Software Design and Implementation

SOFT20091

Nottingham Trent University

April 2019

* Harry Clarkson (N0774446)
* Jonathan Perez (N0753746)
* Mathew Cutler (N0741548)

Contribution Page

Member A: Jonathan Perez (N0753746)

* Class Diagram
* Explanation on cohesion and coupling
* Data Structure (Stacks)
* Explanation of the Internal data structures used.

Member B: Mathew Cutler (N0741548)

* Sequence Diagram
* State Diagram
* User Interface
* Examples and screenshots of the user Interface

Member C: Harry Clarkson (N0774446)

* Component Diagram
* Deployment Diagram
* Searching and Sorting Algorithms
* Explanation on Searching and Sorting Algorithms

Description of the System

The Purpose of the system is to access and edit current film projects that have been published by TrekStar Pictures. The user can view all projects available or a particular project and the materials related to the project. One project can have many materials but only one material can have project. The reason for this is this is because a project may have a material type of DVD at a price of £5 along with other attributes and it may also have another material type of VHS at a price of £6 but another project might have a material type of DVD, but the price may be different for that in particular project with a DVD and/or other attributes.

When the System is running, on load It will write everything from the project file to a stack. The user then has the choice whether to browse projects and its materials or to enter maintenance mode where it can still view everything like it normally would, but it can also create edit and remove projects and its materials related to them.

When browsing all projects, it pops everything of the stack to then display its attributes then it looks at the Material Id’s that are related to the project and then retrieves this information from Materials file that contains them.

When editing a Project, the system requests which project then asks what it would like to change (its attributes or its materials) Depending on which it searches through the stack to find the specified project and to push it back onto the stack again with its updated data.

The same is done for remove however it isn’t pushed back on it is just forgotten about and therefor removed.

On closure of the software the items in the stack then overwrites the current Projects file and the same with materials.

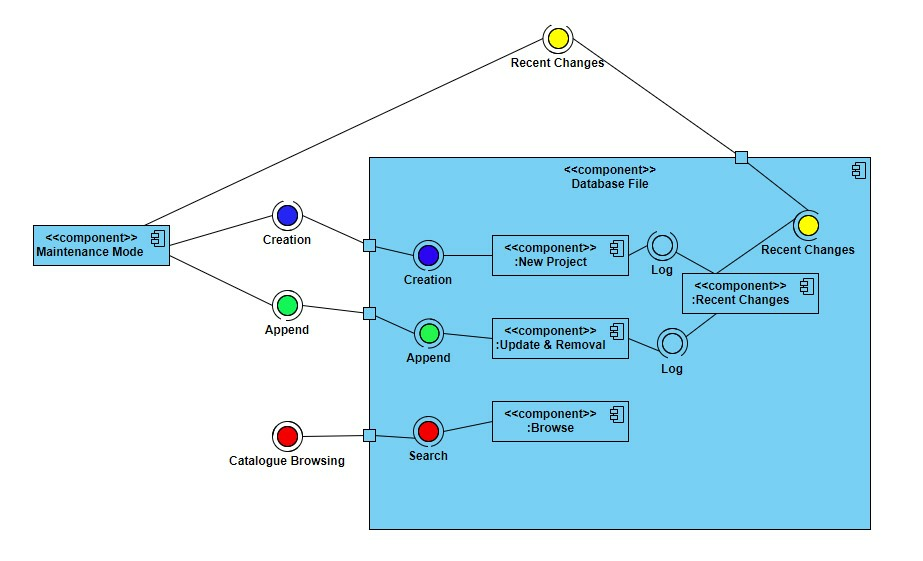
Class Diagram

Explanation of Cohesion and Coupling

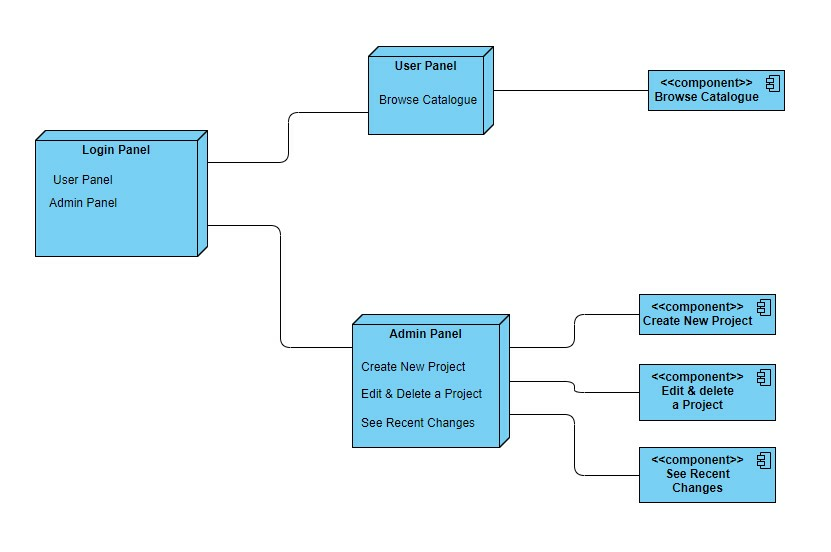
Sequence Diagram

State Diagram

Component Diagram



Deployment Diagram



Explanation of Searching algorithm

It starts off with a “for loop” where it repeats the code inside for the amount of items that are in the project stack. It then uses the requested Project ID to check if the current Project in the stack is the project that contains the correct project ID. If it does then it will set the current project as the project that contains the project ID and set a bool to true to indicate it has been found. At the end of the for loop if the requested project ID is not found then the bool will still be false and therefore will inform the user that the project has not been found.

Plagiarism Declaration

This report and the software it documents are the result of my own work; other contributing group members are acknowledged. Any contributions to the work by third parties, other than tutors, are stated clearly below this declaration. Should this statement prove to be untrue I recognise the right and duty of the Board of Examiners to take appropriate action in line with the university’s regulations on assessment. Names: Harry Clarkson (N0774446) Jonathan Perez (N0753746) Mathew Cutler (N0741548)