Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный технологический университет»

Кафедра информационных систем и технологий

**Отчет по лабораторной работе на тему**:

**«Формирование требований к информационной системе»**

Выполнила:

студентка 4 курса 7 группы ФИТ

Леонов Д.И.

Цель:

Анализ предметной области и формирование требований к информационной системе, подготовка технического задания на разработку программного продукта.

Минск 2023

# 1. Теоретические вопросы

## 1.1 Понятия информационной системы и информационных потоков.

Информационная система – совокупность средств, методов и персонала, используемого для хранения, обработки и передачи информации в интересах достижения поставленной цели

Информационный поток – совокупность циркулируемых сообщений с целью управления, анализа и контроля операций.

## 1.2 Стадии формирования представления о системе.

Всего есть 13 стадий формирования представлений о системе:

1) 1-5: общее цельное «изучение системы»;

2) 6-9: «формирование углубленных представлений о системе»;

3) «Моделирование системы»;

4) 11-13: «сопровождение системы».

## 1.3 Жизненный цикл программного продукта (ЖЦ ПП).

Жизненный цикл программного продукта – это методология проектирования информационных систем, которые описывают процесс работы над продуктом и его сопровождение, представленное в нескольких этапах:

1) Планирование/формирование требований;

2) Проектирование;

3) Реализация;

4) Тестирование;

5) Ввод в действие;

6) Эксплуатация и сопровождение.

## 1.4 Техническое задание (ТЗ).

Техническое задание – это документ, содержащий в себе требования на разработку информационной системы.

## 1.5 Нормативная база в области документирования ПП. Стандарты документирования.

Нормативная база в области документирования ПП:

1) Архитектурная/проектная — обзор программного обеспечения, включающий описание рабочей среды и принципов, которые должны быть использованы при создании ПО;

2) Техническая — документация на код, алгоритмы, интерфейсы, API;

3) Пользовательская — руководства для конечных пользователей, администраторов системы и другого персонала;

4) Маркетинговая.

Стандарты документирования:

1) ГОСТ 34.601-90;

2) ISO/IEC 12207:1995;

3) Custom Development Method (CDM – методика Oracle);

4) Rational Unified Process (RUP);

5) Microsoft Solution Framework (MSF);

6) Extreme Programming (XP).

# 2. Техническое задание к проекту «Интерфейс» бронирования машины

## 2.1 Целевая аудитория

Основная целевая аудитория проекта – юноши и девушки возраста от 18 лет. Так как многие получают водительское удостоверение рано, но не всегда есть средства на покупку автомобиля. Также часть людей постарше, которым срочно необходим автомобиль, если у них возникли неполадки с их личным авто. Резюмируя, можно сказать, что целевая аудитория довольна обширна и имеет разный возраст. Это обязательно нужно учесть при проектировании приложения.

## 2.2 Функциональные требования

- Пользователь:

* Регистрация/Авторизация;
* Просмотр тарифов;
* Каталог автомобилей;
* Просмотр карты с расположением автомобилей;
* Просмотр информации о профиле, о предыдущих заказах;
* Бронирование авто;
* Пополнение баланса;

- Администратор:

* Авторизация;
* Просмотр информации о пользователях;
* Мониторинг арендованных автомобилей;
* Проверка фотоконтроля автомобиля;
* Тех-поддержка пользователей;

## 2.3 Основные системные требования

Приложение состоит из 5 основных компонентов:

* Сервер, хранящий информацию в базе данных и предоставляющий API для мобильного приложения;
* Сервис для карты, отображения автомобилей и данные с их бортового компьютера.
* Сервис для работы с оплатой.
* Мобильное приложение, позволяющее выполнять функции роли пользователя;
* Реляционная СУБД PSQL,

Сервер предоставляет REST API для взаимодействия клиентов с системой. Передача данных осуществляется в формате JSON. Сервер написан на Go, так как приложения, написанные на этом языке, можно легко масштабировать. Сервер предоставляет WEB-интерфейс для администратора со следующим функционалом:

* Авторизация;
* Карта с расположением автомобилей;
* Информация с сервиса карт;
* Заявки на техподдержку;
* Страница для проверки фотоконтроля;
* Информация о пользователях.

Сервис для карты реализован на JAVA, так как этот язык обеспечивает надежность приложений. Сервис должен постоянно отправлять уведомления в систему, и сообщать о различных нарушениях или неполадках автомобилей.

Сервис для работы с оплатой вынесен в отдельный компонент, так как важно не ошибиться при транзакциях связанных с балансом пользователей. Сервис реализован на Python.

Для общения между сервисами использована Kafka, так как она является асинхронным способом общения между сервисами, а также позволяет хранить журнал сообщений.

Мобильное приложение используется для выполнения функций пользователя, написано на языке Flutter, а также имеет локальную базу данных SQLite, содержащую в себе данные пользователя и информацию, необходимую для работы приложения (без подключения к сети интернет) и последующей синхронизацией с сервером. Имеет следующие экраны для роли пользователя:

* Регистрация\авторизация;
* Главный экран;
* Карта с расположением автомобилей;
* Бронирование авто;
* Каталог автомобилей;
* Информация о тарифах;
* Текущая поездка;
* Профиль.

## 2.4 Аппаратные требования

* ОЗУ 1024Mb;
* OS Android 8.0 и выше (API 23+);
* Сетевая пропускная способность около 1Mb/s;
* Внутренняя память не менее 128Mb.