Criterion E - Evaluation

Pharmacy Management Application

Table of contents

Evaluation of the Product based on Success Criteria

Criteria Serial	Success Criteria
1	Profiles for the Front Desk, Pharmacists, Doctors, and the Admin
✔	The profiles could be accessed from the home page of the application. Upon successful account creation and verification of the user by the administrator, the user could access their designated profile and its features.
2	Only the admin profile can verify and remove employees and edit doctor specializations.
✔	The admin profile included features to verify and remove employees and manage doctor specializations. Changes were also reflected in the database. Other user profiles lacked access to these functions, and any attempt to access these pages via manual URL modification triggered appropriate error messages.

3	Admin and pharmacy profiles can view and edit medicines and inventory, and lab tests.
✔	Admin and pharmacy users could successfully view, add, edit, and delete medicines, its inventory and lab tests available. Changes were also reflected in the database. Access from other profiles was prohibited, appropriate error messages were displayed otherwise.
4	The front desk profile can book appointments for new and existing patients.
✔	Front desk users could successfully book appointments with a doctor by entering details for new patients or by selecting the patient from a table for existing patients. Both actions had their specific dedicated pages and a successful request reflected appropriate changes in the database.
5	The doctor profile can write a prescription for booked appointments.
✔	When an appointment was booked from the front desk, it became visible on the respective doctor's home page. The doctor could then write a prescription for the

	appointment, with all changes successfully updated in the database upon request completion.
6	The doctor's profile allows access to the prescription at all times, and the pharmacy profile allows access for some time, abiding by medical rules and regulations.
✔	The doctor profile could access all prescriptions they had previously issued, both complete and incomplete. In contrast, the pharmacy profile could only view incomplete and unpaid prescriptions. Once a patient completes the payment at the front desk, the pharmacy loses access to the prescription and can only view lab tests where data collection has already begun.
7	The pharmacy profile can dispense medicines and enter the required data for lab tests.
✔	Pharmacy users could view medicines awaiting dispensing and lab test requests requiring data collection. They could navigate to the respective requests, dispense the necessary medicines, and update stock in the database upon a successful transaction. Similarly, for prescribed lab tests, users could input details such as

	sample codes and report codes while updating the status accordingly, with all changes accurately reflected in the database.
8	Status trackers for prescribed medicines, lab tests and prescriptions.
✔	Status tracker fields were appropriately defined in the Django models and updated as needed. View permissions dynamically adjusted based on status changes; for instance, once medicines were marked as dispensed, the pharmacy profile could no longer access the medicines section of that prescription.
9	All required fields must be filled out before submitting; to inform the user otherwise appropriately.
✔	On any page, users could not submit a request unless all required fields (marked with an asterisk) were completed. If any mandatory fields were left blank, a notification highlighted the missing fields. Once all required fields were filled, the request was successfully sent to the backend.
10	Sessions will automatically expire after a set period, requiring re-login.

✔	The access token had to be refreshed every 30 mins, as could be observed from the requests made to the backend. Similarly, the refresh token expired in a day, requiring re-login.
11	Structured way of generating tokens at the front desk, for an ordered way of writing prescriptions by the doctor.
✔	Each day, token generation started at 1, with each doctor's counter operating independently. The assigned token number for each appointment was visible to the respective doctor, enabling them to make informed decisions.
12	The system will calculate the total costs for an appointment based on consultations, lab tests, and medicines taken.
✔	The front desk user could see appointments pending payment for each doctor, and could see the amount to be paid. The breakdown was also provided, to verify the calculations.
13	The system will display clear error messages and success notifications for user actions to keep them informed.

✔	Whenever a user makes a request to the database, they receive real-time notifications indicating the response status, whether successful or failed.

Future Recommendation

1. Edit Profile feature: Once a profile is created, there is no way to edit the details of the user. This includes, but is not limited to, changing passwords, name, phone numbers, etc. Implementing this feature will require significant research from the client side to check what is appropriate to be editable and not.
2. Enhanced Prescription features: The layout for writing prescriptions in the doctor application is not intuitive. Furthermore, customised features for different doctor specialisations like animations for exercises, usually prescribed by orthopaedicians, can really help patients. While this requires even more research into the requirements for each specialization, this could be extremely helpful for the patients as well as the doctors to communicate with the patient.