# Eduv9173267 – ITPGA Project Document

# Table of Contents

Table of Contents

[Eduv9173267 – ITPGA Project Document 1](#_Toc207131981)

[Table of Contents 2](#_Toc207131982)

[Project Planning 3](#_Toc207131983)

[What will be covered 3](#_Toc207131984)

[Database Planning 3](#_Toc207131985)

[Designing the database 3](#_Toc207131986)

# Project Planning

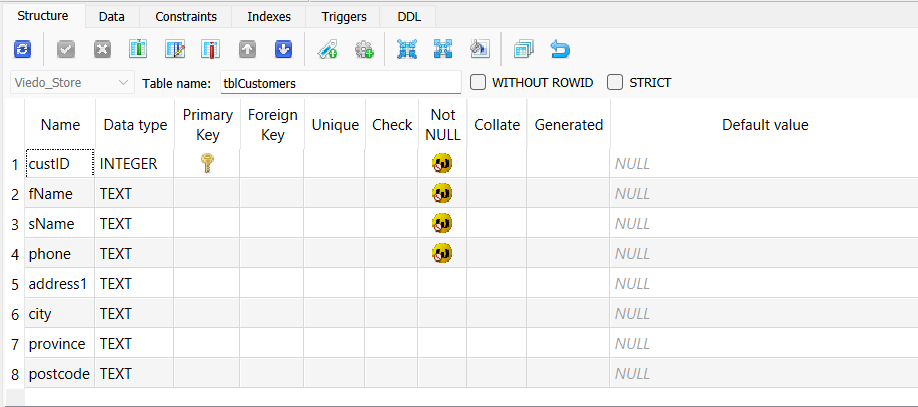
## What will be covered

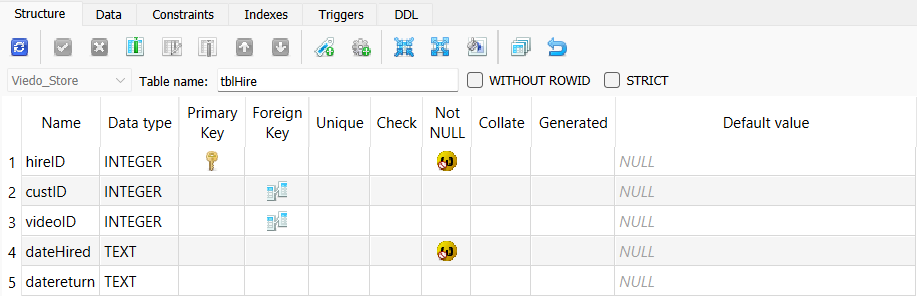
* Program Design
* Project source code
* User documentation
* Databases
  + As well as relevant design planning
* Resources
* System Requirements
  + Optional
* All Rough work

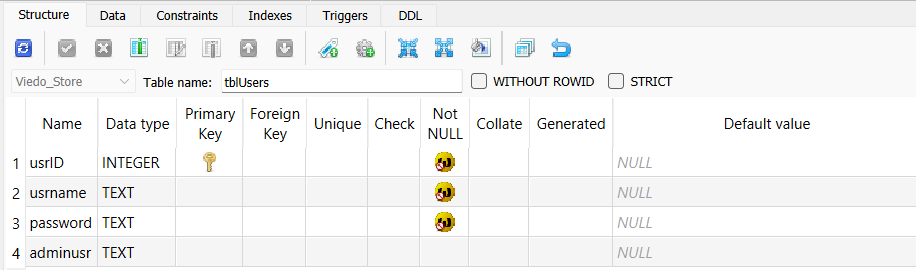
## Database Planning

* The rough planning that was done with pen and paper can be found toward the end of this document.
* The use of SQLite in this project.
  + I will be using SQLite for this project as it has (in my opinion) the most pain free deployment when the database is run locally.

### Designing the database

Seen in the below images are the data structures of all the tables in the database. These images were taken in SQLite Studio GUI. The name of the table can be seen above the “Foreign key” column of the interface

* A screenshot of a computer

  AI-generated content may be incorrect.I would like to note, in the above image “dateAdded” has the “TEXT” data type, SQLite does not specifically have a date and time type. So, when doing my program I will convert the date/time into a String before committing it to the database.
* You will notice that I have added a “hireID” field as the primary key, this was not mentioned in the project specifications but is needed in this case.

## Program Design Planning

### The flow of the application

* User logs in
* The Main Dashboard opens with the following options:
  + Register Customer
    - Data validation, ensure the phone number is unique and has not been used by another customer
  + Register Movie
    - Users must enter the movie name, and type
    - Movies can only be one of the two types, red box, or Black box. Red for new movies, and Black for old movies.
    - If the movie name is the same, only the version should be incremented
  + Add User
    - Prompts the user to enter the details for the new user.
  + Remove User
    - The user must enter to userID of the user they wish to remove
  + Hire out movie
    - The user will be prompted to enter the customers phone number, then the user enterers the videoID of the movie the customer has selected
  + Return move
    - The user will be prompted to enter the videoID of the movie being returned.

### Code planning

* Load tblUsers into an array (Just the usernames and passwords)
  + Display dashboard if valid
  + Else display error
* The use of Dictionary in python. The tables will be read into a dictionary at the start of an operation for easy lookup when needed