

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего
образования «Самарский национальный исследовательский университет имени академика
С.П. Королева»
(Самарский университет)

Институт информатики, математики и электроники
Факультет информатики
Кафедра информационных систем и технологий

ОТЧЕТ ПО ПРАКТИКЕ

Вид практики производственная
(учебная, производственная)

Тип практики преддипломная
(в соответствии с ОПОП ВО)

Сроки прохождения практики: с 04.02.2021 по 17.05.2021
(в соответствии с календарным учебным графиком)

по направлению подготовки 09.04.01 Информатика и вычислительная техника
(уровень магистратуры)
направленность (профиль)
«Автоматизированные системы обработки информации
и управления»

Обучающийся группы № 6223-090401D _____ В.Д. Мавлютов

Руководитель практики,
доцент кафедры ИСТ, к.т.н., доцент _____ О.К. Головнин

Дата сдачи 21.05.2021
Дата защиты 21.05.2021

Оценка _____

Самара 2021

СОДЕРЖАНИЕ

1. Индивидуальное задание на практику.
2. Рабочий график (план) проведения практики.
3. Описательная часть.
4. Заключение.
5. Список использованных источников.
6. Приложение А Листинг программы

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение
высшего образования «Самарский национальный исследовательский университет
имени академика С.П. Королева»
(Самарский университет)

Институт информатики, математики и электроники
Факультет информатики
Кафедра информационных систем и технологий

Индивидуальное задание на практику

Обучающемуся Мавлютову Владимиру Дмитриевичу
группы 6223-090401D

Направление на практику оформлено приказом по университету
от 25.01.2021 № 47-ПР

на кафедру информационных систем и технологий

(наименование профильной организации или структурного подразделения университета)

Тема преддипломной практики: «Автоматизированная система навигации
внутри помещений с помощью инерциальной технологии локального
позиционирования мобильных устройств»

Планируемые результаты освоения образовательной программы (компетенции)	Планируемые результаты практики	Содержание задания
ОПК-1. Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Знать: методологии и технологии проектирования программного и (или) аппаратного обеспечения автоматизированных информационных систем, в том числе информационных систем, методологии теоретических и экспериментальных исследований эффективности	Разработать основные требования к проектируемому программному обеспечению автоматизированной информационной системы Бесплатформенная инерциальная навигация. Провести научные исследования эффективности программной реализации автоматизированной информационной системы Бесплатформенная

<p>ОПК-1.2. Выстраивает структуру теоретического и экспериментального исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте</p>	<p>программной реализации автоматизированной информационной системы. Уметь: применять методологии и технологии проектирования программного и (или) аппаратного обеспечения автоматизированных информационных систем, в том числе информационных интеллектуальных систем, методологии теоретических и экспериментальных исследований для исследования эффективности программной реализации автоматизированной информационной системы. Владеть: навыками проведения научных исследований эффективности программной реализации автоматизированной информационной системы с помощью методов теории вероятностей, математической статистики, искусственного интеллекта, теории оптимизации и</p>	<p>инерциальная навигация.с помощью методов теории вероятностей, математической статистики, искусственного интеллекта, теории оптимизации и вычислительных экспериментов.</p>
---	---	---

	вычислительных экспериментов.	
<p>ОПК-2. Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</p> <p>ОПК-2.2. Разрабатывает оригинальные программные средства, в том числе с использованием современных информационно-коммуникационных и интеллектуальных технологий, для решения профессиональных задач</p>	<p>Знать: современные языки, системы и технологии программирования, в том числе Web- и CALS-технологии и интеллектуальных технологии, для решения задач распознавания и обработки данных, цифровой обработки сигналов и задач управления объектами автоматизации.</p> <p>Уметь: применять современные языки, системы и технологии программирования, в том числе Web- и CALS-технологии и интеллектуальные технологии, для разработки алгоритмов и программ при решении задач распознавания и обработки данных, цифровой обработки сигналов и задач управления объектами автоматизации.</p> <p>Владеть: навыками разработки алгоритмов и программ с использованием современных языков, систем и технологий программирования, в том числе Web- и CALS-технологий и</p>	<p>Разработать алгоритмы и программную реализацию математической модели решения задач Бесплатформенной инерциальной навигации.с использованием современных языков и систем программирования.</p>

	интеллектуальных технологий, для решения задач распознавания и обработки данных, цифровой обработки сигналов и задач управления объектами автоматизации.	
ОПК-3. Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями ОПК-3.3. Осуществляет подготовку научных докладов, публикаций и аналитических обзоров с обоснованными выводами и рекомендациями	Знать: имеющиеся методологии анализа профессиональной информации и технологии подготовки научных докладов, публикаций и аналитических обзоров. Уметь: применять имеющиеся методологии анализа профессиональной информации и технологии подготовки научных докладов, публикаций и аналитических обзоров с обоснованием выводов и рекомендаций. Владеть: навыками разработки аналитических отчётов на основе анализа профессиональной информации для подготовки научных докладов, публикаций и аналитических обзоров, а также навыками обоснования выводов и рекомендаций.	Провести анализ имеющихся методологий анализа профессиональной информации и технологии подготовки научных докладов, публикаций и аналитических обзоров. Разработать аналитический обзор с обоснованием выбора технологии проектирования и технологий программирования для решения задачи Бесплатформенной инерциальной навигации.

<p>ОПК-4. Способен применять на практике новые научные принципы и методы исследований</p> <p>ОПК-4.2. Применяет новые научные принципы и методы исследования для решения профессиональных задач</p>	<p>Знать: методы моделирования на основе теории вероятностей, математической статистики, искусственного интеллекта и теории оптимизации, а также методы проведения научных исследований эффективности программной реализации модели с помощью вычислительных экспериментов для решения профессиональных задач.</p> <p>Уметь: применять методы моделирования с использованием теории вероятностей, математической статистики, искусственного интеллекта и теории оптимизации для разработки математической модели, а также методы проведения научных исследований эффективности программной реализации модели с помощью вычислительных экспериментов, в профессиональных задачах.</p> <p>Владеть: навыками проведения научных</p>	<p>Разработать математическую модель для решения задачи Бесплатформенной инерциальной навигации с использованием методов теории вероятностей, математической статистики, искусственного интеллекта и методов оптимизации.</p>
---	--	---

	исследований с помощью методов теории вероятностей, математической статистики, искусственного интеллекта, теории оптимизации и вычислительного эксперимента.	
ОПК-5. Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем ОПК-5.2. Осуществляет модернизацию программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач	Знать: технологии проектирования программного и (или) аппаратного обеспечения автоматизированных информационных систем. Уметь: применять технологии проектирования программного и (или) аппаратного обеспечения автоматизированных информационных систем для их модернизации. Владеть: навыками модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач.	Провести анализ имеющихся современных языков, систем и технологий программирования, а также технологий проектирования программного обеспечения для решения задач Бесплатформенной инерциальной навигации Разработать программное и аппаратное обеспечение автоматизированной информационной системы Бесплатформенная инерциальная навигация.
ОПК-6. Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации	Знать: методы разработки технической документации по использованию и настройке компонентов	Разработать техническую документацию по использованию и настройке компонентов программно-аппаратного комплекса обработки информации

и автоматизированного проектирования ОПК-6.3. Планирует составление технической документации по использованию и настройке компонентов программно-аппаратного комплекса	программно-аппаратного комплекса обработки информации. Уметь: применять методы разработки технической документации по использованию и настройке компонентов программно-аппаратного комплекса обработки информации. Владеть: навыками разработки технической документации по использованию и настройке компонентов программно-аппаратного комплекса обработки информации.	Бесплатформенная инерциальная навигация.
---	--	--

Дата выдачи задания 04.02.2021.

Срок представления на кафедру отчета о практике 21.05.2021.

Обучающийся группы № 6223-090401D _____ В.Д. Мавлютов

Руководитель практики,
Доцент кафедры ИСТ, к.т.н. _____ О.К. Головин

Рабочий график (план) проведения практики

Дата (период)	Содержание задания	Результаты практики
04.02.2021- 05.02.2021	Оформить индивидуальное задание на практику.	Оформлено и утверждено индивидуальное задание на практику и рабочий график.
06.02.2021- 10.02.2021	Провести анализ имеющихся современных языков, систем и технологий программирования, а также технологий проектирования программного обеспечения для решения задач Бесплатформенной инерциальной навигации.	Обучающийся овладел навыками проведения анализа имеющихся современных языков, систем и технологий программирования, а также технологий проектирования программного обеспечения для решения задач обработки информации и автоматизированного проектирования.
11.02.2021- 15.02.2021	Провести анализ имеющихся методологий анализа профессиональной информации и технологии подготовки научных докладов, публикаций и аналитических обзоров.	Обучающийся овладел навыками проведения анализа имеющихся методологий анализа профессиональной информации и технологии подготовки научных докладов, публикаций и аналитических обзоров
16.02.2021- 20.02.2021	Разработать аналитический обзор с обоснованием выбора технологии проектирования и технологий программирования для решения задачи Бесплатформенной инерциальной навигации.	Обучающийся овладел навыками проведения разработки аналитических обзоров и обоснования выбора технологии проектирования и технологий программирования для решения задачи управления объектом автоматизации или для разрабатываемой информационной системы
22.02.2021- 06.03.2021	Разработать математическую модель для решения задачи Бесплатформенной инерциальной навигации.с использованием методов теории вероятностей, математической статистики, искусственного	Обучающийся овладел навыками разработки математической модели для решения профессиональной задачи с использованием методов теории вероятностей, математической статистики,

	интеллекта и методов оптимизации.	искусственного интеллекта и методов оптимизации.
09.03.2021-15.03.2021	Разработать основные требования к проектируемому программному обеспечению автоматизированной информационной системы Бесплатформенная инерциальная навигация.	Обучающийся овладел навыками разработки основных требований к проектируемому программному обеспечению автоматизированной информационной системы, в том числе информационной интеллектуальной системы.
16.03.2021-07.04.2021	Разработать алгоритмы и программную реализацию математической модели решения задач Бесплатформенной инерциальной навигации с использованием современных языков и систем программирования.	Обучающийся овладел навыками разработки алгоритмов и программной реализации математической модели решения задач управления объектом автоматизации с использованием современных языков и систем программирования.
08.04.2021-29.04.2021	Разработать программное и аппаратное обеспечение автоматизированной информационной системы Бесплатформенная инерциальная навигация.	Обучающийся овладел навыками разработки программного и аппаратного обеспечения автоматизированной информационной системы.
30.04.2021-13.05.2021	Провести научные исследования эффективности программной реализации автоматизированной информационной системы Бесплатформенная инерциальная навигация с помощью методов теории вероятностей, математической статистики, искусственного интеллекта, теории оптимизации и вычислительных экспериментов.	Обучающийся овладел навыками проведения научных исследований эффективности программной реализации автоматизированной информационной системы с помощью методов теории вероятностей, математической статистики, искусственного интеллекта, теории оптимизации и вычислительных экспериментов.
14.05.2021-18.05.2021	Разработать техническую документацию по использованию и настройке компонентов программно-	Обучающийся овладел навыками разработки технической документации по использованию и настройке

	аппаратного комплекса обработки информации Бесплатформенная инерциальная навигация.	компонентов программно-аппаратного комплекса обработки информации.
19.05.2021-20.05.2021	Оформить письменный отчёт по практике.	Обучающийся применил полученные навыки для разработки письменного отчёта по практике. Обучающимся был оформлен и предъявлен руководителю письменный отчет по практике.
21.05.2021	Подготовить устный отчет о прохождении практики.	Подготовлен устный отчет о прохождении практики.
21.05.2021	Защитить отчет по практике	Получен зачёт по практике.

Руководитель практики,
Доцент кафедры ИСТ, к.т.н. _____ О.К. Головнин
(подпись)

Описательная часть

1 АНАЛИЗ ИССЛЕДУЕМОЙ ЗАДАЧИ И НАПРАВЛЕНИЯ ЕЕ РЕШЕНИЯ В ТЕХНОЛОГИЯХ

1.1 Анализ современных языков

Технология программирования не стоит на месте и постоянно развивается с учетом новых потребностей в сфере IT-технологий. Существуют два главных требования для языков программирования: их близость к обычному профессиональному языку определенной сферы деятельности и простота реализации. Противоречивость требований стимулирует непрерывный процесс создания новых языков и доработки существующих.

Язык программирования – система обозначений для описания алгоритмов и структур данных, определенная искусственная формальная система, средствами которой можно выражать алгоритмы [1].

В настоящее время ежегодно число языков программирования пополняется новыми. Ниже рассматриваются несколько самых передовых языков программирования, каждый из которых решает свою определенную работу.

Язык Dart, разработанный компанией Google, как один из языков веб-программирования. Как и в JavaScript, в Dart используются синтаксисы и ключевые слова, похожие на те, которые используются в языке C. Однако одним существенным различием является то, что в то время, как JavaScript основывается на прототипах, объекты в Dart определяются с помощью классов и интерфейсов, как в C++ или Java [2].

Веб-программирование сложно тем, что для клиента, сервера, баз данных и т. д. свои отдельные языки программирования. Инженеры разработали новый язык Ора. Этот язык, созданный с идеей того, чтобы связать в себя интерфейс пользователя, логику сервера и базу данных ввода-вывода. Компилятор самостоятельно решает, где запущена программа и выполняет написанный код. Среда выполнения Ора объединяет собственный веб-сервер и систему управления базой данных, которые не могут быть заменены самостоятельными

альтернативами. Ора поставляется бесплатно и на данный момент доступен для 64-х битных Linux и Mac OS X платформ, другие же порты пока разрабатываются [2].

С другой стороны, в каждом языке существуют свои недостатки, которые ставят созданную на этом языке систему менее безопасной. Для решения этой проблемы ученые нашли способ защиты такой, что позволяет защитить и веб-сайты, и компьютеры пользователей, разработав систему позволяющую использовать несколько языков программирования при создании одной программы.

Система Wyvern является чем-то вроде компилятора одного мета-языка, в состав которого входят все известные и популярные языки программирования. Эта система позволяет программистам разрабатывать веб-страницы и приложения, используя напрямую все возможности C, PHP, HTML, CSS, JavaScript и других языков, и избегая необходимости использования некоторых искусственных приемов, которые являются источниками потенциальной опасности [3].

Данная система не единственная, примерами подобных программ являются – Scheme, ProteaJ, Spoofax и OJ. Они также решают вышеперечисленные задачи.

В продолжение темы, был разработан новый язык программирования под названием Sketch. Особенность языка в том, что он позволяет при написании программ опускать некоторые участки кода, заполняя эти промежутки самостоятельно при этом оптимизируя получившиеся конструкции.

Основной идеей, реализованной в виде компилятора языка Sketch, является метод поиска оптимального решения определенной задачи. Известно, что любая логическая и математическая задача может быть решена несколькими путями, количество которых, в некоторых случаях, может стремиться к бесконечности [4].

Рассмотрев современные языки программирования, некоторые из которых внедряются в программирования, некоторые еще дорабатываются

можно сделать выводы о том, что в настоящее время, имея базу из двух с половиной тысяч языков программирования нет ни одного универсального. Пока такая задача не стоит, поэтому на этом этапе можно выделить систему Wyvern, система, позволяющую использовать несколько разных языков программирования при создании одной программы. Можно предположить, что хотя бы один из рассмотренных языков заменит в будущем популярных в настоящее время языков программирования.

1.2 Анализ современных популярных сред разработки для Android

Среда разработки Android – это внедрение комплексных программных средств с целью создания ПО для гаджета (сокращенная аббревиатура «IDE»). В некоторых случаях может вмещать в себе элементы интегрирования с управленческими системами и разный инструментарий, упрощающий построение графического изображения с нуля.

На сегодняшний день самыми востребованными средами являются:

- Eclipse;
- IntelliJ Idea;
- Android Studio.

Каждая IDE обладает уникальными характеристиками.

Eclipse

Это бесплатная IDE, разработанная некоммерческой компанией Eclipse Foundation. Эта программа (рисунок 1) является базой, которая регулирует процессы создания приложений. Преимущества Эклипс:

- интерфейс переведен на грамотный русский язык (документация прилагается);
- отлично «гоняет» на компьютерах с низкой производительностью;
- имеет дополнительные функции (для серверной работы и анализа базы данных);
- может подключаться к модулям;
- может работать в групповом режиме (когда проект создают несколько человек одновременно).

Эклипс стала популярной несколько лет назад и по-прежнему занимает лидирующие позиции. Хотя после выхода Android Studio (2014 год) Google решил перевести сотрудничество с Eclipse на «второй план».

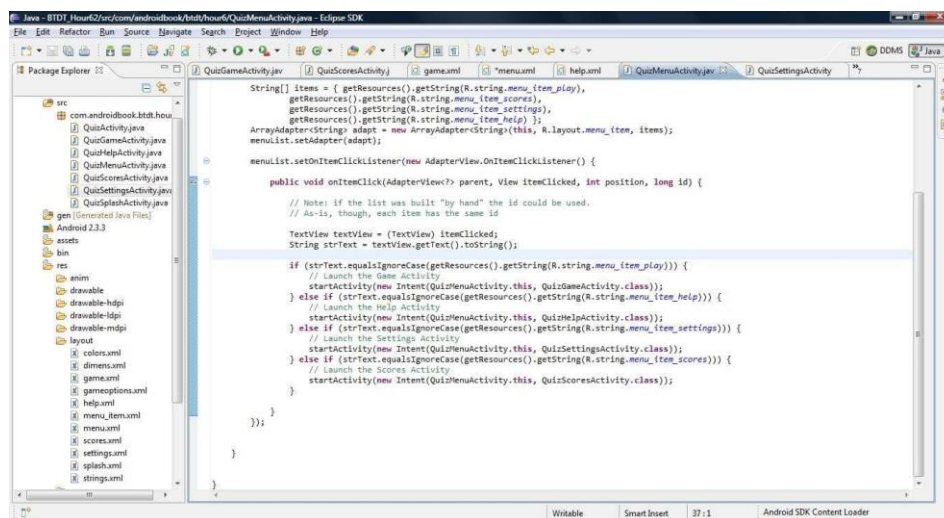


Рисунок 1 – Интерфейс Eclipse

IntelliJ Idea

Данную программу разработала российская компания JetBrains. Интерфейс программы представлен на рисунке 2. Подобно Эклипс, эта среда позволяет создавать приложения и программы на нескольких программных языках. Также IntelliJ Idea не перегружает ПК. Чем она примечательна:

- более оперативная отладка значений;
- предусмотрен автозаполнитель методов;
- есть рефакторинг;
- интерфейс более понятный и лаконичный;
- подходит для тех, кто программирует на Java.

Единственный недостаток – за IntelliJ Idea нужно платить.

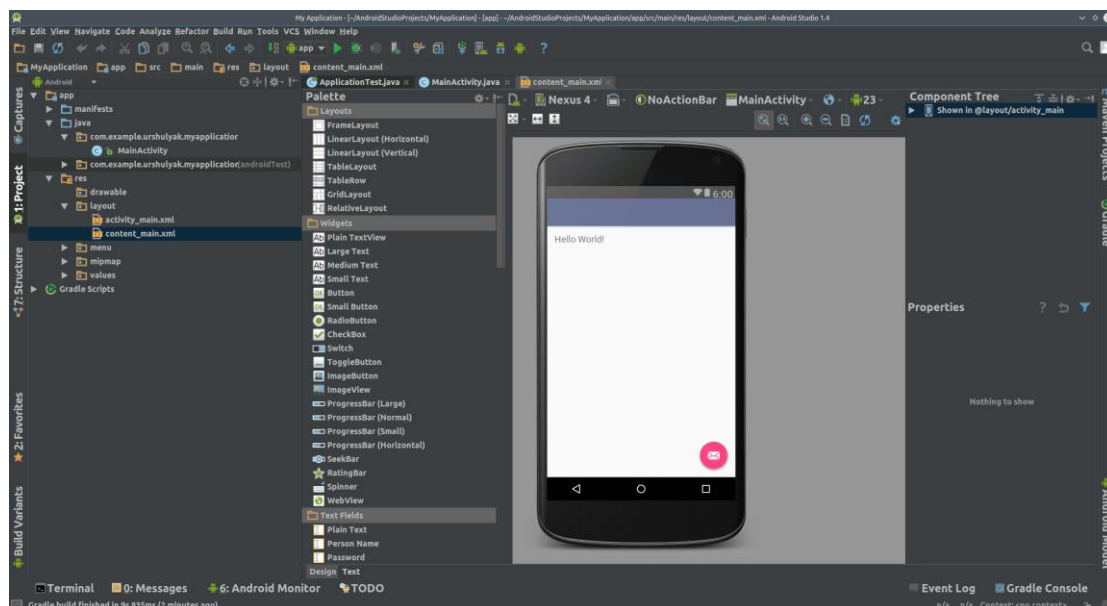


Рисунок 2 – Интерфейс IntelliJ Idea

Android Studio

Google никогда не стоит на месте, и совсем недавно компания принялась за разработку авторской среды для Андроид. Она создала собственную IDE на базе IntelliJ Idea.

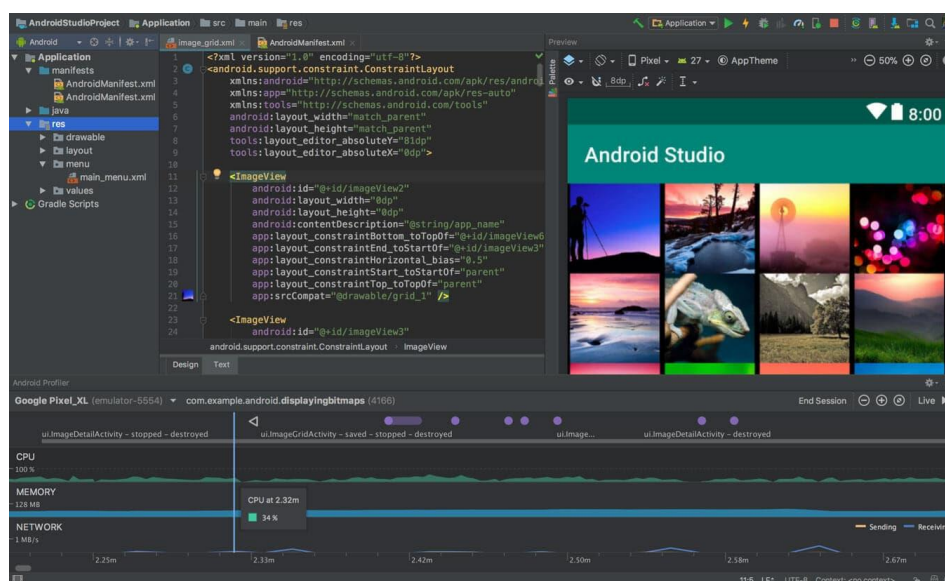


Рисунок 3 – Интерфейс Android Studio

Принципиальных отличий между программами нет, так как нововведений здесь немного. Тем не менее, имя разработчика позволило программе стать достойным конкурентом двум предыдущим IDE за считанные месяцы. Единственный недостаток программы: недостаточное количество функций персонализации в кодовом редакторе и общих настройках [5].

1.3 Выбор языка программирования

Java — один из самых популярных языков программирования. Сейчас на нём работают примерно 9 миллионов человек, а сами Java-приложения работают на семи миллиардах устройств. Этот язык универсален и используется повсюду: мобильные приложения и игры, научные разработки и мобильные телефоны, игровые консоли и серьёзные суперкомпьютеры. Java — это основа практически для всех типов сетевых приложений. Многие исследования называют его языком программирования №1 среди разработчиков. Преимущества у Java:

- Кроссплатформенность;
- Надёжность;
- Объектно-ориентированность;
- Относительная простота;
- Гибкость.

1.4 Выбор среды разработки

Лучший способ разработать приложение для Android — это Android Studio. Android Studio — это интегрированная среда разработки (IDE) для работы с платформой Android.

Android Studio, основанная на программном обеспечении IntelliJ IDEA от компании JetBrains, — официальное средство разработки Android приложений.

Android Studio это часть программного обеспечения, называемая IDE, или интегрированной средой разработки. Он предлагается в виде пакета с Android SDK, который представляет собой не что иное, как набор инструментов, используемых для облегчения разработки Android. Здесь сосредоточено все, что требуется, чтобы начать создавать приложения под Android. Такие функции, как визуальный конструктор, делают процесс более плавным, в то время как расширенные, мощные функции добавляются все время, чтобы предоставить разработчикам доступ к таким вещам, как облачное хранилище.

1.5 Выбор среды проектирования

Для успешной реализации проекта информационная система должна быть, прежде всего, адекватно описана, построены полные и непротиворечивые функциональные и информационные модели информационной системы [6]. Накопленный к настоящему времени опыт проектирования информационных систем показывает, что это логически сложная, трудоемкая и длительная по времени работа, требующая высокой квалификации участвующих в ней специалистов. Поэтому этап проектирования считается важнейшим и сложнейшим этапом жизненного цикла информационной системы.

Чтобы облегчить труд проектировщика, были созданы CASE-средства - программы специального вида. CASE-средства (Computer Aided Software/System Engineering) – позволяют проектировать любые системы на компьютере. CASE-средства – это необходимые элементы системного и структурно-функционального анализа, позволяющие моделировать бизнес-процессы, базы данных, компоненты программного обеспечения, деятельность и структуру организаций.

Visual Paradigm — компания-разработчик программных решений для помощи разработчикам, основная цель программных решений от компании Visual Paradigm — это помощь разработчикам в создании приложений, путем предоставления необходимых инструментов для ускорения работы.

Visual Paradigm выпускает именно то программное обеспечение, которое помогает разработчикам транслировать требования к разрабатываемому программному обеспечению в качественное программное обеспечение, при этом соблюдать высокое качество работы, при минимальных сроках.

2 РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ

Существует несколько вариантов получения информации для построения пути: с помощью данных с акселерометра, магнитометра или гироскопа. Таким образом можно получить различные множества данных.

Так, с помощью акселерометра можно получить ускорение по трем осям. В итоге получается множество данных акселерометра:

$$G = \{\langle g(X), g(Y), g(Z) \rangle\}, \quad (1)$$

где $g_i \in G$ – это точка одного измерения в пространстве.

С помощью гироскопа можно получить гироскопический момент вращения ротора, возникающий при изменении направления оси. Он определяется следующим множеством:

$$M = \{\langle m(X), m(Y), m(Z) \rangle\}, \quad (2)$$

где M – данные с гироскопа, а именно угловая скорость относительно оси симметрии.

$m_i \in M$, где m_i – точка одного измерения для определения пространства.

Для определения магнитных полей используется магнитометр. В итоге получается множество данных:

$$T = \{\langle t(X), t(Y), t(Z) \rangle\} \quad (3)$$

где T – это данные с гироскопа, а именно индукция магнитного поля Земли в определённой точке;

$t_i \in T$ – точка одного измерения для определения пространств.

Результат обработки данных должен выглядеть как множество точек на карте, из которых строится маршрут. Так, например: $p_i = (lt, lg)$ – это координаты точки на карте, где lt – это широта, а lg – долгота. P – это множество точек, из которых строится маршрут.

Из физики известно, что ускорение является первой производной от скорости, то есть характеризует быстроту ее изменения. Соответственно, скорость – это первая производная расстояния. Операцией, обратной

дифференцированию (взятию производной), является интегрирование. Следовательно, если значение производной (измеренное ускорение) известно, то после его интегрирования получим скорость, а после интегрирования скорости получим пройденное расстояние.

Пусть a_N и a_E – измеренные ускорения по направлениям на север и восток, W_N и W_E – составляющие путевой скорости, S_N и S_E – пройденные расстояния по этим же направлениям. Тогда:

$$W_N = \int_0^t a_N dt; \quad W_E = \int_0^t a_E dt; \quad (4)$$

$$S_N = \int_0^t W_N dt; \quad S_E = \int_0^t W_E dt. \quad (5)$$

Современные ИНС осуществляют счисление в географической системе координат, то есть определяют широту и долготу. Если принять Землю за сферу, то текущие широта φ и долгота λ (в радианах) могут быть определены как:

$$lt = lt_0 + \frac{1}{R} \int_0^t W_N dt; \quad lg = lg_0 + \frac{1}{R} \int_0^t \frac{W_N}{\cos(lt)} dt, \quad (6)$$

где R – радиус Земли,

lt_0, lg_0 – начальные координаты, получаемые с магнитометра.

Все расчеты выполняются цифровыми вычислителями, а именно процессором телефона.

3 РАЗРАБОТКА ОСНОВНЫХ ТРЕБОВАНИЙ К ПРОЕКТИРУЕМОМУ ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

3.1 Диаграмма вариантов использования

Моделирование динамического поведения сложной технической системы начинается с разработки диаграммы вариантов использования. Диаграмма вариантов использования (use case diagram) дает наиболее общее представление

функционального назначения системы. Каждый блок из вариантов использования определяет последовательность действий, которые должны быть выполнены. База проектируемой системой при взаимодействии системы с соответствующим актером [7].

Варианты использования представляют собой средство для спецификации требований к системе. Обычно описывается то, что должно происходить в проектируемой системе, какие функции в ней будут поддерживаться. Требуемое поведение системы или субъекта специфицируется одним или несколькими вариантами использования, которые определяются в соответствии с потребностями актеров.

На практике при разработке программных систем рекомендуется придерживаться следующего правила – отдельному варианту использования должно соответствовать некоторое требование к функциональному поведению моделируемой системы [8].

Вовремя проектирование системы инерциальной навигации была разработана диаграмма вариантов использования, которая приведена на рисунке 4.

В системе присутствует авторизация пользователей.

Пользователь может просматривать карту (масштабировать, перемещать, крутить). В зависимости от надобности можно открыть список уже пройденных маршрутов, а затем его посмотреть, где он был пройден, а также информацию о нем. Авторизированный пользователь системы есть возможность найти маршрут до определенного ему места. Для это нужно ввести сначала точку начала, а затем точку окончания.

3.2 Диаграмма классов

Базовыми элементами автоматизированной системы являются классы. Класс – это шаблон, который описывает множество однотипных объектов, имеющих определенный набор параметров и характеристик – атрибуты, операции, отношения и семантику.

В нотации UML классы системы отображаются на диаграмме классов.

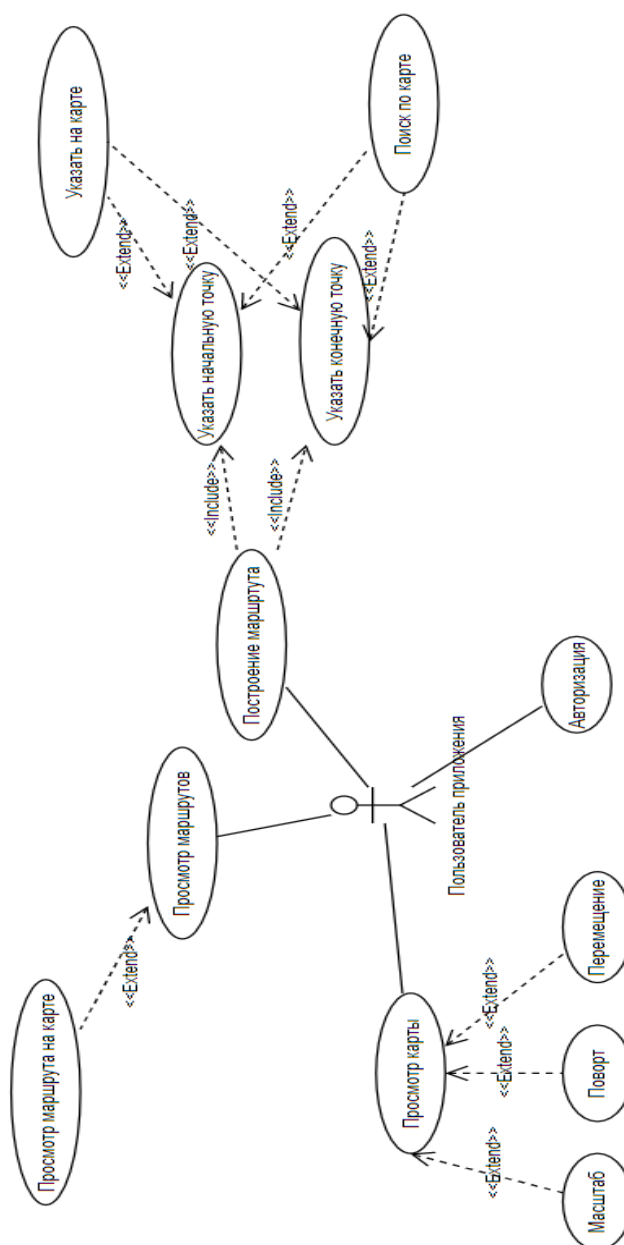


Рисунок 4 – Диаграмма вариантов использования

Диаграмма классов представляет собой набор классов, интерфейсов, их атрибутов и методов, а также взаимосвязей между этими классами [9]. Данная диаграмма предназначена для определения типов объектов разрабатываемой системы и для описания их отношений.

Диаграмма классов системы представлена на рисунке 5.

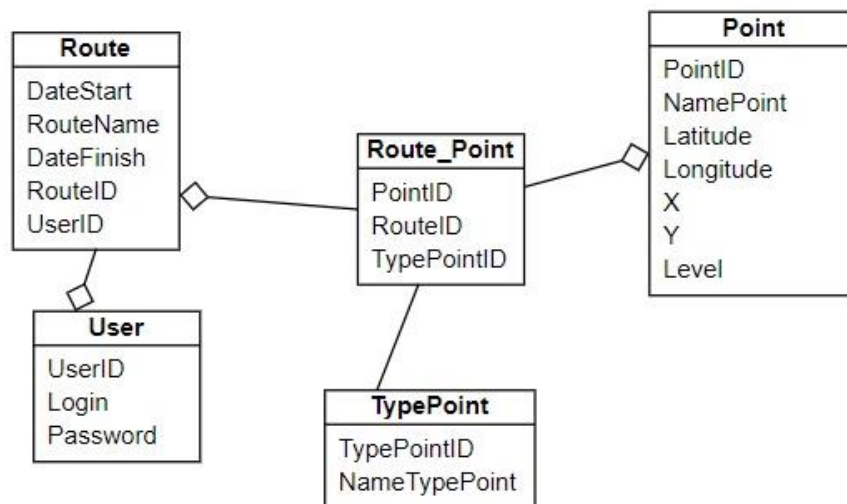


Рисунок 5 – Диаграмма классов

Наиболее важными среди связей диаграммы классов являются следующие:

- зависимость – отношение использования, при котором изменение объекта одного класса влияет на использующий его объект другого класса;
- обобщение, представляющее собой отношение между родительской сущностью и ее потомком;
- ассоциация, показывающая какую-либо связь между объектами различных классов;
- агрегация – это разновидность ассоциации, представляющая собой отношение типа «часть-целое»;
- композиция, являющаяся типом агрегации, при которой объект-часть принадлежит только единственному целому.

3.3 Диаграмма деятельности

При моделировании поведения проектируемой или анализируемой системы возникает необходимость не только представить процесс изменения ее состояний, но и детализировать особенности алгоритмической и логической реализации выполняемых системой операций.

Для моделирования процесса выполнения операций в языке UML используются диаграммы деятельности. Применяемая в них графическая нотация во многом похожа на нотацию диаграммы состояний, поскольку на этих диаграммах также присутствуют обозначения состояний и переходов. Каждое состояние на диаграмме деятельности соответствует выполнению некоторой элементарной операции, а переход в следующее состояние выполняется только при завершении этой операции.

На рисунке 6 изображена диаграмма деятельности системы.

3.4 Диаграмма развертывания

Диаграмма развертывания также является примером физического представления системы. Однако отличие данной диаграммы в том, что она показывает распределенную программную среду, в которой существуют компонентные узлы и их взаимодействие между собой. Диаграмма содержит графическое представление процессов, устройств, связей между ними [10]. Диаграмма развертывания системы представлена на рисунке 7.

3.5 Разработка логической модели данных

Логическая модель базы данных является моделью базы данных системы, представленной в виде совокупности сущностей, атрибутов и связей. Логическая модель позволяет графически представить структуру базы данных, которая не зависит от аппаратной платформы и конечной реализации базы данных.

В процессе проектирования системы была разработана логическая модель базы данных, для описания которой была выбрана методология IDEF1X. На рисунке 8 представлена логическая модель, содержащая основные сущности базы данных системы.

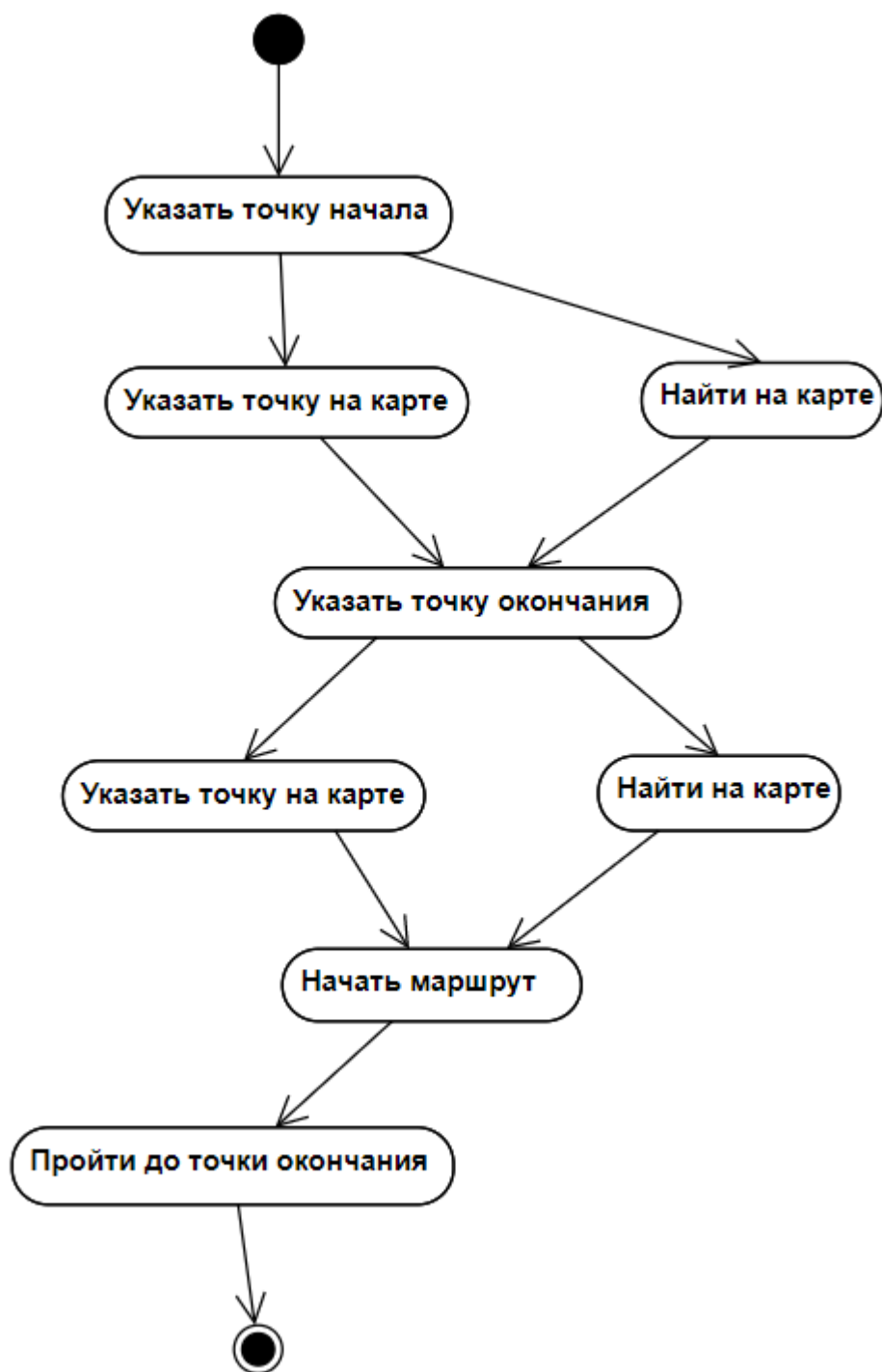


Рисунок 6 – Диаграмма деятельности

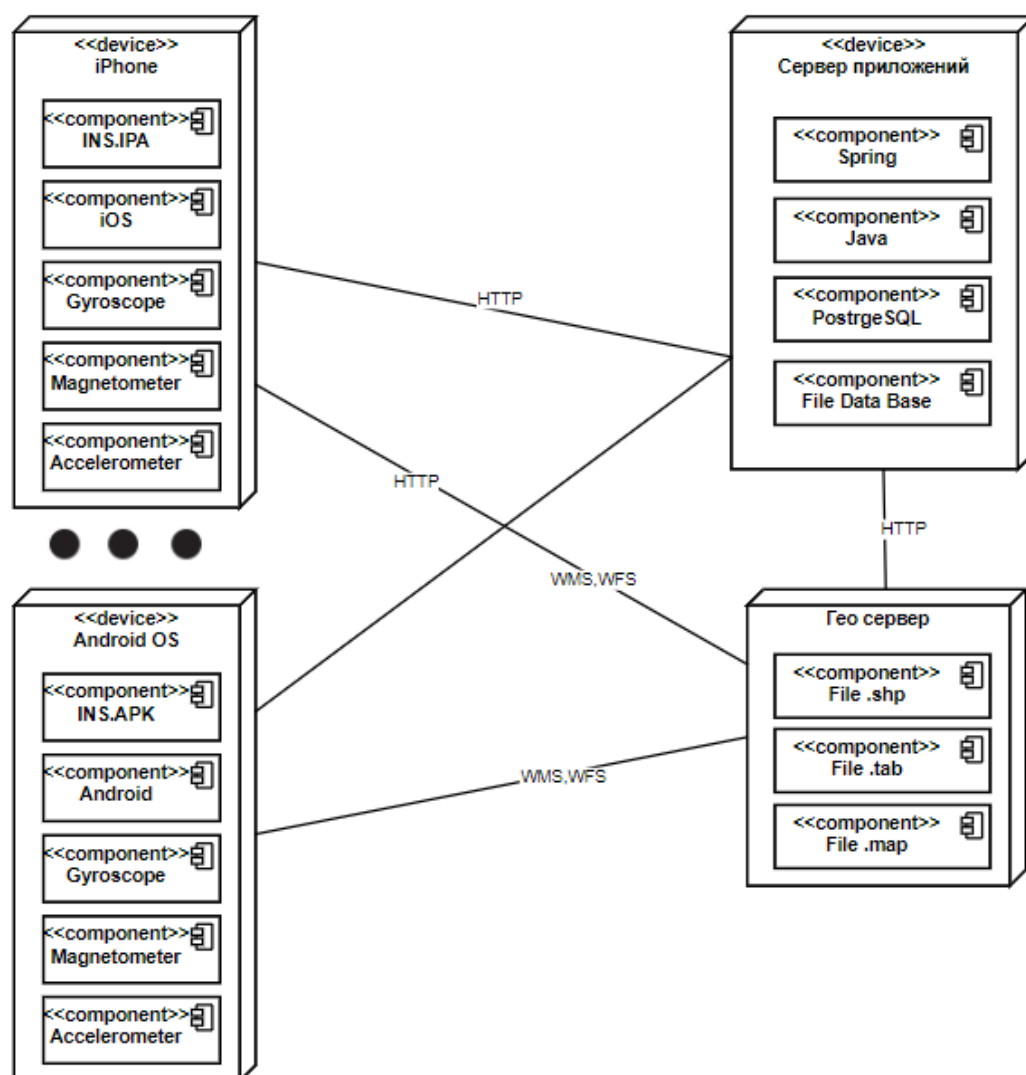


Рисунок 7 – Диаграмма развертывания

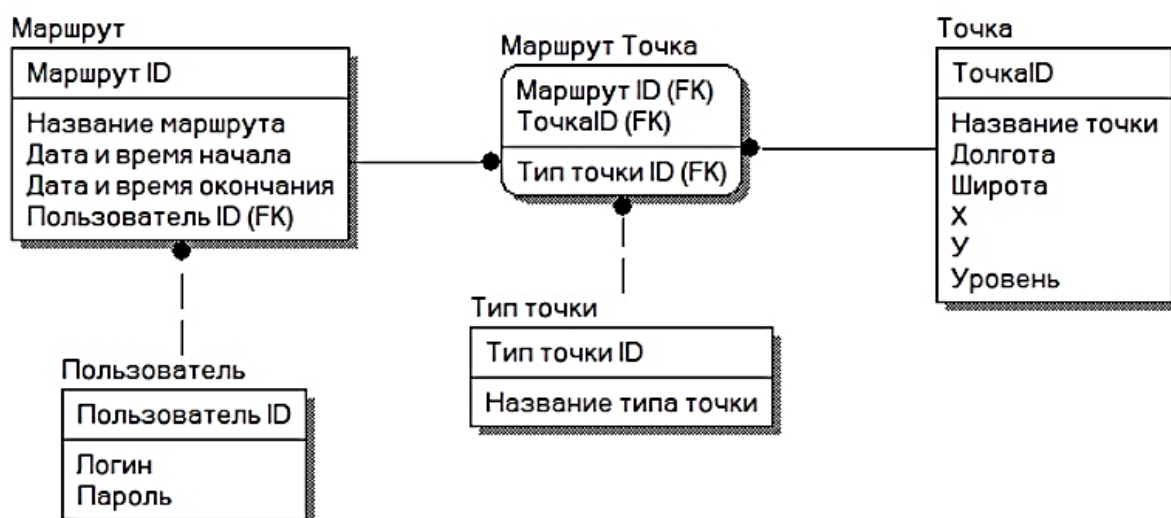


Рисунок 8 – Основные сущности логической модели данных

3.6 Разработка физической модели данных

Под физической моделью базы данных понимают модель, определяющую способы размещения данных в среде хранения и способы доступа к этим данным.

Физическая модель позволяет описывать все детали, которые необходимы для создания базы данных в конкретной СУБД: названия таблиц и столбцов, определения первичных и внешних ключей типы данных, и т.д.

Физическая модель базы данных, представленная на рисунке 9, построена на основе логической модели.

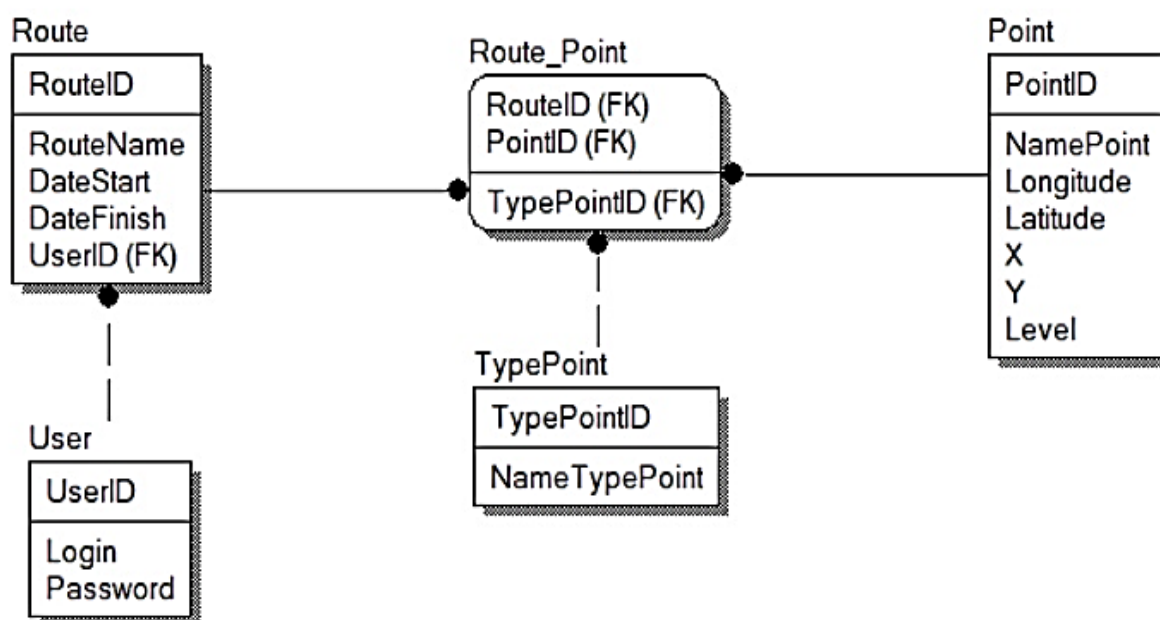


Рисунок 9 – Физическая модель данных

4 РАЗРАБОТКА АЛГОРИТМОВ

4.1 Алгоритм получения данных с помощью акселерометра

Данный алгоритм (рисунок 10) подписывается на событие обновления данных сенсоров, получает информацию о ускорении, преобразовывает в мировые координаты, а также высчитывает пройденную дистанцию. Эти данные накапливаются до момента вызова метода интерфейса `getChanges`, после которого данные обнуляются.

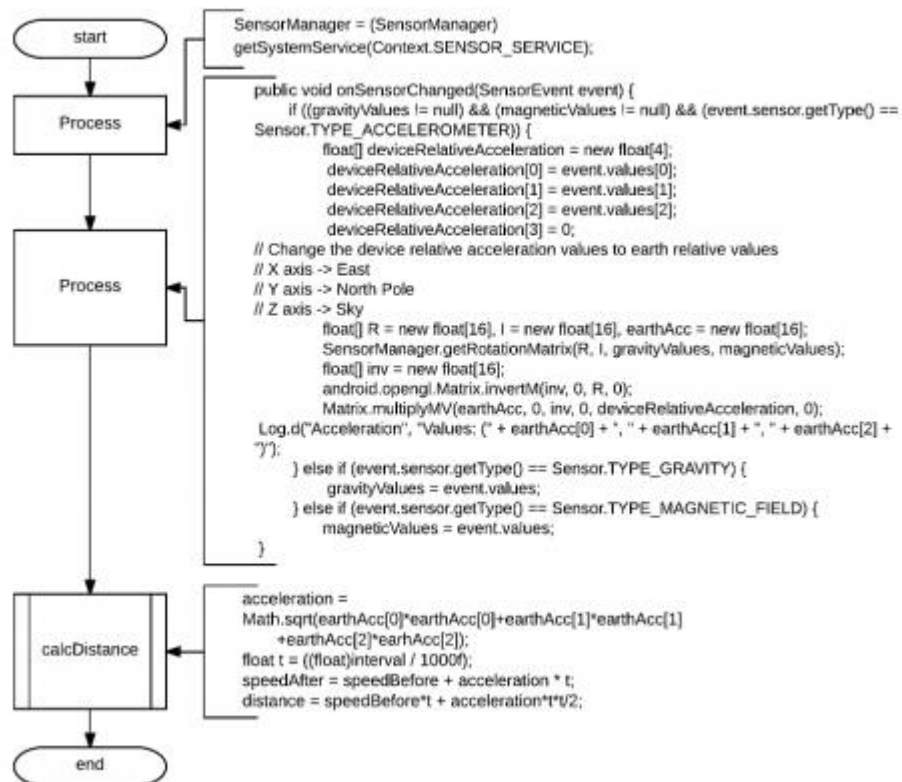


Рисунок 10 – Алгоритм сбора информации с акселерометра

4.2 Алгоритм работы Фильтра Кальмана

Данный алгоритм реализует фильтрацию Калмана (рисунок 11). В конструкторе помимо инициализации также задаются параметры ковариации шума процесса и ковариации шума измерения. С их помощью можно задавать требуемый уровень сглаживания они определяются экспериментально. Матрицу перехода определяем, как единичную (новое состояние равно предыдущему). Матрица отношений измерений и состояний также будет единичной, так как данное преобразование не требуется. Инициализация модуля происходит на старте системы. Дальнейшая работа выполняется по вызову метода `nextStep` из модуля трилатерации. После обработки полученные координаты в виде события отправляются классу уровня представления, отвечающему за вывод на дисплей.

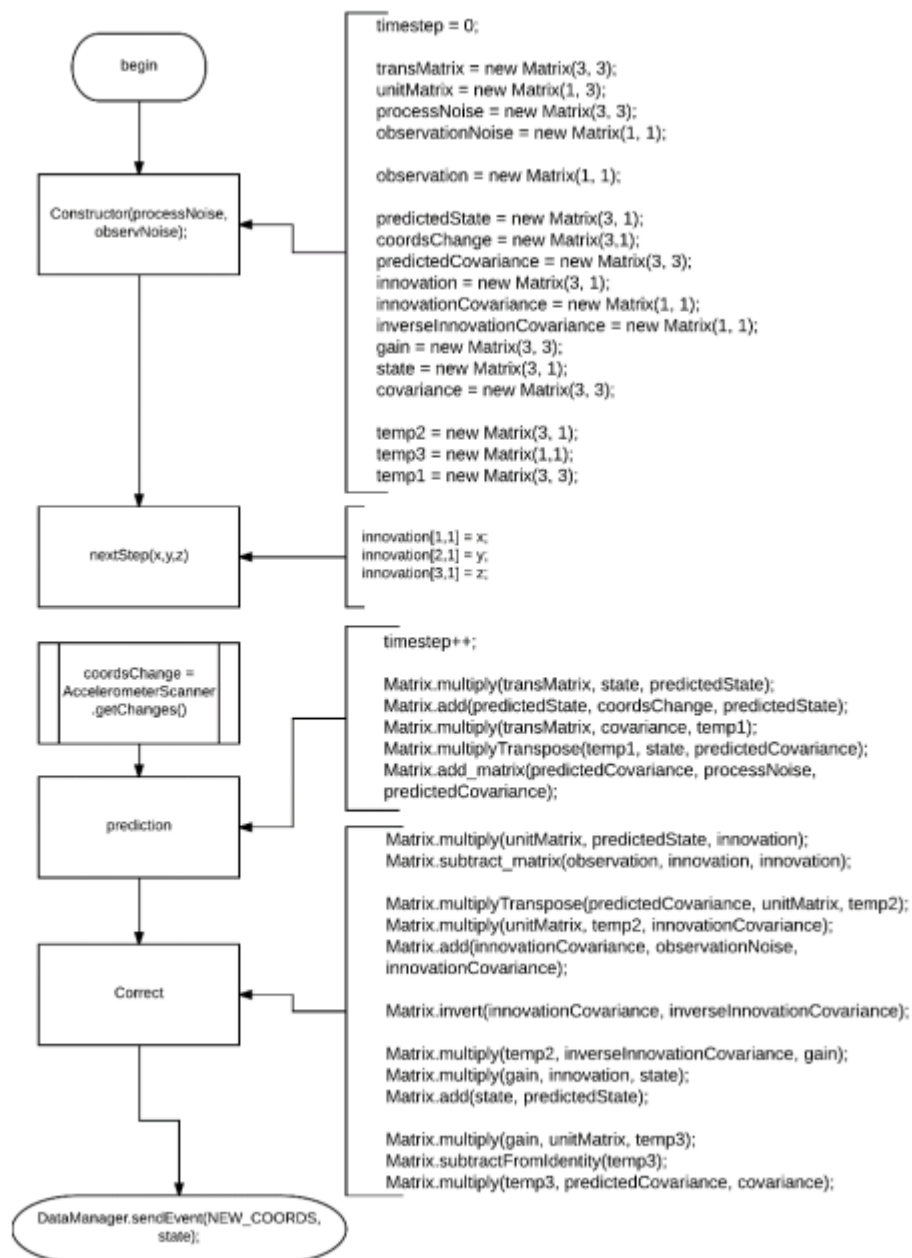


Рисунок 11 – Алгоритм, описывающий работу фильтра Калмана

4.3 Алгоритм А*

На рисунке 12 представлена схема алгоритма, соответствующего поиску кратчайшего маршрута – А*. Поскольку эвристика не оценивает расстояния повторно, А не использует эвристику для поиска подходящего ответа.

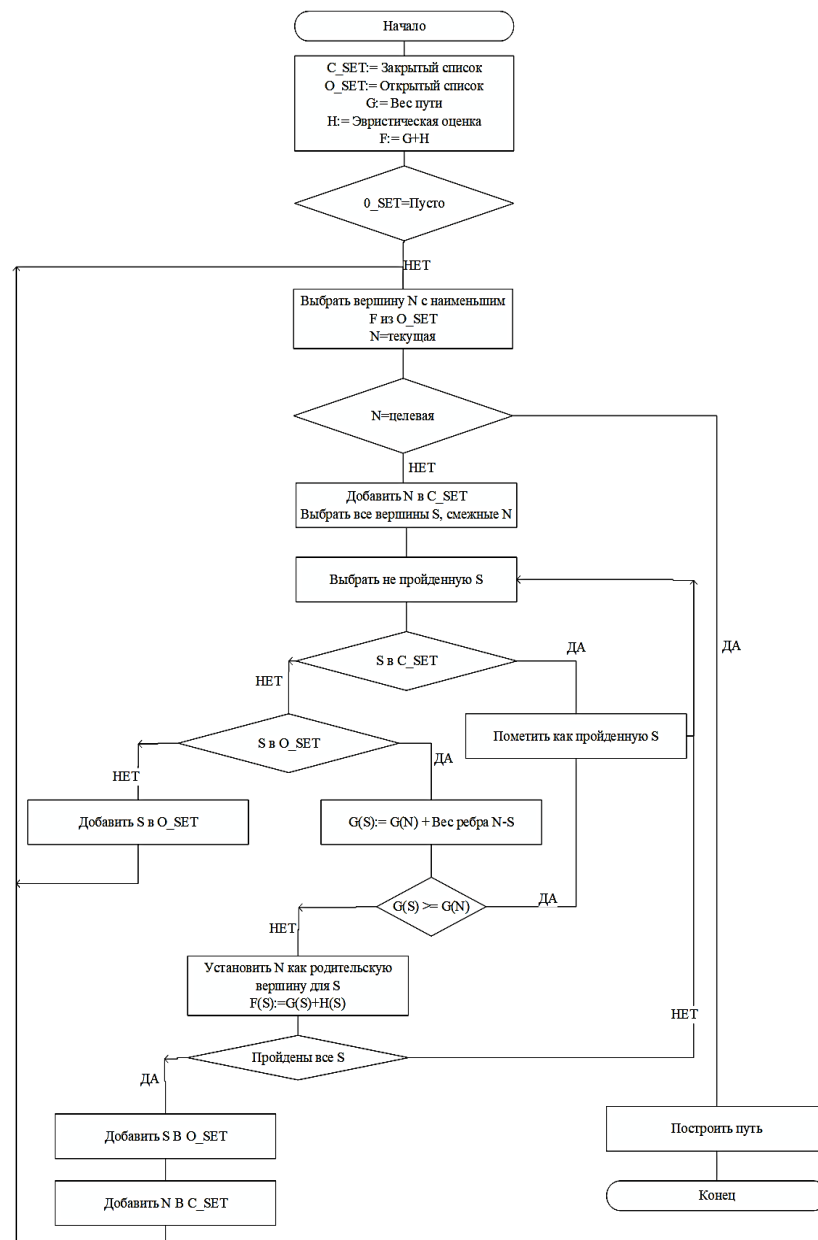


Рисунок 12 – Блок схема алгоритма A*

4.4 Выбор среды проектирования

В качестве среды проектирования информационно-логической модели выбран инструмент Visual Paradigm Online. Он является векторным графическим редактором диаграмм и блок-схем и поддерживает большое количество различных нотаций. Основное преимущество данного инструмента является его открытый исходный код, что делает инструмент бесплатным и надежным, а также возможность получения диаграмм онлайн.

4.5 Выбор языка программирования

В качестве языка программирования был выбран универсальный

объектно-ориентированный язык со строгой типизацией – Java. Основным преимуществом является то, что на нем пишут мобильные приложения на Android, а также самый востребованный язык программирования [11]. Программировать можно практически на всех платформах, язык хорошо спроектирован, логичен и лаконичен.

4.6 Выбор среды программирования

В качестве среды программирования был выбрана Android Studio. Android Studio, основанная на программном обеспечении IntelliJ IDEA от компании JetBrains является официальным средством разработки Android приложений. А в 2017 году компания Google рекомендовала ее как оптимальное средство для создания приложения под Android. Почти все Android-разработчики используют Android Studio.

4.7 Выбор системы управления базами данных

Выбор базы данных (БД) является важным этапом в проектировании системы, потому что БД предопределяет множество нюансов на процессе разработки, а также дальнейшем

В качестве системы управления базами данных (СУБД) была выбрана SQLite. SQLite – это компактная встраиваемая СУБД с открытым исходным кодом.

SQLite – это внутрипроцессная библиотека, которая реализует автономный, безсерверный, транзакционный механизм базы данных SQL с нулевой конфигурацией. Это база данных с нулевой конфигурацией, что означает, что, как и другие базы данных, вам не нужно настраивать ее в вашей системе [12].

5 ПРОВЕДЕНИЕ НАУЧНЫХ ИССЛЕДОВАНИЙ ЭФФЕКТИВНОСТИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ АВТОМАТИЗИРОВАННОЙ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ

5.1 Метрика для расчета точности, погрешность.

Оценка результатов проводилась по параметру GTE (Ground Truth Error) – это максимальная ошибка по отношению к эталонной траектории.

$$GTE = \max(|GTT_i - IMUT_i|), i = 1..I, \quad (6)$$

где I – общее количество точек траектории;

i – порядковый номер точки траектории;

GTT – Ground Truth trajectory;

$IMUT$ – Initial Measurement Unit Trajectory.

5.2 Контрольный пример

Контрольный пример предназначен для проверки правильности работы следующих функций автоматизированной системы инерциальной навигации на базе смартфона:

- построение маршрута от точки А до точки Б;
- просмотр пройденных маршрутов;
- отображение пройденного маршрута;
- проход маршрута и его построение на ГИС;
- отправка пройденного маршрута на сервер.

5.3 Описание исходных данных

Исходные географические данные

Для проверки точности работоспособности системы был пройден маршрут в торговом центре «Космопорт» города Самары. Начальной точкой был ресторан быстрого питания «Макдоналдс», а конечной магазин «ZARA». В начальной точке был найден маршрут до конечной точки. Потом была нажата кнопка, которая начинается отслеживать и строить маршрут пользователя на карте.

Исходные данные контрольного примера

Для выполнения контрольного примера 13.03.2021 был пройден маршрут в ТЦ «Космопорт» в Самаре от «Макдоналдс» до «ZARA». Маршрут был пройден и сохранен с помощью разработанного приложения, которое

установили на мобильный телефон Redmi Note 10 Pro под управлением Android 11.

На рисунке 13 показана запись маршрута во время следования.

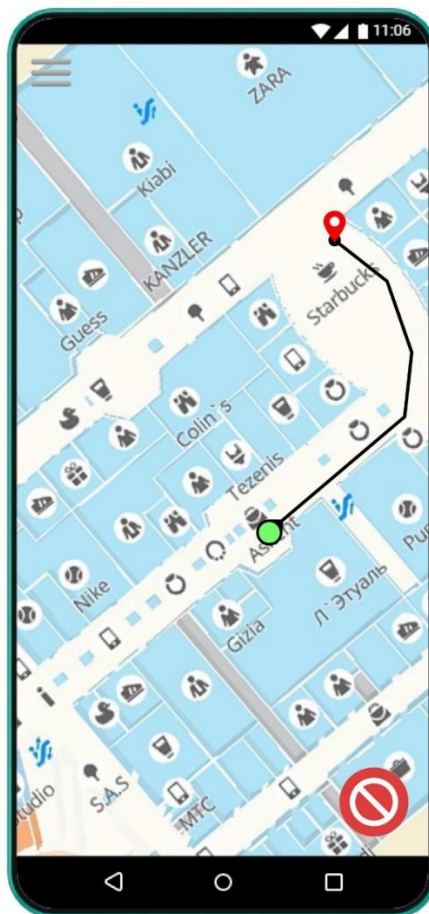


Рисунок 13 – Маршрут до конечной точки

5.4 Исследование эффективности автоматизированной системы

Точность системы

Проведенный контрольный пример дал следующую оценку трехконтурной архитектурной модели инерциальной системы навигации, что максимальная ошибка по отношению к эталонной траектории $GTE = 1,34\%$. В результате сделаны выводы о достаточном достоверном построенном маршруте инерциальной системы навигации.

Результат контрольного примера

В ходе просмотра маршрута, пройденного в ТЦ «Космопорт» был построен маршрут, как на рисунке 14. На нем красной пунктирной линией показан маршрут, построенный системой, а синей линией реальный путь.

Система верно построила маршрут, который ввел пользователь. Но система не смогла построить ровную и до конца точную кривую, это связано с тем, что датчики в Смартфоне не совершенны. Однако это не повлияло на общую картину пройденного маршрута.

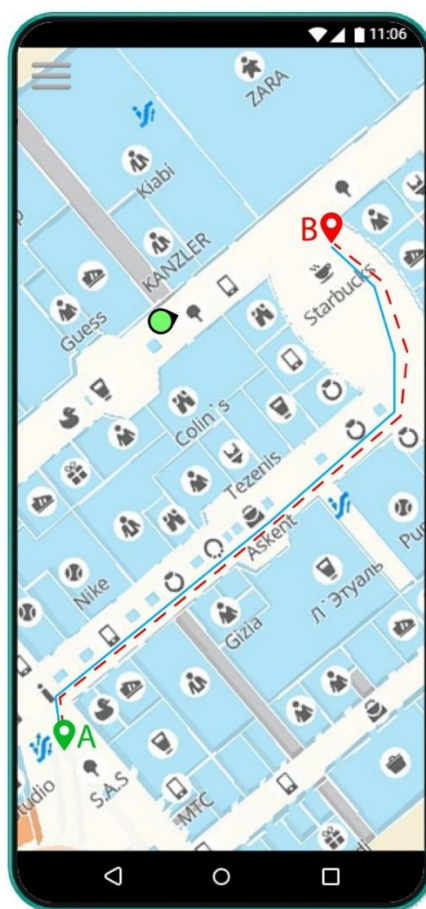


Рисунок 14 – Просмотр пройденного маршрута

6 РАЗРАБОТКА ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ ПО ИСПОЛЬЗОВАНИЮ И НАСТРОЙКЕ КОМПОНЕНТОВ ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА ОБРАБОТКИ ИНФОРМАЦИИ ИЛИ РАЗРАБОТАННОГО ПРОГРАММНОГО

ОБЕСПЕЧЕНИЯ ДЛЯ УПРАВЛЕНИЯ ОБЪЕКТОМ АВТОМАТИЗАЦИИ.

6.1 Назначение системы

В работе рассмотрен подход к позиционированию мобильных устройств при помощи инерциальной навигации. Результатом работы является вывод о том, что данный подход может успешно применяться для мобильных устройств, однако требует немалых усилий для борьбы с накапливаемой погрешностью, для чего следует пользоваться шумовыми фильтрами и географическими особенностями местности, где проводится отслеживание устройства. Наиболее перспективной областью навигация пользователей на крупных предприятиях, рассчитанных на высокую посещаемость – таких, как торгово-развлекательные центры.

Самым главным положительным фактом является, что инерциальные системы будут становиться более актуальными в будущем по мере совершенствования акселерометров и гироскопов, применяемых в мобильных устройствах.

6.2 Требования к техническому обеспечению

Для работы серверной части разработанной системы инерциальной навигации необходимы следующие технические средства:

- IBM-совместимый компьютер;
- не менее 1024 Мбайт оперативной памяти;
- не менее 10549 Мбайт на жестком диске;
- поддерживаемая Windows 10 графическая карта;
- поддерживаемая Windows 10 мышь и клавиатура;
- операционная система Windows 10.

Для работы клиентской части разработанной системы необходимо одно из следующих технических средств:

- мобильное устройство под управлением операционной системы iOS не ниже 10 версии;

- мобильное устройство под управлением операционной системы Android не ниже 6 версии.

ЗАКЛЮЧЕНИЕ

В процессе прохождения преддипломной практики:

- был проведен анализ имеющихся современных языков, систем и технологий программирования, а также технологий проектирования программного и для решения задач Бесплатформенной инерциальной навигации;
- был проведен анализ имеющихся методологий анализа профессиональной информации и технологии подготовки научных докладов, публикаций и аналитических обзоров;
- был разработан аналитический обзор с обоснованием выбора технологии проектирования и технологий программирования для решения задачи Бесплатформенной инерциальной навигации (методологии проектирования – UML, IDEF1X, средства проектирования – VisualParadimOnline, ErWin Data Modeler);
- была разработана математическая модель для решения задачи Бесплатформенной инерциальной навигации;
- были разработаны основные требования к проектируемому программному обеспечению автоматизированной информационной системы Бесплатформенной инерциальной навигации;
- были разработаны алгоритмы и программная реализация математической модели решения задач Бесплатформенной инерциальной навигации с использованием современных языков и систем программирования ((языки программирования Java, среды программирования AndroidStudio, операционная система Android);
- было разработано программное и аппаратное обеспечение автоматизированной информационной системы Бесплатформенной инерциальной навигации;
- были проведены научные исследования эффективности программной реализации автоматизированной информационной системы Бесплатформенной инерциальной навигации;

- была разработана техническая документация по использованию и настройке компонентов программно-аппаратного комплекса обработки информации Бесплатформенной инерциальной навигации.

Таким образом, в процессе выполнения преддипломной практики были освоены все необходимые индикаторы (ОПК-1.2, ОПК-2.2, ОПК-3.3, ОПК-4.2, ОПК-5.2, ОПК-6.3) компетенций (ОПК-1, ОПК-2, ОПК-3, ОПК-4, ОПК-5, ОПК-6).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Языки программирования [Электронный ресурс]. URL: <http://life-prog.ru> (дата обращения: 08.10.2020).
- 2 McALLISTER N. InfoWorld. «10 языков программирования, которые могут перевернуть мир IT» // Компьютерные вести [Электронный ресурс]. URL: <http://www.kv.by/content/325498-10-yazykovprogrammirovaniya-kotorye-mogut-perevernut-mir-it> (дата обращения: 08.10.2020).
- 3 Wyvern – новая система, позволяющая использовать несколько разных языков программирования при создании одной программы // DailyTechInfo [Электронный ресурс]. URL: <http://www.dailytechinfo.org/infotech/6167-wyvern-novaya-sistema-pozvolyayuschaya-ispolzovat-neskolkoraznyh-yazykov-programmirovaniya-pri-sozdanii-odnoy-programmy.html> (дата обращения: 20.12.2020).
- 4 Sketch – новый язык программирования, способный самостоятельно оптимизировать и завершать незаконченные участки кода // DailyTechInfo [Электронный ресурс]. URL: <http://www.dailytechinfo.org/infotech/5691-sketch-novyuy-yazyk-programmirovaniya-sposobnyuy-samostoyatelnooptimizirovat-i-zavershat-nezakonchennye-uchastki-koda.html> (дата обращения: 09.11.2020).
- 5 Топ-3 среды разработки для Android // AndroidAPP [Электронный ресурс]. URL: <https://app-android.ru/blog/environment-develop-android> (дата обращения: 20.11.2020).
- 6 Информационный портал о жизненном цикле информационной системы [Электронный ресурс]. – URL: <https://sites.google.com/site/metodsybd/blok-4-sredstva-proektirovania/4-1-case-sredstva> (дата обращения 01.10.2020).
- 7 Гома, Х. UML. Проектирование систем реального времени, распределенных и параллельных приложений [Текст] / Х. Гома. – Litres, 2017.
- 8 Леоненков, А.В. Самоучитель UML 2 [Текст] / А.В. Леоненков. – БХВ- Петербург, 2013.

- 9 Буч, Г. Введение в UML от создателей языка. Перевод: Н. Мухин [Текст] / Г. Буч, Д. Рамбо, И. Якобсон. – М: ДМК Пресс, 2011. – с. 496
- 10 Грекул, В.И. Проектирование информационных систем [Текст] / В.И. Грекул, Г. Н Денищенко. – 2016.
- 11 Java [Электронный ресурс] // URL: <http://ab.kh.ua/plusi-java-programmirovania/> (дата обращения: 20.03.2021)
- 12 SQLite [Электронный ресурс] // URL: <https://coderlessons.com/tutorials/bazy-dannykh/vyuchit-sqlite/sqlite-kratkoe-rukovodstvo> (дата обращения: 20.03.2021)

ПРИЛОЖЕНИЕ А

ЛИСТИНГ ПРОГРАММЫ

```
import org.springframework.security.core.GrantedAuthority;
import org.springframework.security.core.userdetails.UserDetails;

import javax.persistence.*;
import java.util.Collection;

@Entity
@Table(name = "user_granted")
public class User implements UserDetails {

    @Id
    @Column
    private String userName;

    @Column
    private String password;

    @OneToOne(mappedBy = "userGranted", fetch = FetchType.LAZY)
    private Client client;

    public String getUserName() {
        return userName;
    }

    public void setUserName(String userName) {
        this.userName = userName;
    }
}
```

```
public void setPassword(String password) {  
    this.password = password;  
}
```

@Override

```
public Collection<? extends GrantedAuthority> getAuthorities() {  
    return null;  
}
```

@Override

```
public String getPassword() {  
    return password;  
}
```

@Override

```
public String getUsername() {  
    return userName;  
}
```

@Override

```
public boolean isAccountNonExpired() {  
    return true;  
}
```

@Override

```
public boolean isAccountNonLocked() {  
    return true;  
}
```

```

@Override

public boolean isCredentialsNonExpired() {
    return true;
}

@Override

public boolean isEnabled() {
    return true;
}
}

@Service

public class UserService implements UserDetailsService {

    private UserRepository userRepository;
    private PasswordEncoder passwordEncoder;

    @Autowired

    public UserService(UserRepository userRepository, PasswordEncoder
passwordEncoder) {
        this.userRepository = userRepository;
        this.passwordEncoder = passwordEncoder;
    }

    public void registerUser(User user) {
        String encodedPassword = passwordEncoder.encode(user.getPassword());
        user.setPassword(encodedPassword);
        userRepository.save(user);
    }

    @Override

```

```

        public UserDetails loadUserByUsername(String s) throws
UsernameNotFoundException {
            return userRepository.findById(s).orElseThrow(() -> new
UsernameNotFoundException("No user with such name " + s));
        }
    }
}

```

// класс ИНС

```
public class INS {
```

```

    public DenseMatrix64F Pos_b=new DenseMatrix64F(3,1);//изменение
позиции

```

```

    public DenseMatrix64F Vel_b=new DenseMatrix64F(3,1);//изменении
скорости

```

```
    public DenseMatrix64F Cbn=new DenseMatrix64F(3,3);
```

```
    public DenseMatrix64F grav =new DenseMatrix64F(3,1);
```

```

    public DataAccum dataAccum =new DataAccum();           //sensor
data accumulator

```

```
    //Temporary variables (Save the GC, save the world)
```

```
    private DenseMatrix64F acc=new DenseMatrix64F(3,1);
```

```
    private DenseMatrix64F gyro=new DenseMatrix64F(3,1);
```

```
    DenseMatrix64F rotv=new DenseMatrix64F(3,1);
```

```
    DenseMatrix64F mx_a=new DenseMatrix64F(3,3);
```

```
    DenseMatrix64F mx_b=new DenseMatrix64F(3,3);
```

```
    DenseMatrix64F vr_a=new DenseMatrix64F(3,1);
```

```
    public INS(float[] pos, float[] vel, float[] dcm) {
```

```
        int i;
```

```

        for (i=0;i<3;i++) {
            Pos_b.set(i,pos[i]);
            Vel_b.set(i,vel[i]);
        }

        for (i=0;i<9;i++) {
            Cbn.set(i,dcm[i]);
        }

        //Assume a fixed gravity
        grav .set(2,-SensorManager.GRAVITY_EARTH); //Gravity in
NED (not ENU)
    }

    //Setters and getters
    public void set_dcm(float[] dcm) {
        for (int i=0;i<9;i++) {
            Cbn.set(i,dcm[i]);
        }
    }

    public void setPos(float[] pos) {
        for (int i=0;i<3;i++)
            Pos_b.set(i,pos[i]);
    }

    public void setVel(float[] vel) {
        for (int i=0;i<3;i++)
            Vel_b.set(i,vel[i]);
    }

```

```

public double[] get_dcm() {
    return Cbn.data;
}

public double[] get_pos() {
    return Pos_b.data;
}

public void get_posn(float[] out) {
    CommonOps.mult(Cbn, Pos_b, vr_a);
    out[0]=(float) vr_a.get(0);
    out[1]=(float) vr_a.get(1);
    out[2]=(float) vr_a.get(2);
}

public void get_gravity(float[] out) {
    out[0]=(float) grav .get(0);
    out[1]=(float) grav .get(1);
    out[2]=(float) grav .get(2);
}

//Algorithms
public void update_attI(float[] gdat, float dt) {
    rotv.set(0,gdat[0]*dt);
    rotv.set(1,gdat[1]*dt);
    rotv.set(2,gdat[2]*dt);

    //Convert rotation vector to dcm
    rot2dcm(rotv, mx_a);

```

```

        //Update the dcm
        CommonOps.mult(Cbn, mx_a, mx_b);
        Cbn.set(mx_b);
    }

    public void rot2dcm(DenseMatrix64F rotvec, DenseMatrix64F dcm) {
        double[] rot=rotvec.data;
        double rot_norm=NormOps.fastNormF(rotv);

        if (rot_norm>0) {
            double sr_a=Math.sin(rot_norm)/rot_norm;
            double sr_b=(1-Math.cos(rot_norm))/(rot_norm*rot_norm);

            //Dcm =eye(3)+sr_a*skew(rot)+sr_b*skew(rot)*skew(rot);
            dcm.set(0,0,1+sr_b*(-rot[2]*rot[2]-rot[1]*rot[1]));
            dcm.set(0,1,sr_b*(rot[1]*rot[0])+sr_a*(-rot[2]));
            dcm.set(0,2,sr_b*(rot[2]*rot[0])+sr_a*(rot[1]));
            dcm.set(1,0,sr_b*(rot[0]*rot[1])+sr_a*(rot[2]));
            dcm.set(1,1,1+sr_b*(-rot[2]*rot[2]-rot[0]*rot[0]));
            dcm.set(1,2,sr_b*(rot[2]*rot[1])+sr_a*(-rot[0]));
            dcm.set(2,0,sr_b*(rot[2]*rot[0])+sr_a*(-rot[1]));
            dcm.set(2,1,sr_b*(rot[2]*rot[1])+sr_a*(rot[0]));
            dcm.set(2,2,1+sr_b*(-rot[1]*rot[1]-rot[0]*rot[0]));
        }
    }

    public void update_vell(float[] adat, float dt) {
        acc.set(0, adat[0]);
        acc.set(1, adat[1]);
    }

```



```

        acc.set(2, adat[2]);

        //Specific force (acc=acc+Cbn'*gravity)
        CommonOps.multAddTransA(Cbn, grav , acc);


        //Update vel with specific force (Vel_b=Vel_b+dt*acc)
        CommonOps.addEquals(Vel_b, dt, acc);
    }

    public void update_velII(float[] gdat, float dt) {
        gyro.set(0, gdat[0]);
        gyro.set(1, gdat[1]);
        gyro.set(2, gdat[2]);

        //vel_inc=(cross(Vb,(Cbn'*wie_n)+w))*dt;
        skew(Vel_b, mx_a);
        CommonOps.multAdd(dt, mx_a, gyro, Vel_b);
        //Vel_b=Vel_b+dt*mx_a*gyro
    }

    public void update_posI(float[] gdat, float dt) {
        gyro.set(0, gdat[0]);
        gyro.set(1, gdat[1]);
        gyro.set(2, gdat[2]);

        //Update for the rotation (Pos_b=Pos_b+cross(Pos_b,gyro)
        skew(Pos_b, mx_a);
        CommonOps.multAdd(dt, mx_a, gyro, Pos_b);
    }

```

```
}
```

```
public void update_posII(float dt) {
```

```
    //Update pos with body vel (Pos_b=Pos_b+dt*Vel_b)
```

```
    CommonOps.addEquals(Pos_b, dt, Vel_b);
```

```
}
```

```
public static void skew(DenseMatrix64F vec, DenseMatrix64F smat) {
```

```
    smat.zero();
```

```
    smat.set(0,1,-vec.get(2));
```

```
    smat.set(0,2,vec.get(1));
```

```
    smat.set(1,0,vec.get(2));
```

```
    smat.set(1,2,-vec.get(0));
```

```
    smat.set(2,0,-vec.get(1));
```

```
    smat.set(2,1,vec.get(0));
```

```
}
```

```
public void update(DenseMatrix64F dx) {
```

```
    double sr_a;
```

```
    for (int i=0;i<3;i++) {
```

```
        sr_a=Pos_b.get(i)-dx.get(i);
```

```
        Pos_b.set(i,sr_a);
```

```
        sr_a=Vel_b.get(i)-dx.get(i+3);
```

```
        Vel_b.set(i,sr_a);
```

```
        vr_a.set(i, dx.get(i+6));
```

```

    }

    skew(vr_a,mx_a);
    CommonOps.mult(mx_a, Cbn, mx_b);
    CommonOps.addEquals(Cbn, mx_b);
}

////////////////////////////////////

//Sensor Data Accumulator
class DataAccum {
    private DenseMatrix64F acacc=new DenseMatrix64F(3,1);
    private DenseMatrix64F acgyro=new DenseMatrix64F(3,1);
    private int acina=0, acing=0;    //Accumulator indexes

    //Temporary variables
    private DenseMatrix64F vr_a=new DenseMatrix64F(3,1);

    public void clear() { //Clears the sensor data accumulators
        acacc.zero();
        acgyro.zero();
        acina=0;
        acing=0;
    }

    public void addacc(float[] dat) { //Updates the acc accumulator
        acc.set(0, dat[0]);
        acc.set(1, dat[1]);
        acc.set(2, dat[2]);

        CommonOps.addEquals(acacc, acc);
    }
}

```

```

        acina++;
    }

    public void addgyro(float[] dat) { //Updates the gyro
accumulator
        gyro.set(0, dat[0]);
        gyro.set(1, dat[1]);
        gyro.set(2, dat[2]);

        CommonOps.addEquals(acgyro, gyro);
        acing++;
    }

    public void avacc(float[] out) { //Avarage of acc
        if (acina>0)
            for (int i=0;i<3;i++)
                out[i]=(float) acacc.get(i)/acina;
        else
            { out[0]=0;out[1]=0;out[2]=0;}
    }

    public void avgyro(float[] out) { //Avarage of gyro
        if (acing>0)
            for (int i=0;i<3;i++)
                out[i]=(float) acgyro.get(i)/acing;
        else
            { out[0]=0;out[1]=0;out[2]=0;}
    }

```

//Note: Below the return values are the skewed matrices!!!

```

public void avacc(DenseMatrix64F out) { //Avarage of acc
    if (acina>0) {
        CommonOps.scale(1/acina, acacc, vr_a);
        skew(vr_a, out);
    }
    else
        out.zero();
}

public void avgyro(DenseMatrix64F out) { //Avarage of gyro
    if (acing>0) {
        CommonOps.scale(1/acing, acgyro, vr_a);
        skew(vr_a, out);
    }
    else
        out.zero();
}

};

}

```