**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

**Отчет по практической работе №1**

по дисциплине «Введение в информационные технологии» на тему:

Разработка web-приложений на фреймворке Flask

Выполнил: студент группы БВТ2104

Кравченко Николай Алексеевич

Проверил:

Мкртчян Грач Маратович

Москва

2021

**1 Постановка задачи**

Создать веб-приложение с функциями переводчика на языке программирования Python, с использованием фреймворка Flask.

**Задание**

1. Разработать html страницу. Создать форму с вводом текста. Создать кнопки, нажатие по которым переведет текст.

2. Применить стиль.

3. Подключить API стороннего переводчика.

4. Собрать все с помощью Flask.

2 Результат выполненной работы

Решение задания 1 и 2.

Часть кода страницы html:

<div class="wrapper">

<form class="card border-primary mb-3 myform" action="/" method="post">

<h3 class="card-header">Translate</h3>

<div class="card-body">

<textarea name="textarea" class="form-control">{{ data.source }}</textarea>

</div>

<div class="twobutton">

<div class="confirm">

<button class="btn btn-lg btn-primary" type="submit" name="action" value="ru">To RU</button>

</div>

<div class="confirm">

<button class="btn btn-lg btn-primary" type="submit" name="action" value="en">To EN</button>

</div>

</div>

</form>

<div class="card border-primary mb-3 myform">

<h3 class="card-header">Result</h3>

<div class="card-body">

<textarea name="result\_area" class="form-control">{{ data.result }}</textarea>

</div>

</div>

</div>

Часть кода css стиля:

.wrapper {  
 display: flex;  
 flex-direction: row;  
 margin-top: 80px;  
 margin-bottom: 80px;  
}  
.card-body{  
 flex: 3 0 0;  
}  
.myform {  
 flex: 1 0 auto;  
 min-height: 500px;  
 margin: 10px;  
}  
  
.form-control {  
 position: relative;  
 font-size: 16px;  
 height: 100%;  
 padding: 10px;  
}  
  
input[type="text"] {  
 margin-bottom: 5px;  
}  
.twobutton {  
 width: 100%;  
 flex: 0 1 auto;  
 padding: 1rem 1rem;  
 align-self: flex-end;  
 display: flex;  
 flex-direction: row;  
}  
.confirm{  
 flex: 1 0 0;  
}  
.confirm:first-child {padding-right: 10px}  
.confirm:last-child {padding-left: 10px}  
.btn-lg{  
 width:100%;  
}

Решение задания 3.

Код подключения к IBM API:

import requests, os

api\_key = os.environ['API\_IBM']

headers = {'Content-Type': 'application/json', }

params = (('version', '2018-05-01'),)

url = os.environ['API\_URL']

def get(text: str, lg: str):

result = ""

if text:

lg\_from = 'ru' if lg == 'en' else 'en'

data = f'{{"text": "{text}", "model\_id":"{lg\_from}-{lg}"}}'

response = requests.post(url, headers=headers, params=params, data=data.encode('utf-8'), auth=('apikey', api\_key))

try:

result = response.json()['translations'][0]['translation']

except:

result = response.text

return {"source": text, "result": result}

Решение задания 4.

Код Flask:

from flask import Flask, render\_template, request

from translate import get

app = Flask(\_\_name\_\_)

def parseform():

return get(request.form.get("textarea", ""), request.form.get("action", ""))

@app.route("/", methods=["post", "get"])

def process():

return render\_template("index.html", data=parseform())

@app.route("/api", methods=["post"])

def api():

return parseform()

**3 Вывод**

В результате проделанной работы мы изучили принципы создания Web-приложений на основе фреймворка Flask, основы Html и CSS, методы взаимодействия со сторонними сервисами с помощью API.