

Digi-Cabinet Overview

Digi-Cabinet is a secure and simple to use tool for all institutions to obtain insight of administrative and academic data about their students. This will create a new way for student's data to be stored and even shared amongst institutions. The Digi-Cabinet ecosystem is focused on connecting all users and institutions together for responsible data sharing, organization and analytics. In a world of remote collaboration, big data and digitalization of all information, Digi-Cabinet wants to push education to a new level of insight, convenience and professionalism.

All insight and analytics about students should be at our fingertips. We aim to reduce clutter, and grow physical space by ridding the office of paper and file cabinets. By conversing with individuals within the education field and doing research we have learnt what would be important to an institution. Digi-Cabinet wants to provide a solution to bring useful information and analytical tools to every institution. As the world changes and we interact more within the virtual world we must be able to sustain obtaining and recording information without the risk of that information being destroyed.

This system makes it easy and convenient to store student's data, track student's progress and provide other organizations or government bodies information on a particular student or institution. Doing this will create effective solutions for the advancement of education. With authorization and strict restrictions based on user type, this system can safely provide information that is important to the specific individual or body requesting it. With data storage and authorization, this software will provide everyone with an easier way to have access to information and utilize it. Proper student development is the goal to which this system is being engineered.

Digi-Cabinet will lead the way with cloud technology to improve and secure how we store data about the next generation of students. Ease of use, security, useful data and convenience will allow for new possibilities and improvements in how we look after our students to ensure their success. The possibilities are endless with Digi Cabinet's flexible, scalable SaaS system.

We are focused on making this system available to all institutions from the pre-school to secondary school level. At these institutions, analysing and guiding students to reach their full potential is the aim in the education system. Keeping proper detailed records of students' progression and journey throughout their life will make it easier for even other institutions to determine if a student will be a good fit, or even allowing the current institution to solve problems the student might have.

The reason for such a high level of detail is simply because of how complex education and behaviours are in students. Based on research and interviews with teachers and lecturers, they have informed us on how important it is to understand the student's parents'/guardians'



involvement, student's absences and in today's modern world even the amount of devices and wifi access. We have compiled and will allow for the detailed recording of relevant important data that institutions will find important.

Digi Cabinet's stakeholders are

1. Teachers
 2. Students
 3. Guardians
 4. Administrators
 5. Principals
 6. Deans
 7. Counselors
-

High Level Functionality

This system will allow Administrators to create a new account by *registering* then *logging* in so that they can *input*, *view* and *analyse* the data within the database. An administrator will then create profiles for other users to access the platform. Permission to view or use data will be based on their account type and if they're verified by a higher entity in the system. Requesting permission from entities with higher authorization to temporarily access or mutate data will also be implemented. This system's storage of data will be the primary functionality while offering other features to have analytics and statistics on specific students or institutions. These functionalities will allow for proper safe access of data on students, limiting details based on authorization, while allowing for insight into information.

To allow restrictions, and proper usage of the ability to change and access data about students, authentication is provided. Authentication will allow a level of security so that only the appropriate persons will be identified and authorized to have the ability to access or mutate data. With registration, confirmation of email, verification, authorization and authentication, a trustworthy level of security may be reached. It may be best to let users login to their accounts on every use of the system so that data cannot be easily accessed by outsiders. Authentication and this level of security will allow for safe access and usage of data.

Having multiple account types allows for further proper use of data. This software will limit or grant access to different account types and their relation to the entity that is requested. For example, institution "x's" admin cannot view data on a student from institution "y" unless permission from that institution is granted. While some accounts can see all students, the students themselves can only view their own data and the hierarchy of access/permission goes on. The highest level of permission may only be granted to institution administration which will be able to access all students and share that data with other institutions.

Data is the building block of understanding our world and tackling unseen issues by transforming it into information. Using analytics, the software can be a tool to give users useful insight into the progression and statistics of a student or institution. Algorithms and libraries will facilitate in transforming data into usable information that can be utilized. Some examples of such data are student's biodata, attendance, parent participation in PTA meetings and grades. Other concerns that teachers brought up that make for good examples are when they get a new student with no grades or attendance history from previous forms.

Each authorized body will have access to this information when a student has transferred, and this will create a narrative to the institution that will explain the correlation between student attendance and grades. The parents will be held accountable as their involvement will be evaluated as well.

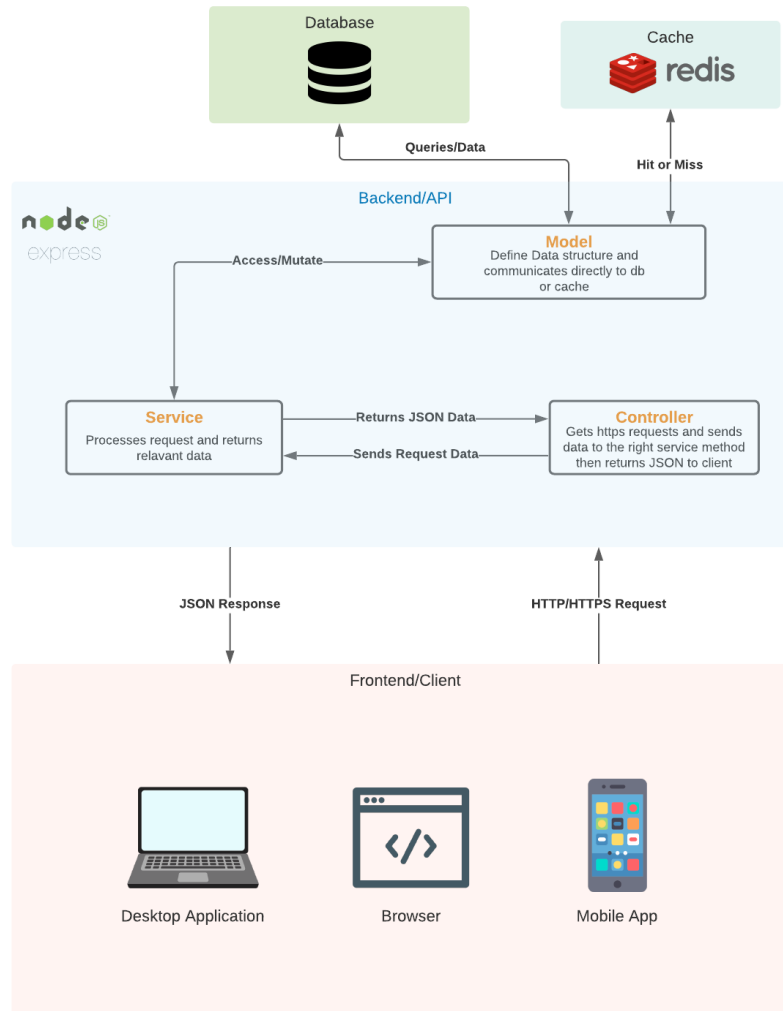
This system achieves many things that will provide users with the ability to safely but conveniently access and enter data. With the access of data, it can become even more useful



with analytics and insight into the information collected. These tasks are done with authentication of users, account types to restrict access, databases, and algorithms and libraries to give insight into data. Functionality like this will bring about effective solutions to assist all in ensuring the success of all.

Architectural Breakdown

This architectural model of the system below will be used for understanding, communication and implementation.



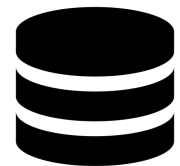
This system will be broken down in three parts. These parts consist of the client, server and database. It will be done in this way to allow for security, flexibility and scalability of the overall ecosystem.

The Frontend or the client side will be developed for different devices so that users can have access in many different ways. This software needs to be accessed by many different users with different incomes and device availability. This made us focus primarily on building a progressive web application (**PWA**) so that users with smartphones, tablets, laptops or desktops may have access to the system. Our secondary focus will be on android and apple mobile apps. This is so because while these apps might give a cleaner better experience it wouldn't serve as many

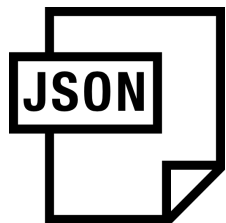
users. **JSON** data and **JWT** will allow for all these devices to use one API/backend to reduce development time and cost. Each HTTPS request sent from the client will have an access token attached to it to authorize each user, and authentication/login will allow for refresh tokens to be created so that the user can create these access tokens for more security and ease of use.



The database is the heart of this system and will use a **noSQL** database to store all data and objects. This communicates directly to the API to *create, read, update or delete* data. A noSQL database was chosen because of the continuous updating of data and scalability.



The backend or the api is the connector between the database and the many different types of clients. This has the responsibility of handling requests, authorizing these CRUD operations and giving appropriate access to the database while performing efficiently. Using a Rest API architecture within the backend, this can achieve these goals and create a system that will grant easy maintenance and continuous development and integration.



Handling requests from many different users requires a standard way of sending and receiving data among all the different possible client types (Desktop Application, Web Application, Mobile App, Wearable Devices). Digi-Cabinet, wanting to be scalable and flexible, a *REST API* serving only JSON data can achieve a standard where all clients can use this data from the database with ease to create the view.



Node.js and **Express.js** are common tools used to create REST APIs. Node.js is a runtime that allows developers to run Javascript off of a browser. Express.js is a module used with node.js to create servers.

Controller | Works out which *service* should be used to process the request based on the CRUD operation and http/https route the client sent. This then returns the data returned from the service as JSON.

Service | Data sent from the controller will be processed and it will determine if the model should access or mutate data through the *model*. The service always returns an object to the controller.



Model | Connecting directly to the database, the model would help to enforce schemas, verify, sanitize data, etc. Within the program itself, these classes will contain accessors and mutators that will directly update the data within the noSQL database. The model will also check the



redis

cache database (Redis) for efficiency. Using key-value peers and noSQL, Redis makes for a fast light database which can be used to minimize calls to the database where queries may take up time.
