**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

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

Кафедра «Математическая кибернетика и информационные технологии»

**Отчет по лабораторной работе №15**

по дисциплине «Введение в информационные технологии» на тему:

Создание бекэнда на FastAPI

Выполнил: Горбунова Ксения БПИ2403

Проверил: Грач Мкртчян Маратович

Москва

2025

Создаём виртуальное окружение

python3 – m venv venv

Активируем его

source venv/bin/activate

Устанавиливаем необходимые зависимости

pip install fastapi Wikipedia

Код:

from fastapi import FastAPI, HTTPException, Depends

import wikipedia

from pydantic import BaseModel, Field, PositiveInt

from typing import Optional

app = FastAPI()

class TitleSearchParams(BaseModel):

word: str

num: Optional[PositiveInt] = Field(None)

class PageRequest(BaseModel):

title: str

class PageResponse(BaseModel):

title: str

references: list[str]

categories: list[str]

@app.get("/titles/{word}")

def search\_titles(params: TitleSearchParams = Depends()) -> list[str]:

return (wikipedia.search(params.word, results=params.num)

if params.num else wikipedia.search(params.word))

@app.post("/page/")

def get\_page(request: PageRequest) -> PageResponse:

try:

page = wikipedia.page(request.title)

except wikipedia.DisambiguationError as e:

raise HTTPException(

400, detail=f"Уточните запрос. Варианты: {e.options[:5]}")

except wikipedia.PageError:

raise HTTPException(404, detail="Страница не найдена")

except Exception as e:

raise HTTPException(500, detail=f"Ошибка сервера: {str(e)}")

return PageResponse(

title=page.title,

references=page.references,

categories=page.categories

)