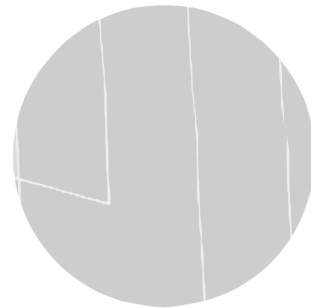




МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
“Национальный исследовательский университет ИТМО”

**ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ
И КОМПЬЮТЕРНОЙ ТЕХНИКИ**



ЛАБОРАТОРНАЯ РАБОТА №1

по дисциплине
“Архитектура Программных Систем”

Работу выполнил:
Студент группы Р3311
Болорболд Аригуун
Преподаватель:
Перл Иван Андреевич

г. Санкт-Петербург
2025 г.

Содержимое

1	Текст задания	3
2	Описание рассматриваемой системы с требованиями к ней	3
2.1	Функциональные требования	3
2.2	Нефункциональные требования	4
3	Оценка трудозатрат, стабильности, приоритетов	5
4	Use-Case диаграмма	6
4.1	Пример прецедента	6
5	Вывод	6



1 Текст задания

Выбрать любую реально существующую систему и описать её в терминах UML. Желательно, чтобы система была не полностью информационной, но опиралась на информационную систему как показано в примере на лекции (Point of sale). Необходимо описать границы системы на разных уровнях, а также описать сценарии использования для нескольких Акторов.

Отчет по работе должен содержать:

1. Титульный лист с указанием автора и номера группы
2. Само задание
3. Описание рассматриваемой системы с требованиями к ней
4. Формальное описание системы с необходимым количеством UML диаграмм
5. Словесное описание сценариев использования для рассматриваемых акторов

2 Описание рассматриваемой системы с требованиями к ней

Рассматриваемая система — касса самообслуживания у магазинов розничной торговой сети «Лента». Для увеличения производительности обслуживания покупателей приходится увеличивать количество касс наряду с количеством кассиров. Данная система является удобным благодаря тому, что при его установке не приходится нанять живую рабочую силу. Выбрал эту систему, потому что это является самым часто используемым мной Point-of-sale.

Пользователь может просканировать продукты магазина, применить скидки и провести оплату за покупку самостоятельно. Администратор (в этом случае наблюдающий сотрудник) может также удалить и аннулировать товары, и валидировать пользователя.

2.1 Функциональные требования

Требования пользователя

- FR0 Система должна предоставлять возможность пользователю сканировать продукт по его баркоду, в частных случаях (молочные, табачные, алкогольные и другие виды продуктов) по специальному QR-коду, а также вручную.
- FR1 Система должна предоставлять возможность пользователю изменить количество продуктов одного и того же типа.
- FR2 Система должна предоставлять возможность пользователю удалить продукт со списка.
- FR3 Система должна отразить пользовательские изменения в списке продукта в реальном времени с соответствующими характеристиками (например, название, цена, количество продукта).
- FR4 Система должна предоставлять возможность пользователю применить скидки.
- FR5 Система должна предоставлять возможность пользователю сканировать купоны для активации вышеупомянутой скидки.
- FR6 Система должна предоставлять возможность пользователю выбрать наличие или отсутствие бумажного чека.
- FR7 Система должна оповещать пользователя о необходимости вызова администратора в нужных случаях (например, при валидации возраста или при удалении продукта).
- FR8 Система должна предоставлять возможность пользователю провести платеж по карте, СБП, купонами и NFC-чипом.
- FR9 Система должна предоставлять возможность пользователю провести платеж по наличным средствам (купюра и монеты).

Требования зарегистрированного пользователя

- FR10 Система должна предоставлять возможность пользователю использовать карту покупателя «Карта №1» (по баркоду (в физическом и цифровом виде) и по номеру телефона) для применения скидок, кэшбеков и спецпредложений.
- FR11 Система должна предоставлять возможность пользователю накопить и использовать баллы и фишки.
- FR12 Система должна отправить электронный чек покупки в электронную почту покупателя при его наличии.

Требования администратора

- FR13 Система должна предоставлять возможность администратору подтвердить действия покупателя (например, удаление продукта).
- FR14 Система должна предоставлять возможность администратору валидировать необходимый по закону статус покупателя для покупки продукта (например, продажа алкоголя не разрешается несовершеннолетним лицам).
- FR15 Система должна предоставлять возможность администратору аннулировать список покупки.
- FR16 Система должна предоставлять возможность администратору отменить денежную транзакцию.

Требования к безопасности:

- SEC0 Система должна использовать защищенный протокол соединения.
- SEC1 Система должна иметь систему ролей, разделение функциональности по ролям и по крайней мере следующие роли: покупатель (незарегистрированный и зарегистрированный) и администратор.
- SEC2 Система должна предоставлять возможность администратору идентифицироваться с помощью специального пропуска с баркодом.
- SEC3 Система должна хранить всю собранную информацию о пользователях на зашифрованном хранилище данных.
- SEC4 Система должна хранить конфиденциальную информацию о пользователях (ФИО, номер телефона и электронная почта) на отдельном защищенном хранилище данных.

2.2 Нефункциональные требования

Требования к удобству использования:

- U0 Система должна быть спроектирована таким образом, чтобы она была проста для использования среднестатистическому покупателю и не требовала от него дополнительной подготовки.
- U1 Система должна формировать полный ответ на действия пользователей в диапазоне от 1 до 3 секунд.
- U2 Система должна быть выполнена в корпоративных цветах МКПАО «Лента».
- U3 Система должна иметь поиск для продуктов.
- U4 Система должна предлагать пользователям мобильной версии сайта скачать мобильное приложение.

Требования к надежности:

- R0 Система должна обеспечивать максимальное время простоя на уровне 1 часа в год с учетом планового технического обслуживания.
- R1 Система должна заранее оповещать о вероятном отказе при наличии сопутствующих факторов (сбоев).

R2 Система должна обеспечивать время полного восстановления после отказа на уровне 5 минут с момента развертывания.

R3 Система должна обеспечивать среднее время между отказами не менее 1 месяца.

R4 Система должна отправлять чек покупки на электронную почту пользователя в промежутке 5 минут после осуществления денежного перевода.

Требования к производительности:

P0 Система должна обеспечивать доступ к хранилищу данных не дольше, чем за 2 секунды, хранилищу изображений – не дольше, чем за 10 секунд.

P1 Система должна быть способным обслуживать не менее 100 пользователей в час.

P2 Система должна обеспечивать время отклика на запрос в пределах 1 секунды.

P3 Система должна быть доступна 24 часа в сутки.

P4 Система должна обеспечивать реактивность в пределах 1 секунды.

P5 Система должна обеспечивать время восстановления не больше двух времен отклика.

Требования к поддерживаемости:

S0 Система должна быть совместима с UNIX-серверной вычислительной мощностью.

S1 Система должна поддерживать локализацию на 30 популярных языков кроме русского.

S2 Система должна поддерживать интернационализацию для 10 географических зон со всей территории Российской Федерации и ближнего зарубежья.

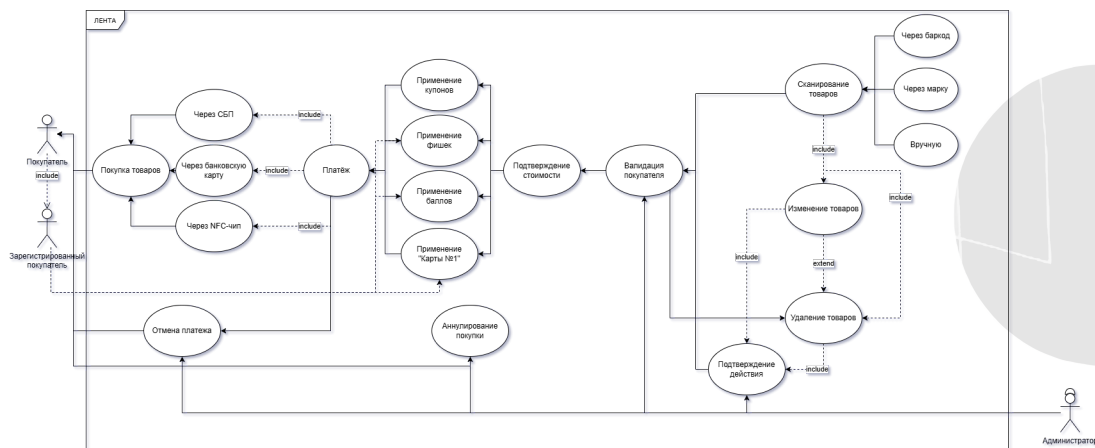
S3 Система должна поддерживать изменения сторонних платформ.

S4 Все функциональные возможности системы должны быть задокументированы в соответствие со стандартом IEEE Std 1016-1998.

3 Оценка трудозатрат, стабильности, приоритетов

№	Статус	Приоритетность	Трудоемкость, чел-ч	Стабильность
FR0	Включено	MUST	40	Высокая
FR1	Включено	MUST	2	Высокая
FR2	Включено	MUST	2	Высокая
FR3	Включено	MUST	4	Средняя
FR4	Одобрено	Should	8	Средняя
FR5	Предложено	Could	25	Средняя
FR6	Предложено	Could	1	Высокая
FR7	Одобрено	MUST	2	Высокая
FR8	Включено	MUST	30	Высокая
FR9	Отклонено	Won't	50	Высокая
FR10	Одобрено	Should	10	Высокая
FR11	Предложено	Could	5	Средняя
FR12	Одобрено	Should	4	Высокая
FR13	Включено	MUST	10	Высокая
FR14	Одобрено	Should	15	Высокая
FR15	Предложено	MUST	10	Высокая
FR16	Одобрено	MUST	25	Высокая

4 Use-Case диаграмма



4.1 Пример прецедента

Прецедент: Добавление «Карты №1»
ID: 1
Краткое описание: Покупатель хочет добавить свою «Карту №1»
Главный актер: Покупатель (пользователь)
Второстепенный актер: нет
Предусловия: Покупатель имеет «Карту №1»
Основной поток: <ul style="list-style-type: none"> 1. Покупатель сканирует продукты 2. Покупатель переходит на окно добавления карты 3. Покупатель добавляет карту за счёт сканирования «Карты №1» в виде баркода
Альтернативный поток: <ul style="list-style-type: none"> 1. Покупатель сканирует продукты 2. Покупатель переходит на окно добавления карты 3. Покупатель вводит номер телефона, связанный с его «Картой №1»

5 Вывод

При выполнении лабораторной работы я изучил как составлять функциональные и нефункциональные требования, как определять их атрибуты. Также я научился строить Use-Case диаграммы на языке UML для описания прецедентов.