Департамент образования и науки Приморского края

Краевое государственное бюджетное профессиональная образовательное учреждение

«Находкинский государственный гуманитарно-политехнический колледж»

Форма обучения: очная

Отделение: техническое

**ОТЧЕТ**

По дисциплине: Разработка кода ИС

Студента: Гладких Кирилла Денисовича

Группа: 731 (Веб)

Специальность: «Информационные системы и программирование»

На тему: Тестирование ИС

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Б.М Дияров

Оценка защиты \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(расшифровка, подпись) дата (ДД.ММ.ГГ)

Находка

2023

**Содержание**

Планирование тестирования [3](#_30j0zll)

[1.](#_1fob9te) Дизайн тестов 4

[2.](#_3znysh7) Подготовка тестовой среды 5

[3.](#_2et92p0) Выполнение тестов 6

[4.](#_1ksv4uv) Заключение и анализ результатов

5. Завершение тестовых работ8

# Планирование тестирования

В мобильной версии Delivery Club будет тестироваться функциональная область приложения, тестирование процесса заказа, возможности поиска ресторанов, выбора блюд, оформление заказа и оплаты. Так-же, будет проводится тестирование фильтров и их сортировка для проверки корректной работы данных функций.

В планы тестирования области так же войдет проверка функций учетных записей пользователей, включая авторизацию, регистрацию, и управление данных профиля. Тестирование уведомлений, включая уведомления о статусе заказа, взаимодействие с ресторанами если приложение поддерживает роли ресторанов - тестирование управления меню и заказами от имени ресторана.

Проверка производительности и надежности приложения, скорость загрузки приложения, выполнения ее функций даже при условии низкой скорости интернета.

Для тестирования приложения понадобится свободный аккаунт для регистрации в приложении, один компьютер, один человек, среда разработки для использования Appium-Python-Client, операционная мобильная система с возможностью установки мобильного приложения.

Время тестирования может варьироваться, но не должно превысить 72 часов с момента написания планирования тестирования.

В рисках тестирования можно отметить отказ работоспособности оборудования и его выход из строя по техническим причинам, поэтому с целью разделения нагрузки будет использована виртуальная среда.

# Дизайн тестов

Методы и подходы к тестированию будут выполнены в соответствии с поведением пользователя и его предпочтениями, а так же, эмуляции непредвиденных обстоятельств в отношении устройства и его влияния на приложения таких как лаги, зависание системы и прочее.

**Тест-кейс 1: Регистрация в приложении и проверка данных**

1. Выбрать аккаунт для регистрации
2. Нажать на иконку пользователя
3. Нажать на кнопку данные
4. Выбрать пол
5. Выбрать дату рождения
6. Сохранить изменения
7. Нажать на кнопку уведомления
8. Проверить наличие уведомлений

**Тест-кейс 2: Добавление дополнительных данных**

1. Нажать на иконку пользователя
2. Выбрать значок “Адреса”
3. Нажать на кнопку “Добавить адрес”
4. Разрешить приложению использовать геопозицию
5. Выбрать адрес

# Подготовка тестовой среды.

В данном случае должны были использоваться инструменты в виде Android-SDK и Appium с использованием библиотеки Appium-Python-Client,

отчет должен был вестись через логи, самое приложение было установлено на устройство с ОС IOS 16.

# Выполнение тестов

**Тест-кейс 1: Ручное тестирование**

Тест-кейс: Статус - успех

Дефекты: Не выявлено

**Тест-кейс 2: Ручное тестирование**

Тест-кейс: Статус - успех

Дефекты: Не выявлено

# Заключение и анализ результатов

Фактические результаты были такие же, как и ожидаемые, приложение прошло тестирование в ручном режиме без использования инструментов автоматизации. Будут добавлены тест-кейсы рассчитанные на автоматическое тестироване

## 

# Завершение тестовых работ

Ручное тестирование не выявило никаких дефектов, в следующем цикле тестирования будут использованы инструменты для автоматического тестирования