

# **MOVIEGPT**

**A PROJECT REPORT  
for  
Mini Project (KCA 353)**

**Session (2024-25)**

**Submitted by**

**Archi Goel  
(2300290140036)**

**Submitted in partial fulfilment of the  
Requirements for the Degree of**

## **MASTER OF COMPUTER APPLICATION**

**Under the Supervision of  
Dr. Amit Kumar Gupta  
Professor**



**Submitted to**

**DEPARTMENT OF COMPUTER APPLICATIONS  
KIET Group of Institutions, Ghaziabad  
Uttar Pradesh-201206  
(December 2024)**

## **DECLARATION**

We hereby declare that the work presented in this report entitled “**MOVIEGPT**”, was carried out by us. We have not submitted the matter embodied in this report for the award of any other degree or diploma from any other University or Institute.

We have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, and results, that are not my original contribution. We have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

We affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, we shall be fully responsible and answerable.

**Name :** Archi Goel  
**Roll No. :** 2300290140036  
**Branch:** MCA

## **CERTIFICATE**

It is to certify that Archi Goel (2300290140036) has carried out the project work having “**Moviegpt (Mini-Project-KCA353)** for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Dr. Amit kumar Gupta**  
**Professor**  
**Department of Computer Applications**  
**KIET Group of Institutions, Ghaziabad**

**Dr. Arun Kumar Tripathi**  
**Head**  
**Department of Computer Applications**  
**KIET Group of Institutions, Ghaziabad**

## ACKNOWLEDGEMENTS

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor **Dr. Amit kumar Gupta** , for his guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi**, Professor and Head, the Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**Archi Goel**  
**(2300290140036)**

## ABSTRACT

**MovieGPT** is an innovative AI-driven system designed to enhance the movie-watching and content creation experience by integrating personalized movie recommendations with generative storytelling. The system leverages advanced **machine learning** algorithms and **natural language processing (NLP)** models, particularly **GPT-4**, to offer personalized film suggestions based on user preferences, watching history, and mood. Additionally, it uses generative AI to create unique and original movie plot ideas, providing a powerful tool for filmmakers, writers, and content creators seeking inspiration or assistance in developing new narratives.

The project combines **collaborative filtering** and **content-based recommendation techniques** to suggest films tailored to users' tastes, offering a dynamic and continually evolving experience. Furthermore, MovieGPT utilizes the capabilities of GPT models to generate entire movie scripts or story outlines based on user input, such as genre, character traits, or themes, enhancing the creative process in filmmaking and scriptwriting. The system also adapts and improves over time through user feedback, ensuring that recommendations and plot suggestions become more accurate and relevant.

Key results of the project include the successful integration of movie recommendation and plot generation systems in a user-friendly interface, along with demonstrated accuracy and creativity in the generated content. The system was tested with various user groups, and the feedback indicated a high level of engagement and satisfaction with both the recommendations and creative assistance provided by the platform.

## TABLE OF CONTENTS

1. Declaration	ii
2. Certificate	iii
3. Acknowledgements	iv
4. Abstract	v
5. Table of Contents	vi
6. List of Abbrevations	vii
7. List of Figures	viii-ix
8. Introduction	10
9. Literature review	13
10. System Analysis and System Design	15
11. Implementation of MovieGPT	20
12. Technology used and setup	24
13. Testing	31
14. Result	40
15. Conclusion	45
16. Future Scope	48
17. Bibliography	50

## LIST OF ABBREVIATIONS

- React – A JavaScript library for building user interfaces.
- JS – JavaScript, the programming language used in React.
- API – Application Programming Interface, used for communication between frontend and backend (likely for Firebase interactions).
- CRUD – Create, Read, Update, Delete, common operations performed on data in databases.
- UI – User Interface, the part of the application that interacts with the user.
- UX – User Experience, the overall experience a user has with the application.
- JSON – JavaScript Object Notation, a lightweight data-interchange format.
- JWT – JSON Web Token, often used for authentication in web applications.
- DB – Database, where the data is stored (Firebase in your case).
- CSS – Cascading Style Sheets, used for styling the frontend.
- HTML – HyperText Markup Language, used for structuring web pages.
- CRUD API – API that facilitates Create, Read, Update, and Delete operations on data.
- SDK – Software Development Kit, a set of tools for developing software (potentially relevant to Firebase).
- OAuth – Open Authorization, a common protocol used for authorization.
- REST – Representational State Transfer, an architectural style for designing networked applications (likely used in the API calls).
- NoSQL – A type of database (relevant for Firebase, which is a NoSQL database).

## **LIST OF FIGURES**

### **1. System Architecture Diagram**

This diagram would showcase how the React frontend communicates with the Firebase backend.

- It can highlight the flow of data between the user interface, the server, and the database.
- Possible components: React (UI), Firebase (Authentication, Firestore Database, Storage), APIs, and User.

### **2. Database Schema**

- This figure would show the structure of the database in Firebase, detailing how data is stored.
- Tables/Collections: Users, Movies, Comments, Likes, etc.
- Relationships between entities (e.g., Users to Comments or Movies to Likes).

### **3. User Flow Diagram**

- This diagram would illustrate the steps a user takes to interact with the application.
- Examples: Registration, logging in, posting a comment, liking a movie, etc.
- Can also include decision points (e.g., if a user is logged in or not).

### **4. Wireframe or UI Design**

- Visual representation of the user interface, showing how the app's screens will look.
- Can include wireframes for pages such as the Home page, Movie page, Login page, Comment section, etc.



## **5. API Flow Diagram**

- Illustrates the flow of API requests and responses between the frontend and backend.
- Example: A user sends a request to Firebase to fetch movie data, and Firebase returns the movie details to the React app.

## **6. Authentication Flow**

- A diagram showing the flow of user authentication using Firebase Authentication.
- It might cover scenarios like user sign-up, login, password recovery, etc.

## **7. Error Handling Flow**

- This diagram could show how errors are managed in the application.
- For example: what happens if an API request fails, or a user attempts to perform an action without being authenticated.

## **8. Data Flow Diagram (DFD)**

- This figure represents how data moves within the system from the user's input to the final storage or output.
- Example: A user submits a movie review, and the data flows from the UI to the API, then to the Firebase database.

## **9. Tech Stack Overview**

- A visual representation of the technologies used in MovieGPT, such as React, Firebase, JavaScript, etc.
- This could be a simple graphic with logos or icons for each technology.

## **10. Component Hierarchy/Structure**

- A diagram showing the structure of the React components in the project.
- Example: Parent components (e.g., App) and child components (e.g., MovieCard, CommentForm) and their relationship.

