**Software Requirements Specification**

**Minimalist Finance Tracker for College Students**

Version 1.0 approved

Prepared by

Mariyam Noor ul ain NUM-BSCS-2022-25

Eshna Eman NUM-BSCS-2022-51

Namal university Mianwali

06/05/2025

Table of Contents

[1. Introduction 4](#_Toc200929939)

[1.1 Purpose 4](#_Toc200929940)

[1.2 Document Conventions 4](#_Toc200929941)

[1.3 Intended Audience and Reading Suggestions 4](#_Toc200929942)

[1.4 Project Scope 4](#_Toc200929943)

[1.5 References 5](#_Toc200929944)

[2. Overall Description 5](#_Toc200929945)

[2.1 Product Perspective 5](#_Toc200929946)

[2.2 Product Features 5](#_Toc200929947)

[2.3 User Classes and Characteristics 5](#_Toc200929948)

[2.4 Operating Environment 6](#_Toc200929949)

[2.5 Design and Implementation Constraints 6](#_Toc200929950)

[2.6 User Documentation 7](#_Toc200929951)

[2.7 Assumptions and Dependencies 7](#_Toc200929952)

[3. Functional Requirements 7](#_Toc200929953)

[3.1 Manual Expense Entry 7](#_Toc200929954)

[3.2 Budget Management 8](#_Toc200929955)

[3.3 Visual Reports & Analytics 9](#_Toc200929956)

[3.4 Export Functionality 10](#_Toc200929957)

[3.5 Parent-Student Connection 10](#_Toc200929958)

[3.6 Fund Transfer System 11](#_Toc200929959)

[3.7 Withdrawal Tracking 11](#_Toc200929960)

[3.8 Emergency Fund Requests 12](#_Toc200929961)

[3.9 Spending Notifications 13](#_Toc200929962)

[3.10 Financial Goal Setting 13](#_Toc200929963)

[3.11 Bank Account Integration 14](#_Toc200929964)

[3.12 Semester-wise Expense Tracking 15](#_Toc200929965)

[4. External Interface Requirements 16](#_Toc200929966)

[4.1 User Interfaces 16](#_Toc200929967)

[4.2 Hardware Interfaces 17](#_Toc200929968)

[4.3 Software Interfaces 17](#_Toc200929969)

[4.4 Communications Interfaces 18](#_Toc200929970)

[5. Other Nonfunctional Requirements 18](#_Toc200929971)

[5.1 Performance Requirements 18](#_Toc200929972)

[5.2 Safety Requirements 18](#_Toc200929973)

[5.3 Security Requirements 18](#_Toc200929974)

[5.4 Software Quality Attributes 19](#_Toc200929975)

[6. Other Requirements 19](#_Toc200929976)

[Appendix A: Glossary 19](#_Toc200929977)

[Appendix B: Issues List 19](#_Toc200929978)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |

# 1. Introduction

### 1.1 Purpose

This Software Requirements Specification (SRS) document describes the requirements for the Minimalist Finance Tracker for College Students. This application is designed to provide a straightforward, user-friendly solution to help students manually log expenses, set budgets, and monitor their spending habits without unnecessary features. The document outlines both functional and non-functional requirements, user interfaces, and system constraints for the application.

### 1.2 Document Conventions

The following conventions are used in this document:

* Requirements are organized by functional area
* Priority levels are indicated as High, Medium, or Low
* TBD (To Be Determined) indicates information that is not yet available
* REQ-X-Y format is used for requirement identification where X is the feature number and Y is the requirement number within that feature

### 1.3 Intended Audience and Reading Suggestions

This document is intended for:

* Development Team: To understand what needs to be built
* Project Managers: To plan project milestones and allocate resources
* Quality Assurance Team: To develop test cases and validation criteria
* Stakeholders: To confirm that requirements meet their needs

For developers, sections 3 and 4 will be most relevant. Project managers should focus on sections 1, 2, and 6. Quality assurance personnel should pay particular attention to sections 3, 4, and 5.

### 1.4 Project Scope

The Minimalist Finance Tracker for College Students aims to address the specific financial management needs of college students by:

* Providing an intuitive platform for recording daily expenses with minimal effort
* Enabling effective budget creation and monitoring across various spending categories
* Offering simple visual representations of spending patterns and financial status
* Supporting financial relationships between students and parents/guardians
* Facilitating emergency fund requests in a structured, accountable manner
* Allowing optional bank account integration for automated transaction tracking
* Encouraging responsible financial habits through goal-setting and notifications
* Functioning offline for core features to ensure accessibility regardless of internet connectivity
* Generating exportable reports for financial planning and analysis
* Tracking expenses on a semester basis for academic period analysis

### 1.5 References

* IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications
* Comprehensive Proposal: Minimalist Finance Tracker for College Students
* [Include any relevant organizational policies, standards, or guidelines]

# 2. Overall Description

### 2.1 Product Perspective

The Minimalist Finance Tracker is a standalone application that can optionally integrate with banking services for transaction verification. It provides a comprehensive solution for students to manage their finances throughout their academic journey while maintaining simplicity and ease of use. The application consists of both a mobile app for students and parents/guardians, and a web portal for more detailed analysis.

### 2.2 Product Features

The major features of the Minimalist Finance Tracker include:

1. Manual Expense Entry
2. Budget Management
3. Visual Reports & Analytics
4. Export Functionality
5. Parent-Student Connection
6. Fund Transfer System
7. Withdrawal Tracking
8. Emergency Fund Requests
9. Spending Notifications
10. Financial Goal Setting
11. Bank Account Integration
12. Semester-wise Expense Tracking

### 2.3 User Classes and Characteristics

1. **Student User**
   * Primary user of the application
   * Records expenses and manages personal budget
   * Requests funds from parents when needed
   * Sets financial goals
   * Generates and reviews financial reports
   * Varying levels of financial knowledge
   * High frequency of use (daily/weekly)
   * Requires intuitive interface for quick expense logging
2. **Parent/Guardian User**
   * Secondary user with connected account
   * Transfers funds to student accounts
   * Reviews and approves/denies emergency fund requests
   * Optionally views student spending reports (if shared by student)
   * Monitors overall financial support provided
   * Moderate frequency of use (weekly/monthly)
   * Requires clear visibility into student spending (as permitted)
   * Various levels of technological proficiency
3. **System Administrators**
   * Manage user accounts and system configuration
   * Low frequency of use (as needed)
   * Require administrative interface and tools

### 2.4 Operating Environment

The Minimalist Finance Tracker will operate in the following environment:

* **Mobile Application:**
  + iOS (Version 14.0 or higher)
  + Android (Version 9.0 or higher)
* **Web Application:**
  + Modern web browsers (Chrome, Firefox, Safari, Edge - latest two major versions)
  + Responsive design for desktop and tablet access
* **Backend Services:**
  + Cloud-based infrastructure
  + Database management system for data storage
  + API integration with banking services (optional)

### 2.5 Design and Implementation Constraints

The following constraints will affect the design and implementation of the Minimalist Finance Tracker:

* Data privacy regulations and compliance with educational record laws
* Secure storage of financial information
* Cross-platform compatibility requirements
* Accessibility standards compliance
* Offline functionality for core features when network connectivity is unavailable
* Bandwidth and data usage considerations for mobile users
* Minimalist design philosophy (avoiding feature bloat)

### 2.6 User Documentation

The following user documentation will be provided with the application:

* In-app tutorials and onboarding guides
* Online user manual with searchable content
* FAQ section and troubleshooting guide
* Video tutorials for key features
* Context-sensitive help within the application

### 2.7 Assumptions and Dependencies

**Assumptions:**

* Students have access to smartphones or computers with internet connectivity
* Users are willing to manually input expense data that cannot be automatically captured
* Parents/guardians consent to being connected to student accounts
* Students prefer straightforward financial tracking over complex automated systems

**Dependencies:**

* Banking API availability for transaction verification (if implemented)
* Cloud service provider reliability for data storage
* Mobile platform guidelines and restrictions
* Third-party authentication services

# 3. Functional Requirements

### 3.1 Manual Expense Entry

**3.1.1 Description and Priority**

The ability for students to manually record expenses is a core functionality of the application. This feature has **High** priority as it forms the foundation for all other features.

**3.1.2 Stimulus/Response Sequences**

* **Stimulus:** Student wants to record a new expense
* **Response:** System displays quick-add expense entry form
* **Stimulus:** Student enters expense details and category
* **Response:** System saves the expense and updates relevant statistics
* **Stimulus:** Student wants to edit or delete an existing expense
* **Response:** System allows modification or removal of the selected expense

**3.1.3 Functional Requirements**

**REQ-1-1:** The system shall provide a quick-add expense option with the following information:

* Amount spent
* Date and time
* Category (from predefined list or custom)
* Notes field for expense context (optional)
* Payment method

**REQ-1-2:** The system shall provide predefined expense categories including but not limited to:

* Food
* Books and supplies
* Entertainment
* Transportation
* Housing
* Utilities
* Clothing
* Health
* Education
* Miscellaneous

**REQ-1-3:** The system shall allow students to create custom expense categories.

**REQ-1-4:** The system shall allow editing or deletion of any expense entry.

**REQ-1-5:** The system shall provide a quick-entry feature for recurring expenses.

### 3.2 Budget Management

**3.2.1 Description and Priority**

Budget management tools help students plan and monitor their spending across categories. This feature has **High** priority.

**3.2.2 Stimulus/Response Sequences**

* **Stimulus:** Student creates or modifies a budget
* **Response:** System stores budget settings and begins tracking against them
* **Stimulus:** Student spends money within a budget category
* **Response:** System updates visual progress indicators for that category

**3.2.3 Functional Requirements**

**REQ-2-1:** The system shall allow category-based budget allocation.

**REQ-2-2:** The system shall support the following budget periods:

* Monthly budget
* Weekly budget
* Semester-based budget

**REQ-2-3:** The system shall provide visual progress indicators for each budget category.

**REQ-2-4:** The system shall alert users when they approach or exceed budget limits.

**REQ-2-5:** The system shall allow budget adjustments during an active period.

**REQ-2-6:** The system shall provide historical budget performance data.

### 3.3 Visual Reports & Analytics

**3.3.1 Description and Priority**

Visual representations of spending data help students understand their financial patterns. This feature has **High** priority.

**3.3.2 Stimulus/Response Sequences**

* **Stimulus:** Student requests to view financial reports
* **Response:** System displays visual representations of financial data
* **Stimulus:** Student applies filters or changes report parameters
* **Response:** System updates visualizations based on selected parameters

**3.3.3 Functional Requirements**

**REQ-3-1:** The system shall provide charts for budget vs. actual comparison.

**REQ-3-2:** The system shall provide a calendar view of daily/weekly expenses.

**REQ-3-3:** The system shall generate expense heat maps by category and time period.

**REQ-3-4:** The system shall allow customization of the time period for expense visualization:

* Daily view
* Weekly view
* Monthly view
* Semester view
* Custom date range

**REQ-3-5:** The system shall display category-wise breakdowns of expenses.

**REQ-3-6:** The system shall provide trend analysis comparing current spending to previous periods.

### 3.4 Export Functionality

**3.4.1 Description and Priority**

The ability to export financial data for external analysis is important for comprehensive financial management. This feature has **Medium** priority.

**3.4.2 Stimulus/Response Sequences**

* **Stimulus:** Student requests to export financial data
* **Response:** System generates and provides the requested export format
* **Stimulus:** Student requests to share a report
* **Response:** System facilitates sharing via selected method

**3.4.3 Functional Requirements**

**REQ-4-1:** The system shall support CSV export for spreadsheet analysis.

**REQ-4-2:** The system shall generate PDF reports with customizable sections.

**REQ-4-3:** The system shall provide data backup and restore options.

**REQ-4-4:** The system shall allow report sharing via email or messaging apps.

**REQ-4-5:** The system shall maintain export history for quick access to previously exported reports.

### 3.5 Parent-Student Connection

**3.5.1 Description and Priority**

The connection between student and parent/guardian accounts facilitates financial support. This feature has **High** priority.

**3.5.2 Stimulus/Response Sequences**

* **Stimulus:** Student initiates parent/guardian connection
* **Response:** System sends connection request to specified parent/guardian
* **Stimulus:** Parent/guardian accepts connection request
* **Response:** System establishes secure connection between accounts

**3.5.3 Functional Requirements**

**REQ-5-1:** The system shall provide a secure account linking process.

**REQ-5-2:** The system shall implement privacy controls for student data, allowing students to determine what information is shared with connected parents/guardians.

**REQ-5-3:** The system shall include a communication thread for financial discussions between connected accounts.

**REQ-5-4:** The system shall maintain transaction history between connected accounts.

**REQ-5-5:** The system shall support multiple parent/guardian connections for a single student account.

**REQ-5-6:** The system shall allow either party to terminate the connection with appropriate notifications.

### 3.6 Fund Transfer System

**3.6.1 Description and Priority**

The ability to transfer funds between parent and student accounts is a key functionality. This feature has **High** priority.

**3.6.2 Stimulus/Response Sequences**

* **Stimulus:** Parent initiates fund transfer to student
* **Response:** System processes transfer and updates both accounts
* **Stimulus:** Parent schedules recurring transfer
* **Response:** System sets up automated transfers based on specified schedule

**3.6.3 Functional Requirements**

**REQ-6-1:** The system shall enable direct in-app transfers between parent and student accounts.

**REQ-6-2:** The system shall support scheduled recurring transfers.

**REQ-6-3:** The system shall provide transfer confirmation and receipts.

**REQ-6-4:** The system shall maintain transfer history and reporting.

**REQ-6-5:** The system shall support optional transfer conditions or requirements.

**REQ-6-6:** The system shall notify both parties when transfers are initiated and completed.

### 3.7 Withdrawal Tracking

**3.7.1 Description and Priority**

Tracking cash withdrawals helps maintain accurate financial records. This feature has **Medium** priority.

**3.7.2 Stimulus/Response Sequences**

* **Stimulus:** Student records a cash withdrawal
* **Response:** System logs withdrawal and updates balance
* **Stimulus:** Student views withdrawal history
* **Response:** System displays list of withdrawals with details

**3.7.3 Functional Requirements**

**REQ-7-1:** The system shall provide cash withdrawal logging functionality.

**REQ-7-2:** The system shall include ATM location mapping for withdrawals.

**REQ-7-3:** The system shall support withdrawal categorization.

**REQ-7-4:** The system shall implement withdrawal limits and alerts.

**REQ-7-5:** The system shall provide balance impact visualization for withdrawals.

### 3.8 Emergency Fund Requests

**3.8.1 Description and Priority**

The ability for students to request emergency funds from parents/guardians provides financial safety. This feature has **High** priority.

**3.8.2 Stimulus/Response Sequences**

* **Stimulus:** Student initiates emergency fund request
* **Response:** System displays fund request form
* **Stimulus:** Student submits request with details
* **Response:** System notifies parent/guardian of request
* **Stimulus:** Parent/guardian approves or denies request
* **Response:** System notifies student of decision and updates transaction history if approved

**3.8.3 Functional Requirements**

**REQ-8-1:** The system shall provide a structured request form with amount and reason fields.

**REQ-8-2:** The system shall limit emergency fund requests to two per month.

**REQ-8-3:** The system shall maintain a historical request archive.

**REQ-8-4:** The system shall notify parents/guardians of fund requests via app notification and email.

**REQ-8-5:** The system shall allow parents/guardians to approve, partially approve, or deny requests.

**REQ-8-6:** The system shall update transaction history and balances when fund requests are approved.

### 3.9 Spending Notifications

**3.9.1 Description and Priority**

Notifications about spending patterns help students stay aware of their financial status. This feature has **Medium** priority.

**3.9.2 Stimulus/Response Sequences**

* **Stimulus:** Student approaches budget threshold
* **Response:** System sends notification warning of approaching limit
* **Stimulus:** Student makes unusually large purchase
* **Response:** System requests confirmation and categorization

**3.9.3 Functional Requirements**

**REQ-9-1:** The system shall provide budget threshold alerts at 50%, 75%, 90%, and 100% of allocated budget.

**REQ-9-2:** The system shall implement large purchase confirmations.

**REQ-9-3:** The system shall detect and alert about unusual spending pattern.

**REQ-9-4:** The system shall provide low balance warnings.

**REQ-9-5:** The system shall allow customization of notification preferences.

### 3.10 Financial Goal Setting

**3.10.1 Description and Priority**

Goal setting functionality encourages positive financial habits. This feature has **Medium** priority.

**3.10.2 Stimulus/Response Sequences**

* **Stimulus:** Student creates a financial goal
* **Response:** System calculates required saving rate and displays goal information
* **Stimulus:** Student contributes toward goal
* **Response:** System updates goal progress
* **Stimulus:** Student achieves goal
* **Response:** System congratulates student and records achievement

**3.10.3 Functional Requirements**

**REQ-10-1:** The system shall enable goal creation with target amounts and deadlines.

**REQ-10-2:** The system shall track progress with visual indicators.

**REQ-10-3:** The system shall provide recommended saving amounts based on goal and timeframe.

**REQ-10-4:** The system shall include goal achievement celebrations.

**REQ-10-5:** The system shall maintain goal history and success rate tracking.

**REQ-10-6:** The system shall allow multiple concurrent goals.

### 3.11 Bank Account Integration

**3.11.1 Description and Priority**

Optional bank integration provides convenience for transaction tracking. This feature has **Low** priority as it is supplemental to the core manual tracking functionality.

**3.11.2 Stimulus/Response Sequences**

* **Stimulus:** Student opts to connect bank account
* **Response:** System securely facilitates bank account connection
* **Stimulus:** New transaction occurs in connected bank account
* **Response:** System imports and categorizes transaction

**3.11.3 Functional Requirements**

**REQ-11-1:** The system shall provide secure bank connection using industry-standard APIs.

**REQ-11-2:** The system shall support automatic transaction import and categorization.

**REQ-11-3:** The system shall enable manual override for automatically assigned categories.

**REQ-11-4:** The system shall include transaction filtering and search capabilities.

**REQ-11-5:** The system shall support multi-account connections and reconciliation.

**REQ-11-6:** The system shall maintain bank connection privacy and security.

### 3.12 Semester-wise Expense Tracking

**3.12.1 Description and Priority**

Tracking expenses on a semester basis helps students understand their spending patterns across academic periods. This feature has **Medium** priority.

**3.12.2 Stimulus/Response Sequences**

* **Stimulus:** Student selects semester view of expenses
* **Response:** System displays semester spending overview
* **Stimulus:** Student requests detailed breakdown of semester expenses
* **Response:** System shows category-wise spending for selected semester

**3.12.3 Functional Requirements**

**REQ-12-1:** The system shall allow students to define semester periods with start and end dates.

**REQ-12-2:** The system shall associate expenses with specific semesters based on the date of the expense.

**REQ-12-3:** The system shall provide semester-specific expense summaries showing:

* Total semester expenses
* Category-wise breakdown within the semester
* Comparison with previous semesters

**REQ-12-4:** The system shall allow students to set semester-specific budgets.

**REQ-12-5:** The system shall track semester-specific expenses separately from overall expenses.

**REQ-12-6:** The system shall provide visualizations comparing expenses across different semesters.

**REQ-12-7:** The system shall retain historical semester data for the duration of the student's academic program.

**REQ-12-8:** The system shall automatically detect and highlight spending pattern changes between semesters, showing percentage increases or decreases in each category.

**REQ-12-9:** The system shall enable students to add semester-specific tags to expenses to track unique costs associated with particular courses or activities (e.g., "Chemistry Lab," "Field Trip").

**REQ-12-10:** The system shall generate semester spending reports that can be shared with parents/guardians, with customizable privacy settings.

**REQ-12-11:** The system shall provide a semester expense forecast feature that predicts likely expenses for the upcoming semester based on historical data from previous semesters.

**REQ-12-12:** The system shall allow the archiving of past semester data while maintaining it for reference and analysis purposes.

**REQ-12-13:** The system shall provide a semester-to-semester migration tool that helps students adjust their budgets based on known changes in upcoming semester expenses (e.g., different housing costs, meal plans, or course materials).

**REQ-12-14:** The system shall generate end-of-semester financial summaries with actionable insights and recommendations for future semester budgeting.

**REQ-12-15:** The system shall support expense categorization by academic purpose, distinguishing between essential academic expenses (textbooks, lab fees) and discretionary academic expenses (optional materials, study aids).

**REQ-12-16:** The system shall provide a visual calendar view that shows spending patterns throughout the semester, highlighting peak spending periods such as beginning of semester, midterms, and finals.

**REQ-12-17:** The system shall implement a semester budget wizard that guides students through creating appropriate budgets based on their specific courses, housing situation, and campus meal plan for each new semester.

**REQ-12-18:** The system shall allow comparison of actual spending against the average spending patterns of anonymized peer groups in the same semester level, providing benchmarks for normal semester expenses.

**REQ-12-19:** The system shall track and display semester-specific financial aid, scholarships, grants, and loans alongside expenses to provide a complete semester financial picture.

**REQ-12-20:** The system shall generate a semester financial health score based on budgeting accuracy, spending patterns, and savings rate to help students gauge their financial management progress each semester.

# 4. External Interface Requirements

### 4.1 User Interfaces

The user interface for the Minimalist Finance Tracker shall be intuitive and minimalist in design, with the following characteristics:

**Student Mobile App:**

* Clean, minimalist design with emphasis on ease of expense entry
* Dashboard showing recent activities and current balance
* Quick-access buttons for common functions
* Visual budget indicators for each category
* Dark/light mode support
* Accessibility features compliant with WCAG 2.1 guidelines

**Parent/Guardian Mobile App:**

* Dashboard focused on student support overview
* Notification center for fund requests and alerts
* Simple fund transfer interface
* Communication thread with connected students

**Web Portal (Both users):**

* Responsive design for various screen sizes
* More detailed analytical tools than mobile version
* Export functionality for reports and data
* Account management features
* Help and support access

### 4.2 Hardware Interfaces

The application shall interface with the following hardware components:

* Camera for receipt scanning and document uploads (optional)
* Fingerprint scanner or facial recognition for biometric authentication (if available on device)
* GPS for ATM location mapping (optional)
* Notification systems of mobile devices

### 4.3 Software Interfaces

The application shall interface with the following software systems:

* Mobile operating systems (iOS, Android)
* Web browsers
* Backend database system for data storage
* Authentication services
* Banking APIs for transaction verification (optional)
* Cloud storage for backup data
* Push notification services
* Analytics engines for spending pattern analysis

### 4.4 Communications Interfaces

The application shall utilize the following communication protocols:

* HTTPS for all web communications
* OAuth 2.0 for authentication
* RESTful APIs for client-server communication
* WebSockets for real-time updates
* SMTP for email notifications
* SMS gateway for text notifications (optional)

# 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

* The application shall load the main dashboard within 3 seconds on devices with standard connectivity
* Expense entry shall be completed within 2 seconds of submission
* Visualization rendering shall occur within 5 seconds for up to 12 months of data
* The system shall support concurrent usage by at least 10,000 users
* Database queries shall return results within 3 seconds for standard operations
* The application shall function with limited or no connectivity for core expense tracking features
* The system shall maintain performance with up to 5 years of historical data per user

### 5.2 Safety Requirements

* The system shall validate all numerical inputs to prevent erroneous financial calculations
* The system shall confirm potentially destructive actions (like deletion of records) with the user
* The system shall maintain audit logs of all financial transactions
* The system shall implement timeout features to prevent unauthorized access to unattended devices

### 5.3 Security Requirements

* User authentication shall be required for all access to financial data
* All financial data shall be encrypted during storage and transmission
* Password recovery shall follow industry standard security practices
* The system shall implement session management with automatic timeout after 15 minutes of inactivity
* The system shall comply with relevant data protection regulations
* The system shall implement rate limiting to prevent brute force attacks
* Multi-factor authentication shall be available as an option for account security
* Bank connection credentials shall never be stored within the application

### 5.4 Software Quality Attributes

* **Reliability:** The system shall have 99.9% uptime excluding scheduled maintenance
* **Usability:** New users shall be able to record their first expense within 1 minute of app installation
* **Maintainability:** The system shall be designed with modular architecture to facilitate updates
* **Portability:** The application shall function consistently across supported platforms
* **Scalability:** The system shall accommodate a 200% increase in user base without performance degradation
* **Interoperability:** The system shall support standard data formats for import/export functionality
* **Localization:** The interface shall support multiple languages and currency formats
* **Simplicity:** The interface shall adhere to minimalist design principles while maintaining functionality

# 6. Other Requirements

* **Data Retention:** The system shall retain user financial data for a minimum of 5 years
* **Regulatory Compliance:** The application shall comply with financial data protection regulations
* **Internationalization:** The system shall support multiple currencies and date formats
* **Accessibility:** The application shall conform to WCAG 2.1 AA standards
* **Offline Functionality:** Core features shall function without internet connectivity

## Appendix A: Glossary

* **Category:** A classification for expenses (e.g., Food, Books)
* **Emergency Fund Request:** A formal request from student to parent/guardian for additional funds
* **Semester:** An academic period defined by start and end dates
* **Transaction:** Any movement of funds into or out of a student's account
* **Visualization:** Graphical representation of financial data
* **Goal:** A financial target with specified amount and deadline
* **Budget Period:** Timeframe for budget allocation (monthly, weekly, semester)

## Appendix B: Issues List

| **ID** | **Description** | **Status** | **Priority** |
| --- | --- | --- | --- |
| 1 | Definition of privacy levels for expense sharing with parents | Open | High |
| 2 | Banking API integration specifications | TBD | Medium |
| 3 | Offline functionality requirements | Open | Low |