МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования   
«САНКТ–ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой №\_\_\_

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

БАКАЛАВРСКАЯ РАБОТА

|  |  |
| --- | --- |
| на тему |  |
|  | |
|  | |

|  |  |
| --- | --- |
| выполнена |  |
| фамилия, имя, отчество студента в творительном падеже | |

|  |  |  |  |
| --- | --- | --- | --- |
| по направлению подготовки | .03. |  |  |
|  | код |  | наименование направления |
|  | | | |
| наименование направления | | | |
| направленности |  |  |  |
|  | код |  | наименование направленности |
|  | | | |
| наименование направленности | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент группы № | 4831 |  |  |  | К.А. Корнющенков |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Руководитель

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

Оглавление

[ВВЕДЕНИЕ 5](#_Toc102999393)

[1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc102999394)

[1.1 Наименование программного изделия 6](#_Toc102999395)

[1.2 Назначение и область применения 6](#_Toc102999396)

[2. Обоснование и описание выбранных технологий разработки 8](#_Toc102999397)

[2.1 Язык разработки Swift 8](#_Toc102999398)

[2.1.1 История создания 8](#_Toc102999399)

[2.1.2 Особенности Swift 8](#_Toc102999400)

[2.2 Облачная база данных Firebase 9](#_Toc102999401)

[3. Что-то 3 10](#_Toc102999402)

[ЗАКЛЮЧЕНИЕ 11](#_Toc102999403)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 12](#_Toc102999404)

[ПРИЛОЖЕНИЕ 1 - Листинг программы 13](#_Toc102999405)

[ПРИЛОЖЕНИЕ 2 - Руководство пользователя 14](#_Toc102999406)

[ПРИЛОЖЕНИЕ 3 - Руководство программиста 15](#_Toc102999407)

* **Введение** - это краткое изложение мотивации написания диплома и поставленной задачи. Шаблон содержания: актуальность темы – обзор аналогов – вывод о необходимости разработки новой программы. Рекомендуемая структура введения должна содержать порядка двух страниц с описанием проблематики, затем анализ имеющихся аналогов и вывод о необходимости разработки нового программного обеспечения. Этот последний пункт обязателен - иначе непонятно, зачем городить огород и создавать что-то новое, если можно взять уже имеющийся продукт.
* **Первый раздел** имеет смысл посвятить расширенному описанию предметной области. Всегда нужно помнить, что Вас готовили не только, как кодировщика, но и как проектировщика и аналитика. Старайтесь показать, что Вы способны разобраться в предметной области, не связанной с IT. В этом же разделе имеет смысл расширить постановку задачи и описать требования к системе.
* **Второй раздел** имеет смысл посвятить проекту приложения. В проекте используйте диаграммы UML и ER. Описывая реализацию, можно приводить НЕБОЛЬШИЕ осмысленные фрагменты кода с пояснениями, что они делают
* **Третий раздел** обычно содержит в себе описание экранных форм приложения со скриншотами и руководством пользователя.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

ВВЕДЕНИЕ

1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

2. РАЗРАБОТКА АЛГОРИТМА ПОСТРОЕНИЯ ГИБКОЙ ДИАГНОСТИЧЕСКОЙ ПРОЦЕДУРЫ АНАЛИЗА ТЕХНИЧЕСКОГО СОСТОЯНИЯ ОБЪЕКТА, ОПТИМАЛЬНОЙ ПО КРИТЕРИЮ МАКСИМУМА ЦЕННОСТИ ИНФОРМАЦИИ

2.1. Схема алгоритма построения гибкой диагностической процедуры анализа технического состояния объекта, оптимальной по критерию максимума ценности информации при использовании дискретных диагностических признаков

2.2. Схема алгоритма построения гибкой диагностической процедуры анализа технического состояния объекта, оптимальной по критерию максимума ценности информации при использовании непрерывных диагностических признаков

3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ГИБКОЙ ДИАГНОСТИЧЕСКОЙ ПРОЦЕДУРЫ АНАЛИЗА ТЕХНИЧЕСКОГО СОСТОЯНИЯ ОБЪЕКТА, ОПТИМАЛЬНОЙ ПО КРИТЕРИЮ МАКСИМУМА ЦЕННОСТИ ИНФОРМАЦИИ

3.1. Выбор языка программирования и среды разработки

3.2. Обоснование и описание выбираемых методов и алгоритмов разработки

3.3. Описание структур данных

3.4. Разработка интерфейса пользователя

3.5. Анализ эффективности разработанной программы

ЗАКЛЮЧЕНИЕ

# ВВЕДЕНИЕ

# ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1 Наименование программного изделия

Приложение по распознаванию достопримечательностей Санкт-Петербурга.

## 1.2 Назначение и область применения

«Приложение по распознаванию достопримечательностей Санкт-Петербурга» может использоваться туристами, жителями города и прочими заинтересованными лицами для решения следующих задачи:

1. Распознавать и определять достопримечательность города по фотографии, которая выбрана пользователем мобильного приложения из галереи мобильного устройства.
2. Распознавать и определять достопримечательность города по наведению главной камеры мобильного девайса на исторический объект.
3. Сокращение времени опознавания достопримечательности.
4. Получение других фотографий для распознанной достопримечательности.
5. Получение краткой исторической справки, распознанного исторического объекта.
6. Отобразить найденную достопримечательность на карте (toDo добавить)

# Обоснование и описание выбранных технологий разработки

## 2.1 Язык программирования Swift

Swift – язык программирования общего назначения, который создан прежде всего для разработки приложения на IOS и macOS.

### 2.1.1 История создания

Язык программирования был представлен общественности в 2014 года на ежегодной всемирной конференции для разработчиков на платформах Apple – Worldwide Developers Conference, которая проводится ежегодно в Apple Park.

Swift был заложен в платформе Next, поэтому в языке можно часто встретить базовые классы, особенно при взаимодействии с Obj-c, которые начинаются с приставки NS, такие как  NSString, NSNumber, NSObject и другие. Разработкой текущий версии языка занимались сотрудника Apple еще с 2010 года. Данный язык не имеет так такового одного «прородителя». Создатели взяли лучшее от разных языков, например, Obj-c, Haskell, Python, С#, С и других.

### 2.1.2 Особенности Swift

Swift является молодым языком программирования, но при этом активно развивается, в нем учтены многие минусы, которые свойственны более старым языкам программирования. Например, Swift хорошо подходит в качестве первого языка программирования. При разработке ЯП создатели преследовали такие цели, как простота изучения и ускорения процесса разработки. Данный язык получил простой и лаконичный синтаксис.

Один из главных преимуществ Swift – высокая производительность. Согласно тестам на производительность язык в 3 раза быстрее своего главного конкурента Obj-c и почти в 8 раз быстрее, чем Python. Помимо скорости Swift обладает хорошим выбором современных библиотек, которые позволяют решать разнообразные задачи начиная от написания базовых клиент-серверных приложения и заканчивания написанием ML и AR.

Но, к сожалению, у языка есть и недостатки. Одним из таких является взаимодействие с Obj-c, для которого используется специальный «мост», который сильно тормозит процесс сборки приложения. Так проект с каким-нибудь крупным приложением спокойно может собираться на 30-40 минут на современных компьютерах.

## 2.2 Облачная база данных Firebase

## Firebase – облачная база данных, которая позволяет хранить и получать сохраненную информацию, а также предоставляет удобный интерфейс для взаимодействия с данными.

### 2.2.1 Преимущества Firebase

ываыв

# Что-то 3

# ЗАКЛЮЧЕНИЕ

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. ISO/IEC/IEEE 29148:2018 International Standard — Systems and software engineering — Life cycle processes — Requirements engineering
2. Виггерс, Карл. Разработка требований к программному обеспечению = Software Requirements: пер. с англ.; 3-е издание, дополненное / Карл Виггерс, Джой Битти — М.: Издательство «Русская редакция», 2014. — 736 с.: ил.
3. IEEE Recommended Practice for SRS / IEEE Std 830-1998 (Revision of IEEE Std 830-1993) [Электронный ресурс]. — University of Alaska Anchorage, 2020 — URL: <http://www.math.uaa.alaska.edu/~afkjm/cs401/IEEE830.pdf>
4. Writing Software Requirements Specifications (SRS) [Электронный ресурс]. — INKtopia Limited, 2020 — URL: <https://techwhirl.com/writing-software-requirements-specifications/>
5. Software Requirements Specification Helps to Protect IT Projects From Failure [Электронный ресурс]. — Belitsoft, 2004-2020 — URL: <https://belitsoft.com/php-development-services/software-requirements-specification-helps-protect-it-projects-failure>

# ПРИЛОЖЕНИЕ 1 - Листинг программы

# ПРИЛОЖЕНИЕ 2 - Руководство пользователя

# ПРИЛОЖЕНИЕ 3 - Руководство программиста