## Chapter 6: Resource Planner

### Project Module

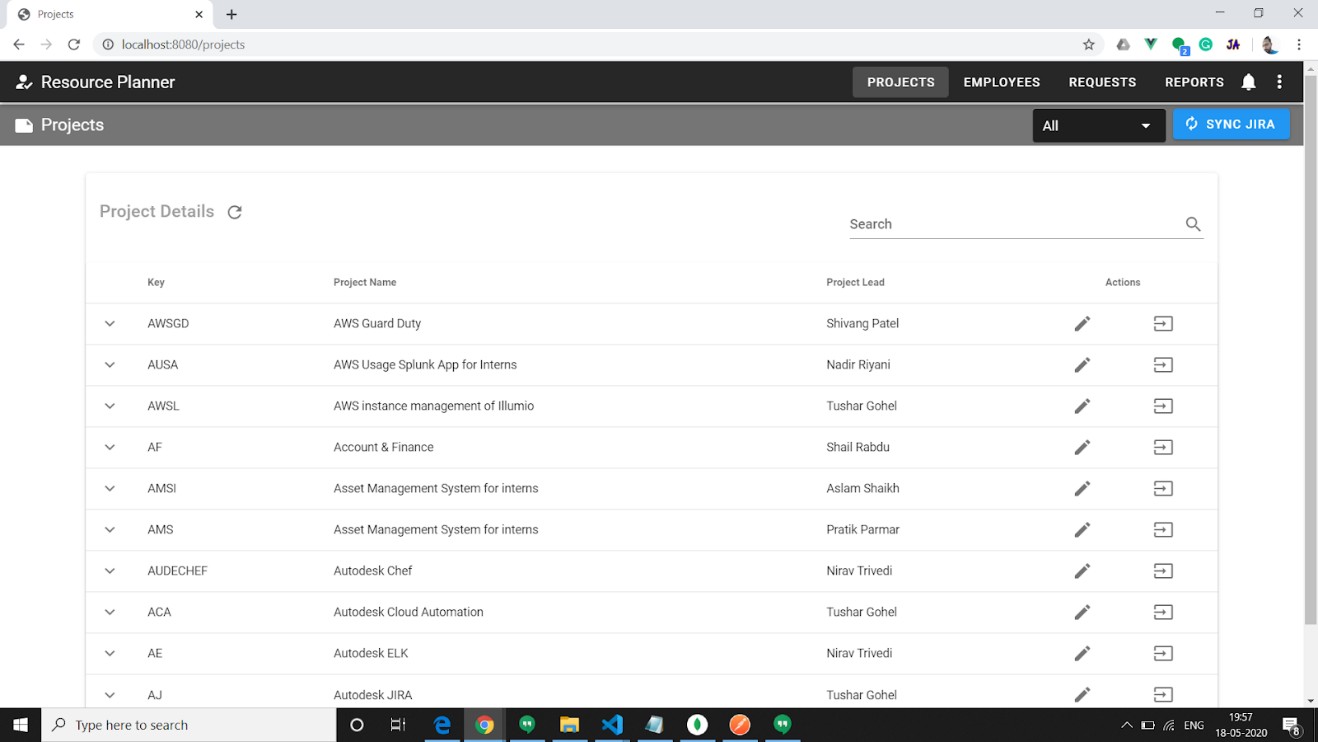


Figure 6. 1 Project listing page view

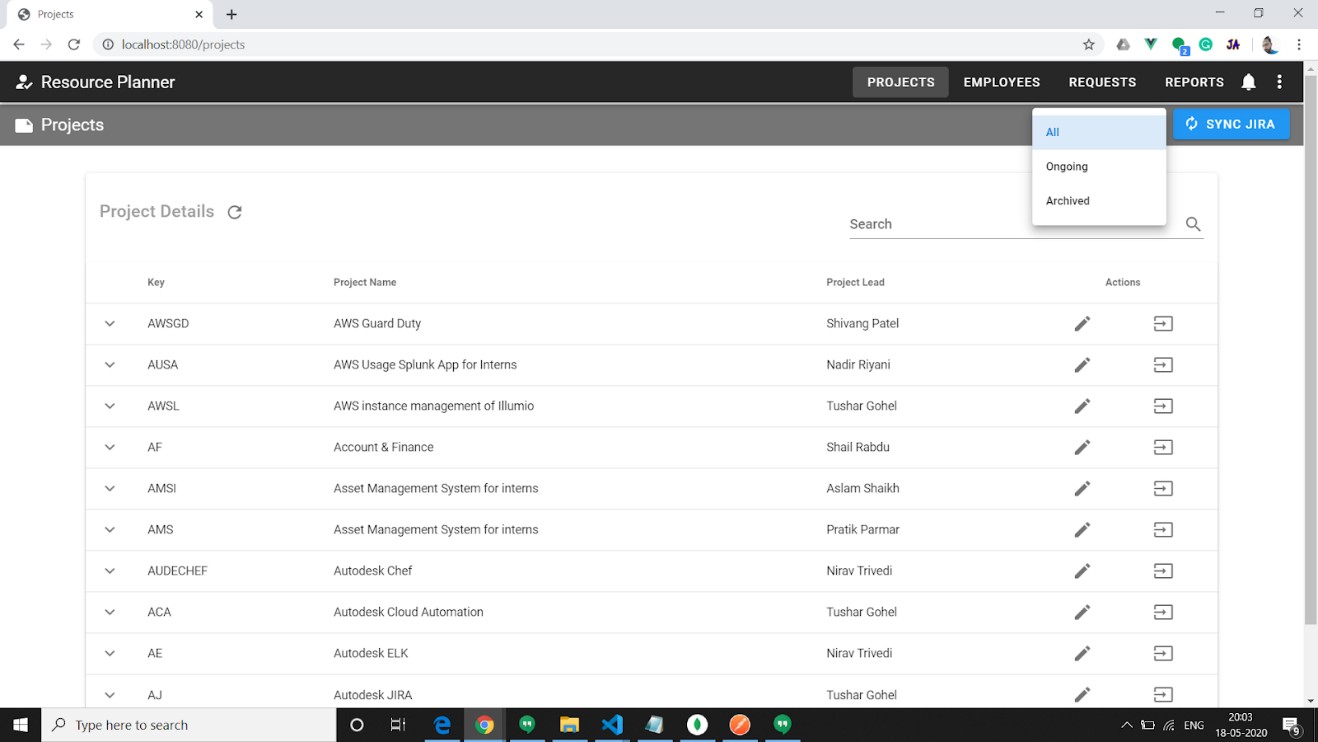


Figure 6. 2 Filters to apply on project

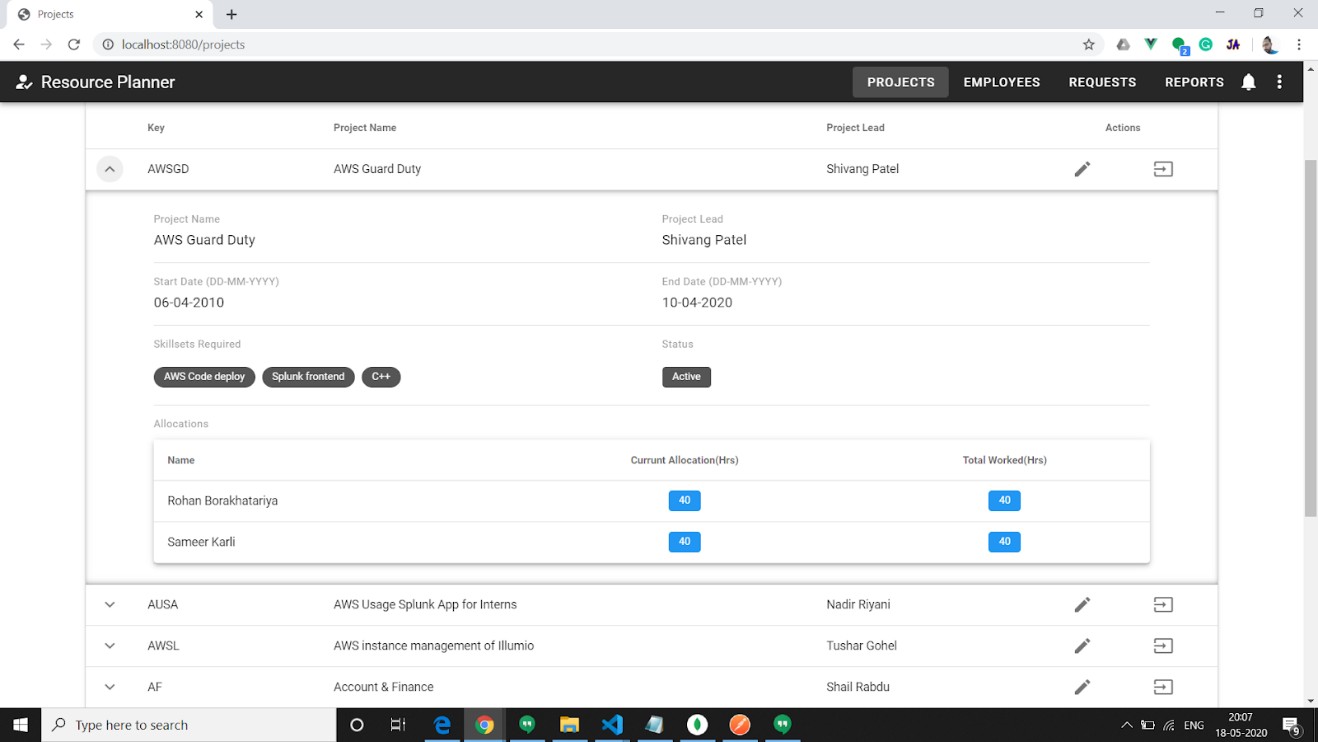


Figure 6. 3 Project expand detail view

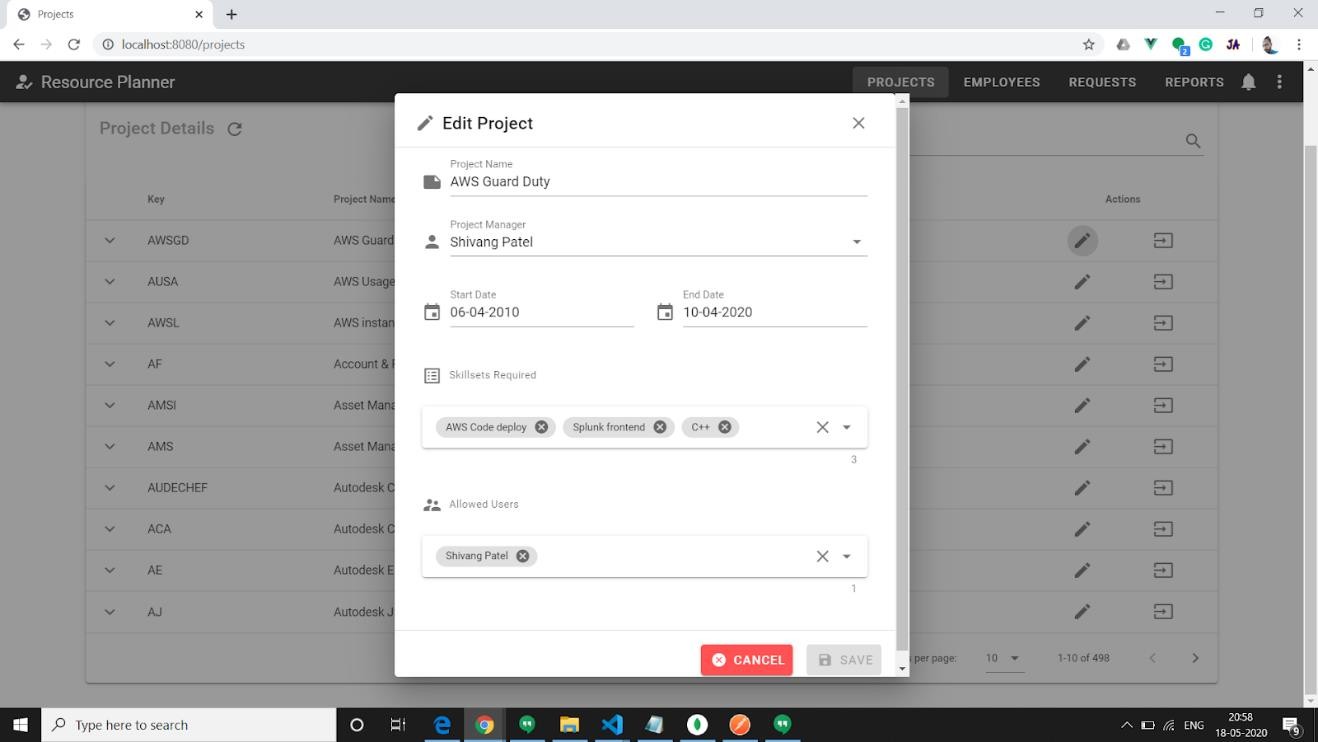


Figure 6. 4 Project edit popup

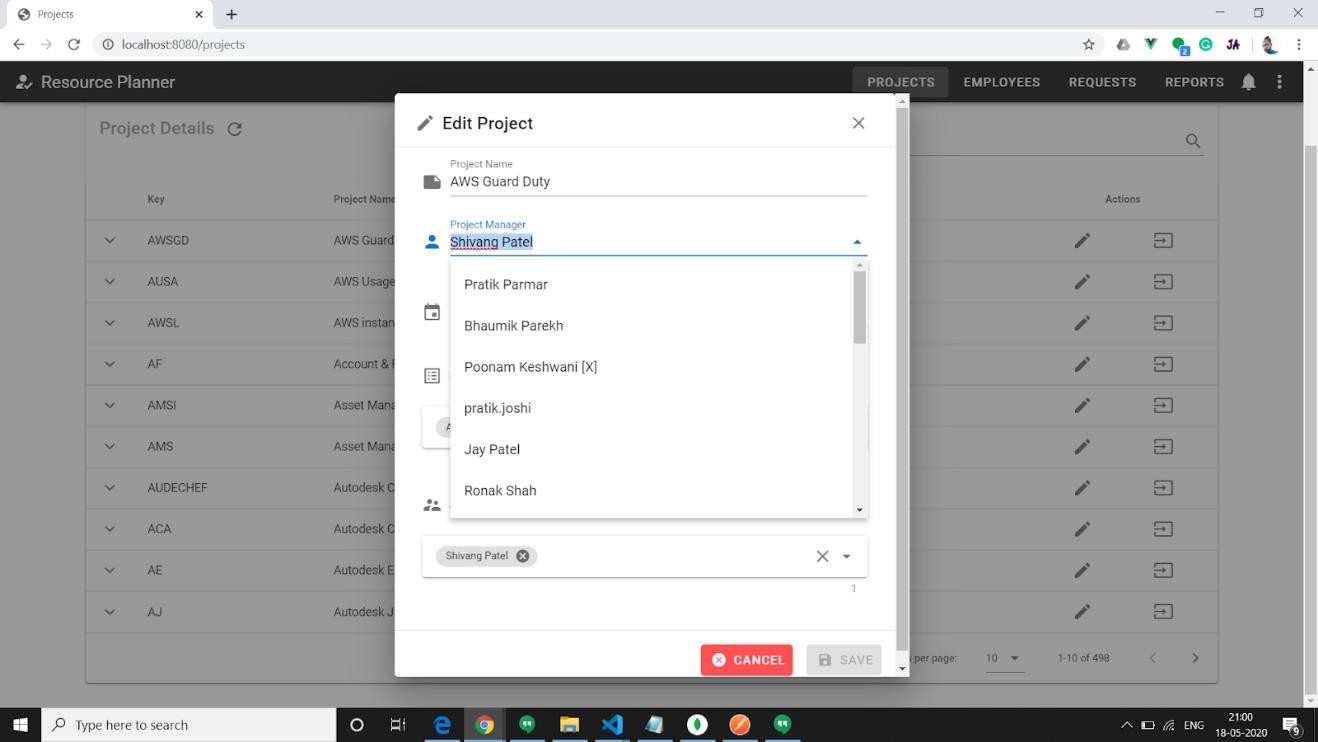


Figure 6. 5 Choosing project manager from dropdown

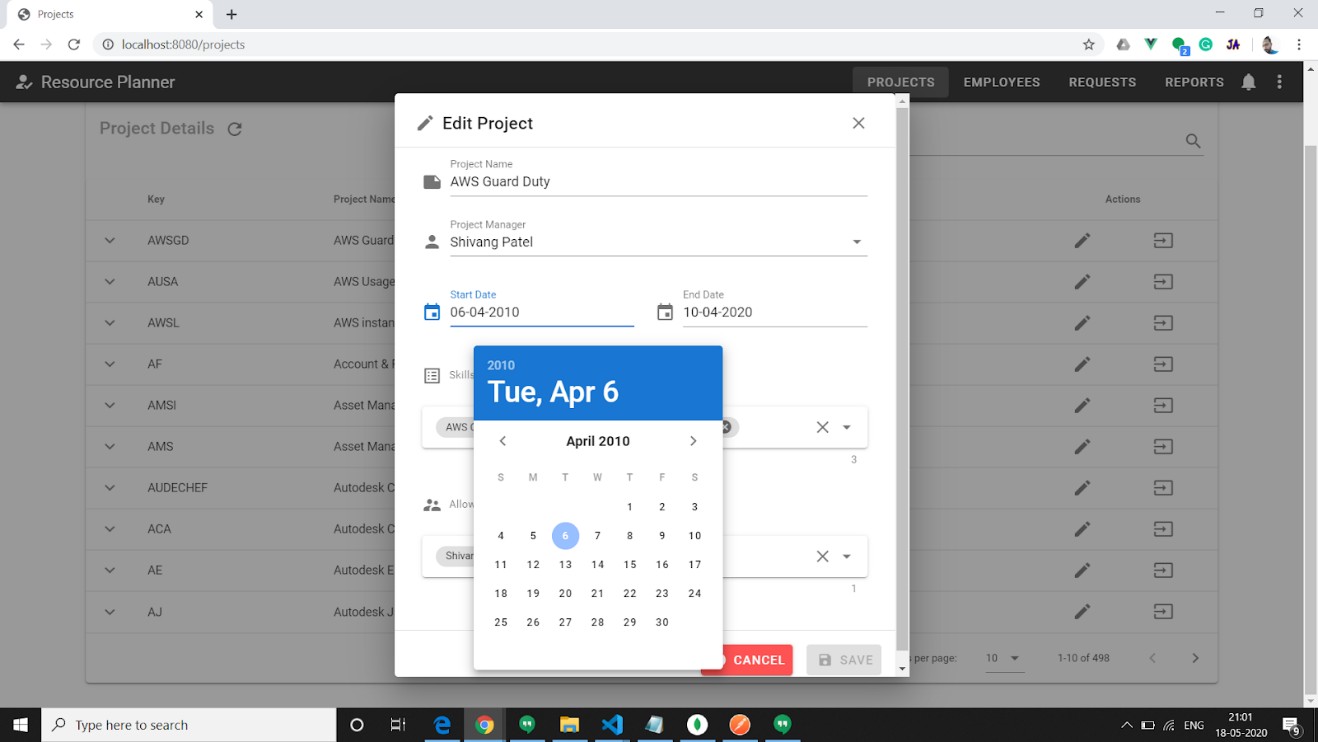


Figure 6. 6 Date selectors

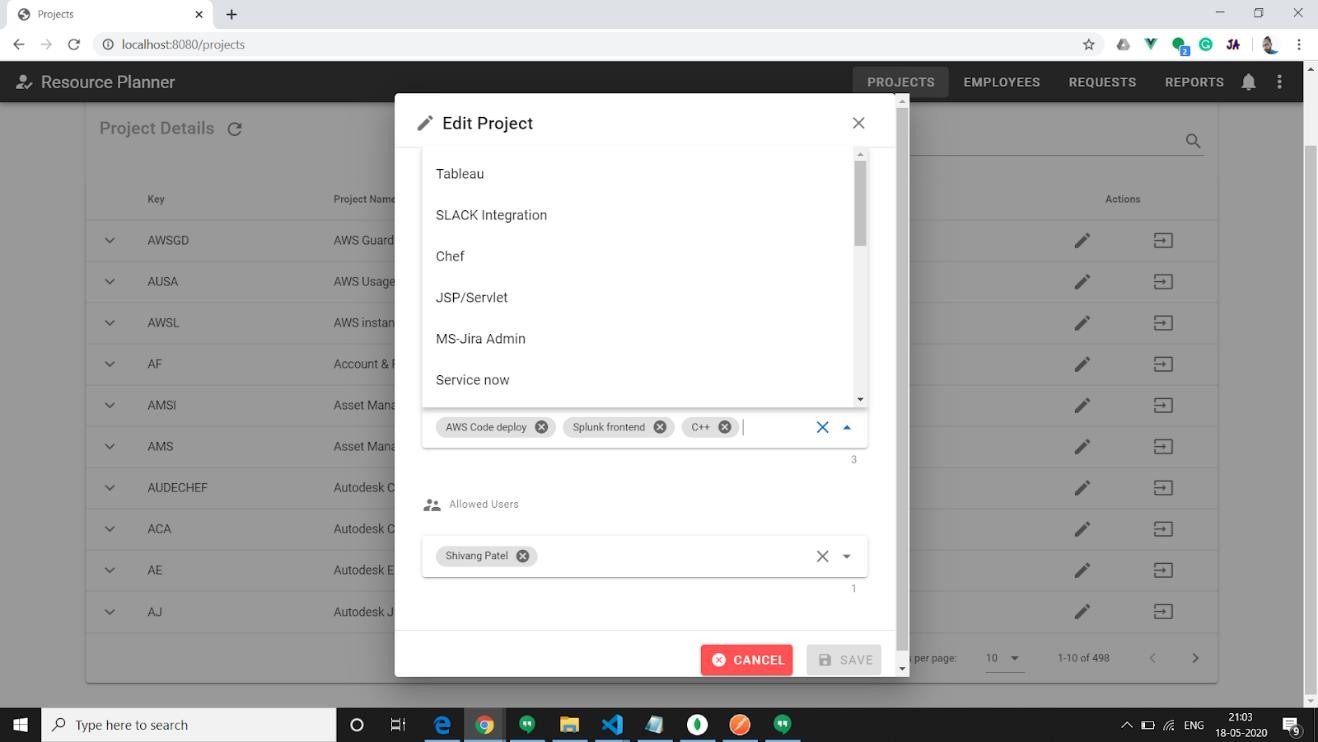


Figure 6. 7 Multiple skill selection

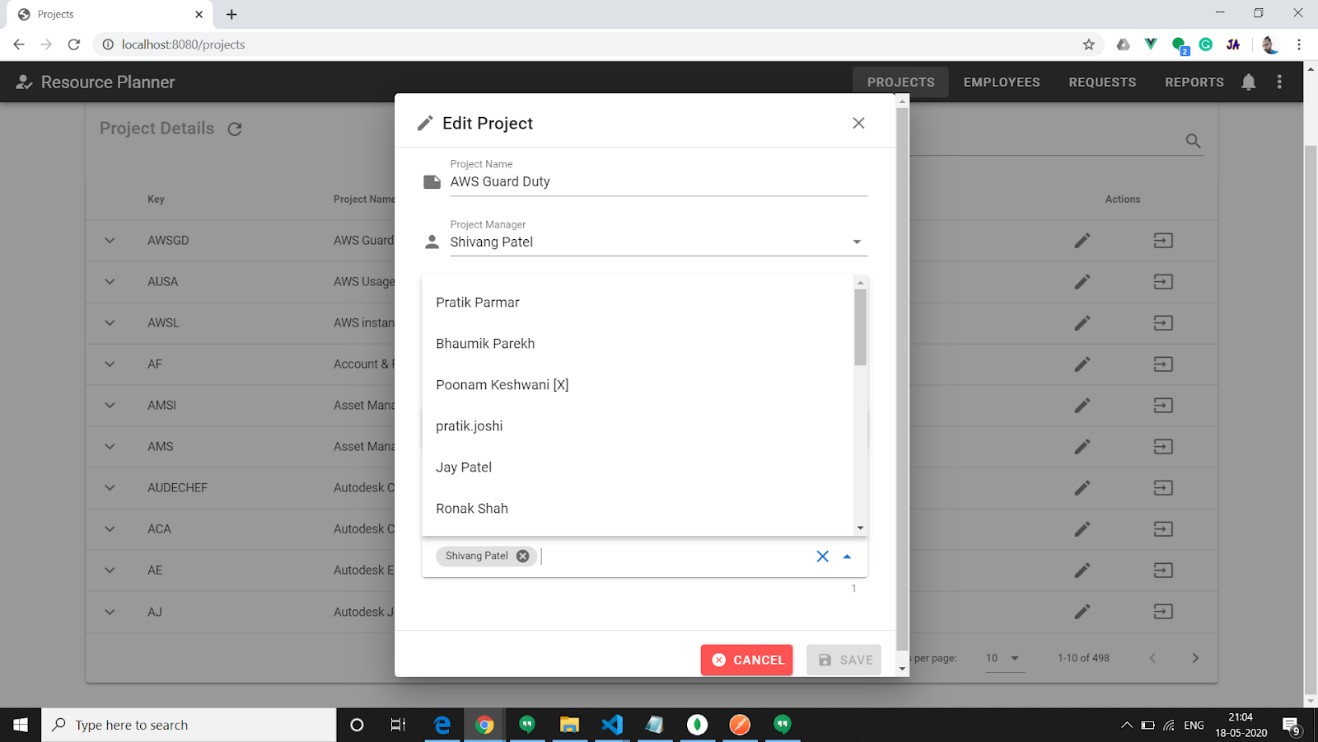


Figure 6. 8 Multiple project manager selection

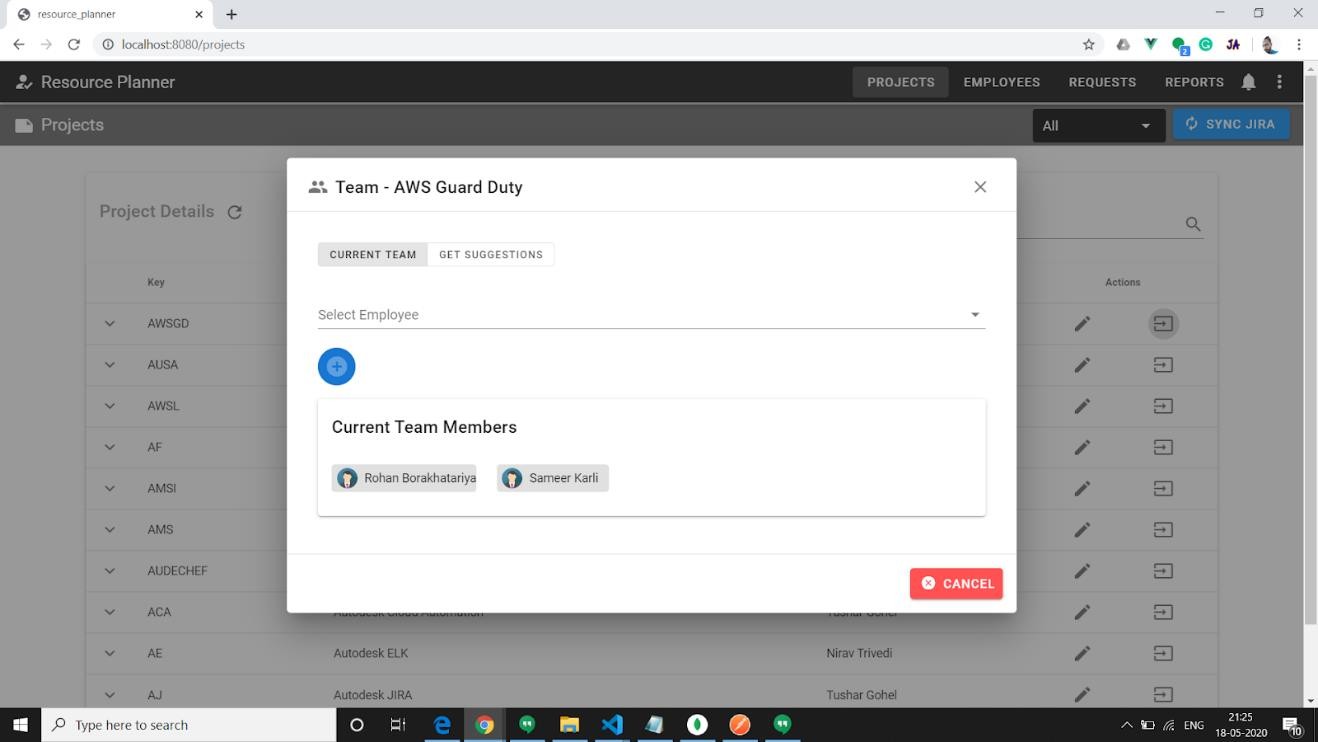


Figure 6. 9 Project team popup

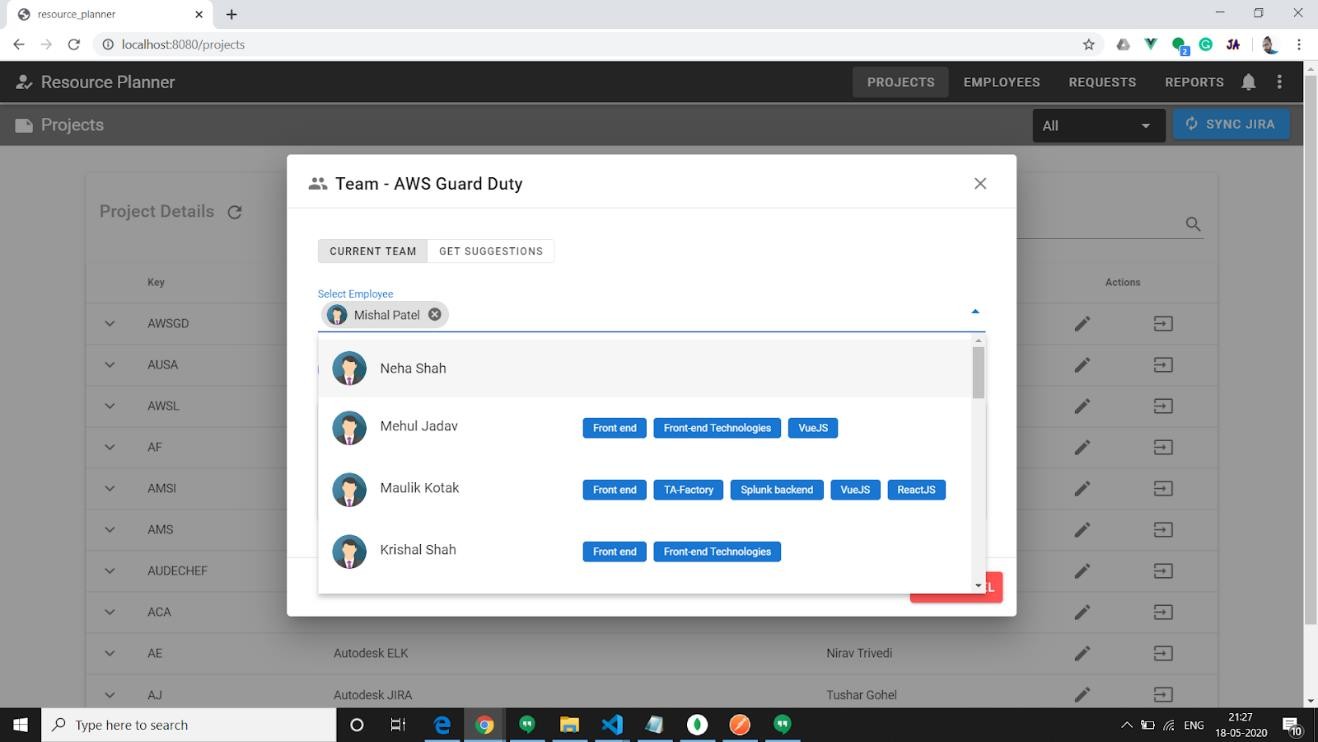


Figure 6. 10 Employee selection

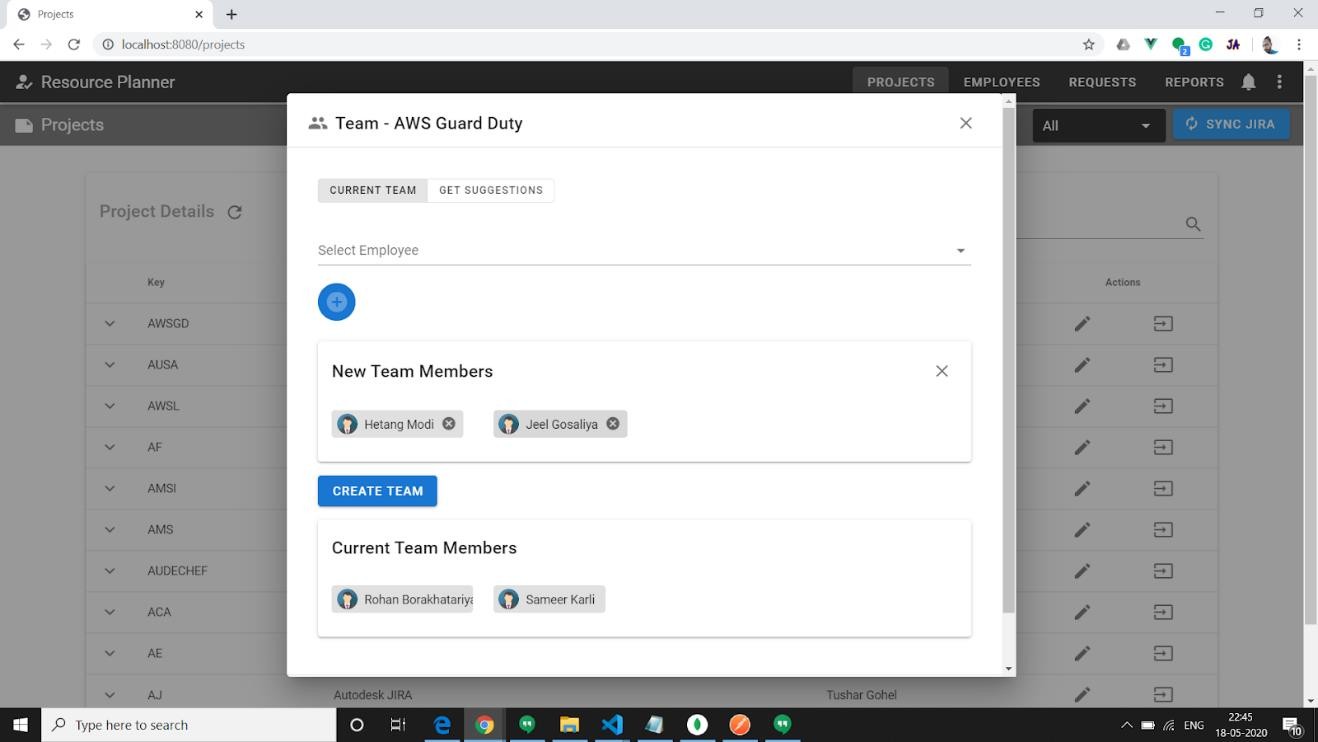


Figure 6. 11 Creating team

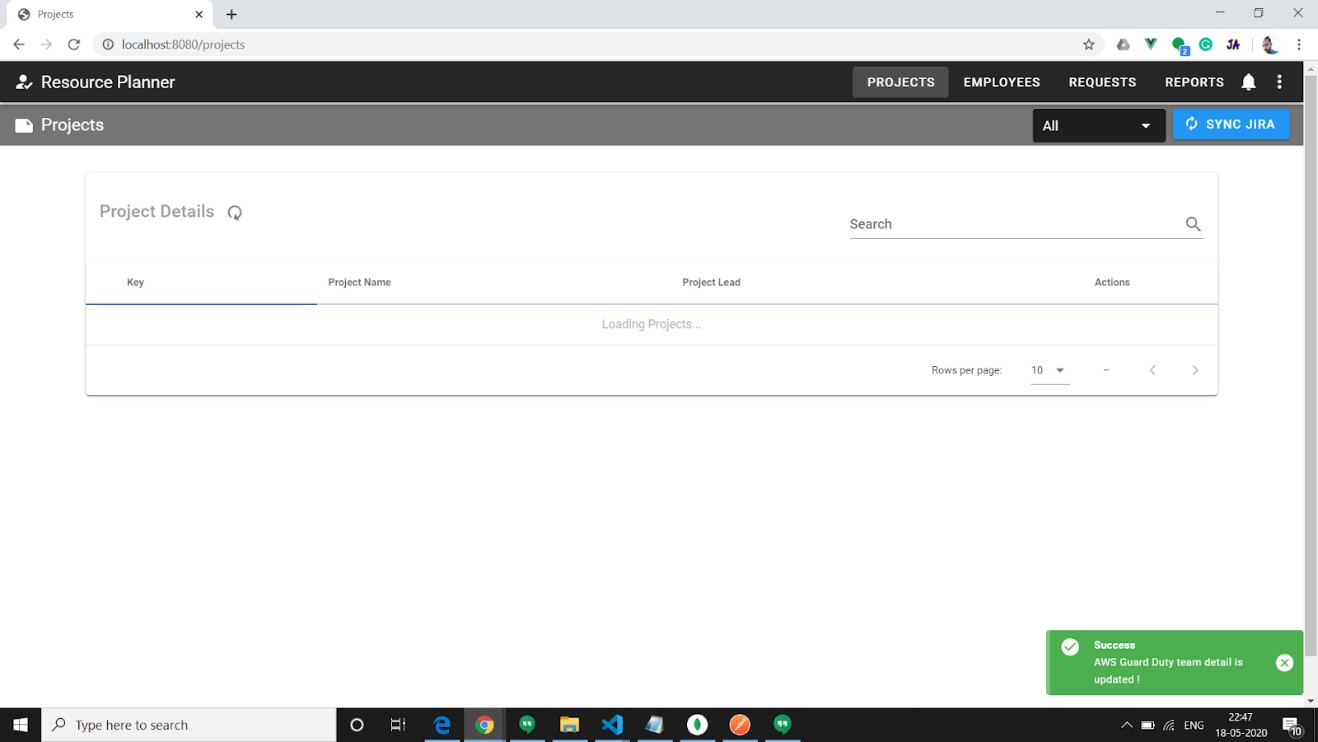


Figure 6. 12 Successful creations of team

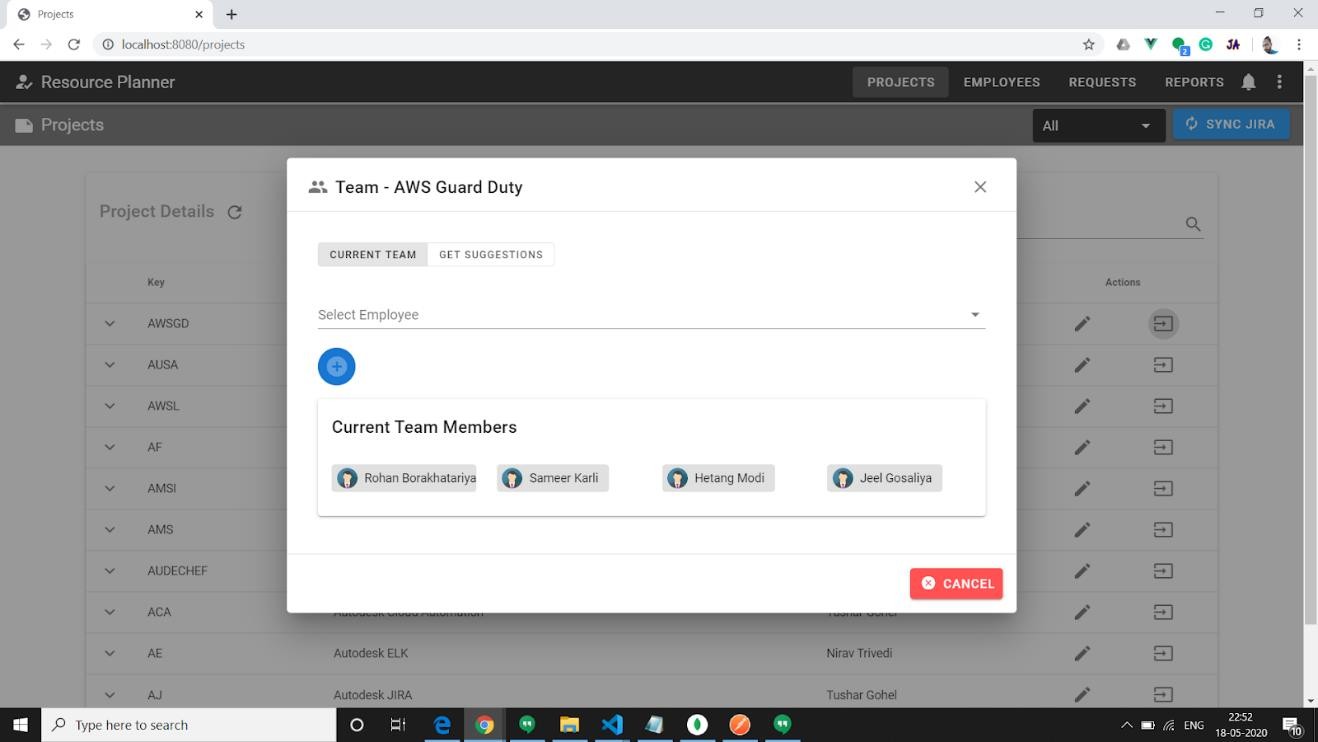


Figure 6. 13 Showing newly created team

### Employee Module

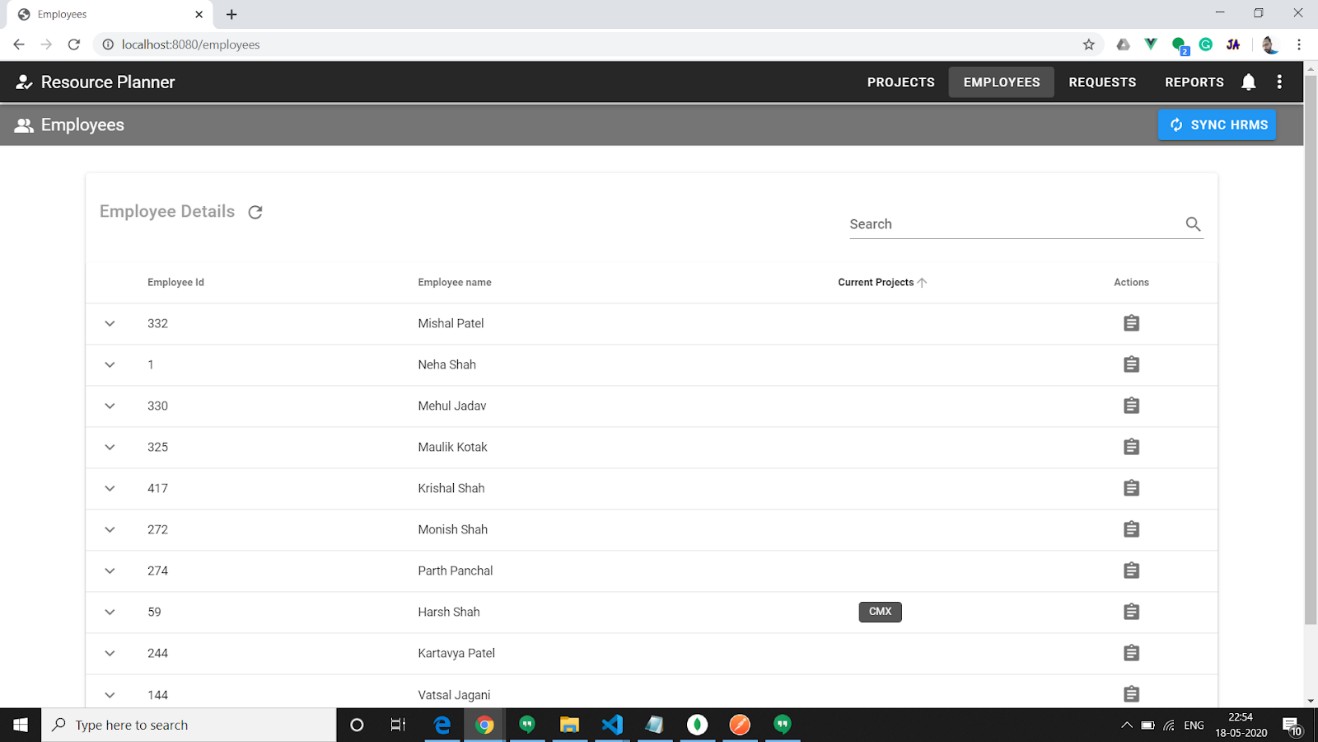


Figure 6. 14 Employee Listing

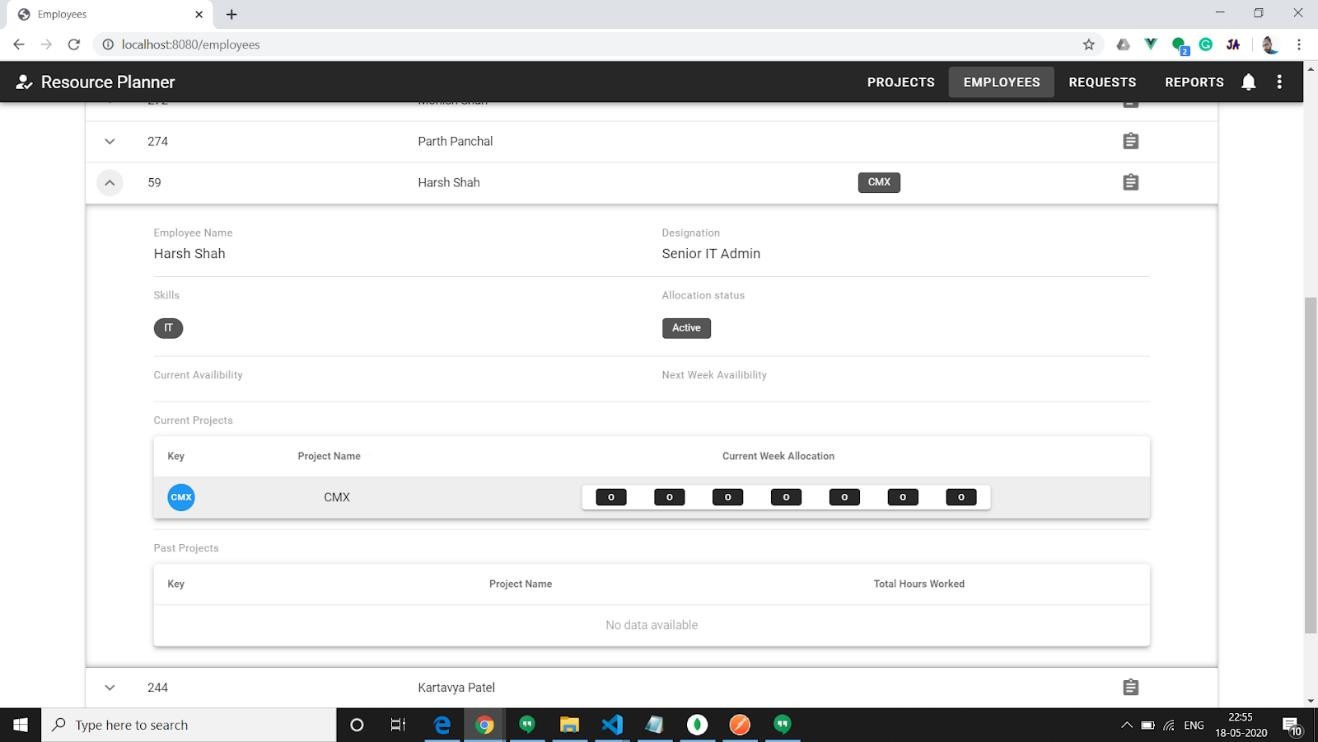


Figure 6. 15 Employee Expand details

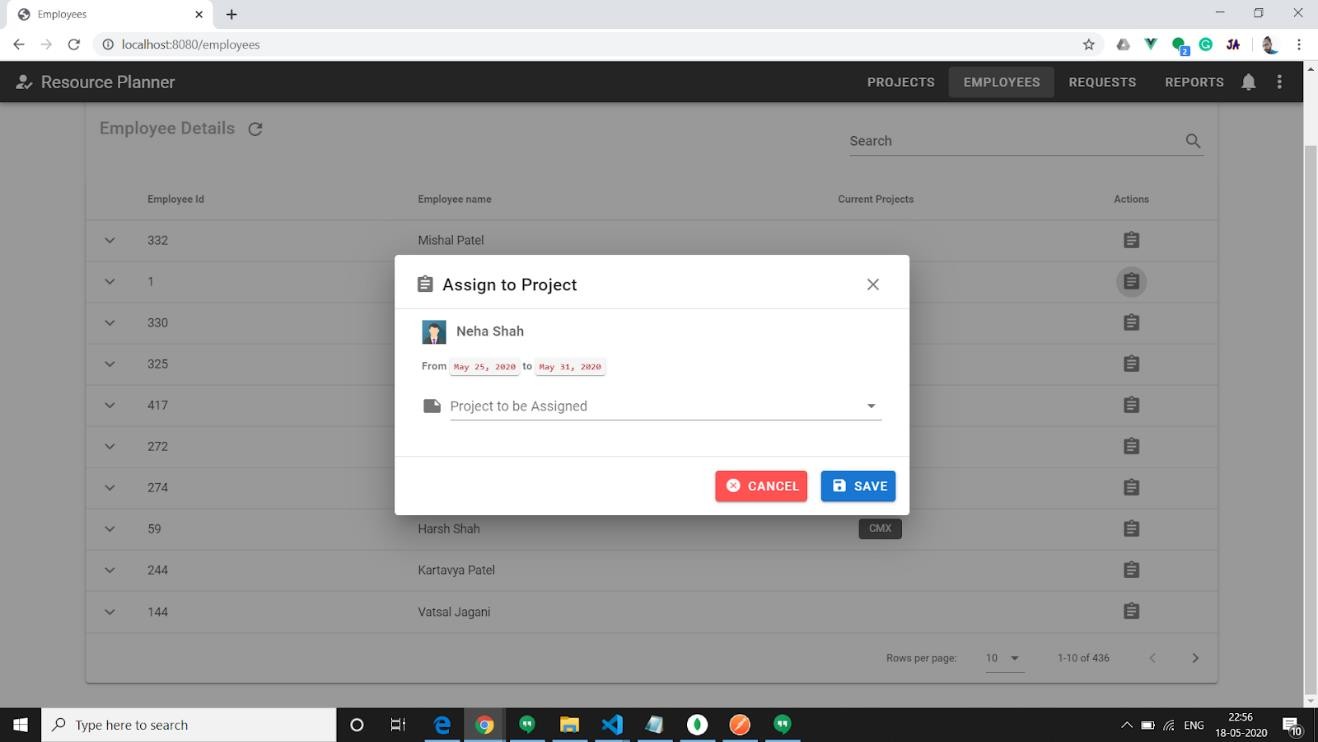


Figure 6. 16 Direct employee assignment to project popup

[Created by partner **Dhruvil Megha**]

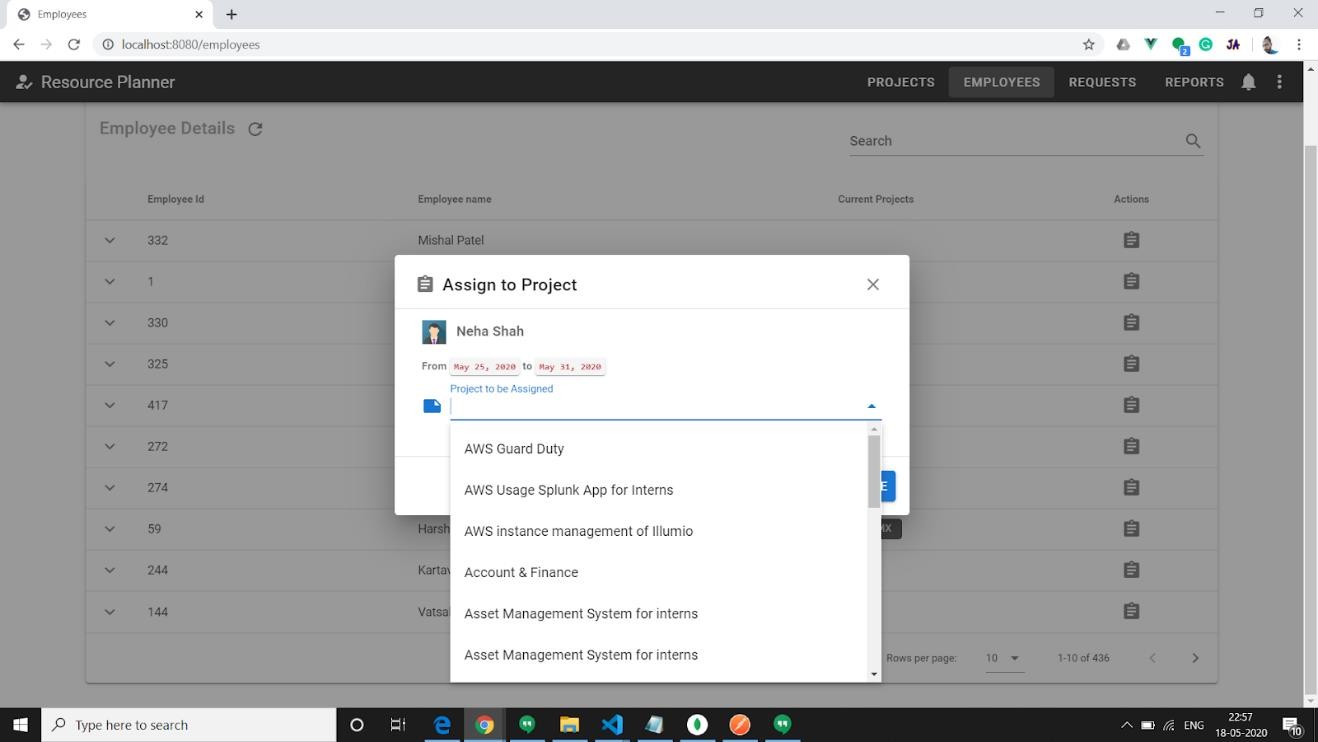


Figure 6. 17 Project selection form drop down

[Created by partner **Dhruvil Megha**]

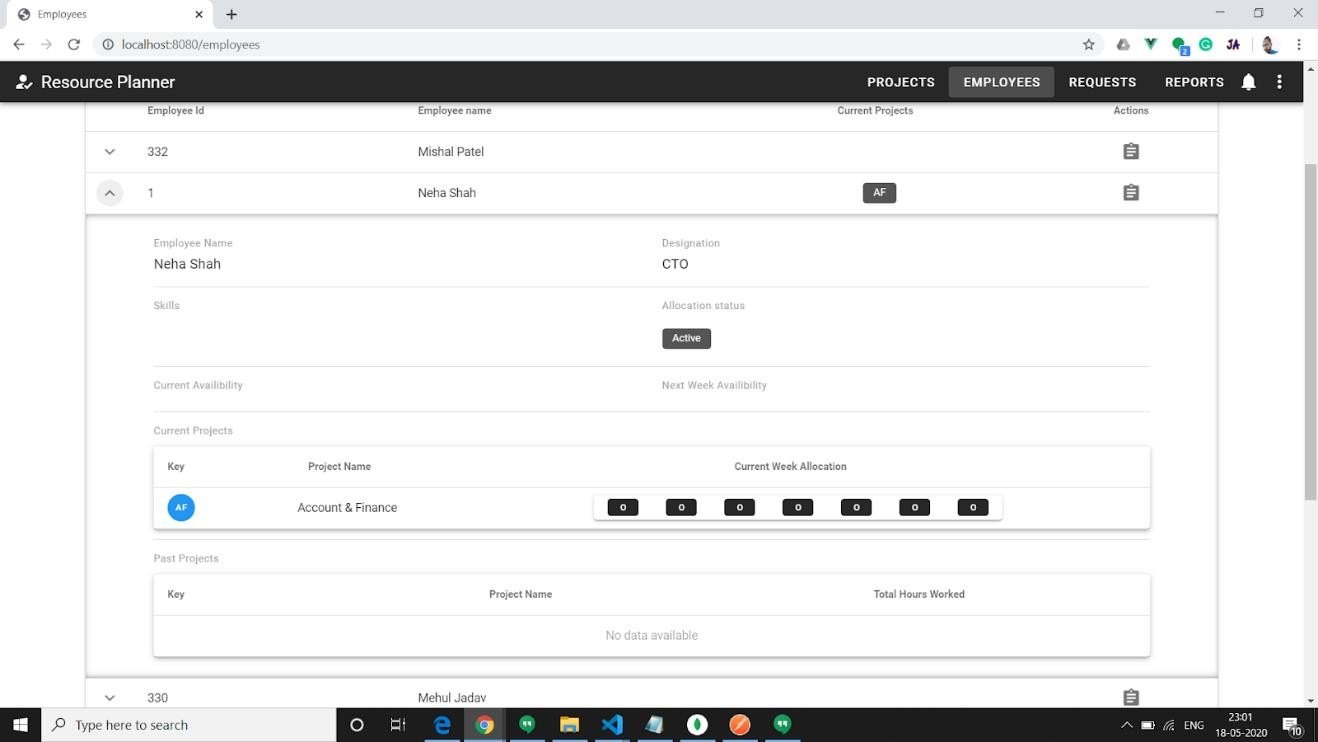


Figure 6. 18 After adding in project

### Request Module

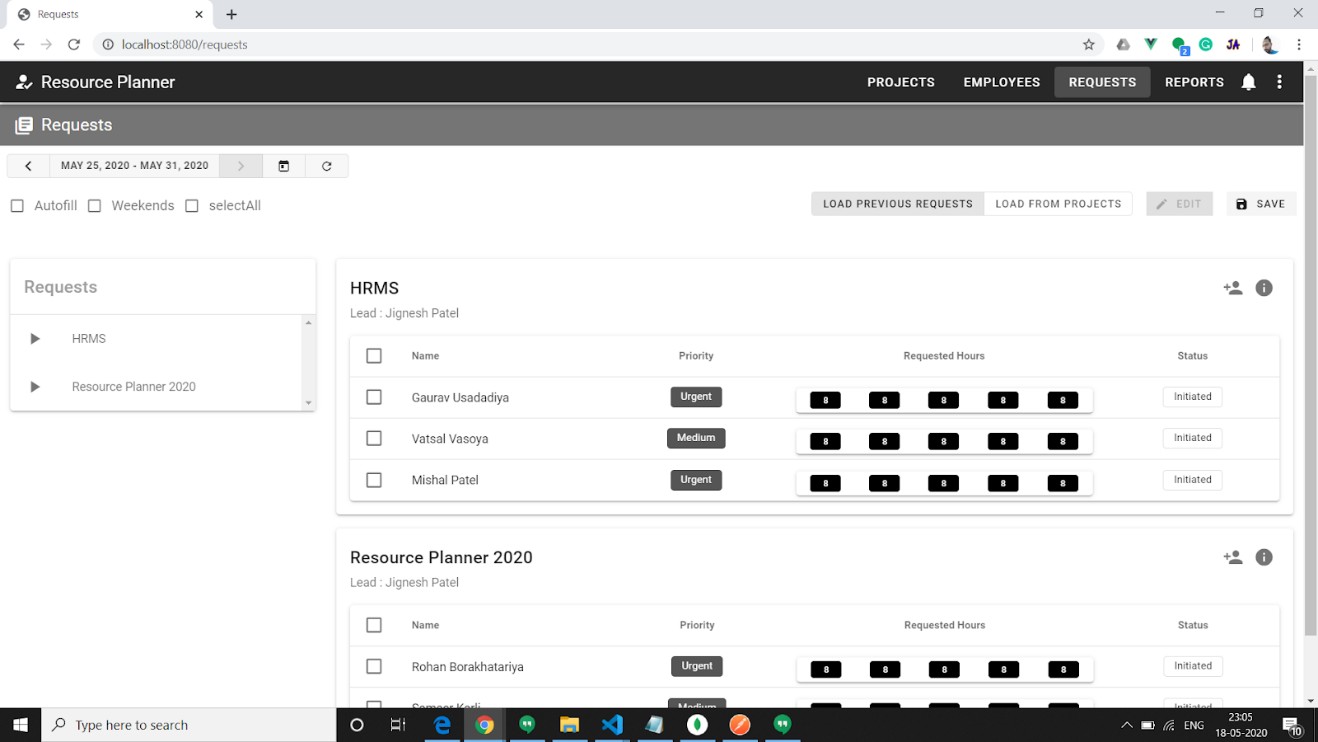


Figure 6. 19 Request view landing page

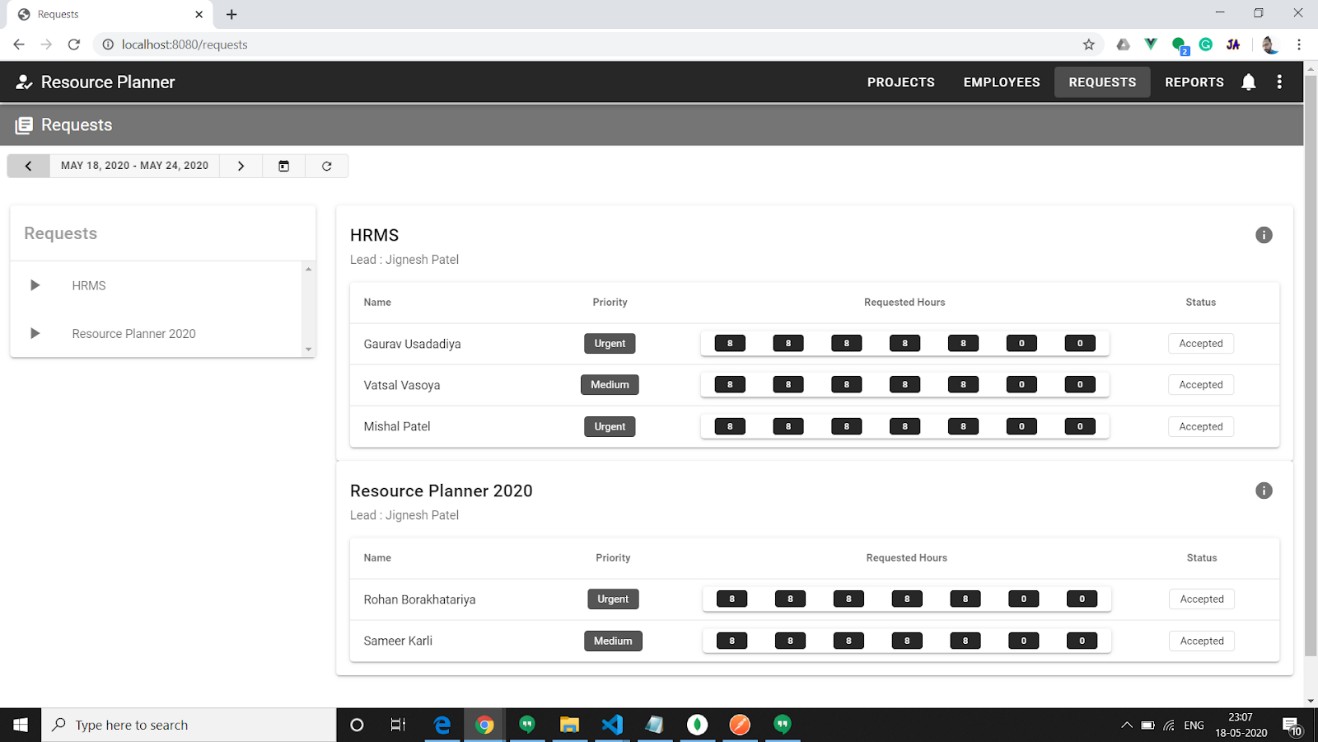


Figure 6. 20 Going back through timeline

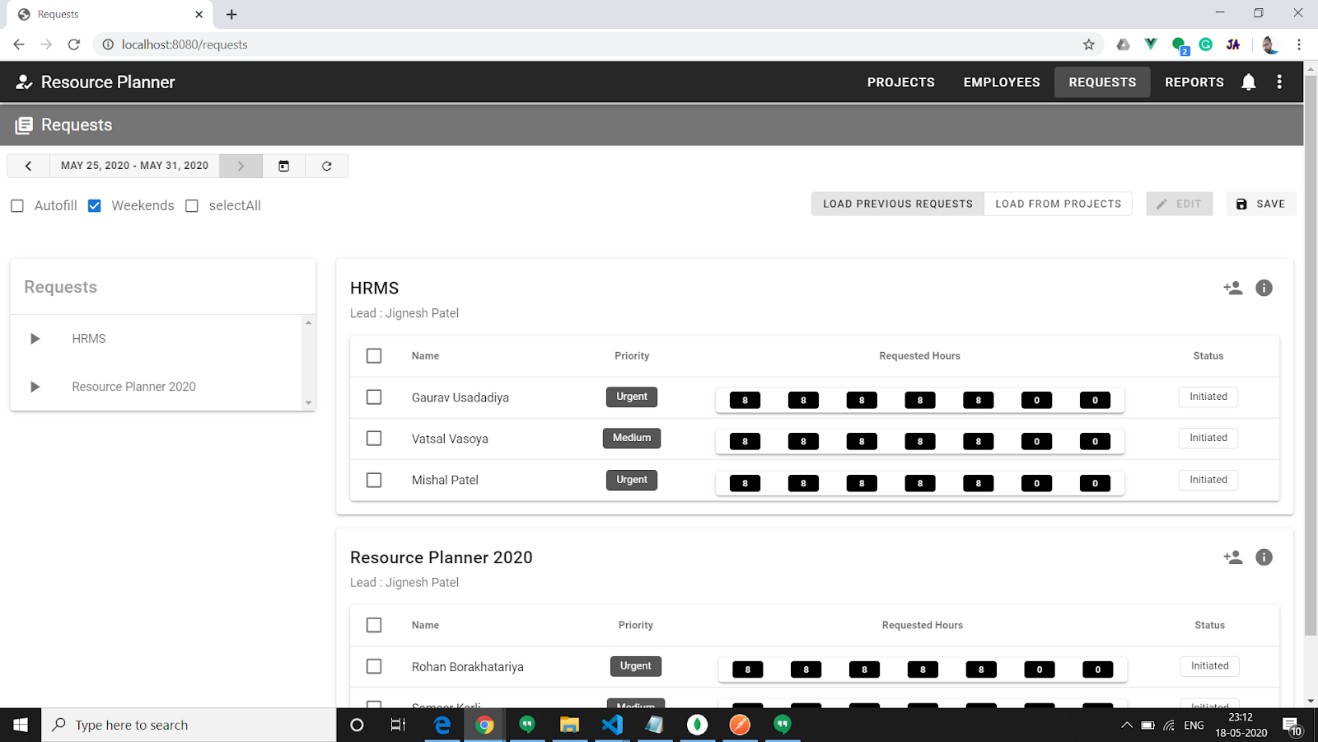


Figure 6. 21 Weekend can be chosen for request

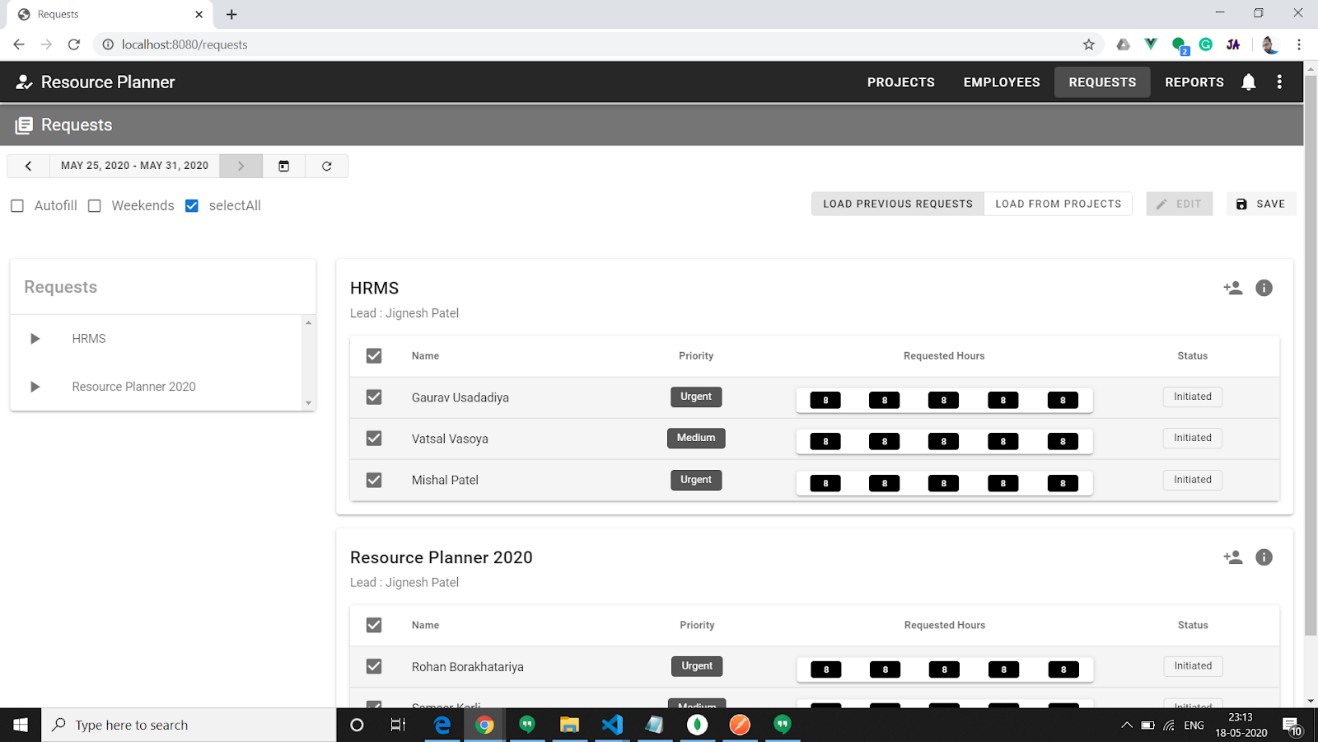


Figure 6. 22 All employee can be selected for request

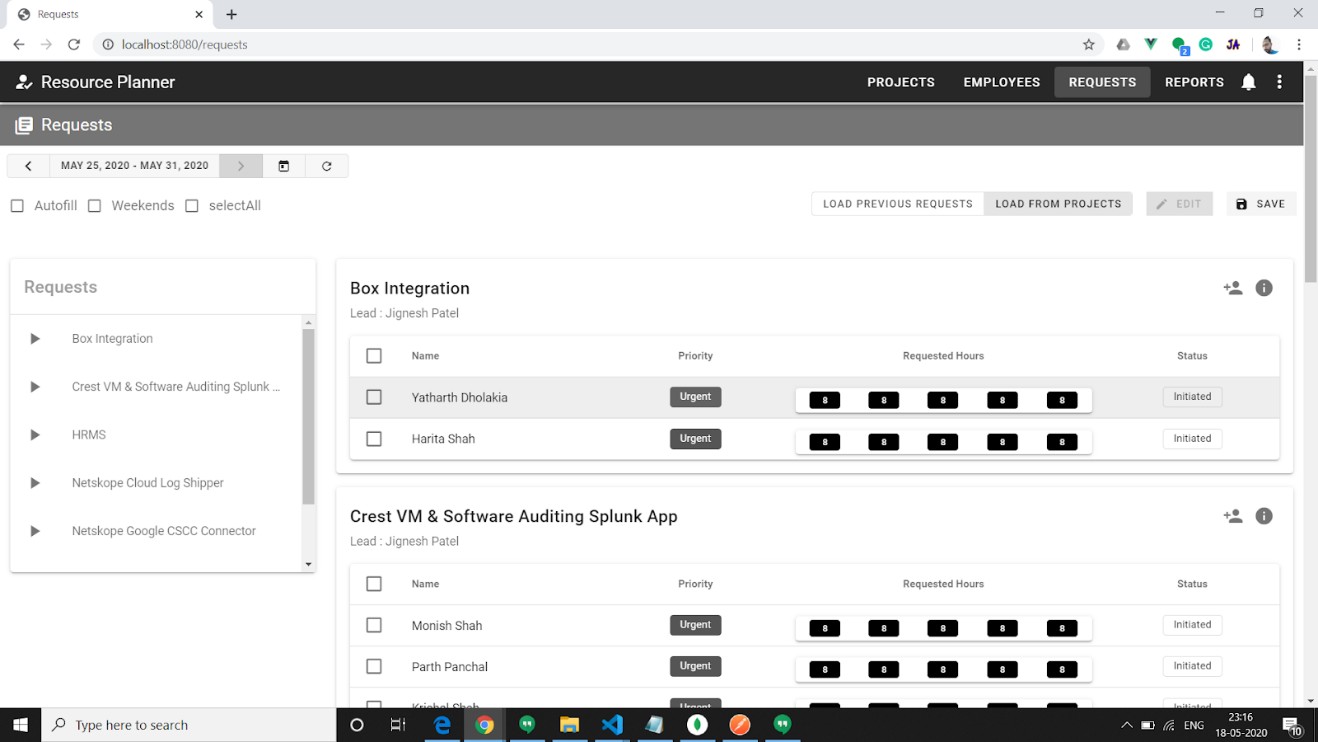


Figure 6. 23 Project can also be loaded from all projects of project manager

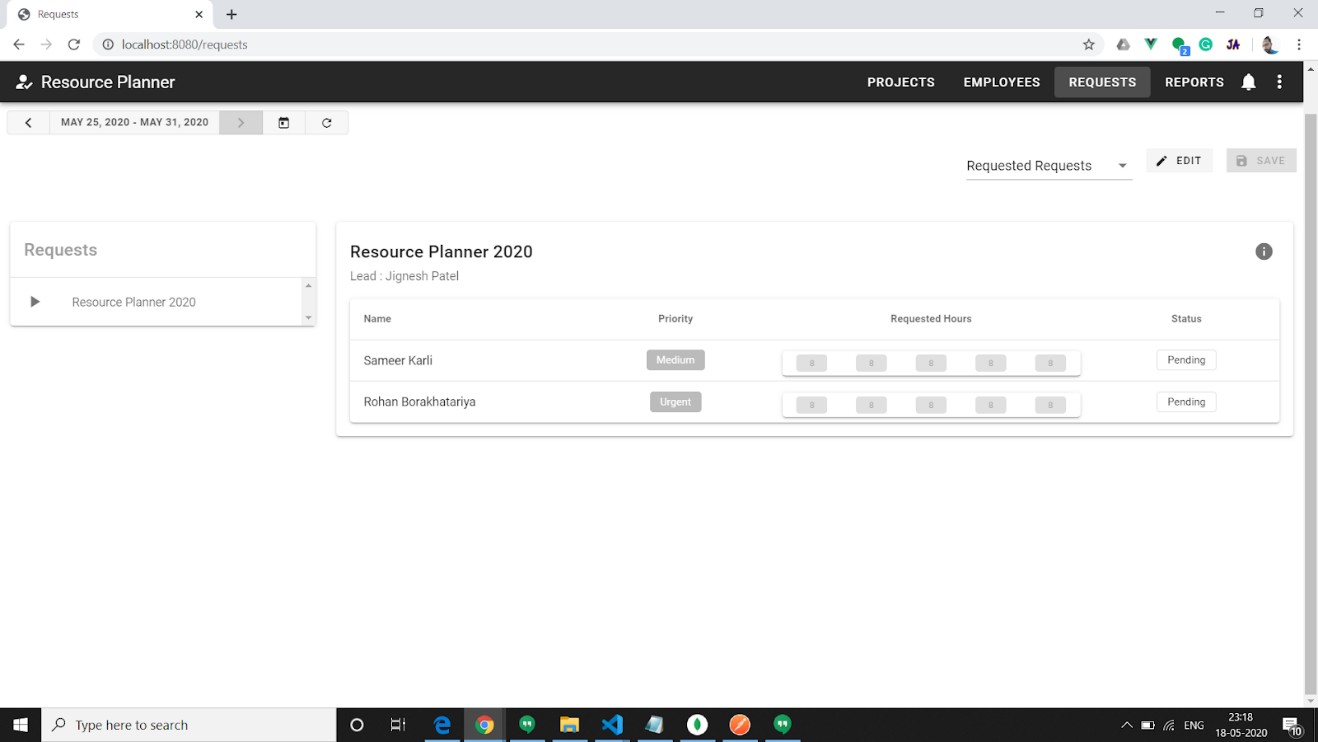


Figure 6. 24 After making request view

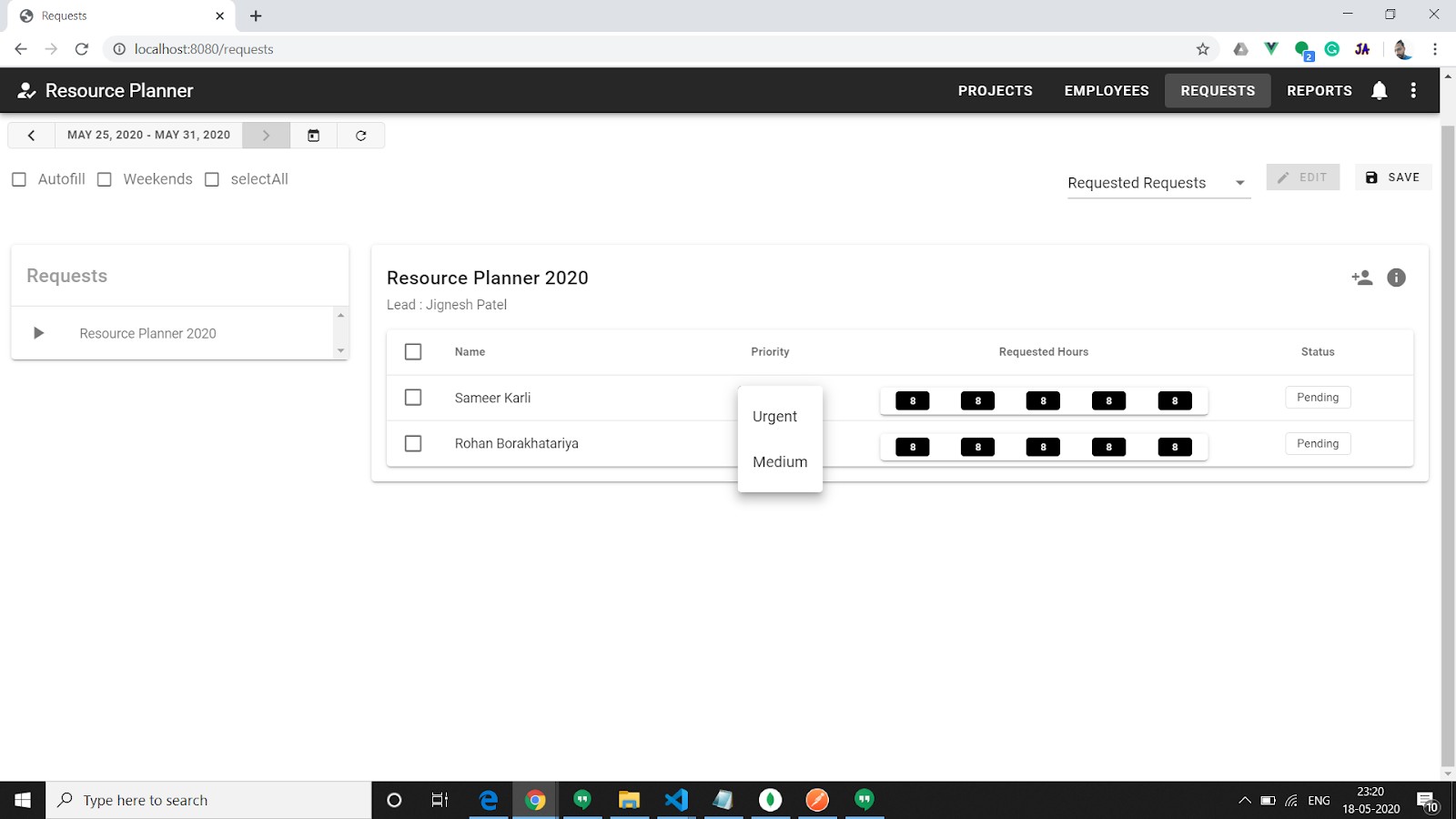


Figure 6. 25 After making request priority is edited

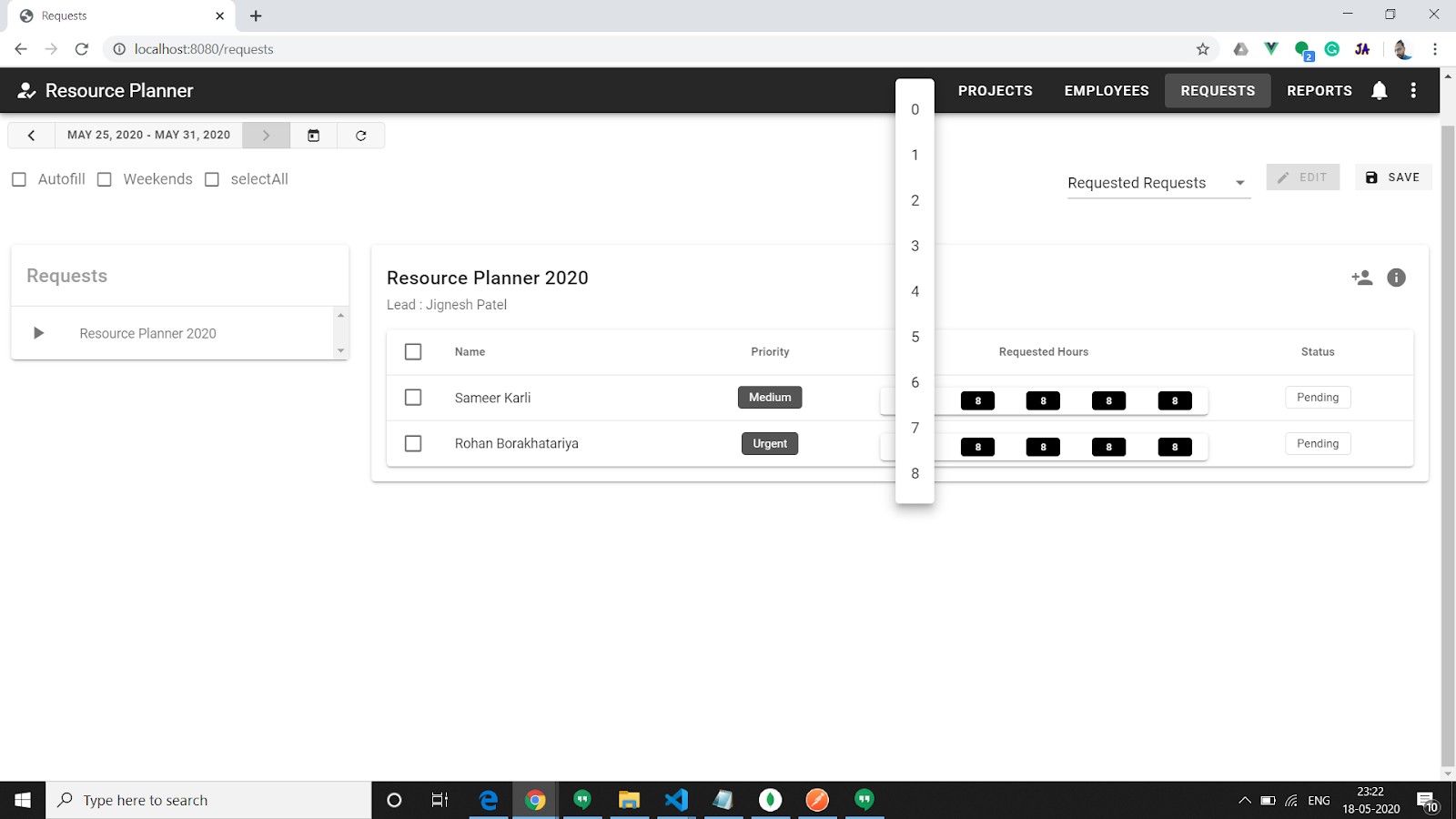


Figure 6. 26 After making request priority is edited

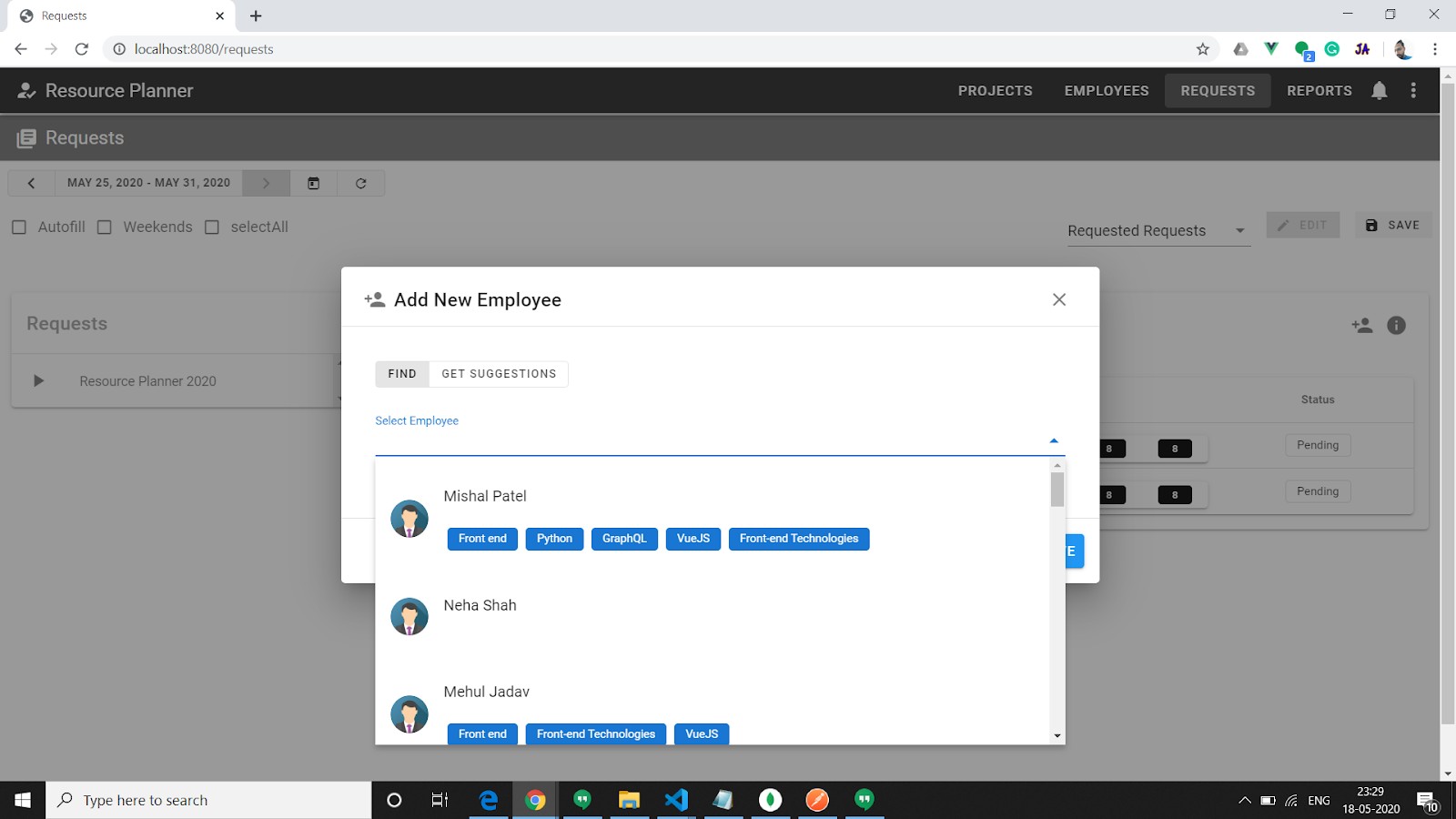


Figure 6. 27 New employees can be added in request from request card

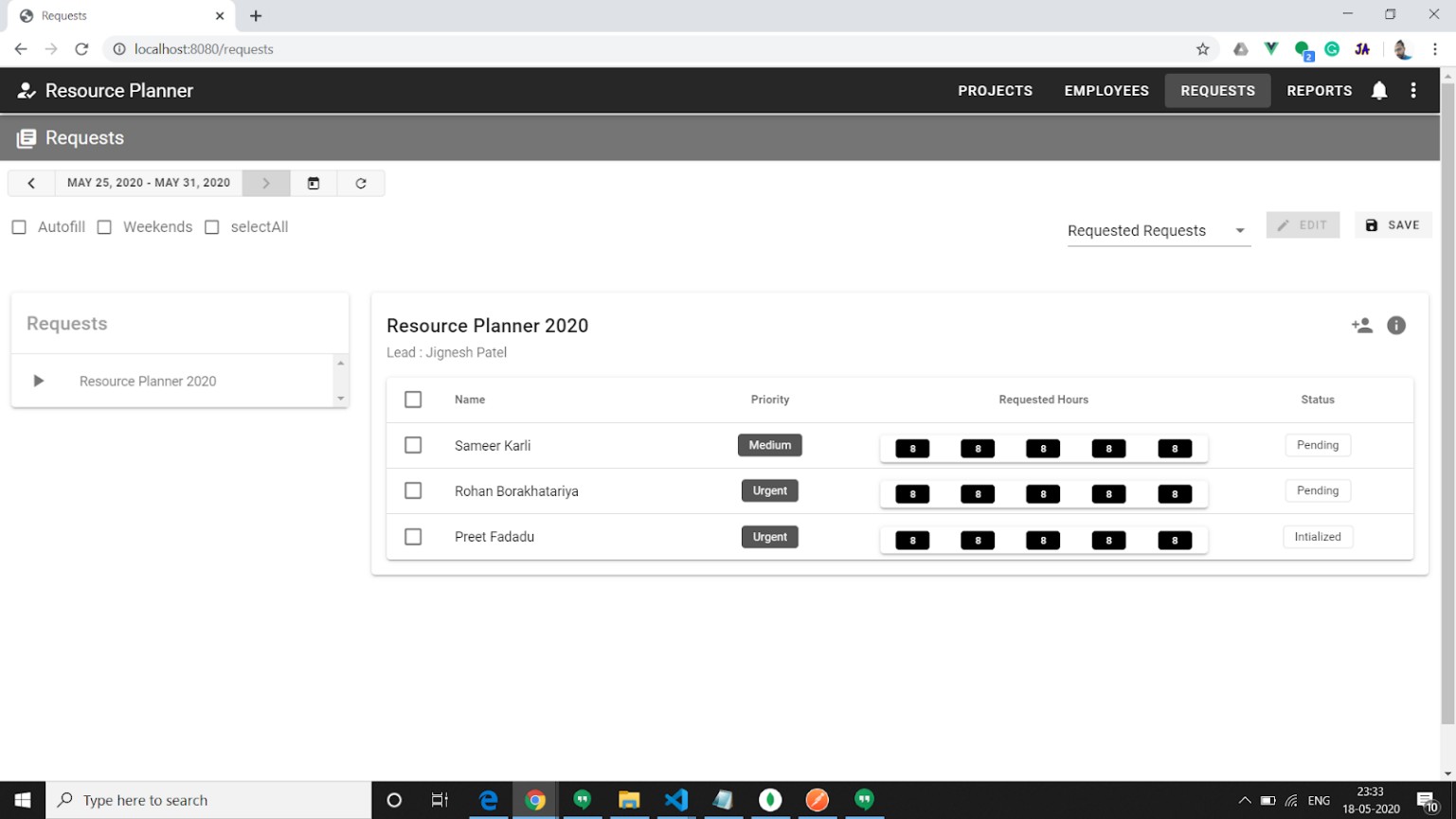


Figure 6. 28 Remaining request view

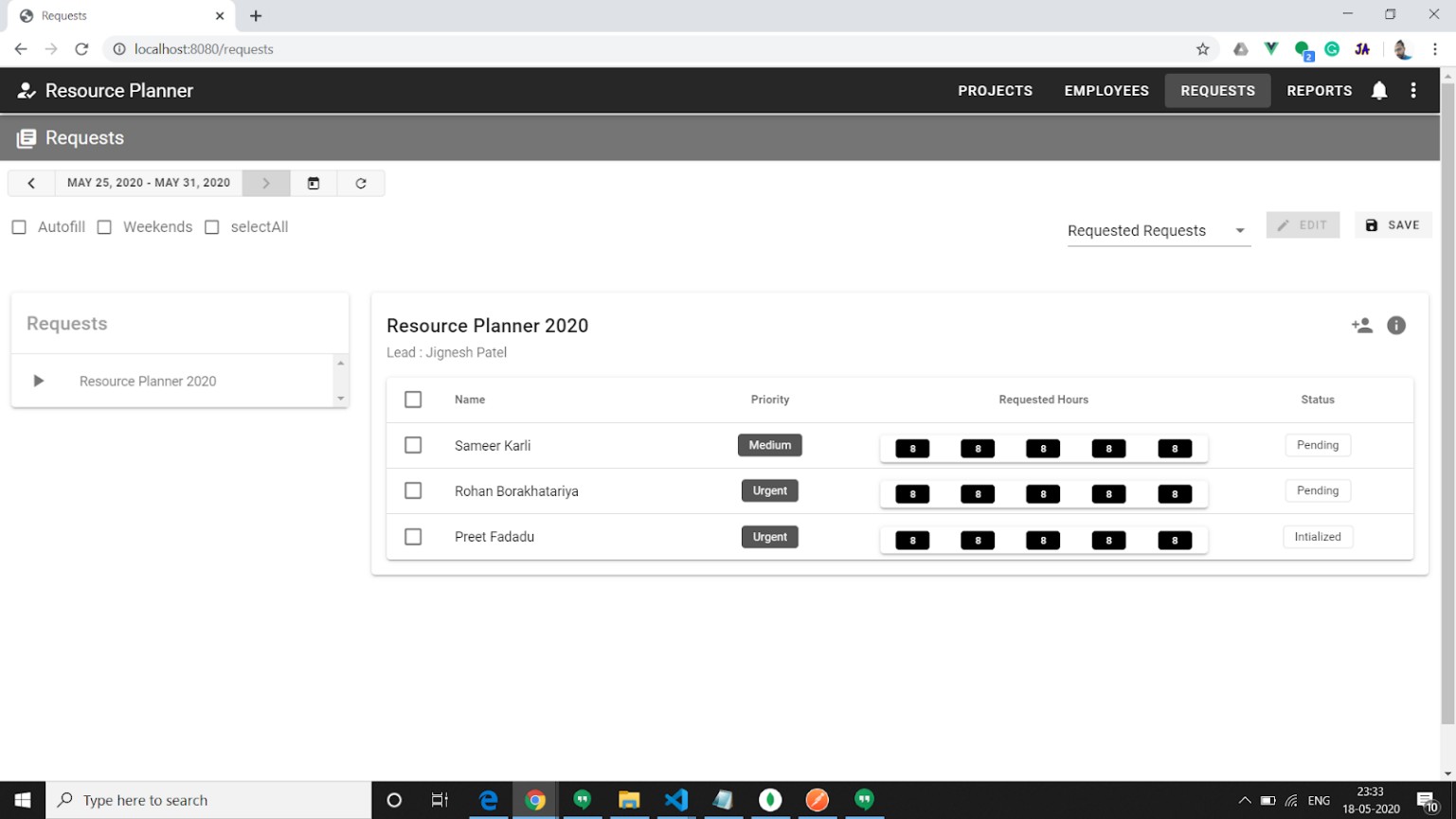


Figure 6. 29 Requested request view

### API Authentication

API authentication is done through the facility fastAPI module provides in backend application. It supports OAuth2 scheme to extract JWT token from header. Then it will rely on decoding function. If that token is correct or uninterrupted then it will allow to access API endpoint. Otherwise, it will through an error.

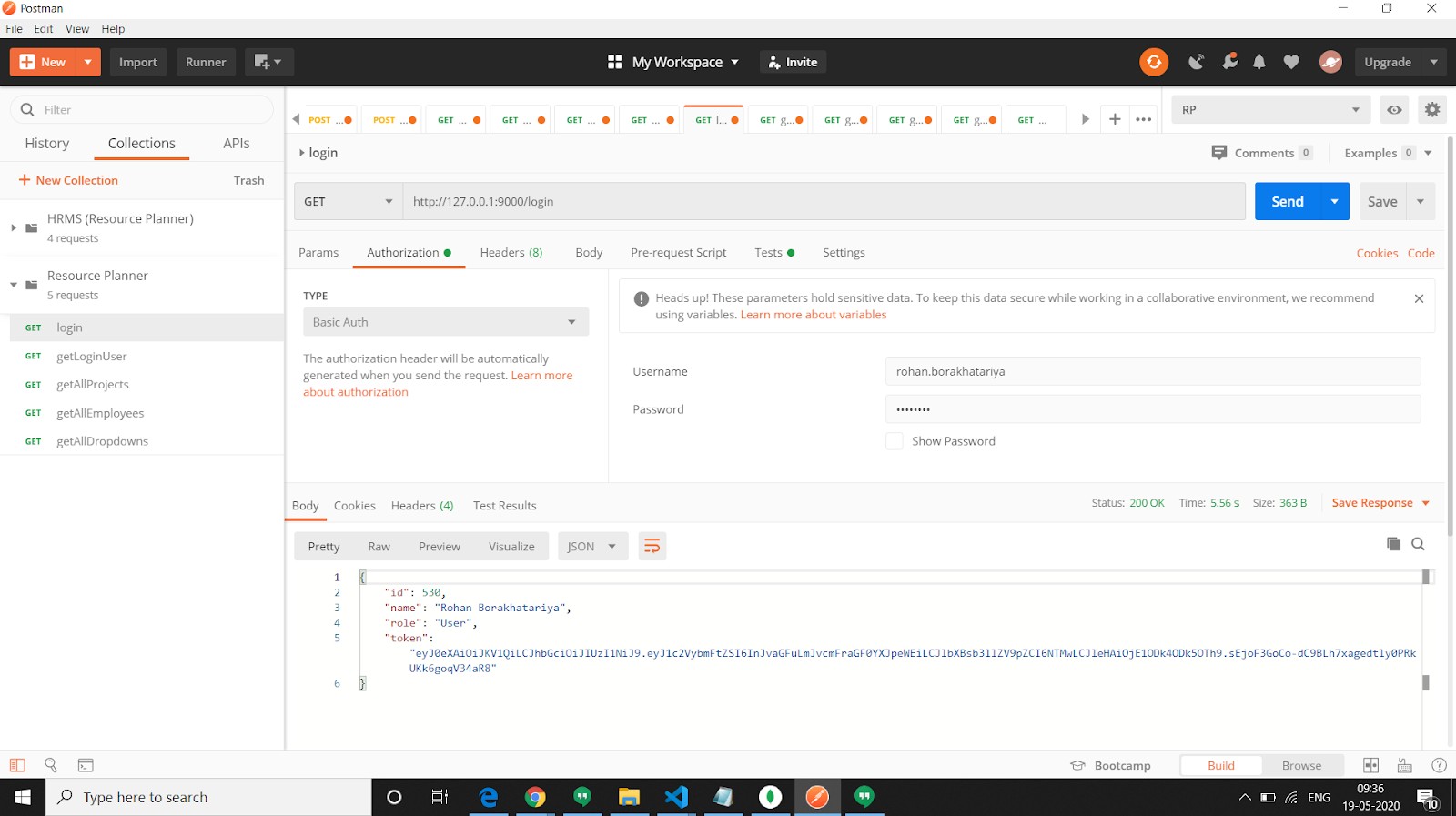


Figure 6. 30 Getting token on login API

Authentication is implemented on backend side. When user send request on login API with username and password his credentials will be checked on Active directory. If he is validated then his Employee Id and, name role and one authentication token is sent in successful response. Response code is 200 OK.

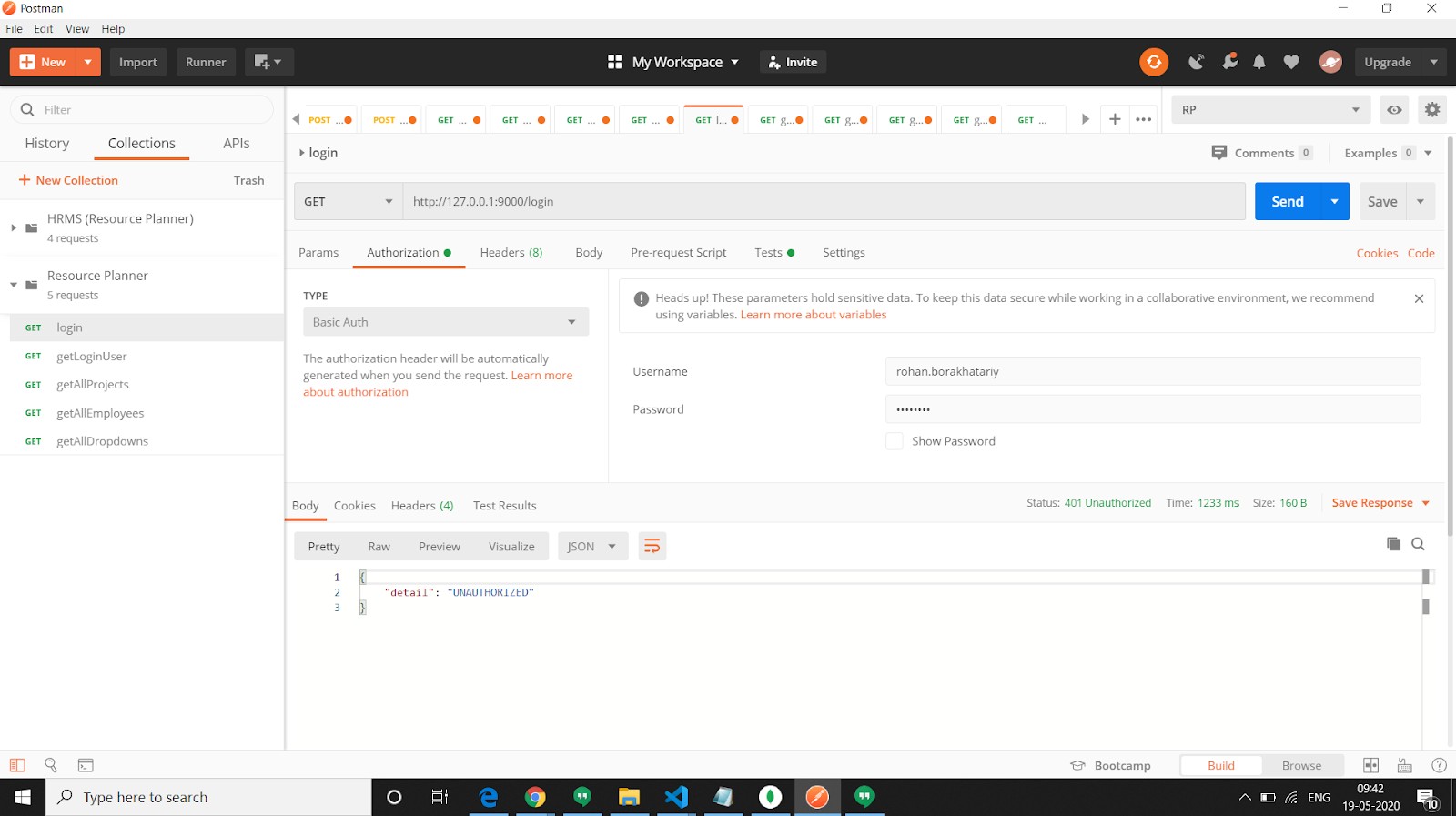


Figure 6. 31 Example of unauthorized access

If wrong credentials is added then response with status code 401 is sent. One message is also sent with “UNAUTHORIZED” message.

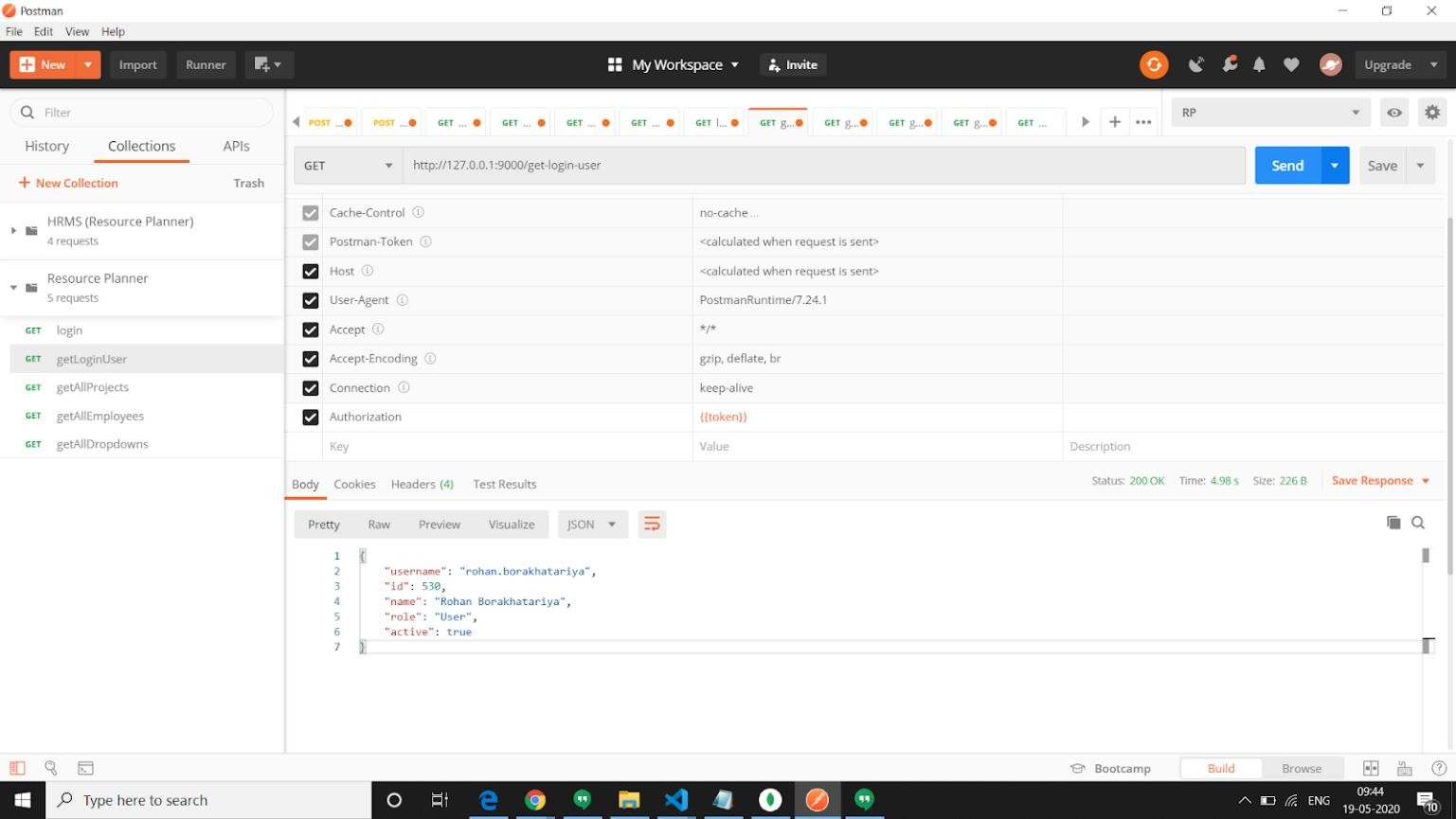


Figure 6. 32 Object that system internally uses to reach API

Now if user using that token hit on any endpoint, then that token is decoded and that employee’s details will be sent.

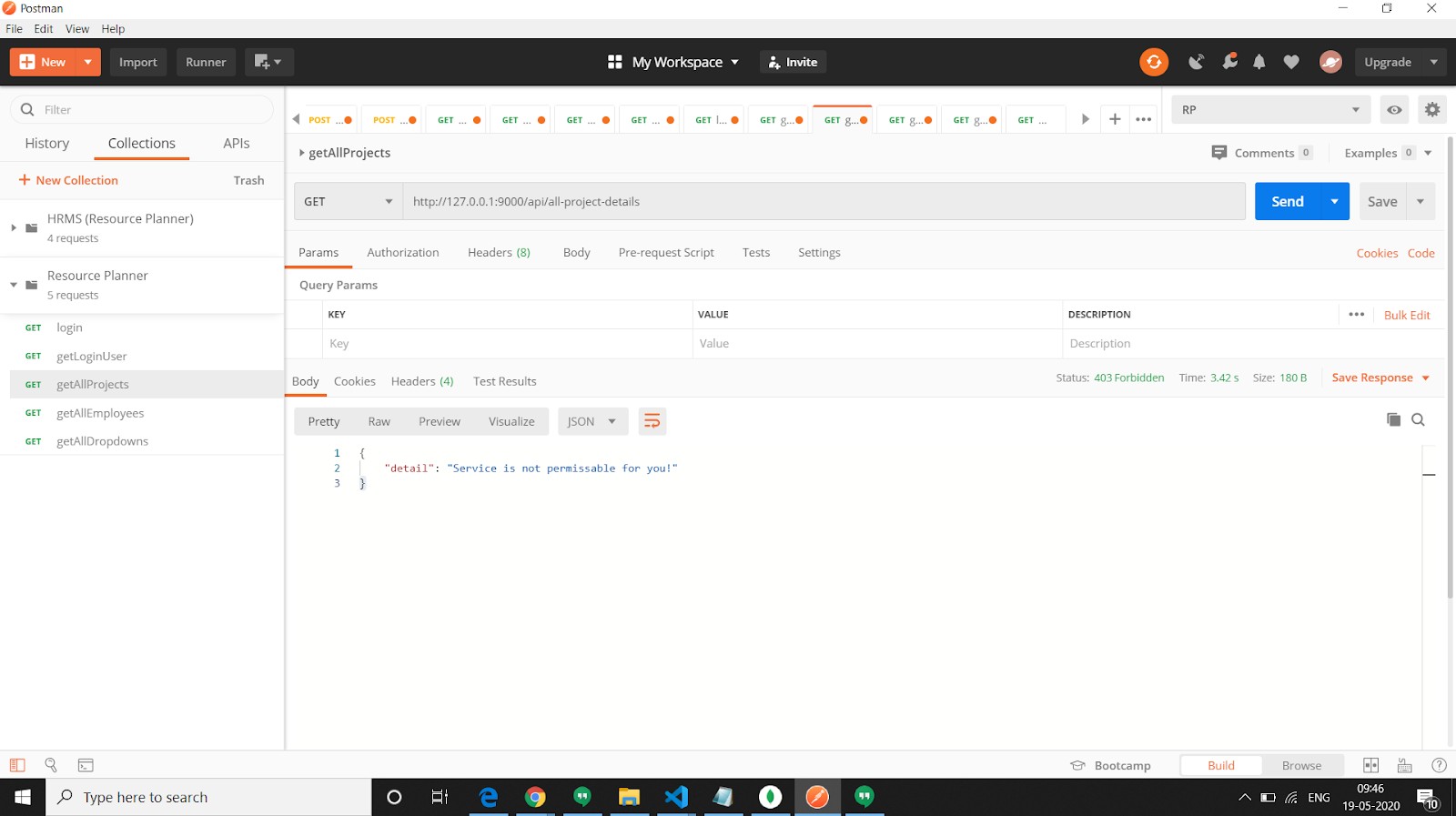


Figure 6. 33 User which is not authenticated trying to reach API

Now if user try to hit on any endpoint, then his role will be checked. If User is normal user, then response code 403 is sent. Also, message will be sent stating service is not available for that user.

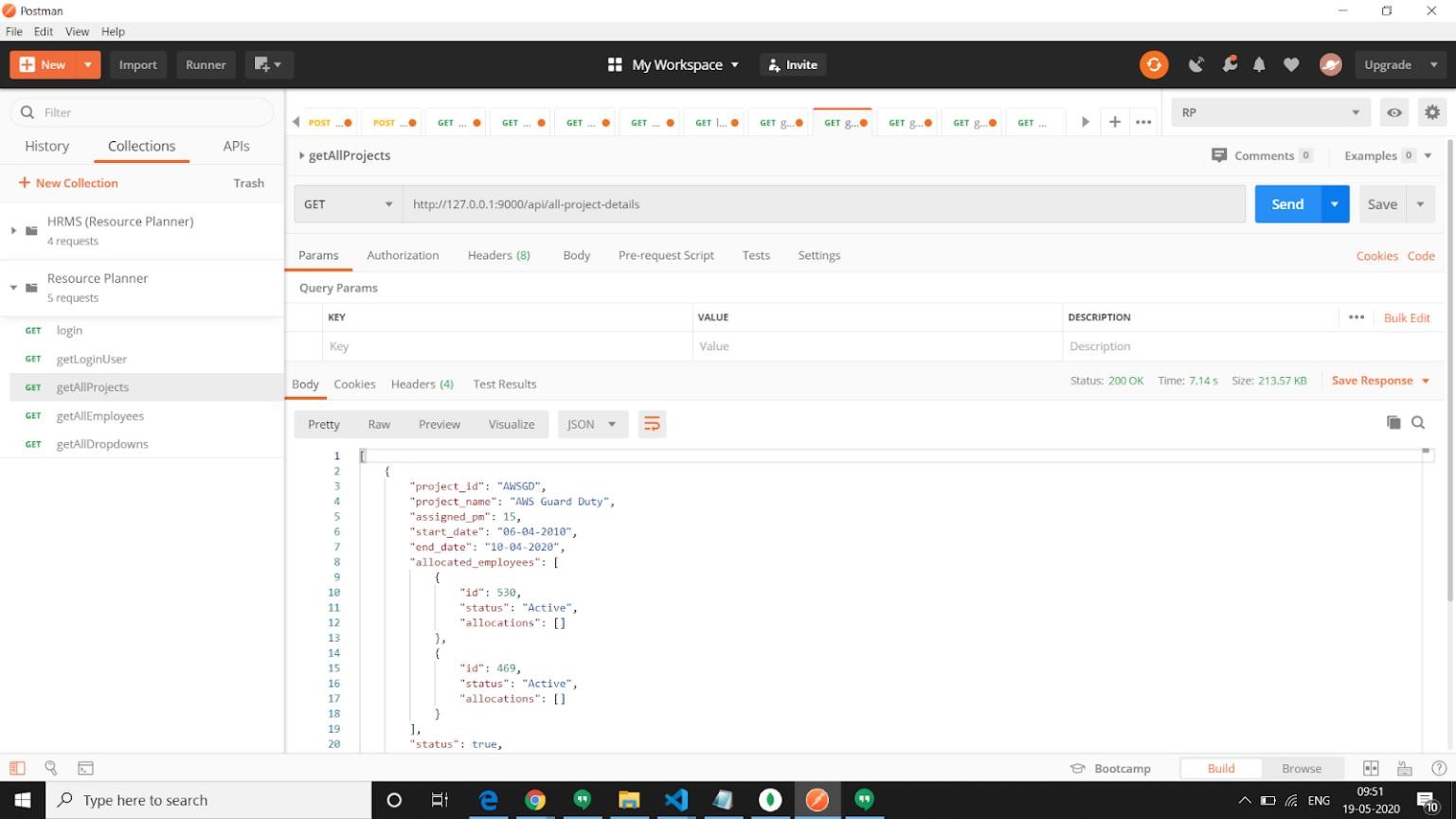


Figure 6. 34 Correct user hits API

And if user have a role of Approver or Lead then he can access that endpoint. And in response data will be send to them.