МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных Технологий

Кафедра Информационных систем и технологий

Специальность 1-98 01 03 «Программное обеспечение информационной безопасности мобильных систем»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Современные технологии программирования мобильных систем»

Тема «Игровой лаунчер»

Исполнитель

студент 3 курса 7 группы Леонов Денис Игоревич

(Ф.И.О.)

Руководитель ассистент Нистюк О.A.

(учен. степень, звание, должность, подпись, Ф.И.О.)

Курсовой проект защищен с оценкой ­

Руководитель Нистюк О.А.

(подпись)

Содержание

Оглавление

[**Введение** 3](#_Toc134567475)

# **Введение**

Курсовой проект представлен в виде полноценного приложения, состоящего из нескольких сервисов, использующих одну базу данных совместно, а также мобильного приложения со своей локальной базой данных. Несмотря на то, что приложение включает в себя несколько сервисов оно не реализует микросервисную архитектуру. Акцентируя внимание на том, что все сервисы используют одну базу данных можно с уверенностью сказать, что приложение является – распределенным монолитом.

Немного углубляясь, можно рассмотреть все сервисы и хранилища этого приложения. Первый сервис – User Service. Он написан на языке Go, соединен с базой данной MySQL, которая расположена на сервере, а также предоставляет пользовательский интерфейс, для реализации которого были выбраны: HTML, CSS, JS. Следующий сервис – Manager Service. Основным инструментом для разработки этого сервиса был выбран TypeScript. Этот сервис соединен с той же базой данный, что и User Service. И наконец мобильное приложение на Flutter. Мобильное приложения является Driver Service и имеет свою локальную базу данных SQLite. Однако стоит учесть, что данные на этот сервис подгружаюся не из MySQL, а с помощью HTTP. Запросы от Driver Service обрабатываются на User Service, благодаря предоставленному REST API.

Все запросы в MySQL базу данных с сервисов, осуществляется с помощью вызова хранимых процедур, а работа Driver Service с локальной базой данных SQLite с помощью соответствующей библиотеки, которая предоставляет функционал похожий на ORM.

Выбор такой архитектуры приложения объясняется бизнес-логикой. Так как позволяет работать сервисам независимо и поддерживать горизонтальное масштабирование.