

RIT Data Center

Team Infinite Diversions

Sashank Agarwal – 1MS16CS090

Anshul Gupta – 1MS16CS018

Guide: Dr. Seema Shedole, Professor CSE
Department

Agenda

To build a Dynamic Data Center which stores the information about the college and has the ability to generate dynamic reports.

Managing the data in a very large and long-established institution is a tedious job as the size of the institution keeps on extending and to maintain the data of every year in a format that is easily accessible becomes very important.

Introduction

RIT Data-Center records all the data regarding the Institution, various committees and departments in the Institution, achievements, and activities conducted by various departments, admission and scholarship details, faculty details etc in a structured manner that can be read, queried and updated in an easier manner.

It provides various levels of authentication and authorization to the members of the Institution to view and update the data in the database. The primary function of the system is to generate dynamic reports for the departments and the Institution in any specified formats for the given requirements whenever required.

Problem Definition

Current Methodology

Currently the data that is being maintained in the Institution is unstructured and it is laborious work to manage the data and access and update whenever necessary. To generate reports when the technical bodies demand in specific formats, the assigned staff has to first contact the respective departments for gathering information and then has to start building the report format with the required attributes and then manually upload the data to the files.

System Requirements

Hardware Requirements

Our hardware requirement is restricted to just one, that is to set up a high-performance server that could take up the load for 30-40 users at a time. We would need a server with minimum this specification -

- RAM - 8GB or more
- Secondary Storage - SSD 80GB
- GPU - Not Required
- Processor - At least 4 core processor with a clock speed of 3 GHz



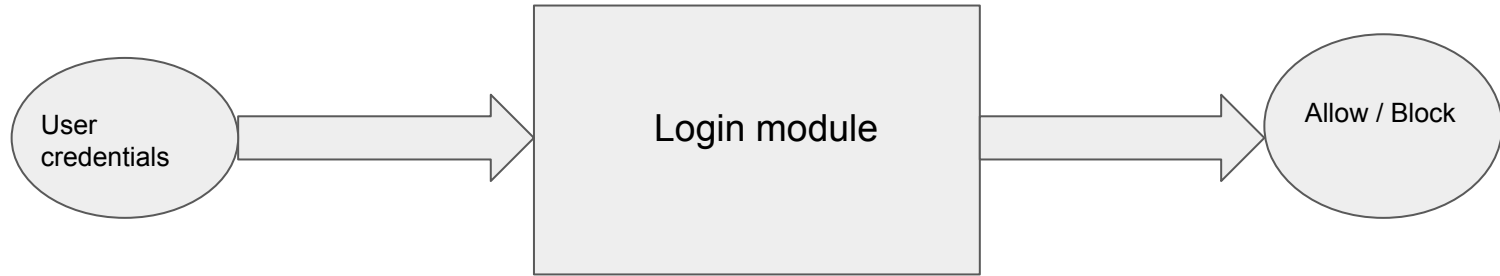
System Requirements

Software Requirements:

- Node JS - for Backend
- MySql - database management
- Node JS - For extracting the data from the raw excel sheets
- EJS View Engine - For Frontend
- HTML/Bootstrap/JS - For making attractive UI

System Architecture

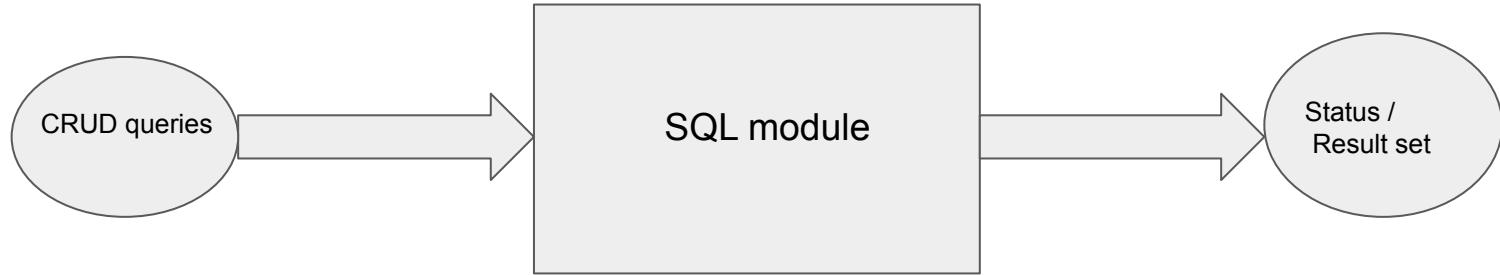
1)Login module



- >The login module takes user input credentials and verifies the data.
- >If the inputs are valid, the user is allowed to use the functionalities of the website.
- >If the inputs are invalid, the user is blocked from the functionalities of the website.

System Architecture

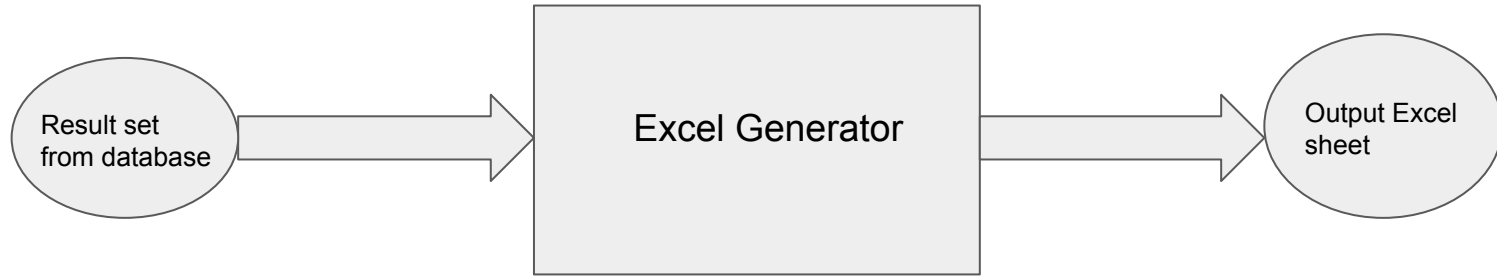
2)SQL module



- >The SQL module takes user queries as input and processes these queries.
- >In case of read query, it returns the result set.
- >If case of insert, delete or update query, it returns the status of execution as output.

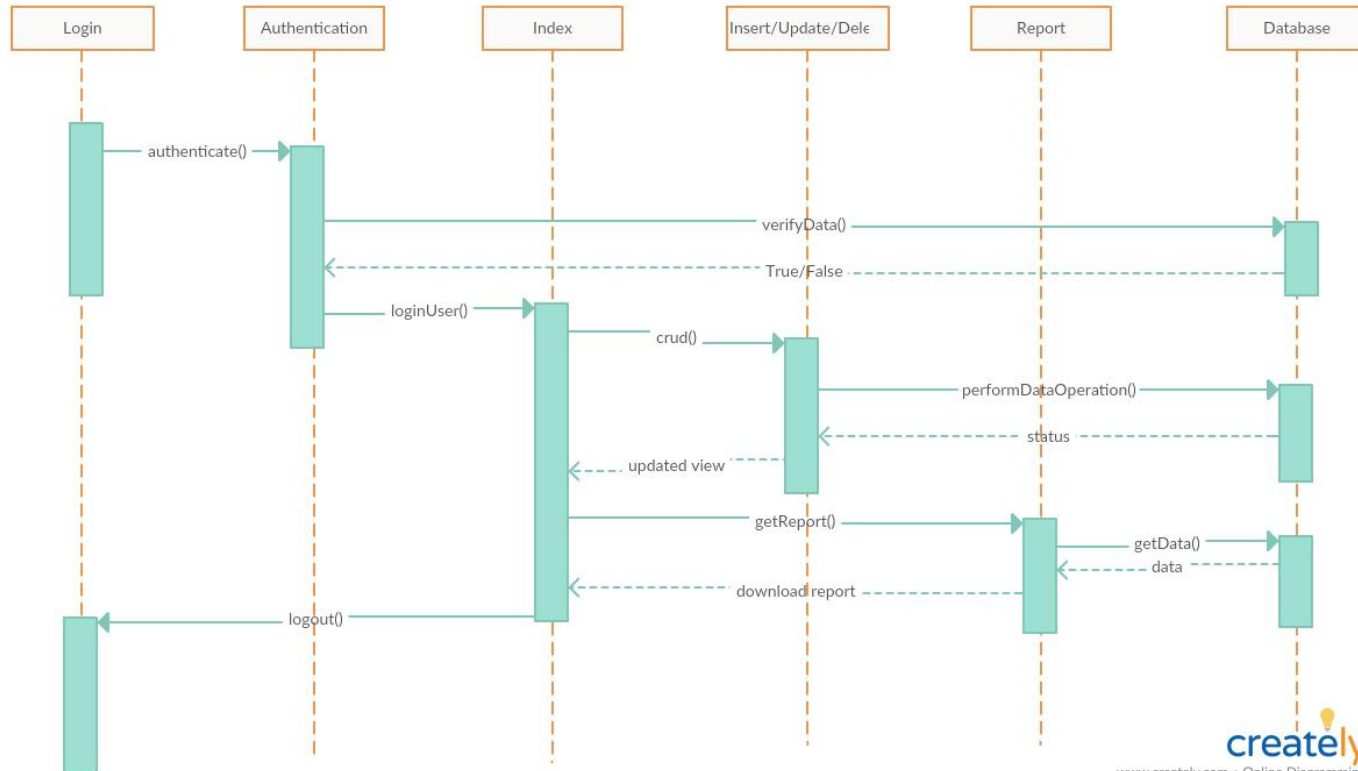
System Architecture

3)Excel Generator

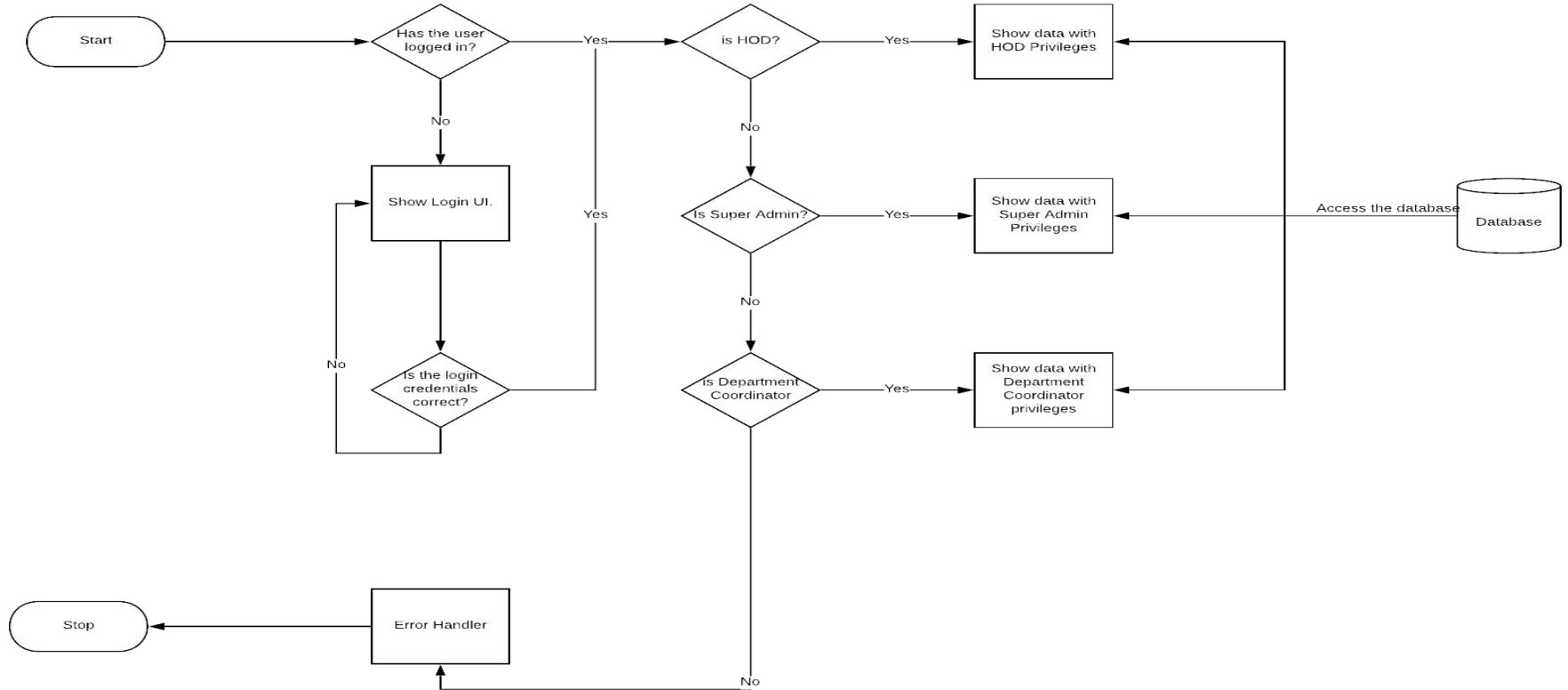


- >The excel generator module, takes result set from the database as an input and generates an output excel file from this result set.
- >Once the excel file is generated, it is downloaded to the user's system.

Sequence Diagram



Flow Chart



Algorithmic Description

Data Cleaning Algorithm:

Algo cleanData(inputExcelFile): returns modifiedExcelSheet

For eachRow in inputExcelFile:

For eachColumn in eachRow:

Remove special Characters from entry[eachRow][eachColumn]

Convert entry[eachRow][eachColumn] to lower case

//Request teammates to suggest any other changes to clean the data

modifiedExcelSheet<--entry[maxRow][maxColumn]

return modifiedExcelSheet

Reason for the choice of Specific Architecture

Due to the structure of the database and the requirements being unstable and frequently Changing we decided to use the following technologies:

We are using SQL for our database, to keep our data structured and have a fast querying and data access. But this fact leads us to restrict our database structure to a minimal change.

Making any changes in the DB structure (due to any change in the requirements) forces us to change almost the entire query related to that schema, which leads to a lot of redundant work.

On the other hand, using a NoSQL Db structure could also have a lot of problems, as it is really difficult to set up the integrity in NoSQL.

We are using nodejs as the framework for back-end. As, It provides Asynchronous I/O . There is a thriving community of nodejs which helps in solving problems encountered during development.

Conclusion

The Data Center will be very helpful for the faculty as they will be able to update their details in the database very easily. The ability to generate reports dynamically for various committees will help save a lots of time.

— Demo

