

PROJECT FUNCTIONALITY DOCUMENT

In the second phase of the project (design report), you need to provide UIs and SQL queries (User Interface Design & Corresponding SQL Statements part) for functionalities of your system. Below, you can find the ones that will be checked. Two of them are common among all project groups and three of them are topic specific. Notice that you may need to alter your E/R diagram to capture all these functionalities. **In this part of the design report, only provide UIs and SQL queries of common functionalities and first topic specific functionality each having its own subsection.**

In the implementation phase, you will be responsible for all functionalities.

COMMON FUNCTIONALITIES

1. Signup and Login functionalities for different user types, if there are no several user types available, admin should be considered. **(DESIGN REPORT)**
2. Additional functional requirement that you introduced into the project context. You should discuss this functionality with your project TA to make sure the scope of the functionality is sufficient. **(DESIGN REPORT)**

TOPIC SPECIFIC FUNCTIONALITIES

Online Course Platform

1. Buy a course (by a standard user) **(DESIGN REPORT)**
 - a. Listing all available courses and applying filters (price, discount, keyword etc.)
 - b. Selecting a course and see its features like price, description, average rating etc.
 - c. Buy/Wishlist the course (See bought courses in user's profile)
 - d. Watch a lecture of the course by selecting it on the course home page
 - i. When a lecture is finished, continue with the next one.
 - ii. Get certificate if all lectures are completed
 - iii. Create notes on lectures (Visible only to user)
 - e. Comment on and rate the course only if the course is completed by the user
 - i. Visible to other users
2. Publish a course (by a course creator)
 - a. Specify course title, description, price etc.
 - b. Create lectures for the course
 - i. Specify lecture title, video length etc.
 - c. Make announcements about the course
 - i. Select a course to make an announcement about
 - ii. Enter announcement text
 - iii. Users that bought the course must be notified of announcements
 - d. Course Q&A
 - i. Select a course and/or lecture to ask a question about (by a standard user)
 - ii. Course creator lists its courses and selects one

- iii. List questions about that question
 - iv. Select one question and answer it
- 3. Site management (by an admin)
 - a. Request refund on a course (by a standard user)
 - i. List all bought courses
 - ii. Select the course to return
 - iii. Specify the reason for the request and send
 - b. Check request and approve/reject (by an admin)
 - i. List available refund requests
 - ii. Select a required and make a decision
 - iii. User must be notified of outcome of request
 - c. Offer discount for a course
 - i. List all discount allowed courses
 - ii. Select a course and apply desired discount
 - iii. Cancel an already applied discount

Social Cataloging Platform for Books

- 1. Track a book progress (by a standard user) **(DESIGN REPORT)**
 - a. List all available books and apply filters (genre, author, keyword)
 - b. Select a book (and the edition) and start tracking
 - i. Indicate page number & date at each progress
 - ii. Show each progress step on book's profile (only visible to user)
 - c. Create lists and add books to these lists
 - i. Specify name of the list
 - ii. List books and select the books to be added
 - d. Add other users as friends
 - i. List all users and select one to add friend (other person should accept the request)
 - ii. Like/comment on their posts
 - iii. Recommend book to friends (if book is read by the user)
- 2. Publish Book (by an author)
 - a. Specify features like title, description etc.
 - b. Create several editions of the book
 - i. Select the book to create its editions
 - ii. Specify edition format, page count, language for each edition
 - iii. Enter translator info if necessary
 - c. Review a book (by a standard user)
 - i. Users selects a book and leave a review
 - ii. Author select his/her book and list all available reviews
 - iii. Author selects a review and replies to it.

3. Site Management (by a librarian)
 - a. Erroneous info correction request (by an author or by a standard user)
 - i. User/Author select a book and creates a request
 - ii. Librarian lists all available requests
 - iii. Select a request and approve/reject it
 - iv. If approve, edit the information about book accordingly
 - b. Reading challenge
 - i. Specify title, deadline, type, book count etc.
 - ii. List all available challenges and join one (by a standard user)
 - iii. See all participants and their progress of challenge
 - iv. Outcome of challenge
 1. If the user is successful, show it in the user's profile
 2. If the user fails, notify the user of failure.

Food Ordering and Delivery System

1. Customer makes an order from a Restaurant **(DESIGN REPORT)**
 - a. Search restaurants and meals in the system through an interface.
 - b. Order meal/s from a particular restaurant by specifying options such as adding/removing certain ingredients, delivery time, and etc.
 - c. List all the orders made by the customer on a history page. Not delivered orders should have an order cycle having different status values.
 - d. Check details of a particular order by listing meals purchased, restaurant served, delivery time, delivery person, and etc.
 - e. Write comments for orders that are finalized/delivered and see the response of the restaurant's owner if available.
2. The Restaurant Owner manages his/her menu and orders.
 - a. List all the orders made by customers considering different values for meal status.
 - b. Finalize an order and ask for a delivery guy assignment. A delivery guy is randomly assigned to an order if his/her status is available.
 - c. List all the comments made by customers and write a response to each comment.
 - d. Add/Remove and/or modify meals inside the menu. Modification can include price change, change of ingredients, and/or managing delivery options.
3. The Delivery guy delivers the order.
 - a. List all the orders handled by the delivery guy including assignment requests highlighting the decision if available.
 - b. Specificity either regions or restaurants to work with to be assigned orders if available.

- c. Accept/Reject delivery assignments. If an assignment is accepted, status of the delivery guy should be changed to “not_available” and all the other assignments waiting for a decision should be automatically rejected.
- d. Finalize an order.

Hospital Database Management System

1. Patient takes an appointment from a doctor. **(DESIGN REPORT)**
 - a. Look for an appointment from a doctor by specifying the department and date; listing available days for the chosen month. Assume for each day, there can only be a single appointment.
 - b. See his/her appointment history in his/her profile.
 - i. See diseases diagnosed for a particular appointment.
 - ii. See symptoms shared with the doctor for a particular appointment.
 - c. List his/her test results history sorted by date.
 - i. General Test View: when clicked on a particular test, list all the results of components.
 - ii. Detailed Component View: when clicked on a particular component inside the General Test View, show previous results of the component if available sorted by date.
2. Doctor sees his/her patients and manages appointments.
 - a. List appointments of patients for the entire month.
 - b. Manage days for the month by cancelling day slots (not appointments).
 - c. Write symptoms and ask for tests if required by choosing from a pre-defined (populated) set of available values for a particular visit. The tests are assigned to a random technician if expertise required matches.
 - d. Diagnose the patient by choosing appropriate diseases among a defined set. Before diagnosing, all the tests asked for have to be “finalized”.
3. Lab Technician carries out tests asked for patient visits.
 - a. List all the tests assigned to the technician by grouping them into “assigned”, “preparing”, “finalized” in the interface.
 - b. Assign scores to a particular component of a test. If any component’s result is determined for a test, the status of the test should be changed to “preparing”. If all components are determined, change the status into “finalized.”

Zoo Database Management System

1. Coordinator creates an event and assigns a cage to a keeper. **(DESIGN REPORT)**
 - a. Create a new event by entering necessary information and selecting one of the listed options of “group tour”, “educational program” or “conservation organization”.
 - b. List all the cages that are not assigned to any keeper yet.
 - c. Select a cage and assign it to a keeper.

- d. List all the cages that are assigned to keepers by this coordinator with necessary information related to each cage.
2. Keeper lists the cages that he/she is responsible for and regularizes the food for animals.
 - a. List all the cages that are assigned to this keeper.
 - b. Select a cage and list all the animals in that cage.
 - c. Select an animal.
 - d. List all training schedules of that animal with necessary information such as training date, topic, etc.
 - e. Select one of them and either remove the selected training schedule or update the *training date* for that animal.
3. Visitors make donations and comment on group tours.
 - a. List all conservation organizations.
 - b. Select a conservation organization.
 - c. Make a donation to that conservation organization by paying the desired amount.
 - i. Each visitor should have "*total amount of money*" for the reduction during the payment.
 - d. List all the group tours that this visitor attended before.
 - e. Select a group tour and write a comment to this group tour.

Hotel Database Management System

1. Guests book a reservation and give a food order. **(DESIGN REPORT)**
 - a. Books a reservation by entering necessary information.
 - b. Creates a food order by selecting one of the listed restaurants and picking one of food options.
 - c. List all the food orders of this guest.
 - d. Select a food order and show detailed information for that food order such as date, delivery status, housekeeper that delivered this order, delivered food, restaurant served, etc.
2. Manager assigns food orders to housekeepers and evaluates training program applications.
 - a. List all food orders that are not assigned to any housekeeper yet.
 - b. Select one of the food orders and assign this order to one of the listed housekeepers.
 - i. The delivery status of that food order should be "*pending*".
 - c. List all of the training programs that were created by this manager.
 - d. Select a training program.
 - e. List the housekeepers applied to this training program.
 - f. Accept or reject the application by updating the acceptance status to "*accepted*" or "*rejected*".

- i. The application can only be accepted/rejected if the acceptance status is *"pending"*.
- 3. Housekeeper delivers assigned food orders and makes applications to training programs.
 - a. List all assigned orders to this housekeeper with necessary information such as date, delivery status, delivered food, restaurant served, guest ordered, etc.
 - b. Deliver a food order by updating the delivery status of that food order to *"delivered"*.
 - i. The food order can only be delivered if the delivery status is *"pending"*.
 - c. List all of the available training programs.
 - d. Apply to a training program by selecting one of them.
 - i. The acceptance status for that applied training program should be *"pending"*.
 - e. List all the training programs applied by this housekeeper with necessary details such as acceptance status, date of program, manager that assigned the delivery, etc.