



COLLEGE OF COMPUTING, INFORMATICS, AND MATHEMATICS

UNIVERSITI TEKNOLOGI MARA

MERBOK, KEDAH

DIPLOMA IN LIBRARY INFORMATICS

(IM144)

PROGRAMMING FOR LIBRARIES

(IML209)

INDIVIDUAL ASSIGNMENT:

NIGHT MOVIE SYSTEM

PREPARED BY:

HAWA NABILA BINTI HANIF (2022625444)

CLASS:

KCDIM144 3F

PREPARED FOR:

SIR. AIRUL SHAZWAN BIN NORSHAHIMI

SUBMISSION DATE:

4 JANUARY 2024

INDIVIDUAL ASSIGNMENT: NIGHT MOVIE SYSTEM

PREPARED BY:

HAWA NABILA BINTI HANIF (2022625444)

COLLEGE OF COMPUTING, INFORMATICS, AND MATHEMATICS

UNIVERSITI TEKNOLOGI MARA

MERBOK, KEDAH

4 JANUARY 2024

## **AKNOWLEDGMENT**

First and foremost, I am overwhelmed in all humbleness and gratefulness to acknowledge my debt to all those who helped me to put these ideas, well above the level of simplicity and into something concrete. I would like to express my special thanks of gratitude to my lecturer, Sir Airul Shazwan bin Norshahimi for encouraging me to continue to write and gives me guidance to complete this assignment which also helped me in doing a lot of research and I came to know about so many new things. Without his help, it is impossible to finish this assignment on time. Secondly, I want to give appreciation to my parents who have continuously supported me from home to help me accomplish this assignment and their guidance, encouragement, and inspiration to me throughout their lives. Next, I also would like to thank my fellow classmates and friends for their help on this task. This assignment makes me more confident to continue my studies and this will be a memorable moment for me.

## **TABLE OF CONTENT**

### **ACKNOWLEDGMENT**

<b>1.0 INTRODUCTION .....</b>	<b>1</b>
<b>2.0 FLOWCHART .....</b>	<b>2</b>
<b>3.0 PYTHON CODE.....</b>	<b>3</b>
<b>4.0 GRAPHICAL USER INTERFACE (GUI) .....</b>	<b>4</b>
<b>5.0 DATABASE.....</b>	<b>5</b>

## 1.0 INTRODUCTION

The Night Movie System emerges as a comprehensive solution for organizing and handling, with a unique focus on delivering films that resonate with and heal your inner child, during the ever-changing entertainment environment. This revolutionary solution, designed specifically for student users, effortlessly blends the power of MySQL's strong relational database management system with an intuitive graphical user interface, giving users a complex and user-friendly movie management experience.

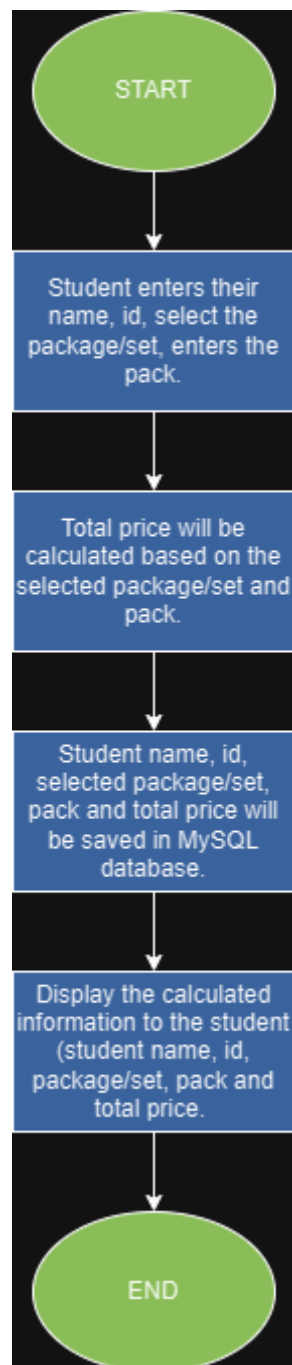
The Night Movie System uses MySQL to store and organize massive quantities of data relating to the chosen movie and user-specific information. MySQL's dependability and scalability allow for the efficient processing of a wide range of data, including student-specific features such as student name, student ID, package type, package packs, and package prices.

With MySQL at its core, the Night Movie System assures data integrity and security not just for movie-related information but also for student information. The relational database structure enables the development of relationships between various data pieces, therefore reducing inconsistencies and improving overall data dependability. User authentication and access restrictions help to strengthen the system's security safeguards.

The Night Movie System prioritizes user experience by providing an easy and visually appealing graphical user interface (GUI) that seamlessly fits student-specific features. Drag-and-drop capability, adaptable design, and a simple layout make it simple to navigate and manage the selected movie and student data. Easy data entry is made possible via straightforward input fields. Students may enter their name and ID, select a set from a dropdown menu, determine the number of persons in each pack, and then click the "Calculate" button. The input is then smoothly processed by the system for a rapid and precise pricing estimate.

Lastly, the Night Movie System goes above and beyond basic movie management by selecting a collection of films particularly intended to cure students' inner child. These carefully chosen films seek to induce nostalgia, joy, and awe, providing users with a healing and enlightening experience.

## 2.0 FLOWCHART




## 3.0 PYTHON CODE

```
File Edit Selection View Go Run Terminal Help ← → SOURCE CODE
night_movie.py X data.sql
C:\Users\HP> HP> OneDrive> Desktop> 208 assignment> night_movie.py > ...
1 import tkinter as tk
2 import mysql.connector
3
4 # Connect to database
5 mydb = mysql.connector.connect(
6     host="localhost",
7     user="root",
8     password="",
9     database="night_movie"
10 )
11
12 # Create a cursor object to execute SQL queries
13 mycursor = mydb.cursor()
14
15 # Function to handle the calculation and database saving
16 def submit():
17     student_name = student_name_entry.get()
18     student_id = student_id_entry.get()
19     set_type = package_var.get()
20     packs = int(packs_entry.get())
21
22     # the price below is to defined the value from selections
23     prices = {
24         "Set A": 6,
25         "Set B": 9,
26         "Set C": 13,
27     }
28
29
30 # Calculate the total price. This will be derived from selection (Package, Pack).
31 total_price = (prices[set_type] * packs)
32
33 # To insert data to database
34 sql = "INSERT INTO data (student_name, student_id, package_type, package_pack, package_price) VALUES (%s, %s, %s, %s, %s)"
35 val = (student_name, student_id, set_type, packs, total_price)
36 mycursor.execute(sql, val)
```

```
File Edit Selection View Go Run Terminal Help ← → SOURCE CODE
night_movie.py X data.sql
C:\Users\HP> HP> OneDrive> Desktop> 208 assignment> night_movie.py > ...
33 # To insert data to database
34 sql = "INSERT INTO data (student_name, student_id, package_type, package_pack, package_price) VALUES (%s, %s, %s, %s, %s)"
35 val = (student_name, student_id, set_type, packs, total_price)
36 mycursor.execute(sql, val)
37 mydb.commit()
38
39 # To print back the output. It will happen in the function collect_data(). the f before the string indicates an f-string in Python.
40 output_label.config(text="Student: {student_name}\nId: {student_id}\nSet: {set_type}, Packs: {packs}, Total Price: RM{total_price}")
41
42
43 # Main Window
44 root = tk.Tk()
45 root.title("Movies")
46 root.geometry('600x700')
47
48 # Page Title
49 label = tk.Label(root, text='Calculate your Set Price', font=("Times New Romans",30, "bold"))
50 label.pack(ipadx=10, ipady=10)
51
52
53 # Prices List by using textbox
54 prices_text = tk.Text(root, height=15, width=45)
55 prices_text.pack(pady=20)
56
57 # The defined list by using pricebox
58 prices_text.insert(tk.END, "Set & Prices:\n\n")
59 prices_text.insert(tk.END, "Set A: Cloudy with a Chance of Meatballs, Popcorn, Soda \nPrice: RM6\n\n")
60 prices_text.insert(tk.END, "Set B: Cloudy with a Chance of Meatballs, Popcorn, Soda, Freeslift \nPrice: RM9\n\n")
61 prices_text.insert(tk.END, "Set C: Cloudy with a Chance of Meatballs, Popcorn, Soda, Meatballs, Blanket \nPrice: RM13\n\n")
62 prices_text.config(state="disabled")
63
64 # Student Name Entry
65 student_name_label = tk.Label(root, text="Student Name:", font= ("Times New Roman",14))
66 student_name_label.pack()
67 student_name_entry = tk.Entry(root)
68 student_name_entry.pack()
```

```
File Edit Selection View Go Run Terminal Help ← → SOURCE CODE
night_movie.py X data.sql
C:\Users\HP> HP> OneDrive> Desktop> 208 assignment> night_movie.py > ...
69 student_name_entry.pack()
70
71 # Student ID Entry
72 student_id_label = tk.Label(root, text="Student Id:", font= ("Times New Roman",14))
73 student_id_label.pack()
74 student_id_entry = tk.Entry(root)
75 student_id_entry.pack()
76
77 # Set Type Dropdown (Label)
78 packs_label = tk.Label(root, text="Choose Your Package:", font= ("Times New Roman",14))
79 packs_label.pack()
80
81 # Set Type Dropdown
82 package_var = tk.StringVar(root)
83 package_var.set("Select your Set") # Default value before your selection
84 trip_dropdown = tk.OptionMenu(root, package_var, "Set A", "Set B", "Set C")
85 trip_dropdown.pack(pady=10)
86
87 # Packs Entry
88 packs_label = tk.Label(root, text="Packs:")
89 packs_label.pack()
90 packs_entry = tk.Entry(root)
91 packs_entry.pack()
92
93 # Save Button
94 save_button = tk.Button(root, text="Calculate", command=submit)
95 save_button.pack(pady=10)
96
97 # Output Label & result
98 label = tk.Label(root, text="Price Package", font=("Times New Romans",12))
99 label.pack(ipadx=10, ipady=10)
100 output_label = tk.Label(root, text="")
101 output_label.pack()
102
103 root.mainloop()
```

## 4.0 GRAPHICAL USER INTERFACE (GUI)

 Movies

—

□

×

# Calculate your Set Price

Set & Prices:

SetA: Cloudy with a Chance of Meatballs, Popcorn, Soda  
Price: RM6


Set B: Cloudy with a Chance of Meatballs, Popcorn, Soda, Freegift  
Price: RM9

Set C: Cloudy with a Chance of Meatballs, Popcorn, Soda, Meatballs, Blanket  
Price: RM13

Student Name:

Student Id:

Choose Your Package:  

Select your Set 

Packs:

Calculate

Price Package



## 5.0 DATABASE

