# Setting up EQPTAS Development Environment

## Pre-requisite tools and runtime

All software installers and CI server licenses are available in EQPTAS team channel [31 Software](https://ykgwoffice.sharepoint.com/:f:/r/sites/1663153802715/Shared%20Documents/R1_06/31%20Software?csf=1&web=1&e=yWo7eE) . Take support from one of the EQPTAS team members if required.

For more details on installation & configuration refer installation guide [EQPTAS R1.06 Installation Guide.docx](https://ykgwoffice.sharepoint.com/:w:/r/sites/1663153802715/Shared%20Documents/R1_06/07%20InstallationGuide/EQPTAS%20R1.06%20Installation%20Guide.docx?d=w13330a96e240458aae06a58b8ddd41a0&csf=1&web=1&e=wtZYZe)

Install below software

1. Windows 10 / 11
2. Git for Windows 64-bit v2.43.0 or higher (<https://git-scm.com/download/win>)
3. Visual Studio Professional 2022 (<https://visualstudio.microsoft.com/vs/>)
   1. MSVC v143 VS 2022 C++ x64/x86 build tools (v 15)
   2. Windows 10 SDK 10.0.17134.0
   3. Desktop Development with C++
4. Install .NET 8.0.101-win-x64 development SDK
5. Visual Studio code version 1.93 or higher
6. Install Node Version Manager (NVM) v1.1.12 or higher from (<https://github.com/coreybutler/nvm-windows/releases> ) and install the below software using NVM
   1. Node.js 20.11.0 (Install using NVM command: “nvm install 20.11.0”) and “nvm use v20.11”
   2. Angular (“npm install -g @angular/cli@17.1.0 --save-dev”)
   3. Typescript (“npm install typescript@5.3 --save-dev”)
7. SQL server 2017/2019 (Developer edition, Basic Install, Database Engine Service)
8. SQL server management studio 20.2 or higher (20.2 is latest as of now)
9. SSRS 2019
10. Report builder 15.0.20073.0
11. Microsoft Reporting Services Projects 2022(Microsoft.DataTools.ReportingServices.vsix)
12. WiX v4.0.4 & Votive2022.vsix
13. ant 1.9.7
14. IntelliJ (Exe file: idealC-2024.1.4.exe Community Edition)
15. CI Server R1.03
16. CI Server R1.03 patches in sequence
    1. R1.03.03
    2. R1.03.04
    3. R1.03.05
    4. R1.03.07
    5. R1.03.08
    6. R1.03.0B
    7. R1.03.0E
    8. R1.03.0F
    9. R1.03.0H
17. Apply CI Server license **NODE\_TAS\_DEV**

## Wix upgrade

1. Install “FireGiant.HeatWave.Dev17.vsix”

## Windows 11 IIS Configuration

1. Go to Control Panel > Programs > Programs and Features and select Turn windows Feature on or off.

A screenshot of a computer program

Description automatically generated

1. Check the Internet Information Services. By default, it will install all necessary items required to host a website. If you want to some other component check required boxes, click OK and Close when it says ‘Windows completed the requested changes’.

A screenshot of a computer

Description automatically generated

1. Select all the check boxes selected in the below image including **Application Initialization and Web Socket Protocol** in Internet Information Services > World Wide Web > Application Development Features.

A screenshot of a computer

Description automatically generated

1. Now open your browser and type in **localhost** and press **ENTER**. You should see a default web page is rendered in your browser.

A screenshot of a computer

Description automatically generated

## Windows user and security

1. Add TASADMIN user to Windows users
   1. Right click on My Computer -> Manage -> Expand Local Users and groups -> right click on User -> click New User -> Create “TASADMIN” -> Password: YokoYoko123 -> click OK
2. Open windows -> “Edit group policy”
   1. In “Local group policy editor window” navigate to “Local computer policy” -> Computer configuration -> Windows settings -> Security settings -> Local policies -> user rights Assignment”
   2. Open log on as a batch job -> Right click and Properties -> Add user or group -> Locations -> Select system name (Ex.”CPC-1234”) -> click OK -> Advanced -> Find Now -> select TASADMIN -> Click OK
   3. Open log on as a service -> Right click and Properties -> Add user or group -> Select system name (Ex.”CPC-1234”) -> click OK -> Advanced -> Find Now -> select TASADMIN -> Click OK

## EQPTAS source code

1. Get access to the EQPTAS project in Azure DevOps from the YTI IT team.
2. Use this command to get source code from the command prompt (normal is ok, no admin privileges required here)

(Please refer End of File for Git related Information - 1.13)

1. Check out the required branch (Ex: R1.06). Developers can use Visual Studio and open the EQPTAS folder to check out the R1.06 Branch.`
2. Update the makefile in the EQPTAS folder as below
3. Change sobjs to objs in <SourceCode>\EQPTAS\makefile

$(CD) Src & make\_unix clean setup **objs** lib spread

1. Comment as below line in <SourceCode>\EQPTAS\Installer\makefile

objs:

# powershell.exe .\SignEqptasFilesYecCertficate.ps1

spread:

# powershell.exe .\SignEqptasInstallerYecCertificate.ps1

1. Append environment variables PATH (System) with the below paths
   1. %EQPTAS\_SOURCE%\Src\FT\Equipments\dev\src
   2. %EQPTAS\_SOURCE%\Src\FT\Equipments\gen\src
   3. C:\Program Files (x86)\WiX Toolset v3.11\bin
   4. C:\apache-ant-1.9.7\bin (Ex C:\apache-ant-1.9.16-bin\apache-ant-1.9.16\bin)
2. Install MANGAL.TTF font from %EQPTAS\_SOURCE%\Support\font folder.

## Build the development environment with the latest source code.

1. Stop CI Server
2. Stop IIS and IP Updater (Run using IISResetStop.cmd as administrator by right click)
3. Clear CI Server data using the command with admin privilege "ci\_clear -a”
4. Execute the below Create DB scripts to create EQPTAS DB from %EQPTAS\_SOURCE%.
   1. CreateDB.cmd
5. Execute the below DB scripts to create EQPTAS Test DB Data from TestDB\_Functionality.cmd or TestDB\_Performance.cmd command
6. Execute the below commands using run as administrator by right click.
   1. Run SETUP.CMD (Run as Administrator by right click)
   2. Run BUILD.CMD (Run as Administrator by right click)
7. RUN INSTALL.CMD (Run as Administrator by right click)
8. Configure “TASADMIN” as the owner of EQPTAS DB using SSMS.
9. Complete the **IIS Configuration** mentioned in the next section.
10. Start IIS and IP Updater (using IISResetStart.cmd)
11. Copy files from EQPTAS\Build\Output\Deploy\tls\wap\cfg\operatorInterfaces\DEPLOY to C:\Users\Public\Yokogawa\tls\wap\cfg\operatorInterfaces\DEPLOY
12. Copy files from %EQPTAS\_SOURCE%\BuildsForTesting\R1.04\TestData\wap\cfg\operatorinterfaces\DEPLOY to C:\Users\Public\Yokogawa\tls\wap\cfg\operatorInterfaces\DEPLOY
13. Start CI Server
14. Load QLI from the build folder
    1. "%EQPTAS\_SOURCE%\QLI\_Load.cmd"
15. Load Test Functionality/Performance QLI from test data which contains BCU and Tank configuration
    1. "%EQPTAS\_SOURCE%\QLI\_Test\_Functionality.cmd or QLI\_Test\_Performance.cmd command"
16. Generate and apply EQPTAS License using the license tool. Better to restart iis after applying license
    1. C:\Program Files (x86)\EQPTASComponents\AdminTools\ Yokogawa.IA.Eqptas.Licensing.LicenseTool.exe
17. Validate symbols and displays in CI Server Edit Module.

## 1.7 Rebuild the development environment with the latest source code

1. Get Latest Code using Git commands
2. Stop CI Server
3. Stop IIS and IP Updater (Run using IISResetStop.cmd as administrator by right click)
4. Clear CI Server data using the command with admin privilege "ci\_clear -a"
5. Execute the below Create DB scripts to create EQPTAS DB from %EQPTAS\_SOURCE%.
   1. CreateDB.cmd
6. Execute the below DB scripts to create EQPTAS Test Data from using TestDB\_Functionality.cmd or TestDB\_Performance.cmd command
7. Configure “TASADMIN” as the owner of EQPTAS DB using SSMS.
8. Execute the below commands using run as administrator by right click.
   1. Run SETUP.CMD (Run as Administrator by right click)
   2. Run BUILD.CMD (Run as Administrator by right click)
9. RUN INSTALL.CMD (Run as Administrator by right click)
10. Start IIS and IP Updater (using IISResetStart.cmd)
11. Copy or update files from EQPTAS\Build\Output\Deploy\tls\wap\cfg\operatorInterfaces\DEPLOY to C:\Users\Public\Yokogawa\tls\wap\cfg\operatorInterfaces\DEPLOY
12. Copy or update files from %EQPTAS\_SOURCE%\BuildsForTesting\R1.02\TestData\wap\cfg\operatorinterfaces\DEPLOY to C:\Users\Public\Yokogawa\tls\wap\cfg\operatorInterfaces\DEPLOY
13. Start CI Server
14. Load QLI from the build folder
    1. "%EQPTAS\_SOURCE%\QLI\_Load.cmd"
15. Load Test Functionality or Performance QLI from test data which contains BCU and Tank configuration
    1. "%EQPTAS\_SOURCE%\QLI\_Test\_Functionality.cmd or QLI\_Test\_Performance.cmd"
16. Validate symbols and displays in CI Server Edit Module.

## IIS Configuration

Below are the IIS Web Sites to be configured with physical path and port numbers. This is done as part of scripts below and no need to do manually

|  |  |  |
| --- | --- | --- |
| **IIS Site Name** | **Physical Path for Virtual Directory** | **Port** |
| EQPTAS TTES | *C:\Program Files (x86)\EQPTASComponents\Angular* | 8090 |
| EQPTAS Services | *C:\Program Files (x86)\EQPTASComponents\Services* | 52789 |
| EQPTAS SignalR Services | *C:\Program Files (x86)\EQPTASComponents\SignalR* | 53789 |
| EQPTAS SAPTAS Services | *C:\Program Files (x86)\EQPTASComponents\SapTasServices* | 54789 |
| MRPL SAP TAS Interface | *C:\Program Files (x86)\EQPTASComponents\MrplSapTasServices* | 56789 |

### Host the Sites/Services in IIS by scripts **automatically**

1. Create a new folder as “EQPTASComponents” in C:\Program Files (x86)\
2. RUN INSTALL.CMD (Run as Administrator by right click)
3. RUN IISCreate.CMD (Run as Administrator by right click)

**The above IIS Configuration will be completed**

### Host the Sites/Services in IIS **manually (not required if above automatic is done)**

1. Create a new folder as “EQPTASComponents” in C:\Program Files (x86)\

A screen shot of a computer

Description automatically generated

1. Copy all the below folders from the Source code output folder **(%EQPTAS\_SOURCE%\Build\Output\Deploy\)** after the successful build (Build EQPTA) to “C:\Program Files (x86)\EQPTASComponents”

Angular  
Services  
SapTasServices  
SignalR

1. Go to the IIS (shortcut: open run box and type **inetmgr)***\*make sure the Windows IIS feature is enabled as mentioned in section 1.3*
2. Right Click on “Sites” and select “Add Website” to host EQPTAS TTES as below: -

*Point to physical path:* ***C:\Program Files (x86)\EQPTASComponents\Angular***

A screenshot of a computer

Description automatically generated

Then assign the port as **8090** and click “Ok”

A screenshot of a computer

Description automatically generated

1. Right Click on “Sites” and select “Add Website” to host EQPTAS Service as below and assign the Port as 52789 and p*oint to the physical path:* ***C:\Program Files (x86)\EQPTASComponents\Services***

A screenshot of a computer

Description automatically generated

Then click “Ok” to complete.

1. Right Click on “Sites” and select “Add Website” to host **EQPTAS SAPTAS Service** as below and assign the Port as **54789 and** *point to physical path* ***C:\Program Files (x86)\EQPTASComponents\SapTasServices***

A screenshot of a computer

Description automatically generated

Then click “Ok” to complete.

1. Right Click on “Sites” and select “Add Website” to host **EQPTAS SignalR Service** as below and assign the Port as **53789 and** *point to physical path* ***C:\Program Files (x86)\EQPTASComponents\SignalR***

A screenshot of a computer

Description automatically generated

## Debugging Angular

 Visual Studio Code

Visual studio code -> open folder -> %EQPTAS\_SOURCE%\Src\TTES\Web\eqptas

View terminal -> execute command “.\ng\_serve.cmd”

### TTES in Chrome

Open Chrome -> press F12 -> Application tab

Storage -> Local storage -> Select TTES website

Key=Value

token: eyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.eyJuYW1laWQiOiIxIiwidW5pcXVlX25hbWUiOiJUQVNBRE1JTiIsImNsaWVudGlkIjoiIiwiZXhwaXJldGltZSI6IjIvMTIvMjAyNCAxMzo1Mzo1NSIsInBhc3N3b3JkZXhwaXJldGltZSI6IjEyLzMxLzk5OTkgMjM6NTk6NTkiLCJuYmYiOjE3MDc3MjI2MzUsImV4cCI6MjAyMzM0MTgzNSwiaWF0IjoxNzA3NzIyNjM1fQ.Vd1NL5lZ9WQ9hTwIY0Bd70p7Kjogk8BlD0qzUoL9jBRmSExKvAPEH3zigtstKHVOTmmq1NQJPRbqANb4eNZDwQ

## Debugging ASP.NET core services

### EQPTAS web services for API

Visual studio 2022 -> Open solution %EQPTAS\_SOURCE%\Src\TTES\Services\Yokogawa.IA.Eqptas.Services\Yokogawa.IA.Eqptas.Services.sln

Start debugging

Click yes/ok for the IIS certificate and other prompts

### EQPTAS web services for SignalR

Visual Studio 2022 -> Open solution %EQPTAS\_SOURCE%\Src\TTES\Services\Yokogawa.IA.Eqptas.Services\Yokogawa.IA.Eqptas.Services.sln

Start debugging the SignalR project

Click yes/ok for the IIS certificate and other prompts

### Debug SAP-TAS services

Visual Studio 2022 -> Open solution %EQPTAS\_SOURCE%\Src\TTES\Services\Yokogawa.IA.sapta.services\Yokogawa.IA.saptas.services.sln

Start debugging the SAPTAS project

Click yes/ok for the IIS certificate and other prompts

### Test API with Postman

New -> request -> Give some name

GET -> <http://localhost:52789/api/cardreader>

Authorization tab -> type -> Bearer Token ->

eyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.eyJuYW1laWQiOiIxIiwidW5pcXVlX25hbWUiOiJUQVNBRE1JTiIsImNsaWVudGlkIjoiIiwiZXhwaXJldGltZSI6IjIvMTIvMjAyNCAxMzo1Mzo1NSIsInBhc3N3b3JkZXhwaXJldGltZSI6IjEyLzMxLzk5OTkgMjM6NTk6NTkiLCJuYmYiOjE3MDc3MjI2MzUsImV4cCI6MjAyMzM0MTgzNSwiaWF0IjoxNzA3NzIyNjM1fQ.Vd1NL5lZ9WQ9hTwIY0Bd70p7Kjogk8BlD0qzUoL9jBRmSExKvAPEH3zigtstKHVOTmmq1NQJPRbqANb4eNZDwQ

Click send

## Debugging Process/FAST

### Open EQPTAS Java modules in Intellij IDE

Open IntelliJ -> file -> Open -> \Src\FT\EQPTAS

### Configuration of Intellij IDE

Java SDK

File -> Project structure -> Project SDK -> Add and choose from CI Server installation folder “C:\Program Files (x86)\Yokogawa\CIServer\tls\jre”

EQPTAs Java Libraries

File -> Project structure -> Libraries -> Add and choose from CI Server installation folder “C:\Program Files (x86)\Yokogawa\CIServer\tls\exe”

### Debug PROCESS/FAST

1. Open F/T performance monitor
2. Stop “opcjexe”, “opcjexe2”, “opcjexe3” and “opcjexe4” processes based on object priority mentioned in the Engineering module.
3. Put break point in PFClasses -> Src -> Builder -> WF.java -> triggerGroupEvent() function
4. Select “executor” or “executor2” or “executor3” or “executor4” program in IntelliJ
5. Start debugger

### Debug Automatic engineering code

1. Open CI performance monitor
2. Stop “JWUIISG” process
3. Put breakpoint in FTEngInterface -> src -> eqptas -> servlets -> FTEngServlet
4. Select “JWUIIS” program in IntelliJ
5. Start debugger

### Debug SAP Demo

1. Run properties -> <http://localhost:53765>

## Debugging Equipment Drivers

### Debug EQPTAS 1010CB BCU

1. Create Loading Arm with MS product
2. Create “Contrec 1010CB - single - TCP/IP” BCU with Line BLINE01.
3. Open F/T performance monitor
4. Stop process EQPBLINE01 in the processes tab of the performance monitor.
5. Visual studio 2022 -> Open solution %EQPTAS\_SOURCE%\Src\FT\Equipments\BatchController\contrec\eqpcb1\src\eqpcb1.sln
6. Configure command arguments **“-n EQPBLINE0**1**”** debugging setting of the project

A screenshot of a computer

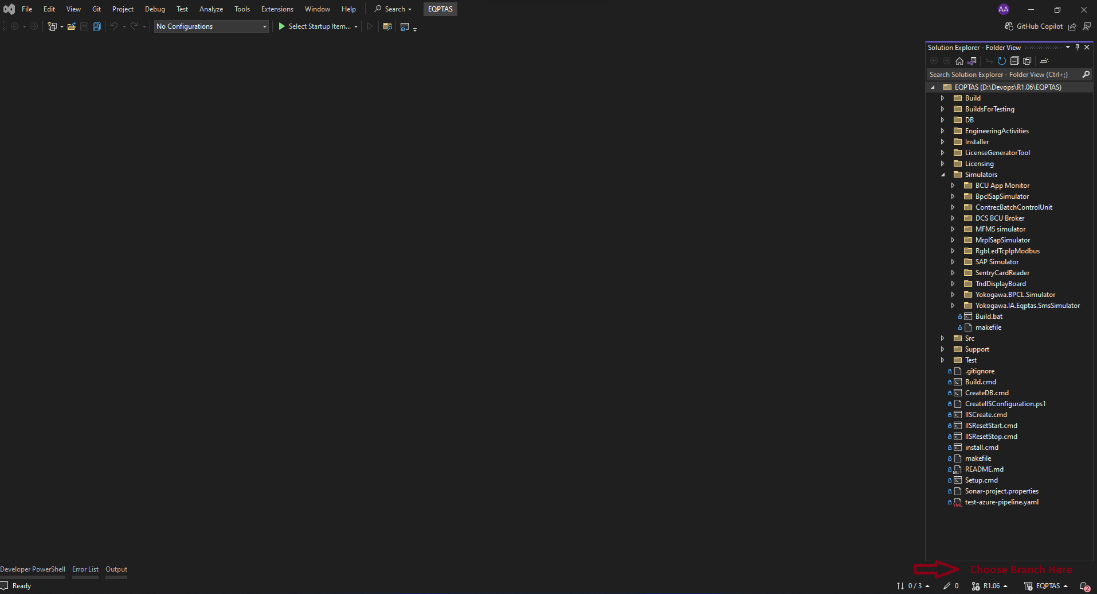
Description automatically generated

1. Start debugging

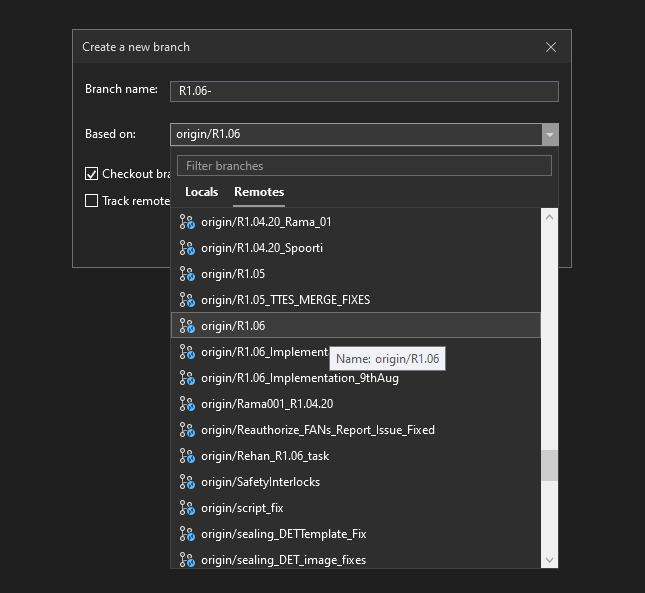
## Git Push and Clone Related information

1.13.1 Git Clone – EQPTAS REPOSITORY

1. Open Visual Studio 2022, and open Terminal window (Location where Repository must be cloned)
2. In terminal window, enter git clone <http://ytidevops:8080/tfs/ytiprojectscollection/_git/EQPTAS.>
3. Navigate to branch, as per the current Project (Ex. R1.06)



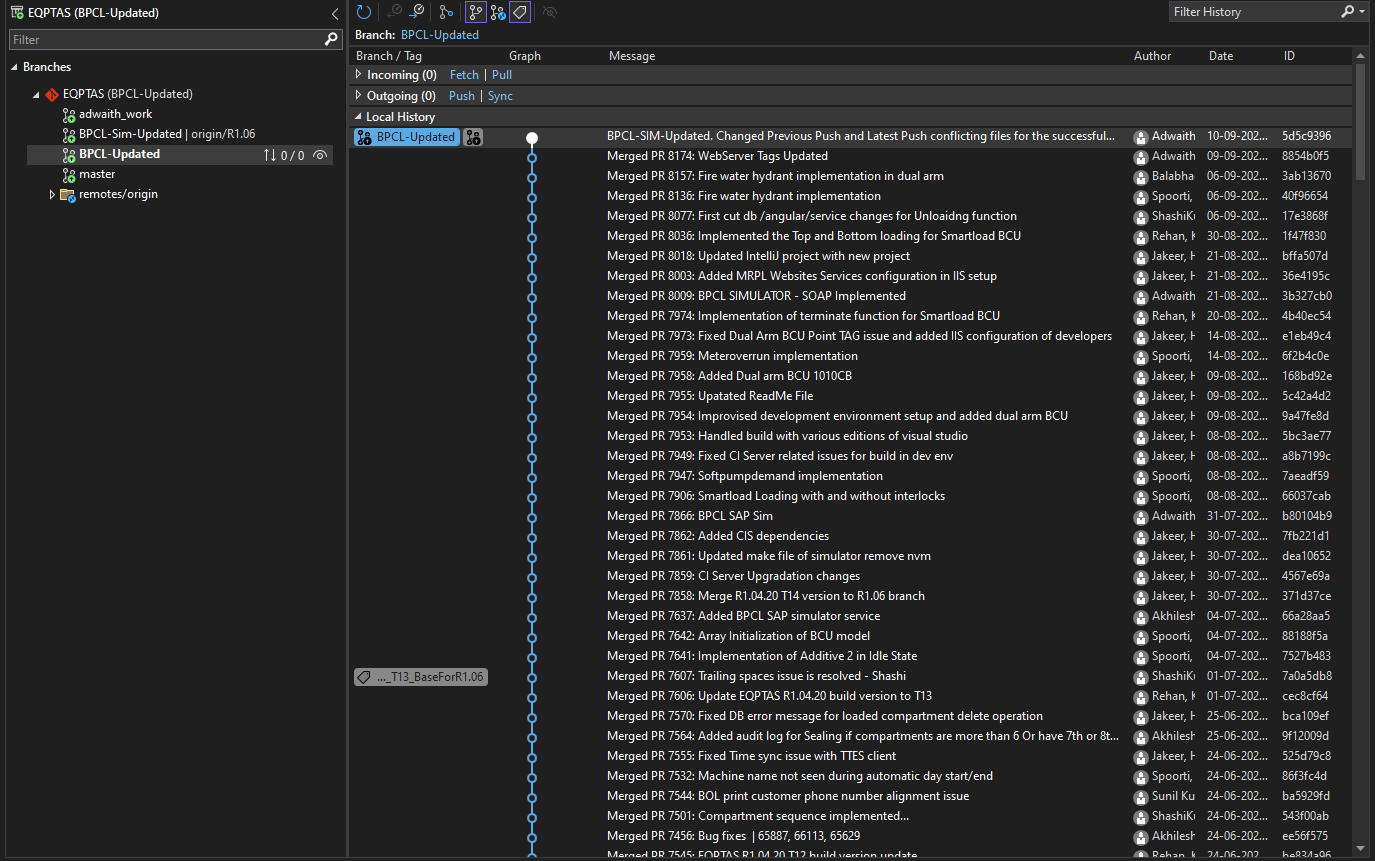
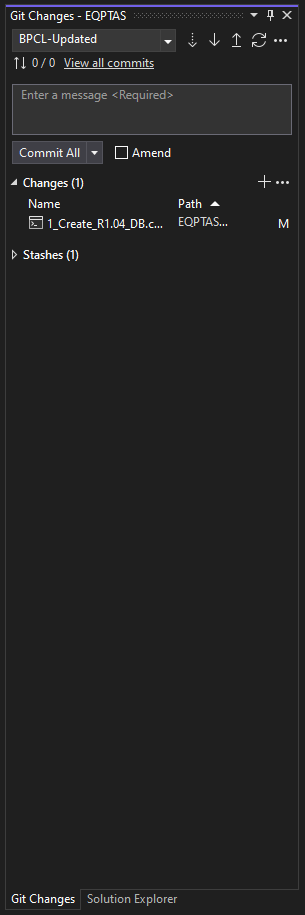
1. Create a new Branch for Local Machine, based on the Remote branch of CurrentProject(Ex. R1.06)



1. Then Start working on the Locally created Branch based on the Remote cloned branch.

1.13.2 - Pushing Code to EQPTAS Branch

1. Based on the Changes made in Local Repository, while committing to the main EQPTAS branch, make sure that the Local Branch created is in sync with the Remote Origin Branch before pushing the code to the Main Branch.



1. Fetch and Pull the Updated code from the required branch to which you are pushing the code. Based on the changes in your Local and Remote branches, the changes will be displayed as in the above window.
2. Recheck again if there are any conflicts in the branches rather than local code changes and latest remote Branch changes before Committing and Pushing the Changes.
3. Once the code is committed and pushed, create a Pull request without any conflicting files within the branch.

## Virtual Machine License Issue Resolution

If you're using a virtual machine and encounter a license issue, you need to comment out the following code section. This will allow the TTES screen to open. However, if you uncomment the code, the TTES screen will not open.

**Steps:**

1. Navigate to the following path:

E:\DevOps\R1.06\EQPTAS\Src\TTES\Services\Yokogawa.IA.Eqptas.Services\Yokogawa.IA.Eqptas.Services\Middlewares\LicenseValidatorMiddleware.cs

2. Comment on the following section of code:

if (!licenseManager.IsLicenseValid())

{

throw new InvalidAppliedLicenseException(EqpTasResource.INVALID\_LICENSE);

}

if (!maxTtesManager.IsMaxTtesValid())

{

throw new InvalidAppliedLicenseException(EqpTasResource.INVALID\_LICENSE);

}

3. Save the changes and restart the application.

This change will allow you to access the TTES screen without the license validation checks.