reviewing expense patterns, alongside sorting purchases supports wiser financial choices over time.

1.2 Context

1.2.1 Background and Setting

Budget limits shape student life across Ethiopian universities. Despite fixed study schedules, money troubles persist because prices climb while pocket funds stay small. Some rely on side work; others wait for help from home that arrives late or not at all. Daily choices involve meals, bus fares, course materials - small needs piling up fast. Thousands carry grades and grocery lists at once.

Where most budgeting still happens through paper logs or simple digital sheets - methods open to mistakes and delays - a different path becomes visible Our project

emerges in this context as a lightweight, offline-capable web app, inspired by global tools like Mint or YNAB but tailored for Ethiopian students using local currency (ETB) and relevant categories (e.g., bus transport, photocopy costs).

1.3 Motivation

1.3.1 Driving Forces

The motivation stems from personal experiences: As university students, our team has witnessed peers overspending on non-essentials, leading to mid-month shortages or debt. A 2025 survey by the Ethiopian Students' Association highlighted that 65% of students struggle with budgeting, often due to lack of tools suited to their lifestyle.

We were inspired by:

- **Accessibility:** Many students can't afford premium apps; it is free and browser-based.
- **Education:** Promoting financial literacy early, aligning with UN SDG 4 (Quality Education) and SDG 8 (Decent Work).
- **Innovation:** Starting as a front-end project, it demonstrates modular design (e.g., reusable components like header.html) while planning for AI-driven insights.
- **Local Relevance:** Categories like "Transport (Taxi/Bajaj/Bus)" address urban Ethiopian challenges, where public transport costs can eat 20-30% of a student's budget.

Ultimately, it motivates us to build a tool that empowers students to achieve financial independence, reducing stress and improving academic focus.

2. Overall Description

2.1 Problem Statement

2.1.1 Challenges and Gaps

University students in Ethiopia, particularly in Addis Ababa, face acute financial management issues:

- **Limited Income:** Average monthly allowance is 2000-5000 ETB, insufficient against inflation (e.g., food prices up 15% in 2025).
- **Tracking Difficulties:** Manual methods (notes, Excel) lead to inaccuracies, forgotten expenses, and no visualizations.
- **Overspending Risks:** Without alerts, students exceed budgets on categories like entertainment (e.g., coffee outings at 50-100 ETB each).
- **Existing Tools Gaps:** Apps like Google Sheets lack mobility; premium tools (e.g., Pocket Guard) require subscriptions and don't support ETB natively.
- **Digital Divide:** Rural-origin students may have limited app experience, needing simple, offline tools.

The core problem: No affordable, student-centric, browser-based tracker that provides real-time summaries, goal tracking, and local relevance.

2.2 Objective

2.2.1 Goals of the Project

General Objective: To develop a responsive, front-end budget tracking application that helps Ethiopian university students monitor expenses, set goals, and gain financial awareness.

Specific Objectives:

1. **User-Friendly Interface:** Create modular pages (e.g., dashboard.html, add-transaction.html) for easy transaction entry and viewing.

2. **Data Management:** Implement local Storage for persistent tracking of income/expenses in ETB.
3. **Insights and Alerts:** Provide summaries, progress bars, and notifications (via alertSystem.js) to prevent overspending.
4. **Scalability:** Design with reusable components (e.g., summary-cards.html) for future backend (PHP/MySQL) and AI features.
5. **Accessibility:** Ensure responsive design (responsive.css) for mobile use in low-bandwidth areas.
6. **Education Impact:** Foster financial literacy through simple visualizations, targeting a 20% reduction in student overspending.

These objectives will be measured via user testing with 50 AASTU students by mid-2026.

2.3 Significance

Impact and Value

Its significance lies in empowering Ethiopia's youth:

- **Individual Level:** Reduces financial anxiety, improving mental health and academic performance (linked to a 15% grade drop from stress, per studies).
- **Societal Level:** Promotes responsible spending, aligning with Ethiopia's youth unemployment reduction goals.
- **Economic Level:** Encourages savings, potentially increasing student entrepreneurship in Ethiopia's startup scene.
- **Educational Level:** Serves as a capstone project for our team, demonstrating software engineering principles (modular JS like app.js).
- **Broader Reach:** Open-source potential could benefit 500,000+ Ethiopian students, with localization for Amharic.
- **Innovation:** Lays groundwork for AI (e.g., spending predictions), contributing to fintech growth in Africa.

Overall, it bridges the gap between technology and student needs, fostering a financially savvy generation.

2.4 Beneficiaries

Target Users and Stakeholders

- **Primary Beneficiaries:** University students in Ethiopia (e.g., AASTU, AAU, Gonder, Haramaya), especially those with limited budgets (18-25 years old). They gain tools for daily tracking.
- **Secondary Beneficiaries:** Parents/families providing allowances, who benefit from reduced financial requests.
- **Institutions:** Universities can integrate our expense tracker into financial literacy workshops.
- **Developers/Team:** Our group of 6 gains practical experience in front-end development.
- **Society:** Contributes to Ethiopia's digital inclusion, benefiting low-income youth in Addis Ababa and beyond.

Estimated Reach: 10,000 users in first year via free distribution

2.5 Operating Environment

The application is built for broad accessibility and cross-platform functionality:

Web Browsers: It works seamlessly with the 2026 versions of Google Chrome, Mozilla Firefox, and Microsoft Edge.

Hardware: Designed to be responsive (responsive.css), it is optimized for use on both desktop and mobile devices, making it suitable for smartphones typically owned by students.

Network: It can function offline, which is essential for students in areas with low bandwidth or limited data availability.

3. Software Requirements Specification (SRS)

3.1 Introduction

The project is a web-based expense tracker. This effort centers on learners handling money matters. Its role involves supporting budget choices through organized tracking online. What it includes now is an interface design, later to connect with data systems behind the scenes.

3.2 Functional Requirements:

Functional requirements define the core behaviors and capabilities the Savr system SHALL provide.

- **User Authentication:** Simulate login/register (login.html, register.html) with local Storage sessions.
- **Transaction Management:** Add/edit/delete transactions (add-transaction.html) with type, category, amount, date, description.
- **Dashboard Views:** Display summaries, lists, progress (dashboard.html, transaction-list.html).
- **Goal Setting:** Set/track savings goals (goals.html).
- **Alerts:** Show notifications for budget limits.

3.3 Non-Functional Requirements:

Non-functional requirements specify the quality attributes, constraints, and performance criteria that Savr must satisfy. These are critical for system acceptance and long-term success.

- **Performance:** Load <2s on standard browsers.
- **Usability:** Intuitive UI, responsive for mobiles (responsive.css).

- **Security:** Basic data privacy via localStorage (future: encryption).
- **Compatibility:** Chrome, Firefox, Edge (2026 versions).
- **Reliability:** 99% uptime (offline capable).

3.4 System Architecture

Modular front-end (HTML/CSS/JS).

Data flow: User input → app.js → local Storage → UI render.

3.5 Assumptions/Dependencies

No server; assumes modern browser. Dependencies: Vanilla JS only.

3.6 Existing System Overview

Current Solutions Analysis

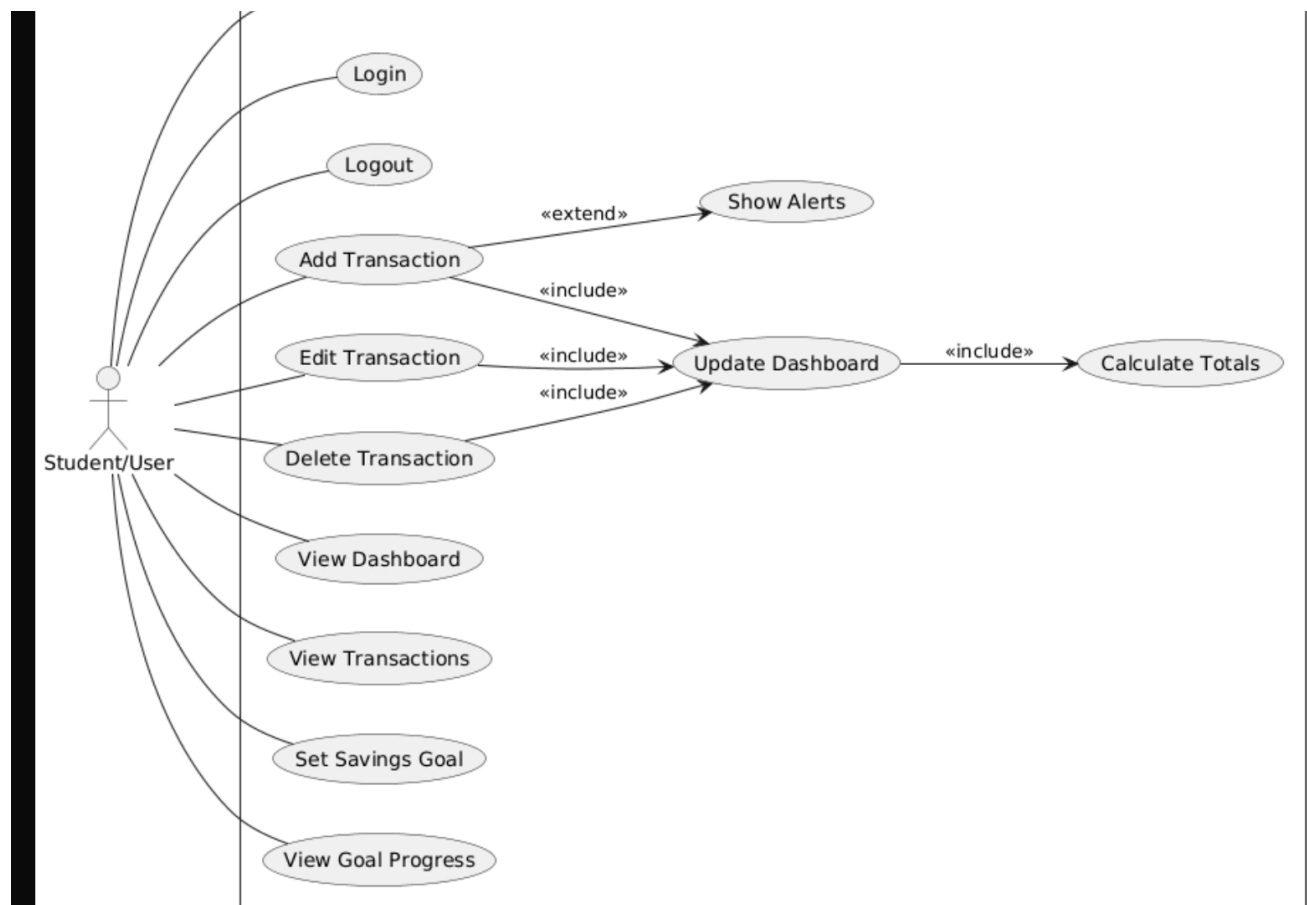
Existing systems include:

- **Manual Methods:** Notebooks/Excel – Cheap but manual, no alerts, prone to loss.
- **Mobile Apps:** Mint, Wallet (global) – Feature-rich but subscription-based, data-heavy, not ETB-focused. Local apps like CBE Mobile lack budgeting.
- **Web Tools:** Google Sheets templates – Free but clunky, no mobile optimization.
- **Limitations:** High data usage, privacy concerns, no student-specific categories (e.g., "Print & Photocopy"). In Ethiopia, 40% of students use basic phones, limiting app access.
- **Gaps:** No offline, modular, free tool for Ethiopian contexts, leading to inefficient tracking.

3.7 Use Case Diagram

3.7.1 Use Case Diagram of Existing System

This visual outlines key functions accessible to students in the existing setup. Interaction with tools like tracking transactions, accessing summaries, defining targets, or reviewing notifications forms part of daily usage patterns. What emerges is a map showing pathways between individuals and system resources. Purpose lies in clarifying boundaries of function alongside movement through tasks.



3.8 Proposed System

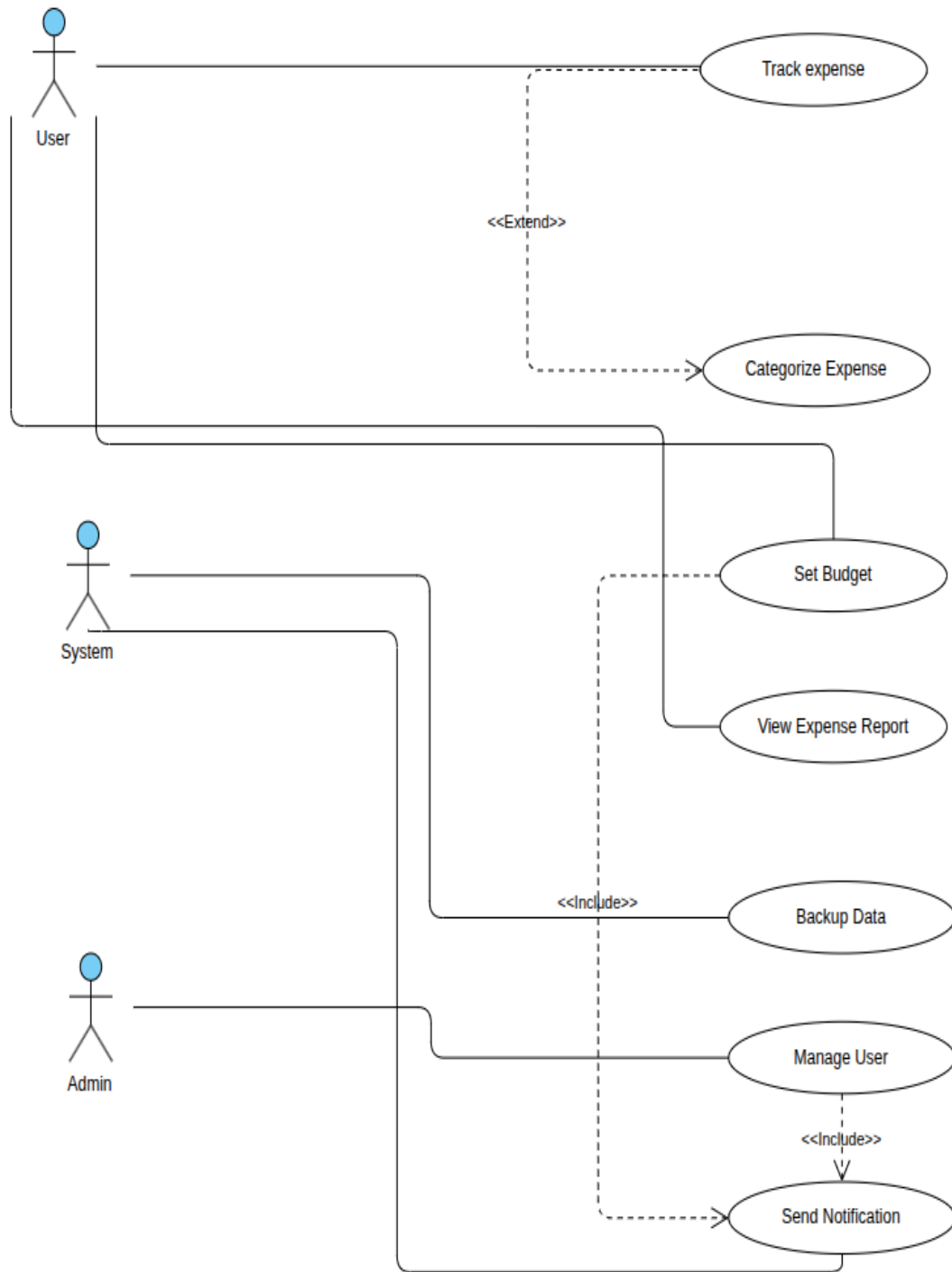
Architecture and Features

The proposed system is a modular front-end app:

- **Components:** Reusable HTML (e.g., header.html), CSS (style.css), JS (app.js for orchestration).
- **Key Modules:** budgetCalculator.js (summaries), uiController.js (rendering), alertSystem.js (notifications).
- **Workflow:** User logs in → Adds transaction → Views dashboard → Sets goals → Receives alerts.
- **Innovations:** Offline via localStorage, responsive design, ETB support.
- **Future:** Integrate PHP/MySQL for multi-user, AI for predictions.
- **Advantages:** Free, simple, scalable – addresses existing gaps with student focus.

3.8.1 Use Case Diagram of Proposed System

Here comes an updated view of how the system operates, now shaped around distinct roles for individuals and those in charge. Where one group handles budgeting, monitors spending patterns, sets targets, another ensures categories are defined properly and records stay accurate. A revised structure shows wider capability - designed not just for narrow tasks but deeper coordination across different responsibilities. Flexibility increases when functions align with user types, allowing smoother handling of diverse financial activities without overlap. Clarity emerges through separation of duties, making oversight possible without limiting personal control over money plans.



Reference

1. *Dennis, B. H. Wixom, and R. M. Roth, Systems Analysis and Design, EMEA Edition. John Wiley & Sons, 2019.*
2. *J. A. Hoffer, J. F. George, and J. S. Valacich, Modern systems analysis and design. 1995.*
3. *I. Jacobson, Object Oriented Software Engineering: a use case driven approach. 1995.*