

УТВЕРЖДАЮ

*Клейменов Илья Валерьевич (Ассистент,
Воронежский Государственный
Университет)*
Руководитель (должность, наименование
предприятия – заказчика АС)

Личная Расшифровка
Подпись подписи

Печать
Дата 31.03.2023

УТВЕРЖДАЮ

*Свиридов Фёдор Юрьевич
(Студент, Воронежский
Государственный Университет)*
Руководитель (должность, наименование
предприятия – разработчика АС)

Личная Расшифровка
Подпись подписи

Печать
Дата 31.03.2023

Приложение для составления университетского расписания
наименование вида АС

Расписание занятий для ВУЗов
наименование объекта автоматизации

«UnivTimetable»
сокращенное наименование АС

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На 2-26 листах
Действует с 31.03.2023

СОГЛАСОВАНО

*Тарасов Вячеслав Сергеевич (Старший
преподаватель, Воронежский Государственный
Университет)*
Руководитель (должность, наименование
предприятия – заказчика АС)

Личная Расшифровка
подпись подписи

Печать
Дата 31.03.2023

Содержание

Содержание	2
1 Общие сведения.....	4
1.1 Понятия и термины	4
1.2 Полное наименование системы и ее условное обозначение	7
1.3 Наименование исполнителя и заказчика приложения	7
1.3.1 Наименование заказчика	7
1.3.2 Наименование исполнителя.....	7
1.4 Перечень документов, на основании которых создается система.....	7
1.5 Плановые сроки начала и окончания работы по созданию системы	8
1.6 Порядок оформления и предъявления заказчику результатов работы	8
2 Назначение и цели создания	9
2.1 Назначение проекта	9
2.2 Цели проекта.....	9
2.3 Задачи проекта.....	9
2.3.1 Для преподавателя	9
2.3.2 Для старосты	9
2.3.3 Для администратора	10
3 Требования к системе	11
3.1 Требования к системе в целом.....	11
3.1.1 Требование к структуре.....	11
3.1.2 Требование к эргономике и технической эстетике	11
3.1.3 Требование к защите информации.....	11
3.1.4 Требование к патентной чистоте.....	11
3.1.5 Требования к используемым технологиям.....	11

3.1.6	Требование к персоналу, обслуживающему приложение	13
3.2	Требования к функциям(задачам)	13
3.2.1	Авторизация	13
3.2.2	Просмотр расписания	14
3.2.3	Перенос занятий.....	16
3.2.4	Создание расписания.....	17
3.2.5	Скачивание расписания.....	18
3.3	Требования к поддерживаемым языкам	18
4	Перспективы развития, модернизации приложения	19
5	Состав и содержание работ по созданию приложения	20
6	Порядок контроля и приёмки приложения	22
7	Требования к документированию	23
8	Источники разработки	24
9	Реквизиты и подписи сторон	25

1 Общие сведения

1.1 Понятия и термины

- Мобильное приложение — программное обеспечение, предназначенное для работы на смартфонах, планшетах и других мобильных устройствах, разработанное для конкретной платформы (iOS, Android и т. д.);
- Android-приложение — программное обеспечение, предназначенное для работы на смартфонах, планшетах и других мобильных устройствах, разработанное для платформы Android;
- Android — это операционная система с открытым исходным кодом, созданная для мобильных устройств на основе модифицированного ядра Linux;
- Клиент — это аппаратный или программный компонент вычислительной системы, посылающий запросы серверу;
- Сервер — выделенный или специализированный компьютер для выполнения сервисного программного обеспечения;
- База данных — это упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. База данных обычно управляется системой управления базами данных (СУБД);
- SQL-запросы — это наборы команд для работы с реляционными базами данных;
- Дизайн-макет — это схематичное изображение финальной идеи с указанием всех деталей. В нем указываются концепция, шрифты, тексты, изображения, расположение всех элементов и общая картина продукта;
- Аутентификация — процедура проверки подлинности, например, проверка подлинности пользователя путем сравнения введенного им пароля с паролем, сохраненным в базе данных;

- Авторизация — предоставление определённому лицу или группе лиц прав на выполнение определенных действий;
- Фреймворк — программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта;
- SQL-инъекция — внедрение в запрос произвольного SQL-кода, который может повредить данные, хранящиеся в БД или предоставить доступ к ним;
- Frontend — клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса;
- Backend — программно-аппаратная часть сервиса, отвечающая за функционирование его внутренней части;
- REST — архитектурный стиль взаимодействия компонентов распределённого приложения в сети;
- API — описание взаимодействия одной компьютерной программы с другой;
- Аккаунт или учетная запись — это персональная страница пользователя или личный кабинет, который создается после регистрации;
- Пользователь — человек, использующий приложение;
- Преподаватель — работник высшего учебного заведения, преподающий какой-либо предмет(ы);
- Староста — человек из числа студентов, ответственный за организационные моменты, связанные с учебной деятельностью своей группы, в том числе за связь с преподавателями и своевременное уведомление одногруппников обо всех изменениях в расписании занятий;

- Администратор — человек, ответственный за обеспечение штатной работы приложения путём добавления/редактирования/удаления ВУЗов и пользователей;
- Расписание — это основной учебно-методический документ, определяющий педагогически целесообразную последовательность учебных занятий в образовательном учреждении на каждый день учебной недели и конкретизирующий таким образом учебный план.

1.2 Полное наименование системы и ее условное обозначение

Полное наименование: «Приложение для составления университетского расписания».

Название приложения: «UnivTimetable».

1.3 Наименование исполнителя и заказчика приложения

1.3.1 Наименование заказчика

Воронежский Государственный Университет.

Факультет Компьютерных Наук.

Кафедра Программирования и Информационных Технологий.

Старший преподаватель Тарасов Вячеслав Сергеевич.

Ассистент Клейменов Илья Валерьевич.

1.3.2 Наименование исполнителя

Воронежский Государственный Университет.

Факультет Компьютерных Наук.

Кафедра Программирования и Информационных Технологий.

3 курс, группа 5.2.

Студенты:

- Свиридов Фёдор Юрьевич;
- Молдавский Илья Михайлович;
- Распопов Павел Сергеевич.

1.4 Перечень документов, на основании которых создается система

- ГОСТ 34.602-89;
- Федеральный закон "Об информации, информационных технологиях и о защите информации" от 27.07.2006 N 149-ФЗ;
- Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ;
- Закон РФ от 07.02.1992 N 2300-1 (ред. от 11.06.2021) "О защите прав потребителей";

— Федеральный закон от 29.12.2012 г. № 273-ФЗ "Об образовании в Российской Федерации".

1.5 Плановые сроки начала и окончания работы по созданию системы

Плановый срок начала работ – март 2023.

Плановый срок окончания работ – июнь 2023.

1.6 Порядок оформления и предъявления заказчику результатов работы

Результаты работ предъявляются Заказчику в следующем виде:

- Техническое задание согласно ГОСТ 34.602-89;
- Работающее, согласно данному Техническому Заданию, мобильное приложение;
- Документация к приложению;
- Презентация в формате видео с демонстрацией функционала приложения;
- Защита проекта.

2 Назначение и цели создания

2.1 Назначение проекта

Создание приложения для составления университетского расписания с учётом вместительности и занятости аудиторий.

2.2 Цели проекта

- Организовать процесс формирования расписания занятий для высших учебных заведений;
- Реализовать процесс перестановок занятий в расписании при помощи предложения доступных вариантов для переноса.

2.3 Задачи проекта

- Предоставление преподавателями нежелательного времени для проведения занятий;
- Обеспечение возможности просмотра расписания на заданный промежуток времени как для старост (для их групп), так и для преподавателей (личное);
- Предоставление преподавателям права переноса занятий через приложение;
- Формирование расписания системой на основе предоставленных преподавателями данных;
- Экспорт расписания в формате Excel таблицы;
- Уведомление старост по электронной почте об изменениях в расписании.

2.3.1 Для преподавателя

- Получение и скачивание личного расписания на заданный промежуток времени;
- Обеспечение возможности переноса занятий.

2.3.2 Для старосты

- Просмотр расписания группы;

- Обеспечение возможности скачивания расписания группы в формате Excel.

2.3.3 Для администратора

- Добавление новых ВУЗов, факультетов, групп, аудиторий и пользователей;
- Редактирование и удаление информации о текущих ВУЗах и пользователях.

3 Требования к системе

3.1 Требования к системе в целом

3.1.1 Требование к структуре

Приложение должно быть построено на трехуровневой архитектуре: клиент (мобильное приложение) – сервер – база данных.

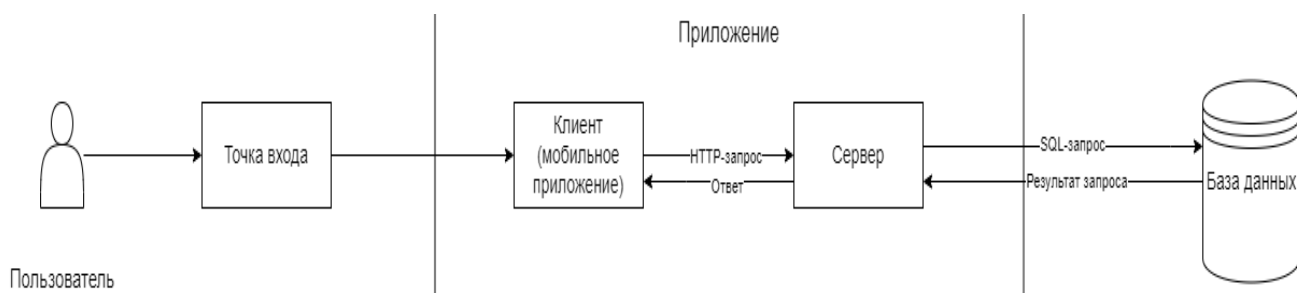


Рисунок 1 - Клиент – Сервер – База данных

3.1.2 Требование к эргономике и технической эстетике

Внешний вид приложения должен соответствовать дизайн-макету. Дизайн-макет будет разработан после утверждения настоящего Технического Задания.

3.1.3 Требование к защите информации

- Обеспечение авторизации и аутентификации пользователей;
- Использование механизмов защиты от SQL-инъекций.

3.1.4 Требование к патентной чистоте

Приложение должно использовать только программное обеспечение с открытым исходным кодом, не нарушая никаких лицензий и патентов. В случае нарушения данного пункта всю ответственность несет сторона Исполнителя.

3.1.5 Требования к используемым технологиям

Ниже приведен перечень используемых технологий, который в ходе разработки может расширяться.

Backend:

- Java - строго типизированный объектно-ориентированный язык программирования общего назначения, разработанный компанией Sun Microsystems;
- Spring Boot — это веб-платформа Java на основе микрослужб с открытым кодом от Spring. Обладает большим набором инструментов;
- PostgreSQL - свободная объектно-реляционная система управления базами данных. Выбрана так как, поддерживает большое количество типов данных, имеет богатую функциональность, поддерживается многими серверами;
- FlyWay - средство обеспечения версионности схем и управления миграциями базы данных. FlyWay имеет поддержку PostgreSQL и интегрируется с Spring Framework;
- Docker - платформа, предназначенная для помощи разработчикам в создании, совместном использовании и запуске современных приложений.

Frontend

- Kotlin - статически типизированный, объектно-ориентированный язык программирования, работающий поверх Java Virtual Machine и разрабатываемый компанией JetBrains;
- Android SDK - универсальное средство разработки мобильных приложений для операционной системы Android.

Инструменты для ведения документации:

- Miro - платформа для совместной работы распределенных команд;
- Swagger - фреймворк для спецификации REST API.

Дополнительный инструментарий:

- Git - распределённая система управления версиями;
- Github - веб-сервис для хостинга IT-проектов и их совместной разработки;

— Trello - облачная программа для управления проектами небольших групп.

3.1.6 Требование к персоналу, обслуживающему приложение

Для обслуживания системы необходимы администраторы. Для старта работы системы достаточно одного администратора. Однако для поддержания работы и привлечения других ВУЗов нужно будет привлекать администраторов, которые будут отвечать за конкретно назначенный им ВУЗ. В обязанности администратора входит:

- Добавление новых ВУЗов, факультетов, аудиторий, групп, сотрудников, старост групп;
- Обновление информации об уже существующих ВУЗах, факультетах, аудиториях, группах, сотрудниках, старостах групп;
- Удаление ВУЗов, факультетов, аудиторий, лабораторий, групп, сотрудников, старост групп.

3.2 Требования к функциям(задачам)

3.2.1 Авторизация

Учётные записи создаются администраторами заранее. Пользователь при входе в систему вводит свою почту и пароль. Приложение не предусматривает сценария для пользователей, не являющихся преподавателями ВУЗа или старостой группы.

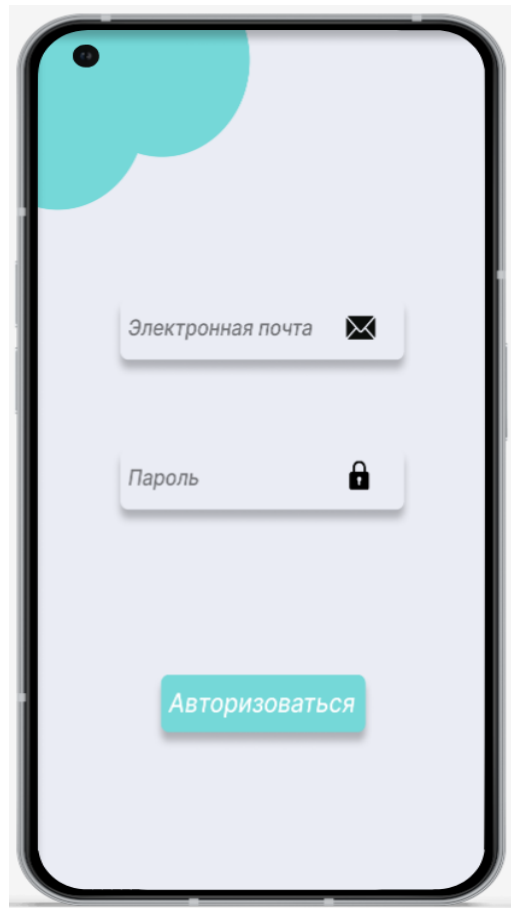


Рисунок 2 - Экран авторизации

3.2.2 Просмотр расписания

Пользователь (преподаватель и староста группы) имеет возможность просматривать расписание. Пользователь с ролью «Староста» может получить расписание своей группы. Пользователь с ролью «Преподаватель» может в свою очередь получить расписание проводимых им лично занятий, то есть свое личное расписание.

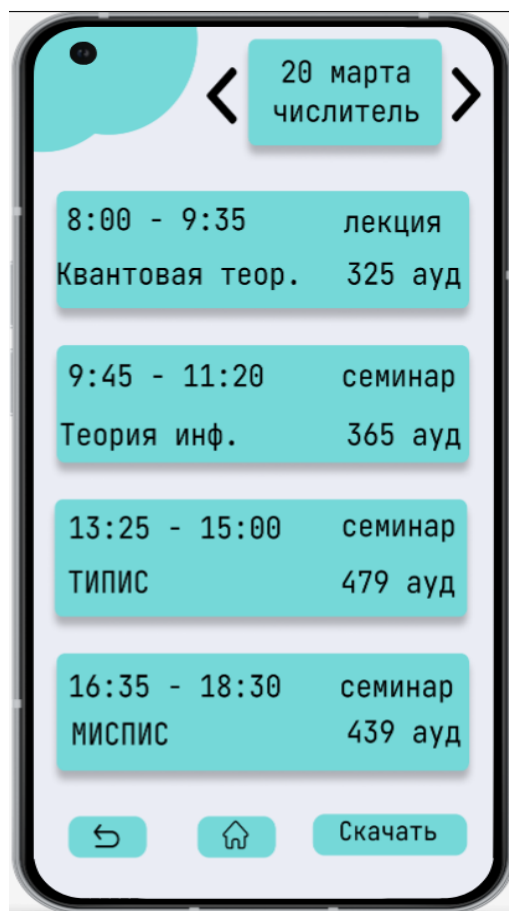


Рисунок 3 - Экран просмотра расписания для студента

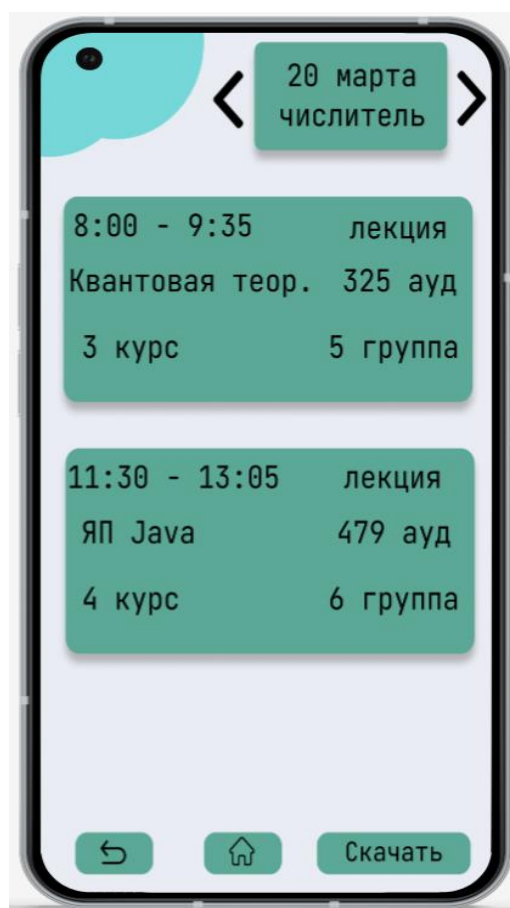


Рисунок 4 - Экран просмотра расписания для преподавателя

3.2.3 Перенос занятий

Пользователь (преподаватель) имеет возможность перенести занятие на другое время или на другой день, предоставляемые системой. При подаче преподавателем заявки на перенос занятия, ему будут предложены все варианты доступного времени для переноса, сначала на оставшиеся дни текущей недели, а в случае недоступности аудиторий с подходящей вместительностью и оборудованием, на следующую неделю. После этого уведомление о переносе приходит на электронную почту пользователю (старосте), связанному с перенесенными занятиями.

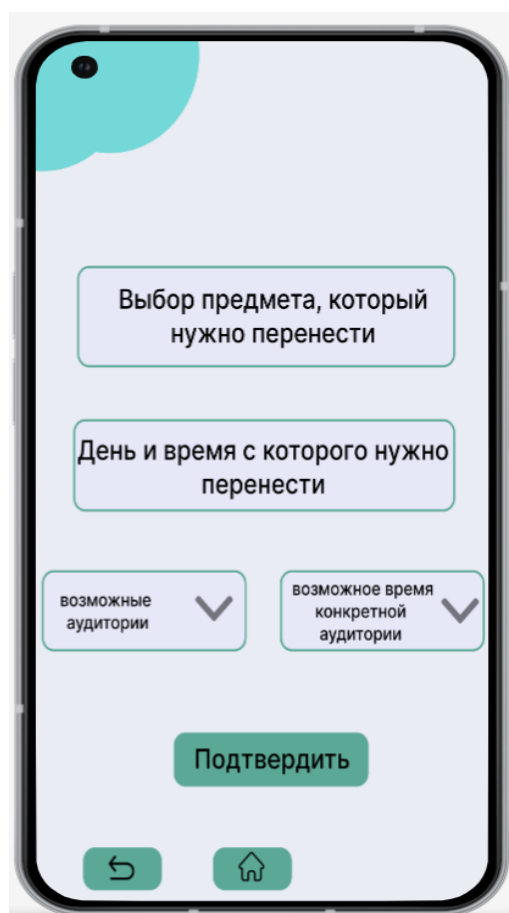


Рисунок 5 - Экран заявки на перенос расписания

3.2.4 Создание расписания

Система генерирует расписание на основе заявок пользователей (преподавателей). Преподаватель добавляет предмет, необходимое количество часов за две недели по числителю и знаменателю конкретной группы. После ввода данных всеми преподавателями приложение сформирует предварительный вариант расписания, учитывая вместительность аудиторий и размеры групп, уведомление об этом будет разослано преподавателям на почту.

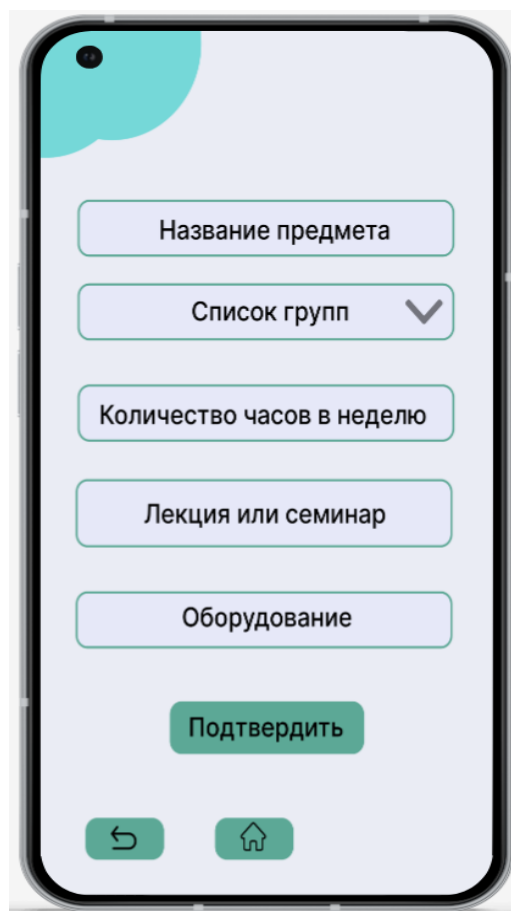


Рисунок 6 - Экран заявки на составление расписания

3.2.5 Скачивание расписания

Пользователь (преподаватель и староста группы) имеет возможность скачать расписание на заданный промежуток времени в формате Excel.

3.3 Требования к поддерживаемым языкам

Приложение будет поддерживать только русский язык.

4 Перспективы развития, модернизации приложения

- Добавление в расписание отображения событий (контрольные работы, аттестации);
- Добавление роли «Студент» и необходимых для неё сценариев
- Отображение домашнего задания к занятиям;
- Добавление поддержки иностранных языков.

5 Состав и содержание работ по созданию приложения

Основные этапы работ, их содержание и приблизительные сроки приведены в Таблице 1.

Таблица 1 - Состав и содержание работ по созданию приложения

Название этапа	Содержание работ	Порядок приемки документов	Сроки
Создание технического задания	Разработка требований к системе	Утверждение технического задания заказчиком	31.03.2023, 21:00 по МСК
Техническое проектирование	Разработка сценариев работы проекта	Описание функционально й схемы в Miro	24.03.2023, 21:00 по МСК
	Разработка дизайна проекта	Изображения дизайн-макета проекта	До 31.03.2023
Разработка программной части	Разработка серверной части	В процессе испытаний	До 31.05.2023
	Разработка системы хранения данных		
	Разработка клиентской части		
Тестирование функциональност и приложения	Проверка соответствия приложения требованиям, указанным в Техническом задании	По техническому заданию	Не позднее 14 дней с завершения предыдущего этапа

	Устранение неисправностей и повторная проверка до полного восстановления функциональности приложения		
Разработка курсового проекта	Разработка Курсового Проекта, содержащего информацию о проекте на основе Технического Задания, отчет о проделанной работе	На основе технического задания	С марта 2023 по июнь 2023
Опытная эксплуатация	Эксплуатация на ограниченной группе пользователей	Ведение отдельного документа	Май – июнь 2023
	Доработка при необходимости и повторная отправка в эксплуатацию		Май - июнь 2023

6 Порядок контроля и приёмки приложения

Контроль разработки приложения осуществляется путем обсуждения текущего этапа разработки среди исполнителей, а также путем встреч с заказчиком.

Готовое приложение со всей необходимой документацией предоставляется заказчику в обозначенный в данном техническом задании срок.

За заказчиком остается осуществление проверки соответствия функциональности приложения согласно Техническому Заданию и приемки приложения. Исполнитель также обязан предоставить заказчику следующий комплект поставки необходимый для защиты проекта:

- Техническое Задание;
- Курсовой Проект;
- Демонстрационная версия приложения со всеми основными сценариями;
- Исходный код приложения;
- Аналитику проекта.

7 Требования к документированию

- Документирование системы ведется в рамках настоящего Технического Задания, составленного в соответствие с ГОСТ 34.602-89;
- Документ об отчете проделанной работы над системой будет представлен в рамках Курсового Проекта на основе настоящего Технического Задания;
- Документирование основных сценариев работы приложения осуществляется в сервисе Miro;
- Документирование серверной части приложения обеспечивается с помощью инструмента Swagger.

8 Источники разработки

1. Руководство по языку Kotlin [Электронный ресурс]. – Режим доступа: <https://metanit.com/kotlin/tutorial/>. - Руководство по языку Kotlin. – (Дата обращения: 23.03.2023).
2. Spring Framework Documentation [Электронный ресурс]. – Режим доступа: <https://spring.io/projects/spring-boot>. – Spring Boot. – (Дата обращения 23.03.2023).
3. Docker Docs [Электронный ресурс]. - Режим доступа: <https://docs.docker.com/>. – Docker overview – (Дата обращения: 23.03.2023).

9 Реквизиты и подписи сторон

СОСТАВИЛИ:

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпись	Дата
Воронежский Государственный Университет	Студент	Свиридов Фёдор Юрьевич		31.03.2023
Воронежский Государственный Университет	Студент	Молдавский Илья Михайлович		31.03.2023
Воронежский Государственный Университет	Студент	Распопов Павел Сергеевич		31.03.2023

СОГЛАСОВАНО:

Наименование организации, предприятия	Должность	Фамилия, имя, отчество	Подпись	Дата
Воронежский Государственный Университет	Старший преподаватель	Тарасов Вячеслав Сергеевич		31.03.2023
Воронежский Государственный Университет	Ассистент	Клейменов Илья Валерьевич		31.03.2023