Санкт-Петербургский Национальный Исследовательский Университет

Информационных Технологий, Механики и Оптики

Факультет Инфокоммуникационных Технологий

**Web-программирование**

**Лабораторная работа 2**

Выполнил

Белоус Я. Р.

Проверила

Марченко Е. В.

Санкт-Петербург, 2024

**Цель:** научиться работать с Git, gulp и создать программу-клиент для просмотра web-страниц.

**Задачи:**

* Изучить работу с Git – клонирование репозиториев, добавление изменений, коммиты и пуш изменений.
* Изучить Gulp и создать простой task
* Создать программу-клиент для просмотра web-страниц с возможностью изменения адресов страниц и интервала показа

**Задание 1**

Для начала была вызвана команда git -v для проверки установки git на компьютер:

****

Рисунок 1 – проверка установки git

Был клонирован репозиторий transport\_python, содержащий файлы лабораторной работы по другой дисциплине:

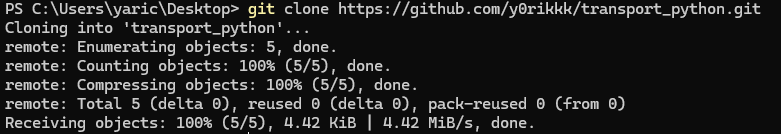
****

Рисунок 2 – клонирование репозитория

После этого были изменены три файла, с каждым из них был создан коммит:

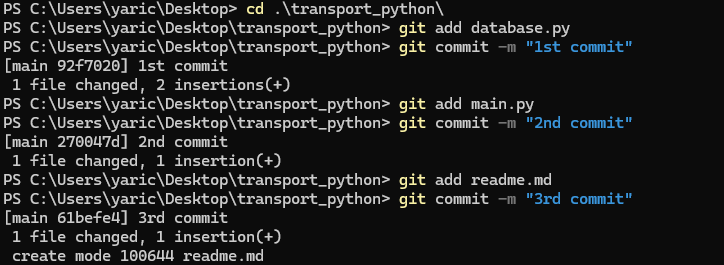
****

Рисунок 3 – добавление изменений и коммитов

Все три коммита отобразились при вызове git log:

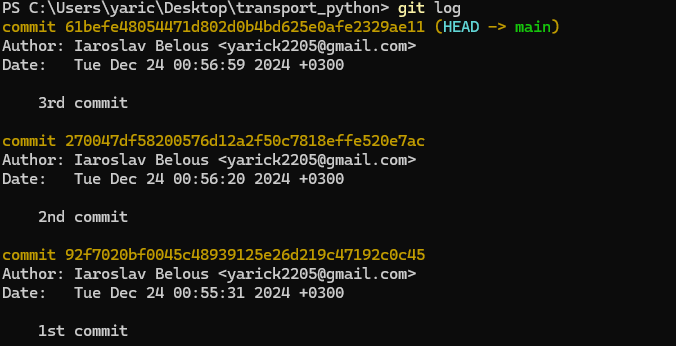
****

Рисунок 4 – просмотр коммитов с помощью git log

После этого все изменения были запушены c помощью git push:

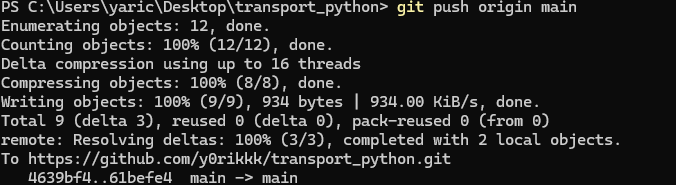
****

Рисунок 5 – пуш изменений

Коммиты отобразились на github:

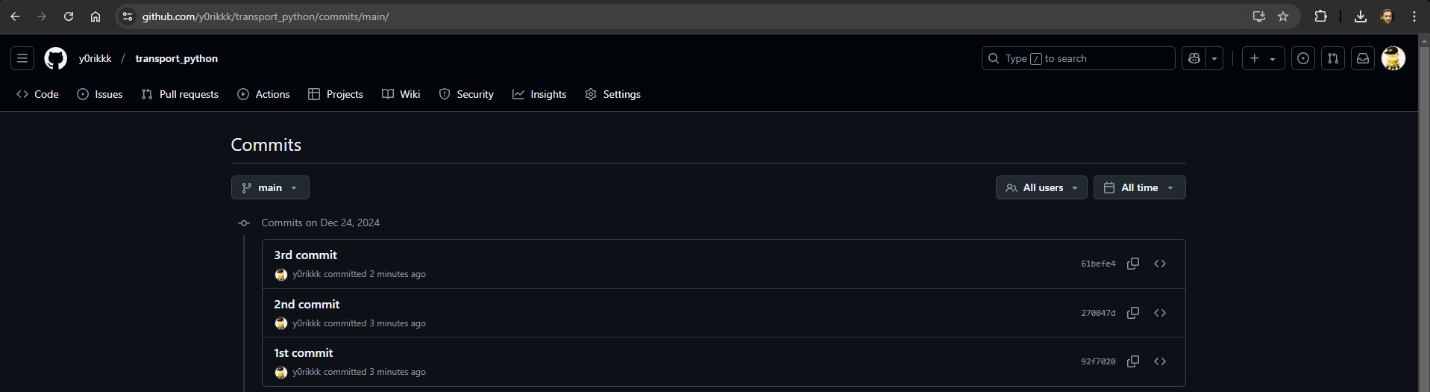
****

Рисунок 6 – коммиты на GitHub

Ссылка на репозиторий: <https://github.com/y0rikkk/transport_python>

**Задание 2**

Сначала были проверены версии node, npm, gulp:

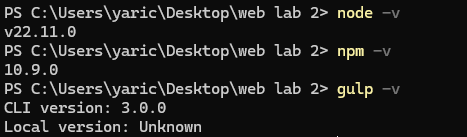
****

Рисунок 7 – проверка версий node, npm, gulp

После этого была настроена и подготовлена к работе директория проекта:

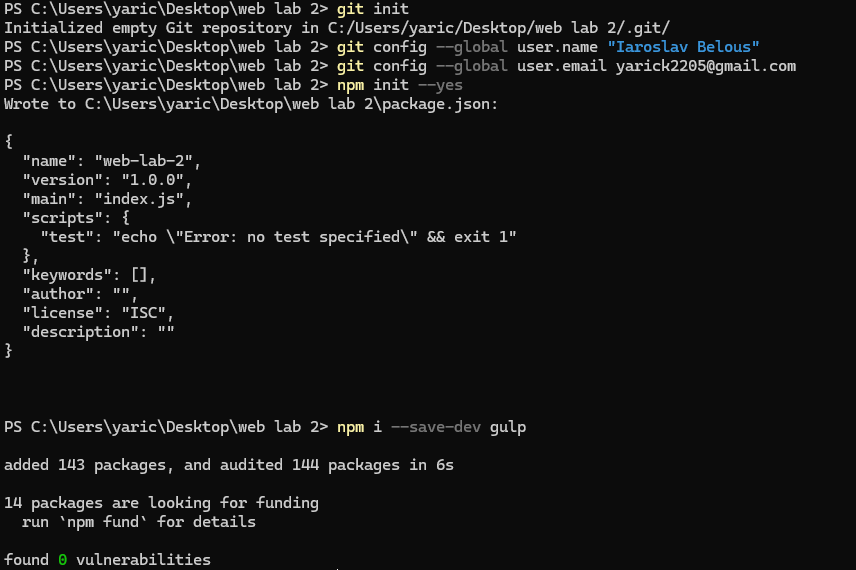
****

Рисунок 8 – подготовка к работе

После этого ещё раз была вызвана команда gulp -v для проверки правильности установки:

****

Рисунок 9 – проверка версии gulp после инициализации проекта

Был создан gulpfile.js и gulp task для автоматического обновления страницы с помощью browsersync:

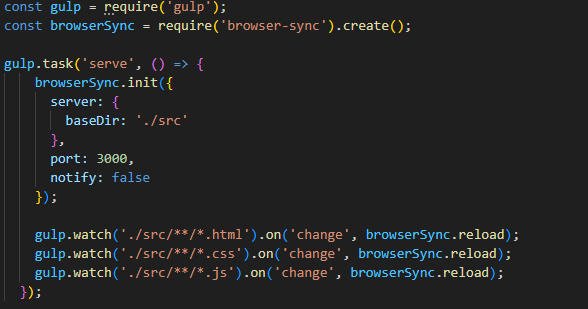


Рисунок 10 – код gulpfile.js

Был запущен этот task:

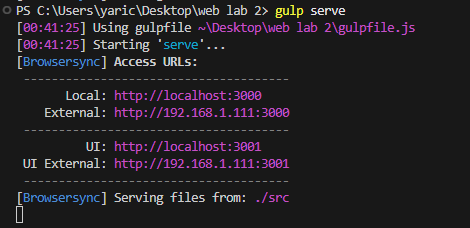


Рисунок 11 – запуск gulp serve

Теперь мы можем увидеть веб-страницу, находящуюся в папке src по адресу http://localhost:3000/



Рисунок 12 – index.html по адресу http://localhost:3000/

При изменении любого файла в папке src страница автоматически обновится:



Рисунок 13 – работа Browsersync

При изменении файла index.html в папке src страница сразу же обновилась.

**Задание 3**

Для выполнения третьего задания были созданы 2 файла: index.html и app.js.

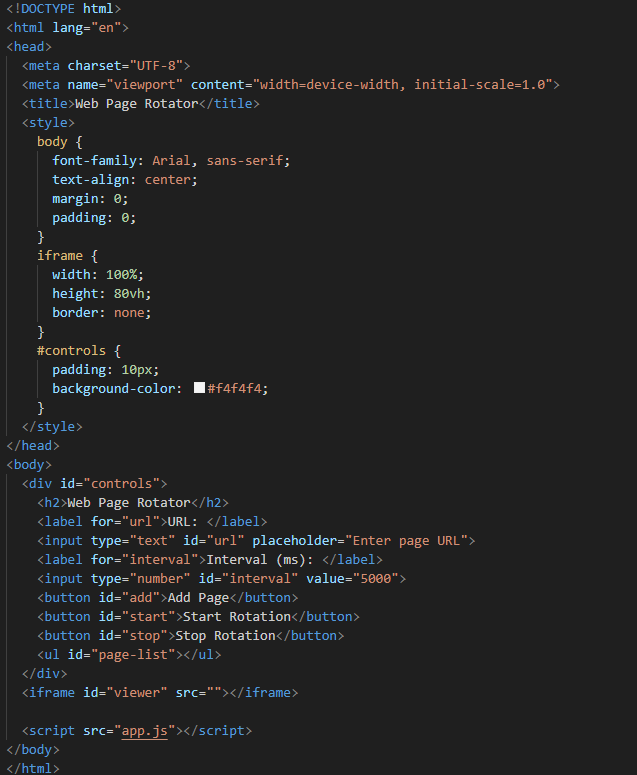


Рисунок 14 – index.html

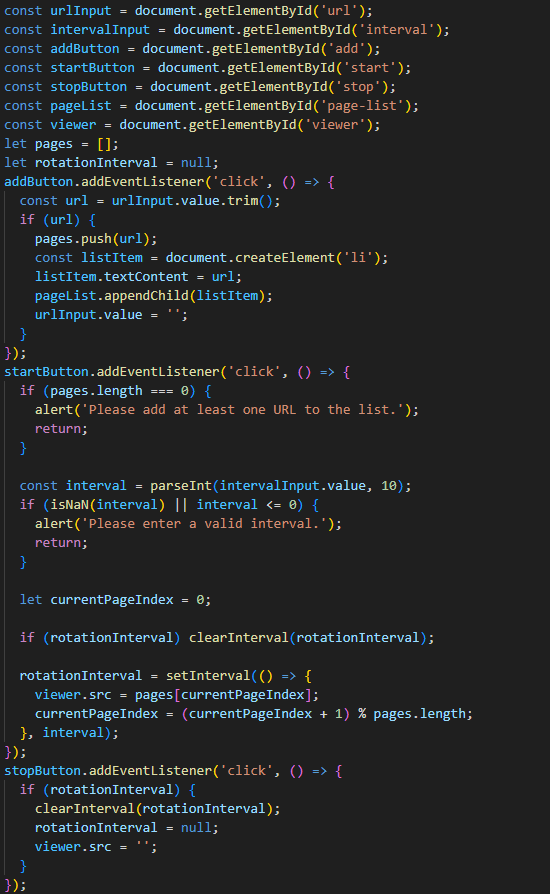


Рисунок 15 – код app.js

В результате получается такая страница, позволяющая просматривать другие страницы в тэге iframe, добавлять страницы, начинать и останавливать ротацию, а так же задавать её интервал:

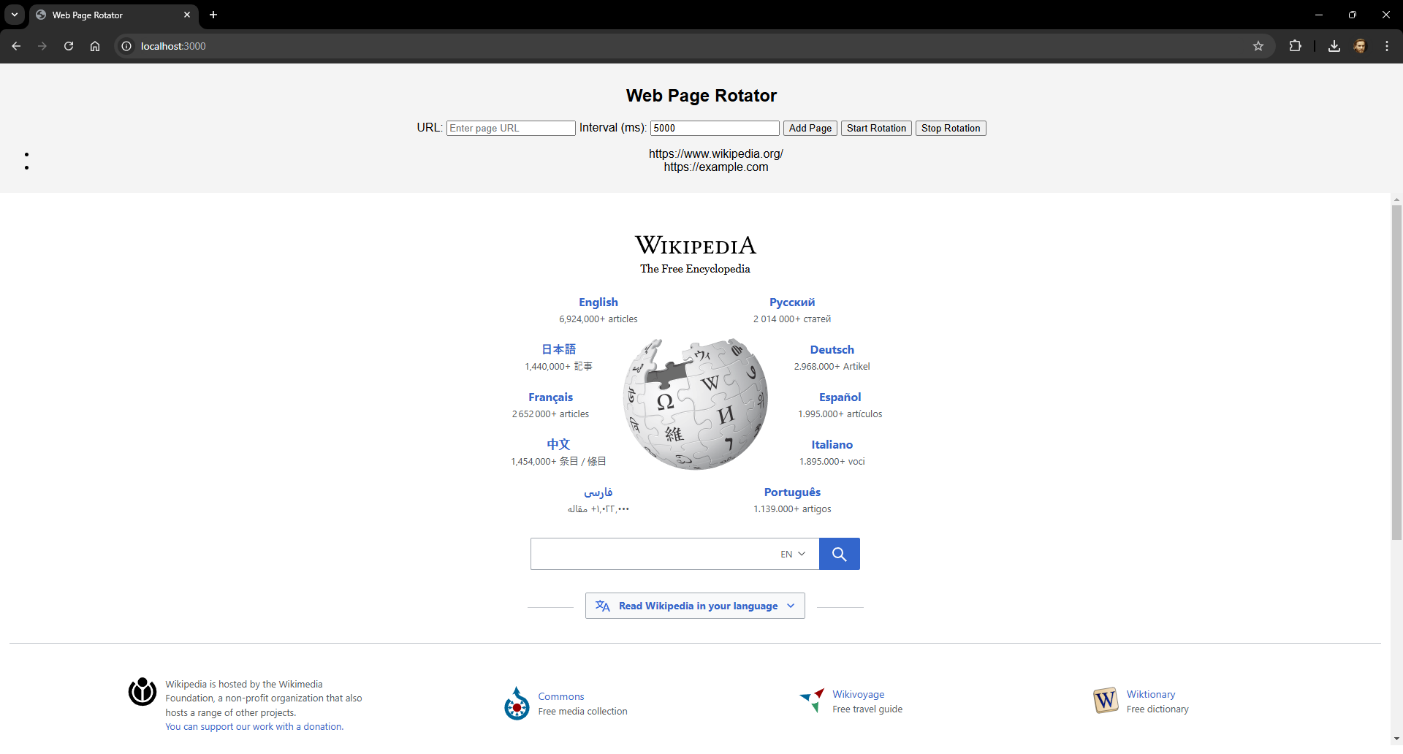


Рисунок 16 – страница в браузере

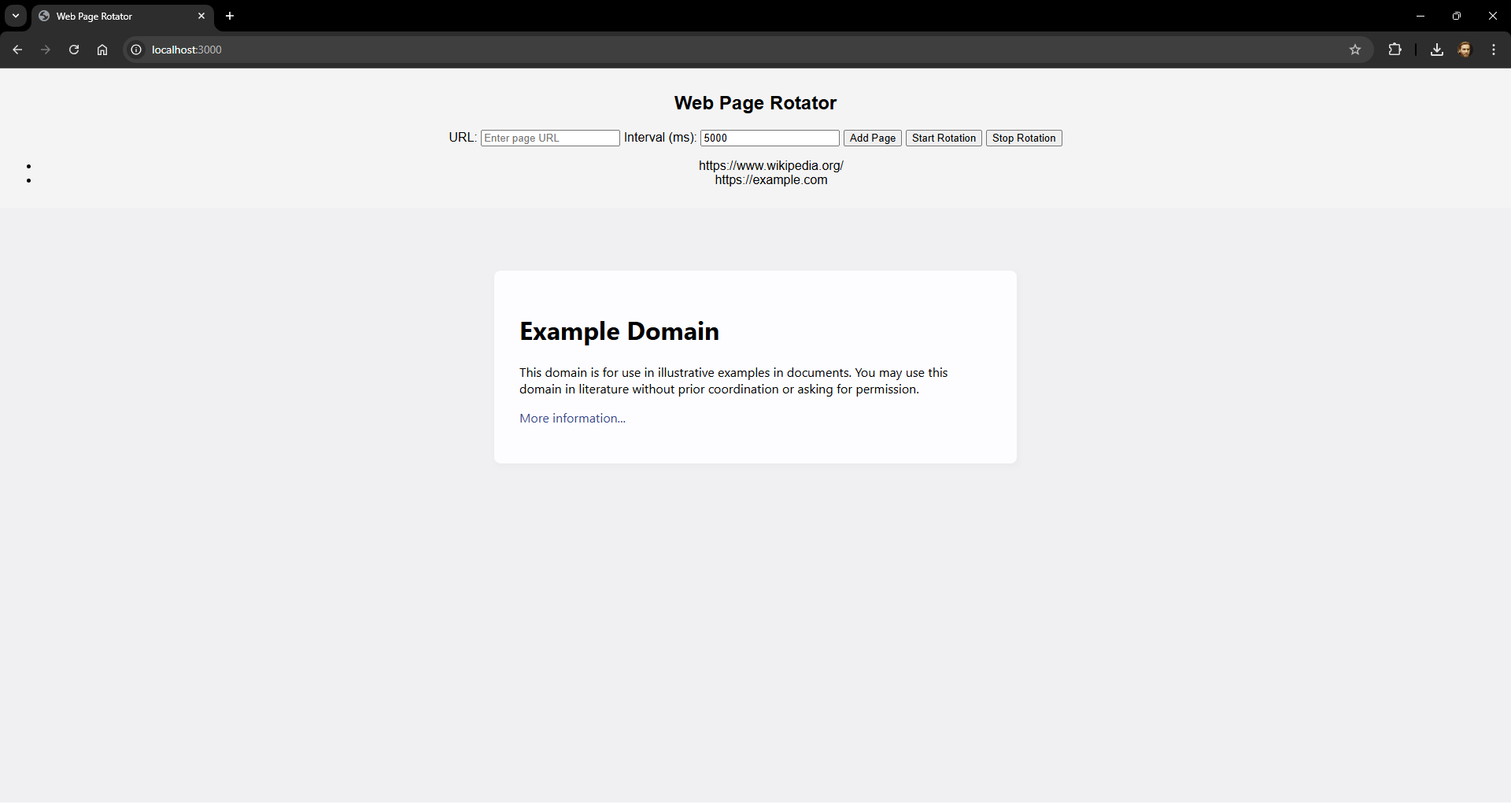


Рисунок 17 – страница в браузере

**Вывод:** были изучены Git, gulp и была создана программа-клиент для просмотра web-страниц.