Deliverable 3

Project/Group Name: Pro TA

Nicolas Stencel, Keenan Jabri, Naumaan Hassan, & Manuel Vargas

Table of Contents

Project/Group Name: Pro TA 1

User Manual 3

- **Introduction** 3
- **Using the proTA application** 4
- **Conclusion** 7

Running Program Instructions 8

- **Program** 8
- **Testing** 8

Test Cases 9

Features and Limitations 10

- **Features:** 10
- **Limitations:** 10

Meeting Minutes 10

Member Contribution Table 13

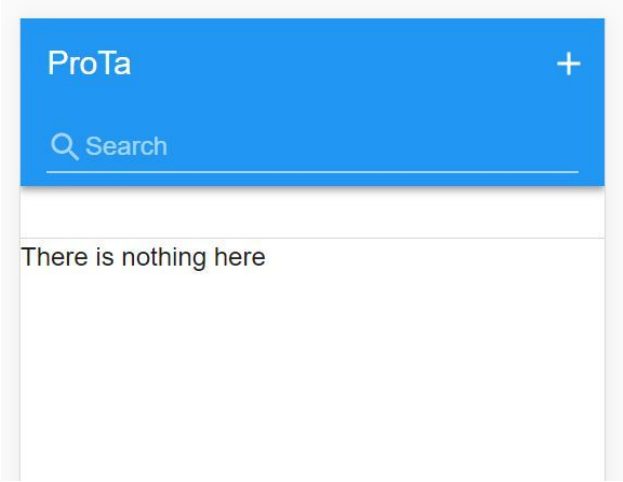
User Manual

Introduction

ProTA is an app that is meant to make the lives of teachers easier by allowing them to forgo the traditional role call method of taking attendance by instead automatically marking students as present or absent based on if and when they swipe their student ID cards through our card scanner. In turn it serves as a benefit to students by ensuring that their costly classroom time that they've paid for is not eaten by teachers calling out every name in the classroom to see if they're tardy or not. Such methods might be a welcome delay in proceedings for students in a primary education environment, where attendance is mandatory and inspiration is near zero, but for students in University this is not only an irritant but a detriment as well, serving to waste time in a class that they attend willingly and for a costly fee.

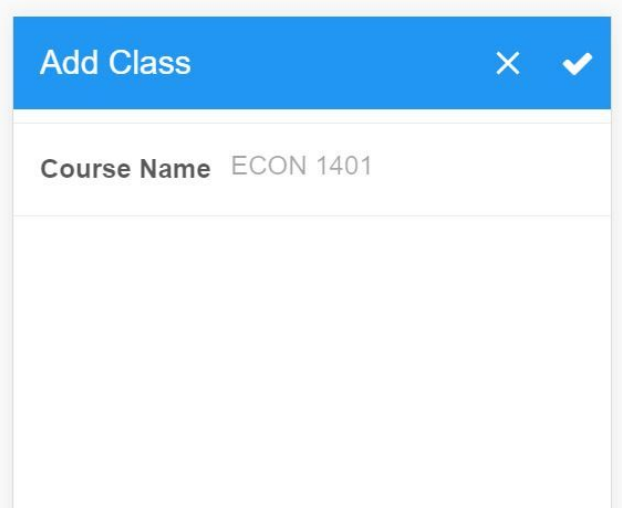
Using the proTA application

When a new user opens up our app for the first time they will be taken to our front page, which will list all the classes they have created. Because a new user cannot have created any classes by this point the page body will be blank.

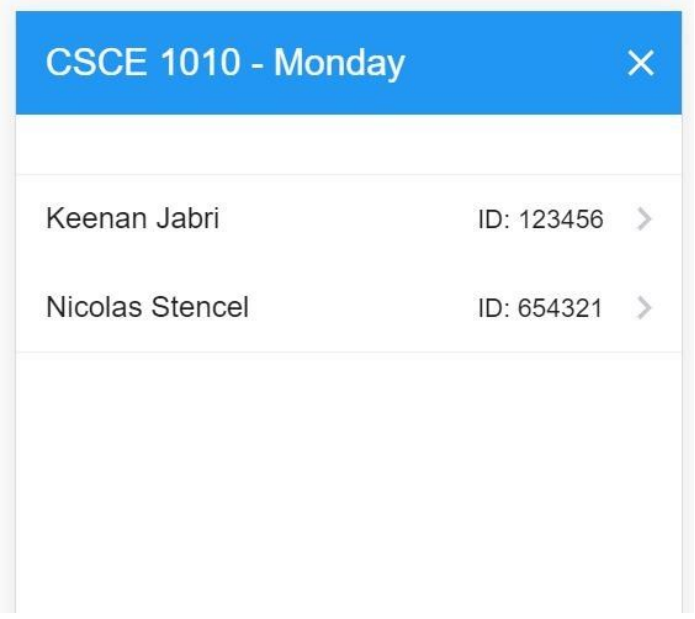


In order to create a new class the user simply has to press the plus button in the top right corner.

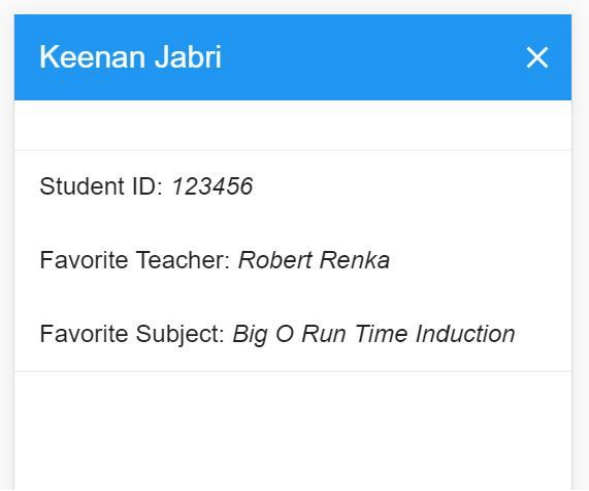
Once the user presses the add class button they will be taken to a popup that will allow them to type the name of the course they want to create. After they enter the name of their course they want to create they can then press the checkmark at the top right corner that will create that new course for them, if for whatever reason they want to leave the page without creating a new course they can press the 'x' button located just to the left of the checkmark that will take them back to the homepage without creating a new course.



Upon clicking the checkmark at the top right corner, and thus creating a new course, the user will automatically be taken to the attendance taking page. This page is what will be open as students swipe into the class. Upon each new student swipe, their name and student ID will be added to the page so that by the end of the class the user will be able to look at the screen and see which students were present on that day.

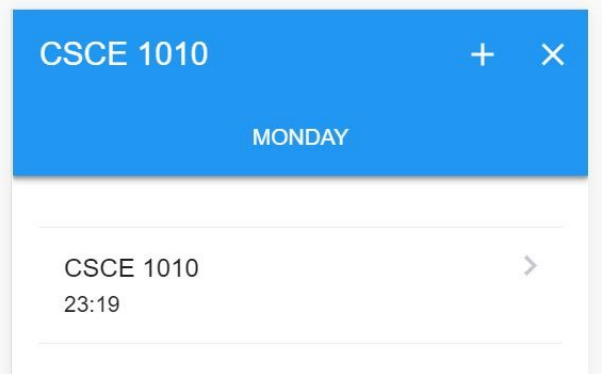


The user will then be able to click upon each student's name to see their ID number. Although this is all at the moment this feature was added as something that will be expanded on later after the semester is over and could potentially contain such information as their student photo, year level, grade average for the class and other pertinent material.



Pressing the 'X' button located at the top right hand corner will take the user back one page. Press said button however many number of pages you have traveled from the home page to get back to the homepage. Once back on the homepage the course that the user created will now be listed.

If the user has created enough classes such that it becomes difficult to find any of their classes by hand they will be able to search for any class using the search bar located at the top of the home page. If the user should click upon any listed class on the homepage they will be taken to a page for that course that will have listed every instance of that course ever created and on which day it was created. The user also has the ability to click upon any of these classes and see which students attended each of these classes. To create a new instance of the class simply press the '+' button located at the top right corner of the screen. The user will then be taken to the attendance screen where students will swipe in.



With that, the functionality of the proTA app has been explained and is now ready for anybody to make use of it as they see fit.

Conclusion

The proTA slogan is "Bringing the classroom into the 21st century" and with what we have accomplished over the course of this semester we are proud to say that we have taken the first steps towards making that goal a reality. By making use of our app teachers can now skip having to call out each student's name, and instead jump right into the lesson being taught that day. Students also benefit from our fast, quick, and easy swipe system in that they no longer have to suffer through listening to their professor callout name after name at the beginning of each class, and can now enjoy the full amount of class time that they paid for.

Higher education is too valuable to waste on archaic teaching practices but unfortunately no convenient alternative has arisen, until now. Just as mobile phones changed the way we communicate, so too will proTA change the way we learn.

Running Program Instructions

Program

For our application to function properly, it needs to be run on a mobile device by installing the package APK stored in our build folder (CSCE-4444-Group-Project\source\build) into an android phone; our app is able to be used on iOS as well but we decided to prioritize our android build with PhoneGap Build .

Since our program was written in javascript, compilation happens at the same time as when you run the program using JIT (Just In Time) compiling. To be able to run a version of the mobile application on your desktop, go to `CSCE-4444-Group-Project\source\build\proTA` and open the `index.html` in your browser. Doing this will limit the storage capabilities that make the application function properly but you won't need to have an android phone.

Testing

We are using Jest to test our core js functionality. To run the test suite, you must install node and npm and change directories over to `CSCE-4444-Group-Project\source\build\proTA\tests`. Now run the command `npm test` which will trigger the jest testing suite and show how many tests have passed or failed.

We found writing our test to be very tedious because of the way we initially wrote our code; since our functions rely heavily on what is currently on the DOM (because we wanted to keep it a single page app), it made creating test cases very difficult because functions were passed div's by how close they are to something using jQuery methods such as Closest.

In order to store data on the device we used localStorage at first but once we switched over to testing on our mobile devices, we realized we had to do a lot of restructuring to make it work as it did on our browser testing environment. Instead we decided to store our data in a file system where we could read and write to files in our directory, that way we would avoid the hassle of using localStorage entirely.

Test Cases

We are building test case in the same way we would run a usability test for a person. We give scenarios in which you would use the application and give expected results.

Course Creation

Scenario: Teacher should be able to add courses to their list of courses
Expected Result: we should see the course added visibly in their list of classes

Student Check-in

Scenario: Student is arriving and needs to check-in for attendance
Expected Result: push a check-in object to the Teacher's course giving name, date, and time.

Modify Student Data - Deletion

Scenario: Teacher knows student has dropped class and should be removed from class.
Expected Result: Pop student teacher selects

Modify Student Data - Addition

Scenario: Student registers late for class, must add to roster.
Expected Result: Push a student object to teacher's course.

Track Student Attendance

Scenario: Teacher needs records of when students came to class.
Expected Result: Teacher able to view list of students

Specific tests are located in `CSCE-4444-Group-Project\source\build\proTA\tests` in the github repository.

Features and Limitations

Features:

We were able to implement many of the ideas that we had planned at the beginning of the semester. Key among these is the fact that teachers do not have to enter in the time that their class starts, or any other tedious details. The only thing they have to input is the name of the course when they first create said course, and the rest will be taken care of automatically by proTA. Creating a new instance of a course is as easy as clicking on it on the homepage, then clicking the '+' button on the course page. The course start time will be set as the time you click that button.

Another major feature we implemented was the different days a course was held. Whereas before we were thinking about simply having a set course where students would swipe in, we now have distinctly marked categories that show which day of the week a course is being held. This makes it easier for the teacher to keep track of attendance throughout the semester.

Limitations:

One of the limitations that was brought up during our peer reviews was the fact that a student could take their friends card and swipe in for them. Unfortunately there is no way for us to prevent this through our app, but in reality this is not a new, unknown problem. It is just as easy for a student to shout "Here!" when the professor calls their friends name as it is to swipe in with a different student ID card. In fact there is more of a deterrent to do this with the swipe method as the phone that the app is running on (the teacher's phone) should be next to the teacher on their desk, so that it is obvious to notice somebody swiping multiple cards at one time.

Another feature that we did not implement due to time constraints was having tardy time. This was strictly due to time constraints. Moving into the future we will improve the app so that teachers pick the tardy time as part of the course set up process. Meaning that if they choose 10 minutes, then any student that swipes in more than 10 minutes after the class starts will be marked as tardy. The app would also keep track of the number of times a student has been tardy throughout the semester so that the teacher would be able to go back and deduct points if they're tardy a certain number of times, if they wish.

Another feature we did not implement due to time constraints is the new semester feature. With this feature a teacher would be able to select one of the courses that they

made and press a button to create that same course but with a different date parameter. This way the teacher would be better able to keep track of all their students.

Meeting Minutes

October 17th, 2017 : 3:45 PM - 4:30 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan, Keenan Jabri

Discussed our progress on Deliverable II and what we still need to work on. Decided that Keenan would take the lead on our coding setup, considering he has the most experience out of all of us, and that he would show us the ropes regarding file storage and JSON card data parsing.

October 19th, 2017 : 3:15 PM - 3:45 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan, Keenan Jabri

Naumaan volunteered to write the user manual and get started on Deliverable III early so that we could focus on the coding aspect of the project. Nicolas proposed we meet outside of class in order to delve deep into the thick of things by getting the proper frameworks and packaging utilities up and running.

October 24th, 2017 : 3:00 PM - 4:00 PM

Group Members Present: Manuel Vargas, Nicolas Stencel

We looked around at the different JavaScript testing platforms to see which one was best for our project. We looked at ones such as Mocha, Chai, and CucumberJS, before Manuel proposed that we use JEST so that we would be able to change our files if necessary in a quick and effective manner. Manuel also volunteered to write the test functions for our project as it was a necessary feature for the Software Development process.

October 31st, 2017 : 3:15 PM - 3:45 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan

This was a progress evaluation day, simply to see where everybody stood on their assigned tasks thus far. All big tasks that had been planned for that point had been fulfilled so we felt there was no need to switch any of our timeline scheduling around. Nicolas suggested we change the color scheme of our app to make it more visually appealing for the user while Naumaan told us he had finished the beginning parts for the user manual. Manuel agreed and told us he was working on the test functionality.

November 2nd, 2017 : 2:00 PM - 2:30 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan, Keenan Jabri

Keenan informed us that he would be updating the pro function which is the backbone of our entire app. It provides different functionalities such as openPage and openPopup that allow us to move between different views. He explained that he was doing this so that it would be easier to use for us to create different views in the html.

November 7th, 2017 : 1:30 PM - 2:30 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan

We tested the app that we had managed to create thus far and made note of the areas that we needed to improve on as well as who should take responsibility for these new tasks. We ended up dividing it between ourselves.

November 9th, 2017 : 3:45 - 4:15 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan

Considering that we had kept up with our project timeline thus far we simply gave a quick progress report on our individual tasks. Manuel mentioned that he was having a brief bit of difficulty with the JEST testing platform but that he should be able to have it handled by the weekend.

November 14th, 2017 : 2:30 PM - 3:45 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Keenan Jabri, Naumaan Hassan

Tested what we had accomplished with the project thus far, namely initial loading of the app and the pro method that allows pages and popups to load. Part of the app involves

swiping between different days that the class is held, for example if a class is Tuesday and Thursday being able to swipe to past classes to see the analytics. We were having trouble getting the swipe motion to work so we messaged Keenan on Slack and he explained that the swipe motion only works in developer mode, phone console.

November 16th, 2017 : 4:00 PM - 4:30 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan, Keenan Jabri

Discussed our project thus far. We were all happy with how things were progressing so we didn't make any significant changes to the timeline or the commitments of any of the group members. Since Keenan had provided invaluable knowledge and experience in setting up the core utilities Nicolas and Manuel agreed to construct different views and popups, such as the page for adding students to a class, as well as setting up rudimentary Teacher and Student classes.

November 19th, 2017 (Collin meeting) 1:00 PM - 6:00 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan, Keenan Jabri

Met at Collin College to get started on the final draft of the project. Keenan explained the app structure with us so that we were thoroughly up to speed and were able to build on top of that, Manuel worked on the course and teacher class. Nicolas got started on accepting user input and storing it locally via the custom built toJSON method and localStorage.

November 26th, 2017 (Starbucks and UTD Meeting) : 2:00 PM - 8:00 PM

Group Members Present: Manuel Vargas, Nicolas Stencel, Naumaan Hassan, Keenan Jabri

Decided to use both of our slack days on Deliverable III so that we could fine tune it for a couple more days to get it perfect. Downloaded the apk file from PhoneGap in order to do final testing on the code. Replaced the localStorage[1] method with the File storage method. Got tabs working for different days of the week, parsed JSON data from the student ID cards and pulled student information from it. Worked on Deliverable III and got the testing methods done using JEST.

---

[1] localStorage: https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage

Member Contribution Table

Member	Contribution Description	Overall Contribution	Note
Nicolas Stencel	Initiated group meetings, took meeting minutes, worked on the coding for the text template views and storage, updated the toJson method, and worked on the Deliverable.	25.5%	Kept track of deadlines, and arranged team meetings as necessary.
Keenan Jabri	Set up the initial working template for the application and worked on the Deliverable. Taught us how to use it and pushed changes as needed. Worked and handled Card Swiping functionality	27%	Lead the project development as well as the vision
Manuel Vargas	Handled code test, unit testing, Deliverable preparation, maintained the github repository.	25.5%	Had to work around not being able to test the DOM parts of our code in JEST unit tests.
Naumaan Hassan	Focused on the graphics for the app as well as took notes for the user manual, and focused on error checking.	22%	Guided presentation and created logo for app.