Step-By-Step Guide

Project folder structure

- OverwatchStatsCode (This contains all of the Visual Studio solutions and projects)
    - OverwatchNightlyUpdaterClient
    - OverwatchStats.Common.Data
    - OverwatchStats.MVC.Test
    - OverwatchStats.MVC
    - OverwatchStats.Service.Web
    - OverwatchStats.Store.Service
    - OverwatchStats.Test
    - OverwatchStats.WCF.Service
    - OverwatchStats.WCF.Tests
    - OverwatchStatsWCF.Tests
- SQL (This Contains the Patches and stored procedures required to run the program)
    - CleanUp
    - Patches
    - StoredProcedures

Project Descriptions

OverwatchNightlyUpdate
This project is for updating the database using the WCF service using an XML populated by players whose stats you wish to track.

OverwatchStats.Common.Data
This project contains all of the common POCO classes to be used by any project wishing to make use of the returns from the WCF service or DB.

OverwatchStats.MVC.Test
Project dedicated to the testing of the OverwatchStats.MVC project

OverwatchStats.MVC
Project which consumes the WCF service and displays the information using MVC.

OverwatchStats.Service.Web
This project is responsible for retrieving data from the PlayOverwatch Website which is then used to populate the database.

OverwatchStats.Store.Service
This project is responsible for accessing the database and call stored procedures.

OverwatchStats.Tests
The project is responsible for the unit testing of any of the service projects.

OverwatchStats.WCF.Service
This project is dedicated to creating a WCF service which access the OverwatchStats.Service.Web project to retrieve player information and uses this information to populate database.

OverwatchStats.WCF.Tests
This project is dedicated to integration tests for the WCF service.

CleanUp
This folder contains all of the patches and stored procs to used during the integration tests

Patches
This folder contains all of the patches required for the database to function

Stored Procedure
This folder contains all of the stored procedures required to run the WCF service

Step-by-Step Guide

Setting Up Database

After downloading the contents of the Git Repository to your local machine the first thing you must do is navigate to the **SQL** folder and run all of the patch files manually in the **Patches** folder starting with Database Create script and then in the order they are numbered (i.e OverwatchStatsPatch001 first and then OverwatchStatsPatch002 next).
*(Note: if you would like to use your own database name just change any mention of the database name in the **DatabaseCreate.sql** script.)*

After this you can then go to the **SQL** folder again and run all of the stored procedures in the **Stored Procedures** folder.

After this you have run the bare minimum required for running the projects if you would also like to run the tests you must also run all of the scripts in the **CleanUp** folder which can also be found in the **SQL** folder.

Setting Up WCF Service

After the Database set up you should be able to run the WCF service navigate to **OverwatchStatsCode** -> **OverwatchStats.WCF.Service** folder and launch the solution file.

After successfully opening in Visual Studio the next step is to change the **Connection Strings** in the Web.config file to your own connection string.

This should now allow your WCF service to run fine.

Setting up MVC project

To launch the MVC project you should navigate to **OverwatchStatsCode** -> **OverwatchStats.MVC** and launch the solution stored in there.
After successfully opening in Visual Studio confirm that your service references for the WCF service are correct and you should be able to launch your MVC app.

Setting Up NightlyUpdater Project

To run the nightly updater project navigate to the **OverwatchStatsCode** -> **OverwatchNightlyUpdaterClient** and launch the solution.
After successfully opening in Visual Studio confirm that your service refernces for the WCF are correct and you should be able to run the updater