

# **Проект «Создание телеграм-бота с целью агрегирования текущих вопросов к обсуждению по проекту и упорядочивания календаря руководителя»**

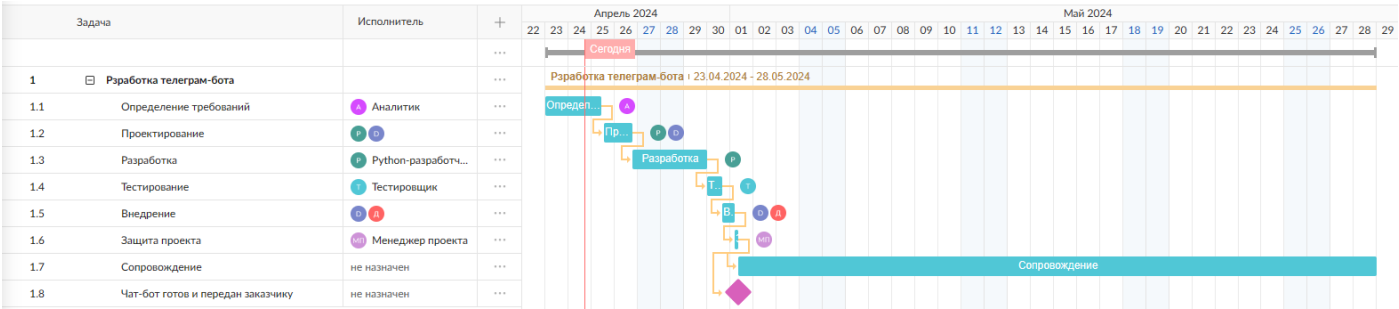
## **Описание команды по ролям PRINCE2**

- Менеджер проекта: организация, планирование (план и бюджет проекта) и контроль работ по проекту.
- Проектная группа:
  - Аналитик: описание процесса AS IS и TO BE, разработка требований к продукту и инструкций пользователя.
  - Python-разработчик: разработка телеграм-бота и базы данных.
  - Тестировщик: функциональное тестирование бота и выявление ошибок.
  - DevOps-инженер: разворачивание и настройка серверной инфраструктуры.
  - Дизайнер: оформление майнд-карты и презентации для защиты проекта.
- Администратор: организация совещаний, взаимодействия между участниками команды, хранение материалов по проекту.

## **План проекта**

- 1) Определение требований:
  - Сбор требований к функциональности бота.
  - Определение основных задач и функций.
  - Разработка пользовательских сценариев.
- 2) Проектирование:
  - Проектирование базы данных для хранения информации о вопросах, задачах и календаре.
  - Проектирование архитектуры бота.
  - Определение интеграций с внешними сервисами.
- 3) Разработка:
  - Написание кода для бота на Python.
  - Создание базы данных и написание SQL-запросов для работы с данными.
- 4) Тестирование:
  - Проведение функционального тестирования всех основных функций бота.
  - Проверка на соответствие требованиям.
  - Выявление и устранение ошибок.
- 5) Внедрение:
  - Разворачивание бота на сервере.
  - Разработка майнд-карты взаимодействия руководителя проектов и членов команды с чат-ботом.
  - Настройка автоматического обновления и мониторинга.
  - Подключение бота к Telegram и другим сервисам.
- 6) Защита проекта:
  - Выступление с презентацией проекта перед заказчиком.
  - Документирование проекта.
- 7) Сопровождение:
  - Поддержка и обновление бота.
  - Дальнейшее развитие функциональности в соответствии с потребностями пользователей.

Диаграмма Ганта проекта:



Бюджет проекта верхнего уровня

Основной статьей расходов в проекте является заработная плата проектной команды.

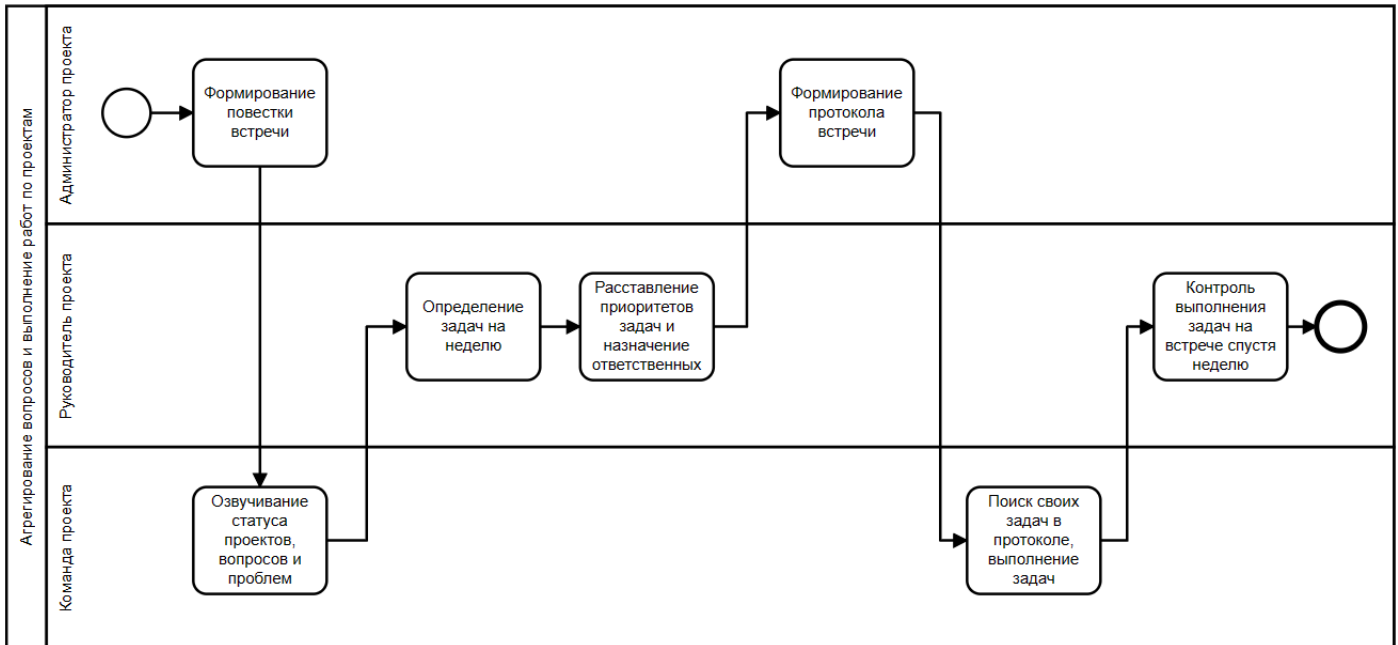
Роль	Участие в проекте (ч)	Ставка/ч (руб.)	ЗП (руб.)
Менеджер проекта	30	1 250	37 500
Аналитик	20	937,5	18 750
Python-разработчик	20	1 250	25 000
Тестировщик	5	1 125	5 625
DevOps-инженер	5	1 125	5 625
Дизайнер	5	687,5	3 437,5
Администратор	30	625	18 750
ИТОГО:			114 687,5

Бюджет проекта:

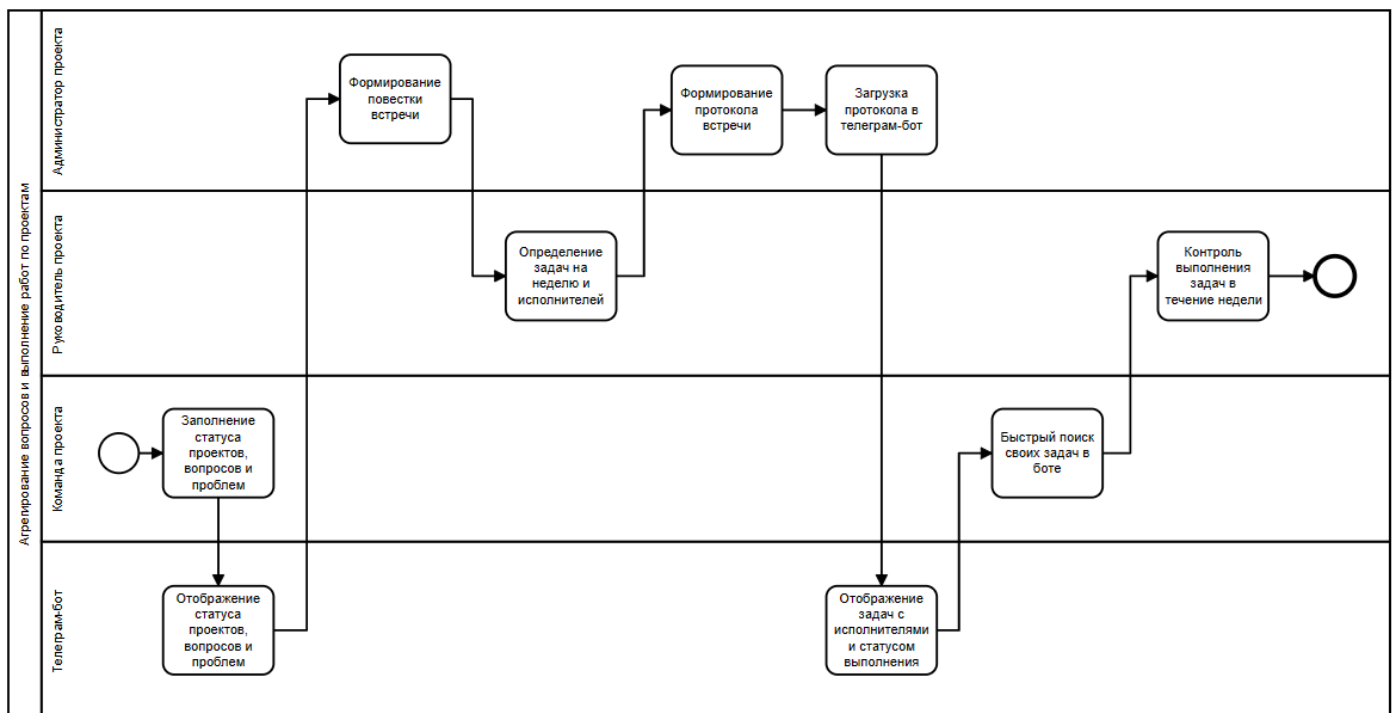
Статья расходов	Сумма (руб.)
ФОТ (ЗП + страховые отчисления 30%)	149 094
Внешний консалтинг	50 000
Хостинг (сервер)	500
Резервный фонд (10%)	19 959
ИТОГО:	219 553

## Модели бизнес-процесса (AS IS и TO BE)

Процесс агрегирования вопросов и выполнения задач AS IS.



Процесс агрегирования вопросов и выполнения задач TO BE.



## Реестр требований

### Функциональные требования

Загрузка документов и извлечение информации по проекту:

- Пользователи должны иметь возможность загружать документы в формате .doc/.docx через интерфейс бота.
- Бот должен распознавать текстовое содержимое загруженных документов и анализировать его.
- Бот должен извлекать информацию из загруженных документов, связанную с проектом, (вопросы, события, задачи и ответственные лица).

### Агрегация вопросов:

- Бот должен иметь возможность принимать вопросы от участников проекта через текстовые сообщения.
- Вопросы должны быть структурированы по темам или категориям, чтобы облегчить их дальнейшее управление.
- Бот должен автоматически агрегировать вопросы и создавать удобный список для дальнейшего рассмотрения.

### Управление календарем:

- Бот должен иметь доступ к календарю руководителя или интегрироваться с его календарным приложением.
- Должна быть возможность добавлять мероприятия, встречи, сроки и другие события в календарь через бота.
- Бот должен отправлять уведомления о предстоящих событиях и напоминания о них.

### Управление задачами:

- Бот должен иметь функционал для создания и отслеживания задач, связанных с проектом.
- Должна быть возможность назначать ответственных за задачи и устанавливать сроки выполнения.
- Бот должен отправлять уведомления о предстоящих задачах и их статусе.
- Пользователи могут добавлять события, встречи и дедлайны в календарь через бота.
- Бот должен автоматически отправлять уведомления о предстоящих событиях.

### Отображение списка задач:

- Пользователи могут просматривать список текущих задач через бота.
- Бот должен показывать статус каждой задачи и информацию о назначенном ответственном лице.

## **Нефункциональные требования**

### Интерфейс пользователя:

- Бот должен иметь интуитивно понятный и легкий в использовании интерфейс для участников проекта.
- Должны быть предоставлены команды или кнопки для выполнения основных функций, таких как добавление вопросов, событий или задач.

### Безопасность:

- Бот должен обеспечивать конфиденциальность и безопасность данных, передаваемых через него.
- Должны быть реализованы меры защиты от несанкционированного доступа к информации о проекте или календарю.

### Интеграция с другими сервисами:

- Бот должен иметь возможность интеграции с другими сервисами, такими как Google Календарь, Trello, Asana и другими инструментами управления проектами.

### Пользовательская настройка:

- Бот должен предоставлять возможность настройки уведомлений и предпочтений пользователей, чтобы каждый участник проекта мог настроить его под свои потребности.

### Логирование и аналитика:

- Бот должен вести лог действий пользователей и предоставлять аналитическую информацию о процессе работы над проектом и его текущем состоянии.
- Аналитика поможет выявить узкие места в процессе управления проектом и принять меры для их улучшения.

### Масштабируемость:

- Бот должен быть способен масштабироваться вместе с увеличением числа пользователей и объема данных.
- Архитектура бота должна быть гибкой и масштабируемой.

### UML-диаграммы