W3H Analysis - Hospital Management

|  |  |
| --- | --- |
| **What?** | **How?** |
| 1. **What are the modules required ?**    * Admin    * Doctors    * Patients    * Staffs 2. **What are the functionalities required for admin module ?** 3. Admin can login using email and password 4. Adding doctors, patients and staffs based on receiving requests. 5. Admin can perform all CRUD operations such as adding, updating and deleting users. 6. **What are the functionalities required for doctors module ?**    1. Doctor can login using email and password    2. Doctors can edit the available time based on their availability    3. Doctors can able to approve the appointments. 7. **What are the functionalities required for patients module ?**    1. Patients can login using email and password    2. Search the specialized doctors based on the category which they required    3. Patients can book the appointment based on the doctor’s availability. 8. **What are the functionalities required for staffs module ?** | **1. Admin**   1. **Login :**   Method 1: admin can login using username and password  **Method 2: admin can login using email ID and password**   1. **Add the doctors :**   Method 1: implement the add doctors using doctor name  Method 2: implement the add doctors using doctor specialization  **Method 3: implement the add doctors using Doctor ID**   1. **Add the patients :**   Method 1: implement the add patients using patient name  **Method 2: implement the add patients using patient ID**   1. **Add the staffs :**   Method 1: implement the add staffs using staff name  **Method 2: implement the add staffs using staff ID**   1. **Performing other CRUD such as (read, update and delete) :**   Method 1: implementing CRUD by manually entering the user id or name or other details (for doctors, patients and staffs)  **Method 2: implementing CRUD by listing all users' details based on the category and give individual delete button for each** |

|  |  |
| --- | --- |
| 1. Staffs can login using email and password 2. Staffs can receive the patients appointment requests and allocate the available doctors to them 3. Can contact the patients to confirm the appointment and get additional details(if required). 4. Staffs can modify the doctor’s appointment slots based on the requests 5. **What are the fields required to add users?**   **Doctors :**  ▫ Doctor ID  ▫ Doctor name  ▫ Email ID  ▫ Specialization  ▫ Preferred shift  ▫ Working hours  **Patients :**  ▫ Patient ID  ▫ Patient name  ▫ Email ID  ▫ Age  ▫ Reason (issue)  ▫ Mobile number  **Staffs :**  ▫ Name  ▫ Email ID  ▫ Appointment details  ▫ Allocated doctors  ▫ Pending appointments   1. **What are the fields required to register for a user ?**   **Doctor :** username, email and password **Patient :** username, email and password **Staff :** username, email and password | **2. Doctor**  **1) Login:**  Method 1: doctor can login using username and password  **Method 2: doctor can login using email ID and password**  **3. Patient**   1. **Login:**   Method 1: patient can login using username and password  **Method 2: patient can login using email ID, mobile number and password**   1. **Search doctors:**   Method 1: patient can search the doctor  by using doctor’s name  **Method 2: patient can search the doctor**  **by using doctor’s id**  Method 3: patient can search the doctor  by using doctor’s specialization  **4. Staff**  **1) Login:**  Method 1: staff can login using username and password  **Method 2: staff can login using email ID and password** |

|  |  |
| --- | --- |
| 1. **Admin:**   **Login :**  **Method 2: admin can login using email ID and password**   * + **Can get the originality of the user without any conflict.**   **Add the doctors:**  **Method 3: implement the add doctors using Doctor ID**   * + **By using ID, we can perform the task very quickly without any complexity**   **Add the patients:**  **Method 2: implement the add patients using patient ID**   * + **By using ID, we can perform the task very quickly without any complexity**   **Add the staffs:**  **Method 2: implement the add staffs using staff ID**   * + **By using ID, we can perform the task very quickly without any complexity**   **Performing other CRUD such as (read, update and delete):**  **Method 2: implementing CRUD by listing all users' details based on the category and giving an individual delete button for each.**   * + **By giving individual delete buttons, we can remove specific user details without conflicting others.**  1. **Doctor**   **Login:**  **Method 2: doctor can login using email ID and password**   * + **Can get the originality of the user without any conflict.**  1. **Patient** | **1) Admin:**   1. To verify the originality of the user,to login using email ID would be the great option instead of providing username. 2. To add the users by providing specific ID will be the better choice instead of providing other credentials and it will avoid the unnecessary stumble 3. Deleting the users by manually providing their details will not be a feasible choice. It may go wrong if you entered invalid details mistakenly.   **2) Doctors:**  1. To verify the originality of the user,to login using email ID would be the great option instead of providing username.  **3) Patients:**  1. To verify the originality of the user,to login using email ID would be the great option instead of providing username.  **4) Staffs:**  1. To verify the originality of the user,to login using email ID would be the great option instead of providing username. |

|  |  |
| --- | --- |
| **Login:**  **Method 2: patient can login using email ID, mobile number and password**   * **Can get the originality of the user without any conflict.**   **Search doctors:**  **Method 2: patient can search the doctor**  **by using doctor’s id**   * **Can retrieve the exact data quickly without difficulty.**  1. **Staff Login:**   **Method 2: staff can login using email ID and password**   * + **Can get the originality of the user without any conflict.** |  |
| **Why?** | **Why not?** |