**Phase 4- Customer Support Logger**

**Githublink:** [**https://github.com/KarakambadiNaveena/CustomerSupportLogger**](https://github.com/KarakambadiNaveena/CustomerSupportLogger)

**=================================================================**

**Steps:**

**AZURE**

* Create a Sqlserver and database in Azure and connect it to SSMS.
* Create userinfo and custloginfo tables in the database and insert values to the userinfo table

**Visual Studio**

* Create a class library project (DAL)
* Create a class library project (DALTest)
* Create ASP.Net Web MVC application project with Docker Support
* Install the EntityFramework in the Project.
* Add Azure Sql connection to it.
* Write the Functionalities to validate the user.
* In the Mvc Project add the references and dependencies of DAL.
* Add the required Models,Controllers and Views .
* Build and run the Project
* Write the Testcases to validate the user in the DALTest library.
* Build and run the TestExplorer.

**JENKINS**

* Push the project to the Git.
* Now goto Jenkins Dashboard.
* Choose Freestyle project.
* Goto Github and copy the repository link.
* Once you save the project is saved in Jenkins.
* Choose Build now on the left to build it.
* Click on build history to check the build status.
* Click on it. Goto Console output.
* Configure the MSBuild in Jenkins to build it. Goto Manage Jenkins.
* Goto Manage Plugins.
* Goto Mange Jenkins -> Global Tool Configuration.
* Choose Add MSBuild
* Save.
* Once the MSBuild plugin is installed successfully goto the project.
* Goto Configure
* Choose visual studio project
* Choose the MSBuild version from the dropdownlist and select msbuild.
* Now goto the project and build the solution.
* It should build the project and should be successful if there are no build errors in the code.

**DOCKER**

* Build and run the project
* Open file explorer and paste docker file
* Open command prompt and executed the commands
* Docker build –t naviiiii/dal .
* After this docker image is created
* Docker run -p 9080:80 naviiiii/dal
* A container is created
* Success now our application is running as a docker image in this url
* http://loacalhost:9080
* Now the application will be opened in the browser.