МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»**

**Московский техникум космического приборостроения**

### СПЕЦИАЛЬНОСТЬ: 09.02.07 Информационные системы и программирование (квалификация «Администратор баз данных»)

**П О Я С Н И Т Е Л Ь Н А Я З А П И С К А**

**к курсовому проекту по теме:**

## Использование фреймворков для безопасного программирования биометрической системы идентификации личности

|  |  |  |
| --- | --- | --- |
| Руководитель разработки от техникума | (подпись, дата) | Гончаренко С.В. |
| Разработчик | (подпись, дата) | Симонян П. Р. |

Москва 2021

**«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»**

**Московский техникум космического приборостроения**

УТВЕРЖДАЮ

Председатель ПЦК спец. 09.02.07

Н.А. Жилкина

« » 2021г.

**З А Д А Н И Е**

### на выполнение курсового проекта

по профессиональному модулю «ПМ.04 Сопровождение и обслуживание программного обеспечения компьютерных систем»

Студент Симонян П.Р. ТИП-51

(фамилия, инициалы, индекс группы)

Руководитель Гончаренко С.В.

(фамилия, инициалы)

График выполнения работы: 25% к 4 нед., 50% к 8 нед., 75% к 12 нед., 100% к 15 нед.

1. ***Тема курсового проекта***

Использование фреймворков для безопасного программирования биометрической идентификации личности

1. ***Техническое задание***

Сравнить возможности фреймворков для безопасного программирования биометрической идентификации личности

1. ***Оформление курсового проекта***
   1. Расчетно-пояснительная записка на листах формата А4.
   2. Перечень графического материала КП (плакаты, схемы, чертежи и т.п.) – схемы алгоритма

Дата выдачи задания «\_ » 2021г.

Руководитель курсового проекта Гончаренко С.В.

# СОДЕРЖАНИЕ

[Введение 4](#_TOC_250005)

1. [Постановка задачи 5](#_TOC_250004)
2. [Теоретические основы 5](#_TOC_250003)
   1. Облачные вычисления 5
   2. Интеграция приложений в Amazon Web Services 5
      1. Сервисы интеграции приложений Amazon Web Services 5
      2. Примеры использования интеграции приложений 5
      3. Перемещение эксабайтов данных с AWS Snowmobile 5
   3. Миграция данных с помощью Amazon Web Services 5
      1. Решения Amazon Web Services для миграции 5
   4. Передача данных в Amazon Web Services 5
      1. Перемещение эксабайтов данных с AWS Snowmobile 5
      2. Работа с потоковыми данными в Amazon Kinesis Streams 5
3. [Практическая реализация 5](#_TOC_250002)
   1. Создание таблицы NoSQL и выполнение запросов к ней 5
   2. Настройка документной базы данных 5
   3. Хранение и извлечение файла с помощью Amazon S3 5
4. Тестирование 5
   1. План тестирования 5
   2. Тест-кейсы 5
   3. Баг-репорты 5
   4. Отчёт о тестировании 5
5. [Результаты 5](#_TOC_250001)
   1. Создание бесплатного аккаунта Amazon Web Services 5
   2. Результаты создания таблицы NoSQL и выполнения запросов к ней 5
   3. Результаты настройки документной базы данных 5
   4. Результаты хранения и извлечения файла с помощью Amazon S3 5
   5. Руководство пользователя 5

[Заключение 5](#_TOC_250000)

Приложение А. Руководство пользователя по созданию таблицы NoSQL и выполнения запросов к ней 5

Приложение Б. Руководство пользователя по настройке документной базы данных 5

Приложение В. Руководство пользователя по хранению. и извлечению файла с помощью Amazon S3 5

Приложение Г. Отчёт о тестировании 5

# ВВЕДЕНИЕ

Благодаря развитию аппаратного обеспечения, в XXI веке большую популярность обрели биометрические датчики, которые позволяют сделать взаимодействие с различными устройствами проще и безопаснее. В отличие от классических данных, используемых при аутентификации, таких как пароль или электронные ключи (токенов), биометрические данные не могут быть переданы, так как индивидуальны для каждого человека. Их подделка и подбор представляются крайне тяжёлыми...

Существует множество фреймворков для реализации функций биометрических сканеров и анализа собранной информации. В данной работе будут рассмотрены самые популярные из них, а также будет проведено их сравнение.

# ПОСТАНОВКА ЗАДАЧИ

Целью курсового проекта является сравнение возможностей фреймворков для безопасного программирования биометрической идентификации личности.

Основания для проведения работы: учебный план специальности, рабочая программа дисциплины, распоряжение по учебному заведению.

Требования к курсовому проекту: требования к структуре документов определены соответствующими стандартами ЕСКД и КСПД; требования к оформлению определены соответствующими методическими указаниями.

Порядок контроля и приёмки курсового проекта: контроль выполнения курсового проекта проводится руководителем поэтапно в соответствии с утверждённым графиком выполнения курсового проекта; на завершающем этапе руководитель осуществляет нормоконтроль представленной исполнителем документации и принимает решение о допуске (недопуске) курсового проекта к защите.

При работе над курсовым проектом учитывать: полноту раскрытия рассматриваемой темы, актуальность выбираемых фреймворков, корректность их сравнения.

Плановые сроки выполнения: 1 семестр 2021 учебного года.

Исходные данные: в курсовом проекте необходимо провести сравнение на основе официальной документации фреймворков и подтверждённой информации, находящейся в открытом доступе.

Состав курсового проекта:

* Текстовые и графические документы;
* Программная и технологическая документация (при необходимости);
* Пояснительная (расчётно-пояснительная) записка;
* Обязательные структурные элементы: титульный лист (одна страница), содержание (от одной страницы), ведение (от одной страницы), список литературы (от одной страницы), заключение (от одной страницы),
* Объём основной части работы: минимум два раздела с минимум двумя параграфами (от пяти страниц) в каждом разделе.

Содержание пояснительной записки:

* Титульный лист;
* Лист задания;
* Содержание;
* Введение;
* Исходные данные;
* Основной раздел (теоретическая, аналитическая и проектно-аналитическая часть);
* Заключение;
* Список использованной литературы;
* Приложения.

Детализированное техническое задание на курсовой проект:

* во введении обосновать актуальность исследуемой темы, степень её изученности, дать критический обзор работ, характеризующих сущность рассматриваемого вопроса, определить цель и задачи, описать применяемые методы исследования;
* в теоретической части изложить теоретические концептуальные основы исследуемой проблемы и собственное понимание существа вопросов исследуемой темы, сформировать своё аргументированное мнение по неоднозначным вопросам;
* в аналитической части изложить особенности функционирования объекта исследования, проанализировать и раскрыть особенности сбора, обработки и подготовки информации о процессах, целесообразно рассмотреть порядок формирования и представления информации об отдельных объектах исследования;
* в проектно-аналитической части разработать и внести предложения по рациональной методологии исследования объекта с использованием современных информационных технологий.
* в заключении изложить основные результаты исследования, сформулировать выводы, предложения и перспективы дальнейшей разработки темы.

# ТЕОРЕТИЧЕСКИЕ ОСНОВЫ

**Биометрия** - это идентификация человека по уникальным биологическим признакам.

В современные устройства и специализированные средства защиты биометрия встраивается все чаще по следующим причинам:

1. Оказывается минимальная когнитивная нагрузка. Для проведения идентификации требуется совершить интуитивные, не требующие умственных усилий или запоминания, действия: коснуться сканера, посмотреть в камеру, произнести фразу.
2. С приходом в массово - потребительские устройства биометрия стала гораздо доступнее и востребованнее.
3. Биометрические сканеры не всегда работают безотказно, но все же остаются достаточно безопасным методом идентификации личности.
4. Тенденция усиления защиты приватности удачно сочетается с тем фактом, что биометрические данные невозможно передать и крайне тяжело подделать.

Остановимся подробнее на последнем пункте.

**Биометрические данные** - уникальные биологические признаки людей. Самыми востребованными являются:

1. Отпечатки пальцев
2. Сетчатка глаза
3. Голос

В отличие от паролей, которые сейчас используются повсеместно, биометрические данные обладают одинаковой сложностью. Это вносит дополнительную защиту, так как пользователь не может сделать этот способ идентификации личности менее надёжным, как если бы он придумал недостаточно сложный и легко угадываемый пароль.

Помимо этого, биометрические данные гораздо тяжелее украсть (или, в случае биометрии, подделать). Скопировать их представляется возможным только для специально обученных людей.

Однако возникают сложности, которые все же не дают гарантию безопасности данного метода идентификации:

1. Все же существуют люди с очень похожими биометрическими признаками. Например, близнецы или люди с похожим тембром голоса.
2. Ключевую роль в обеспечении безопасности и точности биометрии берет на себя разработчик биометрических датчиков и программного обеспечения для них.

Следует учитывать, что надёжность биометрии, в отличие от обычных паролей, сильнее зависит от качества аппаратного и программного продукта, которое обеспечивает её. Потому решения сильно отличаются по качеству, удобству и цене. Продукты для государственных структур, требующие исключительную надёжность и подразумевающие усиленную нагрузку, будут сильно отличаться от решений, используемых в современных смартфонах. Конкретные различия данных продуктов будут рассмотрены в контексте сравнения.

**Фреймворк** - это программный продукт, упрощающий создание и поддержку технически сложных и нагруженных проектов. Обычно задачу программирования биометрических датчиков, методов сбора и обработки информации, полученных с их помощью, берут на себя крупные и проверенные компании или разработчики данных сканеров. Примером вторых могут послужить решения компании BioLink, которая сама выпускает биометрические датчики и комплекты для разработки программного обеспечения (SDK) под них. Но чаще всего встречаются SDK компаний, агрегирующих аппаратное обеспечение разных производителей, чтобы предоставить разработчикам общий интерфейс для их использования. Такими являются Google и Microsoft.

Таким образом, фреймворки способствуют:

А) Усиленной безопасности. Перед тем, как перейти в использование конечным пользователям (в данном случае, это пользователи фреймворков - разработчики), решения для программирования биометрии тщательно тестируются организациями, которые специализируются на этом. В других случаях, такие компании, как, например, Microsoft, имеют все средства для того, чтобы провести качественное тестирование собственных программных решений и предоставить разработчикам безопасный программный продукт.

Б) Облегчённой интеграции. Решения для программирования биометрии, как правило, хорошо задокументированы и не требуют углублённых знаний программиста о работе биометрических сканеров. Ещё проще дела обстоят у разработчиков для массовых платформ: Apple, Google, Microsoft и некоторые другие компании предлагают единые решения для разработчиков под их экосистемы. Пользователю их фреймворков гораздо меньше приходится принимать во внимание разнообразие биометрических датчиков, используемых в конкретных устройствах разных производителей или моделей.

В) Популяризации усиленной безопасности с помощью биометрии. Использование особенностей современных биометрических датчиков может позволить себе любой разработчик, который должен лишь использовать готовое решение в виде фреймворка и изучить документацию к нему. Благодаря этому биометрия пришла в банковскую сферу, сферу онлайн - платежей, а также в другие массовые продукты, предъявляющие специальные требования к безопасности.

Теперь, разобравшись в преимуществах и важности биометрии, стоит рассмотреть некоторые самые популярные решения, которые на данный момент доступны разработчикам.

OpenBR - фреймворк с открытым исходным кодом, позволяющий распознавать лица, возраст и пол по фотографии.

NEUROtechnology Megamatcher SDK - Дорогой профессиональный продукт, ориентированный на крупные компании и государственные структуры, которые предъявляют к безопасности и отказоустойчивости особые требования. Введён в список сравнения для того, чтобы дать представление о том, чем профессиональные решения отличаются от массово - потребительских.

Huawei FIDO (Fast Identity Online) BioAuthn - Фреймворк для использования биометрии на смартфонах и планшетах компании Huawei.

Samsung Knox SDK - Масштабный фреймворк компании Samsung, включающий в себя возможность идентификации личности по отпечатку пальца или радужной оболочки глаза.

Apple Local Authentication Framework - Фреймворк, позволяющий использовать биометрию на всех устройствах компании Apple, аппаратно поддерживающих её: от iPhone до iMac с использованием соответствующей клавиатуры.

Google’s Android Biometric Library - Фреймворк для использования биометрии на устройствах под операционной системой Android.

Microsoft Windows Biometric Framework - Фреймворк для использования биометрии в операционной системе Windows с поддержкой самых разных производителей: от Lenovo до Honor.

Такое разнообразие программных решений определяется не только различием платформ для разработки, но и требованиями конечного пользователя. Например, пользователь смартфона предъявляет требования как к безопасности, так и к удобству и скорости считывания биометрических данных, не говоря уже о цене. Поэтому производители аппаратных и программных решений стараются балансировать между этими характеристиками, пока более профессиональные решения не представляются доступными для использования в массовых продуктах. Развитие в этом направлении происходит постоянно. Некоторое время назад использование достаточно надёжных датчиков отпечатка пальца в бюджетных устройствах было невозможно, но с развитием технологии данная технология стала обыденностью. Как и считывание топологии лица с помощью лазерного сканирования раньше было фантастикой, а сейчас доступно в мобильных устройствах высокого класса, а позже, вероятно, станет такой же обычной практикой, как и датчик отпечатка пальца сейчас.

С другой стороны, биометрия уже давно используется в специализированных решениях обеспечения безопасности. Так, систему Megamatcher используют правительства и силовые структуры некоторых стран, силовые и правоохранительные органы, предприятия, имеющие государственную важность, в том числе аэропорты и банки. Компромиссные и доступные решения не удовлетворяют высоким требованиям данных организаций. Поэтому именно в таких сферах встречаются инновационные и передовые решения, вроде массового распознавания лиц по видео, распознавание особых биометрических признаков вроде вен, отпечатков ладоней, технологии классификации распознанных людей по различным признакам, моментальная идентификация людей для поиска разыскиваемых преступников, безотказные решения, которые могут использоваться для идентификации личности при совершении важных финансовых операций. Ошибки, возможные при идентификации обычного пользователя смартфона, на промышленном и государственном уровне по очевидным причинам просто недопустимы.

Выделим следующие характеристики, которые позволят сравнить данные фреймворки:

1. **Доступные платформы.**

Какие платформы и операционные системы способны взаимодействовать с биометрическими данными через данный фреймворк?

1. **Доступные для сканирования биометрические данные.**

С какими данными может работать фреймворк? Среди них: данные отпечатков пальцев, ладоней, голоса, сетчатки глаза, рисунка вен.

1. **Доступность фреймворка.**

На каких языках программирования осуществляется использование фреймворка? Какие инструменты необходимо приобрести для разработки на нём? Какие представляются возможности отладки фреймворка?

1. **Поддерживаемые модели сканеров.**

Для какого аппаратного обеспечения предназначен фреймворк?

1. **Качество документации.**

Оценивается полнота и актуальность документации, наличие примеров и готовых шаблонов проектов, учебников.

1. **Доступность документации.**

Локализована ли документация? Имеются ли онлайн и оффлайн версии документации?

1. **Дороговизна разработки на данном фреймворке.**

Каких вложений требует минимальная разработка и отладка программ, использующих данный фреймворк?

1. **Уровень поддержки разработчиком фреймворка.**

Чьими силами и как обеспечивается поддержка и совместимость фреймворка в соответствии с развитием биометрии?

1. **Лицензирование.**

По какой лицензии распространяется фреймворк?

Для структурирования информации о фреймворках, было решено поделить их характеристики на три группы:

1. **Функциональные.** Это те характеристики, которые определяют реализацию функций фреймворка в конечных продуктах. Сюда относятся доступные платформы (пункт 1 предыдущего списка), доступные биометрические данные (пункт 2), доступность фреймворка (пункт 3), доступные модели сканеров (пункт 4).
2. **Уровень поддержки документации.** Данные характеристики определяют качество документации и удобство поддержки конечных продуктов, использующих конкретный фреймворк. Сюда относятся: качество документации (пункт 5) и доступность документации (пункт 6).
3. **Финансовая обоснованность использования.** Здесь подразумевается перспективность использования фреймворка. Например, хорошим показателем будет активная поддержка фреймворка или открытость, позволяющая модифицировать его в соответствии с нуждами разработчика. Сюда относится дороговизна разработки на данном фреймворке (пункт 7), уровень поддержки разработчиком (пункт 8) и лицензирование (пункт 9).

# ПРОВЕДЕНИЕ СРАВНЕНИЯ

Изучив все теоретические сведения и определив критерии оценивания фреймворков, можно представить их сравнительную характеристику в виде трёх таблиц, описывающих разные группы характеристик.

Таблица 3.1 - сравнение функциональных характеристик фреймворков.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Фреймворк** | **Доступные платформы** | **Доступные биометрические данные** | **Доступность фреймворка** | **Доступные модели сканеров** |
| OpenBR | Windows, Mac OS X, Debian Linux. | Фотография лица | Для использования OpenBR, его необходимо собрать с помощью CMake. Также необходимо наличие Qt и OpenCV при сборке.  API фреймворка доступен на языке C++, а также «обёртка», как её называют разработчики в официальной документации, в виде Python API.  Отладка фреймворка может производиться на устройстве разработчика, биометрические датчики не требуются. | Фреймворк поддерживает обработку лиц на любых фотографиях. Следовательно, он требует лишь фотографию, содержащую различимое лицо, а не специализированное оборудование. |
| Megamatcher SDK | Windows, Mac OS, Linux, Android, IOS, серверные конфигурации Microsoft Linux | Отпечаток пальца, радужная оболочка, лицо (фотография и 3D модель), голос, отпечаток ладони | Фреймворк необходимо приобрести. Доступен пробный период в 30 дней.  Поддерживаются следующие языки программирования: C#, C++, Java.  Отладка приложения производится на целевой платформе с использованием требующихся биометрических датчиков. | Фреймворк поддерживает десятки камер, датчиков отпечатков пальцев и ладоней. Поддержка также включает в себя биометрические датчики в устройствах на Android и IOS без покупки дополнительного профессионального оборудования. Поддерживается любой микрофон для идентификации по голосу или идентификация по звуковому файлу, содержащему голос. |
| Huawei FIDO BioAuthn | EMUI (смартфоны и планшеты Huawei) | Отпечаток пальца, лицо (фотография или 3D модель) |  |  |
| Samsung Knox SDK | OneUI (смартфоны и планшеты Samsung) |  |  |  |
| Apple Local Authentication Framework | iOS, iPadOS, macOS, Mac | Отпечаток пальца, 3D модель лица |  |  |
| Android Biometric Library | Android 9+ | Отпечаток пальца, лицо (фотография или |  |  |
| Windows Biometric Framework | Windows 10+ | Фотография лица, отпечаток пальца |  |  |

Таблица 3.2 - сравнение уровня поддержки документации фреймворков.

|  |  |  |
| --- | --- | --- |
| **Фреймворк** | **Качество документации** | **Доступность документации** |
| OpenBR | Документация проекта покрывает все возможности фреймворка, включает в себя небольшое пошаговых учебников, подробное описание функций API, готовые примеры программ, руководство по установке фреймворка и всех его зависимостей с нуля. Документация содержит пояснительные изображения. | Документация доступна только на английском языке. Имеется онлайн версия, а оффлайн документация включена в открытый репозиторий GitHub, так что при установке фреймворка она в любом случае будет доступна в виде .md файлов. |
| Megamatcher SDK | Документация фреймворка насчитывает 2859 страниц с подробным описанием всех аспектов работы фреймворка, изображениями, примерами на разных языках программирования, описанием конфигурации баз данных MS SQL Server, MySQL, Oracle, PostgreSQL, SQLite. Нужно отметить полноту документации и удобство ориентирования по ней, а также наличие множества обучающих разделов. | Документация доступна только на английском языке. Файл документации хранится в PDF формате и доступен для скачивания. |
| Huawei FIDO BioAuthn | Документация содержит руководство по всем этапам разработки приложения с биометрией и видеоролики, обозревающие возможности фреймворка. Доступен форум разработчиков. | Документация доступна онлайн на 9 языках включая русский. |
| Samsung Knox SDK |  |  |
| Apple Local Authentication Framework |  |  |
| Android Biometric Library | Документация размещена на глобальной платформе, которая также имеет форум, множество примеров проектов и обучающие видеоролики. | Документация доступна онлайн на 7 языках. Русский язык доступен с помощью встроенного переводчика. |
| Windows Biometric Framework |  |  |

Таблица 3.3 - сравнение финансовой обоснованности использования фреймворков.

|  |  |  |  |
| --- | --- | --- | --- |
| **Фреймворк** | **Дороговизна разработки на фреймворке** | **Уровень поддержки разработчиком** | **Лицензирование** |
| OpenBR | Фреймворк распространяется бесплатно. Это значит, что стоимость разработки на нём определяется только сложностью проекта, а тестирование может проходить на машине разработчика. | Активная стадия разработки фреймворка завершилась в 2015 году, но до сих пор разработчик выполняет оптимизацию и улучшает разработку, публикуя изменения на GitHub. | Фреймворк распространяется под лицензией Apache 2.0. Разработчик не даёт никаких гарантий безотказности и надёжности продукта. |
| Megamatcher SDK | Цены на пожизненную лицензию начинаются от 216 тысяч рублей. Ценовая политика актуальна для крупных компаний и государственных организаций. | Фреймворк обновляется по сей день. Разработчик оказывает индивидуальную поддержку заказчикам и потенциальным покупателям. | Лицензия покупается один раз и на всегда для всех типов используемых компонентов и платформ, на которых планируется использование биометрии. Возможен перенос или обновление лицензии в случае обновления аппаратного обеспечения. SDK покупается один раз на всю компанию, которая собирается интегрировать продукт. |
| Huawei FIDO BioAuthn |  |  |  |
| Samsung Knox SDK |  |  |  |
| Apple Local Authentication Framework |  |  |  |
| Android Biometric Library | Фреймворк бесплатный. Для отладки необходимо устройство на базе Android 9+ или его эмулятор. |  |  |
| Windows Biometric Framework |  |  |  |

# 5. ТЕСТИРОВАНИЕ

* 1. План тестирования

План тестирования (тест-план) – это документ, описывающий весь объем работ по тестированию, начиная с описания тестируемых объектов, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения. Тест-план является важной составляющей любого грамотно-организованного процесса тестирования, так как содержит в себе всю необходимую информацию, описывающую данный процесс.

Условия для плана тестирования:

* возможность решения задач по тестированию;
* построение стратегии тестирования;
* возможность вести учёт всех требуемых ресурсов, как технических, так и человеческих;
* планирование использования ресурсов на тестирование;
* просчёт рисков, возможных при проведении тестирования.

Чек-лист плана тестирования:

* скорость загрузки страниц (интерфейса ПО);
* адаптивность;
* язык и система исчислений, формат времени представлены в привычном формате;
* удобный шрифт и качество контента;
* основной функционал понятный и его не нужно искать;
* отсутствие горизонтальной полосы прокрутки;
* однообразие интерфейса, есть карта сайта;
* присутствие информации о компании и есть рабочее лого;
* нет раздражительных элементов для пользователей (например, реклама со звуком);
* страница 404 сообщает время и причину ошибки, содержит информацию об основных разделах и контактах;
* удобство пользования функционалом;
* удобство пользования функционалом:
* мгновенный скроллинг (кнопки вверх/вниз на длинных страницах);
* варианты выбора динамически изменяются (при начале ввода есть список доступных результатов, также есть возможность не выбирать из списка, если нет подходящего варианта);
* всё, что можно сделать программно, делается;
* присутствует удобный поиск на сайте или в приложении;
* удобство кнопок (нажимается сама кнопка, а не текст на ней или близлежащая область);
* отсутствуют яркие вызывающие цвета и цветной текст;
* цвет зависит от восприятия подсознательного значения (красный – ошибка, зеленый – всё хорошо);
* оптимальный размер активных элементов (ссылки, баннеры, кнопки);
* пояснения всплывают, где это необходимо (например, почему кнопка/ссылка неактивна, не нужно их совсем скрывать);
* строка поиска, контакты, подзаголовки на странице расположены по схеме в виде буквы F (так скользит взгляд пользователя);
* оформление ссылок в однообразных цветах и стиле по всему приложению;
* единственное поле поиска, помощь при отсутствии результатов;
* возможна авторизация через социальные сети;
* есть удобная обратная связь с подтверждением вопроса, отправкой ответа;
* модуль поддержки с человеком, который также отображает время ожидания ответа;
* разбивка на тематические категории.
* обязательные для ввода поля регистрациия пользователя выделены, минимальное, но необходимое количество данных;
* простота заполнения полей регистрациия пользователя (использование подсказок);
* проверка валидности данных ввода по заполнению до отправки формы;
* информационное сообщение об ошибке (также выделение нужного поля);
* функция просмотра пароля;
* подтверждение регистрации;
* есть возможность отказаться от рассылок.
* единообразие в размерах однотипных форм и цветов;
* полезность (изображение с информацией или элемент дизайна);
* высокое качество.

Тестирование проводилось параллелльно с практической реализацией.

* 1. Тест-кейс

Тест-кейс № 1 приведён в таблице № 1.

Таблица № 1

|  |  |
| --- | --- |
| Название | Тест доступности сервиса |
| Функция | Соединение |
| Действие: открыть сайт  htttps:// | Ожидаемый результат: сайт открыт и доступен.  Фактический результат: сайт открыт и доступен |
| Действие: возврат на шаг  назад | Ожидаемый результат: отсутствие ошибки соединения.  Фактический результат: отсутствие ошибки соединения |
| Результат | Тест пройден |

Тест-кейс № 2 приведён в таблице № 2.

Таблица № 2

|  |  |
| --- | --- |
| Название |  |
| Функция |  |
| Действие |  |
| Действие |  |
| Результат |  |

* 1. Баг-репорты

Баг-репорты отсутствуют, т. к. отсутствуют ошибки.

*или*

Баг-репорт № 1:

|  |  |
| --- | --- |
| Краткое описание | Авторизация на главной странице с использованием кириллицы работает не правильно |
| Проект | https://aws.amazon.com/ |
| Компонент приложения | Страница авторизации |
| Номер версии | 0.001 |
| Важность:  S1 Блокирующая (Blocker)  S2 Критическая (Critical) S3 Значительная (Major) S4 Незначительная (Minor) | S3 Значительная (Major) |

|  |  |
| --- | --- |
| S5 Тривиальная (Trivial) |  |
| Приоритет:  P1 Высокий (High) P2 Средний (Medium) P3 Низкий (Low) | P2 Средний (Medium) |
| Статус | Новая |
| Автор | Ф. И. О. |
| Назначен на | Должность «Программист» |
| Шаги воспроизведения | Открываем главную страницу сайта https://aws.amazon.com/=> слева страницы находим раздел: «авторизация»(см. копию экрана):    Введите логин (например, «user»). Нажмите кнопку «Вход» |
| Фактический результат | Запрос не прошел валидацию. (смотри копию экрана): |
| Ожидаемый результат | Поиск прошел удачно, описание требуемого показано верно. |

* 1. Отчёт о тестировании

В отчёте о тестировании суммированы цели и результаты тестирования (описанные в плане тестирования).

Отчёт о тестировании имеет следующую структуру.

* Обозначение продукта тестирования;
* Вычислительные системы, использованные при тестировании (технические средства, программные средства и их конфигурация);
* Дата окончания тестирования;
* Результаты тестирования описания продукта, документации, программ и данных;
* Перечень несоответствий требованиямл, либо перечень несоответствий рекомендациям, либо перечень не учтенных в продукте рекомендаций, либо формулировка того, что продукт не был протестирован на соответствие рекомендациям;
* Формулировка, что результаты тестирования относятся только к протестированным компонентам продукта;
* Формулировка, что полная копия отчета о тестировании не может быть изготовлена без письменного разрешения тетировшиков.

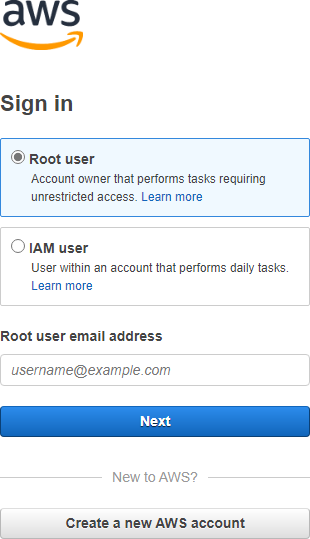
Отчёт о тестировании приведён в приложении Д.

# 5. РЕЗУЛЬТАТЫ

*Если сервис недоступен – раздел исключается.*

* 1. Создание бесплатного аккаунта Amazon Web Services

*Результаты создания бесплатного аккаунта Amazon Web Services с копиями экрана приведены в приложении А. Если сервис недоступен или аккаунт не требуется – пункт исключается.*



* 1. Результаты создания таблицы NoSQL и выполнения запросов к ней

*Результаты создания таблицы NoSQL и выполнения запросов к ней с копиями экрана приведены в приложении Б. Если сервис недоступен – пункт исключается.*

* 1. Результаты настройки документной базы данных

*Результаты создания таблицы NoSQL и выполнения запросов к ней с копиями экрана.*

*Если сервис недоступен – пункт исключается.*

* 1. Результаты хранения и извлечения файла с помощью Amazon S3

*Результаты хранения и извлечения файла с помощью Amazon S3 с копиями экрана. Если сераис недоступен – пункт исключается.*

* 1. Руководство пользователя

*Разработанное (разработанные) Руководство (руководства) пользователя.*

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование руководства пользователя** | **Приложение** |
| 1. | Руководство пользователя по созданию таблицы NoSQL и  выполнения запросов к ней | А |
| 2. | Руководство пользователя по настройке документной базы данных | Б |
| 3. | Руководство пользователя по хранению. и извлечению файла с  помощью Amazon S | В |

# ЗАКЛЮЧЕНИЕ

Целью разработки данного курсового проекта являлась разработка (подготовка) документации и отчётных форм для внедрения программных средств платформы Amazon Web Services интеграции распределённых приложений, миграции и передачи данных.

В ходе написания данной курсовой работы были получены практические навыки в области разработки (подготовки) документации и отчётных форм для внедрения программных средств платформы Amazon Web Services интеграции распределённых приложений, миграции и передачи данных.

Опыт, полученный при написании работы, подтвердил необходимость в правильной выработки требований к разработке (подготовке) документации и отчётных форм для внедрения программных средств платформы Amazon Web Services интеграции распределённых приложений, миграции и передачи данных.

В рамках курсового проектирования получен опыт создания среды AWS Cloud9 для работы с Amazon DocumentDB. Установлена оболочка Mongo, создан кластер Amazon DocumentDB, и подключение к нему, отправлено несколько запросов, вставлены данные и созданы запросы к документам JSON с помощью Amazon DocumentDB.

Создана резервная копия файла в облаке. Для этого создана корзинв Amazon S3 и отправлен в неё файл как объект S3. Amazon S3 обеспечивает надежность хранения объектов на уровне 99,999999999 %, что позволяет гарантировать доступ к ним в любой момент времени. Также извлечена сохраненная копия файла и удалиена корзина.

Созданы таблицы, запрошены таблицы и элементы, управление которыми осущетвлялось из консоли управления AWS.

Заявленные цели полностью достигнуты, однако, возможный недостаток, не влияющий на итоговый результат, проявляется в виде недочёта оформления поянительной записки.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 «Юрайт», 2020

2 «Юрайт», 2020

3 «Amazon», 2020

4 …….

1. https://
2. https://
3. https://
4. https://

9 ……

# ПРИЛОЖЕНИЕ А

## (обязательное)

Руководство пользователя по созданию таблицы NoSQL и выполнения запросов к ней

# ПРИЛОЖЕНИЕ Б

## (обязательное)

Руководство пользователя по настройке документной базы данных

# ПРИЛОЖЕНИЕ В

## (обязательное)

Руководство пользователя по хранению. и извлечению файла с помощью Amazon S3

# ПРИЛОЖЕНИЕ Г

## (обязательное) Отчёт о тестировании

### Наименование подсистемы

Сервис Amazon S3 облачной платформы Amazon Web Services.

### Краткая характеристика

Сервис. Amazon S3 – это безопасное, надежное облачное хранилище с большими возможностями масштабирования для объектов, составляющих статический веб-сайт. Примеры объектов, которые можно хранить: HTML-страницы, файлы CSS, изображения, видеофайлы и объекты JavaScript. Простой веб-интерфейс упрощает использование объектного хранилища Amazon S3 и обеспечивает передачу данных на хранение и их извлечение из любой точки Интернета. Это означает, что надежный доступ к сохраненным данным обеспечен для всех конечных пользователей.

### Вычислительные системы, использованные при тестировании

Компьютер . Операционная система . Программное обеспечение (например, браузер, Office).

### Дата тестирования

14 декабря 2020 г.

### Состав рабочей группы

1. Ф. И. О. (студент (студентка) группы -\_\_, специальность – « »)

### Выявленные в ходе тестирования ошибки (проблемы, замечания)

Ошибок (проблем, замечаний) не выявлено.

*или*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Ошибка** | **Описание** | **Способ устранения** | **Ошибка устранена (да/нет)** |
| 1 | Инстанс | Причины | Процедура восстановления |  |
|  | перестал | подобных | инстанса, который перестал |
|  | отвечать | проблем, как | отвечать на запросы, зависит от |
|  |  | правило, | типа инстанса (на основе EBS |
|  |  | зависят от | или хранилища инстансов). |
|  |  | особенностей | Сначала изучите информацию, |
|  |  | внутренней | которую инстанс выводит в |
|  |  | конфигурации | консоль, чтобы определить, |
|  |  | инстанса | почему перезагрузка повлияла на |
|  |  |  | инстанс. Иногда информации, |
|  |  |  | выведенной в консоль, |
|  |  |  | достаточно, чтобы определить |
|  |  |  | причину ошибки инстанса. При |
|  |  |  | работе в Консоли управления |
|  |  |  | AWS требуется сделать |
|  |  |  | следующее выберите нужный |
|  |  |  | инстанс; для просмотра |
|  |  |  | связанной информации в меню |
|  |  |  | «Instance Actions» выберите |
|  |  |  | пункт «View System Log». Если |
|  |  |  | вы используете инструменты API |
|  |  |  | Amazon EC2 выполните команду |
|  |  |  | ec2-get-console-output. |
| 2 | Потеряны | Чаще всего | В некоторых случаях служба |  |
|  | данные | восстановить | поддержки AWS Support может |
|  |  | данные в | восстановить часть данных при |
|  |  | хранилище | условии, что работа инстанса не |
|  |  | инстансов | была завершена и базовое |
|  |  | невозможно. | оборудование в порядке. Процесс |
|  |  |  | восстановления данных не |
|  |  |  | гарантирует положительного |
|  |  |  | результата и может занять |
|  |  |  | несколько дней, поэтому не |
|  |  |  | следует полагаться на |
|  |  |  | восстановление данных службой |
|  |  |  | поддержки AWS Support как на |
|  |  |  | основную стратегию резервного |
|  |  |  | копирования. |
| 3 | Ошибки | При переходе | Иногда ошибки корневого тома |  |
|  | корневого | инстанса в | для некорректно |
|  | тома | спящий режим | загружающегося инстанса на |
|  |  | сохраняются | основе EBS можно исправить |
|  |  | данные | вручную. Ручное исправление |
|  |  | оперативной | ошибок является сложным |
|  |  | памяти. В | процессом и не рекомендуется |
|  |  | случае | для пользователей, не |
|  |  | остановки | обладающих опытом системного |
|  |  | инстанс | администрирования. Примеры |
|  |  | отключается, а | действий, которые пользователи |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Ошибка** | **Описание** | **Способ устранения** | **Ошибка устранена (да/нет)** |
|  |  | данные | предпринимают на основе |  |
| оперативной | анализа данных консоли после |
| памяти | ошибки инстанса, включают |
| удаляются. В | выполнение команды fsck на |
| обоих случаях | соответствующем томе, |
| данные из | отключение модуля SELinux и |
| корневого | исправление ошибок в файле |
| тома EBS и | fstab. |
| любых |  |
| подключенных |  |
| томов данных |  |
| EBS |  |
| сохраняются. |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |

Результаты тестирования относятся только к протестированным компонентам продукта.

Полная копия отчета о тестировании не может быть изготовлена без письменного разрешения тетировшиков.