Introduction

Pocket Savr is a "Budget Planner with Savings Goals” app designed for individuals to track their expenses and take wiser decisions on their financial planning while getting access to all transaction records. A budget app can be a powerful tool for achieving financial stability, identifying areas for improvement, and reaching financial goals. It promotes discipline, accountability, and ultimately empowers users to take control of their financial lives with greater ease and confidence by optimum utilization of funds.

Objective

Building an app for budget planning that aims to help youth and working professionals to :

- Manage expenses better and take wiser decisions in life
- Track day to day transactions ,
- Educate them about saving wisely,
- Set and achieve strong financial goals

Why of the problem :

- Users spend their money ineffectively without having proper tracking on the categories and expenditures which don’t help them in proper savings disrupting their financial planning

  https://fortune.com/recommends/banking/57-percent-of-americans-cant-afford-a-1000-emergency-expense/

  According to this report if people try to track their expenses their emergency funds could be managed effectively

Key Functionalities

- Expenditure tracking monitors
- Categorization of various spending categories (Grocery / rentals / bills etc)
- Budget limit reminders
- Cash tracking transactions
- Upi / third party wallet transaction tracking via sms
- Monthly / daily / weekly graphs dashboard
- Backup and restoring data (extract to csv / google drive backup)
- Educating people (video courses)
- Split among friends/relatives

Functional specifications : These are feature set level specifications which have to be

- User downloads the app
- Receives a pre-sign ups screens (2-3 sliders with app benefits)
- User authentication , details required (mobile no. , name)
- Call to action button , Get started
- Verification of user (New registration / Existing user)
- Permission access (To scan SMS real time, access location, contacts, camera)
- Connect bank account
- Security layers (risk) (encryption , don't store data on own server / store on user's local device)
- Categorization of various spending categories
- Budget limit reminders
    - Set limit for each category
    - As user is spending in a particular category, user gets information about how well he is utilizing the set budget

Automated reminders when budget utilization

- Tracking Cash transactions
    - P1: In-app notifications to bring back users to update their cash expenses
    - P2: Integrate with Whatsapp for timely reminders
- Budget creation and management

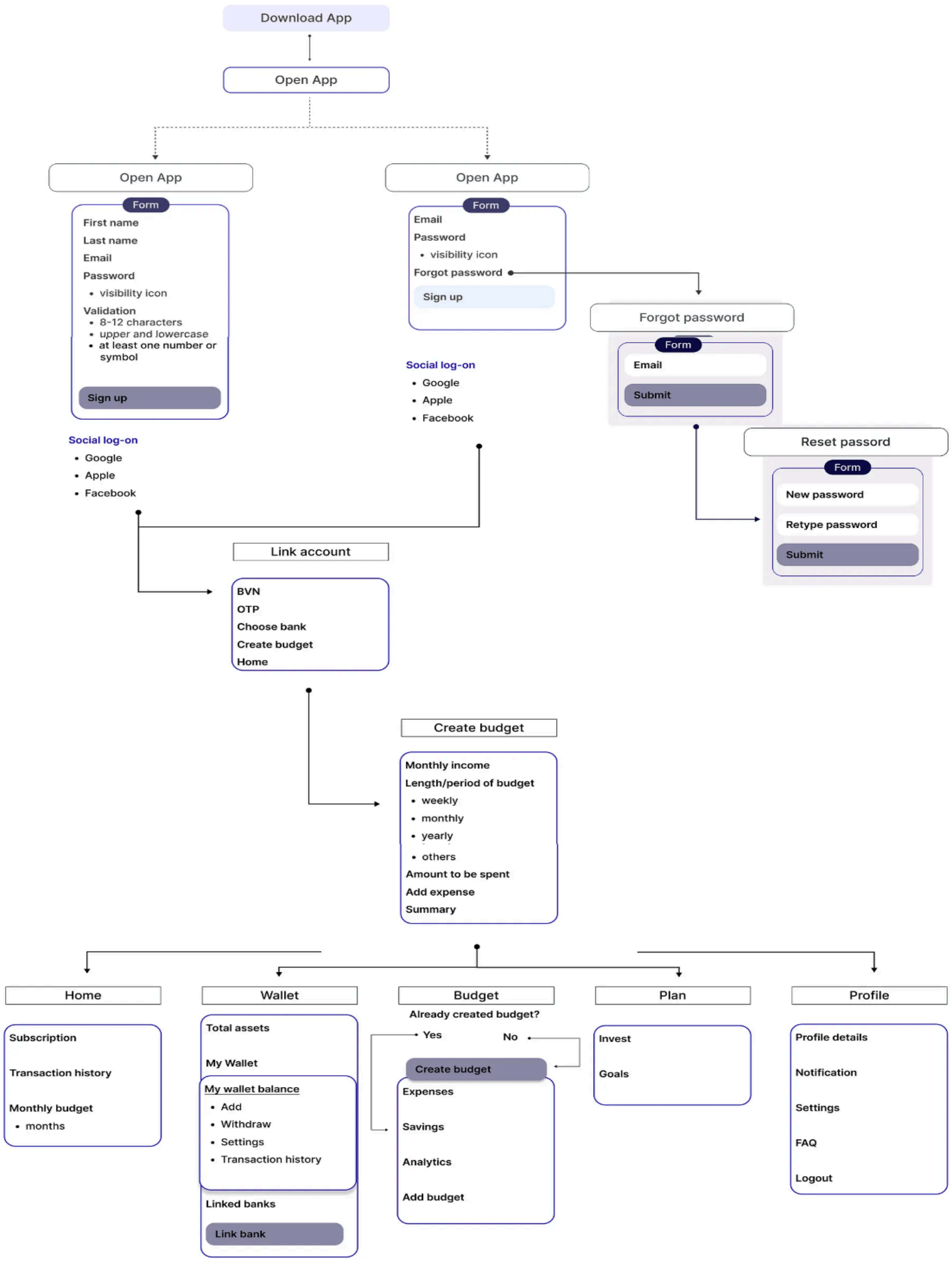
User Flows

Download app > Open app > Swipe sliders > Enter details (Mobile no. , Type your name) > Click Get started >

Verification of user > Click existing user / new user accordingly > Permission access > sms, contacts, camera > click allow > Ensure with security layers (Don't store data on user's server)

Budget creation

Click create budget > Monthly income > Amount to be spent > Select category > Add expense > budget summary


Download App
Open App
Open App
Form
First name
Last name
Email
Password
• visibility icon
Validation
• 8-12 characters
• upper and lowercase
• at least one number or symbol
Sign up
Social log-on
• Google
• Apple
• Facebook
Open App
Form
Email
Password
• visibility icon
Forgot password
Sign up
Social log-on
• Google
• Apple
• Facebook
Forgot password
Form
Email
Submit
Reset passord
Form
New password
Retype password
Submit
Link account
BVN
OTP
Choose bank
Create budget
Home
Create budget
Monthly income
Length/period of budget
• weekly
• monthly
• yearly
• others
Amount to be spent
Add expense
Summary
Home
Subscription
Transaction history
Monthly budget
• months
Wallet
Total assets
My Wallet
My wallet balance
• Add
• Withdraw
• Settings
• Transaction history
Linked banks
Link bank
Budget
Already created budget?
Yes
No
Create budget
Expenses
Savings
Analytics
Add budget
Plan
Invest
Goals
Profile
Profile details
Notification
Settings
FAQ
Logout

Target Audience

- Young Adults: users between 20-30 yrs of age, getting money from their parents for their everyday expenses or users doing part time jobs while studying.
- Working professionals: steady incomes looking to optimize their expenses, save for goals, and track expenses
- Parents/Families: household expenses for shared expenses, family expenses

Competitor Analysis

Competitors - (Axio , Mint, Ynab, Pocket Guard)

Overall weaknesses found from the market research of apps (Ref : Users comments and ratings on play store apps)

1) No proper UI UX
2) Battery draining issues
3) No categorisation of grocery
4) No splitwise feature
5) No sync / account login
6) No data backup
7) Not supporting multiple screens
8) No calendar jumping
9) No Adding family accounts
10) Daily reminder feature
11) No sms detection

Opportunities for Pocket Savr

1) Interactive UI /UX

2) Proper Categorization (Grocery , Rentals, Bills, etc)
3) No Battery issues
4) Account logins for individuals with educating personal finances according to individual user spends
5) Splitting feature for money splitting on travel / food among friends, family
6) Data backup feature (Extract using csv / backup to Google Drive)
7) Family members accounts can be also added in a single user
8) SMS detection with third party transactions / wallets
9) Cash transactions detection

Success Metrics & Measurements

1) NPS (Net promoter score) -

To calculate your Net Promoter Score, subtract the percentage of Detractors from the percentage of Promoters. NPS = % promoters - % detractors.

2) CSAT (Customer satisfaction score)

CSAT, or customer satisfaction score, is a commonly used metric that indicates how satisfied customers are with a company's products or services. It's measured through customer feedback and expressed as a percentage (100% would be fantastic – 0% would be terrible).

3) **User acquisition rate (new users signing up**) - CAC

(Total sales + Marketing expenses)/ (No. of new users) = CAC

4) New feature adoption rate

5) **Number of Daily / monthly active users -**
(Daily active users / monthly active users) * 100

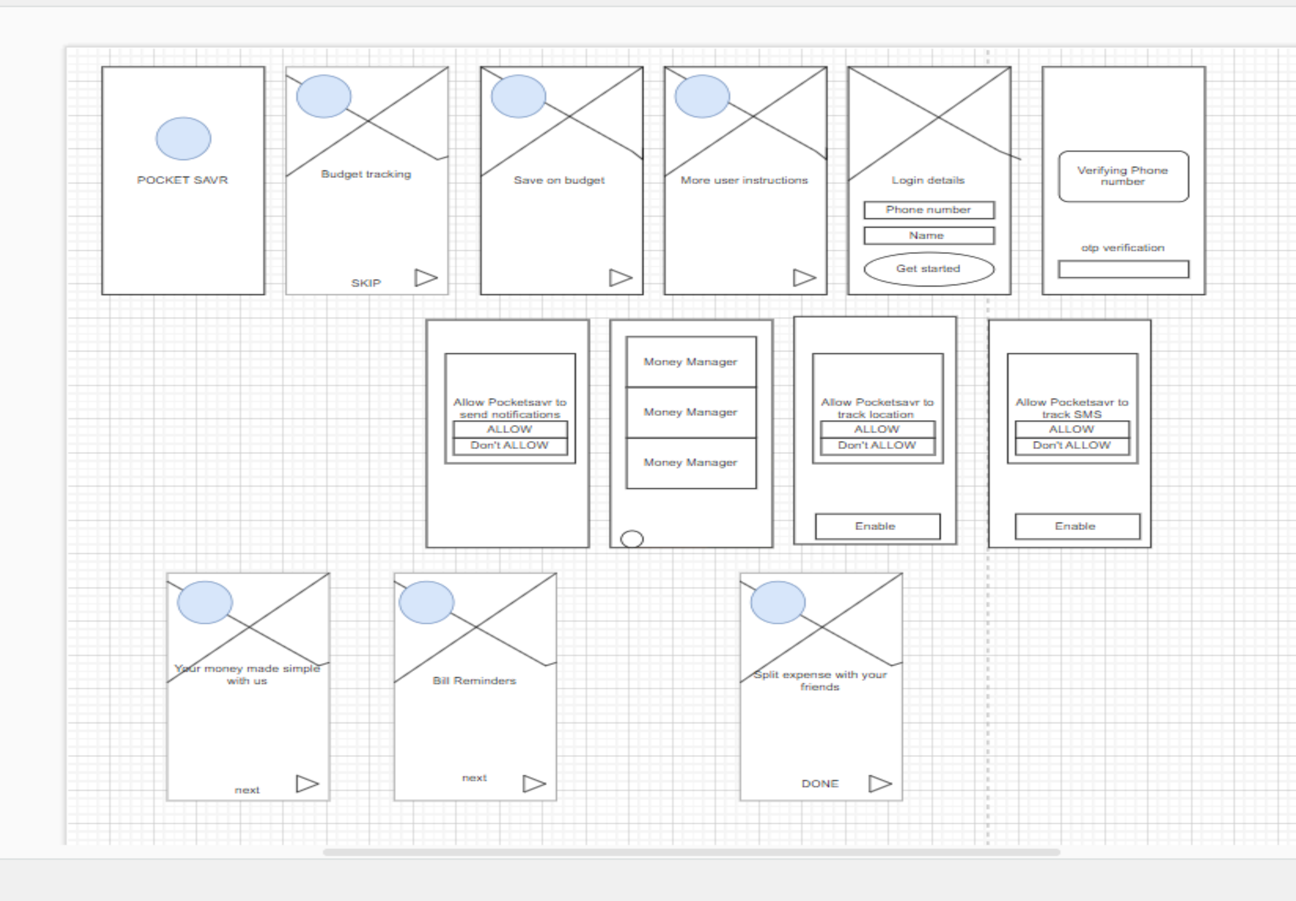
6) **Churn Rate (No. of people leaving the site)**
(Lost customers / Total number of customers in starting period)*100

7) **Customer lifetime value**

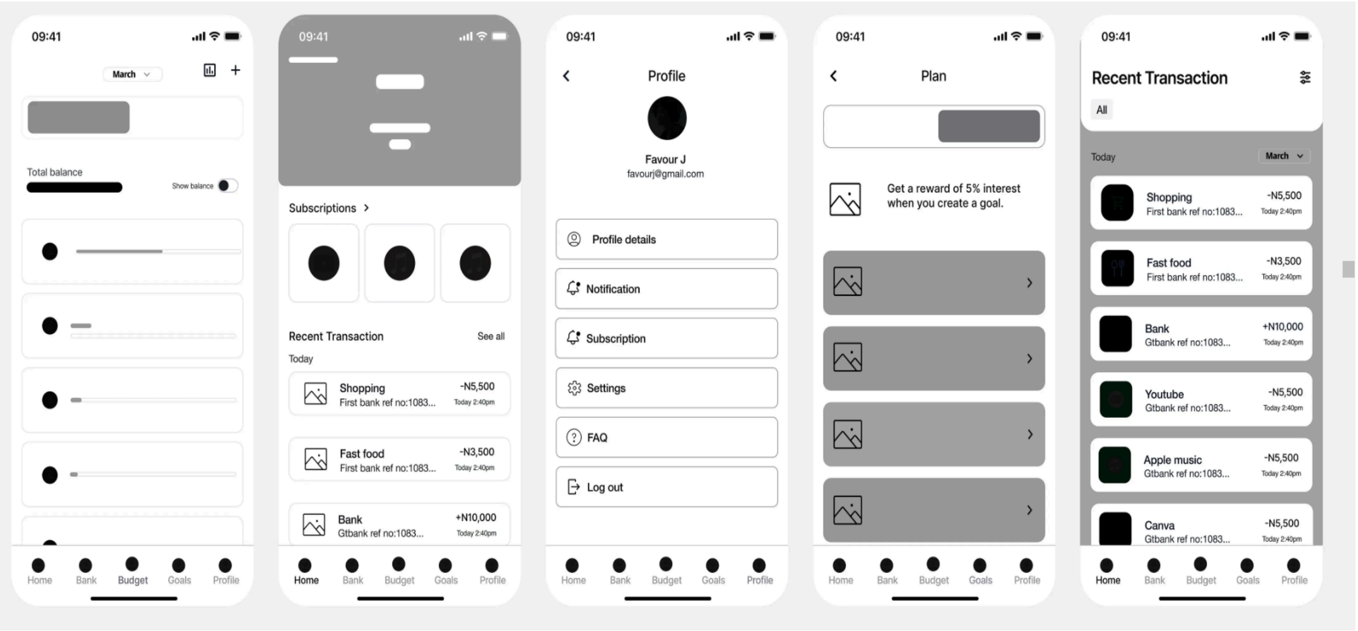
*Customer revenue per year * Duration of the relationship in years – Total costs of acquiring and serving the customer = CLV*

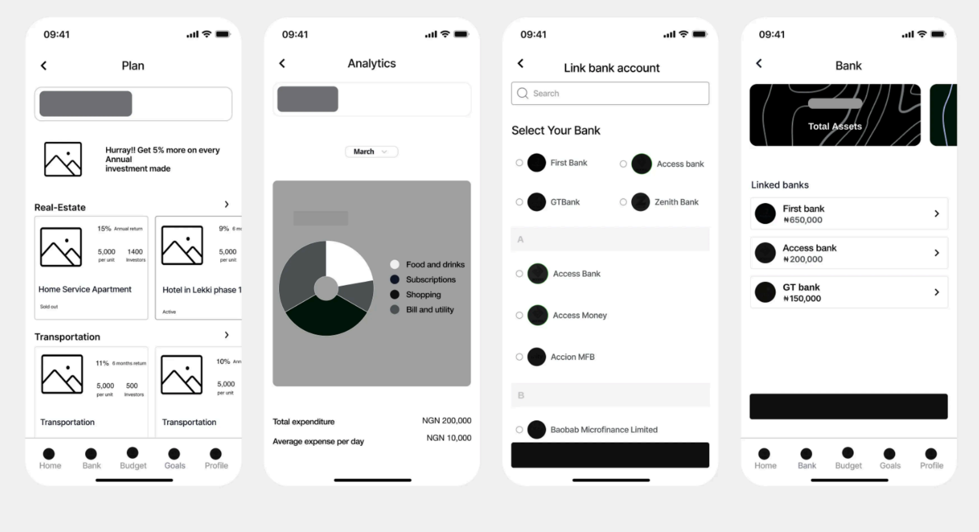
Wireframes & user mockups

https://drive.google.com/file/d/1zuxs3HaYOG06ai37hvPWL_rYhIVbUgN6/view?usp=sharing



Onboarding screen





High-Level Technical Specification Document for Budget Saving Planner App

---

Objective

To design a high-level technical specification for a budget saving planner app, detailing its key features, functionalities, system architecture, and considerations for scalability and security.

Key Features and Functionalities

1. User Authentication and Authorization
   - Sign-up/Login
   - Two-Factor Authentication (2FA)
   - OAuth 2.0 Integration
2. Bank Account Integration
   - Securely connect to user bank accounts
   - Retrieve transaction data
3. Expense Tracking and Categorization

    - Automatic categorization of expenses
    - Manual categorization option
4. Budget Planning and Goal Setting
    - Create and manage budgets
    - Set and track financial goals
5. Alerts and Notifications
    - SMS and push notifications for budget updates and alerts
6. Reports and Insights
    - Generate financial reports
    - Provide insights and recommendations
7. Security Features
    - Data encryption (AES-256, SSL/TLS)
    - Regular security audits
8. User Interface
    - Intuitive and simple design
    - User-friendly navigation

Wireframes/Mockups

1. Login Screen
2. Dashboard
3. Expense Tracking
4. Budget Planning
5. Notifications

Description of Key Components and Interactions

1. User Interface (UI)
    - Components: Login screen, dashboard, expense tracking, budget planning, notifications

- Interaction: Users interact with the UI to input data, view reports, and manage budgets. The UI communicates with the backend services through API calls.

2. Backend Services
   - Components: Authentication service, transaction processing service, budgeting service, notification service
   - Interaction: Backend services handle user authentication, process transactions, manage budget data, and send notifications.
3. Database
   - Components: User database, transaction database, budget database
   - Interaction: The database stores user information, transaction data, and budget details. It interacts with backend services to retrieve and store data.
4. Third-Party APIs
   - Plaid/Yodlee: For bank account integration and transaction retrieval
   - Twilio: For SMS notifications
   - Firebase Cloud Messaging (FCM): For push notifications
5. Security Components
   - Encryption: AES-256 for data at rest, SSL/TLS for data in transit
   - 2FA: Integrated into the authentication service
   - Regular Audits: Conducted to identify and mitigate vulnerabilities

---

Third-Party Services or APIs

1. Plaid/Yodlee: For secure bank account integration and transaction retrieval.
2. Twilio: For SMS notifications.

3. Firebase Cloud Messaging (FCM): For push notifications.
4. Auth0: For authentication and identity management.
5. Mixpanel/Google Analytics: For user behavior and app performance tracking.

Considerations for Scalability and Security

1. Scalability
   - Use microservices architecture to handle different functionalities independently.
   - Implement load balancers to distribute traffic efficiently.
   - Use cloud services (e.g., AWS, Azure) for flexible scaling.
2. Security
   - Implement end-to-end encryption for data security.
   - Regularly update and patch software to protect against vulnerabilities.
   - Conduct frequent security audits and penetration testing.

Data Capture and Tools

1. Data Capture
   - User information (name, email, phone number)
   - Transaction data (date, amount, category)
   - Budget and goal details
   - Notifications and alert preferences
2. Tools
   - Database: PostgreSQL or MongoDB for data storage
   - Analytics: Google Analytics, Mixpanel for tracking user behavior and app performance

    - Security: Snyk for vulnerability scanning, Cloudflare for DDoS protection

---

Conclusion

This high-level technical specification document outlines the essential features, system architecture, and considerations for developing a secure and scalable budget saving planner app. By leveraging robust security measures and integrating third-party APIs, the app aims to provide a seamless and secure user experience.

Go To Market strategy

Pricing models - Free & Subscription based

Subscription based with monthly / quarterly / yearly plans with extra added features like Giving Loans on interest lower than banks depending upon the saving score of an individual

Price Based competitor Analysis

AXIO	Mint	YNAB	Pocket Guard	Pocket Savr
Free/ $4.99 per month/ $49.99 per yr	Free	$14.99 Per month/ $99 per yr	Free $7.99/month/ $7.99 per yr	Free/ $4.60 per month/ $40 per yr

Promotional Plans

Advertising - online and offline

Offline channels - Boards, banners , seminars in offices and colleges targeting students and working professionals

Online Channels used - Social media platforms (Youtube , instagram , linkedin, Emails, Facebook)

Strategy - Spreading awareness about budgeting on social media through creating content

Partnerships and collaborations with banks and other finance apps for promotions

Conducting webinars and free communities on whatsapp and other social media channels to educate people about budgeting , saving and building their financial plans and setting a roadmaps for achieving it.

Generating revenue via displaying Ads on third party apps like finance apps , gaming, banking, e-commerce

Retargetting audience via Push notifications , SMS, Emails, Facebook ads

Rewards and Refferal Systems

Giving coupons on referring the app and rewarding on achieving several saving milestones.

Rewards such as discount on fuels, shopping , etc. which motivates the user to save more money and habit day to day monitoring of their expenses.

Timelines & Deadlines

Week 1-2: Define scope and requirements.

Week 3-5: Design phase.

Week 6: Planning and resource allocation.

Week 7-8: Setup and infrastructure.

Week 9-14: Front-end development.

Week 9-16: Back-end development.

Week 17-20: Integration and testing.

Week 21-23: Beta release and feedback.

Week 24-25: Final adjustments and polishing.

Week 26: Launch.

Week 27+: Post-launch monitoring and support.

Launching Phases

Phase 1 - MVP Launch & feedback generations

Phase 2 or 3 - Extra added features like

Splitwise system , Additions of Family members accounts, Providing Loans

Costing

The cost of building a personal finance app can vary widely depending on the features you want to include, the complexity of the app, and the experience of the development team you hire.

A basic personal finance app with basic features like budgeting, expense tracking, and bill pay can cost as moderate as $35,000 to $75000 to develop.

Market Size and Penetration:

- Total target market: 10 million potential users.
- Expected market penetration in the first year: 1% (100,000 users).

User Acquisition:

- Monthly active users (MAUs) growth rate: 10% compounded monthly.

Revenue Streams:

- Freemium model: Free users and premium users.
- Percentage of users who opt for premium: 5%.
- Monthly subscription fee for premium users: $4.60.
- Yearly subscription fee for premium users: $40.
- Percentage of premium users who opt for yearly plan: 50%.
- Ad revenue from free users: $0.50 per active user per month.

Parameter	Value
Total market size	10000000
Market penetration in first yr	1%
Initial Users	100000
Monthly Growth Rate	10%
Percentage of premium users	5%
Monthly subscription fee	$4.60

Yearly subscription fee	$40
percentage of yearly plan users	50%
Ad revenue per month	$0.50
Budget for creating the app	$40.00
Calculation	**Value**
Total MAUs (First yr)	2,138,428
Total Premium users	106,921
Revenue from monthy subscriptions	$2,951,031.16
Revenue from Yearly subscriptions	$2,138,428.38
Ad revenue from free users	$12,189,041.75
Total revenue	**$17,278,501.28**
Total Budget	**$40,000**
Net Profit/ Loss	**$17,238,501.28**

Problem Solving

Task 4 - Problem-Solving and Adaptability (30 minutes)
Issue - Third Party API has updated its terms of service, introducing a much higher cost & limiting the number of API calls allowed per day

Solution.

1) Increase the perks of app
2) Inform everyone , the team as well as the users itself about the issue so that telling we are on hold
3) Increase the budget and subscription prices
4) Notify people in a creative manner , so that people remain engaged with our app and doesn't lose interest , that the company is adding some features
5) Ask people some time before launching