Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**Иркутский национальный исследовательский технический университет**

|  |
| --- |
| Институт информационных технологий и анализа данных |
| наименование института |
| Отделение прикладной математики и информатики |
| наименование кафедры |

Допускаю к защите

|  |  |
| --- | --- |
| Руководитель |  |
|  | подпись |
|  | Е.А. Фунтикова |
|  | И.О. Фамилия |

|  |
| --- |
| Проектирование и разработка базы данных |
| «Торговая Организация» |
| наименование темы |

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине

|  |
| --- |
| Информатика |

|  |  |  |
| --- | --- | --- |
| 1.000.00.00 – ПЗ | | |
| обозначение документа | | |
| Выполнил студент |  | ИСТб-20-3 | |  |  | |  | Е.А. Житов |
|  |  | шифр | |  | подпись | |  | И.О. Фамилия |
| Нормоконтроль |  |  | |  |  | |  | З.А. Бахвалова |
|  |  |  | |  | подпись | |  | И.О. Фамилия |
| Курсовая работа защищена с оценкой | | | | |  | | | |

Иркутск 2023 г.

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ЗАДАНИЕ**

**на курсовую работу**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **По курсу** | Технологии разработки программных комплексов | | | | | | | | | | | | | | | | |
| **Студенту** | Житову Е.А. | | | | | | | | | | | | | | | | |
|  | (фамилия, инициалы] | | | | | | | | | | | | | | | | |
| **Тема работы** | | Онлайн-регистрация авиабилетов | | | | | | | | | | | | | | | |
| **Исходные данные** | | Создать информационную систему для совершения онлайн-регистрации авиабилетов. Проанализировать предметную область и реализовать систему в соответствии с жизненный циклом ПО. | | | | | | | | | | | | | | | |
| **Рекомендуемая литература** | | | | | | |  | | | | | | | | | | |
| 1. СТО 005-2015 СИСТЕМА МЕНЕДЖМЕНТА КАЧЕСТВА. Учебно-методическая деятельность. Оформление курсовых проектов (работ) и выпускных квалификационных работ технических специальностей (http://www.istu.edu/structure/57/2506/) | | | | | | | | | | | | | | | | | |
|  | | |  | | | | | | | | | | | | | | |
| Дата выдачи задания | | | « | 19 | » | октября | | | | | 2020 г. | | | | |
|  | | | | | | | | | | | | | | | | | |
| Задание получил | | | | | | | |  | | | | |  | | Е.А. Житов | | |
|  | | | | | | | | подпись | | | | |  | | И.О. Фамилия | | |
|  | | | | | | | | |  |  | |  | |  | | |  | |
| Дата представления работы руководителю | | | | | | | | | « | 21 | | » | | мая | | | 2023г. |
|  | | | | | | | | | | | | | | | | | | |
| Руководитель курсовой работы | | | | | | | |  | | | | |  | | З.А. Бахвалова | | |
|  | | | | | | | | подпись | | | | |  | | И.О. Фамилия | | |

**Содержание**

[Формирование требований 4](#_Toc134439782)

[1 Описание предметной области 4](#_Toc134439783)

[2 Описание проблемы 4](#_Toc134439784)

[3 Цель 4](#_Toc134439785)

[4 Словари 5](#_Toc134439786)

[Словарь предметной области 5](#_Toc134439787)

[Словарь по Абботу 6](#_Toc134439788)

[Объектно-ориентированный словарь 7](#_Toc134439789)

[5 Постановка задачи 8](#_Toc134439790)

[6 Функциональные требования ПО 9](#_Toc134439791)

[Проектирование 9](#_Toc134439792)

[Инструменты 9](#_Toc134439793)

[Описание алгоритма работы приложения 12](#_Toc134439794)

[Интерфейс 13](#_Toc134439795)

[Тестирование ПО 16](#_Toc134439796)

[Модульное тестирование 16](#_Toc134439797)

[Функциональное тестирование 16](#_Toc134439798)

[Тестирование интерфейса 19](#_Toc134439799)

[Документация пользователя 20](#_Toc134439800)

[Заключение 21](#_Toc134439801)

[Список использованных источников 22](#_Toc134439802)

## Формирование требований

## 1 Описание предметной области

Аэропорт, занимающейся обслуживанием авиаперевозок. Пассажир, купивший авиабилет использует пришедший ему номер бронирования для самостоятельной онлайн-регистрации. Для совершения авиаперелета, пассажиру необходимо купить билет, зарегистрироваться и получить посадочный талон. Полученный посадочный талон, пассажир предоставляет сотруднику безопасности. Сотрудник безопасности проверяет данные на талоне, паспорт и самого пассажира. После успешной проверки пассажир допускается в зону посадки.

## 2 Описание проблемы

Компанией заказчиком является аэропорт. Аэропорт заключил новые контракты с авиакомпаниями. Количество рейсов будет увеличено. Проблемой является ограниченное пространство в зоне регистрации для планируемого в ближайшее время увеличения пассажиропотока. Аэропорт не имеет возможности расширить зону регистрации.

## 3 Цель

Сократить время нахождения пассажиров в зоне регистрации, исключив столпотворение людей и очереди путем реализации механизма **онлайн-регистрации** для пассажиров без багажа.

## 4 Словари

# Словарь предметной области

|  |  |  |
| --- | --- | --- |
| **№** | **Сущность** | **Описание** |
| 1 | Номер бронирования | Уникальная последовательность символов, которая позволяет однозначно определить пассажира и его рейс. |
| 2 | Номер рейса | Уникальная последовательность символов, которая позволяет однозначно определить рейс. |
| 2 | Маршрутный  Лист | Квитанция (Чек), подтверждающий факт прохождения регистрации. Содержит в себе номер бронирования, паспортные данные пассажира, информацию о рейсе. |
| 3 | Посадочный Талон | Посадочный талон – документ, дающий право пассажиру на посадку в самолет. Чтобы получить талон необходимо предоставить соответствующий маршрутный лист |
| 5 | Онлайн-Регистрация | Процесс регистрации пассажира на рейс, не требующий нахождение пассажира в зоне регистрации. |
| 6 | Зона Регистрации | Специально отведенное место в аэропорте для осуществления регистрации пассажиров. |
| 8 | Паспортные данные | Паспортные данные, предъявляемые пассажиром |
| 9 | Пассажир | Лицо, купившее билет для совершения перелета |
| 10 | Структура посадочного талона | Специальный формат, структура, которой должны соответствовать все посадочные талоны будто оформление или предоставляемая информация |
| 11 | Информация о рейсе | Специальный формат, структура, которой должны соответствовать все рейсы будто оформление или предоставляемая информация |
| 12 | Структура номера бронирования | Специальный формат, структура, которой должны соответствовать все номера бронирования будто оформление или предоставляемая информация |
| 13 | Структура номера рейса | Специальный формат, структура, которой должны соответствовать все номера рейсов будто оформление или предоставляемая информация |
| 14 | Место | Место в самолете. Каждое место пронумеровано и имеет свой номер от 0 до 255 включительно |

## Словарь по Абботу

|  |  |  |
| --- | --- | --- |
| **№** | **Сущность** | **Глагол** |
| 1 | Номер бронирования | Вводить |
| 2 | Маршрутный  Лист | Сгенерировать, Скачать, Распечатать, Отдать |
| 3 | Посадочный Талон | Распечатать, Отдать |
| 5 | Онлайн-Регистрация | Пройти |
| 6 | Зона Регистрации | Избегать |
| 8 | Паспортные данные | Предоставлять, Вводить |
| 9 | Пассажир | Регистрироваться |
| 10 | Структура посадочного талона | Следовать, Соответствовать |
| 11 | Информация о рейсе | Следовать, Соответствовать |
| 12 | Структура номера бронирования | Следовать, Соответствовать |
| 13 | Структура номер рейса | Следовать, Соответствовать |
| 14 | Место | Выбирать |

## Объектно-ориентированный словарь

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Сущность** | **Данные** | **Методы** |
| 1 | Номер бронирования | Номер бронирования | Вводить |
| 2 | Маршрутный  Лист | Паспортные данные  Информация о рейсе | Скачать, Отдать |
| 3 | Талон | Паспортные данные  Информация о рейсе  Номер места, выхода | Получить, Отдать |
| 5 | Онлайн-Регистрация | Номер бронирования Паспортные данные | Пройти |
| 6 | Зона Регистрации | - | Избегать |
| 7 | Паспортные данные | * Фамилия; * Имя; * Отчество; * Пол; * Дата рождения; * Серия паспорта; * Номер паспорта. | Вводить, Предоставлять |
| 8 | Пассажир | * Паспортные данные; * Паспорт; * Маршрутный лист; * Посадочный талон. | Регистрироваться |
| 9 | Структура посадочного талона | * Паспортные данные * Информация о рейсе | Следовать, Соответствовать |
| 10 | Информация о рейсе | * Номер рейса; * Название авиакомпании; * Город отправления и город прибытия * Аэропорт отправления * Аэропорт прибытия * Дата и время отправления и прибытия | Предоставлять |
| 14 | Место | Номер места | Выбирать |

|  |  |  |  |
| --- | --- | --- | --- |
| 11 | Структура номера бронирования | Последовательность, состоящее из 36 символов цифр и английских букв (от a до f) разделенных между собой через знак тире в соотношении **8-4-4-4-12**.  Пример:**123e4567-e89b-12d3-a456-426655440000** | Следовать, Соответствовать |
| 12 | Структура номера рейса | Последовательность, состоящее из 36 символов цифр и английских букв (от a до f) разделенных между собой через знак тире в соотношении **8-4-4-4-12**.  Пример:**123e4567-e89b-12d3-a456-426655440000** | Следовать, Соответствовать |

## 5 Постановка задачи

Создать механизм онлайн регистрациидля аэропорта.

Реализовать возможность пассажиру зарегистрироваться на рейс по его номеру бронирования, используя паспортные данные. Предусмотреть возможность выбора места в зависимости от типа авиабилета (эконом, бизнес). Некоторые авиабилеты могут не предоставлять возможность выбора места или предоставлять, но за дополнительную плату.

Реализовать возможность зарегистрированному пассажиру самостоятельно получить маршрутный лист после прохождения онлайн регистрации.

Пассажир находит по номеру бронирования свой рейс, указывает свои паспортные данные, выбирает место в самолете и подтверждает регистрацию. Если пассажир ввел неверный номер бронирования, то он не сможет найти свой рейс и продолжить регистрацию. Если пассажир на этапе ввода паспортных данных, указал неверные данные, то на 3-ем этапе паспортного контроля сотрудник безопасности не пропустит его, так как он не будет совпадать паспортные данные. После ввода паспортных данных, пассажиру будет предоставлена возможность выбора места (Если авиабилет предусматривает выбор места). Если пассажир не выбрал место, то место после подтверждения регистрации назначается автоматически. Последний этап регистрации – это её подтверждение. После ввода номера бронирования, паспортные данных, места (если возможно), пассажир обязан подтвердить регистрацию путем нажатия на соответствующий элемент интерфейса. В противном случае регистрация не будет действительной и маршрутный лист будет недоступен.

После регистрации пассажир скачивает маршрутный лист, который пассажир подносит к специальному терминалу в аэропорту для получения посадочного талона.

## 6 Функциональные требования ПО

**Онлайн регистрация**

Модуль должен предоставить пассажиру возможность произвести онлайн регистрацию. Пассажиру необходимо выполнить следующий порядок действий:

1. Ввести номер бронирования и получить информацию о своем рейсе;
2. Ввести паспортные данные и пройти проверку;
3. Выбрать место (Если пассажира не устроило место по умолчанию)
4. Завершить регистрацию;

По завершении регистрации, пассажиру необходимо предоставить возможность скачать маршрутный лист.

Предусмотреть исключительные ситуация, которые могут возникнуть в процесса онлайн-регистрации.

## Проектирование

## Инструменты

ПО является веб приложением, которое можно будет открыть в браузере для прохождения онлайн регистрации. Будет использован инструментарий программных средств:

1. **Visual Studio Code** – удобный легкий текстовый редактор кода, благодаря расширениям позволяет легко и просто настроить редактор под данный проект.
2. **Python** – язык программирования. Является самым быстрым языком при прототипировании ПО. Позволяет в кратчайшие сроки реализовать MVP для заказчика.
3. **Flask** – фреймворк для написания API приложения. Требует очень низкий порог вхождения для быстрого написания и поддержки API.
4. **PyTest** – фреймворк для написания авто тестов. Является одним из самых популярных и часто используемых.
5. **PostgreSQL** – реляционная СУБД. Бесплатная с открытым исходным кодом. Необходима для хранения и управления данными.
6. **Selenium** – библиотека, позволяющая провести тестирование интерфейса. Бесплатная, открытый исходный код.

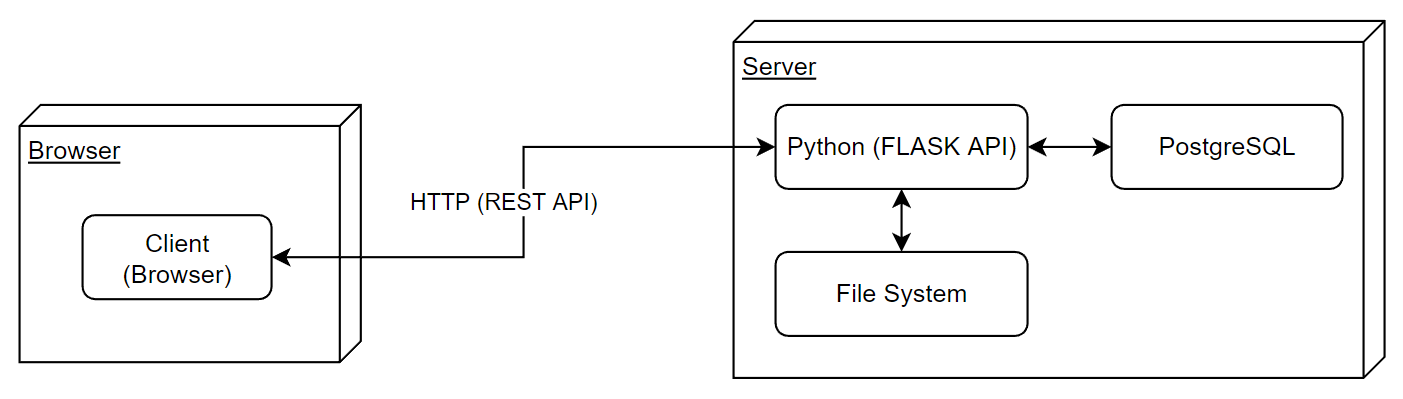


Рисунок X – диаграмма развертывания приложения

На диаграмме развертывания представлена архитектура ПО. Клиент (браузер) используя протокол http получает доступ к приложению, загружая файлы для отображения интерфейса. Клиент также может посылать запросы к API, с помощью которого происходит манипуляция с данными и работой с СУБД. FLASK API – сервер принимающий запросы от клиента. Его основная задача выдача генерация файла и его выдача и манипуляция с данными БД.

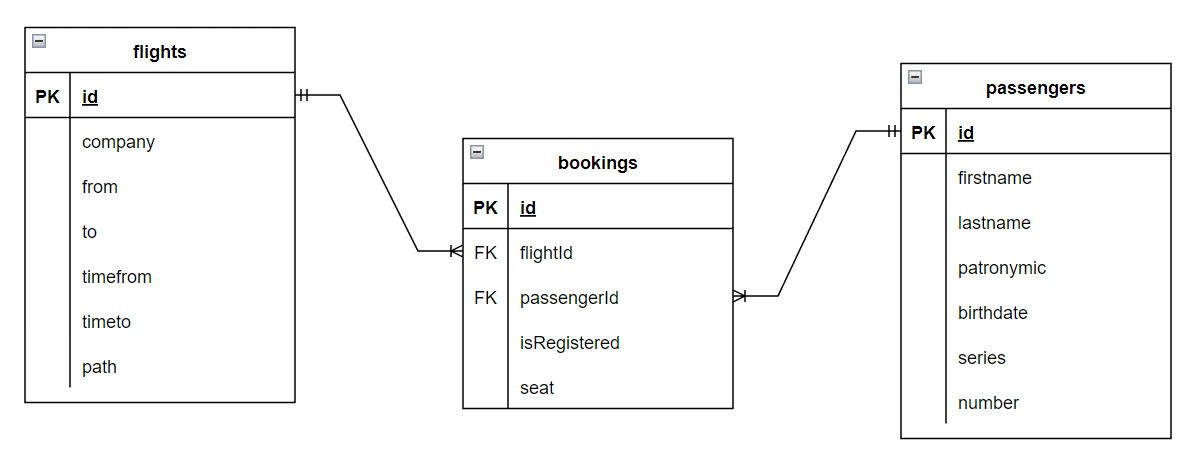


Рисунок X – логическая модель базы данных

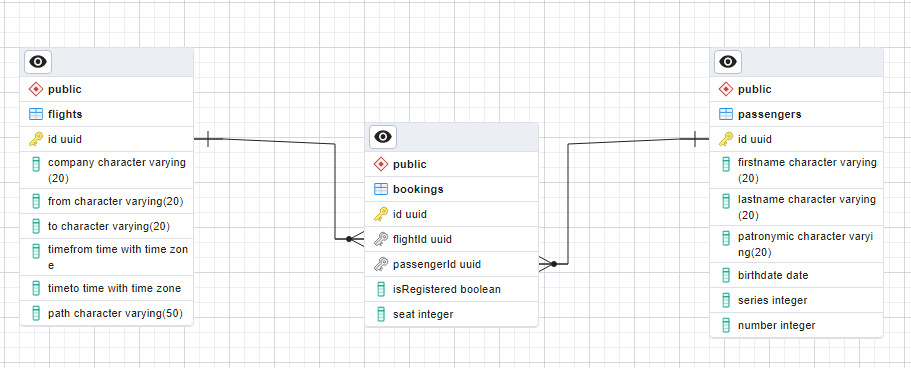
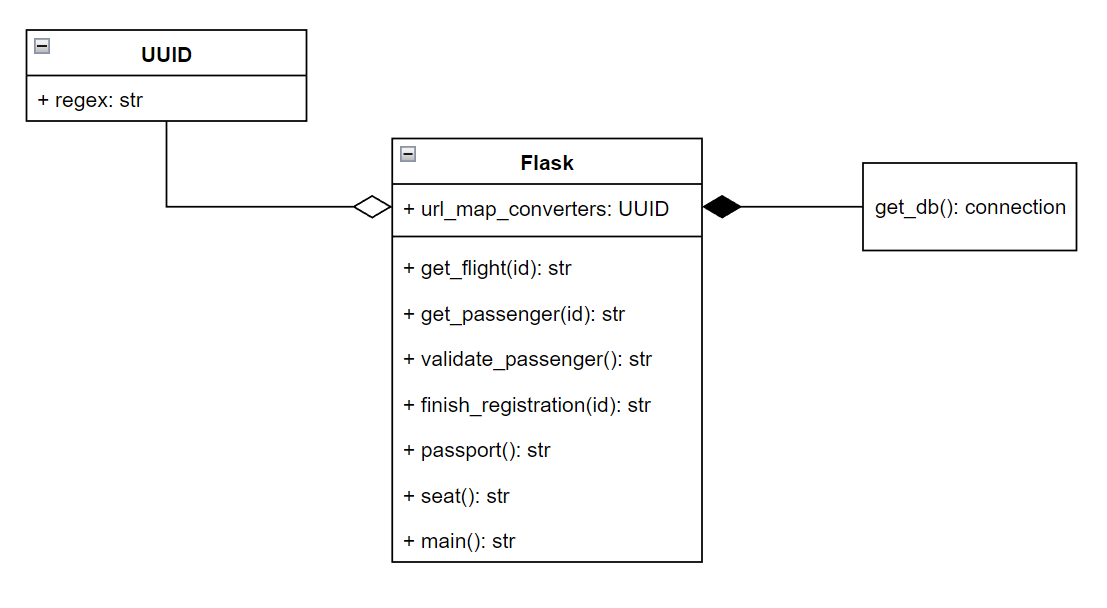


Рисунок X – физическая модель базы данных



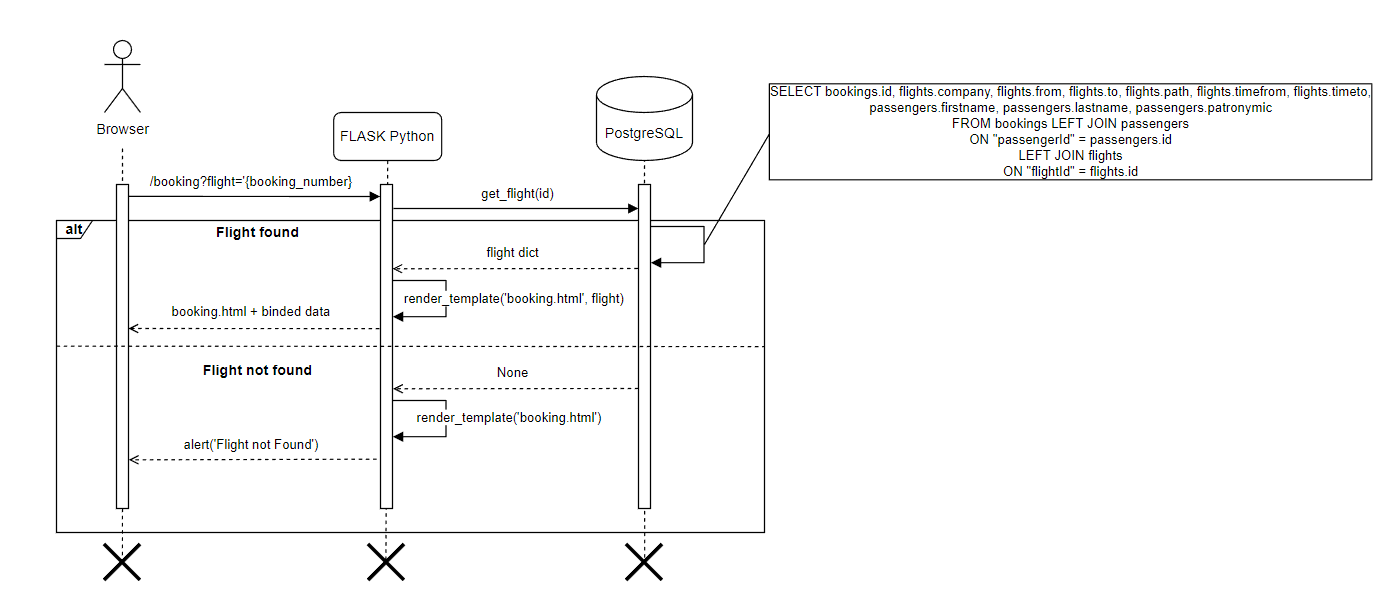
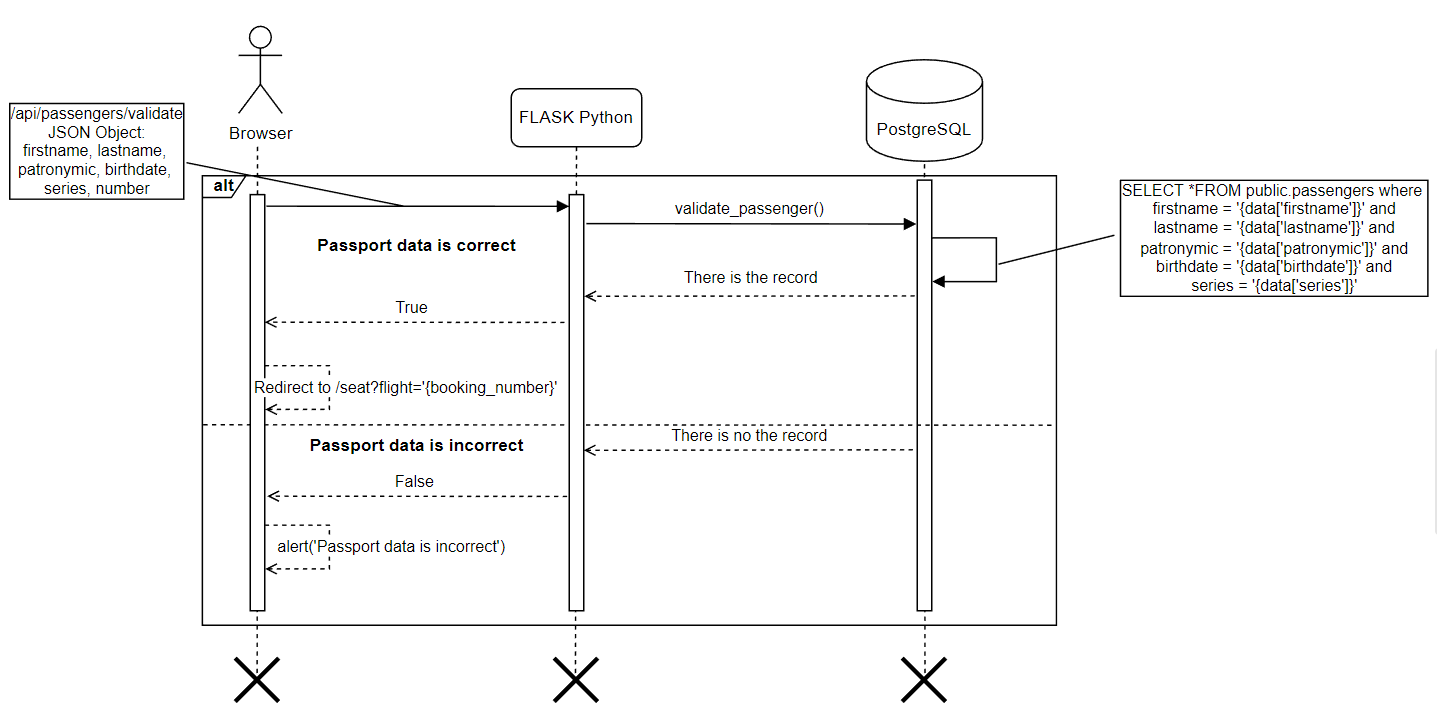
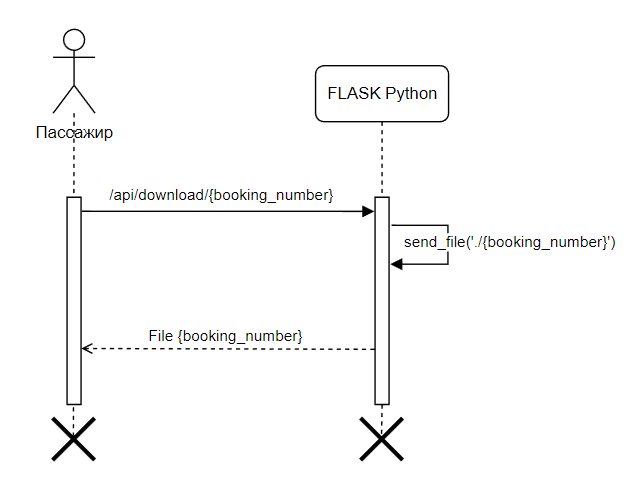


Рисунок X – Use Case ввод номера бронирования





## Описание алгоритма работы приложения

Приложение состоит 2 двух частей frontend и backend. Frontend это html файлы (интерфейс) с информацией, который отдается пользователю. Backend – это API функции-обработчики для манипуляции над данными из БД. Каждая функция обработчик однотипна и очень похожа по сравнению с другими. Каждая функция обработчик должна вернуть содержимое html файла либо иную строку, если функции обработчики имеют в аннотации, маршрут которых начинающейся с **/api**

Функции, начинающиеся с **/api** обрабатывают данные

Остальные функции-обработчики возвращают интерфейс приложения

Протокол между frontend и backend – HTTP

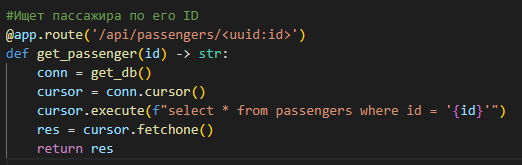


Рисунок X – функция обработчик манипулирующая с данными из БД

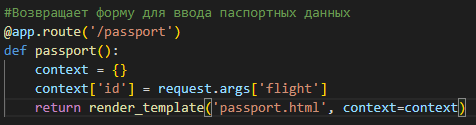


Рисунок X – функция обработчик, возвращает интерфейс (страницу)

## Интерфейс

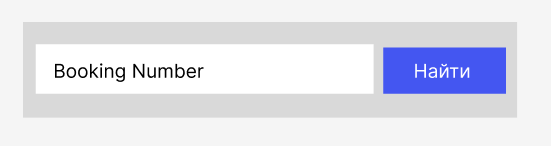


Рисунок X – Начальная страница (Ввод номера бронирования)

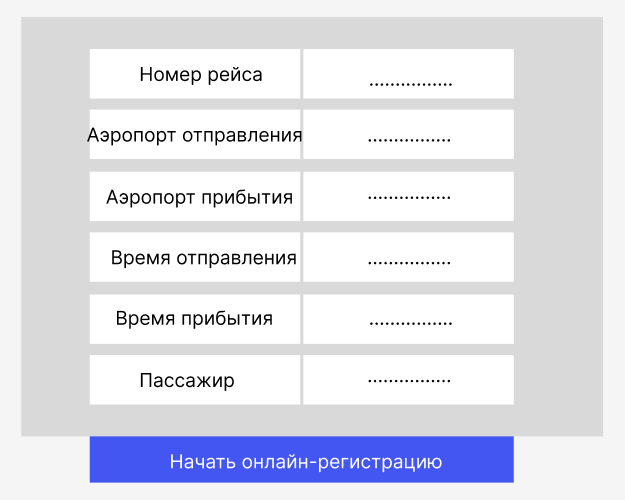


Рисунок X – макет страницы информации о рейсе

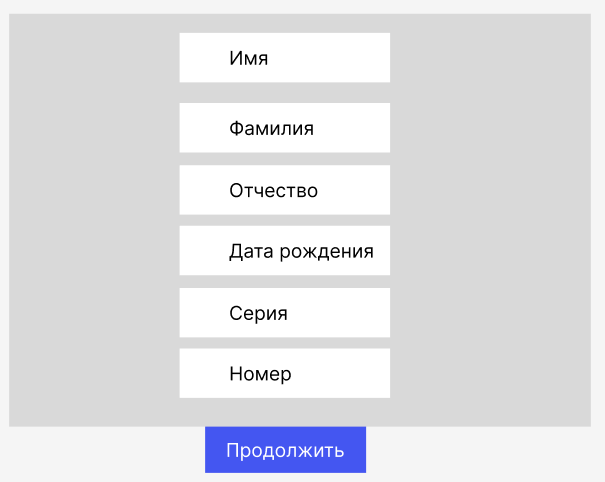


Рисунок X – макет страницы ввода паспортных данных



Рисунок X – макет страницы выбора места

## Тестирование ПО

## Модульное тестирование

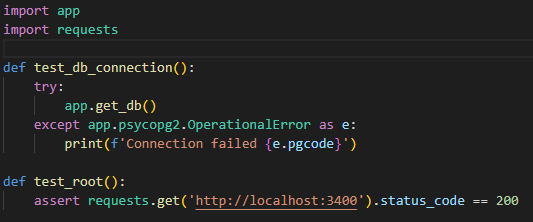
****

Рисунок X – unit тесты

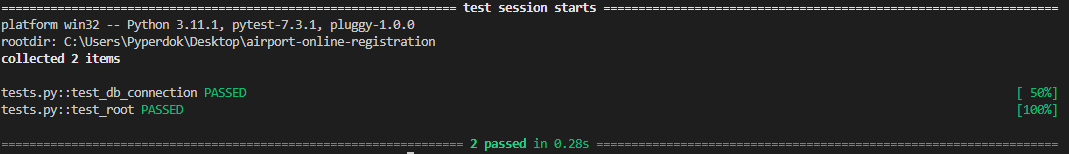


Рисунок X – результат unit тестирования

## Функциональное тестирование

Было проведено функциональное тестирование приложения, осуществляющего процесс онлайн-регистрации пассажира, на основе предоставленных функциональных требований.

Результаты тестирования показали, что все функциональные требования были реализованы и работают корректно:

1. При вводе номера бронирования пассажир получает информацию о своем рейсе.



Рисунок X - Ввод номера бронирования и получение рейса пассажира

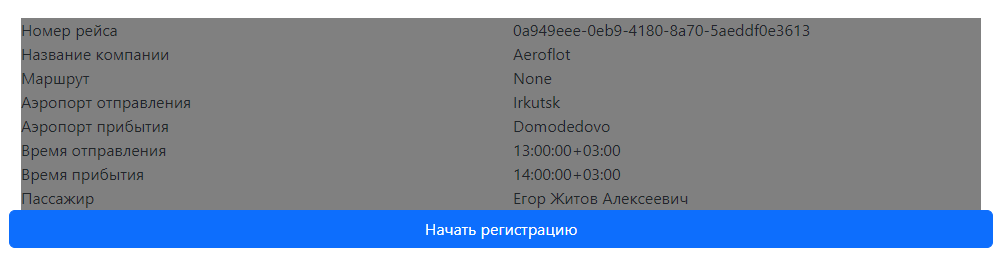


Рисунок X – Результат получения информации о рейсе по его номеру бронирования

1. При вводе паспортных данных пассажир проходит проверку.

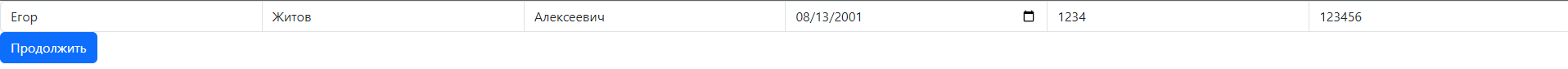


Рисунок X – Ввод паспортных данных пассажира для прохождения проверки

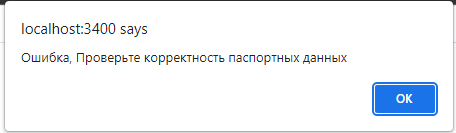
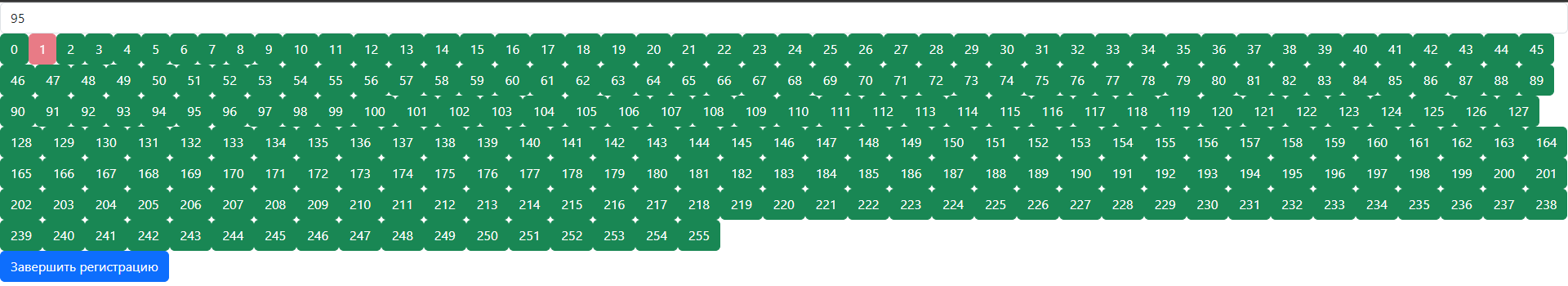


Рисунок X – Вывод ошибки в случае неверных паспортных данных

Рисунок X – Перенаправление на страницу выбора места при успешной проверке

1. Пассажир может выбрать место, если ему не подходит место по умолчанию.

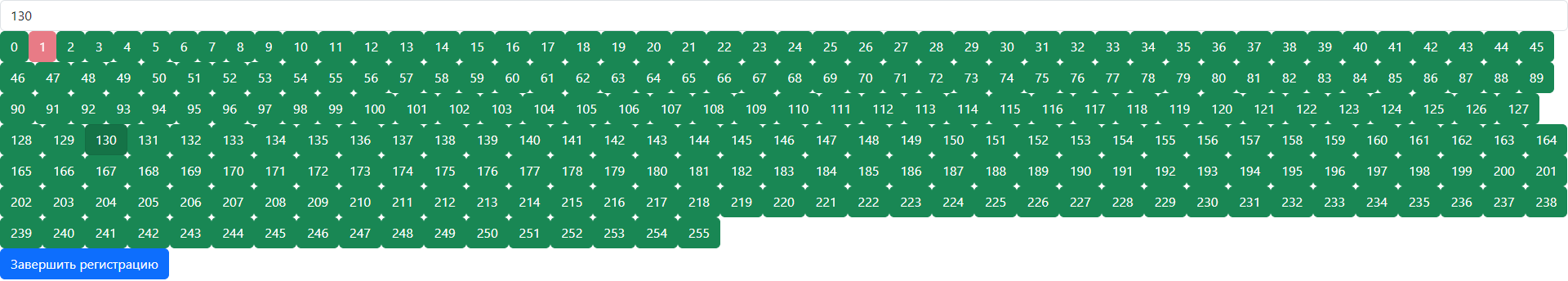


Рисунок X – Пассажир выбрал место 130

1. После выбора места пассажир может завершить регистрацию.

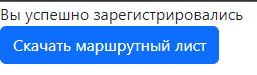


Рисунок X – Пассажир выбрал место 130 и нажал завершить регистрацию

1. После завершения регистрации пассажиру предоставляется возможность скачать маршрутный лист.

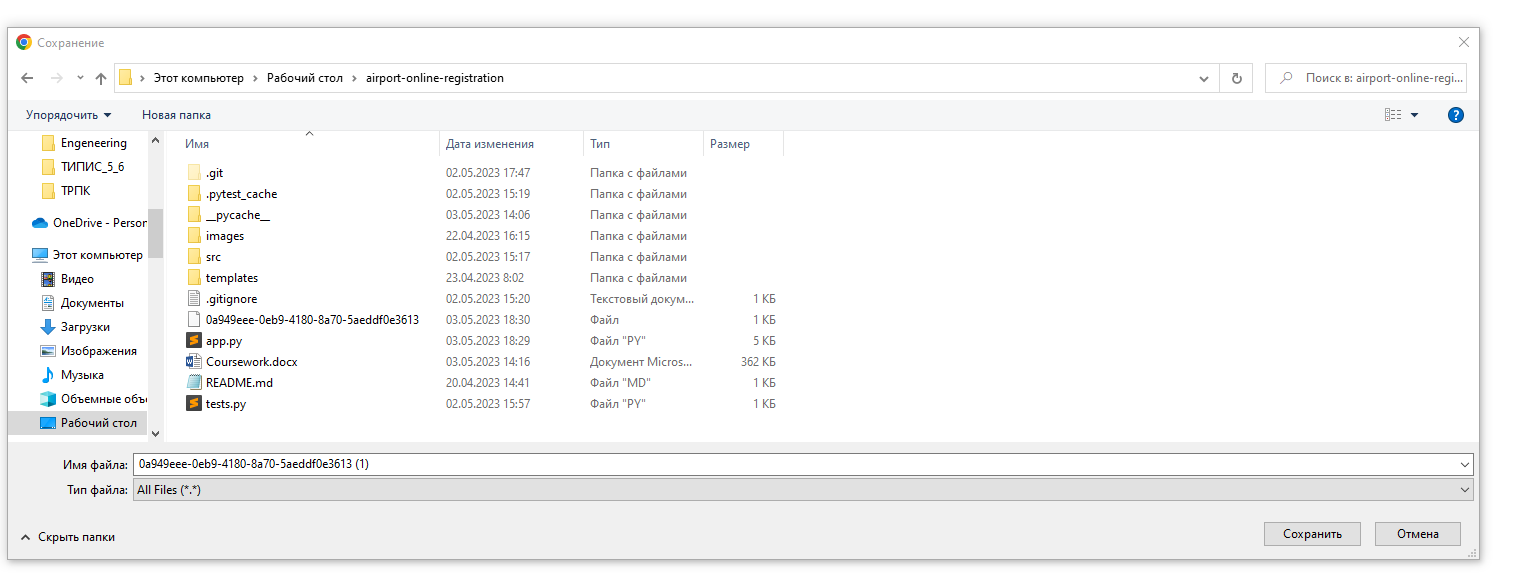


Рисунок X – Скачивание маршрутного листа

Тестирование показало, что все функциональные требования приложения работают без сбоев и соответствуют описанию. Таким образом, приложение готово к использованию пользователями.

## Тестирование интерфейса

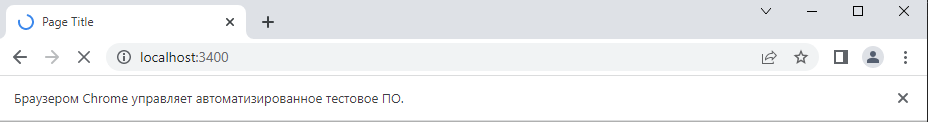
****

Рисунок X – результат End2End тестирования используя Selenium

(Проверка доступности содержимого главной страницы)

Было проведено оценочное тестирование интерфейса приложения, и результаты тестирования показали, что интерфейс приложения удовлетворяет требованиям и является удобным для использования.

В процессе тестирования были оценены следующие аспекты интерфейса:

1. Оформление и дизайн: дизайн приложения выполнен в единообразном стиле, цветовая гамма гармонична и не вызывает дискомфорта, элементы интерфейса выглядят аккуратно и четко.
2. Навигация: приложение имеет простую и понятную структуру навигации, что позволяет пользователям быстро находить нужную информацию и функциональные возможности.
3. Функциональность: все функциональные возможности приложения работают корректно и без задержек, отклик на действия пользователя быстрый и отзывчивый.
4. Удобство использования: приложение легко и интуитивно понятно в использовании, даже для пользователей, не имеющих опыта работы с подобными приложениями.
5. Адаптивность: интерфейс приложения хорошо адаптирован под разные размеры экранов и устройства, что позволяет комфортно использовать приложение на мобильных устройствах и планшетах.

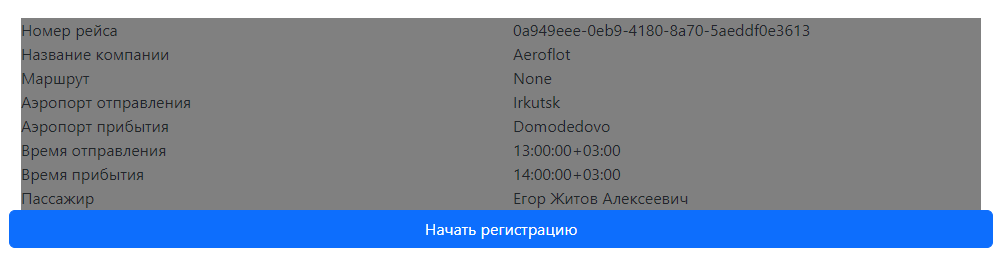
Таким образом, результаты оценочного тестирования показали, что интерфейс приложения удовлетворяет требованиям и является удобным и функциональным для использования пользователями.

## Документация пользователя

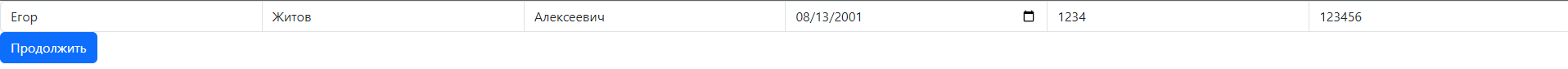
Чтобы начать процесс регистрации, вам необходимо ввести номер бронирования, который вы получили при покупке авиабилета. После того, как вы введете номер, нажмите кнопку "Найти".



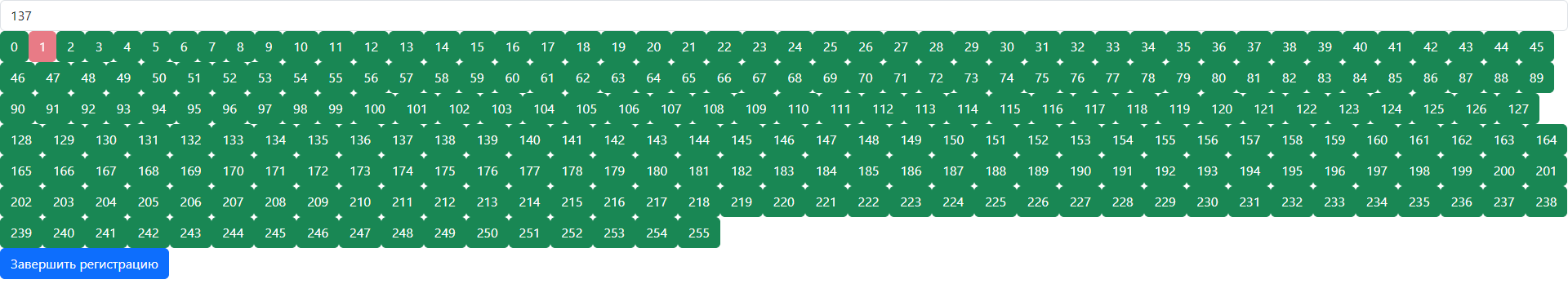
На следующей странице вы увидите информацию о рейсе и вашем билете. Если информация верна, нажмите кнопку "Начать регистрацию".

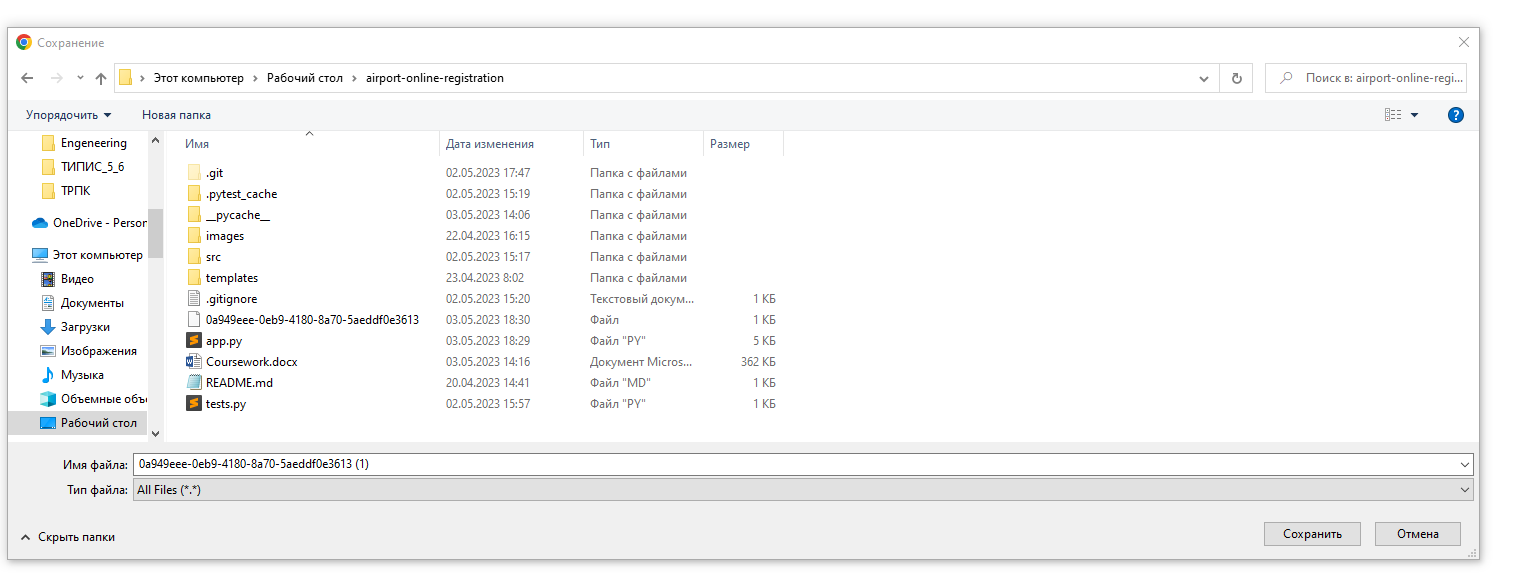


Далее вам будет предложено ввести ваши паспортные данные, такие как Имя, Фамилия, Отчество, Дата рождения, Серия и Номер. После того, как вы введете эту информацию, нажмите кнопку "Продолжить".



На следующей странице вы увидите места и сможете выбрать место, которое вам больше всего нравится. Если вам подходит место, которое автоматически было назначено системой, вы можете его оставить без изменений. Если вы хотите выбрать другое место, выберите любое другое свободное место. После того, как вы выберете место, нажмите кнопку "Завершить регистрацию".



На последней странице вы сможете скачать маршрутный лист, который вам необходимо будет предъявить на регистрации в аэропорту. 

## Заключение

В результате разработки информационной системы для онлайн-регистрации авиабилетов были достигнуты следующие результаты:

1. Улучшены знания языка программирования Python, в том числе его возможностей при создании веб-приложений;
2. Была изучена клиент-серверная архитектура, которая была использована при создании информационной системы;
3. Получены навыки автоматизированного тестирования с использованием библиотеки Selenium, что позволило увеличить скорость разработки и повысить качество решения;
4. Были изучены принципы жизненного цикла ПО, которые были использованы при разработке системы, что обеспечило ее эффективность и надежность;
5. В результате разработки информационной системы была достигнута автоматизация процесса онлайн-регистрации авиабилетов, что позволило повысить качество обслуживания пассажиров и ускорить процесс регистрации.

Таким образом, разработка информационной системы для онлайн-регистрации авиабилетов позволила значительно расширить знания и навыки в области программирования и разработки ПО, а также сделала важный вклад в улучшение качества обслуживания пассажиров.

GitHub репозиторий проекта:

(<https://github.com/Pyperdok/airport-online-registration>)

## Список использованных источников

1. GPT 3.5 // Chat GPT URL: <https://chat.openai.com/> (дата обращения: 08.05.2023).
2. Pytest Docs // Pytest URL: [https://docs.pytest.org/en/7.3.x](https://docs.pytest.org/en/7.3.x%20) (дата обращения: 08.05.2023).
3. Selenium Web Driver // Selenuim URL: <https://www.selenium.dev/documentation/webdriver/> (дата обращения: 08.05.2023).
4. Flask Docs // Flask URL: <https://flask.palletsprojects.com> (дата обращения: 08.05.2023).
5. Questions // StackOverflow URL: <https://stackoverflow.com> (дата обращения: 08.05.2023).