

**Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»**

Факультет программной инженерии и компьютерной техники

Курсовая работа, часть 1

По Информационным Системам

Выполнили:

Ступин Тимур Русланович

Васильев Никита Алексеевич

Группа № Р3308

Поток № 1.3

Преподаватель:

Николаев Владимир Вячеславович

Санкт-Петербург 2025

Содержание

Порядок выполнения работы.....	3
Согласовать с преподавателем предметную область, для которой будет разрабатываться информационная система.....	3
Составить подробное текстовое описание предметной области.....	3
Сформулировать, зачем нужна информационная система для представленной предметной области, какие задачи она позволит решить.....	5
Составить функциональные/нефункциональные требования к разрабатываемой информационной системе.....	6
Построить модели основных прецедентов (прецеденты согласуются с преподавателем), составить их описание.....	10
Предложить архитектуру будущей системы. Согласовать с преподавателем технологии и фреймворки, которые будут использоваться при реализации системы.....	16
Вывод.....	17

Порядок выполнения работы

Согласовать с преподавателем предметную область, для которой будет разрабатываться информационная система.

В ходе согласования была утверждена предметная область: “Предприятие по производству изделий на лазерном станке с ЧПУ”.

Составить подробное текстовое описание предметной области.

LaserWorks — это информационная система для автоматизации бизнес-процессов малого производства, специализирующегося на лазерной резке и обработке материалов (фанеры, ДСП, акрила и т.д.). Целью системы является полная цифровизация всех этапов работы: от получения заказа до его доставки клиенту.

Основные элементы системы:

- Управление заказами клиентов
- Проектирование и подготовка управляющих программ (УП)
- Планирование и контроль выполнения заказов на производстве
- Учет и управление запасами материалов

Ключевые аспекты предметной области:

1. Заказы клиентов:

- Каждый заказ создается менеджером отдела продаж на основе заявки созданной клиентом
- Заказ проходит несколько статусов: *Создан → В обработке → На согласовании → Готов к производству → В производстве → Готов к получению → Завершен.*
- К заказу прикрепляются файлы: эскизы, чертежи, 3D-модели, управляющие программы.

2. Участники системы:

- **Менеджер по продажам** — регистрирует заказы, взаимодействует с клиентами, согласовывает сроки.
- **Конструктор** — оценивает техническую реализуемость, загружает файлы проекта.

- **Оператор станка** — выполняет подготовленную УП на станке, отмечает начало и завершение работ.
- **Менеджер по продажам** — контролирует баланс материалов на складе. Формирует заявки на закупку материалов.
- **Сотрудник склада** — принимает материалы, оформляет отгрузку товаров
- **Клиент** — загружает предварительную информацию о заказе, может просматривать статус своего заказа, получать уведомления, загружать ТЗ и подтверждать получение товара.

3. Производственный процесс:

- После согласования заказа конструктор разрабатывает 3D-модель и УП.
- Далее разработанная модель утверждается клиентом или может быть отправлена на доработку конструктору. Когда согласован с обеих сторон он попадает на очередь в производство и распределяется свободному оператору
- Оператор выполняет работу, фиксируя прогресс в системе.
- Далее изделие попадает к сотруднику склада, который оформляет отгрузку заказа.

4. Управление запасами:

Система ведет учет материалов (фанера, краска, крепеж и др.) на складе.

При достижении минимального уровня запаса («точка заказа») система автоматически уведомляет ответственного сотрудника.

Менеджер по снабжению формирует заявку на закупку, которая может требовать утверждения руководства.

5. Финансовые операции:

- По каждому заказу формируется счет на оплату.
- Система отслеживает оплаты, автоматически обновляет статус заказа и предупреждает о просрочках.
- Финансовый менеджер может формировать отчеты по доходам, расходам и задолженностям.

6. Доставка и отгрузка:

- После завершения производства заказ переходит в статус «Готов к отгрузке».
- Логист выбирает способ доставки (самовывоз, курьер, ТК), оформляет документы.
- Клиент получает уведомление о передаче в доставку и может отслеживать статус.

7. Автоматизация и интеграция:

- Система обеспечивает сквозную автоматизацию всех ключевых процессов.
- Все участники видят актуальную информацию о заказах, что снижает количество ошибок и задержек.

Сформулировать, зачем нужна информационная система для представленной предметной области, какие задачи она позволит решить.

Информационная система **LaserWorks** необходима для повышения эффективности управления малым производством, связанном с лазерной резкой и деревообработкой. Она решает следующие ключевые задачи:

1. Централизация данных о заказах

Все заказы хранятся в единой базе, доступной всем участникам процесса. Это исключает потерю информации, дублирование и путаницу между отделами.

2. Автоматизация согласования и утверждения заказов

Автоматическая отправка 3D-моделей, смет и сроков клиенту через личный кабинет позволяет ускорить процесс согласования и избежать недопонимания.

3. Оптимизация производственного планирования

Система помогает равномерно распределять нагрузку на станки, минимизируя простои и перегрузки. Руководитель видит полную картину загрузки и может оперативно корректировать план.

4. Контроль за материалами и закупками

Автоматическое отслеживание остатков и напоминания о пополнении запасов позволяют избежать остановки производства из-за нехватки сырья.

5. Повышение удовлетворенности клиентов

Клиенты могут видеть статус своего заказа в реальном времени, получать уведомления и загружать документы. Это повышает доверие и лояльность.

6. Снижение человеческого фактора

Автоматизация рутинных операций (выставление счетов, формирование накладных, уведомления) снижает вероятность ошибок и освобождает время сотрудников для более важных задач.

Составить функциональные/нефункциональные требования к разрабатываемой информационной системе.

Функциональные требования

Для неавторизованных клиентов:

- **FR-1.1.1.** Система должна предоставлять возможность зарегистрироваться (с вводом имени, электронной почты, номера телефона и пароля).
- **FR-1.1.2.** Система должна предоставлять возможность авторизоваться через e-mail и пароль.
- **FR-1.1.3.** Система должна предоставлять возможность просматривать каталог изделий.
- **FR-1.1.4.** Система должна предоставлять возможность просматривать карточку изделия.

Для авторизованных клиентов:

- **FR-1.2.1.** Система должна предоставлять возможность формировать заявку на заказ (с заполнением информации о заказе).
- **FR-1.2.2.** Система должна предоставлять возможность просматривать список заказов.
- **FR-1.2.3.** Система должна предоставлять возможность редактировать заказ.
- **FR-1.2.4.** Система должна предоставлять возможность просматривать 3D-модель заказа.
- **FR-1.2.5.** Система должна предоставлять возможность утверждать заказ.
- **FR-1.2.6.** Система должна предоставлять возможность отклонять заказ с комментарием.
- **FR-1.2.7.** Система должна предоставлять возможность переходить в чат по заказу, включая просмотр истории и отправку сообщений.

Для работников:

- **FR-2.1.** Система должна предоставлять работнику возможность входа (логин/пароль).
- **FR-2.2.** Система должна предоставлять работнику возможность выхода из системы.
- **FR-2.3.** Система должна предоставлять работнику возможность изменять статус заказа.

Для менеджера по продажам:

- **FR-3.1.1.** Система должна предоставлять менеджеру по продажам возможность просматривать список заявок.
- **FR-3.1.2.** Система должна предоставлять менеджеру по продажам возможность формировать заказ на основе заявки.
- **FR-3.1.3.** Система должна предоставлять менеджеру по продажам возможность взаимодействовать с клиентом через чат по заказу (просмотр и отправка сообщений).

Для конструктора:

- **FR-3.2.1.** Система должна предоставлять конструктору возможность прикреплять 3D-модель изделия к заказу.
- **FR-3.2.2.** Система должна предоставлять конструктору возможность прикреплять управляющую программу (УП) к заказу.
- **FR-3.2.3.** Система должна предоставлять конструктору возможность просматривать комментарии клиента в случае отклонения заказа.
- **FR-3.2.4.** Система должна предоставлять конструктору возможность утверждать заказ.
- **FR-3.2.5.** Система должна предоставлять конструктору возможность отклонять заказ.

Для оператора станка с ЧПУ:

- **FR-3.3.1.** Система должна предоставлять оператору станка с ЧПУ возможность выгружать файл управляющей программы (УП).
- **FR-3.3.2.** Система должна предоставлять оператору станка с ЧПУ возможность выгружать файл 3D модели.
- **FR-3.3.3.** Система должна предоставлять оператору станка с ЧПУ возможность зафиксировать начало и конец производства.

Для сотрудника склада:

- **FR-3.4.1.** Система должна предоставлять сотруднику склада возможность принимать товар и фиксировать фактические значения.
- **FR-3.4.2.** Система должна предоставлять сотруднику склада возможность устанавливать статус заказа на “Готов к отправке”.

Для менеджера по снабжению:

- **FR-3.5.1.** Система должна предоставлять менеджеру по снабжению возможность просматривать остатки материалов на складе.
- **FR-3.5.2.** Система должна предоставлять менеджеру по снабжению возможность формировать заявки на закупку материалов.
- **FR-3.5.3.** Система должна уведомлять менеджера по снабжению о недостатке материалов.
- **FR-3.5.4.** Система должна предоставлять менеджеру по снабжению возможность добавлять ресурсы.
- **FR-3.5.5.** Система должна предоставлять менеджеру по снабжению возможность удалять ресурсы.

Для администратора:

- **FR-3.7.1.** Система должна предоставлять администратору возможность создавать аккаунты работников.
- **FR-3.7.2.** Система должна предоставлять администратору возможность удалять аккаунты работников.

Нефункциональные требования

Usability (Требования к удобству использования):

- **UR-001** Интерфейс должен быть, основан на знакомых пользователям паттернах взаимодействия, логично структурирован, с четкими инструкциями.
- **UR-002** Единый стиль ошибок и предупреждений (цвета, иконки).
- **UR-003** В системе должна быть реализована удобная навигация (доступная в основных разделах страницы, дающая пользователю однозначное представление о том, куда ведет ссылка), позволяющая пользователю перейти в нужный раздел.
- **UR-004** Система должна быть адаптирована под работу с устройств с разным расширением экрана. (Мобильные устройства: 320px–768px, планшеты: 769px–1024px, десктоп 1025px+).
- **UR-005** В системе должен быть реализован header, включающий в себя логотип, поисковую строку, корзину, ссылку на профиль, и footer с возможностью перейти на основные страницы сайта (контакты поддержки, ссылки на политики (конфиденциальность, возврат), социальные сети).

- **UR-006** Система должна предоставлять пользователю возможность зарегистрироваться и совершить покупку в течение 10 минут после первого посещения сайта.
- **UR-007** Система должна быть адаптирована к региональным форматам, соответствующем региону пользователя.
- **UR-008** Соответствие стандартам WCAG 2.1 для доступности.
- **UR-009** Система должна предоставлять веб-интерфейс, совместимый с последними версиями браузеров Chrome 133+, Firefox 135+, Safari и Edge 133+.
- **UR-010** Система должна обновляться в реальном времени результаты поиска при применении или изменении фильтров без перезагрузки страницы.

Reliability (Требования к надежности):

- **RR-001** Система должна быть доступна 99.9% времени в течение года (максимальное время простоя — 8 часов 45 минут в год).
- **RR-002** Плановое техническое обслуживание должно проводиться в период минимальной нагрузки и не чаще 1 раза в месяц.
- **RR-003** Среднее время между критическими сбоями (MTBF) должно составлять не менее 1000 часов. Для некритичных компонентов допустимо $MTBF \geq 500$ часов.
- **RR-004** Среднее время восстановления после сбоя (MTTR) не должно превышать 1 час для критических сбоев. Для некритичных компонентов — $MTTR \leq 4$ часа.
- **RR-005** Система должна предоставлять безопасную передачу данных пользователей по протоколу HTTPS
- **RR-006** Система должна контролировать доступ к защищенным ресурсам с использованием JWT токенов
- **RR-007** Данные пользователей должны быть зашифрованы с использованием алгоритма bcsuip и храниться на серверах с сертификатом безопасности SSL.

Performance (Требования к производительности):

- **PR-001** Максимальное время ответа для пользовательских операций — 2 секунды при скорости интернет-соединения 100 Мбит/с.
- **PR-002** Среднее время ответа по всем операциям — 500 мс при скорости интернет-соединения 100 Мбит/с.

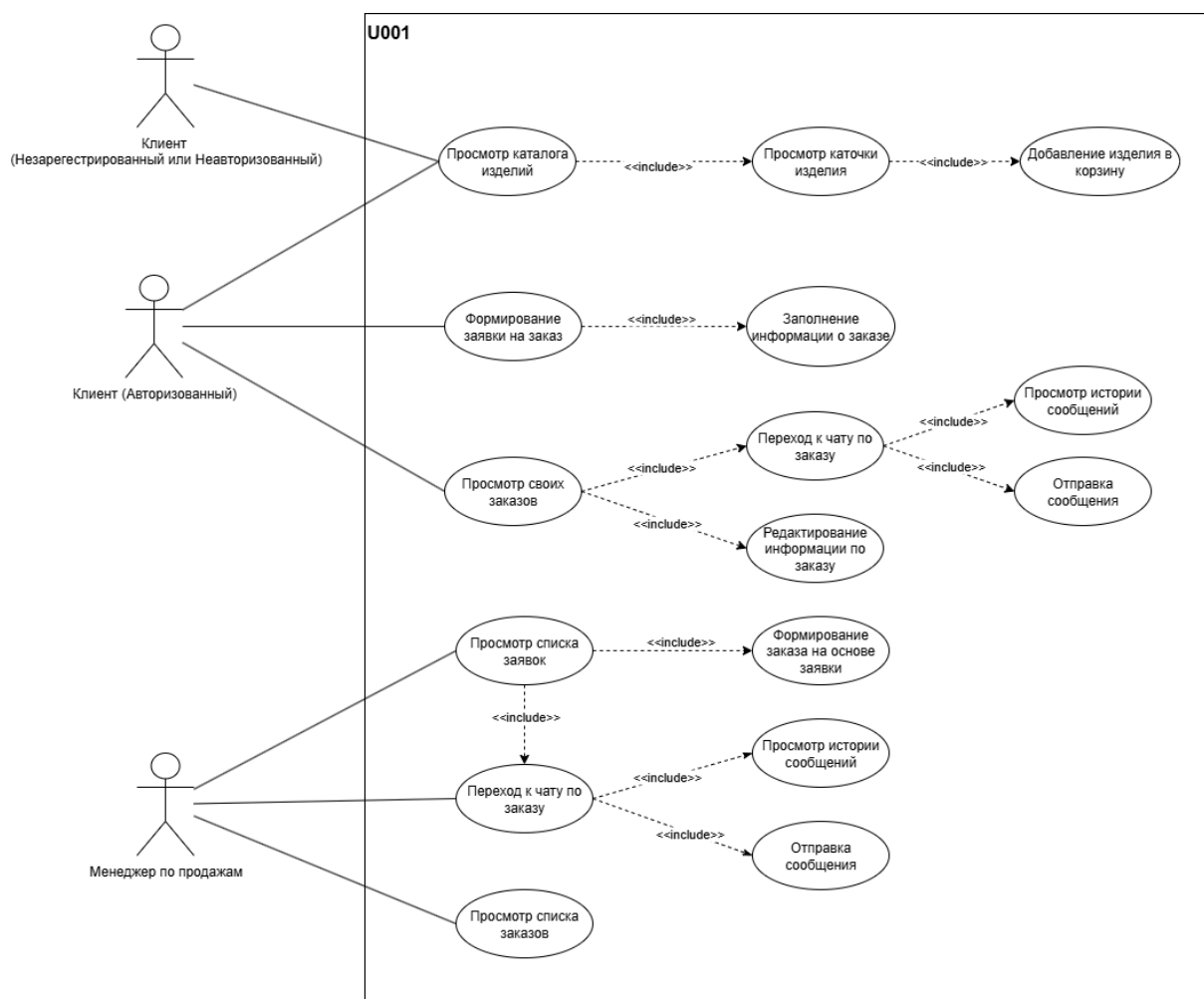
- **PR-003** Время загрузки страницы для десктопа — $\leq 1,5$ секунды, для мобильных устройств — ≤ 3 секунды (при 3G-соединении).
- **PR-004** Платформа должна поддерживать одновременное обслуживание не менее 100 активных пользователей
- **PR-005** Асинхронная обработка некритичных операций (например, отправка email-уведомлений).

Design Constraints (Ограничения разработки)

- **DR-001** Технологический стек: frontend должен быть разработан с помощью TypeScript с использованием фреймворка React 18+, Sass, Vite; backend должен быть реализован с помощью языка Java 17+ с фреймворком Spring Boot 3+, средство для управления базами данных – PostgreSQL и MinIO (для хранения файлов).
- **DR-002** Процесс разработки должен следовать методологии Scrum (Agile) с двухнедельными спринтами.
- **DR-003** Для управления версиями кода должен использоваться Git.

Построить модели основных прецедентов (прецеденты согласуются с преподавателем), составить их описание.

Прецедент: U001
ID: 1
Краткое описание: <i>Клиент</i> создает заявку, <i>Менеджер отдела продаж</i> регистрирует ее в системе.
Главный актер: <i>Клиент, Менеджер отдела продаж</i>
Второстепенные акторы: -
Предусловия: <i>Клиент авторизован и находится на сайте.</i>
Основной поток: <ol style="list-style-type: none"> 1. <i>Клиент</i> формирует заявку на изделие через каталог или вручную, прикладывая фото, эскизы и описание. 2. <i>Менеджер</i> получает уведомление о новой заявке. 3. <i>Менеджер</i> создает в системе новый заказ. 4. Система присваивает заказу уникальный ID и статус “Создан”.
Альтернативный поток: <ol style="list-style-type: none"> 1. <i>Менеджер</i> отправляет сообщение <i>Клиенту</i> с уточнениями по заказу. 2. <i>Клиент</i> получает сообщение от <i>Менеджера</i>. 3. <i>Клиент</i> отправляет ответ <i>Менеджеру</i>.

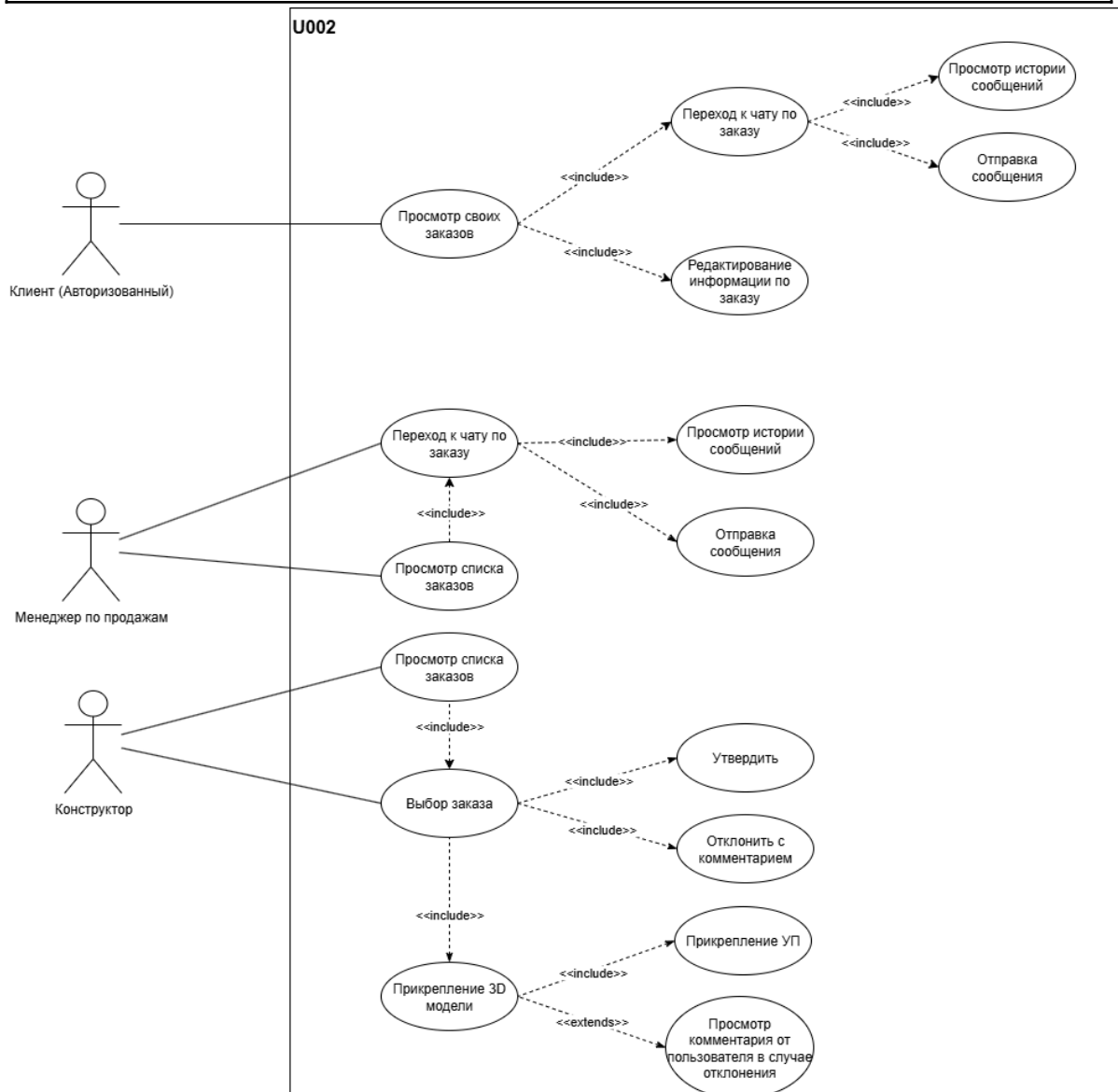


Прецедент: U002
ID: 2
Краткое описание: <i>Конструктор</i> проводит техническую экспертизу и расчет стоимости, после чего <i>Менеджер</i> согласует эти условия с <i>Клиентом</i> .
Главный актер: <i>Конструктор</i>
Второстепенные акторы: <i>Менеджер по продажам, Клиент</i>
Предусловия: Заказ имеет статус “Создан”.
Основной поток: <ol style="list-style-type: none"> 1. <i>Конструктор</i> видит в системе заказы со статусом “На согласовании”. 2. <i>Конструктор</i> изучает предоставленные материалы. 3. <i>Конструктор</i> создает 3D-модель и управляющую программу изделия и загружает её в Систему. 4. <i>Конструктор</i> рассчитывает нормы расхода материала и заносит эти данные в Систему 5. <i>Конструктор</i> рассчитывает стоимость изделия.

6. Система выдает заказу статус “Согласован конструктором”

Альтернативный поток:

1. *Конструктор* просматривает заявку и оставляет комментарии на доработку.
2. *Система* меняет статус заказа на «На доработке».
3. *Клиент* отвечает на комментарии конструктора по заказу.
4. *Конструктор* получает комментарий от пользователя и вносит изменения.



Прецедент: U003

ID: 3

Краткое описание: *Оператор ЧПУ* запускает управляющую программу, фиксирует начало и завершение работы, а также проблемы при производстве.

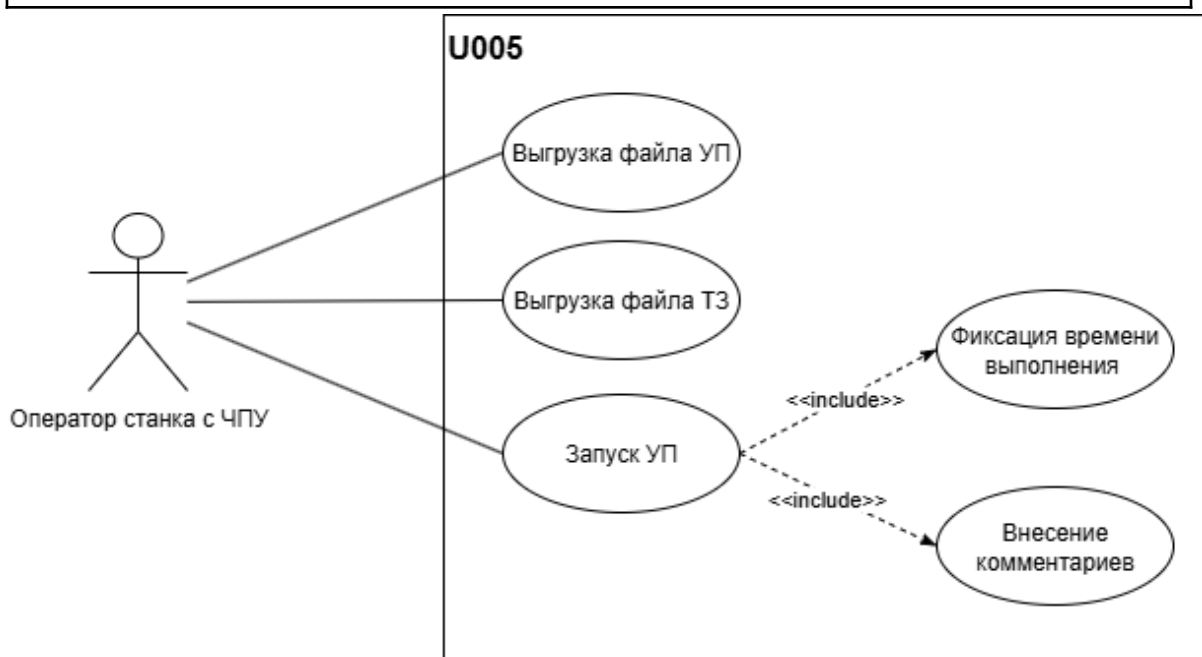
Главный актер: *Оператор станка с ЧПУ*

Второстепенные актеры: нет

Предусловия: Заказ находится в статусе “Готов к изготовлению”.

Основной поток:

1. Система выдает оператору заказ для производства
2. Система выдает заказу статус “В производстве”
3. Оператор запускает управляющую программу
4. Система отмечает время начала работы
5. Система отмечает время окончания работы
6. Система автоматически рассчитывает общее время выполнения заказа на станке.

Альтернативный поток:**Прецедент: U004****ID:** 4

Краткое описание: При выполнении клиентских заказов *Конструктор* фиксирует данные об используемых материалах, что позволяет отслеживать расход и остатки.

Главный актер: *Конструктор*

Второстепенные актеры: нет

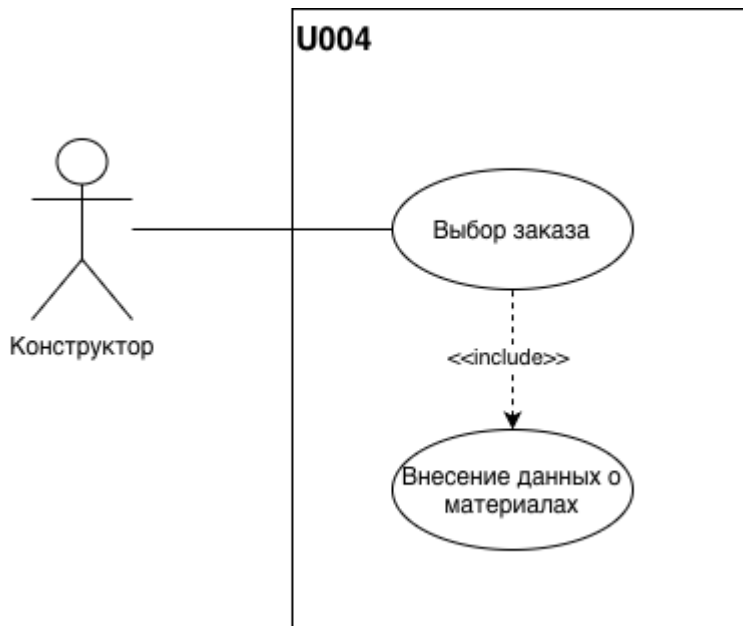
Предусловия: Имеется активный заказ клиента на изготовление изделия.

Основной поток:

1. *Конструктор* при технической оценке заказа вносит в систему данные о требуемом количестве и типе материалов.
2. Система автоматически списывает использованные материалы со склада после выполнения заказа.
3. Система обновляет текущие остатки на складе в режиме реального времени.

4. Система сохраняет все данные об использовании в истории расхода по каждому заказу.

Альтернативный поток:



Прецедент: U005

ID: 5

Краткое описание: Менеджер по снабжению создает заявку на закупку требуемого материала, указывая параметры поставки.

Главный актер: Менеджер по снабжению

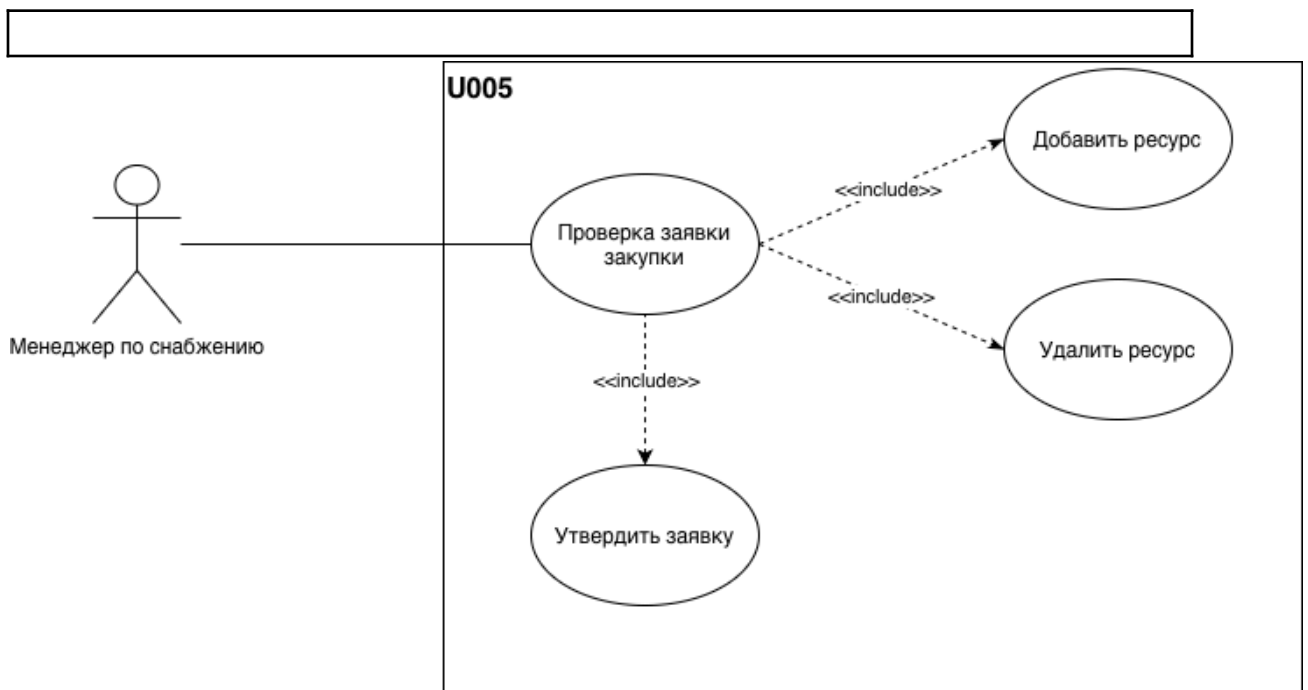
Второстепенные акторы: нет

Предусловия: Система уведомила о необходимости закупки, либо инициатива исходит от менеджера.

Основной поток:

1. Менеджер по снабжению переходит в раздел заявок и создает новую заявку на закупку.
2. В заявке указываются:
 - наименование материала;
 - требуемое количество;
 - дополнительная информация

Альтернативный поток:



Прецедент: U006
ID: 6
Краткое описание: После получения материалов на склад, <i>Сотрудник склада</i> регистрирует факт приемки и система обновляет запасы.
Главный актер: <i>Сотрудник склада</i>
Второстепенные акторы: Нет
Предусловия: Материалы фактически поступили на склад.
Основной поток: <ol style="list-style-type: none"> <i>Сотрудник склада</i> проверяет соответствие полученных материалов по накладной. <i>Сотрудник склада</i> подтверждает приемку в системе. Система автоматически обновляет складские остатки. Система присваивает заказу статус “Завершено”.
Альтернативный поток: <ol style="list-style-type: none"> <i>Сотрудник склада</i> обнаруживает расхождение в поставке или брак. <i>Сотрудник склада</i> вносит фактическое значение полученного материала. Система сохраняет баланс материалов исходя из фактически принятого количества



Предложить архитектуру будущей системы. Согласовать с преподавателем технологии и фреймворки, которые будут использоваться при реализации системы.

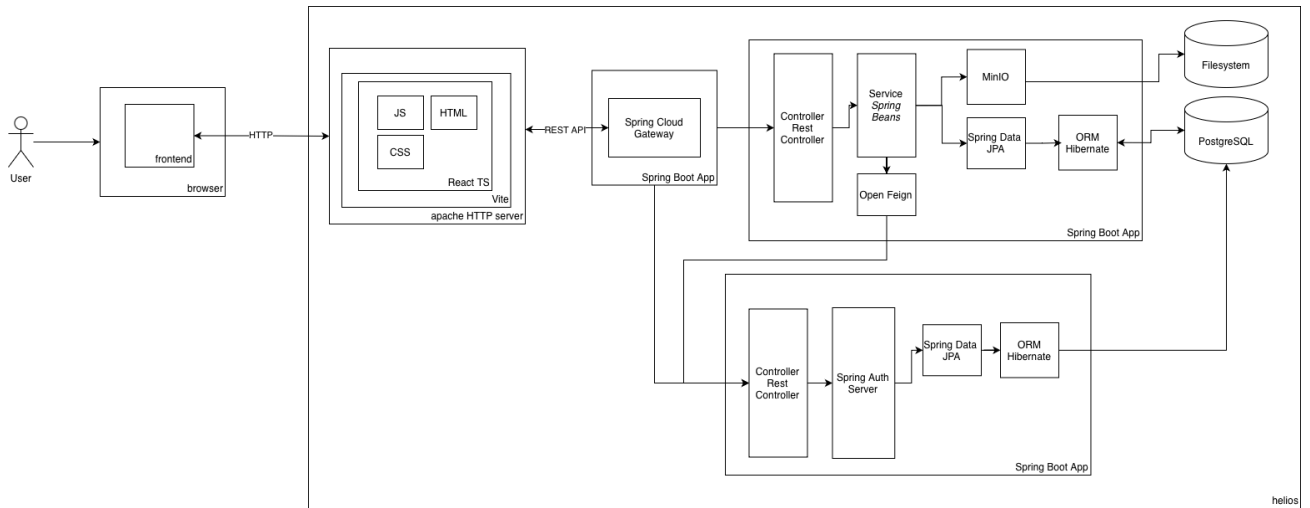
Система будет реализована по принципу **клиент-серверной архитектуры** с разделением на два уровня:

- Frontend (клиентская часть);
- Backend (серверная часть).

Общение между фронтендом и бэкендом – через REST API. Сервер работает с базой данных через ORM и репозитории.

Технологии и компоненты:

Frontend	Backend
React + TypeScript	Spring Boot
React Router	Spring Web
Axios	Spring Security
Formik + Yup	Spring Data JPA
React Query	Hibernate
Tailwind CSS	PostgreSQL
Material UI	Flyway
Vite	JWT
i18next	Lombok



Вывод

В результате выполнения первого этапа курсовой работы была детально описана предметная область — система управления производством в цехе лазерной резки. Определены ключевые бизнес-процессы, сформулированы цели системы и ее основные функции. Разработаны функциональные и нефункциональные требования, построены модели прецедентов и предложена клиент-серверная архитектура с использованием современных технологий. На данном этапе заложена прочная основа для дальнейшего проектирования, разработки и тестирования информационной системы **LaserWorks**.