COMSATS University Islamabad

## Abbottabad Campus

**Assignment no 02**

**Submitted by:**

**Syed Hussain Mustafa (SP22-BSE-159)**

**Muhammad Bilal (FA20-BSE-071)**

**Subject:**

**SDA**

**Submitted to:**

**Sir Mukhtiar Zamin**

**Date: 28/12/2024**

**Framework: Django:**

**1. Evolution of Django’s Architecture**

Below is a detailed history of Django versions with major features, release notes, and architectural diagrams.

**Version 1.0 (September 2008)**

* **Major Features**:
  + Officially marked as the first production-ready version.
  + Introduced the Model-View-Template (MVT) architecture.
  + Robust ORM (Object-Relational Mapping) for database abstraction.
  + Built-in template engine for rendering dynamic web pages.
* **Architectural Diagram:**  
  Django’s Model-View-Template (MVT) structure:

rust

Copy code

User <--> View <--> Model <--> Database

|

--> Template

The **Model** handles data and database interactions, the **View** processes requests and fetches data, and the **Template** renders HTML for users.

* **Release Notes**:
  + Focused on stability and backward compatibility.
  + Built-in admin interface for managing data.
  + URL routing system with regular expressions.

**Version 1.8 (April 2015)**

* **Major Features**:
  + Long-Term Support (LTS) version.
  + Native support for PostgreSQL-specific features (JSONField, HStoreField).
  + Enhanced QuerySet API for complex database queries.
  + Template-based rendering system improvements.
* **Architectural Enhancements:** Middleware and modular design were improved for scalability.
* **Release Notes**:
  + Improved form handling and performance.
  + Allowed seamless integration with third-party libraries.

**Version 2.0 (December 2017)**

* **Major Features**:
  + Dropped Python 2 support (fully Python 3).
  + Simplified URL routing with the path() function.
  + Support for JSON-based rendering.
  + Enhanced async capabilities for modern web apps.
* **Architectural Diagram:**

rust

Copy code

User <--> Async Views <--> Model <--> Database

|

--> Template

Django adopted asynchronous processing in views, enabling real-time application performance.

* **Release Notes:**
  + Focused on compatibility with modern web standards.
  + Improved the ORM for handling queries.

**Version 4.0 (December 2021)**

* **Major Features**:
  + Full support for async views, middleware, and database queries.
  + Introduction of time zone handling at the ORM level.
  + Simplified custom model fields creation.
* **Architectural Enhancements**: Real-time and async capabilities were enhanced for high-performance systems.
* **Release Notes**:
  + Improved scalability for large-scale web applications.
  + Added support for type hinting.

**Version 4.2 (April 2023) (Current Version)**

* **Major Features:**
  + Better CSRF handling and secure form rendering.
  + Form Rendering API for flexibility in web forms.
  + Optimized ORM queries for better performance.
* **Architectural Diagram**:

rust

Copy code

User <--> Async Views <--> ORM <--> Database

|

--> Template

--> Middleware

Enhanced middleware system for security, caching, and other processes.

* **Release Notes**:
  + Focused on developer productivity and security.
  + Continued improvements in async support and ORM performance.

**2. Contributions Section**

Since you have two members in your group, the contributions can be divided as follows:

* **Member 1**: Research and document versions 1.0 to 2.0. Create architectural diagrams for these versions.
* **Member 2**: Research and document versions 2.0 to 4.2. Create architectural diagrams for these versions.

**Architectural Diagrams:**

I’ll generate detailed architecture diagrams for Django’s major versions and share them shortly. Let me start working on them!



Here’s the architectural diagram for **Django 1.0's MVT framework**. It represents how users interact with the Model, View, and Template components, along with their connection to the database.

I’ll proceed with additional diagrams for other versions and finalize the details for the report. Let me know if you’d like any modifications!