Mozilla Firefox

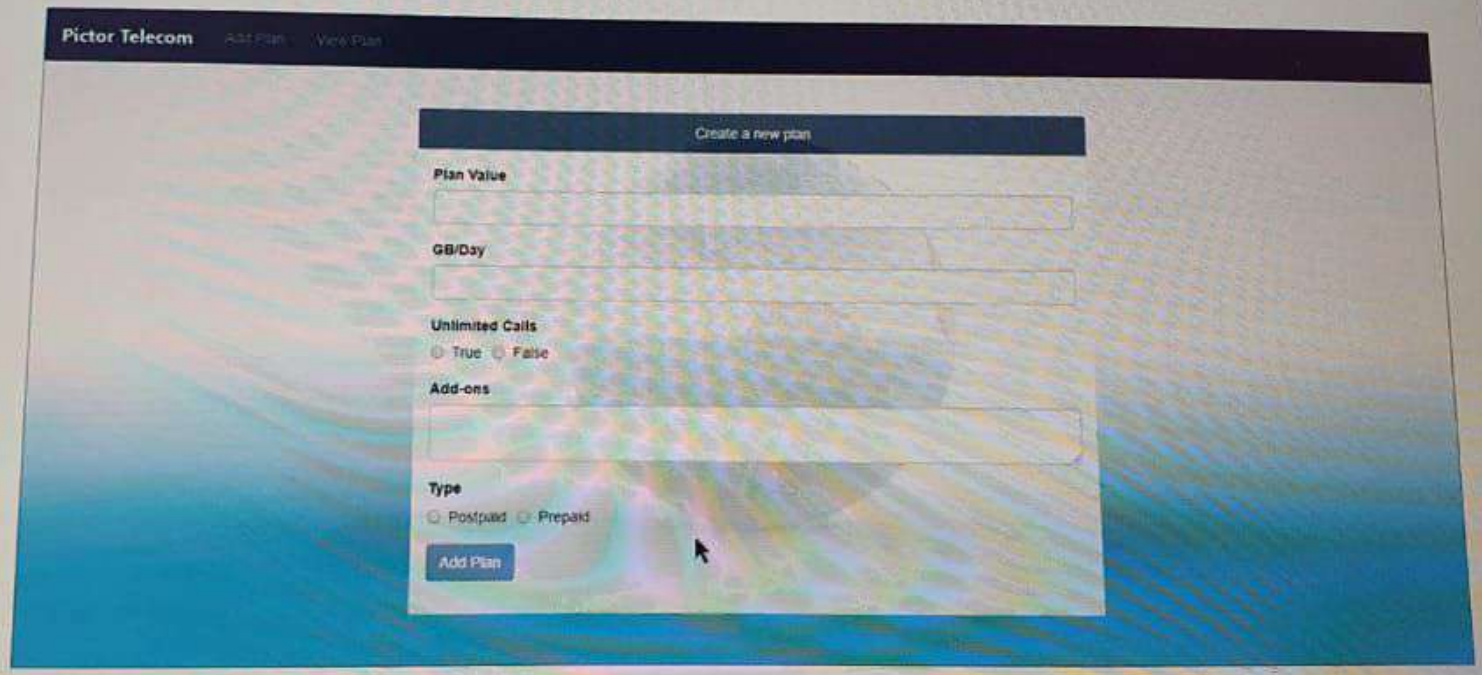
/home/user/Desktop/user_re

Firefox Privacy Notice —

AngCertProjJest

file:///home/user/Desktop/user_repo/QP/index.html

1. When **"http://localhost:4200"** is entered in the browser then the following form should be displayed:



2. Add validations for form controls as below. Use field names as mentioned below and use validations inside a

Execute

View Result

Mozilla Firefox

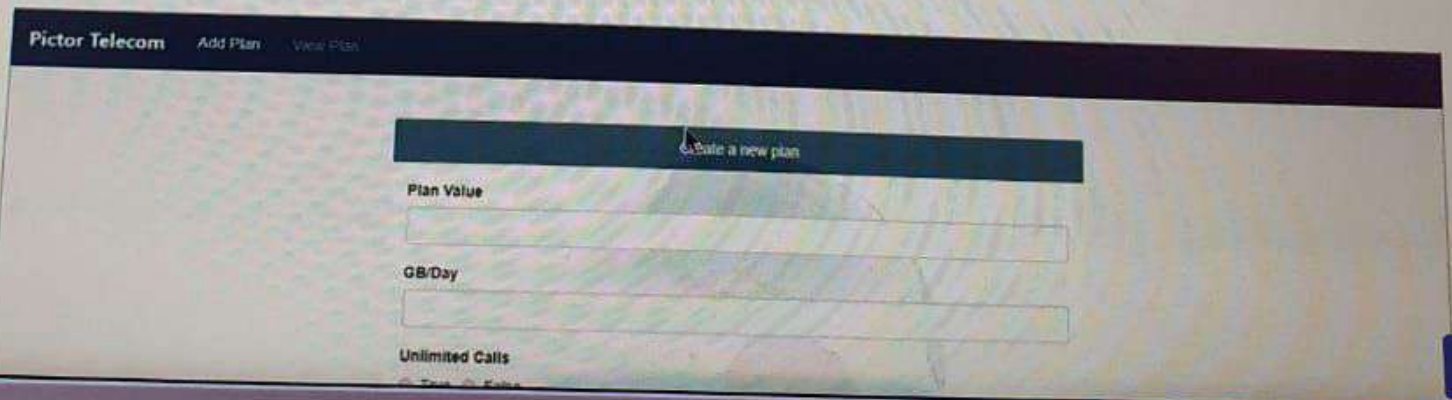
/home/user/Desktop/user_re | Firefox Privacy Notice — | AngCertProjJest

file:///home/user/Desktop/user_repo/QP/index.html

2. Add validations for form controls as below. Use field names as mentioned below and use validations inside a component only.

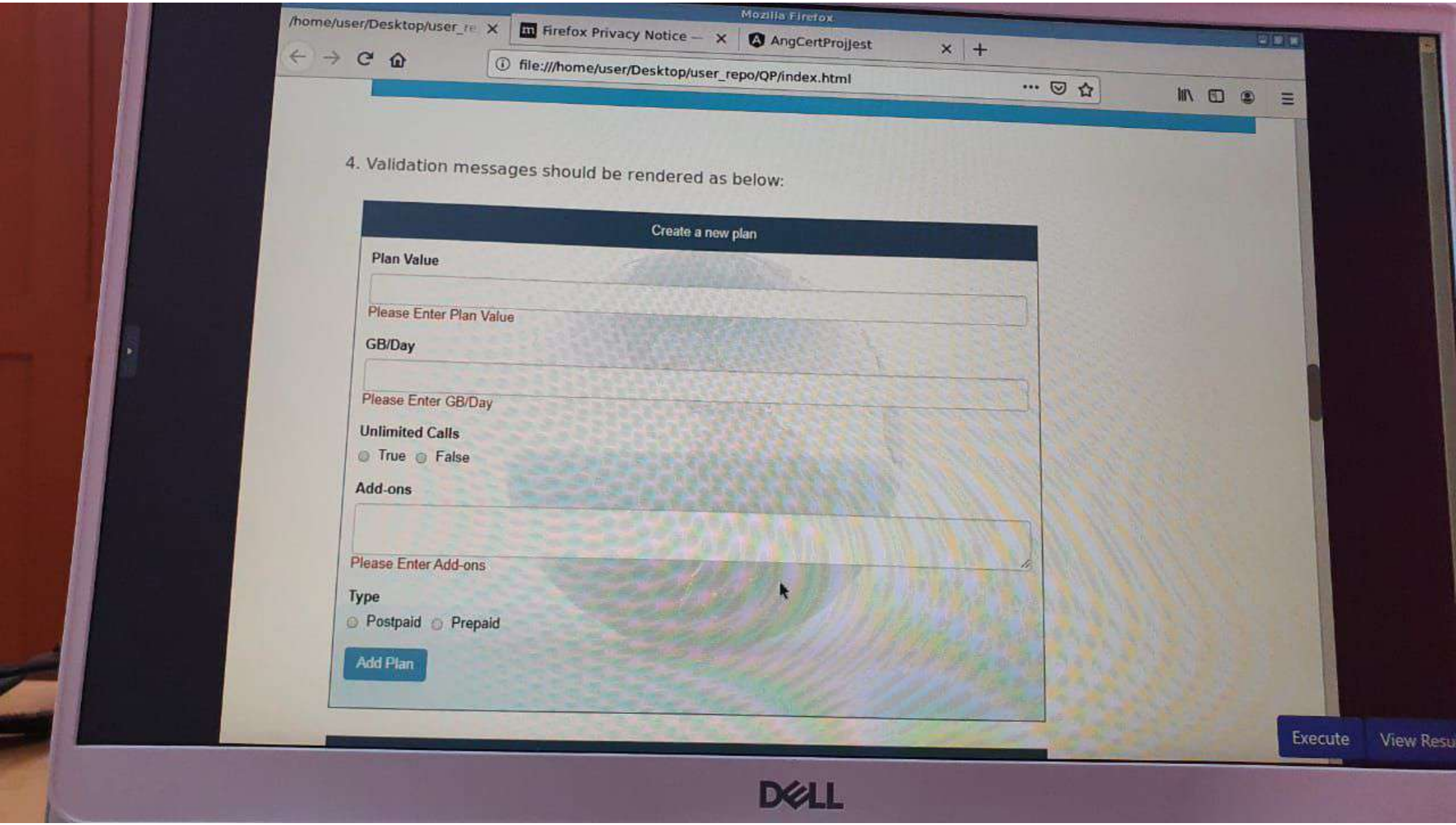
Field name	Validations Required
planValue	Required validation
data	Required validation, Maximum 200
unLimitedCalls	Required validation
addOns	Required validation, Pattern (only alphabets and special characters)
planType	Required validation

3. When **Add Plan** is clicked view should be displayed as below:



Execute | View Result

DELL

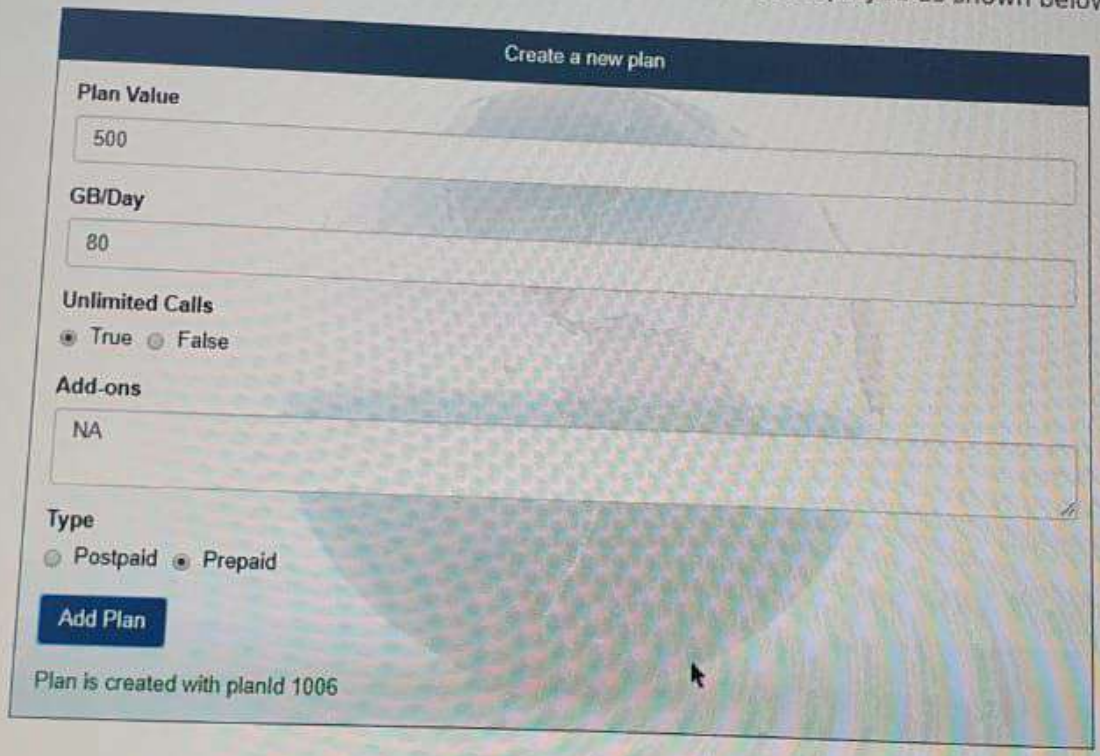

Mozilla Firefox
/home/user/Desktop/user_re
Firefox Privacy Notice —
AngCertProjJest
file:///home/user/Desktop/user_repo/QP/index.html
4. Validation messages should be rendered as below:
Create a new plan
Plan Value
Please Enter Plan Value
GB/Day
Please Enter GB/Day
Unlimited Calls
True
False
Add-ons
Please Enter Add-ons
Type
Postpaid
Prepaid
Add Plan
Execute
View Resu
DELL

Mozilla Firefox

/home/user/Desktop/user_r... | Firefox Privacy Notice — | AngCertProjJest

file:///home/user/Desktop/user_repo/QP/index.html

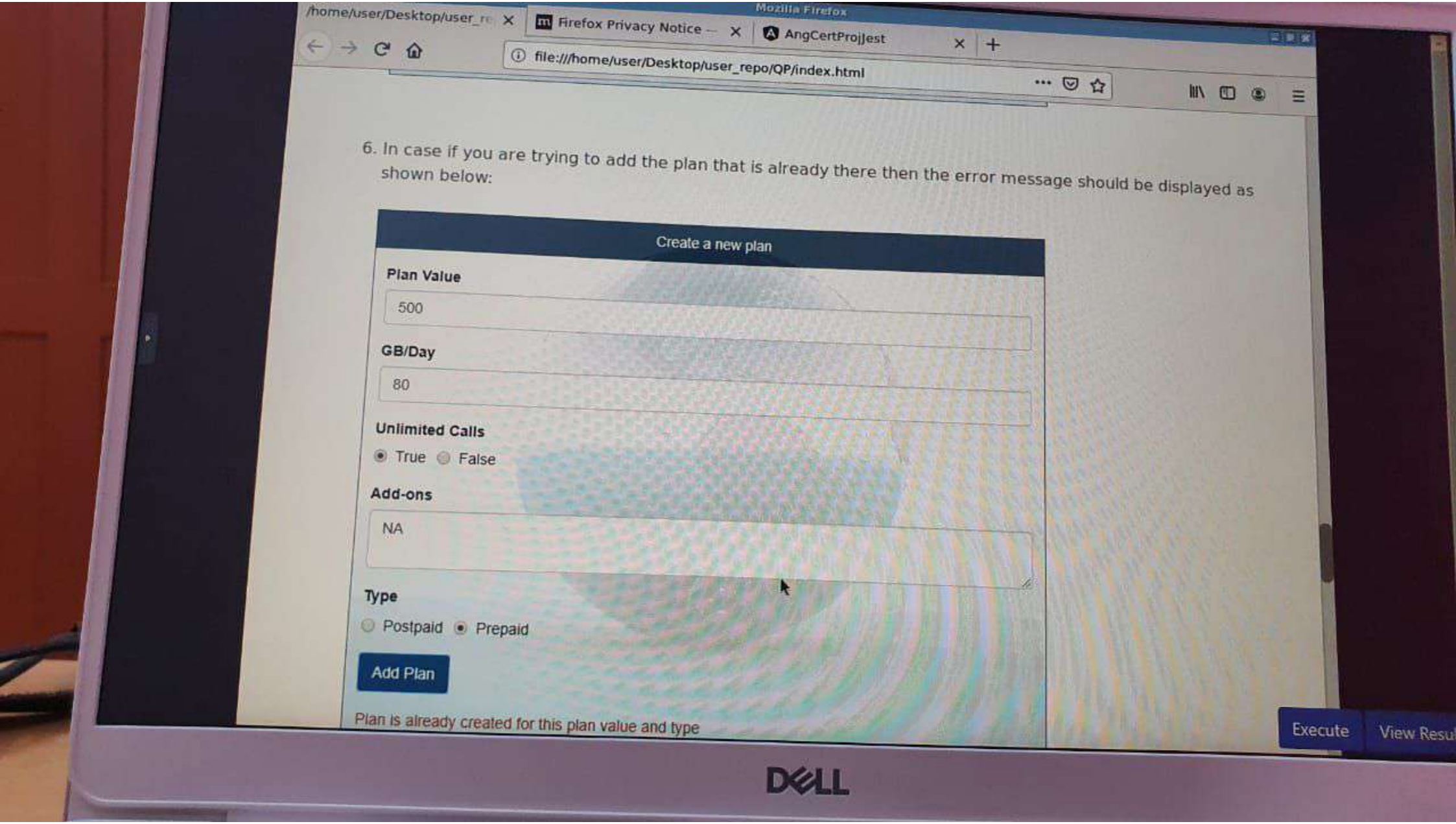
5. On successful execution, the success message should be displayed as shown below:



6. In case if you are trying to add the plan that is already there then the error message should be displayed as shown below:

Execute | View Result

DELL


Mozilla Firefox
/home/user/Desktop/user_re
Firefox Privacy Notice —
AngCertProjJest
file:///home/user/Desktop/user_repo/QP/index.html
6. In case if you are trying to add the plan that is already there then the error message should be displayed as shown below:
Create a new plan
Plan Value
500
GB/Day
80
Unlimited Calls
True
False
Add-ons
NA
Type
Postpaid
Prepaid
Add Plan
Plan is already created for this plan value and type
Execute
View Resu
DELL

Mozilla Firefox

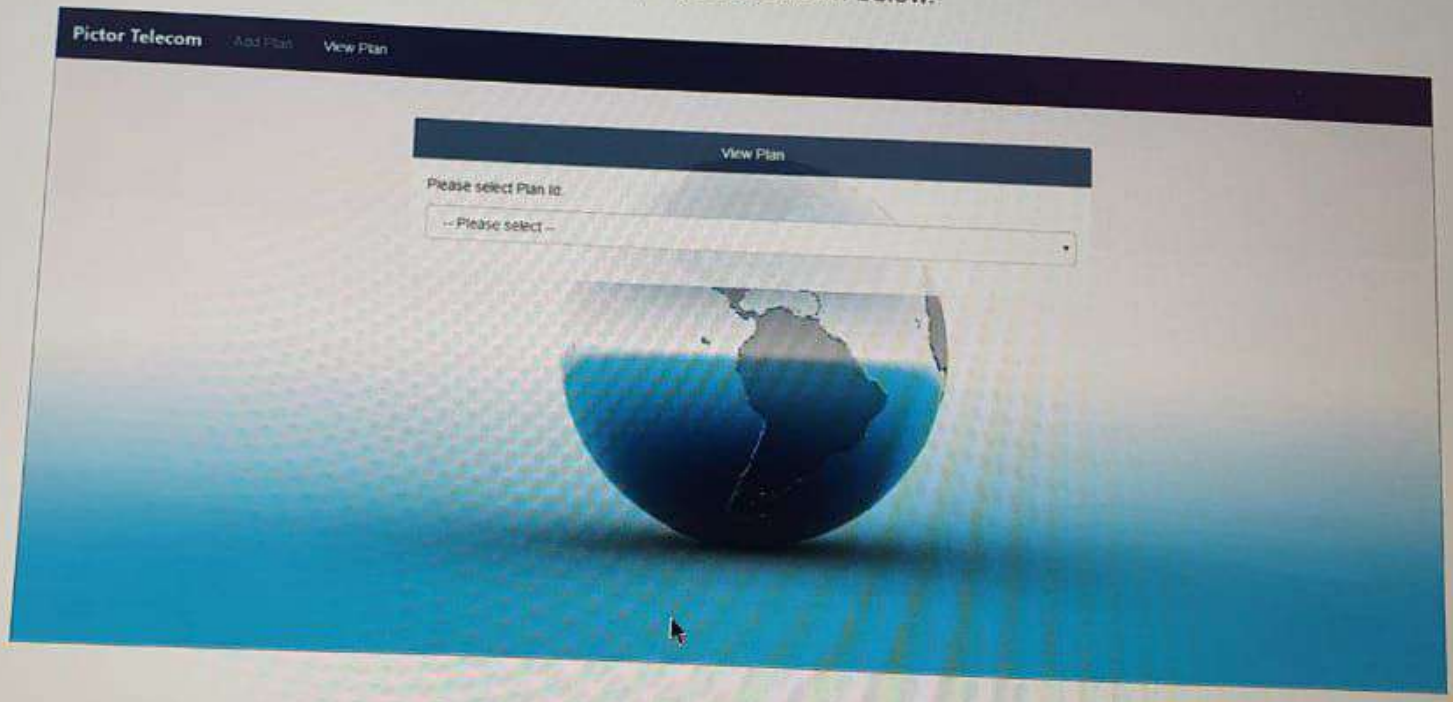
/home/user/Desktop/user_re

Firefox Privacy Notice —

AngCertProjJest

file:///home/user/Desktop/user_repo/QP/index.html

7. When **View Plan** is clicked view should be displayed as shown below:



8. Plan Ids should be populated into the dropdown

Execute

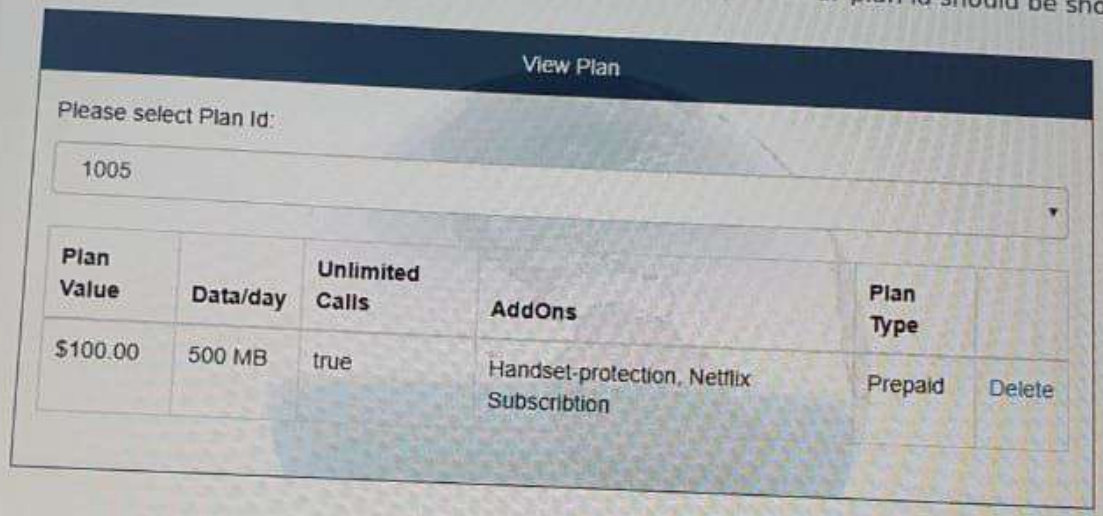
View Resu

DELL

Firefox Privacy Notice — | AngCertProjJest

file:///home/user/Desktop/user_repo/QP/index.html

9. When user selects the plan id then the details of that particular plan id should be shown as below:



10. Transform the

- **Plan Value field -** Render the value in the format shown above.
- **data field -** If the **data** is less than 1, it should add unit as "MB" else it should add unit as"GB "

Note: Use custom pipe for transforming the data field

2. If user clicks delete, it should delete the planId and display following message

Execute View Res

DELL

Firefox Privacy Notice — | AngCertProjJest

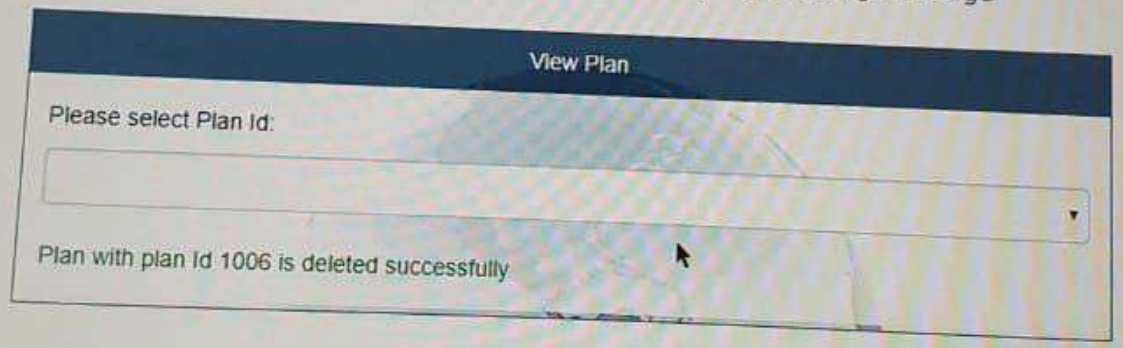
file:///home/user/Desktop/user_repo/QP/index.html

10. Transform the

- **Plan Value field -** Render the value in the format shown above.
- **data field -** If the **data** is less than 1, it should add unit as "MB" else it should add unit as"GB "

Note: Use custom pipe for transforming the data field

3. If user clicks delete, it should delete the planId and display following message

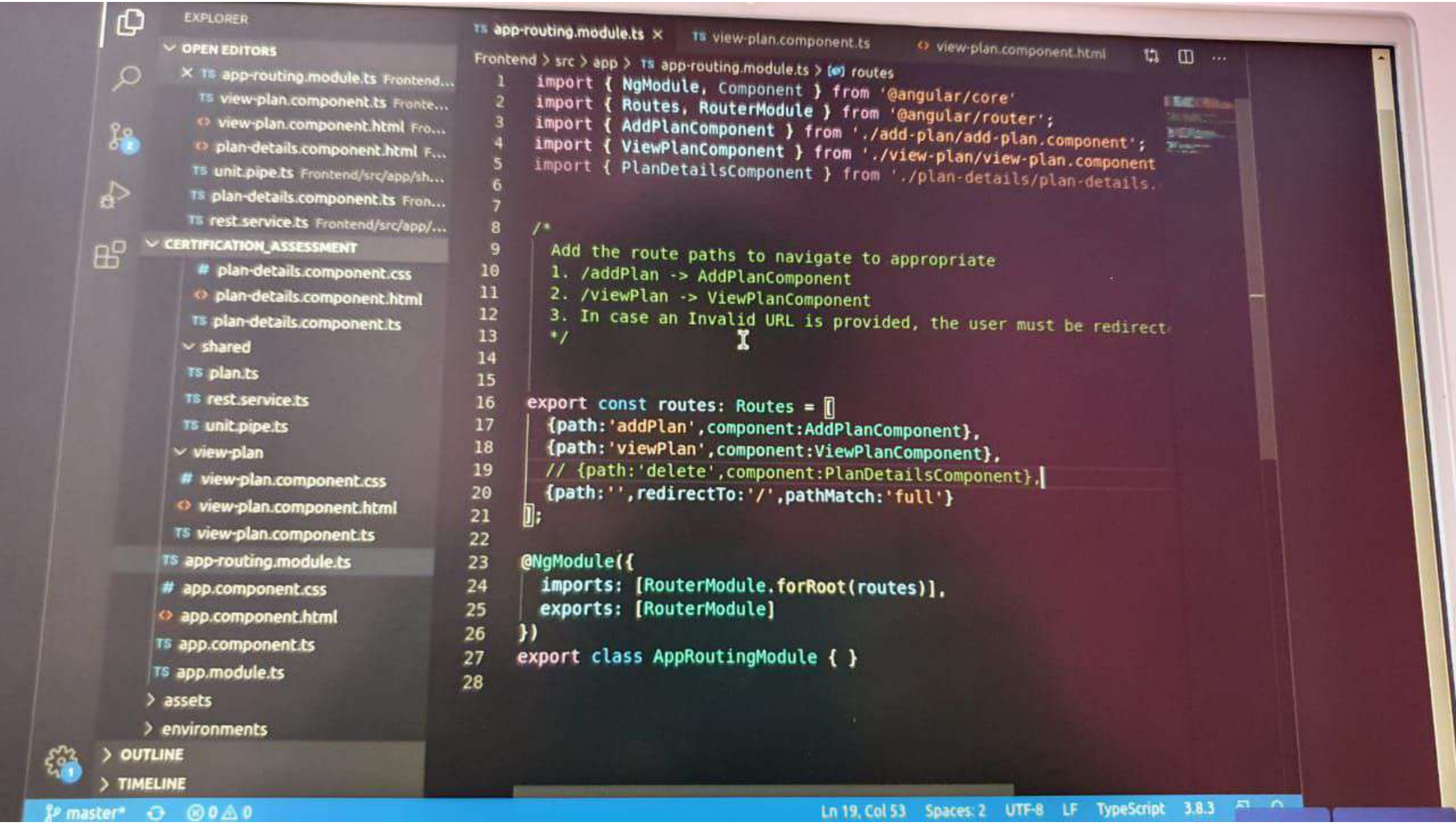


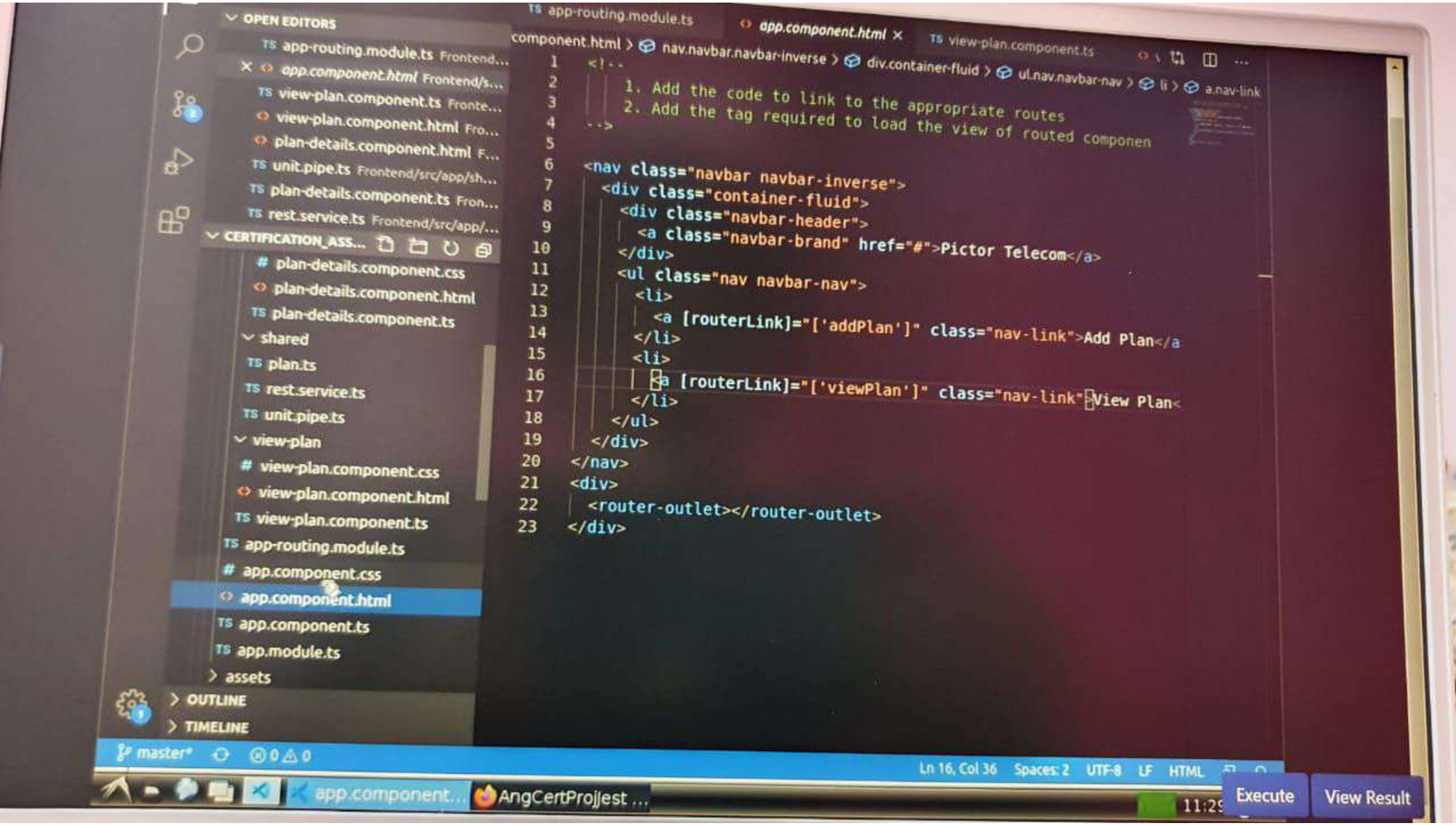
~~~~~~~~~~~~~~~~ All the Best ~~~~~~~~~~~~~~~~

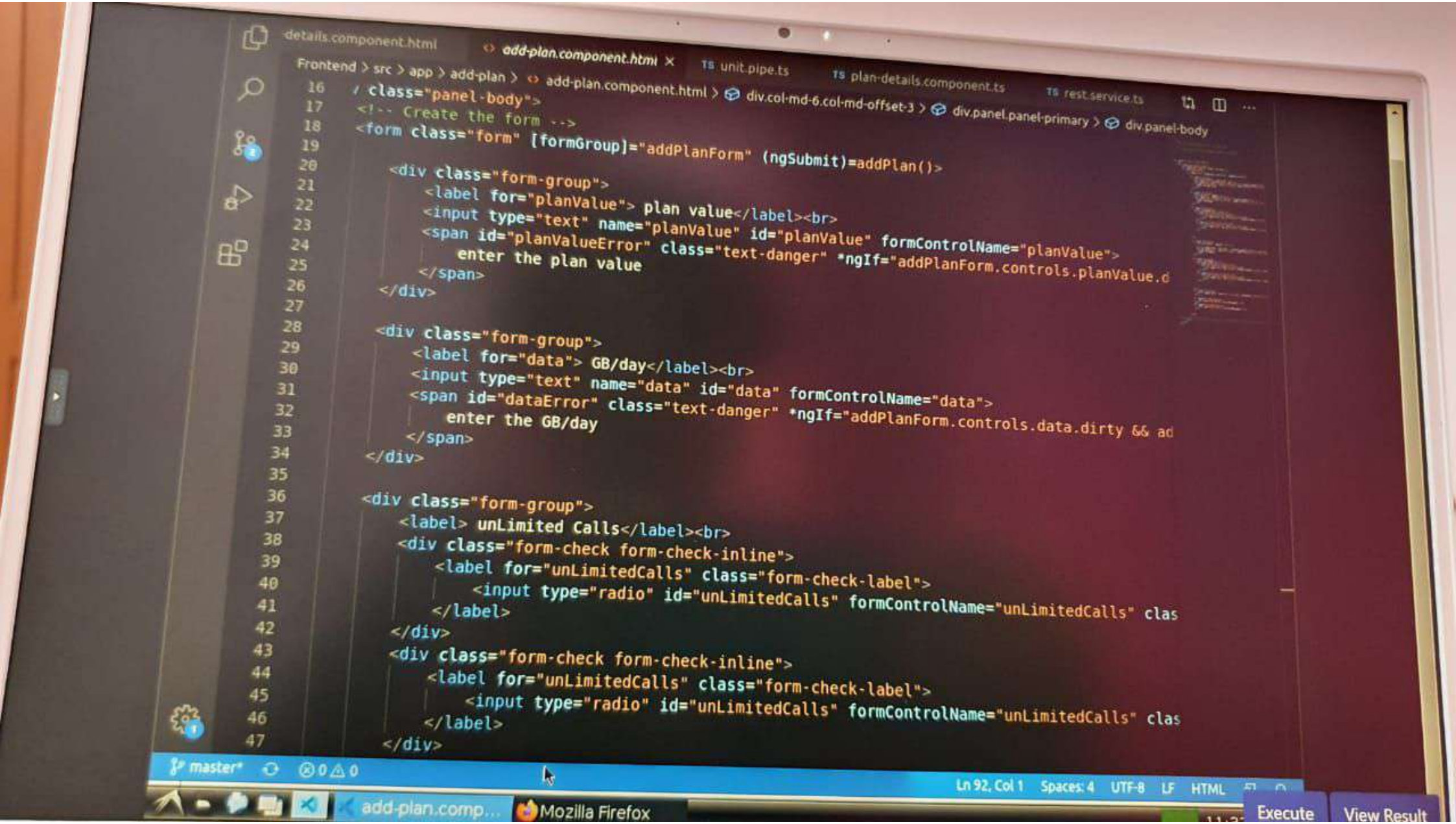
PREV

Execute View Result

DELL
~~~~~~~~~~~~~~~~


EXPLORER
OPEN EDITORS
app-routing.module.ts Frontend...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASSESSMENT
plan-details.component.css
plan-details.component.html
plan-details.component.ts
shared
plan.ts
rest.service.ts
unit.pipe.ts
view-plan
view-plan.component.css
view-plan.component.html
view-plan.component.ts
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
environments
OUTLINE
TIMELINE
app-routing.module.ts
view-plan.component.ts
view-plan.component.html
Frontend > src > app > app-routing.module.ts > routes
import { NgModule, Component } from '@angular/core'
import { Routes, RouterModule } from '@angular/router';
import { AddPlanComponent } from './add-plan/add-plan.component';
import { ViewPlanComponent } from './view-plan/view-plan.component
import { PlanDetailsComponent } from './plan-details/plan-details.
/*
Add the route paths to navigate to appropriate
1. /addPlan -> AddPlanComponent
2. /viewPlan -> ViewPlanComponent
3. In case an Invalid URL is provided, the user must be redirect
*/
export const routes: Routes = [
{path:'addPlan',component:AddPlanComponent},
{path:'viewPlan',component:ViewPlanComponent},
// {path:'delete',component:PlanDetailsComponent},
{path:'',redirectTo:'/',pathMatch:'full'}
];
@NgModule({
imports: [RouterModule.forRoot(routes)],
exports: [RouterModule]
})
export class AppRoutingModule { }
master*
0 0
Ln 19, Col 53
Spaces: 2
UTF-8
LF
TypeScript
3.8.3


OPEN EDITORS
app-routing.module.ts Frontend...
app.component.html Frontend/s...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASS...
plan-details.component.css
plan-details.component.html
plan-details.component.ts
shared
plan.ts
rest.service.ts
unit.pipe.ts
view-plan
view-plan.component.css
view-plan.component.html
view-plan.component.ts
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
OUTLINE
TIMELINE
app-routing.module.ts
app.component.html
view-plan.component.ts
component.html > nav.navbar.navbar-inverse > div.container-fluid > ul.nav.navbar-nav > li > a.nav-link
1 <!--
2 1. Add the code to link to the appropriate routes
3 2. Add the tag required to load the view of routed componen
4 -->
5
6 <nav class="navbar navbar-inverse">
7
8
9 <a class="navbar-brand" href="#">Pictor Telecom</a>
10
11 <ul class="nav navbar-nav">
12 <li>
13 <a [routerLink]="['addPlan']" class="nav-link">Add Plan</a
14 </li>
15 <li>
16 <a [routerLink]="['viewPlan']" class="nav-link">View Plan<
17 </li>
18 </ul>
19
20 </nav>
21
22 <router-outlet></router-outlet>
23
master*
0 0
Ln 16, Col 36
Spaces: 2
UTF-8
LF
HTML
app.component...
AngCertProjJest ...
Execute
View Result


details.component.html
add-plan.component.html
unit.pipe.ts
plan-details.component.ts
rest.service.ts
Frontend > src > app > add-plan > add-plan.component.html > div.col-md-6.col-md-offset-3 > div.panel.panel-primary > div.panel-body
/ class="panel-body">
<!-- Create the form -->
<form class="form" [formGroup]="addPlanForm" (ngSubmit)=addPlan()>

<label for="planValue"> plan value</label>
<input type="text" name="planValue" id="planValue" formControlName="planValue">
<span id="planValueError" class="text-danger" *ngIf="addPlanForm.controls.planValue.d
enter the plan value

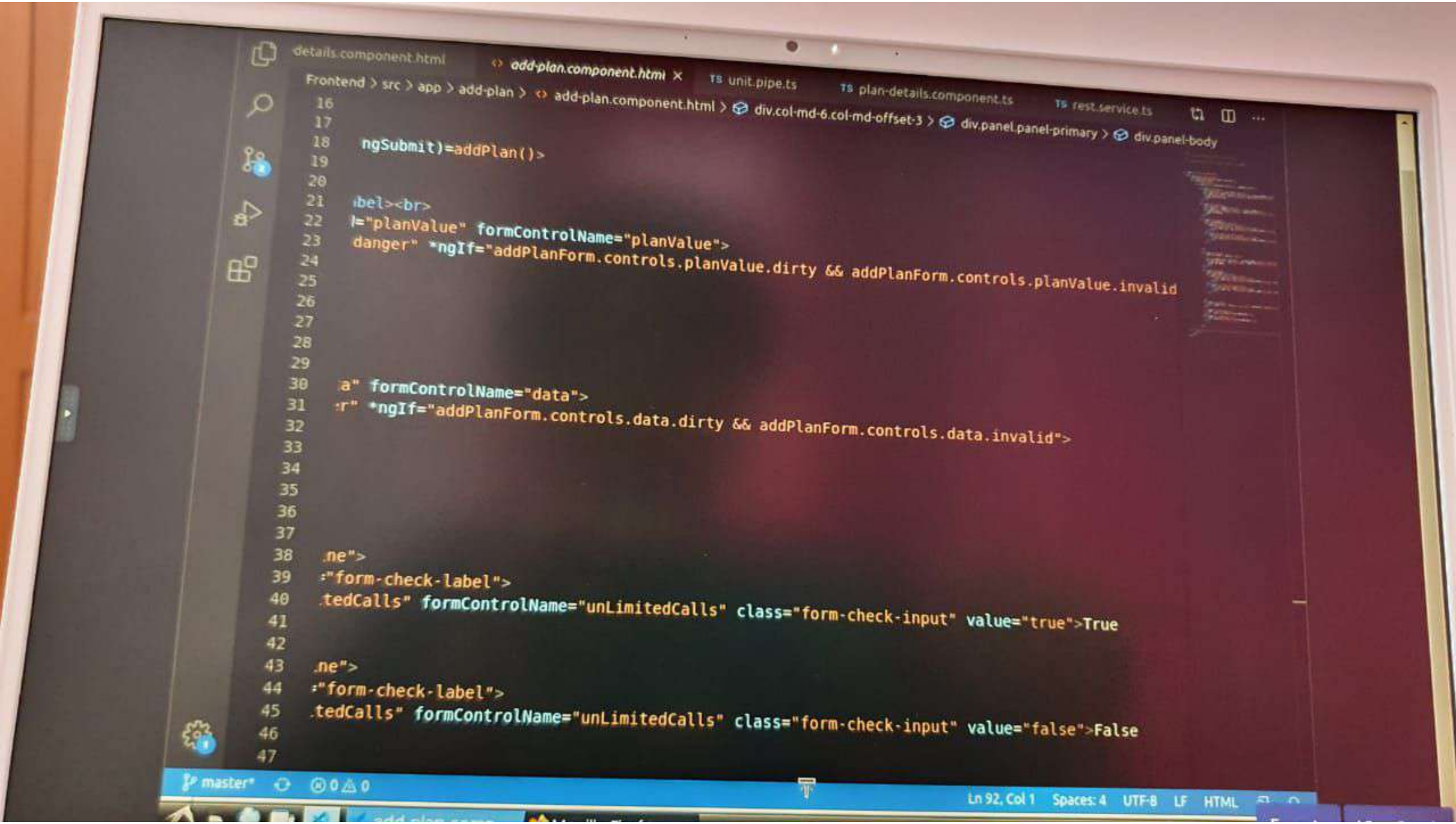
<label for="data"> GB/day</label>
<input type="text" name="data" id="data" formControlName="data">
<span id="dataError" class="text-danger" *ngIf="addPlanForm.controls.data.dirty && ad
enter the GB/day

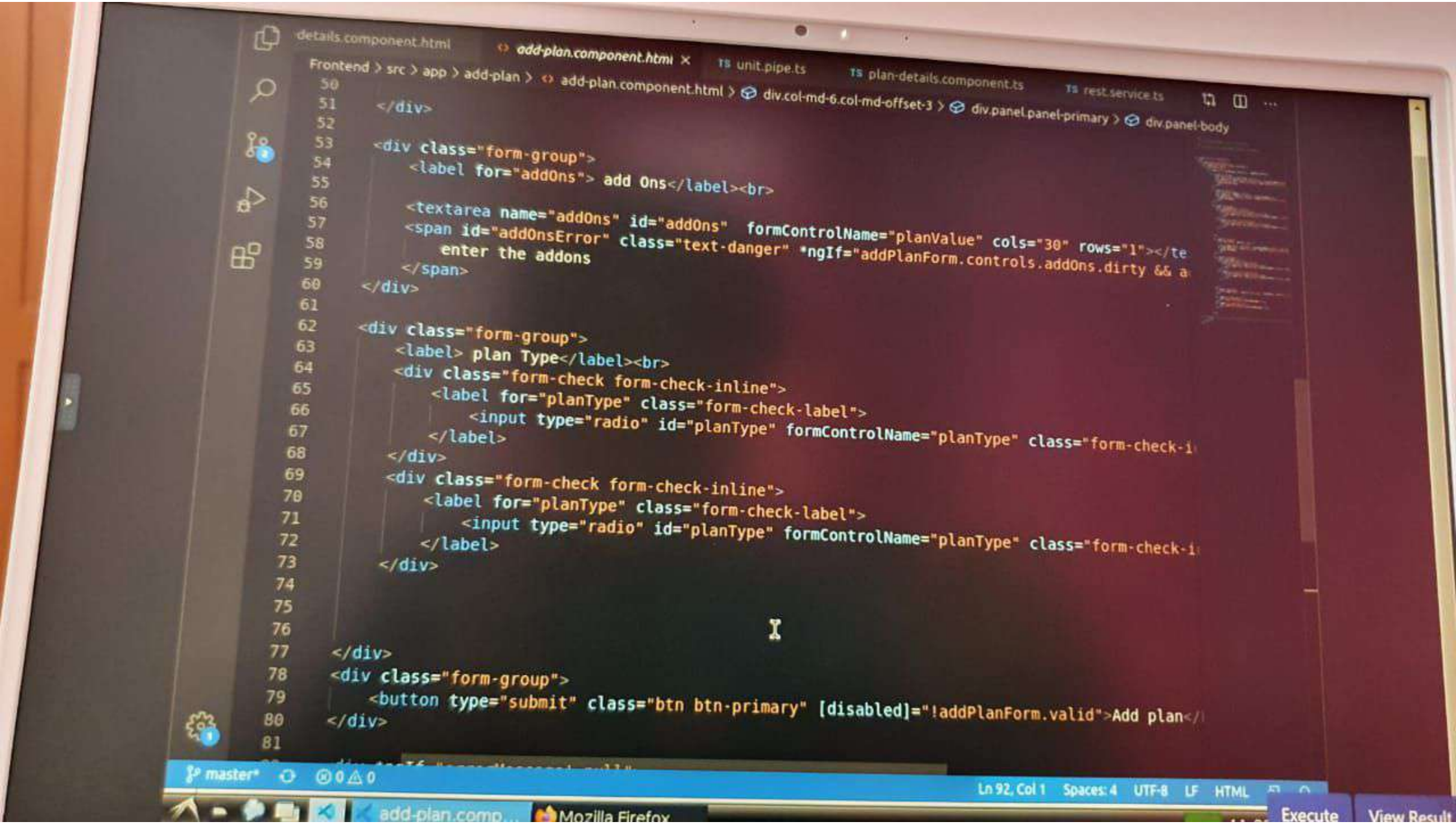
<label> unLimited Calls</label>

<label for="unLimitedCalls" class="form-check-label">
<input type="radio" id="unLimitedCalls" formControlName="unLimitedCalls" clas
</label>

<label for="unLimitedCalls" class="form-check-label">
<input type="radio" id="unLimitedCalls" formControlName="unLimitedCalls" clas
</label>

master*
0 0
Ln 92, Col 1
Spaces: 4
UTF-8
LF
HTML
add-plan.comp...
Mozilla Firefox
Execute
View Result


details.component.html
add-plan.component.html
unit.pipe.ts
plan-details.component.ts
rest.service.ts
Frontend > src > app > add-plan > add-plan.component.html > div.col-md-6.col-md-offset-3 > div.panel.panel-primary > div.panel-body
ngSubmit)=addPlan()>
bel>
="planValue" formControlName="planValue">
danger" *ngIf="addPlanForm.controls.planValue.dirty && addPlanForm.controls.planValue.invalid
a" formControlName="data">
r" *ngIf="addPlanForm.controls.data.dirty && addPlanForm.controls.data.invalid">
ne">
:"form-check-label">
tedCalls" formControlName="unLimitedCalls" class="form-check-input" value="true">True
ne">
:"form-check-label">
tedCalls" formControlName="unLimitedCalls" class="form-check-input" value="false">False
master*
Ln 92, Col 1
Spaces: 4
UTF-8
LF
HTML


details.component.html
add-plan.component.html
unit.pipe.ts
plan-details.component.ts
rest.service.ts
Frontend > src > app > add-plan > add-plan.component.html > div.col-md-6.col-md-offset-3 > div.panel.panel-primary > div.panel-body

<label for="addOns"> add Ons</label>
<textarea name="addOns" id="addOns" formControlName="planValue" cols="30" rows="1"></te
<span id="addOnsError" class="text-danger" *ngIf="addPlanForm.controls.addOns.dirty && a
enter the addons

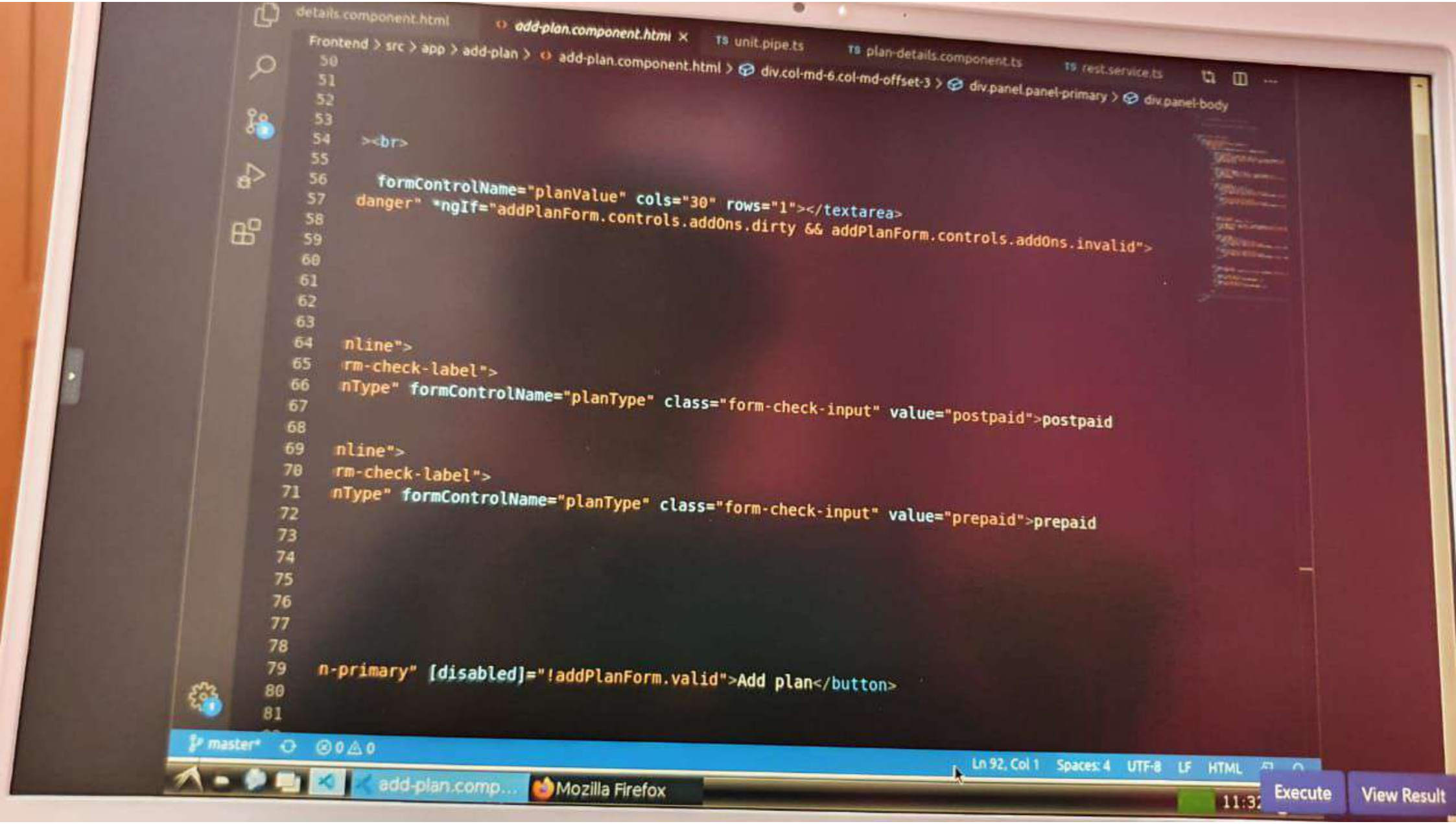
<label> plan Type</label>

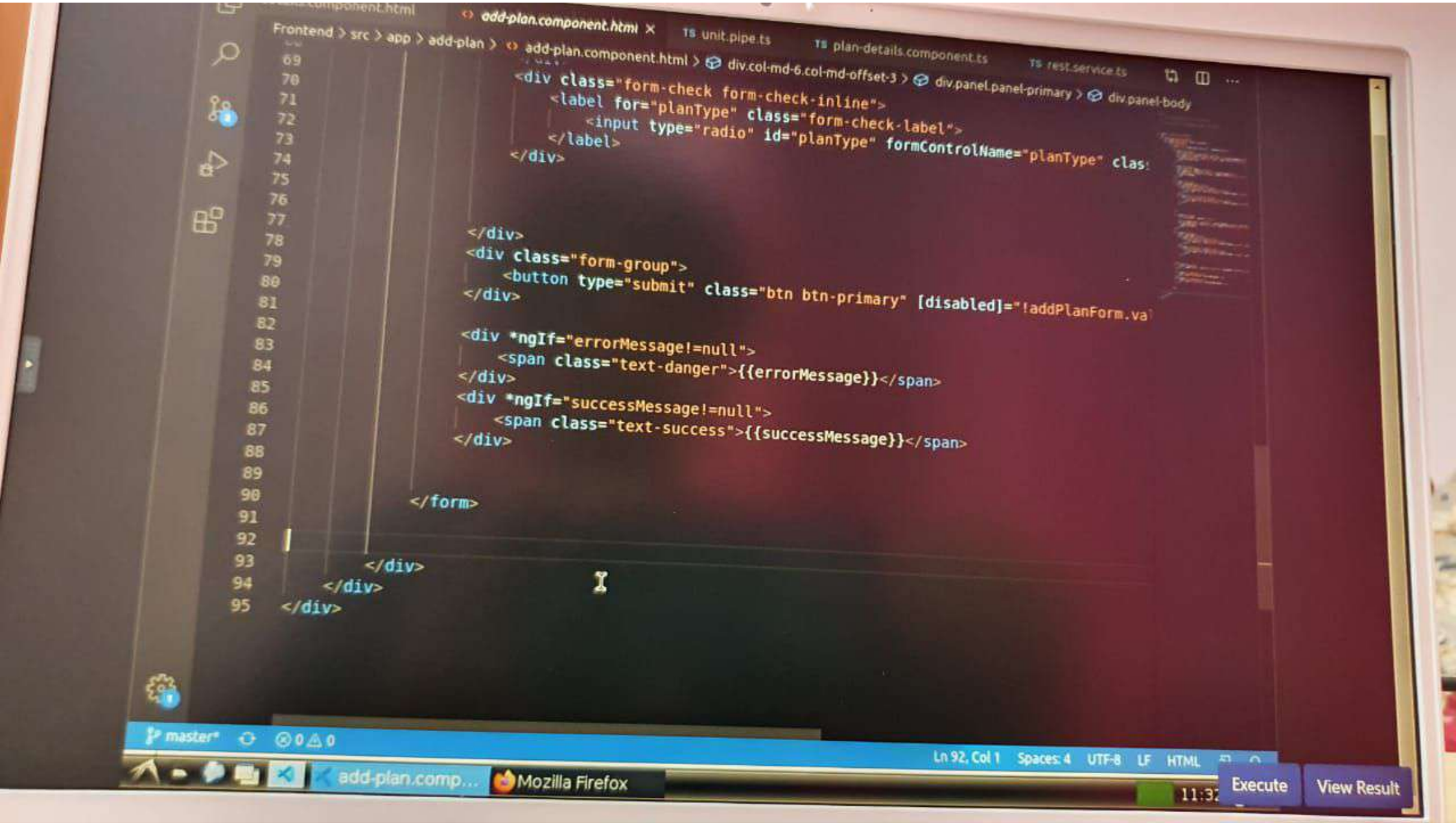
<label for="planType" class="form-check-label">
<input type="radio" id="planType" formControlName="planType" class="form-check-i
</label>

<label for="planType" class="form-check-label">
<input type="radio" id="planType" formControlName="planType" class="form-check-i
</label>

<button type="submit" class="btn btn-primary" [disabled]="!addPlanForm.valid">Add plan</

master*
Ln 92, Col 1
Spaces: 4
UTF-8
LF
HTML
Mozilla Firefox
Execute
View Result


details.component.html
add-plan.component.html
unit.pipe.ts
plan-details.component.ts
rest.service.ts
Frontend > src > app > add-plan > add-plan.component.html > div.col-md-6.col-md-offset-3 > div.panel.panel-primary > div.panel-body
>
formControlName="planValue" cols="30" rows="1"></textarea>
danger" *ngIf="addPlanForm.controls.addOns.dirty && addPlanForm.controls.addOns.invalid">
nline">
rm-check-label">
nType" formControlName="planType" class="form-check-input" value="postpaid">postpaid
nline">
rm-check-label">
nType" formControlName="planType" class="form-check-input" value="prepaid">prepaid
n-primary" [disabled]="!addPlanForm.valid">Add plan</button>
master*
Ln 92, Col 1
Spaces: 4
UTF-8
LF
HTML
add-plan.comp...
Mozilla Firefox
Execute
View Result


add-plan.component.html
unit.pipe.ts
plan-details.component.ts
rest.service.ts
Frontend > src > app > add-plan > add-plan.component.html > div.col-md-6.col-md-offset-3 > div.panel.panel-primary > div.panel-body

<label for="planType" class="form-check-label">
<input type="radio" id="planType" formControlName="planType" clas
</label>

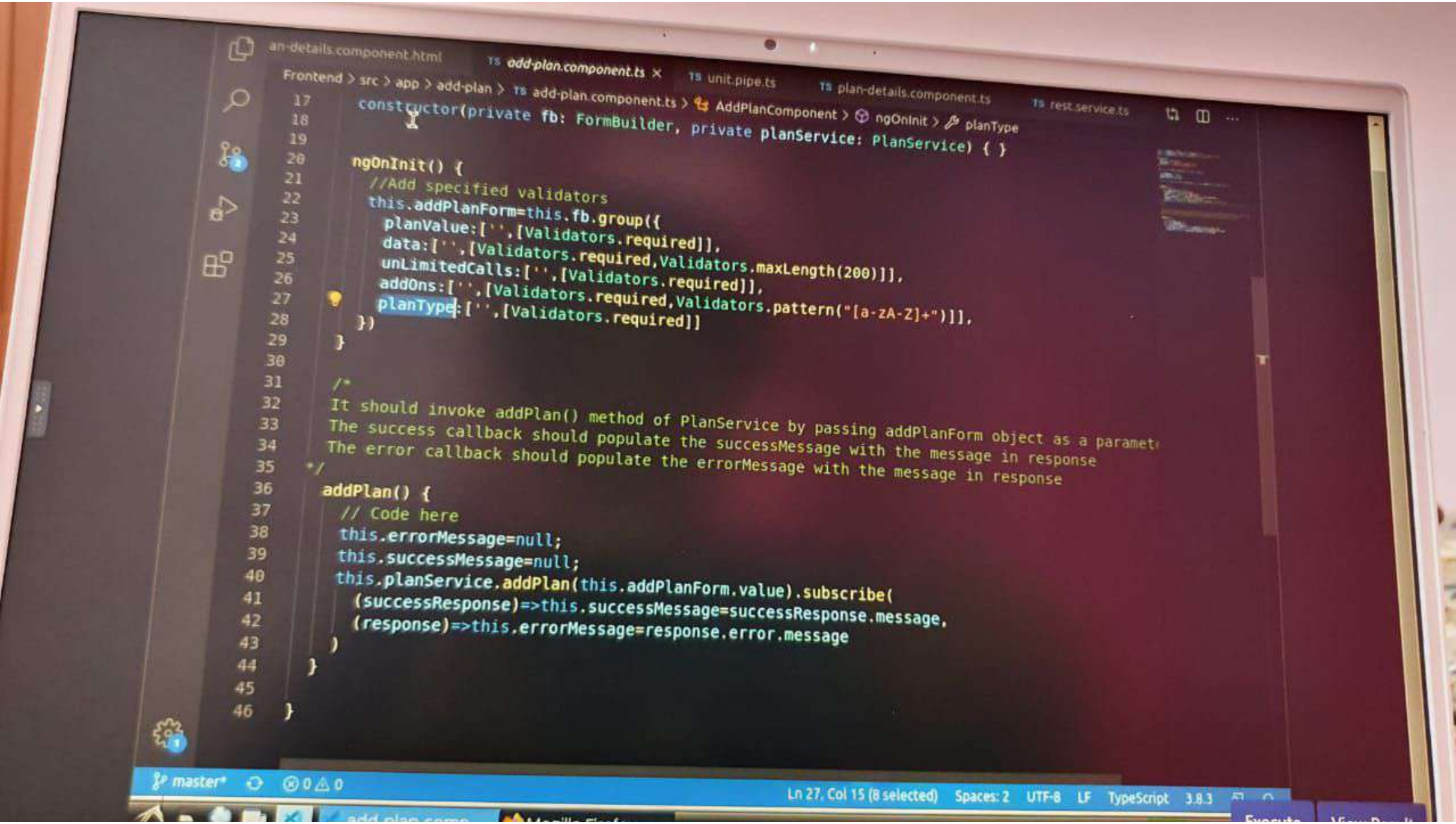
<button type="submit" class="btn btn-primary" [disabled]="!addPlanForm.va

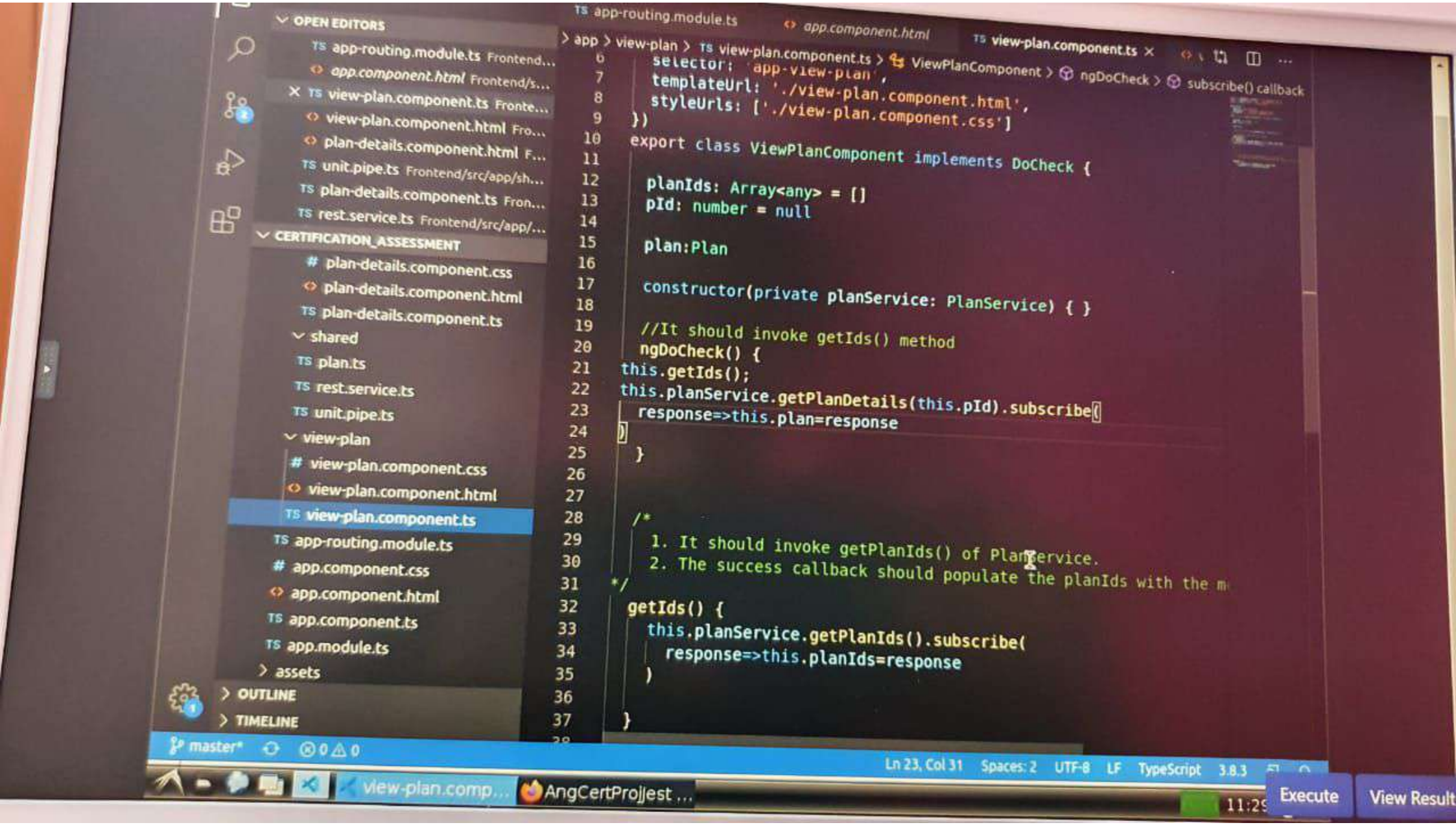
<div *ngIf="errorMessage!=null">
{{errorMessage}}

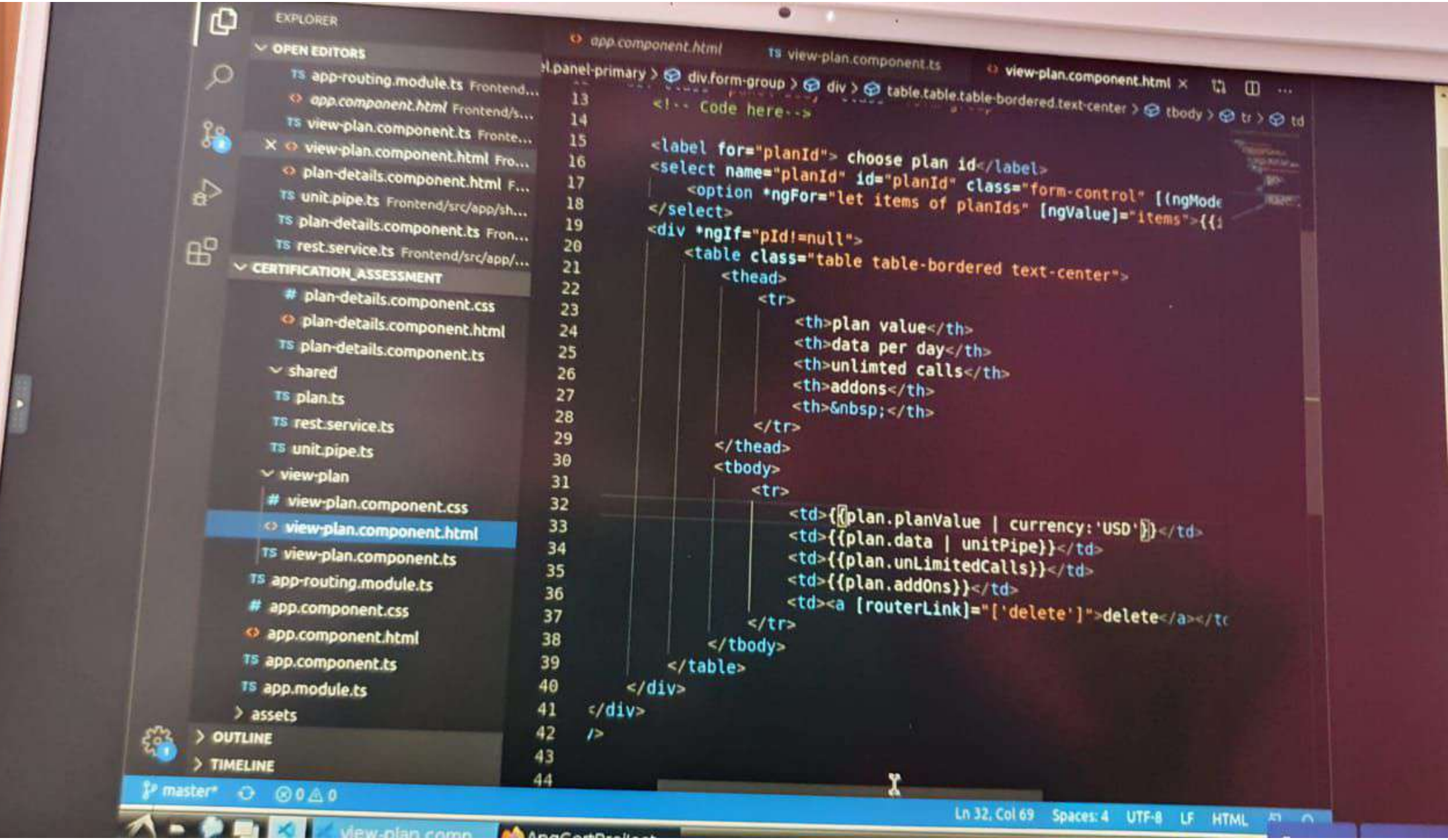
<div *ngIf="successMessage!=null">
{{successMessage}}

</form>

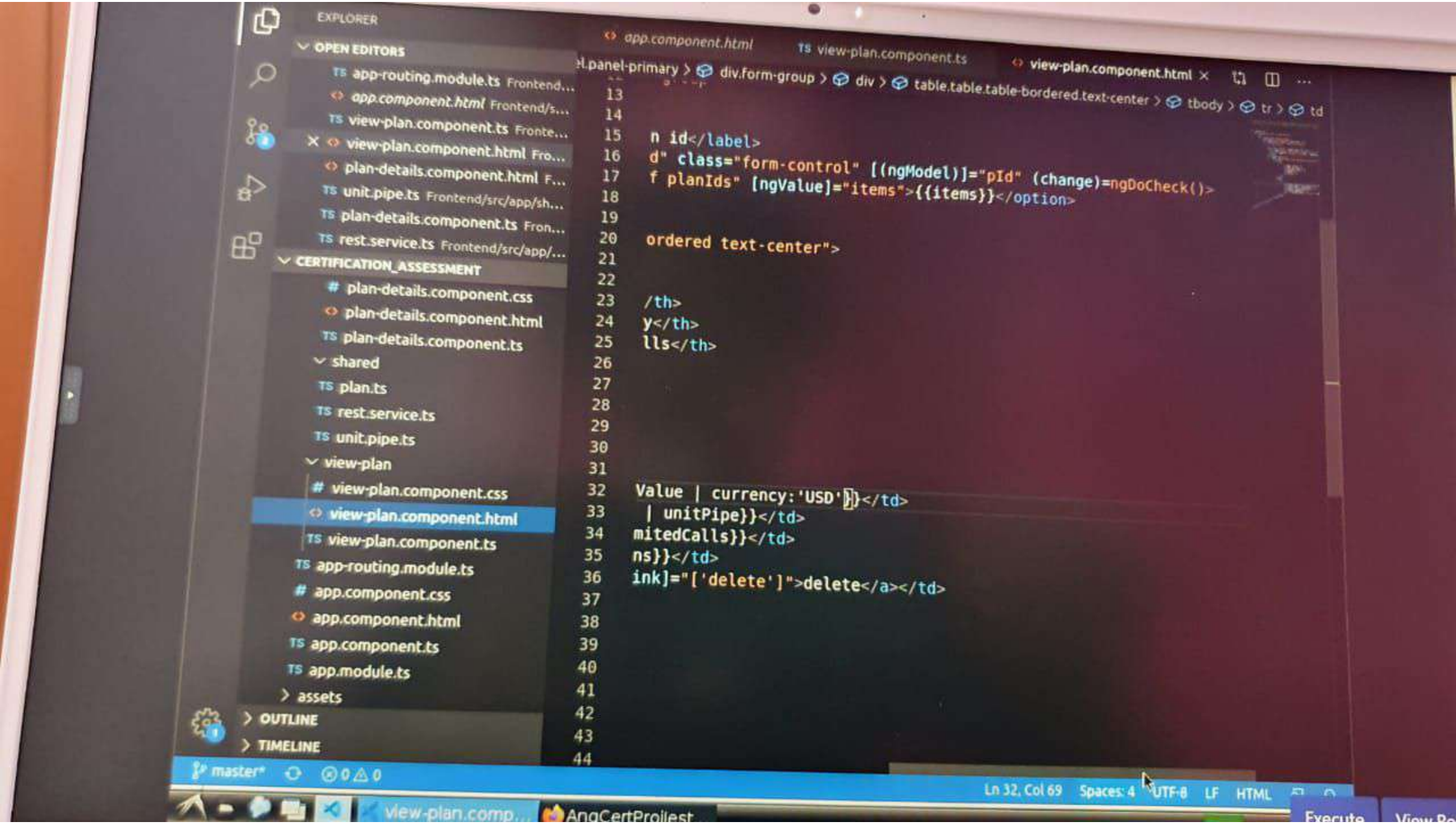
master*
Ln 92, Col 1
Spaces: 4
UTF-8
LF
HTML
add-plan.comp...
Mozilla Firefox
Execute
View Result

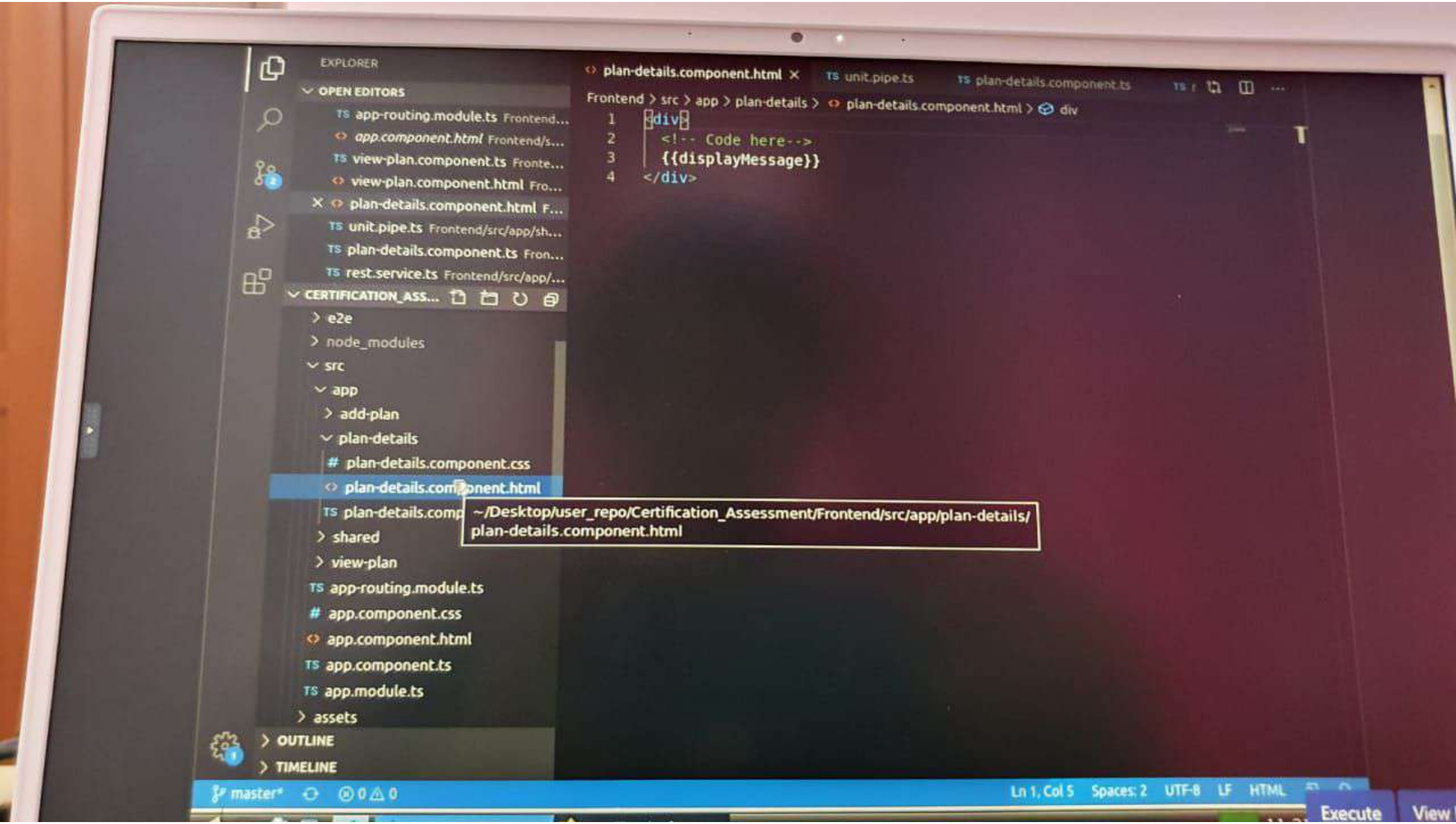

an-details.component.html
add-plan.component.ts
unit.pipe.ts
plan-details.component.ts
rest.service.ts
Frontend > src > app > add-plan > add-plan.component.ts > AddPlanComponent > ngOnInit > planType
constructor(private fb: FormBuilder, private planService: PlanService) { }
ngOnInit() {
//Add specified validators
this.addPlanForm=this.fb.group({
planValue:['',[Validators.required]],
data:['',[Validators.required,Validators.maxLength(200)]],
unLimitedCalls:['',[Validators.required]],
addOns:['',[Validators.required,Validators.pattern("[a-zA-Z]+")]],
planType:['',[Validators.required]]
})
}
/*
It should invoke addPlan() method of PlanService by passing addPlanForm object as a paramet
The success callback should populate the successMessage with the message in response
The error callback should populate the errorMessage with the message in response
*/
addPlan() {
// Code here
this.errorMessage=null;
this.successMessage=null;
this.planService.addPlan(this.addPlanForm.value).subscribe(
(successResponse)=>this.successMessage=successResponse.message,
(response)=>this.errorMessage=response.error.message
)
}
}
master*
0 0
Ln 27, Col 15 (8 selected)
Spaces: 2
UTF-8
LF
TypeScript
3.8.3

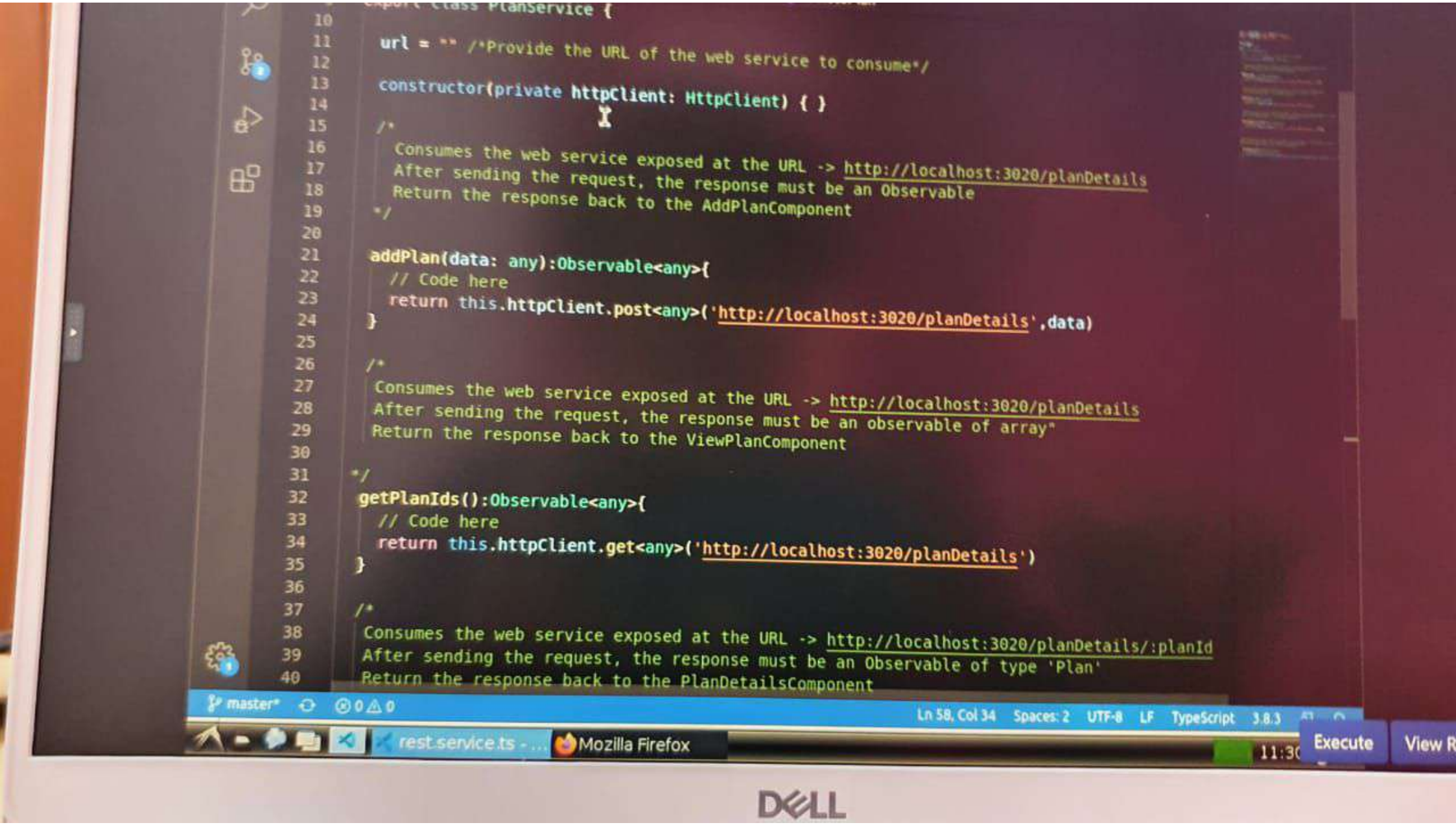

OPEN EDITORS
app-routing.module.ts Frontend...
app.component.html Frontend/s...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASSESSMENT
plan-details.component.css
plan-details.component.html
plan-details.component.ts
shared
plan.ts
rest.service.ts
unit.pipe.ts
view-plan
view-plan.component.css
view-plan.component.html
view-plan.component.ts
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
OUTLINE
TIMELINE
app-routing.module.ts
app.component.html
view-plan.component.ts
app > view-plan > view-plan.component.ts > ViewPlanComponent > ngDoCheck > subscribe() callback
selector: 'app-view-plan',
templateUrl: './view-plan.component.html',
styleUrls: ['./view-plan.component.css']
})
export class ViewPlanComponent implements DoCheck {
planIds: Array<any> = []
pId: number = null
plan:Plan
constructor(private planService: PlanService) { }
//It should invoke getIds() method
ngDoCheck() {
this.getIds();
this.planService.getPlanDetails(this.pId).subscribe(
response=>this.plan=response
)
}
/*
1. It should invoke getPlanIds() of PlanService.
2. The success callback should populate the planIds with the m
*/
getIds() {
this.planService.getPlanIds().subscribe(
response=>this.planIds=response
)
}
master*
Ln 23, Col 31
Spaces: 2
UTF-8
LF
TypeScript
3.8.3
view-plan.comp...
AngCertProjJest ...
Execute
View Result

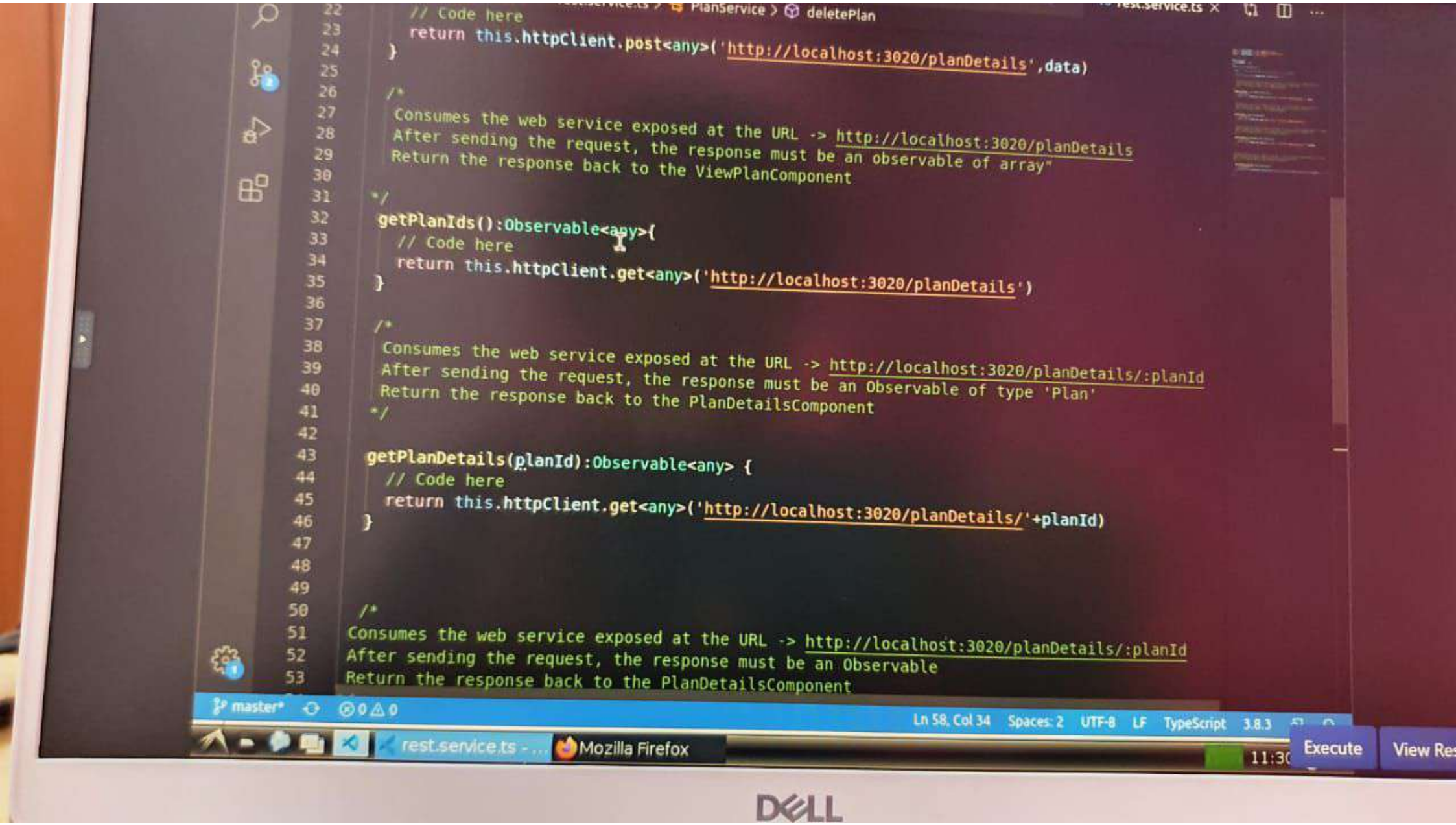

EXPLORER
OPEN EDITORS
app-routing.module.ts Frontend...
app.component.html Frontend/s...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASSESSMENT
plan-details.component.css
plan-details.component.html
plan-details.component.ts
shared
plan.ts
rest.service.ts
unit.pipe.ts
view-plan
view-plan.component.css
view-plan.component.html
view-plan.component.ts
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
OUTLINE
TIMELINE
app.component.html
view-plan.component.ts
view-plan.component.html
div.form-group > div > table.table.table-bordered.text-center > tbody > tr > td
<!-- Code here-->
<label for="planId"> choose plan id</label>
<select name="planId" id="planId" class="form-control" [(ngMode
<option *ngFor="let items of planIds" [ngValue]="items">{{
</select>
<div *ngIf="pId!=null">
<table class="table table-bordered text-center">
<thead>
<tr>
<th>plan value</th>
<th>data per day</th>
<th>unlimted calls</th>
<th>addons</th>
<th> </th>
</tr>
</thead>
<tbody>
<tr>
<td>{{plan.planValue | currency:'USD'}}</td>
<td>{{plan.data | unitPipe}}</td>
<td>{{plan.unLimitedCalls}}</td>
<td>{{plan.addOns}}</td>
<td><a [routerLink]="['delete']">delete</a></t
</tr>
</tbody>
</table>

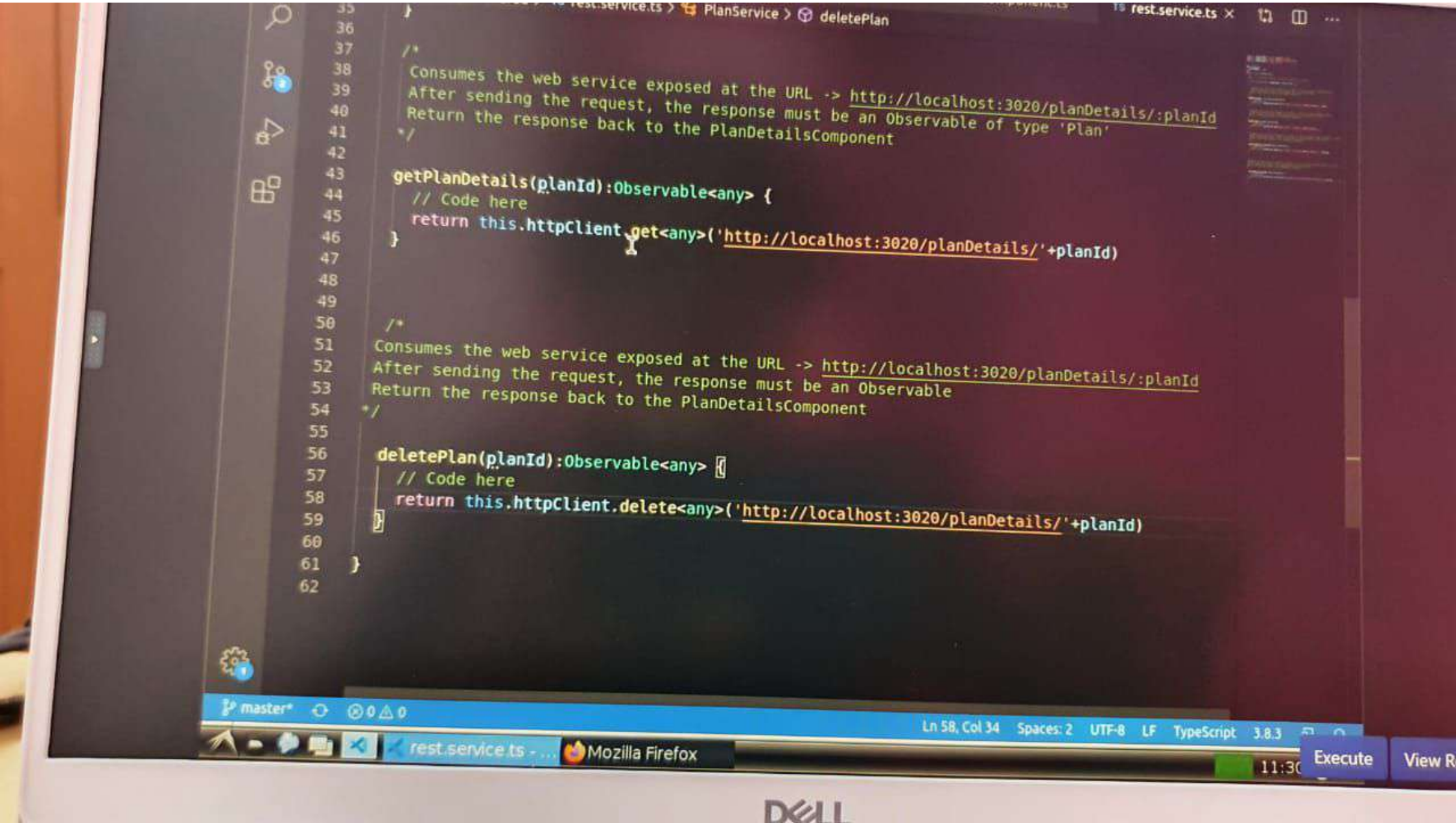
/>
master*
Ln 32, Col 69
Spaces: 4
UTF-8
LF
HTML

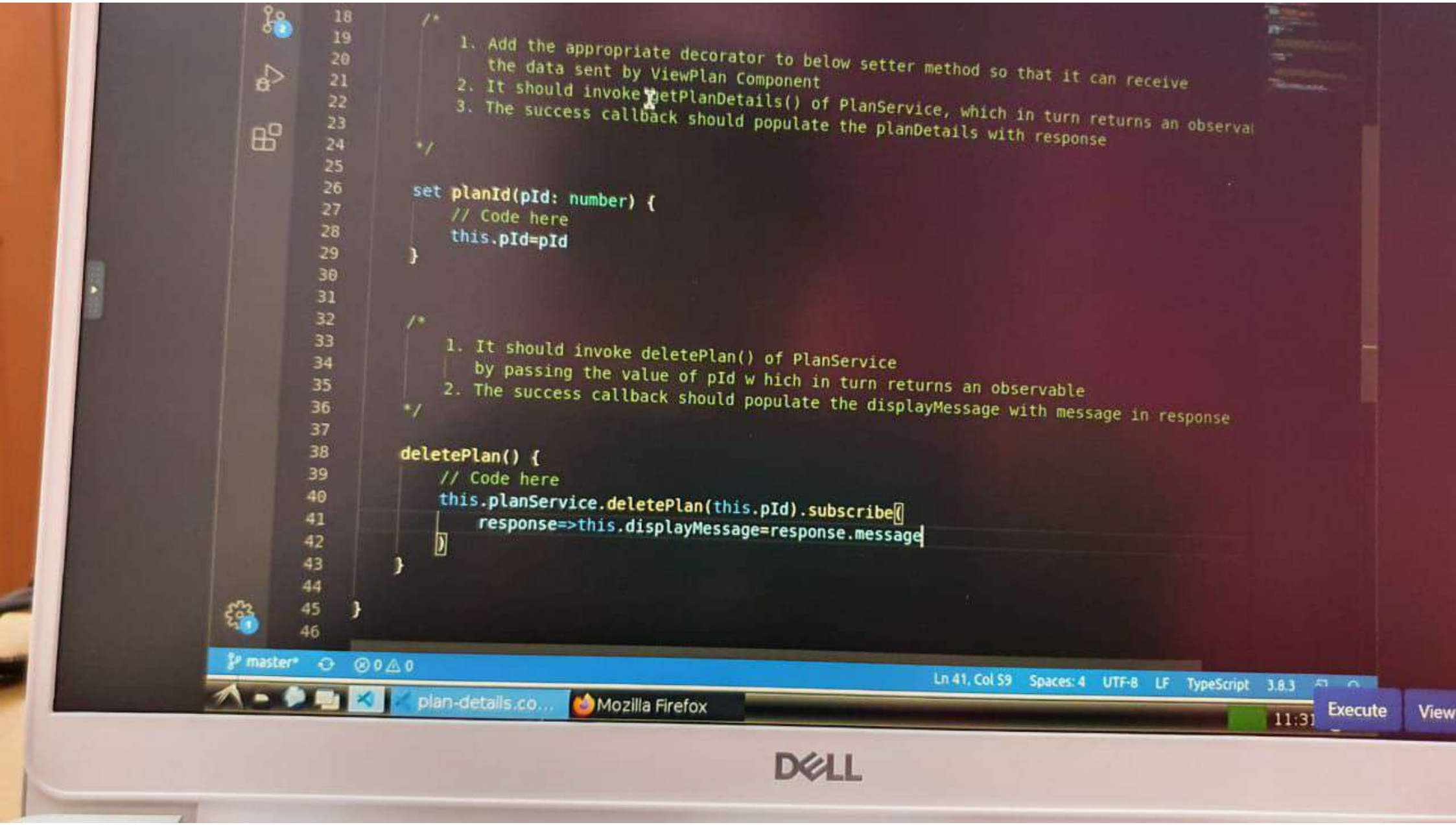

EXPLORER
OPEN EDITORS
app-routing.module.ts Frontend...
app.component.html Frontend/s...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASSESSMENT
plan-details.component.css
plan-details.component.html
plan-details.component.ts
shared
plan.ts
rest.service.ts
unit.pipe.ts
view-plan
view-plan.component.css
view-plan.component.html
view-plan.component.ts
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
OUTLINE
TIMELINE
app.component.html
view-plan.component.ts
view-plan.component.html
div.form-group
div
table.table.table-bordered.text-center
tbody
tr
td
n id</label>
d" class="form-control" [(ngModel)]="pId" (change)=ngDoCheck()>
f planIds" [ngValue]="items">{{items}}</option>
ordered text-center">
/th>
y</th>
lls</th>
Value | currency:'USD'}}</td>
| unitPipe}}</td>
mitedCalls}}</td>
ns}}</td>
ink]="['delete']">delete</a></td>
master*
Ln 32, Col 69
Spaces: 4
UTF-8
LF
HTML
view-plan.comp...
AngCertProjtest
Execute

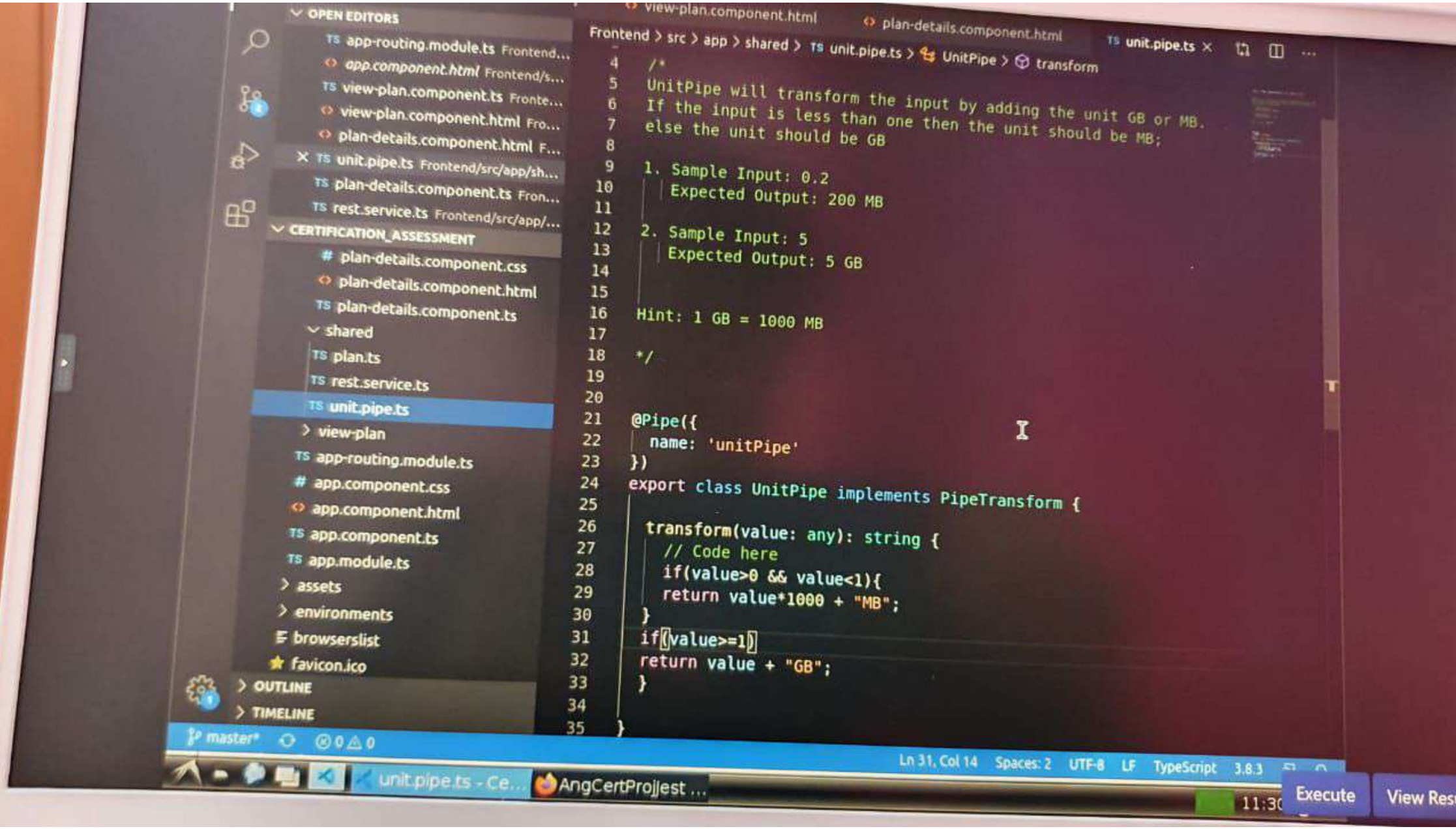

EXPLORER
OPEN EDITORS
app-routing.module.ts Frontend...
app.component.html Frontend/s...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASS...
e2e
node_modules
src
app
add-plan
plan-details
plan-details.component.css
plan-details.component.html
plan-details.comp
shared
view-plan
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
OUTLINE
TIMELINE
plan-details.component.html
unit.pipe.ts
plan-details.component.ts
Frontend > src > app > plan-details > plan-details.component.html > div
1
2 <!-- Code here-->
3 {{displayMessage}}
4
~/Desktop/user_repo/Certification_Assessment/Frontend/src/app/plan-details/plan-details.component.html
master*
Ln 1, Col 5
Spaces: 2
UTF-8
LF
HTML
Execute
View


export class PlanService {
url = "" /*Provide the URL of the web service to consume*/
constructor(private httpClient: HttpClient) { }
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails
After sending the request, the response must be an Observable
Return the response back to the AddPlanComponent
*/
addPlan(data: any):Observable<any>{
// Code here
return this.httpClient.post<any>('http://localhost:3020/planDetails',data)
}
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails
After sending the request, the response must be an observable of array*
Return the response back to the ViewPlanComponent
*/
getPlanIds():Observable<any>{
// Code here
return this.httpClient.get<any>('http://localhost:3020/planDetails')
}
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails/:planId
After sending the request, the response must be an Observable of type 'Plan'
Return the response back to the PlanDetailsComponent
master*
Ln 58, Col 34
Spaces: 2
UTF-8
LF
TypeScript
3.8.3
rest.service.ts - ...
Mozilla Firefox
Execute
View R


// Code here
return this.httpClient.post<any>('http://localhost:3020/planDetails',data)
}
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails
After sending the request, the response must be an observable of array"
Return the response back to the ViewPlanComponent
*/
getPlanIds():Observable<any>{
// Code here
return this.httpClient.get<any>('http://localhost:3020/planDetails')
}
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails/:planId
After sending the request, the response must be an Observable of type 'Plan'
Return the response back to the PlanDetailsComponent
*/
getPlanDetails(planId):Observable<any> {
// Code here
return this.httpClient.get<any>('http://localhost:3020/planDetails/'+planId)
}
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails/:planId
After sending the request, the response must be an Observable
Return the response back to the PlanDetailsComponent
master*
Ln 58, Col 34
Spaces: 2
UTF-8
LF
TypeScript
3.8.3
rest.service.ts - ...
Mozilla Firefox
Execute
View Res


rest.service.ts
PlanService > deletePlan
rest.service.ts
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails/:planId
After sending the request, the response must be an Observable of type 'Plan'
Return the response back to the PlanDetailsComponent
*/
getPlanDetails(planId):Observable<any> {
// Code here
return this.httpClient.get<any>('http://localhost:3020/planDetails/'+planId)
}
/*
Consumes the web service exposed at the URL -> http://localhost:3020/planDetails/:planId
After sending the request, the response must be an Observable
Return the response back to the PlanDetailsComponent
*/
deletePlan(planId):Observable<any> {
// Code here
return this.httpClient.delete<any>('http://localhost:3020/planDetails/'+planId)
}
}
master*
Ln 58, Col 34
Spaces: 2
UTF-8
LF
TypeScript
3.8.3
rest.service.ts - ...
Mozilla Firefox
Execute
View R


/*
1. Add the appropriate decorator to below setter method so that it can receive
the data sent by ViewPlan Component
2. It should invoke getPlanDetails() of PlanService, which in turn returns an observal
3. The success callback should populate the planDetails with response
*/
set planId(pId: number) {
// Code here
this.pId=pId
}
/*
1. It should invoke deletePlan() of PlanService
by passing the value of pId w hich in turn returns an observable
2. The success callback should populate the displayMessage with message in response
*/
deletePlan() {
// Code here
this.planService.deletePlan(this.pId).subscribe(
response=>this.displayMessage=response.message
)
}
}
master*
Ln 41, Col 59
Spaces: 4
UTF-8
LF
TypeScript
3.8.3
plan-details.co...
Mozilla Firefox
Execute
View
DELL


OPEN EDITORS
app-routing.module.ts Frontend...
app.component.html Frontend/s...
view-plan.component.ts Fronte...
view-plan.component.html Fro...
plan-details.component.html F...
unit.pipe.ts Frontend/src/app/sh...
plan-details.component.ts Fron...
rest.service.ts Frontend/src/app/...
CERTIFICATION_ASSESSMENT
plan-details.component.css
plan-details.component.html
plan-details.component.ts
shared
plan.ts
rest.service.ts
unit.pipe.ts
view-plan
app-routing.module.ts
app.component.css
app.component.html
app.component.ts
app.module.ts
assets
environments
browserslist
favicon.ico
OUTLINE
TIMELINE
view-plan.component.html
plan-details.component.html
unit.pipe.ts
Frontend > src > app > shared > unit.pipe.ts > UnitPipe > transform
/*
UnitPipe will transform the input by adding the unit GB or MB.
If the input is less than one then the unit should be MB;
else the unit should be GB
1. Sample Input: 0.2
Expected Output: 200 MB
2. Sample Input: 5
Expected Output: 5 GB
Hint: 1 GB = 1000 MB
*/
@Pipe({
name: 'unitPipe'
})
export class UnitPipe implements PipeTransform {
transform(value: any): string {
// Code here
if(value>0 && value<1){
return value*1000 + "MB";
}
if(value>=1)
return value + "GB";
}
}
master*
Ln 31, Col 14
Spaces: 2
UTF-8
LF
TypeScript
3.8.3
unit.pipe.ts - Ce...
AngCertProjJest ...
Execute
View Res