**DIGITAL CAMPUS NOTICEBOARD**

*By*

**MANKALA LIKHITH KUMAR**

**(Regd. No. 23VV1A1232)**

Under the supervision of

**MRS.MADHUMITA CHANDA**

**Department of Information Technology**

**JNTU-GV CE, Vizianagaram**



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA-VIZIANAGARAM**

**VIZIANAGARAM – 533003 ANDHRA PRADESH, INDIA MARCH 2025.**

****

Regd.No:23VV1A1232

**CERTIFICATE**

**Certified that this is a bonafide Record of Practical work done by Mr Mankala Likhith Kumar Of 2nd B.Tech 2nd Semester Class in Django Framework laboratories College during the year 2024-25**

**No of experiments done and certified:13**

**Head Of The Department Lecture in Charge**

DEPARTMENT OF INFORMATION TECHNOLOGY

JNTU-GURAJADA VIZIANAGARAM

COLLEGE OF ENGINEERING VIZIANAGARAM (A)

VIZIANAGARAM Website:www.jntugvcev.edu.in

# Subject Name: DJANGO FRAMEWORK Subject Code:R232212SE01

# Year: 2025 Regulation: R23

# COURSE OUTCOMES

|  |  |  |
| --- | --- | --- |
| **NBA**  **Subject Code** | **Course Outcomes** | |
|  | **CO1** | Design and build static as well as dynamic web pages and interactive web-based applications . |
| **CO2** | Web development using Django framework. |
| **CO3** | Analyze and create functional website in Django and deploy Django Web Application on Cloud . |

**CO-PO Mapping**

Mapping of Course Outcomes (COs) with Program Outcomes (POs)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | | **Program Outcomes (POs)** | | | | | | | | | | | | | | | |
| **PO 1** | **PO 2** | **PO 3** | **PO 4** | **PO 5** | **PO 6** | **PO 7** | **PO 8** | **PO 9** | **PO 10** | **PO 11** | **PO 12** | **PSO 1** | **PSO 2** | **PSO 3** |
|  | **CO1** | **3** | **1** | **3** | **1** | **3** | **1** | **1** | **1** | **2** | **3** | **2** | **1** | **3** | **3** | **2** |
| **CO2** | **3** | **2** | **3** | **1** | **3** | **1** | **1** | **1** | **2** | **2** | **2** | **2** | **3** | **3** | **3** |
| **CO3** | **2** | **3** | **3** | **3** | **3** | **2** | **2** | **2** | **2** | **3** | **3** | **3** | **3** | **3** | **3** |

Enter correlation levels 1,2 and 3 as defined below:

**1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High) If there is no correlation, put “-”**

Signature of the Instructor

# INDEX - DJANGO FRAMEWORK

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.NO** | **DATE** | **CONCEPT** | **PAGES** | **MARKS** | **SIGNATURE** |
| 1 | 13-12-2024 | Introduction and Objectives | 5-7 |  |  |
| 2 | 13-12-2024 | Libraries and Technologies Used | 8-12 |  |  |
| 3 | 20-12-2024 | Step-by-Step Guide to Installing Django | 13-15 |  |  |
| 4 | 27-12-2024 | System Architecture | 16-18 |  |  |
| 5 | 03-01-2025 | Implementation and Features | 19-22 |  |  |
| 6 | 24-01-2025 | Setting Up App-Level URLs | 23-25 |  |  |
| 7 | 24-01-2025 | Working with Views in Django | 26-28 |  |  |
| 8 | 31-02-2025 | Integration and Configuration-Settings | 29-32 |  |  |
| 9 | 17-03-2025 | Working with Templates in Django | 33-64 |  |  |
| 10 | 07-03-2025 | Explanation of Key Django Files | 65-66 |  |  |
| 11 | 07-03-2025 | Database Django | 67-69 |  |  |
| 12 | 27-03-2025 | Deploying Django Applications on Cloud Platforms And Conclusion | 70-73 |  |  |
| 13 | 04-03-2025 | Front-End Developer Certificate By Infosys Springboard | 74-75 |  |  |

Signature of the Instructor



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Introduction and Objectives

7. Date of Experiment : 13-12-2024

8. Date of Submission of Report :13-12-2024

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**Project Name :- Digital Campus Notice Board**

**Introduction**

The **Digital Campus Noticeboard** is a web-based platform designed to streamline university announcements and important notices. Traditional methods of distributing information, such as physical noticeboards and email chains, often lead to inefficiencies, missed updates, and delays in communication. This project aims to solve these issues by providing a centralized digital platform where students, faculty, and administrators can access real-time notices efficiently.

With an easy-to-use interface, the **Digital Campus Noticeboard** ensures that university updates, exam notifications, event announcements, and other critical information are accessible anytime. The system allows administrators to post categorized notices.

This project is built using the **Django framework** for backend processing and **SQLite** as the database. The combination of these technologies provides a robust, scalable, and secure environment for managing and displaying notices.

**2. Objectives**

The **Digital Campus Noticeboard** project aims to create an efficient and user-friendly platform for managing university announcements. The key objectives of this project are:

1. **Centralized Information Hub**
   * Provide a single platform where students and faculty can access all university notices in one place.
2. **Real-time Updates**
   * Ensure that notices are published instantly and accessible to users without delays.
3. **Categorized Notices**
   * Organize announcements into categories such as academics, events, exams, and sports to enhance accessibility.
4. **User Accessibility**
   * Allow both students and faculty members to access notices anytime, from any device.
5. **Admin Notice Management**
   * Enable university administrators to post, edit, and categorize notices efficiently.
6. **Search and Filter Functionality**
   * Implement a search bar to help users find specific notices quickly.
7. **Notification System**
   * Provide instant alerts for urgent announcements to ensure timely communication.
8. **Secure and Scalable System**
   * Use Django and SQLite to create a secure, scalable, and reliable platform.
9. **Improved Communication**
   * Reduce dependency on physical noticeboards and manual communication methods, improving information dissemination.
10. **Eco-friendly Solution**

* Minimize paper usage by transitioning to a digital platform for university announcements.

By achieving these objectives, the **Digital Campus Noticeboard** enhances the efficiency, accessibility, and effectiveness of communication within the university.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Libraries and Technologies Used

7. Date of Experiment : 13-12-2024

8. Date of Submission of Report :20-12-2024

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**3.Libraries for Web development**

Python has a vast ecosystem of libraries that simplify **web development**. These libraries help with tasks like building web applications, handling databases, managing APIs, and more.

Here’s a categorized breakdown of some of the best libraries:

**1. Tkinter - GUI Applications**

Tkinter is Python's built-in library for creating Graphical User Interfaces (GUIs).  
It is used for building desktop applications using buttons, labels, text fields, and other widgets.  
Easy to learn and suitable for small to medium-sized projects.

**Key Features:**

1. Drag-and-drop interface using tools like PAGE (Python Automatic GUI Generator).
2. Supports creating complex applications with events, forms, and dialogues.
3. Cross-platform (works on Windows, macOS, and Linux).

**Example: Simple Tkinter App**

***python***

***CopyEdit***

***import tkinter as tk***

***def on\_click():***

***label.config(text="Hello, " + entry.get())***

***root = tk.Tk()***

***root.title("Simple Tkinter App")***

***label = tk.Label(root, text="Enter your name:")***

***label.pack()***

***entry = tk.Entry(root)***

***entry.pack()***

***button = tk.Button(root, text="Click Me!", command=on\_click)***

***button.pack()***

***root.mainloop()***

**A screenshot of a computer screen

AI-generated content may be incorrect.Use Case:** Building a simple calculator, to-do list, or contact management app.

**2. BeautifulSoup4 - Web Scraping**

BeautifulSoup4 (BS4) is a Python library used for web scraping.  
It parses HTML and XML documents and extracts data from websites.  
Often used for gathering data from pages that don’t have APIs.

**Key Features:**

1. Easy-to-use HTML parser.
2. Supports extracting elements using tags, attributes, and classes.
3. Compatible with other libraries like Requests and lxml.

**Example: Scraping a Website**

***python***

***import requests***

***from bs4 import BeautifulSoup***

***url = 'https://example.com'***

***response = requests.get(url)***

***soup = BeautifulSoup(response.text, 'html.parser')***

***for heading in soup.find\_all('h1'):***

***print(heading.text)***

**Use Case:** Extracting news articles, stock data, or product details from e-commerce sites.

**3. CherryPy - Web Framework**

CherryPy is a minimal, object-oriented web framework for Python.  
It allows you to build web applications quickly without needing additional tools.  
It has a built-in web server and is often used for lightweight applications.

**Key Features:**

1. Supports RESTful APIs.
2. Integrated web server (no external server needed).
3. Great for small-scale projects like internal tools.

**Example: Simple CherryPy App**

***python***

***import cherrypy***

***class HelloWorld:***

***@cherrypy.expose***

***def index(self):***

***return "Hello, CherryPy!"***

***if \_\_name\_\_ == '\_\_main\_\_':***

***cherrypy.quickstart(HelloWorld())***

**Use Case:** Developing REST APIs, small admin dashboards, or internal web tools.

**4. Bottle - Lightweight Web Framework**

Bottle is a micro web framework for Python.  
It is designed for small web applications and has no dependencies other than Python.  
It is similar to Flask but with fewer features, making it ideal for small projects.

**Key Features:**

1. Built-in development server.
2. Supports routing, templates, and form handling.
3. Easily deployable on platforms like Heroku and PythonAnywhere.

**Example: Simple Bottle App**

**python**

***from bottle import Bottle, run***

***app = Bottle()***

***@app.route('/')***

***def home():***

***return "Hello, Bottle Framework!"***

***run(app, host='localhost', port=8080)***

**Use Case:** Building simple REST APIs, prototypes, or web applications with minimal complexity.

**4. Technologies Used**

The **Digital Campus Noticeboard** project utilizes modern web development technologies to ensure a secure, efficient, and scalable platform for university announcements. The key technologies used in this project are:

**1. Backend Development**

1. **Django (Python Framework)**
   1. Django is a high-level Python web framework that follows the **Model-View-Template (MVT)** architecture.
   2. It provides built-in security features, rapid development capabilities, and a structured approach to web application development.
   3. Used for handling user authentication, database interactions, and backend logic.
2. **SQLite (Database Management System)**
   1. A lightweight, embedded SQL database engine used to store notices, user details, and other relevant data.
   2. Provides a simple and efficient database solution for small-to-medium-scale applications like this noticeboard.
   3. Ensures quick retrieval and management of notices.
3. **Frontend Development**
4. **HTML (HyperText Markup Language)**
   * 1. Used for structuring web pages and displaying notices, user profiles, and other interface elements.
5. **CSS (Cascading Style Sheets)**
   1. Applied for styling the user interface to improve the visual appeal and responsiveness of the platform.
   2. Ensures a clean and modern UI design.
6. **JavaScript**
   1. Enhances interactivity on the platform, such as dynamic updates, form validation, and search functionality.

**3. Additional Technologies**

1. **Django Admin Panel**
   1. Provides an easy-to-use interface for university administrators to manage notices.
2. **Git & GitHub**
   1. Version control system (Git) and cloud repository (GitHub) for tracking project development and collaboration.

By leveraging these technologies, the **Digital Campus Noticeboard** ensures a **fast, secure, and user-friendly** platform for university-wide communication.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Django Installation

7. Date of Experiment : 20-12-2024

8. Date of Submission of Report :27-12-2024

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**5.Django & Installation**

Django is a **high-level Python web framework** that allows developers to build secure, scalable, and maintainable web applications. It follows the **Model-View-Template (MVT)** architectural pattern and provides a wide range of built-in features to simplify web development.

It is commonly used for:

1. Web applications
2. API development
3. Content management systems
4. E-commerce platforms

**Installation Of Django**

**Step 1: Install Python**

Django requires Python, so ensure it is installed on your system.

* Check if Python is installed:

***python –version***

**Step 2: Install Pip (Python Package Manager)**

Pip is usually installed with Python. Verify using

***pip –version***

**Step 3: Create a Virtual Environment**

A virtual environment helps keep dependencies isolated.

1. Open your terminal or command prompt.
2. Navigate to your desired project directory:

***cd path/to/your/project***

Create a virtual environment:

***python -m venv env***

Activate the virtual environment:

***python -m venv env***

**Step 4: Install Django**

Once the virtual environment is active, install Django using Pip:

***pip install Django***

**Step 5: Create a Django Project**

Now that Django is installed, create separate projects for **Digital Campus Noticeboard**.

***django-admin startproject digital\_noticeboard***

***cd digital\_noticeboard***

**Step 6: Create Django Apps**

In Django, you can create different apps for different features.

**Example Apps:**

**For Digital Campus Noticeboard**

1. Notices (For managing notices)
2. AdminPanel (For admin management)

***python manage.py startapp notices***

***python manage.py startapp adminpanel***

***Step 7: Run the Server***

Start the development server

***python manage.py runserver***

Access your app at:

1. **http://127.0.0.1:8000/**
2. **http://127.0.0.1:8000/admin** (For the admin panel)



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : System Architecture

7. Date of Experiment : 27-12-2024

8. Date of Submission of Report :03-01-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**6. System Architecture**

The **Digital Campus Noticeboard** follows a structured **three-tier architecture**, ensuring a well-organized and scalable system. The architecture consists of three main layers:

**1. Presentation Layer (Frontend)**

1. This layer is responsible for the **user interface (UI)** and user interaction.
2. Technologies used: **HTML, CSS, JavaScript, and Bootstrap** for designing a responsive and interactive UI.
3. It includes:
   1. **Home Page** – Displays an overview of notices.
   2. **Notifications Page** – Shows categorized university notices.
   3. **Account Section** – Enables user login and profile management.
   4. **Contact Form** – Allows users to submit feedback or queries.

**2. Application Layer (Backend – Django Framework)**

1. Acts as the **business logic layer**, processing user requests and interacting with the database.
2. Built using **Django (Python framework)**, which follows the **Model-View-Template (MVT)** architecture.
3. Key responsibilities:
   1. Handling user authentication (login, signup).
   2. Managing notices (creating, editing, deleting).
   3. Processing search and filtering functionalities.
   4. Rendering templates dynamically based on user roles.

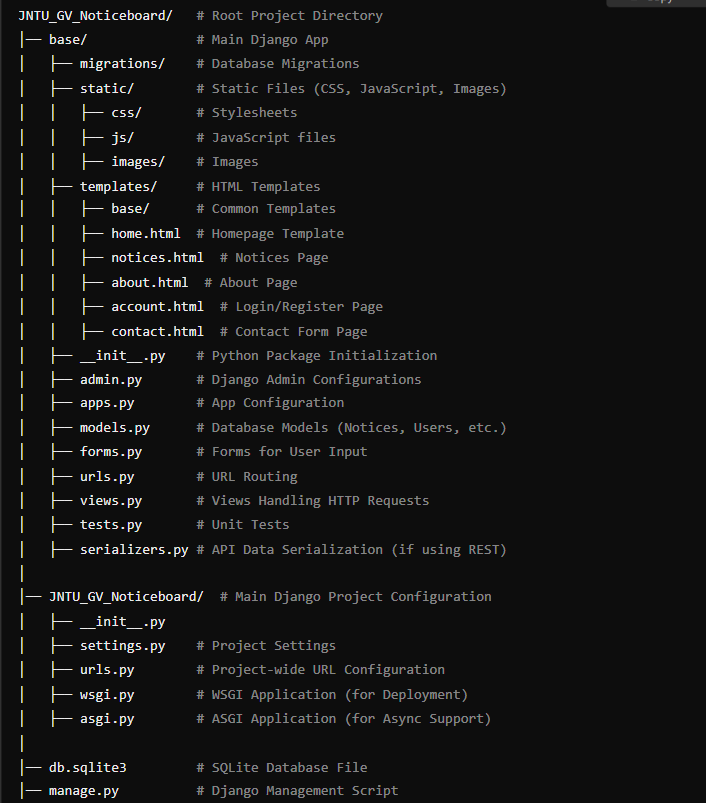
**MVT (Model-View-Template) Breakdown:**

1. **Model (M)** – Defines the database structure (e.g., storing notices, user profiles).
2. **View (V)** – Handles requests and fetches data from the database.
3. **Template (T)** –Displays the processed data to users via HTML.

**3. Data Layer (Database – SQLite)**

* Stores all project-related data, including:
  1. **User details** (students, faculty, admins).
  2. **Notices** (categories, timestamps, content).
  3. **Feedback messages** from the contact form.
* SQLite is chosen for its **lightweight** and **efficient** performance for small-to-medium-sized applications.

This **three-tier system architecture** ensures that the **Digital Campus Noticeboard** is **secure, scalable, and efficient** in handling university-wide announcements.





**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Features and Implementation

7. Date of Experiment : 03-01-2025

8. Date of Submission of Report :24-02-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**7. Features**

The Digital Campus Noticeboard is designed to enhance university communication by providing a centralized platform for posting and accessing notices. Below are the key features:

**1. Admin Features**

1. **Notice Management**
   1. Admins can post, edit, and delete notices in different categories.
2. **Categorization of Notices**
   1. Notices are organized into categories such as Academics, Exams, Events, Sports, and General Updates for easy access.
3. **Secure Admin Panel**
   1. Only authorized administrators can manage notices using the Django Admin Panel

**2. Student & Faculty Features**

1. **Real-time Access to Notices**
   1. Students and faculty can view newly published notices instantly.
2. **User Authentication System**
   1. Users must log in or sign up to access certain features.
3. **Search Functionality**
   1. Users can search notices by keywords for quick retrieval of relevant announcements.
4. **Mobile-Friendly UI**
   1. A responsive design ensures accessibility on desktops, tablets, and smartphones.

**3. Additional Features**

1. **Urgent Notification System**
2. Important notices are highlighted and can trigger alerts for users.
3. **Contact & Feedback Form**
4. Users can submit queries or feedback, which will be stored for admin review**.**
5. **Fast & Lightweight Performance**
6. Uses SQLite database for quick data retrieval and Django caching for optimized performance.
7. **Secure & Scalable**

Built with Django security features to protect user data and prevent unauthorized access

**8. Implementation Details – Project Structure**

The **Digital Campus Noticeboard** project follows a structured **Django project architecture**, ensuring maintainability and scalability. Below is an explanation of the **directory structure** from the provided image:

**Root Directory: DIGITAL NOTICE BOARD**

This is the main folder containing all files related to the project, including the virtual environment (.venv), the Django project (university), the database (db.sqlite3), and the Django management script (manage.py).

**Main Django Project Folder: university/**

* This is the main Django project directory that contains settings and configurations.

1. It includes the primary settings.py, urls.py, and wsgi.py files (not shown in the image but typically present).

**Django App Folder: base/**

This is the core Django application where all the main functionalities are implemented. It contains:

1. **migrations/**
   * Stores database migration files that Django generates to track model changes.
2. **static/base/images/**
   * Contains static files such as images, CSS, and JavaScript used for frontend design.
3. **templates/ (HTML Templates)**
   1. Holds all the frontend HTML files used for rendering the web pages.
   2. Files included:
      1. about.html – Displays information about the university.
      2. account.html – Manages user login/signup and profiles.
      3. contact.html – Contains the contact/feedback form.
      4. home.html – The homepage displaying an overview of notices.
      5. notices.html – Shows all university notices.
4. **Python Files in base/**
   1. \_\_init\_\_.py – Marks this folder as a Python package.
   2. admin.py – Registers models to be managed via the Django admin panel.
   3. apps.py – Defines the configuration for the base application.
   4. models.py – Contains database models (e.g., Notices, Users).
   5. tests.py – Used for writing test cases to ensure functionality.
   6. urls.py – Defines URL patterns for routing requests.
   7. views.py – Contains functions to handle requests and return responses.

**Database File: db.sqlite3**

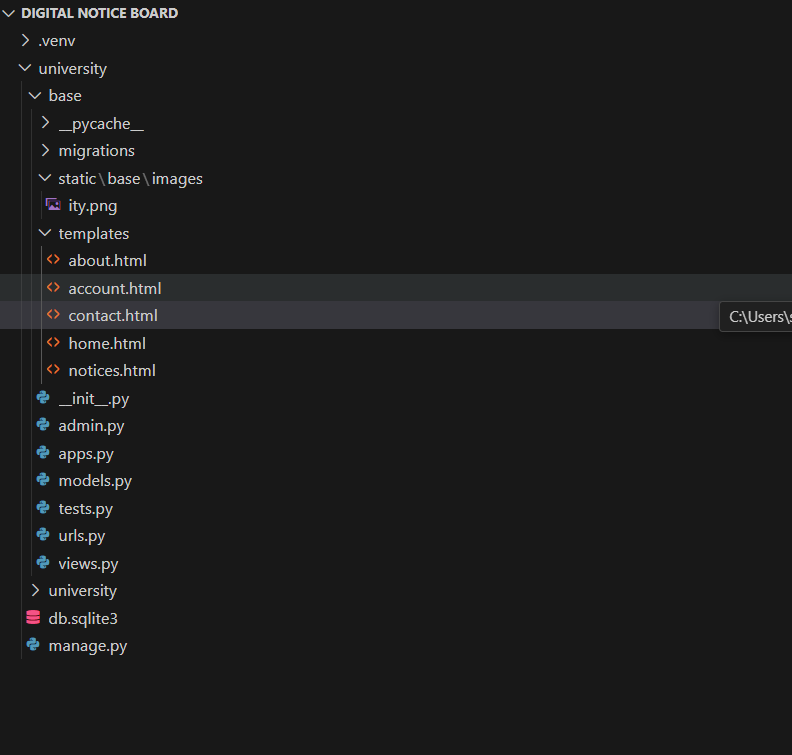
* SQLite database file where all application data (notices, users, etc.) is stored.

**Django Management Script: manage.py**

* The main script for managing the Django project (running the server, migrations, etc.).

**Summary**

This **structured architecture** ensures:  
**Separation of concerns** (backend, frontend, database).  
 **Scalability and maintainability** for future enhancements.  
 **Easy navigation** for developers working on the project.





**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Setting Up App-Level URLs

7. Date of Experiment : 24-02-2025

8. Date of Submission of Report :24-02-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**9.URL Configuration**

In Django, urls.py is responsible for defining URL patterns that direct user requests to the appropriate views. It acts as a mapping system, connecting different web pages to specific functions that generate responses.

**Explanation of the Code**

**1.Importing Required Modules**

* 1. The necessary modules are imported to enable URL configuration.
  2. The path function is used to define URL routes.
  3. The views module is imported to link URLs with corresponding view functions.

1. **Defining URL Patterns**
   1. A list named urlpatterns is created to store URL mappings.
   2. Each entry in the list specifies a route and the view function responsible for handling requests to that route.
2. **URL Mappings and Their Functions**
   1. The first pattern maps the root URL to a function that renders the homepage.
   2. The second pattern directs requests for notices to a function that displays notifications.
   3. The third pattern links to an account-related page, where users can log in or view their profile.
   4. The fourth pattern maps to an "About" page that provides information about the project.
   5. The fifth pattern links to a contact page where users can submit queries or feedback.

**URLS In Project**

This Django code defines the URL routing for a web application using the urlpatterns list. It begins by importing the path function from django.urls and the views module from the current directory. The urlpatterns list contains multiple path() entries that map specific URL paths to their corresponding view functions in views.py. The empty string '' maps to the home view, serving as the homepage of the application. The 'notices/' path routes to the notices view, which likely displays all notices or announcements. Similarly, 'account/' leads to the user account page handled by the account view, 'about/' points to the about view that provides information about the website or organization, and 'contact/' links to the contact view for users to reach out or submit inquiries. Each path also includes a name parameter, which assigns a unique name to the URL, making it easier to reference within Django templates using the {% url %} template tag. Overall, this URL configuration allows users to navigate different parts of the site by mapping URLs to their respective logic and templates

.**How It Works**

When a user visits a specific URL in the browser, Django checks the urls.py file to find a matching pattern. If a match is found, the corresponding function in views.py is executed to generate a response.

**CODE:-**

***from django.urls import path***

***from . import views***

***urlpatterns = [***

***path('', views.home, name='home'), # Homepage***

***path('notices/', views.notices, name='notices'), # Notices page***

***path('account/', views.account, name='account'), # Account page***

***path('about/', views.about, name='about'), # About page***

***path('contact/', views.contact, name='contact'), # Contact page***



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Working with Views in Django

7. Date of Experiment : 24-01-2025

8. Date of Submission of Report :31-01-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

]

**10.Views in Django**

In Django, views.py contains functions that handle user requests and return responses. Each view function is responsible for rendering a webpage or processing data before displaying it to the user.

**Explanation of the Code (Without Code Snippets)**

1. **Importing Necessary Modules**
   1. A function is imported to help render HTML templates.
   2. This function allows the views to return properly formatted web pages.
2. **Defining View Functions**
   1. Each function represents a webpage and takes a request as input.
   2. The function returns a rendered template corresponding to the requested page.
3. **View Functions and Their Purpose**
   1. The first function loads the homepage when users visit the root URL.
   2. The second function is responsible for displaying notices when users navigate to the notices section.
   3. The third function handles user account-related pages, where login and profile information are shown.
   4. The fourth function displays an "About" page, providing details about the project.
   5. The fifth function loads a contact page where users can submit feedback or inquiries.

**How Views Works In Project:-**

This Django views.py code defines five simple view functions that handle different pages of a website. Each view takes a request object as an argument and uses Django’s render() function to return an HTML page as a response. The home function renders the homepage using the home.html template. The notices function displays the notices page by rendering notices.html. Similarly, the account, about, and contact functions render account.html, about.html, and contact.html, respectively. These views are connected to the URL patterns defined in the urls.py file. When a user visits a specific URL, the corresponding view is called, and the appropriate HTML template is displayed in the browser. This structure separates the logic (views) from the presentation (templates), making the web app organized and easy to manage.

**How It Works**

When a user visits a URL, Django directs the request to the appropriate function in views.py. The function then fetches the required HTML template and sends it as a response to the user.

**CODE:-**

***from django.shortcuts import render***

***def home(request):***

***return render(request, 'home.html') # Render the homepage***

***def notices(request):***

***return render(request, 'notices.html') # Render the notices page***

***def account(request):***

***return render(request, 'account.html') # Render the account page***

***def about(request):***

***return render(request, 'about.html') # Render the about page***

***def contact(request):***

***return render(request, 'contact.html') # Render the contact page***



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Integration and Configuration

7. Date of Experiment : 31-02-2025

8. Date of Submission of Report :17-03-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**11.Django Settings**

**Introduction**

The settings.py file in Django is a core configuration file that controls how the Django project functions. It contains various settings related to the database, installed applications, security, static files, templates, and other essential configurations.

**Explanation of Key Configurations**

**1. Installed Applications**

1. This section lists all the Django apps that are part of the project.
2. It includes built-in Django applications (such as authentication and admin panel) and custom applications developed for the project.

**2. Middleware**

1. Middleware is a framework that processes requests and responses before they reach the view or the user.
2. It helps with security, session management, and various request modifications.

**3. Templates Configuration**

1. This section defines how Django should locate and render HTML templates.
2. It includes settings related to the template engine and the directories where templates are stored.

**4. Static Files and Media**

1. Static files (CSS, JavaScript, images) are stored separately and served using this configuration.
2. Media files (user-uploaded content) are also configured here.

**5. Database Configuration**

1. This section defines how Django connects to the database.
2. It includes settings such as database engine, name, username, password, and host.
3. SQLite is commonly used for small projects, while larger applications use PostgreSQL or MySQL.

**6. Authentication and User Management**

1. Settings related to user authentication, login redirection, and password validation are defined in this section.
2. It ensures that only authorized users can access certain parts of the application.

**7. Security Settings**

Security-related configurations, such as allowed hosts, CSRF protection, and session management, are included.

**8. Debug Mode**

1. Debug mode helps developers troubleshoot errors during development.
2. When enabled, Django provides detailed error messages for debugging.
3. In a production environment, debug mode is turned off for security reasons.

**9. Internationalization and Time Zone**

1. This section allows configuring the language and time zone settings for the project.
2. It ensures that the application functions correctly across different regions.
3. The settings.py file plays a crucial role in defining how a Django project operates. It manages everything from application settings and security configurations to database connections and static file handling. Proper configuration ensures smooth functionality, security, and scalability of the Django project.

**CODE:-**

***INSTALLED\_APPS = [***

***'django.contrib.admin', # Admin site***

***'django.contrib.auth', # Authentication system***

***'django.contrib.contenttypes', # Content type framework***

***'django.contrib.sessions', # Session framework***

***'django.contrib.messages', # Messaging framework***

***'django.contrib.staticfiles', # Static file management***

***'base', # Custom app***

***]***

***MIDDLEWARE = [***

***'django.middleware.security.SecurityMiddleware', # Security enhancements***

***'django.contrib.sessions.middleware.SessionMiddleware', # Session support***

***'django.middleware.common.CommonMiddleware', # Common HTTP middleware***

***'django.middleware.csrf.CsrfViewMiddleware', # CSRF protection***

***'django.contrib.auth.middleware.AuthenticationMiddleware', # Authentication support***

***'django.contrib.messages.middleware.MessageMiddleware', # Message framework***

***'django.middleware.clickjacking.XFrameOptionsMiddleware', # Clickjacking protection***

***]***

***ROOT\_URLCONF = 'university.urls' # Main URL configuration***

***TEMPLATES = [***

***{***

***'BACKEND': 'django.template.backends.django.DjangoTemplates', # Use Django template engine***

***'DIRS': [BASE\_DIR / 'templates'], # Custom templates directory***

***'APP\_DIRS': True, # Enable app-level templates***

***'OPTIONS': {***

***'context\_processors': [***

***'django.template.context\_processors.debug', # Debugging context***

***'django.template.context\_processors.request', # Request object in templates***

***'django.contrib.auth.context\_processors.auth', # Auth context***

***'django.contrib.messages.context\_processors.messages', # Messages context***

***],***

***},***

***},***



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Working with Templates

7. Date of Experiment : 17-03-2025

8. Date of Submission of Report :07-03-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

]**12. Templates in the Digital Campus Noticeboard**

What are Templates in Django?

Templates in Django are HTML files used to render dynamic content. Django uses the Model-View-Template (MVT) architecture, where:

1. Models handle the data.
2. Views process user requests.
3. Templates display the data in a structured format.

Django’s template engine allows using dynamic placeholders, loops, and conditions within HTML files to make pages interactive.

**Templates Used in the Project**

In the Digital Campus Noticeboard, the templates/ folder contains several HTML files that define different pages of the platform. The following templates are used:

1. **home.html (Home Page)**

1. Displays an overview of the platform.
2. Provides quick access to notices and important announcements.
3. Includes navigation links to different sections of the website.

2. **notices.html (Notices Page)**

1. Displays all university notices and announcements.
2. Notices are categorized (e.g., Academics, Events, Sports, Exams).
3. Users can search for specific notices using keywords.

3. **about.html (About Page)**

1. Provides information about the university and its mission.
2. May include details about the campus, departments, and administration.

4. **account.html (Account Page)**

1. Manages user authentication (login/signup).
2. Displays user profiles for students and faculty.
3. Users can update their personal details and passwords.

5. c**ontact.html (Contact Form Page)**

1. Allows users to submit feedback or queries.
2. Collects details like name, email, message, and sends it to the admin.

The Digital Campus Noticeboard uses five main templates for displaying notices, user accounts, contact forms, and other essential pages. These templates ensure a structured, user-friendly, and responsive experience**.**

**13.Home Page Template**

**Home Template Explanation**

The **Home Template** of the **JNTU-GV Digital Campus Noticeboard** project serves as the landing page, providing an overview of the university and its digital campus services. It is designed with a clean, structured layout, engaging visuals, and interactive elements to enhance user experience.

**Key Components of the Home Template**

**1. Header Section**

1. Displays the **JNTU-GV logo** along with the **university name and details**.
2. Gives a professional and authentic look to the website.

**2. Navigation Bar**

* A top menu bar that includes **links to major sections** like:

**Home**

**Notices**

**About**

**Contact**

**Account** (For login/signup)

* Improves accessibility and allows users to navigate smoothly between pages.

**3. Headline Scroller (Latest Announcements)**

A **scrolling text section** displaying **important updates** like:

Admission notifications

Exam schedules

Placement drives

Upcoming workshops and events

Adds a dynamic element to keep users informed.

**4. Welcome Section**

Provides a brief **introduction about JNTU-GV** and its significance.

Gives users an idea about the university’s history, mission, and focus on education.

**5. Image Slider (Campus Highlights)**

A **carousel-style image slider** displaying:

University campus images

Infrastructure highlights

Events and activities

1. Enhances user engagement with visual content.
2. Comes with **navigation arrows** and **dot indicators** for easy browsing.

**6. Footer Section**

* Contains **contact details** of JNTU-GV, including:

University address

Social media links (Facebook, LinkedIn, Twitter, Instagram, YouTube)

Privacy policy

Copyright notice and credits to the developer.

Provides **quick access to external resources** and **university’s official channels**.

**User Experience & Design Enhancements**

1. **Color Scheme:** Uses a professional and attractive gradient background.
2. **Typography:** Uses the "Poppins" font for a **modern and clean look**.
3. **Animations & Effects:**

The **headline scroller continuously scrolls** to catch user attention.

The **image slider automatically transitions** but can also be manuallcontrolled.

* + **Hover effects on navigation links** for better interactivity.

1. **Responsive Design:**
   * The template is **mobile-friendly** and adapts to different screen sizes.
   * The **footer and social media icons resize** for smaller screens.

**Purpose of the Home Template**

1. Acts as the **entry point** for students and faculty to access notices and updates.
2. Provides **quick navigation** to key sections.
3. Keeps users engaged with **interactive elements** like the **headline scroller and image slider**.
4. Establishes the **university’s online presence** in an informative and visually appealing manner.

This template ensures a **user-friendly, informative, and visually engaging experience** for students, faculty, and visitors accessing the **JNTU-GV Digital Campus Noticeboard**.

**CODE:-**

***<!DOCTYPE html><html lang="en"><head><meta charset="UTF-8"><meta name="viewport" content="width=device-width, initial-scale=1.0"><title>JNTU-GV Digital Campus Noticeboard</title>***

***<!-- Fonts and Icons -->***

***<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet">***

***<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css">***

***<!-- Inline CSS -->***

***style>body{font-family:'Poppins',sans-serif;margin:0;padding:0;background:linear-gradient(to right,#007bff,#6610f2);color:#fff;scroll-behavior:smooth}header{display:flex;align-items:center;background-color:rgba(255,255,255,0.95);padding:15px 20px;box-shadow:0px 5px 10px rgba(0,0,0,0.1);border-radius:0 0 15px 15px}header img{height:80px;margin-right:15px}header div{color:#333}nav{background-color:#002855;padding:12px 0;box-shadow:0 4px 6px rgba(0,0,0,0.3)}nav ul{list-style:none;margin:0;padding:0;display:flex;justify-content:center}nav ul li{margin:0 25px;position:relative}nav ul li a{text-decoration:none;color:white;font-size:18px;font-weight:bold;transition:color 0.3s;position:relative;padding-bottom:5px}nav ul li a:hover{color:#FFC107}nav ul li a::after{content:"";width:0;height:2px;background-color:#FFC107;position:absolute;bottom:0;left:50%;transform:translateX(-50%);transition:width 0.3s ease-in-out}nav ul li a:hover::after{width:100%}.headline-wrapper{width:100%;overflow:hidden;background:#002855;padding:10px 0;position:relative}.headline-scroller{display:flex;gap:50px;white-space:nowrap;animation:scroll-left 15s linear infinite}.headline-scroller p{color:white;font-size:18px;font-weight:bold;display:inline-block;margin:0;padding:0 20px;animation:rotate-text 4s infinite alternate}@keyframes scroll-left{from{transform:translateX(100%)}to{transform:translateX(-100%)}}@keyframes rotate-text{0%{transform:rotate(0deg)}100%{transform:rotate(3deg)}}.headline-wrapper:hover .headline-scroller{animation-play-state:paused}section{padding:20px;margin:20px;background:rgba(255,255,255,0.95);border-radius:15px;box-shadow:0px 4px 10px rgba(0,0,0,0.2);animation:fadeIn 1s ease-in-out;color:#333}.slider{position:relative;max-width:900px;height:300px;margin:auto;overflow:hidden;border-radius:10px;box-shadow:0px 5px 15px rgba(0,0,0,0.4)}.slides{display:flex;transition:transform 0.6s ease-in-out;width:400%}.slides img{width:100%;height:100%;object-fit:cover;border-radius:10px}.prev,.next{position:absolute;top:50%;transform:translateY(-50%);background-color:rgba(0,0,0,0.5);color:white;border:none;padding:12px;cursor:pointer;font-size:22px;border-radius:50%;transition:background-color 0.3s ease}.prev:hover,.next:hover{background-color:rgba(255,193,7,0.8)}.prev{left:10px}.next{right:10px}.dots{text-align:center;margin-top:10px}.dot{height:10px;width:10px;margin:0 5px;background-color:#bbb;border-radius:50%;display:inline-block;transition:background-color 0.3s ease;cursor:pointer}.active-dot{background-color:#ff8800}#footer{background:url('footer-bg.jpg') center/cover no-repeat;padding:30px 0;position:relative}footer{color:white;text-align:center}.footer-content{backdrop-filter:blur(5px);background:rgba(0,0,0,0.5);padding:20px;border-radius:5px;max-width:800px;margin:auto}.footer-content p{margin:10px 0;font-size:14px}strong{color:orange}.social-icons a{color:white;margin:0 10px;font-size:20px;transition:0.3s;text-decoration:none}.social-icons a:hover{color:orange}.privacy{display:block;margin:10px 0;color:white;text-decoration:none}.privacy:hover{text-decoration:underline}@media(max-width:600px){.footer-content{padding:15px}.social-icons a{font-size:16px;margin:0 8px}.footer-content p{font-size:12px}}</style></head>***

***<body>***

***<!-- Header Section -->***

***<header><img src="https://tse3.mm.bing.net/th?id=OIP.ZMmOL1kqTR2H1yYs4ZZKhgHaHc&pid=Api&P=0&h=180" alt="JNTU-GV Logo"><div><h1>JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA VIZIANAGARAM</h1><p>VIZIANAGARAM - 535 003, A.P (Established by Andhra Pradesh Act No.22 of 2021)</p></div></header>***

***<!-- Navigation -->***

***<nav><ul><li><a href="{% url 'home' %}">Home</a></li><li><a href="{% url 'notices' %}">Notices</a></li><li><a href="{% url 'about' %}">About</a></li><li><a href="{% url 'contact' %}">Contact</a></li><li><a href="{% url 'account' %}">Account</a></li></ul></nav>***

***<!-- Headline Scroller -->***

***<div class="headline-wrapper"><div class="headline-scroller"><p>📢 Admission Open for 2025-26! Apply Now.</p><p>🏆 JNTU-GV Won Best University Award 2025!</p><p>📝 Exam Notification: Semester Exams Begin from April 15.</p><p>🎓 Placement Drive: Top MNCs Hiring on March 20.</p><p>📅 Workshop on AI & ML – Register Before March 10.</p></div></div>***

***<!-- Home Section -->***

***<section id="home"><h1>Welcome to JNTU-GV Digital Campus</h1><p>The Jawaharlal Nehru Technological University-Gurajada, Vizianagaram (JNTU-GV) is a public university located in Vizianagaram, Andhra Pradesh. It was established on 12 January 2022. The university was named after Gurajada Apparao, a noted Indian playwright, dramatist, poet, and writer known for his works in Telugu theatre. The university mainly focuses on engineering.</p>***

***<!-- Image Slider -->***

***<div class="slider" id="slider"><div class="slides"><img src="https://tse4.mm.bing.net/th?id=OIP.lZZr6dW7qSNgiL7YOqvOLgHaCU&pid=Api&P=0&h=180" alt="Image 1"><img src="https://tse1.mm.bing.net/th?id=OIP.-jFFHOj47ri8M-vKBY1QmAHaCx&pid=Api&P=0&h=180" alt="Image 2">{% load static %}<img src="{% static 'base/images/ity.png' %}" alt="My Image"></div>***

***<button class="prev" onclick="moveSlide(-1)">&#10094;</button><button class="next" onclick="moveSlide(1)">&#10095;</button></div>***

***<!-- Slider Dots -->***

***<div class="dots"><span class="dot" onclick="currentSlide(0)"></span><span class="dot" onclick="currentSlide(1)"></span><span class="dot" onclick="currentSlide(2)"></span></div></section>***

***<!-- Footer Section -->***

***<section id="footer"><footer><div class="footer-content"><p><strong>Contact Us:</strong> JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY-GURAJADA VIZIANAGARAM, DWARAPUDI, VIZIANAGARAM, ANDHRA PRADESH - 535 003, Andhra Pradesh, INDIA.</p>***

***<div class="social-icons"><a href="https://www.facebook.com/JntuVizianagaram/" target="\_blank"><i class="fab fa-facebook-f"></i></a><a href="https://in.linkedin.com/school/jntugv/" target="\_blank"><i class="fab fa-linkedin-in"></i></a><a href="https://x.com/JNTU\_Gurajada" target="\_blank"><i class="fab fa-twitter"></i></a><a href="https://www.instagram.com/jntu\_gurajada/" target="\_blank"><i class="fab fa-instagram"></i></a><a href="https://www.youtube.com/@JNTUGV" target="\_blank"><i class="fab fa-youtube"></i></a></div>***

***<a href="#" class="privacy">Privacy & Policy</a>***

***<p>© Copyright 2024 <strong>JNTU-GV Vizianagaram</strong>. All Rights Reserved.</p>***

***<p>Designed and Developed by <strong>M.Likhith Kumar, JNTU-GV</strong></p></div></footer></section>***

***<!-- Slider Script -->***

***<script>***

***document.addEventListener("DOMContentLoaded", function () {***

***let scroller = document.querySelector(".headline-scroller");***

***let clone = scroller.innerHTML;***

***scroller.innerHTML += clone;***

***});***

***let currentIndex = 0;***

***const slides = document.querySelector('.slides');***

***const dots = document.querySelectorAll('.dot');***

***let autoSlideInterval;***

***function updateSlider() {***

***slides.style.transform = `translateX(-${currentIndex \* 100}%)`;***

***dots.forEach(dot => dot.classList.remove('active-dot'));***

***dots[currentIndex].classList.add('active-dot');***

***}***

***function moveSlide(direction) {***

***currentIndex += direction;***

***if (currentIndex < 0) currentIndex = slides.children.length - 1;***

***if (currentIndex >= slides.children.length) currentIndex = 0;***

***updateSlider();***

***resetAutoSlide();***

***}***

***function currentSlide(index) {***

***currentIndex = index;***

***updateSlider();***

***resetAutoSlide();***

***}***

***function autoSlide() {***

***autoSlideInterval = setInterval(() => {***

***moveSlide(1);***

***}, 3000);***

***}***

***function resetAutoSlide() {***

***clearInterval(autoSlideInterval);***

***autoSlide();***

***}***

***autoSlide();***

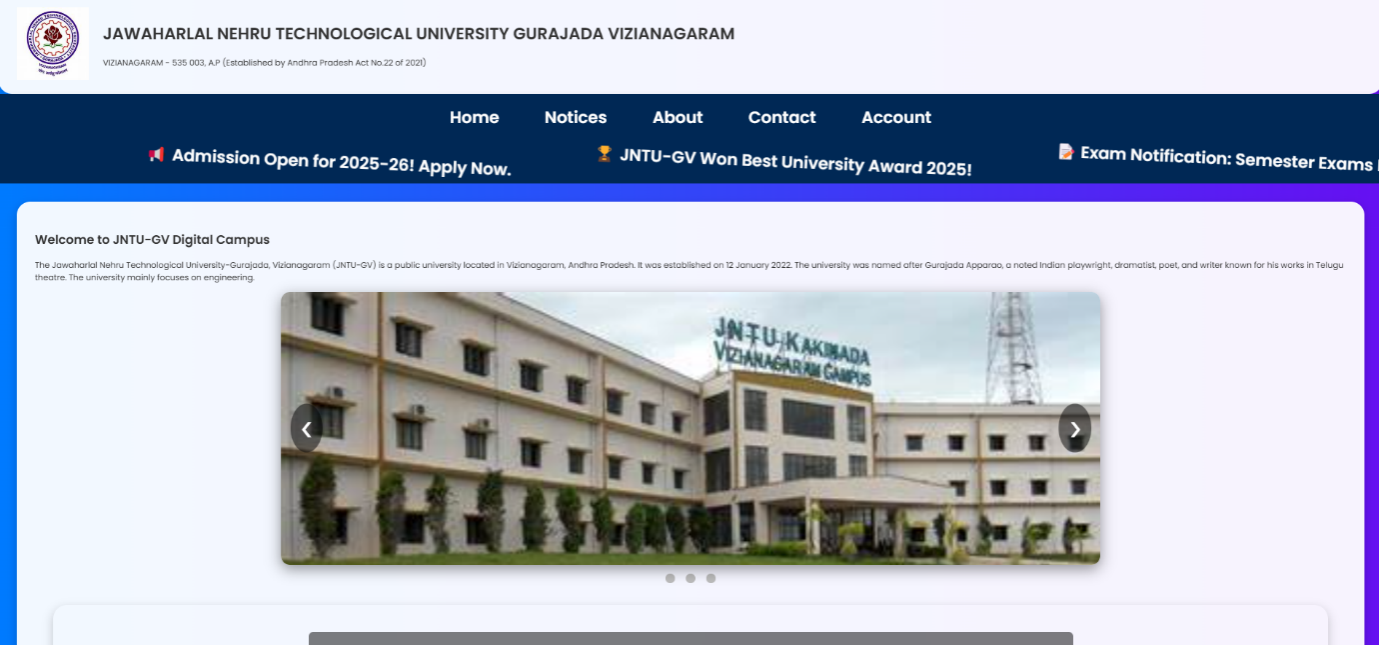
***document.getElementById('slider').addEventListener('mouseover', () => clearInterval(autoSlideInterval));***

***document.getElementById('slider').addEventListener('mouseleave', resetAutoSlide);***

***</script>***

***</body></html***

>**OUTPUT:-**



**14.Notices Page Templates**

**Notifications Template Explanation (Without Code)**

The **Notifications Template** in the **JNTU-GV Digital Campus Noticeboard** serves as the **central hub** for students and faculty to view categorized **university notices and updates**. It is structured to be **organized, interactive, and user-friendly**, allowing users to quickly access the information they need.

**Key Components of the Notifications Template**

**1. Header Section**

1. Displays the **university logo** and name at the top of the page.
2. Provides **institution details**, ensuring an official and professional look.

**2. Navigation Bar**

1. Contains **links to different sections** of the platform:
   * **Home** – Redirects to the homepage.
   * **Notices** – Current page displaying all notifications.
   * **About** – Information about the university.
   * **Contact** – Provides a contact form for feedback.
   * **Account** – For user login and profile management.
2. Enhances **navigation and accessibility** for users.

**3. Sidebar (Notice Categories)**

A vertical **sidebar menu** allows users to filter notices by category:

**Exams** – Displays all examination-related updates.

**Events** – Lists cultural and academic events happening at JNTU-GV.

**Sports** – Provides updates on university sports activities.

**Academic Events** – Highlights workshops, seminars, and other educational events.

**Interactive buttons** allow users to switch between categories seamlessly.

**4. Main Content Section (Notifications Display)**

Displays notices **dynamically** based on the selected category.

Each notice includes:

**Date Box** – Clearly shows the date of the announcement.

**Notice Content** – Title and description of the notification.

**Hover Effects** – Notices slightly enlarge when hovered over, making them more engaging.

**5. Interactive Features**

1. **Dynamic Category Switching:**

When a user clicks a **sidebar button**, the relevant **notices appear smoothly**, while others are hidden.

This **eliminates page reloading**, ensuring a seamless experience.

1. **Hover Effects:**

Notices change **background color and scale slightly** when hovered over.

Improves **visual appeal and user engagement**.

**6. Styling and Design**

**Modern and Clean UI:**

Uses a **purple gradient sidebar** for a sleek look.

White background for notices, enhancing readability.

**Animations & Transitions:**

**Fade-in effects** for notices.

**Smooth sidebar interactions** when switching categories.

**Mobile Responsiveness:**

The page adapts to different screen sizes.

Ensures easy access across **phones, tablets, and desktops**.

**Purpose of the Notifications Template**

**Categorized Notice Display:** Helps users quickly find relevant announcements.  
**Improved User Experience:** Interactive buttons and animations make browsing easier.  
**Real-Time Updates:** Displays latest notices efficiently.  
 **Modern and Professional Look:** Clean design ensures better engagement.

This template ensures that students and faculty can **quickly and easily access important university updates**, improving **communication and efficiency** within JNTU-GV

**CODE:-**

***<!DOCTYPE html>***

***<html lang="en">***

***<head>***

***<meta charset="UTF-8" />***

***<meta name="viewport" content="width=device-width, initial-scale=1.0" />***

***<title>Notifications | JNTU-GV University</title>***

***</head>***

***<body style="margin:0;">***

***<!-- Header -->***

***<header style="display:flex;align-items:center;background:rgba(255,255,255,0.95);padding:15px 20px;box-shadow:0 5px 10px rgba(0,0,0,0.1);border-radius:0 0 15px 15px;">***

***<img src="https://tse3.mm.bing.net/th?id=OIP.ZMmOL1kqTR2H1yYs4ZZKhgHaHc&pid=Api&P=0&h=180" alt="JNTU-GV Logo" style="height:80px;margin-right:15px;">***

***<div style="color:#333;">***

***<h1 style="margin:0;font-size:20px;">JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA VIZIANAGARAM</h1>***

***<p style="margin:0;font-size:14px;">VIZIANAGARAM - 535 003, A.P (Established by Andhra Pradesh Act No.22 of 2021)</p>***

***</div>***

***</header>***

***<!-- Navigation -->***

***<nav style="background:#002855;padding:12px 0;box-shadow:0 4px 6px rgba(0,0,0,0.3);">***

***<ul style="list-style:none;margin:0;padding:0;display:flex;justify-content:center;">***

***<li style="margin:0 25px;"><a href="{% url 'home' %}" style="color:#fff;text-decoration:none;font-size:18px;font-weight:bold;">Home</a></li>***

***<li style="margin:0 25px;"><a href="{% url 'notices' %}" style="color:#fff;text-decoration:none;font-size:18px;font-weight:bold;">Notices</a></li>***

***<li style="margin:0 25px;"><a href="{% url 'about' %}" style="color:#fff;text-decoration:none;font-size:18px;font-weight:bold;">About</a></li>***

***<li style="margin:0 25px;"><a href="{% url 'contact' %}" style="color:#fff;text-decoration:none;font-size:18px;font-weight:bold;">Contact</a></li>***

***<li style="margin:0 25px;"><a href="{% url 'account' %}" style="color:#fff;text-decoration:none;font-size:18px;font-weight:bold;">Account</a></li>***

***</ul>***

***</nav>***

***<!-- Main Layout -->***

***<div style="display:flex;min-height:calc(100vh - 170px);">***

***<!-- Sidebar -->***

***<div class="sidebar" style="width:250px;background:linear-gradient(to bottom,#6a0dad,#8e44ad);padding:20px;display:flex;flex-direction:column;align-items:center;box-shadow:3px 0 10px rgba(0,0,0,0.2);">***

***<button onclick="showSection('exams', this)" class="active" style="width:100%;padding:12px;border:2px solid #fff;background:transparent;color:#fff;font-size:16px;font-weight:bold;margin-bottom:10px;border-radius:5px;cursor:pointer;">Exams</button>***

***<button onclick="showSection('events', this)" style="width:100%;padding:12px;border:2px solid #fff;background:transparent;color:#fff;font-size:16px;font-weight:bold;margin-bottom:10px;border-radius:5px;cursor:pointer;">Events</button>***

***<button onclick="showSection('sports', this)" style="width:100%;padding:12px;border:2px solid #fff;background:transparent;color:#fff;font-size:16px;font-weight:bold;margin-bottom:10px;border-radius:5px;cursor:pointer;">Sports</button>***

***<button onclick="showSection('academic', this)" style="width:100%;padding:12px;border:2px solid #fff;background:transparent;color:#fff;font-size:16px;font-weight:bold;margin-bottom:10px;border-radius:5px;cursor:pointer;">Academic</button>***

***</div>***

***<!-- Content -->***

***<div style="flex:1;padding:20px;background:#f5f5f5;font-family:'Arial',sans-serif;">***

***<div style="text-align:center;font-size:24px;font-weight:bold;color:#5d1a99;margin-bottom:20px;">📢 Notifications | JNTU-GV University</div>***

***<!-- Exams -->***

***<div id="exams" class="notification-container">***

***<h2>📚 Exam Notifications</h2>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">29 <small>Nov 2023</small></span> 📅 Results of I B.Tech I Sem Regular & Supplementary Exams</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">29 <small>Nov 2023</small></span> 📅 Results of II B.Tech II Sem Supplementary Exams</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">05 <small>Mar 2025</small></span> 📝 Special Supplementary Exams Application (IV B. Pharmacy II Sem)</div>***

***</div>***

***<!-- Events -->***

***<div id="events" class="notification-container" style="display:none;">***

***<h2>🎉 Events</h2>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">10 <small>Jan 2025</small></span> 🔥 IT YUKTA 2K25</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">15 <small>Feb 2025</small></span> 🎭 EKTHRA 2K25</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">20 <small>Feb 2025</small></span> 🎉 ECENTRA 2K25</div>***

***</div>***

***<!-- Sports -->***

***<div id="sports" class="notification-container" style="display:none;">***

***<h2>⚽ Sports</h2>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">29 <small>Feb 2025</small></span> 🏐 Volleyball Selected Players List</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">29 <small>Feb 2025</small></span> 🏋️ Weightlifting List</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">29 <small>Feb 2025</small></span> 🧘 Yoga (Men & Women) Selected List</div>***

***</div>***

***<!-- Academic -->***

***<div id="academic" class="notification-container" style="display:none;">***

***<h2>📖 Academic Events</h2>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">15 <small>Mar 2025</small></span> 🛠️ Skill Development Workshop</div>***

***<div><span style="display:inline-block;width:60px;font-weight:bold;">10 <small>Apr 2025</small></span> 📌 One-Day Orientation Program for Freshers</div>***

***</div>***

***</div>***

***</div>***

***<!-- JavaScript -->***

***<script>***

***function showSection(id, btn) {***

***document.querySelectorAll('.notification-container').forEach(section => section.style.display = 'none');***

***document.getElementById(id).style.display = 'block';***

***document.querySelectorAll('.sidebar button').forEach(b => b.classList.remove('active'));***

***btn.classList.add('active');***

***}***

***</script>***

***</body>***

</html>**OUTPUT:-**

A screenshot of a computer

AI-generated content may be incorrect.

**15.Contact Page Templates**

**Explanation of the Contact Us Template (Without Code)**

This **Contact Us** template is designed for the **JNTU-GV Digital Campus Noticeboard** project, allowing users to reach out with queries, feedback, or concerns. Below is a breakdown of its key sections:

**1. Header Section**

1. Displays the **JNTU-GV** logo on the left.
2. Shows the **full name of the university** and its **location** (Vizianagaram, Andhra Pradesh).
3. Uses a **clean and structured layout** with a background shadow effect for clarity.

**2. Navigation Bar**

A **horizontal menu bar** with the following navigation links:

**Home** (Redirects to the homepage).

**Notices** (Displays university-related announcements).

**About** (Provides details about the university).

* + **Contact** (Current page for feedback and inquiries).

**Account** (Allows users to log in or sign up).

Uses a **hover effect** where links change color to yellow when the user interacts.

**3. Contact Form Section**

A **form inside a centered container**, allowing users to submit feedback.

Includes the following fields:

**Name (Optional):** Users can enter their name (not mandatory).

**Email (Required):** Ensures users provide a valid email for follow-ups.

**Subject (Required):** Captures the main topic of the query.

**Message (Required):** A text area where users enter their message.

A **Submit Button** with a green background, which changes color when hovered over.

**4. Footer Section**

1. A **background image** adds a professional look.
2. Displays the **university's full contact details**, including location.
3. **Social Media Links**:

Facebook

LinkedIn

Twitter (X)

Instagram

YouTube

1. A **Privacy & Policy** link for legal information.
2. Copyright notice indicating that **JNTU-GV owns the content**.
3. Mentions **Likhith Kumar as the developer** of the page.

**5. Styling & Design Features**

1. Uses **Poppins font** for a modern and readable look.
2. **Responsive layout**, adjusting elements for different screen sizes.
3. **Hover effects** on navigation links and buttons for interactivity.
4. **Shadow effects** on elements to enhance visibility.

**Overall Purpose**

This **Contact Us page** serves as a user-friendly interface for students, faculty, and visitors to **communicate with JNTU-GV**. The **responsive design, structured layout, and interactive elements** make it easy to use and visually appealing.

**CODE:-**

***<!DOCTYPE html>***

***<html lang="en">***

***<head>***

***<meta charset="UTF-8">***

***<meta name="viewport" content="width=device-width, initial-scale=1.0">***

***<title>Contact Us</title>***

***<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet">***

***<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css">***

***<style>body{font-family:Arial,sans-serif;margin:0;padding:0;background-color:#f4f4f4}header{display:flex;align-items:center;background-color:rgba(255,255,255,0.95);padding:15px 20px;box-shadow:0px 5px 10px rgba(0,0,0,0.1);border-radius:0 0 15px 15px}header img{height:80px;margin-right:15px}header div{color:#333}nav{background-color:#002855;padding:12px 0;box-shadow:0 4px 6px rgba(0,0,0,0.3)}nav ul{list-style:none;margin:0;padding:0;display:flex;justify-content:center}nav ul li{margin:0 25px;position:relative}nav ul li a{text-decoration:none;color:white;font-size:18px;font-weight:bold;transition:color 0.3s;position:relative;padding-bottom:5px}nav ul li a:hover{color:#FFC107}nav ul li a::after{content:"";width:0;height:2px;background-color:#FFC107;position:absolute;bottom:0;left:50%;transform:translateX(-50%);transition:width 0.3s ease-in-out}nav ul li a:hover::after{width:100%}.container{width:50%;margin:50px auto;background:white;padding:20px;box-shadow:0px 0px 10px rgba(0,0,0,0.1);border-radius:8px}h2{text-align:center;color:#333}.form-group{margin-bottom:15px}label{display:block;font-weight:bold}input,textarea{width:100%;padding:10px;margin-top:5px;border:1px solid #ccc;border-radius:5px}button{background:#28a745;color:white;border:none;padding:10px;width:100%;cursor:pointer;border-radius:5px;font-size:16px}button:hover{background:#218838}#footer{background:url('</head>***

***<body>***

***<header>***

***<img src="https://tse3.mm.bing.net/th?id=OIP.ZMmOL1kqTR2H1yYs4ZZKhgHaHc&pid=Api&P=0&h=180" alt="JNTU-GV Logo">***

***<div>***

***<h1>JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA VIZIANAGARAM</h1>***

***<p>VIZIANAGARAM - 535 003, A.P (Established by Andhra Pradesh Act No.22 of 2021)</p>***

***</div>***

***</header>***

***<nav>***

***<ul>***

***<li><a href="{% url 'home' %}">Home</a></li>***

***<li><a href="{% url 'notices' %}">Notices</a></li>***

***<li><a href="{% url 'about' %}">About</a></li>***

***<li><a href="{% url 'contact' %}">Contact</a></li>***

***<li><a href="{% url 'account' %}">Account</a></li>***

***</ul>***

***</nav>***

***<div class="container">***

***<h2>Contact Us</h2>***

***<form action="submit\_feedback.php" method="POST">***

***<div class="form-group">***

***<label for="name">Name (optional):</label>***

***<input type="text" id="name" name="name">***

***</div>***

***<div class="form-group">***

***<label for="email">Email:</label>***

***<input type="email" id="email" name="email" required>***

***</div>***

***<div class="form-group">***

***<label for="subject">Subject:</label>***

***<input type="text" id="subject" name="subject" required>***

***</div>***

***<div class="form-group">***

***<label for="message">Message:</label>***

***<textarea id="message" name="message" rows="5" required></textarea>***

***</div>***

***<button type="submit">Send Feedback</button>***

***</form>***

***</div>***

***<section id="footer">***

***<footer>***

***<div class="footer-content">***

***<p>***

***<strong>Contact Us:</strong> JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY-GURAJADA VIZIANAGARAM,***

***DWARAPUDI, VIZIANAGARAM, ANDHRA PRADESH - 535 003, Andhra Pradesh, INDIA.***

***</p>***

***<div class="social-icons">***

***<a href="https://www.facebook.com/JntuVizianagaram/" target="\_blank">***

***<i class="fab fa-facebook-f"></i>***

***</a>***

***<a href="https://in.linkedin.com/school/jntugv/" target="\_blank">***

***<i class="fab fa-linkedin-in"></i>***

***</a>***

***<a href="https://x.com/JNTU\_Gurajada" target="\_blank">***

***<i class="fab fa-twitter"></i>***

***</a>***

***<a href="https://www.instagram.com/jntu\_gurajada/" target="\_blank">***

***<i class="fab fa-instagram"></i>***

***</a>***

***<a href="https://www.youtube.com/@JNTUGV" target="\_blank">***

***<i class="fab fa-youtube"></i>***

***</a>***

***</div>***

***<a href="#" class="privacy">Privacy & Policy</a>***

***<p>© Copyright 2024 <strong>JNTU-GV Vizianagaram</strong>. All Rights Reserved.</p>***

***<p>Designed and Developed by <strong>M.Likhith Kumar, JNTU-GV</strong></p>***

***</div>***

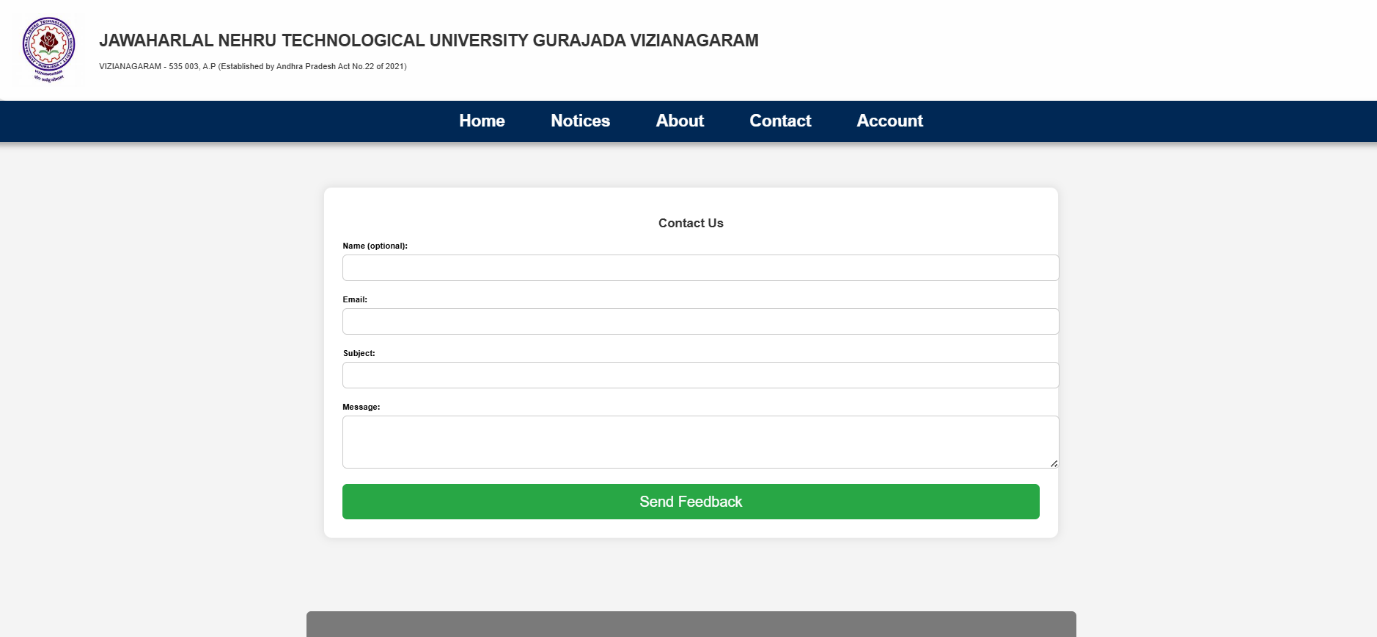
***</footer>***

***</section>***

***</body>***

***</html>***

**OUTPUT:-**



**16.Account Page Templates**

Account Page Template - JNTU-GV Digital Campus Noticeboard

The Account Page in the JNTU-GV Digital Campus Noticeboard project provides a user-friendly interface for students and faculty to log in or sign up. This page is designed with a modern and interactive UI, ensuring seamless user authentication.

Key Features

**1. Header and Navigation Bar**

1. The page includes a header displaying the official JNTU-GV logo and university details.
2. A navigation bar allows users to switch between different sections such as Home, Notices, About, Contact, and Account.

**2. Authentication Section**

1. Users are given two primary options:
   * Login (For existing users)
   * Sign Up (For new users)
2. The interface provides buttons to switch between login and signup forms smoothly.

**3. Login Form**

1. A simple login form where users can enter their username and password.
2. On successful login, the authentication section disappears, simulating a real login experience.

**4. Signup Form**

1. The signup form allows new users to enter their details, including:

Username, Email, Password, Phone Number, and Address

A dropdown selection to choose between Student or Faculty

Additional fields based on the role:

Students: Roll Number and Branch

Faculty: Designation

1. A profile picture upload option is included, with a preview feature that displays the selected image.

**5. Animations and Effects**

1. The page utilizes smooth transitions and animations, including fade-in effects and slide-in forms, to enhance the user experience.
2. The profile picture preview appears dynamically when an image is uploaded.

**6. Footer Section**

1. The footer contains university contact details and social media links (Facebook, LinkedIn, Twitter, Instagram, and YouTube).
2. A privacy policy link is also included.
3. The page credits the developer, M. Likhith Kumar, for designing and developing the project.

**User Experience Enhancements**

1. Responsive Design: The page adapts to different screen sizes, ensuring compatibility with mobile and desktop devices.
2. JavaScript-based Form Toggle: Users can easily switch between login and signup forms without reloading the page.

This Account Page serves as the gateway for students and faculty to access personalized features within the JNTU-GV Digital Campus Noticeboard project

**CODE:-**

***<!DOCTYPE html>***

***<html lang="en">***

***<head>***

***<meta charset="UTF-8">***

***<meta name="viewport" content="width=device-width, initial-scale=1.0">***

***<title>Account Section</title>***

***<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet">***

***</head>***

***<body>***

***<header>***

***<div class="header-content">***

***<img src="https://tse4.mm.bing.net/th?id=OIP.\_KgEoeUCp0qBL0-lpgNXmQHaHi&pid=Api&P=0&h=180" alt="JNTU GV Logo">***

***<div class="header-text">***

***<h2>JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA VIZIANAGARAM</h2>***

***<p>VIZIANAGARAM-535 003, A.P</p>***

***<p>(Established by Andhra Pradesh Act No.22 of 2021)</p>***

***</div>***

***</div>***

***</header>***

***<nav>***

***<a href="{% url 'home' %}">Home</a>***

***<a href="{% url 'internships' %}">Internships</a>***

***<a href="{% url 'about' %}">About</a>***

***<a href="{% url 'account' %}">Account</a>***

***</nav>***

***<section id="account">***

***<h2>Account</h2>***

***<div class="account-form" id="authForm">***

***<input type="file" id="profilePic" accept="image/\*">***

***<input type="text" id="name" placeholder="Full Name (Only for Signup)">***

***<input type="email" id="email" placeholder="Email">***

***<input type="password" id="password" placeholder="Password">***

***<input type="text" id="phone" placeholder="Phone Number (Only for Signup)">***

***<input type="text" id="address" placeholder="Address (Only for Signup)">***

***<button onclick="handleAuth()">Login / Sign Up</button>***

***<p id="toggleText">Don't have an account? <a href="#" onclick="toggleAuthMode()">Sign Up</a></p>***

***</div>***

***<div class="profile" id="userProfile">***

***<img id="profileImage" src="https://via.placeholder.com/120" alt="Profile Picture">***

***<h3 id="userName"></h3>***

***<p><strong>Email:</strong> <span id="userEmail"></span></p>***

***<p><strong>Phone:</strong> <span id="userPhone"></span></p>***

***<p><strong>Address:</strong> <span id="userAddress"></span></p>***

***<button class="logout-btn" onclick="logout()">Logout</button>***

***</div>***

***</section>***

***<script>***

***let isSignUp = false;***

***let users = JSON.parse(localStorage.getItem("users")) || {};***

***function toggleAuthMode() {***

***isSignUp = !isSignUp;***

***document.getElementById("name").style.display = isSignUp ? "block" : "none";***

***document.getElementById("phone").style.display = isSignUp ? "block" : "none";***

***document.getElementById("address").style.display = isSignUp ? "block" : "none";***

***document.getElementById("toggleText").innerHTML = isSignUp ?***

***`Already have an account? <a href="#" onclick="toggleAuthMode()">Login</a>` :***

***`Don't have an account? <a href="#" onclick="toggleAuthMode()">Sign Up</a>`;***

***}***

***function handleAuth() {***

***let email = document.getElementById("email").value;***

***let password = document.getElementById("password").value;***

***if (isSignUp) {***

***let name = document.getElementById("name").value;***

***let phone = document.getElementById("phone").value;***

***let address = document.getElementById("address").value;***

***let profilePic = document.getElementById("profilePic").files[0];***

***if (!name || !email || !password || !phone || !address) {***

***alert("All fields are required!");***

***return;***

***}***

***if (users[email]) {***

***alert("User already exists. Please login.");***

***return;***

***}***

***let reader = new FileReader();***

***reader.onload = function (e) {***

***users[email] = { name, email, password, phone, address, profilePic: e.target.result };***

***localStorage.setItem("users", JSON.stringify(users));***

***alert("Signup successful! Please log in.");***

***toggleAuthMode();***

***};***

***if (profilePic) {***

***reader.readAsDataURL(profilePic);***

***} else {***

***users[email] = { name, email, password, phone, address, profilePic: "https://via.placeholder.com/120" };***

***localStorage.setItem("users", JSON.stringify(users));***

***alert("Signup successful! Please log in.");***

***toggleAuthMode();***

***}***

***} else {***

***if (!users[email] || users[email].password !== password) {***

***alert("Invalid email or password!");***

***return;***

***}***

***localStorage.setItem("loggedInUser", email);***

***showProfile(users[email]);***

***}***

***}***

***function showProfile(user) {***

***document.getElementById("authForm").style.display = "none";***

***document.getElementById("userProfile").style.display = "block";***

***document.getElementById("profileImage").src = user.profilePic;***

***document.getElementById("userName").innerText = user.name;***

***document.getElementById("userEmail").innerText = user.email;***

***document.getElementById("userPhone").innerText = user.phone;***

***document.getElementById("userAddress").innerText = user.address;***

***}***

***function logout() {***

***localStorage.removeItem("loggedInUser");***

***document.getElementById("authForm").style.display = "block";***

***document.getElementById("userProfile").style.display = "none";***

***}***

***window.onload = function () {***

***let loggedInUser = localStorage.getItem("loggedInUser");***

***if (loggedInUser && users[loggedInUser]) {***

***showProfile(users[loggedInUser]);***

***}***

***};***

***</script>***

***</body>***

***</html>***

**OUTPUT:-**

A screenshot of a computer

AI-generated content may be incorrect.

**17.About Page Templates**

**Explanation of the JNTU-GV Leaderboard Template**

This template is designed for the **JNTU-GV Leaderboard** webpage, which showcases key university personnel in an interactive and visually appealing format. The page includes a header, navigation bar, a leaderboard section, and additional details about the university. Below is a breakdown of its key sections:

**1. Header Section**

1. Displays the **JNTU-GV logo** and the **university name**.
2. Provides a **short description** about the university, including its establishment under the **Andhra Pradesh Act No.22 of 2021**.

**2. Main Content - Leaderboard Section**

1. Displays key officials of JNTU-GV, including:

**Chancellor (Governor of Andhra Pradesh)**

**Vice Chancellor**

**Registrar**

**Principal**

**Vice Principal**

1. Each person’s profile includes:

**Profile Picture**

**Name and Designation**

**Brief Description**

**Contact Details (if available)**

1. Clicking on a profile **reveals more details** in an animated dropdown effect.

**Leaderboard Layout**

1. **Top Row**: Displays high-ranking officials (Governor, Vice Chancellor, Registrar).
2. **Bottom Row**: Displays college-level administrators (Principal, Vice Principal).
3. Responsive design ensures proper alignment across different screen sizes.

**3. About JNTU-GV Section**

1. Provides a **brief history and introduction** to JNTU-GV.
2. Includes an **explanation of the university’s focus on engineering education**.

**4. Engineering Departments Section**

1. Describes the **various engineering departments** available at JNTU-GV.
2. Uses **a visually appealing card layout** to display department information.

**5. Styling & Animations**

1. **Gradient Background**: Smooth color transition from purple to blue.
2. **Hover Effects**: Buttons and links change color upon hovering.
3. **Interactive Profiles**: Clicking on a person expands additional details.
4. **Shadow Effects**: Cards and containers have subtle shadows for a modern look.

**6. Mobile Responsiveness**

1. Uses **flexbox and media queries** to ensure the page adapts to smaller screens.
2. **Leaderboard layout changes** on mobile:
   * Instead of a row-based layout, profiles are **stacked vertically**.
   * Ensures readability and usability on mobile devices.

This template effectively presents **key university officials** in an **interactive leaderboard format**, while maintaining a **clean, modern, and responsive design**. The page provides useful information about JNTU-GV and ensures easy navigation for users.

**CODE:-**

***<!DOCTYPE html>***

***<html lang="en">***

***<head>***

***<meta charset="UTF-8" />***

***<meta name="viewport" content="width=device-width, initial-scale=1.0"/>***

***<title>JNTU-GV Leaderboard</title>***

***<!-- Fonts and Icons -->***

***<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;600&display=swap" rel="stylesheet"/>***

***<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css"/>***

***<style>***

***/\* General Styling \*/***

***body { font-family: 'Poppins', sans-serif; margin: 0; padding: 20px; color: white; background: linear-gradient(#6a11cb, #2575fc); text-align: center; overflow-x: hidden; }***

***header, nav, .container, section, footer { margin: auto; max-width: 1200px; }***

***/\* Header \*/***

***header { display: flex; align-items: center; background: #fff; color: #333; padding: 15px 20px; border-radius: 0 0 15px 15px; box-shadow: 0 5px 10px rgba(0,0,0,0.1); }***

***header img { height: 80px; margin-right: 15px; }***

***/\* Navigation \*/***

***nav { background: #002855; box-shadow: 0 4px 6px rgba(0,0,0,0.3); }***

***nav ul { list-style: none; display: flex; justify-content: center; gap: 30px; margin: 0; padding: 10px 0; }***

***nav a { color: #fff; text-decoration: none; font-weight: bold; }***

***/\* Main Container \*/***

***.container { background: #fff; color: #000; padding: 20px; border-radius: 10px; box-shadow: 0 0 15px rgba(0,0,0,0.2); }***

***/\* Leaderboard Cards \*/***

***.leaderboard { display: flex; flex-wrap: wrap; justify-content: center; gap: 30px; margin-top: 20px; }***

***.leader { background: #fff; width: 260px; padding: 15px; border-radius: 20px; box-shadow: 0 4px 8px rgba(0,0,0,0.1); cursor: pointer; }***

***.leader img { width: 100px; height: 100px; object-fit: cover; border-radius: 50%; margin-bottom: 10px; }***

***/\* Section Styling \*/***

***section { margin: 40px auto 0; color: #fff; }***

***section div span { display: inline-block; margin: 5px 8px; }***

***/\* Footer Styling \*/***

***footer { margin-top: 50px; padding: 20px; background: rgba(0, 0, 0, 0.2); }***

***footer a { color: white; margin: 0 8px; font-size: 18px; text-decoration: none; }***

***</style>***

***</head>***

***<body>***

***<!-- University Header -->***

***<header>***

***<img src="https://tse3.mm.bing.net/th?id=OIP.ZMmOL1kqTR2H1yYs4ZZKhgHaHc&pid=Api&P=0&h=180" alt="Logo"/>***

***<div>***

***<h1 style="margin:0;font-size:20px;">JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURAJADA VIZIANAGARAM</h1>***

***<p style="margin:5px 0;">VIZIANAGARAM - 535 003, A.P (Established by Act No.22 of 2021)</p>***

***</div>***

***</header>***

***<!-- Navigation Bar -->***

***<nav>***

***<ul>***

***<li><a href="{% url 'home' %}">Home</a></li>***

***<li><a href="{% url 'notices' %}">Notices</a></li>***

***<li><a href="{% url 'about' %}">About</a></li>***

***<li><a href="{% url 'contact' %}">Contact</a></li>***

***<li><a href="{% url 'account' %}">Account</a></li>***

***</ul>***

***</nav>***

***<!-- Main Content -->***

***<div class="container">***

***<img src="https://jntugv.edu.in/static/media/jntugvcev.b33bb43b07b2037ab043.jpg" style="width:100px;margin-bottom:10px;"/>***

***<h2>Jawaharlal Nehru Technological University - Gurajada Vizianagaram (JNTU-GV)</h2>***

***<p>The Jawaharlal Nehru Technological University-Gurajada, Vizianagaram (JNTU-GV) is a public university...</p>***

***<!-- Leaderboard Section -->***

***<h2 style="margin-top:30px;">Leaderboard</h2>***

***<div class="leaderboard">***

***<div class="leader" onclick="toggleDetails(this)"><img src="..." alt="Chancellor"/><h3>Shri Justice (Retd) Syed Abdul Nazeer</h3><p><strong>Hon'ble Governor of AP</strong></p><div><p>Bio of the Chancellor...</p></div></div>***

***<div class="leader" onclick="toggleDetails(this)"><img src="..." alt="VC"/><h3>Dr D. Rajya Lakshmi</h3><p><strong>Vice Chancellor</strong></p><div><p>Bio of the VC...</p></div></div>***

***<div class="leader" onclick="toggleDetails(this)"><img src="..." alt="Registrar"/><h3>Prof. G.Jaya Suma</h3><p><strong>Registrar</strong></p><div><p>Bio of the Registrar...</p></div></div>***

***<div class="leader" onclick="toggleDetails(this)"><img src="..." alt="Principal"/><h3>Prof. R. Rajeswara Rao</h3><p><strong>Principal i/c</strong></p><div><p>Bio of the Principal...</p></div></div>***

***<div class="leader" onclick="toggleDetails(this)"><img src="..." alt="VP"/><h3>Dr. G. J. Naga Raju</h3><p><strong>Vice Principal</strong></p><div><p>Bio of the VP...</p></div></div>***

***</div>***

***</div>***

***<!-- About Section -->***

***<section>***

***<h2>About JNTU-GV</h2>***

***<p>History and background of the university...</p>***

***</section>***

***<!-- Departments Section -->***

***<section>***

***<h2>Engineering Departments</h2>***

***<p>Explore the various engineering disciplines...</p>***

***<div>***

***<span>Computer Science & Engineering</span>***

***<span>Electronics & Communication Engineering</span>***

***<span>Mechanical Engineering</span>***

***<span>Civil Engineering</span>***

***<span>Electrical & Electronics Engineering</span>***

***</div>***

***</section>***

***<!-- Footer -->***

***<footer>***

***<p><strong>Contact Us:</strong> Address info...</p>***

***<div>***

***<a href="#"><i class="fab fa-facebook-f"></i></a>***

***<a href="#"><i class="fab fa-linkedin-in"></i></a>***

***<a href="#"><i class="fab fa-twitter"></i></a>***

***<a href="#"><i class="fab fa-instagram"></i></a>***

***<a href="#"><i class="fab fa-youtube"></i></a>***

***</div>***

***<a href="#" style="display:block;margin:10px 0;">Privacy & Policy</a>***

***<p>© 2024 <strong>JNTU-GV Vizianagaram</strong>. All Rights Reserved.</p>***

***<p>Designed by <strong>M.Likhith Kumar, JNTU-GV</strong></p>***

***</footer>***

***<!-- Toggle Details Script -->***

***<script>***

***function toggleDetails(el) {***

***document.querySelectorAll('.leader').forEach(l => { if (l !== el) l.classList.remove('active'); });***

***el.classList.toggle('active');***

***}***

***</script>***

***</body>***

***</html>***

**OUTPUT:-**

A screenshot of a computer

AI-generated content may be incorrect.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Key Django Files

7. Date of Experiment : 07-03-2025

8. Date of Submission of Report :07-03-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**18.Explanation of Key Django Files**

1. **db.sqlite3 (Database File)**

This is the **default database file** used by Django when SQLite is selected.

It stores all the **data related to users, internships, applications, and other models**.

The database is automatically created when running migrations.

1. **admin.py (Admin Panel Configuration)**

This file allows **customization of the Django Admin interface**.

It is used to **register models** so they can be managed through the admin dashboard.

Admins can **add, edit, delete, and view records** from the database via the Django Admin panel.

1. **apps.py (App Configuration)**

This file contains the **configuration for the Django app** (like base).

It helps Django recognize the app within the project.

The class in this file is automatically created when an app is generated.

1. **models.py (Database Models)**

Defines the **structure of database tables** using Django’s ORM (Object-Relational Mapping).

Each model represents a **table**, and its attributes represent **columns**.

Django automatically converts models into database tables when migrations are applied.

1. **forms.py (Forms Handling)**

This file is used to create **HTML forms using Django’s Form API**.

Forms in Django handle **user input, validation, and data submission**.

They help in managing sign-ups, logins, and internship applications efficiently.

1. **static/ (Static Files)**

This directory stores **CSS, JavaScript, and images** used in the web pages.

It ensures styling and interactivity for the website.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Database In Djnago

7. Date of Experiment : 07-03-2025

8. Date of Submission of Report :24-03-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**19.Database in Django(sqlite3)**

**Database in Django**

In **Django**, the database is an essential component used to store, manage, and retrieve data using Django's **Object-Relational Mapping (ORM)**. ORM acts as a bridge between the application and the database, allowing you to interact with the database using Python code instead of raw SQL.

Django supports several databases like:

* **SQLite** (default for development)

**Database for JNTU-GV Digital Campus Noticeboard**

You used **SQLite** as the database for both projects, which is an excellent choice for development and small to medium-scale applications. SQLite is a lightweight, file-based database that doesn’t require a separate database server. It offers simplicity and reliability.

**Database Structure Overview**

**1. Digital Campus Noticeboard Database**

**Tables and Models:**

1. **Notice:** Stores all college announcements and notices.
2. **Category:** Categorizes notices (e.g., Academic, Events, Exams).
3. **User:** Handles student and faculty authentication.
4. **Notification:** Tracks notifications sent to users.

**Setting Up the Database**

1. By default, Django uses **SQLite** (a lightweight, file-based database).
2. You can configure other databases like **MySQL**, **PostgreSQL**, or **Oracle** in the settings.py file.

**CODE:-**

***# DATABASE configuration using SQLite3 (for lightweight development use)***

***DATABASES = {***

***'default': {***

***'ENGINE': 'django.db.backends.sqlite3', # Use SQLite as the database engine***

***'NAME': BASE\_DIR / 'dq.sqlite', # Database file name (stored in project directory)***

***}***

***}***

***from django.db import models***

***from django.contrib.auth.models import User # Django's built-in User model for authentication***

***class Category(models.Model):***

***name = models.CharField(max\_length=50) # Category name field (max 50 characters)***

***def \_\_str\_\_(self):***

***return self.name # Display category name in admin/interface***

***class Notice(models.Model):***

***title = models.CharField(max\_length=200) # Title of the notice***

***content = models.TextField() # Detailed content/body of the notice***

***category = models.ForeignKey(Category, on\_delete=models.CASCADE) # Link to a Category; delete notice if category is deleted***

***posted\_by = models.ForeignKey(User, on\_delete=models.CASCADE) # Link to the User who posted it; delete if user is deleted***

***posted\_at = models.DateTimeField(auto\_now\_add=True) # Automatically add timestamp when notice is created***

***def \_\_str\_\_(self):***

***return self.title # Display title in admin or string representation***



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Deploying Django Applications

7. Date of Experiment : 21-03-2025

8. Date of Submission of Report :27-03-2025

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**20.Deploying Django Web Application on Cloud**

**What is Deployment?**

Deployment is the process of making a **Django web application live on the internet** so users can access it. This involves hosting your app on a **cloud server** like **AWS, Googl**

**e Cloud, DigitalOcean, Heroku, or PythonAnywhere**.

**Features**

**Scalability** – Handle more users without performance issues.  
 **Security** – Protect user data with SSL and secure databases.  
 **Global Accessibility** – Users can access your app from anywhere.  
 **Continuous Deployment** – Easily update your app with new features.

Here's a step-by-step guide to **Register on GitHub**, **Create a Django website with login and registration** pages, and **Configure Django to handle static files**.

**Step 1: Register on GitHub**

1. Go to [GitHub](https://github.com/) and click **Sign up**.
2. Enter your **Username, Email, and Password**.
3. Complete the verification and click **Create Account**.
4. Verify your email by clicking the link in your inbox.

**Step 2: Push to GitHub**

1. Initialize Git in your project:

***git init***

2.Connect to GitHub:

***git remote add origin*** [***https://github.com/your-username/django-login-app***](https://github.com/your-username/django-login-app)***.***

3.Add and commit changes:

***gitgit add .***

***git commit -m "Initial Commit: Login and Registration App"***

4.Push to GitHub:

***git branch -M main***

***git push -u origin main***

You have successfully built a **Django website** with login, registration, and static file management.

Your code is now available on **GitHub**.

**21.CONCLUSION**

The **JNTU-GV Digital Campus Noticeboard** and **Internship Portal** projects have been successfully developed using Django, HTML, CSS, JavaScript, and SQLite. These platforms provide an efficient and user-friendly way for students and faculty to access important information, including college notices, internship opportunities, and account management.

The **Digital Campus Noticeboard** ensures that students and faculty stay updated with key announcements, academic notifications, and other essential communications. **Scalability** – Designed to support future enhancements and additional features.

In conclusion, these platforms enhance the digital experience at JNTU-GV, making information dissemination and internship applications more accessible and efficient. Future improvements may include additional features like push notifications, AI-powered recommendations, and more advanced security measures to further improve the system.

**22.FUTURE ENHANCEMENTS**

The JNTU-GV Digital Campus Noticeboard and Internship Portal have been successfully developed with essential features, but there is always room for improvement and expansion. Below are some potential future enhancements to further optimize the platforms:

**1. Advanced Notification System**

1. Implement push notifications for real-time alerts on new notices and internship opportunities.
2. Allow email and SMS alerts for important updates, deadlines, and announcements.

**2. AI-Powered Features**

1. Use AI-based recommendations to suggest relevant internships based on a student’s skills and academic background.
2. Implement a chatbot assistant to help users navigate the portal and answer common queries.

**3. Internship Application Tracking**

1. Introduce a dashboard for students to track their internship applications, status updates, and feedback.
2. Provide faculty access to monitor student applications and guide them accordingly.

**4. Role-Based Access & Security Enhancements**

1. Implement multi-level authentication to enhance security for students, faculty, and admins.
2. Enable admin controls to manage notices, internship postings, and user permissions efficiently.

**5. Internship Certificate & Evaluation System**

1. Allow students to receive digital certificates upon completing internships.
2. Implement a performance evaluation system where mentors can provide feedback and ratings.

**6. Mobile App Development**

1. Develop a mobile version of the noticeboard and internship portal for seamless access.
2. Integrate mobile-friendly UI/UX for better navigation on smartphones and tablets.

**7. Enhanced UI/UX Design**

1. Improve the interface with modern design elements and a more interactive user experience.
2. Add dark mode and customizable themes for user preference.

**8. Internship Collaboration with Companies**

1. Partner with more companies to provide a wider range of internship opportunities.
2. Create a company portal where recruiters can directly post internships and review student applications.

**9. Community & Discussion Forums**

1. Introduce a discussion forum where students can share internship experiences and tips.

**10. Integration with Learning Platforms**

1. Connect the portal with e-learning platforms like Coursera, Udemy, and NPTEL for skill development.
2. Provide access to training resources to prepare students for internships.

By implementing these enhancements, the JNTU-GV Digital Campus Noticeboard and Internship Portal can become more efficient, interactive, and beneficial for students and faculty. These future improvements will ensure that the platforms remain scalable, secure, and valuable for the academic community.

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**JNTU-GURAJADA VIZIANAGARAM**

**COLLEGE OF ENGINEERING VIZIANAGARAM (A)**

**VIZIANAGARAM**

**Dr.Ch. Bindu Madhuri**

Asst. Professor & HOD Email: hod.it@intugvcev.edu.in

1. Name of the Laboratory : Django Framework Lab

2. Name of the Student : M.Likhith Kumar

3. Roll No : 23VV1A1232

4. Class : 2ND B.Tech 2ND Semester

5. Academic Year : 2024-25

6. Name of Experiment : Front-end Developer Certificate

7. Date of Experiment : 04-03-2025

8. Date of Submission of Report :

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Ability And Activity** | **Weightage Of**  **Marks** | **Day To Day**  **Evalution**  **Score** |
| 1 | Aim Objective, Tools required | 3 |  |
| 2 | Theory, Algorithm and Observations | 3 |  |
| 3 | Implementation | 3 |  |
| 4 | Schematic diagrams,Architecture, workflow,Flowchart | 3 |  |
| 5 | Tidiness of his/her working area, proper maintenance of system during and after experiment. | 3 |  |
|  | Total Score | 15 |  |

DATE:

Signature of Faculty:

**Front-End Developer Certificate By Infosys Springboard:-**

