**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

УТВЕРЖДАЮ

Зам. директора по УПР

С.Ю. Назаров\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_20\_\_\_г.

**ОТЧЕТ**

**по производственной практике (по профилю специальности)**

По профессиональному модулю УП 02.01 - «Разработка модулей программного обеспечения для компьютерных систем»

(код и наименование)

Специальность09.02.07 Информационные системы и программирование

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Гильманова Лилия Русланован

(Фамилия, имя, отчество)

Место прохождения практики: Энгельса 23, ИСПО

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

Руководитель практики

от учебной организации Прокофьев А.А.

(подпись) (расшифровка подписи)

Руководитель с места

прохождения практики

(подпись) (расшифровка подписи)

Итоговая оценка по практике \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

М.П.

Санкт-Петербург

2024

**ЗАДАНИЕ**

**на учебную практику (по профилю специальности)**

По профессиональному модулю УП 02.01 - «Разработка модулей программного обеспечения для компьютерных систем»

(код и наименование)

Специальность09.02.07 Информационные системы и программирование

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Гильманова Лилия Руслановна

(Фамилия, имя, отчество)

Место прохождения практики: Энгельса 23, ИСПО

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

**Виды работ, обязательные для выполнения** *(переносится из программы, соответствующего ПМ):*

1. Разработка спецификаций отдельных компонент
2. Конструирование пользовательского интерфейса
3. Проектирование и разработка модулей программного обеспечения
4. Модульное тестирование
5. Документирование программных продуктов

**Индивидуальное задание:**

Задание выдал «9» декабря 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Прокофьев А.А.

(подпись)  (Ф.И.О.)

Задание получил «9» декабря 2024 г. \_\_\_\_\_\_\_\_\_\_\_\_ Гильманова Л.Р.(подпись)  (Ф.И.О.)

СОГЛАСОВАНО

Председатель ПЦК

\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_ 202\_\_ г.

**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

**ДНЕВНИК**

**прохождения производственной практики**

**(по профилю специальности)**

По профессиональному модулю УП 02.01 - «Разработка модулей программного обеспечения для компьютерных систем»

(код и наименование)

Специальность09.02.07 Информационные системы и программирование

(код и наименование специальности)

Студент(ка) 4 курса 42919/4 группы

Гильманова Лилия Руслановна

(Фамилия, имя, отчество)

Место прохождения практики Энгельса 23, ИСПО

(наименование и адрес организации)

Период прохождения практики

с «9» декабря 2024 г. по «21» декабря 2024 г.

Руководитель практики

от учебной организации Прокофьев А.А.

(подпись) (расшифровка подписи)

Руководитель с места

прохождения практики .

(подпись) (расшифровка подписи)

Итоговая оценка по практике \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

М.П.

Санкт-Петербург

2024

**Содержание дневника**

|  |  |  |
| --- | --- | --- |
| Дата | Виды выполненных работ и заданий по программе практики | Подпись руководителя практики |
| **1** | **2** | **3** |
| 18.11.24 | Постановка задачи. Изучение предметной области. Анализ требований. Изучение аналогов программного продукта. |  |
| 19.11.24 | Разработка технического задания. |  |
| 20.11.24 | Анализ методов решения. Анализ программных средств |  |
| 21.11.24 | Разработка UML-диаграмм с использованием графических редакторов. |  |
| 22.11.24 | Разработка спецификаций для структурных единиц. Составление алгоритмов структурных единиц. |  |
| 23.11.24 | Разработка макета пользовательского интерфейса в различных режимах работы. |  |
| 25.11.24 | Конструирование и настройка пользовательского интерфейса. Работа над дизайном пользовательского интерфейса. |  |
| 26.11.24 | Разработка программных модулей в первой итерации. Кодирование структурных данных. |  |
| 27.11.24 | Откладка и тестирование структурных единиц с использованием тест-кейсов. |  |
| 28.11.24 | Разработка программных модулей во второй итерации. Расширение и добавление функций. |  |
| 29.11.24 | Создание справочной системны приложения. |  |
| 30.11.24 | Откладка и тестирование единиц второй итерации с использованием тест-кейсов. |  |
| 02.12.24 | Рефакторинг и оптимизация программы. |  |
| 03.12.24 | Разработка методики испытаний. Откладка и тестирование программного продукта на уровне модуля. |  |
| 04.12.24 | Испытания программы. Тест готовой системы. Тест приемлемости. |  |
| 06.12.24 | Определение перечня программных документов. Изучение ГОСТ в ЕСПД. Документирование программного обеспечения в соответствии с Единой системой программной документации, с использованием инструментальных средств. |  |
| 07.12.24 | Настройка и выпуск приложения. Инсталляция приложения. |  |

Содержание

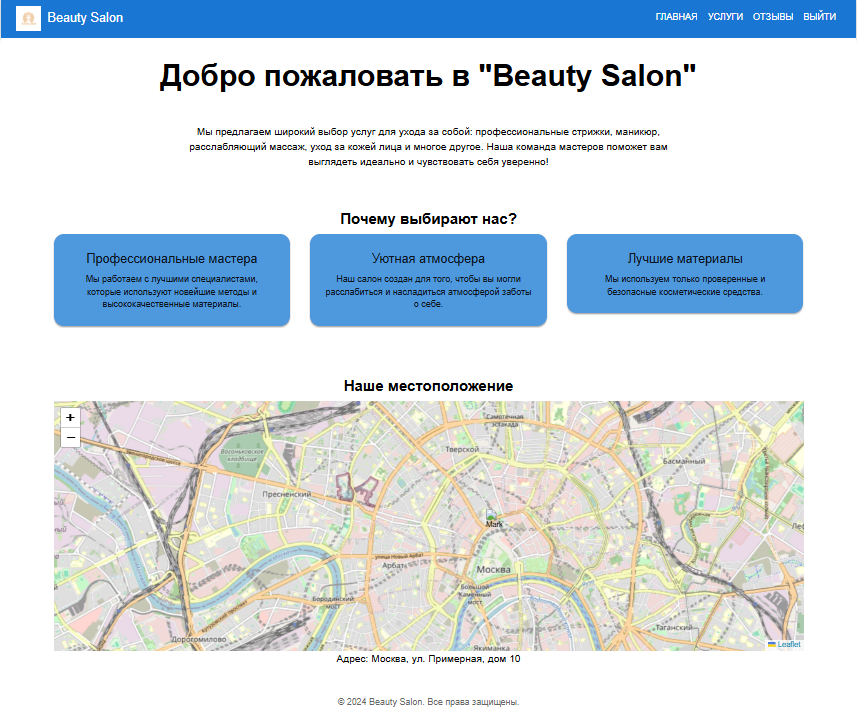
[Задание 1. Разработка технического задания 11](#_Toc185330379)

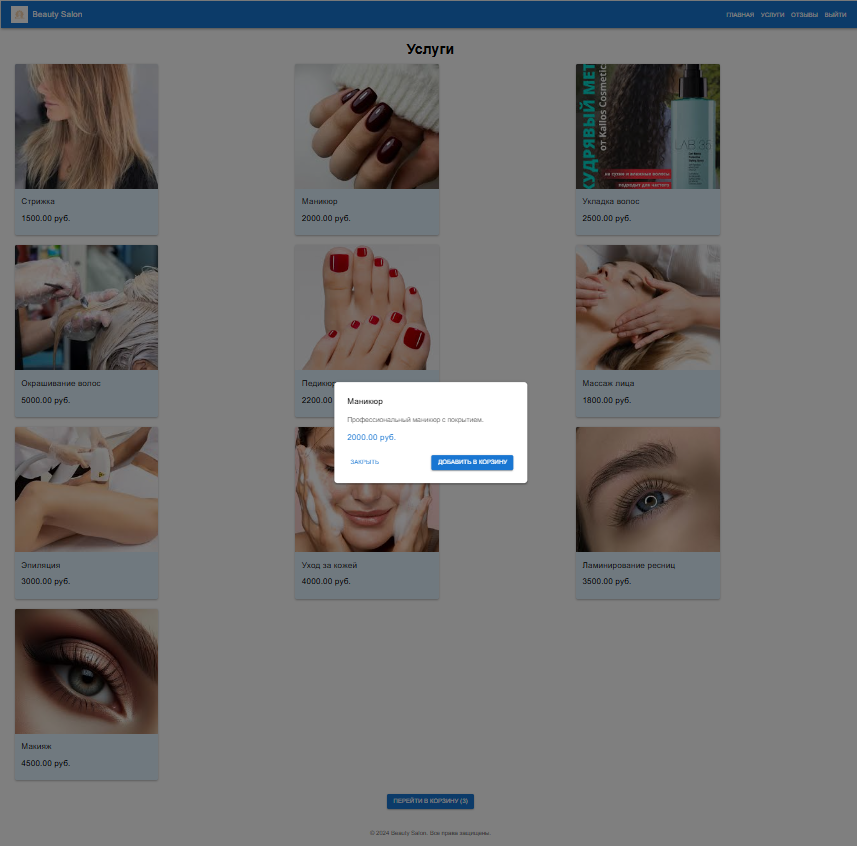
[Задание 2. Разработка алгоритмов и диаграмм 14](#_Toc185330380)

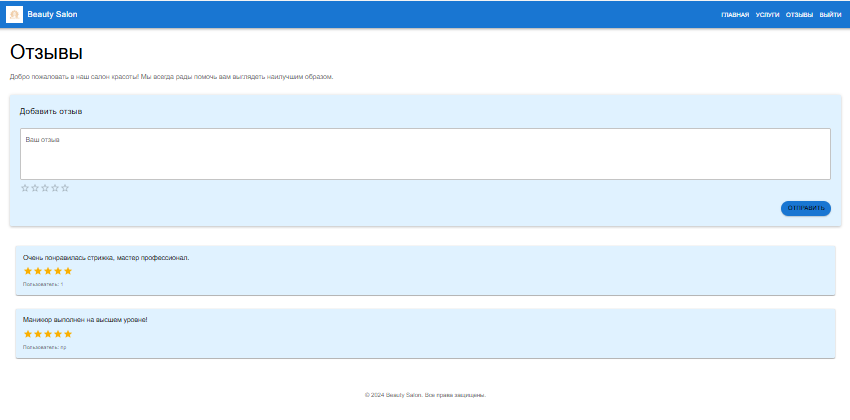
[Рисунок 4. ER-диаграмма базы данных. 17](#_Toc185330381)

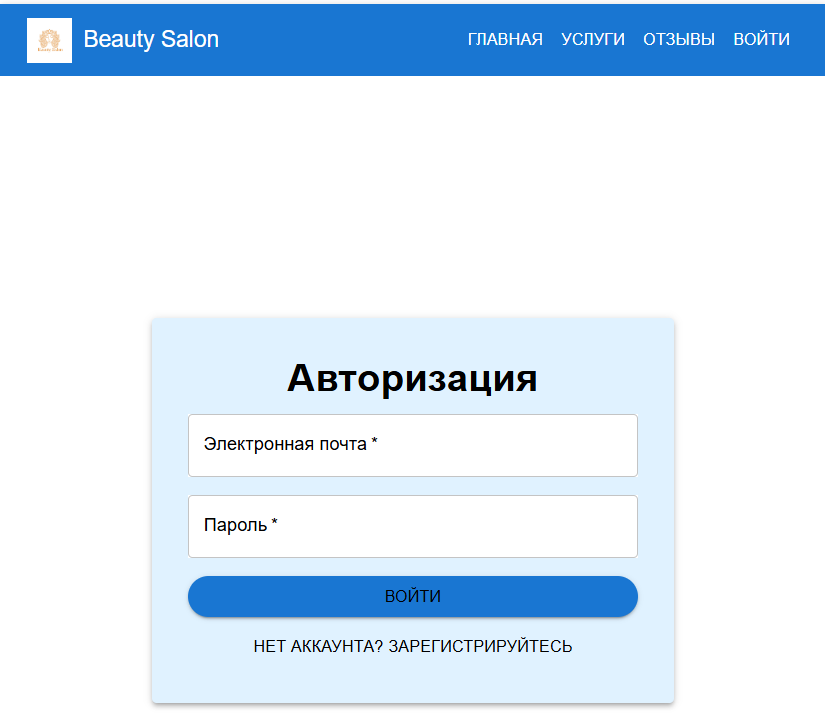
[Задание 3. Разработка макетов программы 17](#_Toc185330382)

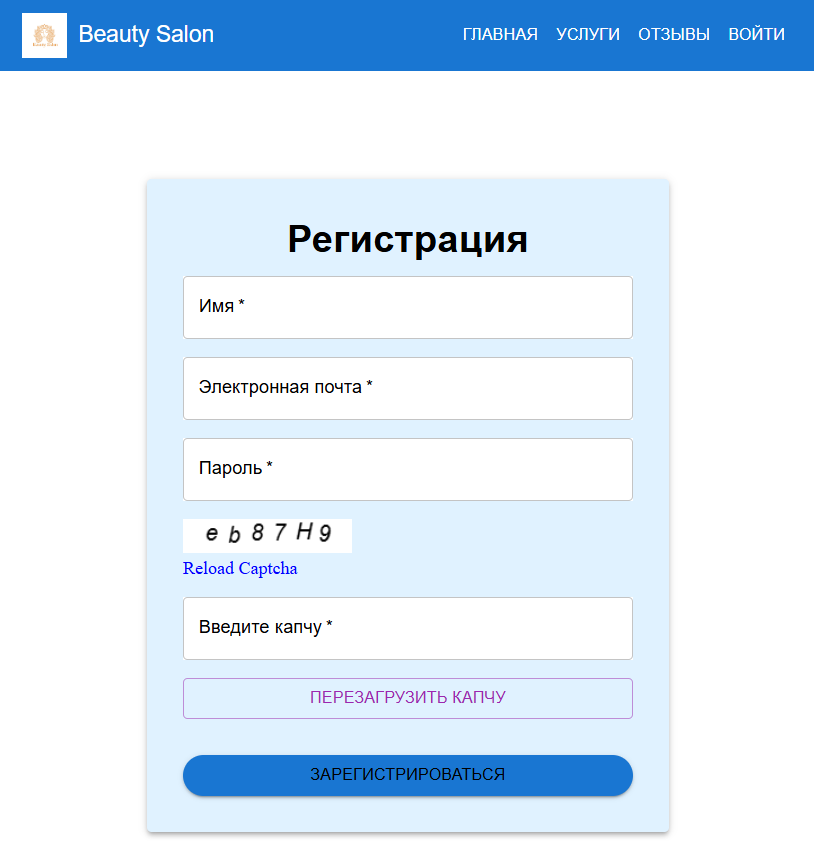
[Задание 4. Разработка программы 25](#_Toc185330383)

[ 25](#_Toc185330384)

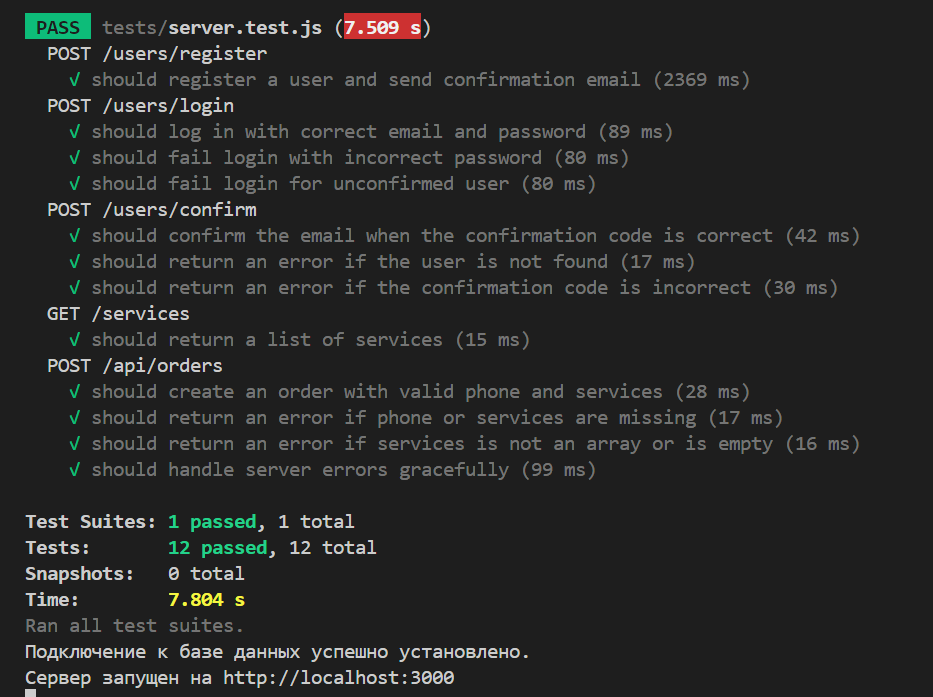
[ 26](#_Toc185330385)

[ 27](#_Toc185330386)

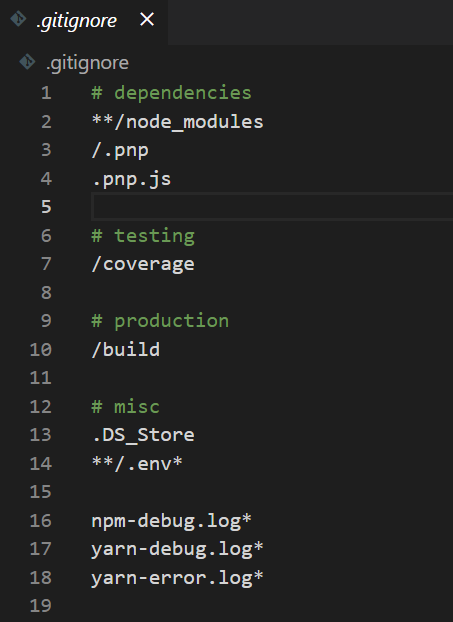
[ 27](#_Toc185330387)

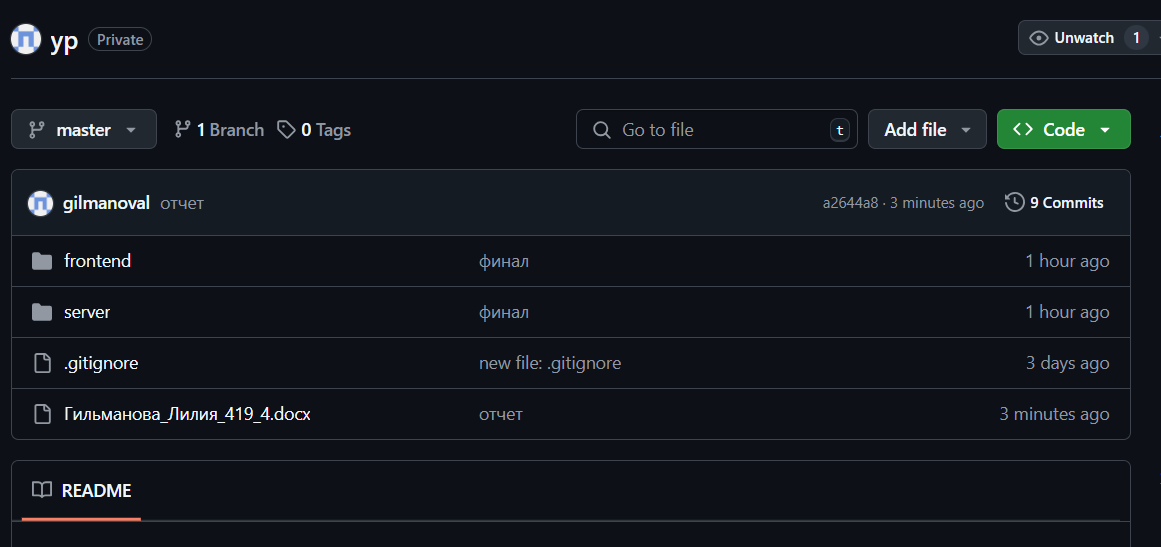
[ 28](#_Toc185330388)

[Задание 5. Тестирование 28](#_Toc185330389)

[ 39](#_Toc185330390)

[Задание 6. GitHub. Оценка проекта 39](#_Toc185330391)

[ 40](#_Toc185330392)

[ 41](#_Toc185330393)

[Рисунок 37. Git-репозиторий. 41](#_Toc185330394)

Задание 1. Разработка технического задания

Вариант «Салон красоты»

Описание предметной области

Основной целью данного модуля является автоматизация процессов учета и управления записью клиентов на предоставление услуг в салоне красоты. Модуль помогает сотрудникам салона эффективно и прозрачно обрабатывать заявки, управлять расписанием мастеров и отслеживать текущее состояние заказов.

Разработка программного модуля для учета заявок на услуги салона красоты — это создание программного продукта, который позволяет салонам красоты оптимизировать процесс взаимодействия с клиентами, улучшить планирование работы и повысить удовлетворенность клиентов.

Задача разработки такого модуля состоит в создании удобного и функционального инструмента для управления заявками, отслеживания их статуса и организации рабочего времени мастеров. Это помогает минимизировать ошибки при записи, снизить время на обработку запросов и обеспечить высокий уровень сервиса.

Программный модуль позволяет салону красоты повысить эффективность работы, сократить затраты времени на ручное управление процессами, а также создавать условия для предоставления качественного обслуживания клиентам.

Основные функции и возможности модуля

Запись на услуги:

Клиенты могут оставлять заявки на предоставление услуг (например, стрижка, окрашивание, маникюр и др.), указывая предпочитаемую дату, время и мастера. Заявка содержит информацию о клиенте (ФИО, номер телефона) и подробности услуги.

Регистрация заявок:

Оператор или администратор салона регистрирует заявки в системе, присваивая им уникальный номер. Это позволяет отслеживать каждую заявку и контролировать её выполнение.

Обработка заявок:

Система помогает распределять заявки между мастерами, назначать время и уведомлять клиентов. Возможно изменение записи в случае изменений пожеланий клиента или занятости мастера.

Управление расписанием мастеров:

Система отображает расписание сотрудников, помогает планировать их рабочее время и избегать пересечения заявок.

Отчетность и анализ:

Возможность формирования отчетов по выполненным услугам, времени работы мастеров, количеству клиентов, популярным услугам. Это помогает оптимизировать процессы салона и выявлять ключевые точки роста.

Техническое задание

1. Общие сведения
   1. Наименование проекта: Разработка программного модуля для учета заявок на услуги салона красоты.
   2. Заказчик: Салон красоты «Эстетика».
   3. Исполнитель: ИСПО СПБПУ.
2. Функциональные требования
   1. Возможности работы с заявками:

- Добавление новой заявки с параметрами:

- Уникальный номер;

- Дата и время записи;

- Услуга;

- ФИО клиента;

- Контактный номер;

- Назначенный мастер;

- Редактирование заявки:

- Изменение услуги, времени, мастера.

* 1. Управление расписанием мастеров:

- Отображение текущего расписания;

- Поиск свободного времени для записи.

* 1. Учет предоставленных услуг:

- Возможность отслеживания затрат материалов;

- Ведение истории выполненных работ.

* 1. Формирование отчетности:

- Количество заявок по мастерам и услугам;

- Среднее время выполнения услуг;

- Доходы от предоставленных услуг за период.

1. Нефункциональные требования

- Адаптивный дизайн (в google chrome на ПК и в google chrome на телефоне android версии 13+ должен быть без ошибок)

* 1. Безопасность:

- Доступ к системе по логину и паролю (логин и пароль хранится в зашифрованном виде в базе данных);

- Разграничение прав пользователей (администратор, мастер, клиент).

* 1. Удобство использования:

- Интуитивный интерфейс (при наведении на элементы меняется курсор и цвет, обозначая возможность нажать на него);

* 1. Производительность:

- Обеспечение времени отклика меньше 1 секунды;

- Поиск записей по базе менее 1 секунды.

1. Требования к реализации
   1. Язык программирования: JavaScript React 17, Node.js .
   2. СУБД: Postgres.
2. Требования к документации
   1. Техническое задание.
   2. Руководство пользователя.
3. Руководство по стилю
   1. Шрифт: Стандартный, читаемый (Arial, Calibri).
   2. Цветовая схема: Светлая, с акцентами на ключевых элементах интерфейса. #ff9aff, #b4ffff, #000000, #b40068

Задание 2. Разработка алгоритмов и диаграмм

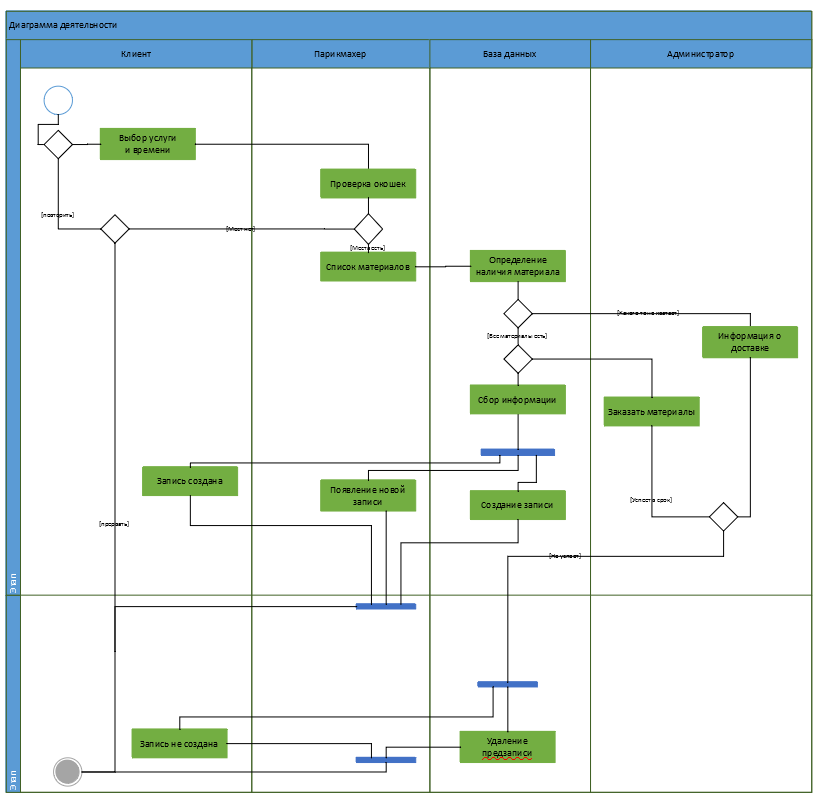


Рисунок 1. Диаграмма активности.

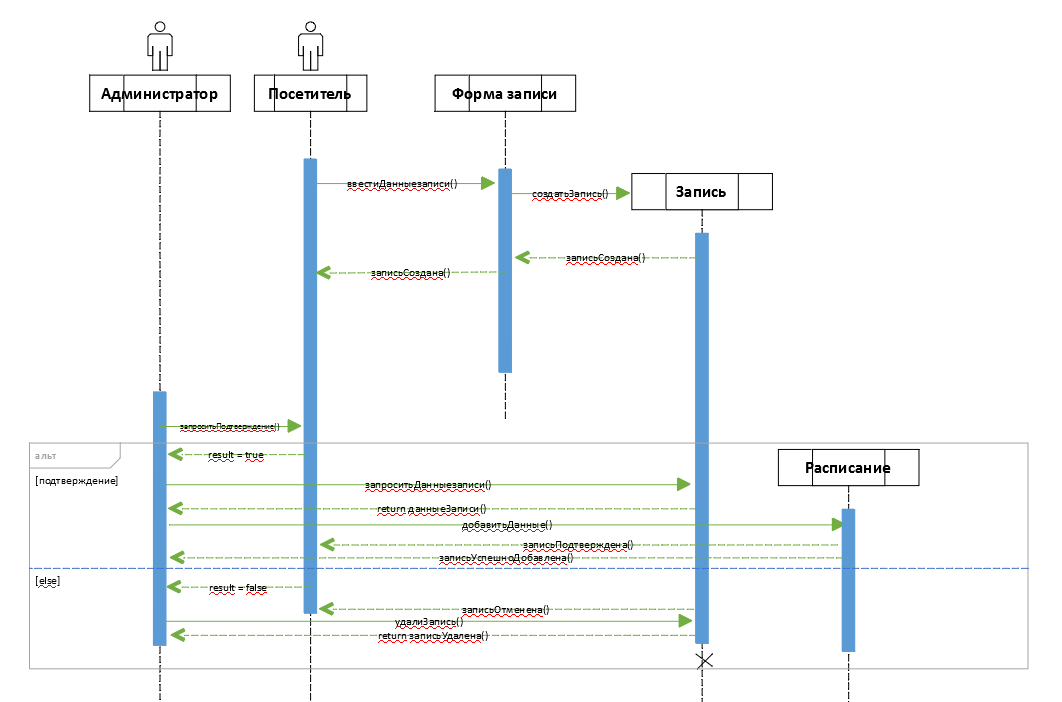


Рисунок 2. Диаграмма последовательности.

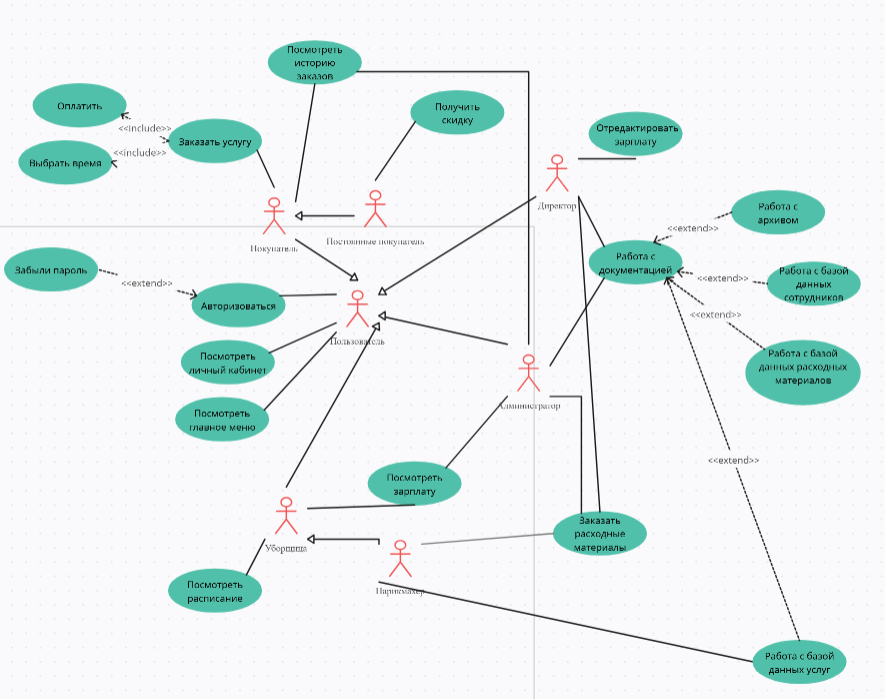


Рисунок 3. Диаграмма вариантов использования.

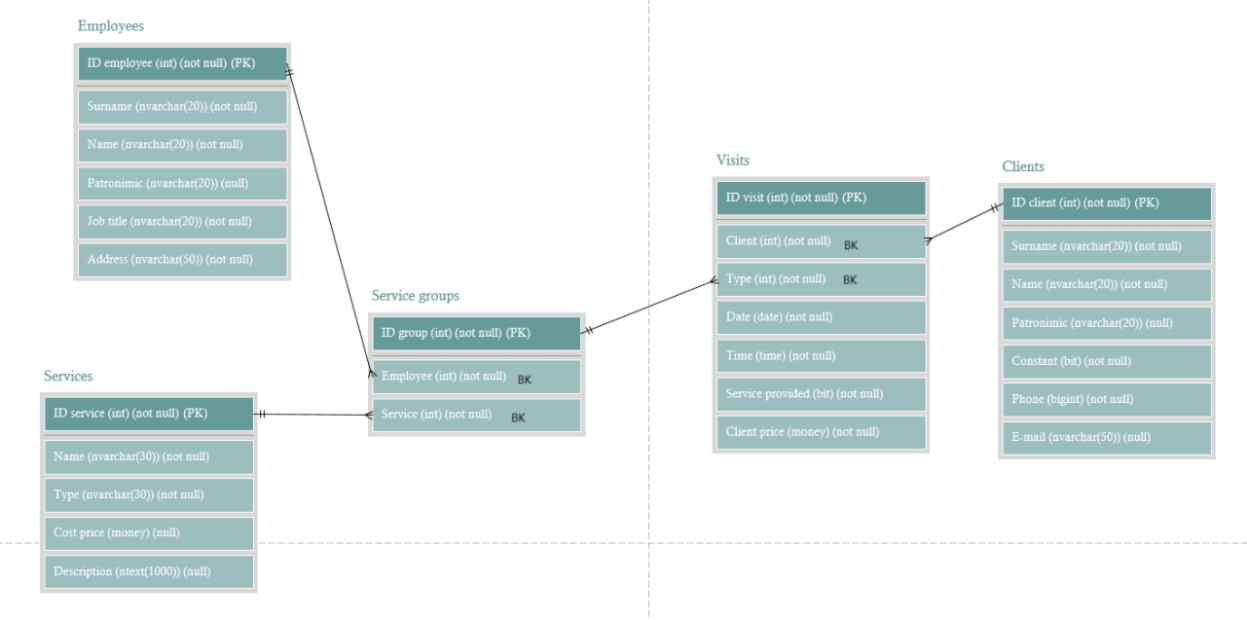


Рисунок 4. ER-диаграмма базы данных.

Задание 3. Разработка макетов программы

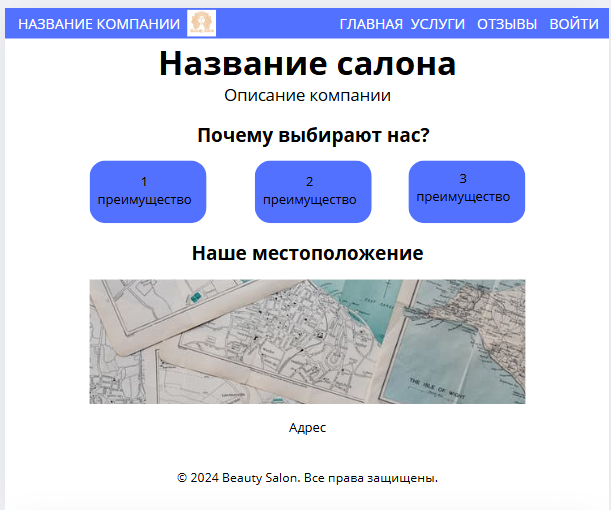


Рисунок 5. Mockup макет страницы «Главная».

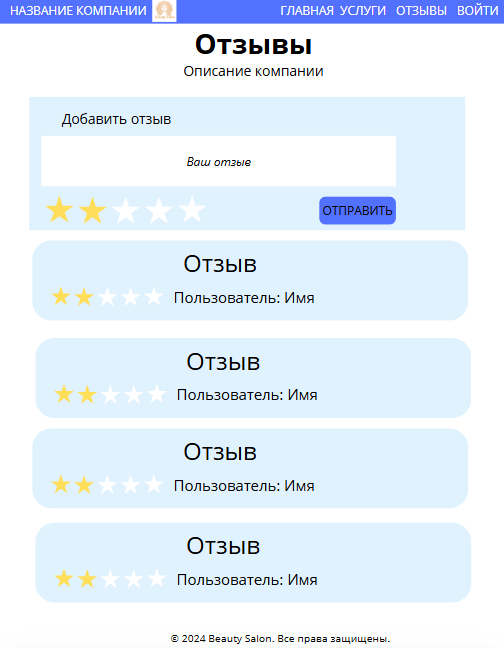


Рисунок 6. Mockup макет страницы «Отзывы».



Рисунок 7. Mockup макет страницы «Услуги».



Рисунок 8. Mockup макет страницы «Администрирование».

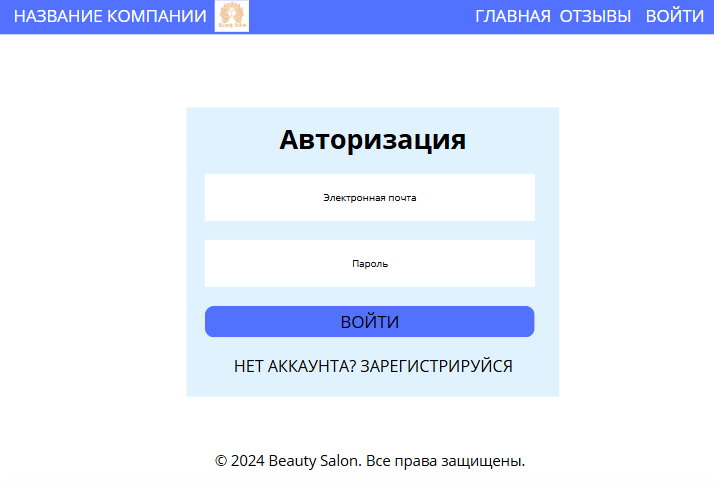


Рисунок 9. Mockup макет страницы «Авторизация».

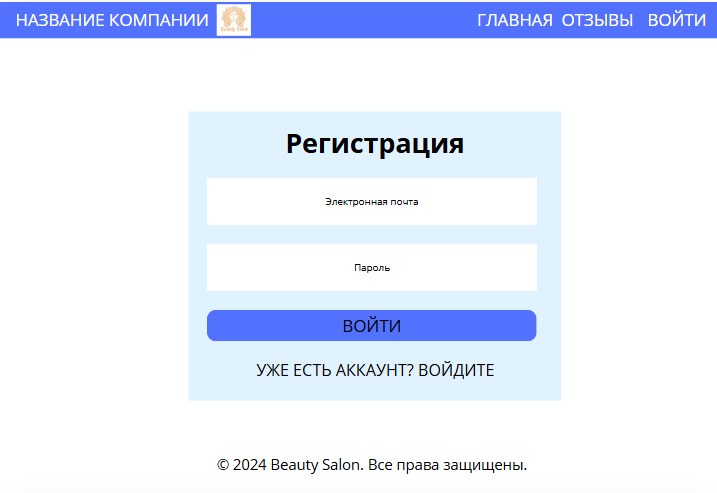


Рисунок 10. Mockup макет страницы «Регистрация».

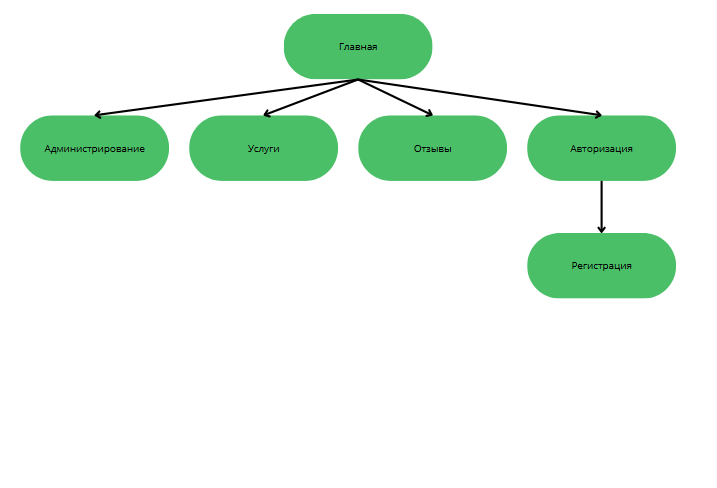


Рисунок 11. Карта навигации.



Рисунок 12. Wireframe макет страницы «Главная».

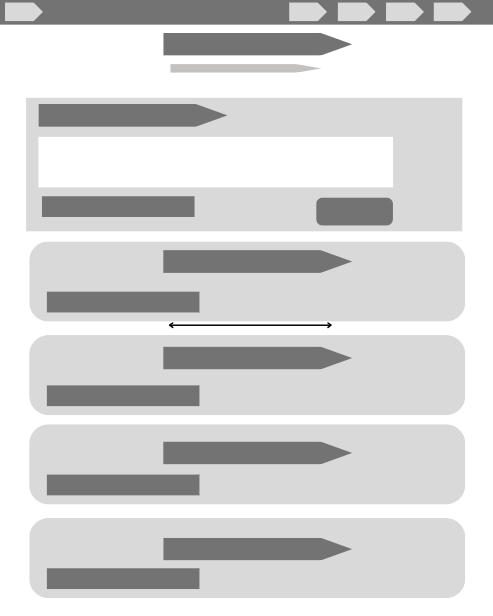


Рисунок 13. Wireframe макет страницы «Отзывы».

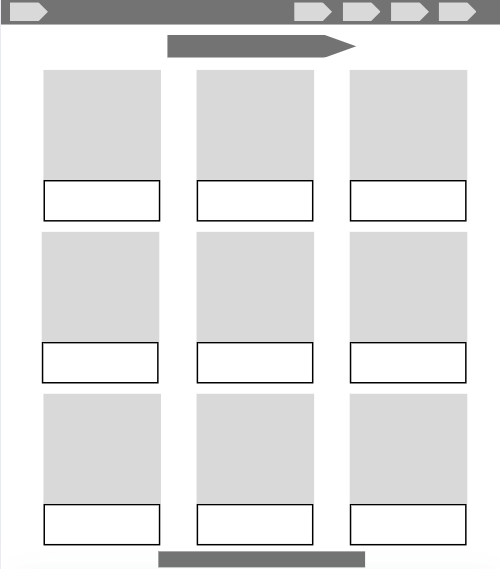


Рисунок 14. Wireframe макет страницы «Услуги».

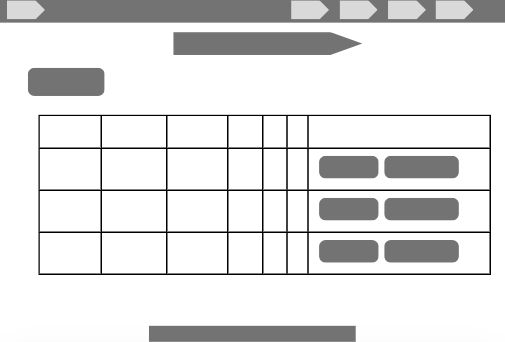


Рисунок 15. Wireframe макет страницы «Администрирование».

Задание 4. Разработка программы

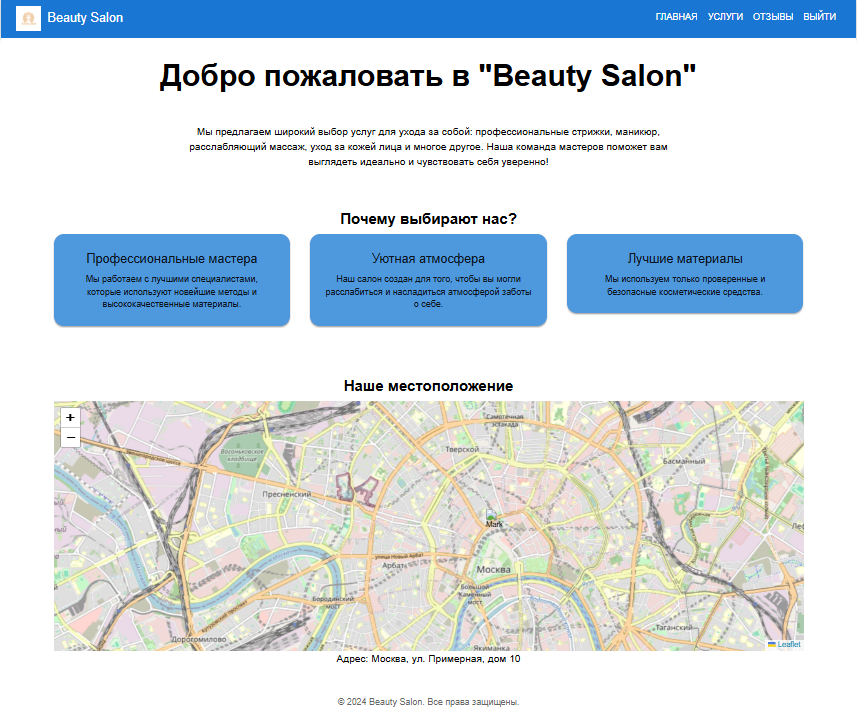


Рисунок 16. Страница «Главная».

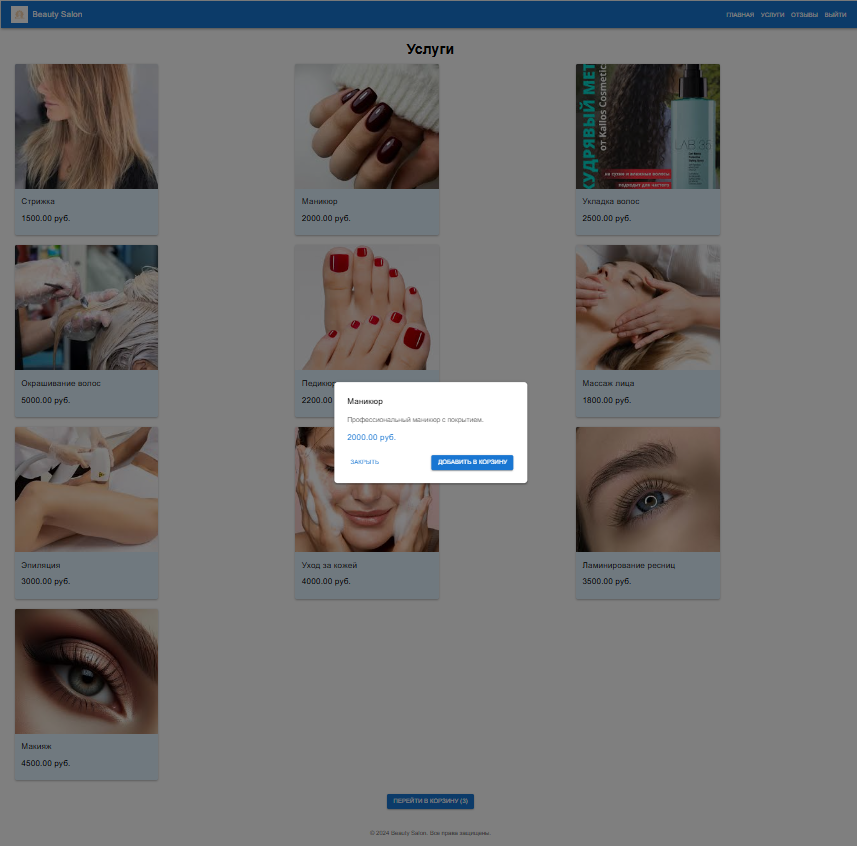


Рисунок 17. Страница «Услуги».

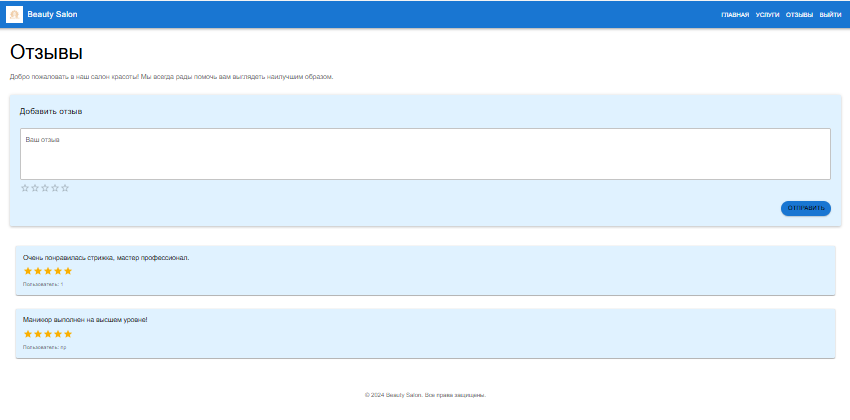


Рисунок 18. Страница «Отзывы».

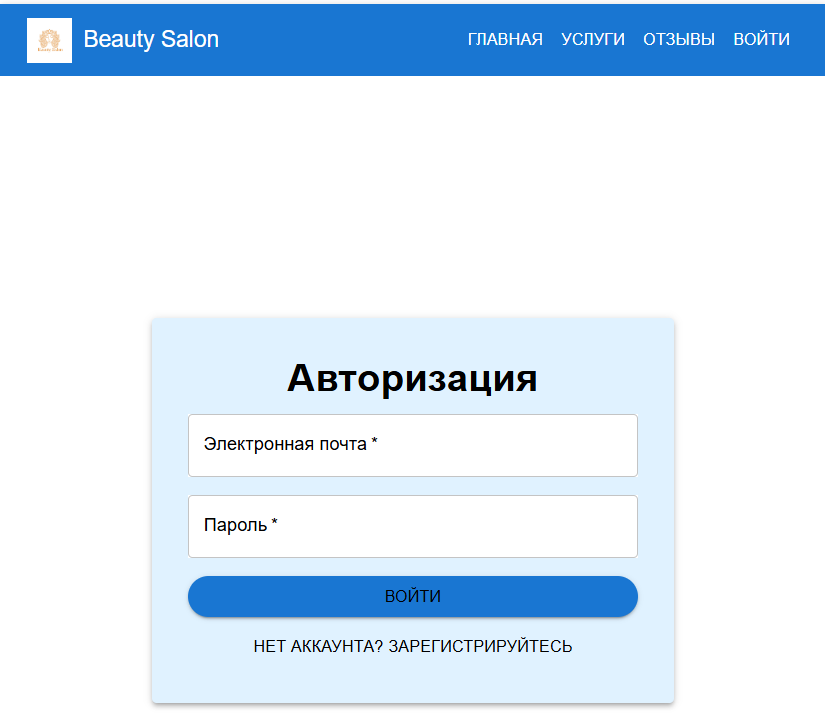


Рисунок 19. Страница «Авторизация».

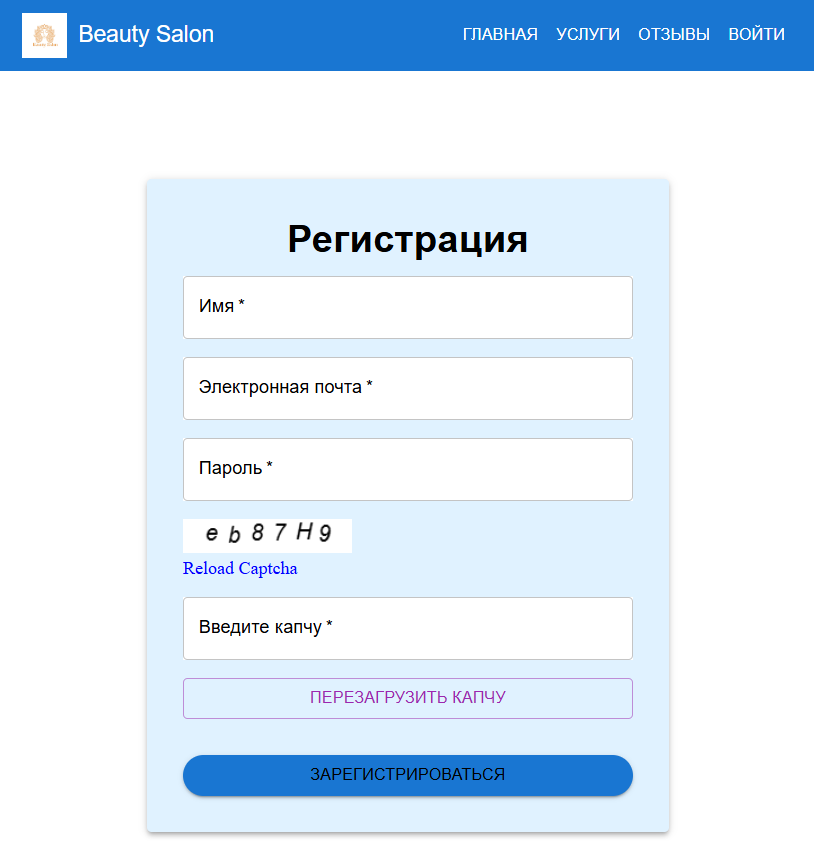


Рисунок 20. Страница «Регистрация».

Задание 5. Тестирование

Аннотация теста:

|  |  |
| --- | --- |
| **Название проекта** | Beauty Salon |
| **Рабочая версия** | 1.0 |
| **Имя тестирующего** | Гильманова Л.Р. |
| **Дата(ы) теста** | 16.12.2024 |

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_1 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | registeringNewUserWithValidData  (Регистрация нового пользователя с валидными данными) |
| **Краткое изложение теста** | Проверка успешной регистрации нового пользователя на сайте салона красоты при введении всех корректных данных. |
| **Этапы теста** | 1. Открыть сайт по http://localhost:3001/ 2. Перейти на страницу регистрации (/register). 3. Ввести в поля формы:  * Имя: Иван * Электронная почта: gilmanova3004@gmail.com * Пароль: StrongPassword123 * Введите капчу: ввести корректную капчу  1. Нажать кнопку "Зарегистрироваться". 2. В модальном окне с информацией, что код отправлен на почту нажать “ОК” 3. Ввести в поле“Код подтверждения”: шестизначный код который пришел на почту 4. Нажать кнопку “Подтвердить почту” |
| **Тестовые данные** | Имя: Иван  Электронная почта: gilmanova3004@gmail.com  Пароль: StrongPassword123 |
| **Ожидаемый результат** | Появляется сообщение: "Почта подтверждена! Теперь можете войти."  Перенаправляет на страницу /login |
| **Фактический результат** | Появляется сообщение: "Почта подтверждена! Теперь можете войти." (Рисунок 5)  Перенаправляет на страницу /login |
| **Статус** | Зачёт (Рисунок 21-25) |
| **Предварительное условие** | Пользователь не должен быть зарегистрирован в системе.  Пользователь не должен быть авторизован.  Иметь доступ к электронной почте |
| **Постусловие** | В базе данных user появляется новый пользователь |
| **Примечания/комментарии** | В электронную почту нужно ввести такую, к которой есть доступ. |

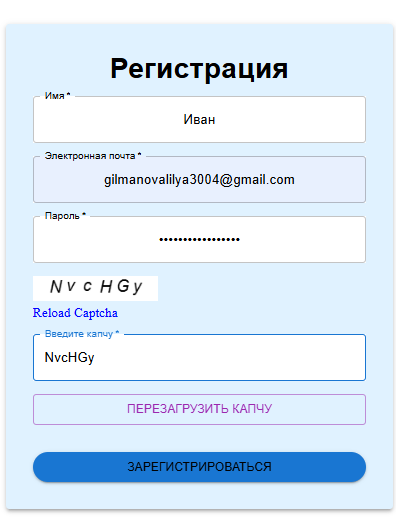


Рисунок 21.

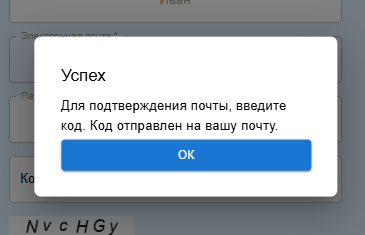


Рисунок 22.

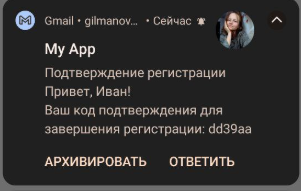


Рисунок 23.

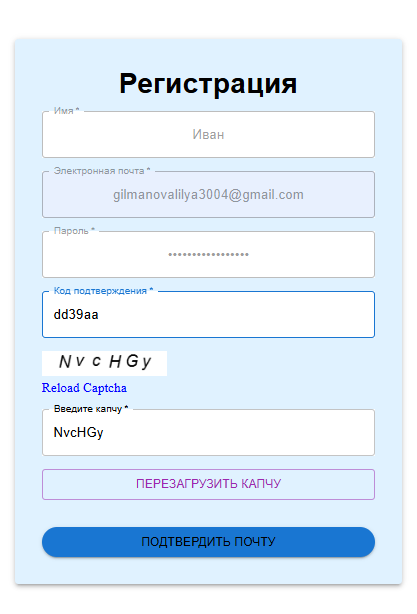


Рисунок 24.

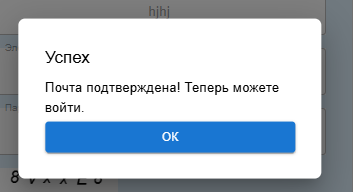


Рисунок 25.

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_2 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | attemptedRegisterWithInvalidIEmailAddress  (Попытка регистрации с невалидным email) |
| **Краткое изложение теста** | Проверка ошибки при попытке регистрации с некорректным email (отсутствует @). |
| **Этапы теста** | 1. Открыть сайт по http://localhost:3001/ 2. Перейти на страницу регистрации (/register). 3. Ввести в поля формы:  * Имя: Анна * Электронная почта: gilmanova3004gmail.com * Пароль: StrongPassword123 * Введите капчу: ввести корректную капчу  1. Нажать кнопку "Зарегистрироваться". |
| **Тестовые данные** | Имя: Анна  Электронная почта: gilmanova3004gmail.com  Пароль: StrongPassword123 |
| **Ожидаемый результат** | Появляется сообщение:  Адрес электронной почты должен содержать символ “@”. В адресе “gilmanovalilya3004gmail.com” отсутствует символ “@” |
| **Фактический результат** | Появляется сообщение:  Адрес электронной почты должен содержать символ “@”. В адресе “gilmanovalilya3004gmail.com” отсутствует символ “@”(Рисунок 27) |
| **Статус** | Зачёт (Рисунок 26-27) |
| **Предварительное условие** | - |
| **Постусловие** | Пользователь не добавлен в базу данных. |
| **Примечания/комментарии** | - |

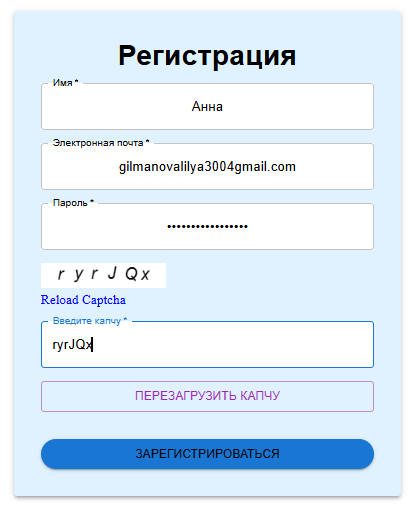


Рисунок 26.

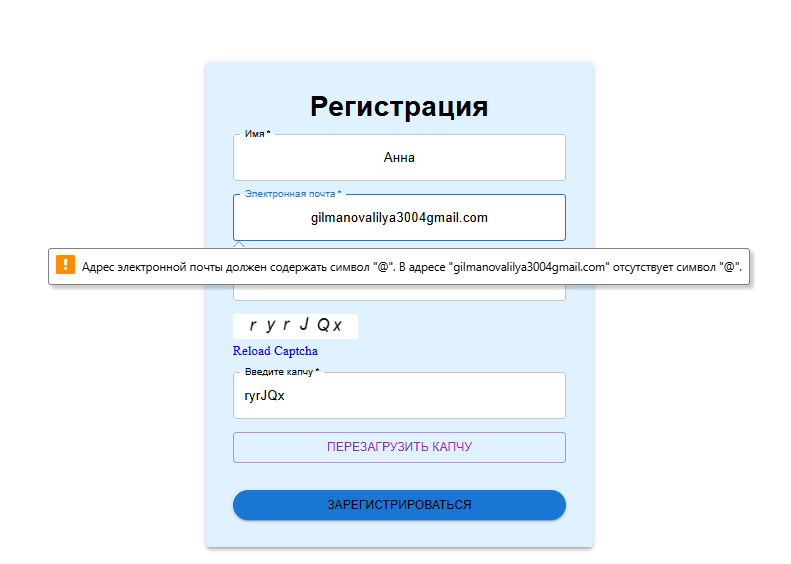


Рисунок 27.

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_3 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | attemptRegisterWithEmptyData  (Попытка регистрации с пустыми данными) |
| **Краткое изложение теста** | Проверка корректной обработки ошибки при попытке зарегистрировать нового пользователя с пустыми полями на странице регистрации. |
| **Этапы теста** | 1. Открыть сайт по http://localhost:3001/ 2. Перейти на страницу регистрации (/register). 3. Оставить все поля формы пустыми (Имя, Фамилия, Электронная почта, Пароль, Подтверждение пароля). 4. Нажать кнопку "Зарегистрироваться". |
| **Тестовые данные** | Имя: (пустое поле)  Электронная почта: (пустое поле)  Пароль: (пустое поле)  Введите капчу: (пустое поле) |
| **Ожидаемый результат** | У первого незаполненного поля появляется сообщение:  Заполните это поле |
| **Фактический результат** | У первого незаполненного поля появляется сообщение:  Заполните это поле |
| **Статус** | Зачёт (Рисунок 28) |
| **Предварительное условие** | - |
| **Постусловие** | Пользователь не добавлен в базу данных user. |
| **Примечания/комментарии** | - |

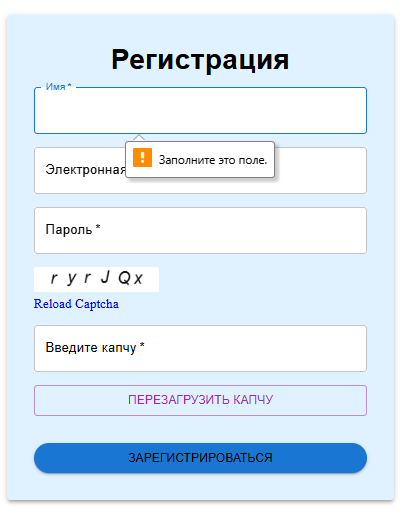


Рисунок 28.

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_4 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | attemptLogWithIncorrectPassword  (Попытка авторизации с неверным паролем) |
| **Краткое изложение теста** | Проверка того, что система корректно отклоняет авторизацию при введении неверного пароля. |
| **Этапы теста** | 1. Открыть сайт по http://localhost:3001/ 2. Перейти на страницу на страницу авторизации (/login). 3. Ввести в поля формы:   Электронная почта: gilmanovalilya3004@gmail.com  Пароль: WrongPassword123   1. Нажать на кнопку “Войти”. 2. Проверить, что система выводит ошибку. |
| **Тестовые данные** | Электронная почта: gilmanovalilya3004@gmail.com  Пароль: WrongPassword123 |
| **Ожидаемый результат** | Появляется сообщение:  Неверные учетные данные |
| **Фактический результат** | Появляется сообщение:  Неверные учетные данные (Рисунок 30) |
| **Статус** | Зачёт (Рисунок 29-31) |
| **Предварительное условие** | В базе данных должен быть пользователь с электронной почтой gilmanovalilya3004@gmail.com и пароль должен быть не WrongPassword123 |
| **Постусловие** | Пользователь не авторизован в системе и не имеет активную сессию.  (При переходе на страницу отзывов /reviews. Нет раздела“Добавить отзыв”Рисунок 11) |
| **Примечания/комментарии** | - |

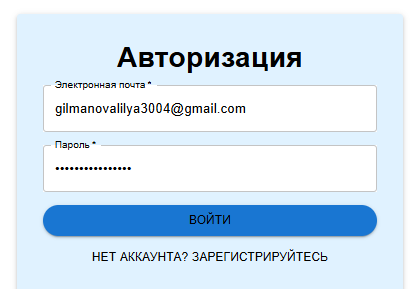


Рисунок 29.

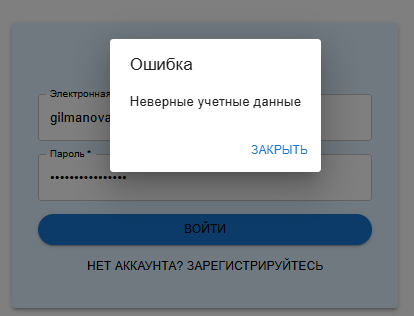


Рисунок 30.

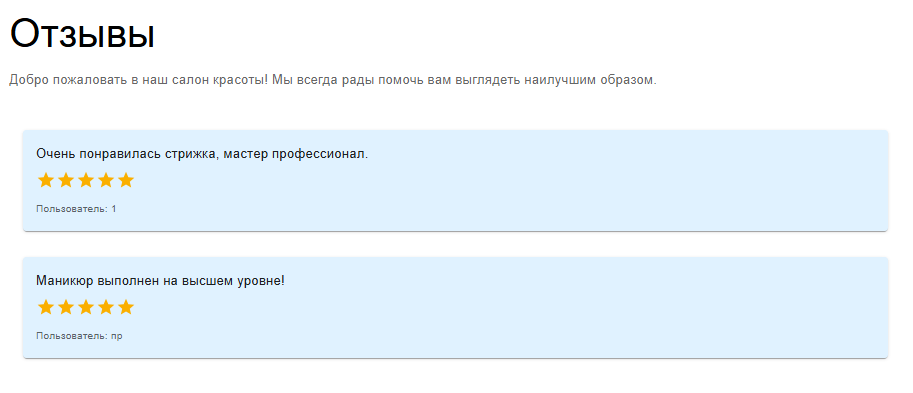


Рисунок 31.

|  |  |
| --- | --- |
| **Тестовый пример #** | TC\_UI\_5 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | authorizationWithValidData  (Авторизация с валидными данными) |
| **Краткое изложение теста** | Проверка корректной авторизации пользователя с действительными данными (правильный email и пароль). |
| **Этапы теста** | 1. Открыть сайт по http://localhost:3001/ 2. Перейти на страницу на страницу авторизации (/login). 3. Ввести в поля формы:   Электронная почта: gilmanovalilya3004@gmail.com  Пароль: StrongPassword123   1. Нажать на кнопку “Войти”. 2. В модальном окне с информацией, что код выполнен успешно нажать “ОК” |
| **Тестовые данные** | Электронная почта: gilmanovalilya3004@gmail.com  Пароль: StrongPassword123 |
| **Ожидаемый результат** | Появляется сообщение:  Вход выполнен успешно! |
| **Фактический результат** | Появляется сообщение:  Вход выполнен успешно! (Рисунок 33) |
| **Статус** | Зачёт (Рисунок 32-34) |
| **Предварительное условие** | Пользователь должен быть зарегистрирован в системе с действительными данными. Пользователь должен иметь подтвержденную электронную почту. |
| **Постусловие** | Пользователь авторизован в системе и имеет активную сессию. (При переходе на страницу отзывов /reviews. Присутствует раздел “Добавить отзыв”) |
| **Примечания/комментарии** | - |

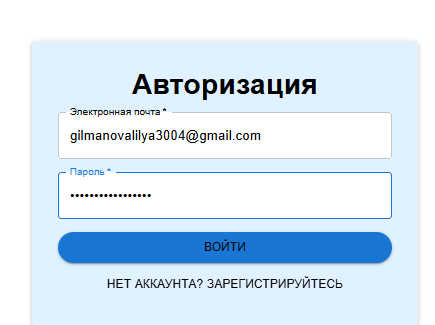


Рисунок 32.

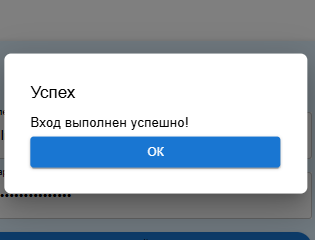


Рисунок 33.

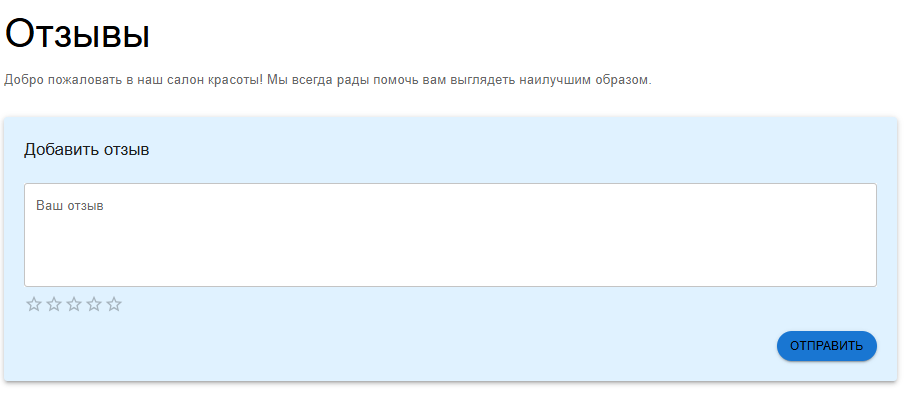


Рисунок 34.

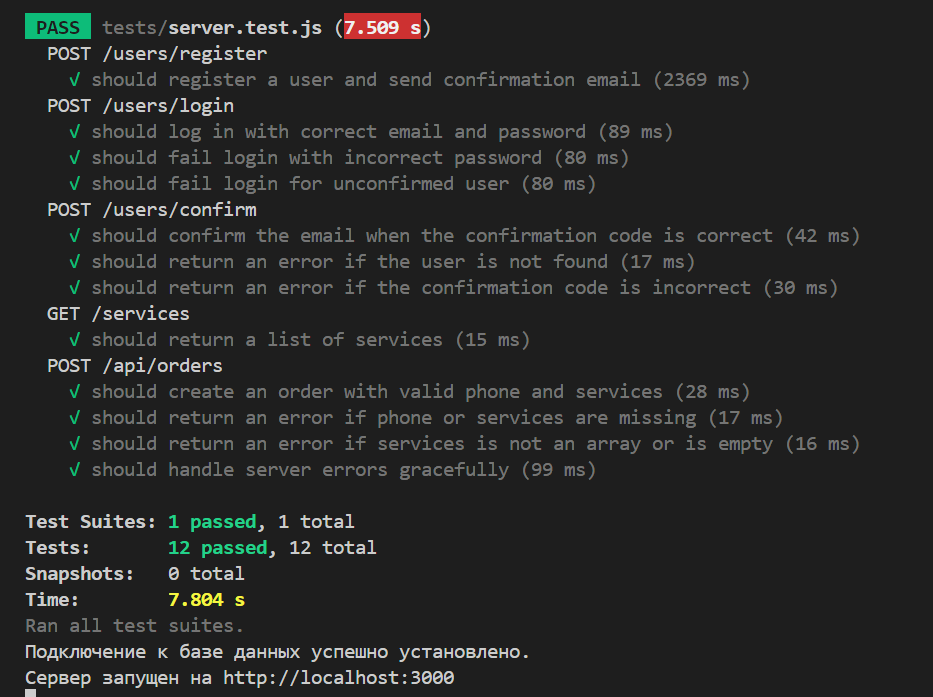


Рисунок 35. Unit-тесты.

Задание 6. GitHub. Оценка проекта

Для добавления проекта в репозиторий использовались следующие команды:

git remote add origin https://github.com/gilmanoval/yp.git

git init

git add .

git commit -m "Initial commit"

git push

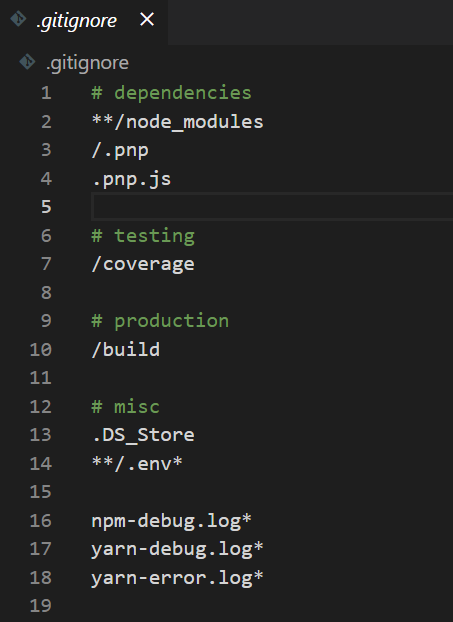


Рисунок 36. Файл gitignore.

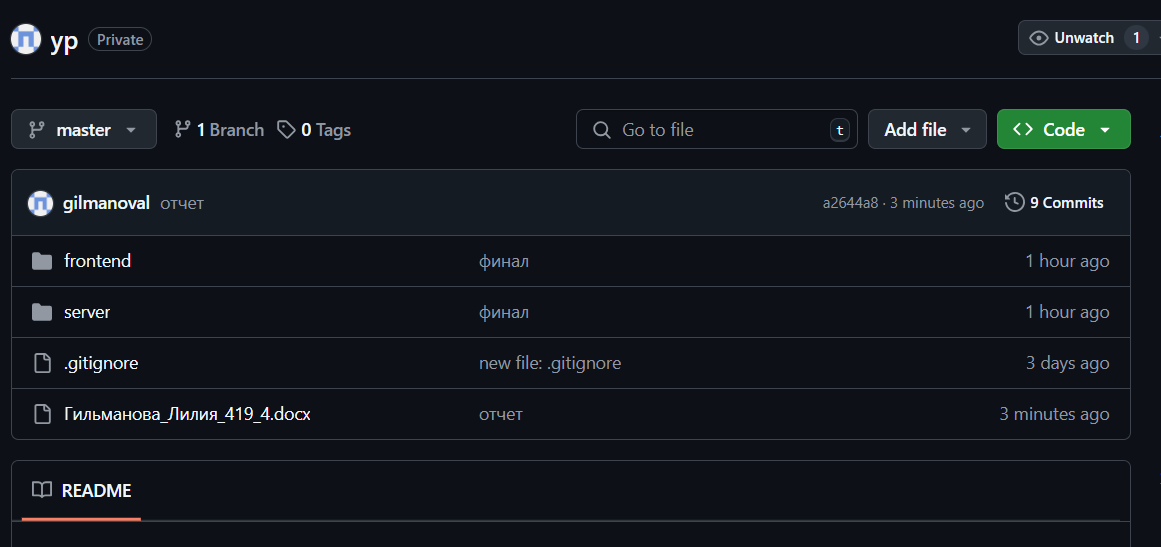


Рисунок 37. Git-репозиторий.