17-21

Friday 8 march  
Friday 15 march

Thursday 21 march

Monday 25 march

Thursday 4 april

Monday 8 april

Monday 15 april (Final presentation)

Description  
Functional requirements and non functional requirements  
  
  
  
  
  
  
Recipe Sharing Website: Create a platform where users can share their favourite recipes, search for recipes by specific ingredients, and save those they want to try. You could implement advanced search functionality and recipe ratings.

1. **Planning and Design**: Before diving into coding, it's important to plan and design the structure of your web application. Define the main requirements such as user registration, recipe uploading, ingredient-based recipe search, saving favorite recipes, and adding ratings. You can use tools like use case diagrams and flowcharts to assist you in the design process.
2. **Setup Development Environment**: Make sure you have a proper development environment set up to start working on your project. This may involve installing a code editor such as Visual Studio Code or PyCharm and using a web development framework such as Django, Flask, or Ruby on Rails.
3. **Backend Development**: Start by developing the backend of your web application. This involves setting up a database (MySQL) to store recipes and user information, as well as writing APIs to handle operations like creating, reading, updating, and deleting recipes. If you're using Python, you might consider using Django or Flask to streamline backend development.
4. **Frontend Development**: Once the backend is complete, focus on developing the frontend of your web application. This includes designing and implementing the user interface to allow users to interact with the site. You can use HTML, CSS, and JavaScript to create web pages and utilize frontend frameworks like React, Angular, or Vue.js to simplify development.
5. **Implementing Core Features**: After completing the basic backend and frontend, start implementing the core features of your web application. This may include creating forms for adding new recipes, creating pages for viewing recipes, implementing search functionality for ingredients, and allowing users to save favorite recipes.
6. **Testing and Debugging**: Make sure to thoroughly test your web application to identify and fix any bugs or issues. You can use testing tools like pytest for the backend and Jest or Cypress for the frontend.
7. **Deployment**: Finally, when your web application is ready to be published, you can set up a production server and deploy your application. You can use hosting services like Heroku, AWS, or DigitalOcean to host your web application.

**Applicazione di pianificazione giornaliera**: Un'applicazione che consente agli utenti di pianificare le proprie attività giornaliere, impostare promemoria e tener traccia dei compiti completati. Puoi includere funzionalità come notifiche push e un'interfaccia semplice e intuitiva.

**Sito web per la condivisione di ricette**: Crea una piattaforma in cui gli utenti possano condividere le proprie ricette preferite, cercare ricette per ingredienti specifici e salvare quelle che desiderano provare. Potresti implementare funzionalità di ricerca avanzata e di valutazione delle ricette.

**Chatbot per la pianificazione del budget**: Sviluppa un chatbot che aiuti gli utenti a creare e gestire il proprio budget finanziario. Il chatbot potrebbe fornire consigli su come risparmiare denaro, tenere traccia delle spese e impostare obiettivi finanziari.

**Generatore di meme personalizzati**: Crea un'applicazione che consenta agli utenti di creare facilmente meme personalizzati utilizzando modelli predefiniti e caricando le proprie immagini. Potresti anche includere funzionalità per condividere i meme direttamente sui social media.

**Sistema di raccomandazione di libri/film/musica**: Realizza un'applicazione che suggerisca libri, film o musica agli utenti in base ai loro gusti personali e alle loro preferenze passate. Puoi utilizzare algoritmi di raccomandazione basati su contenuti simili o sui comportamenti degli utenti.

**Quiz a tema**: Crea un sito web o un'applicazione che offra quiz su argomenti popolari come cinema, musica, storia o cultura generale. Puoi includere una vasta gamma di domande e livelli di difficoltà per intrattenere gli utenti.

Daily Planning Application: An application that allows users to plan their daily activities, set reminders, and keep track of completed tasks. You can include features such as push notifications and a simple, intuitive interface.

Budget Planning Chatbot: Develop a chatbot that helps users create and manage their financial budget. The chatbot could provide advice on saving money, track expenses, and set financial goals.

Book/Film/Music Recommendation System: Build an application that suggests books, films, or music to users based on their personal tastes and past preferences. You can use recommendation algorithms based on similar content or user behavior.

Themed Quizzes: Create a website or application that offers quizzes on popular topics such as cinema, music, history, or general culture. You can include a wide range of questions and difficulty levels to entertain users.