TUTORLY WRITE-UP

DANIEL BRISENO, MATTHEW RAYMOND

As longtime employees of the TLT, we've noticed a few inefficiencies in the way that things are run. Most are relatively benign, but there is one in particular that we think needs a major overhaul: the student/tutor check-in process. This process, although good-intentioned, requires an enormous amount of work by the tutoring staff. Just to illustrate how much work it is, we want to walk you through the process from beginning to end.

(1) When a tutor arrives at the TLT, they sign in on paper sign-in sheet.

- They must fill out their arrival time, all subjects tutored, and table number.

(2) Tutors must also sign in on their own paper timesheet (which they keep and turn in bi-weekly)

- They must fill out the date and time that they tutored

(3) Once a student/tutee arrives at the TLT, they have to manually sign in through an Ipad using their student ID

(4) Once a student finds a tutor (which is often by asking around) the student signs in on tutor's paper timesheet

- They include their student ID, name, the time they arrived, and the class they're being tutored for

(5) Once the tutoring session is finished, the student signs out on tutor's paper timesheet

- They must include the time-out and their signature

(6) When their shift has ended, the tutor must sign out on their paper time sheet

- They must record the time they left, as well as the difference between their arrival and departure times

(7) Before they leave the room, however, a tutor must record their time of departure on a TLT paper sign-in sheet

(8) On a biweekly basis, the tutor must turn in their paper timesheet to the TLT office

- When submitting, they must also include the current pay-period and their student ID

(9) At the same time, a tutor must manually fill out an online timesheet with the dates, times, duration that they worked, and the type of shift (regular work, overtime, sick-pay)

(10) Once all documents are submitted, the TLT Management (2 people) manually input timesheet data for all tutors to Excel and compare this document to an online time-sheet. When there are missing fields or inconsistent/absent Tutee arrival and departure times, they try to contact the tutees to fix this information

(11) As a result, tutors are often sent emails about bad or missing information on time-sheets, are told to correct it. Often, the student doesn't remember what they put on their timesheet and has to guess.

- Pay is withheld until corrections are made. Tutors also held liable for a Tutee's mistakes, such as not signing in, not giving a signature, and not signing out.

(12) Tutors must then submit a corrected paper timesheet, which is then manually re-verified by TLT management.

(13) Finally, the TLT front desk attendants and management manually input sign-in sheet data

- We estimate the TLT is at least a year behind on this data-entry, meaning that the data cannot be used for any meaningful data analysis.

This requires 4 mediums (Excel, PeopleSoft, and two types of papers), and is understandably prone to errors and bottlenecks. Mistakes abound, and since they are often impossible to correct, people resort to best-guess. Tutors cannot be expected to babysit tutees and make sure that they fill out the form three times while simultaneously tutoring multiple students. Data-entry is also extremely backlogged because of the extreme load that TLT staff is being put under, and as a result, the data that's collected is at best misleading, at worst false. Tutors' shifts do not coincide with rush hours, leading to a surplus of tutors for some subjects, and a shortage for others. As a result, some tutors will have upwards of 15 students while multiple tutors have none. However, this doesn't have to be the case. Through this project, we hope to streamline this incredibly labor-intensive, error-prone, and distributed system and bring it down to three easy steps:

(1) Swipe your ID to check-in

(2) Swipe your ID to check-out

(3) Let SQL do the rest

On the surface, our application is very simple. Let's say that you're a student. When you arrive at the TLT, you swipe your ID at a touchscreen kiosk, which brings up a tutor-selection menu. You tap the tutor you're here to see, and press continue. That's it. To check out, you simply swipe your ID again and you're good to go. The kiosk also contains a list of classes that TLT offers, a list of tutors who work at the TLT, those tutors' schedules, which tutors are checked in, which tutors tutor a given

class, and which classes a tutor tutors. These allow students to service themselves, and frees up the front desk for other, less-repetitive tasks.

When a tutor arrives for work, they follow a very similar process. They swipe their ID and choose whether they're here to be tutored or tutor. If they're here to be tutored, they follow the same path as a student. Otherwise, they enter a passcode[1] (since this counts for their timesheets), choose the table that they're going to be sitting at, and click continue. To check-out, all you do is swipe your card.

The back end is much more complicated. Our UI was built with jQuery using the Electron framework, resulting in an application that is platform-independent. The backend was built using Node.js, and the database aspect was implemented using MySQL hosted in Google Cloud servers. This gives us an application that is not only extremely portable, but also simple to maintain and extremely scalable.

In order to keep our application as simple to update as possible, we use javascript almost exclusively for making database calls and displaying data (all of which is done programmatically based on return values). Almost all of the business logic is implemented in MySQL using stored procedures. This makes the programming slightly more complicated but makes updates massively easier. Instead of having to manually install a new version of the application after every update, we simply change the stored procedures and the application will automatically have new logic. This allows for simple and non-intrusive bug fixes.

Because this is a simple check-in and check-out system, the business logic isn't too complicated and mainly relies on nested if-else statements. However, due to the asynchronous nature of database calls, there is a decent number of linked callbacks in the javascript.

[1]To be implemented in a later version

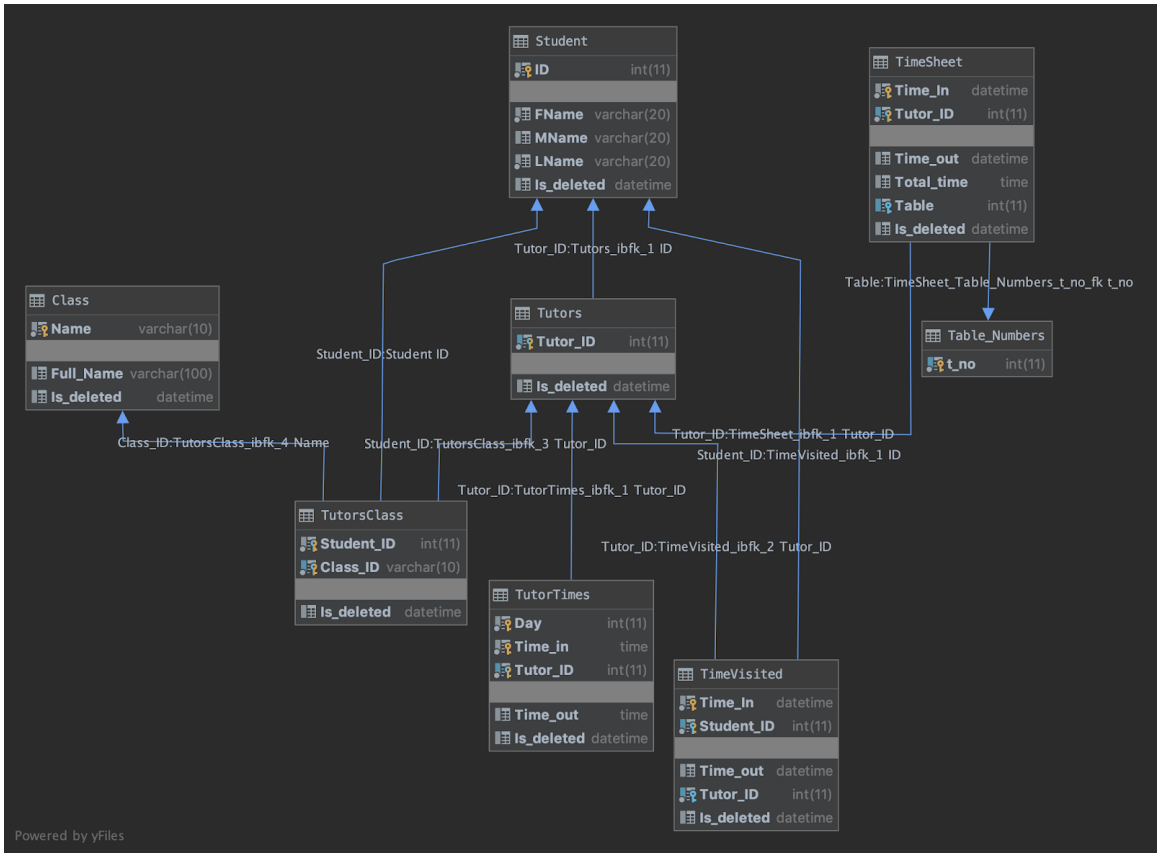

FIGURE 1. Database Schema Diagram for Tutorly

Already, our application massively cuts down on repetition and wasted time. However, we want to develop this into a full-fledged application for use in the tutoring center. As we develop it further, we intend to add the following features:

- Move the application from electron onto a node-based web server, which can then be accessed from a website.
- A front-desk login screen so that simple errors and basic admin tasks can be attended to by the front desk attendant.
- A system that maintains an even distribution of tutees across the tutors, encouraging healthier workloads.

- A separate application that allows the TLT Management staff to make admin-level changes (such as adding/updating/deleting tutors, changing tutor schedules, updating tutor times, viewing data analytics, etc.) As the project stands, the "tablet" electron application has very limited data analysis and record-editing abilities (export tables to CSV and add/drop tutors) to be in accordance with the assignment specification. The true tablet application will have none of these capabilities, instead they will be present in the TLT Management app.
- The TLT Management application would also contain a streamlined data-analytics screen that allows the TLT Management staff to see when surges happen, what classes they need more tutors for, and what classes rarely get tutors.
- An option to create a google calendar link so tutors can easily keep track of their schedules.

We believe that the tutoring center is in desperate need of modernization. With our application, we hope to provide a measure of necessary automation and make life better for the tutors, tutees, and TLT Management staff. We intend to continue working on this project over the summer, and we're excited to see where it takes us.