МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”**

Факультет *компьютерных наук*

Кафедра *программирования и информационных систем*

*Система обработки полей от вредителей квадрокоптером*

*Курсовой проект*

*09.03.02 Информационные системы и технологии*

Допущен к защите

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_П.Н. Парамонов*, 3 курс, д/о*

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.Ю. Сигарев *3 курс, д/о*

Воронеж 2020

# Введение

Несмотря на то, что Российское общество находится на стадии постиндустриального развития, значительную часть экономики занимает сельское хозяйство. Оно является основным поставщиком продуктов питания и сырья для многих отраслей промышленности. Нормальное функционирование сельского хозяйства обеспечивает большую часть физических потребностей человека. Развитие сельского хозяйства обеспечивало развитие общества, наглядно это можно увидеть во времена аграрных революций, например «Зеленой». В конце ХХ в. в сельском хозяйстве многих развивающихся стран происходила техническая и социальная революция, чему способствовали научнотехнические достижения и инновации, внесенные в аграрный сектор. Результаты сельскохозяйственного производства все в большей степени стали определяться уровнем технического оснащения отрасли. В последние десятилетия выросло количество тракторов и сельхозтехники, увеличилось применение удобрений и средств защиты растений, улучшилась агротехника земледелия и животноводства, постепенно модернизируется структура хозяйств.

Решение проблем сельского хозяйства может быть сконцентрировано в следующем направлении: инвестиции в отрасли, обеспечивающие рост производительности (создание инфраструктуры доступа на рынки, производство сельскохозяйственной техники и химикатов, развитие биотехнологий, развитие агротехники и науки и т. д.) В частности, имеет место быть применение беспилотных летательных средств для проведения рутинных сельскохозяйственных операций, таких например, как обработка от вредителей. В связи с этим возникает потребность в средствах информатизации компаний-посредников, занимающихся операциями данного рода.

# Постановка задачи

Цель курсовой работы: реализовать веб приложение, которое будет отвечать следующим требованиям:

* Понятный пользователю интерфейс
* Возможность делать заказ на обработку поля от вредителей
* Возможность просматривать историю заказов
* Возможность просматривать текущие заказы
* Возможность для менеджера просматривать текущие пестициды на склады, а так же удалять их
* Возможность для операторов дронов смотреть свои текущие заказы, а так же завершать их и обновлять прогресс.

Для достижения этих целей выделены задачи:

* Разработка Back-end часть приложения, которая отвечает за бизнес логику проекта и развернута на сервере приложения
* Разработка Базы Данных, для хранения больших объемов данных о пользователях, заказах, пестицидов и т.д
* Разработка Front-end части приложения, которая отвечает за предоставления интерфейса пользователю

# Анализ предметной области

* 1. Описание предметной области

Для решения некоторых сельхоз проблем в сфере обрабатывания полей от вредителей, такие как заправка техники топливом, содержание самой техники, ее обслуживание, неудобность довольно таки большой техники на малых земельных участках и т.д, было решено отказаться от этой всей техники в пользу беспилотных дронов. Ведь они не нуждаются в топливе, нужно лишь их зарядить, следовательно, не загрязняется окружающая среда, а также идет экономия на топливе. Ведь куда дешевле полностью зарядить дрон, нежели полностью заправить трактор. Так как дроны очень маленькие по сравнению с тракторами, то содержание гораздо легче их и обслуживание так же проще. Еще из-за своего размера, проще обрабатывать маленькие участки.

Для всех этих решений было придуман сервис “Agrin”. Который включается в себя, возможность сделать заказ на обработку полей дронами, а также просмотр этих заказов, для клиента. Просмотр списка пестицидов, возможность удаления и добавления пестицидов на склад для менеджера. А также просмотр списка доступных заказов, обновление и завершение заказа для оператора дронов.

* 1. Сравнение с аналогами

Среди аналогов на просторах интернета мы смогли выделить, при беглом просмотре первых страниц поисковика, только один web сервис:

Дронцентр (<https://dronecentre.ru/services/for-agriculture/>). На этом сервисе предлагают такие услуги как покупка дронов разных категорий, обучение пилотированию на дронах, фото и видеосъемка с дронов. Что касается сель-хоз промышленности, на этом сервисе предоставляется такая услуга как, фото-съемка земельного участка, и тем самым получение такой информации как:

● точный объем обрабатываемых и/или пригодных к обработке полей;  
● территория пастбищ, сенокосов и других угодий;  
● количество, размер, расположение пустых участков;

Из плюсов можно выделить:

* Возможность покупки и обучение пилотирования дрона. Тем самым можно в дальнейшем самому обрабатывать свой участок.
* Съемки различных категорий

Из минусов выделим и самый главный:

* Отсутствие обработки поля, а только лишь расчеты для этой обработки, что является одной из основных целей нашего проекта.
  1. Пользователи системы

Система предназначена для работы:

* Менеджера системы
* Оператора системы
* Клиента системы
  1. Анализ задач

Задачи, связанные для гостя:

* + 1. Задача регистрации пользователя

Задача регистрации пользователя разбивается на следующие этапы:

* Неавторизированный пользователь нажимает на кнопку “Регистрация”
* Появляется форма для регистрации
* Пользователь заполняет эту форму
* Данные отправляются на сервер
  + 1. Задача авторизации пользователя

Данная задача разбивается на следующие этапы:

* Неавторизированный пользователь нажимает на кнопку “Войти” 2. Появляется форма для ввода данных
* Пользователь вводит свои данные и нажимает на кнопку “Войти”.
* Данные отправляются на сервер система авторизует пользователя
  + 1. Задача создания заказа

Данная задача разбивается на следующие этапы:

* Пользователь нажимает в навигационном меню кнопку “Заказ”
* Появляется форма заказа
* Пользователь вводит требуемые данные
* Система делает расчеты и перенаправляет пользователя на страницу выбора даты заказа
* Пользователь выбирает дату
* Пользователь на след страницу выбирает оплату или же сразу оплачивает заказ

Задачи для авторизированного пользователя:

1. Задача просмотра заказов

Данная задача разбивается на следующие этапы:

* Пользователь нажимает на кнопку “Посмотреть историю своих заказов”
* Система выдает пользователю историю его заказов

1. Задача просмотра текущих заказов

Данная задача разбивается на следующие этапы:

* Пользователь нажимает на кнопку “Посмотреть текущие заказы”
* Система выдает пользователю текущие его заказ

Задачи для менеджера:

1. Задача просмотра пестицидов

Данная задача разбивается на следующие этапы:

* Менеджер нажимает на кнопку “Посмотреть пестициды”
* Система выдает список текущих пестицидов

1. Задача удаления пестицидов

Данная задача разбивается на следующие этапы:

* Менеджер нажимает на кнопку “Посмотреть пестициды”
* Система выдает список пестицидов
* Менеджер выбирает нужные пестициды и удаляет их

Задачи для оператора:

1. Задача просмотра текущих заказов оператора

Данная задача разбивается на следующие этапы:

* Оператор нажимает на кнопку “Текущие заказы”
* Система выдает список текущих заказов

1. Задача обновления прогресса текущего заказа

Данная задача разбивается на следующие этапы:

* Оператор нажимает на кнопку “Текущие заказы”
* Система выдает список текущих заказов
* Оператор выбирает нужный заказ
* Оператор обновляет прогресс заказа

1. Задача завершения текущего заказа

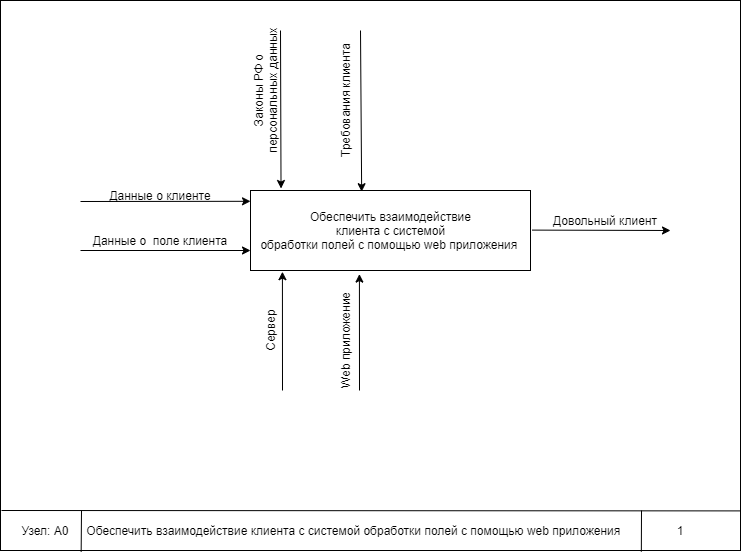
Данная задача разбивается на следующие этапы:

* Оператор нажимает на кнопку “Текущие заказы”
* Система выдает список текущих заказов
* Оператор выбирает нужный заказ
* Оператор завершает заказ
  1. Графическое описание работы системы

Для удобства описания работы системы была использована графическая нотация UML.

* + 1. Диаграмма idef0

Для описания работы системы в целом была разработана диаграмма IDEF0. Контекстная диаграмма представлена на Рисунке 1.



*Рис. 1 Диаграмма IDEF0*

По ней мы видим основную цель “Обеспечить взаимодействие клиента с системой обработки полей с помощью web приложения”.

Механизмы:

* Сервер
* Web приложение

Входные данные:

* Данные о клиенте
* Данные о поле клиента

Управляющие средства:

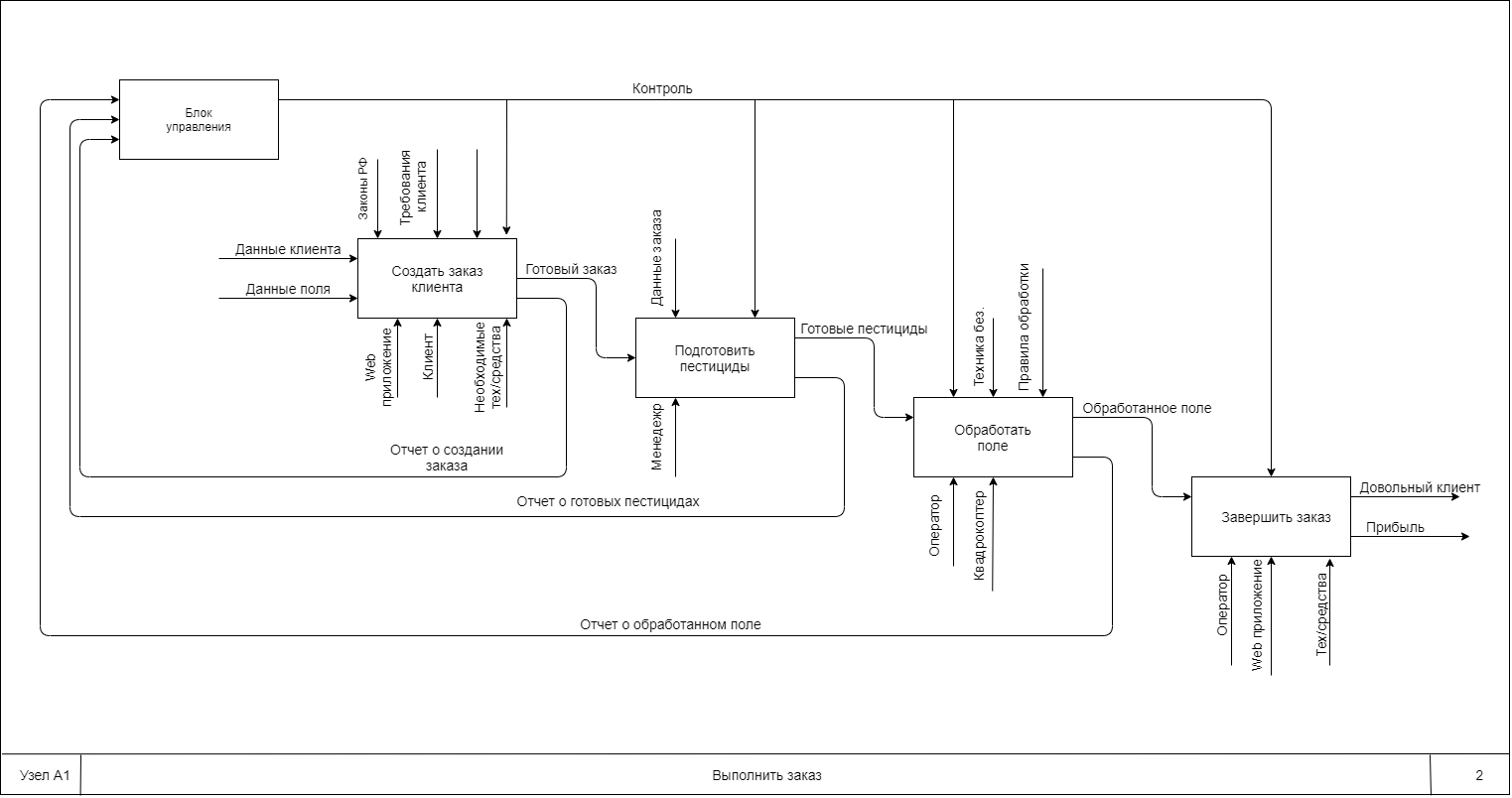
* Закон РФ о персональных данных
* Требования клиента

Выход:

* Довольный клиент

На втором уровне декомпозиции диаграммы у нас будут 5 функциональных блоков:

* Блок управления
* Создать заказ клиента
* Подготовить пестициды
* Обработать поле
* Завершить заказ



*Рис.2 IDEF0 на втором уровне декомпозиции*

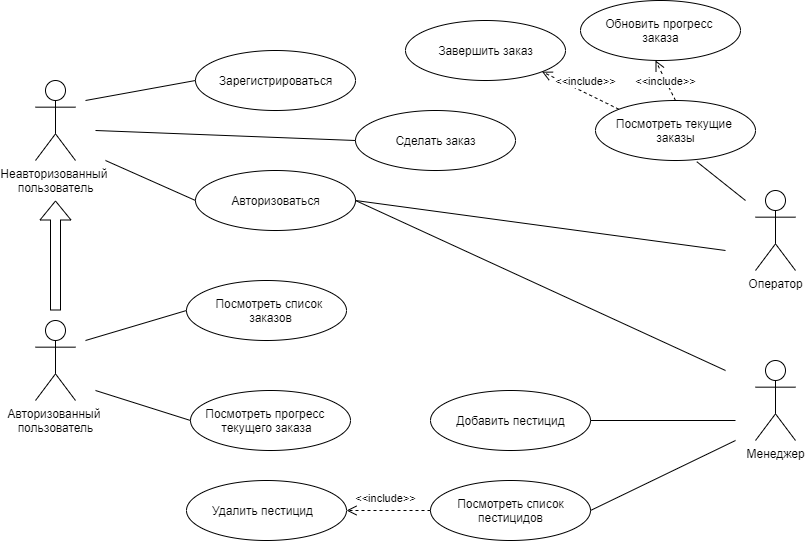
Блок управления выступает здесь за контроль над всеми блоками, который посылает каждому блоку свои руководства, а также получает отчет о выполнение каждого функционального блока.

* + 1. Диаграмма прецедентов

Для описания системы на концептуальном уровне была спроектирована диаграмма прецедентов, представленная на Рисунке 3.

Из основных участников можно выделить:

* Менеджер
* Оператор
* Авторизированный пользователь
* Неавторизированный пользователь



*Рис.3 UseCase*

У актера “Неавторизированный пользователь” следующие варианты использования:

* Зарегистрироваться
* Авторизоваться
* Сделать заказ

Актер “Авторизированный пользователь” наследует варианты использования от “неавторизированный пользователь” и у него так же есть и свои:

* Посмотреть список заказов
* Посмотреть список текущих заказов

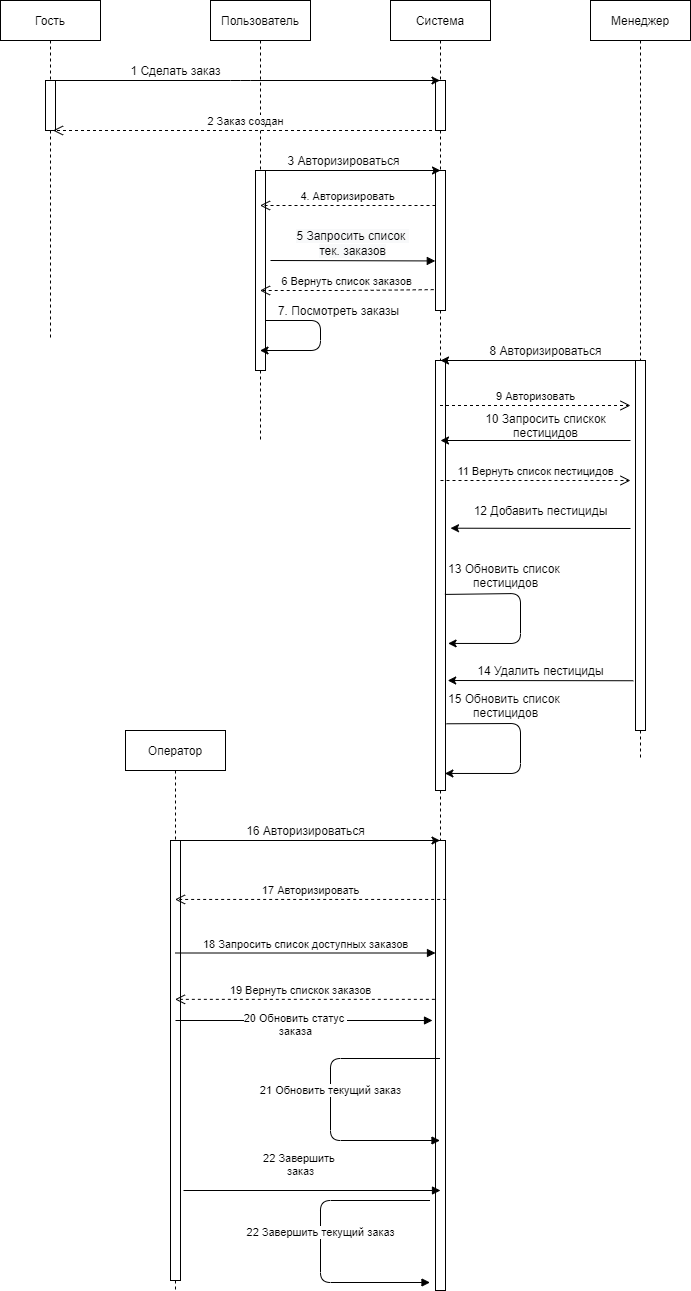
У актера “Менеджер” следующие варианты использования:

* Добавить пестицид
* Посмотреть список пестицидов
  + Удалить пестицид

У последнего актера “Оператор” следующие варианты использования:

* Посмотреть текущие заказы
  + Обновить прогресс заказа
  + Завершить заказ
    1. Диаграмма последовательностей

Для демонстрации взаимодействия пользователя, администратора и программного продукта была создана диаграмма последовательностей, представленная Рисунке 4.



*Рис 4. Диаграмма последовательности*

* + 1. Диаграмма коммуникаций

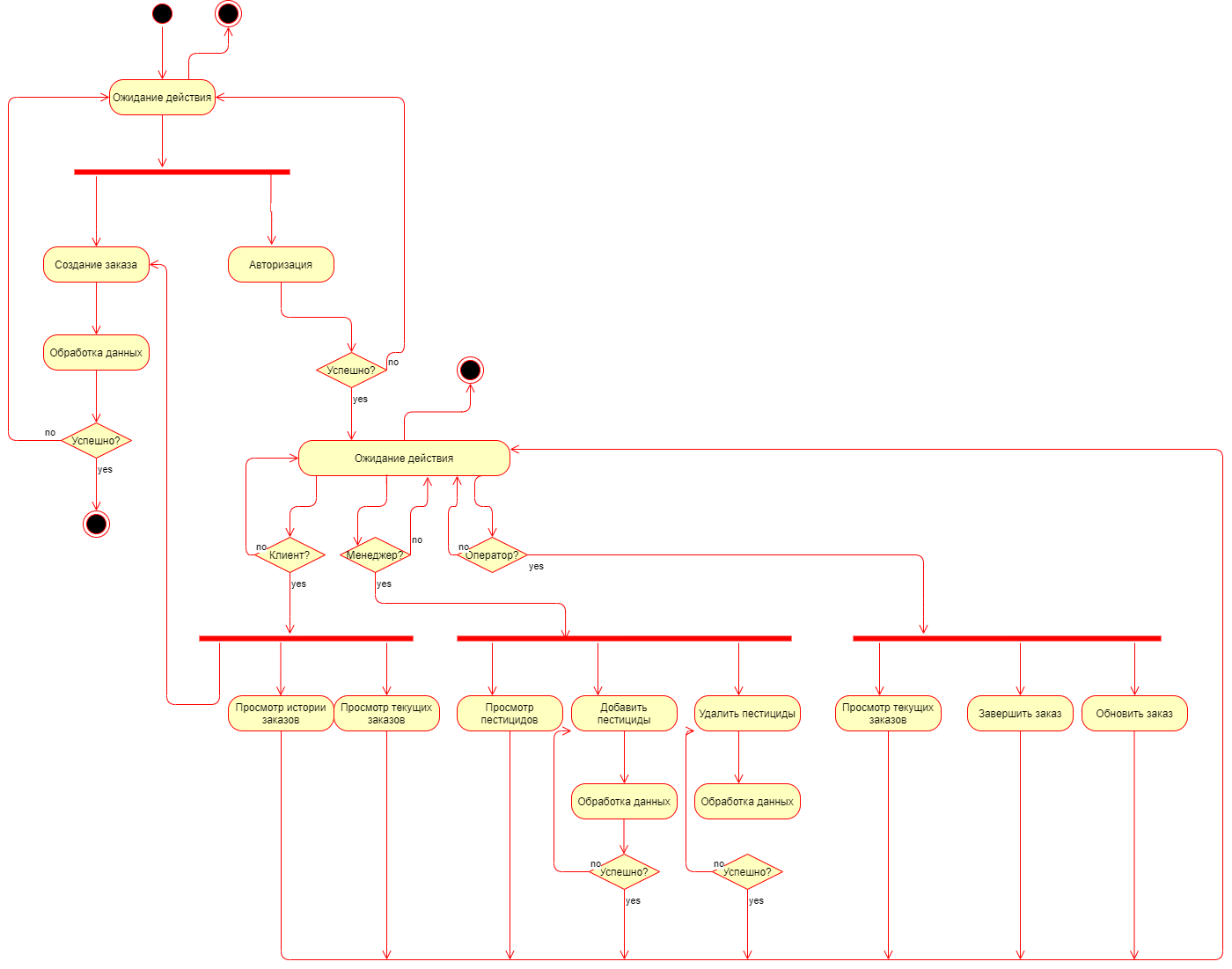
Для визуализации взаимодействия Пользователя, Менеджера, Оператора с системой, была спроектированная диаграмма коммуникаций на Рисунке 5.



*Рис.5 Диаграмма коммуникаций*

* + 1. Диаграмма состояний

Диаграмма состояний, изображенная на Рисунке 6, отражает возможные состояния системы. При запуске приложения система находится в ожидании выбора действия.



*Рис.6 Диаграмма состояний*

На данной диаграмме мы видим, что когда пользователь заходит на web приложение, приложение ждет от пользователя выбор:

* Авторизоваться
* Создать заказ

Если пользователь выберет создание заказа, приложение потребует ввести необходимые данные, далее пользователь отправляет свои данные. Далее идет обработка полученных данных от пользователя. Если они валидны, то происходит создание заказа. Если же введенные данные не валидны, то приложение сначала показывает информацию об этом, потом приложение опять ожидает действие от пользователя.

Второе состояние, которое может быть – это Авторизация пользователя или же Регистрация пользователя. Пользователь вводит необходимые данные, если они валидны, то система авторизует пользователя с доступной ей ролью: пользователь, менеджер, оператор.

Далее приложение опять ждет от пользователя действий. Если он клиент:

* Просмотр истории заказов
* Просмотр текущих заказов
* Создание заказа

При просмотре истории заказов, пользователю приложение выдает список всех проведенных заказов и опять ждет действие от пользователя.

При просмотре текущих заказов, пользователю приложение выдает список всех текущих на данный момент заказов и ждет дальнейшее действие.

При создании заказа, пользователь проводит те же самые действия, описанные при неавторизованном пользователе.

Если пользователь Менеджер:

* Просмотр пестицидов
* Добавление пестицидов
* Удаление пестицидов

При просмотре пестицидов, менеджеру приложение выдает список всех пестицидов на складе и ждет дальнейшее действие.

При добавлении пестицидов, менеджер вводит данные о добавляющихся пестицидах, если данные валидны приложение добавляет пестициды, иначе выводит сообщение о ошибке и ждет дальнейшее действие.

При удалении пестицидов, менеджер выбирает необходимые пестициды для удаления, если данные валидны приложение удаляет выбранные пестициды, иначе выводит сообщение о ошибке и ждет дальнейшее действие.

Если пользователь Оператор:

* Просмотр текущих заказов
* Обновление прогресса заказа
* Завершение заказа

При просмотре текущих заказов, оператору приложение выдает список всех доступных ему заказов и ждет дальнейшее действие.

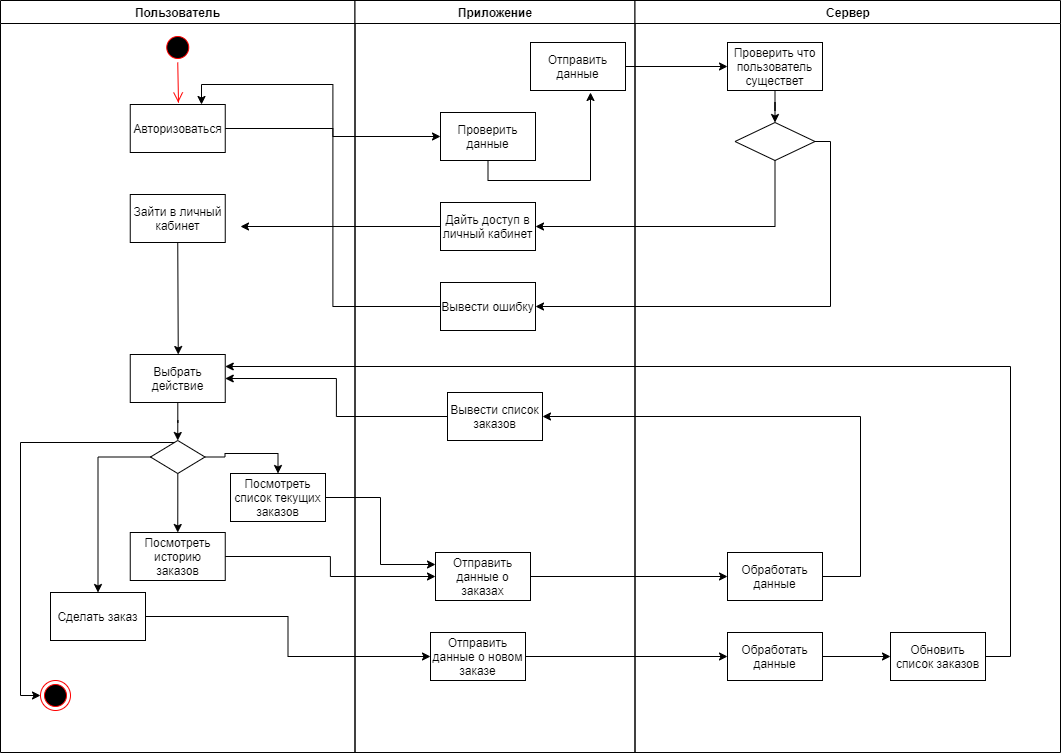
При обновлении прогресса, оператор вводит данные которые необходимо обновить, приложение обновляет прогресс заказа и ждет дальнейшее действие.

При завершении заказа, оператор завершает заказ приложение завершает заказ и ждет дальнейшее действие.

Если пользователю больше нечего делать в приложении, при ожидании действия, пользователь может выйти.

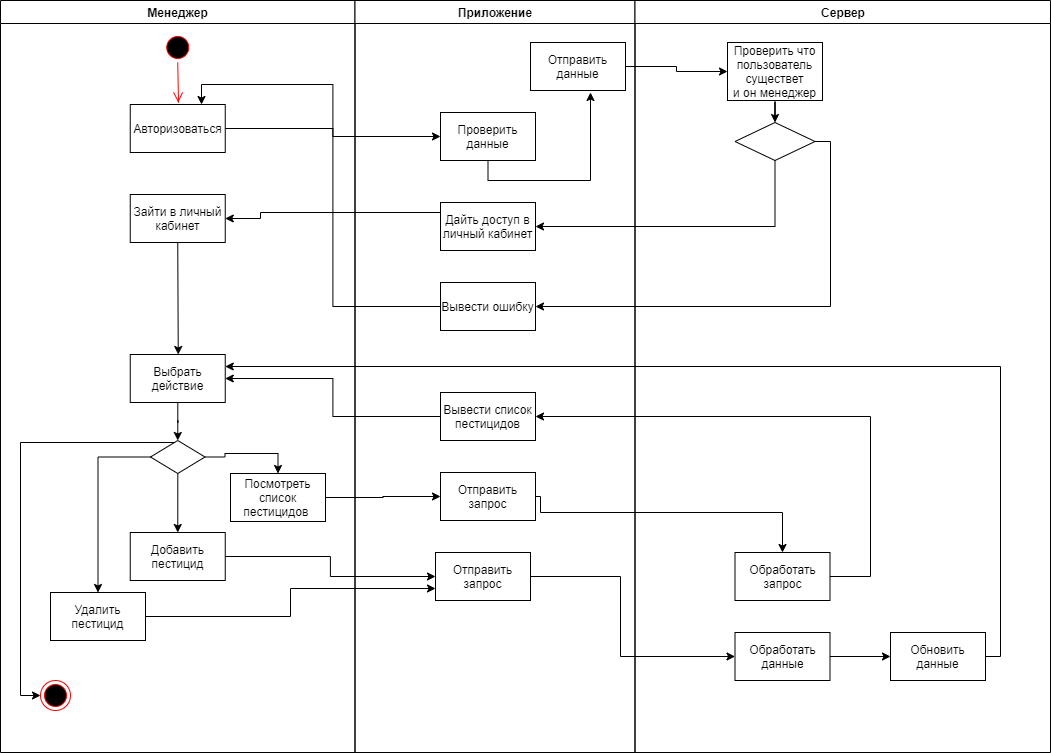
* + 1. Диаграмма активностей

С помощью диаграммы активности, представленной на Рисунке 7, была продемонстрирована работа клиента с приложением.



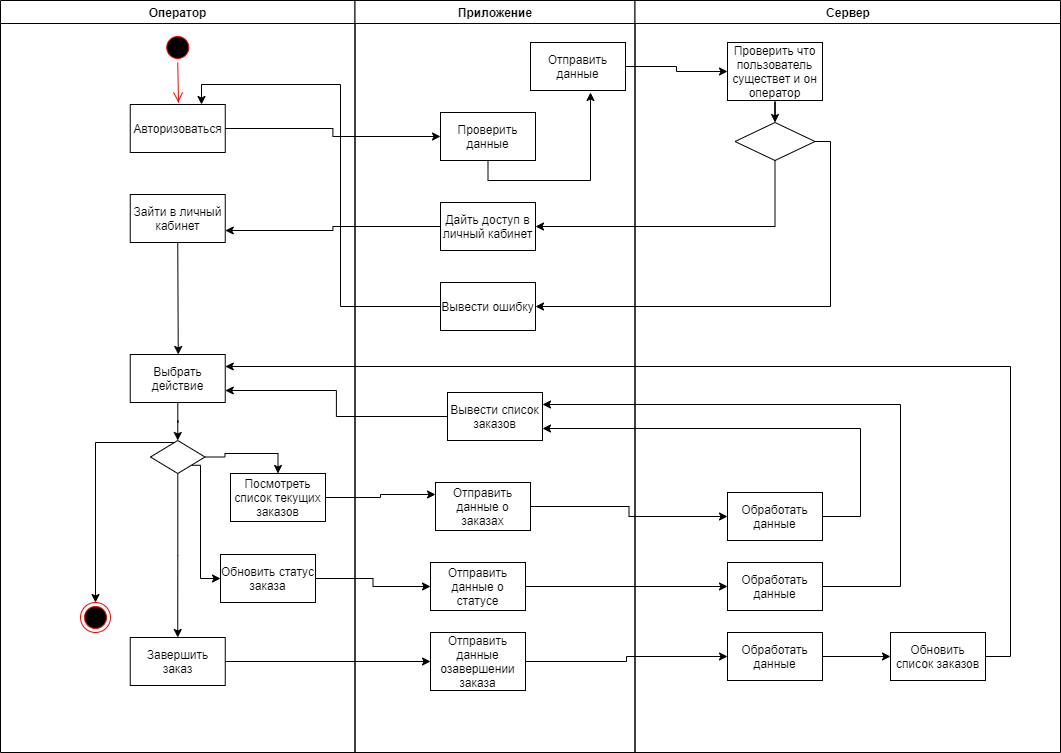
*Рис.7 Диаграмма активностей клиента*

С помощью диаграммы активности, представленной на Рисунке 8, была продемонстрирована работа менеджера с приложением.



*Рис. 8 Диаграмма активностей менеджера*

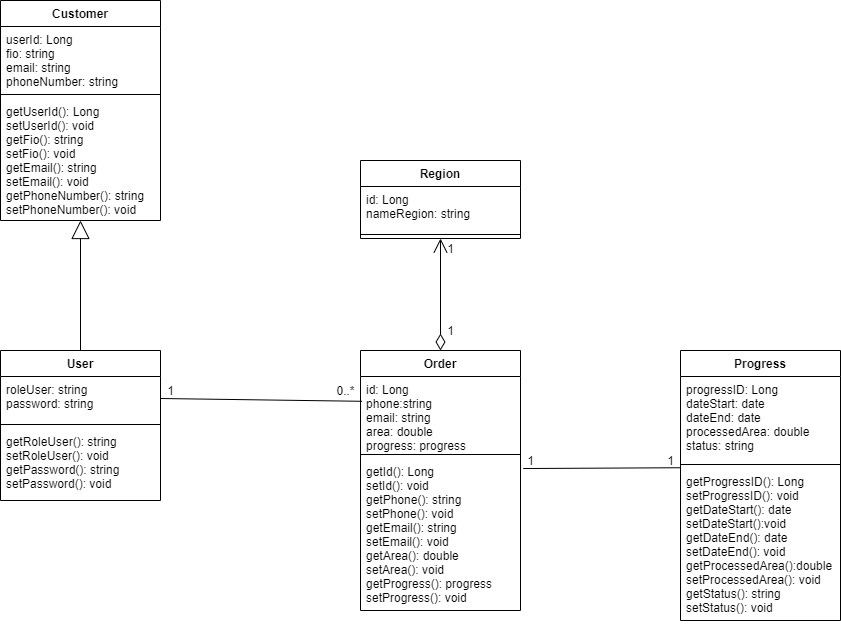
С помощью диаграммы активности, представленной на Рисунке 9, была продемонстрирована работа оператора с приложением.



*Рис. 9 Диаграмма активостей оператора*

* + 1. Диаграмма классов

В ходе разработки приложения была составлена данная диаграмма сущностей на Рисунке 10



*Рис.10 Диаграмма классов*

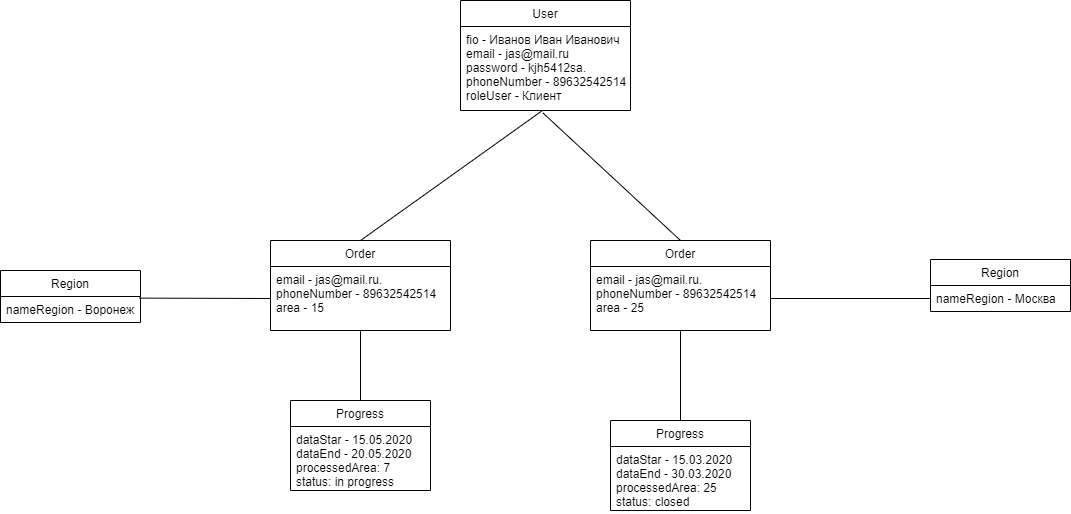
Это классы:

* Customer. С полями id, ФИО, почты, номера телефона.
* User. С полями от Customer, а так же его роль, и пароль
* Order. С полями id, телефон, почта, площадь, прогресс
* Region. С полями id, название региона.
* Progress. С полями id, дата начала, дата окончания, выполненная площадь, статус.

Так же у каждого класса, есть для всех полей get() и set().

* + 1. Диаграмма объектов

Для лучшего понимания диаграммы классов была составлена диаграмма объектов, представленная на Рисунке 11.



*Рис.11 Диаграмма объектов*

* + 1. ER-диаграмма

Для работы с базой данной была спроектирована ER-диаграмма, представленная на Рисунке 12.

На ней представлены 5 сущностей:

* User – пользователь

Атрибуты:

* User\_id – PK
* Role – роль пользователя
* Fio – ФИО пользователя
* Email –почта пользователя
* Pass- пароль пользователя
* Phone – номер телефона пользователя
* Order – заказ

Атрибуты:

* Order\_id – PK
* Progress\_id – FK на таблицу Progress
* Region\_id – FK на таблицу Region
* Phone\_number –телефон для заказа
* Email – почта для заказа
* Area – площадь
* Progress – прогресс заказа

Атрибуты:

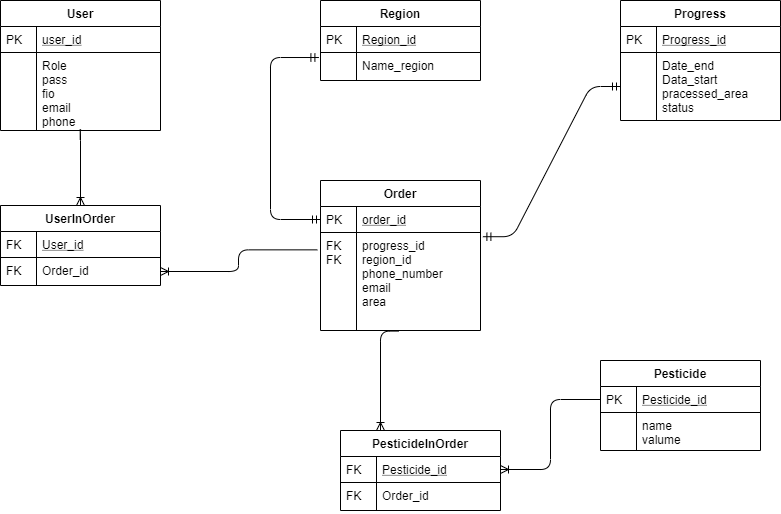
* Progress \_id – PK
* Data\_start –Дата начала заказа
* Data\_end –Дата окончания заказа
* Processed\_area – в для заказа
* Area – площадь
* Status – статус заказа
* Region – регион заказа

Атрибуты:

* Region \_id – PK
* Name –Название региона
* Pesticide – пестициды заказа

Атрибуты:

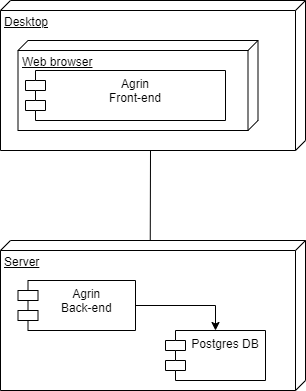
* Pesticide\_id – PK
* Name – название
* value – кол-во



*Рис. 12 ER-диаграмма*

* + 1. Диаграмма развертывания

На Рисунке 13 приведена диаграмма развертывания, визуализирующая элементы и компоненты разрабатываемой системы, которые существуют на этапе ее исполнения.



*Рис. 13 Диаграмма развертывания*