# Budget Tool Requirements Document

## Overview

The \*\*Budget Tool\*\* is a web-based budgeting application designed to let users create, edit, and track income and expenses in an intuitive, visually appealing layout inspired by a budget worksheet template. The tool calculates savings dynamically and organizes expenses into categories and subcategories.

---

## Features & Functionality

### Budget Management

- Default budget name: \*\*Personal Budget\*\*.

- User can edit the budget name directly via a text field.

- Edit and Delete buttons available for the overall budget.

### Income

- Ability to \*\*add, edit, and delete income entries\*\*.

- Each income entry contains:

- \*\*Amount\*\* (monthly, numeric input).

- \*\*Source\*\* (text input).

- \*\*Comments\*\* (optional text input).

- After adding or editing, form fields reset to default values.

- Income is listed below the Income section with:

- Source name.

- Amount.

- Edit button next to each entry.

- \*\*Total Income\*\* is calculated and displayed at the bottom of the section.

### Expenses

- Ability to \*\*add, edit, and delete expense entries\*\*.

- Each expense entry contains:

- \*\*Expense Category\*\* (dropdown, editable list of predefined categories).

- \*\*Amount\*\* (monthly, numeric input).

- \*\*Pay Day\*\* (dropdown, 1–30).

- \*\*Payee\*\* (text input).

- \*\*Notes\*\* (optional text input).

- Expense categories are grouped:

- If a category has \*\*subcategories\*\* (e.g., `Food-Lunch`, `Food-Groceries`), it is displayed under the main category (`Food`).

- Subcategories are \*\*indented and italicized\*\*.

- Standalone categories (e.g., `Gas`, `Mortgage`) do not display a subcategory structure.

- Each category and subcategory has an \*\*Edit button\*\*.

- Delete functionality is accessed via the \*\*Edit dialog\*\*.

- \*\*Total Expenses\*\* are calculated and displayed at the bottom of the section.

### Savings Calculation

- Automatically calculated as:

- `Savings = Total Income - Total Expenses`

- Displayed at the bottom of the budget in a bold font.

- If negative, value is styled in \*\*red\*\*.

- If positive, value is styled in \*\*green\*\*.

---

## Design & UI Elements

### Layout

- Header styled like a budget worksheet:

- \*\*Teal header bar\*\* with title: \*BUDGET WORKSHEET\*.

- Subtitle: \*Get your budget in order once and for all\*.

- \*\*Card-style container\*\* with rounded corners, soft shadows, and section dividers.

### Section Headers

- \*\*Income Section:\*\* Pink header bar.

- \*\*Expenses Section:\*\* Green header bar.

- \*\*Savings Section:\*\* Border-top divider, large bold calculation.

### Forms

- Modal dialogs (`Dialog`) for adding/editing income and expense entries.

- Inputs reset after submission.

- Save and Delete buttons inside dialogs.

### Styling

- Modern, minimal, color-coded sections.

- Subcategories are italicized and indented.

- Positive/negative values are color-coded (green/red).

---

## Developer Notes & Coding Decisions

1. \*\*State Management\*\*

- `useState` used for income, expenses, new entry forms, and editing indexes.

- Separate states for new income (`newIncome`) and new expense (`newExpense`).

2. \*\*Category Grouping Logic\*\*

- Expense categories split by `-`.

- Main categories displayed with totals.

- Subcategories listed beneath main categories only when they exist.

3. \*\*Editing & Deleting\*\*

- Editing is done by pre-filling the input fields inside a dialog.

- Deletion is handled inside the Edit dialog (not directly in the main list).

4. \*\*Calculations\*\*

- Totals computed dynamically with `reduce`.

- Savings recalculated on each state update.

5. \*\*UI Libraries\*\*

- \*\*shadcn/ui\*\* components: `Button`, `Card`, `Input`, `Select`, `Dialog`.

- TailwindCSS for styling.

6. \*\*Reset Behavior\*\*

- After saving, forms reset to default values.

- Ensures consistent UX and prevents old values from persisting.

7. \*\*Accessibility & UX\*\*

- Clear headings, color coding, and spacing.

- Confirmation dialogs for delete actions (handled within edit flow).

---

## Future Enhancements (Optional)

- Add \*\*Expected vs. Actual vs. Difference\*\* columns for income and expenses.

- Export budget to \*\*CSV, PDF, or Excel\*\*.

- Add recurring income/expense scheduling.

- Support multiple budgets with tabs or a dashboard.

- Authentication for saving budgets per user profile.