Thea Gregory

5291240

# Report – Video Store Program Visual Basic.NET/MySQL

## Introduction

### About the Project

The goal of this project was to use Visual Basic.NET and MySQL to create a project that mimics the functionality required to administrate a video rental store from the late 20th century. It was designed to test the ability to integrate all skills learned in block 3 so that they could all come together into one more complicated project that demonstrates its creator’s competence in the subject material.

### About the Report

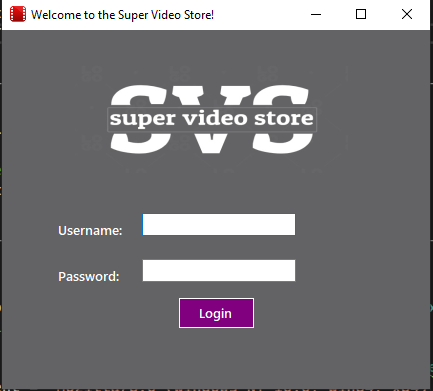
This report serves to document the steps and challenges undertaken in the creation of this software project. It will contain detailed screenshots, walkthroughs, and a post-mortem of issues encountered during the creation process.

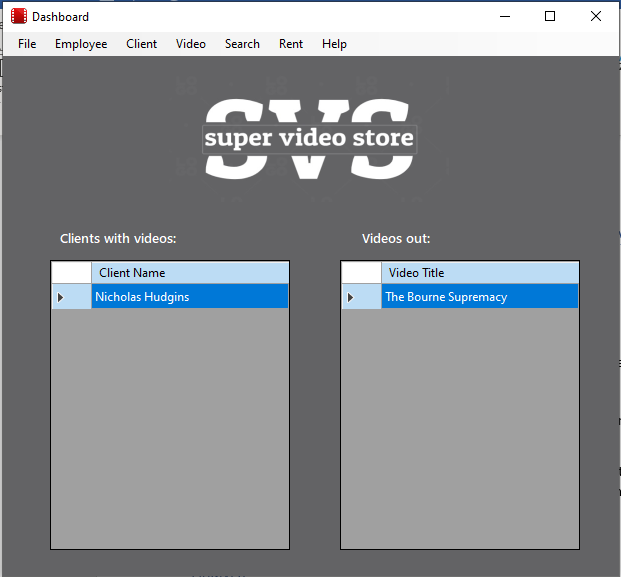
## Analyze

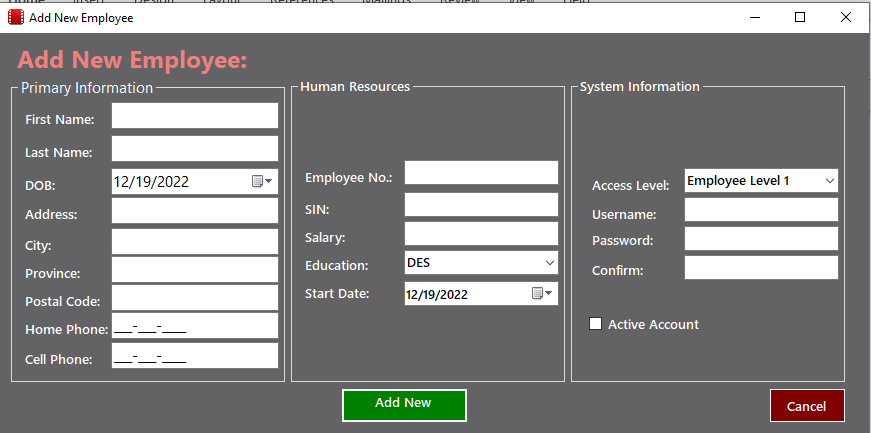
### How did you plan on realizing the project?

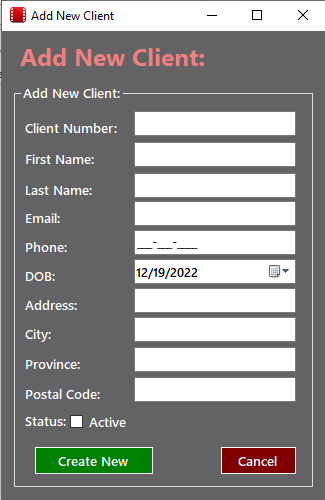
I decided early on to address the project one use case at a time. I would design and test as I went, only moving to the next case once I was satisfied that the current one was completed to the best of my ability. My approach was systematic and methodical, with the aim of reducing distractions and becoming lost in too many details at once.

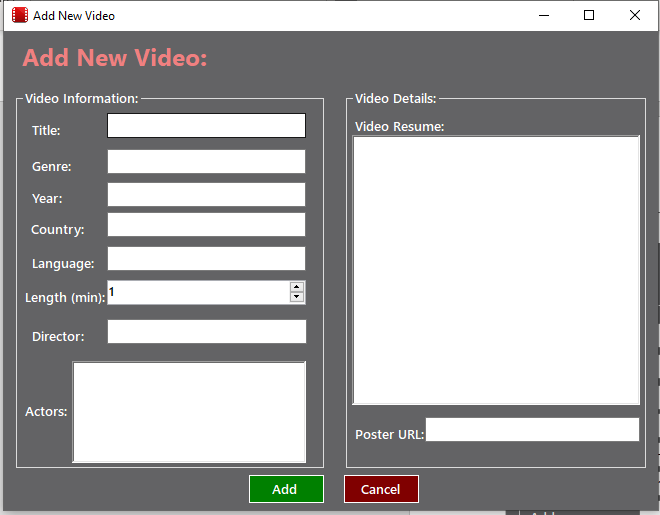
Screenshots:

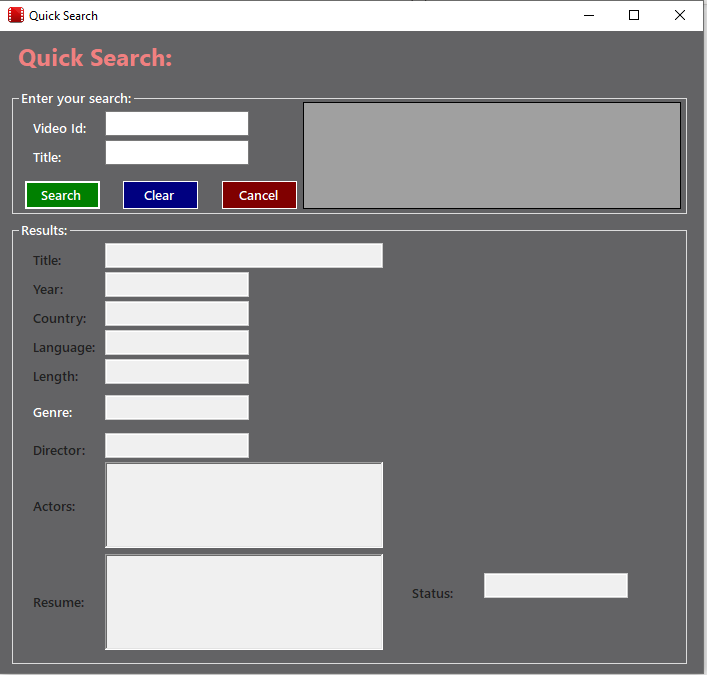


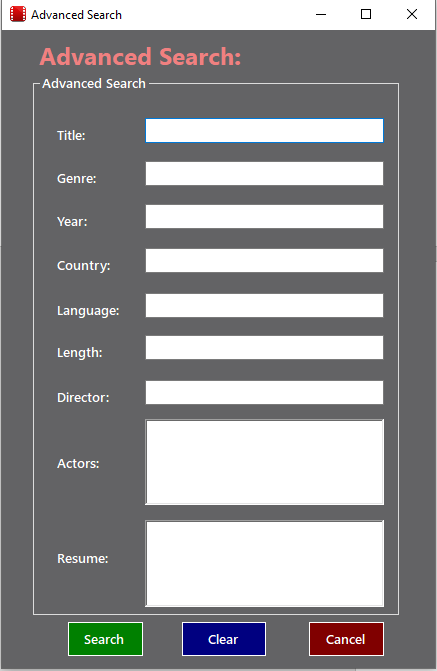


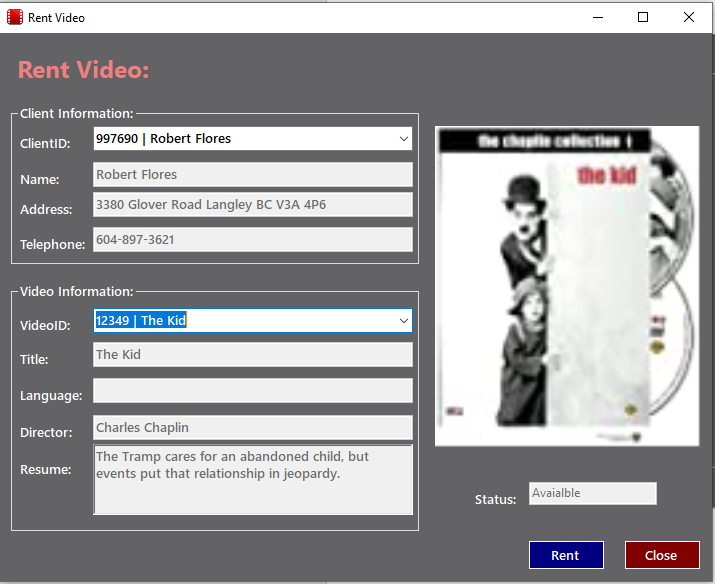


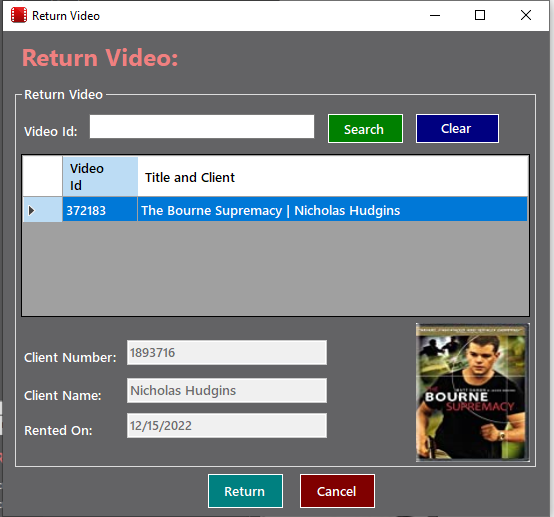












## Solution

My solution was to create a streamlined application that was intuitive and aesthetically pleasing. I followed the directives laid out in the project guidelines, but adding a unique flavor to the interface and functionality. There was a great deal of freedom left to us, so I took advantage of that to create something that represents my skills as a programmer while still delivering an efficient and useful piece of software.

## Walkthrough

When the application launches, the user is directed to login with a username and password. Only an active user is granted entry into the system, even if they enter the appropriate password. Once they have logged in, the user is presented with a dashboard which displays the clients who have rented movies and the movies that have been rented. This is for reference. Then, there is a menu that links to the functions of the software. The Employee menu, available only to administrators, grants the power to list, create, edit, and delete users.

The Client and Video menus are available to all, but their functions are limited by the user’s access level, as granted by the administrator. However, all users can view clients and movies, as well as create new ones.

All users can use the quick and advanced search functions. The quick search will return information on the movie, while advanced search will allow the user to search for many fields and allow them to rent any movie that comes back in the results.

The Rent menu gives both the rent and return functions, which are available to all users. Rent allows the user to pick a client from a list, as well as a movie. If the movie is available, it is rented to the user. Return will allow the user to set the status of the movie back to available and allow it to be rented again.

# Discussion

## What happened throughout the project?

This project was a test of endurance. There were many use cases, each with its own unique challenges. The name of the game was persistence, working at each until it gave the intended result. I would define a goal for the day and work towards it. I also made great use of Google, using it to refresh my memory on MySQL functions, as well as Visual Basic commands and syntax help.

## Problem, Encounter, Solution?

Towards the end of the project, I began to have trouble keeping everything straight. The project was huge, bigger than anything I had ever worked on before. To keep my mind focused on everything, I maintained a logical separation between classes. In my case, I found my DBManager class was becoming far too large and not well defined. So, I broke it up so that I had several DBManager classes. One for Employee, client, video, search, and rent. This allowed me to simplify my code and made my functions easier to find and keep track of.

## Did you manage to finish the project?

I completed the project in the allotted time.

## What would you have done differently?

I would have divided my DBManager classes earlier on, as I was losing time sifting through my functions looking for which one to call. This lapse caused a lot of my early inefficiency, and I would have finished the project with more testing time had I done this.