

Date: 29th April-2025

DJANGO - A FRAMEWORK FOR CREATING MODERN WEB APPLICATIONS

Jalolov Tursunbek Sadriddinovich

Associate Professor, Asian International University, PhD

Maxdiyeva Manzura Muxiddinovna,

student at Asian International University

Annotation: This article provides detailed information about the Django framework, one of the modern web programming tools. Django is an open source framework written in the Python programming language, designed for developing fast, secure and scalable web applications. The article analyzes Django's architecture, main components, advantages, security system, real-life applications, modules, admin panel, URL routing system, templates and database system. The process of creating an application using Django is also considered in a practical way. This article serves as an important theoretical and practical guide for students, teachers and web developers.

Login

Today, web technologies are developing day by day. Web applications are widely used in almost all aspects of human life, including education, medicine, trade, service and other areas. Modern, convenient and effective programming frameworks play an important role in creating these systems. The Django framework is one of such tools, it was developed by Adrian Holovaty and Simon Willison in 2005 and is still one of the most popular web frameworks today. Django is based on the Python programming language and is distinguished by its simplicity, extended capabilities, security and fast performance. Django is mainly based on the DRY (Don't Repeat Yourself) and MTV (Model-Template-View) architectural principles. It allows you to develop, test and configure complex systems in a short time.

Main part

1. Django architecture and principles.

Django is based on the MTV (Model-Template-View) architecture. This architecture consists of the following components: - Model: The layer that relates to the database. It allows you to work with the database without writing SQL queries through ORM (Object Relational Mapping). - Template: Responsible for creating HTML pages and displaying them dynamically.

- View: Accepts user requests, retrieves the necessary data, and displays the required HTML pages.

2. Django main components

- URLconf: Django redirects URL requests sent by the user to the appropriate View function. - ORM: Using the Django ORM system, you can work directly with the database through Python. - Admin panel: Django automatically creates an admin interface based on the database. - Forms: Django makes it easy to validate and store user-entered data through



Date: 29th April-2025

forms. - Security mechanisms: Django is protected against CSRF, XSS, SQL injection and other security threats.

3. Advantages of Django

- Speed: Django allows you to quickly create an application and run it in a short time.
- Simplicity: Django's syntax is easy to understand because it is based on Python .
- Security: There are built-in security modules.
- Extensibility: Django's capabilities can be extended through modules.
- Large community and documentation: Django's documentation is very rich and complete .

4. Steps to create an application using Django

Install Django : pip install django

2. Create a project: django-admin startproject myproject 3. Create an application: python manage.py startapp blog 4. Create and migrate models 5. Configure the admin panel 6. Configure URL routing 7. Create templates and views 8. Run the server: python manage.py runserver

5. Real applications and Django - Instagram- Disqus- Pinterest-

Some modules of Mozilla Firefox **6. Comparison of Django and other frameworks**

| Framework | Language | Strengths |
|---------------|------------|-------------------------------|
| Django | Python | Speed, security, admin panel |
| Laravel | PHP | Blade templates, Artisan CLI |
| Express.js | JavaScript | Flexibility, Adaptability |
| Ruby on Rails | Ruby | Convention over configuration |

Conclusion

is one of the most powerful, reliable, and widely used frameworks in web development today . It allows developers to create functional web applications in a simple and fast way . Django's security, admin interface, advanced features, and large community have made it very popular among developers. With Django, you can quickly, securely, and easily create systems of any complexity. This makes it a relevant tool in today's digital age.

REFERENCES:

1. Django Project Documentation: <https://docs.djangoproject.com/>
2. A. Holovaty, J. Kaplan-Moss. The Django Book. Version 2.0.3. Python official documentation: <https://docs.python.org/>
- 3/4. W. Vincent. Django for Beginners: Build websites with Python and Django.
5. RealPython.com Articles
6. GitHub: Django open source repository
7. StackOverflow: Django development FAQ



Date: 29th April-2025

8. DjangoGirls tutorial: <https://tutorial.djangogirls.org/>
9. MDN Web Docs: Web development with Django
10. Online articles and blog posts in Uzbek

