

DOI <https://doi.org/10.30525/978-9934-26-277-7-220>

USING GINJA TEMPLATES TO CREATE E-COMMERCE PLATFORMS

ELEKTRON TIJORAT PLATFORMALARINI YARATISHDA JINJA SHABLONLARIDAN FOYDALANISH

Soliev B. N.

*Assistant
Fergana branch of the Tashkent University
of Information Technologies named after
Muhammad Musa al-Khorazmi
Fergana, Uzbekistan,
Independent researcher-PhD*

Soliyev B. N.

*Muhammad Muso al-Xorazmiy nomidagi
Toshkent axborot texnologiyalari
universiteti Farg'ona filiali assistenti
Farg'ona, O'zbekiston,
Mustaqil izlanuvchi-PhD*

Abdurasulova D. B. kizi

*Assistant
Fergana branch of the Tashkent University
of Information Technologies named after
Muhammad Musa al-Khorazmi
Farg'ona, O'zbekiston,
Mustaqil izlanuvchi-PhD*

Abdurasulova D. B. kizi

*Muhammad Muso al-Xorazmiy nomidagi
Toshkent axborot texnologiyalari
universiteti
Farg'ona filiali assistenti
Farg'ona, O'zbekiston,
Mustaqil izlanuvchi-PhD*

Scientific supervisor: **Yakubov M. S.**

*Doctor of Technical Sciences, Academician
of International Informatization Sciences,
Professor at the Department
of Information Technologies
Tashkent University of Information
Technologies named
after Muhammad Musa al-Khorazmi
Fergana, Uzbekistan*

Ilmiy rahbar: **Yakubov M. S.**

*Muhammad Muso al-Xorazmiy nomidagi
Toshkent axborot texnologiyalari
universiteti "Axborot texnologiyalari"
kafedrası professori, texnika fanlari
doktori, xalqaro axborotlashtirish
fanlari akademigi*

Bugungi kun ommaviy axborot vositalariga nazar solinsa, mamlakatimizdagi axborot-kommunikatsiya texnologiyalari sohasidagi yutuqlar haqida ko'plab maqolalarni ko'rish mumkin. Mamlakatimizda mahalliy va xalqaro internet kanallarining axborotni o'tkazuvchanlik darajasi oshib borayotgani axborot va kommunikatsiya texnologiyalari asosida elektron tijorat tizimini joriy qilishga ham zamin yaratmoqda. Xususan, elektron hujjat aylanishi tizimlari joriy etilmoqda, elektron to'lovlar rivojlantirilmoqda va elektron tijorat sohasida yaratilgan normativ-

huquqiy baza takomillashtirilmoqda. Respublikamizda O‘zbekiston Respublikasining "Elektron tijorat to‘g‘risida" gi qonunini amalga oshirish maqsadida zamonaviy axborot-kommunikatsiya texnologiyalaridan keng foydalangan holda elektron tijoratni yanada rivojlantirish bo‘yicha keng ko‘lamli chora-tadbirlar amalga oshirildi. Elektron tijorat yo‘nalishidagi qonunlarning qabul qilinishi 2004 yilga borib taqaladi. «Elektron tijorat to‘g‘risida»gi[1] O‘zbekiston Respublikasi Qonunini amalga kiritish haqida O‘zbekiston Respublikasi Oliy Majlisining Qarori 614-II-son bilan 29.04.2004 sanasida imzolangan. Keyinchalik 2004 yilda qabul qilingan qonunga o‘zgartirishlar kiritilgan. “«Elektron tijorat to‘g‘risida»gi O‘zbekiston Respublikasi Qonuniga o‘zgartirish va qo‘shimchalar kiritish haqida” [2] O‘zbekiston Respublikasining Qonuni qabul qilindi [3].

Elektron tijorat veb sahifalarini turli yo‘nalishlarda yaratish mumkin. Tayyor platformalardan foydalanib yaratish mumkin. WordPress platformasida WooCommerce plagini bilan, Joomla platformasida VirtueMarket platformasi bilan, OpenCart platformasi, OcStore platformasi, 1cBitrix va boshqa ko‘plab tayyor yechimlarni misol qilib keltirish mumkin. Ushbu platformalarning afzalliklari shundaki, veb dasturlash bo‘yicha tajribasiz foydalanuvchilar ortiqcha qiyinchiliksiz o‘z onlayn magazinlarini yaratib olishlari mumkin bo‘ladi.

Yaratiluvchi onlayn magazin professional bo‘lishi talab etilganda, veb dasturlashga murojaat qilish kerak bo‘ladi. Shulardan biri sifatida Python dasturlash tilida ishlovchi Django freymvorkini misol qilib keltirish mumkin.

Django – bu Python dasturlash tilidagi veb ilovalar uchun erkin veb freymvork hisoblanadi. Loyihalash MVC shabloni asosida yaratiladi. Ushbu loyiha Django Software Foundation tomonidan qo‘llab-quvvatlanadi. Uning dastlabki versiyasi 2005 yil 21 iyulda taqdim etilgan.

Django freymvorkida yaratilgan sayt bir necha ilovalar ko‘rinishida yaratilib, ulardan uzib-ulab foydalanish mumkin. Ushbu holat uni boshqa freymvorlardan farqini ko‘rsatadi [4].

Ushbu freymvorkning yaratuvchilari sifatida Russell Keyt-Magi (Russell Keith-Magee), Adrian Golovaty (Adrian Holovaty), Saymon Villison (Simon Willison), Djeykob Kaplan-Moss (Jacob Kaplan-Moss), Uilson Mayner (Wilson Miner) lar tilga olinadi.

Ushbu freymvorkda “Hello World” ni chiqarish uchun quyidagi misol keltirilgan:

```
1 from django.http import HttpResponse
2
3 def hello(request):
4     return HttpResponse("Hello world")
```

Rasm 1. Django freymvorkida “Hello World” ni chiqarish

Django freymvorki keng tarqalgan bo‘lib, unda ko‘plab yo‘nalishda saytlar yasash mumkin. Itproger.com saytida keltirilgan ma‘lumotda aytilishicha, quyidagi keltirilgan veb saytlar aynan Python dasturlash tili va Django freymvorkidan foydalanilib yaratilgan. YouTube (Google kompaniyasiga tegishli dunyodagi yirik videoxosting hisoblanadi. Ushbu saytning ommabopligini ortiqcha maqtashga hojat yo‘q deyish mumkin.), Google Search (Google kompaniyasi anchadan beri o‘z mahsulotlarida Python dasturlash tilidan foydalanib keladi. Ushbu tilning imkoniyatlarini yaxshi tushunib olishgan va imkon qadar undan ko‘proq foydalanishga harakat qilishadi. O‘z mahsulotlarida Python dasturlash tili va Django freymvorkidan foydalanishadi.), DropBox (Dunyodagi yirik bulutli platforma hisoblanadi. U yerda istalgan ma‘lumotni saqlash, kompyuter va boshqa mobil qurilmalar bilan sinxronizatsiya qilish mumkin.), Instagram (Dunyodagi yirik ijtimoiy tarmoq hisoblanadi. Foydalanuvchilar o‘z fotografiyalari, videolari va boshqa ma‘lumotlari bilan bo‘lishishlari mumkin, layk va boshqa reytinglarda ishtirok etishlari mumkin.), Reddit (Reddit – bu yirik ijtimoiy yangiliklar loyihasi sayti hisoblanadi. Foydalanuvchilar turli mavzularda maqolalar joylashlari, ular uchun izohlar qoldirishi, ovoz berishlari mumkin bo‘lgan sayt), Pinterest (Pinterest resursi instagram ijtimoiy tarmog‘iga o‘xshab ketadi. Bu yerda foydalanuvchilar turli fotografiyalari bilan o‘rtoqlashadilar. Faqatgina bu yerda foydalanuvchi o‘zi qiziqtirgan rasmlarini osonlik bilan topishi mumkin), Quora (Quora sayti “Отеты Mail.ru” saytiga o‘xshash bo‘lib, foydalanuvchilar o‘zlari qiziqtirgan ko‘plab savollariga javoblar topishlari mumkin.), Yahoo Maps (Yahoo o‘z platformasida asosan Node.js dan foydalangan. Lekin xaritalar bilan ishlashda Python tilidan ham foydalanib ko‘rishgan. MDH davlatlarida bu onlayn xarita unchalik ommabop emas), Spotify (musiqalarni bepul topish va eshitish), Onion (satirik gazeta veb sayti) kabilar.

Ushbu freymvork asosida veb sayt yaratishda nimalarni bilish talab etiladi? Ushbu freymvork Python tilida ishlashini e‘tiborga olsak,

o‘z-o‘zidan Python dasturlash tilini bilish kerak bo‘ladi. Shu bilan birga, ushbu freymvork veb saytlar yaratishga mo‘ljallanganini hisobga olinsa, minimal darajada – HTML va CSS bilimlarini egallagan bo‘lish, qo‘shimchasiga esa – JavaScript va PHP bilish foydadan holi bo‘lmaydi.

Veb sahifalarda Django freymvorkining interaktiv ma‘lumotlarini HTML kodlari orasida aks ettirish uchun esa maxsus kodlardan foydalaniladi. Jinja deb nomlanuvchi maxsus kodlar orqali shartlarni tekshirish, massivlarni sikl orqali chiqarish, o‘zgaruvchilarni chop etish va boshqa ma‘lumotlar bilan bevosita interaktiv kontent yaratish mumkin bo‘ladi.

Jinja – Python dasturlash tili uchun shablonizator hisoblanadi. Bu Django shablonizatori kabi bo‘lib, Python ifodalarni bajarilishi uchun muhit bo‘lib xizmat qiladi. Bu matnli shablonizator hisoblanadi. Shuning uchun istalgan ko‘rinishdagi belgilashlarni yaratish imkonini beradi. BSD litsenziyasi bilan litsenziyalangan [6]. Jinja da ifodalar ikkitadan figurali qavs ichida beriladi. Masalan:

```
>>> from jinja2 import Template
>>>
>>> Template("{ { 10 + 3 } }").render()
'13'
>>> Template("{ { 10 - 3 } }").render()
'7'
>>>
>>> Template("{ { 10 // 3 } }").render()
'3'
>>> Template("{ { 10 / 3 } }").render()
'3.3333333333333335'
>>>
>>> Template("{ { 10 % 3 } }").render()
'1'
>>> Template("{ { 10 ** 3 } }").render()
'1000'
>>>
```

Ushbu muhitda o‘zgaruvchilarni chop etish mumkin. Quyidagi misolda oddiy o‘zgaruvchilarni chop etish ko‘rsatilgan:

```
>>> Template("{ { var } }").render(var=12)
'12'
>>> Template("{ { var } }").render(var="hello")
'hello'
>>>
```

Bundan tashqari ro'yxatlar, lug'atlar, kortejlar va foydalanuvchi klasslarini ham chop etish mumkin [7]:

```
>>> Template("{ { var[1] } }").render(var=[1,2,3])
'2'
>>> Template("{ { var['profession'] } }").render(var={'name': 'tom',
'age': 25, 'profession': 'Manager' })
'Manager'
>>> Template("{ { var[2] } }").render(var=("c", "c++", "python"))
'python'
>>> class Foo:
... def __str__(self):
... return "This is an instance of Foo class"
...
>>> Template("{ { var } }").render(var=Foo())
'This is an instance of Foo class'
>>>
```

Jinja da funksiya yaratib, undan foydalanish mumkin. Funksiyadan foydalanish uchun uni chaqirish kerak bo'ladi:

```
>>> def foo():
... return "foo() called"
...
>>>
>>> Template("{ { foo() } }").render(foo=foo)
'foo() called'
>>>
```

Avvallari ma'lumotni chop etish uchun HttpResponse dan foydalanilgan. Ushbu yondashuv noto'g'ri hisoblanadi. Sababi, ko'plab HTML teglarida joylashgan ma'lumotlar bir satr orqali uzatiladi. Buning o'rniga esa render funksiyasidan foydalanish tavsiya qilinadi. Ushbu funksiya ko'plab va yirik HTML-shablonlarni Jinja shablonizatori orqali ixchamlab uzatish imkoniyatiga ega. HTML shablonlarni chaqirishda parametr sifatida request majburiy parametri va shablonli fayl joylashgan joyi ko'rsatiladi [8].

Shablon yaratish. Barcha shablonlar templates umumiy papkasida joylashgan bo'ladi. Django avtomatik ravishda barcha papkalarni bitta yagona templates papkasiga birlashtiradi. Mana shu xususiyat tufayli bir xil nomli fayllar muammo keltirib chiqarishi mumkin. Muammoli vaziyatdan chiqish uchun esa barcha shablonlarni ilova nomi bilan qo'shimcha papkaga joylashtirish tavsiya etiladi. Masalan, <home/templates/home/shablon.html>

ko‘rinishda. Barcha HTML fayllar uchun asosiy strukturani yaratib olish mumkindir:

```
<body>
  {% block content %}
  {% endblock %}
</body>
```

Rasm 2. HTML fayllar uchun bazaviy struktura

Bazaviy shablon ichida HTML bloklar yaratib olib, “block content” tarkibini o‘zgartirish mumkin bo‘ladi. Buning uchun yangi HTML fayl yaratib olib, bazaviy fayl strukturasi bilan meros olish kerak bo‘ladi:

```
{% extends "PATH TO HTML FILE" %}

{% block content %}
  <div>ADD ANY CODE HERE</div>
{% endblock %}
```

Rasm 3. Bazaviy HTML fayl strukturasi o‘zgartirish

Uncha katta bo‘lmagan kodni yaratib, uni boshqa shablonlarga qo‘shish mumkin. Bunday dasturiy kod bo‘laklarini alohida includes nomli papka ichida joylashtirib borish tavsiya qilinadi. Papka ichida oddiy HTML fayl yaratib olinadi va u yerda Jinja belgilash amalga oshiriladi:

```
{% block content %}
  <div>ADD ANY CODE HERE</div>
{% endblock %}
```

Rasm 4. HTML faylda Jinja belgilash kodi

Bunday fayllarni qo‘shib olish uchun «include» direktivasidan foydalaniladi:

```
{% include "PATH TO FILE" %}
```

Rasm 5. HTML faylni ilova qilish

Yuqorida keltirilgani kabi shablonizatorlar ro‘yxati juda keng. Python dasturlash tili bilan bog‘liq ro‘yxatda esa Django, Genshi, Jinja, Jinja2, Kid, Mako, web2py kabi nomlarni misol qilib keltirish mumkin.

Genshi – bu Pythonda yozilgan XMLga asoslangan lug‘atlar uchun shablon mexanizmi . Genshi yaratilgan mahsulotni XML-asosidagi tillarga, odatda HTML ga osongina kiritish va hujjatlar orasidagi elementlarni qayta ishlatish uchun ishlatiladi. Genshi sintaksisi Kid ga asoslangan , ammo uning arxitekturasi boshqacha. Genshi shablonlarni Python kodiga kompilyatsiya qilish o‘rniga, oqimga asoslangan API yordamida shablonlarni dinamik ravishda qayta ishlash orqali shablonlarni tezroq qayta ishlash vaqtida o‘zining ba’zi funksiyalarini amalga oshirishni maqsad qilgan [9].

Mako – Pythonda yozilgan shablon kutubxonasi. Mako o‘rnatilgan Python (ya’ni Python Server sahifasi) tili bo‘lib, u komponentlashtirilgan tartib va meros haqidagi tanish g‘oyalarni aniqlaydi. Mako shabloni Reddit tomonidan qo‘llaniladi [10].

Web2py – bu Python dasturlash tilida yozilgan ochiq manbali veb-ilovalar tizimi. Web2py veb-ishlab chiquvchilarga Python yordamida dinamik veb-kontentni dasturlash imkonini beradi. Web2py noldan veb-shakllarni ishlab chiqish kabi zerikarli veb-ishlab chiqish vazifalarini kamaytirishga yordam berish uchun mo‘ljallangan, garchi veb-ishlab chiquvchi kerak bo‘lsa, noldan shakl yaratishi mumkin [11].

Iqtiboslar:

1. O‘zbekiston Respublikasi Oliy Majlisining Qarori, «Elektron tijorat to‘g‘risida»gi O‘zbekiston Respublikasi Qonunini amalga kiritish haqida, 614-II-son 29.04.2004, <https://lex.uz/docs/165486>
2. O‘zbekiston Respublikasining Qonuni, «Elektron tijorat to‘g‘risida»gi O‘zbekiston Respublikasi qonuniga o‘zgartish va qo‘shimchalar kiritish haqida, O‘RQ-385-son 22.05.2015, <https://lex.uz/docs/2650293>

3. Soliev B.N., Rasulov A.M., O‘zbekistonda elektron tijorat va onlayn kurslar LMS tizimi asosida rivojlanishi istiqbollari, “O‘zbekistonda ilmiy-amaliy tadqiqotlar” mavzusida respublika 22-ko‘p tarmoqli ilmiy masofaviy konferensiya 16-qism, №22, noyabr 2020 yil, Toshkent, 7–9 betlar.

4. Django – Материал из Википедии, <https://ru.wikipedia.org/wiki/Django>

5. 10 самых популярных сайтов написанных на Django, <https://itproger.com/news/10-samih-populyarnih-saytov-napisannih-na-django>

6. Jinja – <https://ru.wikipedia.org/wiki/Jinja>

7. Основы шаблонизатора Jinja – <https://pythonru.com/uroki/7-osnovy-shablonizatora-jinja>

8. Шаблонизатор Jinja – <https://itproger.com/course/django/4>

9. Genshi (templating language) – [https://en.wikipedia.org/wiki/Genshi_\(templating_language\)](https://en.wikipedia.org/wiki/Genshi_(templating_language))

10. Mako (template engine) – [https://en.wikipedia.org/wiki/Mako_\(template_engine\)](https://en.wikipedia.org/wiki/Mako_(template_engine))

11. web2py – <https://en.wikipedia.org/wiki/Web2py>

DOI <https://doi.org/10.30525/978-9934-26-277-7-221>

USE OF MOBILE TECHNOLOGIES AND LEARNING TOOLS IN METHODOLOGICAL TRAINING OF THE FUTURE MATHEMATICS TEACHERS

Soia O. M.

*Candidate of Pedagogical Sciences,
Senior Lecturer at the Department of Mathematics and Computer Science
Vinnytsia Mykhailo Kotsiubynskyi State Pedagogical University
Vinnytsia, Ukraine*

When preparing a lesson with the use of mobile technologies, the teacher must organize it based on the goals and adhere to the main didactic principles (systematicity and consistency, accessibility, differentiated approach, scientific and others). The use of mobile phones is an addition to the lesson, not a substitute for the teacher, so the use of smartphones and other devices should not occupy more than half of the lesson.