Project Plan

Financial Budget App

Industry Partner	
Primary Instructor	Laily Ajellu
Team Member	Trang Nguyen
Team Member	Adam Simcoe
Team Member	Nhan Tran
Team Member	Nhu Ly
Team Member	Christian Do

Document Revision History

Revision #	Date
1.0	October 3rd, 2024
2.0	February 2nd, 2025

Table of Contents

1. Executive Summary

Objective	The primary objective of the BUDGET BOSS project is to develop a user-friendly budget-tracking application that empowers individuals to effectively manage their finances. The application will utilize mobile platforms to provide users with seamless access to their financial information anytime and anywhere, enhancing convenience and usability. By incorporating features such as income and expense tracking, automated calculations for milestones, visual graphs for data clarity, and personalized budgeting recommendations, BUDGET BOSS seeks to enhance financial habits among its users.
Corporate Goals Addressed	1. Customer Satisfaction 2. Customer Retention 3. Strong Market Positioning 4. Social Responsibility 5. Operational Efficiency
Planned Start Date	September 18, 2024
Planned End Date	April 10, 2025

2. Project Approvers, Reviews and Distribution List

Project Role	Name	E-mail	Date
Project Approver	Laily Ajellu	Laily.Ajellu@georgebrown.ca	Feb 2, 2025
Project Reviewer	Laily Ajellu	Laily.Ajellu@georgebrown.ca	Feb 2, 2025
Project Manager	Trang Nguyen	Trang.Nguyen3@georgebrown.ca	Feb 2, 2025
Requirement Engineer	Nhu Ly	HuynhYenNhu.Ly@georgebrown.ca	Feb 2, 2025
Development Team	Nhan Tran	Nhan.Tran@georgebrown.ca	Feb 2, 2025
Marketing Team	Adam Simcoe	Adam.Simcoe@georgebrown.ca	Feb 2, 2025
Financial Team	Christian Do	Christian.Do@georgebrown.ca	Feb 2, 2025

3. Scope

In Scope	Out of Scope
Dashboard for tracking income and expenses.	Tax filing assistance or other financial advisory functionalities.
Budget Creation and Management Tools	International currency conversion or exchange rate features.
Notification system for reminders about user-set saving goals.	Integration with other investment platforms or potential stock trading functionalities.
Data visualization such as graphs and charts to present expense/saving patterns	Integration with end-users' banking account.
User profile system to allow users to register and save their information accordingly.	Corporate financial management features.
Data Security	API integration for potential Ontario tax service integration.

4. Deliverables

Deliverable	Description
Income & Expense Overview	Central dashboard summarizing total income, expenses, and balance with categorized breakdowns (e.g., food, bills, entertainment).
Date-Filtered Financial Insights	Filters to view income and expense data by daily, weekly, or monthly ranges for detailed analysis of financial patterns.
Real-Time Updates	Newly added transactions instantly appear on the dashboard for accurate tracking.
Budget Setup	Allows users to create and assign budgets to specific categories such as groceries, rent, and leisure.
Budget Adjustment Tools	Functionality to modify budgets in real time, adapting to evolving financial needs.
Overspending Alerts	Notifications are sent when expenses approach or exceed defined budget limits.
Configurable Reminders	Users can set up and customize notifications for saving goals and budget reviews.
Goal Progress Tracking	Automated updates on savings milestones to keep users on track.
Financial Report Visualization	Line and semi-circular pie charts that display income vs. expenses over time for trend evaluation.
Expense Breakdown Charts	Pie charts to display the distribution of expenses by category.

Savings Progress Visualization	Progress bars tracking how close users are to meeting their savings goals.
Custom Category Creation	Users can create their own categories for expenses, income, or budgets.
Secure User Authentication	Registration and login system using secure authentication methods (e.g. JWT).
User Preference Management	Features to manage preferences, including notification settings.
Data Encryption	Implement AES encryption for secure storage and transmission of sensitive financial information.
Strong Password Hashing	Use hashing algorithms to securely store passwords and prevent unauthorized access (e.g. bcrypt or werkzeug).
Regular Security Audits	Conduct periodic reviews of security practices to identify and mitigate vulnerabilities, protecting against data breaches.

5. Assumptions

This project makes the following assumptions:

1. **End-User Profile**
   End-users will be individuals managing personal finances, not businesses or corporations. The app will focus on features designed for personal finances, without the need for business-specific functionalities. The user interface and experience will be simplified to cater to individual users, ensuring easy navigation.
2. **Device Accessibility**
   End-users will have access to smartphones or tablets with internet connectivity. Mobile devices with internet are widely accessible, and their availability is critical for users to leverage the app's functionalities effectively.
3. **User Engagement**
   End-users will regularly interact with the app to track their finances and set savings goals. Consistent user engagement is vital for achieving financial goals and tracking progress, aligning with the app's primary purpose.
4. **Technology Availability**
   The necessary technology (hardware and software) will be accessible and compatible with project requirements, ensuring Budget Boss runs smoothly on supported devices with internet access and sufficient hardware resources. All required software components will be kept up to date to ensure optimal performance, security, and seamless integration.
5. **Alignment of Deliverables**
   The identified project deliverables align with the project's objectives. This assumption guarantees that all efforts are directed toward meeting the app's intended goals and addressing user needs effectively.
6. **Stakeholder Consensus**
   All stakeholders have provided their recommendations and requirements, with no significant changes expected during the project. This assumption ensures stability in the project scope, avoiding disruptions caused by shifting expectations.
7. **Feasibility of Project Schedule**
   The project schedule will be achievable within the allocated timeframes. This assumption ensures that tasks can be completed on time without overburdening the team, leading to a balanced workflow.

8. **Accurate Task Dependencies**
   Task dependencies will be accurately identified. Proper sequencing of tasks is critical to avoid delays, ensuring an efficient and smooth project workflow.
9. **Sufficient Resources**
   The project team will have sufficient time and resources to complete all project deliverables. This assumption is vital to ensure that the team can meet objectives and deliver high-quality results without resource constraints.
10. **Quality Assurance**
    The project deliverables will meet or exceed quality standards. By adhering to established quality standards, the app will be reliable, user-friendly, and secure, meeting stakeholder and user expectations.
11. **Timely Feedback**
    Stakeholders will provide timely feedback throughout the project. This assumption is crucial to identify and address potential issues early, keeping the project on track and aligned with stakeholder expectations.

6. Dependencies

The following are the internal and external dependencies that will have to be acknowledged and addressed

- **External Financial APIs:** Reliance on third-party APIs, such as YNAB API, for real-time data. Any downtime or changes could impact the functionality of the program.
- **Data Security Compliance**
  The app must comply with data privacy laws, such as PIPEDA or GDPR, to mitigate legal risks and ensure robust protection of user data. Technical dependencies include implementing encryption algorithms and secure storage solutions.
- **Development Team Capacity**
  The project timeline depends on the team's ability to meet deadlines and address technical challenges like debugging, feature integration, and API connections.
- **Marketing and User Onboarding**
  The app's success relies on timely promotion and effective onboarding strategies to engage users. Dependencies include integrating analytics tools to track user behavior and optimizing workflows for a smooth onboarding process.
- **User Feedback Loop**
  Continuous iteration is dependent on the collection and processing of user feedback post-launch. Dependencies include creating feedback forms, using analytics platforms, and implementing agile development cycles to make timely updates.
- **Operating System and Device Compatibility**
  The app's performance hinges on its ability to run efficiently on various devices and OS versions. Dependencies include thorough testing on emulators and physical devices using Expo's tools, as well as maintaining compatibility with the latest version of React Native, Expo SDKs, and OS updates.
- **App Store Approval**
  The app's release is subject to approval by app stores, and any issues in compliance with their guidelines can cause delays. Dependencies include adhering to store policies, implementing in-app purchase systems where applicable, and meeting security standards during submission.

- **Database Management**
  The app's core functionalities, like user profiles and financial records, depend on secure and efficient database operations. Dependencies include choosing a reliable database, implementing optimized queries, and ensuring secure connections.
- **Third-Party Libraries**
  The app's functionality may depend on external libraries for features like chart visualization, push notifications, and authentication. Dependencies include library updates and compatibility with project requirements.

7. Risk Management

Potential Risk	Severity (H/M/L)	Likelihood (H/M/L)	Management Strategy
Data Breach or Security Vulnerability	H	M	Regular security audits, data encryption, and compliance with security regulations.
Inadequate User Testing	H	L	Implement a comprehensive user testing plan, including beta testing and feedback collection.
Compliance with New Financial Regulations	H	L	Monitor regulatory changes and plan updates to ensure compliance.
User Privacy Concerns	H	M	Implement transparent data usage policies, provide clear privacy settings, and follow best practices.
Inadequate Documentation	L	L	Maintain detailed documentation throughout development to ensure smooth handover and easy maintenance.
Development Delays	M	L	Allocate buffer time in project timelines and regularly assess team capacity and progress.
Poor User Adoption	H	M	Strong marketing campaign and onboarding strategies, focus on user-friendly experience.
Budget Overrun	M	L	Implement strict budget monitoring and adjust scope or timeline as needed.
Market Competition	H	H	Conduct thorough market research to identify trends and differentiate the app effectively.
App Store Rejection	M	L	Ensure the app meets all platform guidelines before submission; maintain clear communication with stores.
Delayed Market Feedback	M	L	Collect feedback from users early and regularly and implement a continuous improvement process.
Outdated Devices Compatibility	L	L	Test the app on older devices to ensure compatibility and minimize issues for users with legacy hardware.

8. Communication

Reporting

The following reports will be produced.

Report	Audience	Frequency
Product backlog	Project manager, Requirement Engineer, Development team,	Bi - weekly, before each sprint and as needed
Sprint backlog	Project manager, Requirement Engineer, Development team,	Daily
Project Plan	Project Manager, Requirement Engineer, Development Team, Stakeholders	Beginning of the project
Project Status report	Project Manager, Requirement Engineer, Development Team, Stakeholders	Bi - weekly, before each sprint and as needed
Closure Report	Project Manager, Requirement Engineer, Development Team, Stakeholders	Ending of the project
Risk Management Report	Project Manager, Requirement Engineer, Development Team, Stakeholders	Weekly or as required
Change Request Report	Project Manager, Requirement Engineer, Development Team, Stakeholders	As required or when changes occur
Resource Allocation Report	Project Manager, Development Team, HR Team	Bi-weekly or as required

Meetings

The following meetings/communication will be established.

Meeting	Purpose	Attendees	Frequency
Daily Standups	Discuss progress, next steps, and roadblocks.	Project manager, Requirement Engineer, Development team,	Daily

Bi-Weekly Stakeholder Meetings	Update stakeholders on project status, gather feedback, and align on goals and expectations.	Project manager, Requirement Engineer, Development team, Marketing team, Financial team	Bi-weekly
Ad-hoc Meetings	Scheduled as needed to address urgent issues, ensure timely decision-making, and adapt to project changes.	Project manager, Requirement Engineer, Development team, Marketing team, Financial team	Ad-hoc

9. Task Listing (WBS- Work Breakdown Structure)

Reference	Tasks	Duration	Dependency
1	**Project Initiation**	**4 weeks**	**None**
1.1	Define project scope	2 weeks	None
1.2	Identify stakeholders	1 week	1.1
1.3	Develop project charter	1 week	1.1, 1.2
2	**Project Planning**	**5 weeks**	**1.3**
2.1	Create project plan	2 weeks	1.3
2.2	Risk management planning	1 week	2.1
2.3	Resource allocation	1 week	2.2
2.4	Define communication plan	1 week	2.3
3	**Design Phase**	**5 weeks**	**2.4**
3.1	UI/UX design	2 weeks	2.4
3.2	Prototype development	2 weeks	3.1
3.3	Design review and approval	1 week	3.2
4	**Development Phase**	**11 weeks**	**3.3**
4.1	Setup development environment	1 week	3.3

4.2	Database design	3 weeks	4.1
4.4	Implement core functionalities	5 weeks	4.3
4.5	Security implementation	2 weeks	4.4
5	**Testing phase**	**3 weeks**	**4.5**
5.1	Unit testing	1 week	4.5
5.2	Integration testing	1 week	5.1
5.3	User acceptance testing (UAT)	1 week	5.2
6	**Deployment Phase**	**5 weeks**	**5.3**
6.1	Finalize app for launch	1 weeks	5.3
6.2	App store submission	1 week	6.1
6.3	Marketing and promotion planning	3 weeks	6.2
7	**Post-Launch Activities**	**6 weeks**	**6.2, 6.3**
7.1	Gather user feedback	1 week	6.2, 6.3
7.2	Monitor app performance	2 weeks	6.2, 6.3
7.3	Plan for updates and feature enhancements	2 weeks	7.1,7.2
7.4	Conduct retrospective analysis	1 week	7.3

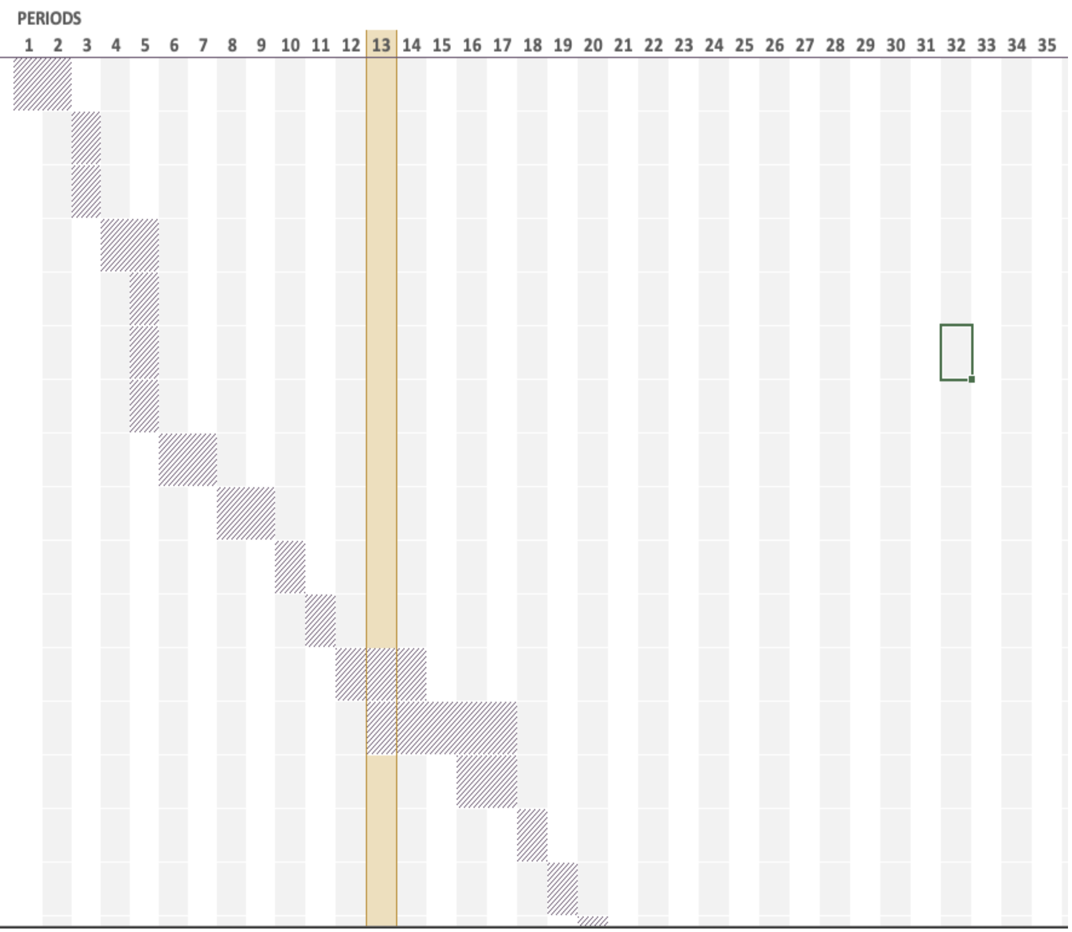
10. Gantt Chart

Budget Boss

Select a period to highlight at right. A legend describing the charting follows.

13 Plan Duration | Actual Start | % Complete | Actual (beyond plan) | % Complete (beyond plan)

ACTIVITY	PLAN START	PLAN DURATION
Define project scope	1	2
Identify stakeholders	3	1
Develop project charter	3	1
Create project plan	4	2
Risk management planning	5	1
Resource allocation	5	1
Define communication plan	5	1
UI/UX design	6	2
Prototype development	8	2
Design review and approval	10	1
Setup development environment	11	1
Database design	12	3
Implement core functionalities	13	5
Security implementation	16	2
Unit testing	18	1
Integration testing	19	1

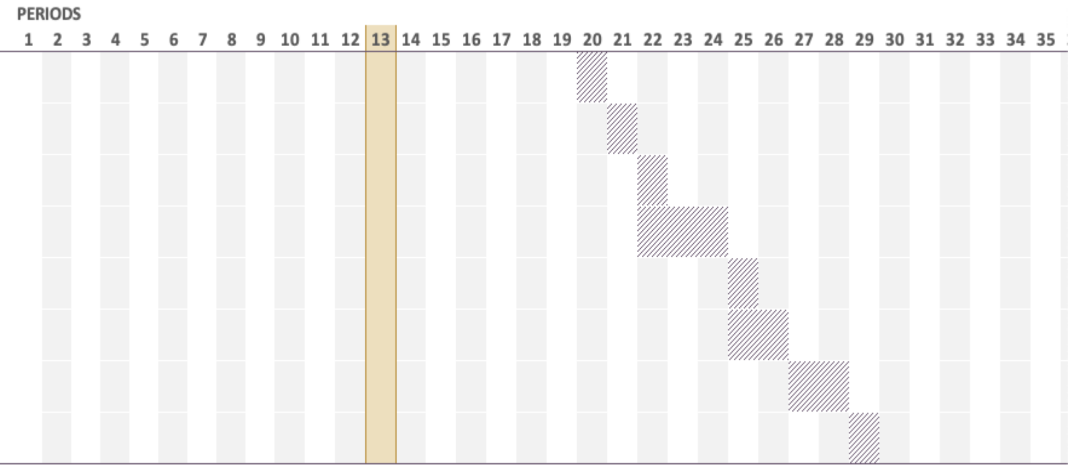


Budget Boss

Select a period to highlight at right. A legend describing the charting follows.

13 Plan Duration Actual Start % Complete Actual (beyond plan) % Complete (beyond plan)

ACTIVITY	PLAN START	PLAN DURATION
User acceptance testing (UAT)	20	1
Finalize app for launch	21	1
App store submission	22	1
Marketing and promotion planning	22	3
Gather user feedback	25	1
Monitor app performance	25	2
Plan for updates and feature enhancements	27	2
Conduct retrospective analysis	29	1



11. Milestones

Major Activity or Milestone	Estimated Milestone Target date	Owner/Reviewer Team Members
Project Initiation	Oct 3, 2024	Trang, Adam, Nhan, Nhu, Christian
Design Phase	Nov 7, 2024	Trang, Adam, Nhan, Nhu, Christian
Development Phase	Jan 22, 2025	Trang, Adam, Nhan, Nhu, Christian
Testing	Mar 14, 2025	Trang, Adam, Christian
Deployment	Mar 28, 2025	Nhan, Nhu, Trang
Post Launch Survey	Apr 5, 2025	Adam, Nhan, Christian

12. RAM – Responsibility Assignment Matrix

Task/Activity	Trang	Adam	Nhan	Nhu	Christian
Project Initiation	R	A	C	C	C
Define scope, identify stakeholders, and charter development	R	C	C	C	R
Project Planning	C	C	C	R	A
Create plan, risk management, and resource allocation	R	C	C	R	A
Design Phase	A	C	C	R	R
UI/UX design, prototype development, and review	R	C	C	A	C
Development Phase	R	C	A	C	C
Environment setup, database design, and core implementation	R	C	C	A	C
Testing Phase	C	R	A	C	R
Unit testing and UAT	R	A	C	C	C
Deployment Phase	A	C	C	R	C
Finalize app and marketing planning	R	A	C	C	C
Post Launch	C	C	A	R	C
Feedback gathering and monitoring	C	R	A	C	C

R = Responsible: The person who performs the task.
A = Accountable: The person who is ultimately answerable for the task.
C = Consulted: The person who provides input or advice.
I = Informed: The person who needs to be kept up-to-date on the progress of the task.

13. Approval

Project Role	Name	Signature	Date
Project Manager	Trang Nguyen	Trang Nguyen	Feb 2, 2025
Requirement Engineer	Nhu Ly	Nhu Ly	Feb 2, 2025
Development Team	Nhan Tran	Nhan Tran	Feb 2, 2025
Marketing Team	Adam Simcoe	Adam Simcoe	Feb 2, 2025
Financial Team	Christian Do	Christian Do	Feb 2, 2025