

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Волгоградский государственный технический университет»
Факультет электроники и вычислительной техники
Кафедра программного обеспечения автоматизированных систем

ОТЧЕТ

О прохождении учебной практики: Практика по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности
на Кафедре Программного обеспечения автоматизированных систем
наименование кафедры или научной лаборатории университета

Руководитель практики от
университета _____ Донская А. Р.
должность *подпись*

Студент гр. ПриИн-266 _____ Ермолаев Е. С.
подпись

Студент гр. ПриИн-266 _____ Куканов Б. А.
подпись

Студент гр. ПриИн-266 _____ Мащенко И. Д.
подпись

Студент гр. ПриИн-266 _____ Стариков И. А.
подпись

Отчет защищен с оценкой _____

Волгоград 2025г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Волгоградский государственный технический университет»
Факультет электроники и вычислительной техники
Кафедра программного обеспечения автоматизированных систем

УТВЕРЖДАЮ

и. о. зав. кафедрой ПОАС

(подпись) Сычёв О. А.

«30» июня 2025г.

ЗАДАНИЕ

на учебную практику

Практика по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности

Студенту <u>Ермолаеву Егору Сергеевичу</u> (фамилия, имя, отчество)	Группа <u>ПрИн-266</u>
Студенту <u>Куканову Борису Алексеевичу</u> (фамилия, имя, отчество)	Группа <u>ПрИн-266</u>
Студенту <u>Мащенко Ивану Денисовичу</u> (фамилия, имя, отчество)	Группа <u>ПрИн-266</u>
Студенту <u>Старикову Илье Антоновичу</u> (фамилия, имя, отчество)	Группа <u>ПрИн-266</u>

1. Изучить язык программирования Python и его библиотеки Flask, Pandas и Plotly.
2. Разработать систему для анализа данных о потреблении энергии в умных домах.
3. Подготовить отчет о практике, содержащий соответствующий документ о проделанной работе.

Дата выдачи задания

«30» июня 2025г.

Руководитель практики от университета

Донская А. Р.
(подпись)

Студент гр. ПрИн-266 _____ Ермолаев Е. С.
подпись

Студент гр. ПрИн-266 _____ Куканов Б. А.
подпись

Студент гр. ПрИн-266 _____ Мащенко И. Д.
подпись

Студент гр. ПрИн-266 _____ Стариков И. А.
подпись

Оглавление

Введение	4
1. Внешняя спецификация	4
1.1 Описание программы	4
1.2 Функциональные требования	4
1.3 Нефункциональные требования	5
2. Внутренняя спецификация	5
2.1 Описание решения	5
2.2 Вызовы функций и потоки данных	6
3. Программа и методика испытаний	8
3.1 Цель испытаний	8
3.2 Аспекты тестирования	8
3.3 Результаты работы программы	8
3.4 Оценка точности результатов	10
Заключение	10
Список использованных источников	10

Введение

Целью учебной практики является выработка профессиональных навыков.

Основные задачи учебной практики:

- получение задания на разработку программного изделия;
- изучение основных принципов написания программ на языке программирования Python;
- изучение основных принципов работы библиотек Flask, Pandas и Plotly;
- разработка программы для анализа и визуализации данных о потреблении энергии в умных домах.

1. Внешняя спецификация

1.1 Описание программы

Программа EnerGraph представляет собой панель мониторинга данных об использовании энергии в умных домах. Основная цель системы — анализ и визуализация данных о потреблении электричества и показаниях датчиков температуры и влажности в помещениях. Система позволяет пользователям загружать статистические данные и анализировать их с помощью удобного веб-приложения.

Ссылка на программу - <https://github.com/Mikasa-Hayashi/EnerGraph>

1.2 Функциональные требования

Программа должна обеспечивать загрузку пользователем CSV-файла со статистическими данными. Если загруженный файл имеет неподдерживаемый формат (например, отсутствуют необходимые поля), то пользователю выводится сообщение об этом. Если же все данные корректны, то пользователь перенаправляется на страницу со следующими графиками:

1. График энергопотребления всех приборов;
2. График энергопотребления бытовых приборов и света (раздельно);

3. График почасового энергопотребления;
4. График дневного энергопотребления;
5. График зависимости энергопотребления от средней температуры в доме;
6. График зависимости энергопотребления от средней влажности в доме;
7. График зависимости энергопотребления от разности температуры снаружи и внутри дома;
8. График зависимости энергопотребления от разности влажности снаружи и внутри дома;
9. Гистограмма среднего энергопотребления по часам дня;
10. Гистограмма среднего энергопотребления по дням недели.

1.3 Нефункциональные требования

Программа должна обрабатывать большие объёмы статистических данных без особых задержек. Интерфейс веб-приложения при этом должен быть интуитивно понятным и адаптивным, чтобы пользователи могли комфортно работать на устройствах с различной ориентацией экранов.

Код программы должен быть простым и понятным, чтобы сторонние разработчики могли без особых проблем вносить в систему свои доработки.

2. Внутренняя спецификация

2.1 Описание решения

Система мониторинга энергопотребления реализована как веб-приложение, состоящее из клиентской части на HTML/CSS/JavaScript и серверной части на Python с использованием фреймворка Flask. Основной алгоритм работы системы строится на последовательной обработке данных: при получении CSV-файла через веб-интерфейс серверная часть выполняет загрузку и предварительную обработку данных с помощью библиотеки Pandas. Визуализация данных реализована через библиотеку Plotly, которая

обеспечивает построение интерактивных графиков с возможностью детального изучения отдельных временных периодов.

Входные данные

Система принимает CSV-файлы следующего формата:

Обязательные столбцы:

date – дата и время измерения.

Appliances – потребление энергии бытовыми приборами (в ваттах).

lights – потребление энергии освещением.

T1...TN, RH_1...RH_N – показания датчиков температуры и влажности (где N – число датчиков).

Дополнительные параметры:

T_out – температура снаружи;

Press_mm_hg – атмосферное давление;

RH_out – влажность снаружи;

Windspeed – скорость ветра;

Visibility – видимость;

Выходные данные

Интерактивные графики, которые описаны в функциональных требованиях (пункт 1.2)

2.2 Вызовы функций и потоки данных

Пользователь загружает файл со статистическими данными на странице index.html с помощью метода POST[/load-file]. Обработчик этого метода load_file() сохраняет файл в папке backend/datasets, вызывает метод data_handler.load_data() и проводит проверку корректности данных с помощью функции data_handler.validate_data(). Если данные имеют недопустимый формат,

то пользователю отображается страница `load_error.html` с описанием ошибки. Если данные корректны, то пользователь увидит страницу `graphs.html`, на которой будут отображены графики (обработчики `graph_1-10`). По нажатию на кнопки пользователь может увидеть аналитику по каждому из графиков, что обрабатывается в соответствующих функция, которые видны на диаграмме вызовов (рисунок 1). Диаграмма потоков данных показана на рисунке 2.

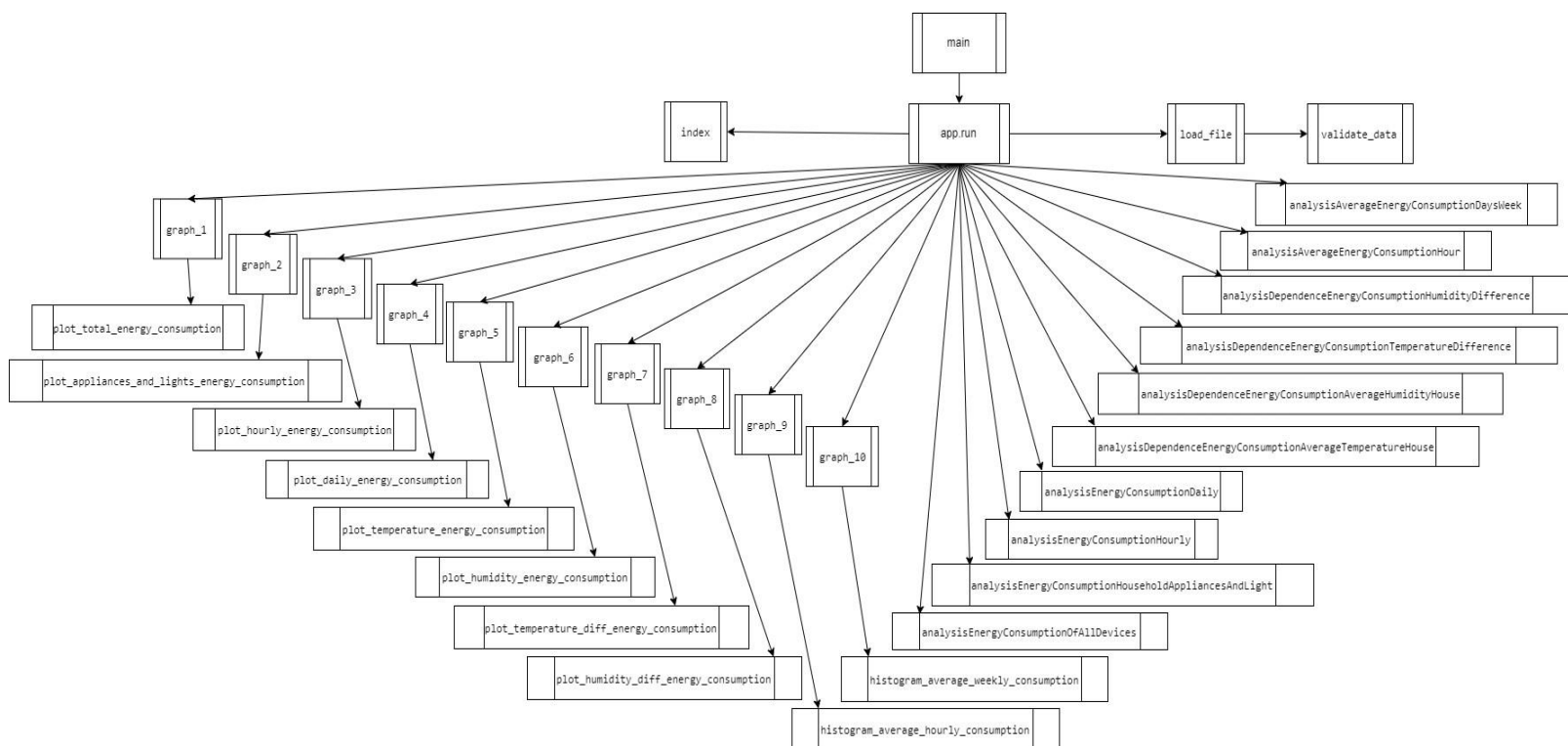


Рисунок 1. Диаграмма вызовов функций.

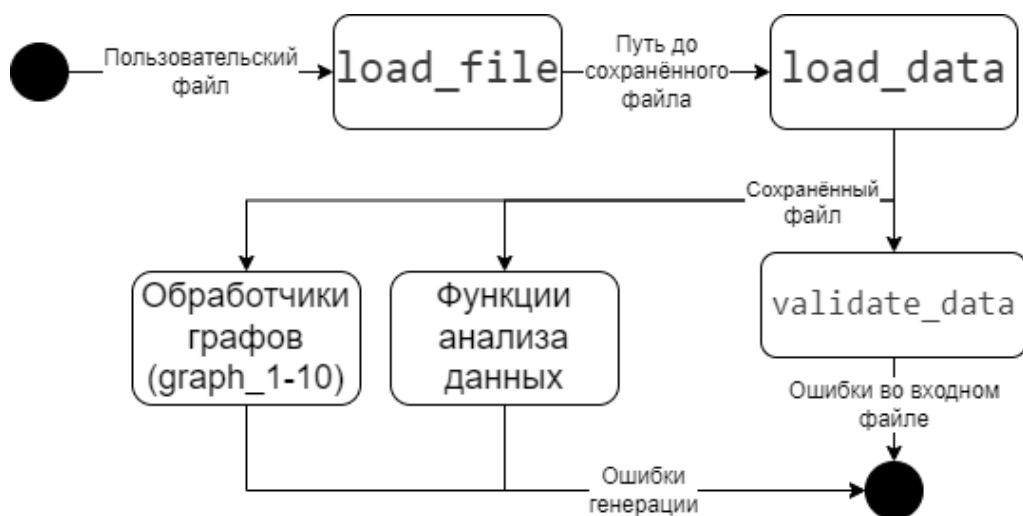


Рисунок 2. Диаграмма потоков данных

3. Программа и методика испытаний

3.1 Цель испытаний

Целью испытаний является выявление различных ошибок, которые могут возникнуть при работе с программой. Необходимо выявить основные аспекты тестирования и ситуации, которые могут приводить к ошибкам.

3.2 Аспекты тестирования

Аспекты тестирования представлены в таблице 1.

Таблица 1. Аспекты тестирования.

Аспект	Значение
Размер входного файла	Небольшой(< 10 мб), средний, большой(>100 мб)
Расширение входного файла	CSV, TXT
Количество датчиков (комнат)	Небольшое(< 5), среднее, большое(> 20)
Наличие дополнительных параметров	Отсутствуют, присутствуют
Частота измерений	Небольшая(каждую минуту), средняя, большая(каждый час или больше)
Период времени измерения	Несколько дней, недель, месяцев, лет

3.3 Результаты работы программы

1. Попытка загрузки файла, где отсутствуют столбцы date, lights, Appliances (рисунок 3).

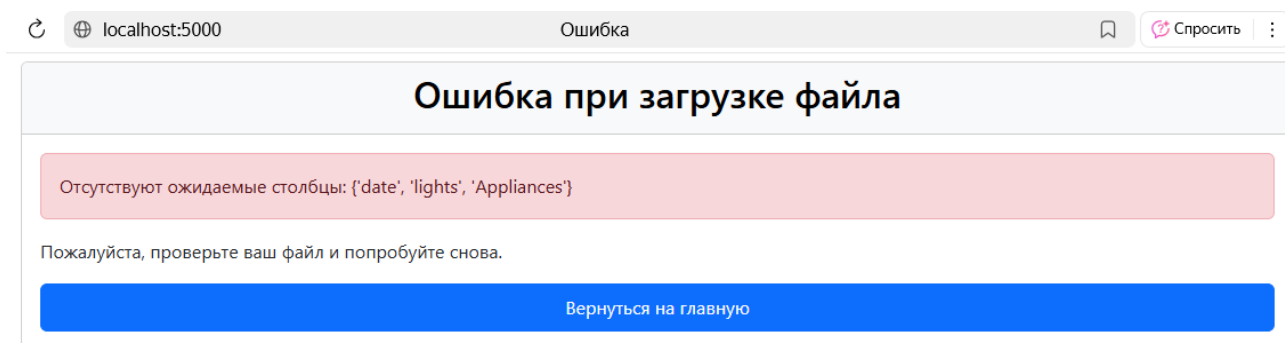


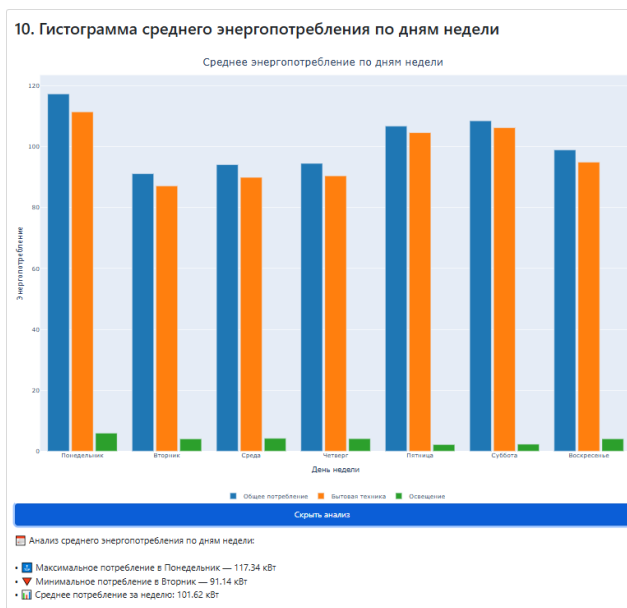
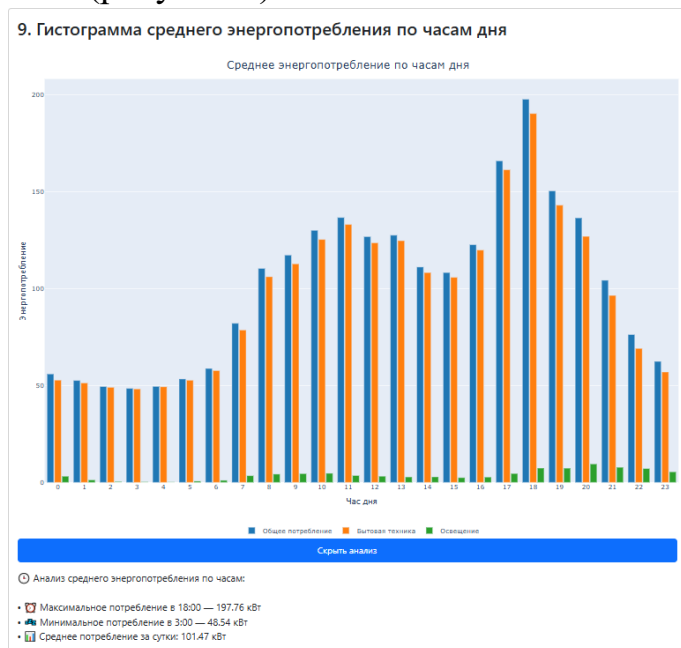
Рисунок 3. Ошибка – отсутствуют обязательные столбцы.

2. Общее энергопотребление (рисунок 4), потребление света и бытовых приборов по отдельности (рисунок 5).



Рисунки 4-5. Графики энергопотребления.

3. Гистограммы среднего потребления по часам дня (рисунок 6), по дням недели (рисунок 7).



Рисунки 6-7. Гистограммы среднего энергопотребления.

3.4 Оценка точности результатов

Аналитические функции обрабатывают статистические данные достаточно точно и выводят результаты с незначительными округлениями. При отрисовке графической информации необходимо понимать, что этот способ представления данных наглядный, но допускает небольшую погрешность.

Заключение

Разработанная программа успешно решает задачу мониторинга и анализа энергопотребления в умных домах. Реализованы все функциональные требования, включая загрузку данных, визуализацию и анализ. В дальнейшем можно реализовать большее число графиков и добавить новые типы анализа. В ходе выполнения задания были приобретены профессиональные навыки и выполнены намеченные цели.

Список использованных источников

1. Flask's documentation / [Электронный ресурс] // flask.palletsprojects.com : [сайт]. — URL: <https://flask.palletsprojects.com/en/stable/> (дата обращения: 10.07.2025).
2. Pandas documentation / [Электронный ресурс] // pandas.pydata.org : [сайт]. — URL: <https://pandas.pydata.org/docs/> (дата обращения: 10.07.2025).
3. Шпаргалка по визуализации данных в Python с помощью Plotly / [Электронный ресурс] // habr.com : [сайт]. — URL: <https://habr.com/ru/articles/502958/> (дата обращения: 10.07.2025).