Overview

This application is a comprehensive Hospital Management System designed to facilitate the secure registration, onboarding, and management of patients, doctors, receptionists, and hospital workers. It provides a centralized platform for maintaining accurate and up-to-date records, supporting user account creation, and ensuring authenticated access to system resources. The primary purpose of the application is to streamline hospital administrative and clinical operations by enabling efficient data management and secure user authentication within a healthcare environment.

Key Personas

- Admin
- Doctor
- Receptionist

Key Functionalities

Admin:

- Register new patients and store their records securely in the system
- Update patient details and ensure data integrity
- Delete patient records with confirmation and feedback
- Manage doctor registration including adding, editing, and deleting doctor records
- View, search, and filter lists of doctors, receptionists, and hospital workers
- Register and onboard new receptionists and hospital workers with validation and feedback
- Process and validate all registration submissions for doctors, receptionists, and workers
- Receive notifications for successful or failed registrations and take corrective actions
- Authenticate user credentials and provision system access for all user types
- Manage user account registrations and verify outcomes

Doctor:

- Search for and list patients to access clinical information
- Submit registration details to create a user account
- Log in to the system using username and password
- Receive feedback on registration and authentication outcomes

Receptionist:

- Initiate patient management actions: add, update, search/list, or delete patient records
- Add new patient details and register them in the system
- Edit and update patient information as needed
- View and fill out registration forms to create user accounts
- Receive notifications for successful or failed registration attempts
- Log in to the system and gain access upon successful authentication

Activity 1: Manage Patient Records

User Action 1: Initiate Patient Management Action

Description:
As a Receptionist, I want to initiate patient management actions so that I can efficiently manage patient records for hospital operations.

- The system allows the user to select a patient management action type: Add, Update, Search/List, or Delete, from the main interface.

- When the user chooses to add a new patient, the system displays a patient registration form with required fields and validates the input before submission.

- When the user chooses to search for or list patients, the system displays a paginated table of patient records with options to edit or delete each record.

- When the user chooses to update an existing patient, the system allows selection of a patient and displays a pre-filled update form for editing patient details.

- When the user chooses to delete a patient, the system allows selection of a patient and confirms the deletion action before removing the record from the system.

- The system provides feedback to the user after each action (add, update, delete) indicating success or failure.

User Action 2: Add New Patient

Description:
As a Receptionist, I want to add new patient details so that new patients can be registered and their information is available for clinical and administrative use.

- The receptionist can access the 'Add Patient' form from the hospital management system interface.

- The 'Add Patient' form requires entry of all mandatory patient fields including first name, last name, gender, phone, city, email, age, and address.

- The form validates that all required fields are completed before allowing submission.
- Upon successful submission, the new patient record is inserted into the database and is immediately available for search and listing.
- The system provides feedback to the receptionist confirming successful registration or indicating any errors in the data entry process.
- The newly added patient appears in the patient list and can be accessed for future update or deletion actions.

User Action 3: Register Patient in System

Description:
As an Admin, I want the system to store new patient records so that patient data is securely saved and accessible for future reference.

- The user can access the 'Add Patient' form from the hospital management system interface.
- The 'Add Patient' form requires entry of all mandatory patient fields including first name, last name, gender, phone, city, email, age, and address.
- The system validates that all required fields are filled and displays an error if any are missing or invalid.
- Upon successful submission, the backend inserts the new patient record into the database.
- The user receives confirmation that the patient has been registered successfully.
- If the registration fails due to incorrect data or system error, the user is notified with an appropriate error message.
- The new patient record is immediately available for search and listing in the patient management system.

User Action 4: Search or List Patients

Description:
As a Doctor, I want to search for or list patients so that I can quickly access patient information for clinical decision-making.

- The user can access a search or list interface for patients from the hospital management system.
- The system displays a paginated table of patient records with relevant details such as name, gender, age, contact information, and address.
- The user can enter search criteria (such as name, mobile number, or other patient attributes) to filter the patient list.

- The system retrieves and displays only those patient records that match the entered search criteria.
- The user can view all patients in the system when no search criteria are entered.
- The patient list includes options to edit or delete each patient record.
- The system provides feedback if no matching patient records are found for the entered search criteria.
- The patient list is updated in real time after any edit or delete action is performed.

User Action 5: Select Patient to Edit

Description:
As a Receptionist, I want to select a patient to edit so that I can update their information when changes are needed.

- The user can view a list of patients with an option to select a patient for editing.
- When a patient is selected, the system displays a form pre-filled with the patient's current details.
- The form includes all editable patient fields such as first name, last name, gender, phone, city, email, age, and address.
- The system retrieves and displays the correct patient information based on the selected patient identifier.
- If the patient record cannot be found, the system displays an appropriate error message to the user.
- The user can proceed to update the patient information only after selecting a patient from the list.

User Action 6: Update Patient Details

Description:
As a Receptionist, I want to update patient details so that patient records remain accurate and up-to-date.

- The user can select a patient from the list to edit, and the update form is pre-filled with the patient’s current details.
- The update form validates all required fields before allowing submission (e.g., first name, last name, gender, phone, city, email, age, address).
- The user can modify patient details and submit the updated information to the backend system.

- The backend system updates the patient record in the database with the new details provided.
- The user receives feedback indicating whether the update was successful or if there was an error.
- The updated patient information is immediately reflected in the patient list after a successful update.

User Action 7: Process Patient Update in System

Description:
As an Admin, I want the system to process and save updated patient information so that the database reflects the latest patient data.

- When the admin submits updated patient information, the backend receives all required fields and validates the input before processing.
- The system updates the patient record in the database with the new information provided by the admin.
- If the update is successful, the system provides a confirmation message to the admin indicating the patient record was updated.
- If the update fails due to invalid data or system error, the admin receives an error message and is prompted to try again.
- The updated patient information is immediately reflected in subsequent searches or listings of patient records.
- All changes to patient data are auditable and maintain data integrity throughout the update process.

User Action 8: Select Patient to Delete

Description:
As an Admin, I want to select a patient to delete so that I can remove outdated or incorrect patient records from the system.

- The admin can view a list of all patients in the system with an option to delete each patient record.
- When the admin selects a patient to delete, the system prompts for confirmation before proceeding.
- Upon confirmation, the selected patient record is removed from the database.
- After deletion, the admin is redirected to the admin home page and receives feedback that the patient record was successfully deleted.
- If the deletion fails, the admin receives an error message and the patient record remains in the system.

- Only authorized users (admin) can access the delete patient functionality.

User Action 9: Delete Patient from System

Description:
As an Admin, I want the system to delete patient records so that the database remains clean and only contains relevant patient information.

- When an admin selects a patient to delete, the system prompts for confirmation before proceeding with the deletion.
- Upon confirmation, the system removes the selected patient record from the database.
- After deletion, the system redirects the admin to the admin home page or patient list view.
- The deleted patient no longer appears in any patient search or list results.
- If the deletion fails, the system displays an error message to the admin.

Activity 2: Manage Doctor Registration and Records

User Action 1: Fill Doctor Registration Form

Description:
As an Admin, I want to fill out a doctor registration form with required details so that I can onboard new doctors efficiently and ensure all necessary information is collected.

- The doctor registration form must include fields for unique ID, first name, last name, gender, phone, city, email, age, address, and qualification.
- Client-side validation must ensure all required fields are filled before allowing form submission.
- Upon successful submission, the form data must be sent via HTTP POST to the backend for processing and storage.
- If the registration is successful, the admin must receive a success alert and be redirected to the admin home page.
- If the registration fails, the admin must receive an error alert and be redirected back to the add doctor form for correction.

User Action 2: Submit Doctor Registration

Description:
As an Admin, I want to submit the completed doctor registration form so that the doctor's information is processed and stored in the system.

- The admin can submit the doctor registration form only after all required fields are filled and client-side validation passes
- Upon form submission, the doctor’s details are sent via HTTP POST to the backend for processing and storage
- If the registration is successful, the admin receives a success alert and is redirected to the admin home page
- If the registration fails, the admin receives an error alert and is redirected back to the add doctor form for correction
- The backend processes the registration by inserting the doctor’s details into the database, including all required fields such as ID, name, gender, phone, city, email, age, address, and qualification

User Action 3: Process Doctor Registration

Description:
As an Admin, I want the system to process and validate doctor registration data so that only accurate and complete records are added to the database.

- The system must validate all required fields on the doctor registration form before allowing submission, including unique ID, first name, last name, gender, phone, city, email, age, address, and qualification.
- Upon form submission, the system must send the doctor registration data to the backend for processing via HTTP POST.
- The backend must attempt to insert the new doctor record into the database only if all validations pass.
- If the registration is successful, the system must display a success alert to the admin and redirect to the admin home page.
- If the registration fails, the system must display an error alert and redirect the admin back to the add doctor form for correction.
- The system must handle backend errors gracefully and log exceptions without exposing sensitive information to the admin.

User Action 4: Notify Registration Success

Description:
As an Admin, I want to receive a success notification and be redirected to the admin home page when a doctor is registered successfully so that I know the process was completed.

- When a doctor registration is successful, the admin receives a visible success notification alert.

- After a successful doctor registration, the admin is automatically redirected to the admin home page.
- The success notification is only shown if the doctor record is inserted into the database without errors.
- If the registration is not successful, no success notification is shown and the admin is not redirected to the admin home page.

User Action 5: Notify Registration Failure

Description:
As an Admin, I want to receive an error notification and be redirected back to the registration form if doctor registration fails so that I can correct any issues and try again.

- If doctor registration fails during submission, the admin receives an error alert notification indicating the failure
- After the error notification, the admin is automatically redirected back to the doctor registration form
- The registration form is reloaded with the ability for the admin to correct and resubmit the doctor details

User Action 6: View Doctor List

Description:
As an Admin, I want to view a searchable and filterable list of all registered doctors so that I can easily find and manage doctor records.

- The admin can access a page displaying a list of all registered doctors with their details including ID, first name, last name, gender, mobile, city, email, age, address, date, and qualification.
- The admin can use a search bar to filter the doctor list by first name or last name.
- The doctor list updates in real-time or upon search submission to reflect the search criteria entered by the admin.
- Each doctor record in the list provides options for editing or deleting the record.
- The admin can view the full list of doctors when no search criteria are entered.

User Action 7: Search and Manage Doctors

Description:
As an Admin, I want to search for doctors by name and access options to

edit or delete their records so that I can keep doctor information up-to-date and accurate.

- The admin can view a list of all registered doctors with their details displayed in a table format.
- The admin can use a search bar to filter the list of doctors by first or last name.
- The search results update dynamically to show only doctors matching the entered search criteria.
- Each doctor record in the list includes options to edit or delete the record.
- Selecting the edit option for a doctor navigates the admin to the appropriate page or form for editing the doctor's details.
- Selecting the delete option for a doctor prompts for confirmation before removing the record from the list.
- If no doctors match the search criteria, the list displays an appropriate message indicating no results found.

Activity 3: Manage Receptionist Onboarding and Records

User Action 1: Display Add Receptionist Form

Description:
As an Admin, I want to view a web form for entering new receptionist details so that I can onboard new receptionists efficiently and ensure all required information is collected.

- The Add Receptionist form is displayed to the Admin user and includes fields for first name, last name, and phone number.
- All required fields in the Add Receptionist form must be filled before the form can be submitted.
- Submitting the form sends the receptionist details to the backend via HTTP POST for processing.
- If the receptionist record is inserted successfully, the Admin is notified with a success message and redirected to the admin home page.
- If the receptionist record insertion fails due to incorrect data, the Admin is alerted and redirected back to the Add Receptionist form for correction.

User Action 2: Submit Receptionist Registration

Description:
As an Admin, I want to submit the receptionist registration form so that the new receptionist's details are sent to the backend for processing and storage.

- The receptionist registration form must display all required fields for first name, last name, and phone number before submission.
- Submitting the registration form sends receptionist details to the backend via HTTP POST for processing.
- The backend must validate the submitted receptionist details and record the current date and time of registration.
- If the receptionist record is inserted successfully into the database, the user is notified of success and redirected to the admin home page.
- If the receptionist record insertion fails due to incorrect data, the user is alerted and redirected back to the registration form for correction.

User Action 3: Process Receptionist Submission

Description:
As an Admin, I want the backend to validate and process receptionist registration submissions so that only valid data is inserted into the system and errors are handled appropriately.

- The backend must receive receptionist registration submissions via HTTP POST and extract all required fields (first name, last name, phone).
- The backend must validate that all required receptionist fields are present and not empty before attempting to insert into the database.
- The backend must record the current date and time for each receptionist registration and include it in the database record.
- If the receptionist record is inserted successfully, the system must notify the user of success and redirect to the admin home page.
- If the receptionist record insertion fails due to incorrect or missing data, the system must alert the user and redirect back to the add receptionist form for correction.
- No receptionist record should be inserted into the database if required fields are missing or invalid.

User Action 4: Handle Successful Receptionist Registration

Description:
As an Admin, I want to be notified when a receptionist is registered

successfully so that I know the onboarding process is complete and can proceed with other tasks.

- When a receptionist registration is successful, the admin receives a notification indicating the success of the operation.
- After successful registration, the system redirects the admin to the admin home page.
- The notification of successful registration is displayed immediately after the receptionist record is inserted into the database.
- No notification of success is shown if the receptionist registration fails.

User Action 5: Handle Failed Receptionist Registration

Description:
As an Admin, I want to be alerted if receptionist registration fails due to incorrect data so that I can correct the information and resubmit the form.

- When receptionist registration fails due to incorrect data, the system displays an alert message to the admin indicating the failure.
- After a failed registration attempt, the admin is redirected back to the add receptionist form to correct and resubmit the information.
- The system does not insert any receptionist record into the database if the submitted data is incorrect or incomplete.
- The admin can immediately attempt to correct the data and resubmit the form after being notified of the failure.

User Action 6: View and Manage Receptionist List

Description:
As an Admin, I want to view, search, edit, and delete receptionist records in a dynamic table so that I can efficiently manage receptionist information and keep records up to date.

- The receptionist list displays all registered receptionists in a dynamic table format.
- The receptionist list can be filtered by first name or last name using a search input.
- Each receptionist record in the table includes options to edit or delete the record.
- Editing a receptionist record allows the admin to update the receptionist’s details and save changes.
- Deleting a receptionist record removes the receptionist from the list and the database.

- The table updates in real time to reflect any changes made by editing or deleting records.
- If no receptionist matches the search query, the table displays an empty result or appropriate message.

Activity 4: Manage Hospital Worker Registration and Records

User Action 1: Navigate to Add Worker Page

Description:
As an Admin, I want to access the 'Add Worker' page and view a registration form so that I can begin registering a new hospital worker with all required details.

- The Admin can successfully navigate to the 'Add Worker' page from the main navigation menu.
- The 'Add Worker' page displays a registration form with required fields for first name, last name, and phone number.
- The registration form enforces client-side validation to ensure all required fields are filled before submission.
- If the Admin submits the form with missing required fields, an error message is displayed and the form is not submitted.
- Upon successful submission of the form, the data is sent to the backend for processing.

User Action 2: Submit Worker Registration Form

Description:
As an Admin, I want to submit the completed worker registration form so that the new worker's details are sent to the backend for processing and addition to the database.

- The worker registration form requires first name, last name, and phone number fields to be completed before submission.
- Client-side validation prevents submission if any required field is missing.
- Upon form submission, the entered worker details are sent to the backend for processing.
- If the backend successfully adds the new worker to the database, the admin receives a success notification and is redirected to the confirmation or worker list page.

- If the backend fails to add the worker due to a database error or invalid data, the admin receives an error notification and is prompted to correct the issue or retry.

User Action 3: Process Worker Registration

Description:
As an Admin, I want the system to process the submitted worker registration data so that the new worker record is correctly stored in the database.

- The system receives the submitted worker registration form and extracts the required fields: first name, last name, and phone number.
- The backend attempts to insert the new worker record into the database using the provided registration data.
- If the database operation is successful, the new worker record is stored and the admin is notified of successful registration.
- If the database operation fails due to an error or invalid data, the admin is notified of the failure and prompted to correct the issue or retry.
- The system ensures that only valid and complete worker registration data is processed and stored in the database.

User Action 4: Check Worker Registration Result

Description:
As an Admin, I want the system to check if the worker registration was successful or failed so that I receive appropriate feedback and can take further action if needed.

- The system determines if the worker registration was successful or failed after the admin submits the registration form.
- If the registration is successful, the admin receives a notification confirming the worker was added and is redirected to the appropriate page.
- If the registration fails due to a database error or invalid data, the admin receives an error notification and is prompted to correct the issue or retry.

User Action 5: Handle Worker Registration Success

Description:
As an Admin, I want to be notified when a worker is registered successfully so that I can confirm the addition and proceed with other administrative tasks.

- When a worker registration is successful, the admin receives a clear notification confirming the addition.

- The system redirects the admin to the worker list or confirmation page after successful registration.
- The notification of success is displayed immediately after the registration is processed and before any further actions can be taken.
- The newly registered worker appears in the worker list accessible to the admin after successful registration.
- No notification of failure is shown when the registration is successful.

User Action 6: Handle Worker Registration Failure

Description:
As an Admin, I want to be notified if worker registration fails so that I can correct any errors and retry the registration process.

- If worker registration fails due to a database error or invalid data, the admin is notified with a clear error message indicating the failure.
- The system prompts the admin to correct any errors in the registration form and allows them to retry the registration process.
- The registration failure notification is displayed immediately after the failed attempt, without requiring the admin to refresh or navigate away from the page.
- No worker record is added to the database if the registration fails.

User Action 7: Access Admin Worker List Page

Description:
As an Admin, I want to access the worker list page so that I can view all registered hospital workers and manage their records efficiently.

- The admin can successfully navigate to the Admin Worker List page from the application interface.
- The Admin Worker List page displays a table containing all registered hospital workers with their relevant details.
- The admin can enter a search query to filter workers by first or last name, and the results update accordingly in the table.
- The admin can view the full list of workers when no search criteria are entered.
- The worker list is fetched from the database and displayed without errors.
- The table of workers is styled and readable for management purposes.

User Action 8: Search or View Workers

Description:
As an Admin, I want to search for workers by name or view the full list so that I can quickly find and manage specific worker records.

- The admin can access the worker list page and view all registered workers displayed in a table format.
- The admin can enter a search query to filter workers by first or last name and see only matching results in the list.
- The search results update dynamically to reflect the entered search criteria without requiring a page reload.
- If no search criteria are entered, the full list of workers is displayed.
- The admin can view all relevant details for each worker in the list, including first name, last name, and phone number.
- The system displays an appropriate message if no workers match the search criteria.

Activity 5: Register Users and Onboard Accounts

User Action 1: Display Registration Form

Description:
As a Receptionist, I want to view and fill out a registration form so that I can create a new user account for myself or others in the Hospital Management System.

- The registration form is displayed to the user and includes fields for username, password, and password confirmation.
- The registration form requires all fields to be filled before allowing submission.
- The user can submit the registration form and the data is sent to the backend via HTTP POST.
- If the registration is successful, the user is notified and redirected to the login page.
- If the registration fails, the user is notified and redirected back to the registration page.

User Action 2: Submit Registration Form

Description:
As a Doctor, I want to submit my registration details so that my account can be created and I can access the system.

- The registration form must require the user to enter a username, password, and password confirmation before submission is allowed.
- The registration form must not be submitted unless all required fields are filled.
- Upon submission, the registration details must be sent to the backend via HTTP POST for processing.
- The backend must attempt to insert the submitted credentials into the database upon receiving the registration data.
- If the registration is successful, the user must be notified with a success message and redirected to the login page.
- If the registration fails, the user must be notified with a failure message and redirected back to the registration page.

User Action 3: Process Registration in Backend

Description:
As an Admin, I want the backend to securely process and store user registration data so that new accounts are properly created in the database.

- The backend receives registration data submitted from the registration form via HTTP POST.
- The backend validates that all required fields (username, password, password confirmation) are present in the registration data.
- The backend attempts to insert the new user credentials into the database.
- If the database insertion is successful, the backend triggers a success notification and redirects the user to the login page.
- If the database insertion fails, the backend triggers a failure notification and redirects the user back to the registration page.
- User credentials are stored securely in the database upon successful registration.

User Action 4: Check Registration Outcome

Description:
As an Admin, I want the system to verify if registration was successful or

failed so that users receive accurate feedback and the database remains consistent.

- The system must determine if the registration attempt was successful or failed based on the result of the database operation.
- If registration is successful, the user must be notified with a success message and redirected to the login page.
- If registration fails, the user must be notified with a failure message and redirected back to the registration page.
- The database must only contain new user credentials if the registration was successful.
- Users must receive feedback immediately after the registration attempt, indicating the outcome.

User Action 5: Notify Registration Success

Description:
As a Receptionist, I want to be notified when my registration is successful so that I can proceed to log in and use the system.

- The system displays a notification to the user when registration is successful.
- The user is redirected to the login page after successful registration.
- The notification clearly indicates that registration was completed successfully.
- The notification and redirection occur only if the user credentials are successfully inserted into the database.

User Action 6: Notify Registration Failure

Description:
As a Doctor, I want to be notified if my registration fails so that I can correct any errors and try again.

- The system displays a clear notification to the user when registration fails.
- The user is redirected back to the registration page after a failed registration attempt.
- The notification message indicates that registration was unsuccessful and prompts the user to try again.
- The registration failure notification is shown immediately after the backend determines that user credentials were not inserted into the database.

- The user can reattempt registration after being notified of the failure.

Activity 6: Authenticate Users and Provision System Access

User Action 1: Display Login Form

Description:
As a Receptionist, I want to see a login form requiring my username and password so that I can securely access the Hospital Management System.

- The login form is displayed to the user when accessing the Hospital Management System.
- The login form requires both a username and a password before allowing submission.
- The user can enter their username and password and submit the login form.
- If the submitted credentials are valid, the user is authenticated and granted access to the Hospital Management System home page.
- If the submitted credentials are invalid, an error message is displayed and the user is redirected back to the login page to retry.

User Action 2: Submit Login Credentials

Description:
As a Doctor, I want to submit my username and password so that I can authenticate and gain access to the system.

- The login form is displayed and requires both username and password before allowing submission.
- When the user submits the login form, the entered credentials are sent to the backend for authentication.
- If the submitted credentials match a stored user record, the user is authenticated and granted access to the Hospital Management System home page.
- If the submitted credentials do not match any stored user record, an error message is displayed and the user is redirected back to the login page to retry.

User Action 3: Authenticate User Credentials

Description:
As an Admin, I want the system to verify submitted credentials against

stored records so that only authorized users can access the Hospital Management System.

- The login form is displayed and requires both username and password before allowing submission.
- When a user submits the login form, the entered credentials are sent to the backend for authentication.
- The backend retrieves the user record from the database and compares the submitted credentials with the stored records.
- If the credentials are valid, the user is authenticated and granted access to the Hospital Management System home page.
- If the credentials are invalid, an error message is displayed and the user is redirected back to the login page to retry.
- Only authenticated users are able to access the Hospital Management System.

User Action 4: Grant System Access

Description:
As a Receptionist, I want to be granted access to the system upon successful authentication so that I can perform my duties.

- The login form is displayed and requires both username and password before allowing submission.
- When valid credentials are submitted, the user is authenticated and redirected to the Hospital Management System home page.
- If invalid credentials are submitted, an error message is displayed and the user is redirected back to the login page to retry.
- Only authenticated users are granted access to the system; unauthenticated users cannot proceed beyond the login page.

User Action 5: Handle Authentication Failure

Description:
As a Doctor, I want to receive an error message and be prompted to retry if my login fails so that I know my credentials were incorrect and can attempt again.

- When a user submits invalid login credentials, an error message is displayed indicating that the username or password is incorrect.
- After authentication failure, the user is redirected back to the login page to attempt login again.
- The login form is presented again after a failed authentication, allowing the user to re-enter credentials.

- No access to the Hospital Management System is granted unless valid credentials are provided.