Write a comprehensive lesson allocation PHP script using this format and flow. when the script is first clicked, it checks and displays all classes whose department\_id is the same as that of the logged-in user. (table classes (department\_id)) these classes are represented and displayed in blocks arranged in a column of 3 and are clickable. when a specific block of class is clicked, it opens a page with the name of the class at the top and all subjects allocated to it below which are clickable as well (subjects table and display all the subject\_name whose class\_id is the current or opened). when a specific subject is clicked, it opens and lists all lecturers qualified to teach that particular subject and whose department\_id is the same as that of the logged-in user (lecturer\_subjects table (subject\_id) and hod\_subjects table (subject\_id)). One lecturer is selected then another form opens to allow the selection of the start time, the end time, and the day of the week (Monday to Friday). All that data is saved in the table allocations (allocations table (id, lecturer\_id, subject\_id, class\_id, course, level, start\_time, end\_time, duration, day\_of\_week) the durations is the difference between the end time and start time. the script should not allow the same lecturer to be allocated more than one lesson in the same class. in scenarios where there are subjects not allocated to lecturers and there are no free lecturers who are in the same department as the logged-in user, it checks for any other lecturer who qualifies to teach that subject (lecturer\_subjects table (subject\_id)) and prompts the logged-in user to send a notification to the HOD of that lecturer (uses the department\_id of the chosen lecturer from the table lecturers and sends the notification to the user whose user\_role is HOD and the department\_id resembles that of the lecturer from the table users) to request approval to allocate the lecturer (the notification message should contain the subject name, start time and end time and the day), (notifications table (id,user\_role, message, status, notification\_time, lesson\_id, user\_id, is\_sent)) the user then waits for a response from the HOD of the lecturer, if approved, an email and notification are sent to the affected lecturer notifying him/her of the approved request, and once allocated, their HOD cannot allocate them lessons whose hours collide with that which the lecturer was allocated in the other department. if the request is rejected, a reason is given to the requesting user. upon successful allocation, the data is stored in the allocations table (allocations table (id, lecturer\_id, subject\_id, class\_id, course, level, start\_time, end\_time, duration, day\_of\_week), and an appropriate message alert is displayed. Employ appropriate mechanism to allow smooth and quick opening of those different forms