

Name: Alexander Ray

Title: *Cinnamon: Money Management in a Spicier Package*

References: Jacob Hallberg for idea on encapsulating user information and addresses.

Both course Teaching Assistants for help structuring my class diagram, understanding requirements.

Wireframe Mockflow for UI Mockups

Visual Paradigm for class diagram

https://en.wikipedia.org/wiki/Strategy_pattern

https://en.wikipedia.org/wiki/Factory_method_pattern

Project Summary: *Cinnamon* is a personal money management app—akin to Mint—to help track spending and upcoming expenses in the context of the user’s priorities, including support for report generation. This project will facilitate smarter, more conscientious spending habits from its users and will incubate a goal-oriented mindset to money management.

Specifically, *Cinnamon* will provide utilities for spending tracking for day-to-day purchases. Factoring in information about the user including location and income, *Cinnamon* can track spending and provide up-to-date information as well as report generation with a number of different options.

Project Requirements:

Use Case ID	Use Case Name	Description
1	Sign Up	As a user, I want to be able to sign up for <i>Cinnamon</i> and set attributes about myself including location and income so that my information can be included in generated reports
2	Log In	As a user, I want to be able to log in to <i>Cinnamon</i> so that I can access my spending information
3	Log Out	As a user, I want to be able to log out of <i>Cinnamon</i> so that my spending information is no longer visible
4	Add Account	As a user, I want to be able to add a new account so that I can track spending out of individual sources
5	View Spending	As a user, I want to be able to view each spending instance so that I can track my expenses
6	Log Spending	As a user, I want to be able to log a spending instance, including the type, amount, account, and date so that they're available as a list or in reports
7	Generate Report	As a user, I want to be able to generate reports spending so that I can access this data with external programs

UI Mockups and Class Diagram

Notes:

After extensive conversations with both TA's, I settled on separating most controller logic out into controller classes instead of keeping functionality with a model class. An example of this is having a `AccountController` to handle adding an account instead of leaving that functionality inside `User`. As this method is passed a `User` object, it's intended to do something along the lines of `userObj.getAccounts.add()` within that method.

Also, related to controllers, after speaking with both TAs we decided controllers do not need state in this case and therefore do not have attributes.

In general, I was seeking to lay out the framework for a factory method design pattern for spending instances and the strategy design pattern for generating reports (as we did in class). In the future, I'm considering memento for saving state or possibly a decorator to add functionality to report generation.

The mockup shows a window titled "Cinnamon" with a standard macOS-style title bar (three small circles on the left). The main content area is centered and contains the following elements:

- A heading "Sign Up" in a simple, hand-drawn font.
- A text input field labeled "Username".
- A text input field labeled "Password".
- An "Income" label followed by a text input field containing "20000" and a small spinner control (up and down arrows).
- A text input field labeled "Street Address".
- A text input field labeled "City".
- A text input field labeled "State".
- A text input field labeled "Zip".
- A rectangular button labeled "Sign Up" positioned below the address fields.

Figure 1: UI Mockup for signup page. User adds username, password, and all necessary information. After this page, there are no options in current mockups to alter this data.

The image shows a UI mockup for a login page within a window titled "Cinnamon". The window has a standard macOS-style title bar with three small circles on the left. The main content area is centered and contains the following elements:

- The text "Login to Cinnamon" is centered above the input fields.
- A text input field labeled "Username" is positioned below the title.
- A text input field labeled "Password" is positioned below the "Username" field.
- A button labeled "Login" is positioned below the "Password" field.

Figure 2: UI Mockup for login page. Self-explanatory.

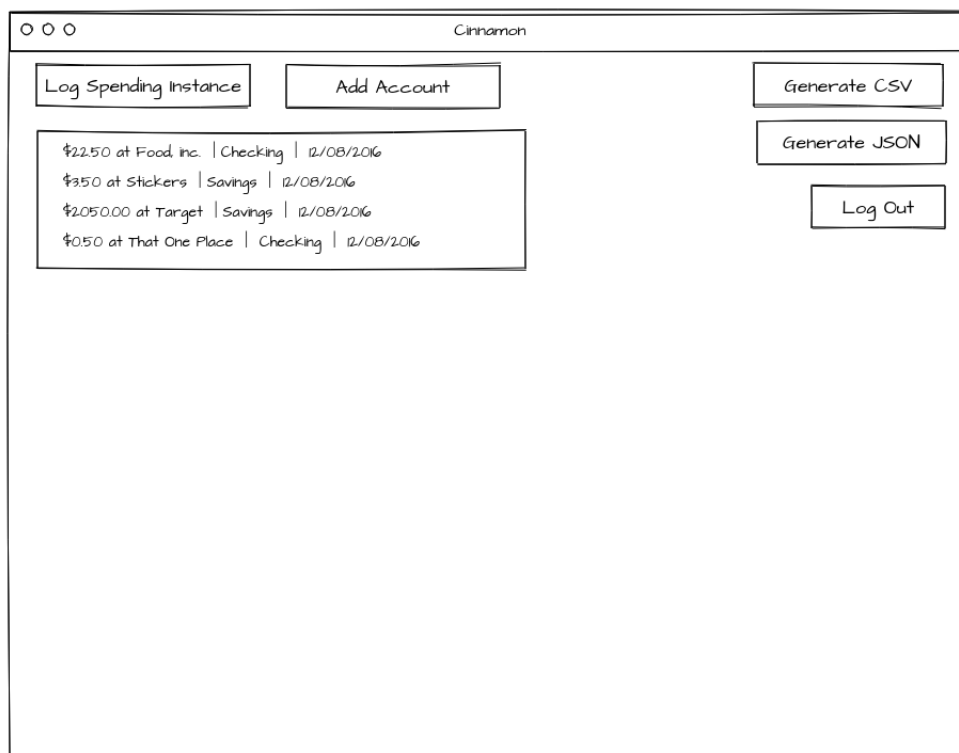
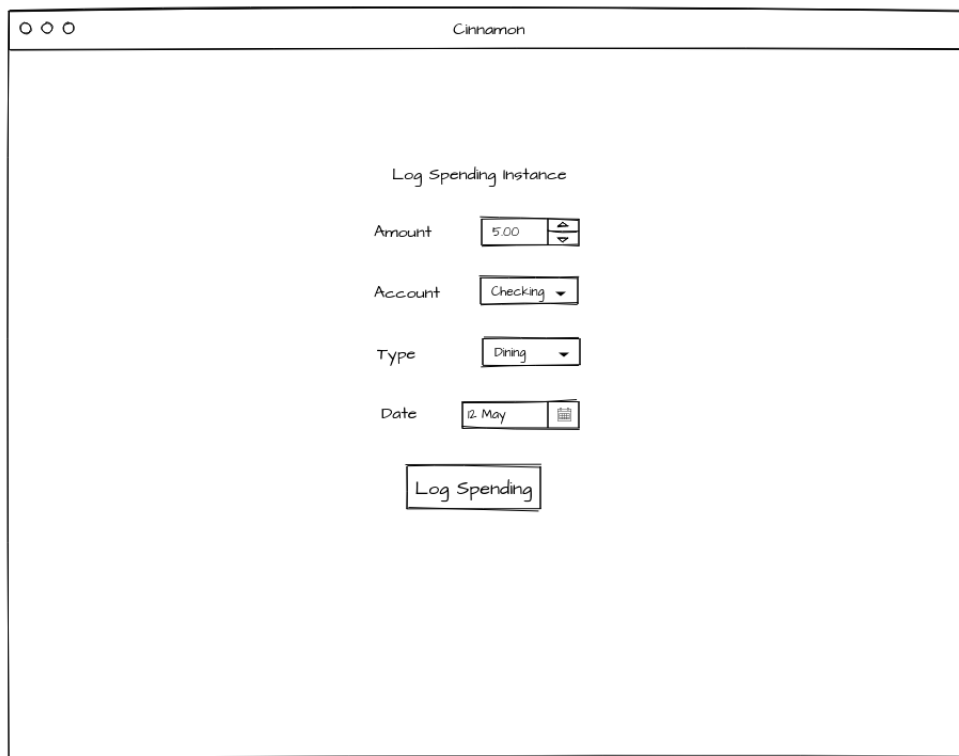


Figure 3: UI Mockup for home page. A main view to show spending instances. Also includes ways to log spending instances, generate reports, add accounts, and log out.



A UI mockup for a 'Log Spending Instance' page. The window has a title bar with three window control buttons and the text 'Cinnamon'. The main content area is titled 'Log Spending Instance' and contains four input fields: 'Amount' with a numeric input showing '5.00' and up/down arrows; 'Account' with a dropdown menu showing 'Checking'; 'Type' with a dropdown menu showing 'Dining'; and 'Date' with a date input showing '12 May' and a calendar icon. Below these fields is a 'Log Spending' button.

Figure 4: UI Mockup for log spending page. Form inputs for a spending instance. Options for accounts will be from the user's set accounts.



A hand-drawn UI mockup for an "Add Account" page. The mockup is contained within a rectangular frame representing a window. At the top of the window is a title bar with three small circles on the left and the text "Cinnamon" on the right. Centered in the main area of the window is the text "Add Account". Below this text is a horizontal rectangular input field with the word "Name" written inside it. Directly beneath the input field is a rectangular button with the word "Add" written inside it.

Figure 5: UI Mockup for add account page. Form input to add a new account.

