Library Management

Submitted to: Anthony Pagnotta

Ancy Tom.

Student ID: 199693500

5/29/2019

Library Check-in/Check-out System

About the application (high level)

Library check-in/check-out is a simple web application that allows the user to add a member to library, check-in and check-out a book. There are 3 menus on the web application for these. Each has a form asking for various details and a button to perform the operations.

Application screenshots below for reference:

Book Check-in

Name	Ancy Tom
Student ID	19953345
Book Name	Mobile Strategy ▼
Book ID	AlgBk0002 ▼
Taken date	05/15/2019

Check-in   Book check-in noted

Add new Member

Name		Enter student name
Student ID		Enter student ID
Phone		Enter Phone
Address		Enter address
City	Select City ▼	
Email		Enter email

Add New Member
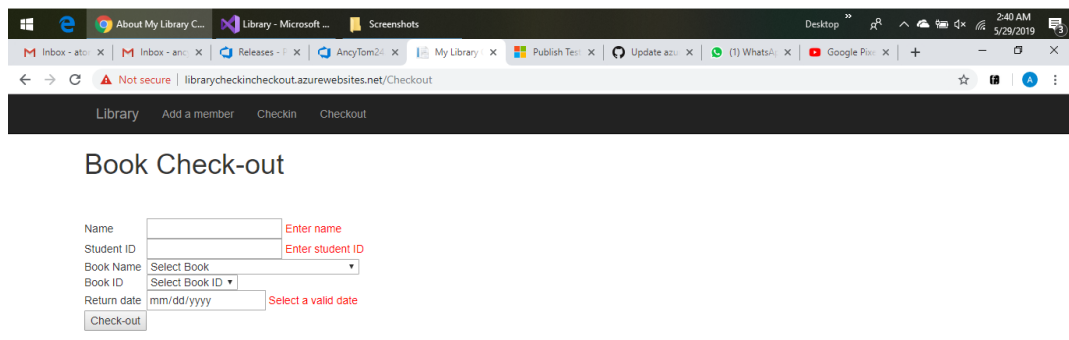
Library   Add a member   Checkin   Checkout

Add new Member

Name	Ancy Tom
Student ID	19953345
Phone	7059750421
Address	627 Macdonald Avenue
City	Sault Ste. Marie ▼
Email	ancytom24@gmail.com
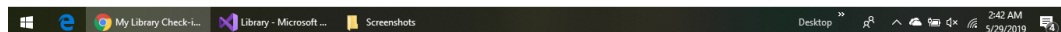
Add New Member   Member Added

Library   Add a member   Checkin   Checkout

Book Check-out

Name		Enter name
Student ID		Enter student ID
Book Name	Select Book ▼	
Book ID	Select Book ID ▼	
Return date	mm/dd/yyyy	Select a valid date

Check-out

About the application (low level)

Library app is created in .net framework 4.7 with c#. On click of buttons, form details are saved into a txt file. Versioning is controlled by GitHub and build and release is done in Azure. Unit test for .net framework is also used for testing.

Tools Used:

- Visual studio 2019 – IDE for developing
- GitHub – versioning
- Visual studio unit test for .net framework application – unit testing
- Azure DevOps – creating build and release pipeline, deployment.
- Windows OS

Versioning

GitHub is a hosting service for version control using Git. It offers all of the distributed version control and source code management functionality of Git as well as adding its own features [1].

Creating a new repository in GitHub

- Sign in to GitHub account
- Select Add Repo
- Enter desired name
- Click create repository button.
- My repo url : https://github.com/AncyTom24/Library.git

Connecting code from visual studio 2019 to GitHub

- Create a project in vs 2019
- View -> Team Explorer -> Home
- Select Sync menu from team explorer
- Select option to link to Git repo
- Signing to GitHub
- Add repository name
- Push code to git.
- Now, every time we have a change in code, we can push it to git.
- Before we start working on the code we should pull the latest code from git if more than one person is making changes to code.
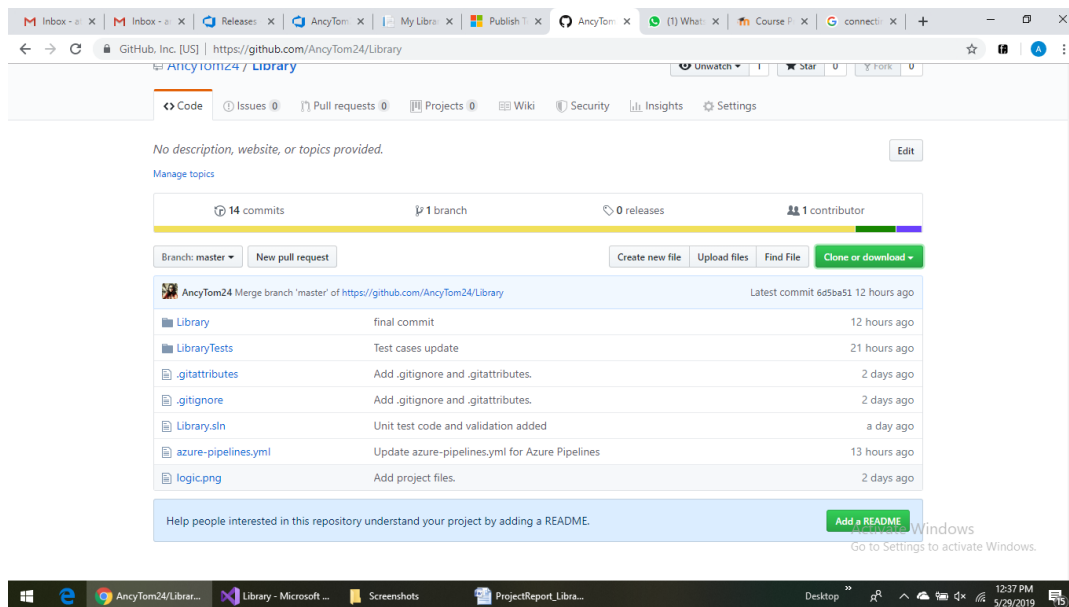
Azure DevOps

- Sign-in to azure devOps.
- Add a build pipeline.
- Select code source from GitHub.

- Select repository.
- Review pipeline YML, make necessary changes and run. – This was quite difficult step for me because I wasn't sure what should be added to the YML.
- Once it is successfully run, a build pipeline is created.
- Now, we need to create a release pipeline.
  - Sign-in to portal.azure.com.
  - Subscribe to azure to create a resource.
  - Create resource.
  - Get free trial subscription key.
- Go to release tab in azure DevOps.
- Select new release pipeline.
- Select template – Azure app service deployment.
- Select our subscription and resource name in the form.
- Fill out other details like artifact configuration path.
- Save -> Create release.
- Select release – deploy – Success/Fail.
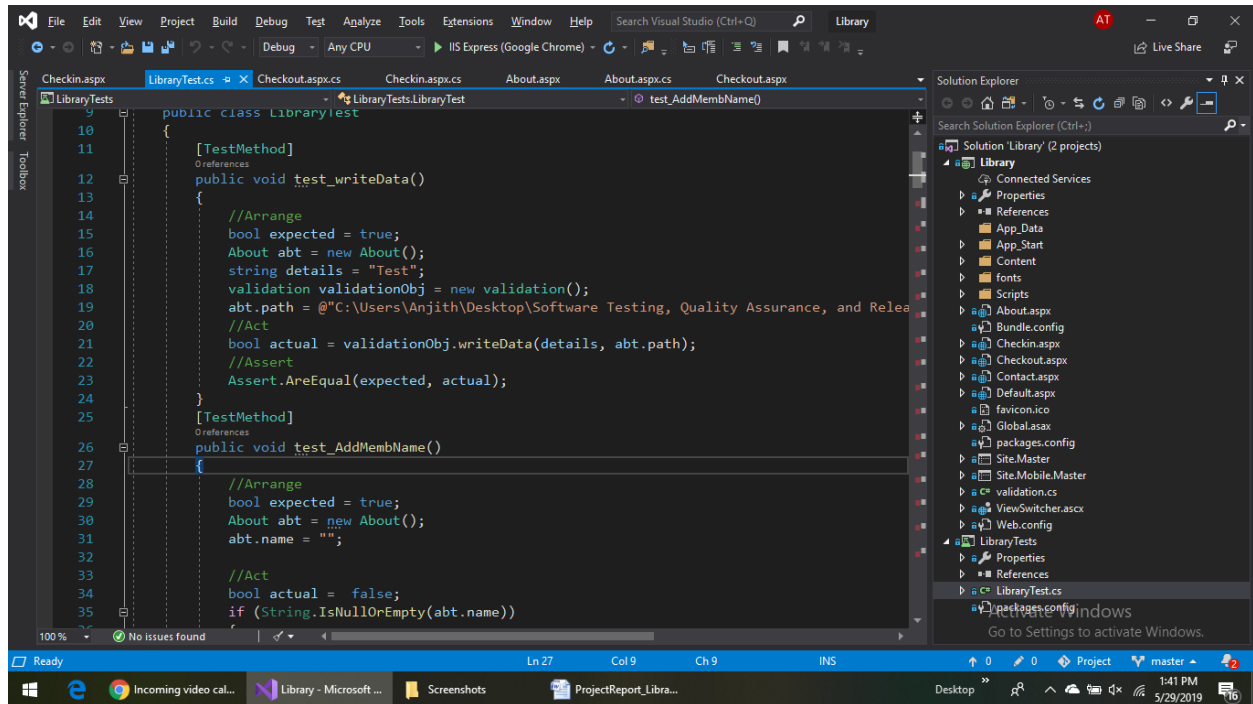- If failed, check logs and correct errors and deploy again.

Publish to azure from vs

- Build the solution.
- Select publish from context menu.
- Add azure account to vs.
- Select configuration settings and click publish.
- Published URL: http://librarycheckincheckout.azurewebsites.net/Checkout.

Testing

       A testing project for .net framework is created in the same solution folder for unit testing the app. various test cases are:

- Required filed validation and testing
- Writing data to txt file checking

Reference

[1]Wikipedia