ABOUT DJANGO

Jalolov Stand up Sadriddinovich
Assistant of Asian International University
Odilova Fatima Altiboyevna
Asia international University student

Anatasia. *Django is a powerful Python web framework designed to make fast and usable web applications. This material article of D go takes its architecture, outputs and components. Special attention is paid to MVC (Model-View-Controller) model, routing system, transformation and control management. Conclusion Django's book and experience speaks volumes for its popularity in modern web programming.*

ENTER.

Django is a high-quality Python web framework designed to accelerate the development of web applications. With a 'batteries in' production philosophy, it provides developers with many plug-ins and products to get production-ready, scalable and reliable web applications. Djan supports fast loading by using modern technologies such as automatic clear code generation and component recovery and minimizing manual labor.

Django's main goal is to help developers focus on the business logic of an add-on, not on the fabric. It has become one of the most popular for getting simple and complex web applications for production, with many installations such as ORM (Object-Relational Manufacturing), authentication and management system.

the main part

1. Django architecture

Django defines a Model-View-Template (MVT) architectural pattern similar to the popular MVC (Model-View-Controller) pattern. Features of MVT:

Components responsible for working with model data and interacting with the database. Django is a transformation to automate database query processing from an ORM that allows developers to work with data through Python classes.

The view contains the business logic of the application and tells the user what data to download and how to process it. In Django, a view returns an HTTP response containing data in various formats (HTML, JSON, etc.).

Template - Mandatory for displaying data in the form of HTML pages. Django relies on its templating system to make it easy to display this logic and data.

2. Django core components

2.1. Django ORM

One of the main features of Django is its ORM (Object-Relational Mapping) - this system works with databases through Python objects, without using SQL queries directly. ORM makes SQL queries based on automatic models (classes), simplifying programs and data management, working with databases more intuitive and productive.

An example of a simple model in Django:

python

```
import django.db
classbook(models.Model):
title = models.CharField(max_length=200)
author = models.CharField(max_length=100)
published_date = models.DateField()
```

moreover, it is possible to interact with the database through methods such as .save(), .filter(), .all() and others without typing SQL queries manually.

2.2. Routing and URL configuration

Routing in Django is controlled by `urls.py` files, which clear which views block requests for which URLs. This is to make it easier to associate URLs with specific functions or view classes.

An example of configuring a URL:

python

```
from django.urls import path
from import views
urlpatterns = [
path(books/, views.book_list, name=book_list),
path(books/<int:id>/, views.book_detail, name=book_detail),
]
```

2.3.Django template system

DjangoBlons support tags and filters to modify HTML content and dynamically display data. The main advantage of templates is the simplicity of editing: developers can separate the logic from the display and write and manage the pages.

Example of a simple template:

html

```
<!DOCTYPE html>
<html>
<head>
<title>List of books</title>
</head>
<body>
<h1>Books</h1>
<ul>
{% for books in books %}
<li>{{ book.title }} by {{ book.author }}</li>
```

```
{% for end %}
</ul>
</body>
</html>
```

2.4. In relation to himself

Django pays a lot of attention. It has defense mechanisms against common attacks, such as:

Cross-Site Scripting (XSS): Templates provide data protection by default to protect against malicious scripting.

Cross-Site Request Forgery (CSRF): protection against request forgery.

SQL Injection: Gets SQL injection by autoloading from the ORM.

Password Encrypt: Configure Django user passwords from hashing methods to be removed for protection.

2.5. Django control panel

One of Django's most attractive features is its automated control panel. This panel provides a convenient interface for support and data management based on practical models. It performs CRUD (create, read, update, delete) operations without requiring any additional code.

An example of adding a model to the admin panel:

python

```
from django.contrib import admin
from .models import Book
admin.site.register(Book)
```

3. Django's and damage

Advantages:

Fast production release: Django allows you to release your applications in production thanks to many release automation tools.

- justification: Django provides great things to protect applications from public access.
- Scalability: Django supports both small downloads and complex enterprise applications.
- Community and documentation: a large selection of resources, documentation and an active community.

Disadvantages:

- Compound for minor use: D can be added for minor additions.
- Slow version updates: Although the framework is maintained, updating to newer versions may require additional effort.

Summary.

Django is a powerful and easy-to-use tool for web developers, suitable for both simple websites and complex enterprise systems. It allows you to speed up the production process with the automation of many cutting tools, while allowing for high-speed control and scalability. Django continues to be active and remains a popular choice among web developers thanks to its community. for smaller, lighter frameworks like Flask may be preferable. benefit, Django is getting excellent support for developing applications with a focus on benefit, reliability and usability.

USED LITERATURE LIST :

1. Sadriddinovich, J. T. (2024). BASICS OF PSYCHOLOGICAL SERVICE. PSIXOLOGIYA VA SOTSIOLOGIYA ILMIY JURNALI, 2(4), 61-67.
2. Jalolov, T. S. (2024). PYTHONNING MATEMATIK KUTUBXONALARINI O'RGANISH: KENG QAMROVLI QO'LLANMA. BIOLOGIYA VA KIMYO FANLARI ILMIY JURNALI, 2(5), 71-77.
3. Django rasmiy hujjatlari. https://www.djangoproject.com/
4. Yangi boshlanuvchilar uchun Django, Uilyam S. Vinsent.
5. Djangoning ikki qoshig'i, Audrey Roy Grinfeld, Daniel Roy Grinfeld.
6. Django REST Framework Documentation. https://www.django-rest-framework.org/