Api Project

1-Create user and password for SQl if it not exist.

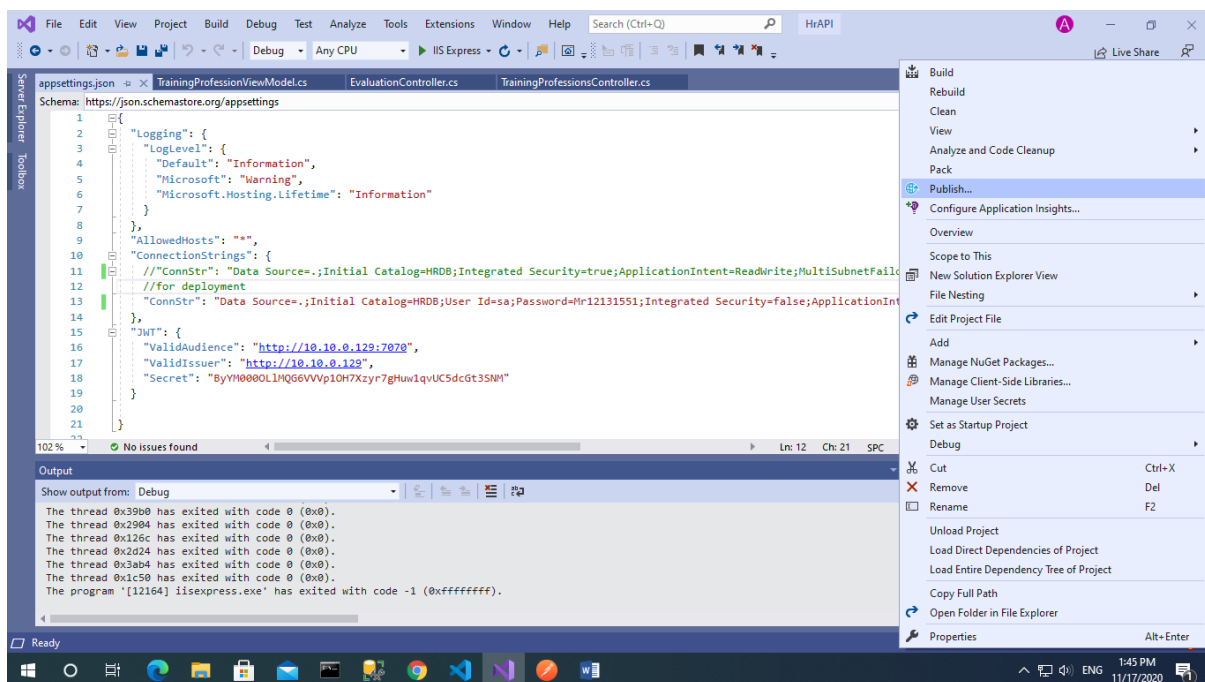2-In Api project in change connection string in "appsettings.json"

Form

```
"ConnStr": "Data Source=.;Initial Catalog=HRDB;Integrated Security=true;ApplicationIntent=ReadWrite;MultiSubnetFailover=False"
```
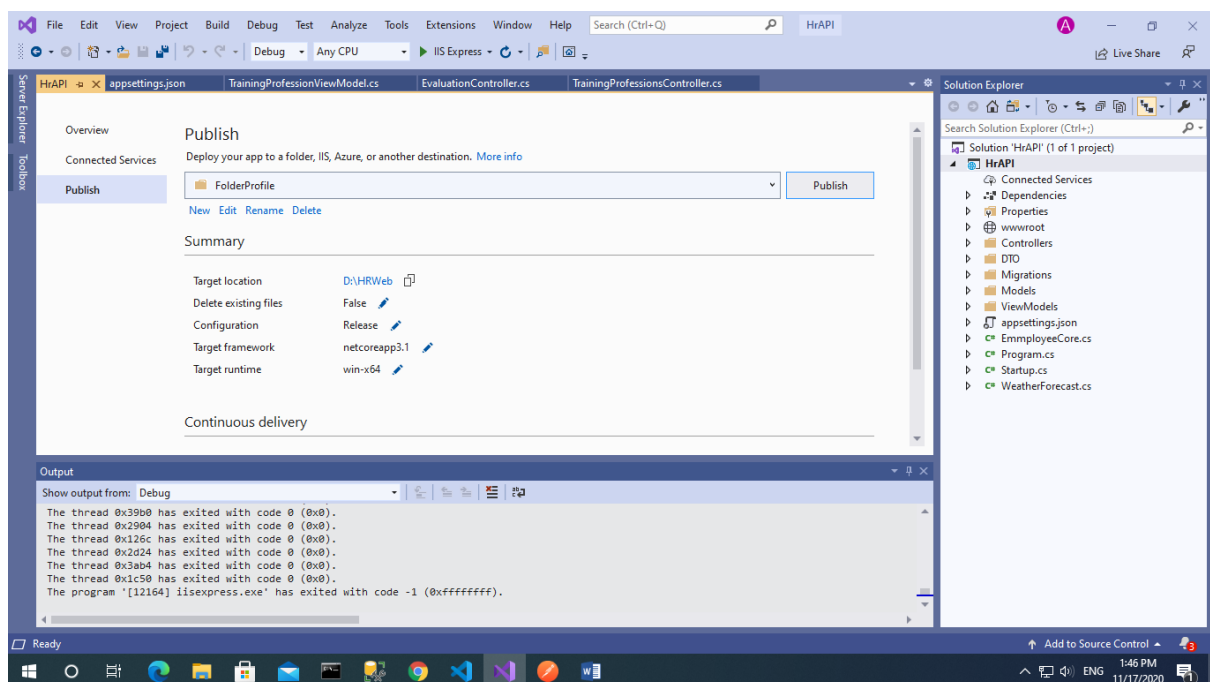
To

```
"ConnStr": "Data Source=.; Initial Catalog=HRDB;User Id=sa;Password=Mr12131551;Integrated Security=false;ApplicationIntent=ReadWrite;MultiSubnetFailover=False"
```

3-create folder in your disk and give it name

4- Right click on your project name and select Publish

5-In target location select your created folder in your disk

6-press publish

Angular Project

1-Change and href in your "navbar or sidbar html" href only which redirect to component

```html
<!-- Submenu content -->
<div id='attendance' class="collapse sidebar-submenu">
    <a [routerLink]="['/attendanceIN']" class="list-group-item list-group-item-action bg-dark text-white">
        <span class="menu-collapsed" i18n>IN</span>
    </a>
    <a [routerLink]="['/attendanceout']" class="list-group-item list-group-item-action bg-dark text-white">
        <span class="menu-collapsed" i18n>OUT</span>
    </a>
</div>
```

Not to open list in your "navbar or sidbar interface"

```html
<a *ngIf="isAdmin()" href="#attendance" data-toggle="collapse" aria-expanded="false"
class="bg-dark list-group-item list-group-item-action flex-column align-items-start">
    <div class="d-flex w-100 justify-content-start align-items-center">
        <span class="fa fa-dashboard fa-fw mr-3"></span>
        <span class="menu-collapsed" i18n>Attendance</span>
        <span class="submenu-icon ml-auto"></span>
    </div>
</a>
```

2- Change your **environment Prod** like Picture

In your environment Prod put your IP instead of localhost and put your Api Project folder name before Api like Picture

```
<> side-nav.component.html        TS environment.prod.ts ×

src > environments > TS environment.prod.ts > ...
  1    export const environment = {
  2      production: true,
  3      ApiURL:'http://10.10.0.129/HRWeb/api'
  4    };
  5    |
```

3-In your App.Module import

Import {HashLocationStrategy, LocationStrategy} from '@angular/common';

 Providers: [{provide: LocationStrategy, useClass: HashLocationStrategy}],

Note: this for solve refresh page in deployment in angular

When you refresh page you will see

```
Server Error

  404 - File or directory not found.
  The resource you are looking for might have been removed, had its name changed, or is temporarily unavailable.
```

4- Run command **ng build –prod** the output in terminal like

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL                    2: cmd      ∨   +  ⊞  🗑  ∧  ×

D:\HR Main\HR>ng build --prod

chunk {} runtime.98bc6fdfbd07cf4762b5.js (runtime) 2.28 kB [entry] [rendered]
chunk {1} main.593346814e9b4758acf3.js (main) 1.58 MB [initial] [rendered]
chunk {2} polyfills.6282a28eb8950eb48467.js (polyfills) 36.7 kB [initial] [rendered]
chunk {3} styles.d022b8878ee50c60753d.css (styles) 218 kB [initial] [rendered]
chunk {4} 4.da8b69ede99b060786e2.js () 195 kB  [rendered]
chunk {5} 5.149181af3deb0cb8d1e7.js () 15.6 kB  [rendered]
chunk {6} 6.a156c084566558a92615.js () 162 kB  [rendered]
Date: 2020-11-17T12:12:20.296Z - Hash: 91d7704b2f68397a9d0e - Time: 54602ms
```

VM

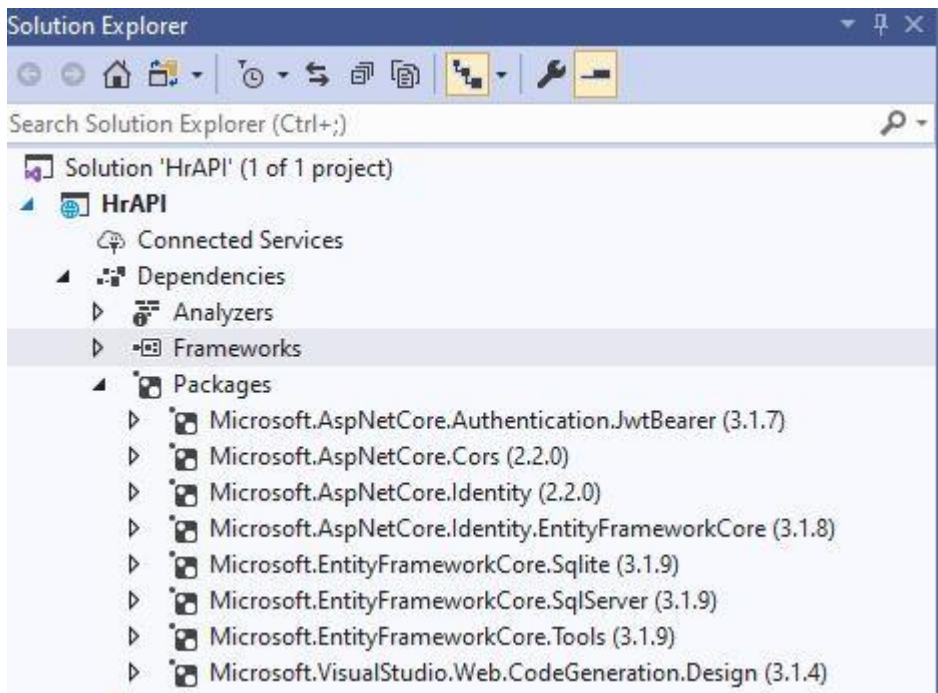1-Open Remote Desktop Connection and write your IP to connect

2-In wwwroot in **C** partition put your Published folder from **Api project**

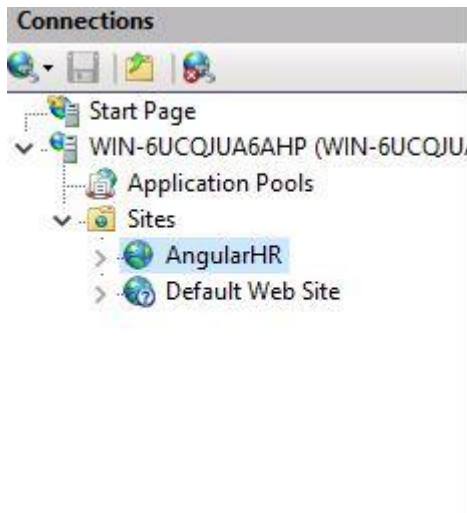3- Create folder and set name to it like **AngularProject**

4- Put all files in **dist** in angular project to **AngularProject**

5- Install **aspnetcore-runtime, dotnet-runtime, dotnet-hosting, dotnet-sdk**
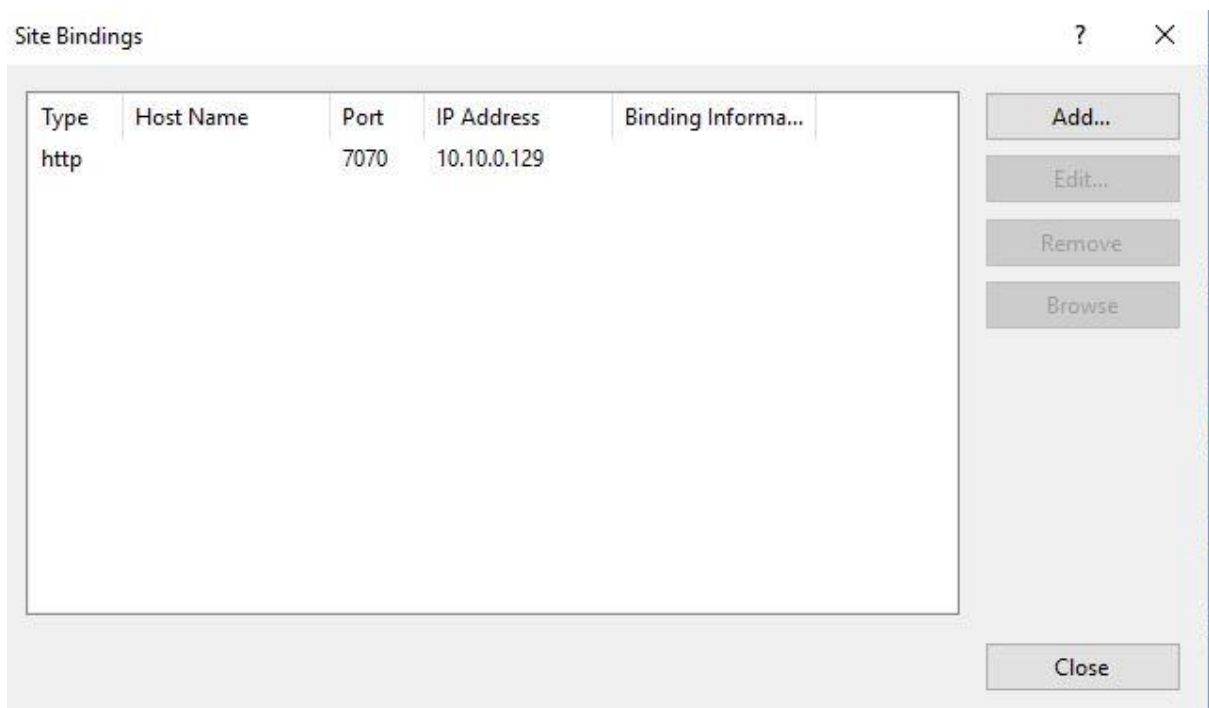
As per your project version

6-In VM server you will see your angularProject

7-Right click on **AngularProject** and choose edit binding

8-write your IP and Port

9- On default Website right click and choose edit binding and set port **80**

10- Now you can browse your project by IP from your browser