UpQuest
Low-Fi Prototype Testing Report

Team Saraswati:
Tyler B. (testing + report + sketches)
Matt C. (testing + report + sketches)
Kat G. (testing + report + sketches + presentation)

Word count: ~2100

Introduction

Mission Statement / Value Proposition

The student data that keeps on giving

Problem/Solution Overview

Teachers face a problem that they start the school year with limited information about the knowledge level of their students. In the remainder of the school year their students progress is tracked on an ad-hoc basis with quizzes. Our application serves to provide a method for teachers to give formative assessments to their students to understand the materials that they already know and what they need to teach. This would help to customize the learning experience for each classroom and each student.
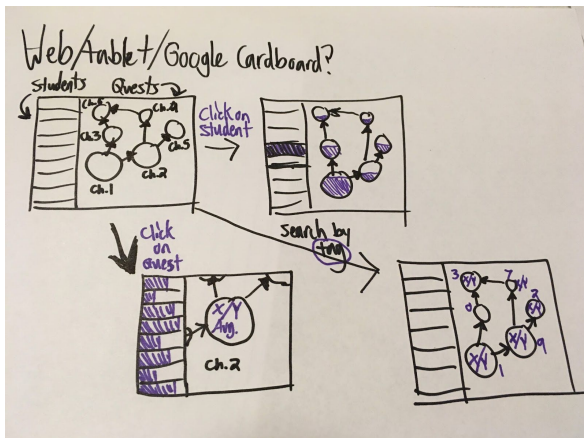
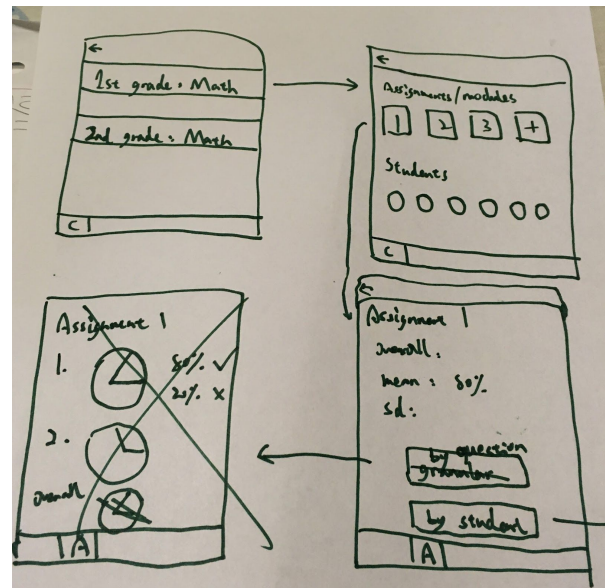**Sketches**
Concept Sketches
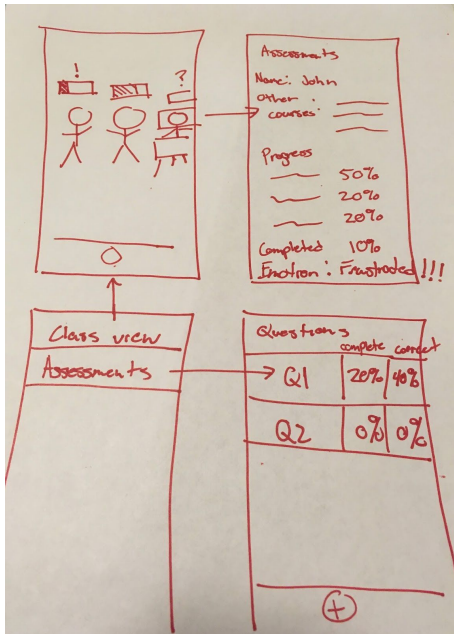
Concept Sketch Series #1 : Augmented reality capability for iphone
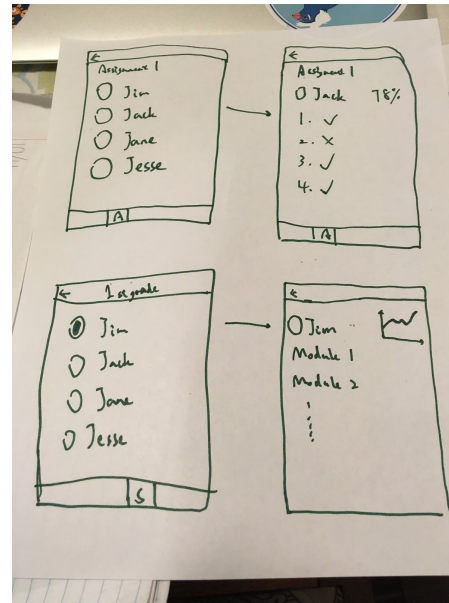
Concept Sketch Series #2 : Teacher iphone student view

Concept Sketch Series #3 : Concept maps on a tablet display for modules

Concept Sketch Series #4 : Teacher iphone quiz/module view
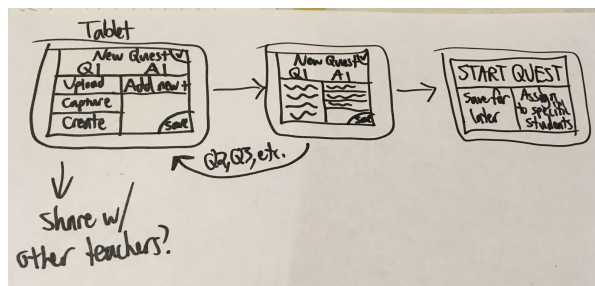
Concept Sketch Series #5 : Teacher tablet photo quiz importer
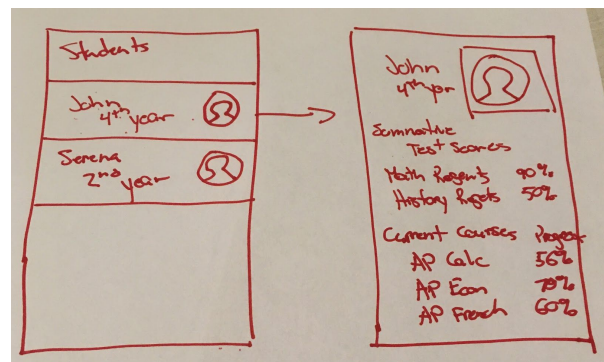
Concept Sketch Series #6 : Teacher iphone student profile view

Storyboards / Detailed UI Sketches

Teacher mobile app: Student data and
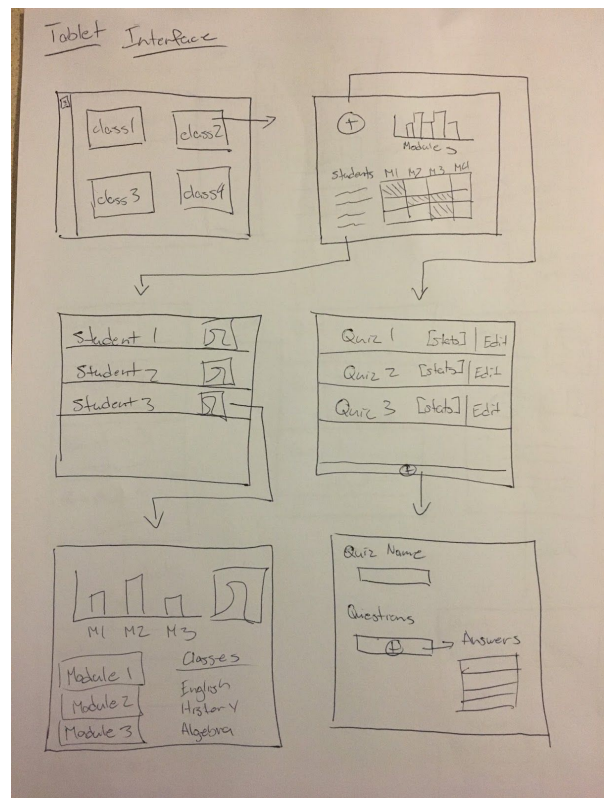
Teacher web app (optimized for desktop

quiz creation interface	and tablet browsers): Quiz creation and results dashboard

Selected Interface Design

Task 1

Task 2

Glasses

Algebra

European History

Biology

Algebra
Quiz 1 completed — Score 75%
Quiz 2 completed — 20%

Algebra
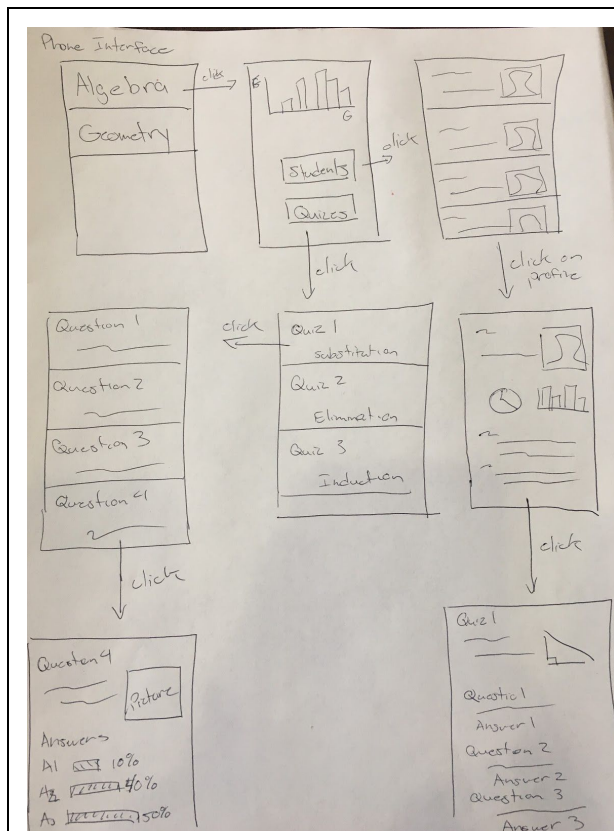Supporting Material — Score — Resume
Quiz 2 completed — 20%

What is x? 3 / 4
A. 6
B. 14
C. 15
D. 20
Next →

Incorrect :(
Supporting Material
Explanation
Retry
Next →

Correct!
→

Task 3

Students
Alex | Ann | Bob

Alex
semester [____]
Score 100, 60
M1 M2 M3 M4 Module
M1 60 | M2 90 | M3 70

M1
Q1. 3 / 4 ?
✓ A. 5
✗ B. 10
C. 7
D. 9
Q2. 1 / 1 ?
A. 2
B. √2
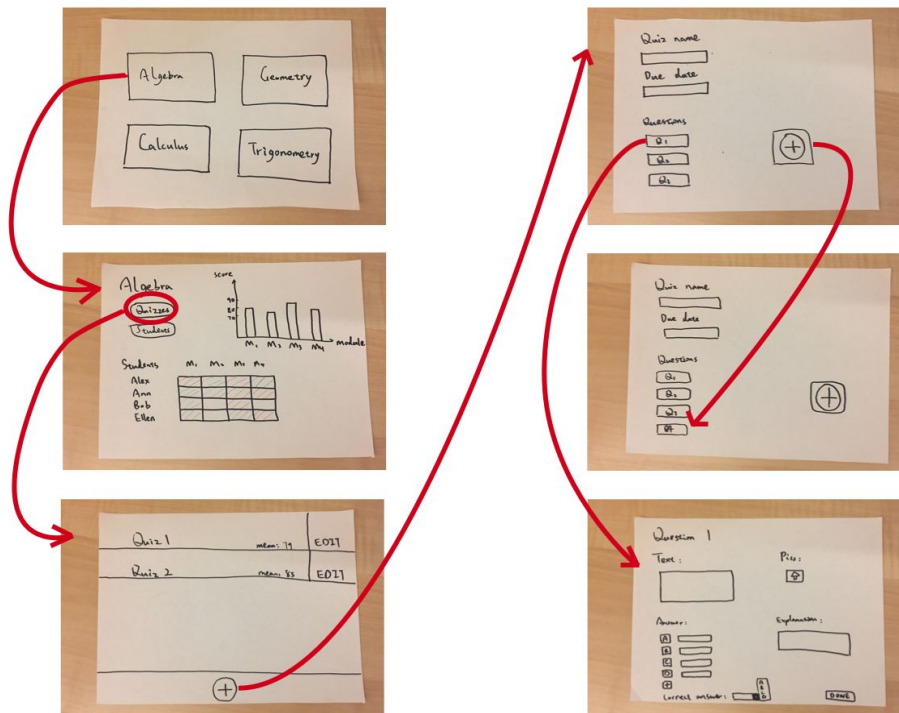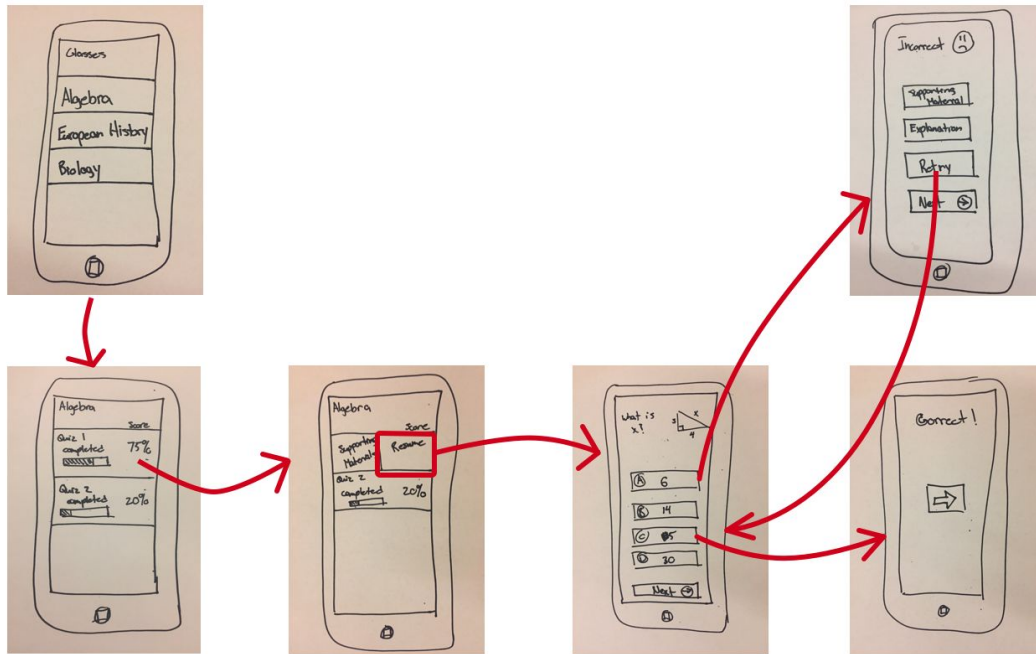
Reasoning for selection

This design was chosen based on the power of its data-visualizing abilities for teachers and the simplicity of the mobile assessment interface for students. We ended up going with a couple of assumptions: (a) that teachers wouldn't derive enough benefit from our functionalities on a smartphone, and (b) that teachers should be able to mandate assessments and then assign (or not assign) actual credit as they wished. This let us keep the student's end clean and straightforward while allowing immediate feedback capabilities. For the teacher, the focus was on giving the flexibility to upload any type of assessment and presenting the results of assessments in a small number of highly useful ways to avoid clutter and confusion.
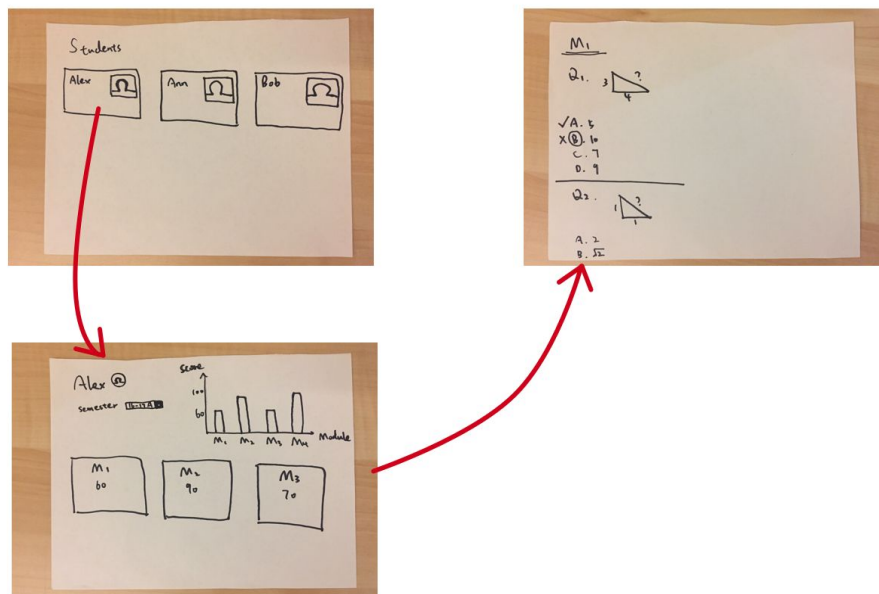
One benefit of the design we went with is that it lets teachers see their students' knowledge on individual vs. class-wide levels as well as single-concept (or even single-question) vs. course-long levels. A drawback of this is the challenge that it creates as far as not overloading the teacher with options and with making those options intuitive. The teacher mobile app had the advantage of letting the teacher see grades and work on assessments while on the go, which many teachers often are. One pitfall, as we mentioned, would have been that the high-level class data and quiz creation interfaces would have been hard to keep uncluttered; another more glaring one was that teachers might not use it. As we later learned, teachers prefer to do most if not all class-related work on their laptop or desktop computer.

Prototype Description
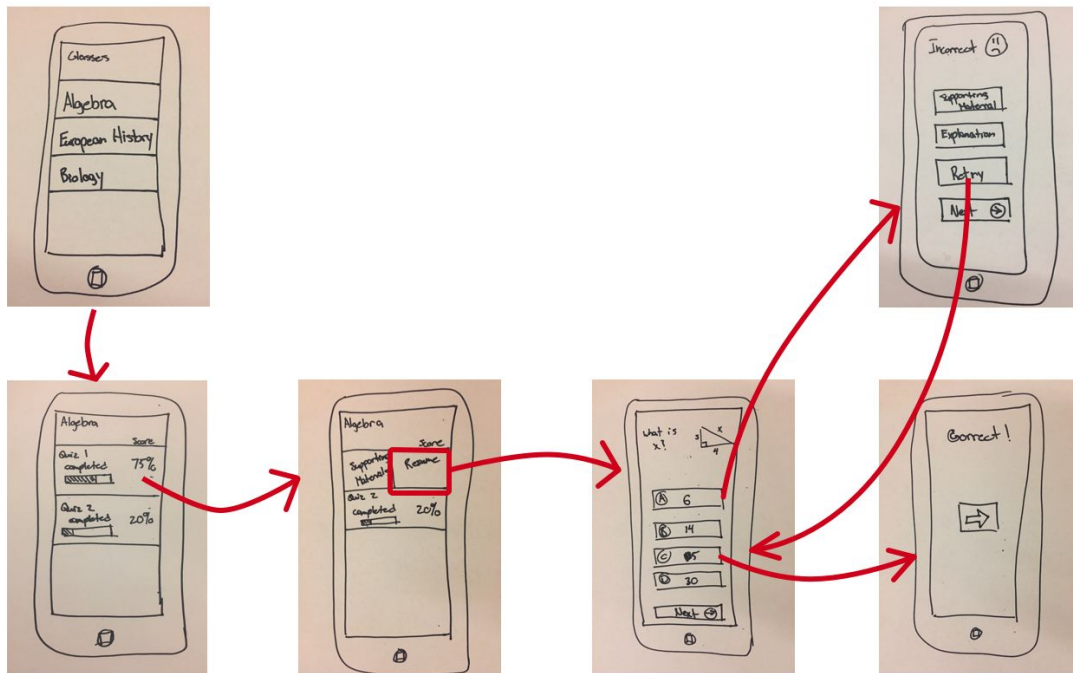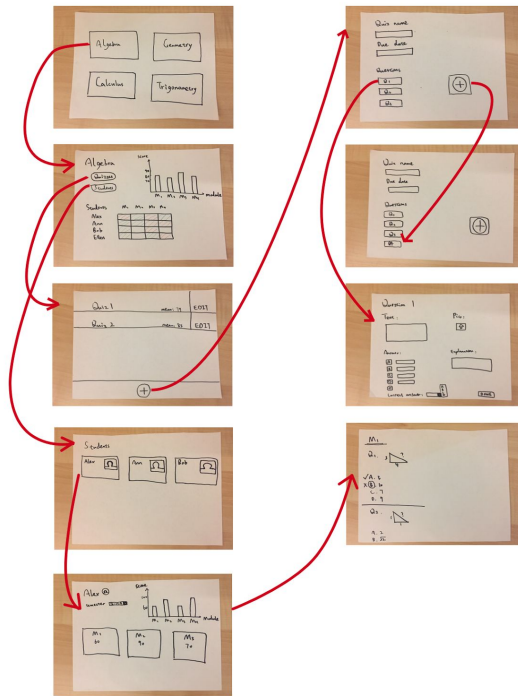
Teacher's home page	T's class page	T's quizzes page	T's creating quiz page
After clicking plus sign	T's creating a question page	Student's home page	S' class screen
After clicking a quiz	S' quiz-taking screen	S' got it wrong	S' retried and got it right
T's students screen	T's individual student	T's individual quiz of	

	screen	an individual student screen	

We created a paper and sharpie prototype with two interfaces -- web/tablet interface for the teacher, and iPhone interface for the students.

Teacher interface:

A teacher logs in and sees all her classes on the homepage. Clicking on one of the classes leads her to the particular class' page. On the class page there are two buttons that will lead her to the *quiz* page and the *student* page. There is a histogram of the class average scores, as well as a spreadsheet containing each student's performance overtime. A red square indicates that the student has fallen under a threshold set by the teacher, while a green square indicates the opposite. A blank square means that the student didn't do the particular quiz. Clicking on the squares can lead her to a particular student's particular quiz page.

Clicking on the *quizzes* button, the teacher can see all the quizzes she made, as well as add quizzes. When creating a new quiz, she can type in or upload screenshots for each question.

To track student progress, she can go back to the class page and tap on *students* button or the colored squares. Say she clicks on *students*, she will see all her students listed in tiles. Clicking on a student's tile leads her to the *student* page, where she can see a student's scores in the current semester or other semesters by scrolling the *semester* bar. Clicking on one module leads her to see how the student fared in a particular quiz.

Student interface:

The student's homepage contains all the classes he's currently enrolled in. He can click on a class and get to the *class* page, where he sees all the quizzes of the class. He can see his scores and the percentage completed in each quiz. Clicking on a quiz gives him the options of resuming or reading support materials. Clicking on *resume*, he will see one question at a time. Giving a wrong answer leads him to a set of options, while giving the right answer will lead him to the next question.

Method

Participants
#1. High school Math teacher for 2 years
#2. High school Math teacher for 4 years

#3. High school Earth System Science teacher for 4 years
Recruited by cold-email.
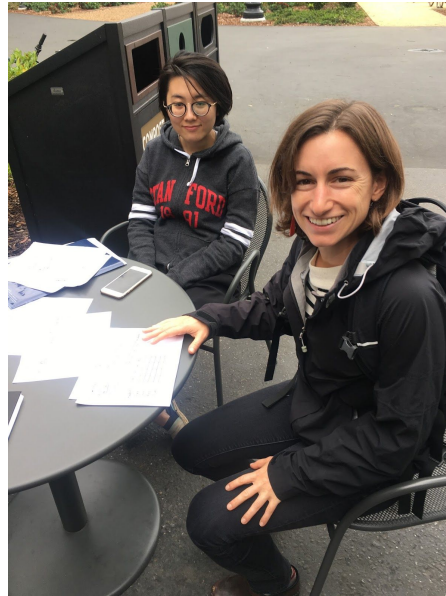
Environment
We did one test at Quinn High School, one at the Graduate School of Education at Stanford, one at Coupa Cafe near Green Library.

Tasks
Task 1: The teacher gives a quiz to the students.
Task 2: The student does the quiz.
Task 3: The teacher checks the student's progress

Since we only heard back from teachers, we had the teachers play the student's role in Task 2.

Procedure
All three of our tests follow the same procedure. Tyler introduces our interface first, and then Matt facilitates the user to go through the 3 tasks. Throughout the tests Kat observes and takes notes. We were unable to do a demo using a different task because our interface only supports the tasks we have.

Test Measures
- **Contradictions in behavior** and anything the interviewee did differently than what we thought was the "correct" movement. This helped us see where the user's attention was drawn and replace our intuitions with theirs.
- **Where they got stuck**. If the person had to stop, that means they had to think about what to do, which is a sign of an incorrect assumption about the flow of the product.
- **Points of confusion** where the interviewee expressed their confusions or asked one of us for guidance.

Results

The results we got from the participants are generally positive. Teachers were able to move on in every task, though sometimes a bit unsure of what to do.

Below are the major issues we found in the tests.

Task 1:
- When adding quiz questions, the 3 questions we put on the screen as placeholders are confusing, because teachers didn't know if they should click the *add(plus sign)* button or click on the questions.
- Due date is not selectable.
- On question 1 screen, the scroll bar for correct answer sometimes was ignored, or created confusions.
- Didn't understand the *explanation* text box.

Task 2:
- The home screen for students did not have notifications.
- On the quizzes page, they thought there should be a new quiz. But they only found 2 uncompleted quizzes.
- The *next* button under the 4 answers was ignored once.

Task 3:
- misinterpreted the display of right answer as "the student got it right on the second try"
- Hope the wrong answers make more sense, so that teachers know why students get it wrong.

Discussion

Our usability tests revealed a wealth of helpful specifics and high-level insights about the strengths and weaknesses of our designs so far. The testers helped us focus our attention on features that could use extending and expanding so as to better address the biggest pain points as well as features that could be moved, streamlined, or removed altogether.

On a completely functional level, our buttons for "action" items like creating new assessments (as the teacher), starting new quizzes (as the student), and moving from screen to screen could have been clearer and more pronounced in general. In our discussions, the testers were also fairly clear on the fact that clunky interfaces and repetitive/administrative tasks were some of the biggest problems that they normally face. Making it easy to perform such actions, like bringing in questions from external assessments or exporting assessment results in spreadsheet form, could be very fruitful.

Likewise, in the student's mobile app, the problems were more about the specific implementation we started with. The flow for taking a quiz was relatively straightforward, as it's meant for students going through assessments already prepared by the teacher, but it produced its own share of interesting responses. For example, after choosing a class, it wasn't clear

whether each quiz in the list for that class had been completed or could be resumed, hinting that we could remove some of that information on past quizzes or put it elsewhere. One teacher brought up another good point that might be addressed with functionality changes: how would we prevent students from cheating?

All three testers being teachers, and so the interfaces for creating assessments and analyzing their results were scrutinized the most. In general, the aspects of the teacher app that needed the most refining had to do with formatting and layout rather than the nuts and bolts. Task 3 in particular seemed somewhat ambiguous, as it wasn't 100% clear from the flow how one was "supposed" to reach the most comprehensive view of the class's knowledge.

Interestingly, the testers diverged in how they wanted to create assessments; it seemed that being able to pull from common question "templates" and creating your own questions from scratch would both be valuable. They did, however, express a general desire for control over parts of the assessment process that they saw as more personal or variable from teacher to teacher, like the names of particular units, assessments, or "tags" (discussed later).

All in all, our belief was affirmed that teachers really do care about gathering the most and the best information possible from mountains of gradebook data for the purpose of differentiating and tailoring content. Furthermore, they had some really insightful requests and useful suggestions about how to do so. One tester asked how she might break students up into sections (e.g. based on a particular idea that certain students are having trouble with) or see how many attempts a student made on a certain question. Two testers independently touched on notions like giving relevant explanations or links to online content after getting questions wrong, or "tagging" specific questions or assessments with short labels a la Piazza so that you can sort your class data in a customized way. The teachers really wanted to know why students got particular answers wrong and what trends exist in their students' knowledge, which will be a strong guiding principle in our designs going forward.

Appendix

Word count:

Demo Script

Hi [Enter Name]! Thank you for your participation in this project. The goal of this project is to test the usability of a solution that we have created for teachers to track their students progress. The idea is that teachers need a way to generate formative assessments for their students to understand that individual students are struggling so they can customize their teaching methods accordingly. In this exercise we are going to present you with three different tasks. For each task you are going to land at a start screen and you need to interact with our paper based user interface as though it were a real device. Feel free to speak aloud so that we can better understand your thought process as you work through the interface. Before we start please look over and sign this consent form, detailing what the information collected here will be used for.

For the first task your goal is to create a quiz on a tablet application for your students in a algebra class that you are teaching.

For the second task your goal as a student is to complete the quizzes presented to you.

For the third task your goal is to look over the results of the quizzes given to your students and look at the results of a particular student on a particular quiz.

Notes from three tests:

- Test #1 Observations:
    - Task #1
        - not able to select due date
        - got stuck when inputting text for the question
        - asked a question with no answer
        - didn't understand the "correct answer" scroll bar
    - Task #2
        - hope there's a quiz 3 he just created
        - ask about scribble on the screen
        - no supporting materials
        - didn't get to click on next after answering the question
    - Task #3
        - task is ambiguous
        - no aggregate
        - order was messed up

- Test #2 Observations
    - Task 1
        - Confused about how to create first question (click on question or plus sign button?)
        - Unclear how to create another question after the first
    - Task 2
        - How would student know there's a quiz? Notification about particular class maybe?
        - What would it look like when the student is starting a new quiz?
    - Task 3
        - Would want to click on the lowest score first (problem student)
        - Would want to somehow identify which answers reveal which thought processes
        - Correcting most common mistakes is important
        - Bag system might help?

Overall

"Reteach concept X" notification based on aggregated data (like from tags)

Would want to customize/differentiate homework based on quiz results

How to avoid cheating?

Lesson plans tend to be diverse, personal

- Test #3 Observations
  - Task #1
    - People not using questions tab
    - Understood the picture button
    - Did not know what explanation box is used for
  - Task #2
    - Screen with all quizzes is unclear - are those already taken?
    - Did not click on the next button after the question
  - Task #3
    - Not sure about what M stands for (module)
    - Understood display graphic
    - Did not know how he got the answer wrong - how many others got the answer wrong in the same way?
    - How would you display multiple tries?