John Wesley Williams
Eduv9275706

ITPGA0 PROJECT

ITPGA0 DOCUMENTATION

INDEX

John Wesley Williams
Eduv9275706

2.1 Description od application:

The Video Store Management System is a client-server application that used to manages customer and movie registrations, movie rentals and returns , and to ensure that only registered customers can access the system. The system has a backend server which handles the database operations and a frontend client server which provides the user interface.

2.2 Installation and setup

To install we simply download MySQL WORKBENCH and finish setting it up. Then ew create our database, CREATE DATABASE IF NOT EXISTS eduvos; Next we create our 3 tables and insert data into them. Now we install a python idle, be sure to add the mysql.connector package to your idle to make sure it runs. Now we use our idle to create the servers. Once done make sure you run your system server first in one console then open another and run your client server. Make sure you have mysql open in the background.

Main Menu Options:

1. Register Customer

- Enter customer phone number (10 digits)
- System checks if customer exists
- If new customer, enter: First Name, Surname, Address
- Customer ID is automatically generated

2. Register Movie

- Enter movie name (max 40 characters)
- Select movie type: R (Red box - New) or B (Black box - Old)
- Video ID and Version are automatically generated

3. Hire Movie

- Enter customer phone number for verification
- Enter Video ID of requested movie
- System checks availability
- Movie is marked as hired with current date

4. Return Movie

- Enter Video ID of movie to return
- System updates return date to current date
- Movie becomes available for hire again

5. Exit System

- Closes the application gracefully

3. Directions on how to handle error messages

Common Error Messages:

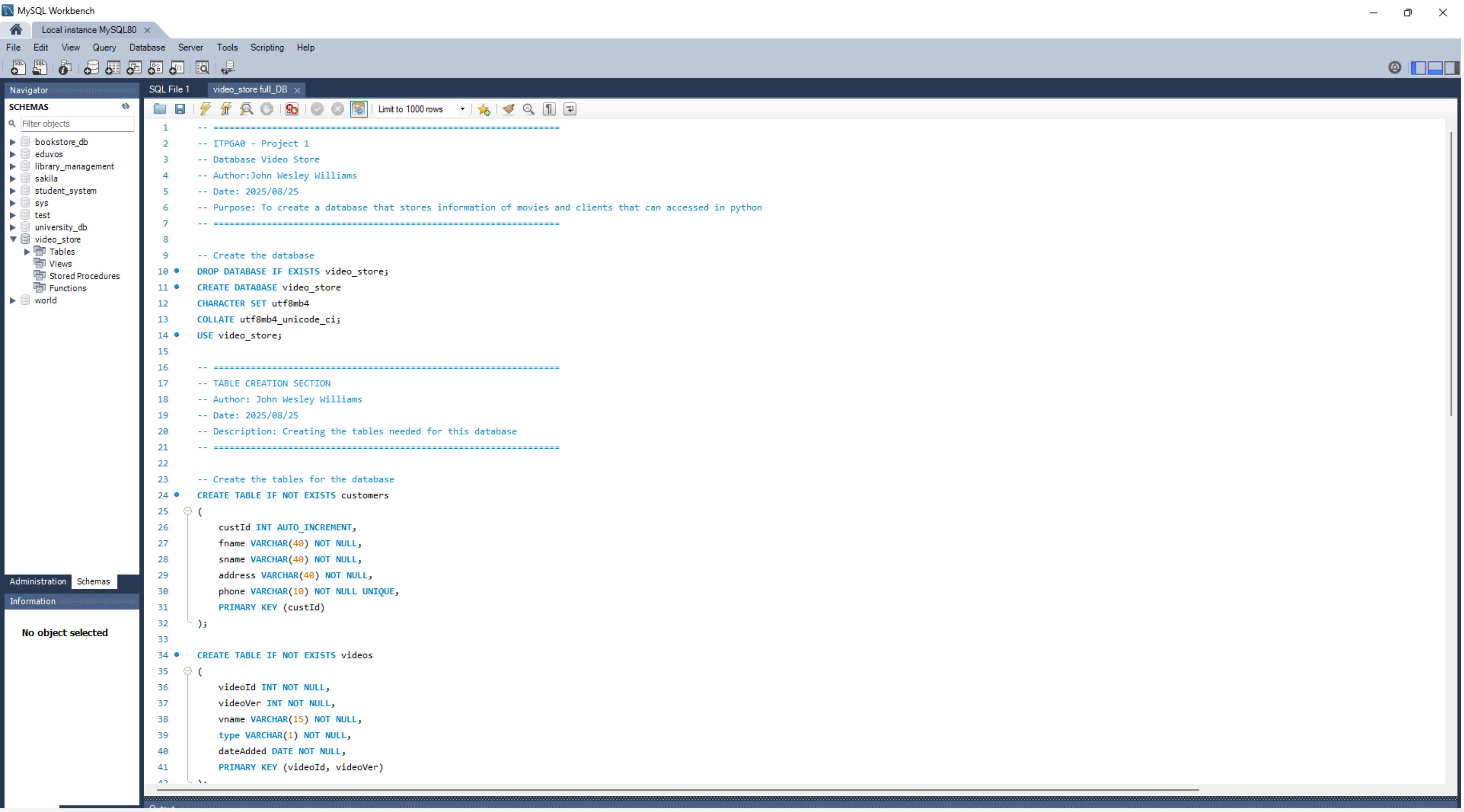
- "Customer already exists" – Customer is already registered on the system
- "Movie has already been rented" – Movie is not available for renting as it hs been rented by another customer.
- "Movie has not been rented or already returned" – This means the movie is available for renting.
- "Invalid phone number format" - Phone number validation has failed due to not inputting digits.
- "Unable to connect to server" – The server is facing network related issues.

Troubleshooting:

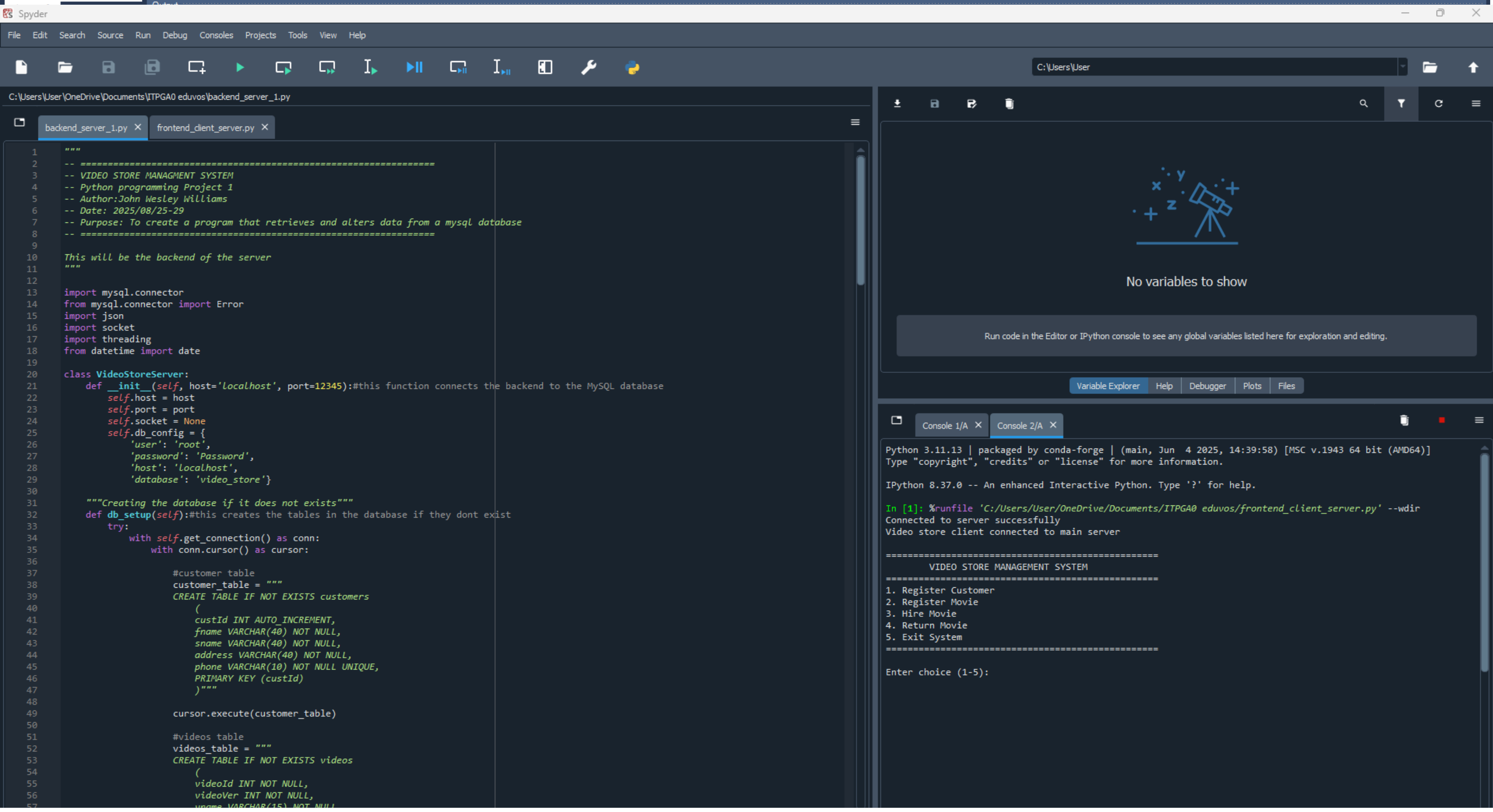
1. Connection Issues: Check if backend server is running
2. Database Errors: Verify MySQL port details are correct and database existence
3. Port Conflicts: Change port number if 12345 is occupied
4. Firewall Issues: Ensure port 12345 is not blocked

4.Screenshots of the entire application.

John Wesley Williams
Eduv9275706



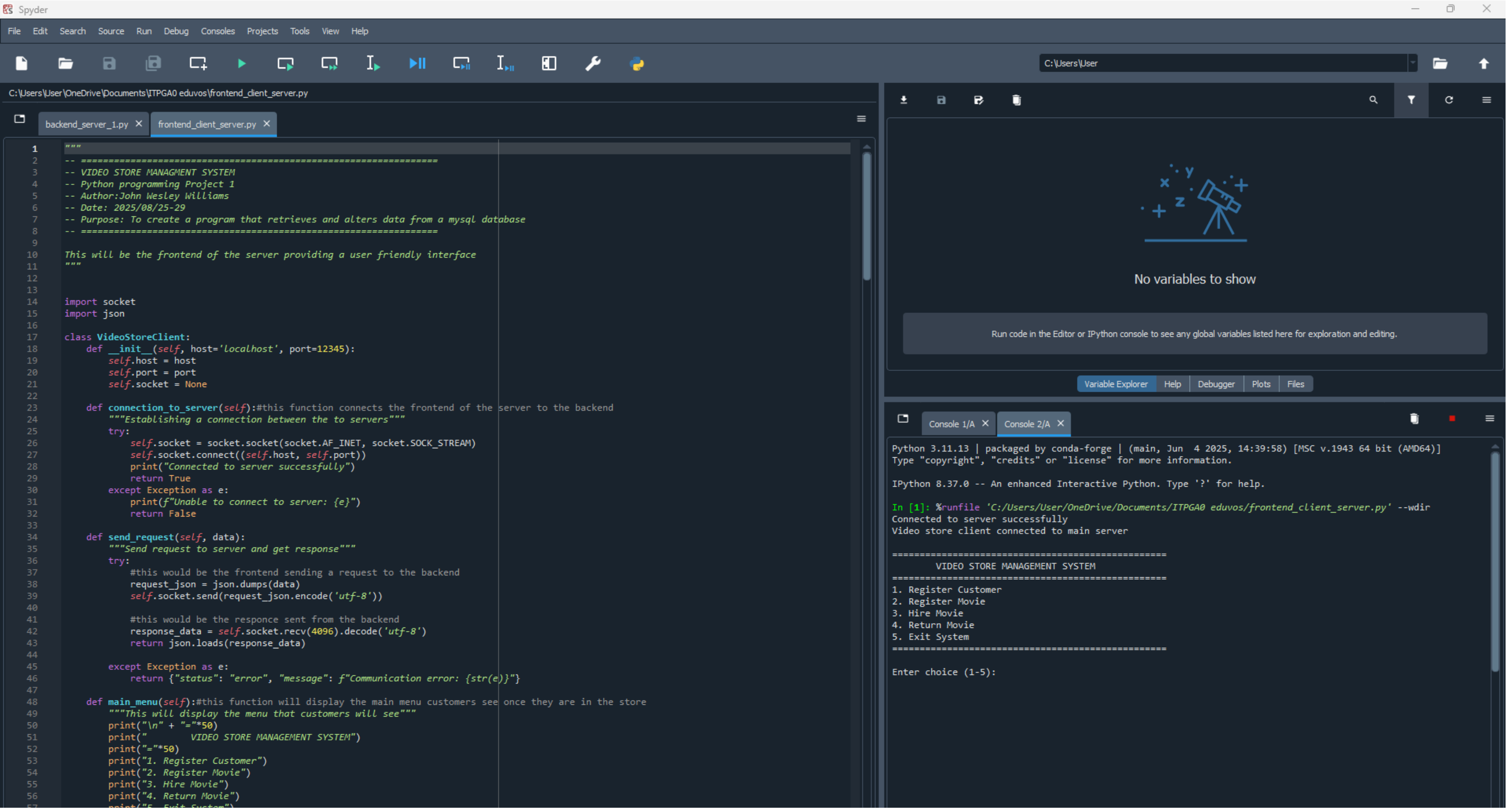
```sql
-- ============================================================
-- ITPGA0 - Project 1
-- Database Video Store
-- Author:John Wesley Williams
-- Date: 2025/08/25
-- Purpose: To create a database that stores information of movies and clients that can accessed in python
-- ============================================================

-- Create the database
DROP DATABASE IF EXISTS video_store;
CREATE DATABASE video_store
CHARACTER SET utf8mb4
COLLATE utf8mb4_unicode_ci;
USE video_store;

-- ============================================================
-- TABLE CREATION SECTION
-- Author: John Wesley Williams
-- Date: 2025/08/25
-- Description: Creating the tables needed for this database
-- ============================================================

-- Create the tables for the database
CREATE TABLE IF NOT EXISTS customers
(
    custId INT AUTO_INCREMENT,
    fname VARCHAR(40) NOT NULL,
    sname VARCHAR(40) NOT NULL,
    address VARCHAR(40) NOT NULL,
    phone VARCHAR(10) NOT NULL UNIQUE,
    PRIMARY KEY (custId)
);

CREATE TABLE IF NOT EXISTS videos
(
    videoId INT NOT NULL,
    videoVer INT NOT NULL,
    vname VARCHAR(15) NOT NULL,
    type VARCHAR(1) NOT NULL,
    dateAdded DATE NOT NULL,
    PRIMARY KEY (videoId, videoVer)
);
```


Spyder
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\User
C:\Users\User\OneDrive\Documents\ITPGA0 eduvos\backend_server_1.py
backend_server_1.py
frontend_client_server.py
No variables to show
Run code in the Editor or IPython console to see any global variables listed here for exploration and editing.
Variable Explorer
Help
Debugger
Plots
Files
Console 1/A
Console 2/A

```python
"""
-- ================================================================
-- VIDEO STORE MANAGMENT SYSTEM
-- Python programming Project 1
-- Author:John Wesley Williams
-- Date: 2025/08/25-29
-- Purpose: To create a program that retrieves and alters data from a mysql database
-- ================================================================

This will be the backend of the server
"""

import mysql.connector
from mysql.connector import Error
import json
import socket
import threading
from datetime import date

class VideoStoreServer:
    def __init__(self, host='localhost', port=12345):#this function connects the backend to the MySQL database
        self.host = host
        self.port = port
        self.socket = None
        self.db_config = {
            'user': 'root',
            'password': 'Password',
            'host': 'localhost',
            'database': 'video_store'}

    """Creating the database if it does not exists"""
    def db_setup(self):#this creates the tables in the database if they dont exist
        try:
            with self.get_connection() as conn:
                with conn.cursor() as cursor:

                    #customer table
                    customer_table = """
                    CREATE TABLE IF NOT EXISTS customers
                        (
                        custId INT AUTO_INCREMENT,
                        fname VARCHAR(40) NOT NULL,
                        sname VARCHAR(40) NOT NULL,
                        address VARCHAR(40) NOT NULL,
                        phone VARCHAR(10) NOT NULL UNIQUE,
                        PRIMARY KEY (custId)
                        )"""

                    cursor.execute(customer_table)

                    #videos table
                    videos_table = """
                    CREATE TABLE IF NOT EXISTS videos
                        (
                        videoId INT NOT NULL,
                        videoVer INT NOT NULL,
                        vname VARCHAR(15) NOT NULL
```

```
Python 3.11.13 | packaged by conda-forge | (main, Jun  4 2025, 14:39:58) [MSC v.1943 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.37.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: %runfile 'C:/Users/User/OneDrive/Documents/ITPGA0 eduvos/frontend_client_server.py' --wdir
Connected to server successfully
Video store client connected to main server

==================================================
        VIDEO STORE MANAGEMENT SYSTEM
==================================================
1. Register Customer
2. Register Movie
3. Hire Movie
4. Return Movie
5. Exit System
==================================================

Enter choice (1-5):
```

John Wesley Williams
Eduv9275706


Spyder
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\User
C:\Users\User\OneDrive\Documents\ITPGA0 eduvos\frontend_client_server.py
backend_server_1.py
frontend_client_server.py
No variables to show
Variable Explorer
Help
Debugger
Plots
Files
Console 1/A
Console 2/A

Run code in the Editor or IPython console to see any global variables listed here for exploration and editing.

```python
"""
-- =================================================================
-- VIDEO STORE MANAGMENT SYSTEM
-- Python programming Project 1
-- Author:John Wesley Williams
-- Date: 2025/08/25-29
-- Purpose: To create a program that retrieves and alters data from a mysql database
-- =================================================================

This will be the frontend of the server providing a user friendly interface
"""


import socket
import json

class VideoStoreClient:
    def __init__(self, host='localhost', port=12345):
        self.host = host
        self.port = port
        self.socket = None

    def connection_to_server(self):#this function connects the frontend of the server to the backend
        """Establishing a connection between the to servers"""
        try:
            self.socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            self.socket.connect((self.host, self.port))
            print("Connected to server successfully")
            return True
        except Exception as e:
            print(f"Unable to connect to server: {e}")
            return False

    def send_request(self, data):
        """Send request to server and get response"""
        try:
            #this would be the frontend sending a request to the backend
            request_json = json.dumps(data)
            self.socket.send(request_json.encode('utf-8'))

            #this would be the responce sent from the backend
            response_data = self.socket.recv(4096).decode('utf-8')
            return json.loads(response_data)

        except Exception as e:
            return {"status": "error", "message": f"Communication error: {str(e)}"}

    def main_menu(self):#this function will display the main menu customers see once they are in the store
        """This will display the menu that customers will see"""
        print("\n" + "="*50)
        print("        VIDEO STORE MANAGEMENT SYSTEM")
        print("="*50)
        print("1. Register Customer")
        print("2. Register Movie")
        print("3. Hire Movie")
        print("4. Return Movie")
        print("5. Exit System")
```

```
Python 3.11.13 | packaged by conda-forge | (main, Jun  4 2025, 14:39:58) [MSC v.1943 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.37.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: %runfile 'C:/Users/User/OneDrive/Documents/ITPGA0 eduvos/frontend_client_server.py' --wdir
Connected to server successfully
Video store client connected to main server

==================================================
        VIDEO STORE MANAGEMENT SYSTEM
==================================================
1. Register Customer
2. Register Movie
3. Hire Movie
4. Return Movie
5. Exit System
==================================================

Enter choice (1-5):
```

Bibliography:

https://mylms.vossie.net/course/view.php?id=31666

https://mylms.vossie.net/course/view.php?id=31103

I used everything I could find in the mylms course as reference