

Федеральное государственное автономное образовательное учреждение высшего
образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра «Информатика и информационные технологии»

Направление подготовки/ специальность: Информационные системы и технологии

ОТЧЕТ

по проектной практике

Студент: Синицина Диана Александровна Группа: 241-336

Место прохождения практики: Московский Политех, кафедра Информатика и
информационные технологии

Отчет принят с оценкой _____ Дата _____

Руководитель практики: Рябчикова Анна Валерьевна, кафедра «Информатика и
информационные технологии»

Москва 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

1. Общая информация о проекте:
 - Название проекта
 - Цели и задачи проекта
2. Общая характеристика деятельности организации (*заказчика проекта*)
 - Наименование заказчика
 - Организационная структура
 - Описание деятельности
3. Описание задания по проектной практике
4. Описание достигнутых результатов по проектной практике

ЗАКЛЮЧЕНИЕ (*выводы о проделанной работе и оценка ценности выполненных задач для заказчика*)

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

ПРИЛОЖЕНИЯ (*при необходимости*)

ВВЕДЕНИЕ

1. Общая информация о проекте:

Название проекта:

«Мой дом» - Чат-бот, в котором собрана вся необходимая информация по общежитию студенческого городка.

Цели и задачи:

Цель проекта: Создание удобного и эффективного приложения для студентов Московского политеха, предоставляющего всю необходимую информацию по проживанию в общежитии и обеспечивающего легкий доступ к основным услугам.

Задачи:

1. Разработка интерфейса приложения, удобного для использования студентами.
2. Формирование базы данных.
3. Разработка и внедрение функционального чат-бота.
4. Обучение бота на типичных вопросах студентов.
5. Проведение тестирования приложения, корректировка найденных ошибок, сбор обратной связи.
6. Создание платформы для взаимодействия студентов с администрацией общежития через бота.
7. Запуск рекламной кампании и информирование студентов о возможностях приложения «Мой дом».

2. Общая характеристика деятельности организации (заказчика проекта)

Наименование заказчика: Администрация управления Московским Политехническим Университетом

Организационная структура: Администрация управления Московского Политехнического Университета (Московский Политех) – это руководящий

орган, отвечающий за стратегическое развитие, управление и координацию деятельности университета. В её структуру входят следующие ключевые подразделения:

1. Ректорат:

- Ректор
- Проректоры (по учебной, научной, административнохозяйственной работе и др.)

2. Учебные департаменты/управления

- Управление академической политики
- Учебно-методическое управление
- Департамент международного сотрудничества

3. Научно-исследовательский блок

- Управление научной деятельности
- Центры инновационного развития
- Лаборатории и научные группы

4. Административно-хозяйственные подразделения

- Финансово-экономическое управление
- Управление имущественного комплекса
- Отдел кадров

5. Социально-воспитательные структуры

- Департамент по работе со студентами
- Культурно-массовый отдел
- Спортивное управление

6. Информационно-технологическая инфраструктура

- IT-департамент
- Центр цифровых образовательных технологий

Описание деятельности: Администрация управления Московского Политехнического Университета обеспечивает:

1. Разработку и реализацию стратегии развития университета.

2. Организацию учебного процесса и контроль качества образования.
3. Координацию научно-исследовательских проектов и инновационных программ.
4. Управление финансовыми, материальными и кадровыми ресурсами.
5. Взаимодействие с государственными органами, бизнес-партнёрами и международными образовательными учреждениями.
6. Создание условий для всестороннего развития студентов и сотрудников.

3. Описание задания по проектной практике:

- Настройка Git и репозитория
- Написание документов в Markdown
- Создание статического веб-сайта
- Взаимодействие с организацией-партнёром
- Отчет о практике
- Практическая реализация технологии на выбор

4. Описание достигнутых результатов по проектной практике:

В ходе выполнения проектной практики были успешно реализованы все поставленные задачи, включая базовую и вариативную части.

1. Организация проекта и работа с Git

- Создан GitHub-репозиторий с полной структурой проекта.
- Освоены ключевые команды Git:
- Клонирование, создание веток (git branch).
- Регулярные коммиты с осмысленными сообщениями (git commit -m).
- Синхронизация с удалённым репозиторием (git push/pull).

2. Документирование в Markdown

Подготовлены файлы:

- README.md – описание проекта и инструкции по запуску.
- REPORT.md – отчёт о практике.
- JOURNAL.md – журнал изменений с хронологией работы.

3. Разработка текстового редактора

Реализован функционал:

- Работа с файлами:
 - 1) Открытие/сохранение в кодировке UTF-8.
 - 2) Подтверждение выхода (messagebox).
- Пользовательский интерфейс:
 - 1) Смена тем (тёмная/светлая) и шрифтов (Arial, Calibri, Times New Roman).
 - 2) Статус-бар с отображением позиции курсора.
- Дополнительные возможности:
 - 1) Поиск текста с подсветкой результатов.
 - 2) Горячие клавиши (Ctrl+O, Ctrl+S, Ctrl+F).

4. Взаимодействие с партнёрской организацией

- Посещены пары-лекции
- Было произведено совместное проектирование
- Были распределены задачи в команде

5. Вариативная часть: Создание текстового редактора

```
from tkinter import *
from tkinter import messagebox
from tkinter import filedialog
from tkinter import ttk

def change_theme(theme):
    text_field['bg'] = view_colors[theme]['text_bg']
    text_field['fg'] = view_colors[theme]['text_fg']
    text_field['insertbackground'] = view_colors[theme]['cursor']
    text_field['selectbackground'] = view_colors[theme]['select_bg']

def change_fonts(fontss):
    text_field['font'] = fonts[fontss]['font']

def notepad_exit():
    answer = messagebox.askokcancel('Выход', 'Вы точно хотите выйти?')
    if answer:
        root.destroy()

def open_file():
    file_path = filedialog.askopenfilename(title='Выбор файла',
                                           filetypes=((('Текстовые документы (txt)', '*.txt'),
                                                         ('Все файлы', '*.*'))))
    if file_path:
        text_field.delete('1.0', END)
        with open(file_path, encoding='utf-8') as f:
            text_field.insert('1.0', f.read())
        update_status_bar()
```

```

def save_file():
    file_path = filedialog.asksaveasfilename(filetypes=((('Текстовые документы (txt)', '*.txt'),
                                                         ('Все файлы', '*.*'))))
    if file_path:
        with open(file_path, 'w', encoding='utf-8') as f:
            text = text_field.get('1.0', END)
            f.write(text)
            update_status_bar()

def find_text():
    search_window = Toplevel(root)
    search_window.title("Поиск текста")
    search_window.geometry("300x100")

    Label(search_window, text="Введите текст для поиска:").pack(pady=5)

    search_entry = Entry(search_window, width=30)
    search_entry.pack(pady=5)
    search_entry.focus_set()

def search():

    text_field.tag_remove('found', '1.0', END)

    search_term = search_entry.get()

    if search_term:
        start_pos = '1.0'
        while True:

            start_pos = text_field.search(search_term, start_pos, stopindex=END)

            if not start_pos:
                break

            end_pos = f"{start_pos}+{len(search_term)}c"

            text_field.tag_add('found', start_pos, end_pos)

            start_pos = end_pos

        text_field.tag_config('found', background='yellow', foreground='black')

    Button(search_window, text="Найти", command=search).pack(pady=5)

    search_window.bind('<Return>', lambda event: search())

```

```

def update_status_bar(event=None):

    cursor_pos = text_field.index(INSERT)
    line, column = cursor_pos.split('.')

    total_lines = text_field.index('end-1c').split('.')[0]

    status_bar.config(text=f"Строка: {line}, Колонка: {column} | Всего строк: {total_lines}")

root = Tk()
root.title('Текстовый редактор')
root.geometry('800x600')

main_menu = Menu(root)

file_menu = Menu(main_menu, tearoff=0)
file_menu.add_command(label='Открыть', command=open_file, accelerator="Ctrl+O")
file_menu.add_command(label='Сохранить', command=save_file, accelerator="Ctrl+S")
file_menu.add_separator()
file_menu.add_command(label='Закрыть', command=notepad_exit)
main_menu.add_cascade(label='Файл', menu=file_menu)

edit_menu = Menu(main_menu, tearoff=0)
edit_menu.add_command(label='Поиск', command=find_text, accelerator="Ctrl+F")
main_menu.add_cascade(label='Правка', menu=edit_menu)

view_menu = Menu(main_menu, tearoff=0)
view_menu_sub = Menu(view_menu, tearoff=0)
font_menu_sub = Menu(view_menu, tearoff=0)

view_menu_sub.add_command(label='Темная', command=lambda: change_theme('dark'))
view_menu_sub.add_command(label='Светлая', command=lambda: change_theme('light'))
view_menu.add_cascade(label='Тема', menu=view_menu_sub)

```



```

font_menu_sub.add_command(label='Arial', command=lambda: change_fonts('Arial'))
font_menu_sub.add_command(label='Calibri', command=lambda: change_fonts('Calibri'))
font_menu_sub.add_command(label='Times New Roman', command=lambda: change_fonts('TNR'))
view_menu.add_cascade(label='Шрифт', menu=font_menu_sub)
main_menu.add_cascade(label='Вид', menu=view_menu)

root.config(menu=main_menu)

text_frame = Frame(root)
text_frame.pack(fill=BOTH, expand=1)

scroll_y = Scrollbar(text_frame)
scroll_y.pack(side=RIGHT, fill=Y)

scroll_x = Scrollbar(text_frame, orient=HORIZONTAL)
scroll_x.pack(side=BOTTOM, fill=X)

text_field = Text(text_frame,
                  yscrollcommand=scroll_y.set,
                  xscrollcommand=scroll_x.set,
                  wrap=NONE,
                  padx=10,
                  pady=10,
                  spacing3=10,
                  width=30,
                  font='Arial 14')
text_field.pack(expand=1, fill=BOTH)

scroll_y.config(command=text_field.yview)
scroll_x.config(command=text_field.xview)

status_bar = Label(root, text="Готово", bd=1, relief=SUNKEN, anchor=W)
status_bar.pack(side=BOTTOM, fill=X)

```

```

view_colors = {
    'dark': {
        'text_bg': 'black',
        'text_fg': 'lime',
        'cursor': 'brown',
        'select_bg': '#8d917a'
    },
    'light': {
        'text_bg': 'white',
        'text_fg': 'black',
        'cursor': '#A5A5A5',
        'select_bg': '#FAEEDD'
    }
}

fonts = {
    'Arial': {
        'font': 'Arial 14'
    },
    'Calibri': {
        'font': 'Calibri 14'
    },
    'TNR': {
        'font': ('Times New Roman', 14)
    }
}

change_theme('dark')

root.bind('<Control-o>', lambda e: open_file())
root.bind('<Control-s>', lambda e: save_file())
root.bind('<Control-f>', lambda e: find_text())

text_field.bind('<KeyRelease>', update_status_bar)
text_field.bind('<ButtonRelease>', update_status_bar)

update_status_bar()

root.mainloop()

```

6. Презентация результатов

- Подготовлен статический сайт на HTML/CSS:
 - 1) Главная страница с описанием проекта.
 - 2) Страницы: "О проекте", "Участники", "Журнал", "Ресурсы" .

Загружены отчёты в форматах DOCX и PDF в СДО (LMS).

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Python 3 Documentation: Tkinter [Электронный ресурс]. – Режим доступа: <https://docs.python.org/3/library/tkinter.html> (дата обращения: 01.11.2023).
2. Создание контента для вашего первого веб-сайта [Электронный ресурс] // MDN Web Docs. – Режим доступа: https://developer.mozilla.org/ru/docs/Learn_web_development/Getting_started/Your_first_website/Creating_the_content (дата обращения: 01.11.2023).
3. Введение в HTML [Электронный ресурс] // HTML Academy. – Режим доступа: <https://htmlacademy.ru/blog/html/introduction-to-html> (дата обращения: 01.11.2023).
4. CSS: Cascading Style Sheets [Электронный ресурс] // MDN Web Docs. – Режим доступа: <https://developer.mozilla.org/en-US/docs/Web/CSS> (дата обращения: 01.11.2023).
5. CSS: что это такое и как работает [Электронный ресурс] // SkillFactory. – Режим доступа: <https://blog.skillfactory.ru/glossary/css/> (дата обращения: 01.11.2023).
6. Вёрстка сайта: инструкция для новичков [Электронный ресурс] // Skillbox. – Режим доступа: <https://skillbox.ru/media/code/vyerstka-sayta-instruktsiya-dlya-novichkov/> (дата обращения: 01.11.2023).

ЗАКЛЮЧЕНИЕ

В ходе проектной практики я самостоятельно разработала текстовый редактор на Python с использованием Tkinter, реализовав ключевые функции: работу с файлами (открытие/сохранение), смену тем оформления, выбор шрифтов, поиск по тексту с подсветкой и статус-бар для навигации.

Параллельно я создала сайт-визитку проекта на HTML/CSS с описанием проекта ПД «Мой дом», журналом изменений и полезными ресурсами.

Весь процесс разработки сопровождался грамотной документацией в Markdown - от README с инструкциями до технического отчета, а также блок-схемой основных алгоритмов.

Особое внимание уделила организации рабочего процесса: настроила GitHub-репозиторий, освоила Git (ветки, коммиты, пуши), регулярно фиксировала изменения.

Завершила проект подготовкой видео-презентации и финальных отчетов в DOCX/PDF форматах.

Эта практика дала мне бесценный опыт самостоятельной разработки - от идеи до рабочего продукта, научила грамотно планировать этапы работы, документировать процесс и презентовать результаты.

Несмотря на сложности, особенно с реализацией поиска и интеграцией статус-бара, я справилась со всеми задачами и теперь вижу возможности для улучшения редактора, такие как добавление вкладок, подсветки синтаксиса и облачной синхронизации.

ПРИЛОЖЕНИЯ

Git-репозиторий: <https://github.com/dayinkywww/practika.git>