***Календарный план проекта и распределение задач по исполнителям***

|  |  |  |
| --- | --- | --- |
| Цель | Исполнитель | Дата |
| **Продумать алгоритм обработки заявки** | Марчёнок, Емельянов, Подолец | до 20.11.19 |
| **Спроектировать БД** | Марчёнок, Емельянов, Подолец | до 26.11.19 |
| **Оформление сайта** |  | до 5.12.2019 |
| Проработка дизайна сайта | Марчёнок, Подолец | до 26.11.2019 |
| Верстка сайта | Подолец | до 5.12.2019 |
| **Реализация прототипа** |  | до 15.12.2019 |
| Написание модели данных | Марчёнок | до 29.11.2019 |
| DAO | Емельянов, Подолец | до 5.12.2019 |
| Реализация логики сайта | Марчёнок, Емельянов | до 13.12.2019 |
| Реализация телеграм-бота | Емельянов | до 15.12.2019 |
| **Тестирование прототипа** |  | до 20.12.2019 |
| Unit-тесты | Марчёнок, Емельянов, Подолец | до 20.12.2019 |
| Мануальное тестирование | Подолец | до 20.12.2019 |

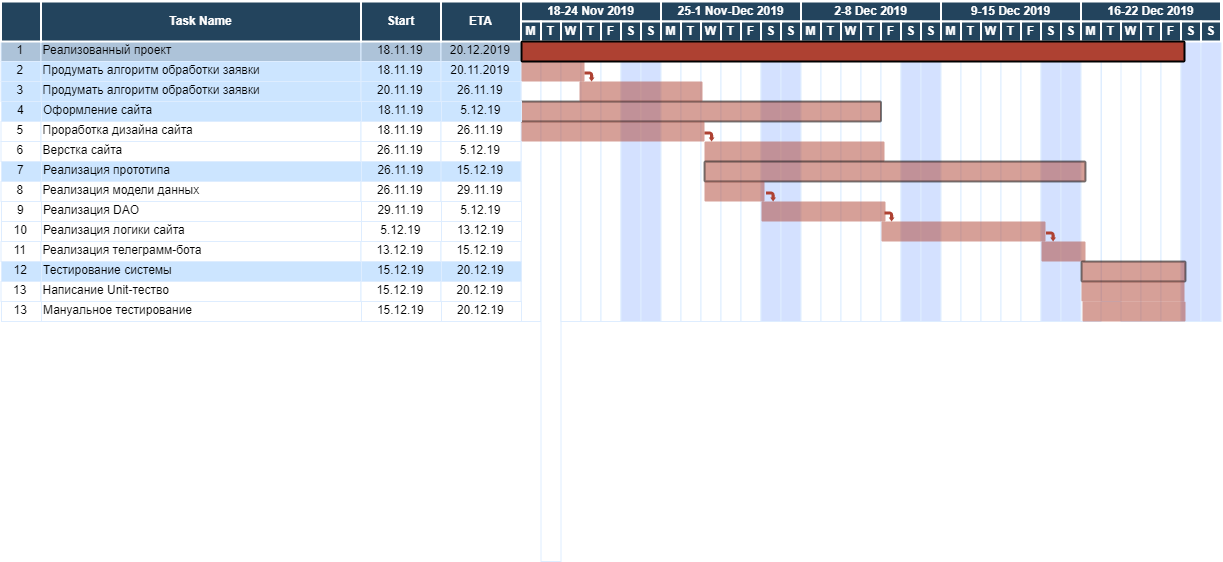


Рисунок 1. Диаграмма Ганта

***State of Art***

*Обзор актуальных технологий*

--

*Анализ аналогов*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сайт\Критерии | Удобство | Функционал | Дизайн | Актуальность | Оценка |
| [ЯГражданин](https://igrajdanin.ru/) | 6 | 9 | 8 | 8 | 7,75 |
| [ЭкоПроблемы](https://ecoproblems.ru/) | 9 | 8 | 9 | 7 | 8,25 |
| [РосЯма](https://rosyama.ru/) | 7 | 7 | 7 | 8 | 7,25 |

Выводы: ЭкоПроблемы является наиболее привлекательным и удобным аналогом.

**Анализ предметной области**

*Описание предметной области*

Программная система по привлечению внимания к проблемам города, которые волнуют граждан.

*Словарь предметной области*

Заявка – это заявление, заполненное в указанной форме, с просьбой опубликовать волнующую актера проблему.

Проблема – это проблема или инцидент, волнующие граждан и требующие решения

Новость – это значимое событие, которое произошло в городе за последнее время.

*Пользователи системы и роли*

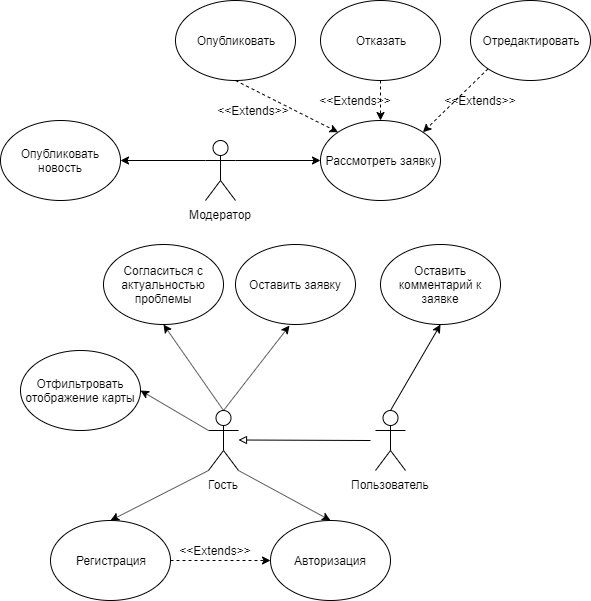
Гость может оставлять заявку, просматривать выложенные проблемы и новости, соглашаться с актуальностью проблемы.

Зарегистрированный пользователь может оставлять комментарии к заявкам, оставлять заявку, просматривать выложенные проблемы и новости, соглашаться с актуальностью проблемы.

Модератор рассматривает отправленные заявки, может выложить заявку или отказать.

Редактор выкладывает новости.

*Прецеденты*



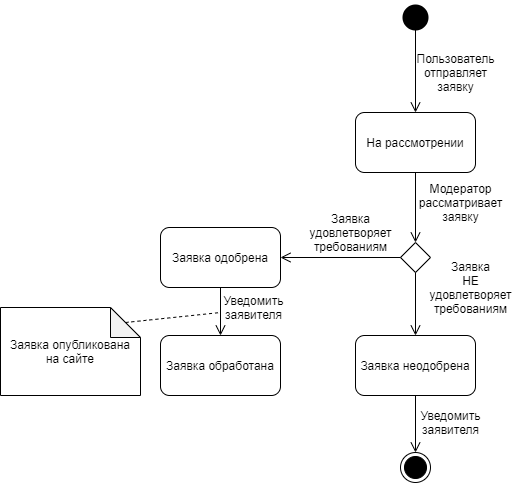
*Описание прецедентов*

Название: «Оставить заявку»  
Предусловие: -.  
Действующее лицо: Гость, зарегистрированный пользователь.  
Основной поток: Актер нажимает на кнопку «Сообщить о проблеме».  
Открывается форма для заполнения заявки.  
Актер заполняет поля «название», «адрес», «категория», «описание», «фотография».   
Актер отправляет заявку.   
Альтернативный поток: -.  
Постусловие: модератору приходит уведомление об отправленной заявке.

Название: «Оставить комментарий к заявке»  
Предусловие: -.  
Действующее лицо: Зарегистрированный пользователь.  
Основной поток: Актер выбирает проблему, выложенную на сайте.  
Открывается форма проблемы.   
В поле «Оставить комментарий» актер пишет комментарий.  
Актер нажимает кнопку «Отправить комментарий»  
Альтернативный поток: -.  
Постусловие: -.

Название: «Согласиться с актуальностью проблемы»  
Предусловие: -.  
Действующее лицо: Гость, зарегистрированный пользователь.  
Основной поток: Актер выбирает проблему, выложенную на сайте.  
Открывается форма проблемы.  
Актер нажимает кнопку «Согласиться с актуальностью проблемы».  
Альтернативный поток: -.  
Постусловие: -.

*Описание документов/сущностей предметной области*



*Анализ бизнес-процессов*

*Функциональные требования к системе*

* 1. Система должна создавать заявку, отправлять на рассмотрение, опубликовывать заявку.
  2. Система должна отправить пользователю уведомление о том, что его заявка на рассмотрении/опубликована/отклонена.
  3. Система должна создавать нового пользователя, редактировать или **удалять (?)** пользователя.
  4. Система должна фильтровать проблемы по категориям.
  5. Система должна выводить топ актуальных проблем.
  6. Система должна отображать новости указанного города.
  7. Система должна поддерживать комментарии к проблемам.
  8. Система должна поддерживать отправку заявок через приложение *«Telegram»*.
  9. Система должна поддерживать отображение местонахождения проблем на карте.
  10. Система должна поддерживать различные категорий проблем.
  11. Система должна поддерживать рейтинговую таблицу проблем по количеству голосов пользователей.
  12. Система должна возможность комментирования проблем только от зарегистрированных пользователей.

*Нефункциональные требования к системе*

1. Пользовательский интерфейс должен быть удобным, чтобы каждый пользователь мог интуитивно понять на какую кнопку нажать, чтобы совершить нужное действие.
2. Поддержка клиент-серверной архитектуры.

**Макет программной системы**

*Структура интерфейса, навигация*

*Эскизы интерфейса пользователя*

