Recipe Sharing Platform:

Front-End: The User Interface

The front-end is the visual and interactive part of your website, built using HTML for structure, CSS for styling, and JavaScript for functionality.

- **Recipe Card Layout**: Recipes will be displayed in a responsive card layout. Each "card" is a block containing a recipe's image, title, and a short description. "Responsive" means the layout automatically adjusts to look good on different screen sizes, like desktops and mobile phones, which can be achieved using CSS tools like Flexbox or Grid.

- **Search and Filters**: To help users find recipes, you'll add search and filter functions. This includes a search bar to look for recipes by name and category filters (e.g., "Dessert," "Dinner") that users can click to narrow down the displayed recipes. JavaScript is used to dynamically show or hide recipe cards based on the user's selections.

- **Recipe Submission Form**: This is an HTML form where users can enter details for a new recipe, such as the title, ingredient list, and instructions. The form needs validation, which is a check to ensure all required fields are filled out correctly before the recipe can be submitted.

Back-End: The Engine Room

The back-end works behind the scenes to power the website's features. For this project, you'll use PHP as the programming language and MySQL as the database.

- **User Authentication**: This is the system that allows users to register for a new account and log in. When a user signs up, their information (like a username and a secured password) is stored in the database. The login page checks the entered details against the stored information to grant access.

- **Database Management with MySQL**: MySQL is the database that will store all the website's information. You will need to create separate tables for users, recipes, and comments to keep the data organized. For example, the users table holds account information, while the recipes table stores all the recipe details.

- **Recipe Management with PHP (CRUD)**: PHP will be used to manage the recipes in the database through CRUD operations: Create, Read, Update, and Delete.

  - **Create**: When a user submits a new recipe, a PHP script saves that information into the recipes table.

  - **Read**: When a user wants to view recipes, PHP retrieves the data from the database and sends it to the front-end to be displayed.

  - **Update & Delete**: You will write PHP scripts to allow users to edit or remove the recipes they have posted.

Member F1 – UI and pages

Member F1 focuses on all the main pages and how recipes are visually shown. This includes:

- Build base layout: header, footer, navigation links (Home, Browse Recipes, Add Recipe, Login/Register).

- Create the recipe listing page with a responsive card grid (cards show image, title, short description, and category).

- Create the single recipe detail page (title, ingredients, steps, author, comments section placeholder).

- Apply consistent styling (colors, typography, spacing) with CSS, including mobile and tablet breakpoints using Flexbox or Grid.

Member F2 – Forms, search, and UX

Member F2 handles all interactive front-end features and validation. Tasks:

- Build the recipe submission form (title, category dropdown, ingredients textarea, steps textarea, optional image URL).

- Add client-side validation (required fields, max lengths, basic format checks) and user-friendly error messages.

- Implement search bar and category filter UI that can later call back-end endpoints (e.g., via form submits or simple JavaScript).

- Ensure all pages are fully responsive and accessible (labels on inputs, readable contrast, proper button states).

Member B1 – Setup, DB, and auth

Member B1 sets up the PHP–MySQL environment and all user-related logic. Tasks:

- Set up project structure (public index.php, includes for header/footer, config file for DB connection) and connect to MySQL.

- Design and create MySQL tables: users (id, name, email, password_hash, created_at), recipes, and comments base structure.

- Implement user registration (form handling in PHP, password hashing, inserting into users table).

- Implement login and logout using sessions (check credentials against DB, set session variables, secure logout).

Member B2 – Recipe & comment logic

Member B2 focuses on recipe and comment operations plus search/filter logic. Tasks:

- Implement PHP CRUD for recipes:

    - Create: handle recipe submission, validate server-side, insert into recipes.

    - Read: list recipes for home/browse page and show recipe details by id.

    - Update/Delete: allow the recipe owner to edit or remove their recipes.

- Implement comments: insert comments linked to a recipe and user, and fetch them for display on the recipe detail page.

- Implement server-side search and category filtering (SQL queries with WHERE title LIKE ? and WHERE category = ?) and wire them to the front-end forms.

- Add basic security and validation (escape output to prevent XSS, use prepared statements to prevent SQL injection).

AI Model Extra:
Phase 1: Setup & Data (Days 1-3)

Member B2 (AI Focus)

1. **Dataset Acquisition:** Go to Kaggle and download a large recipe dataset (e.g., RecipeNLG or Food.com).

2. **Data Cleaning (In Colab):** Write a script to load the data and keep only two columns: Ingredients (Input) and Instructions (Output). Remove recipes that are too short or have missing data.

3. **Preprocessing:** Convert the text data into a numerical format the AI can understand (Tokenization). Save this processed data for training.

Members F1, F2, B1 (Web Focus)

1. **F1:** Create the wireframes for the Home, Recipe Listing, and the new "AI Chef" page.

2. **B1:** Set up the local server (XAMPP/WAMP) and create the MySQL database with tables for Users and Recipes.

3. **F2:** Build the HTML structure for the standard "Submit Recipe" form to ensure you know what fields are needed.

Phase 2: Development & Training (Days 4-7)

Member B2 (AI Focus)

1. **Model Design:** In Google Colab, design a Neural Network (Sequence-to-Sequence or LSTM). It needs an "Encoder" to read ingredients and a "Decoder" to write the recipe.

2. **Training Session:** Run the training process. This is where the computer learns. It will take time (hours). Monitor the "Loss" (error rate) to ensure it goes down.

3. **Testing:** Manually test the model inside Colab by typing a list of ingredients and checking if the output makes sense.

Members F1, F2, B1 (Web Focus)

1. **F1:** Code the CSS/Bootstrap to make the site responsive. Design the specific "AI Result Card" to look different from normal recipes.

2. **F2:** Build the **"AI Chef" Interface**. It needs a text input for ingredients and a "Generate" button. Add a "Loading..." animation because the AI will take a few seconds to reply.

3. **B1:** Build the User Registration and Login systems. Ensure users can save recipes to the database.

Phase 3: The Connection (Days 8-9)

Member B2 (AI Focus)

1. **Build the API:** Inside the Colab notebook, create a simple web server (using Flask) that listens for incoming messages.

2. **Create the Tunnel:** Install a tunneling tool (Ngrok) in the notebook. This generates a public web link (e.g., http://random-name.ngrok.io) that connects directly to the running Colab notebook.

3. **Define Output:** Ensure the API returns the data in a clean JSON format (e.g., Title, Instructions) so the website can read it easily.

Members F2 & B1 (Web Focus)

1. **B1 (The Bridge):** Write a script on the website backend that accepts the user's ingredients, sends them to B2's "Ngrok Link," and waits for the answer.

2. **F2 (The Display):** Connect the "Generate" button to B1's script. When the answer comes back, format it nicely into the "AI Result Card" F1 designed.

Files Structure:

RECIPE SHARING PLATFORM - COMPLETE FILE STRUCTURE

Organized by: HTML → CSS → JavaScript → PHP → MySQL

RECIPE SHARING PLATFORM/

│

├── templates/

│   ├── pages/

│   │   ├── home.html   [MEMBER F1: UI and Pages]

│   │   ├── login.html   [MEMBER F1: UI and Pages]

│   │   ├── register.html   [MEMBER F1: UI and Pages]

│   │   ├── add-recipe.html   [MEMBER F1: UI and Pages]

│   │   ├── edit-recipe.html   [MEMBER F1: UI and Pages]

```
│  │      ├── recipe-detail.html  [MEMBER F1: UI and Pages]

│  │      ├── profile.html      [MEMBER F1: UI and Pages]

│  │      ├── my-recipes.html    [MEMBER F1: UI and Pages]

│  │      ├── favorites.html     [MEMBER F1: UI and Pages]

│  │      ├── search-results.html  [MEMBER F1: UI and Pages]

│  │      ├── category.html      [MEMBER F1: UI and Pages]

│  │      ├── about.html        [MEMBER F1: UI and Pages]

│  │      ├── contact.html      [MEMBER F1: UI and Pages]

│  │      ├── 404.html         [MEMBER F1: UI and Pages]

│  │      └── 500.html        [MEMBER F1: UI and Pages]

│  └── components/

│      ├── recipe-card.html    [MEMBER F1: UI and Pages]

│      ├── comment-item.html   [MEMBER F1: UI and Pages]

│      ├── comment-form.html   [MEMBER F1: UI and Pages]

│      ├── filter-sidebar.html  [MEMBER F1: UI and Pages]

│      ├── search-bar.html     [MEMBER F1: UI and Pages]

│      ├── pagination.html     [MEMBER F1: UI and Pages]

│      ├── user-card.html      [MEMBER F1: UI and Pages]

│      ├── recipe-meta.html    [MEMBER F1: UI and Pages]

│      ├── breadcrumb.html     [MEMBER F1: UI and Pages]

│      ├── alert.html        [MEMBER F1: UI and Pages]

│      └── modal.html        [MEMBER F1: UI and Pages]

│

├── assets/

│   ├── css/
```

```
│ │   ├── style.css          [MEMBER F1: UI and Pages]

│ │   ├── variables.css      [MEMBER F1: UI and Pages]

│ │   ├── reset.css          [MEMBER F1: UI and Pages]

│ │   ├── typography.css     [MEMBER F1: UI and Pages]

│ │   ├── layout.css         [MEMBER F1: UI and Pages]

│ │   ├── responsive.css     [MEMBER F1: UI and Pages]

│ │   ├── components/

│ │   │   ├── nav.css        [MEMBER F1: UI and Pages]

│ │   │   ├── footer.css     [MEMBER F1: UI and Pages]

│ │   │   ├── cards.css      [MEMBER F1: UI and Pages]

│ │   │   ├── buttons.css    [MEMBER F1: UI and Pages]

│ │   │   ├── forms.css      [MEMBER F1: UI and Pages]

│ │   │   ├── modals.css     [MEMBER F1: UI and Pages]

│ │   │   ├── alerts.css     [MEMBER F1: UI and Pages]

│ │   │   ├── pagination.css [MEMBER F1: UI and Pages]

│ │   │   ├── comments.css   [MEMBER F1: UI and Pages]

│ │   │   ├── sidebar.css    [MEMBER F1: UI and Pages]

│ │   │   └── breadcrumb.css [MEMBER F1: UI and Pages]

│ │   └── pages/

│ │       ├── home.css       [MEMBER F1: UI and Pages]

│ │       ├── login.css      [MEMBER F1: UI and Pages]

│ │       ├── register.css   [MEMBER F1: UI and Pages]

│ │       ├── recipe-detail.css [MEMBER F1: UI and Pages]

│ │       ├── profile.css    [MEMBER F1: UI and Pages]

│ │       ├── add-recipe.css  [MEMBER F1: UI and Pages]
```

```
|  |     └── search.css    [MEMBER F1: UI and Pages]
|  └── js/
|     ├── main.js        [MEMBER F2: Forms & UX]
|     ├── config.js       [MEMBER F2: Forms & UX]
|     ├── utils.js        [MEMBER F2: Forms & UX]
|     ├── api.js        [MEMBER F2: Forms & UX]
|     ├── components/
|     |   ├── validation.js  [MEMBER F2: Forms & UX]
|     |   ├── search.js     [MEMBER F2: Forms & UX]
|     |   ├── filter.js     [MEMBER F2: Forms & UX]
|     |   ├── comments.js    [MEMBER F2: Forms & UX]
|     |   ├── favorites.js   [MEMBER F2: Forms & UX]
|     |   ├── modal.js      [MEMBER F2: Forms & UX]
|     |   ├── image-preview.js [MEMBER F2: Forms & UX]
|     |   ├── pagination.js   [MEMBER F2: Forms & UX]
|     |   ├── toast.js      [MEMBER F2: Forms & UX]
|     |   └── dropdown.js    [MEMBER F2: Forms & UX]
|     └── pages/
|        ├── home.js      [MEMBER F2: Forms & UX]
|        ├── recipe-detail.js [MEMBER F2: Forms & UX]
|        ├── add-recipe.js   [MEMBER F2: Forms & UX]
|        ├── profile.js    [MEMBER F2: Forms & UX]
|        └── search.js     [MEMBER F2: Forms & UX]
|
├── [ROOT PHP FILES]
```

```
│   ├── index.php          [MEMBER B1: Setup & Auth]

│   ├── login.php          [MEMBER B1: Setup & Auth]

│   ├── register.php          [MEMBER B1: Setup & Auth]

│   ├── logout.php          [MEMBER B1: Setup & Auth]

│   ├── add-recipe.php        [MEMBER B2: Recipe Logic]

│   ├── edit-recipe.php        [MEMBER B2: Recipe Logic]

│   ├── delete-recipe.php      [MEMBER B2: Recipe Logic]

│   ├── recipe-detail.php      [MEMBER B2: Recipe Logic]

│   ├── profile.php          [MEMBER B1: Setup & Auth]

│   ├── my-recipes.php        [MEMBER B2: Recipe Logic]

│   ├── favorites.php        [MEMBER B2: Recipe Logic]

│   ├── search.php          [MEMBER B2: Recipe Logic]

│   ├── category.php          [MEMBER B2: Recipe Logic]

│   ├── about.php          [MEMBER B1: Setup & Auth]

│   ├── contact.php          [MEMBER B1: Setup & Auth]

│   ├── 404.php          [MEMBER B1: Setup & Auth]

│   └── 500.php          [MEMBER B1: Setup & Auth]

│

├── config/

│   ├── database.php          [MEMBER B1: Setup & Auth]

│   ├── config.php          [MEMBER B1: Setup & Auth]

│   └── session.php          [MEMBER B1: Setup & Auth]

│

├── includes/

│   ├── header.php          [MEMBER B1: Setup & Auth]
```

```
│   ├── footer.php          [MEMBER B1: Setup & Auth]

│   ├── nav.php             [MEMBER B1: Setup & Auth]

│   └── functions.php        [MEMBER B1: Setup & Auth]

│

├── controllers/

│   ├── AuthController.php    [MEMBER B1: Setup & Auth]

│   ├── UserController.php    [MEMBER B1: Setup & Auth]

│   ├── RecipeController.php   [MEMBER B2: Recipe Logic]

│   ├── CommentController.php  [MEMBER B2: Recipe Logic]

│   ├── SearchController.php   [MEMBER B2: Recipe Logic]

│   ├── CategoryController.php  [MEMBER B2: Recipe Logic]

│   └── FavoriteController.php  [MEMBER B2: Recipe Logic]

│

├── models/

│   ├── Database.php         [MEMBER B1: Setup & Auth]

│   ├── User.php            [MEMBER B1: Setup & Auth]

│   ├── Recipe.php           [MEMBER B2: Recipe Logic]

│   ├── Comment.php          [MEMBER B2: Recipe Logic]

│   ├── Category.php          [MEMBER B2: Recipe Logic]

│   └── Favorite.php         [MEMBER B2: Recipe Logic]

│

├── middleware/

│   ├── auth.php           [MEMBER B1: Setup & Auth]

│   ├── guest.php           [MEMBER B1: Setup & Auth]

│   └── csrf.php            [MEMBER B1: Setup & Auth]
```

```
|
├── validators/
│   ├── UserValidator.php      [MEMBER B1: Setup & Auth]
│   ├── RecipeValidator.php     [MEMBER B2: Recipe Logic]
│   └── CommentValidator.php    [MEMBER B2: Recipe Logic]
|
├── api/
│   ├── recipes/
│   │   ├── get-recipes.php     [MEMBER B2: Recipe Logic]
│   │   ├── get-recipe.php      [MEMBER B2: Recipe Logic]
│   │   ├── create-recipe.php   [MEMBER B2: Recipe Logic]
│   │   ├── update-recipe.php   [MEMBER B2: Recipe Logic]
│   │   ├── delete-recipe.php   [MEMBER B2: Recipe Logic]
│   │   └── filter-recipes.php  [MEMBER B2: Recipe Logic]
│   ├── comments/
│   │   ├── get-comments.php    [MEMBER B2: Recipe Logic]
│   │   ├── add-comment.php     [MEMBER B2: Recipe Logic]
│   │   ├── update-comment.php  [MEMBER B2: Recipe Logic]
│   │   └── delete-comment.php  [MEMBER B2: Recipe Logic]
│   ├── users/
│   │   ├── get-user.php        [MEMBER B1: Setup & Auth]
│   │   ├── update-profile.php  [MEMBER B1: Setup & Auth]
│   │   └── change-password.php [MEMBER B1: Setup & Auth]
│   ├── favorites/
│   │   ├── add-favorite.php    [MEMBER B2: Recipe Logic]
```

```
│  │    ├── remove-favorite.php  [MEMBER B2: Recipe Logic]
│  │    └── get-favorites.php   [MEMBER B2: Recipe Logic]
│  ├── search.php          [MEMBER B2: Recipe Logic]
│  ├── categories.php       [MEMBER B2: Recipe Logic]
│  └── upload-image.php      [MEMBER B2: Recipe Logic]
│
├── database/
│  ├── schema.sql          [MEMBER B1: Setup & Auth]
│  ├── seed.sql            [MEMBER B1: Setup & Auth]
│  └── migrations/
│     ├── 001_create_database.sql      [MEMBER B1: Setup & Auth]
│     ├── 002_create_users_table.sql     [MEMBER B1: Setup & Auth]
│     ├── 003_create_recipes_table.sql    [MEMBER B1: Setup & Auth]
│     ├── 004_create_comments_table.sql   [MEMBER B1: Setup & Auth]
│     ├── 005_create_categories_table.sql [MEMBER B1: Setup & Auth]
│     ├── 006_create_favorites_table.sql  [MEMBER B1: Setup & Auth]
│     ├── 007_create_recipe_categories.sql [MEMBER B1: Setup & Auth]
│     └── 008_add_indexes.sql         [MEMBER B1: Setup & Auth]
│
└── [SUPPORTING FILES]
   ├── .htaccess          [MEMBER B1: Setup & Auth]
   ├── .env             [MEMBER B1: Setup & Auth]
   ├── .gitignore         [MEMBER B1: Setup & Auth]
   ├── composer.json       [MEMBER B1: Setup & Auth]
   ├── package.json        [MEMBER B1: Setup & Auth]
```

```
├── README.md           [MEMBER B1: Setup & Auth]
└── assets/images/
    ├── logo.png         [MEMBER F1: UI and Pages]
    ├── favicon.ico       [MEMBER F1: UI and Pages]
    ├── default-recipe.jpg  [MEMBER F1: UI and Pages]
    ├── default-avatar.png  [MEMBER F1: UI and Pages]
    └── uploads/          [MEMBER B2: Recipe Logic] (Content Dir)
```