Sprint Report #5

0.1 Team Members:

Marcus Berger
Dicheng Wu
Sponsor:
Jeff McGough

0.2 Prototype Progress

Sprint five was dedicated to finishing the last of the functions and clean up of the project. First the client modified some requirements for the team. After this the user stories remaining from sprint four were tackled by the group to produce the remaining payroll and billing interface. This included logging hours, and payments. After this was done the team started on bug fixes and quality of life updates such as new buttons and removal functions. The team is currently working on finishing up the student registration, and tax calculation for wages. If progress continues at this rate the prototype should be completed functionally however without some work the prototype may not reach some of the teams desired standard.

0.3 Project deliverables of Sprint 5:

Sprint 5 deliverables were the remaining functionality for the project. If the sprint was successful the main project will be done or very close to done functionally at the end of the sprint.

0.3.1 Sprint 5 Backlog

1. Admin and teacher update crossover

2. Ignoring case in address forms

3. Add rejected students to approved/ reject and provide current status

4. Multiple pay rates

5. Admin list function

6. Enter staff hours

7. Sprint 4 rollover (remaining payroll functions)

8. Add and remove class location

9. Removal Functions (Admin, Class, Student, Teacher)

10. Refunds and check boxes for fees and tuition

11. Quality of life updates to some functions

12. Bug Fixes

13. Student registration

14. Employee wages

15. Begin User Guide for documentation

0.4 Sprint 5

During sprint five the team tackled the remaining functionally, bugs, and quality updates in an attempt to finish the base project. Removal functions to clean out the database were added, and some client requested quality of life updates were added. Bugs and glitches were patch up in many functions. Another job of this sprint was to finish the rollover from sprint four since some payroll functions still needed work. Lastly the team tackled student registration since the student interface was dropped from the project requirements in sprint 4. The registration is still in progress at this time, as the team failed to complete this functionality in time for the end of the sprint.
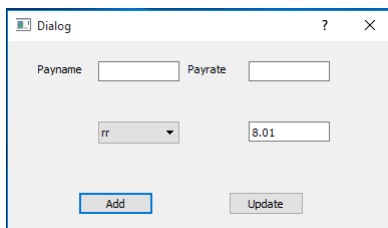
0.5 Prorated Refunds

The prorated refunds code will keep track of how much a class costed a student. If the student drops a class it will refund the remaining worth of the class to the students school credit field. This file has now been completed.

0.6   Enter Staff Hours

This page needed to undergo changes after meeting with client. The new version will allow a teacher to log their in class hours, office hours, trip hours, etc. These hours will then be paid out at their different rates respectfully. The hours will be logged as single numbers not by days or weeks. This function is now completed
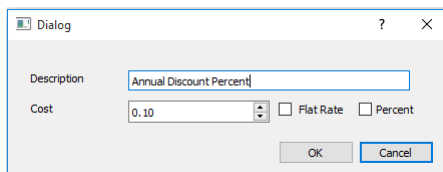
0.7   Enter Teacher Wages

This was handled in the database in previous sprints assuming one pay rate. However in talks with the client the academy pays different pay rates based on what part of academy work they are doing. So the page will need to be updated once the list of possible pay rates have been provided by the client. This pages and its updates are still in progress.

0.8   Enter Tuition Rates and Fees

Updated these pages to provide quality of life updates to the function requested by the client.

0.9   Bug Fixes

During the course of testing and the client meeting some bugs or inconsistencies were discovered within the project. The bugs found and fixed are:
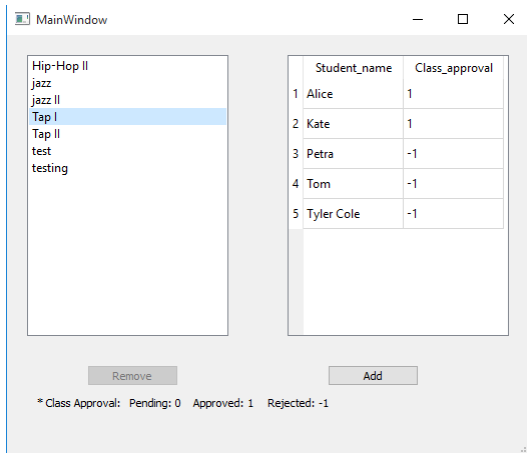
1. Student search was not showing all the students. Status: Fixed

2. Search edit need to be turned off and advance search modifications. Status: Fixed

3. Search, role, and schdules need to have dynamic not static times. Status: Fixed

4. Allow for time changes on schdule. Status: Fixed

5. Some of the back buttons in the teacher interface did not work correctly. Status: Fixed

6. Forms seem to be bugged at times, however can not recreate bug. Status: In testing possibly fixed.

0.10   Updates Crossover

One of the updates that needed to be completed this sprint was the need for admin and teacher updates to crossover if something was changed. This means that if someone is an admin and they update there phone number through the "update my information form" then the phone number will be updated in both the teacher table and the admin table. This way the system will not have conflicting information in different places.

0.11   Approved/Rejected Student Updates

A update requested by the client was the ability to see rejected students in the approved/rejected pages just in case the academy changed its mind about a student. The page underwent slight modifications to the way information was displayed to make this change efficiently

MainWindow

	Student_name	Class_approval
1	Alice	1
2	Kate	1
3	Petra	-1
4	Tom	-1
5	Tyler Cole	-1

Hip-Hop II
jazz
jazz II
Tap I
Tap II
test
testing

Remove   Add

* Class Approval: Pending: 0   Approved: 1   Rejected: -1

0.12   Admin List

In creating some of the updates it became clear that a specific admin list was needed to see who had admin access to the system. This was done through a simple list view of the names that can then be selected to display more detailed information.

0.13   Add/Remove Locations

Another requested feature was the ability to add and remove locations for classes in the database. This feature is necessary because the academy sometimes teaches classes in different places based on need. The team accomplished this using a simple dialog that connects to the location table in the database. When the user updates or adds a new class the location combo box provides a list of locations in the database and an add location option.

0.14 Removal Functions

These are simple functions that pull up a dialog box where the user can select a name and remove all the information related to that user from the system. The removal functions that exist in the system are: teacher, admin, student, and class. Teacher will remove that persons information from the system. Admin removes that persons admin rights and then ask if the whole user should be removed.

0.15 Sprint 5 Issues

Some issues that were encountered during this sprint were:

1. Spring semester ramp up - This was by far the biggest issue this sprint and the main cause of the failures this sprint. Other classes up to the point of blocking work on many projects. Staggered due dates meant that when the team wanted to work, there was always another projects that needed to be completed in a short amount of time. By the end much of our time had been taken and the sprint began to fall behind.

2. Requirement redefining - Due to drop the student interface the functionally needs to be added to the GUI which created some extra work for the team.

3. Failure to complete backlog - The team did not finish all of the assign backlog this sprint.

0.16 Client Interactions

Client interactions came in the form of required reviews and a demo of the prototype at the end of the sprint. Topics included:

1. Progress reports on where we are in the project

2. Asking questions to refine functionality, understand academy needs, and update requirements.

3. Discuss the future of the project and idea on how to execute them.

4. Prototype the current version of the prototype.

0.17 Group Meeting

The group has a standard meeting time of TTH 10:00 - 12:00 and once during the weekend and at home as needed.
   Meeting can continue pass these times if needed, and other times during the weekend. However extra weekend times fluctuate and do not remain constant from week to week.

0.18 Work Distribution

Marcus:

1. Bug fixes

2. Removal functions

3. Admin and teacher update crossover

4. Ignore Case on address forms

5. Add rejected students to appoved/ reject and provide current status

6. Add/remove class location
7. Prorated refunds
8. Quality of life updates
9. Sprint report
10. Start User Guide
11. Trello management

Dicheng:
1. Bug fixes
2. Pay rates
3. Admin list
4. Staff hours
5. Student registration (in progress)
6. Wage calculations (in progress)

Together:
1. Updated database construction
2. Update prototype structure
3. Coded some functionality