

Digital Story Generator Using GenAI

Abstract:

The **Digital Story Generator Using GenAI** is a web-based application designed to create and manage digital stories using Generative Artificial Intelligence. In the modern digital world, storytelling has evolved from traditional writing methods to intelligent automated systems. This project provides a platform where users can generate creative stories instantly by giving simple text prompts.

The system consists of two main modules: **Admin and Customer**. The **Administrator** manages the entire system by handling customer details, managing story categories, adding and viewing story books, publishing notifications, and monitoring all generated and user-submitted stories. The admin ensures proper system control and content management.

The **Customers** can register and log in to the system to explore and create stories. Users can generate stories by providing prompts to the GenAI model, which produces creative and meaningful content automatically. Customers can also add their own stories, view all stories, explore today's and top stories, comment on stories, and access story books. Additionally, users can manage their profiles, view notifications, and interact with other users through comments.

The GenAI-based story generation feature is the core of the system, enabling users to quickly generate unique and engaging stories without manual writing effort. This system enhances creativity, improves user engagement, and provides a modern digital storytelling platform.

Overall, the **Digital Story Generator Using GenAI** simplifies content creation, promotes creativity, and demonstrates the effective use of artificial intelligence in storytelling applications.

Introduction

In today's digital era, content creation has become an essential part of communication, entertainment, and education. Storytelling, which was traditionally done manually, now benefits from advancements in Artificial Intelligence. Generative AI (GenAI) has made it possible to automatically create meaningful and creative content based on user inputs, reducing the time and effort required for writing.

The **Digital Story Generator Using GenAI** is designed to provide a web-based platform where users can generate stories instantly by giving simple prompts. This system leverages artificial intelligence to produce creative and engaging stories, making storytelling accessible to everyone, including beginners and non-writers.

The system consists of two main modules: **Admin and Customer**. The **Admin** manages the platform by handling customer details, managing story categories, adding and viewing story books, publishing notifications, and monitoring all stories in the system. The **Customer** can register and log in to the system, generate stories using prompts, add their own stories, view all stories, explore top and recent stories, and interact with other users through comments.

Additionally, the system provides features such as story categorization, notifications, and profile management, enhancing the overall user experience. The Digital Story Generator Using GenAI aims to simplify content creation, encourage creativity, and demonstrate the practical use of artificial intelligence in modern digital applications

Existing System

In the existing system, story creation is primarily done manually by writers or users without the support of intelligent automation. Users rely on their own creativity, writing skills, and time to develop stories, which can be challenging for beginners or those lacking writing experience. There is no automated system that can generate stories instantly based on user input.

Most available platforms allow users to read or publish stories, but they do not provide advanced features like AI-based story generation. Users must spend significant time thinking, drafting, and editing content. Additionally, there is limited support for organizing stories into categories, tracking trending or top stories, and interacting through comments in a structured manner.

There is also no centralized system where administrators can manage story content, categories, notifications, and users efficiently. Communication between users and administrators is often unstructured, and there is no proper mechanism to manage story books or provide real-time updates

Drawbacks

The traditional story creation and management systems have several limitations that reduce efficiency and user engagement. The major drawbacks are as follows:

1. **Manual Story Creation** – Users must write stories on their own, which requires time, effort, and creativity.
2. **No AI Support** – There is no automated system to generate stories based on user prompts.
3. **Time-Consuming Process** – Creating, editing, and finalizing stories manually takes a long time.
4. **Limited Creativity for Beginners** – New users may struggle to generate ideas and write engaging stories.

5. **No Centralized Content Management** – Stories, categories, and user data are not managed in a structured system.
6. **Lack of Personalization** – Systems do not provide customized story generation based on user input.
7. **No Real-Time Story Generation** – Users cannot instantly generate content based on prompts.
8. **Limited User Interaction** – Commenting and feedback features are either unavailable or not well-structured.
9. **No Trending or Top Story Features** – Users cannot easily discover popular or recent stories.
10. **Poor Notification System** – Users are not informed about updates, new stories, or announcements.
11. **Inefficient Content Organization** – Stories are not properly categorized or structured.
12. **Limited Accessibility** – Some platforms do not provide easy access to manage or explore stories efficiently

Proposed System

The proposed **Digital Story Generator Using GenAI** is a web-based application designed to automate and enhance the process of story creation using Generative Artificial Intelligence. This system provides a centralized platform where users can generate creative and meaningful stories instantly by providing simple text prompts.

In this system, the **Admin** manages the entire platform by handling customer details, managing story categories, adding and viewing story books, and publishing notifications. The admin can also monitor all user-generated and AI-generated stories to ensure proper content management and system functionality. Additionally, the admin can manage user accounts and maintain system security.

The **Customers** can register and log in to the system to access various features. They can generate stories by providing prompts to the GenAI model, which automatically creates unique and engaging content. Customers can also add their own stories, view all stories, explore today's stories and top stories, and interact with other users by adding comments. The system allows users to view story books, receive notifications, and manage their profiles securely.

The core feature of this system is the **GenAI-based Story Generator**, which reduces the effort required for content creation and enhances creativity by providing instant story outputs. The system also includes features such as story categorization, comment management, notifications, and profile management, making it a complete digital storytelling platform.

Overall, the proposed system provides a modern, efficient, and user-friendly solution for storytelling by combining artificial intelligence with web technologies, improving user engagement and content accessibility

Advantages

The proposed Digital Story Generator Using GenAI offers several benefits that improve content creation, user experience, and system efficiency. The key advantages are:

1. **Automated Story Generation** – Users can generate stories instantly using AI by providing simple prompts.
2. **Saves Time and Effort** – Reduces the time required for writing and editing stories manually.
3. **Enhances Creativity** – Helps users generate new ideas and creative content easily.
4. **User-Friendly Platform** – Simple and easy-to-use interface for all types of users.
5. **Centralized Content Management** – Admin can manage stories, categories, users, and notifications efficiently.
6. **Story Categorization** – Stories are organized into categories for easy browsing and access.
7. **Interactive Features** – Users can add comments and interact with other users' stories.
8. **Trending and Latest Stories** – Users can view today's stories and top stories easily.
9. **Notification System** – Admin can send updates, announcements, and alerts to users.
10. **Profile Management** – Users can manage personal details and update account information.
11. **Secure Login System** – Provides authentication and password management for security.
12. **Scalable System** – Can be extended with advanced AI models and additional features.
13. **Accessible Anywhere** – Users can access the system from any location using the internet.
14. **Digital Story Book Access** – Users can explore and read story books online.
15. **Improved User Engagement** – Encourages users to create, read, and interact with content regularly

Modules

Admin Modules

The Admin Module is responsible for managing the entire system. The administrator can securely log in and perform various operations such as viewing customer details, managing story categories, adding and viewing story books, and publishing notifications. The admin can also view all stories generated by users and AI,

monitor system activities, and ensure proper content management. Additionally, the admin can change the password and log out securely

Customers Modules

The Customer Module allows users to register and log in to the system. Customers can generate stories using AI by providing prompts, add their own stories, and view all stories available in the system. They can explore features like today's stories and top stories, interact with other users by adding comments, and view story books. Customers can also receive notifications, manage their profiles, change passwords, and log out securely

Scope and Objective of the Project

The main objective of the **Digital Story Generator Using GenAI** is to develop a web-based platform that enables users to generate creative and meaningful stories automatically using Generative Artificial Intelligence. The system aims to simplify the storytelling process by allowing users to provide simple prompts and receive complete stories instantly.

Another objective is to provide a centralized platform where users can create, view, and manage stories efficiently. The system also focuses on improving user engagement by enabling features such as commenting, viewing top and recent stories, and accessing story books. Additionally, the admin can manage users, categories, notifications, and story content to ensure smooth system operation and content organization.

The project also aims to encourage creativity, reduce manual effort in content creation, and demonstrate the practical implementation of AI in real-world applications

The scope of the **Digital Story Generator Using GenAI** includes user registration and login functionality, AI-based story generation using prompts, and management of stories and categories. Customers can create and explore stories, interact through comments, and view trending and latest content. The system also includes features such as notification management, profile updates, and digital story book access.

The administrator manages the entire system by handling customer data, adding categories, managing stories, and sending notifications. The system ensures secure access and efficient data management.

In the future, the system can be enhanced with advanced AI models, multi-language story generation, voice-based input, image-based storytelling, mobile application support, and integration with social media platforms. These enhancements can make the system more interactive, intelligent, and user-friendly

Software Requirements

- Operating System : Windows
- User Interface : HTML, CSS , JS and Bootstrap
- Programming Language : Python
- Web Applications : Django
- IDE/Workbench : Pycharm
- Database : MYSQL, SQL, SQLYOG
- Server Deployment : Apache Tomcat

Hardware Requirements

- Processor : Intel I3 Processor
- RAM : 4 GB
- Monitor : 15 inch color monitor or LED
- Hard disk : 256 GB
- Key board : Standard 102 keys
- Mouse : Optical

Functional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So it's essential to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions.

HOME

ABOUT

CONTACT

ADMIN

➤ Admin Functions

- Login
- Home
- View Customers
- View Contacts
- All Stories
- Add Category
- View Categories
- Add Notifications
- View Notifications
- Add Story Book
- View Story Book
- Change Password
- Logout

CUSTOMER

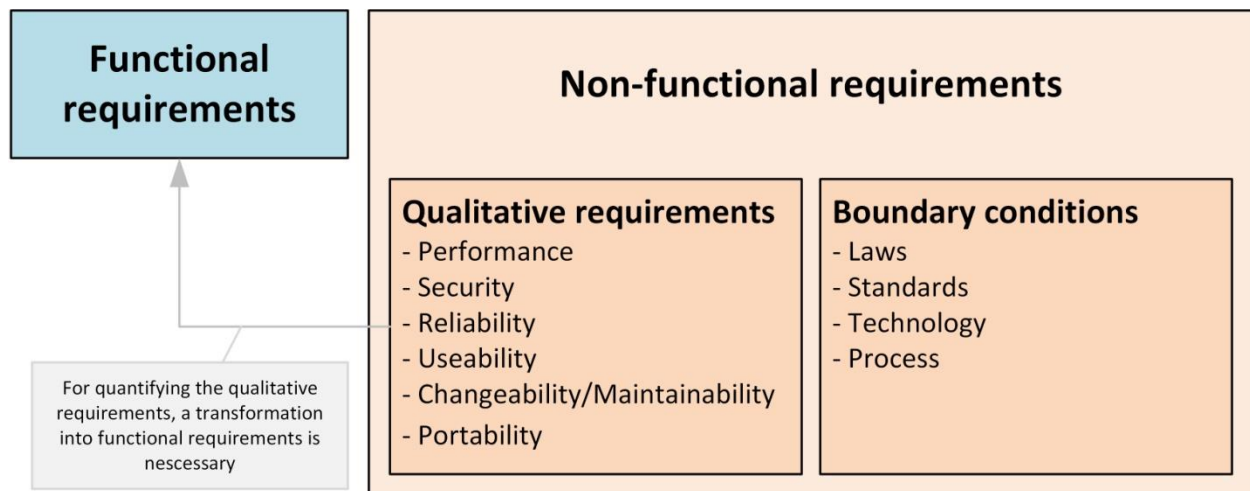
➤ Customer Functions

- Registration
- Login
- Home
- Add Story
- My Story
- All Stories
- Generates Storyies
- Add Comments
- View Comments
- Today Stories
- Top Stories
- View Story Book
- View Notification

- My Profile
- Change Password
- Logout

Non Functional Requirements

Non-functional testing is a type of software testing that verifies non-functional aspects of the product, such as performance, stability, and usability. Whereas functional testing verifies whether or not the product does what it is supposed to, non-functional testing verifies how well the product performs.



The system should shows below non-functional requirements

1. High Security
2. Multi browser and operating system compatibility
3. Mobile responsiveness
4. High quality and performance
5. Memory optimized and User Friendliness
6. Easy maintenance and scalable capability
7. It's should support good accessibility and usability

8. It should follow regulatory and compliance policies

Feasibility Study

The possibility of the undertaking is broke down during this stage and strategic plan is advanced with an exceptionally broad arrangement for the task and a few quotes. During project examination the likelihood investigation of the proposed project is to be completed. This is often to ensure that the proposed project isn't a weight to the organization. For practicality examination, some comprehension of the many necessities for the project is prime.

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are.

Economic Feasibility

This examination is completed to see the financial effect that the framework will wear the association. The measure of store that the organization can fill the innovative work of the framework is constrained. The consumptions must be advocated. Accordingly, the created framework also inside the budget and this was accomplished in light of the very fact that an outsized portion of the advancements utilized are uninhibitedly accessible. Just the modified items must be bought. Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. And also in this project work all software is free and open source. It can be developed using our personal machines. So its economically its feasible to complete the project.

Technical Feasibility

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The system is developed for platform Independent environment. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

This investigation is completed to see the technical practicality, that is, the specialized prerequisites of the framework. Any framework created must not have a popularity on the accessible specialized assets. This may prompt high requests on the accessible specialized assets. This may prompt high requests being assail the customer. The created framework must have a humble prerequisite, as just insignificant or invalid changes are required for actualizing this framework.

Social Feasibility

The part of study is to see the degree of acknowledgment of the framework by the client. This incorporates the way toward preparing the client to utilize the framework effectively. The client must not feel undermined by the framework, rather should acknowledge it as a requirement. The degree of acknowledgment by the clients exclusively relies upon the techniques that are utilized to show the client about the framework and to form him familiar with it. His degree of certainty must be raised with the goal that he's likewise able to make some helpful analysis, which is invited, as he's the last client of the framework.

Operational Feasibility

This region of attainability ensures that the undertaking fulfills all the prerequisites that were talks about during introductory necessity examination stage. This undertaking met all the necessities proposed before. It ensured that the modules examined during the proposing of the framework were actualized with stable precision and with no mistakes.

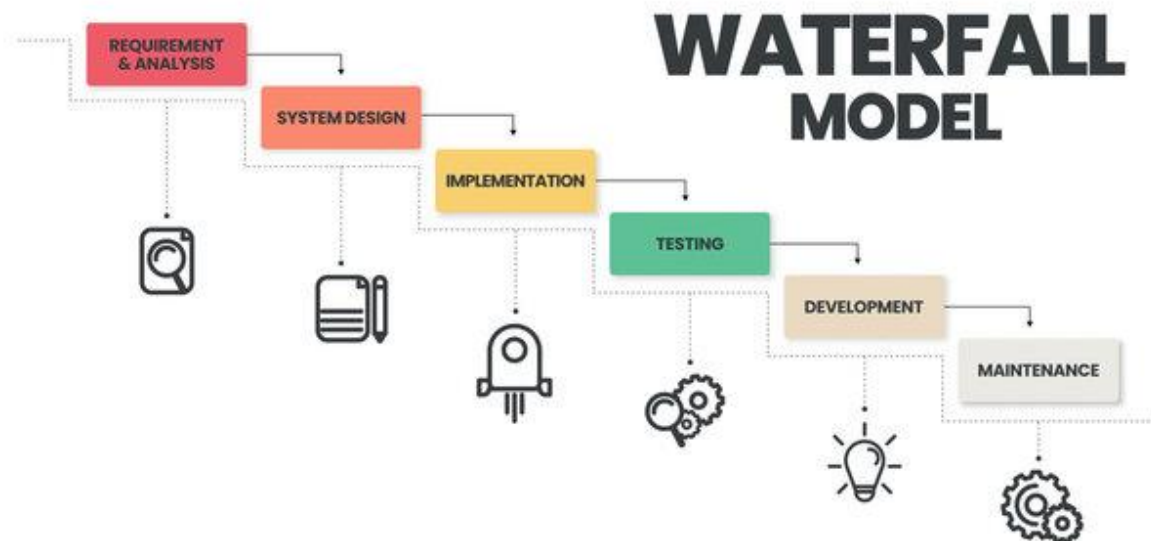
Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Process Models

1. Waterfall Model

Waterfall model indicates that all the SDLC steps will be going on step by step. In this model once one stage completed then only it will start the process. This will be implemented when the requirements well known and clearly defined.

In this model we can implement all the steps one by one. First we will do requirements gathering i.e in this step business analyst will collect the all functional requirements from the client and will discuss with the manager and development team. Second step is analysis where people can analyses the requirements so that they will do risk basement, risk mitigation, risk controlling steps. And also they will do the cost-time analysis to estimate the cost. They will also do the detail feasible study over the requirements of project.



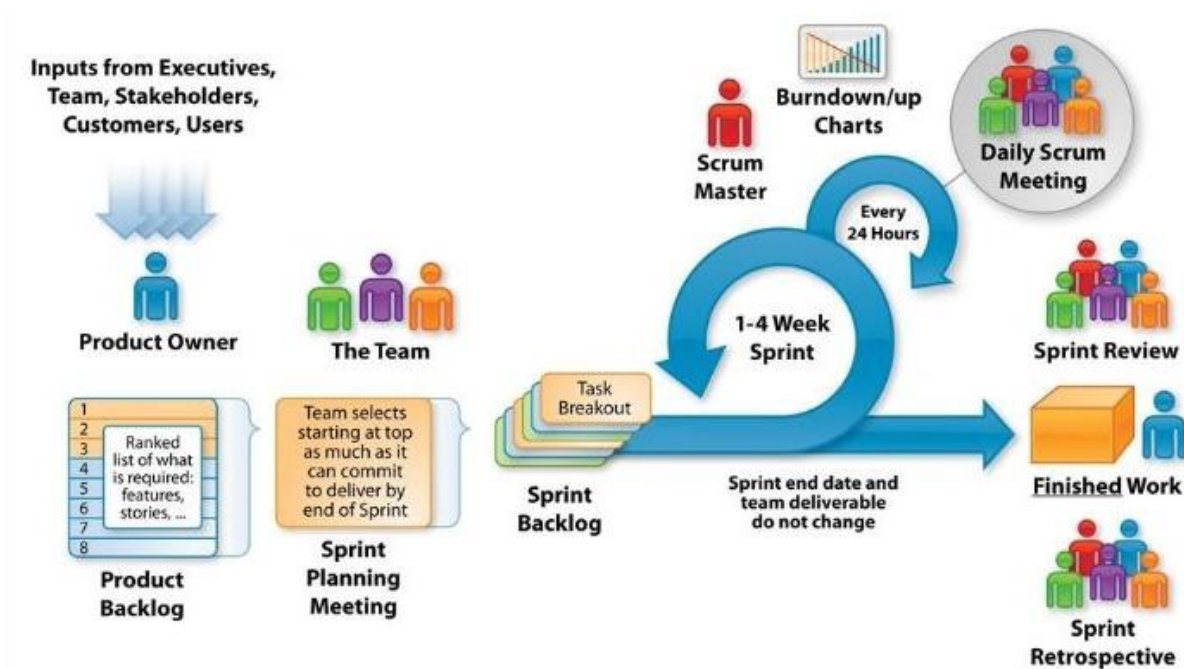
In third phase called design page generally people will design the project i.e architect will prepare the UML diagrams and prototypes for project. And then developers will start programming or coding based on design . Finally testing team will test the content to identify and fix the bugs. After successful user acceptance testing application will be deployed and maintained in the client machine.

The waterfall model was selected as the SDLC model due to the following reasons:

- When the requirements are very clear and not changed.
- Available technology is enough to implement.
- Project is easy and simple to understand to finish.
- And if we don't have any confusion in requirements.
- We can easily manage the review process and task management.
- Every stage defined clearly and with timelines

2. Agile Scrum Model

Agile scrum methodology is the combination of the agile philosophy and the scrum framework. Agile means “incremental, allowing teams to develop projects in small increments. Scrum is one of the many types of agile methodology, known for breaking projects down into sizable chunks called “sprints.” Agile scrum methodology is good for businesses that need to finish specific projects quickly.



Agile scrum methodology is a project management system that relies on incremental development. Each iteration consists of two- to four-week sprints, where the goal of each sprint is to build the most important features first and come out with a potentially deliverable product. More features are built into the product in subsequent sprints and

are adjusted based on stakeholder and customer feedback between sprints. Whereas other project management methods emphasize building an entire product in one operation from start to finish, agile scrum methodology focuses on delivering several iterations of a product to provide stakeholders with the highest business value in the least amount of time.

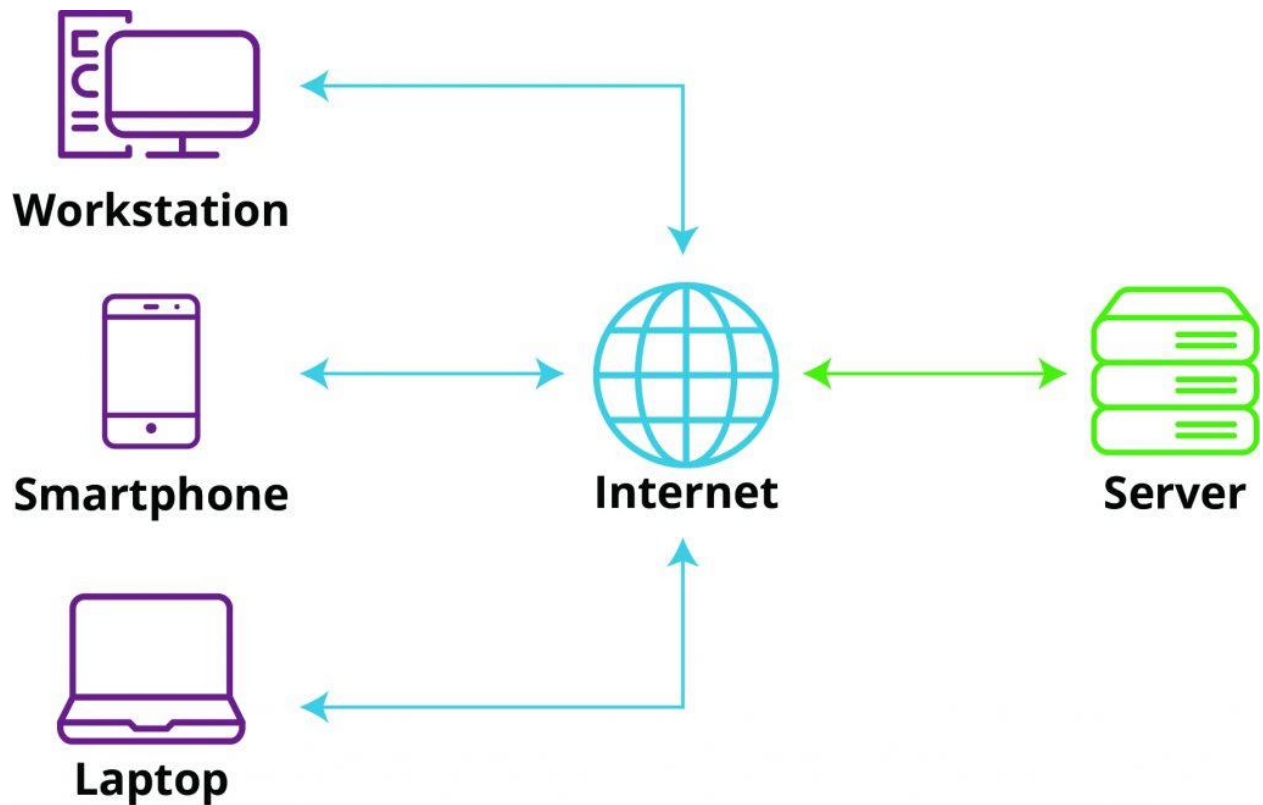
Agile scrum methodology has several benefits. First, it encourages products to be built faster, since each set of goals must be completed within each sprint's time frame. It also requires frequent planning and goal setting, which helps the scrum team focus on the current sprint's objectives and increase productivity.

The greatest benefit of agile scrum methodology is its flexibility. With the sprint-based model, the scrum team typically receives feedback from stakeholders after each sprint. If there are any problems or changes, the scrum team can easily and quickly adjust product goals during future sprints to provide more valuable iterations. This way, stakeholders are happier because they get exactly what they want after being involved every step of the way.

Client Server Architecture

Client server architecture, architecture of a computer network in which many clients (remote processors) request and receive service from a centralized server (host computer). Client computers provide an interface to allow a computer user to request services of the server and to display the results the server returns.

The Client-server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters called clients. In the client-server architecture, when the client computer sends a request for data to the server through the internet, the server accepts the requested process and delivers the data packets requested back to the client. Clients do not share any of their resources. Examples of the Client-Server Model are Email, World Wide Web, etc.



Web Application MVC Architecture

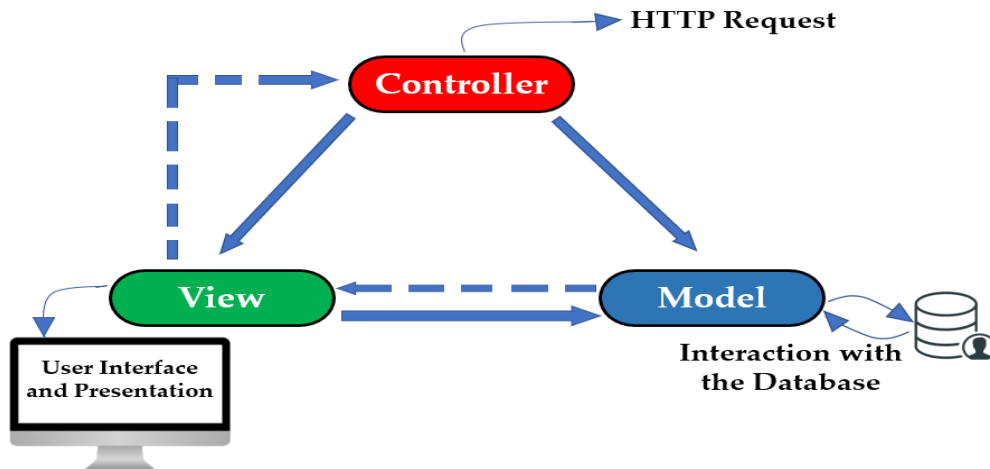
The Model-View-Controller (MVC) is a well-known design pattern in the web development field. It is a way to organize our code. It specifies that a program or application shall consist of data model, presentation information and control information. The MVC pattern needs all these components to be separated as different objects.

The MVC pattern architecture consists of three layers:

- **Model:** It represents the business layer of application. It is an object to carry the data that can also contain the logic to update controller if data is changed.
- **View:** It represents the presentation layer of application. It is used to visualize the data that the model contains.

- **Controller:** It works on both the model and view. It is used to manage the flow of application, i.e. data flow in the model object and to update the view whenever data is changed.

Model-View-Controller Architecture Block Diagram



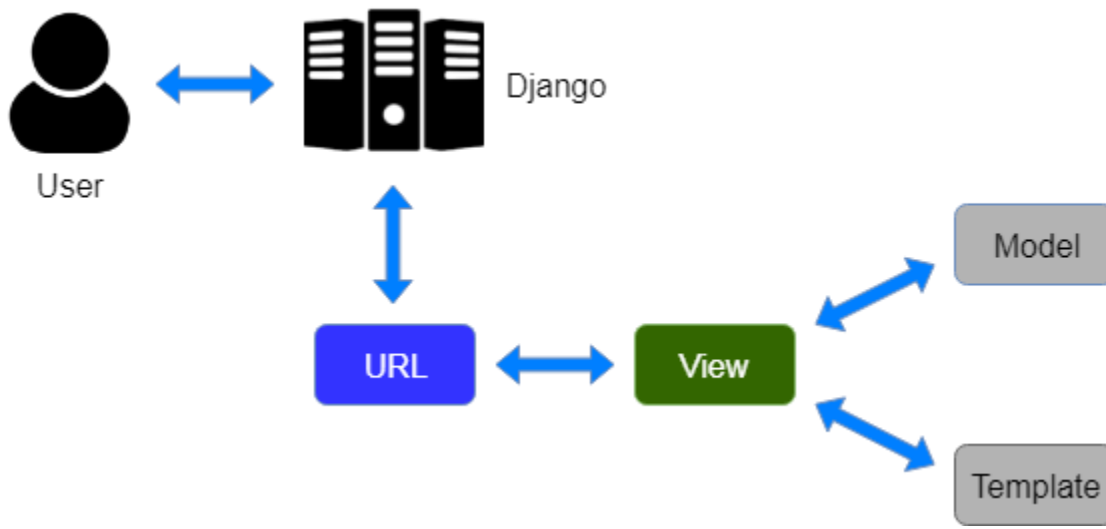
Web Application MVT Architecture

The MVT (Model View Template) is a software design pattern. It is a collection of three important components Model View and Template. The Model helps to handle database. It is a data access layer which handles the data.

The Template is a presentation layer which handles User Interface part completely. The View is used to execute the business logic and interact with a model to carry data and renders a template.

The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySQL, Postgres).

The View is the user interface — what you see in your browser when you render a website. It is represented by HTML/CSS/Javascript and Jinja files.



Conclusion

The **Digital Story Generator Using GenAI** provides an innovative and efficient solution for creating digital stories using artificial intelligence. By allowing users to generate stories instantly through simple prompts, the system reduces the effort required for manual writing and enhances creativity for users of all skill levels.

The system enables administrators to manage users, story categories, story books, and notifications effectively, ensuring smooth operation and organized content management. Customers can create, explore, and interact with stories through features such as comments, top stories, and today's stories, making the platform engaging and user-friendly.

This project demonstrates the practical application of Generative AI in content creation and highlights its ability to transform traditional storytelling into a modern, automated process. In the future, the system can be enhanced with advanced AI capabilities, multi-language support, voice-based inputs, and mobile applications to provide a more interactive and intelligent storytelling experience.

Overall, the Digital Story Generator Using GenAI serves as a scalable and creative platform that promotes digital content creation and showcases the potential of artificial intelligence in real-world applications.