UCS 2201

FUNDAMENTALS AND PRACTICES OF SOFTWARE DEVELOPMENT

TIMETABLE MANAGEMENT SYSTEM

Review 4

MOVIE SCHEDULING IN A MULTIPLEX MALL

- **A REPORT**

Presented By:

Team Best Practices- Batch 2

Heera Ethirajan – 3122 21 5001 033

Hemasree Senthil – 3122 21 5001 034

Jahnavi Murali – 3122 21 5001 038

Submitted On 26.08.2022

Signature: Team members

Signature: Faculty

TABLE OF CONTENTS

LIST OF FIGURES

1. Abstract

The scheduling of a movie theatre is one such problem, existing in the film industry, that is quite amenable to model building and optimization.

In this project, we have used a two-tier approach to schedule movies in a multiplex mall of capacity ‘N’ screens and 6 slots per screen.

In the first tier, we generate the priority score for each movie from the information we collected through user survey. Using the calculated priority score, we compute the number of showings per movie, wherein the movie with the maximum score is allotted maximum screening.

In the second tier, we deal with our main scheduling problem, wherein we plot the movies with a higher priority score to prime showings.

This way we satisfy the demand of the audience, ensure success of the movies, and maximize the theatre’s revenue/ profit.

Keywords: Movie Scheduling, Priority, Slots, Prime slots, Screens, Python, Django, HTML, CSS, Optimization, Profit.

2. Introduction

The motion picture industry is a prominent economic activity with total worldwide box office revenue of $ 21.3 billion in 2021, of which $ 412 million is in India. Many problems in the film industry are quite suitable/apt to model building and optimization, as movie executives have increasingly recognized. One such problem is the detailed scheduling of a movie theater.

Presently, this programming is mostly done manually with pencil and paper by specialists in the theater company. It is believed that an analytical system can help in movie programming. This would not only relieve theaters from a repetitive labor-intensive task, but also achieve a better performance in scheduling than the current mental, manual procedure.

3. Review of Literature

Appropriate international literature pertaining to movie scheduling and timetable optimization has been cited below.

Jehoshua Eliashberg, Quintus Hegie et al. proposed a model where the movie scheduling problem is formulated as a generalized set partitioning problem. The latter is solved with an algorithm based on column generation techniques. The proposed model not only made movie scheduling easier and less time consuming, but also generated schedules that would attract more visitors than the current ‘intuition-based’ schedules.

Ahmad Muklason et al. proposed methodology that could be divided into two main parts: the method to generate initial feasible solution and the method to optimize the initial solution. To generate initial solution, a greedy algorithm was employed whereas to optimize the initial solution, tabu-search algorithm was employed.

Sanjeev Swami et al. proposed a two-tier integrated application of the model to show how the model can be applied to realistic decision making. This application involves updating data, from one time window to another. The approaches followed in the two tiers of the integrated application are quite general in that they can incorporate a sophisticated demand prediction model, managerial judgments, or a combination of both.

4. Design

4.1. Architecture design:

Fig. 4.1. (a) Website/Frontend architecture

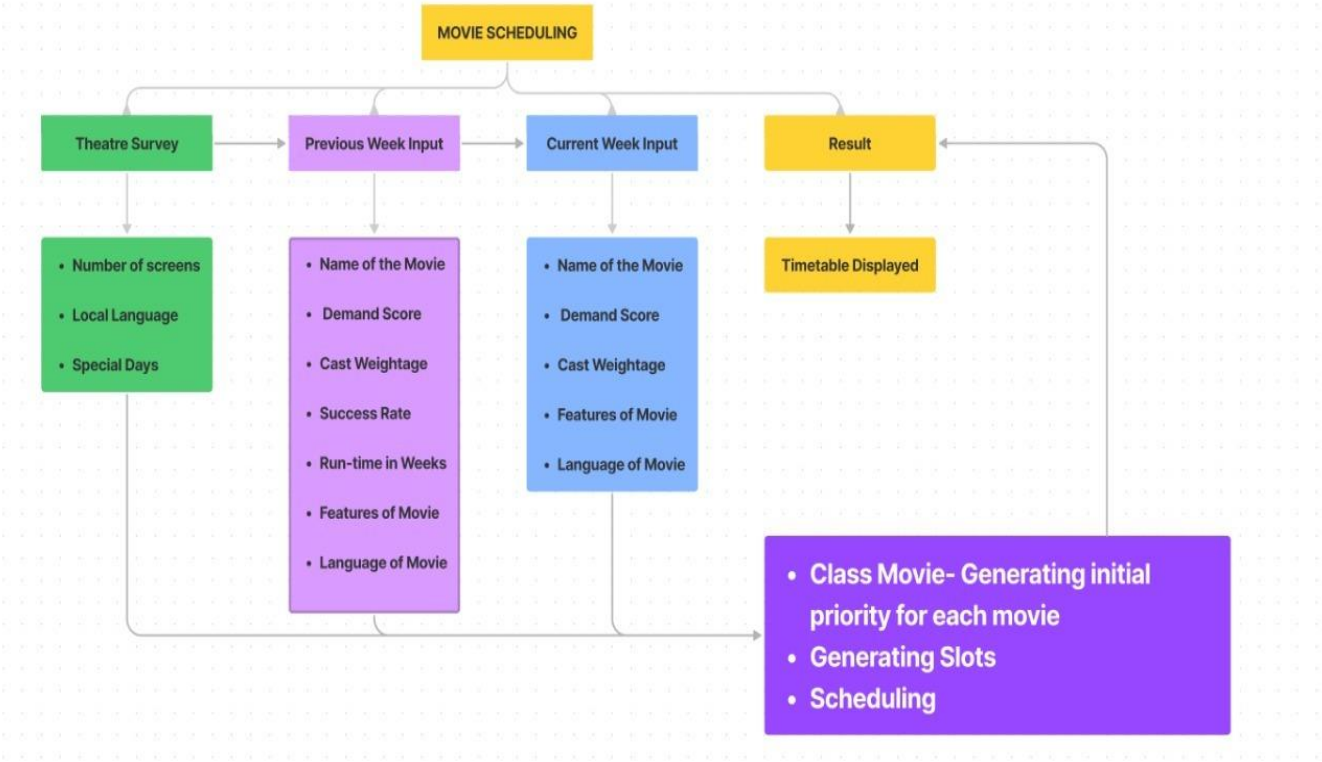
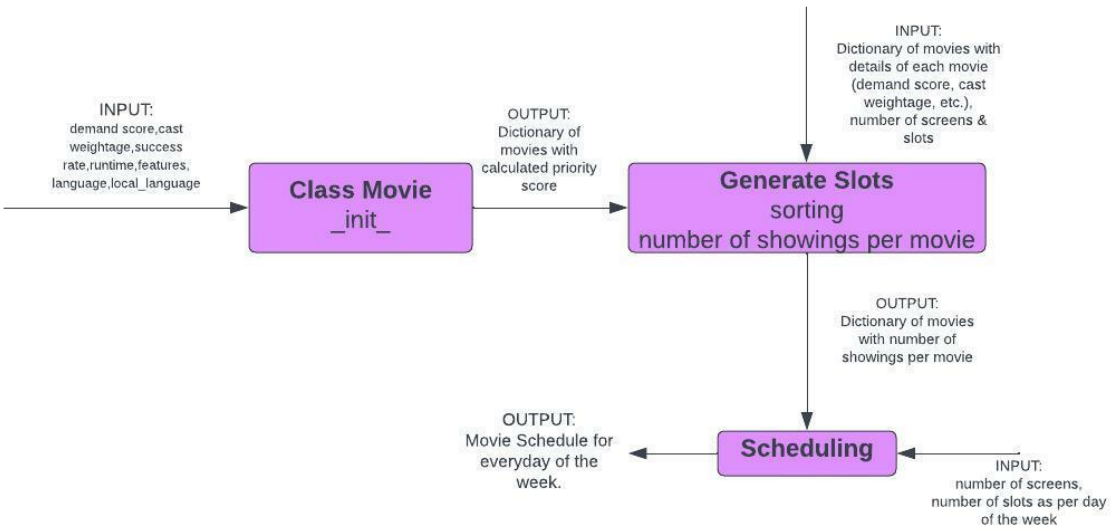


Fig 4.1.(b) Backend architecture



4.2. Module design:

Algorithm: Class Movie- function _init_ (self, demand_score, cast_weight, success_rate, time_running, feature, language ,local_language):

The *class Movie* involves generation of priority score for each movie from the information we collected through user survey. The priority score is calculated by the formula: (Demand Score + Cast Weightage + Success Rate)/2. It is also influenced by several other soft constraints such as, Features, Language and Runtime of the movie and the days of the week.

1. If 20<success_rate<=50 priority reduced by 50%.
2. If success_rate<20 priority =0.
3. If runtime=0 or 1 priority will remain the same.
4. If runtime=2 priority will reduce by 25%.
5. If runtime=3 priority will reduce by 50%.
6. If runtime=4 priority will reduce by 75%.
7. If runtime>4 and demand_score>5, priority will reduce by 80%.
8. If runtime>4 and demand score<5, priority =0
9. If the language of the locality is the same as the language of the movie, priority will remain the same, else will reduce by 25%.
10. If the movie has features like 3D and Dolby Atmos, its priority will increase by .5.

Algorithm: Module GenerateSlots(Details,Movie,number_of_screens, nslots)

In the first tier, module Generate Slots, using the calculated priority score, we compute the number of showings per movie, wherein the movie with the maximum score is allotted maximum screening.

Input: Details, Movie, number_of_screens, nslots

Output: Dictionary of movies with number of showings per day.

1. Sorting the dictionary Movie by priority score(item) using sorted() function and storing in dictionary d.
2. Inverting the dictionary d to find the number of movies with same priority score and storing it in dictionary inverse.
3. Further sorting of movies with the same priority based on demand score, cast weightage and success rate. Store the sorted order of movies in a list.
4. According to the number of movies, generate a list in decreasing order of unequal percentages of number of showings(number_of_screens x nslots). Store the number of showings per movie in a dictionary p_movie of movies.
5. Check if the total number of showings generated is greater than the capacity of the theatre. If yes, reduce the number of shows from the low priority movies present at the end of p_movie accordingly.
6. Return p_movie.

Algorithm: Module Scheduling(number_of_screens, p_movie, l, u)

In the second tier-*module Scheduling*-deals with our main scheduling problem. The forthcoming week's scheduling plan is visualized as a two-dimensional (screen-by- slot) matrix, then that matrix contains only "empty cells" before the second tier. After the first tier has been successfully executed, some of those empty cells are 'filled' given the day of the week. The movies with a higher priority score to prime showings and the rest to the remaining showings.

Input: number_of_screens,p_movie, low limit of slot, upper limit of slot.

Output: Schedule per day of the week.

1. Initialize every element of the matrix schedule with '-'
2. Hard code the prime slots depending upon the values l and u to a list prime_slots.
3. Fill the prime slots first with the high priority movies with maximum showtimes first.
4. After the prime slots have been filled, fill the remaining slots with the remaining movies.
5. Return the matrix schedule.

5. Implementation and Results:

We have created a web application, integrating our scheduling algorithm, using Django and HTML and CSS. The web application includes

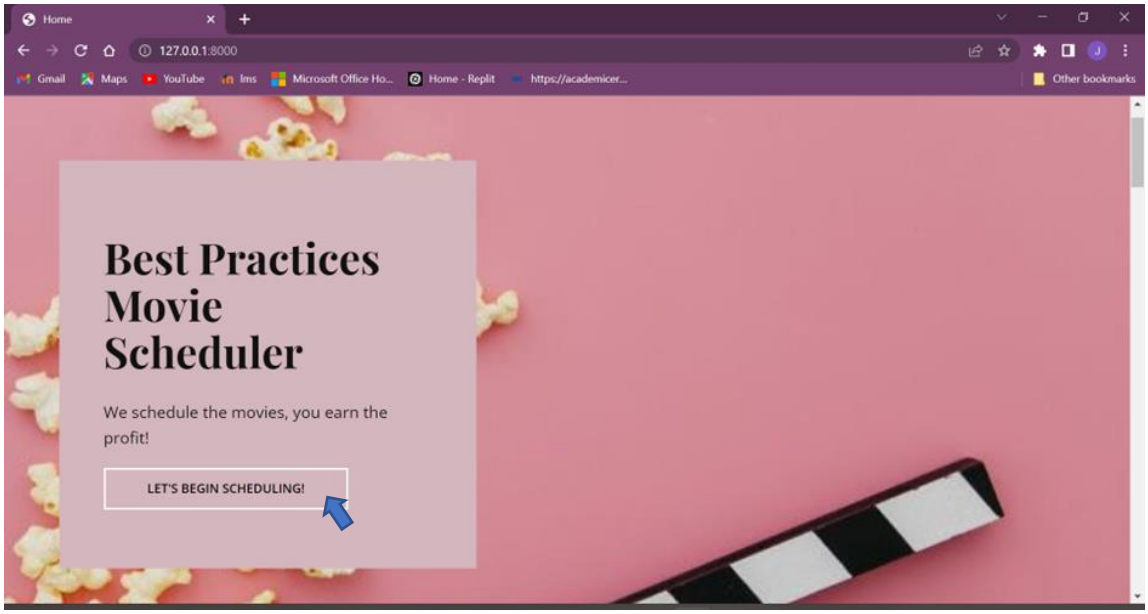
1. An about page that displays details about our algorithm and its working.
2. Three forms/ survey pages:
    i. Form to collect general theatre and locality details (number of screens, language of the locality and special days/festivities occurring in the forthcoming week).
    ii. Form to collect details about movies that are already released (Movie name, demand score, cast weightage, success rate, runtime, language, and features of the movie).

iii. Form to collect details about movies that are newly releasing that week (Movie name, demand score, cast weightage, language, and features of the movie).

3. A result page that displays the generated movie schedule for the week.

5.1. About Page

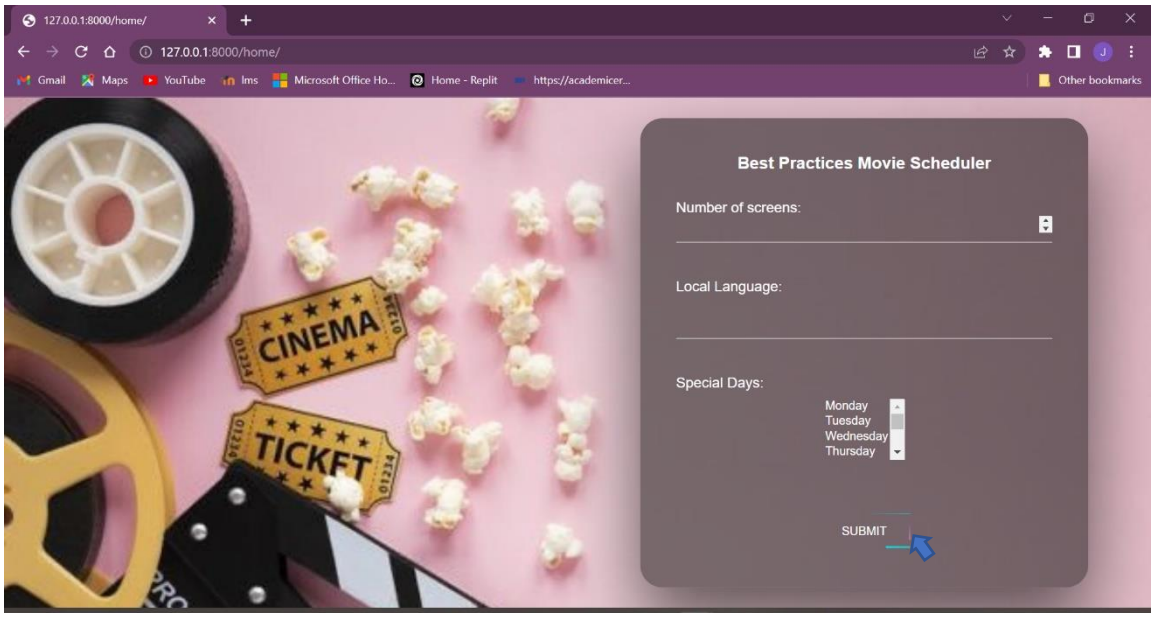
Fig 5.1. About page



On clicking the ‘LET’S BEGIN SCHEDULING’ button, one will be redirected to the first form/ survey page.

5.2. First form/Survey Page

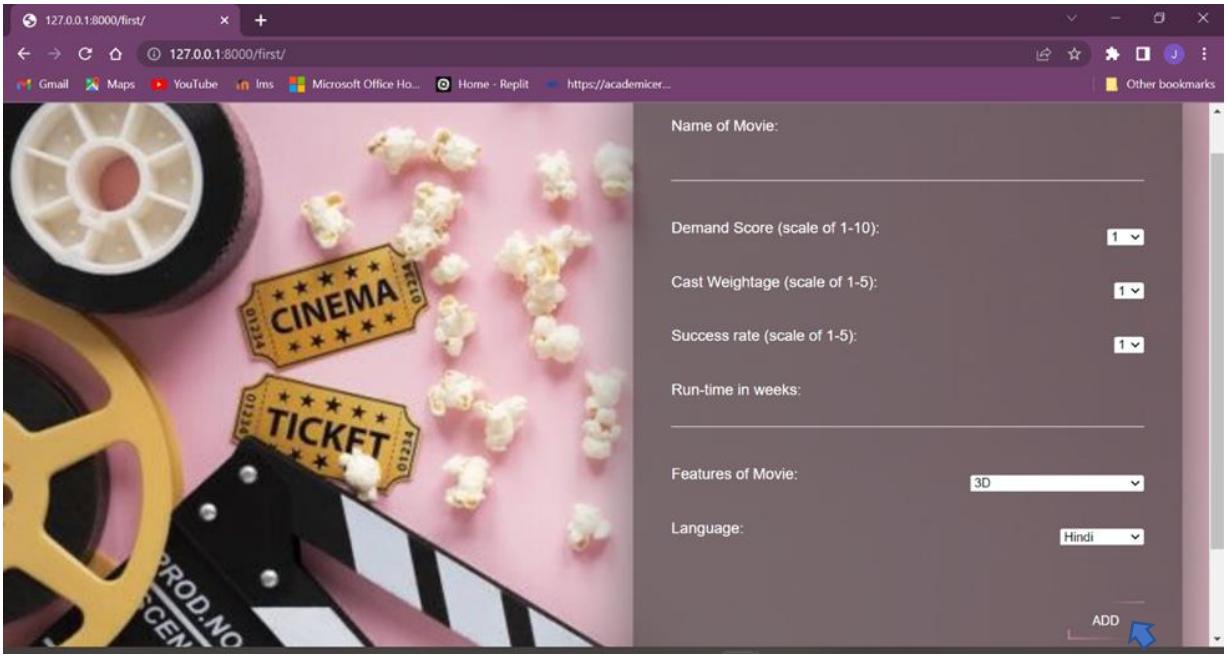
Fig 5.2. First form/survey page



After filling the necessary details, the user is redirected to the second form/ survey page on clicking the ‘SUBMIT’ button. The input is received by the server end in the form of a list of dictionaries.

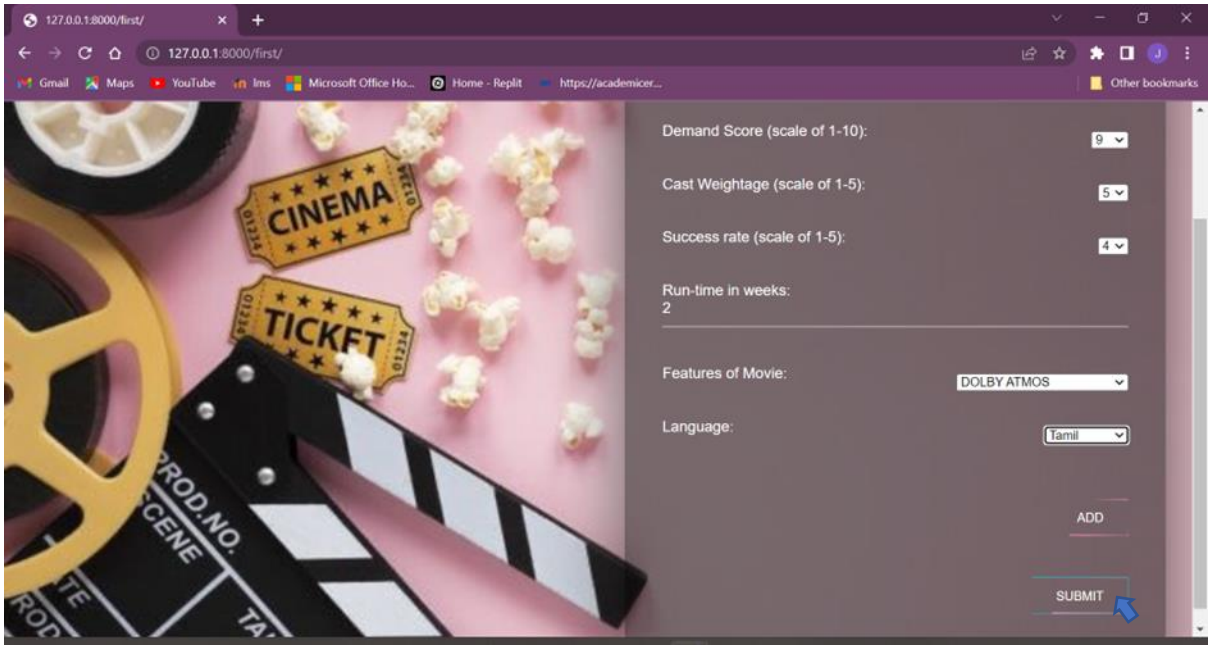
5.3. Second form/ Survey Page

Fig 5.3.a. Second form/ survey page



After filling the necessary details, the user is redirected to the same page on clicking the ‘ADD’ button to go on adding details about the next movie.

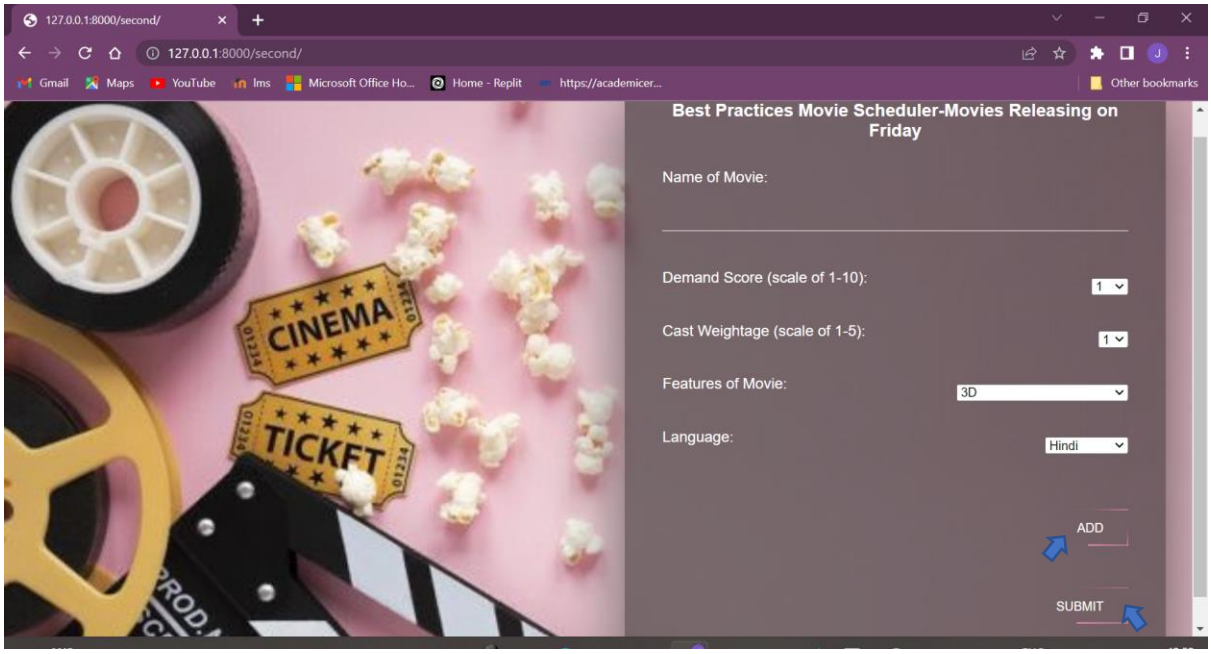
Fig 5.3.b. Second form/survey page



The page redirects to the third form by clicking the ‘SUBMIT’ button, after filling the details for the last movie. The input is received by the server end in the form of a list of dictionaries.

5.4.Third form/ Survey Page

Fig 5.4. Third form/survey page



After filling the necessary details, the user is redirected to the same page on clicking the ‘ADD’ button to go on adding details about the next movie. The page redirects to the result page by clicking the ‘SUBMIT’ button, after filling the details for the last movie. The input is received by the server end in the form of a list of dictionaries.

5.5. Result Page

The input received from all the three forms is converted to a dictionary of dictionaries, which is the structure supported by our backend. The priority for each movie is computed in *class Movie* using the formula mentioned before and also, after considering the various soft constraints i.e., Features, Language and Runtime of the movie.

The number of showings per movie is computed in the *module Generate Slots* considering the number of screens in the theatre and number of slots per day (Weekdays- 4 slots, Weekends-5 slots and Special Days- all 6 slots). Lastly, the schedule is generated in *Scheduling module*.

The generated schedule is displayed to the user in the result page as a sequence of HTML tables.

Fig 5.5.a. Result Page Sample-1

127.0.0.1:8000/result/

Gmail Maps YouTube lms Microsoft Office Ho... Home - Replit https://academicer... Other bookmarks

Best Practices Movie Scheduler

Your schedule has been generated!!

Monday

Screens	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6
Screen1	-	Harry Potter	Harry Potter	Vikram	Thiruchitrambalam	-
Screen2	-	Dune	Lord of the Rings	Thiruchitrambalam	Thiruchitrambalam	-
Screen3	-	Lord of the Rings	Gangubhai Kathiawadi	Vikram	Thiruchitrambalam	-
Screen4	-	Dune	Dune	Thiruchitrambalam	Thiruchitrambalam	-
Screen5	-	Dune	Dune	Vikram	Vikram	-
Screen6	-	Gangubhai Kathiawadi	Gangubhai Kathiawadi	Vikram	Thiruchitrambalam	-
Screen7	-	Dune	Shamshera	Vikram	Thiruchitrambalam	-
Screen8	-	Harry Potter	Lord of the Rings	Vikram	Vikram	-
Screen9	-	Lord of the Rings	Lord of the Rings	Thiruchitrambalam	Vikram	-
Screen10	-	Gangubhai Kathiawadi	Dune	Thiruchitrambalam	Vikram	-
Screen11	-	Harry Potter	Shamshera	Thiruchitrambalam	Thiruchitrambalam	-
Screen12	-	Gangubhai Kathiawadi	Dune	Vikram	Vikram	-
Screen13	-	Gangubhai Kathiawadi	Dune	Vikram	Thiruchitrambalam	-
Screen14	-	Lord of the Rings	Harry Potter	Dune	Dune	-
Screen15	-	Gangubhai Kathiawadi	Lord of the Rings	Vikram	Vikram	-
Screen16	-	Lord of the Rings	Lord of the Rings	Thiruchitrambalam	Vikram	-

Movie schedule for every day of the week is generated. In the above picture, the movie schedule for Monday is seen. We can see that only 4 slots (Slots 2 through 6) have been filled (for Monday is a weekday) with movies that are already running in the theatre.

Fig 5.5.b. Result Page Sample-2

127.0.0.1:8000/result/

Gmail Maps YouTube lms Microsoft Office Ho... Home - Replit https://academicer... Other bookmarks

Screen16	-	Dune	Harry Potter	Dune	Thiruchitrambalam	-

Friday

Screens	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6
Screen1	Vikram	Brahmastra	Lord of the Rings	Lord of the Rings	Legend	Thiruchitrambalam
Screen2	Thiruchitrambalam	Dune	Gangubhai Kathiawadi	Lord of the Rings	Legend	Thiruchitrambalam
Screen3	Vikram	Dune	Harry Potter	Dune	Vikram	Legend
Screen4	Thiruchitrambalam	Brahmastra	Rocketry	Gangubhai Kathiawadi	Legend	Legend
Screen5	Legend	Rocketry	Harry Potter	-	Vikram	Legend
Screen6	Legend	Dune	Gangubhai Kathiawadi	Brahmastra	Thiruchitrambalam	Vikram
Screen7	Thiruchitrambalam	Lord of the Rings	Shamshera	Brahmastra	Vikram	Vikram
Screen8	Legend	Dune	Rocketry	Shamshera	Thiruchitrambalam	Thiruchitrambalam
Screen9	Vikram	Dune	Rocketry	Dune	Legend	Vikram
Screen10	Thiruchitrambalam	Lord of the Rings	Lord of the Rings	Brahmastra	Legend	Vikram
Screen11	Vikram	Brahmastra	Rocketry	Dune	Thiruchitrambalam	Vikram
Screen12	Vikram	Dune	Brahmastra	Lord of the Rings	Dune	Vikram
Screen13	Thiruchitrambalam	Rocketry	Brahmastra	Gangubhai Kathiawadi	Legend	Legend
Screen14	Vikram	Rocketry	Harry Potter	Dune	Vikram	Legend
Screen15	Legend	Dune	Brahmastra	Gangubhai Kathiawadi	Thiruchitrambalam	Thiruchitrambalam
Screen16	Legend	Rocketry	Rocketry	Rocketry	Thiruchitrambalam	Vikram

Saturday

Screens	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6

In the above picture, the movie schedule for Friday can be seen. We can see that all 6 slots have been filled (even though Friday is a weekday, it is a special day as per user input) with movies that are already running as well as new movies that are released that week.

6. Conclusion:

Having taken into consideration, audience demand and success (performance) of the movies in the previous weeks, in the problem of scheduling movies, we have generated the number of showings per movie and have plotted the movies with a high priority score to prime showtimes. This ensures optimization of the schedule and maximizes the revenue of the theatre/multiplex. The Best Practices Movie Scheduler fulfills its problem statement and can be considered as an efficient alternative to the current, manual timetabling procedure.

7. References:

1. Sanjeev Swami, Jehoshua Eliashberg and Charles B. Weinberg. Article: SilverScreener: A Modeling Approach to Movie Screens Management. *Marketing Science 1999 INFORMS Vol. 18, No. 3, 1999, pp. 352–372*
2. Corey Schafer. Python Django Tutorial: Full-Featured Web App Part 1 - Getting Started. *www.youtube.com.*
3. Tech with Tim. Django For Beginners - Full Tutorial. *www.youtube.com*
4. Slider Revolution, March 21, 2021. https://www.sliderrevolution.com/resources/css-forms.
5. Nicepage CSS-HTML Templates application.