

June 01, 2023

I explored various websites to get ideas about these topics like

=>What is Middleware? Architecture of Middleware.

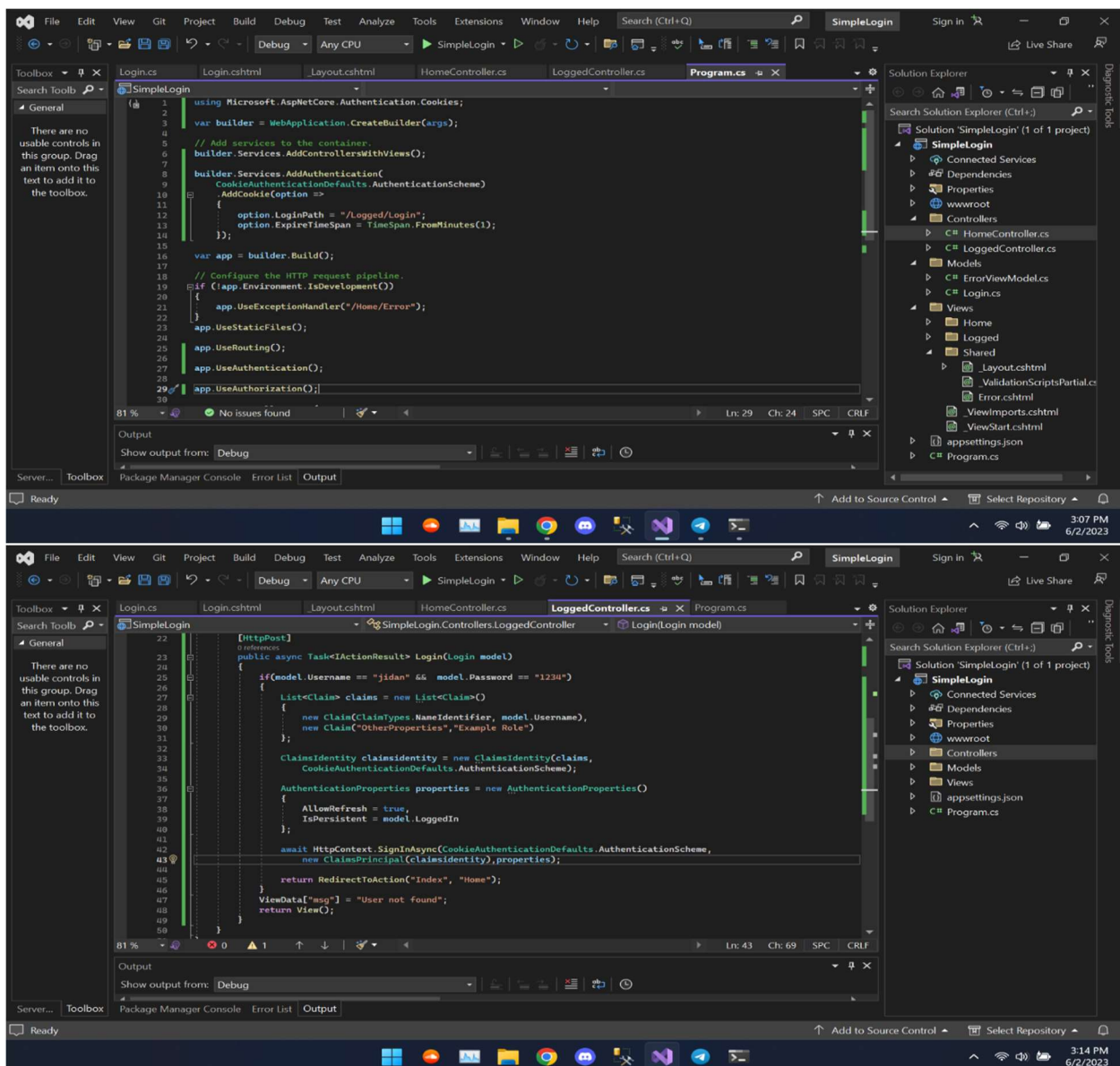
=>What is Authorization and different type of authorization strategies

=>What is Authentication and authentication factors

=>Looked for how to inject service and tried to inject service in .Net Core

June 02, 2023

Learned how to implement Authorization and Authentication using middleware and implemented those using ASP.NET Core. Tried to get idea about how to implement API in .NET Core.



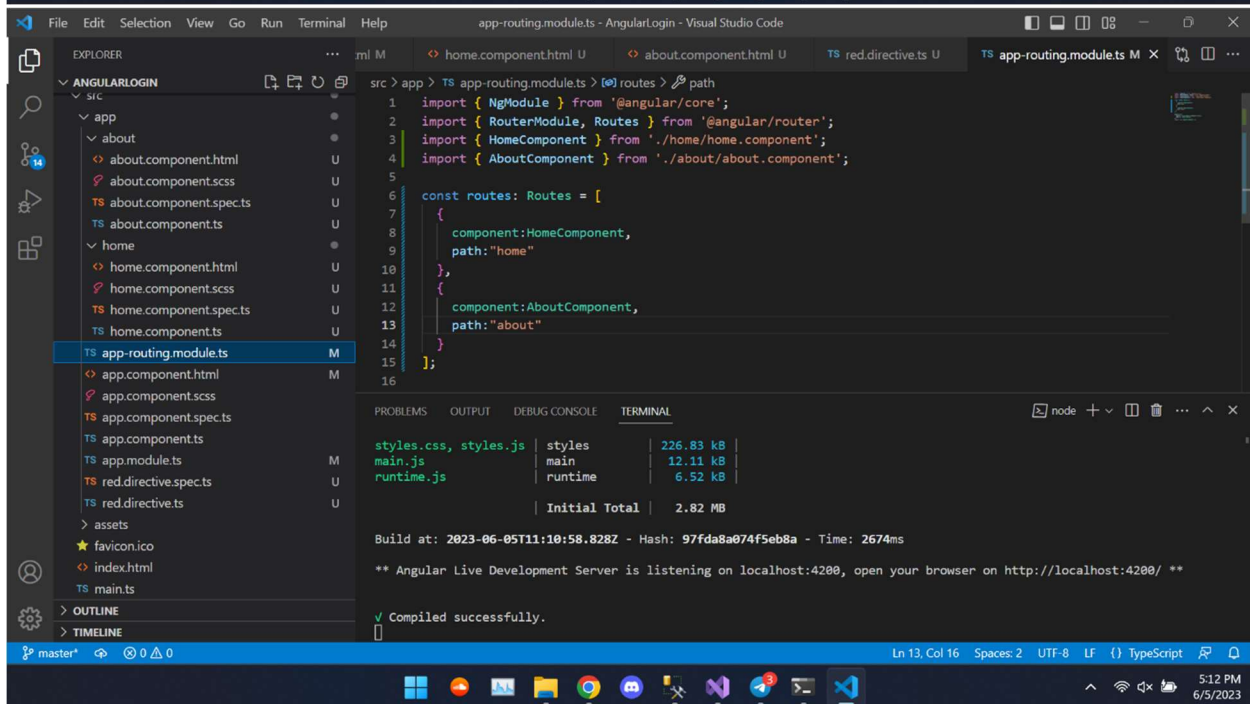
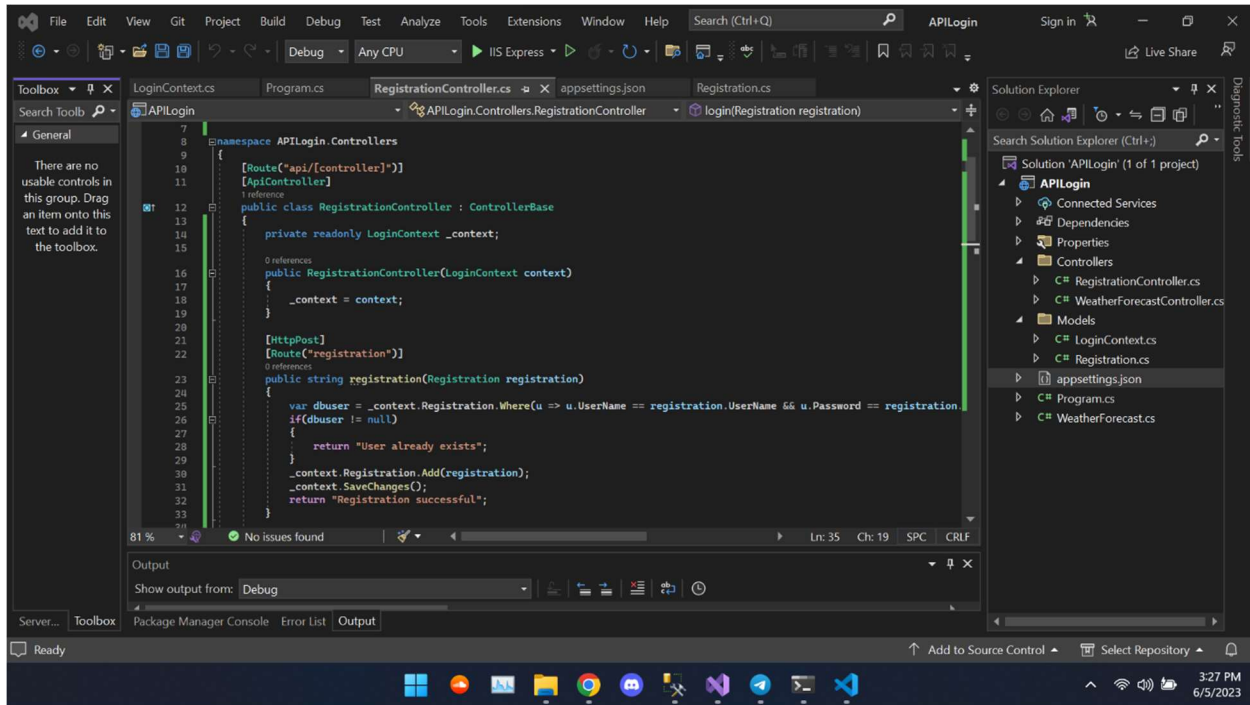
June 05, 2023

Built Web API for simple login and registration with ASP.NET Core and checked that the API's are working fine by using Swagger. Learned the basics about Angular like

=> Component in Angular, how component is created

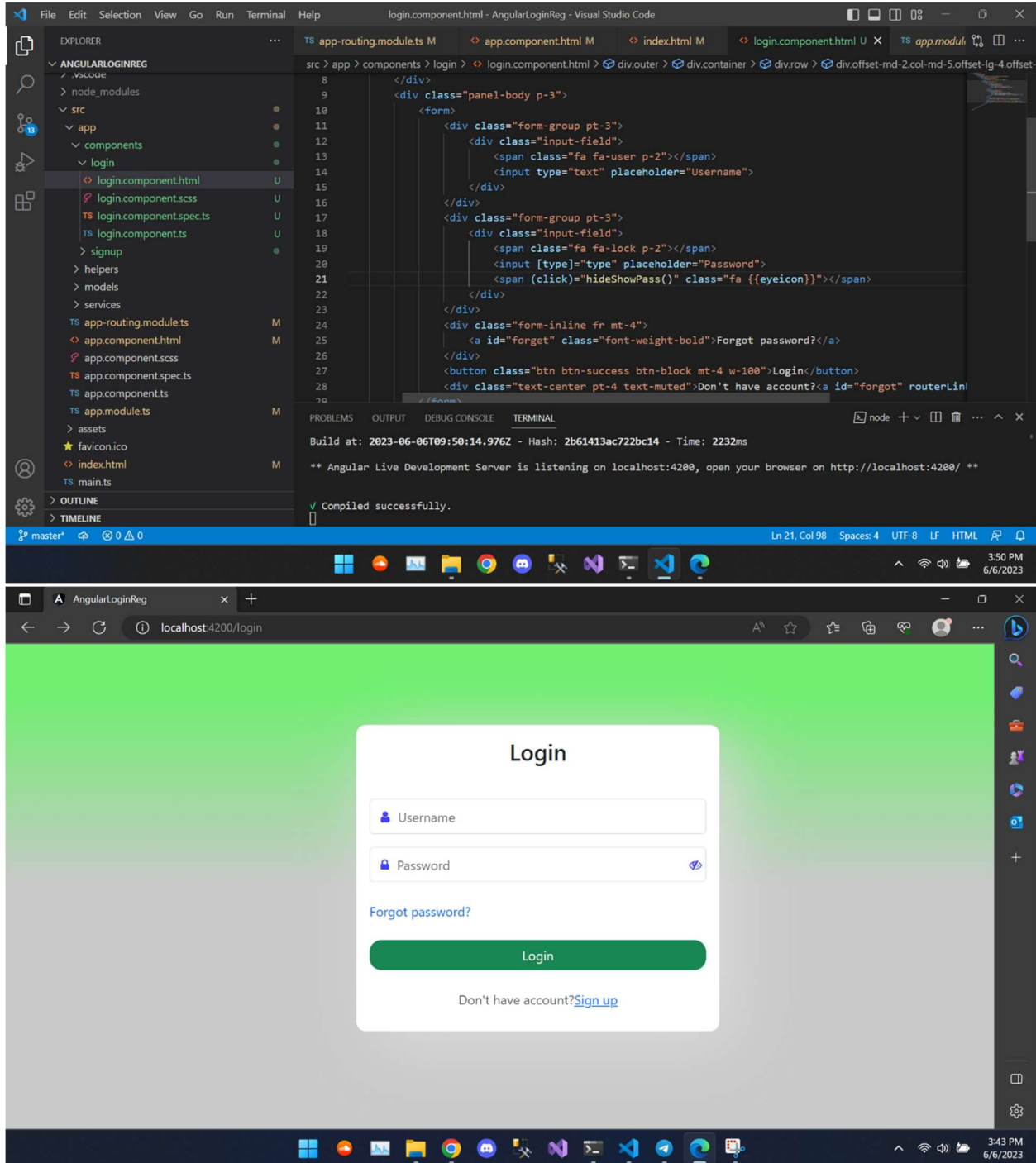
=> Routing, how routing is done in Angular

=> Directives of Angular, how to define a custom directive



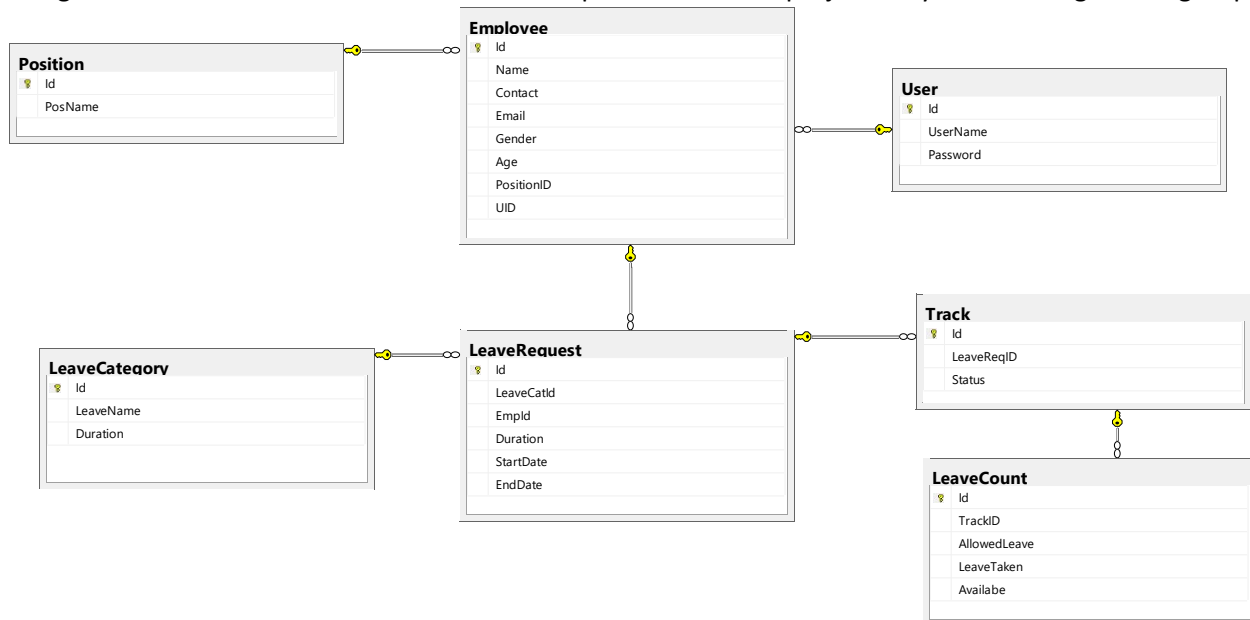
June 06, 2023

Created a simple login form in angular. Viewed the backend and frontend structures of the projects that are done in the company. Learned about the lazy loading and eager loading in angular. Implemented lazy loading to check how it works when application starts.



June 07, 2023

A project was assigned and first task was to develop the database according to requirements. Therefore, designed the database with relationships for the project by discussing in group.



June 08, 2023

Updated the database of the given project. Still there might be change needed. Therefore, tried to build some basic UI structure to get more idea about the database by creating queries.

The screenshot shows an Excel spreadsheet with a database schema and a form structure. The schema includes tables: LeaveRequestTable, CategoryTable, and PositionTable. The form structure includes fields: LeaveRequestForm, Choose Category (Select * from Category), Duration, RequestDate, StartDate, EndDate, Comment, and Status. A yellow box labeled 'SEND REQUEST' is highlighted, with an arrow pointing to an SQL query: INSERT INTO LeaveRequest (LeaveRequestID, CategoryID, EmployeeID, Duration, RequestDate, StartDate, EndDate, Comment, Status) VALUES ('123456', '789', '456', '5', '2023-06-08', '2023-06-08', '2023-06-08', '5', '2023-06-08').

Table	Field	PK	FK	Table
LeaveRequestTable	LeaveRequestID	PK		
	CategoryID	FK	CategoryTable	
	EmployeeID	FK	EmployeeTable	
	Duration			
CategoryTable	CategoryID	PK		
	CategoryName			
PositionTable	PositionID	FK		
	PositionName			

Form Structure:

- LeaveRequestForm
- Choose Category (Select * from Category)
- Duration
- RequestDate
- StartDate
- EndDate
- Comment
- Status

SQL Query:

```
INSERT INTO LeaveRequest (LeaveRequestID, CategoryID, EmployeeID, Duration, RequestDate, StartDate, EndDate, Comment, Status) VALUES ('123456', '789', '456', '5', '2023-06-08', '2023-06-08', '2023-06-08', '5', '2023-06-08')
```

June 09, 2023

Finalized the database for the assigned project. Developed queries for different actions. Build the database in MS SQL Server and executed the queries to see the results.

The screenshot displays an Excel spreadsheet titled "LMS1.xlsx" with a dashboard for a leave management system. The dashboard includes a "Request Leave Form" with input fields for LeaveType, Duration, RequestDate, StartDate, EndDate, Status, ViewAdmin, and EditDisabled. It also features a "BalanceDetails" section with "Accept" and "Decline" buttons, and a "Leave Request Table" with columns for LeaveRequestID, LeaveTypeID, EmployeeID, Duration, RequestDate, StartDate, EndDate, Comment, and Status. The spreadsheet contains several SQL queries for database operations, including a "Send Request" query, a "Join Employee" query, an "Insert Into LeaveRequest" query, and a "Select LeaveType" query. The bottom of the spreadsheet shows a "LoginForm" and a "Dashboard" tab.

Request Leave Form

LeaveType	Duration	RequestDate	StartDate	EndDate	Status	ViewAdmin	EditDisabled
Casual	10	6/9/2023	6/10/2023	6/20/2023	Accepted		

BalanceDetails

EmployeeID	LeaveTypeID	Duration	RequestDate	Status
1	Casual	10	6/9/2023	Accepted
2	Vacational	15	6/10/2023	Pending
1	Sick	15	6/9/2023	Pending

Leave Request Table

LeaveRequestID	LeaveTypeID	EmployeeID	Duration	RequestDate	StartDate	EndDate	Comment	Status
1	1	1	10	6/9/2023	6/10/2023	6/20/2023	Sick	Accepted
2	2	1	15	6/9/2023	6/10/2023	6/25/2023	Sick	Pending

Employee

LeaveType	Allowance	Used	Available
Casual	10	10	0
Vacational	10	0	10
Sick	20	0	20

SQL Queries

```
SELECT * FROM LeaveType
```

```
SELECT * FROM LeaveRequest
```

```
INSERT INTO LeaveRequest (LeaveTypeID, EmployeeID, Duration, RequestDate, StartDate, EndDate, Comment, Status) VALUES (SELECT LeaveTypeID FROM LeaveType WHERE LeaveType = 'Sick', '1', '2023-06-09', '2023-06-10', '2023-06-25', 'Sick', 'Pending')
```

```
SELECT LeaveTypeID, EmployeeID, Duration, RequestDate, StartDate, EndDate, Comment, Status FROM LeaveRequest
```

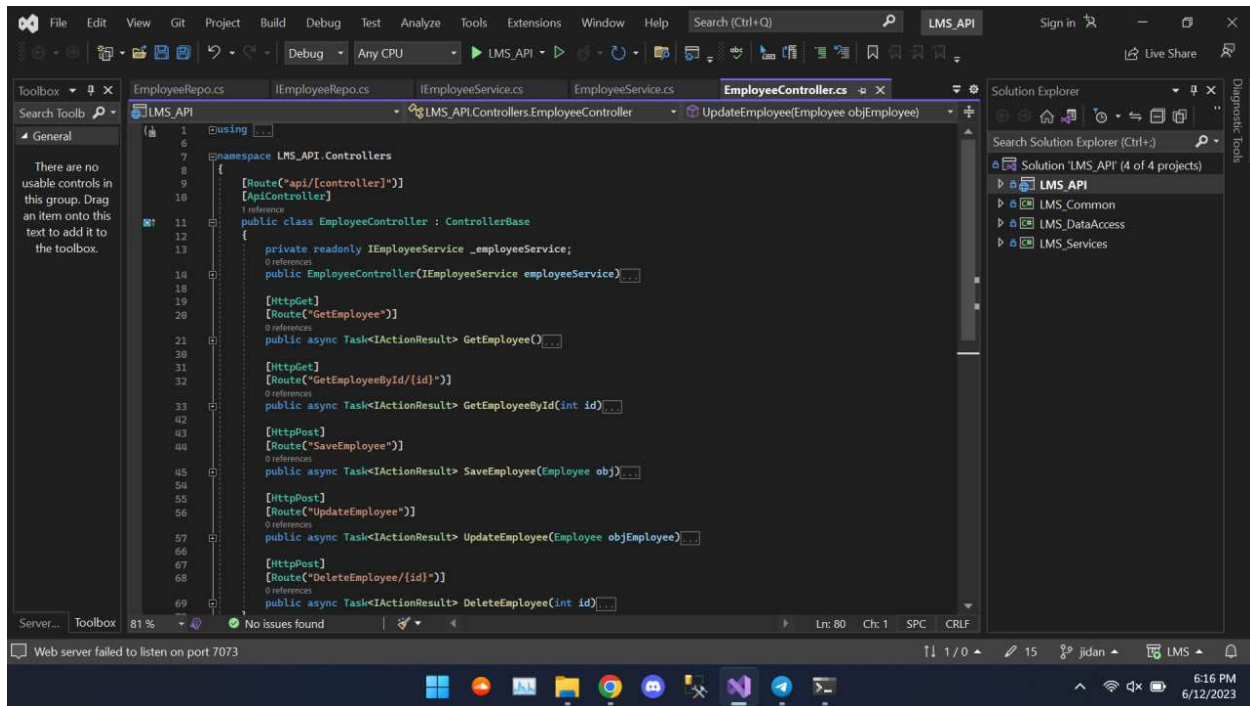
Dashboard

Ready

5:21 PM 6/9/2023

June 12,2023

Did the Employee CRUD and Login of the LMS project using ASP.Net Core Web API.

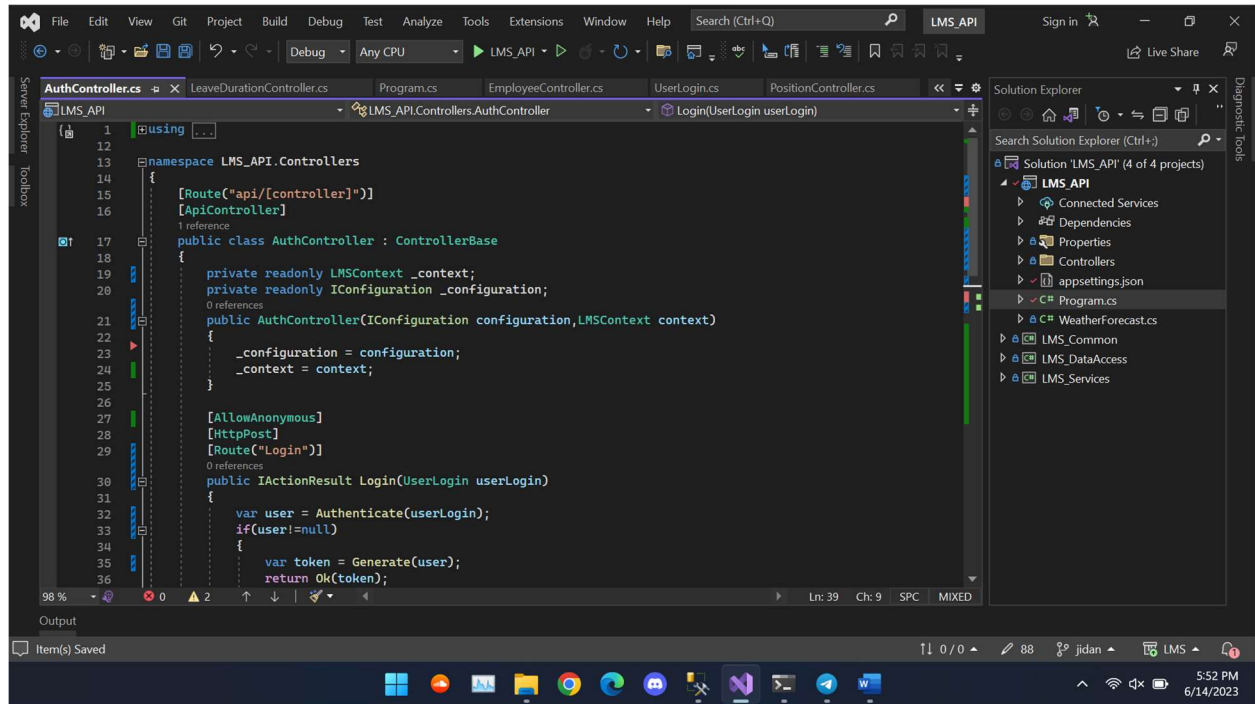


June 13,2023

Did the CRUD operation for all the entities of LMS project as well as login authentication(JWT) in .Net Core Web API.

June 14,2023

Did the authentication (JWT) and authorization in .Net Core Web API.

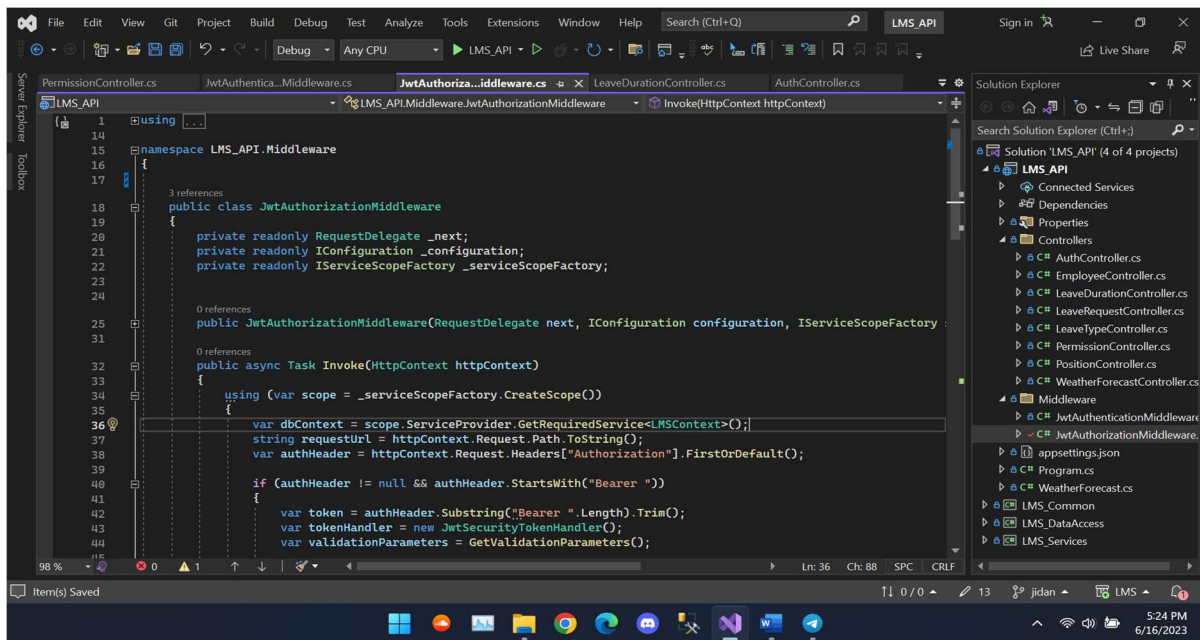


June 15,2023

Learning the JWT authentication and authorization by using custom middleware of the LMS project.

June 16,2023

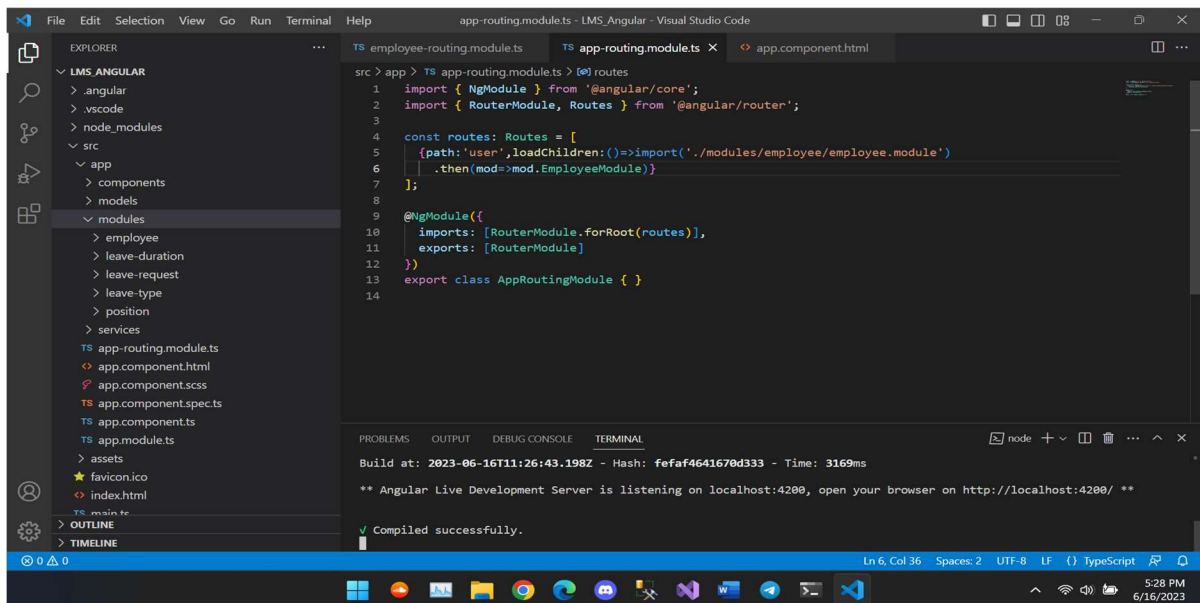
Completed the JWT authentication and authorization using custom middleware of the LMS project in .Net Core Web API. Started building the frontend of the LMS project in Angular by defining the components, modules and services.



The screenshot displays the Visual Studio Code editor with the 'JwtAuthorizationMiddleware.cs' file open. The code defines a custom middleware class for JWT authentication and authorization. The class is named 'JwtAuthorizationMiddleware' and is located within the 'LMS_API.Middleware' namespace. It implements the 'IMiddleware' interface. The code includes the following logic:

- Private fields: `readonly RequestDelegate _next;`, `readonly IConfiguration _configuration;`, and `readonly IServiceScopeFactory _serviceScopeFactory;`
- Constructor: `JwtAuthorizationMiddleware(RequestDelegate next, IConfiguration configuration, IServiceScopeFactory serviceScopeFactory)`
- `Invoke` method: An asynchronous method that takes an `HttpContext` as a parameter. It creates a service scope, retrieves the database context, and extracts the authorization header from the request. It then validates the token and sets the validation parameters.

The Solution Explorer on the right shows the project structure, including the 'Middleware' folder where the 'JwtAuthorizationMiddleware.cs' file is located.



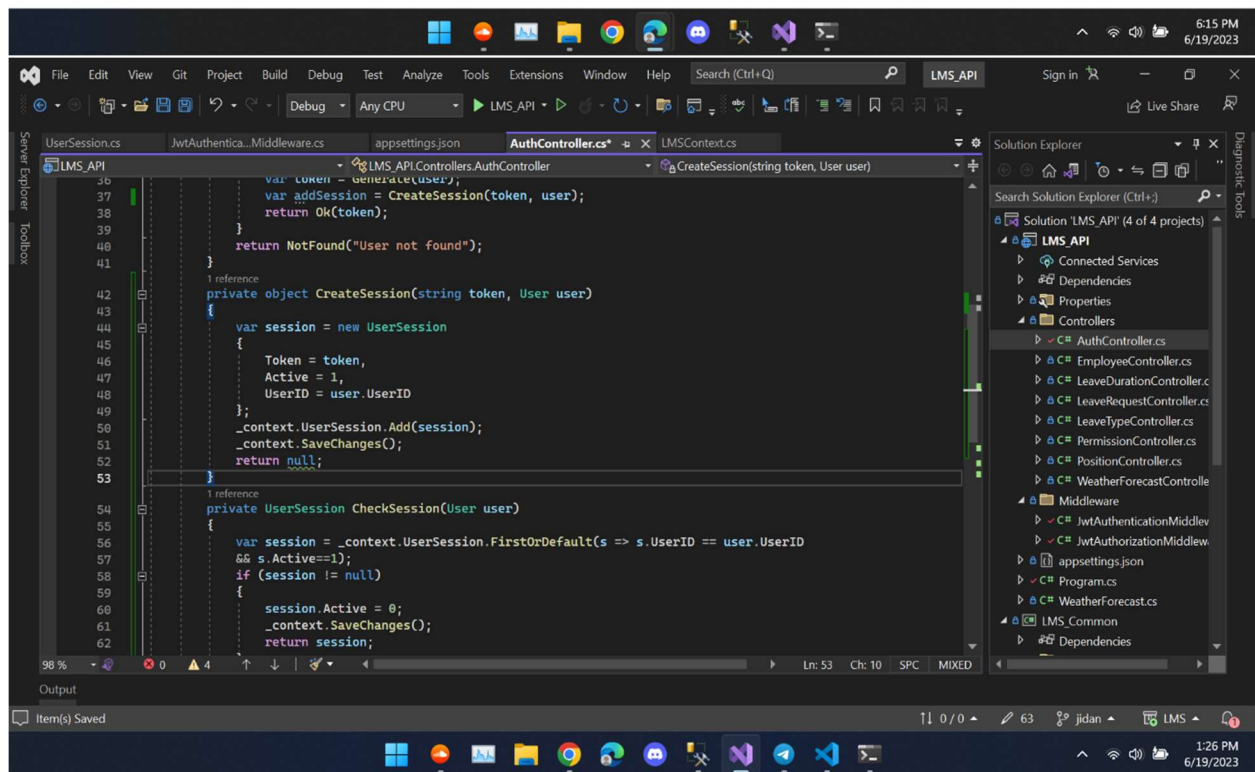
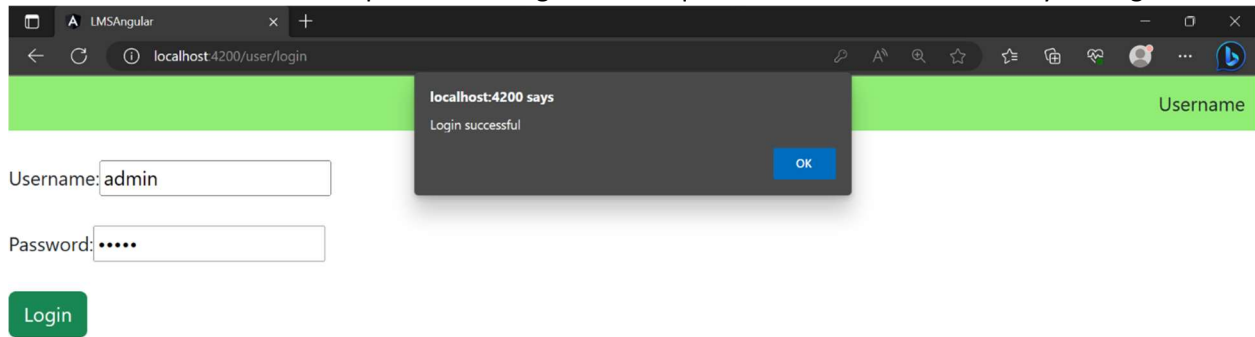
The screenshot displays the Visual Studio Code editor with the 'app-routing.module.ts' file open. The code defines the routing module for the Angular application. The module is named 'AppRoutingModule' and is located within the 'app-routing' namespace. The code includes the following logic:

- Imports: `NgModule` from '@angular/core' and `RouterModule, Routes` from '@angular/router'.
- Routes: A constant array of routes, including a route for the 'user' path that loads the 'EmployeeModule'.
- Module configuration: The `@NgModule` decorator is used to configure the module, including the imports and exports.

The terminal output at the bottom shows the build process and the message: "Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/".

June 19,2023

Created session for user. Implemented Login and RequestLeave in the frontend by calling the API.



JIDAN.LeaveManagement - dbo.UserSession - Microsoft SQL Server Management Studio

File Edit View Project Debug Query Designer Tools Window Help

Object Explorer

Connect - JIDAN (SQL Server 12.0.2000 - JIDANV)

- Databases
 - System Databases
 - CareerPortal
 - CoreLoginReg
 - LeaveManagement
 - Database Diagrams
 - Tables
 - System Tables
 - FileTables
 - dbo.Employee
 - dbo.LeaveDuration
 - dbo.LeaveRequest
 - dbo.LeaveType
 - dbo.Permission
 - dbo.Position
 - dbo.User
 - dbo.UserSession
 - Views
 - Synonyms
 - Programmability
 - Service Broker
 - Storage
 - Security
 - Product

JIDAN.LeaveManagement - dbo.UserSession

SessionID	Token	UserID	Active
1	eyJhbGciOiJIUzI1NiIsInR5cCI6I	2	0
2	eyJhbGciOiJIUzI1NiIsInR5cCI6I	2	1
NULL	NULL	NULL	NULL

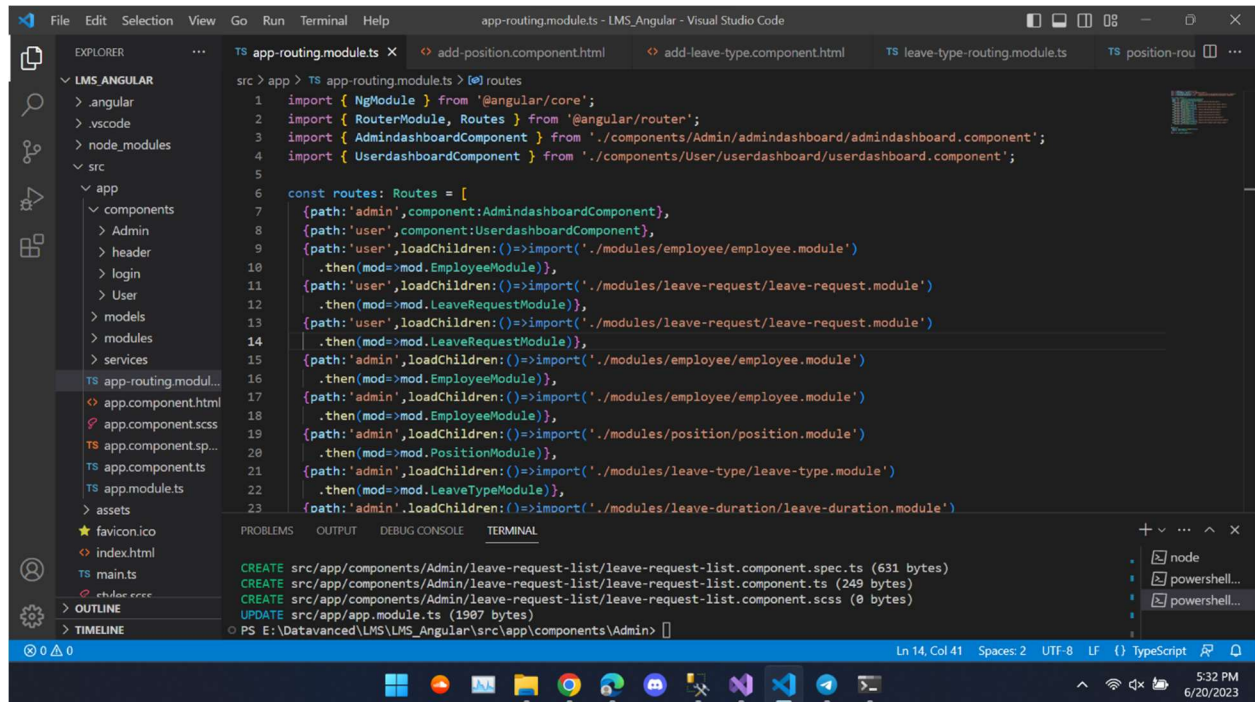
1 of 2 | Cell is Read Only.

Ready

1:19 PM 6/19/2023

June 20,2023

Modified some code of authorization in the backend. Did most of the routing for different components in the frontend.



```
src > app > TS app-routing.module.ts > routes
1 import { NgModule } from '@angular/core';
2 import { RouterModule, Routes } from '@angular/router';
3 import { AdminDashboardComponent } from './components/Admin/admindashboard/admindashboard.component';
4 import { UserDashboardComponent } from './components/User/userdashboard/userdashboard.component';
5
6 const routes: Routes = [
7   {path: 'admin', component: AdminDashboardComponent},
8   {path: 'user', component: UserDashboardComponent},
9   {path: 'user', loadChildren: () => import('./modules/employee/employee.module')
10    .then(mod => mod.EmployeeModule)},
11   {path: 'user', loadChildren: () => import('./modules/leave-request/leave-request.module')
12    .then(mod => mod.LeaveRequestModule)},
13   {path: 'user', loadChildren: () => import('./modules/leave-request/leave-request.module')
14    .then(mod => mod.LeaveRequestModule)},
15   {path: 'admin', loadChildren: () => import('./modules/employee/employee.module')
16    .then(mod => mod.EmployeeModule)},
17   {path: 'admin', loadChildren: () => import('./modules/employee/employee.module')
18    .then(mod => mod.EmployeeModule)},
19   {path: 'admin', loadChildren: () => import('./modules/position/position.module')
20    .then(mod => mod.PositionModule)},
21   {path: 'admin', loadChildren: () => import('./modules/leave-type/leave-type.module')
22    .then(mod => mod.LeaveTypeModule)},
23   {path: 'admin', loadChildren: () => import('./modules/leave-duration/leave-duration.module')}
```

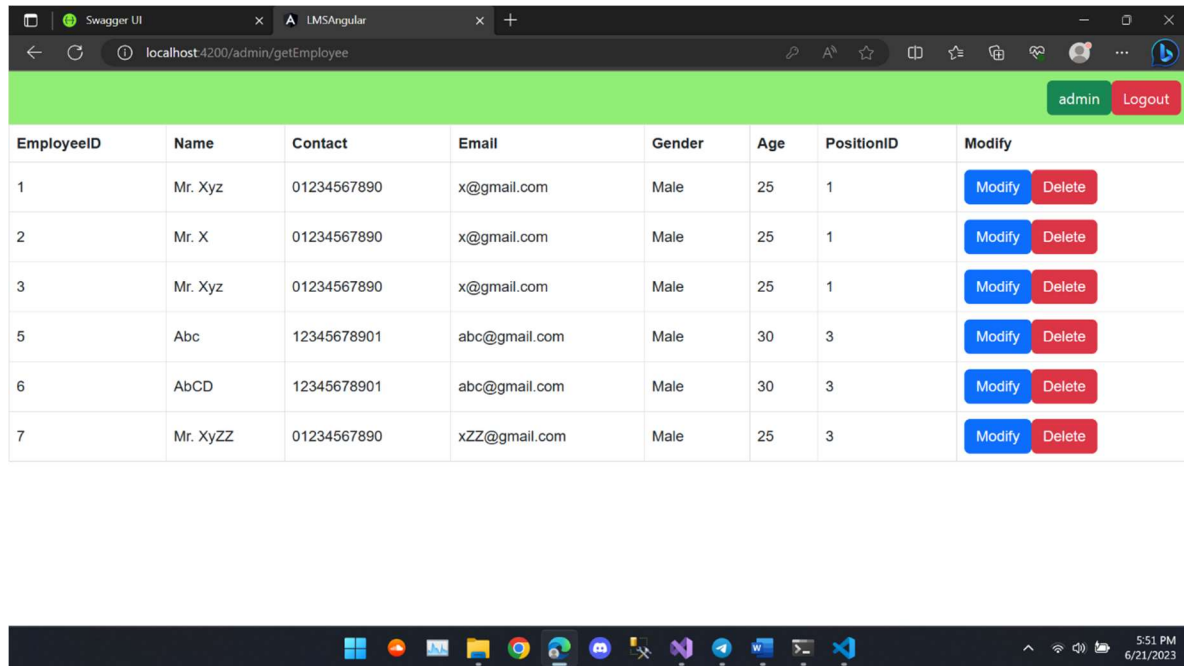
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
CREATE src/app/components/Admin/leave-request-list/leave-request-list.component.spec.ts (631 bytes)
CREATE src/app/components/Admin/leave-request-list/leave-request-list.component.ts (249 bytes)
CREATE src/app/components/Admin/leave-request-list/leave-request-list.component.scss (0 bytes)
UPDATE src/app/app.module.ts (1987 bytes)
PS E:\Datavanced\LMS\LMS_Angular\src\app\components\Admin>
```

Ln 14, Col 41 Spaces: 2 UTF-8 LF TypeScript 5:32 PM 6/20/2023

June 21,2023

Modified login by using interceptor. Retrieved data from database for some components in the frontend by calling API.



EmployeeID	Name	Contact	Email	Gender	Age	PositionID	Modify
1	Mr. Xyz	01234567890	x@gmail.com	Male	25	1	<button>Modify</button> <button>Delete</button>
2	Mr. X	01234567890	x@gmail.com	Male	25	1	<button>Modify</button> <button>Delete</button>
3	Mr. Xyz	01234567890	x@gmail.com	Male	25	1	<button>Modify</button> <button>Delete</button>
5	Abc	12345678901	abc@gmail.com	Male	30	3	<button>Modify</button> <button>Delete</button>
6	AbCD	12345678901	abc@gmail.com	Male	30	3	<button>Modify</button> <button>Delete</button>
7	Mr. XyZZ	01234567890	xZZ@gmail.com	Male	25	3	<button>Modify</button> <button>Delete</button>

June 22,2023

Implement AUTH Guard in angular for protecting the routing of the LMS project. Did the CRUD for employee by calling the API's in Angular.

June 23,2023

Learned how to pass data from parent to child and child to parent component by using `@Input()`, `@Output` decorator and the EventEmitter. Implemented some components to retrieve data from database using API as well as implemented modals in angular

abc Logout

Apply New Leave

LeaveRequestID	LeaveType Name	Employee Name	Duration	RequestDate	StartDate	EndDate	Comment	Status
5	Vacational	Abc	10	2023-06-19T00:00:00	2023-06-20T00:00:00	2023-06-30T00:00:00	asd	pending

LeaveTypeID:

EmployeeID:

Duration:

RequestDate:

StartDate:

EndDate:

Comment:

Status:

July 06,2023

Learned how to use bootstrap modals, implemented @input () decorator to pass data from parent to child component. Implemented those in the User Dashboard of the LMS project. Also, got to know about pipes in angular.

abc Logout

Request for Leave

LeaveType Name	Employee Name
Vacational	Abc
Sick	Abc
Casual	Abc
Casual	Abc

Edit Leave Request Form

Leave Type:

EmployeeID:

Duration:

RequestDate:

StartDate:

EndDate:

Comment:

Status:

Close Save

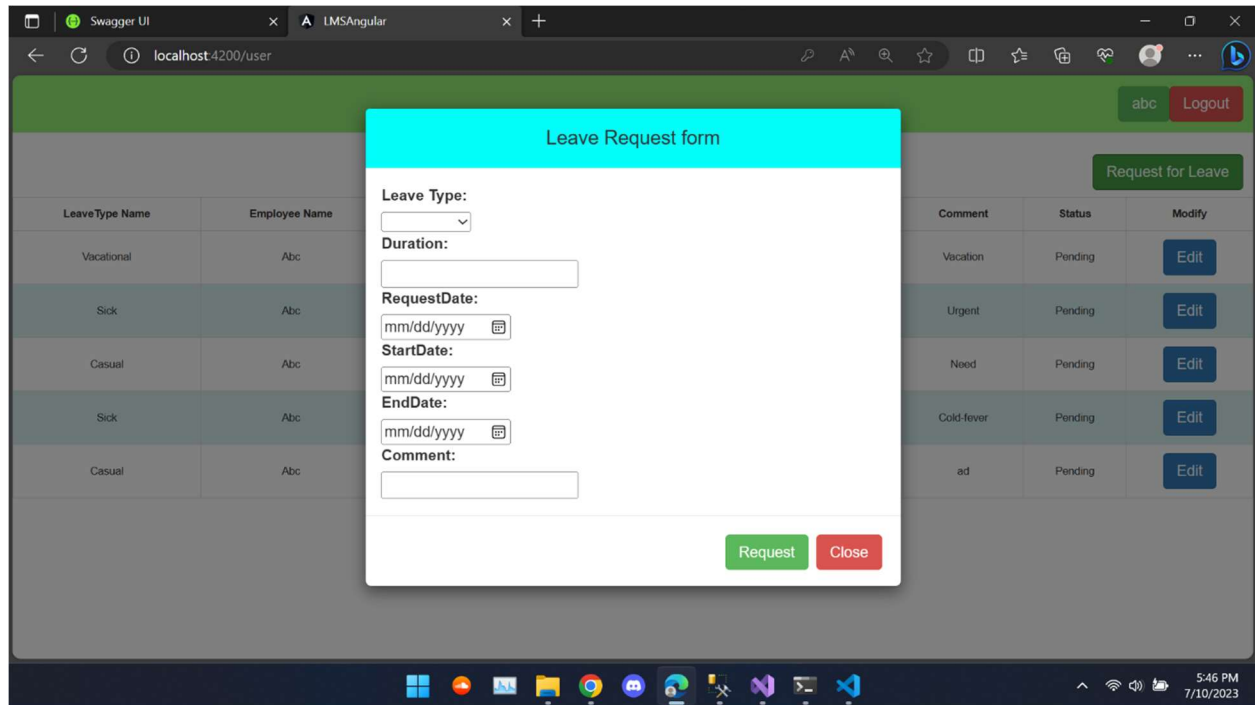
Date	Comment	Status	Modify
06-30	Vacation	Pending	Edit
07-07	Urgent	Pending	Edit
07-12	Needed	Pending	Edit
07-17	Needed	Pending	Edit

July 7, 2023

Learned @input(), @output() decorator and EventEmitter in Angular.

July 10, 2023

Implemented Request for leave form to apply for leave using @Output() decorator and EventEmitter.



July 11, 2023

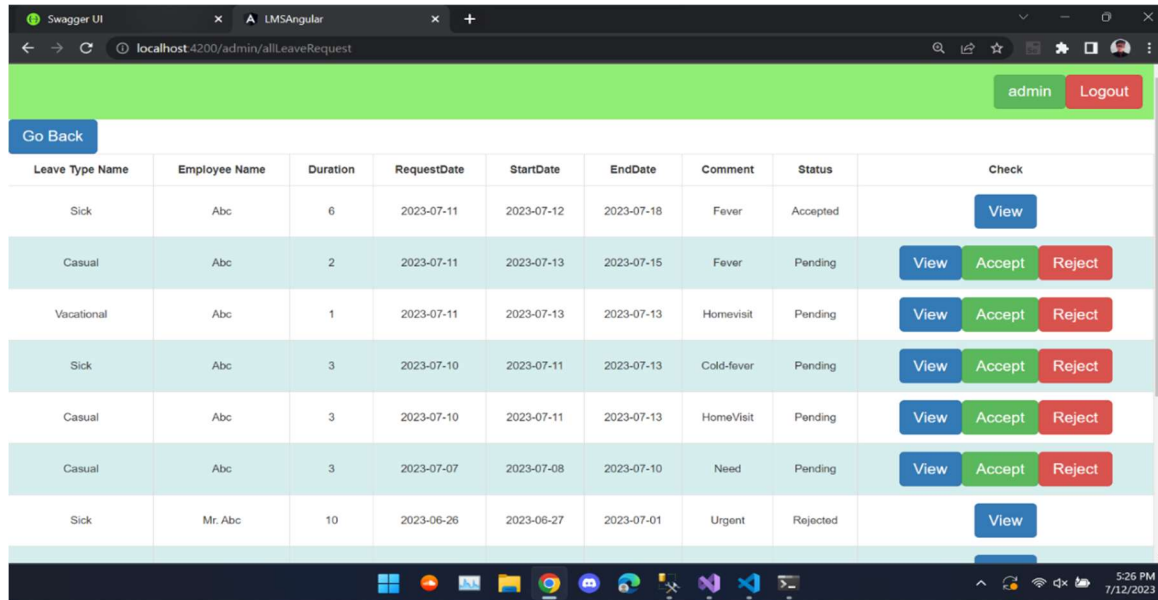
Completed implementing user dashboard of the LMS project. Passed data through components using @Input(), @Output() decorator and EventEmitter.

The screenshot displays a web application running in a browser. The browser's address bar shows 'localhost:4200/user'. The application has a green header bar with a user profile 'abc' and a 'Logout' button. Below the header, there is a 'Request for Leave' button. The main content area features a table with columns: Leave Type Name, Employee Name, Duration, RequestDate, StartDate, EndDate, Comment, Status, and Modify. The table contains eight rows of leave requests, all with a status of 'Pending'. Each row has an 'Edit' button in the 'Modify' column.

Leave Type Name	Employee Name	Duration	RequestDate	StartDate	EndDate	Comment	Status	Modify
Vacation	Abc	10	2023-06-19	2023-06-20	2023-06-30	Urgent	Pending	Edit
Sick	Abc	10	2023-06-26	2023-06-27	2023-07-07	Urgent	Pending	Edit
Casual	Abc	3	2023-07-07	2023-07-08	2023-07-10	Need	Pending	Edit
Sick	Abc	3	2023-07-10	2023-07-11	2023-07-13	Cold-fever	Pending	Edit
Casual	Abc	3	2023-07-10	2023-07-11	2023-07-13	HomeVisit	Pending	Edit
Sick	Abc	6	2023-07-11	2023-07-12	2023-07-18	Fever	Pending	Edit
Casual	Abc	2	2023-07-11	2023-07-13	2023-07-15	Fever	Pending	Edit
Vacational	Abc	1	2023-07-11	2023-07-13	2023-07-13	Home visit	Pending	Edit

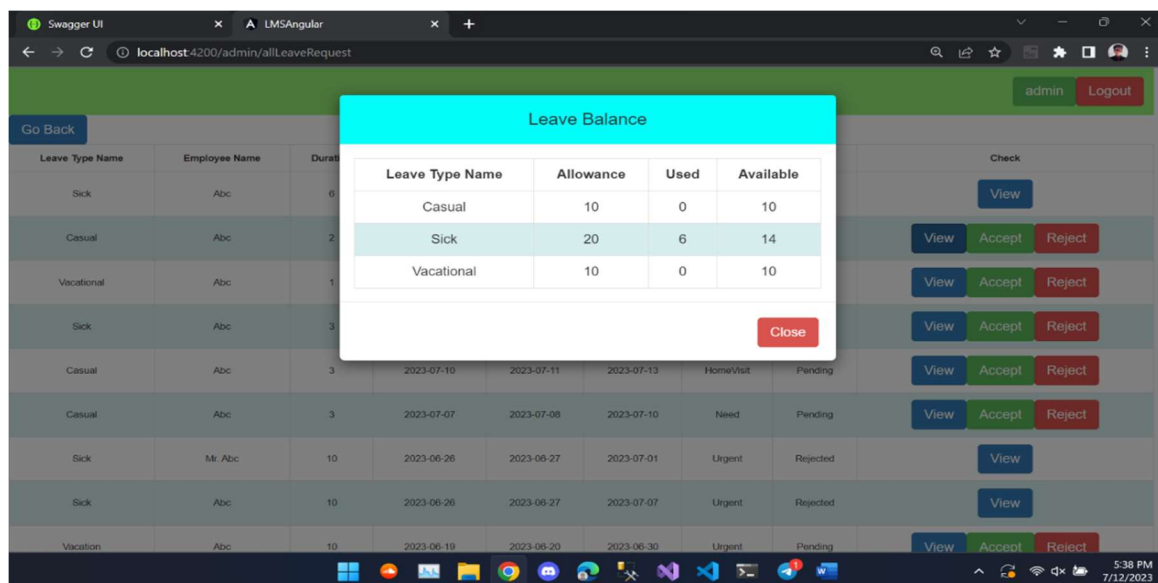
July 12, 2023

Implemented how 'Admin' will handle the Leave Requests of the LMS project. Admin can check all the LeaveRequests and by clicking on the "View" button of a particular request, it will show the leave balance of the requested employee. By checking the leave balance admin can either accept the request by clicking "Accept" button or deny by clicking on "Reject" button.



The screenshot shows a web browser window with the URL `localhost:4200/admin/allLeaveRequest`. The page has a green header with 'admin' and 'Logout' buttons. Below the header is a 'Go Back' button. The main content is a table of leave requests. The table has columns: Leave Type Name, Employee Name, Duration, RequestDate, StartDate, EndDate, Comment, Status, and Check. The 'Check' column contains 'View' buttons. The table lists several requests, including Sick, Casual, and Vacational leaves for employees 'Abc' and 'Mr. Abc'. The status of the requests varies, including 'Accepted', 'Pending', and 'Rejected'.

Leave Type Name	Employee Name	Duration	RequestDate	StartDate	EndDate	Comment	Status	Check
Sick	Abc	6	2023-07-11	2023-07-12	2023-07-18	Fever	Accepted	<button>View</button>
Casual	Abc	2	2023-07-11	2023-07-13	2023-07-15	Fever	Pending	<button>View</button> <button>Accept</button> <button>Reject</button>
Vacational	Abc	1	2023-07-11	2023-07-13	2023-07-13	Homevisit	Pending	<button>View</button> <button>Accept</button> <button>Reject</button>
Sick	Abc	3	2023-07-10	2023-07-11	2023-07-13	Cold-fever	Pending	<button>View</button> <button>Accept</button> <button>Reject</button>
Casual	Abc	3	2023-07-10	2023-07-11	2023-07-13	HomeVisit	Pending	<button>View</button> <button>Accept</button> <button>Reject</button>
Casual	Abc	3	2023-07-07	2023-07-08	2023-07-10	Need	Pending	<button>View</button> <button>Accept</button> <button>Reject</button>
Sick	Mr. Abc	10	2023-06-26	2023-06-27	2023-07-01	Urgent	Rejected	<button>View</button>



The screenshot shows the same web browser window as the previous one, but with a modal window titled 'Leave Balance' open. The modal contains a table with columns: Leave Type Name, Allowance, Used, and Available. The table lists the leave balance for 'Casual', 'Sick', and 'Vacational' leave types. The 'Sick' row is highlighted. The modal also has a 'Close' button.

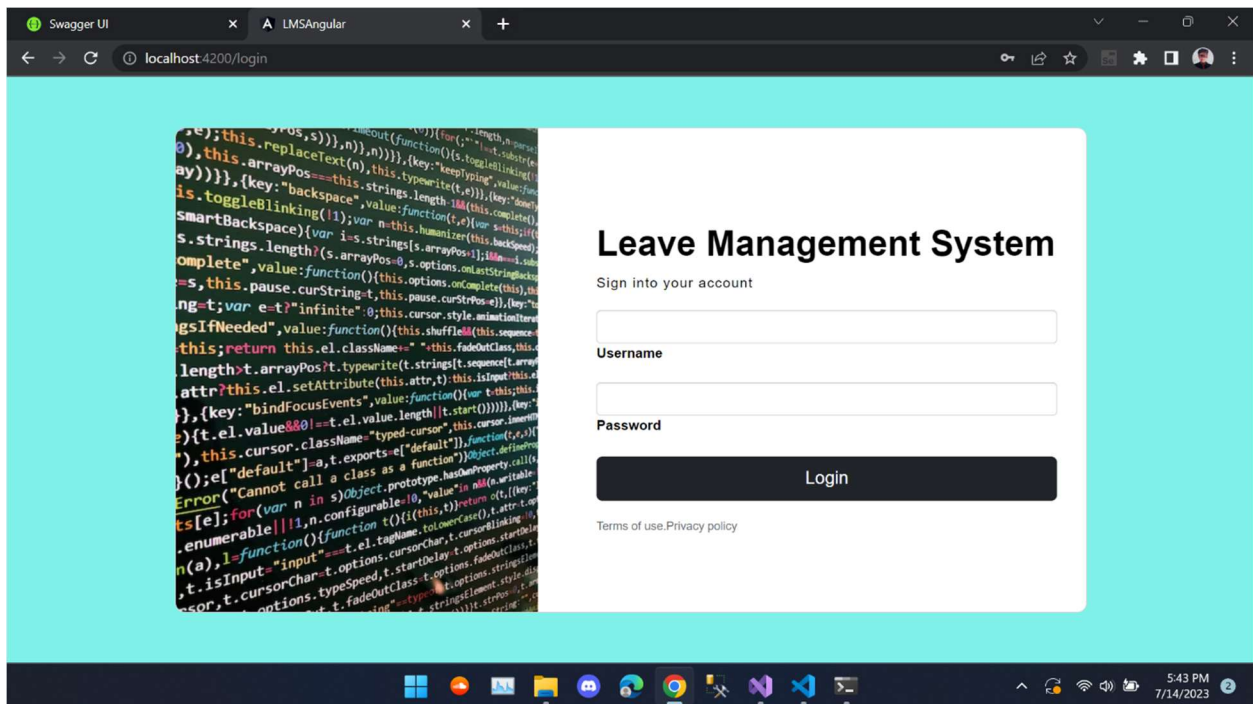
Leave Type Name	Allowance	Used	Available
Casual	10	0	10
Sick	20	6	14
Vacational	10	0	10

July 13, 2023

Implemented how employee will view the rejected leave request and the rejection cause of the LMS project. Tried to update the navigation bar of admin.

July 14, 2023

Completed implementing the LMS project.



Swagger UI

LMSAngular

+

localhost:4200/user

Leave Management System

Logout

Request for Leave

Leave Type Name	Duration	RequestDate	StartDate	EndDate	Comment	Status	Modify
Sick	6	2023-07-11	2023-07-12	2023-07-18	Fever	Pending	
Casual	2	2023-07-11	2023-07-13	2023-07-15	Fever	Rejected	
Vacational	1	2023-07-11	2023-07-13	2023-07-13	Home visit	Pending	
Sick	3	2023-07-10	2023-07-11	2023-07-13	Cold-fever	Rejected	
Casual	3	2023-07-10	2023-07-11	2023-07-13	HomeVisit	Pending	
Casual	3	2023-07-07	2023-07-08	2023-07-10	Need	Accepted	
Sick	10	2023-06-26	2023-06-27	2023-07-07	Urgent	Accepted	
Vacation	10	2023-06-19	2023-06-20	2023-06-30	Urgent	Rejected	

5:42 PM
7/14/2023

Swagger UI

LMSAngular

+

localhost:4200/admin/allLeaveRequest

Leave Management System

Leave Request Employee

Logout

Leave Type Name	Employee Name	Duration	RequestDate	StartDate	EndDate	Comment	Status	Check
Sick	Mr. XyZZ	4	2023-07-14	2023-07-15	2023-07-18	Cold	Accepted	
Sick	Abc	6	2023-07-11	2023-07-12	2023-07-18	Fever	Pending	
Casual	Abc	2	2023-07-11	2023-07-13	2023-07-15	Fever	Rejected	
Vacational	Abc	1	2023-07-11	2023-07-13	2023-07-13	Home visit	Pending	
Sick	Abc	3	2023-07-10	2023-07-11	2023-07-13	Cold-fever	Rejected	
Casual	Abc	3	2023-07-10	2023-07-11	2023-07-13	HomeVisit	Pending	
Casual	Abc	3	2023-07-07	2023-07-08	2023-07-10	Need	Accepted	
Sick	Mr. Abc	10	2023-06-26	2023-06-27	2023-07-01	Urgent	Accepted	
Sick	Abc	10	2023-06-26	2023-06-27	2023-07-07	Urgent	Accepted	
Vacation	Abc	10	2023-06-19	2023-06-20	2023-06-30	Urgent	Rejected	
Vacational	Mr. Abc	15	2023-06-10	2023-06-10	2023-06-25	Vacation	Pending	
Casual	Mr. Xyz	10	2023-06-09	2023-06-10	2023-06-20	Sick	Accepted	
Sick	Mr. Xyz	15	2023-06-09	2023-06-10	2023-06-25	Sick	Pending	

5:42 PM
7/14/2023