CITE

30003389

Programming 3

AT3 Project Outline

Table of Contents

[What data structures are you using? 1](#_Toc26861341)

[Doubly Linked List 1](#_Toc26861342)

[Lists 1](#_Toc26861343)

[Where are you using hashing techniques? 2](#_Toc26861344)

[What sorting algorithm are you using and how is this different from selection and bubble sort? 3](#_Toc26861345)

[What search technique are you using? 4](#_Toc26861346)

[What third party libraries are you using? 5](#_Toc26861347)

[Where can I find documentation for this? 5](#_Toc26861348)

[A mock-up of the GUI 6](#_Toc26861349)

[PipeServer 6](#_Toc26861350)

[PipeClient 6](#_Toc26861351)

[What source control are you using? 7](#_Toc26861352)

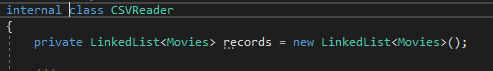
[What code standards are you enforcing? 8](#_Toc26861353)

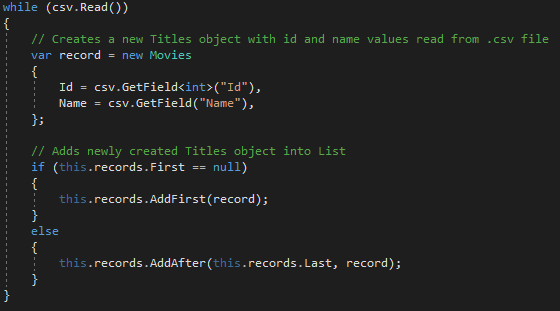
[What tests are you going to run? 9](#_Toc26861354)

# What data structures are you using?

### Doubly Linked List

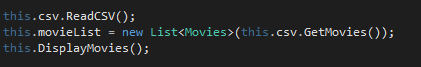
I am using a doubly linked list within the CSVReader class. This is used to read in the CSV data which is then used throughout other functions of the PipeClient solution





### Lists

The data from the doubly linked list is then imported into a list to enable various functions such as the Search and MergeSort Algorithms.

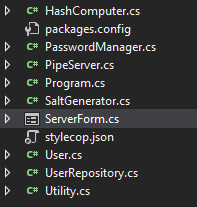




# Where are you using hashing techniques?

The hashing techniques I have used are stored within classes in the PipeServer solution.

The Password Manager classes uses the HashComputer and SaltGenerator to hash and salt passwords that are then saved within a user object. The user object is then stored within a UserRepository and the hash data can then be accessed and utilised from that object.



# What sorting algorithm are you using and how is this different from selection and bubble sort?

I am using a Merge Sort algorithm to sort through pre-shuffled CSV import data, which is then reproduced in a sorted state within a listbox on the GUI. Merge sort differs from selection and bubblesort in its implementation; it is a divide and conquer algorithm that separates the objects recursively until they are singular objects, then merges them together to form a complete sorted data structure.

Bubblesort is an iterative approach to sorting, and continually matches elements against each other, swapping them adjacently if required until the entire data structure is sorted. Higher complexity data structures can take an extremely long time to sort with this method.

Selection sort is another iterative approach to sorting. Unlike bubblesort, the selection sort does not match elements to their adjacent counterparts, instead finding the highest (or lowest) value element and placing them at the end or beginning of the data structure. Once again, as this is an iterative approach to sorting, large and complex data structures can take a significant amount of time to sort with this method.

# What search technique are you using?

I’m using an iterative search within the PipeClient as the data may be sorted or unsorted at the time of the search. The search takes in data through a text box and then matches it against the appropriate field of an element contained within the list. An index integer is then supplied to the LstOutput box to use as an Index Selection.

# What third party libraries are you using?

I am using the CSVHelper third party library to import and export CSV data to the PipeClient solution

I am also using the StyleCop library to enforce coding standards throughout the PipeServer and PipeClient solutions and contained classes within.

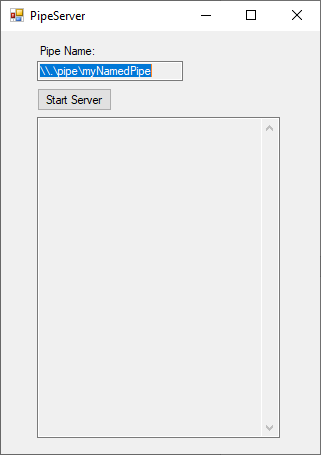
# Where can I find documentation for this?

The documentation for CSVHelper can be found [here](https://joshclose.github.io/CsvHelper/getting-started/).

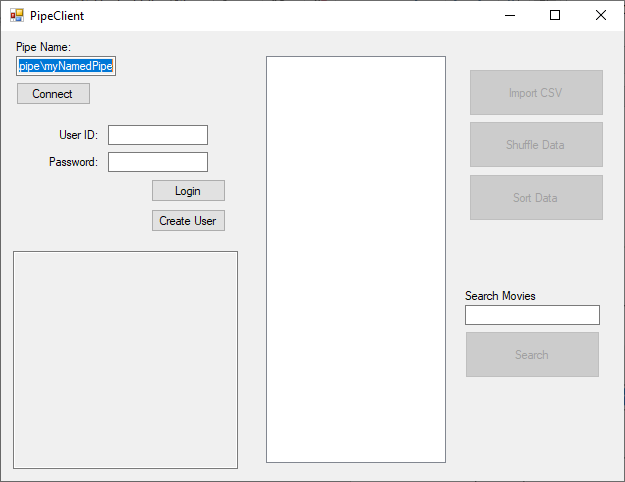
The documentation for StyleCop can be found [here](https://github.com/DotNetAnalyzers/StyleCopAnalyzers/tree/master/documentation).

# A mock-up of the GUI

### PipeServer

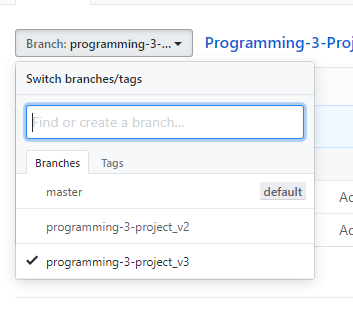


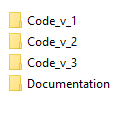
### PipeClient



# What source control are you using?

I am using GitHub as my source control. The repository containing the solutions and documentation can be found [here](https://github.com/kyerpotts/Programming-3-Project).





# What code standards are you enforcing?

I am using the third party library StyleCop to enforce strict coding standards throughout the PipeServer and PipeClient solutions.

More information pertaining to the StyleCop standards can be found at

<https://github.com/StyleCop/StyleCop>

# What tests are you going to run?

* IPC connection tests between Server and Client
* User Creation
* User Login
* CSV Import
* Data Shuffle
* Data Sort
* Search function
* Close Application hotkey