МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«Новосибирский национальный исследовательский государственный университет»

(Новосибирский государственный университет, НГУ)

Структурное подразделение Новосибирского государственного университета – Высший колледж информатики Университета (ВКИ НГУ)

КАФЕДРА ИНФОРМАТИКИ

**РАРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ СБОРА ДАННЫХ С АКСЕЛЕРОМЕТРА В СИСТЕМ ANDROID ПРИМЕНИТЕЛЬНО К ИЗМЕРЕНИЮ СЕЙСМИЧЕСКИЧ ПРОЦЕССОВ**

Квалификация программист

|  |  |
| --- | --- |
| Руководитель  к.ф.м.н с.н.с лаб.стат.прочности | Ларичкин А.Ю.  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2025 г. |
| Студент 3 курса  гр. 2207сг1 | Рыбалов М.С.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2025 г. |

Новосибирск

2025

**СОДЕРЖАНИЕ**

[ПЕРЕЧЕНЬ СОКРАЩЕНИЙ, УСЛОВНЫХ ОБОЗНАЧЕНИЙ И ТЕРМИНОВ 4](#_gjdgxs)

[ВВЕДЕНИЕ 5](#_30j0zll)

[1 ПОСТАНОВКА ЗАДАЧИ ВКР 6](#_1fob9te)

[1.1 Бизнес-требования 6](#_3znysh7)

[1.2 Пользовательские требования 7](#_2et92p0)

[1.3 Системные требования 7](#_tyjcwt)

[1.4 Требования к пользовательскому интерфейсу 7](#_3dy6vkm)

[1.5 План-график выполнения ВКР 8](#_1t3h5sf)

[2 АНАЛИЗ ТРЕБОВАНИЙ И ОПРЕДЕЛЕНИЕ СПЕЦИФИКАЦИЙ 9](#_4d34og8)

[2.1 Описание предметной области задачи ВКР 9](#_2s8eyo1)

[2.1.1 Информационные объекты предметной области и взаимосвязи между ними 9](#_17dp8vu)

[2.1.2 Информационные потребности пользователей 10](#_3rdcrjn)

[2.1.3 Методы работы с информационными объектами предметной области 10](#_26in1rg)

[2.1.3.1 Способы хранения информации об объектах предметной области 10](#_lnxbz9)

[2.1.3.2 Математические модели, используемые для обработки информации 11](#_35nkun2)

[2.1.3.3 Применяемые программные технологии обработки информации, основанные на математических моделях 11](#_1ksv4uv)

[2.1.3.4 Способы интерпретации и визуального представления информации 11](#_44sinio)

[2.1.3.5 Технологии получения и передачи информации 12](#_2jxsxqh)

[2.1.4 Обзор существующих программных реализаций решения задачи 12](#_z337ya)

[2.1.5 Концептуальное обоснование разработки 12](#_3j2qqm3)

[2.2 Классы и характеристики пользователей 12](#_1y810tw)

[2.3 Функциональные требования 13](#_4i7ojhp)

[2.3.1 Определение функциональных возможностей 13](#_2xcytpi)

[2.3.2 Описание прецедентов 13](#_1ci93xb)

[2.4 Нефункциональные требования 14](#_3whwml4)

[3 ВЫБОР ПРОГРАММНЫХ СРЕД И СРЕДСТВ РАЗРАБОТКИ 15](#_2bn6wsx)

[3.1 Сравнительный анализ имеющихся возможностей по выбору средств разработки 15](#_qsh70q)

[3.2 Характеристика выбранных программных сред и средств 15](#_3as4poj)

[4 АЛГОРИТМЫ РЕШЕНИЯ ПОСТАВЛЕННОЙ ЗАДАЧИ 16](#_1pxezwc)

[4.1 Этапы реализации 16](#_49x2ik5)

[4.2 Пользовательский интерфейс 16](#_2p2csry)

[4.2.1 Описание взаимодействия с пользователем 16](#_147n2zr)

[4.2.2 Определение операций пользователей и составление функциональных блоков 16](#_3o7alnk)

[4.2.3 Проектирование структуры экранов и схемы навигации 17](#_23ckvvd)

[4.2.4 Разработка дизайна интерфейса 17](#_ihv636)

[4.3 Входные, выходные и промежуточные данные 18](#_32hioqz)

[4.4 Реализация используемых методов хранения, обработки и передачи информации об объектах предметной области 18](#_1hmsyys)

[4.4.1 Методы хранения данных 18](#_41mghml)

[4.4.2 Алгоритмы реализации используемых математических моделей 19](#_2grqrue)

[4.4.3 Алгоритмы использования применяемых программных технологий обработки данных 20](#_vx1227)

[4.4.4 Алгоритмы применения методов графического анализа данных 20](#_3fwokq0)

[4.4.5 Алгоритмы использования технологий передачи данных 20](#_1v1yuxt)

[4.5 Описание архитектурного решения 21](#_4f1mdlm)

[4.5.1 Структурная организация программной системы 21](#_2u6wntf)

[4.5.2 Архитектура программного кода 21](#_19c6y18)

[5 ТЕСТИРОВАНИЕ И ОПТИМИЗАЦИЯ 22](#_3tbugp1)

[5.1 План тестирования 22](#_28h4qwu)

[5.2 Результаты тестирования и оптимизация 22](#_nmf14n)

[6 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 23](#_37m2jsg)

[ЗАКЛЮЧЕНИЕ 24](#_1mrcu09)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 25](#_46r0co2)

[ПРИЛОЖЕНИЯ 26](#_2lwamvv)

[Приложение А 26](#_111kx3o)

[Приложение Б 26](#_3l18frh)

[Приложение В 26](#_206ipza)

[Приложение Г 26](#_4k668n3)

[Приложение Д 26](#_2zbgiuw)

[Приложение Е 27](#_1egqt2p)

# ПЕРЕЧЕНЬ СОКРАЩЕНИЙ, УСЛОВНЫХ ОБОЗНАЧЕНИЙ И ТЕРМИНОВ

{Принятые в ПЗ малораспространенные сокращения, условные обозначения, символы, единицы и специфические термины должны быть представлены в виде отдельного списка.

Если сокращения, условные обозначения, символы, единицы и термины повторяются в ПЗ менее трех раз, отдельный список не составляют, а расшифровку дают непосредственно в тексте ПЗ при первом упоминании.

Даже если в работе не используются сокращения и аббревиатуры, следует привести **СПИСОК СПЕЦИАЛЬНЫХ ТЕРМИНОВ, ХАРАКТЕРИЗУЮЩИХ ПРЕДМЕТНУЮ ОБЛАСТЬ ЗАДАЧИ И ТЕХНОЛОГИИ**, применяемые для её решения.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

# ВВЕДЕНИЕ

Сейсмическая активность является одной из ключевых областей изучения в современной науке, поскольку она существенно влияет на безопасность инфраструктуры и жизни людей. Мониторинг сейсмических процессов позволяет не только своевременно предупреждать о возможных природных катаклизмах, но и исследовать геофизические особенности территорий, что имеет важное значение для научных и прикладных задач.

В России, с её значительной территорией, расположенной в зонах повышенной тектонической активности, вопросы контроля сейсмической активности остаются чрезвычайно актуальными. Особенно это касается Сибири, где активное освоение недр и проведение шахтёрских работ сопряжены с риском возникновения индуцированных сейсмических событий. Наблюдение за такими процессами позволяет минимизировать риски для горнодобывающих предприятий и прилегающих населённых пунктов.

В Институте гидродинамики имени М.А. Лаврентьева СО РАН уже ведутся работы, связанные с мониторингом и анализом сейсмической активности. Однако существующие системы сбора данных зачастую громоздки, требуют значительных затрат на оборудование и обслуживание, а также не обеспечивают необходимой мобильности.

Целью работы является разработка программного приложения для смартфонов на платформе Android, позволяющего собирать, анализировать и визуализировать данные с датчиков различного назначения для мониторинга сейсмических процессов.

Для достижения цели были поставлены следующие задачи:

1. Провести анализ существующих решений и технологий в области мобильного мониторинга сейсмической активности.
2. Определить требования к программному обеспечению, включая функциональные и нефункциональные аспекты.
3. Разработать архитектуру мобильного приложения для работы с данными сейсмических датчиков.
4. Создать пользовательский интерфейс, обеспечивающий удобство работы с приложением.
5. Реализовать механизм сбора данных с подключаемых датчиков через стандартные интерфейсы связи.
6. Разработать алгоритмы предварительной обработки и анализа данных.
7. Внедрить функции визуализации результатов мониторинга в реальном времени.
8. Провести тестирование разработанного приложения, включая проверку его работы в полевых условиях.

Используемые технологии включают Android Studio и язык программирования Kotlin, что позволяет создавать современные и производительные мобильные приложения.

Практическая значимость проекта заключается в создании доступного и удобного инструмента для оперативного сбора и анализа данных сейсмической активности. Предлагаемая система повысит мобильность и эффективность мониторинга, а также обеспечит возможность её использования в удалённых регионах.

# 1 ПОСТАНОВКА ЗАДАЧИ ВКР

## 1.1 Бизнес-требования

Основной задачей Института гидродинамики имени М.А. Лаврентьева является мониторинг и анализ сейсмических процессов, особенно в Сибири и на объектах, где ведутся шахтёрские работы. Разработка мобильного приложения для сбора и анализа данных сейсмических датчиков значительно улучшит оперативность и точность мониторинга.

Бизнес-требования включают:

1. Оперативность и мобильность: Существующие стационарные системы требуют больших затрат на установку и обслуживание. Мобильное приложение обеспечит доступность мониторинга в полевых условиях.
2. Улучшение безопасности: Быстрое получение данных о сейсмических колебаниях позволит снизить риск для сотрудников шахт и предприятий.
3. Автоматизация процессов анализа: Приложение позволит автоматически собирать и обрабатывать данные с различных датчиков, минимизируя ручной труд и вероятность ошибок.

## 1.2 Пользовательские требования

Система предназначена для сбора и анализа данных с различных датчиков, что требует четкой архитектуры и надежной интеграции компонентов. Важной задачей является обеспечение бесперебойной работы всех модулей и правильной обработки полученных данных. Особое внимание уделяется стабильности и производительности приложения при работе с большими объемами информации. Разрабатываемое приложение должно предоставлять пользователям следующие возможности:

1. Сбор данных в реальном времени: Получение данных с подключённых сейсмических датчиков и отображение их на экране мобильного устройства.
2. Анализ данных: Встроенные алгоритмы для обработки полученной информации и её первичного анализа.
3. Визуализация данных: Построение графиков и диаграмм на основе собранных данных, удобное для восприятия как специалистами, так и студентами.
4. Хранение и экспорт данных: Возможность сохранять записи и экспортировать их для дальнейшего анализа на других устройствах.
5. Уведомления о критических изменениях: Оповещения при превышении заданных пороговых значений сейсмической активности.
6. Настраиваемые параметры: Пользователи смогут задавать свои пороговые значения, настраивать типы подключаемых датчиков и их параметры.
7. Многопользовательский доступ: Обеспечение разграничения прав доступа для специалистов и студентов.
8. Доступ к справочной информации: Интегрированный справочник и инструкция по работе с приложением.

Подробнее технические требования и функции приложения приведены в Приложении А.

## 1.3 Системные требования

Система предназначена для сбора и анализа данных с различных датчиков, что требует четкой архитектуры и надежной интеграции компонентов. Важной задачей является обеспечение бесперебойной работы всех модулей и правильной обработки полученных данных. Особое внимание уделяется стабильности и производительности приложения при работе с большими объемами информации. Приложение состоит из нескольких основных модулей:

1. Модуль сбора данных: Отвечает за подключение к сейсмическим датчикам через стандартные интерфейсы (Bluetooth, USB, Wi-Fi).
2. Модуль обработки данных: Обеспечивает фильтрацию шумов, обработку сигналов и вычисление параметров сейсмической активности.
3. Модуль визуализации: Построение графиков и диаграмм для отображения данных в реальном времени.
4. Модуль управления данными: Позволяет сохранять записи и экспортировать их в различных форматах (CSV, JSON).
5. Пользовательский интерфейс: Интуитивно понятный интерфейс с разграничением доступа по ролям.

Система разрабатывается на базе Android Studio с использованием языка Kotlin и предназначена для работы на устройствах с операционной системой Android версии 7.0 и выше.

## 1.4 Требования к пользовательскому интерфейсу

Пользовательский интерфейс должен быть интуитивно понятным и удобным как для специалистов, так и для обычного пользователя. Требования к интерфейсу включают:

1. Главный экран: Отображение текущих данных с датчиков и основных параметров сейсмических процессов.
2. Графики в реальном времени: Визуализация показателей в виде графиков и диаграмм.
3. Меню навигации: Доступ к различным модулям приложения (настройки, архив данных, отчёты).
4. Интерактивность: Пользователь должен иметь возможность масштабировать графики, просматривать детальную информацию по каждому событию.
5. Сохранение данных: Интуитивно понятная кнопка для сохранения текущих данных и их экспорта.
6. Дизайн: Минималистичный, с акцентом на функциональность. Цветовая схема должна облегчать восприятие информации.

## 1.5 План-график выполнения ВКР

В данном разделе представлен общий план-график выполнения работ по созданию системы для сбора и анализа данных с сейсмических датчиков. Планирование основано на методологии последовательной разработки, включающей ключевые этапы от анализа требований до тестирования и внедрения системы. Диаграмма Ганта представлена на рисунке 1.

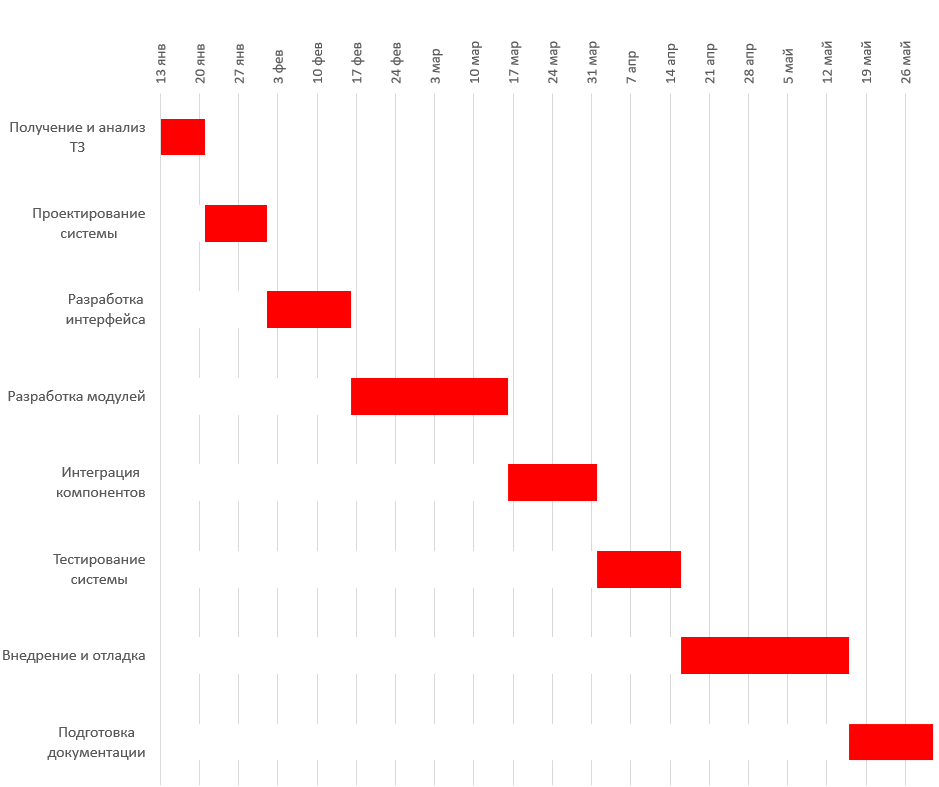


Рисунок 1. План-График работы в виде диаграммы Ганта

# 2 АНАЛИЗ ТРЕБОВАНИЙ И ОПРЕДЕЛЕНИЕ СПЕЦИФИКАЦИЙ

## 2.1 Описание предметной области задачи ВКР

### 2.1.1 Информационные объекты предметной области и взаимосвязи между ними

Современные системы сбора и анализа данных с датчиков активно развиваются благодаря их важной роли в решении научных и прикладных задач. Предметная область сейсмических процессов характеризуется сложными взаимосвязями между природными явлениями, физическими процессами и технологиями обработки данных. Создание мобильного приложения, способного интегрировать данные с различных сенсоров, требует глубокого анализа этих взаимосвязей и структуры данных. Рассмотрим ключевые информационные объекты, их характеристики и способы взаимодействия.

Информационные объекты в контексте мобильного приложения включают:

1. Датчики: устройства, собирающие информацию о вибрациях, ускорениях и изменениях в окружающей среде. Примеры: акселерометры, геофоны.
2. Сырые данные: данные, полученные непосредственно с датчиков, которые требуют предварительной обработки.
3. Обработанные данные: результаты преобразования исходной информации с применением фильтрации, анализа и других методов.
4. Модели сейсмических процессов: математические и графические представления данных для анализа активности земной коры.
5. Пользовательские интерфейсы и отчеты: элементы взаимодействия приложения с пользователем, включая графики, таблицы и текстовые интерпретации.

Взаимосвязи между объектами обеспечивают последовательность и полноту обработки данных. Например, данные с акселерометров передаются в модуль обработки, где они преобразуются в удобный для анализа формат. Математические модели помогают интерпретировать результаты, делая их доступными для пользователей с разным уровнем подготовки.

Каждый информационный объект имеет свои особенности, влияющие на работу системы. Например, выбор типа датчиков определяет точность и частоту регистрации событий, а алгоритмы обработки данных влияют на качество аналитических выводов. Основное внимание уделяется обеспечению интеграции данных, что делает систему надежным инструментом для мониторинга и изучения сейсмической активности.

### 2.1.2 Информационные потребности пользователей

Эффективное взаимодействие пользователей с системой сбора и анализа данных напрямую зависит от удобства и наглядности представления информации. В разрабатываемом приложении для мониторинга и анализа сейсмических процессов предусмотрено использование различных форм отображения данных, которые адаптируются под потребности специалистов и обучающихся в данной области. Основная цель – предоставить исчерпывающие данные в удобной и легко интерпретируемой форме, поддерживая высокий уровень точности и информативности.

Информация представляется в следующих формах:

1. Графики сейсмической активности – временные ряды, отражающие амплитуду колебаний, регистрируемую сенсорами.
2. Спектральные диаграммы – визуализация частотного анализа сейсмических данных, позволяющая выделить ключевые частотные компоненты.
3. Картографические данные – отображение расположения источников сейсмической активности на карте с возможностью масштабирования и выбора временных диапазонов.
4. Результаты обработки данных, включая амплитуды, частоты, координаты и временные отметки событий.
5. Сводные отчеты, описывающие динамику изменений параметров за выбранный период.
6. Фильтры для выбора данных по определенным параметрам (временной интервал, тип сенсора, географическая область).
7. Вкладки и панели управления, где пользователи могут переключаться между различными представлениями данных.
8. Исходные данные: Непрерывный поток измерений от акселерометров, включающий амплитуды и частоту колебаний с временной меткой.
9. Обработанные данные: Усредненные и сглаженные значения амплитуд, спектральные характеристики, выявленные пиковые события, автоматически обозначенные как потенциальные сейсмические явления.

Методы работы с информацией, реализованные в приложении, направлены на интеграцию, фильтрацию и интерпретацию данных, предоставляя пользователям результаты в удобном виде. Такой подход к отображению данных позволяет пользователям разного уровня подготовки решать прикладные задачи, включая оценку рисков, прогнозирование событий и обучение.

### 2.1.3 Методы работы с информационными объектами предметной области

В приложении для мониторинга и анализа сейсмических процессов используются разнообразные подходы для обработки данных. Выбор методов и технологий направлен на обеспечение высокой точности анализа, удобства для пользователей и масштабируемости системы. Рассмотрим основные методы работы с информацией, включая способы хранения, обработки и визуализации данных, а также технологии передачи информации.

#### 2.1.3.1 Способы хранения информации об объектах предметной области

Хранение информации в приложении организовано с учетом особенностей предметной области. Важнейшая задача – сохранить исходные данные, результаты обработки и вспомогательные метаданные в удобной для анализа и дальнейшей обработки форме.

Информация хранится в реляционной базе данных, что позволяет четко структурировать данные:

1. Таблица "Сенсоры" содержит уникальные идентификаторы, параметры сенсоров (тип, диапазон измерений, точность), а также их местоположение.
2. Таблица "Данные измерений" содержит временные ряды, поступающие с сенсоров, включая амплитуды, частоты и временные метки.
3. Таблица "События" фиксирует обработанные данные: обнаруженные аномалии, их координаты, время регистрации и классификацию.

Использование реляционной базы данных обосновано следующими причинами:

1. Логическая целостность данных: строгие ограничения гарантируют отсутствие дублирования.
2. Гибкость при обработке: SQL-запросы позволяют извлекать данные для анализа в различных разрезах.
3. Интеграция с внешними системами: данные могут экспортироваться в популярные форматы (CSV, JSON) для использования в сторонних аналитических приложениях.

Также система поддерживает работу с файлами, например, логами и отчетами. Структура каталогов организована по временным интервалам и типам данных, что упрощает поиск нужной информации.

#### 2.1.3.2 Математические модели, используемые для обработки информации

Анализ данных в приложении осуществляется с помощью математических моделей, которые позволяют извлекать ключевые параметры из потоков измерений. Применяются как классические методы, так и более сложные подходы.

Анализ временных рядов:

Фурье-преобразование: разложение сигналов на частотные компоненты для выделения доминирующих частот. Это особенно важно для выявления характерных сейсмических волн. Формула Фурье-преобразования:

Методы фильтрации:

1. Применяется сглаживание методом скользящего среднего, что помогает устранить шумы и выделить главные тенденции.
2. Используются высоко- и низкочастотные фильтры для выделения сейсмических сигналов определенных диапазонов.

Волновое уравнение:

Применяется для моделирования распространения упругих волн в геологических структурах, где u – смещение, c – скорость распространения волны.

Математические модели адаптированы к специфике сейсмических данных, что позволяет извлекать максимальную полезную информацию.

#### 2.1.3.3 Способы интерпретации и визуального представления информации

Методы интерпретации и визуального представления данных играют ключевую роль в анализе информации о сейсмических процессах. Они позволяют преобразовать массивы чисел и параметров в удобные для восприятия графики, диаграммы и схемы. Это значительно упрощает процесс анализа данных как для профессионалов, так и для тех, кто только осваивает область сейсмологии. Визуальные инструменты являются мостом между теоретическими расчётами и их практическим применением, позволяя эффективно выявлять закономерности и аномалии в поведении исследуемых объектов. В рамках разработки системы используются графические методы, включающие построение линейных графиков, гистограмм, спектральных диаграмм и трёхмерных моделей, чтобы отразить изменения сейсмических параметров в пространстве и времени. Например, графики временных рядов предоставляют пользователям возможность отслеживать амплитуду вибраций или частотный спектр на конкретных участках за определённый промежуток времени. Спектральные диаграммы используются для анализа частотных характеристик сейсмических сигналов, выявляя доминирующие частоты и их изменения.

Гистограммы применяются для анализа распределения данных, таких как амплитуды колебаний или плотность событий в заданных интервалах времени. Для сложных данных, включающих пространственные компоненты, применяется трёхмерная визуализация, которая помогает анализировать географическое распределение источников сейсмических процессов.

Пользовательский интерфейс приложения включает интерактивные элементы: пользователи могут увеличивать, перемещать и масштабировать графики для более детального анализа. Данные могут отображаться как в реальном времени, так и с возможностью ретроспективного анализа. Таким образом, представленные методы позволяют использовать визуальные инструменты как для изучения глобальных закономерностей, так и для детального изучения отдельных событий.

#### 2.1.3.5 Технологии получения и передачи информации

В современном мире важнейшим компонентом обработки данных является их своевременное получение и передача. Для сейсмологических исследований это особенно актуально, поскольку работа с такими процессами требует оперативного взаимодействия с устройствами, фиксирующими данные, и удалёнными системами хранения. Эффективные технологии получения и передачи информации являются основой точной и достоверной работы системы.

Разрабатываемая система взаимодействует с датчиками сейсмических процессов через стандартизированные протоколы передачи данных, такие как Bluetooth и Wi-Fi. Эти технологии позволяют устройству получать данные в режиме реального времени с минимальной задержкой, что крайне важно для мониторинга сейсмической активности. Например, при работе с акселерометрами данные о вибрациях передаются через Bluetooth-соединение с частотой обновления до 100 Гц.

Для работы с облачными сервисами используется передача данных через REST API, что позволяет интегрировать приложение с внешними платформами для хранения и анализа больших объёмов данных. API предоставляет гибкость в форматах передачи данных (например, JSON или XML), что упрощает их обработку и дальнейшее использование.

Кроме того, реализована функция локального сохранения данных в памяти устройства с последующей синхронизацией. Это особенно полезно в случаях, когда доступ к интернету временно недоступен. После восстановления соединения приложение автоматически передаёт накопленные данные на сервер, обеспечивая их сохранность и доступность для анализа.

Интеграция данных из различных источников позволяет не только обрабатывать текущие параметры, но и использовать сторонние библиотеки данных для сравнения, прогнозирования и интерпретации сейсмической активности. Такой подход гарантирует гибкость и адаптивность системы в различных условиях эксплуатации.

### 2.1.4 Обзор существующих программных реализаций решения задачи

Современный рынок предлагает множество мобильных приложений, направленных на сбор и обработку данных с различных датчиков. Однако большинство из них ориентированы на решение узкоспециализированных задач или обладают ограниченным функционалом, что делает их менее удобными для комплексного анализа сейсмических процессов. В данном разделе приведены три аналогичных программных средства с их достоинствами и недостатками.

* 1. Physics Toolbox Suite

Ссылка: https://www.vieyrasoftware.net/physics-toolbox-suite  
Physics Toolbox Suite — это приложение, предоставляющее доступ к множеству инструментов для измерений с использованием встроенных сенсоров смартфона. Программа позволяет регистрировать параметры, такие как ускорение, давление, звуковые колебания, магнитное поле и многие другие, с возможностью записи данных для дальнейшего анализа.

* Плюсы: Обширный набор инструментов и датчиков. Возможность экспорта данных в популярные форматы для последующего анализа. Дружественный интерфейс и поддержка нескольких платформ.
* Минусы: Отсутствие специализированных алгоритмов для обработки сейсмических данных. Ограниченные возможности визуализации данных. Сложности с калибровкой датчиков для точных научных измерений.

A screenshot of a computer

Description automatically generated

Рисунок 2 – Physics Toolbox Suite

2. Sensor Kinetics Pro

Ссылка: https://www.innoventions.com/sensor-kinetics  
Sensor Kinetics Pro — это приложение для мониторинга и анализа работы встроенных датчиков смартфона. Оно предлагает пользователям гибкие настройки отображения данных и позволяет получать графики в реальном времени.

* Плюсы: Простота в использовании и быстрая настройка параметров. Поддержка реального времени для отображения данных сенсоров. Возможность настройки фильтров для подавления шума в данных.
* Минусы: Отсутствие специализированных методов обработки сейсмических процессов. Невозможность работы с большими объемами исторических данных. Не поддерживает комплексный анализ с использованием нескольких сенсоров одновременно.

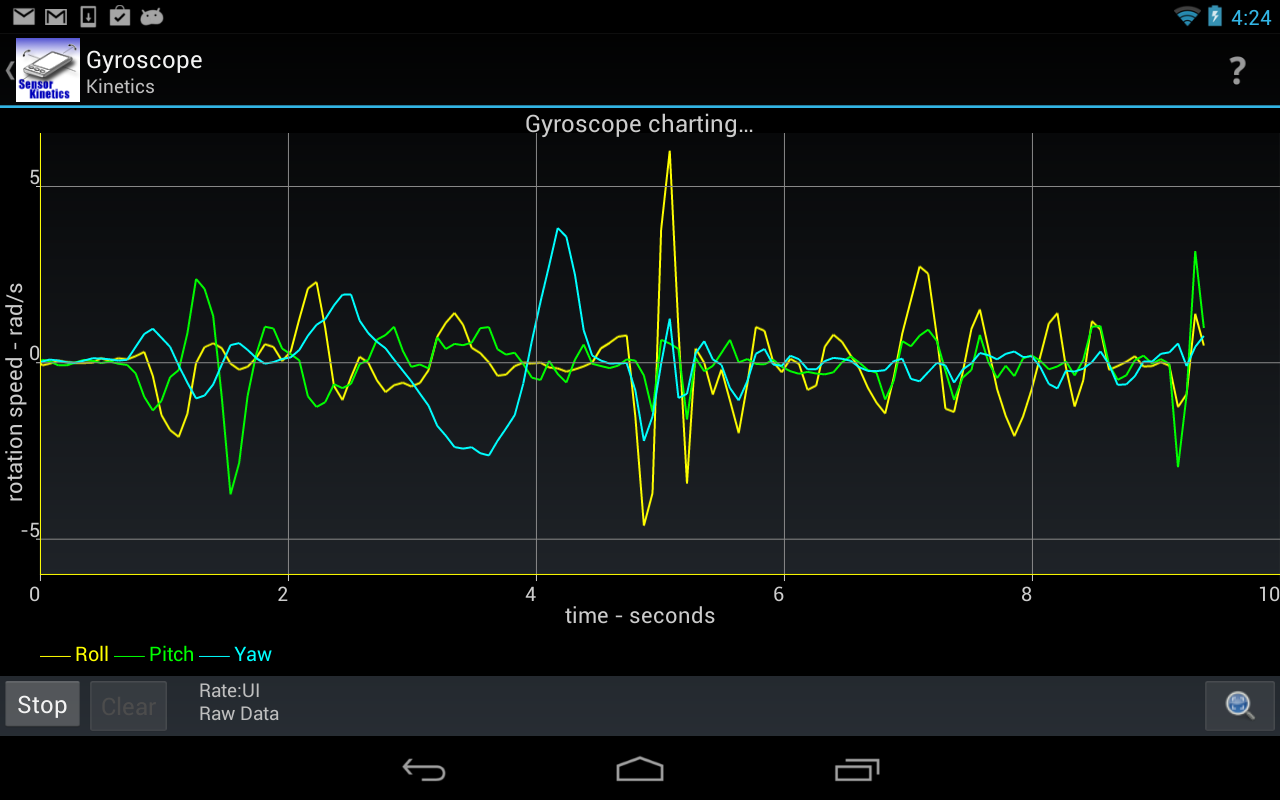


Рисунок 3 – Sensor Kinetics Pro

3. MyShake

Ссылка: https://myshake.berkeley.edu/  
MyShake — мобильное приложение, специально разработанное для сбора и анализа данных о сейсмической активности. Оно также позволяет пользователям подключаться к глобальной сети мониторинга землетрясений, предоставляя возможность наблюдения за сейсмическими процессами в режиме реального времени.

* Плюсы: Специализированная направленность на анализ сейсмических процессов. Доступ к глобальным данным о землетрясениях через сеть. Встроенные алгоритмы для распознавания сейсмических событий.
* Минусы: Ограниченный доступ к данным, собранным другими пользователями. Отсутствие функций для глубокой аналитики или визуализации данных. Невозможность настройки интерфейса под индивидуальные задачи пользователя.

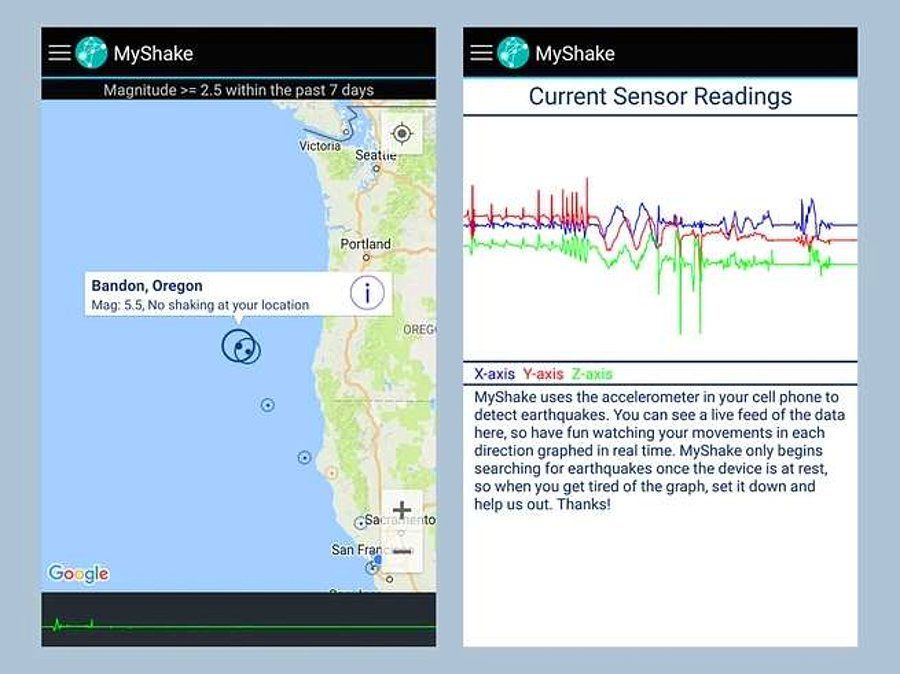


Рисунок 4 – MyShake

Анализ представленных аналогов показывает, что существующие решения либо имеют недостаточную функциональность для углубленного анализа сейсмических процессов, либо слишком узкоспециализированы. Разрабатываемое программное средство будет объединять их сильные стороны, предлагая широкие возможности по сбору, обработке, визуализации и интерпретации данных, что делает его уникальным на рынке.

## 2.2 Классы и характеристики пользователей

В данном мобильном приложении выделяется только один тип пользователя — сотрудник, который использует устройство для получения и анализа сейсмических данных с помощью встроенного акселерометра.

Основные функции пользователя включают:

* Сбор данных акселерометра для анализа сейсмической активности.
* Просмотр и сохранение результатов измерений.
* Передача данных для дальнейшей обработки.

На основании предметной области можно выделить только один тип пользователей. Таблица 1, расположенная ниже, показывает какими характеристиками будут обладать пользователи разрабатываемого ПС.

Таблица 1–Характеристика пользователей

|  |  |
| --- | --- |
| Пользователь | Сотрудник |
| Социальные характеристики | Мужчины и женщины от 18 лет. Русскоязычные. Средний уровень владения смартфонами и мобильными приложениями. |
| Мотивационно-целевая среда | Интерес к сбору и анализу сейсмических данных. Высокая мотивация к использованию специализированного ПО. |
| Навыки и умения | Базовые навыки работы с мобильными устройствами и установкой приложений. |
| Требования к ПО и ИС | Удобность. Надежность сбора данных. Возможность хранения и передачи результатов измерений. |
| Задачи пользователя | Сбор сейсмических данных. Сохранение результатов. Передача данных на сервер. |
| Рабочая среда | Мобильное устройство с установленным приложением. Доступ к интернету для передачи данных. |

Исходя из таблицы характеристики пользователей можно выделить юскейсы. Далее следует таблица 2 с описанием всех юскейсов пользователя программного средства.

Таблица 2–Реестр юскейсов

|  |  |  |  |
| --- | --- | --- | --- |
| Пользователь | Потребности | UseCase | Сценарий |
| Сотрудник | Сбор данных акселерометра для анализа сейсмической активности. | Сбор данных акселерометра. | * Инициализировать акселерометр. * Начать запись данных. * Сохранить файл. |
|  | Просмотр и сохранение результатов измерений. | Просмотр и сохранённых данных. | * Открыть сохраненные данные. * Просмотреть графики/таблицы. |
|  | Передача данных для дальнейшей обработки. | Передача данных. | * Выбрать файл с данными. * Выбрать способ отправки. * Отправить данные. |

На основе таблицы 2, с помощью языка UML, было создано наглядное представление всех UseCase. На рисунке 4 представлена диагрмма UseCase.

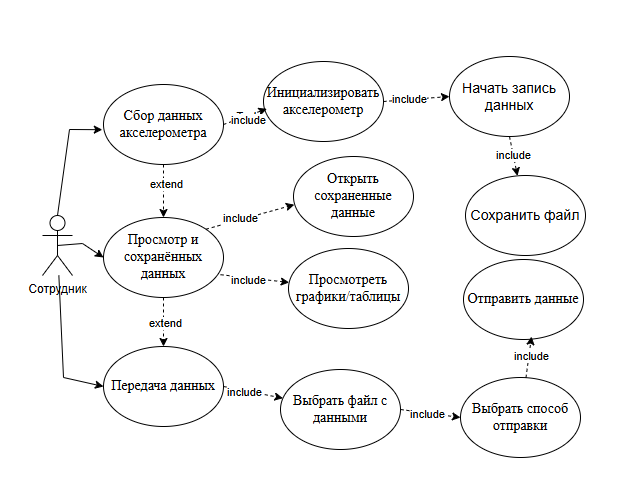


Рисунок – 5 Диаграмма UseCase

При помощи таблицы Реестр юскейсов разработана UML диаграмма активности для пользователя Сотрудник представлена на рисунке 6

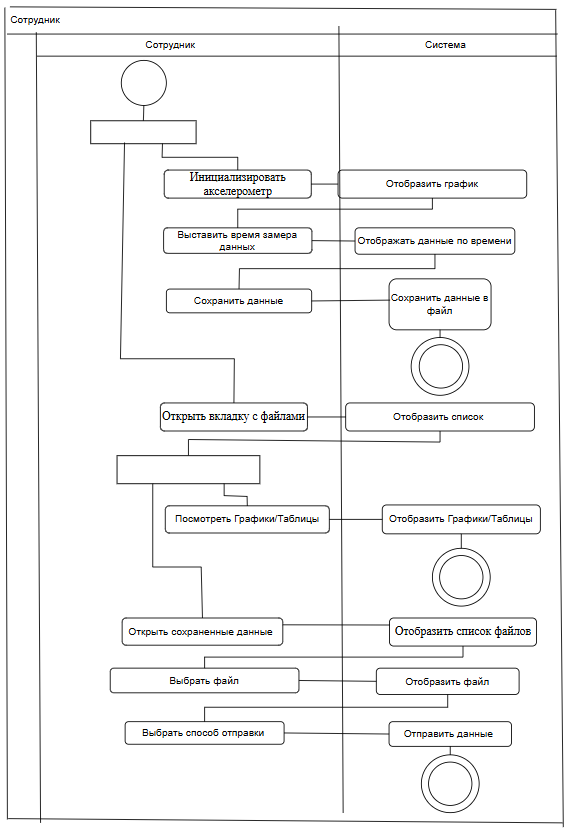


Рисунок – 6 Диаграмма активности пользователя Сотрудник

Ниже на рисунке 7 приведена диаграмма объектов разрабатываемого приложения.

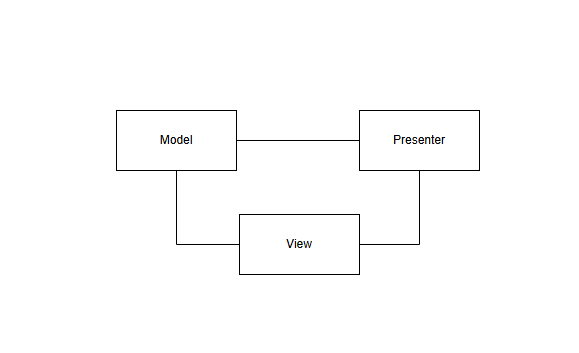


Рисунок – 7 Диаграмма объектов MVP

Далее на рисунке 8 изображена диаграмма последовательностей для usecase Создание нового файла данных.

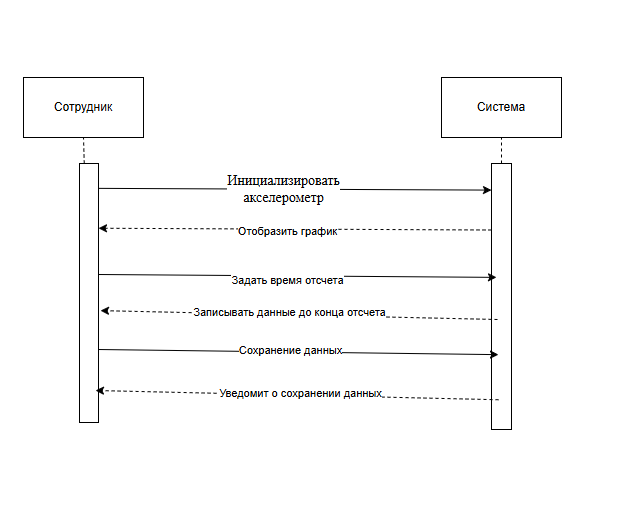


Рисунок – 8 Диаграмма последовательностей

Диаграмма Взаимодействия изображённая на рисунке 9 является альтернативным представлением диаграммы последовательностей



Рисунок – 9 Диаграмма взаимодействия

На рисунке 10 в качестве альтернативы диаграммы классов изображена диаграмма компонентов.

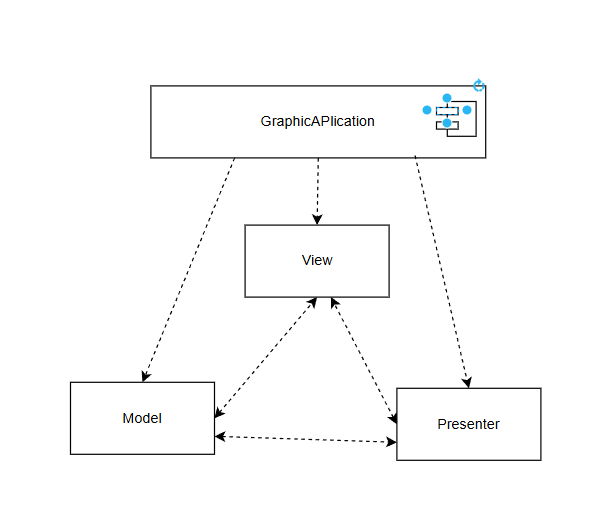


Рисунок – 10 Диаграмма компонентов

## 2.3 Функциональные требования

{Функциональные требования – определяют функциональность (поведение) программной системы, которая должна быть создана разработчиками для предоставления возможности выполнения пользователями действий в рамках бизнес-требований и в контексте пользовательских требований.

Группа функциональных требований определяет набор задач, которые система должна выполнять. }

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 2.3.1 Определение функциональных возможностей

{**Необходимо привести:**

* общее текстовое описание возможностей программы,
* возможности разных категорий пользователей при работе с ПС в виде реестра юскейсов (включая пошаговую реализацию вариантов использования!),
* UML диаграмму вариантов использования ПС.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 2.3.2 Описание прецедентов

{Следует привести текстовые либо табличные описания прецедентов. Обязательно рассмотреть основные успешные сценарии для каждого из описанных прецедентов!}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

## Нефункциональные требования

1. **Требования к производительности и масштабируемости**

* Приложение должно быстро обрабатывать запросы на запуск, остановку сбора данных и выполнение анализа.
* Время обработки данных для расчета АЧХ не должно превышать 5 секунд при объеме данных до 3 минут.
* Приложение должно корректно функционировать при сборе данных в фоновом режиме на протяжении не менее 1 часа без потери производительности.

1. **Требования к переносимости и совместимости**

* Приложение должно поддерживаться на устройствах под управлением Android версии 8.0 (API Level 26) и выше.
* Программное обеспечение не должно конфликтовать с другими приложениями и процессами, работающими на устройстве, включая Bluetooth-соединения.
* Приложение должно корректно отображать графические элементы интерфейса на экранах разных размеров (адаптивный дизайн).

1. **Требования к надежности, доступности и ремонтопригодности.**

* Приложение должно быть устойчиво к аварийным сбоям. В случае их возникновения данные должны сохраняться автоматически для предотвращения их утраты.
* Доступность приложения должна составлять не менее 99% времени работы устройства.
* Среднее время восстановления работы приложения после сбоя не должно превышать 2 минут.

1. **Требования к безопасности**

* Система должна предотвращать несанкционированный доступ к сохраненным данным.
* Данные должны сохраняться в безопасных форматах (\*.csv, \*.txt, \*.xml), исключающих вероятность их повреждения.
* Передача данных через Bluetooth и в социальные сети должна быть защищена стандартными механизмами шифрования.

1. **Требования к локализации**

* Приложение должно поддерживать русский язык интерфейса. Возможность выбора английского языка добавляется по желанию пользователя.
* Все даты и время должны быть отображены в формате, принятом для региона пользователя.

1. **Требования к удобству использования (юзабилити)**

* Интерфейс приложения должен быть интуитивно понятным и минималистичным, с четкой навигацией между настройками, данными и визуализацией.
* Пользователь должен иметь доступ к инструкциям или подсказкам для выполнения основных операций, таких как настройка параметров сбора данных.
* Визуализация данных должна быть представлена в удобных для чтения графиках и таблицах.

Условия разработки:

1. Среда разработки:

* Android Studio: Основная среда разработки.
* **Языки программирования**: Java/Kotlin для реализации функциональности.
* Технологии и библиотеки: Bluetooth API, библиотека для работы с графиками (например, MPAndroidChart).

1. Ограничения и стандарты:

* Разработка должна вестись в соответствии с рекомендациями Google по проектированию Android-приложений.
* Исходный код должен быть структурированным, а ключевые процессы покрыты комментариями для облегчения дальнейшего сопровождения.

1. План тестирования:

* Проведение функционального, пользовательского и нагрузочного тестирования на устройствах с разными версиями Android.
* Обеспечение стабильности работы при тестировании на реальных устройствах с различными аппаратными характеристиками.

1. Сроки выполнения:

* Все этапы разработки должны быть завершены к 1 мая 2025 года.

# 3 ВЫБОР ПРОГРАММНЫХ СРЕД И СРЕДСТВ РАЗРАБОТКИ

{Провести анализ возможных сред и средств разработки программной системы.

На основе анализа осуществить выбор программного инструментария.

Обосновать выбор программных средств разработки приложения, охарактеризовав выбранные среды и средства, привести их достоинства, недостатки, сложности применения.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

## 3.1 Сравнительный анализ имеющихся возможностей по выбору средств разработки

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

## 3.2 Характеристика выбранных программных сред и средств

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

# 4 АЛГОРИТМЫ РЕШЕНИЯ ПОСТАВЛЕННОЙ ЗАДАЧИ

## 4.1 Этапы реализации

{Привести список этапов реализации ПС или ПТС.

Могут использоваться два способа описания:

- словесный: в виде последовательности шагов по реализации;

- графический: в виде блок-схем, обязательно с пояснениями.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

## 4.2 Пользовательский интерфейс

### 4.2.1 Описание взаимодействия с пользователем

{В рамках проектирования пользовательского интерфейса разрабатывается логика диалога с пользователем. Разрабатывается диалоговый пользовательский интерфейс – продумывается пользовательский опыт в виде пользовательских сценариев или пользовательских диалогов. Опираясь на уже разработанную диаграмму вариантов использования, следует определить множество необходимых диалогов. При проектировании диалога необходимо определить структуру диалога (основные сообщения).

**Привести алгоритмы взаимодействия пользователей с ПС в рамках прецедентов с использованием UML-диаграмм последовательности.**

**Общую схему взаимодействия пользователя с ПС, учитывающую возможные сценарии развития диалога, отобразить с использованием** **UML-диаграмм активности**.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.2.2 Определение операций пользователей и составление функциональных блоков

{Определить операции пользователей, которые они смогут выполнять в рамках прецедентов. Привести пронумерованный список операций пользователя. Совокупность операций пользователя определяет множество действий, выполняемых пользователем с помощью объектов интерфейса (*например: нажатие на кнопку, запускающую расчетную функцию*). Каждая операция соответствует определенной реакции системы (например: появляется всплывающее окно с результатами расчета).

Выделить отдельные функциональные блоки. Под функциональным блоком подразумевают группу функций (операций), связанных по назначению или области применения или группу функций информационного наполнения. Как правило, операции, относящиеся к одному функциональному блоку, располагаются на одном графическом окне или на нескольких окнах, связанных операциями перехода.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.2.3 Проектирование структуры экранов и схемы навигации

{На этом этапе, проектируются взаимосвязи форм (окон) приложения и взаимосвязи объектов внутри форм.

Основываясь на сценариях работы и ролях пользователей, формируется структура экранов системы, т.е. определяется количество экранов, функциональность каждого из них, навигационные связи между ними, формируется структура меню и других навигационных элементов.

**Привести навигационную схему.** Навигационная схема показывает механизм распределения функций и задач между экранами.

Навигационная схема определяет, каким образом пользователи смогут перемещаться между различными экранами.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.2.4 Разработка дизайна интерфейса

{На этом этапе определяется визуальный стиль приложения, цветовая схема, создаются макеты экранов и элементов интерфейса. Cоздаются иконки, фоны, шрифты и другие графические элементы, которые будут использоваться в приложении.

На данном этапе разрабатываются интерфейсы конкретных экранов системы (состав, взаимное расположение интерфейсных элементов).

**Привести планы отдельных экранных форм, прототипы окон, страниц** и т.п.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

## 4.3 Входные, выходные и промежуточные данные

{В данном разделе описывается структура входных, выходных и про-межуточных данных. Например, входные данные поступают на вход программы в виде файла, значит, структура файла должна быть полностью описана. Аналогичное требование выполняется для промежуточных и выходных данных.

*Например, для сайтов входной информацией могут быть текстовые файлы (указать в каком формате), графическая информация (указать формат) и т. п. Выходная информация – это HTML, PHP и т. п. страницы, видимые в окне браузеров (указать, каких).*}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

## 4.4 Реализация используемых методов хранения, обработки и передачи информации об объектах предметной области

{Описать, как именно будут применяться в задаче методы, описанные в п. 2.1.3.

В ***п. 4.4.1- 4.4.5 привести алгоритмы реализации выбранных методов для конкретной задачи (тема вкр)***}

**СЛЕДУЕТ ВЫБРАТЬ ПУНКТЫ, СООТВЕТСТВУЮЩИЕ ПРИМЕНЯЕМЫМ МЕТОДАМ ХРАНЕНИЯ, ПРЕДСТАВЛЕНИЯ, ИНТЕРПРЕТАЦИИ ИЛИ ОБРАБОТКИ ИНФОРМАЦИИ, описанным в п. 2.1.3.1-2.1.3.5)**

### 4.4.1 Методы хранения данных

{В этом разделе необходимо указать тип и описать структуру базы данных, если таковая имеется в проекте.

При описании структуры базы данных **описываются все таблицы** в виде:

- имя таблицы и её назначение, т.е. для хранения какой информации предназначена данная таблица;

- описание всех полей таблицы с указанием типа, назначения, первичных и внешних ключей.

**Приводится ER-диаграмма**, иллюстрирующая схему связей между таблицами.

Для объектной базы данных приводится описание структуры данных всех классов, аналогично тому, как это сделано для таблиц, добавляется только описание методов.

Если в выпускной квалификационной работе разработаны и созданы инструменты для работы с БД, их можно описать в данном разделе.

**ЕСЛИ В ПРОЕКТЕ ПРЕДУСМОТРЕНО НЕКОЕ ДРУГОЕ ХРАНИЛИЩЕ ДАННЫХ** (*не бд, но, например, используется система каталогов*), **СЛЕДУЕТ ПОДРОБНО ОПИСАТЬ ЕГО СТРУКТУРУ**.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.4.2 Алгоритмы реализации используемых математических моделей

{**Привести разработанные алгоритмы расчета** по используемой математической модели.

Разработка мат.модели не входит в задачи ВКР, описание мат.постановки приводится в разделе Описание предметной области с указанием источника информации.

Разработка алгоритма расчета по модели - одна из задач ВКР. Алгоритм приводится **в виде текстового описания** - последовательность действий, выполняемых расчетной программой, можно при описании расчетного алгоритма **использовать диаграммы активности UML**.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.4.3 Алгоритмы использования применяемых программных технологий обработки данных

{При разработке сложных программных систем, использующих существующие программные технологии, основанные на мат.моделях и предназначенные для решения конкретных задач, следует **привести алгоритмы применения этих технологий** для решения задач ВКР.

*Например, разработка ПС с применением нейронных сетей включает следующий сценарий:*

*- выбор архитектуры нейронной сети,*

*- формирование датасета для обучения сети,*

*- обучение сети.*

*В данном случае, сама нейронная сеть не разрабатывается, применяется готовая программная технология, которая адаптируется под задачу. Алгоритм этой адаптации и следует описать.*}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.4.4 Алгоритмы применения методов графического анализа данных

{**Привести разработанные алгоритмы использования графичекого способа представления и интерпретации информации (*пошагово: какие данные выбираются для отображения? Какие способы визуализации являются предпочтительными для данной задачи (тема ВКР)? Какие выводы можно сделать на основе проведенного графического анализа? Привести примеры*)}**

### 4.4.5 Алгоритмы использования технологий передачи данных

{Описать, как в задаче ВКР реализована передача данных между компонентами программной системы (межмодульная передача данных); между разрабатываемой программной системой и внешними источниками (потребителями) данных.

(*например, может быть описано: роль общей базы данных в процессе передачи данных, виды запросов, протоколы передачи данных, технология организации локальных адресов, программно-алгоритмическое обеспечение связи «клиент-сервер» и т.д.*).

## 4.5 Описание архитектурного решения

### 4.5.1 Структурная организация программной системы

{В этом разделе следует показать организацию программной системы, структурных элементов и объединение этих элементов в более крупные подсистемы. Также можно показать, какое место разрабатываемое ПС (ПМ) занимает в более крупной системе.

Можно использовать статические UML диаграммы: диаграммы компонентов, диаграммы развертывания. С использованием **диаграммы компонентов UML** (она иллюстрирует архитектуры компонентов программного обеспечения и зависимости между ними) показать, как компоненты соединяются вместе для формирования программной системы. С помощью **диаграммы развертывания UML** показать, как программная система будет физически развернута на аппаратном обеспечении.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

### 4.5.2 Архитектура программного кода

{В этом разделе должна быть описана структура кода ПС, т. е.:

- из каких функциональных блоков (файлов, модулей, процедур, функций, классов) состоит ПС;

- приведено описание каждого блока с его названием и назначением;

- приведена графическая схема взаимосвязи этих блоков.

Для описания структуры кода можно использовать **диаграммы классов UML**.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

# 5 ТЕСТИРОВАНИЕ И ОПТИМИЗАЦИЯ

## 5.1 План тестирования

Для проверки качества работы мобильного приложения для записи данных с акселерометра выбраны пять типов тестирования: функциональное, юзабилити, нагрузочное, тестирование совместимости и тестирование **удобства**.

1. Функциональное тестирование

* Проверка работы основного функционала:
* Чтение данных с акселерометра в реальном времени.
* Сохранение данных в файл.
* Корректность отображения текущих показателей акселерометра.
* Тестирование стабильности работы при запуске/остановке записи.
* Проверка корректности обработки ошибок, например:
* Недостаточно места для сохранения данных.
* Сбой работы акселерометра.

1. Юзабилити тестирование

* Оценка интуитивности интерфейса:
* Простота начала и остановки записи данных.
* Понятность структуры меню и настроек.
* Проверка читаемости информации (значения акселерометра в реальном времени, сообщения об ошибках).

1. Нагрузочное тестирование

* Тестирование длительной записи данных (не менее 1 часа).
* Оценка производительности при высокой частоте записи данных (например, 100 Гц и выше).
* Анализ потребления ресурсов устройства:
* Уровень заряда батареи.
* Использование процессора и оперативной памяти.
* Проверка стабильности работы при одновременной работе других приложений.

1. Тестирование совместимости

* Проверка корректной работы приложения на различных версиях Android (например, начиная с Android 8.0 и выше).
* Тестирование на устройствах с различными характеристиками:  
   Различные производители и модели телефонов.
* Устройства с разным объемом оперативной памяти.
* Проверка совместимости с разными разрешениями экрана.

1. Тестирование удобства

* Оценка расположения и размера кнопок для управления записью данных.
* Проверка простоты настройки частоты записи и других параметров.
* Оценка логичности и доступности элементов интерфейса для пользователя.
* Тестирование сценариев с минимальными действиями пользователя (например, быстрый старт записи).

## 5.2 Результаты тестирования и оптимизация

{Описать процесс тестирования (пошагово, с подтверждением скриншотами), проведенного в соответствии с разработанным планом тестирования ПС. Оформить заключение о проведенном тестировании.

По результатам тестирования ПС может быть проведена техническая, алгоритмическая или программная оптимизация. Описать процесс оптимизации, подтверждая соответствующими скриншотами }

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

# 6 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

{В этот раздел могут быть включены следующие пункты:

- порядок установки и настройки программного средства на компьютер пользователя или первичная настройка технического средства;

- действия пользователя в случае сбоя ПС или ПТС;

- последовательность действий пользователя для решения своих основных задач при работе с программным или техническим средством.

Следует привести образы экранов (ScreenShots) с пояснениями для лучшего восприятия раздела. Приветствуется создание помощи по установке и эксплуатации ПС или ПТС.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

# ЗАКЛЮЧЕНИЕ

{Заключение – последовательное, логически стройное изложение полученных итогов и их соотношение с общей целью и конкретными задачами, поставленными и сформулированными во введении.

Заключение отражает оценку работы, подчеркивает актуальность и практическую её значимость, и включает рекомендации по практическому использованию ее результатов.

Заключение должно содержать краткие выводы по результатам ВКР.

Выводы формулируются по пунктам так, как они должны быть оглашены в конце доклада на защите.

- В этом разделе необходимо указать решена задача полностью или частично:

- подвести итоги проделанной работы – что сделано для решения поставленной задачи, это может быть:

* + разработанное и реализованное ПС или ПТС. Следует детализировать, что было сделано, покомпонентно, в соответствии с составленным планом разработки;
  + предложен новый подход (технология) к решению подобных задач;
  + предложен новый метод реализации ПС или ПТС.

Следует отметить возможные точки роста (развития) ПС или ПТС.

*Например:*

* + *добавить новые функции (блоки, режимы работы);*
  + *распространить на новый класс задач, другой тип данных.*

В Заключении следует:

* + привести объем созданного программного средства (в любых единицах, например, в килобайтах, строках кода, классах, процедурах, функциях ит.д.).
  + - отразить апробацию работы: выступления на семинарах, публикации, выступления на конференциях и конкурсах (полученные дипломы, грамоты и т.п.), акты о внедрении, реально действующий сайт (адрес) и т.д.}

Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст Текст

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

{Список опубликованных печатных, либо интернет- источников информации, используемых при написании текста ПЗ.

Сведения об источниках можно располагать в порядке появления ссылок на источники в тексте документа.

Источники необходимо нумеровать арабскими цифрами без точки и печатать с абзацного отступа.

Ссылки на использованные источники следует приводить в квадратных скобках в тексте, в конце предложения.

Примеры библиографического описания источников}

1. Численное исследование процессов переноса и трансформации газовых и аэрозольных примесей в шлейфе выбросов Норильского промышленного района / В.Ф. Рапута, В. Симоненков, Б.Д. Белан, Т.В. Ярославцева / «Оптика атмосферы и океана», 31, № 6, 2018 – C. 438-439.
2. Вострикова Л.Г. Финансовое право [Электронный ресурс] : учебник / Л.Г. Вострикова. – Электрон. текстовые дан. – М.: Равновесие: Юстинформ, 2005. – 1 электрон. опт. диск (СD). – Загл. с контейнера.
3. Исследовано в России: многопредмет. науч. журн. / Моск. физ.-техн. ин-т. – Электрон. журн. – Долгопрудный : МФТИ, 1998.– URL : http://zhurnal.mipt.rssi.ru, свободный. –Яз. рус. – (Дата обращ. ……..)
4. Мозолин В. П. Модернизация права собственности в экономическом измере-нии [Электронный ресурс] / В. П. Мозолин. – Электрон. ст. – М., 2011. – URL : http://www.norma-verlag.com/journal/2011/1, свободный. – Яз. рус. – Аналог печат. изд. (Журнал российского права. – 2011. – № 1). – (Дата обращ. 23.06.2011).

# ПРИЛОЖЕНИЯ

## Приложение А

**Техническое задание**

**Название темы:** Разработка приложения для сбора данных с акселерометра в системе Android применительно к измерению сейсмических процессов.

**Заказчик:** Ларичкин Алексей Юрьевич **Исполнитель:** Рыбалов Максим Сергеевич

**Функциональные требования**

1. Определение доступностиакселерометра: Приложение должно проверять наличие акселерометра на устройстве. При наличии акселерометра необходимо выводить его параметры, включая максимальную частоту сбора данных.
2. Настройка параметров сбора данных: Пользователь должен иметь возможность настраивать частоту сбора данных в диапазоне от 100 Гц до максимальной доступной на устройстве. Устанавливать интервал сбора данных (от 20 секунд до 3 минут). Задавать время и дату начала измерений. Запускать сбор данных в фоновом режиме.
3. Сохранение данных: Собранные данные должны сохраняться в виде текстового файла на устройстве. Формат файла: \*.csv, \*.txt или \*.xml. Структура данных: название устройства, его уникальный идентификатор (серийный номер или аналог), дата/время, значения акселерометра по осям X, Y, Z, координаты измерения.
4. Обработка данных: Приложение должно выполнять анализ данных с использованием преобразования Фурье для получения амплитуд и частот по осям X, Y, Z.
5. Графическое отображение: Визуализация данных акселерометра и спектрального анализа (АЧХ).
6. Дополнительные функции: Возможность задать название измерения. Передача файлов данных в социальные сети и по Bluetooth.

**Пользовательские требования**

1. **Интерфейс приложения:** Простой и интуитивно понятный интерфейс, с удобным отображением графиков данных и результатов анализа.
2. **Требования к устройству:** Устройство должно быть оснащено акселерометром и поддерживать операционную систему Android версии 8.0 (API Level 26) или выше.
3. **Языковая поддержка:** Приложение на русском языке (опционально добавить английский).

**Условия разработки**

1. **Среда разработки:** Android Studio (основная разработка).
2. **Технологии и библиотеки:** Bluetooth API для Android SDK. Java/Kotlin для основной разработки.

**Тестирование**

1. **Функциональное тестирование** Проверка корректности работы акселерометра и других используемых сенсоров. Тестирование настройки параметров сбора данных (частоты, интервала, времени начала). Проверка сохранения данных в текстовый файл и их дальнейшей обработки (включая преобразование Фурье).
2. **Пользовательское тестирование:** Оценка удобства интерфейса. Проверка выполнения всех пользовательских сценариев.
3. **Тестирование производительности:** Измерение времени обработки данных акселерометра (включая построение графиков и расчет АЧХ). Проверка стабильности работы приложения при длительном сборе данных и в фоновом режиме на устройствах с разными версиями Android.

**Сроки выполнения:** 1 мая 2025г.

**Подпись представителя:** \_\_\_\_\_\_\_\_\_\_\_\_\_

## Приложение В

{Содержит различные схемы, диаграммы, иллюстрирующие алгоритмы работы разработанной программной системы.}

## Приложение Г

{Содержит результаты тестовых испытаний на различных тестовых данных.

*Например, при наличии в составе ПС расчетно-графического модуля, полученные в результате работы, иллюстрации и таблицы, по которым можно судить о корректной работе программы*.}

## Приложение Д

{Содержит фрагменты листинга программного кода.

**Обязательно наличие поясняющего текста (для чего предназначен фрагмент кода, функция, класс)!**

Привести не менее 10 страниц кода. Текстовые пояснения оформляются в соответствии с ГОСТ.

Параметры оформления листинга кода: размер шрифта 10, допускается расположение в 2 столбца, одинарный междустрочный интервал..}

## Приложение Е

{При наличии, можно привести Акт о внедрении в эксплуатацию разработанной программной системы.}