8/28/2023

|  |
| --- |
| Anvil-Procurement |

# User Story

# As a corporate procurement manager,

# I want a web app that facilitates purchasing from commonly used vendors,

# So that I can efficiently manage budgets, track spending, access a list of frequently purchased items, and generate and email purchase orders (POs).

# Acceptance Criteria:

# Budget Tracking:

# The app should allow me to set budget limits for specific categories or vendors. I should receive notifications when the budget limit for a category or vendor is close to being exceeded. The app should provide visual representations of budget utilization. Vendor Management:

# I should be able to maintain a list of commonly used vendors with their contact information. The app should allow me to categorize vendors for easier management. Purchase Tracking:

# Every purchase made through the app should be recorded and associated with the appropriate category and vendor. I should be able to filter and sort purchases based on date, category, vendor, and amount. Frequently Purchased Items:

# The app should compile a list of frequently purchased items based on historical data. This list should help me identify potential bulk purchase opportunities or negotiate better deals. Purchase Orders (POs):

# I should be able to create and customize purchase orders directly within the app. POS should automatically include vendor information, item details, quantities, and pricing. After creating a PO, I should have the option to review and edit it before finalizing. Once finalized, the app should generate a printable version of the PO in PDF format. I should be able to email the finalized PO directly from the app to the respective vendor. User-Friendly Interface:

# The app should have an intuitive and user-friendly interface to ensure easy navigation and usage. Additional Notes:

# Integrations with accounting and finance systems could enhance the accuracy of budget tracking. The app should maintain data security and user authentication to protect sensitive corporate information. Regular updates and improvements to the app should be part of its development roadmap.

# Front End Tecnology

# React

# Vite

# MUI

# Back End Tecnology

# Sqlize

# Node

# express

# express router

# File Structure (React)

# |-- components/

# | |-- common/

# | | |-- Button.js

# | | |-- Input.js

# | | |-- ...

# | |-- dashboard/

# | | |-- BudgetTracker.js

# | | |-- PurchaseHistory.js

# | | |-- ...

# | |-- vendors/

# | | |-- VendorList.js

# | | |-- VendorDetails.js

# | | |-- ...

# |-- containers/

# | |-- DashboardContainer.js

# | |-- VendorContainer.js

# | |-- ...

# |-- context/

# | |-- BudgetContext.js

# | |-- VendorContext.js

# | |-- ...

# |-- hooks/

# | |-- useBudget.js

# | |-- useVendors.js

# | |-- ...

# |-- services/

# | |-- api.js

# |-- App.js

# |-- index.js

# |-- index.css

# SQL Schema

# CREATE TABLE users (

# id SERIAL PRIMARY KEY,

# username VARCHAR(50) NOT NULL,

# email VARCHAR(100) NOT NULL,

# password VARCHAR(100) NOT NULL

# );

# CREATE TABLE vendors (

# id SERIAL PRIMARY KEY,

# name VARCHAR(100) NOT NULL,

# contact\_name VARCHAR(100),

# contact\_email VARCHAR(100),

# contact\_phone VARCHAR(20)

# );

# CREATE TABLE categories (

# id SERIAL PRIMARY KEY,

# name VARCHAR(100) NOT NULL

# );

# CREATE TABLE items (

# id SERIAL PRIMARY KEY,

# name VARCHAR(200) NOT NULL,

# description TEXT,

# category\_id INT REFERENCES categories(id),

# vendor\_id INT REFERENCES vendors(id)

# );

# CREATE TABLE budgets (

# id SERIAL PRIMARY KEY,

# user\_id INT REFERENCES users(id),

# category\_id INT REFERENCES categories(id),

# budget\_amount DECIMAL(10, 2) NOT NULL

# );

# SQL Schema cont.

# CREATE TABLE purchases (

# id SERIAL PRIMARY KEY,

# user\_id INT REFERENCES users(id),

# item\_id INT REFERENCES items(id),

# purchase\_date DATE NOT NULL,

# quantity INT NOT NULL,

# unit\_price DECIMAL(10, 2) NOT NULL

# );

# CREATE TABLE purchase\_orders (

# id SERIAL PRIMARY KEY,

# user\_id INT REFERENCES users(id),

# vendor\_id INT REFERENCES vendors(id),

# order\_date DATE NOT NULL,

# total\_amount DECIMAL(10, 2) NOT NULL

# );

# CREATE TABLE po\_items (

# id SERIAL PRIMARY KEY,

# purchase\_order\_id INT REFERENCES purchase\_orders(id),

# item\_id INT REFERENCES items(id),

# quantity INT NOT NULL,

# unit\_price DECIMAL(10, 2) NOT NULL

# );

# Basic Wireframe

--------------------------------

| Corporate Dashboard |

--------------------------------

| [Budget Tracker] |

|------------------------------|

| Budget: $10,000 |

| Spent: $7,500 |

| Remaining: $2,500 |

|------------------------------|

| [Purchase History] |

|------------------------------|

| Date Item $ |

|------------------------------|

| 2023-08-01 Office Supplies $ |

| 2023-08-02 Electronics $ |

| 2023-08-03 Stationery $ |

| ... $ |

|------------------------------|

| [Frequently Purchased Items] |

|------------------------------|

| 1. Office Paper |

| 2. Ballpoint Pens |

| 3. Laptops |

| 4. ... |

|------------------------------|

| [Create Purchase Order] |

|------------------------------|

| Vendor: [Dropdown] |

| Item: [Dropdown] |

| Quantity: [Input] |

|------------------------------|

| [Submit PO] |

--------------------------------