|  |  |  |
| --- | --- | --- |
| *Лабораторная*  *работа* 1 | *Выполнили* | *Абдулрахман А.*  *Нагоди М.З.*  *Магомедов М.О.*  *Шеху А.У.* |
| *Группа* | *ПрИн*-366/367 |
| *Преподаватель* |  |
| *Дата сдачи* |  |

**1. Введение**

Этот документ представляет собой план тестирования  
приложения чата в реальном времени. Описывает функциональность  
тестируемого приложения и функциональность, которая не будет тестироваться. В  
документе описываются потребности группы в успешном тестировании,  
критерии приемлемости, методы тестирования и  
используемый подход. Документ содержит описание рисков на пути  
реализации проекта и разделение зон ответственности.

**2. Тестируемая функциональность:**

- Проверьте, может ли пользователь отправлять и получать почту.

-Убедитесь, что время доставки сообщения и любая другая ожидаемая  
информация о сообщении отображаются правильно для пользователя.

-Пользователь должен иметь возможность видеть статус “ввод", когда  
получатель сообщения вводит ответ.

- Проверьте, может ли пользователь копировать и вставлять сообщения и их  
части.

- пользователь может видеть, когда была опубликована работа (время, дата)  
, соответствующее уведомление должно быть отображено всем кандидатам

-Убедитесь, что push-уведомления приложения работают правильно.  
Срок поставки

- Проверьте, может ли пользователь изменять настройки уведомлений  
(включать и выключать звук уведомлений, выбирать тип  
отображаемых уведомлений и т.д.).

**3. Тестовые сценарии**

После того как вы определились с приложением, которое хотите протестировать,  
рекомендуется провести быстрый мозговой штурм его функций  
, чтобы придумать идеи для ваших тестовых примеров.

Многое должно прийти на ум, когда вы начинаете думать о том, какую функциональность тестировать, вот мой список:

* -Главная страница
* Кнопки
* Пользователи
* Безопасность учетной записи
* Войдите в систему
* Выход из системы
* Отправка писем
* Просмотр онлайн-рекрутеров
* Уведомления

1. Присоединяйтесь к платформе

•Посетите домашнюю страницу

2.Получать уведомления, когда пользователь вводит

3. Уведомлять о присоединении или уходе пользователя.

4. Все кнопки функциональны

•Если пользователь нажимает на кнопку "Присоединиться к чату”, он должен перейти на страницу  
чата

•Если пользователь нажмет кнопку "Отправить", введенное сообщение  
должно быть отправлено.

•Если пользователь нажмет на значок настроек, он  
сможет настроить чат и внести изменения.

**4. Подход к тестированию**

Интерфейс приложения, видимая функциональность и удобство  
использования Приложение будет протестировано вручную с  
привлечением целевой аудитории (юзабилити-тестирование).  
Тестовые сценарии представлены выше. Выпуск новых версий приложения  
будет сопровождаться тестированием дыма. Внутренняя структура приложения  
будет протестирована с помощью модульных тестов с использованием методологий “черного ящика” и “белого ящика".

**5. Критерии успеха/неудачи тестирования**

Критерии успеха:

-Успешно завершенные дымовые испытания;

- Модульные тесты успешно завершены;

-Хорошие отзывы от юзабилити-тестирования;

- приложение не вылетает и работает нормально (никаких негативных  
отзывов на эту тему во время юзабилити-тестирования и тестирования  
командой разработчиков /тестировщиков);

-логика интерфейса удобна и интуитивно  
понятна (отрицательных отзывов на эту тему во время юзабилити-тестирования и  
тестирования командой разработчиков/тестировщиков нет).

Критериями отказа будут являться неудачное дымовое тестирование, а  
также наличие неудачных тестов структуры программы и неоднократные  
негативные отзывы о той или иной функциональности или удобстве использования программы.

**6. Критерии продолжения / прекращения тестирования**

Тестирование полностью прекращается, если внутренние тесты  
функций чата терпят неудачу. В противном случае пользовательское тестирование не выполняется для  
модулей, внутренние тесты которых не были пройдены.

**7. Обязанности тестировщиков**

-Выявлять и анализировать ошибки и проблемы, возникающие при работе с  
программными продуктами.

-Разработка алгоритмов тестирования сценариев.

-Разработка автотестов.

-Документирование выявленных дефектов

**8. Требования к методам испытаний**

Мы рекомендуем использовать следующие инструменты:

-Django Rest Framework (DRF)