

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук

Кафедра программирования и информационных технологий

Техническое задание
на разработку мобильного приложения
«FreelanceFinder»

Исполнители

_____ М.И. Капустин

_____ Т.А. Коротаев

_____ О.А. Киреев

Заказчик

_____ В.С. Тарасов

Воронеж 2024

СОДЕРЖАНИЕ

1 Общие сведения.....	4
1.1 Наименование приложения	4
1.2 Наименование заказчика.....	4
1.3 Наименование исполнителя	4
1.4 Основания для разработки.....	4
1.5 Плановые сроки начала и окончания работ	5
1.6 Термины и сокращения	5
2 Цели и назначение создания автоматизированной системы	8
2.1 Цели и назначение приложения	8
2.2 Цели создания приложения	8
3 Характеристика объектов автоматизации	9
4 Требования к автоматизированной системе.....	11
4.1 Требования к структуре и функционированию системы	11
4.2 Показатели назначения системы.....	11
4.3 Требования к защите информации от несанкционированного доступа.....	12
4.4 Требования к аутентификации	13
4.5 Требования к численности и квалификации персонала	13
4.6 Требования к патентной чистоте	13
4.7 Требования к масштабируемости и открытости	14
4.8 Обработка ошибок.....	14
4.9 Требования к функциям, выполняемым системой.....	15
4.10 Требования к видам обеспечения	16
4.10.1 Требования к информационному обеспечению	16
4.10.2 Требования к программному обеспечению приложения	16
4.10.3 Требования к лингвистическому обеспечению.....	17
5 Состав и содержание работ по созданию сервиса	18

6	Порядок контроля и приемки автоматизированной системы	19
7	Порядок разработки	20
8	Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие	21
9	Требования к документированию для системы	23
10	Источники разработки программного обеспечения	24
11	Структура приложения	25
12	Навигация по приложению	26
12.1	Основное навигационное меню	26
13	Описание страниц приложения	27
13.1	Приветственная страница для всех запусков.....	27
13.2	Страница входа	27
13.3	Страница регистрации	27
13.4	Страница профиля	28
13.5	Страница редактирования.....	28
13.6	Страница поиска заказов	28
13.7	Страница поиска фрилансеров.....	29
13.8	Боттом меню	29
14	Функциональные требования.....	30
14.1	Для неавторизованного пользователя:	30
14.2	Для авторизованного пользователя (Фрилансер).....	30
14.3	Для авторизованного пользователя (Заказчик):	30
14.4	Для администратора:	31
15	Нефункциональные требования	31
16	Подписи сторон	33
	ПРИЛОЖЕНИЕ А	34

1 Общие сведения

1.1 Наименование приложения

Полное наименование: FreelanceFinder.

1.2 Наименование заказчика

Старший преподаватель Тарасов Вячеслав Сергеевич, кафедра программирования и информационных технологий.

1.3 Наименование исполнителя

Исполнитель: студент Киреев Олег Александрович. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Капустин Максим Иванович. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и Информационных Технологий.

Исполнитель: студент Коротаев Тимофей Александрович. Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и Информационных Технологий.

1.4 Основания для разработки

Клиенту требуется мобильное приложение, которое будет служить платформой для связи заказчика и фрилансера. Приложение должно предоставлять удобную навигацию, позволять пользователям заказчикам добавлять новые заказы с текстовыми описаниями, а также оставлять обратную связь о пользователе, выполнившем заказ. В свою очередь,

пользователь-фрилансер должен иметь возможность оставить анкету, а также просматривать анкеты других пользователей. Должна быть предусмотрена возможность редактирования своих заказов для заказчиков.

1.5 Плановые сроки начала и окончания работ

Срок начала работ – февраль 2024

Плановый срок окончания работ – май 2024

1.6 Термины и сокращения

Таблица 1 - Термины и сокращения

Мобильное приложение	Программное обеспечение, предназначенное для работы на смартфонах, планшетах и других мобильных устройствах, разработанное для конкретной платформы.
Клиент	Это аппаратный или программный компонент вычислительной системы, посылающий запросы серверу.
Сервер	Выделенный или специализированный компьютер для выполнения сервисного программного обеспечения.
База данных	Упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. База данных обычно управляется системой управления базами данных.

HTTP	Это протокол, позволяющий получать различные ресурсы, например HTML-документы. Протокол HTTP лежит в основе обмена данными в Интернете.
GitHub	Крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки.
Администратор	Человек, имеющий доступ к расширенной функциональности мобильного клиента.
Пользователь	Авторизованный в мобильном клиенте человек, пользующийся функциональностью мобильного клиента.
Гость	Неавторизованный в мобильном клиенте человек, пользующийся ограниченной функциональностью мобильного клиента.
Аутентификация	Процедура проверки подлинности, например, проверка подлинности пользователя путем сравнения введенного им пароля с паролем, сохраненным в базе данных.
Авторизация	Предоставление определенному лицу или группе лиц прав на выполнение определенных действий.
Контент	Наполнение мобильного клиента.
Фреймворк	Программное обеспечение, облегчающее разработку и объединение

	разных компонентов большого программного проекта.
Заказчик	Тип пользователя, который заинтересован в выполнении задачи по разработке программного обеспечения.
Фрилансер	Тип пользователя, который заинтересован в получении материального заработка за выполнение задач по разработке программного обеспечения.
Аккаунт	Персональная страница пользователя или личный кабинет, который создается после регистрации на сайте.
Личный кабинет	Раздел сервиса, в котором Пользователь может получить доступ к своим данным.
API	Описание взаимодействия одной компьютерной программы с другой.
REST API	Стиль архитектуры программного обеспечения для построения распределенных масштабируемых веб-сервисов.
Back-end	Программно-аппаратная часть сервиса.
Front-end	Клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса.

2 Цели и назначение создания автоматизированной системы

2.1 Цели и назначение приложения

Назначением приложения является создание удобной платформы для поиска и найма специалистов с функциями создания, просмотра и редактирования заказов; отклика на заказы и предложением заказа специалисту.

2.2 Цели создания приложения

Цели создания:

- создание удобной и эффективной платформы для пользователей с целью поиска и найма квалифицированных специалистов для выполнения различных проектов и задач;
- обеспечение удобного и прозрачного взаимодействия между заказчиками и специалистами, с целью эффективного осуществления процесса найма и выполнения проектов;
- обеспечение стабильной прибыли компании за счет предоставления услуг в продвижении заказов.

3 Характеристика объектов автоматизации

Для администрирования данного продукта необходим минимум один сотрудник из штата заказчика. Для взаимодействия с данной системой выделяют следующие виды пользователей:

- неавторизованный пользователь;
- авторизованный пользователь (Фрилансер);
- авторизованный пользователь (Заказчик);
- администратор.

Неавторизованный пользователь имеет возможность в процессе взаимодействия с данной системой выполнять следующие действия:

- просмотреть список всех заказов;
- увидеть рейтинг заказчика в приложении;
- просмотреть заказ;
- осуществить поиск заказов в приложении;
- зарегистрироваться;
- осуществить поиск фрилансеров в приложении;
- увидеть рейтинг фрилансера в приложении.

Авторизованный пользователь имеет возможность в процессе взаимодействия с данной системой выполнять следующие действия:

- возможности неавторизованного пользователя;
- изменение персональных данных;
- возможность добавить, изменить или удалить заказ;
- возможность оставить обратную связь фрилансеру, выполнившему заказ;
- возможность предложить фрилансеру заказ;

— возможность выбрать фрилансера для выполнения из откликов.

Авторизованный пользователь-фрилансер имеет возможность в процессе взаимодействия с данной системой выполнять следующие действия:

- возможности неавторизованного пользователя;
- изменение персональных данных;
- возможность откликнуться на заказ;
- возможность принять предложение заказа;
- возможность оставить обратную связь о заказчике.

Администратор имеет возможность в процессе взаимодействия с данной системой выполнять следующие действия:

- блокировать заказы;
- блокировать пользователей с автоматическим удалением его заказов.

4 Требования к автоматизированной системе

4.1 Требования к структуре и функционированию системы

Пользователь взаимодействует с серверной частью приложения при помощи клиентского приложения (мобильное приложение). Сервер посредством REST API возвращает пользователю необходимые данные. Администратор взаимодействует с серверной частью приложения при помощи клиентского приложения (мобильное приложение).

Серверная часть приложения включает в себя:

- сервис регистрации и авторизации;
- API сервис;
- сервис для работы с базой данных;
- сервис обработки данных.

Back-end:

- Язык программирования: Kotlin 1.6;
- Фреймворк: Spring 6.1.4;
- СУБД: PostgreSQL 16.2;
- ORM: Hibernate;
- Средство построения REST API: Spring Boot 3.2.3;
- Средство авторизации и аутентификации: Spring Security/

Front-end:

- Язык программирования: Dart 3.2.6;
- Фреймворк: Flutter 3.16.9;
- Средство авторизации и аутентификации: JWT.

4.2 Показатели назначения системы

- реализация просмотра списка всех доступных заказов и профилей

- фрилансеров на платформе;
- реализация добавления и удаления профиля фрилансера после авторизации в системе;
- реализация добавления и удаления профиля заказчика после авторизации в системе;
- реализация возможности просмотра рейтинга и отзывов о профилях фрилансеров;
- реализация возможности просмотра рейтинга и отзывов о профилях заказчиков;
- реализация поиска профилей фрилансеров и их работ по различным критериям;
- реализация поиска заказов по различным критериям.

4.3 Требования к защите информации от несанкционированного доступа

Spring включает несколько механизмов защиты по умолчанию. Вот некоторые из них:

- CSRF-защита (Cross-SiteRequestForgery): Spring автоматически генерирует уникальный токен для каждой сессии пользователя и вставляет его в формы. При отправке формы этот токен проверяется для предотвращения атак CSRF;
- защита от атак XSS (Cross-SiteScripting): Spring Security может предотвратить атаки XSS по умолчанию, блокируя выполнение вредоносных скриптов на стороне клиента;
- ограничение доступа к ресурсам: Spring Security предоставляет возможность ограничения доступа к определенным ресурсам вашего REST API с использованием ролей и разрешений. Это позволяет

определить, какие пользователи могут выполнять определенные операции.

4.4 Требования к аутентификации

- В системе должна быть реализована идентификация и проверка доступа при входе в систему по логину, почте и паролю длиной не менее 6 символов.
- Система защиты должна подвергаться проверке подлинности идентификации — осуществлять аутентификацию.
- Система не должна предоставлять доступ к защищенным данным неавторизованным пользователям.
- Система защиты должна обладать способностью надежно связывать полученную идентификацию со всеми действиями данного пользователя.

4.5 Требования к численности и квалификации персонала

- В системе конкретных требований к численности персонала не приводится.
- В системе предполагается наличие ролей пользователей — администратор, авторизованный и неавторизованный пользователь.
- Пользователь с ролью администратор должен обладать знаниями и навыками необходимыми для поддержания работоспособности системы.
- Авторизованные и неавторизованные пользователи, работающие с данной системой, должны обладать базовыми навыками работы на мобильном устройстве

4.6 Требования к патентной чистоте

Проект не должен нарушать никаких патентных прав и лицензий.

4.7 Требования к масштабируемости и открытости

Проект должен быть спроектирован таким образом, чтобы обеспечивать гибкость и масштабируемость. При добавлении новой функциональности необходимо стремиться к минимизации изменений в существующем программном коде, чтобы сократить риски появления новых ошибок и снизить затраты на тестирование и отладку.

4.8 Обработка ошибок

Приложение должно обеспечивать вывод соответствующих сообщений об ошибках в случае их возникновения. Для этого необходимо предусмотреть обработку основных типов ошибок:

- некорректный ввод данных;
- системный сбой.

Если пользователь вводит неверные данные, например, неправильный формат электронной почты или пароля, то приложение должно выводить ясные и конкретные сообщения об ошибке. Например:

«Неверный формат электронной почты. Пожалуйста, проверьте введённый адрес и попробуйте снова.»

«Пароль должен содержать не менее 8 символов, включая буквы верхнего и нижнего регистра, цифры и специальные символы.»

Системный сбой:

В случае системного сбоя, когда приложение не может обработать запрос пользователя из-за технических проблем, должны выводиться

соответствующие сообщения об ошибке с указанием кода ошибки. Например:

«Извините, произошла ошибка сервера (500). Наши специалисты уже работают над её устранением. Попробуйте повторить попытку позже.»

«Не удалось установить соединение с сервером (502). Пожалуйста, проверьте ваше подключение к интернету и повторите попытку.»

4.9 Требования к функциям, выполняемым системой

Приложение должно предоставлять следующие возможности для неавторизованного пользователя:

- возможность осуществить поиск заказов и фрилансеров по фильтрам;
- возможность просмотра заказа, профиля фрилансера и заказчика.

Приложение должно предоставлять следующие возможности для авторизованного пользователя (Фрилансера):

- возможность осуществить поиск заказов и фрилансеров по фильтрам;
- возможность редактирования персональных данных в личном кабинете;
- возможность просмотра заказа, профиля фрилансера и заказчика;
- возможность откликнуться на заказ;
- возможность принять заказ;
- возможность оставить отзыв о заказчике выполненного заказа.

Приложение должно предоставлять следующие возможности для авторизованного пользователя (Заказчика):

- возможность осуществить поиск заказов и фрилансеров по фильтрам;
- возможность редактирования персональных данных в личном кабинете;
- возможность просмотра заказа, профиля фрилансера и заказчика;

- возможность размещения заказа;
- возможность принятия отклика на заказ от фрилансера;
- возможность оставить отзыв об исполнителе выполненного заказа.

Приложение должно предоставлять следующие возможности для администратора системы:

- возможность осуществить поиск заказов и фрилансеров по фильтрам;
- возможность просмотра заказа, профиля фрилансера и заказчика;
- возможность редактировать и удалять информацию о заказах и пользователях;
- возможность просмотра профилей и заказов, которые были отправлены на модерацию.

4.10 Требования к видам обеспечения

4.10.1 Требования к информационному обеспечению

Требования по применению систем управления базами данных: PostgreSQL.

4.10.2 Требования к программному обеспечению приложения

Требования к программному обеспечению клиентской части:

- приложение должно устанавливаться и работать на мобильных устройствах под управлением операционной системы Android.

Требования к программному обеспечению серверной части:

- серверная часть приложения должна быть реализована на языке программирования Kotlin с использованием фреймворка Spring;
- в качестве СУБД необходимо использовать PostgreSQL.

В результате разработки данные требования могут расширяться.

4.10.3 Требования к лингвистическому обеспечению

Приложение должно поддерживать русский язык. Язык ввода-вывода данных русский.

5 Состав и содержание работ по созданию сервиса

Состав и содержание работ по созданию сервиса включают в себя следующие этапы:

- сбор необходимой информации, постановка целей, задач системы, которые в будущем должны быть реализованы 16.02.24 – 01.03.24;
- анализ предметной области, анализ конкурентов и построение структуры требований, ведущих к решению поставленных задач и целей 01.03.24 – 13.03.24;
- построение модели программы, описание спецификаций данных, определение связей между сущностями, разработка модели БД 13.03.24 - 23.04.24;
- разработка рабочего проекта, состоящего из написания программного кода, отладки и корректировки кода программы 13.03.24 – 01.05.24;
- проведение тестирования программного обеспечения 16.05.24 – 01.06.24.

6 Порядок контроля и приемки автоматизированной системы

Предварительные отчёты по работе будет проводиться во время рубежных аттестаций:

- 1 аттестация (середина марта 2024) - создан репозиторий проекта на GitHub, распределены задачи проекта в таск-менеджере Trello, создан проект Miro с общей логикой системы, предоставлены промежуточные результаты по курсовому проекту и готовое техническое задание;
- 2 аттестация (конец апреля 2024) – написан программный код, реализующий 100% описанной функциональности, реализована БД и ее взаимодействие с сервером;
проведена отладка и доработка программного кода, проведено тестирование по работе системы;
- 3 аттестация (конец мая 2024) - разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

7 Порядок разработки

- инициация проекта: сбор информации о проекте, определение целей, задач и требований к системе;
- сбор требований: анализ предметной области, изучение потребности пользователей, бизнес-процессы и конкуренты;
- проектирование: разработка архитектуры системы, модели данных и базы данных, проектирование интерфейсов, алгоритмов и структуры программного кода;
- разработка: разработка программного обеспечения в соответствии с техническим заданием. Этот этап включает в себя написание программного кода, отладку и корректировку программы;
- тестирование: проводится тестирование системы, которое включает в себя функциональное, нагрузочное, регрессионное и другие виды тестирования;
- развертывание: происходит развертывание на рабочих серверах;
- сопровождение и поддержка: осуществляется мониторинг работы системы, устранение возникающих ошибок и неполадок, а также внесение изменений и дополнений в программное обеспечение в соответствии с изменяющимися требованиями заказчика.

8 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие

- анализ готовности объекта автоматизации: проводится оценка готовности объекта автоматизации к вводу автоматизированной системы в действие;
- подготовка технической документации: разрабатывается техническая документация, необходимая для ввода автоматизированной системы в действие. Подготовка аппаратного обеспечения: проводится подготовка аппаратного обеспечения, необходимого для работы автоматизированной системы;
- установка и настройка программного обеспечения: проводится установка и настройка программного обеспечения, необходимого для работы автоматизированной системы;
- тестирование и отладка: после установки и настройки программного обеспечения проводится тестирование и отладка автоматизированной системы;
- подготовка персонала: проводится подготовка персонала, который будет эксплуатировать автоматизированную систему;
- ввод автоматизированной системы в действие: после завершения всех работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие происходит ее ввод в эксплуатацию;
- сопровождение и поддержка: после ввода автоматизированной системы в действие начинается этап сопровождения и поддержки. В этот период мониторятся работа системы, устраняются ошибки и неполадки, а также вносятся изменения и дополнения в ПО по изменяющимся требованиям заказчика.

9 Требования к документированию для системы

- техническое задание: Документ, содержащий общие требования к системе, описание функциональности, требования к аппаратному и программному обеспечению, требования к интерфейсу и взаимодействию серверной и мобильной частей системы;
- описание архитектуры системы: Документ, содержащий описание архитектуры системы, включая серверную и мобильную части, взаимодействие между ними, используемые технологии и протоколы, схему базы данных и т.д.;
- описание интерфейсов: Документ, содержащий описание интерфейсов системы, включая графический интерфейс пользователя, API-интерфейсы, протоколы обмена данными и т.д.;
- справочная информация: Документ, содержащий справочную информацию по системе, включая глоссарий терминов, описание используемых технологий, алгоритмов и т.д.

10 Источники разработки программного обеспечения

- требования заказчика;
- нормативные документы;
- анализ бизнес-процессов;
- исследование рынка.

11 Структура приложения

Приложение должно содержать следующие страницы/виды страниц:

- приветственная страница;
- страница входа;
- страница поиска заказов;
- страница заказа;
- страница поиска фрилансеров;
- страница профиля пользователя;
- страница редактирования профиля;
- страница регистрации;
- боттом меню;
- страница откликов;
- страница предложений задач;
- страница обращений администратору.

12 Навигация по приложению

12.1 Основное навигационное меню

Основным навигационным меню должно выступать нижнее меню (боттом меню).

Нижнее меню должно включать в себя следующие кнопки (для неавторизованного пользователя):

- заказы;
- фрилансеры;
- личный кабинет.

При переходе на страницу профиля пользователю будет высвечиваться уведомление с предложением зарегистрироваться или войти в систему.

Нижнее меню должно включать в себя следующие кнопки (для авторизованного пользователя):

- заказы;
- фрилансеры;
- личный кабинет.

13 Описание страниц приложения

13.1 Приветственная страница для всех запусков

Доступна каждому пользователю при любом запуске приложения.
Содержит логотип приложения.

13.2 Страница входа

Данная страница доступна для авторизованных пользователей.
Содержит:

- поле ввода логина;
- поле ввода почты пользователя;
- поле ввода пароля;
- поле повтора пароля;
- ссылка на регистрацию;
- ссылка на восстановление пароля;
- кнопка входа;
- кнопка продолжения без входа.

13.3 Страница регистрации

Данная страница доступна для неавторизованных пользователей.
Содержит:

- поле ввода логина;
- поле ввода почты пользователя;
- поле ввода пароля;
- поле повторного ввода пароля;
- переключатель выбора роли;

- ссылка на вход;
- кнопка зарегистрироваться;
- кнопка продолжения без регистрации.

13.4 Страница профиля

Данная страница доступна из боттом меню. В правом верхнем углу расположена иконка редактирования. В верхней части страницы находится имя пользователя.

Также в профиле имеется три раздела:

- статическая информация о пользователе;
- текущий заказ;
- все выполненные заказы.

13.5 Страница редактирования

Страница редактирования доступна из профиля пользователя. Отображает общие поля для редактирования.

13.6 Страница поиска заказов

Поиск позволяет фрилансерам эффективно находить заказы, используя ключевые слова или фильтры. Страница поиска имеет следующую функциональность:

- поле поиска. Пользователи могут вводить ключевые слова для поиска заказов;
- результаты поиска. Показываются карточки заказов с краткой информацией о них.

13.7 Страница поиска фрилансеров

Эта страница предоставляет пользователям возможность просмотра профилей фрилансеров.

Функции страницы:

- поиск фрилансеров по фильтрам;
- список фрилансеров с краткой информацией о них.

13.8 Боттом меню

Нижнее меню, доступное всем пользователям с главной страницы. Содержит кнопки для перехода на все основные страницы приложения. Ниже перечислены различные кнопки, которое может содержать боттом меню.

- поиск заказов;
- поиск фрилансеров;
- личный кабинет.

14 Функциональные требования

Для каждой группы пользователей предусмотрены свои функции.

14.1 Для неавторизованного пользователя:

- просмотр заказов: Возможность просматривать заказы без возможности отклика;
- просмотр фрилансеров: Возможность просматривать фрилансеров без возможности предложения заказа;
- регистрация и вход: Возможность регистрации нового аккаунта. Вход в существующий аккаунт для получения полного доступа к функциям приложения.

14.2 Для авторизованного пользователя (Фрилансер):

- отклик на заказ: Возможность просмотра заказов и отклика на интересующий;
- просмотр других фрилансеров;
- принятие заказа: Возможность принять заказ, предложенный заказчиком;
- сбор отзывов: Возможность оставить обратную связь о заказчике;
- личный профиль: Возможность редактировать личные данные, включая имя пользователя, почту, информацию о пользователе, контактную информацию, стоимость и пароль.

14.3 Для авторизованного пользователя (Заказчик):

- просмотр заказов: Возможность просмотра заказов без возможности отклика;
- предложение фрилансеру заказа: Возможность просмотра фрилансеров с возможностью оставить предложение о выполнении заказа;
- личный профиль: Возможность редактировать личные данные, включая имя пользователя, почту, информацию о пользователе, контактную информацию и пароль;
- принятие отклика: Возможность принять отклик фрилансера;
- сбор отзывов: Возможность оставить обратную связь о фрилансера.

14.4 Для администратора:

- управление пользователями: Возможность просматривать и удалять пользовательские аккаунты;
- управление заказами: Возможность редактировать и удалять заказы пользователей;
- рассмотрение жалоб: Возможность редактировать и удалять заказы и пользовательские аккаунты по поступившей жалобе.

15 Нефункциональные требования

- безопасность – приложение должно обеспечивать достаточную защиту данных пользователей;
- полезность – приложение должно помогать пользователям решать их задачи;
- удобство использования – приложение должно быть просто в работе;
- масштабируемость – приложение должно иметь возможность легко расширяться и дополнять функциональность.

16 Подписи сторон

ЗАКАЗЧИК:

Ст. преп. Тарасов В.С.

_____/_____/

«_____» _____ 20____ г.

ИСПОЛНИТЕЛЬ:

Рук. команды 6-4 Капустин М.И.

_____/_____/

Разработчик Коротаев Т.А.

_____/_____/

Разработчик Киреев О.А.

_____/_____/

«_____» _____ 20____ г.

ПРИЛОЖЕНИЕ А

Диаграммы, иллюстрирующие работу системы:

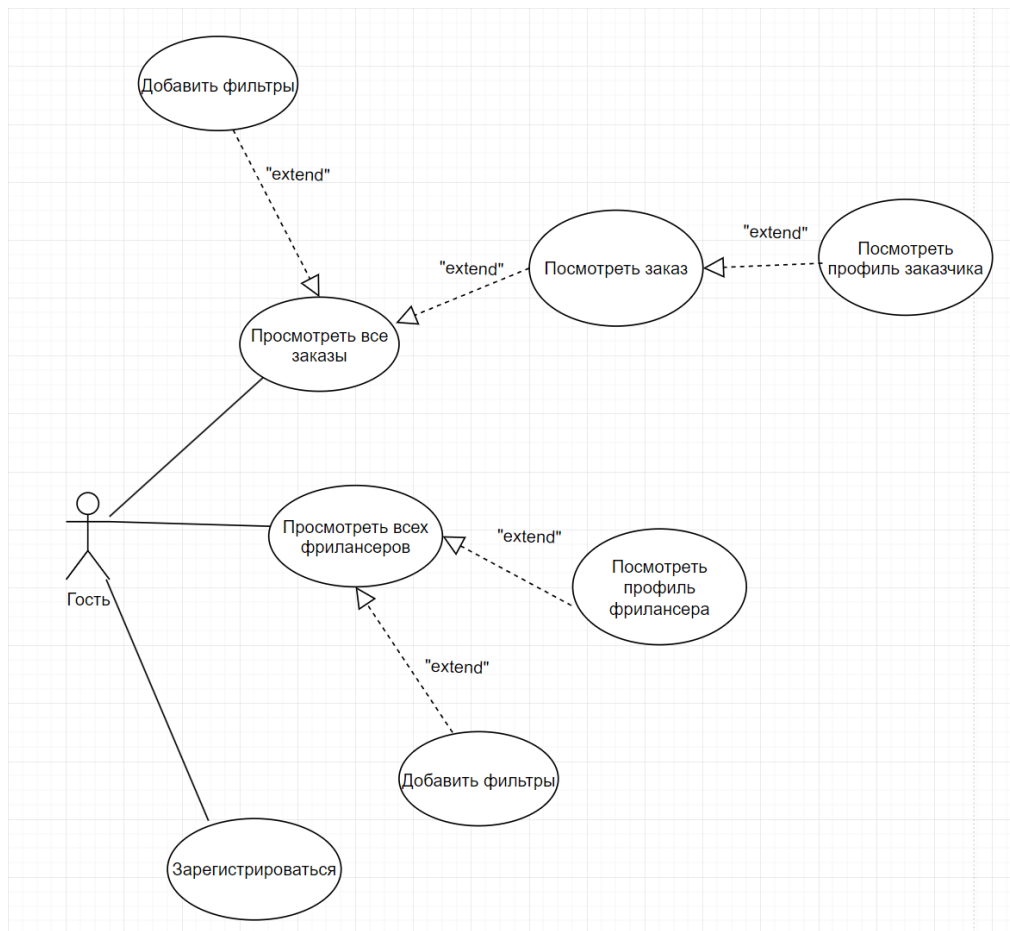


Рисунок 1 - Диаграмма прецедентов гостя

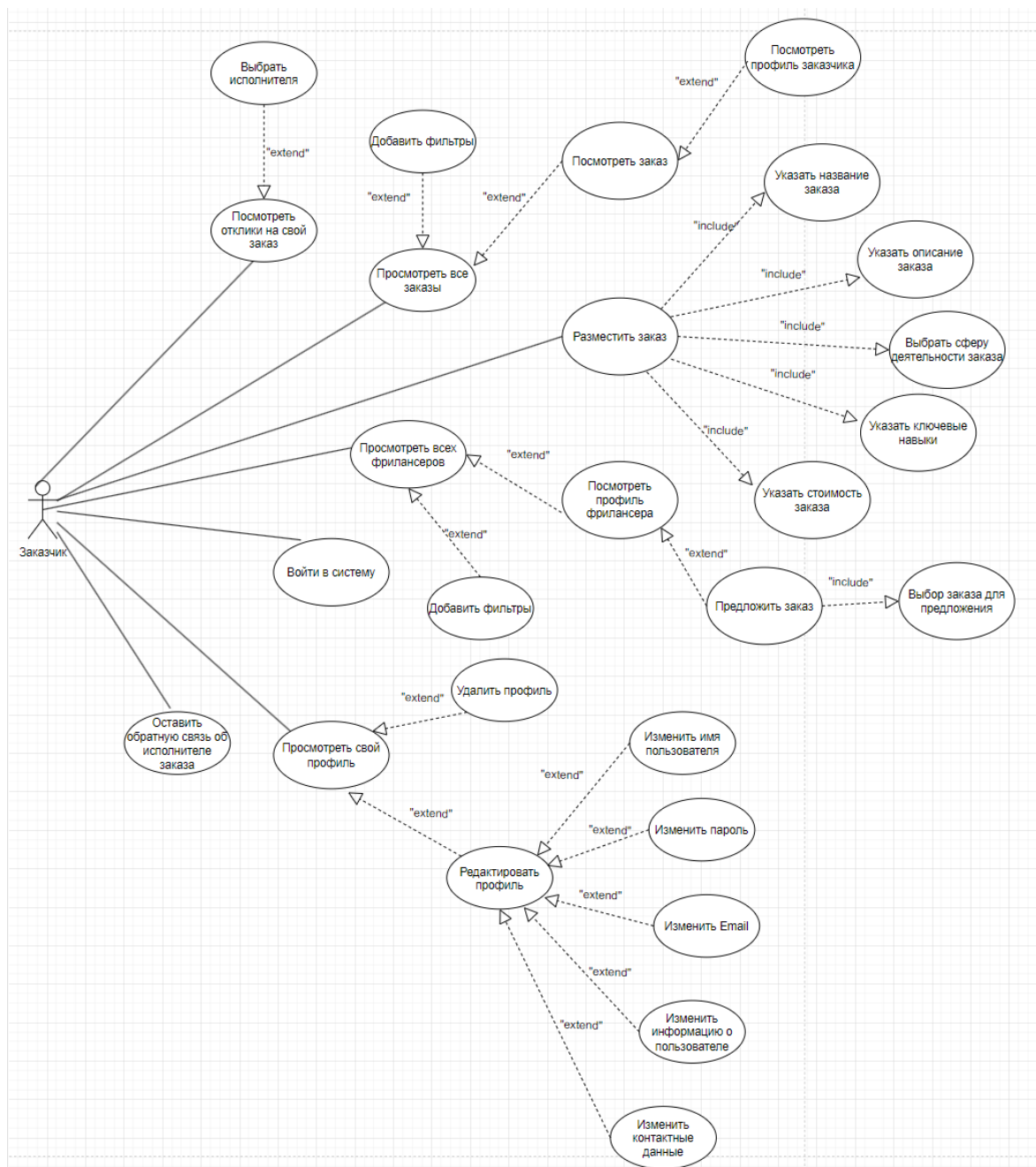


Рисунок 2 - Диаграмма прецедентов заказчика

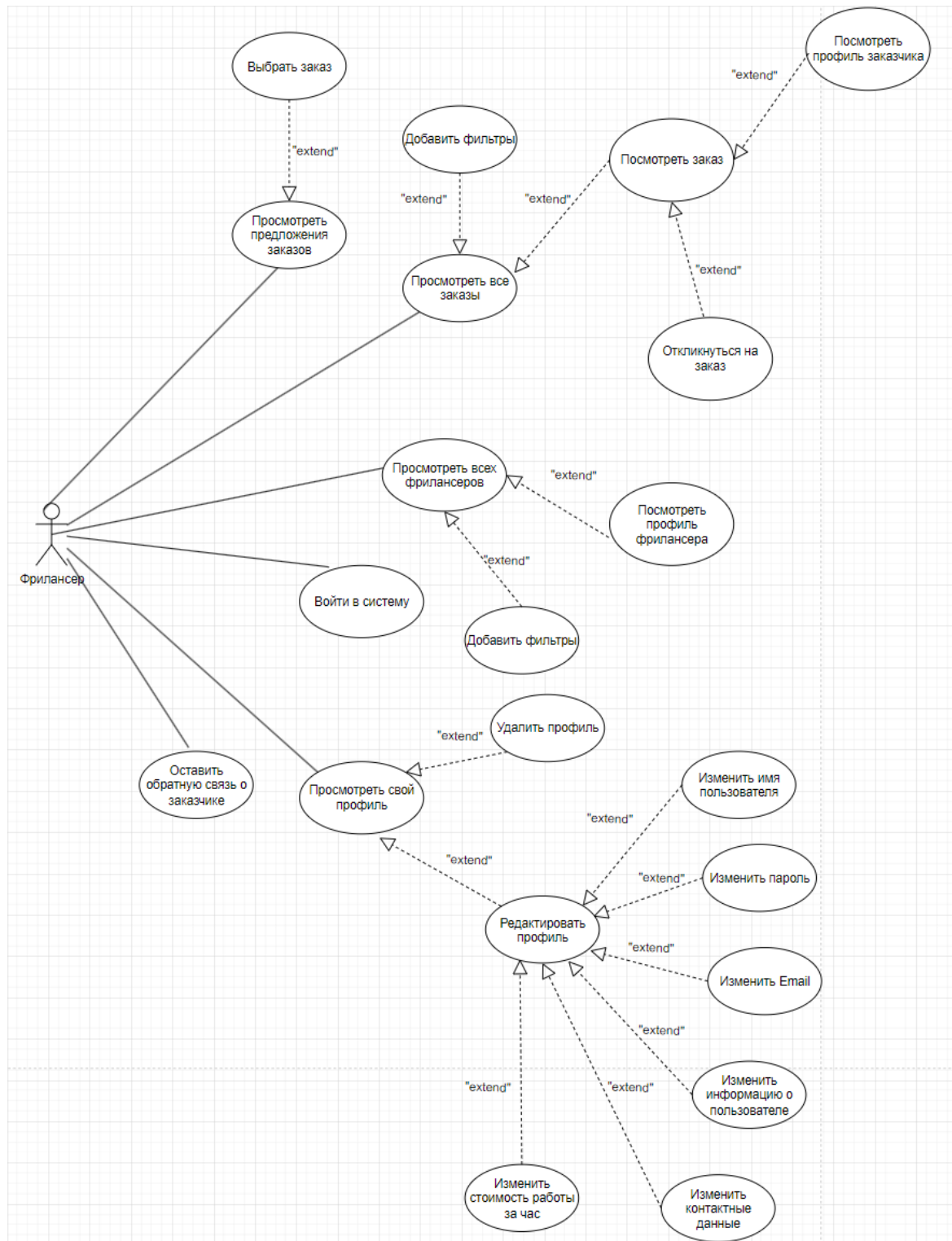


Рисунок 3 - Диаграмма прецедентов фрилансера

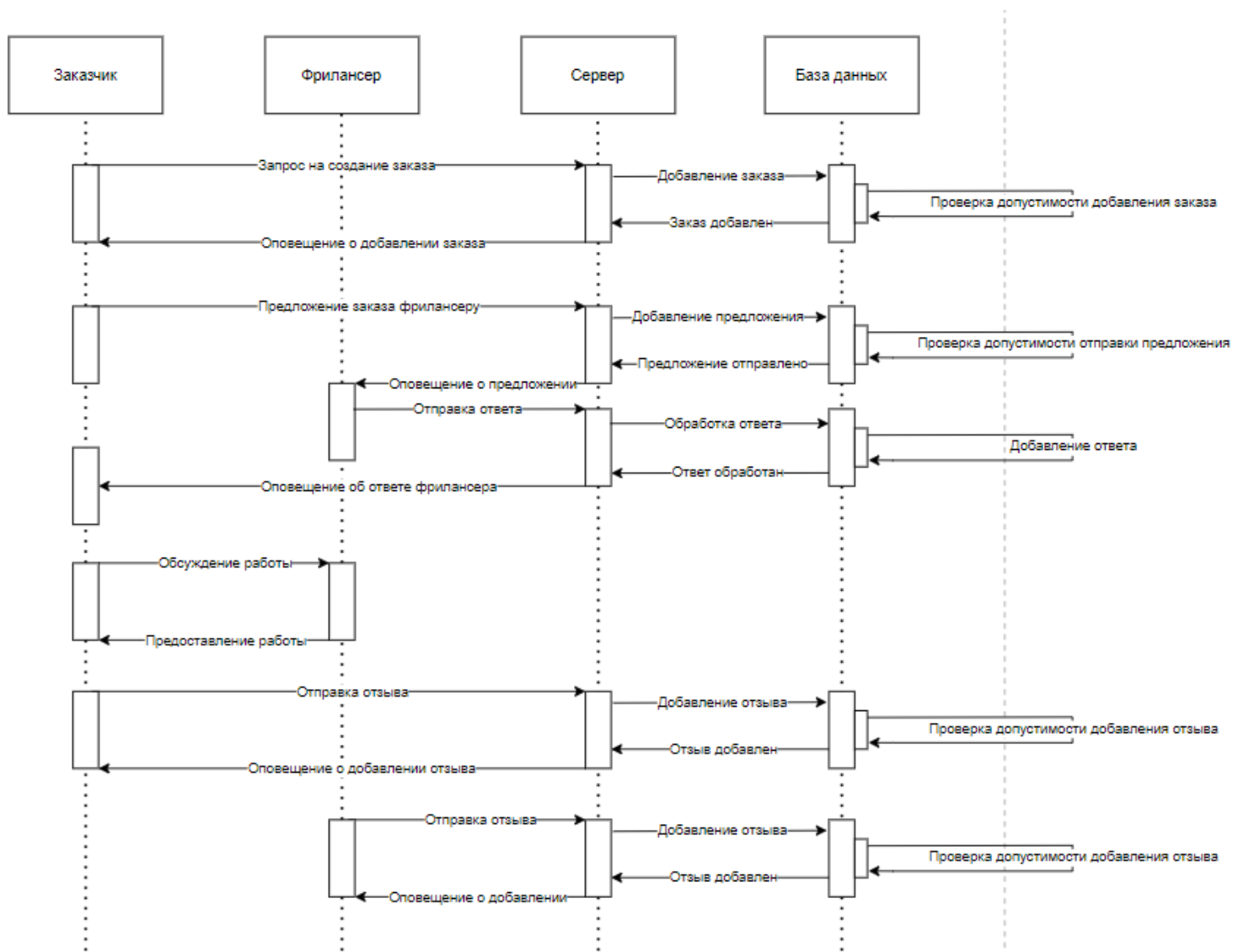


Рисунок 5 - Диаграмма последовательности

Состояния для фрилансера

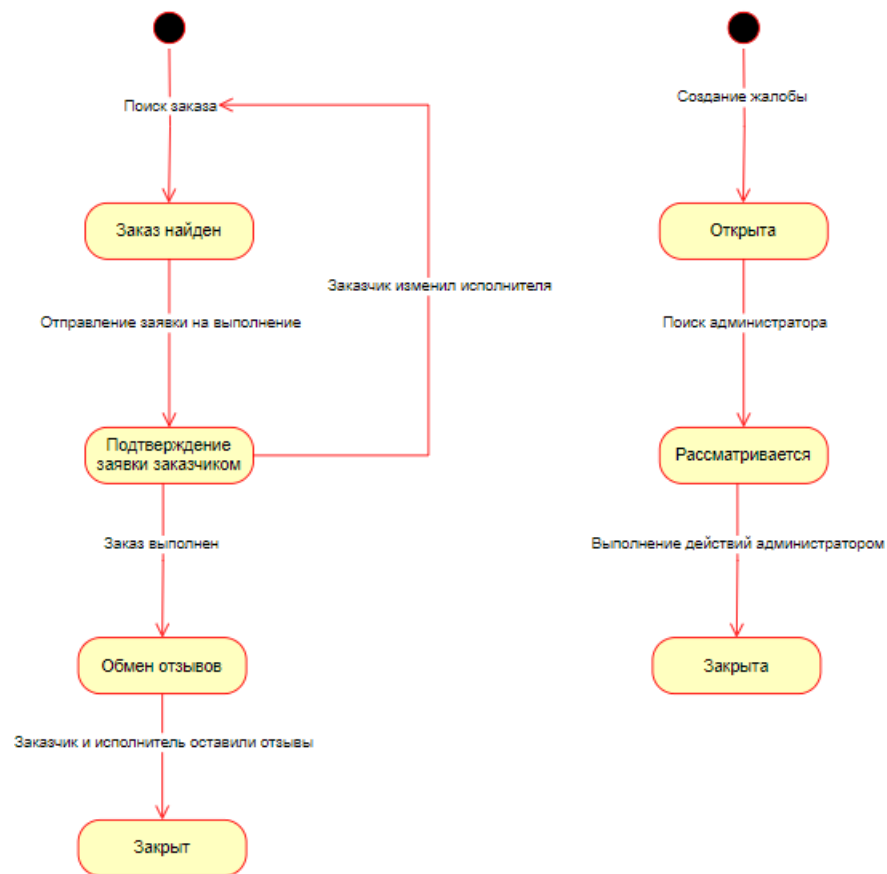


Рисунок 6 - Диаграмма состояний фрилансера

Состояния для гостя

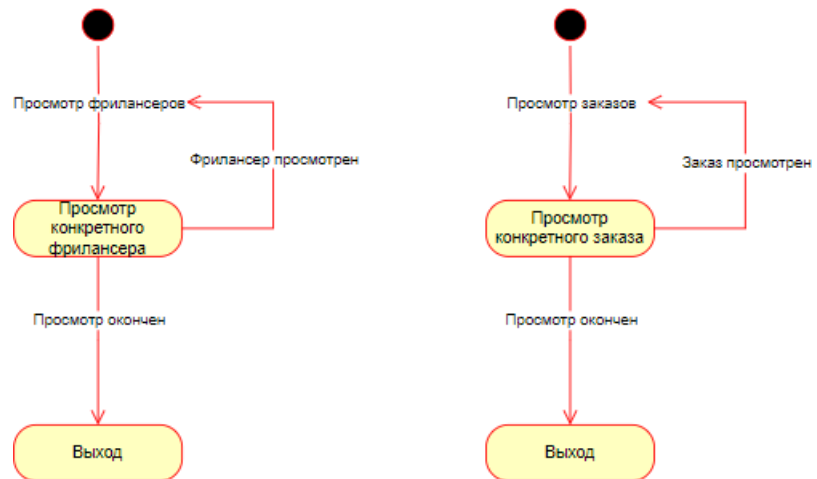


Рисунок 7 - Диаграмма состояний гостя

Состояния для заказчика

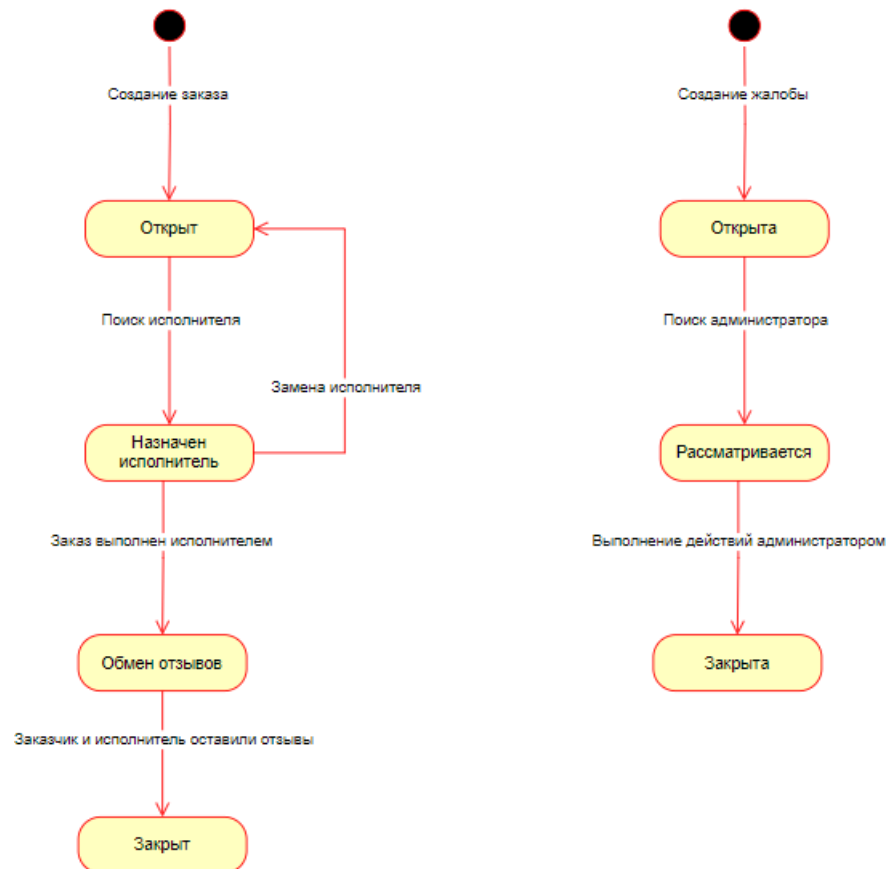


Рисунок 8 - Диаграмма состояний заказчика

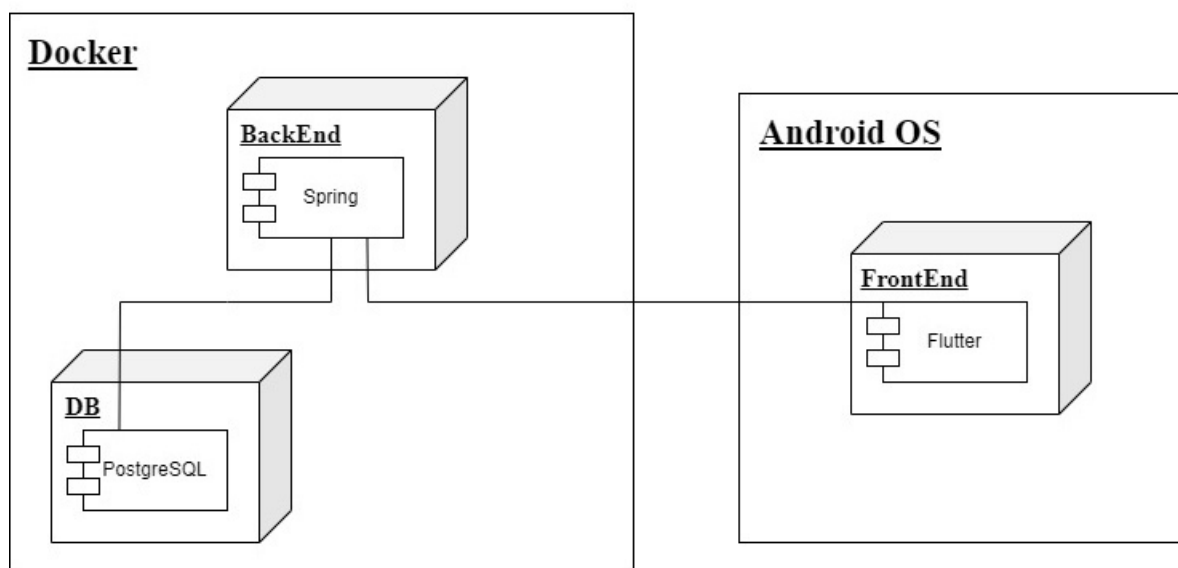


Рисунок 9 - Диаграмма развёртывания

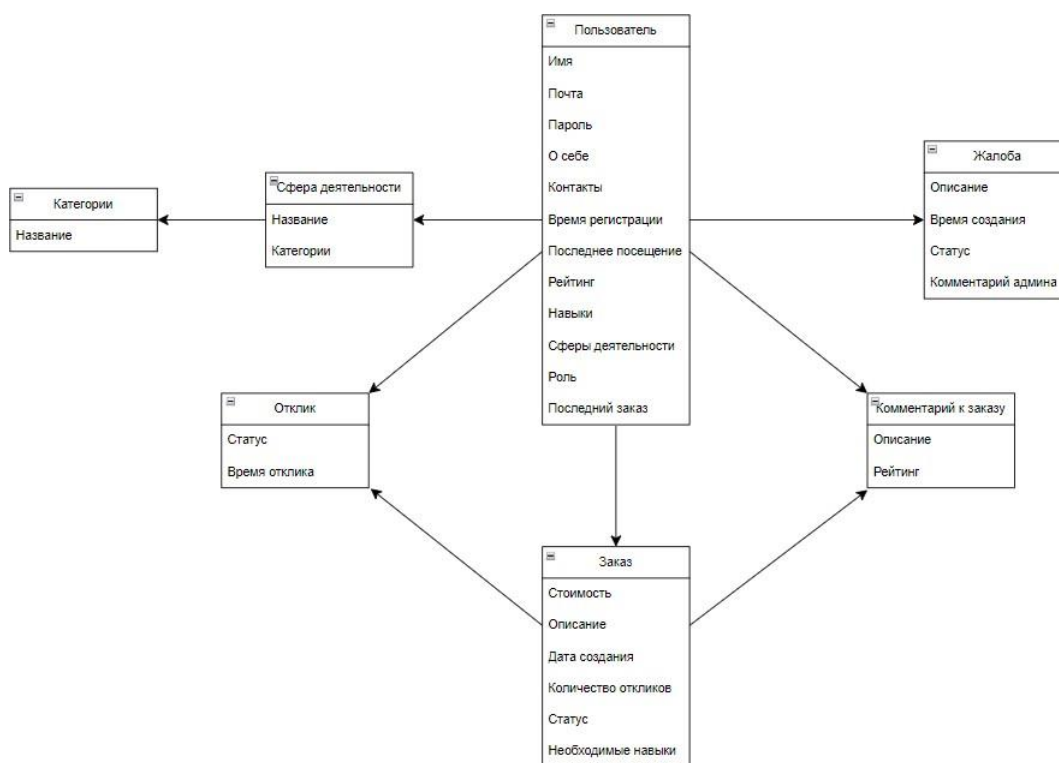


Рисунок 10 - Диаграмма объекта

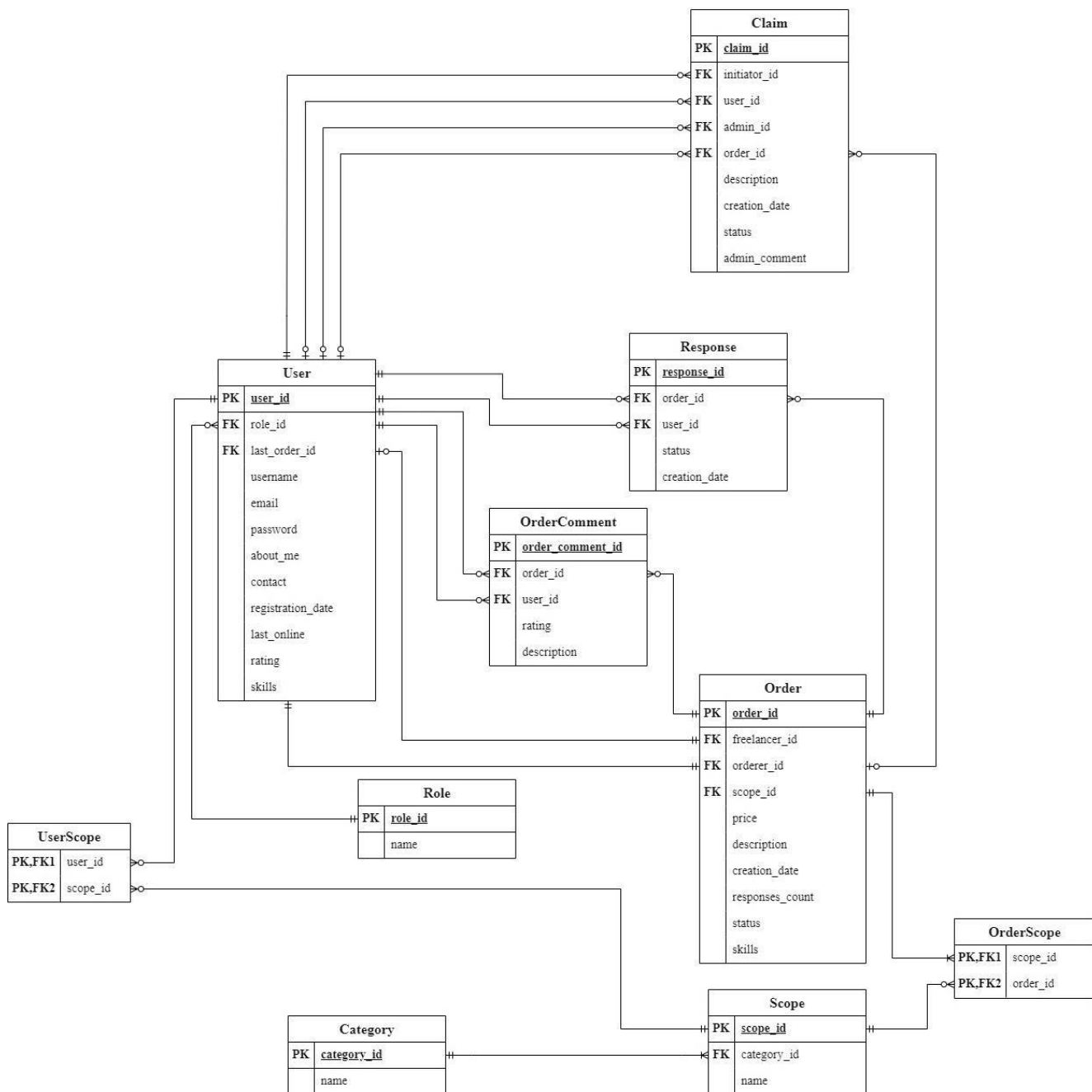


Рисунок 11 - Диаграмма ER

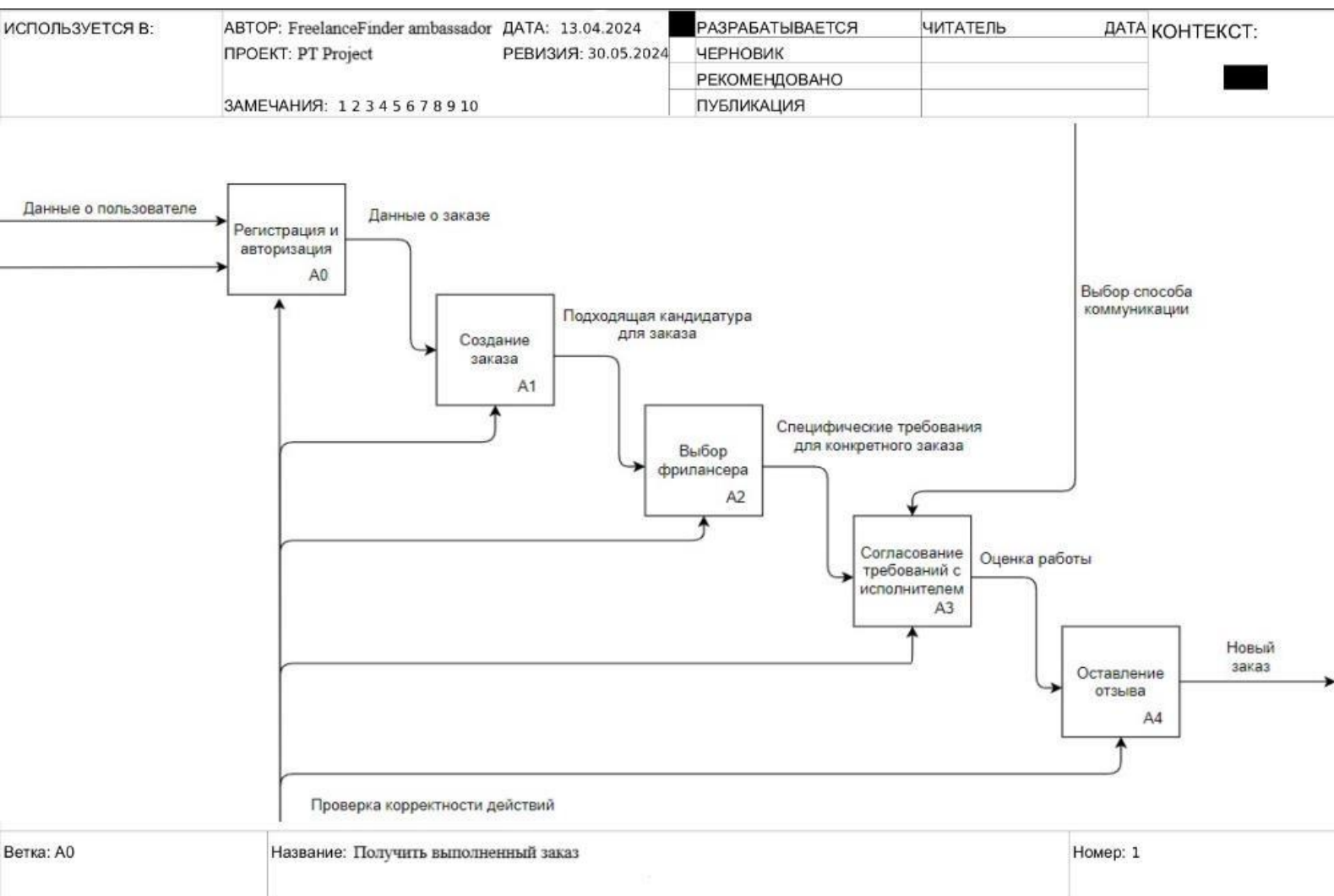


Рисунок 12 - Диаграмма IDEF0-1

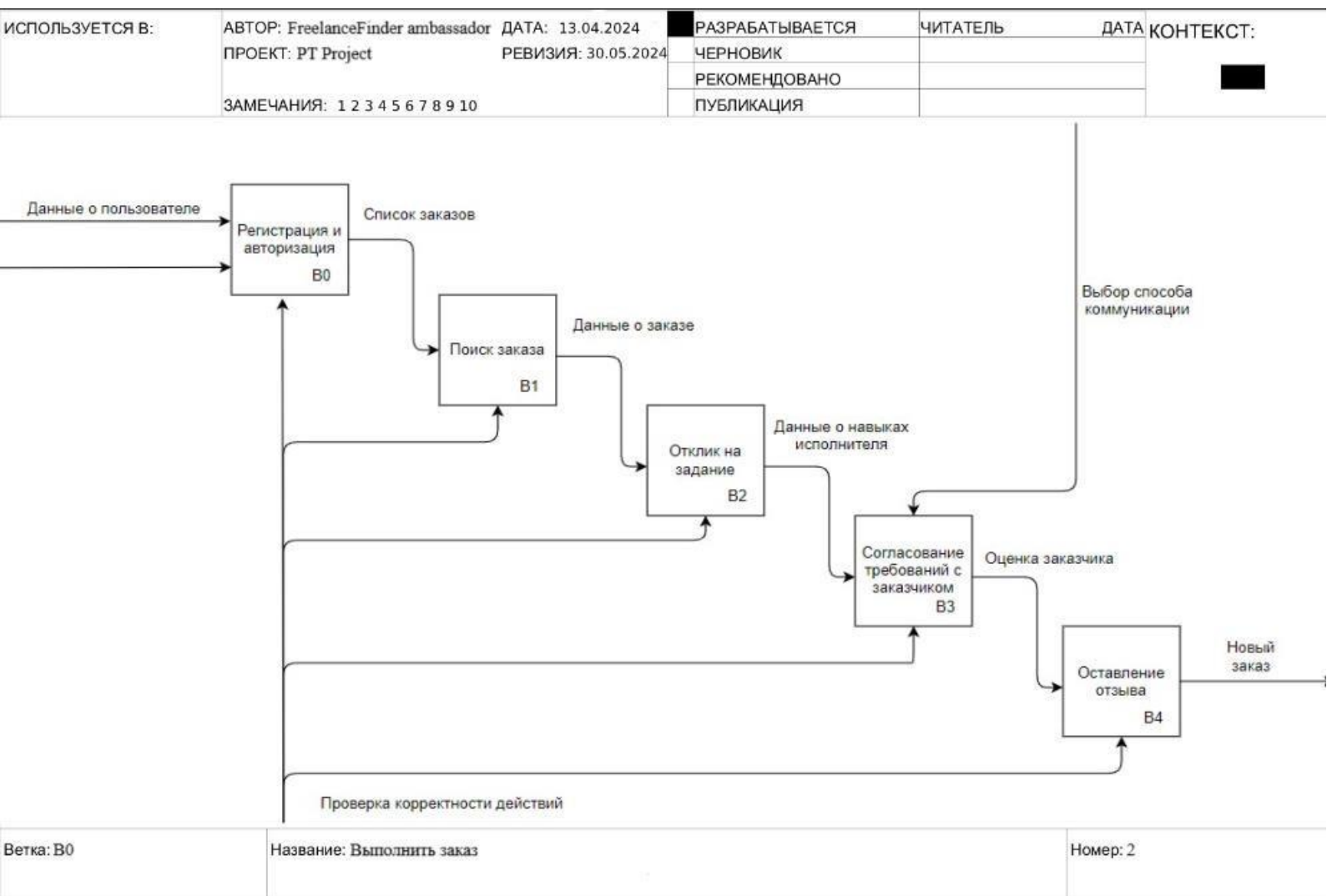


Рисунок 13 - Диаграмма IDEF0-2

Активность админа

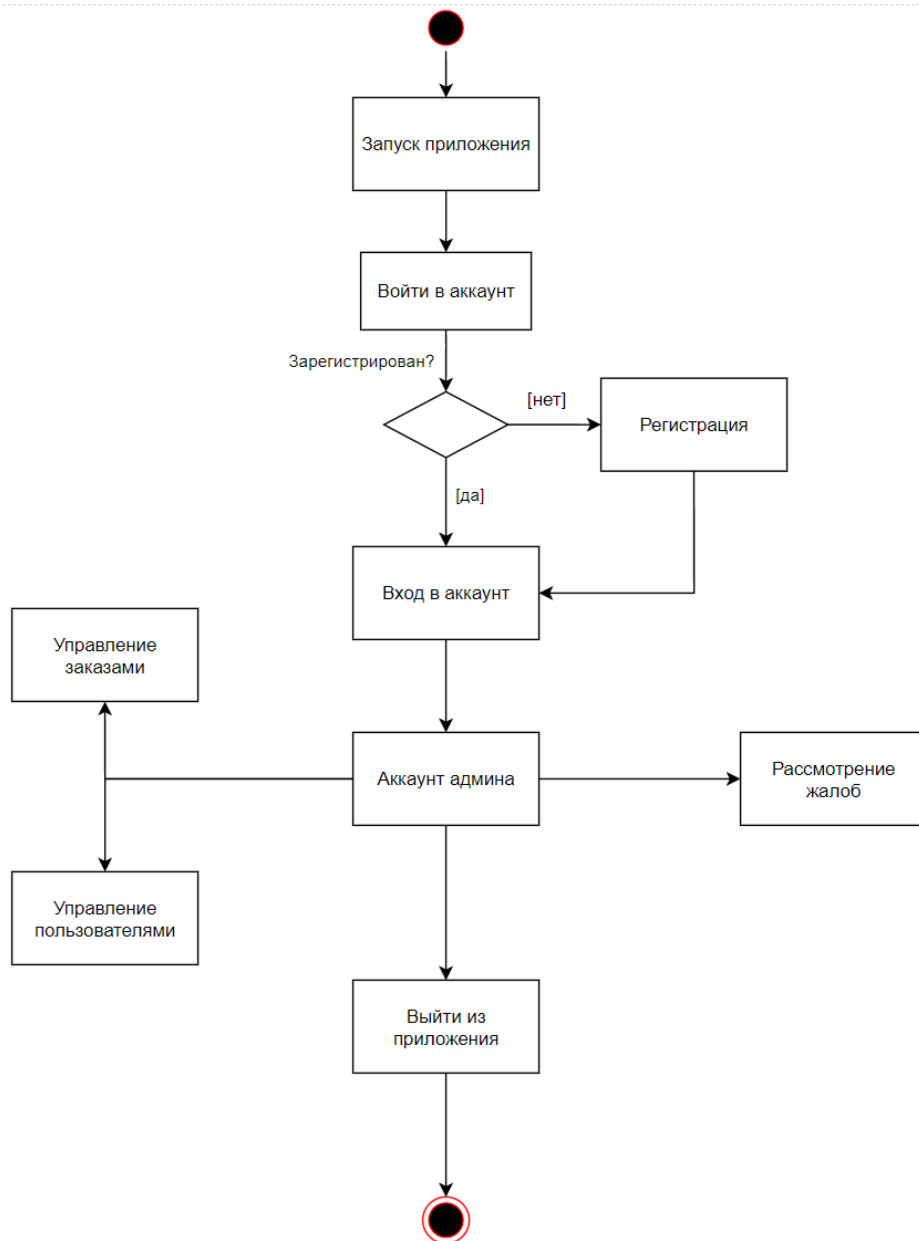


Рисунок 14 - Диаграмма активности администратора

Активность гостя

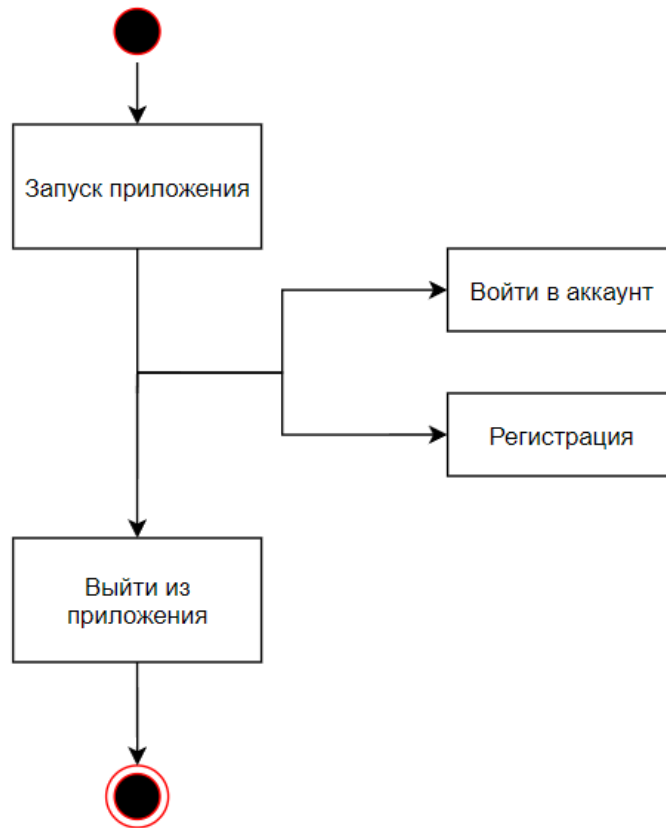


Рисунок 15 - Диаграмма активности гостя

Активность пользователя

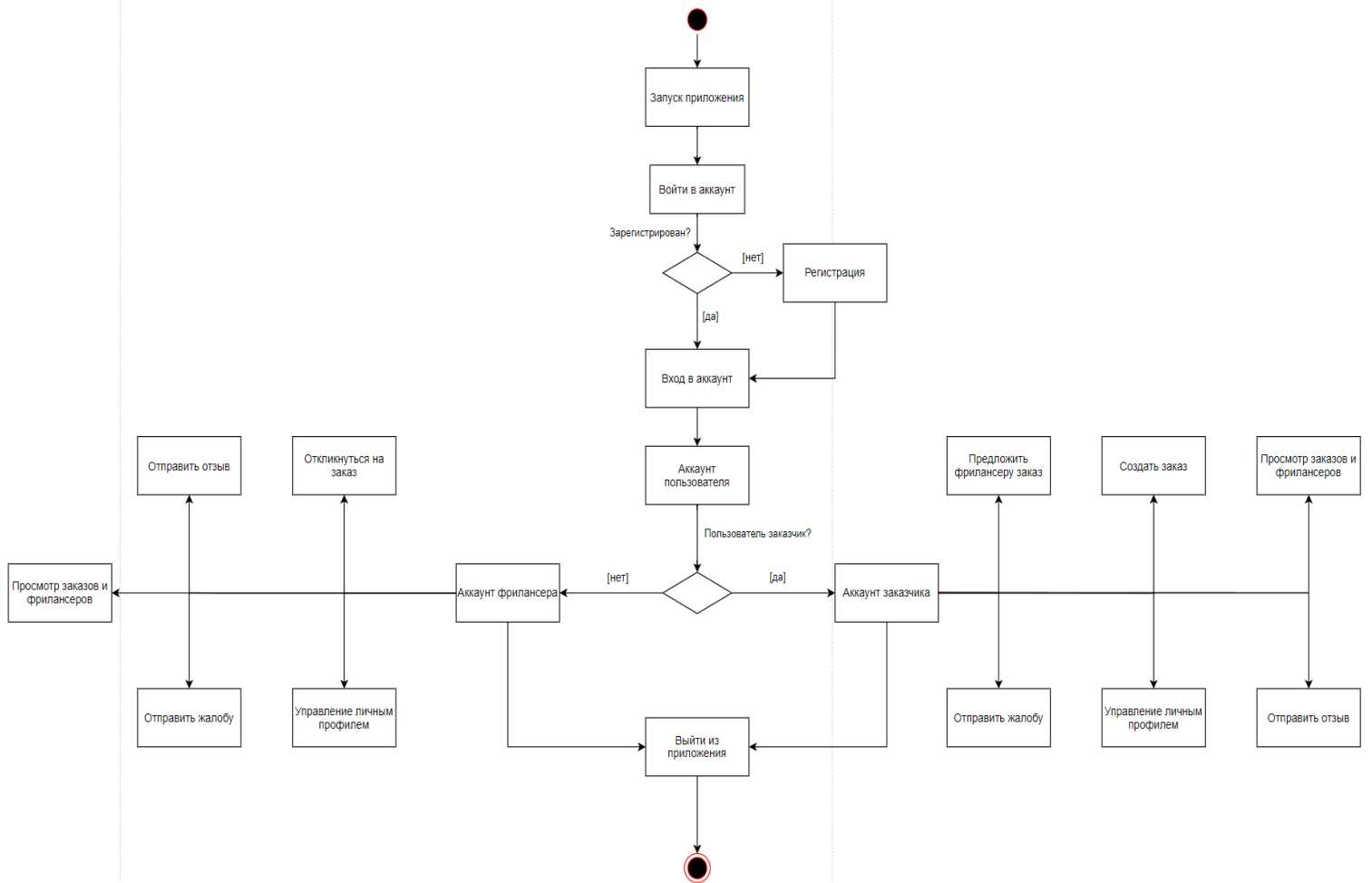


Рисунок 16 - Диаграмма активности пользователя