

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 учебная
(учебная, производственная)

Тип практики технологическая (проектно-технологическая) практика
(в соответствии с ОПОП ВО)

Сроки прохождения практики: с 01.09.2020 по 26.12.2020
(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. _____О.К. Головнин

Дата сдачи 26.12.2020

Дата защиты 26.12.2020

Оценка _____

Самара 2020

СОДЕРЖАНИЕ

1. Индивидуальное задание на практику.
2. Рабочий график (план) проведения практики.
3. Описательная часть.
4. Список использованных источников.

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 26.08.2020 г. № 318-ПР

на кафедру информационных систем и технологий
(наименование профильной организации или структурного подразделения университета)

Тема НИР: «Исследование математического аппарата и технологий
построения инерциальных навигационных систем»

Планируемые результаты освоения образовательной программы (компетенции)	Планируемые результаты практики	Содержание задания
ОПК-5. Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем ОПК-5.2. Осуществляет модернизацию программного и аппаратного обеспечения информационных	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	Провести анализ имеющихся технологий проектирования программного и (или) аппаратного обеспечения для решения задач автоматизированной информационной системы Бесплатформенная инерциальная навигация. Модернизировать имеющееся программное и аппаратное обеспечения для автоматизированной информационной системы Бесплатформенная

и автоматизированных систем для решения профессиональных задач	информационных и автоматизированных систем для решения профессиональных задач.	инерциальная навигация.
ОПК-6. Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации и автоматизированного проектирования ОПК-6.2. Анализирует техническое задание, разрабатывает и оптимизирует программный код для решения задач обработки информации и автоматизированного проектирования	Знать: современные языки, системы и технологии программирования для решения задач обработки информации и автоматизированного проектирования. Уметь: применять современные языки, системы и технологии программирования для оптимизации программных кодов при решении задач обработки информации и автоматизированного проектирования. Владеть: навыками оптимизации программных кодов при решении задач обработки информации и автоматизированного проектирования.	Провести анализ имеющихся современных языков, систем и технологий программирования, для автоматизированной информационной системы Бесплатформенная инерциальная навигация. Разработать программное и аппаратное обеспечения для автоматизированной информационной системы Бесплатформенная инерциальная навигация.
ОПК-7. Способен адаптировать зарубежные комплексы обработки информации и автоматизированного проектирования к нуждам отечественных предприятий ОПК-7.2. Осуществляет настройку интерфейса,	Знать: имеющиеся методы разработки пользовательских шаблонов, подключения библиотек, добавления новых функций в зарубежные комплексы обработки информации для приведения в соответствие с национальными стандартами и интеграции с отраслевыми информационными системами. Уметь: применять имеющиеся методы	Разработать или настроить интерфейс для автоматизированной информационной системы Бесплатформенная инерциальная навигация. Сделать описание используемых библиотек и зарубежных комплексов обработки информации.

<p>разработку пользовательских шаблонов, подключение библиотек, добавление новых функций в зарубежные комплексы обработки информации для приведения в соответствие с национальными стандартами и интеграции с отраслевыми информационным и системами</p>	<p>разработки пользовательских шаблонов, подключения библиотек, добавления новых функций в зарубежные комплексы обработки информации для приведения в соответствие с национальными стандартами и интеграции с отраслевыми информационными системами. Владеть: навыками разработки пользовательских шаблонов, подключения библиотек, добавления новых функций в зарубежные комплексы обработки информации для приведения в соответствие с национальными стандартами и интеграции с отраслевыми информационными системами.</p>	
<p>ОПК-8. Способен осуществлять эффективное управление разработкой программных средств и проектов ОПК-8.2. Реализует методы и средства разработки программного обеспечения, методы управления проектами разработки программного</p>	<p>Знать: основные методы и средства разработки программного обеспечения, методы управления проектами разработки программного обеспечения, способы организации проектных данных, нормативно-технические документы (стандарты и регламенты) по разработке программных средств и проектов. Уметь: применять методы и средства разработки программного обеспечения, методы управления проектами разработки программного обеспечения,</p>	<p>Разработать проект автоматизированной информационной системы Бесплатформенная инерциальная навигация.</p> <p>Сделать описание способов организации проектных данных и используемых нормативно-технических документов, предметной области для решения задач Бесплатформенная инерциальная навигация.</p>

обеспечения, способы организации проектных данных, нормативно-технические документы (стандарты и регламенты) по разработке программных средств и проектов	способы организации проектных данных, нормативно-технические документы (стандарты и регламенты) для разработки программных средств и проектов. Владеть: навыками разработки программного обеспечения, управления проектами разработки программного обеспечения, организации проектных данных, использования нормативно-технических документов (стандартов и регламентов) для разработки программных средств и проектов.	
---	--	--

Дата выдачи задания 01.09.2020.

Срок представления на кафедру отчета о практике 26.12.2020.

Руководитель практики,

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

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

Рабочий график (план) проведения практики

Дата (период)	Содержание задания	Результаты практики
01.09.2020	Получение задания на практику	Получено задание на практику
01.09.2020	Утверждение задания руководителем практики.	Утверждены индивидуальное задание и рабочий план-график.
02.09.2020 — 10.09.2020	Провести анализ имеющихся современных языков, систем и технологий программирования, для автоматизированной информационной системы Бесплатформенная инерциальная навигация.	Обучающийся приобрел навыки анализа имеющихся современных языков, систем и технологий разработки Бесплатформенная инерциальная навигация.
11.09.2020 — 19.09.2020	Провести анализ имеющихся технологий проектирования программного и (или) аппаратного обеспечения для решения задач автоматизированной информационной системы Бесплатформенная инерциальная навигация.	Обучающийся приобрел навыки анализа имеющихся технологий проектирования программного и (или) аппаратного обеспечения для решения задач Бесплатформенная инерциальная навигация.
21.09.2020 — 05.10.2020	Разработать проект автоматизированной информационной системы Бесплатформенная инерциальная навигация.	Обучающийся научился проектировать автоматизированную информационную систему Бесплатформенная инерциальная навигация.
06.10.2020 — 21.10.2020	Сделать описание способов организации проектных данных и используемых нормативно-технических документов, предметной области для решения задач Бесплатформенная инерциальная навигация.	Обучающийся научился делать описание способов организации проектных данных и используемых нормативно-технических документов, предметной области для решения задач Бесплатформенная инерциальная навигация.
22.10.2020 — 30.11.2020	Разработать программное и аппаратное обеспечения для автоматизированной информационной системы Бесплатформенная инерциальная навигация.	Обучающийся научился разрабатывать программное и аппаратное обеспечения для автоматизированной информационной системы Бесплатформенная инерциальная навигация.
01.12.2020	Разработать интерфейс для	Обучающийся приобрел

— 09.12.2020	автоматизированной информационной системы Бесплатформенная инерциальная навигация.	навыки разработки интерфейса для автоматизированной информационной системы Бесплатформенная инерциальная навигация.
10.12.2020 — 19.12.2020	Сделать описание используемых библиотек и зарубежных комплексов обработки информации.	Обучающийся приобрел навыки описания используемых библиотек и зарубежных комплексов обработки информации.
21.12.2020 — 23.12.2020	Написание, оформление письменного отчета и его предъявление руководителю практики.	Написан и оформлен письменный отчет по практике.
24.12.2020 — 25.12.2020	Подготовка устного отчета о прохождении практики.	Подготовлен устный отчет о прохождении практики.
26.12.2020	Защита отчета. Получение зачета по практике.	Получен зачет 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 разным инструментарий, упрощающий построение графического изображения с нуля.

На сегодняшний день самыми востребованными средами являются:

- Eclipse;
- IntelliJ Idea;
- Android Studio.

Каждая IDE обладает уникальными характеристиками.

1.2.1 Eclipse

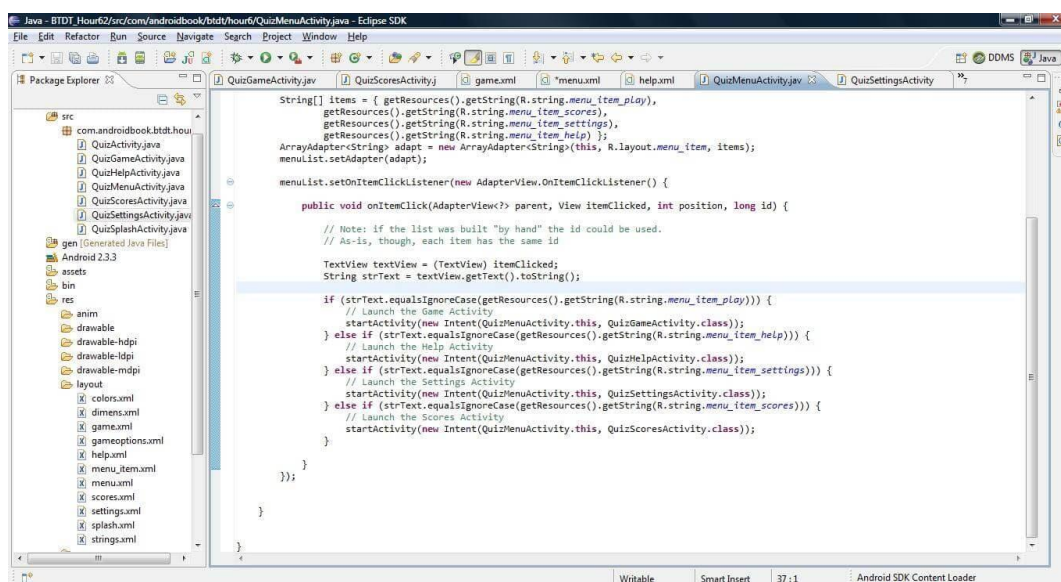


Рисунок 1 – Интерфейс Eclipse

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 Андроид Студии (2014 год) Google решил перевести сотрудничество с Eclipse на «второй план».

1.2.2 **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.

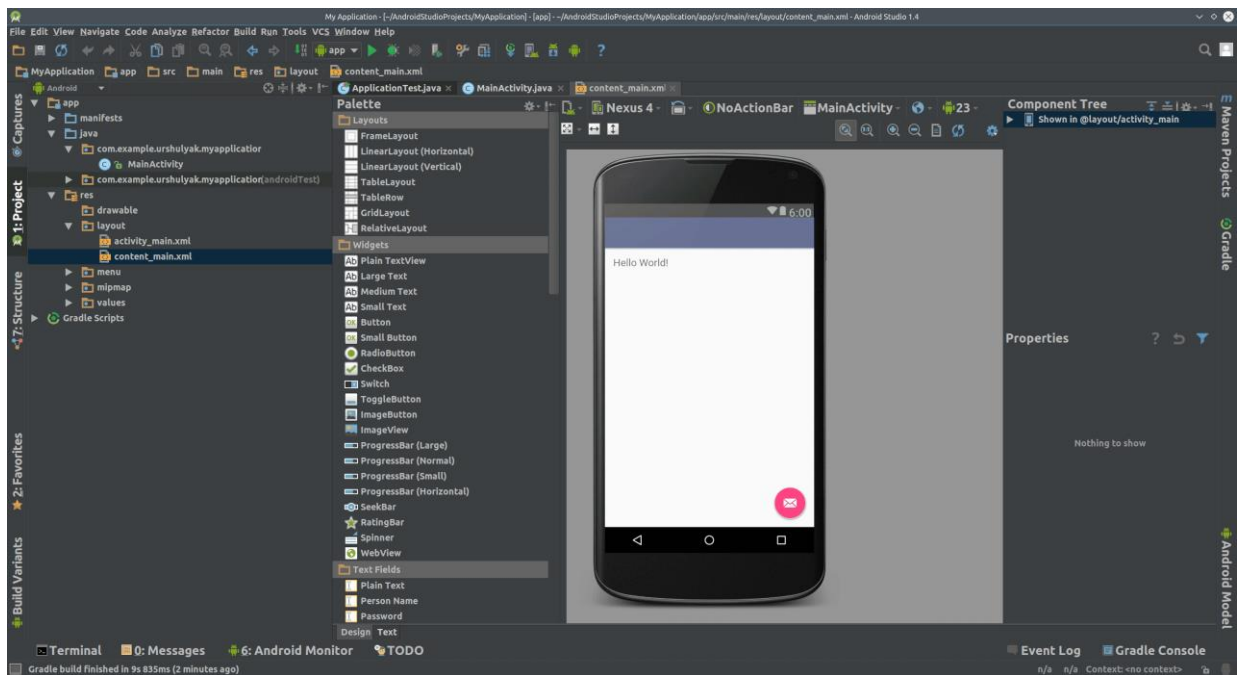


Рисунок 2 – Интерфейс IntelliJ Idea

1.2.3 Android Studio

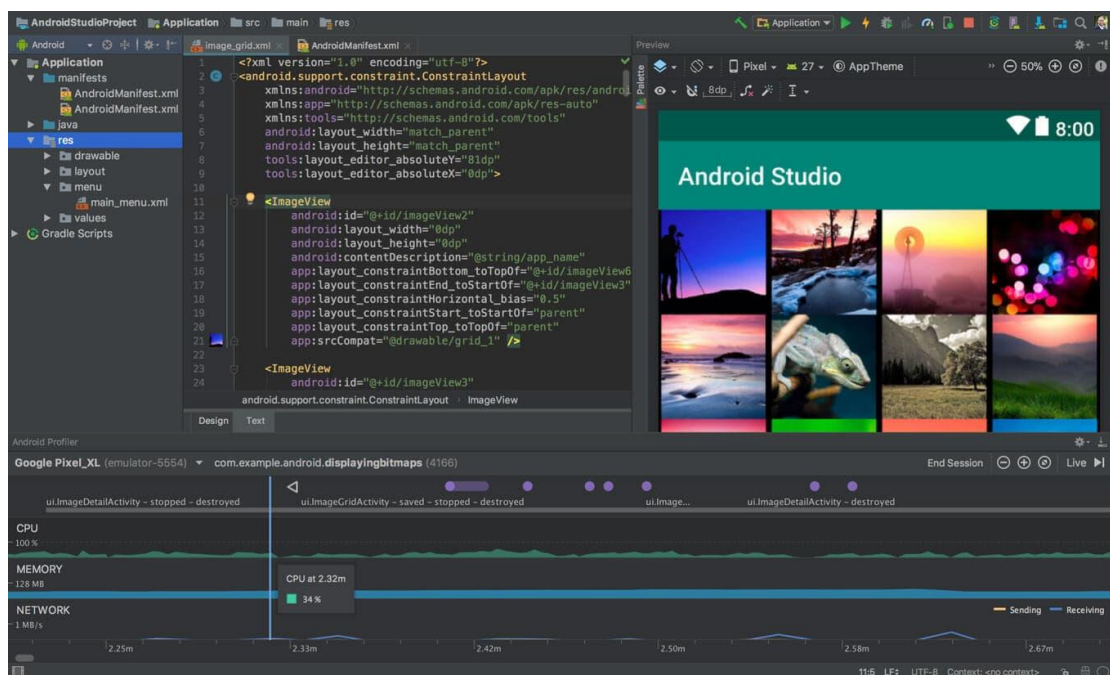


Рисунок 3 – Интерфейс Android Studio

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.

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 проекта автоматизированной информационной системы

2.1 Разработка логической модели данных

Для создания качественной информационной системы необходимо построить логическую модель данных. Логическая модель дает графическое

представление структуры предметной области. В ней становится понятно, какой информацией система должна управлять.

Одним из основных средств построения логической модели в настоящее время являются ER-диаграммы.

На рисунке 4 показана логическая модель базы данных проектируемой системы. На ER – модели показаны сущности, такие как справочники и основные.

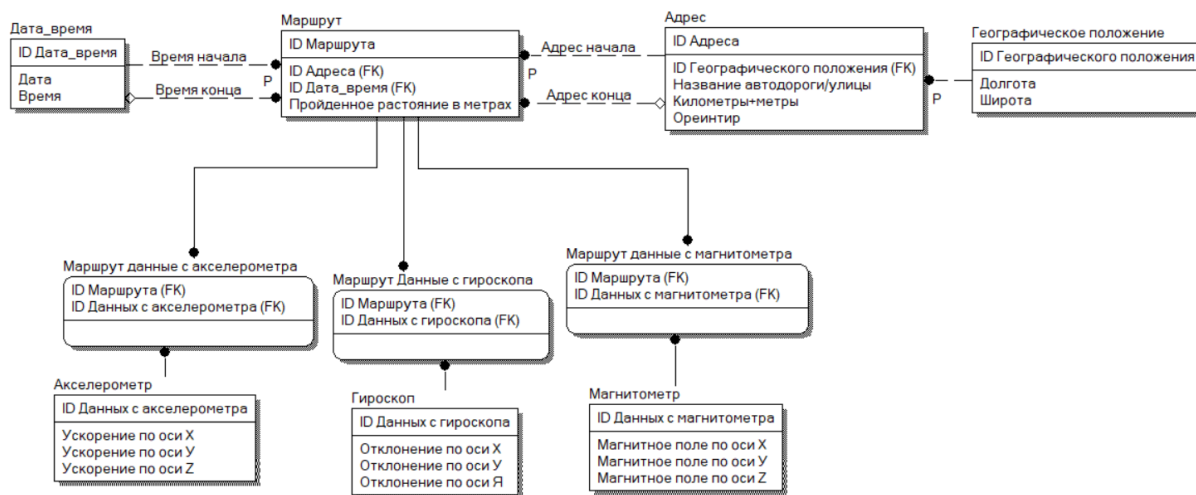


Рисунок 4 – Логическая модель базы данных

3 Разработка приложения

3.1 Алгоритм получения данных с помощью акселерометра

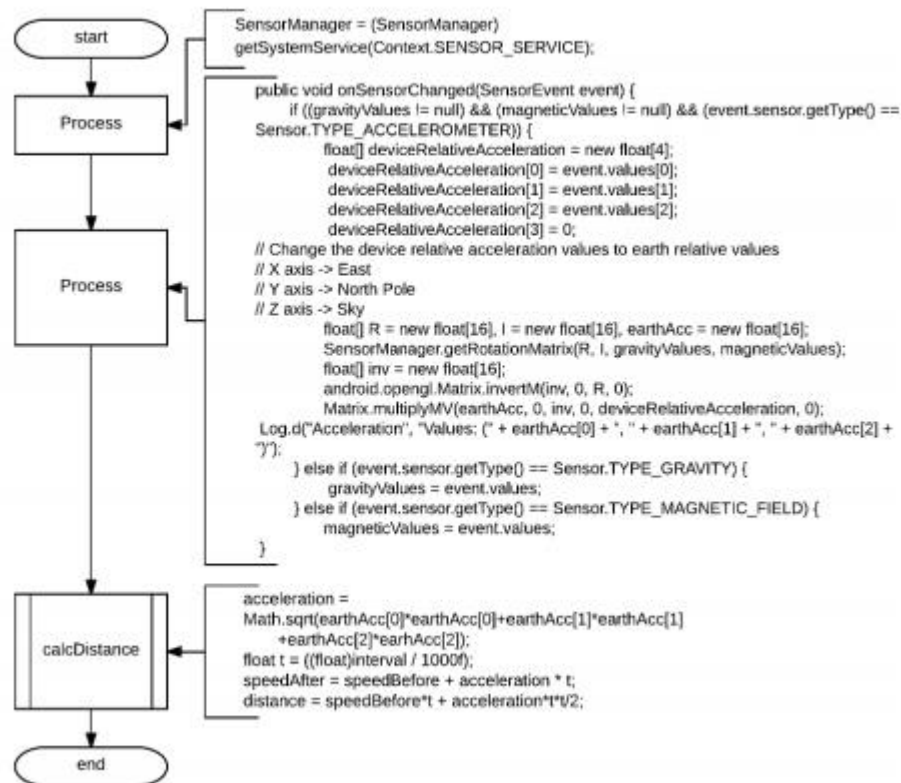


Рисунок 5 – Алгоритм сбора информации с акселерометра

Данный алгоритм (рисунок 5) 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.

3.2 Алгоритм работы Фильтра Кальмана

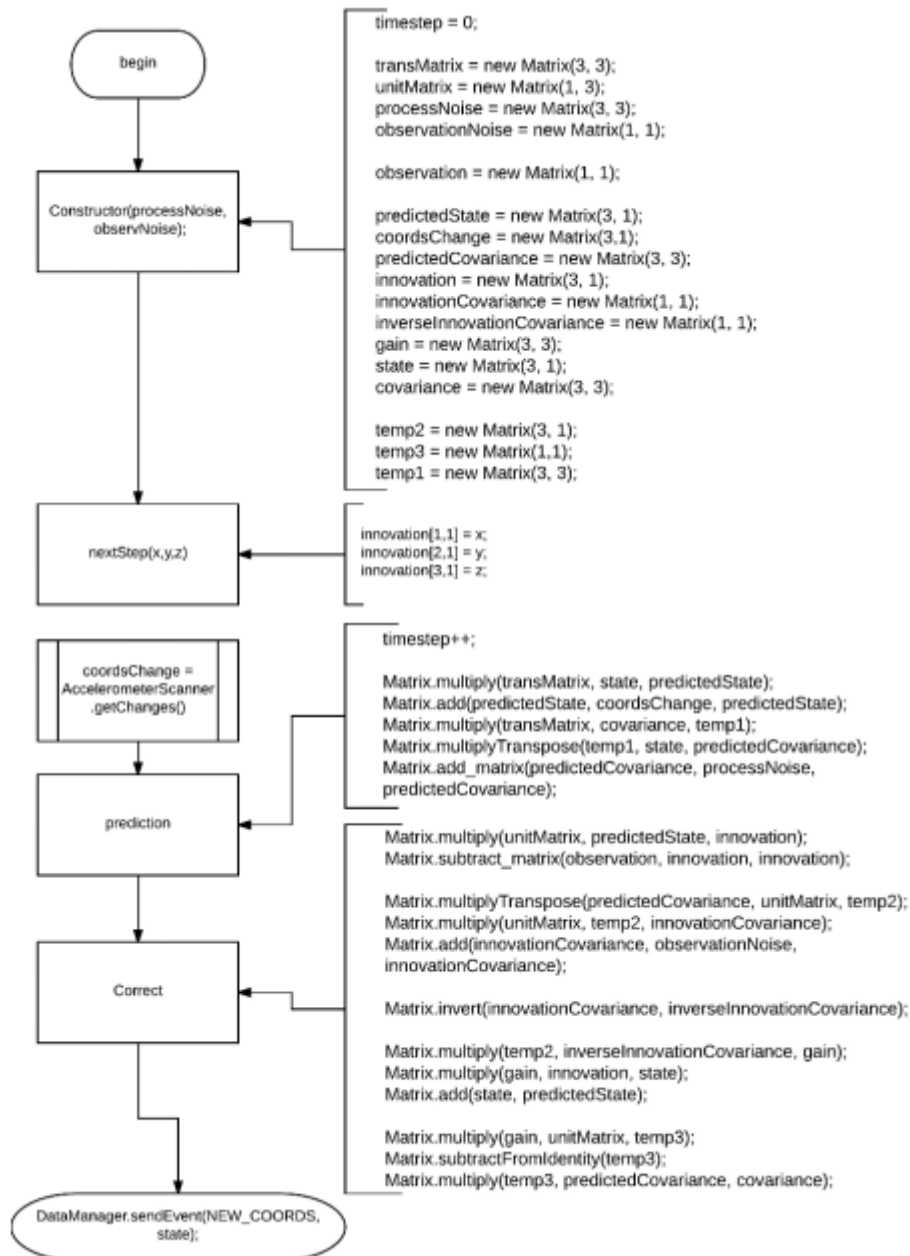


Рисунок 6 – Алгоритм, описывающий работу фильтра Калмана

Данный алгоритм реализует фильтрацию Калмана (рисунок 6). 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.

3.3 Архитектура работы с картой

Класс оформлен по шаблону проектирования «одиночка». Инициализируется при запуске приложения, в ходе которой выделяется место в памяти и кешируются необходимые изображения карт. Количество используемой памяти определяется как 30% от доступной приложению. Интерфейс содержит всего один метод `getMapBitmap` с названием карты в параметре. В случае наличия изображения в памяти оно 50 отдается в возвращаемом параметре, иначе требуемое изображение считывается из ресурсов приложения, записывается в кеш и также возвращается. Данный подход необходим для уменьшения времени задержки в отрисовке наиболее часто запрашиваемых карт

3.4 Архитектура ОС Android

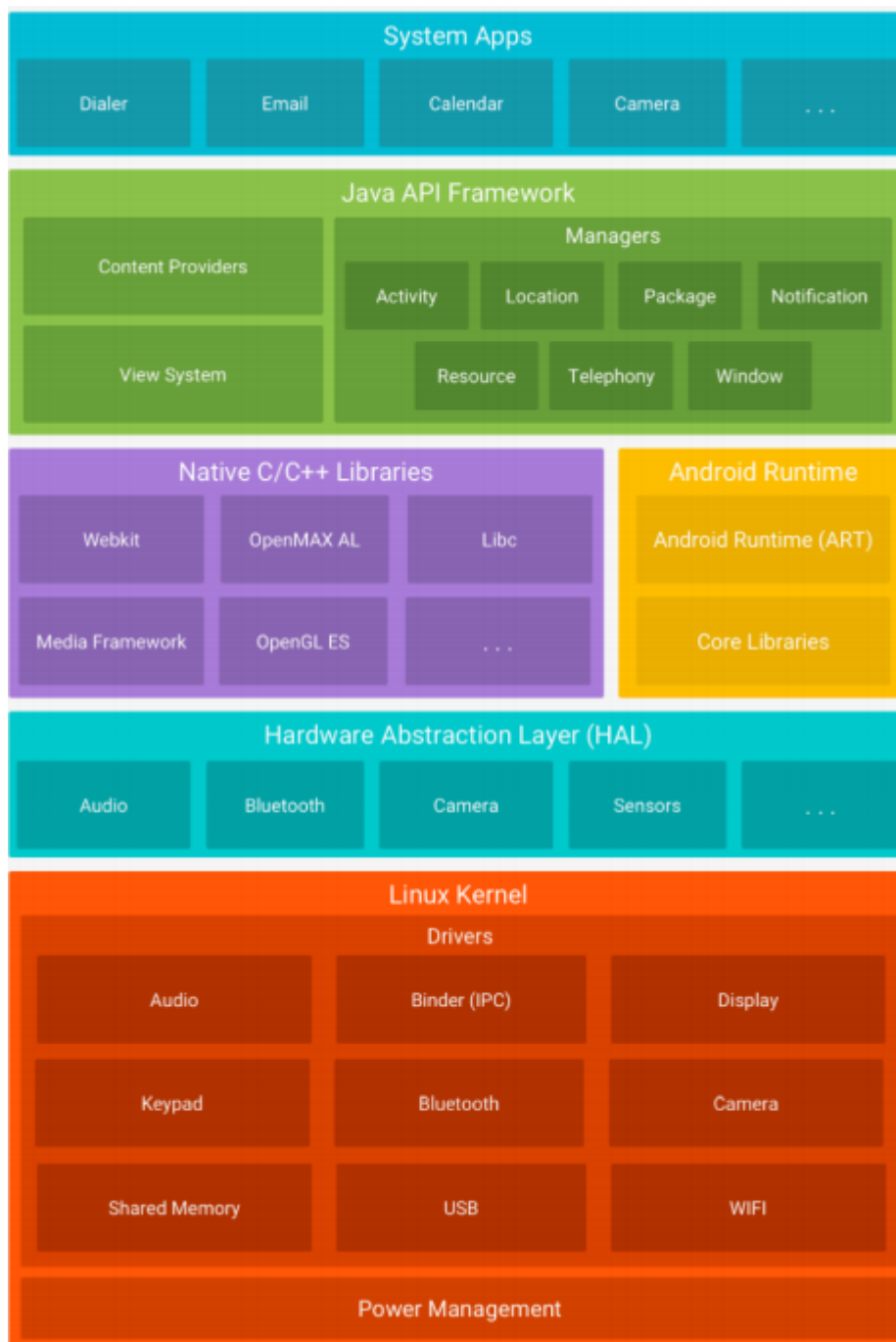


Рисунок 7 – Архитектура ОС Android

Android — это программный стек с открытым исходным кодом, основанный на Linux, созданный для широкого спектра устройств и форм-факторов. На следующей диаграмме показаны основные компоненты платформы Android (рисунок 7).

Основой платформы Android является ядро Linux.

Например, Android Runtime (ART) полагается на ядро Linux в вопросах, связанных с базовыми возможностями, таких как поточное и низкоуровневое управление памятью.

Использование ядра Linux позволяет Android использовать ключевые функции безопасности и позволяет производителям устройств разрабатывать аппаратные драйверы для известного ядра.

Уровень абстрагирования оборудования (HAL) предоставляет стандартные интерфейсы, которые предоставляют возможности аппаратного обеспечения устройства для высокоуровневой инфраструктуры Java API. HAL состоит из нескольких модулей библиотеки, каждый из которых реализует интерфейс для определенного типа аппаратного компонента, такого как камера или модуль Bluetooth. Когда базовый API делает вызов для доступа к аппаратным средствам устройства, система Android загружает библиотечный модуль для этого аппаратного компонента.

Для устройств, работающих под управлением Android версии 5.0 (уровень API 21) или выше, каждое приложение запускается в своем собственном процессе и с собственным экземпляром Android Runtime (ART). ART создан для запуска нескольких виртуальных машин на устройствах с низкой памятью, выполняя DEX-файлы, формат байт-кода, разработанный специально для Android, оптимизированный для потребления минимального объема памяти.

Android также включает в себя набор основных библиотек времени исполнения, которые обеспечивают большую часть функциональных возможностей языка программирования Java, включая некоторые функции языка Java 8, которые использует инфраструктура Java API.

3.5 Общая архитектура приложения

Экосистема средств разработки под Android развивается очень быстро. Каждую неделю создаются новые инструменты, обновляются существующие библиотеки, появляются новые возможности. Следовательно, и общая архитектура приложений развивается вслед за ними. В течение прошлого года в Android-сообществе начали набирать популярность отдельные архитектурные шаблоны, так как MVP, или MVVM.

Внимательно изучив данные шаблоны, можно прийти к выводу, что в целом они достаточно просты и схожи. Но основным отличием является использование MVVM механизма связывания данных, позволяющего автоматически изменять представление изменением 48 состояния View-модели. В случае MVP изменение уровня представления ложится на плечи разработчика.

Ввиду того, что перед выводом на экран данные необходимо обработать, мы будем придерживаться архитектурного шаблона MVP (рисунок 8).

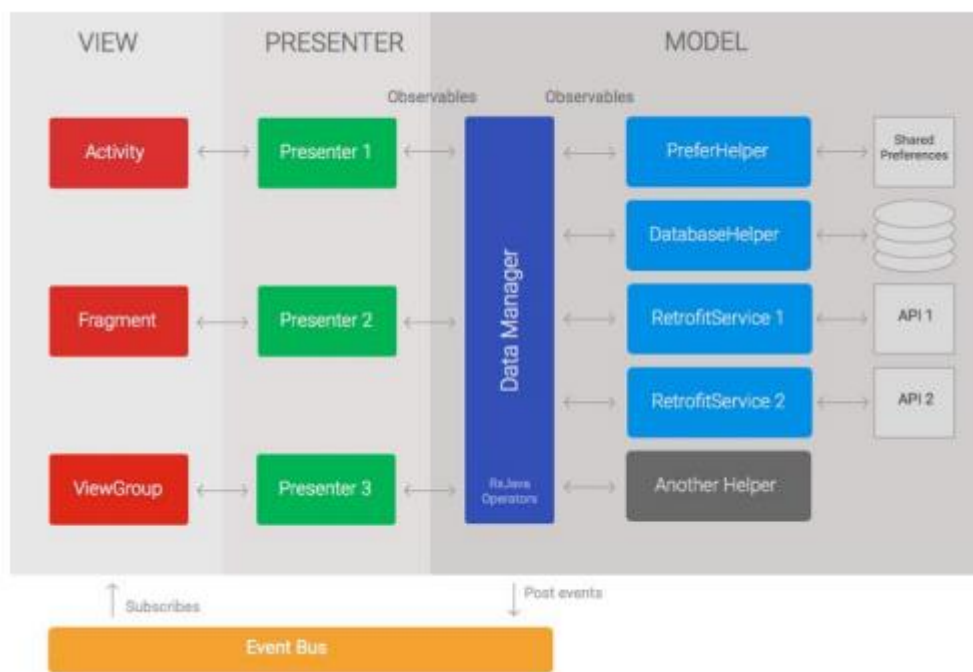


Рисунок 8 – Архитектурный шаблон MVP

4 Описание разработанного интерфейса

Интерфейс пользователя представляет собой представление, которое пользователь видит в первую очередь – это карта (рисунок 9). Как только загружается карта, запускается алгоритм определения текущего местоположения пользователя.

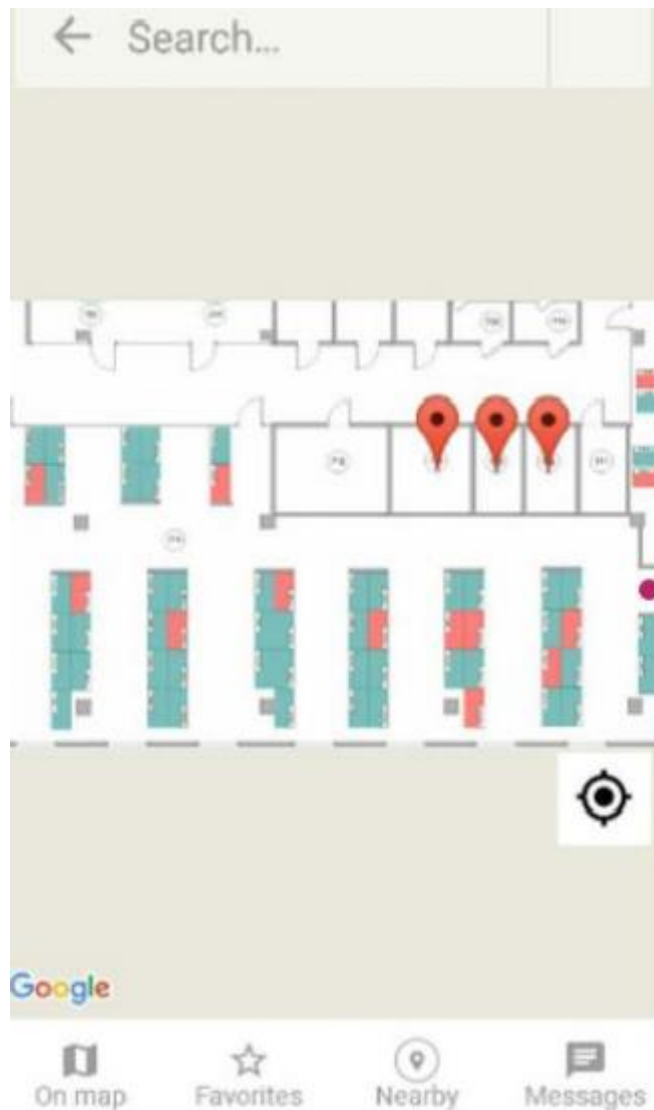


Рисунок 9 – Начальный экран

Также, в приложении есть функционал, позволяющий выбирать пройденные маршруты, а также осуществлять поиск и построение маршрута от текущего местоположения до интересующего пользователя места. Построение маршрута на рисунке 10.

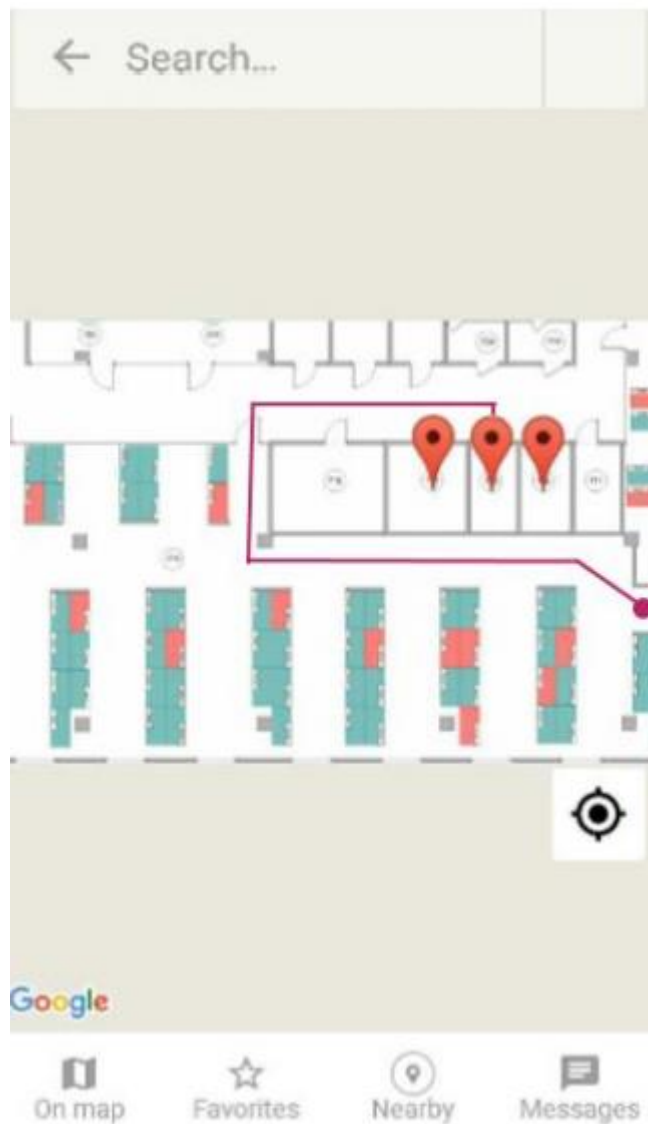


Рисунок 10 – Построенный маршрут

По достижении пользователя конечной точки, на экране возникает сообщение о том, что пользователь достиг конца маршрута. Приложение также имеет следующие возможности:

- возврат к текущему местоположению пользователя;
- просмотр пройденных маршрутов;
- масштабирование карты.

В дальнейшем планируется расширить дополнительный функционал такими возможностями, как чат, более точный пройденный маршрут с дополнительными фильтрами, поиск мест на карте.

ЗАКЛЮЧЕНИЕ

В процессе прохождения технологической (проектно-технологической) практики:

- проведен анализ имеющихся современных языков, систем и технологий программирования, а также технологий проектирования программного и (или) аппаратного обеспечения для решения задач Бесплатформенная инерциальная навигация;

- проведена разработка проекта автоматизированной информационной системы Бесплатформенная инерциальная навигация;

- сделано описание способов организации проектных данных и используемых нормативно-технических документов, предметной области для задач Бесплатформенная инерциальная навигация;

- разработано программное и аппаратное обеспечения для автоматизированной информационной системы Бесплатформенная инерциальная навигация;

- разработан интерфейс системы Бесплатформенная инерциальная навигация,

- сделано описание используемых библиотек и зарубежных комплексов обработки информации.

Таким образом в процессе прохождения практики были освоены все необходимые индикаторы ОПК-5.2, ОПК-6.2, ОПК-7.2, ОПК-8.2 компетенций ОПК-5, ОПК-6, ОПК-7, ОПК-8.

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-yazykov-programmirovaniya-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-novyy-yazyk-programmirovaniya-sposobnyy-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).