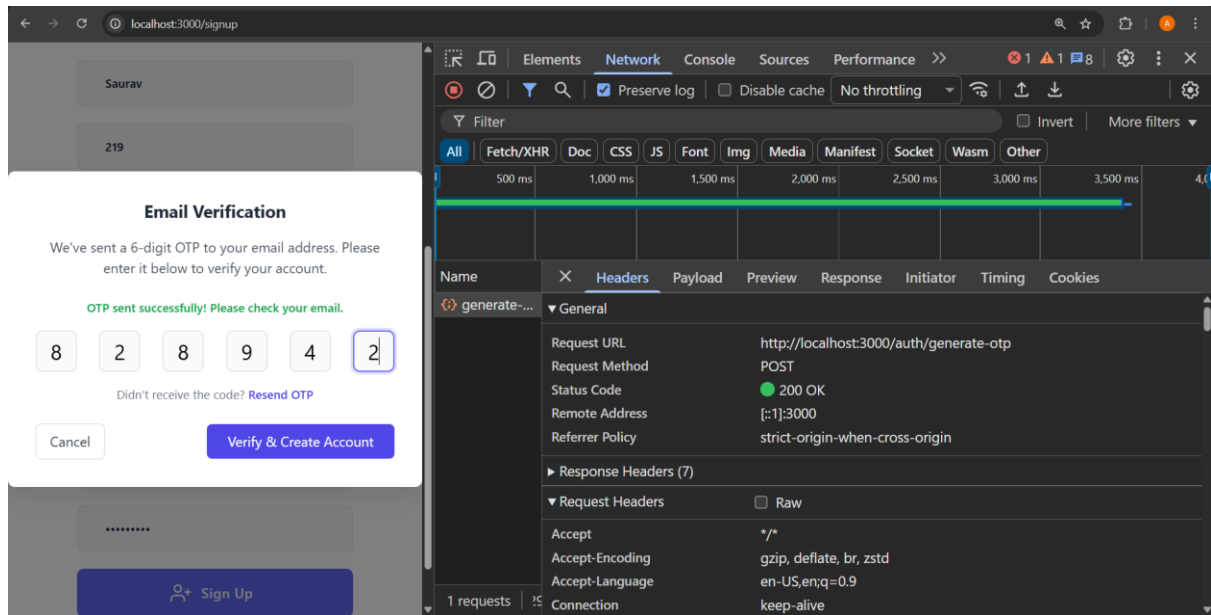


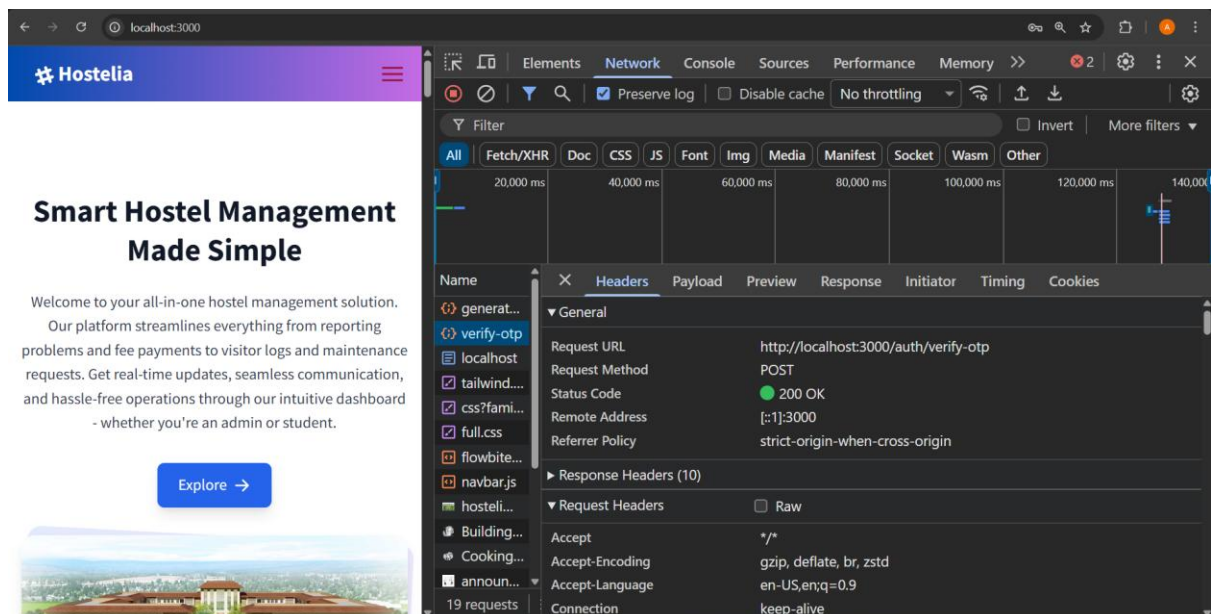
Network Tab Evidence

1. Signup

➔ OTP generation for authentication.
“/auth/generate-otp”



➔ OTP verification.
“/auth/verify-otp”



2. Logout/Login

➔ “auth/logout”

The screenshot shows the Hostelia application running on localhost:3000. The main heading is "Smart Hostel Management Made Simple". Below it, a welcome message states: "Welcome to your all-in-one hostel management solution. Our platform streamlines everything from reporting problems and fee payments to visitor logs and maintenance requests. Get real-time updates, seamless communication, and hassle-free operations through our intuitive dashboard - whether you're an admin or student." An "Explore" button is visible. The browser's Network tab is open, showing a list of requests. The selected request is "logout" with a status of 302 Found. The request details are as follows:

General	
Request URL	http://localhost:3000/auth/logout
Request Method	POST
Status Code	302 Found
Remote Address	[::1]:3000
Referrer Policy	strict-origin-when-cross-origin

The response headers are listed below:

Response Headers (14)	
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding	gzip, deflate, br, zstd

➔ “auth/login”

The screenshot shows the Hostelia application running on localhost:3000/dashboard. The main heading is "Dashboard". Below it, a user profile is displayed for "Saurav" (Student ID 219) with the email "saurav.s23@iiits.in" and the year "UG-3". The browser's Network tab is open, showing a list of requests. The selected request is "login" with a status of 200 OK. The request details are as follows:

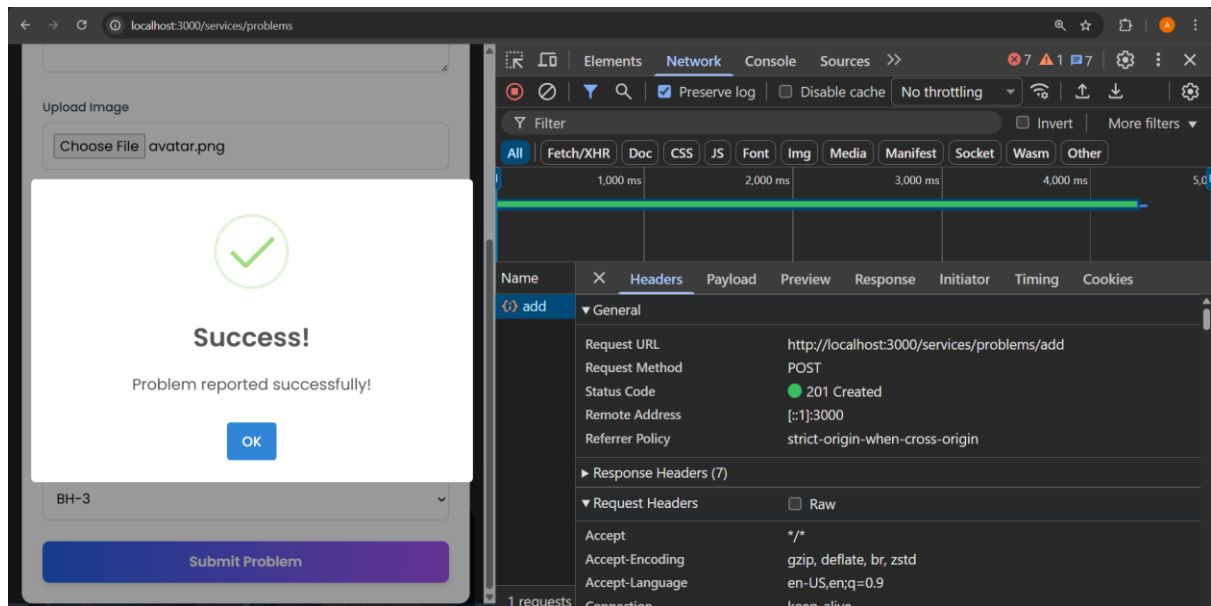
General	
Request URL	http://localhost:3000/auth/login
Request Method	POST
Status Code	200 OK
Remote Address	[::1]:3000
Referrer Policy	strict-origin-when-cross-origin

The response headers are listed below:

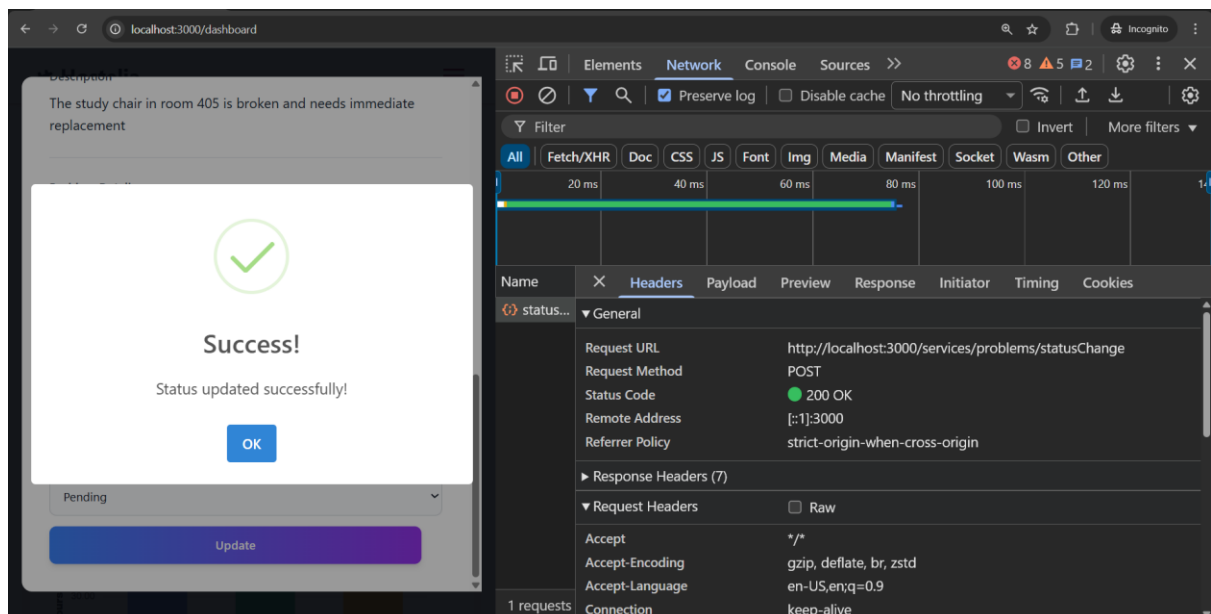
Response Headers (10)	
Accept	*/*
Accept-Encoding	gzip, deflate, br, zstd
Accept-Language	en-US,en;q=0.9

3.Services

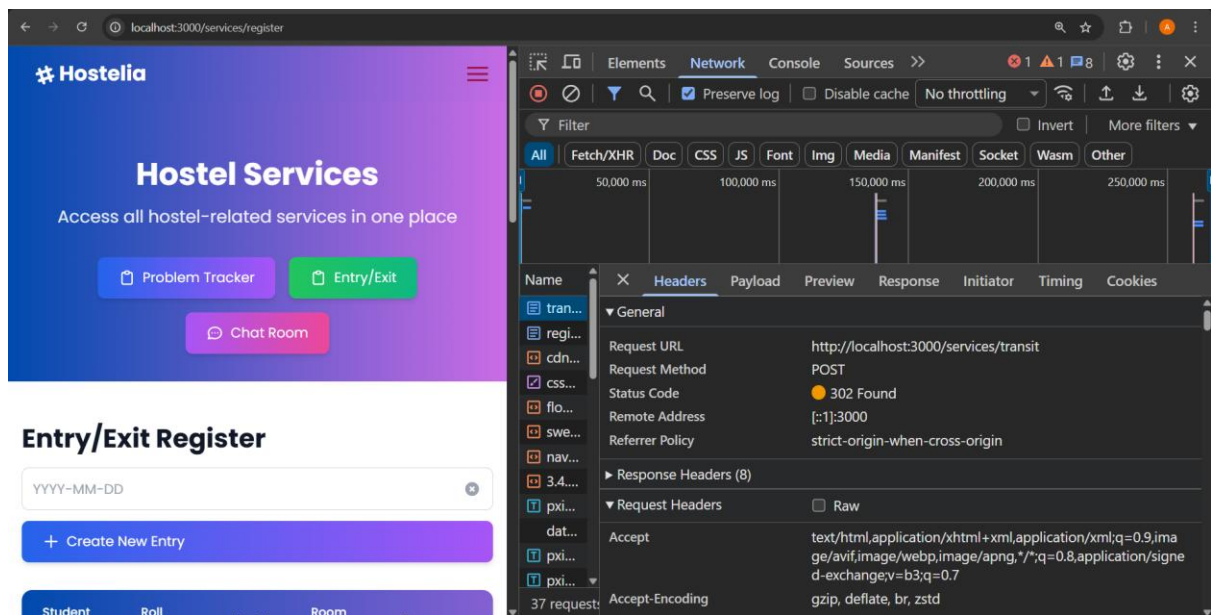
➔ “/services/problems/add”



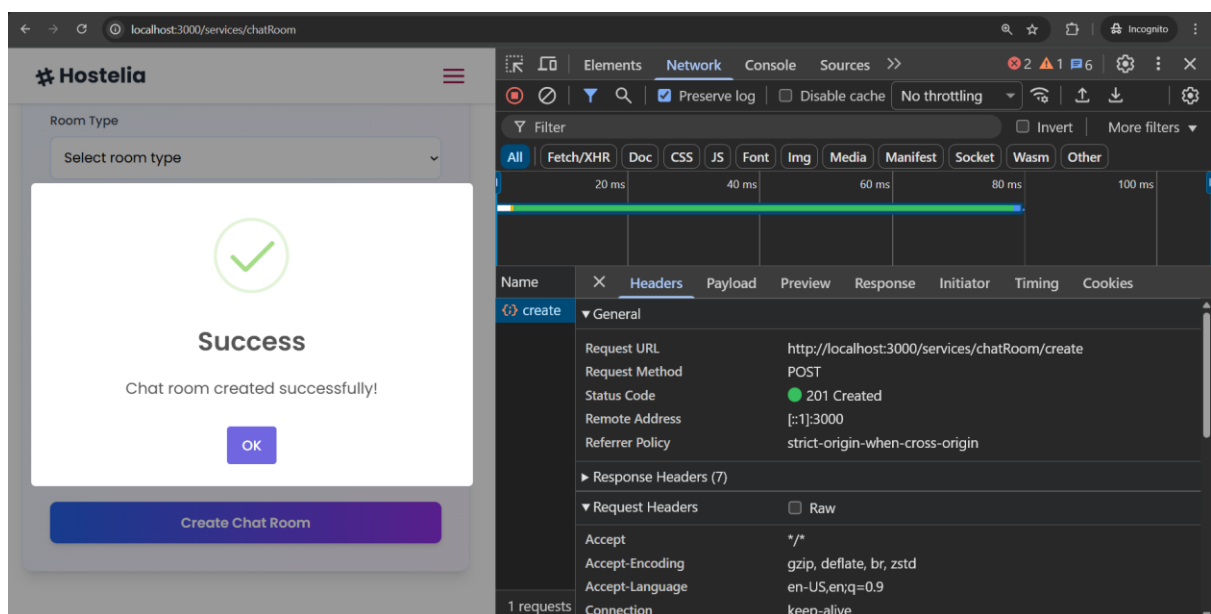
➔ Problems Status –
“/services/problems/statusChange”



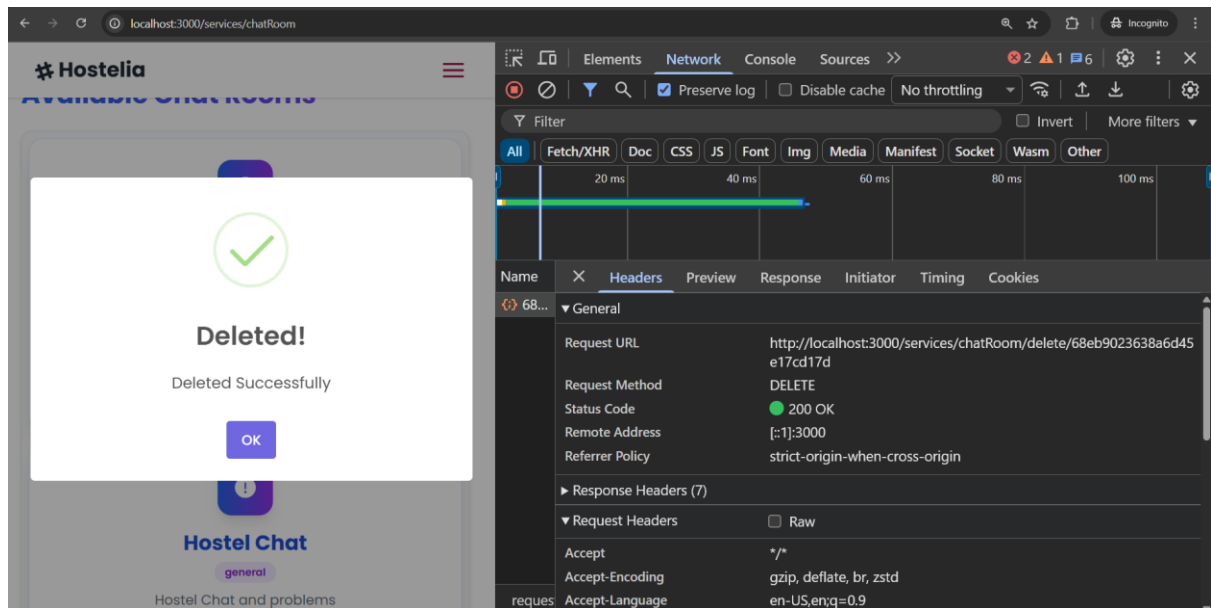
➔ Hostel Entry/Exit – “/services/transit”



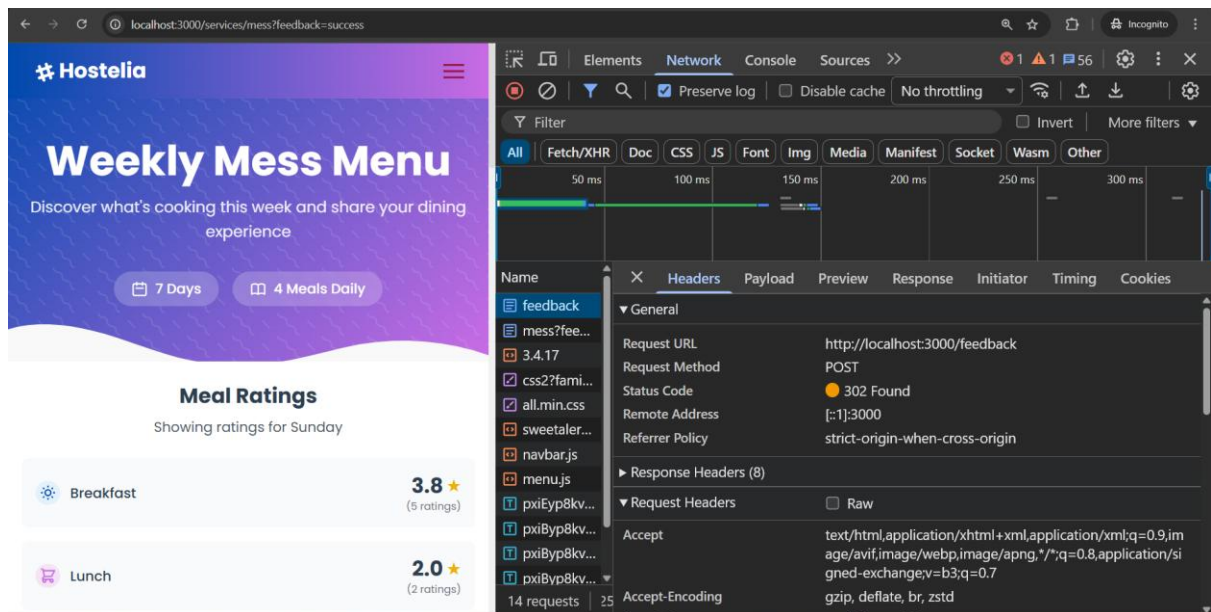
➔ Chat room creation – “services/chatRoom/create”



➔ Chat room deletion – “/services/chatRoom/delete/:id”



➔ Mess menu feedback – “/feedback”



➔ Fees Confirmation – “/services/fee-status”

The screenshot displays a web application interface on the left and a Chrome DevTools Network tab on the right. The web application, titled "Student Details", shows personal information for a student named Saurav (Roll No: 219, Email: saurav.s23@iiits.in, Year: UG-3) and hostel information (Hostel: BH-3, Room No: 117). The hostel fees are listed as "Paid" for the Hostel Fee and "Unpaid" for the Mess Fee, both with "Change" buttons. A "Send Reminder" button is visible at the bottom. The Network tab shows a POST request to `http://localhost:3000/services/fee-status` with a status code of 200 OK. The request headers include `Accept: */*`, `Accept-Encoding: gzip, deflate, br, zstd`, and `Accept-Language: en-US,en;q=0.9`. The response headers include `Content-Type: application/json` and `strict-origin-when-cross-origin`.

➔ Fees Reminder – “send-fee-reminder”

The screenshot displays the same web application interface on the left, but now with a confirmation modal open. The modal shows a green checkmark and the text "Email Sent! Fee reminder sent to Saurav", with an "OK" button. The "Send Reminder" button is still visible. The Network tab shows a POST request to `http://localhost:3000/send-fee-reminder` with a status code of 200 OK. The request headers are the same as in the previous screenshot. The response headers include `Content-Type: application/json` and `strict-origin-when-cross-origin`.

→ Creating Wardens (Only by admins) – “/services/users/add-warden”

