МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ижевский государственный технический университет имени М. Т. Калашникова»

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение»

Работа защищена с оценкой

«\_\_\_\_»

Дата \_\_\_\_\_\_\_\_\_\_

Подпись \_\_\_\_\_\_\_/\_\_\_\_\_\_\_

ОТЧЕТ ПО ПРЕДДИПЛОМНОЙ ПРАКТИКЕ

«Система управления учебным процессом. Серверная часть»

Выполнил:

Студент группы Б08-191-2 М.Л. Поздеев

Руководитель:

к.т.н., доцент кафедры ПО И.О. Архипов

Рецензия:

степень достижения поставленной цели работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

полнота разработки темы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

уровень самостоятельности работы обучающегося\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

недостатки работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

СОДЕРЖАНИЕ

[1. РАЗРАБОТКА СЕРВЕРА АВТОРИЗАЦИИ 3](#_Toc40635445)

[1.1. Описание постановки задачи 3](#_Toc40635446)

[1.1.1. Характеристика задачи 3](#_Toc40635447)

[1.1.2. Входная информация 3](#_Toc40635448)

[1.1.3. Выходная информация 4](#_Toc40635449)

[1.2. Описание алгоритма регистрации пользователей 4](#_Toc40635450)

[1.2.1. Назначение и характеристика 4](#_Toc40635451)

[1.2.2. Используемая информация 5](#_Toc40635452)

[1.2.3. Результаты решения 5](#_Toc40635453)

[1.2.4. Алгоритм решения 5](#_Toc40635454)

1. РАЗРАБОТКА СЕРВЕРА АВТОРИЗАЦИИ

1.1. Описание постановки задачи

1.1.1. Характеристика задачи

Задача – разработать сервер авторизации.

Сервер авторизации разрабатывается с целью обеспечения безопасности данных, хранимых в системе. Для реализации возможности масштабирования системы в будущем на большое количество университетов – было решено разработать собственный сервер авторизации, который будет предоставлять доступ мобильным приложениям пользователей к API системы.

Разработка сервера авторизации связана с разработкой сервера API системы. С помощью проверки доступа к методам API будет проверяться работоспособность сервера авторизации.

Разработка задачи будет завершена при успешном прохождении тестирования сервера.

1.1.2. Входная информация

Перечень входных данных при регистрации пользователей:

* номер зачетной книжки (в случае, если пользователь – студент);
* идентификатор преподавателя в системе (в случае, если пользователь – преподаватель);
* уникальный логин пользователя;
* пароль.

Перечень входных данных при авторизации пользователей:

* логин пользователя, созданный при регистрации;
* пароль пользователя, созданный при регистрации.

Перечень входных данных при валидации токена доступа:

* токен доступа.

Перечень входных данных при обновлении токена доступа:

* токен восстановления доступа:

Вся вышеперечисленная информация передается на сервер с помощью http запросов.

1.1.3. Выходная информация

Перечень выходных данных при регистрации пользователей:

* сообщение об успешной или о неудачной регистрации.

Перечень выходных данных при авторизации пользователей:

* токен доступа;
* токен восстановления доступа;
* идентификатор пользователя в системе.

Перечень выходных данных при валидации токена доступа:

* сообщение об успешной или неудачной валидации.

Перечень выходных данных при обновлении токена доступа:

* токен доступа;
* токен восстановления доступа.

Вся вышеперечисленная информация выдается сервером в ответ на http запросы.

1.2. Описание алгоритма регистрации пользователей

1.2.1. Назначение и характеристика

Регистрация необходима для создания пользователя в системе и его сопоставления с сущностью студента или преподавателя в базе данных.

Алгоритм должен сохранять данные пользователей в базе данных, при этом проверив существование этого пользователя как студента или преподавателя в системе. Пароли пользователей должны записываться в базу в виде хеша таким образом, чтобы одинаковые пароли имели разное представление в базе данных. Полученный хеш должен быть устойчивым к атакам перебора паролей.

1.2.2. Используемая информация

Для алгоритма регистрации используются:

1. номер зачетной книжки студента или идентификатор преподавателя в системе;
2. уникальный логин пользователя;
3. пароль пользователя.

1.2.3. Результаты решения

В результате выполнения алгоритма формируется следующая информация:

1. оповещение клиентского приложения об успешной или неудачной регистрации;
2. сохранение регистрационных данных пользователя (логин, пароль в виде хеша, идентификатор пользователя) в базе данных.

1.2.4. Алгоритм решения

Схема алгоритма:



Описание алгоритма:

1. сервер авторизации получает регистрационные данные от клиентского приложения;
2. сервер авторизации сопоставляет полученный номер зачетной книжки или идентификатор преподавателя с данными в базе;
3. если такого студента или преподавателя не существует, то сервер посылает на клиентское приложение ответ с сообщением об ошибке регистрации и алгоритм завершается;
4. если такой студент или преподаватель существует в университете, то алгоритм выполняется дальше;
5. сервер авторизации генерирует «соль» для хеширования пароля – случайный текст, который добавится к введенному паролю для обеспечения уникальности хешей при введении пользователями одинаковых паролей;
6. сервер генерирует хеш пароля и добавлением «соли»;
7. сервер сохраняет регистрационные данные в базу данных пользователей системы;
8. сервер посылает на клиент ответ с сообщением об успешной регистрации и алгоритм завершается.