VAANI - Voice to Expense PRD

Strategic Context

Why Start with Expenses?

While VAANI's vision encompasses multiple use cases like invoice creation and party management, we're strategically starting with expense addition because:

1. **Lower Barrier to Entry**: Expense recording doesn't involve sensitive customer information, making users more comfortable with voice input
2. **High Frequency Use Case**: MSMEs record expenses multiple times daily, providing immediate value
3. **Simpler Implementation**: Less complex than invoice creation while still being a significant technical milestone
4. **Privacy Comfort**: User research shows hesitation using voice for invoices due to security concerns about customer data
5. **Proven Need**: Current expense tracking shows significant under-reporting; voice can capture expenses that go unrecorded

Success Hypothesis

Target: 100% increase in expense recording

- Users currently adding X expenses should add at least 2X expenses after voice activation
- Focus on capturing the ~60% of expenses that currently go unrecorded due to friction

Executive Summary

VAANI's first use case focuses on expense addition through voice commands, enabling MSMEs to quickly record business expenses in their natural language. This feature supports 8 languages with auto-detection and provides intelligent categorization using a combination of historical data and ML models.

Feature Overview

Core Capability

Users can add business expenses by speaking naturally in their preferred language. The system processes voice input to extract:

- Item name(s)
- Amount(s)
- Auto-categorizes expenses using historical data + AI

Key Design Principles

- **Voice-First**: Natural language processing in 8 supported languages
- **Data-Driven Categorization**: Leverages 1000+ pre-mapped items from 100K+ users
- **Intelligent Defaults**: Auto-categorization with smart defaults
- **Visual Confirmation**: Always show preview before committing
- **Flexible Editing**: Allow corrections in preview screen
- **Progressive Disclosure**: Guide users when information is missing

Detailed User Flow

```
flowchart TD
    Start([User Taps VAANI Button]) --> VoiceInterface[Voice Interface Opens]
    VoiceInterface --> ProcessInput[Process Voice Input]

    ProcessInput --> ExtractData[Extract: Amount, Item, Category]

    ExtractData --> ItemCheck{Item Identified?}

    ItemCheck -->|NO| AskItem[Ask: What did you spend on?]
    ItemCheck -->|YES| AmountCheck{Amount Identified?}

    AskItem --> UserProvidesItem[User Provides Item Details]
    UserProvidesItem --> AmountCheck

    AmountCheck -->|NO| AskAmount[Ask: What amount did you spend?]
    AmountCheck -->|YES| ItemLookup{Check Item in Excel Database}

    AskAmount --> UserProvidesAmount[User Provides Amount]
    UserProvidesAmount --> ItemLookup

    ItemLookup -->|Found in 1000 Items List| UseExcelCategory[Use Pre-mapped Category from
Excel]
    ItemLookup -->|Not Found| CategoryProcess{Was Category Mentioned in Voice?}

    CategoryProcess -->|YES| ValidateCategory{Validate Against Excel Categories}
    CategoryProcess -->|NO| SuggestCategory[ML Model Suggests Category]

    ValidateCategory -->|Found in Excel| UseVoiceCategory[Use Voice-Identified Category]
    ValidateCategory -->|Not Found| SuggestCategory

    SuggestCategory --> CheckExisting{Match with User's Existing Categories}
    CheckExisting -->|Match >70%| UseExisting[Use Existing Category]
    CheckExisting -->|Match <70%| CreateNew[Create New Category as Direct Expense]
    CheckExisting -->|No User Categories| UseGeneral[Use 'General' Category]

    UseExcelCategory --> ShowPreview[Show Confirmation Screen]
    UseVoiceCategory --> ShowPreview
```

```
    UseExisting --> ShowPreview
    CreateNew --> ShowPreview
    UseGeneral --> ShowPreview

    ShowPreview --> UserReview{User Reviews Details}

    UserReview -->|Edit Category| CategoryDropdown[Show Category Dropdown + Create New Option]
    CategoryDropdown --> UserSelectsCategory[User Selects/Creates Category]
    UserSelectsCategory --> ShowPreview

    UserReview -->|Edit Amount/Item| EditForm[Open Edit Form]
    EditForm --> UserEdits[User Makes Changes]
    UserEdits --> ShowPreview

    UserReview -->|Confirm| ValidateFields{All Fields Valid?}

    ValidateFields -->|NO| ShowError[Show Specific Error Message]
    ShowError --> ShowPreview

    ValidateFields -->|YES| SaveExpense[(Save Expense to Database)]

    SaveExpense --> ShowSuccess[Show Success Message/GIF]
    ShowSuccess --> PostSaveActions[Post-Save Actions]

    PostSaveActions --> ViewAll[View All Expenses Button]
    PostSaveActions --> AddAnother[Add Another Expense Button]

    UserReview -->|Cancel| CancelConfirm{Confirm Cancel?}
    CancelConfirm -->|YES| ReturnHome[Return to Voice Screen]
    CancelConfirm -->|NO| ShowPreview
```

Requirements Table

	Component	Description	Notes
1	Voice Input	Support 8 languages with auto-detection	Hindi, English, Tamil, Telugu, Bengali, Marathi, Gujarati, Kannada
2	Voice Input	Process Hinglish (code-mixed) commands	Common in Indian context
3	Data Extraction	Extract item names from voice	Can be single or multiple items
4	Data Extraction	Extract amount from voice	Required field

5	Data Extraction	Handle multiple items in single command	Max 10 items per command
6	Data Extraction	Single item interpretation for combined names	"Chai Samosa" = single item unless prices specified separately
7	Excel Database	Load 1000 items with pre-mapped categories	Historical data from 100K+ users
8	Excel Database	Load 1000 validated expense categories	2-3 years of usage patterns
9	Categorization	Check Excel database first for item match	Primary categorization method
10	Categorization	Auto-categorize based on item names	ML model as secondary method
11	Categorization	Check similarity with existing categories	70% threshold for matching
12	Categorization	Create new category if no match found	All new categories are "Direct Expense"
13	Categorization	Default to "General" for uncategorized	Fallback category
14	Preview Screen	Show editable expense details	Item, amount, category, date
15	Preview Screen	Allow category change	Dropdown with existing categories
16	Preview Screen	Allow creating new category	Text input option
17	Preview Screen	Edit individual item amounts	For multi-item expenses
18	Voice Tag	Add "Added via voice" indicator	Visual indicator in expense list

19	Success Action	Show "View all expenses" CTA	Primary action after save
20	Success Action	Option to "Add another expense"	Secondary action
21	Error Handling	Handle unsupported commands	Clear messaging about current capabilities
22	Settings	Request more use cases option	Feedback collection
23	Settings	Show 6-7 predefined options	Common requested features
24	Settings	Allow custom feedback text	Free text input
25	Settings	Max 3 selections for feature requests	Limit user choices
26	Response Language	Always respond in English	Regardless of input language
27	Data Limit	Process maximum 10 items per command	Ignore items beyond 10

Platform Considerations

	Feature/Component	Mobile	Desktop	Notes
1	VAANI Voice Button	✅ Available	❌ Not Available	Floating button on all mobile screens
2	Voice Input Processing	✅ Available	❌ Not Available	Mobile-only in MVP
3	Voice Interface UI	✅ Available	❌ Not Available	Listening animation,

				transcription display
4	Expense Preview/Edit Screen	✅ Available	❌ Not Available	After voice processing
5	Settings - Request Features	✅ Available		Accessible from both platforms
6	Voice-Added Tag Display	✅ Available		Shows "Added via voice" indicator
7	Expense List View	✅ Available	✅ Available	Standard expense management
8	View Voice-Added Expenses	✅ Available	✅ Available	Filter/view voice expenses
9				
10				
11				
12				

Edge Cases & Scenarios

Input Processing

	Scenario	Example Input	System Behavior
1	Multiple items with individual prices	"Chai 60 rupees, samosa 80 rupees"	Creates single expense with 2 line items if prices identified separately

2	Combined item name	"Chai Samosa for 140 rupees"	Creates single item "Chai Samosa" with ₹140
3	Multiple items with total amount	"Chai and samosa for 140 rupees"	Creates single expense, may split proportionally
4	Unclear amount	"Around 100 rupees"	Processes as exactly ₹100
5	No amount mentioned	"Bought vegetables"	Prompts: "Please mention the amount"
6	No item mentioned	"Spent 500"	Prompts: "What was this expense for?"
7	Amount in words	"Paanch sau rupees"	Converts to ₹500
8	Mixed language	"Taxi ke liye two hundred diye"	Processes correctly as ₹200 for Taxi
9	More than 10 items	Lists 11+ items	Processes first 10 items only
10	Zero or negative amount	"Minus 100 rupees"	Shows error: "Please enter a valid amount"
11	Extremely large amount	"10 crore rupees for building"	Confirms with user due to large amount
12	Item in Excel database	"Petrol 500 rupees"	Auto-categorizes to "Fuel" from Excel
13	Item not in database	"Zoom subscription 1000 rupees"	ML model suggests "Software" category

Categorisation Logic

Three-Tier Categorization System

1. **Excel Database Lookup (Primary)**
   - 1000 pre-mapped items from 2-3 years of usage data
   - 1000 categories validated by 100K+ Vyapar users

- Direct mapping for common expenses

2. **ML Model Suggestion (Secondary)**
   - When item not found in Excel database
   - Uses context and patterns from similar businesses
   - 70% similarity threshold for matching
3. **User Categories (Tertiary)**
   - Checks user's existing categories
   - Creates new category if no match found
   - Defaults to "General" as last resort

> Need to work more on this. WIP

	Scenario	Behavior	Priority
1	Item found in Excel database	Uses pre-mapped category	1st
2	Category mentioned and in Excel	Uses voice-identified category	2nd
3	ML model suggests category	Checks against user's existing categories	3rd
4	Similar category exists (>70% match)	Assigns to similar category	3rd
5	No match found (<70% similarity)	Creates new category with item name	4th
6	Multiple categories possible	Selects highest confidence match	•
7	User creates custom category	Always creates as "Direct Expense" type	•
8	No categories in system	Defaults to "General" category	Last

Error Scenarios

	Error	User Message	Recovery

1	Voice recognition fails	"Couldn't understand. Please try again"	Retry button
2	Network timeout	"Connection lost. Please check internet"	Offline mode or retry
3	Unsupported command	"Currently, I can only add expenses. More features coming soon!"	Show supported commands
4	Invalid amount format	"Please mention a valid amount"	Voice retry or manual input
5	System error	"Something went wrong. Please try again"	Fallback to manual entry
6	Excel database load failure	"Using basic categorisation"	Falls back to ML model only

Unsupported Commands Handling

When users try commands outside expense addition:

Detection Patterns

- "Add a sale" / "Create invoice" → Sales flow
- "Add party" / "Add customer" → Party management
- "Show reports" / "View profit" → Analytics
- "Change settings" → Configuration

Response Flow

```
User: "Create a sale for Ramesh"
         ↓
System: "I can currently only add expenses.
         This feature is coming soon!"
         ↓
Show: [Add Expense] [Request Features] [Learn More]
```

Settings Screen

"Request More Use Cases" Flow

1. **Entry Point**: Settings → VAANI Settings → Request Features

2. **Screen Layout**:

```
What would you like VAANI to do?

□ Create sales/invoices
□ Add new parties/customers
□ View reports & analytics
□ Manage inventory
□ Send payment reminders
□ Change app settings
□ Generate GST returns

[Select up to 3]

OR

[Text field: Tell us what you need...]

[Submit]
```

3. **Submission**: Sends to product team for prioritization

Data Model

Expense Entity (Voice-Added)

```
{
  "expense_id": "EXP_20250827_001",
  "items": [
    {
      "name": "Chai Samosa",
      "amount": 140,
      "quantity": 1
    }
  ],
  "total_amount": 140,
  "category": "Food",
  "category_type": "Direct Expense",
  "category_source": "excel_database", // or "ml_model" or "user_created"
  "date": "2025-08-27",
  "payment_type": "Cash",
  "added_via": "voice",
  "voice_metadata": {
    "language_detected": "hi-IN",
    "confidence_score": 0.92,
    "original_transcript": "chai samosa ke liye 140 rupees",
    "categorization_method": "excel_match" // or "ml_suggestion" or "user_default"
  }
}
```

Important Notes

⚠️ Critical Requirements:

1. Excel database of 1000 items and 1000 categories must be loaded on app start or backend support
2. All new categories created via voice are "Direct Expense" type
3. Maximum 10 items processed per voice command
4. Response always in English regardless of input language
5. Voice tag/indicator must be visible in expense list
6. No edit/delete operations via voice in MVP
7. Create operations only for expense use case
8. "Chai Samosa" as single phrase = single item unless prices are specified separately
9. Category matching uses 70% similarity threshold from ML model
10. Excel database takes priority over ML suggestions for categorization

Success Metrics

- **Primary Goal**: 100% increase in expense recording for voice users
- 80% successful expense creation from voice
- <3 seconds processing time
- 70% auto-categorization accuracy via Excel database
- 60% of users complete expense addition without manual edits
- 40% of users who try voice use it again within 7 days
- Capture previously unrecorded expenses (target: 60% of missing expenses)

Dependencies

- Data Science team: Language models and categorization ML
- Backend team: Expense API modifications for voice metadata
- Mobile team: Voice UI implementation
- Design team: Voice interaction patterns and animations
- Analytics team: Excel database preparation from historical data

Appendix

Supported Languages

1. Hindi (hi-IN)
2. English (en-IN)
3. Tamil (ta-IN)
4. Telugu (te-IN)

5. Bengali (bn-IN)
6. Marathi (mr-IN)
7. Gujarati (gu-IN)
8. Kannada (kn-IN)