JSC «Kazakh-British Technical University»
Faculty of Information Technology
Department of Electrical Engineering and Computer Science

APPROVED BY
Dean of FIT
Hajiyev F. A. _____
«____»_____ 20__.

SYLLABUS

Discipline: Advanced Django
Number of credits: 3
Term: Fall 20__
Instructors full name: Bobur Mukhsimbayev

Personal Information about the Instructor	Time and place of classes		Contact information
	Classes	Office Hours	e-mail
Bobur Mukhsimbayev	According to the schedule	Room 262, will be appointed	bobur.muhsimbaev@gmail.com

COURSE DURATION: 3 credits, 15 weeks, 60 class hours

COURSE DESCRIPTION

This course is designed to continue knowledge about back-end web development from previous subjects. Especially, for BackEnd development using framework Django.

Django is a web development framework that assists in building and maintaining quality web applications. Django helps eliminate repetitive tasks making the development process an easy and time saving experience. This course gives a complete understanding of Django.

This course is designed for developers who want to learn how to develop quality web applications using the smart techniques and tools offered by Django.

One of the main concepts of the course, is Django REST Framework. Django REST framework is a flexible and fully-featured library with modular and customizable architecture that aims at building sophisticated web APIs and uses Python and Django

COURSE OBJECTIVES

The objective of this course is to provide to the student real world task from industry and find best solution for them and working in team.

COURSE OUTCOMES

At the end of the course students will know:

- Advanced skill level of Python programming
- Tasks regarding the Django Signals
- Celery, RabbitMQ
- Difference between eager and lazy loading
- Working with Debugging tools
- Advanced Django Rest Framework techniques, File Upload
- Management commands
- Django Channels
- Deploy to remote server
- Services for working with ML models

LITERATURE

1. Daniel Greenfeld - Two Scoops of Django 1.11 - 2017
2. The Django Book - MIT 2015
    a. http://gsl.mit.edu/media/programs/south-africa-summer-2015/materials/djangobook.pdf
3. Adrian Holovaty, Jacob Kaplan-Moss, et al
    a. https://media.readthedocs.org/pdf/djangobook/latest/djangobook.pdf
4. Tutorials
    a. https://www.django-rest-framework.org/
    b. https://tutorial.djangogirls.org/en/
    c. https://djangoforbeginners.com
    d. https://docs.djangoproject.com/en/2.1/intro

Week	Class work		Laboratory works
	Topic	Lecture	
1	**Introduction. Recover knowledge:** ● Python programming language ● Django framework ● DRF ● REST API	1	*Laboratory work #1*
2	**User identification manipulation** ● Registration ● Authentication ● JWT ● User extending ● Profile ● User roles	2	*Laboratory work #2*
3	**Advanced Views:** ● Class Based views ● Function Based views ● Viewsets ● Extending CBV ● Mixins ● Action decorators ● Router registration	3	*Laboratory work #3*
4	**Advanced Model manipulations:** ● Model Manager ● Model QuerySet ● Field Types, ChoiceField, JSONField ● Abstract class ● Inheritance of models	4	*Laboratory work #4.*

5	**Advanced Serializers:** ● Custom Fields ● Nested Fields ● TimestampField ● read_only, write_only, required, source ● Serializer extending	5	*Laboratory work #5*
6	**Logging:** ● Logger levels ● Logging calls ● Logger formatters ● Configuration	6	*Laboratory work #6*
7	**File Uploading:** ● upload_to ● validators ● MultiPartParser	7	*Laboratory work #7*
8	*Midterm*		
9	**Django Signals:** ● pre_save ● post_save ● pre_delete ● post_delete	8	*Laboratory work #8*
10	**Complex SQL requests using ORM, request optimizations:** ● Debugging tools ● Eager and lazy loading ● annotate, aggregate ● Max, Min, Avg, Q, F ● select_related ● prefetch_related	9	*Laboratory work #9*
11	**Background tasks, periodic tasks:** ● Celery ● RabbitMQ ● crontab ● djang_celery_beat	9,10	*Laboratory work #10*
12	**Caching:** ● Frequent request response caching ● Redis ● Memcached **Tests** ● TDD ● unittest	11	*Laboratory work #11*

13	**Management commands:** ● System control with management command scripts ● cron tasks for report using management commands ● Python parsers, Scrapy	12	*Laboratory work #12*
14	**Django channels:** ● Async requests ● real time application **Deploy to remote server**	13	*Laboratory work #13*
15	*Endterm*	14	
16-17	Final Exam		Open-questions

Academic Policy

KBTU standard academic policy is used.

- Cheating, duplication, falsification of data, plagiarism, and crib are not permitted under any circumstances!
- Attendance is mandatory.

  Attention. Missing 20% attendance to lessons, student will be taken from discipline with filling in F (Fail) grade.

  Students must participate fully in every class. While attendance is crucial, merely being in class does not constitute "participation". Participation means reading the assigned materials, coming to class prepared to ask questions and engage in discussion.
- Students are expected to take an active role in learning.
- Written assignments (independent work) must be typewritten or written legibly and be handed in time specified. Late papers are not accepted!
- Students must arrive to class on time.
- Students are to take responsibility for making up any work missed.
- Make up tests in case of absence will not normally be allowed.
- Mobile phones must always be switched off in class.
- Students should always be appropriately dressed (in a formal/semi-formal style).
- Students should always show tolerance, consideration and mutual support towards other students.