# TripSplit – Software Product Proposal

CSC4898 – Senior Design Project

Team Submission – November 16, 2025

# 1. Product Information and Team Members

**Product Name and Team Name**: TripSplit

**Team members and Roles:**

| Name | Email | Role |
|---|---|---|
| **Allie Miller** | millera31@spu.edu | Frontend Development, UI/UX, Testing |
| **Shadman Shahzahan** | shahzahans@spu.edu | Backend Development |
| **Dawit Kelile** | keliled@spu.edu | AI/ML Integration, Frontend |
| **Kristen Lowe** | lowek3@spu.edu | Backend Support, QA, Bug Tracking |
| **Pedro Luiz** | teixeiraandp@spu.edu | Scrum Master, Lead Developer |

**Project Sponsor:**

- **Andy Cameron:** acameron@spu.edu

**External Advisors**

- **Martin Krienke:** mlkrienke@hotmail.com

# 2. Product

## 2.1 Problem Statement

Managing shared travel expenses is often messy and disorganized. Groups frequently lose receipts, forget who paid for what, and struggle with splitting itemized costs fairly. Existing tools like Splitwise help track totals but do not support detailed item-by-item scanning or real-time balance updates during trips. Travelers need a simple, automated solution that keeps expenses clear, accurate, and stress-free.

### 2.2 Product Description

TripSplit is a mobile application designed to streamline shared travel expenses. The app allows users to create trip groups, log expenses, and use AI-powered receipt scanning to extract totals and itemized entries. TripSplit automatically calculates fair splits and provides real-time balance updates, so groups can manage expenses transparently and efficiently.

## 3. Value Proposition

Group travel often leads to confusion over payments, lost receipts, awkward money discussions, and uneven expense contributions. TripSplit eliminates these challenges by:

- Automating receipt entry through AI scanning
- Providing instant balance updates showing who owes whom
- Supporting both even and custom splits for accuracy
- Offering a clean intuitive interface made for travel
- Keeping a trip expense history organized for reference

TripSplit removes the stress and awkwardness of shared finances so travelers can stay present, enjoy their trip, and leave the math to the app.

## 4. Intended Audience and Personas

### 4.1 Target Audience

- College students planning group trips
- Professionals traveling together
- Families splitting vacation expenses
- Any group needing transparent, automated cost management

### 4.2 Persona 1 – The Organizer (Jenna, 21)

**Occupation:** Undergraduate student majoring in Marketing

**Location:** Seattle, WA

**Bio:** Jenna loves planning weekend getaways with her friends. She scouts locations, books lodging and activities, and keeps the group organized. However, when it comes to handling shared expenses, she ends up doing the “trip accountant” role; collecting receipts, reminding friends to pay, and calculating who owes what.

**Goals:**

- To quickly log all group expenses so no one gets left behind
- To automate expense-splitting so she doesn't have to manually do the math
- To keep an easily accessible record of past trips for future reference

**Pain Points:**

- Regular calculations create confusion and delays
- Some friends forget to pay
- Receipts and things she paid for become forgotten after time

**Motivation:** She wants to reduce her admin burden so she can enjoy the trip rather than manage it

**Quote:** "I just want to know everyone paid their share and get back to enjoying the trip."

## 4.3 Persona 2 – The Fair Split Focused (Sarah, 26)

**Occupation:** Financial Analyst

**Location:** San Diego, CA

**Bio:** Sarah is very conscientious about fairness and budgeting. When she travels with friends, she wants an item-by-item breakdown. She dislikes getting stuck paying extra because someone else ordered a lot more than her. She values precision and transparency.

**Goals:**

- To have accurate itemized expense splits (especially when people order differently)
- To have visual proofs (e.g., scanned receipts)
- To avoid paying more than her fair share or being overcharged

**Pain Points:**

- Apps that only allow "even split" frustrate her
- People disagreeing later about who ordered which item

**Motivation:** She wants complete transparency and control.

**Quote:** "If I ordered less, I shouldn't end up paying more. Let's make the split clear."

### 4.4 Persona 3 – The Easygoing Traveler (Marcus, 24)

**Occupation:** Software Engineering Intern
**Location:** Portland, OR

**Bio:**
Marcus loves spontaneous trips and hanging out with friends. He doesn't enjoy dealing with numbers or receipts and tends to "go with the flow" when it comes to money. He's the kind of friend who says, "Just tell me what I owe." However, because he's not always paying attention, he sometimes forgets pending payments or accidentally underpays, leading to awkward moments later. He values convenience and doesn't like anything that feels too complicated.

**Goals:**

- To pay his share without having to think too much
- To get clear notifications about what he owes and when
- To avoid misunderstandings or owing people money for too long

**Pain Points:**

- Gets overwhelmed by apps that require a lot of manual input
- Forgets to pay after trips unless reminded
- Dislikes group tension caused by delayed payments

**Motivation:** He wants a simple, no-stress way to stay on top of his expenses without doing extra work.

**Quote:** "Just send me what I owe. I'll pay it immediately if it's simple."

# 5. MVP Product Backlog

## 5.1 Epics and User Stories

**Epic A: User & Trip Management**

- Users can create accounts and log in
- Users can update their profile and upload a profile picture
- Users can create new trips

- Users can add friends and search users by username
- Users can add or remove participants in a trip

**Epic B: Expense Entry**

- Users can log shared expenses (meals, gas, lodging, activities)
- Users can split expenses evenly or customize splits
- Users can mark expenses as settled

**Epic C: Receipt Scanning**

- Users can scan receipts and extract totals, date, and line items
- The system detects and prevents duplicate receipt uploads

**Epic D: Balance and Notifications**

- The app shows real time balances showing who owes whom what
- Users receive notifications for new expenses and settlements (i.e. “Sam paid you $20 for “Dinner at The Spaghetti Factory”)

**Epic E: Trip Feed and History**

- Users can view a trip activity feed
- Users can review past trips and export summaries

## 5.2 Technology Tasks

- Build backend API and database schema
- Implement secure user authentication
- Create React Native UI components
- Develop receipt OCR integration
- Implement offline first data storage
- Set up Android deployment pipeline
- Build unit and integration tests

## 5.3 Spikes

- Compare OCR providers (AWS Textract, Google Vision, Tesseract)
- Research offlice capabilities
- Prototype receipt duplicate detection
- Evaluate cross platform constraints for iOS support

# 6. Technology

## 6.1 Platform Availability

- Android (MVP release)
- IOS (post–MVP) using same React Native codebase

Android is chosen first for faster testing and easier deployment.

## 6.2 Deployment Environment (Tech Stack)

**Frontend:**

- React Native
- JavaScript

**Backend:**

- Node.js
- Express
- SQL (Relational Database)

**AI/OCR System:**

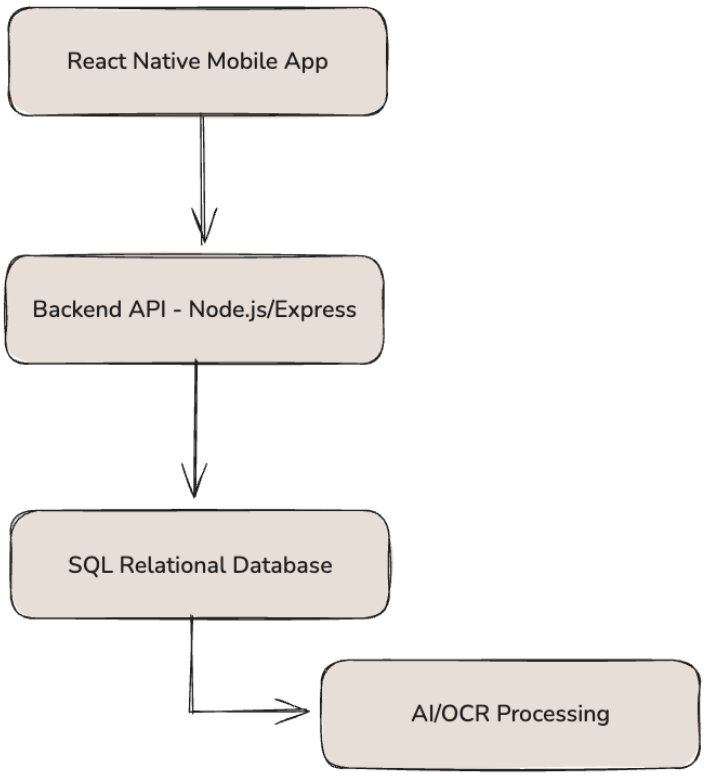
- Google ML Kit OCR library
    - If ML Kit doesn't meet accuracy needs, we will evaluate **Google Vision API** or **AWS Textract** for cloud-based OCR.

**Hosting and DevOps:**

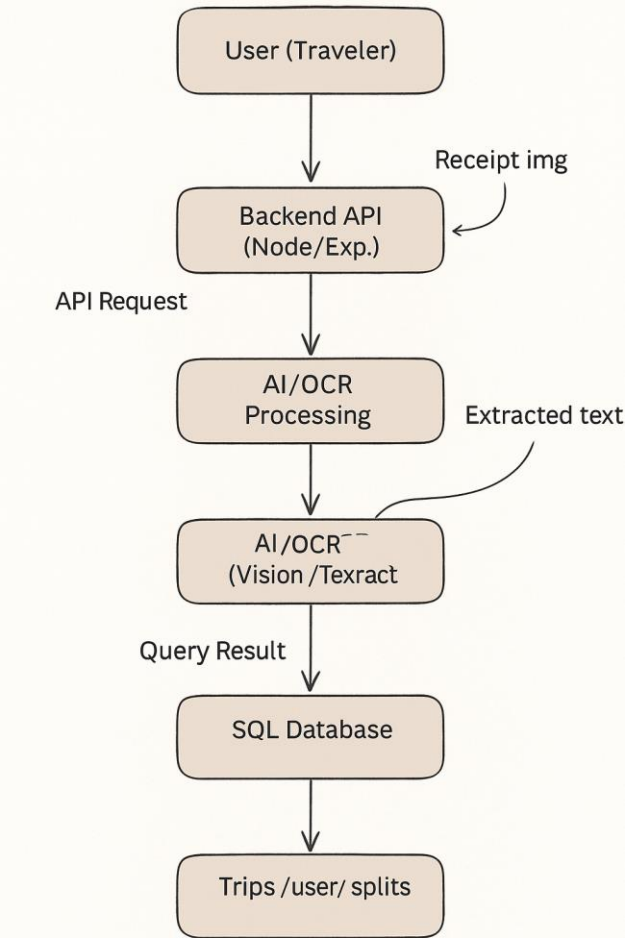
- Github Repository for version control
- AWS (likely **EC2 or AWS Amplify)** for backend hosting and deployment
- Android Studio for emulator testing and build generation

## 6.3 Software Architecture and Data Flow

**Architecture Overview:**


React Native Mobile App
Backend API - Node.js/Express
SQL Relational Database
AI/OCR Processing

**Data Flow Diagram & Description:**




1. A user creates or joins a trip
2. The user logs an expense or uploads a receipt
3. If a receipt is uploaded, the image is sent to the AI/OCR service for extraction (totals, taxes, line items, merchant, date)
4. The backend returns the extracted line items to the mobile app
5. The user reviews the line items and assigns each item to one or more travelers (who ordered what)
6. The app sends the final item-to-traveler assignments to the backend
7. The backend calculates each person's share and stores the expense, splits, and balances in the database
8. The app updates the balances and trip activity feed in real time

## 6.4 Development Tools

- Visual Studio Code

- Github
- Balsamiq for UI design
- Android Studio Emulator
- Teams for communication

## 6.5 Programming Languages

- JavaScript (frontend + backend)

## 6.6 AI Integration

**AI/OCR Provider Selection Criteria**

Since we have not yet finalized which OCR provider we will use (Google ML Kit, Google Vision API, or AWS Textract), our team will select a provider based on the following criteria:

- **Accuracy of Text Extraction:**
  Ability to reliably detect totals, taxes, dates, merchant names, and itemized line entries from a wide range of receipt types, including low-quality or poorly lit images.

- **Cost and Pricing Model:**
  Whether the provider offers a free tier (such as ML Kit), or charges per request (Vision API, Textract). We will compare cost per receipt scan during MVP and long-term usage.

- **Processing Speed / Latency:**
  Fast response times are essential. Our target is to produce receipt results in **under 2 seconds**. ML Kit provides offline, on-device performance, while cloud APIs may introduce delays.

- **Offline Support:**
  Since travelers may have limited connectivity, offline functionality is a strong advantage. ML Kit supports on-device OCR, while Vision API and Textract require internet access.

- **Ease of Integration with React Native:**
  The provider should have strong documentation, existing React Native libraries, and a straightforward API to reduce development time.

- **Line-Item Extraction Quality:**
  Cloud providers generally offer more accurate itemized line parsing. We will compare the ability to detect individual items, quantities, and prices during our OCR spike.

- **Rate Limits and Scalability:**
  APIs often include per-minute limits. We must ensure users can upload multiple receipts quickly without hitting usage caps.
- **Security and Privacy:**
  Whether the receipt image stays on the device (ML Kit) or must be uploaded to the cloud (Vision, Textract). On-device processing improves privacy and reduces risk.

**Conclusion:**

These criteria will guide our evaluation of OCR providers and ensure we select the option that balances accuracy, cost, performance, and travel-friendly offline capabilities.

# 7. Competitive Analysis

## 7.1 Similar Products

- **Venmo**
  - Payment platform only, not built for trips or cost tracking.
- **Splitwise**
  - Shared bill tracking.
- **Settle Up**
  - Helps track shared expenses internationally, but the interface can feel cluttered and less intuitive for quick trip logging.
- **Tricount**
  - Group expense tracking trips, but limited customization for itemized splits.

## 7.2 How TripSplit is Different

- Built specifically for travel groups
- AI driven itemized receipt scanning
- Offline support for remote travel scenarios
- Duplicate receipt detection