

Project: Clinic Reservation System (Phase 1: 100 points + 20 points bonus) Imagine

that we are designing a web application for clinic appointment reservations. It is required to implement the following features:

- 1- Sign In (10 points)
- 2- Sign up (10 points)
- 3- Doctor set his schedule. (Inserting a slot) (10 points)
- 4- Patients select doctor, view his available slots, then patient chooses a slot. (10 points)
- 5- Patient can update his appointment by change the doctor or the slot. (10 points)
- 6- Patient can cancel his appointment. (10 points)
- 7- Patients can view all his reservations. (10 points)

Expectation:

- As per the stack specified for each team (setting the team tech stack is generated randomly [Link here](#))
- you need to implement.
 - Front end layer consists of simple screens to simulate successful flow
 - Backend layer contains 7 features APIs
 - Database layer
- Successful integration between the 3 layers (10 points)
- Application build successfully, flows are successfully implemented (10 points)
- Code (front end, backend) pushed on github (10 points)

Bonus (20 points):

Integrate a messaging feature like kafka or rabbitMq to notify doctor when a patient reserve, update or cancel a reservation.

By creating topic called "clinic_reservation"

Event structure:

{ "doctorId": "111",

"patientId": "222",

"Operation": "ReservationCreated" **or** "ReservationUpdated" or "ReservationCancelled" }

Create page in doctor with one button to fetch the doctor's messages.

Ground rules:

- Projects will be submitted via google classroom.
- Deadline: 11 November 2023 11:59 pm
- Late Submission till 13 November 2023 11:59 pm with penalty -20 points AND bonus part will not be considered.
- Violating the technology stack for team: -20 points
- You need to submit one zip folder its name following schema:
{GroupNum_ID1_ID2_ID2_TA name.zip}
- Violating naming convention: -5 points
- Cheating cases: -5 Marks (not points) from the actual year work grades.

Project grading criteria : [Link](#)

- Additional resources: (suggested resources you can start with) ○ Go lang :
<https://medium.com/@rasoky.maulana/create-a-crud-rest-api-using-mysql-golang-part-2-5af4942abdd2>
 - Python: <https://techarise.com/restful-api-in-python-mysql/>
 - NodeJs: <https://www.bezkoder.com/node-js-rest-api-express-mysql/>
 - C # : <https://www.youtube.com/watch?v=O5hKoBV3vaU>
 - React: <https://www.bezkoder.com/react-crud-web-api/>
 - Angular <https://www.bezkoder.com/angular-16-crud-example/>

Suggested screens: feel free to add whatever you want but make it simple.

Hello, User (user type: Patient)

My Appoinments

Appointment		Doctor		
10/11/2023	1:00 PM	Dr.Ahmed	Edit	Cancel
15/03/2022	5:00 PM	Dr.Mostafa	Edit	Cancel
30/9/2021	1:00 PM	Dr.Ali	Edit	Cancel
15/04/2021	5:00 PM	Dr.Mohamed	Edit	Cancel

Create new appointment

Choose Doctor	drop down
Appointment	

Reserve

Hello, User (user type: Doctor)

My Slots

Date	Hour		
10/11/2023	1:00 PM	Edit	Cancel
15/03/2022	5:00 PM	Edit	Cancel

Create new slot

Date	10/11/2023
Hour	1:00 PM

Add Slot

Email

Password

User Type: ☐ Patient ☐ Doctor

Sign up

Email

Password

Login