**Mingus Financial Profile Screen - Implementation Guide**

**Screen Purpose**

Collect essential financial data needed for cash flow forecasting, milestone planning, and personalized insights. This bridges the gap between your health/career questionnaires and your financial analytics features.

**Database Schema Requirements**

-- Financial profile data table

CREATE TABLE user\_financial\_profile (

id uuid PRIMARY KEY DEFAULT gen\_random\_uuid(),

user\_id uuid REFERENCES auth.users(id) NOT NULL,

monthly\_income decimal(10,2),

income\_frequency text CHECK (income\_frequency IN ('weekly', 'bi\_weekly', 'monthly', 'semi\_monthly')),

income\_sources jsonb, -- Array of income sources with amounts

monthly\_fixed\_expenses decimal(10,2),

monthly\_variable\_expenses decimal(10,2),

current\_savings decimal(10,2),

current\_debt decimal(10,2),

emergency\_fund\_months decimal(3,1),

financial\_goals jsonb, -- Array of goals with target amounts and dates

risk\_tolerance text CHECK (risk\_tolerance IN ('conservative', 'moderate', 'aggressive')),

budgeting\_experience text CHECK (budgeting\_experience IN ('none', 'basic', 'intermediate', 'advanced')),

created\_at timestamp DEFAULT now(),

updated\_at timestamp DEFAULT now()

);

**Frontend Implementation Prompts**

**1. Financial Profile Component Structure**

**Prompt for Developer:**

Create a React component called FinancialProfile.tsx for the Mingus onboarding flow. The component should:

1. Use a multi-step form with progress indicator

2. Include the following sections in order:

- Income Information

- Monthly Expenses

- Current Financial Position

- Financial Goals

- Risk Assessment

3. Style with Tailwind CSS using the Mingus color scheme:

- Primary: #1a1a2e

- Secondary: #16213e

- Accent: #00d9ff

- Success: #4caf50

4. Include form validation and error handling

5. Save progress automatically on each step

6. Connect to the /api/financial-profile endpoint

The component should feel modern, trustworthy, and specifically designed for young African American professionals aged 25-35.

**2. Income Information Section**

**Prompt for Developer:**

Create the Income Information section for the Financial Profile screen. Include:

1. Primary income amount input with monthly conversion display

2. Income frequency selector (Weekly, Bi-weekly, Monthly, Semi-monthly)

3. Secondary income sources with add/remove functionality

4. Income stability assessment (1-10 scale: "How stable is your primary income?")

5. Next pay date picker for cash flow calculations

UI Requirements:

- Use currency formatting for all money inputs

- Show real-time monthly equivalent calculations

- Include helpful tooltips explaining why we need this information

- Use progressive disclosure - start simple, allow adding complexity

Validation:

- Primary income required and must be > 0

- Total income should be reasonable (warn if < $20k or > $500k annually)

- Income frequency required

**3. Monthly Expenses Section**

**Prompt for Developer:**

Create the Monthly Expenses section with smart categorization:

1. Fixed Expenses (rent, insurance, loans):

- Pre-populated common categories

- Custom category addition

- Total fixed expenses calculation

2. Variable Expenses estimation:

- Spending category sliders (Food, Entertainment, Shopping, etc.)

- Percentage-based or dollar amount options

- Visual breakdown pie chart

3. Expense-to-income ratio warning:

- Show percentage of income going to expenses

- Color-coded warnings (green < 50%, yellow 50-80%, red > 80%)

4. Quick setup options:

- "I'll connect my bank later" button

- "Import from another app" option

- "Manual entry" (current flow)

Include helpful copy like: "Don't worry about being exact - we'll help you refine this over time."

**4. Current Financial Position Section**

**Prompt for Developer:**

Create the Current Financial Position section focusing on:

1. Current Savings:

- Checking account balance

- Savings account balance

- Emergency fund months calculation

- Visual progress bar for emergency fund goal

2. Current Debt:

- Total debt amount (auto-calculated in expenses)

- Monthly debt payments (auto-calculated in expenses)

- Debt-to-income ratio display

- Debt payoff timeline estimation

3. Financial Health Score:

- Real-time calculation based on inputs

- Simple 0-100 score with color coding

- Brief explanation of score factors

4. Reassuring messaging:

- "This information stays private and secure"

- "We're here to help improve your financial health"

- Progress indicators showing they're building toward better financial wellness

**5. Financial Goals Section**

**Prompt for Developer:**

Create the Financial Goals section with:

1. Goal Categories:

- Emergency Fund (auto-suggested based on expenses)

- Major Purchase (house, car, etc.)

- Vacation/Travel

- Debt Payoff

- Investment/Retirement

- Custom goal

2. For each goal:

- Goal name and category

- Target amount

- Target date

- Current progress (optional)

- Priority level (1-5 stars)

3. Goal Feasibility Analysis:

- Auto-calculate monthly savings needed

- Show if goal is realistic given current income/expenses

- Suggest timeline adjustments if needed

4. Visual Goal Tracker:

- Timeline view of all goals

- Conflict detection (overlapping major expenses)

- Progress visualization

UI should be inspiring and motivating, not overwhelming.

**6. Risk Assessment Section**

**Prompt for Developer:**

Create the Risk Assessment section with:

1. Risk Tolerance Quiz:

- 3-5 scenario-based questions

- Visual scenarios rather than just text

- Results in conservative/moderate/aggressive classification

2. Financial Experience Assessment:

- Budgeting experience level

- Investment experience

- Previous financial app usage

3. Preferences:

- Preferred notification frequency

- How aggressive coaching should be

- Areas of focus (saving vs. debt vs. investing)

4. Final Summary:

- Review of all entered information

- Financial health score

- Next steps preview

- "Complete Setup" button leading to dashboard

Include encouraging copy about their financial journey starting now.

**Backend Implementation Prompts**

**1. API Endpoint Creation**

**Prompt for Backend Developer:**

Create a comprehensive financial profile API in backend/routes/financial\_profile.py:

1. POST /api/financial-profile - Save/update user's financial profile

2. GET /api/financial-profile/{user\_id} - Retrieve profile data

3. POST /api/financial-profile/calculate-health-score - Real-time score calculation

4. PUT /api/financial-profile/goals - Update specific goals

5. DELETE /api/financial-profile/goals/{goal\_id} - Remove goal

Include:

- Input validation for all financial data

- Automatic calculations (debt-to-income ratio, emergency fund months, etc.)

- Integration with existing daily\_cashflow calculations

- Trigger cash flow forecast recalculation when profile updates

Security: Ensure all financial data is properly encrypted and access-controlled.

**2. Integration with Existing Systems**

**Prompt for Backend Developer:**

Update the existing cash flow calculation system to use financial profile data:

1. Modify calculate\_enhanced\_daily\_cashflow() function to:

- Pull income data from financial profile instead of hardcoded values

- Use actual expense data for more accurate forecasting

- Factor in financial goals as future expenses

2. Update onboarding progress tracking:

- Mark financial profile completion

- Calculate onboarding percentage

- Trigger welcome email when financial setup complete

3. Create financial profile validation:

- Check for required fields before allowing dashboard access

- Provide completion percentage for partial profiles

- Generate suggestions for missing data

**User Experience Flow**

**1. Entry Points**

* **From Registration**: After email verification, before other questionnaires
* **From Dashboard**: If financial profile incomplete, show prominent setup card
* **From Settings**: Allow editing/updating profile anytime

**2. Progressive Disclosure Strategy**

1. **Start Simple**: Just income and major expenses
2. **Build Confidence**: Show immediate insights and calculations
3. **Add Complexity**: Goals, detailed categories, preferences
4. **Provide Value**: Financial health score and first recommendations

**3. Completion Incentives**

* Show financial health score improving in real-time
* Unlock dashboard features as sections complete
* Provide immediate cash flow forecast preview
* Celebrate milestones ("You're 60% complete!")

**Integration with Existing Features**

**1. Health-Spending Correlations**

* Use financial profile to personalize spending categories
* Compare actual spending to budgeted amounts
* Factor in financial stress when analyzing health impacts

**2. Job Security Analytics**

* Use income and savings data to calculate employment risk impact
* Recommend emergency fund targets based on actual expenses
* Personalize job security recommendations based on financial position

**3. Milestone Planning**

* Populate initial milestones from financial goals
* Use income/expense data to calculate milestone feasibility
* Provide realistic timeline recommendations

**Success Metrics**

* **Profile Completion Rate**: Target 85%+ of registered users
* **Data Quality**: Complete profiles lead to 3x better engagement
* **User Satisfaction**: Financial insights rated as "very helpful" by 80%+ users
* **Retention**: Users with complete profiles have 60% higher 30-day retention

**Privacy and Security Considerations**

* All financial data encrypted at rest and in transit
* Clear privacy policy explaining data usage
* Option to delete financial data anytime
* No sharing of individual financial information
* Aggregate, anonymized insights only

This financial profile screen is the critical missing piece that will make your health-finance correlation insights truly powerful and personalized.