

Plateful: plan, shop, eat, repeat.

GROUP 2-8 PROJECT ANALYSIS

Anette Hardy Sosa & Shulin Xu

CONTENTS

Project Description	2
Story Mapping (User)	2
(Selected) User Stories	3
Story 1 – Signup	3
Story 2 – Managing multiple dates	4
Story 3 – Managing meals for individual days	5
Story 4 - Shopping List	6
Story 5 - Recipe Management.....	7
Domain Model	8
Conceptual Model	9
Logical Model	9

PROJECT DESCRIPTION

Title: Plateful

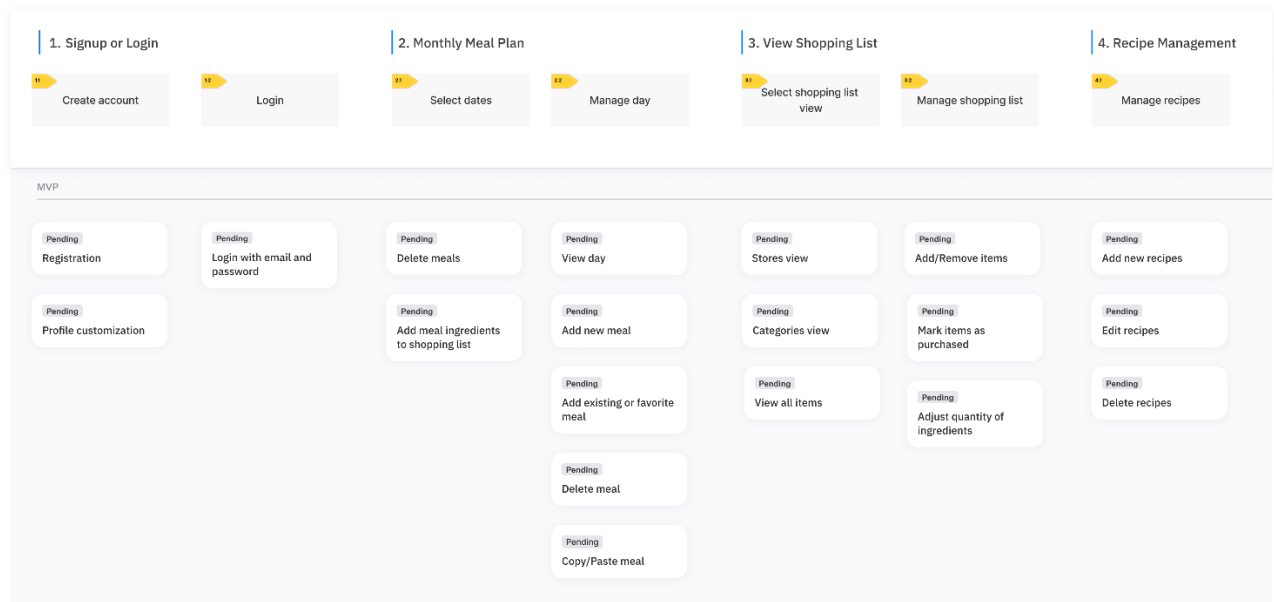
Description: Plateful is a personalized meal planning app that enables users to organize their monthly meals, store their favourite recipes, and manage shopping lists based on their meal plans. The app offers a fully customizable experience, where every recipe and meal plan are unique to each user, ensuring complete control over their meal choices and preferences.

Functional Requirements

Plateful will have the following core functionalities:

- Meal Planning:
 - Users can assign meals to specific days and generate a monthly overview
 - Users can copy or repeat meals across different days
- Recipe Management
 - Users can add, edit, delete, and store their recipes
 - Users can categorize recipes by type (e.g., breakfast, lunch, dinner)
 - Users can favourite recipes for quick access
- Shopping List
 - Users can manually select specific days from their meal plan to generate a shopping list
 - Users can manually add or remove items from the shopping list
 - Users can view the shopping list and toggle between entire list view, category view, or store view
- User Accounts & Personalization
 - Each user will have their own personalized account with unique recipes and meal plans.

STORY MAPPING (USER)



(SELECTED) USER STORIES

Story 1 – Signup

As a new user, I **want to** register for a Plateful account and customize my profile, **so that I can** start using the app.

Wireframe

The wireframe shows a web browser window titled "Plateful". Inside the window is a "Create Account" form. The form contains the following elements:

- Two input fields for "First Name" and "Last Name" side-by-side.
- A single input field for "Username".
- A single input field for "Email".
- A single input field for "Password".
- A single input field for "Repeat Password".
- A black button labeled "Sign-up".
- Below the button, the text "Already have an account? Login".

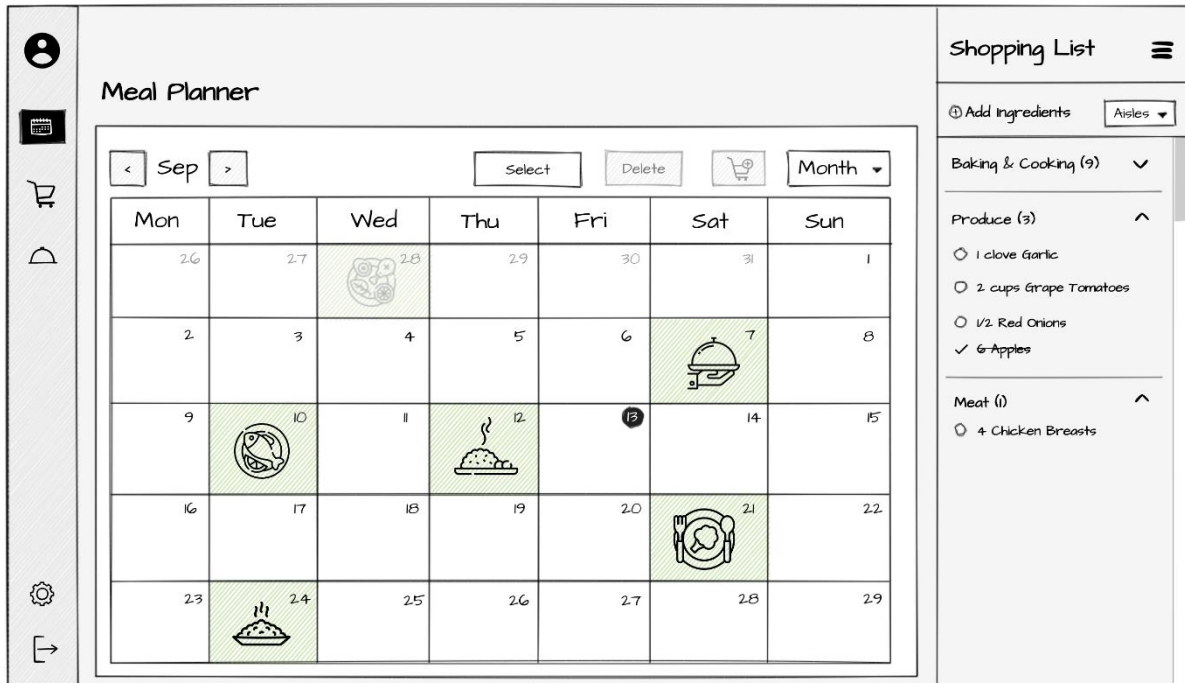
Acceptance Criteria:

- A user can create an account using a unique email address.
- The email must be validated to check if it is unique and properly formatted.
- If the email is already associated with an account, an error message is displayed: "Email already in use."
- The password must be at least 8 characters long, contain one uppercase letter, one number, and one special character.
- The password confirmation field must match the password entered. If the password does not match, an error is shown.
- All fields must be filled out to complete the registration. If any fields are left blank, an error is shown for each invalid field.
- Upon successful registration, the user is logged in and redirected to the home page (calendar).

Story 2 – Managing multiple dates

As a user, I want to select multiple days, **so that I can** add meals to my shopping list or delete multiple meals efficiently.

Wireframe:



Acceptance Criteria:

- The calendar displays all meals per day in an overview with their names or pictures. Users can toggle between weekly and monthly view.
 - If the calendar fails to load meals, an error message is displayed: *"Unable to load meals. Please try again later."*
- Users can select one or multiple days on the calendar to perform bulk actions (adding to the shopping list or deleting meals).
 - Users will only be able to select days that contain planned meals.
- When selecting multiple days, all meals' ingredients for the selected days are consolidated into a shopping list.
 - Users can review and modify the shopping list.
 - If there is an issue generating the shopping list, display error: *"Unable to generate shopping list. Please try again."*
- Users can delete multiple meals across different dates at once.
 - Deleting meals prompts a confirmation dialog before removal.
 - If deleting the meals fails due to a system issue, display: *"Unable to delete meals. Please try again."*

Story 3 – Managing meals for individual days

As a user, I want to manage meals for a selected date, **so that I can** view, add, edit, delete, or copy meals for a specific day efficiently.

Wireframe:

The wireframe illustrates a Meal Planner application. It features a sidebar on the left with icons for user profile, calendar, shopping cart, home, settings, and a plus sign. The main area is titled 'Meal Planner' and contains a calendar for the month of September. The calendar grid shows days from Monday to Sunday, with dates 1 through 29. Some days have meal icons: a bowl of soup on Wednesday (28), a bowl of food on Saturday (7), a bowl of food on Sunday (15), a bowl of food on Monday (21), and a bowl of food on Tuesday (24). Above the calendar grid are controls for selecting a date, deleting a meal, and a shopping cart icon. To the right of the calendar is a 'Shopping List' panel. It includes a section for 'Add Ingredients' with a dropdown menu set to 'Asides'. Below this, there are sections for 'Baking & Cooking (9)', 'Produce (3)', and 'Meat (1)'. The 'Produce' section lists items like '1 clove Garlic', '2 cups Grape Tomatoes', '1/2 Red Onions', and '6 Apples'. The 'Meat' section lists '4 Chicken Breasts'.

Acceptance Criteria:

- The user can click on a specific day to view meal details in a pop-up.
 - The pop-up displays the meal's picture, name, ingredients, cooking time, instructions, and category.
 - If the meal details fail to load in the pop-up, an error message is displayed: *"Unable to load meal details. Please try again."*
 - Users can edit or delete the meal directly from the pop-up.
 - If the deletion fails, display the error message: *"Unable to delete the meal. Please try again."*
- The user can add a new meal to a specific day by providing the required information in a form.
 - The system prevents creating duplicate meals for the same day.
 - If a meal with the same name already exists on the selected day, display: *"A meal with this name already exists for the selected date."*
 - Required fields like meal name, ingredients, and instructions must be filled before saving.
 - If required fields are empty, display an error message next to the empty fields.
- The user can add an existing or favourite meal to the selected day.
 - A pop-up displays previously added or favourite meals, allowing users to choose one.
 - If there is a system issue fetching the list of meals, display: *"Unable to load favourite or existing meals. Please try again."*
 - If the meal fails to be added, display: *"Unable to add the meal. Please try again."*
- The user can copy and paste meals from one day to another within the calendar.
 - Users can copy a meal from past or current dates and paste it to a selected date.
 - If the meal already exists for the target day, display: *"A meal with this name already exists on the selected date."*

Story 4 - Shopping List

As a user, I want to see the generated shopping list from my selected days, **so that I can** shop only for the ingredients I need during a selected period.

Wireframe:

The wireframe shows a mobile application interface for a shopping list. On the left is a vertical sidebar with icons for a user profile, calendar, shopping cart, home, settings, and a back arrow. The main content area is titled 'Shopping List' and includes a 'Stores' dropdown menu and a '+ Add Items' button. Below the title, there are three columns representing different stores: 'Albert Heijn' (blue background), 'Colruyt' (orange background), and 'No Store Selected' (light grey background). Each column contains a list of items with checkboxes. In the 'Albert Heijn' column, some items are checked and greyed out. The 'Colruyt' column has a scroll bar on its right side.

Albert Heijn	Colruyt	No Store Selected
<input type="checkbox"/> 1 clove Garlic	<input type="checkbox"/> 5' Bread	<input type="checkbox"/> Soy Sauce
<input type="checkbox"/> 10 Tomatoes	<input type="checkbox"/> Salt	<input type="checkbox"/> Olive Oil
<input type="checkbox"/> 5 Paprikas	<input type="checkbox"/> Pepper	<input type="checkbox"/> Strawberries
<input type="checkbox"/> 3 Pizzas	<input type="checkbox"/> Bottled Water	
<input checked="" type="checkbox"/> 4 Beef Steaks	<input type="checkbox"/> Cheese	
<input checked="" type="checkbox"/> Rice	<input type="checkbox"/> Butter	
<input checked="" type="checkbox"/> Greek Yogurt	<input type="checkbox"/> Ice cream	
<input checked="" type="checkbox"/> Orange Juice	<input type="checkbox"/> Chicken Breasts	
	<input type="checkbox"/> Onions	

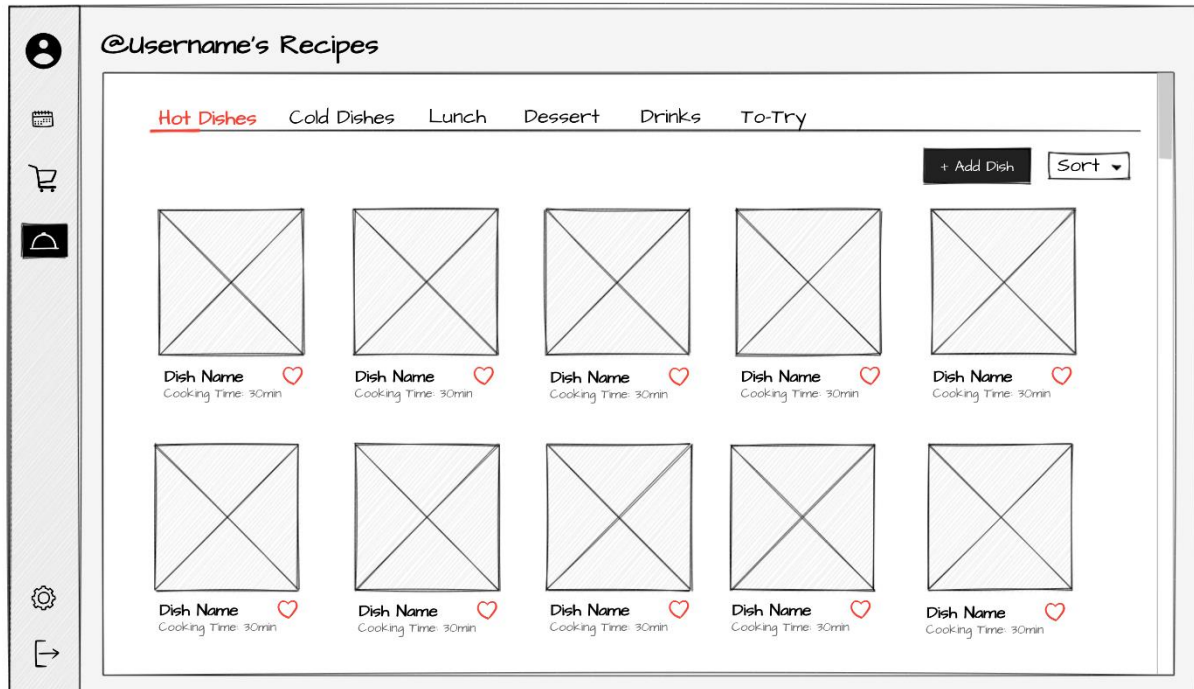
Acceptance Criteria:

- The shopping list includes the ingredients from the recipes assigned to the selected days.
- The user can toggle between different list views: stores, category, view all (ordered alphabetically).
- Users can check off items from the list while shopping. Checked-off items are visually crossed out and greyed out.
- The shopping list is automatically saved to the database after any changes.
- If there's a connection issue or failure to save changes, display: *"Unable to save changes. Please try again."*
- Users can add or remove items manually from the shopping list.
- Users can increase or decrease the quantity of items.
- Deleting an item prompts a confirmation dialog: *"Are you sure you want to remove this item from your shopping list?"*
- If the shopping list is empty, display: *"Your shopping list is empty."*

Story 5 - Recipe Management

- **As a user, I want to** add, edit, delete, and store my recipes, **so that I can** have easy access to the recipes I like to cook frequently.

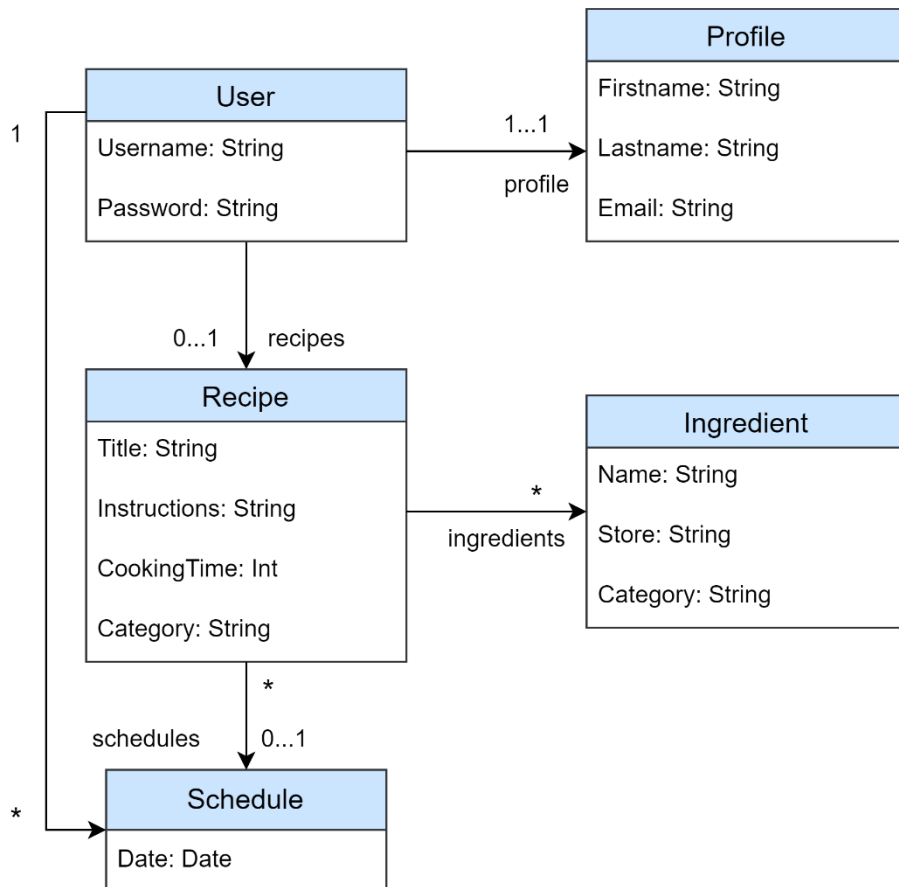
Wireframe:



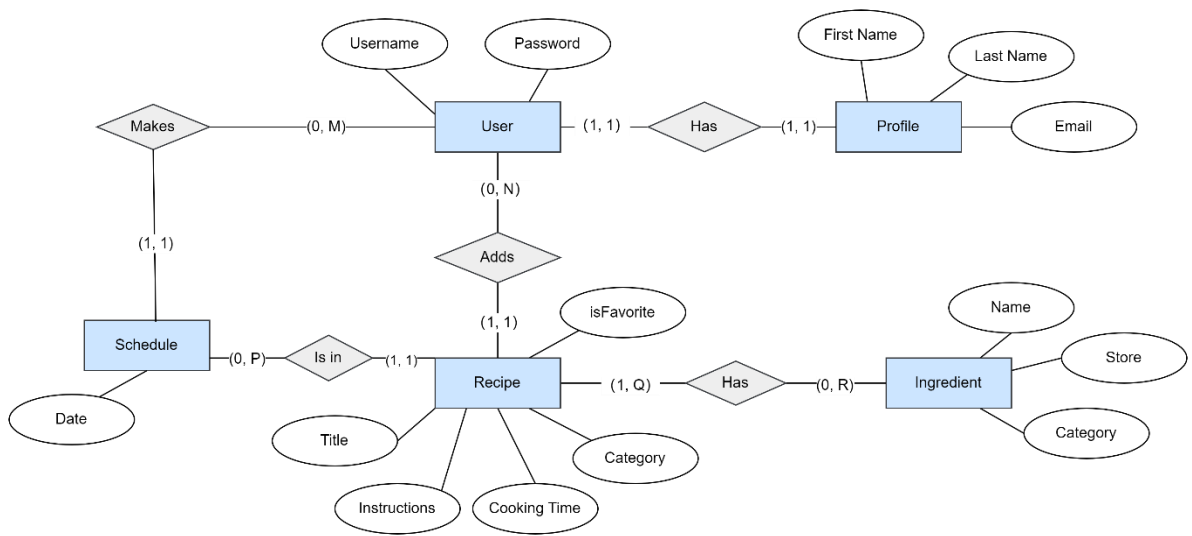
Acceptance Criteria:

- Users can create a recipe with fields for title, ingredients, instructions, cooking time, categories and isFavourite.
 - The form validates that the title, ingredients, and instructions fields are filled in.
 - If any required fields are empty, an error message should display for each invalid field.
- Users can edit and update existing recipes.
 - If the user attempts to leave the editing page without saving changes, display: *"You have unsaved changes. Are you sure you want to leave?"*
- Users can delete recipes, and the recipe is removed from the database.
 - Deleting a recipe prompts a confirmation dialog: *"Are you sure you want to delete this recipe?"*
- Users can mark recipes as favourites.
- If the user tries to save a recipe with duplicate titles, display: *"A recipe with this title already exists"*.

DOMAIN MODEL



CONCEPTUAL MODEL



LOGICAL MODEL

