**План управления программным проектом (SPMP) для менеджера бюджета**

Утверждаю

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

20.02.25 Михайлов Иван: Создание первой версии документации

**1. Введение**

**1.1 Обзор проекта**

Этот проект организован для разработки программного приложения "Менеджер бюджета" — инструмента, который поможет пользователям отслеживать свои доходы и расходы. Проект выполняется командой из трех студентов в рамках учебного курса по разработке программного обеспечения. Приложение будет разработано с использованием языка программирования Python, библиотеки Flet для создания пользовательского интерфейса и базы данных SQLite3 для хранения данных. Разработка будет проходить в три этапа (итерации), чтобы обеспечить постепенное улучшение функциональности и качества продукта.

**1.2 Результирующие артефакты проекта**

Следующие материалы должны быть поставлены в указанные сроки.

**Версия 1 (прототип)** с документацией — вторая неделя марта 2025 года

**Версия 2** с документацией — третья неделя апреля 2025 года.

**Версия 3 (финальная)** с документацией — третья неделя мая 2025 года.

Документация для каждой версии включает: SPMP (План управления проектом, данный документ), SRS (Спецификация требований к программному обеспечению), SDD (Проектная документация программного обеспечения), STP (План тестирования программного обеспечения), исходный код, исполняемый файл приложения, руководство пользователя.

**1.3 Развитие SPMP**

Этот документ будет поддерживаться и обновляться лидером проекта (назначение роли будет указано в разделе 2). Лидер проекта отвечает за актуальность SPMP, внося изменения по мере необходимости, не реже одного раза в неделю. Документ будет находиться под управлением конфигурации (например, с использованием Git), чтобы отслеживать все изменения. SPMP в основном следует рекомендациям стандарта IEEE 1058.1-1987, адаптированным под учебный проект.

**1.4 Ссылочные материалы**

Основное руководство: Software Engineering: an Objected-Oriented Perspective E. Braude, Wiley, 2000

Руководство по Python: официальная документация Python (python.org)

Документация Flet: официальный сайт библиотеки (flet.dev)

Документация SQLite3: официальная документация (sqlite.org)

**1.5. Аббревиатуры**

SPMP — Softivare Project Management Plan. План управления программным  
проектом (данный документ)  
SRS \_ Software Requiremcnts Specification. Спецификация требований к про-  
граммному обеспечению,  
SDD — Softwvare Design Document. Проектная документация программного  
обеспечения.

STD – документация по тестированию программного обеспечения.

**Flet** — Библиотека для создания интерфейсов на Python.

**SQLite3** — Легковесная система управления базами данных.

**Git** — Система управления версиями.

**2. Организация проекта**

**2.1 Модель процесса**

Проект разрабатывается с использованием инкрементной модели разработки, которая включает итеративное добавление функционала и его тестирование. Основные этапы:  
- Анализ требований и проектирование базы данных (SQLite);  
- Разработка графического интерфейса (Tkinter, Kivy);  
- Интеграция функциональности ввода, обработки и отображения данных;  
- Тестирование и отладка;  
- Оптимизация и адаптация под мобильные устройства.

**2.2 Организационная структура**

Проект выполняется командой из трех студентов: разработчик (по совместительству лидер проекта) аналитик, тестировщик. Каждый участник команды будет проводить инспектирование работы других участников.

**2.3** **Организационные рамки и взаимосвязи**

Взаимодействие с преподавателями: получение рекомендаций и корректировка направлений работы. Обмен опытом с другими студентами, обсуждение технических решений. Использование открытых ресурсов (документация SQLite, Tkinter, Kivy) для разработки.

**2.4 Ответственности за проект**

- Разработка и поддержка кода (Михайлов Иван)  
- Тестирование с использованием автоматизированных и ручных методов (Сергачев Семен)  
- Документирование, ведение отчетности по проекту, включая описание кода, техническую документацию и пользовательские инструкции (Михайлов Иван)  
- Финальное представление проекта, подготовка демонстрации и отчетности для учебного процесса (вся команда)

*1*

**3. Управляющий процесс**

**3.1 Цели и приоритеты**

Высший приоритет имеет достижение заданного уровня качества. На втором месте по приоритетности стоит выполнение проекта в срок. Третий приоритет имеет выполнение максимального количества требований. Четвертый приоритет имеет написание программного кода, который можно улучшить в дальнейшем.

**3.2 Допущения, зависимости и ограничения**

Отсутствуют.

**3.3 Управление рисками**

Риск №1 Отсутствие опыта работы с графическими интерфейсами у команды.

Риск №2 Команда не обладает достаточными навыками работы с базами данных.

Риск №3 Знание недостаточного количества модулей необходимых для написания программы.

**3.4 Механизмы мониторинга и контроля**

Вся команда будет встречаться на совещании в начале каждой фазы (определение требований, проектирование, реализация) каждой итерации. Должны проводиться еженедельные совещания по проекту каждую среду с 16:00 до 18:00. Следует принять все меры к тому, чтобы на этих совещаниях рассматривались сразу все общие для команды дела. Участники команды должны зарезервировать время в четверг с 12 до 14 для проведения дополнительных совещаний, если понадобится. Лидер команды должен предупредить участников о проведении дополнительного совещания не позднее 15:30 в среду.

**3.5 План расстановки кадров**

Under construction

**4. Технический процесс**

SRS описывает некоторые аспекты требуемого технологического проекта. Здесь рассматриваются те аспекты, которые не установлены явно в SRS.

**4.1 Методы, инструменты и технологии**

В нашем программном проекте реализация ведется на языке Python в среде разработки PyCharm. Во всех случаях используется объектно-ориентированный подход.

**4.2 Документация программного обеспечения**

См. SQAP

**5. Распределение работ, план-график и бюджет**

**5.1 Распределение работ**

**5.2 Зависимости**

Вторая итерация зависит от результатов первой

**5.3 Потребности в ресурсах**

В процессе разработки программного проекта будут заняты три студента. Три ноутбука со следующими техническими характеристиками: Asus Vivobook S и 2 Macbook air M3.

**5.4 Выделение бюджета и ресурсов**

Отсутствует.

**5.5 План-график**

**6. Дополнения**

**6.1 Указатель**

Должен быть представлен.

**6.2 Приложения**

Должны быть представлены.

*1.1 Описание проекта*  
Проект представляет собой приложение для учета личных финансов с графическим интерфейсом. Пользователь сможет добавлять доходы и расходы, просматривать историю транзакций и анализировать данные с помощью диаграмм.

*1.2 Функциональные требования*

- Реализация регистрации нескольких пользователей  
- Ввод и сохранение доходов и расходов в базе данных SQLite.

- Категоризация трат

- Установка месячного бюджета и распределение денежных средств по категориям.

- Анализ действий пользователя (предупреждение о превышение трат на определенную категорию)

- Возможность установки цели по накоплению денег

- Создание графического интерфейса.   
- Возможность добавления нескольких транзакций без перезапуска приложения.   
- Экспорт данных в CSV.   
- Отображение финансовых данных в виде диаграмм.   
- Перенос приложения на мобильные устройства с использованием Kivy.