Classy: Architectural Design

CS 400-ON Group: Delicate Garbage

Core questions addressed in this document:

- How the system will be organized as a set of architectural components, where each of these components provides a subset of the overall system functionality
- How these architectural components will be distributed and communicate with each other
- What technologies will be used in building the system and what components should be reused

Core Architectural Components

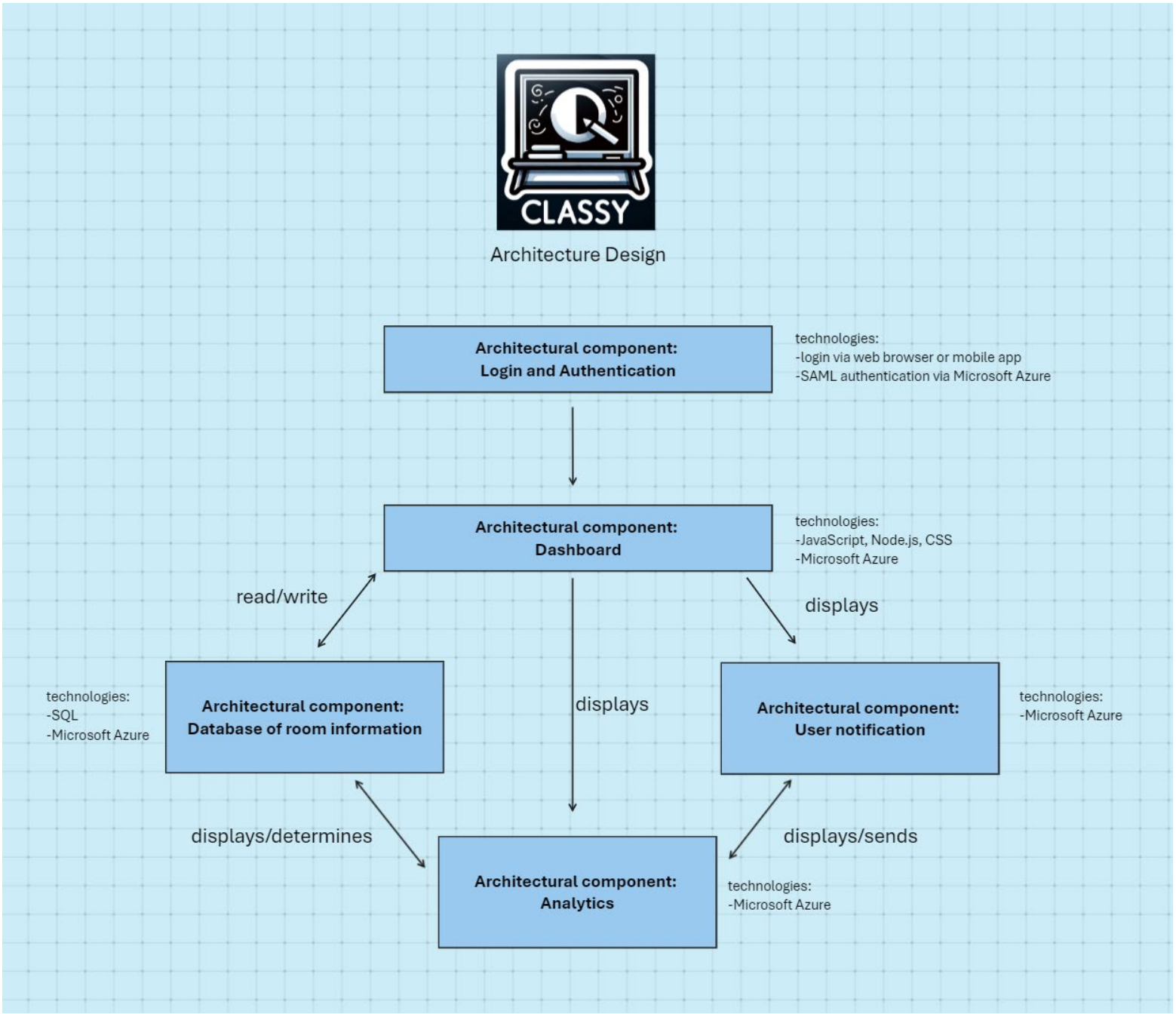
The primary UI components will include the following:

- Login and authentication
- A user dashboard displaying relevant information along with visual representations of data
- Database writing/reading for logging information about rooms, inventory, and condition
- Data analytics
- Notifications

The architectural design is shown in a diagram format on the next page.

CLASSY

Architecture Design



Core Application Functionality

The architectural design's five core components provide the basis for numerous application services, including the following:

- A secure log-in and authentication process that allows users to access the application through either a web browser or a standalone mobile application.
- The ability to view a dashboard interface that surfaces all relevant information for a given user, based on that user's role. Eventually, this dashboard will provide a virtual simulated image of the classroom and its contents, along with meaningful visual descriptions of key items and conditions. For example, if the floor needs to be cleaned, those who teach in that classroom would see a red floor for that classroom's visual depiction in the dashboard, indicating a problematic condition.
- The ability to read information from the application's database, for the purpose of surfacing relevant information in the dashboard.
- The ability to write information to the application's database, for the purpose of updating information on classrooms, inventories, and conditions.
- The ability to conduct data analytics—both out-of-the-box and customized by the user. These analytics would be displayed to the user in the dashboard.
- The ability to send and receive notifications through the user dashboard; these notifications can be user-generated as well as automatically triggered based on conditions set in the user dashboard. An example of the latter might be a notification sent to the user when a previous user of the same classroom has indicated a problematic condition in the classroom, such as a mess on the floor or insufficient stock of dry-erase markers.

Technologies

In terms of the technologies required in the building of the Classy system architecture, each is addressed below.

Login and authentication

Classy will operate as a cloud-based software application accessed through a web browser or a mobile application. The completed webpage will provide links to the iOS and Android app stores for the mobile application. For convenience and ease of use, the application will run entirely through the cloud.

The app will have a client-side login screen. The login gateway will serve as authentication and communication with the servers for the user's respective school, using SAML

authentication powered by Microsoft Azure. Upon successful login, the user will find their unique and customizable user interface presented in the Dashboard, the application's landing page.

User dashboard

The user dashboard will display relevant information from the Classy database, key data analytics, and visual representations of important data. This component will rely on the JavaScript programming language, using Node.js, HTML and CSS, all hosted in the cloud by Microsoft Azure. This will provide a robust dashboard environment that combines webpage interactivity, dashboard interactivity, visual page elements, visual analytics displays, and more. Given that this is the core UI of the Classy application, this will be the standout unique feature.

Database

All data used in the software application will be stored and manipulated using structured query language (SQL), relying on Microsoft Azure SQL Database, as part of the application's reliance on Azure for web services. The database will primarily contain data related to the logging of information about rooms, inventory, and condition. The data compiled to the database will also be used for running analytics and triggering notifications to users about room states, inventories, and conditions.

Data analytics

Data analytics will be logged server-side, and will be conducted on the data populating the database component of the application's architecture. Because Microsoft Azure's SQL Database will be used for the database component, Azure's data analytics capabilities will be utilized for the analytics component of Classy's design architecture as well. Key out-of-the-box analytics will be included for all users. In terms of functional analytics, Classy will calculate and display information about all classrooms to which a given user has access. It will identify problematic classrooms on account of accumulating issues identified by users. Summary data on stock and inventories will be provided, too; for example, if a given classroom is commonly short on dry-erase markers, a user attached to that room would see that given analytic information. In terms of user analytics—useful to administrator accounts for Classy clients—this will include how many requests the user makes and what

they are asking for, which features are used and how frequently, how much storage they are using and queries used in the search feature.

Notifications

The notification architectural component of Classy will be built through Microsoft Azure's Notification Hubs, based on Node.js programming that will configure key notifications around set conditions—triggered by data in the database component and corresponding analytics (both also in Azure). For example, if a user is assigned to a given classroom, that user would receive notifications when a problematic condition is sent to the database for that particular classroom. Multiple users would receive the same notification, and all of this would be configurable by a customer's local administrator account, as well as by the end user, to a lesser extent.