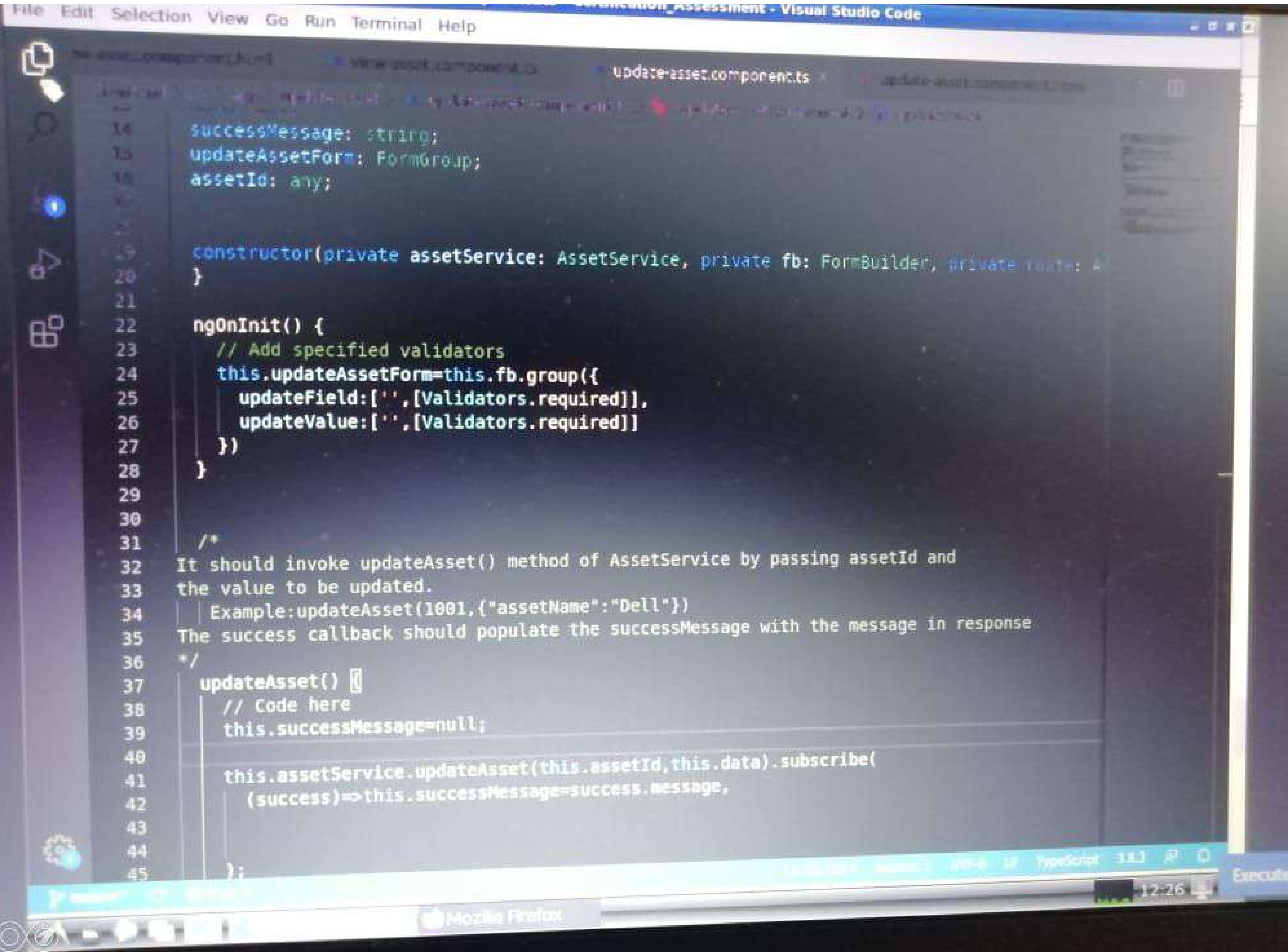
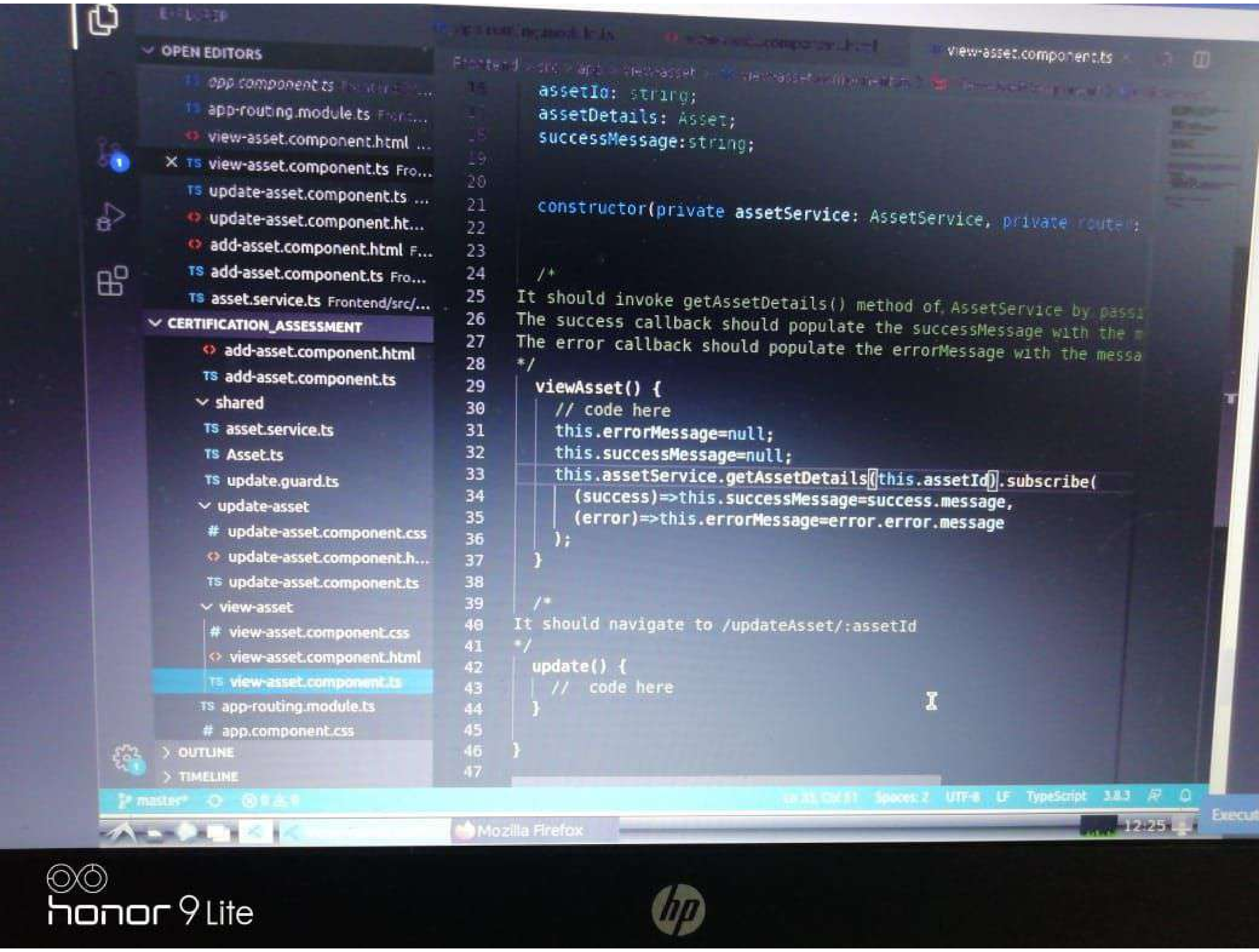
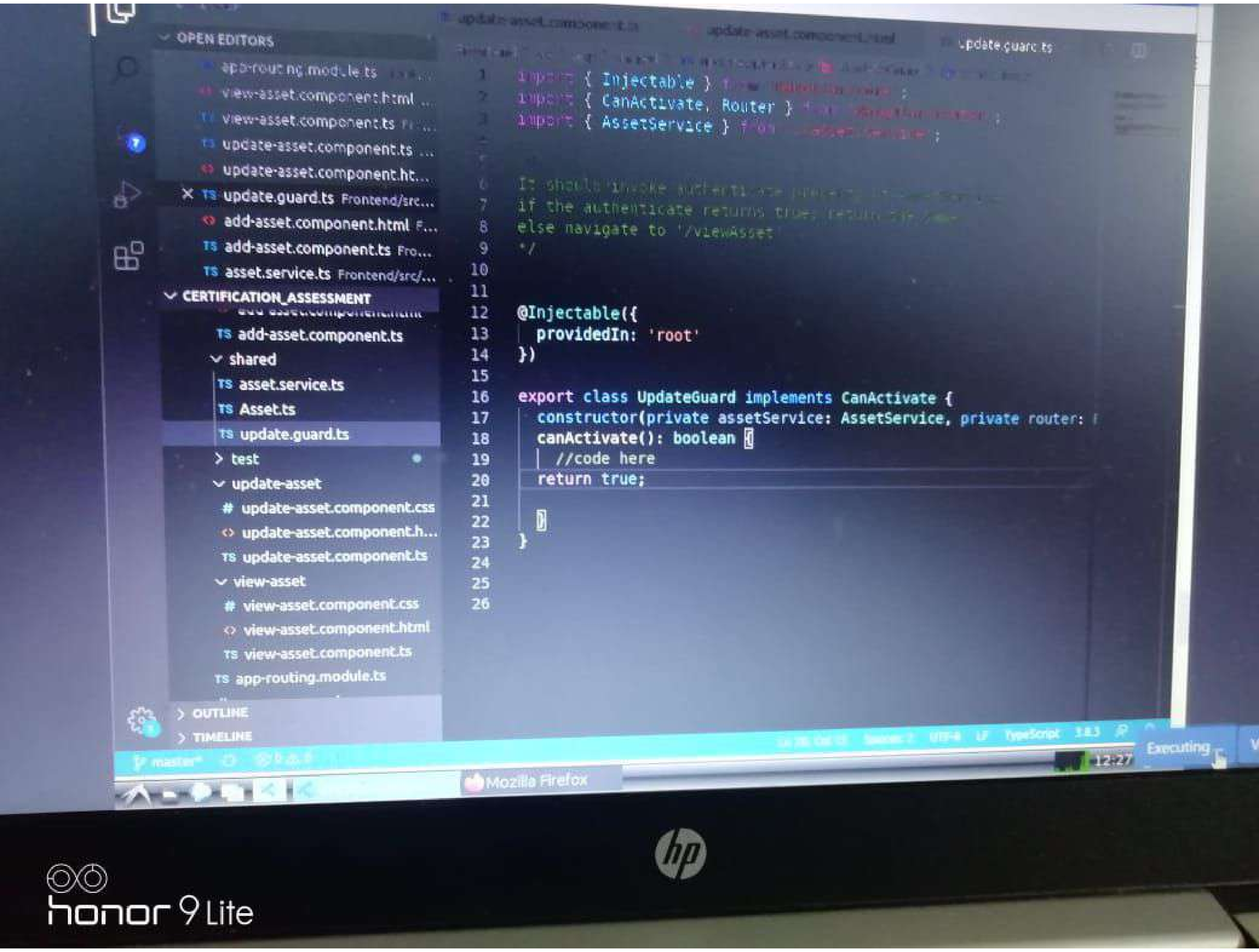

File Edit Selection View Go Run Terminal Help
_Assessment - Visual Studio Code
update-asset.component.ts
TypeScript
12:26
Mozilla Firefox
honor 9 Lite

```
  successMessage: string;
  updateAssetForm: FormGroup;
  assetId: any;


  constructor(private assetService: AssetService, private fb: FormBuilder, private route: A
  }

  ngOnInit() {
    // Add specified validators
    this.updateAssetForm=this.fb.group({
      updateField:['',[Validators.required]],
      updateValue:['',[Validators.required]]
    })
  }


  /*
It should invoke updateAsset() method of AssetService by passing assetId and
the value to be updated.
   Example:updateAsset(1001,{"assetName":"Dell"})
The success callback should populate the successMessage with the message in response
*/
  updateAsset() {
    // Code here
    this.successMessage=null;

    this.assetService.updateAsset(this.assetId,this.data).subscribe(
      (success)=>this.successMessage=success.message,


    );
```



```
  assetId: string;
  assetDetails: Asset;
  successMessage:string;


  constructor(private assetService: AssetService, private route


  /*
It should invoke getAssetDetails() method of AssetService by passi
The success callback should populate the successMessage with the m
The error callback should populate the errorMessage with the messa
*/
  viewAsset() {
    // code here
    this.errorMessage=null;
    this.successMessage=null;
    this.assetService.getAssetDetails(this.assetId).subscribe(
      (success)=>this.successMessage=success.message,
      (error)=>this.errorMessage=error.error.message
    );
  }

  /*
It should navigate to /updateAsset/:assetId
*/
  update() {
    //  code here
  }

}
```



OPEN EDITORS

- app-routing.module.ts
- view-asset.component.html
- view-asset.component.ts
- update-asset.component.ts
- update-asset.component.ht...
- × update.guard.ts Frontend/src...
- add-asset.component.html F...
- add-asset.component.ts Fro...
- asset.service.ts Frontend/src/...

CERTIFICATION_ASSESSMENT

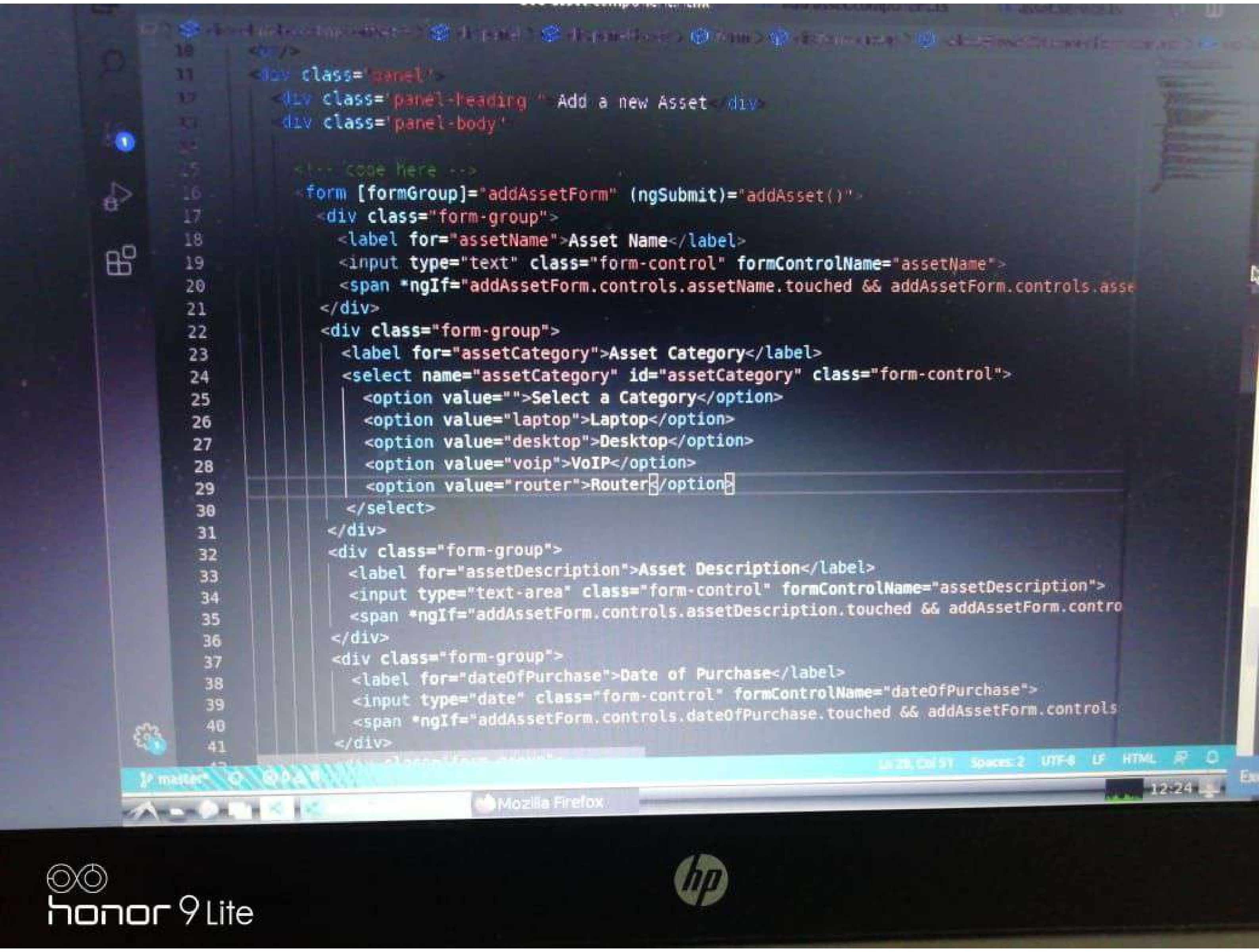
- add-asset.component.ts
- shared
  - asset.service.ts
  - Asset.ts
  - update.guard.ts
- test
- update-asset
  - update-asset.component.css
  - update-asset.component.h...
  - update-asset.component.ts
- view-asset
  - view-asset.component.css
  - view-asset.component.html
  - view-asset.component.ts
- app-routing.module.ts

OUTLINE

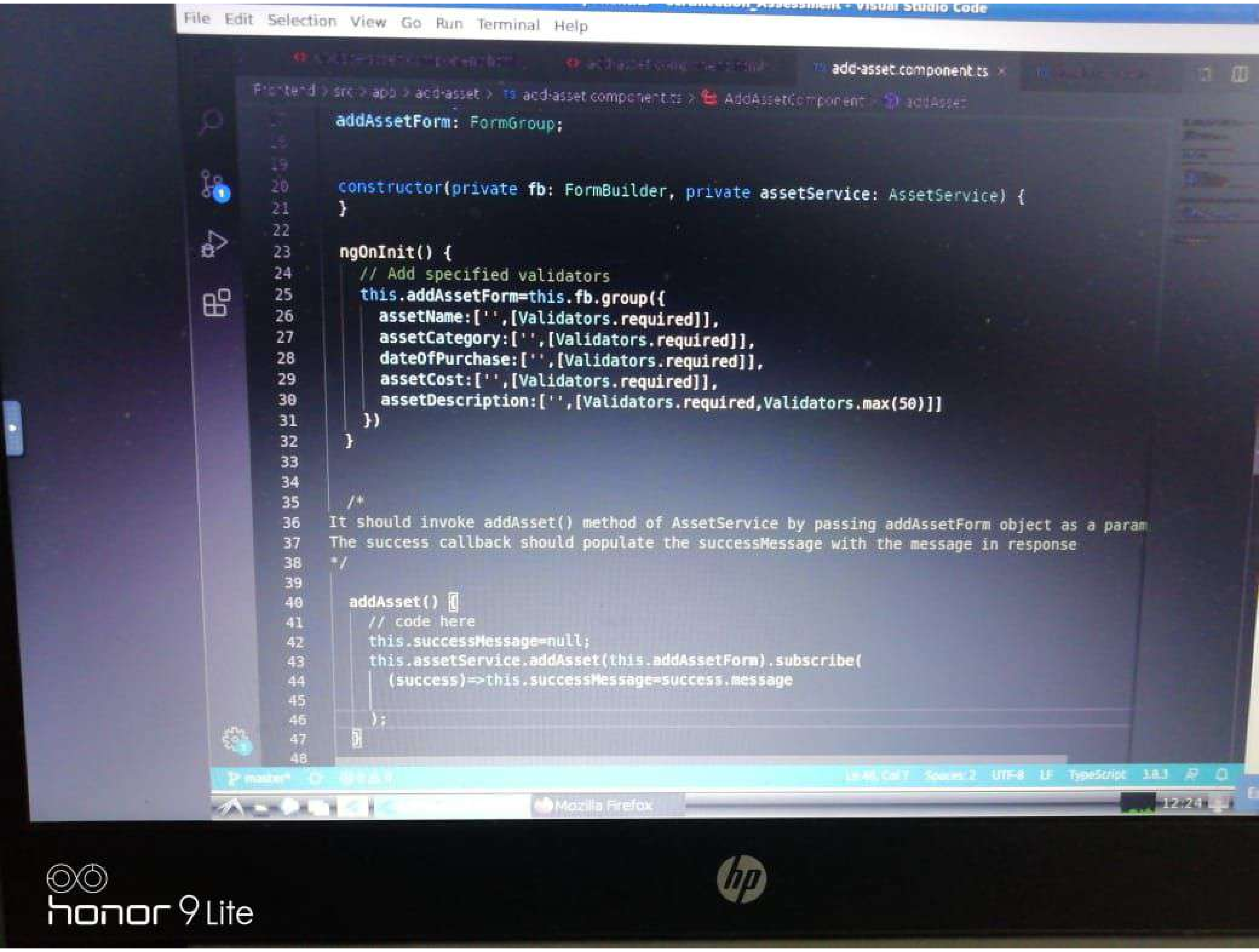
TIMELINE

update.guard.ts

```
import { Injectable } from [illegible];
import { CanActivate, Router } from [illegible];
import { AssetService } from [illegible];


It should invoke authenticate [illegible]
if the authenticate returns true, [illegible]
else navigate to '/viewAsset'
*/


@Injectable({
  providedIn: 'root'
})

export class UpdateGuard implements CanActivate {
  constructor(private assetService: AssetService, private router: 
  canActivate(): boolean {
    //code here
  return true;

  }
}
```

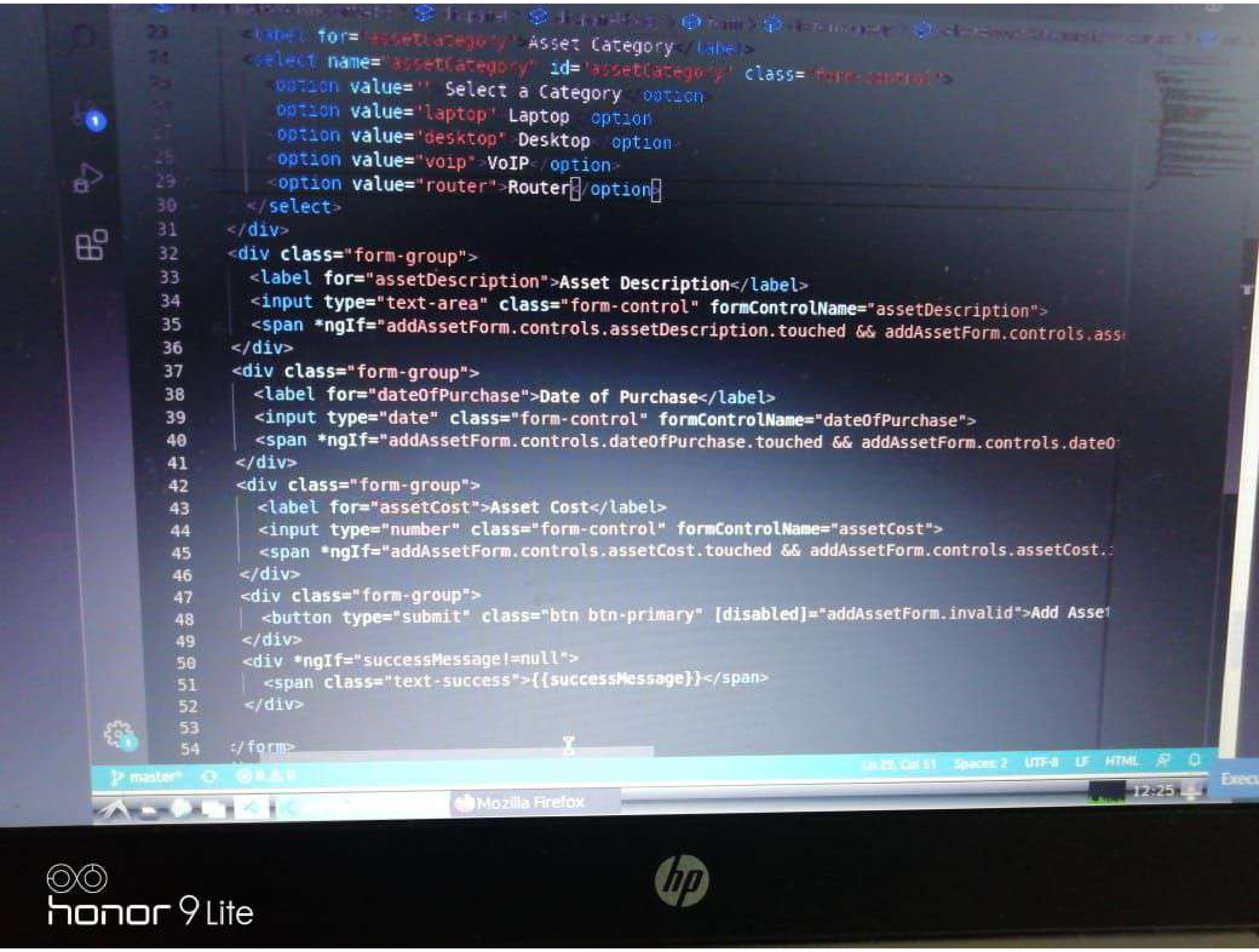


```html
<br/>
<div class='panel'>
  <div class='panel-heading'>Add a new Asset</div>
  <div class='panel-body'>

    <!-- code here -->
    <form [formGroup]="addAssetForm" (ngSubmit)="addAsset()">
      <div class="form-group">
        <label for="assetName">Asset Name</label>
        <input type="text" class="form-control" formControlName="assetName">
        <span *ngIf="addAssetForm.controls.assetName.touched && addAssetForm.controls.asse
      </div>
      <div class="form-group">
        <label for="assetCategory">Asset Category</label>
        <select name="assetCategory" id="assetCategory" class="form-control">
          <option value="">Select a Category</option>
          <option value="laptop">Laptop</option>
          <option value="desktop">Desktop</option>
          <option value="voip">VoIP</option>
          <option value="router">Router</option>
        </select>
      </div>
      <div class="form-group">
        <label for="assetDescription">Asset Description</label>
        <input type="text-area" class="form-control" formControlName="assetDescription">
        <span *ngIf="addAssetForm.controls.assetDescription.touched && addAssetForm.contro
      </div>
      <div class="form-group">
        <label for="dateOfPurchase">Date of Purchase</label>
        <input type="date" class="form-control" formControlName="dateOfPurchase">
        <span *ngIf="addAssetForm.controls.dateOfPurchase.touched && addAssetForm.controls
      </div>
```

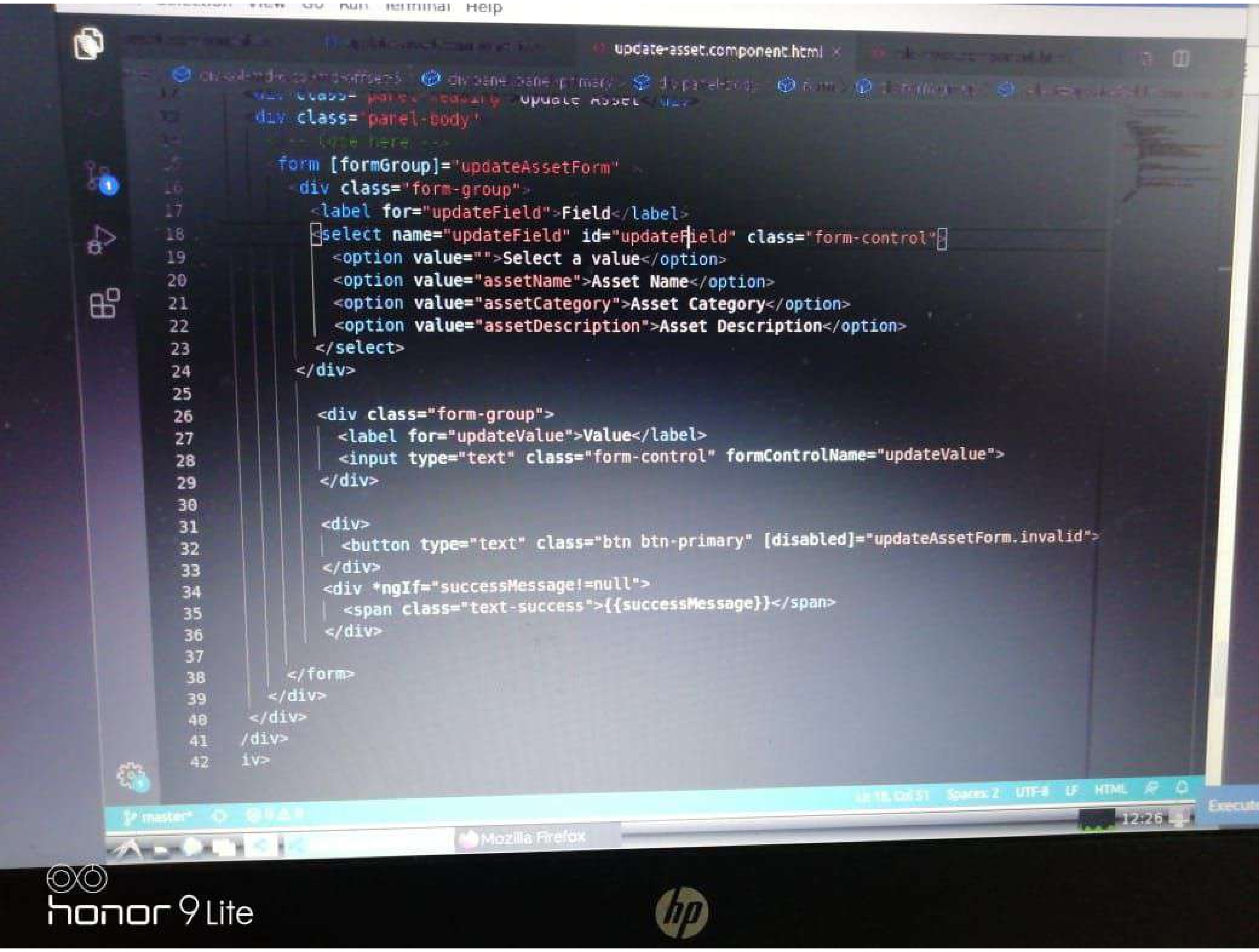


Visual Studio Code
File Edit Selection View Go Run Terminal Help
add-asset.component.ts
Frontend > src > app > add-asset > add-asset.component.ts > AddAssetComponent > addAsset
master*
Spaces: 2 UTF-8 LF TypeScript
Mozilla Firefox
12:24
honor 9 Lite

```
  addAssetForm: FormGroup;


  constructor(private fb: FormBuilder, private assetService: AssetService) {
  }

  ngOnInit() {
    // Add specified validators
    this.addAssetForm=this.fb.group({
      assetName:['',[Validators.required]],
      assetCategory:['',[Validators.required]],
      dateOfPurchase:['',[Validators.required]],
      assetCost:['',[Validators.required]],
      assetDescription:['',[Validators.required,Validators.max(50)]]
    })
  }


   /*
It should invoke addAsset() method of AssetService by passing addAssetForm object as a param
The success callback should populate the successMessage with the message in response
*/

  addAsset() {
    // code here
    this.successMessage=null;
    this.assetService.addAsset(this.addAssetForm).subscribe(
      (success)=>this.successMessage=success.message

    );
  }
```

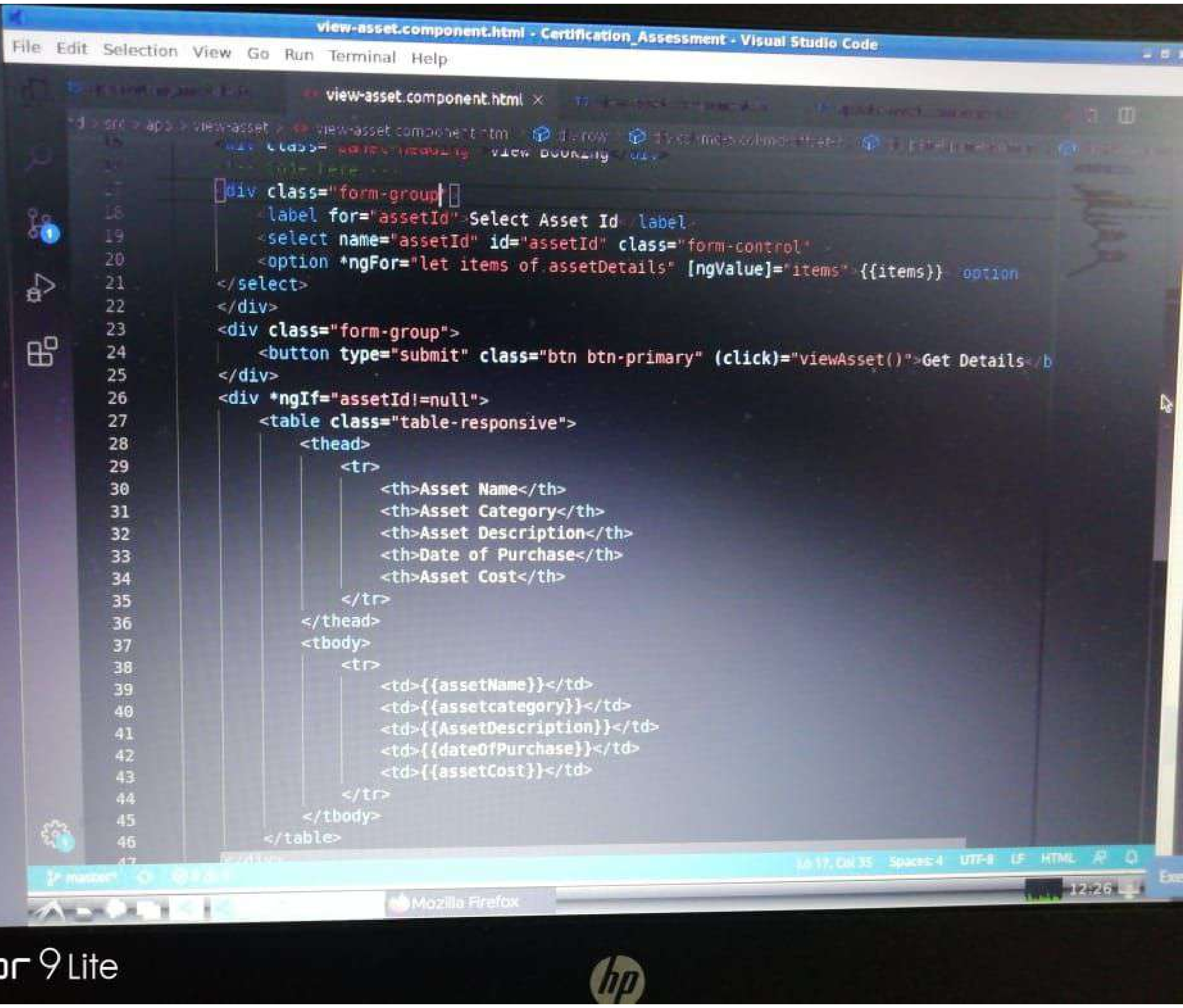


```
<label for="assetCategory">Asset Category</label>
<select name="assetCategory" id="assetCategory" class="form-control">
  <option value="">Select a Category</option>
  <option value="laptop">Laptop</option>
  <option value="desktop">Desktop</option>
  <option value="voip">VoIP</option>
  <option value="router">Router</option>
</select>
</div>
<div class="form-group">
  <label for="assetDescription">Asset Description</label>
  <input type="text-area" class="form-control" formControlName="assetDescription">
  <span *ngIf="addAssetForm.controls.assetDescription.touched && addAssetForm.controls.ass
</div>
<div class="form-group">
  <label for="dateOfPurchase">Date of Purchase</label>
  <input type="date" class="form-control" formControlName="dateOfPurchase">
  <span *ngIf="addAssetForm.controls.dateOfPurchase.touched && addAssetForm.controls.dateO
</div>
<div class="form-group">
  <label for="assetCost">Asset Cost</label>
  <input type="number" class="form-control" formControlName="assetCost">
  <span *ngIf="addAssetForm.controls.assetCost.touched && addAssetForm.controls.assetCost.
</div>
<div class="form-group">
  <button type="submit" class="btn btn-primary" [disabled]="addAssetForm.invalid">Add Asse
</div>
<div *ngIf="successMessage!=null">
  <span class="text-success">{{successMessage}}</span>
</div>

</form>
```

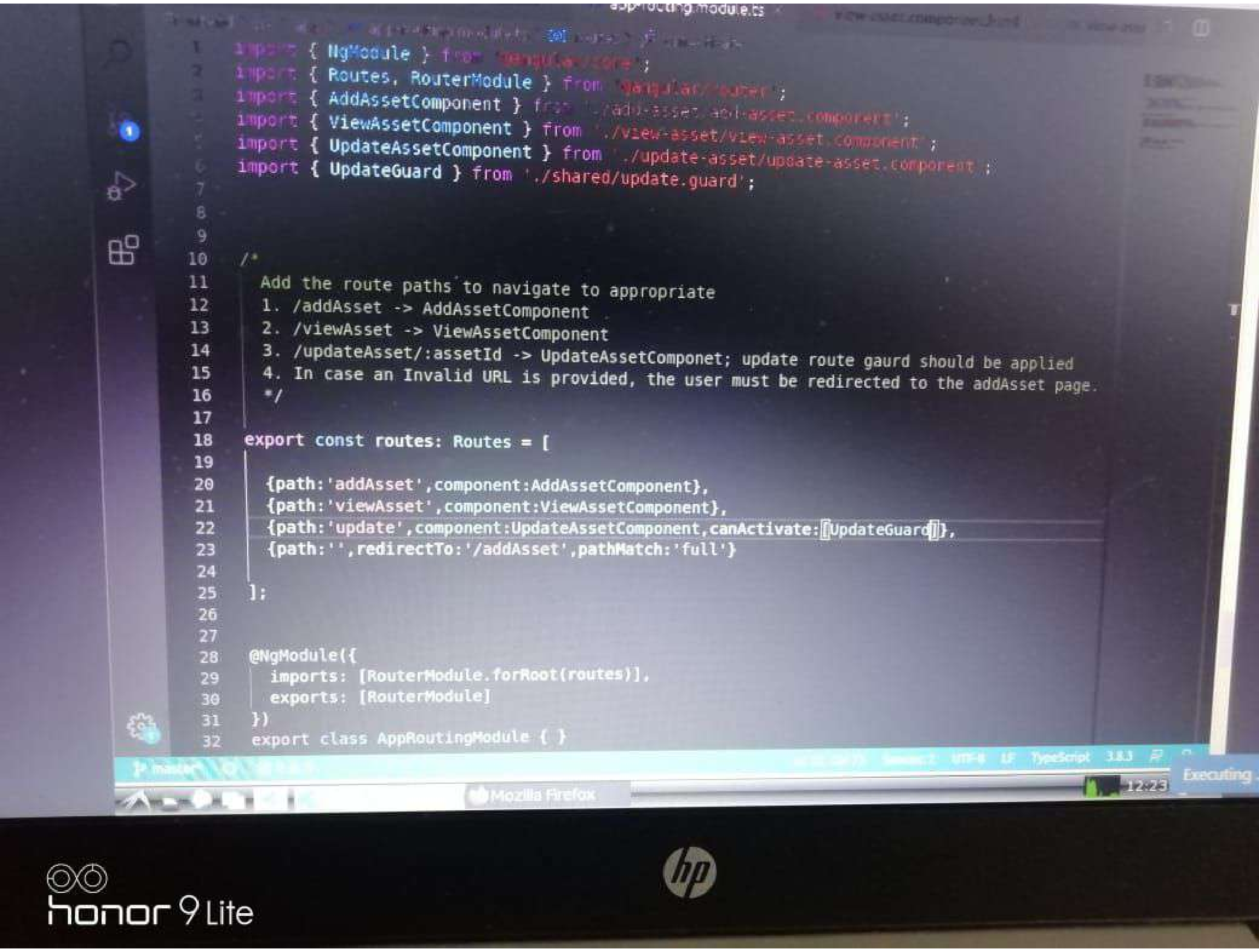


Selection View Go Run Terminal Help
update-asset.component.html
master*
Mozilla Firefox
12:26
honor 9 Lite

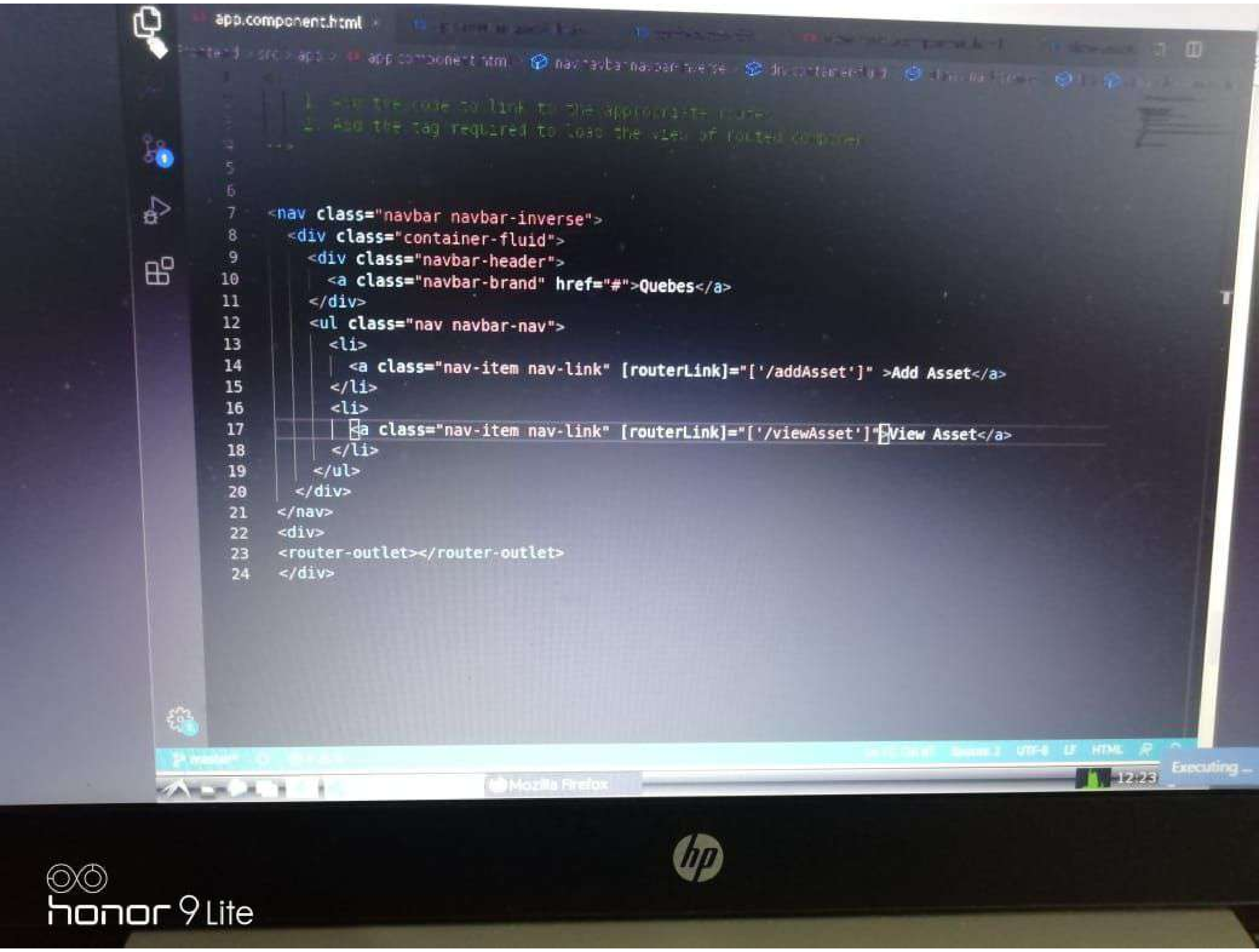
```
        <div class="panel-heading">Update Asset</div>
        div class='panel-body'
          <!-- Code here -->
          form [formGroup]='updateAssetForm'>
            div class='form-group'>
              <label for="updateField">Field</label>
              <select name="updateField" id="updateField" class="form-control">
                <option value="">Select a value</option>
                <option value="assetName">Asset Name</option>
                <option value="assetCategory">Asset Category</option>
                <option value="assetDescription">Asset Description</option>
              </select>
            </div>

              <div class="form-group">
                <label for="updateValue">Value</label>
                <input type="text" class="form-control" formControlName="updateValue">
              </div>

              <div>
                <button type="text" class="btn btn-primary" [disabled]="updateAssetForm.invalid">
              </div>
              <div *ngIf="successMessage!=null">
                <span class="text-success">{{successMessage}}</span>
              </div>

          </form>
        </div>
      </div>
    /div>
    iv>
```


view-asset.component.html - Certification_Assessment - Visual Studio Code
File
Edit
Selection
View
Go
Run
Terminal
Help
view-asset.component.html
src > app > view-asset > view-asset.component.htm
Ln 17, Col 35
Spaces: 4
UTF-8
LF
HTML
master*
Mozilla Firefox
12:26
honor 9 Lite
hp

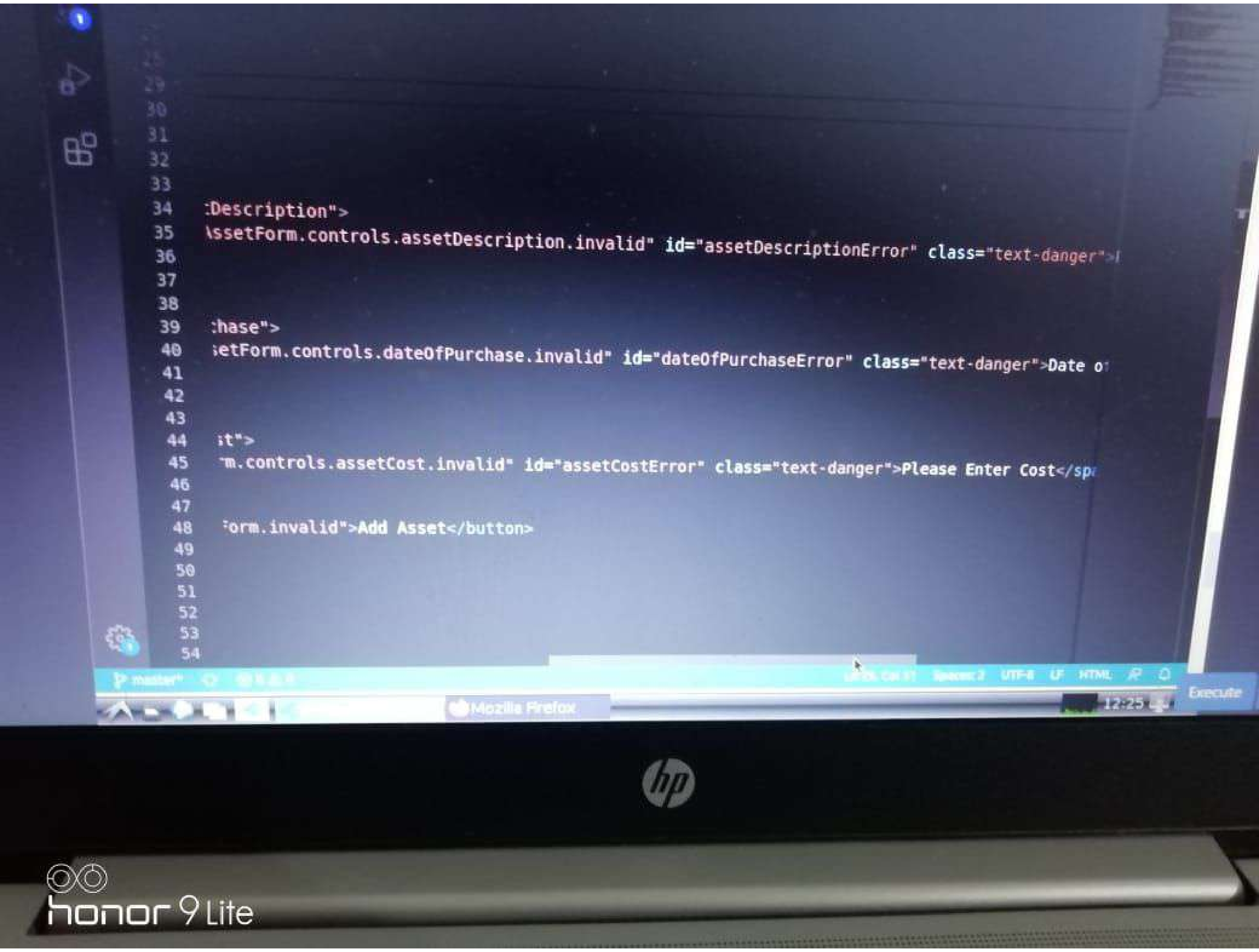
```
    class="[illegible]" View Booking</div>

<div class="form-group">
    <label for="assetId">Select Asset Id</label>
    <select name="assetId" id="assetId" class="form-control">
    <option *ngFor="let items of assetDetails" [ngValue]="items">{{items}}</option>
</select>
</div>
<div class="form-group">
    <button type="submit" class="btn btn-primary" (click)="viewAsset()">Get Details</b
</div>
<div *ngIf="assetId!=null">
    <table class="table-responsive">
        <thead>
            <tr>
                <th>Asset Name</th>
                <th>Asset Category</th>
                <th>Asset Description</th>
                <th>Date of Purchase</th>
                <th>Asset Cost</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>{{assetName}}</td>
                <td>{{assetcategory}}</td>
                <td>{{AssetDescription}}</td>
                <td>{{dateOfPurchase}}</td>
                <td>{{assetCost}}</td>
            </tr>
        </tbody>
    </table>
```



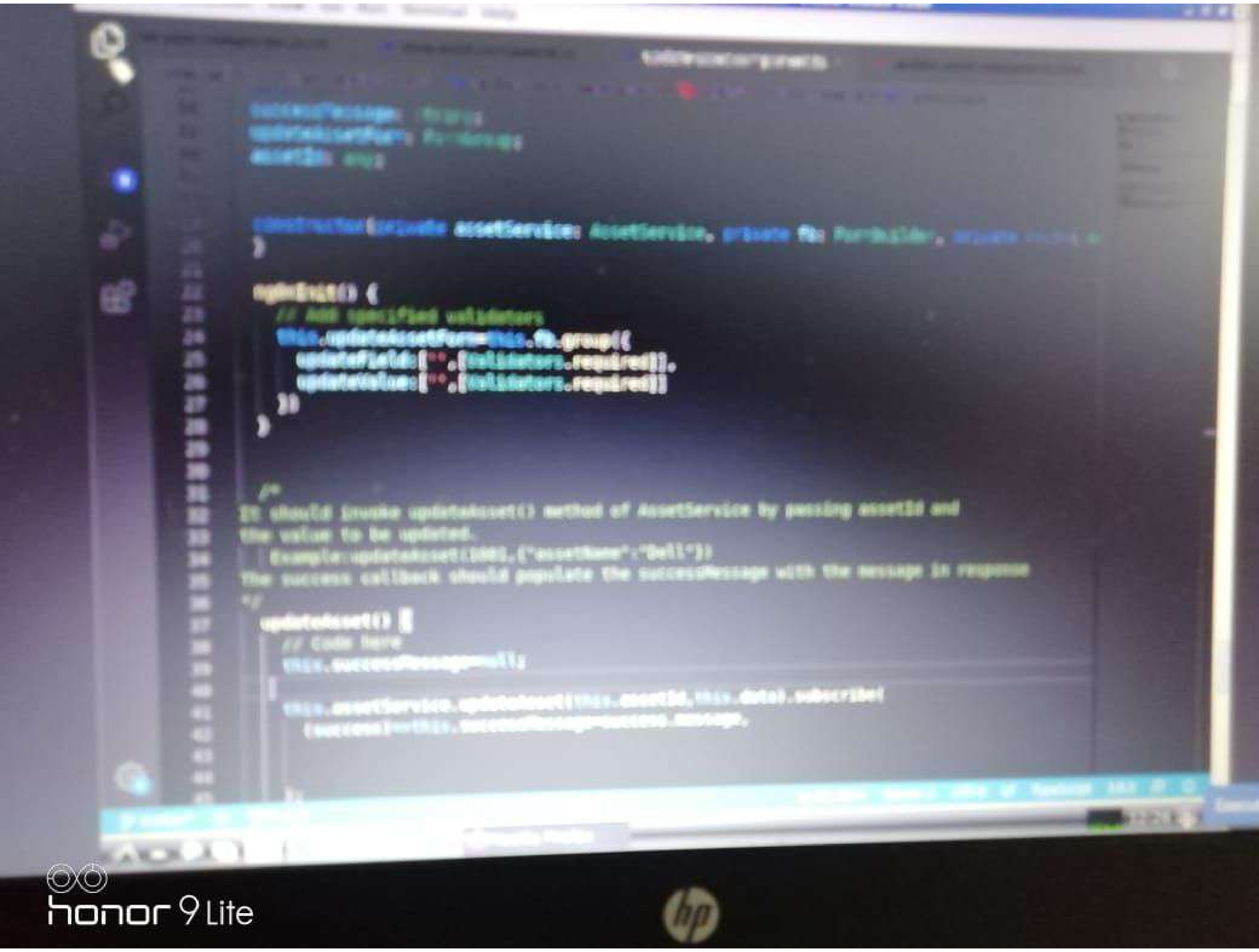
```typescript
import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router';
import { AddAssetComponent } from './add-asset/add-asset.component';
import { ViewAssetComponent } from './view-asset/view-asset.component';
import { UpdateAssetComponent } from './update-asset/update-asset.component';
import { UpdateGuard } from './shared/update.guard';



/*
  Add the route paths to navigate to appropriate
  1. /addAsset -> AddAssetComponent
  2. /viewAsset -> ViewAssetComponent
  3. /updateAsset/:assetId -> UpdateAssetComponet; update route gaurd should be applied
  4. In case an Invalid URL is provided, the user must be redirected to the addAsset page.
  */

export const routes: Routes = [

  {path:'addAsset',component:AddAssetComponent},
  {path:'viewAsset',component:ViewAssetComponent},
  {path:'update',component:UpdateAssetComponent,canActivate:[UpdateGuard]},
  {path:'',redirectTo:'/addAsset',pathMatch:'full'}

];


@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```



```html
<!--
    1. Add the code to link to the appropriate route
    2. Add the tag required to load the view of routed component
-->


<nav class="navbar navbar-inverse">
  <div class="container-fluid">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">Quebes</a>
    </div>
    <ul class="nav navbar-nav">
      <li>
        <a class="nav-item nav-link" [routerLink]="['/addAsset']" >Add Asset</a>
      </li>
      <li>
        <a class="nav-item nav-link" [routerLink]="['/viewAsset']">View Asset</a>
      </li>
    </ul>
  </div>
</nav>
<div>
<router-outlet></router-outlet>
</div>
```



```
:Description">
\ssetForm.controls.assetDescription.invalid" id="assetDescriptionError" class="text-danger">



:hase">
;etForm.controls.dateOfPurchase.invalid" id="dateOfPurchaseError" class="text-danger">Date o



;t">
'm.controls.assetCost.invalid" id="assetCostError" class="text-danger">Please Enter Cost</sp


'orm.invalid">Add Asset</button>
```



```
successMessage: String;
updateAssetForm: FormGroup;
assetId: any;


constructor(private assetService: AssetService, private fb: FormBuilder, private [illegible]
}

ngOnInit() {
  // Add specified validators
  this.updateAssetForm=this.fb.group({
    updateField:["",[Validators.required]],
    updateValue:["",[Validators.required]]
  })
}


 /*
It should invoke updateAsset() method of AssetService by passing assetId and
the value to be updated.
  Example:updateAsset(1001,{"assetName":"Bell"})
The success callback should populate the successMessage with the message in response
*/
 updateAsset() {
   // Code here
   this.successMessage=null;

   this.assetService.updateAsset(this.assetId,this.data).subscribe(
     (success)=>this.successMessage=success.message,
```