Task – 1,2,3,4

**Summary**

By following these steps, you can create a structured project in Jira, organize your work into epics and user stories, and manage your progress through sprints. This approach ensures that you have a clear roadmap for your project and can effectively track and manage your tasks.

**1. Create a Project in Jira**

* **Step 1:** Log in to your Jira account.
* **Step 2:** Navigate to the top menu and click on "Projects" > "Create Project".
* **Step 3:** Choose the project template that best fits your needs (e.g., Scrum, Kanban, etc.).
* **Step 4:** Fill in the project details such as the project name, key, and any other required fields.
* **Step 5:** Click "Create" to finalize the project creation.

**Example:**

* **Project Name:** Client Management System
* **Project Key:** CMS

**2. Create Epics for Clients and Client Meetings**

* **Step 1:** Open your project and go to the "Backlog" view.
* **Step 2:** Click on "Create Epic" in the Epics panel (usually found on the left side).
* **Step 3:** Fill in the epic details such as the epic name and summary.

**Example Epics:**

* **Epic 1:** Client Onboarding
  + **Summary:** Manage the process of onboarding new clients.
* **Epic 2:** Client Meetings
  + **Summary:** Schedule and document meetings with clients.

**3. Create User Stories in the Epics**

* **Step 1:** In the "Backlog" view, select the epic you want to create user stories for.
* **Step 2:** Click on "Create Issue" and choose "Story".
* **Step 3:** Fill in the details of the user story, including the title, description, and acceptance criteria.
* **Step 4:** Assign the story to the relevant epic.

**Example User Stories for Epic 1: Client Onboarding:**

* **Story 1:** As a project manager, I want to create a new client profile so that I can manage client information.
  + **Description:** Create fields for client name, contact details, business details, etc.
  + **Acceptance Criteria:** New client profile can be saved and retrieved.
* **Story 2:** As a sales representative, I want to track the onboarding progress of a new client so that I can ensure all steps are completed.
  + **Description:** Include steps such as initial contact, document collection, first meeting, etc.
  + **Acceptance Criteria:** Onboarding progress is updated and visible in the client's profile.

**Example User Stories for Epic 2: Client Meetings:**

* **Story 1:** As a project manager, I want to schedule a client meeting so that I can discuss project requirements.
  + **Description:** Include date, time, agenda, and participants.
  + **Acceptance Criteria:** Meeting is scheduled and notifications are sent.
* **Story 2:** As a team member, I want to document the outcomes of a client meeting so that there is a record of decisions and action items.
  + **Description:** Create a meeting notes section in the client profile.
  + **Acceptance Criteria:** Meeting notes can be saved and retrieved.

**4. Create and Start a Sprint for the Project**

* **Step 1:** In the "Backlog" view, click on "Create Sprint" at the top of the backlog section.
* **Step 2:** Name the sprint and set the sprint goal.
* **Step 3:** Add user stories from the backlog to the sprint by dragging them into the sprint section.
* **Step 4:** Set the sprint duration (e.g., 2 weeks).
* **Step 5:** Click "Start Sprint" to begin the sprint.

**Example Sprint:**

* **Sprint Name:** Sprint 1
* **Sprint Goal:** Complete initial client onboarding and schedule first meetings.
* **Sprint Duration:** 2 weeks

Task – 5

Web Page Designs for Client Creation and Client Meetings

**1. Client Creation Page**

Description: The Client Creation page is designed to facilitate the entry of new client information into the system. The page features a clean and intuitive layout with clearly labelled input fields for essential client details.

**Components:**

Header: Displays the title "Create New Client" prominently at the top of the page.

**Form Fields:**

**Client Name:** A text input field for entering the name of the client.

**Contact Person:** A text input field for entering the name of the primary contact person for the client.

**Contact Email:** A text input field for entering the email address of the contact person.

**Contact Phone:** A text input field for entering the phone number of the contact person.

**Business Details:** A larger text area for entering additional business information about the client.

**Submit Button:** A button to submit the form and save the client information.

**Styling:** The form is centrally aligned with a responsive design to ensure usability on various devices. Each form field is well-spaced with consistent labelling and placeholder text to guide users through the input process.

**2. Client Meetings Page**

**Description:** The Client Meetings page is designed to schedule and document meetings with clients. The layout is structured to provide an overview of upcoming meetings and the ability to add new meetings easily.

**Components:**

**Header:** Displays the title "Client Meetings" at the top of the page.

Meeting Schedule:

**Date and Time:** Input fields for selecting the date and time of the meeting.

**Agenda:** A text area for entering the meeting agenda.

**Participants:** A multi-select dropdown for choosing participants from a list of contacts.

**Location:** A text input field for entering the meeting location (or a checkbox for virtual meetings with an input for the link).

**Submit Button:** A button to add the meeting to the schedule.

**Meeting List:**

Upcoming Meetings: A section that lists all scheduled upcoming meetings with details such as date, time, agenda, and participants.

**Past Meetings:** A section for reviewing past meetings with the option to view meeting notes and outcomes.

**Styling:** The page features a modern, professional design with sections clearly demarcated. Input fields and buttons are styled to be user-friendly and accessible, ensuring that users can easily navigate and manage their meetings. Responsive design principles are applied to ensure functionality across various devices and screen sizes.

Task - 6,9

**Summary**

By following these steps, you will have a functional client management web application built with Angular. The application will include components for creating clients and managing client meetings, with a clean and user-friendly interface. This setup ensures scalability and maintainability, allowing for future enhancements and feature additions.

**Developing a Client Management Web Application using Angular**

**1. Setting Up Your Angular Project**

**Description:** First, set up an Angular project using the Angular CLI. This will create the necessary project structure and files to start development.

**Steps:**

1. **Install Angular CLI:** If you haven't already, install the Angular CLI globally

npm install -g @angular/cli

1. **Create a New Angular Project:** Use the CLI to create a new project

ng new client-management, cd client-management

1. **Serve the Project:** Start the development server to ensure everything is set up correctly.

ng serve -o

**2. Creating Components for Client Management**

**2.1 Client Creation Component**

**Description:** This component will handle the creation of new clients. It will include a form for inputting client details.

**Steps:**

1. **Generate the Component**

ng generate component client-creation

1. **Add Form Fields:**
   1. Client Name
   2. Contact Person
   3. Contact Email
   4. Contact Phone
   5. Business Details
   6. Submit Button

**HTML Structure:** Create a form in the client-creation.component.html file with input fields for each of the above details and a submit button.

**Styling:** Style the form in the client-creation.component.css file to ensure it looks clean and professional.

**2.2 Client Meetings Component**

**Description:** This component will manage client meetings, allowing users to schedule new meetings and view existing ones.

**Steps:**

1. **Generate the Component:**

ng generate component client-meetings

1. **Add Meeting Scheduling Form:**
   1. Date and Time
   2. Agenda
   3. Participants
   4. Location (or Virtual Meeting Link)
   5. Submit Button
2. **Display Meeting List:**
   1. Upcoming Meetings
   2. Past Meetings

**HTML Structure:** Create the meeting scheduling form and meeting lists in the client-meetings.component.html file.

**Styling:** Style the meeting components in the client-meetings.component.css file.

### **3. Services for Client Management**

**Description:** Create Angular services to handle data operations such as creating clients and scheduling meetings. This ensures a separation of concerns and keeps the components clean.

#### 3.1 Client Service

**Steps:**

1. **Generate the Service:**

ng generate service client

1. **Methods:**
   1. createClient(clientData): Method to create a new client.
   2. getClients(): Method to fetch all clients.

**3.2 Meeting Service**

**Steps:**

1. **Generate the Service:**

ng generate service meeting

1. **Methods:**
   1. scheduleMeeting(meetingData): Method to schedule a new meeting.
   2. getMeetings(): Method to fetch all meetings.

### **4. Running and Testing the Application**

**Description:** Run the application and test each component to ensure everything works as expected.

**Steps:**

1. **Start the Development Server:**

Ng serve -o

1. **Navigate to the Application:** Open a web browser and go to <http://localhost:4200>.
2. Test Client Creation:

* Navigate to the client creation component and test creating new clients

1. Test Client Meetings

* Navigate to the client meetings component and test scheduling and viewing meetings.

Task – 7

**Database and Table Design for Client Management and Meetings**

**1. Creating the Database**

**Description:** A database named ClientMeetingDB is created to store all client-related data and meeting information. This centralized database will facilitate efficient data management and retrieval.

**2. Creating the Clients Table**

**Description:** The Clients table stores detailed information about each client. Key fields include:

* **ClientID:** A unique identifier for each client.
* **FirstName and LastName:** Personal names of the client.
* **Email and PhoneNumber:** Contact details.
* **Address Information:** Includes fields for address, city, state, zip code, and country.
* **CreatedAt:** A timestamp of when the client record was created.

This table ensures comprehensive capture of client details for effective management.

**3. Creating the Meetings Table**

**Description:** The Meetings table stores information about each meeting. Key fields include:

* **MeetingID:** A unique identifier for each meeting.
* **MeetingDate:** The date and time of the meeting.
* **Subject:** The topic or agenda of the meeting.
* **Location:** The place where the meeting will be held.
* **Notes:** Any additional information or notes about the meeting.
* **CreatedAt:** A timestamp of when the meeting record was created.

This table allows for detailed tracking and scheduling of client meetings.

**4. Creating the Clients\_Meetings Table**

**Description:** The Clients\_Meetings table is a junction table that links clients to their respective meetings. Key fields include:

* **ClientID:** The unique identifier for the client.
* **MeetingID:** The unique identifier for the meeting.

This table establishes a many-to-many relationship between clients and meetings, allowing for efficient association and retrieval of related records.

Task – 8

**Description:**

This feature file outlines scenarios for managing client meetings. It includes three key scenarios that cover different aspects of meeting management:

1. **Scheduling a Client Meeting:**
   * **Objective:** To schedule a new meeting with a client.
   * **Steps:**
     + Log in as a project manager.
     + Navigate to the "Client Meetings" section.
     + Schedule a new meeting by entering client details, selecting a date and time, and providing an agenda.
     + Save the meeting and ensure the client receives a notification email.
2. **Updating a Client Meeting:**
   * **Objective:** To update the details of an existing meeting.
   * **Steps:**
     + Log in as a project manager.
     + Access an existing meeting with the client.
     + Edit the meeting date and agenda.
     + Save the changes and verify that the client receives an updated notification email.
3. **Cancelling a Client Meeting:**
   * **Objective:** To cancel a scheduled meeting with a client.
   * **Steps:**
     + Log in as a project manager.
     + Select an existing meeting to cancel.
     + Confirm the cancellation and ensure the client receives a cancellation notification email.

Each scenario is written in Gherkin syntax, which provides a clear and structured way to describe the behavior of the application in various contexts. The scenarios help ensure that all aspects of client meeting management are covered and tested appropriately.

Task – 10

**Description:** To manage and collaborate on your code, you need to synchronize your local project with a remote repository on GitHub. This process involves setting up a GitHub repository and using Git commands to push your local code to this repository.