Server Side Programming Assessment

Tom Willington, Computer Science Group 2

Design

There are three stage to my design; Front-end website design, Database design and OOP code design. I will begin with my front-end design.

Front end design-

The first step in designing my website was to decide upon a front-end framework such as Bootstrap, Fbootstrapp, MetroUI or BootMetro. I have previously used Bootstrap so I was keen to try a different framework. Eventually I decided upon Metro UI, Metro UI is a front end framework from Microsoft. The reasons for choosing metro UI over other frameworks was firstly the style portrayed with Metro was a very clean aesthetically pleasing design and secondly they had lots of useful features which were readily available, such as a drop down menu and live tiles.

Secondly, I thought I should look at other DVD rental or streaming websites with good designs. The main movie streaming company was Netflix and so I began to examine the website for features that I liked and could reproduce and use within my system. I also examined the navigation and search features of the site to give myself an idea of how I would implement my website design.

The features I found to be affective in making the design aesthetically pleasing and simple were:

- The use of one single main colour (in the case of Netflix it was red).
- Having a small header and the rest as pictures of the DVD artwork.
- Within the header they had to the left the Home, Browse and other functionality based options whereas on the right there was an account button which had a drop-down to other account based options, I liked this design as it meant all the option could be put on the header and still be separated.

Subsequently, when I had a few ideas of what I wanted to implement I went back to metro UI to see what pre-build features I could implement. Metro UI provided a wide variety of features and among many I used were the following:

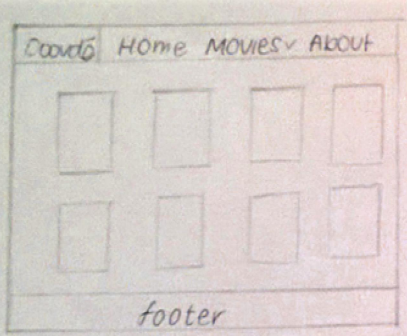
- Drop down menu bar
- Login Page
- Pre-build add to basket button

From these I drew up my draft designs with these feature implemented then I began to add my bespoke details, such as DVD pictures and Logos. See image 1.

After these draft designs I began more final designs in which I have decided the colour to be green and the font will remain as Segoe UI which is the standard font for Metro UI. I left the font so the web pages would be consistent with the metro UI design. I decided to use green as my main colour as I was trying to keep a tertiary colour but didn’t want to use red as it was too similar to netflix. Metro UI provided the menu bar and the option to change to their green colour scheme.

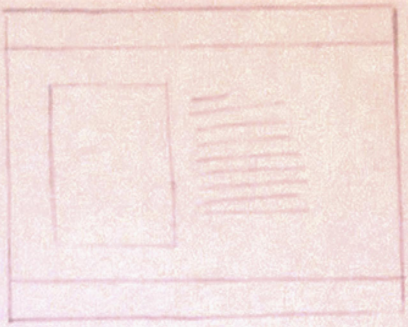
When the design was complete I designed my logo and other images to be used on my site.

I have also included screenshots of my finished design after stage 3. In which you can see my logo design and finished menu bar.



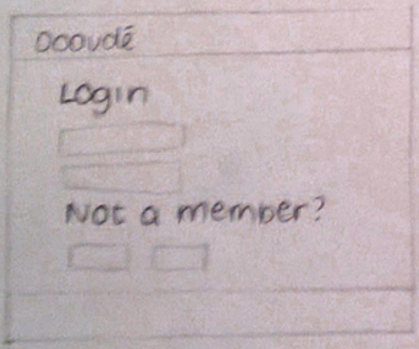
Home Page

- image dimensions 150 × 215
- On-Click take to DVD page.



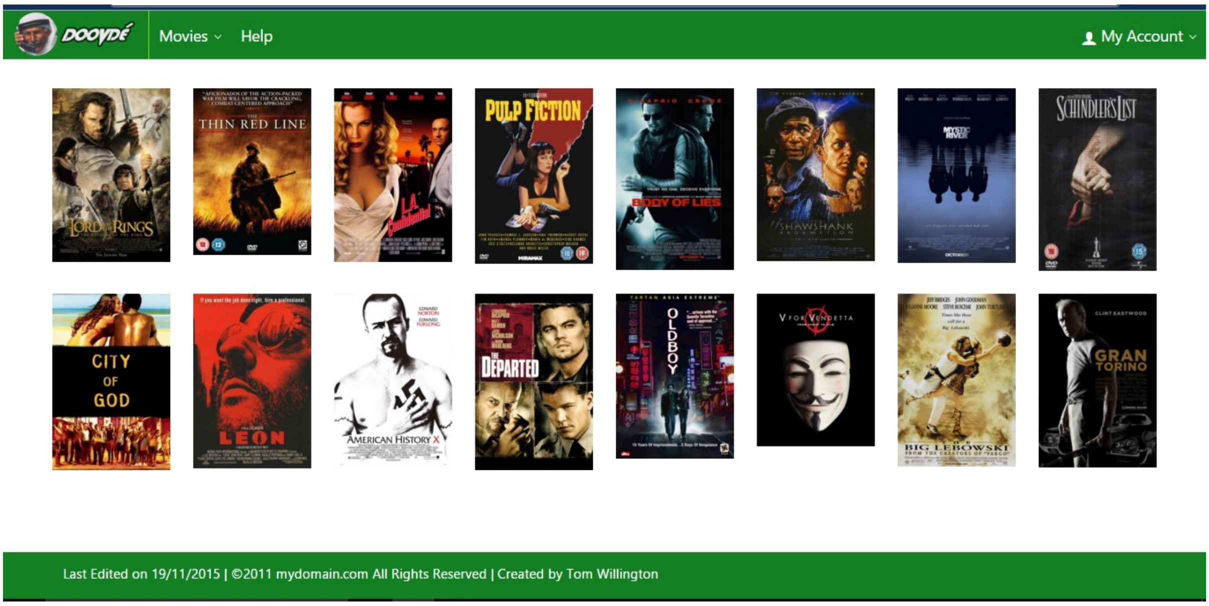
DVD Page

- image dimensions 333 × 470
- Header and footer identical.

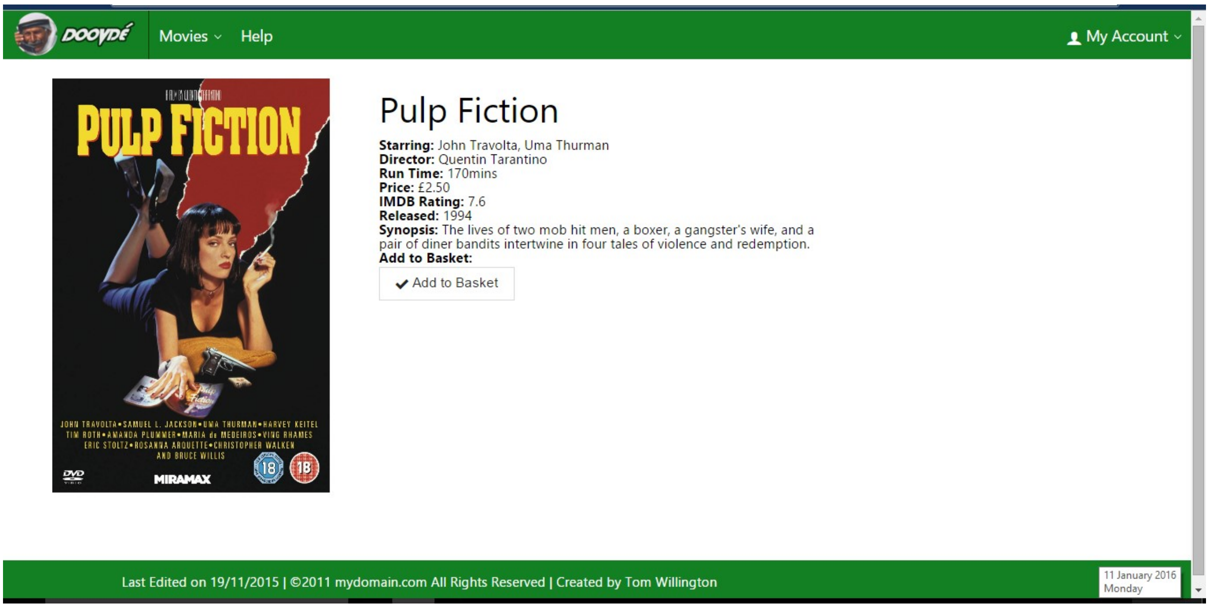


Login Page

- Footer the same
- Header just logo.



Last Edited on 19/11/2015 | ©2011 mydomain.com All Rights Reserved | Created by Tom Willington

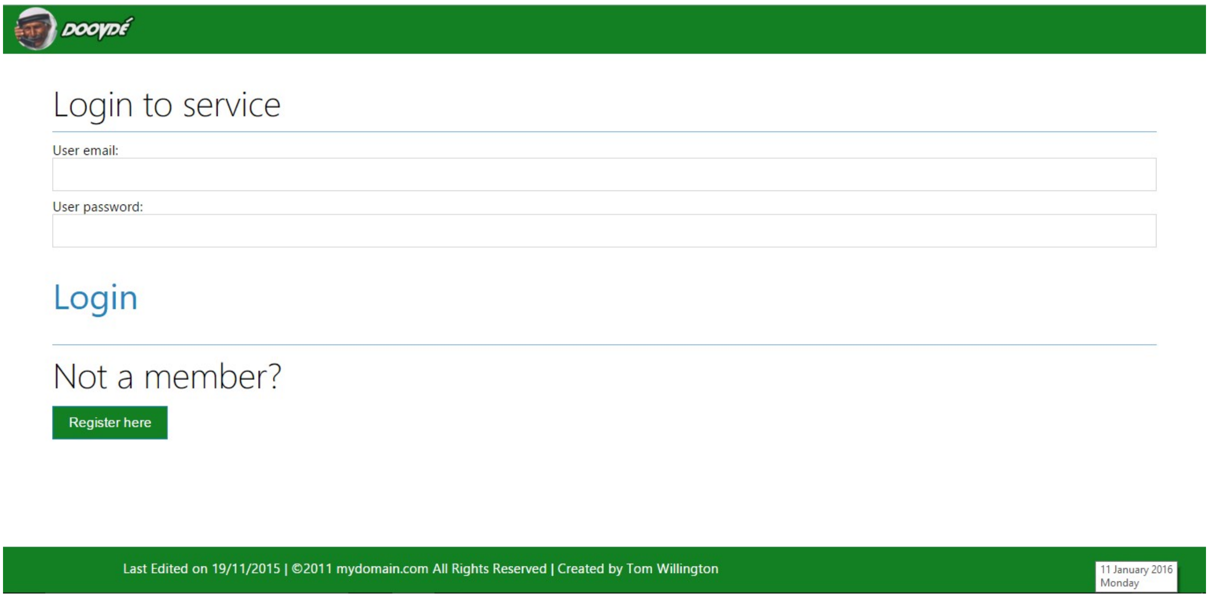


Pulp Fiction

Starring: John Travolta, Uma Thurman
Director: Quentin Tarantino
Run Time: 170mins
Price: £2.50
IMDB Rating: 7.6
Released: 1994
Synopsis: The lives of two mob hit men, a boxer, a gangster's wife, and a pair of diner bandits intertwine in four tales of violence and redemption.
Add to Basket:

Last Edited on 19/11/2015 | ©2011 mydomain.com All Rights Reserved | Created by Tom Willington

11 January 2016
Monday



Login to service

User email:

User password:

Not a member?

Last Edited on 19/11/2015 | ©2011 mydomain.com All Rights Reserved | Created by Tom Willington

11 January 2016
Monday

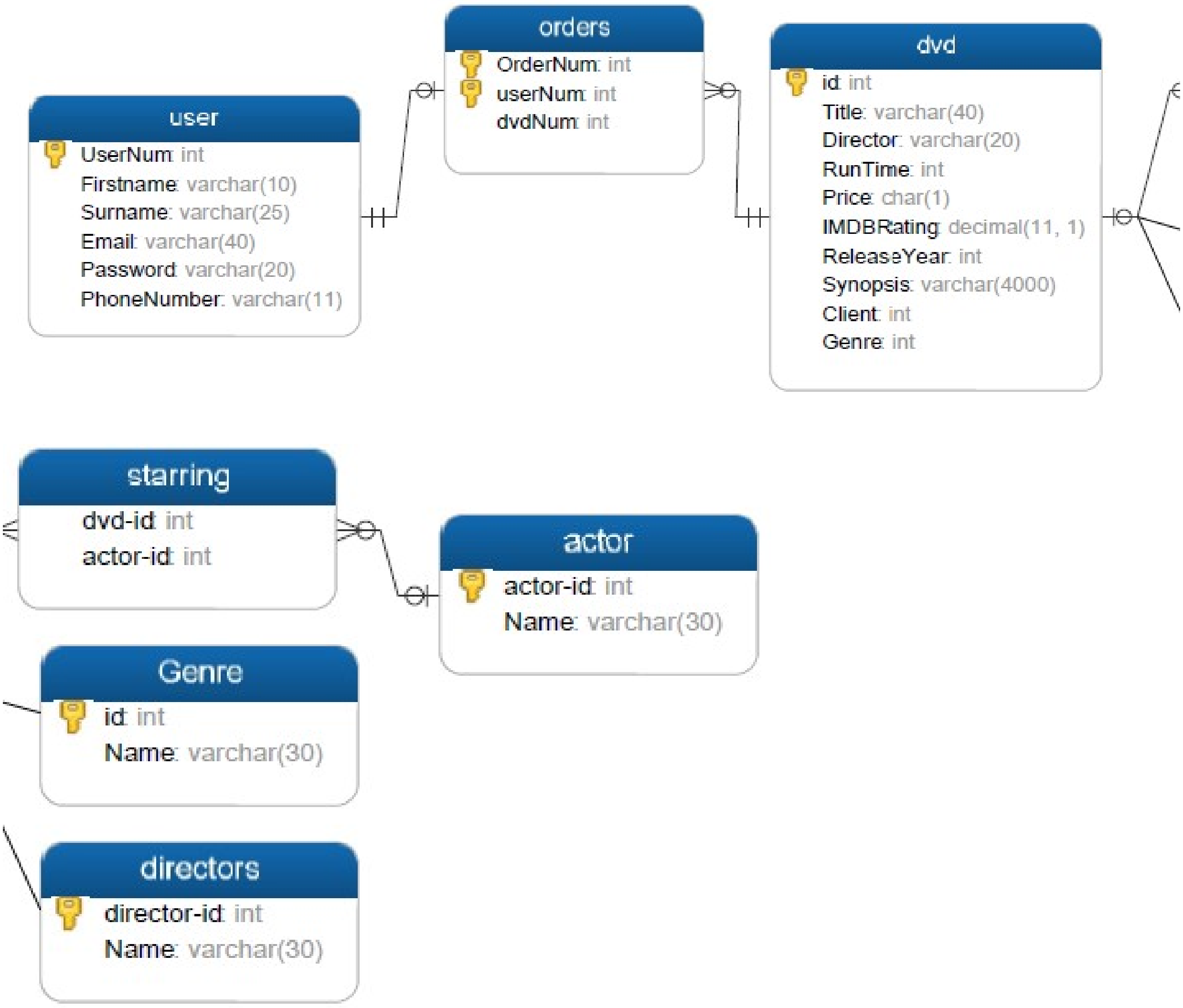
Database design-

For my database I was planning to use Navicat and included in this software was a modelling application which was used to design databases. So I firstly began writing all the table names and their fields. I would then go on to use the model application to insert and manage relationships between the tables. Once I had finished I had a model I could use to implement my database from.

Below is my model that I used to design my database, this is not in any way the finished product during production I will surely make many changes. However with this I can ensure that ive visualised how my database will look before beginning my implementation.

Below I have used a few different types of relationship, one being a many-many to which I must take an alternative approach. I make another table which details each specific relationship. For example the actor to dvd relationship is many to many, as a dvd can have multiple actors and an actor can be in multiple dvds. So I made a separate table which contained two fields one was the actor id and the other was a dvd id both were foreign keys, this way I can have multiple relationships.

Many to one relationship are simpler, for example directors can direct multiple films whereas there can only be one director for a film so I place a director id within the dvd table which was a foreign key in another relation with further details about the directors.



OOP Code design-

As I will be aiming to use PHP in an OOP context, I will have to give some thought about how I will be designing my classes, method and other things. Firstly I will be using MVC design pattern to store my files. So for every .phtml file within views I will need a .php file to access it acting as a controller. But within models is where I will use OOP, I will need a class for the Database, DVD and the Library. These are just to start with I have no doubt I will need more classes.

The database class will just establish the PDO connection and have methods such as getDatabase() to return the instance of the class I am using.

The DVD class will have all the same fields as the DVD relation in the database and will store information about the DVDs and return the information from get methods.

The library class will manage the basket and the session.

Testing

My testing strategy would be to test certain functionality of the site at stage of implementation then repeat all the tests at the end so firstly, I need to separate my implementation into intervals in between which I will complete my testing. The bases of my thinking is almost waterfall, I won't move on until I'm happy with the pervious stage or it will get confusing to manage errors with multiple sections not working. Below I have listed the stages of my implementation and therefore testing.

1. Design a web template (includes logo design)
2. Complete other web pages templates leaving blank anything which requires back-end functionality.
3. Design and populate database
4. Establish PDO connection to database
5. Retrieve and display DVD information
6. Add filters: titles, directors and genres
7. Add to basket functionality
8. Add login functionality with self-registration
9. Add searching ability

Below I have testing requirements for each stage of development once they have been met implementation can continue.

Stage 1:

-Test all layout and design features are working correctly and the template should work as normal
-Test all images are loaded correctly
W3C Code check on html pages
-No outstanding errors

Stage 2:

-Test all navigational links on all pages
-Test all images are loaded correctly
-W3C Code check on html pages
-No outstanding errors

Stage 3:

-Test databases by populating accordingly
-Compare to original model make note of changes and decide whether they are beneficial or not
-No outstanding errors

Stage 4:

-To test connection successfully var_dump some information from the table
-No outstanding errors

Stage 5:

-load a DVD page which displays all the information of certain DVD
-Check all the images and display them on a home page with links to each specific DVD.
-No outstanding errors

Stage 6:

-Test by selecting all filter and seeing result
-Click on the navigational links to ensure they work

Stage 7:

-Successfully add an item to the baskets and ensure it disappears from the list of available purchases

Stage 8:

-Login and use sessions to access the website.

Stage 9:

-Searching filters work successful
-No outstanding errors

Once they are all complete I will go back and perform all test again.

Self-Evaluation

What I feel I performed well on-

Firstly one of my strengths in my opinion is my design and use of Metro UI. Using this framework I feel my design is clean and simple as well as being aesthetically pleasing. The use of a single colour along with the help of the pre-build features from metro UI I feel worked well to present the information nicely.

Also I felt that my use of a PDO to access my database and how I put my DVD information within a class resulted in a malleable OOP system that could be improved in the future.

I feel I took an organised approach, I planned out all my testing before I began to implement and I had a good idea of how I wanted my design to look before beginning.

What I feel I could have done better on-

Mid way through my implementation I encountered problems which set me back and I had little contingency time so my project is incomplete. This lack of functionality is detrimental to the project and renders it incomplete and therefore a failure. I had begun at a very fast pace and made a long of progress as such I underestimated the following sections and began to focus my effort in other modules. This lack of understanding the task at hand has cost the website functionality. I had planned properly including factors such as testing and design but what I failed to consider was time.

Things to consider for my next project-

As previously mentioned time management cost me marks in this project and next time I will incorporate time management techniques such as Gantt charts and schedules. Something I also will improve on is countering my lack of understanding because my underestimation of what I needed to do ended up costing me in the functionality of my system.

Further improvements on my system-

If the deadline was extended I would firstly finish my system by adding a basket, searching and a login system using sessions.

Beyond that I could add multiple pictures for each DVD and have a slideshow on the DVD page. Finally I could include the use of live tiles to show information when the cursor is scrolled over the DVD picture.

<u>My self-assessment marking..</u>

For the first one I would place myself in the 100-80% as I have met the requirements as I have used Metro UI very well and in a considerable amount of code.

1. Secondly I would place myself in the 100-80% as my database was designed very well and populated accordingly.
2. Then I would place myself in the 80-60% as I feel I have shown my capabilities to code in an OOP paradigm in MVC using a PDO however I feel I could have had more code.
3. 20-0% as login not implemented.
4. 100-80% for DVD information retrieval as it works successfully.
5. 20-0% as basket not implemented.
6. I would place myself in the 80-60% bracket as I have given an informative report addressing all sections; however I feel I could have gone into more detail in certain areas.
7. 20-0% as no additional feature were used.