PROG1121J Database Programming Project

Project: Database Driven Application

Using a topic of your choosing and instructor approval, create a data-driven, Windows Forms desktop application that will create, manipulate, search and browse records within a SQL Server database. The database design will contain three tables. Two tables will be related through a many-to-many relationship.

The minimum application requirements will contain the following forms

- An MDI form container
  - Complete with status strip updated by child forms
  - Menu for launching all child forms
  - Tool strip for common application functions
- About form
  - List all Application Properties
  - Open from the MDI Toolstrip and Menu
- Splash Screen
  - Displayed at launch
  - Application name
  - Current Version number
- Login form
  - Create a Login database table with pre-populated login accounts. A user cannot access the application unless successfully logged in.
    - Fields
      - User
      - Password
- Maintenance form for each entity in the database. This form must accomplish the following
  - Present all database entity attributes with an appropriate Windows form control
  - Allow navigation forward, back, first and last records within the database table
  - Buttons for Create, Update and Delete, with enable property managed according to the current state of the form
  - Child form will update the MDI status label according to the current action (See attached screenshots for examples)
    - Examples include but not limited to
      - Adding…
      - 1 of 50 records
      - Ready…
- Maintenance form for the many-to-many relationship
- Two Browse forms
  - Allow browsing of each entity and show the related records within a DataGridView
  - Update mdi status strip label (See attached screenshots)
- Create two business rules that are enforced within the system (Requires instructor approval)

PROG1121J Database Programming Project

Submission Requirements

- Private GitHub Repository (Instructor added as collaborator) with project source code and artifacts
- Database creation script added to GitHub project repository
- Database seed data added to GitHub project repository
    - Clean test data included for all tables
- OneClick Deployment for installation
- Product Presentation and Demonstration
    - Introduce the product with a short PowerPoint presentation
        - Include business rules
    - Product walkthrough demonstrating all features (Not a code review)
        - Launched from OneClick installation

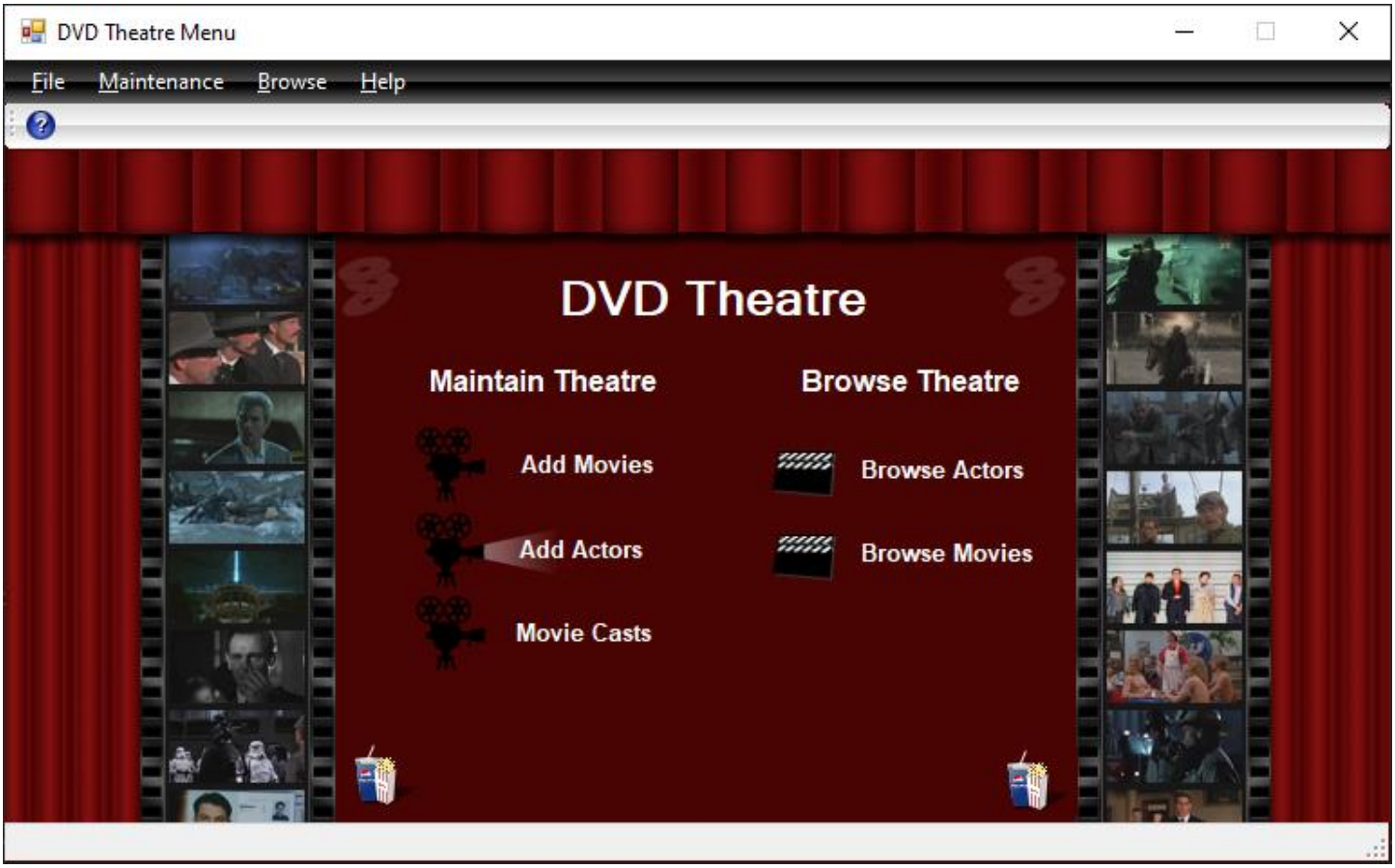
Deliverable Schedule

- Project GitHub repository
    - May 22st 11:59pm Instructor added as collaborator
- Project topic and business rules
    - May 22st 11:59pm Word document memo format checked into GitHub in a folder called **Artifacts**
- ER Diagram version 1
    - May 25 11:59pm checked into GitHub in the **Artifacts** folder. Ensure the ERD is clearly versioned in its naming (Example **CollegeCodeDirectoryERD_V1.drawio**, **CollegeCodeDirectoryERD_V1.pdf**). Use design software of your choosing. Ensure the repository has your native format file and a pdf version.
    - Must include Login table in ERD
- ER Diagram version 2
    - May 27th 11:59pm revised database design based on instructor feedback, checked into GitHub in the **Artifacts** folder. Ensure the ERD is clearly versioned in its naming (Example **CollegeCodeDirectoryERD_V2.drawio**, **CollegeCodeDirectoryERD_V2.pdf**).
- Mid Project Check In and Progress Review
    - June 3rd instructor meeting to review current progress of the project
- Final Code Check in
    - June 7th 11:59pm
- Presentation
    - Week of June 8th - June 12th

PROG1121J Database Programming Project

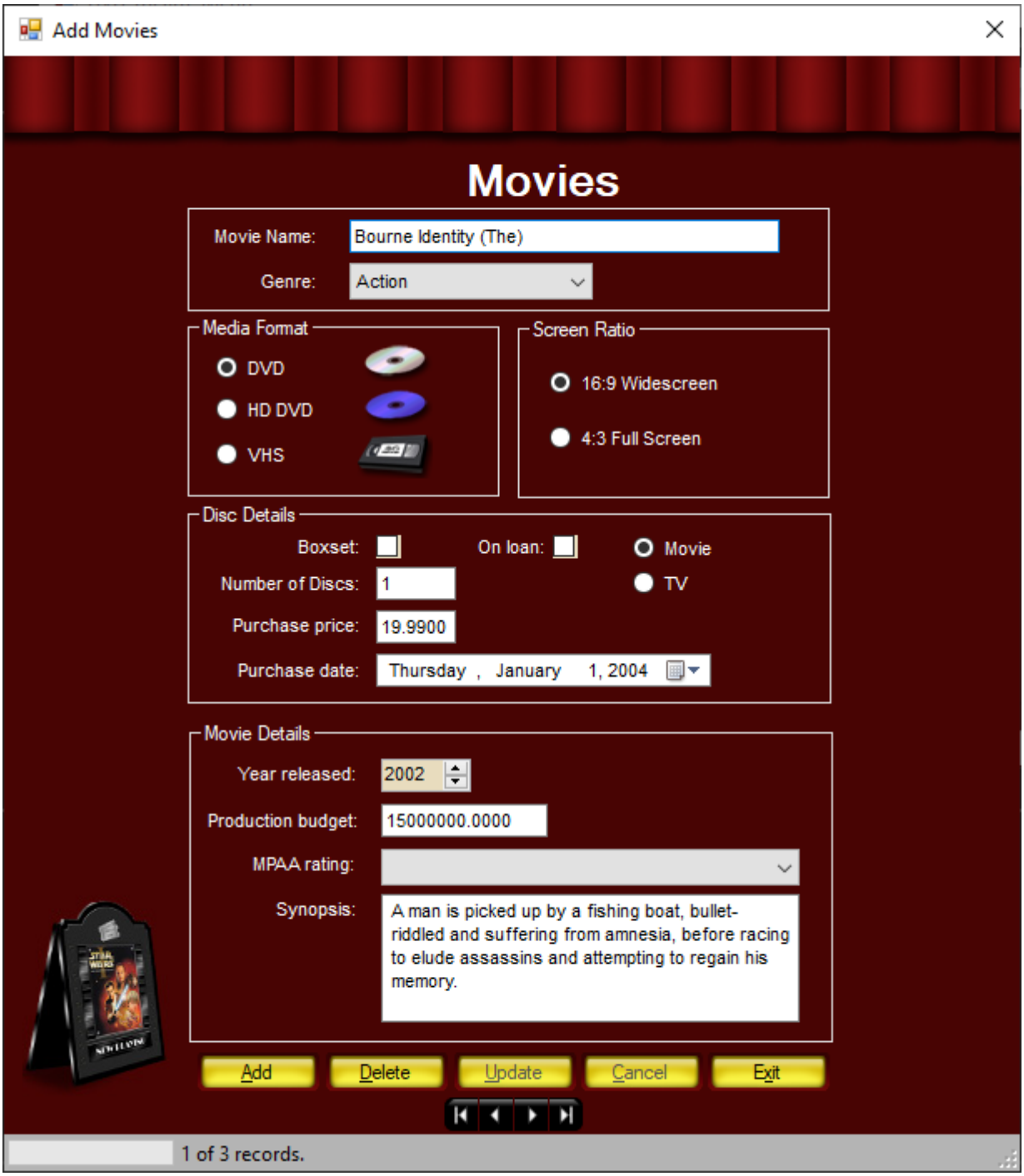
Examples

MDI Container



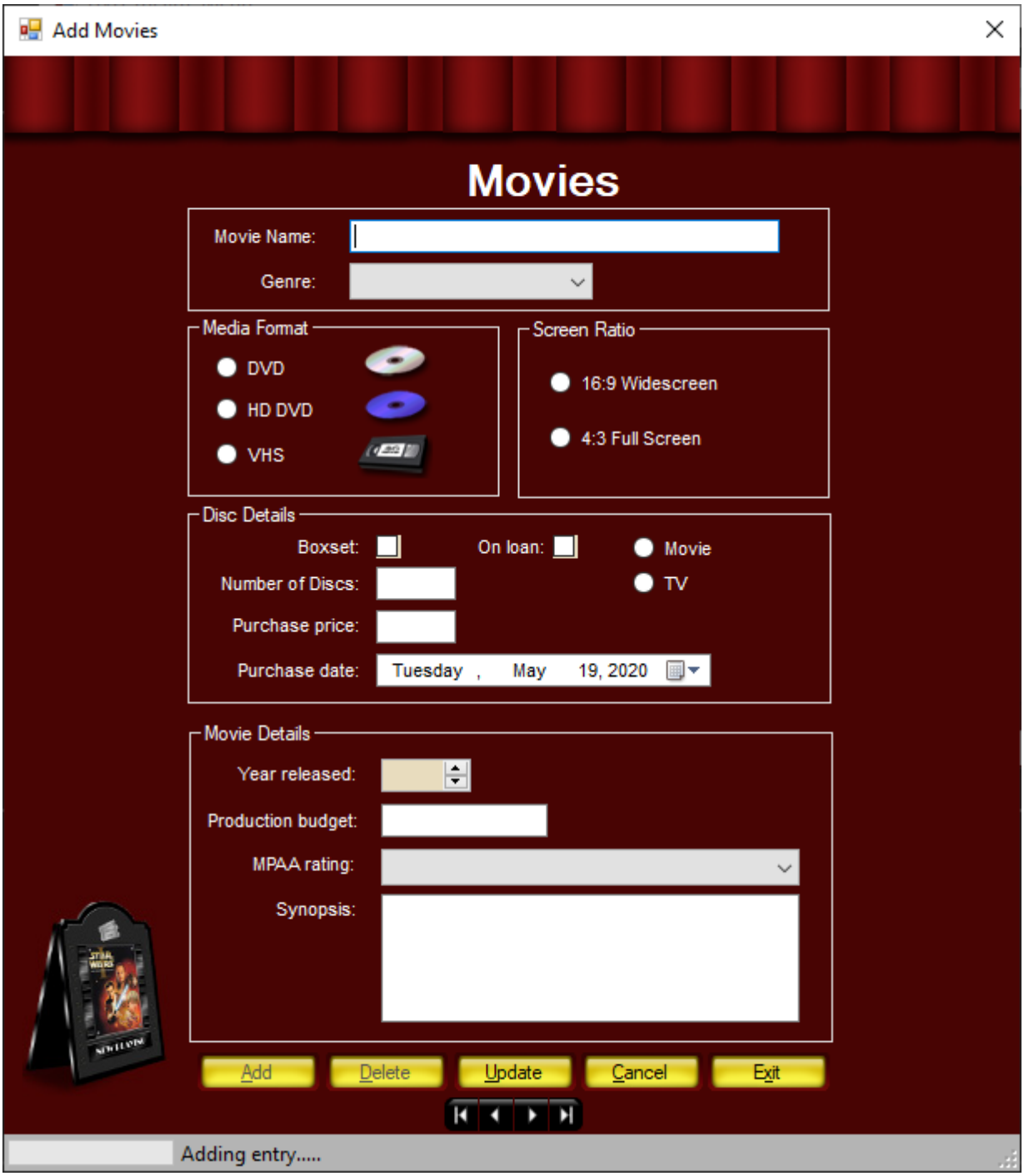
PROG1121J Database Programming Project

Entity Maintenance Form



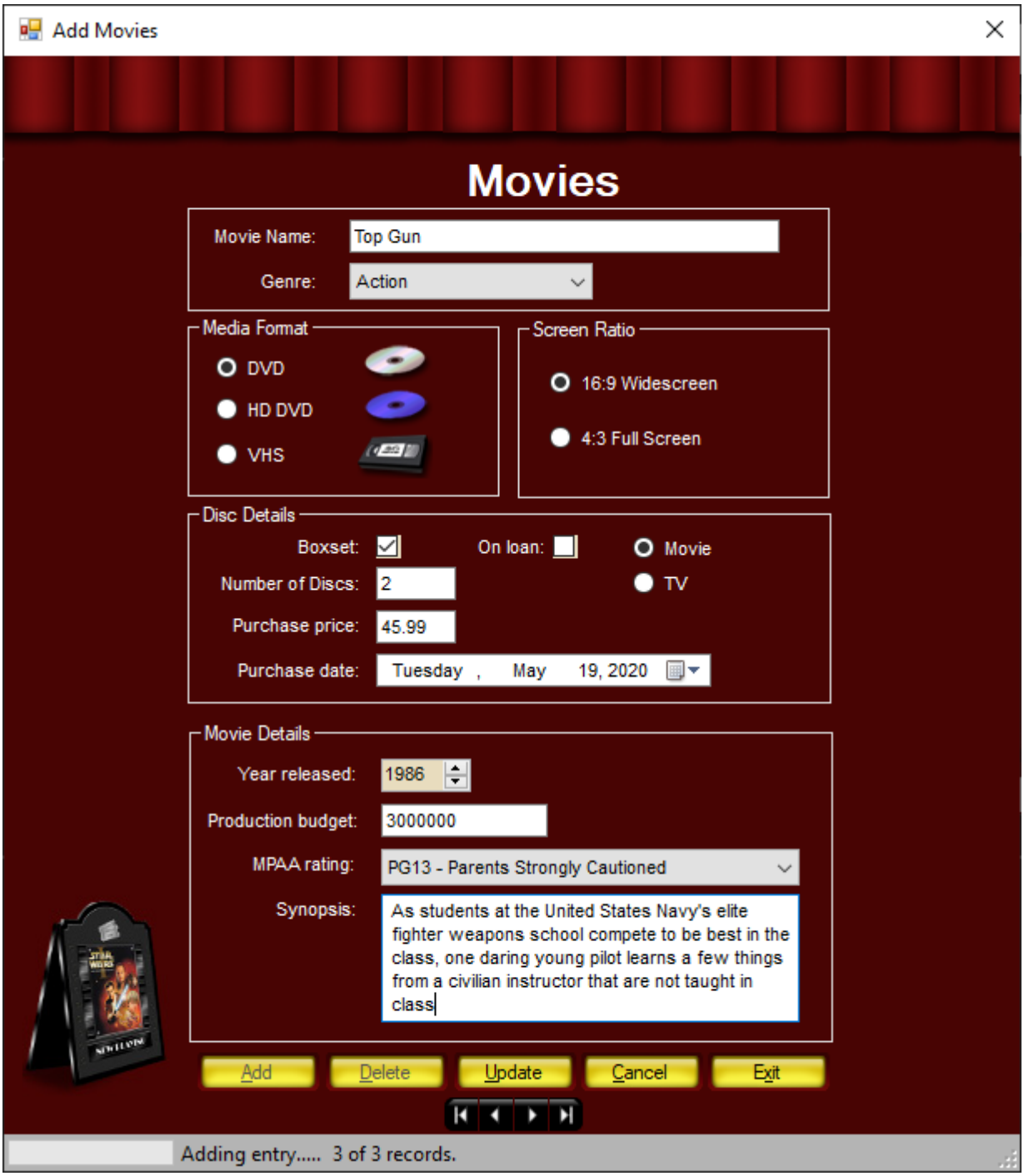
NBCC

PROG1121J Database Programming Project

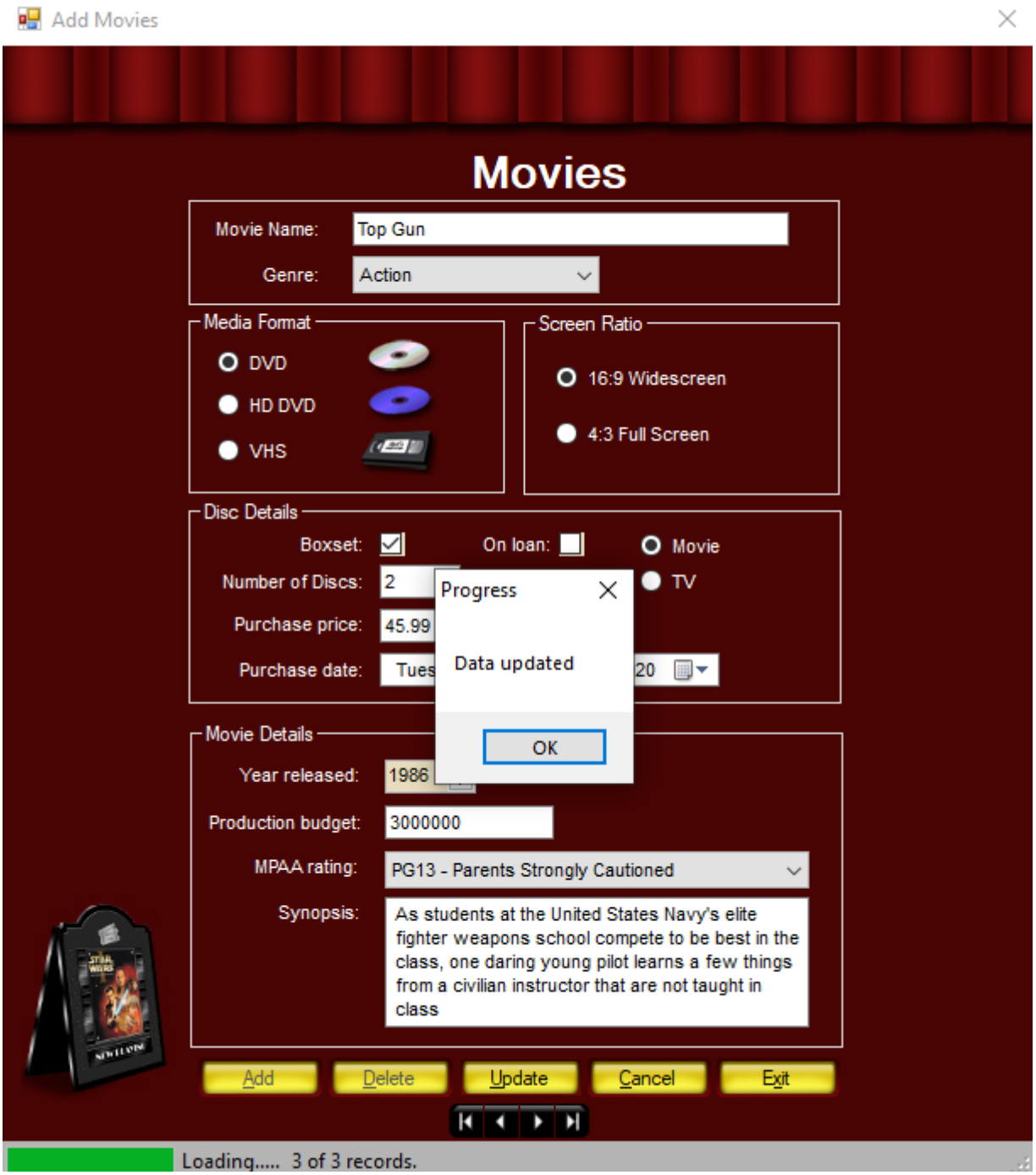


NBCC

PROG1121J Database Programming Project



PROG1121J Database Programming Project



PROG1121J Database Programming Project

Browse Forms