ClassLens

Class feedback that's *actually* useful

Team

Amy Liu	**Front-end Engineer**
Hope Casey-Allen	**Back-end Engineer**
Yash Tambawala	**Product Manager**

Value Proposition

Class feedback that's *actually* useful

Table of Contents

Problem and Solution Overview

Problem

Currently, there is no organized way for teachers and students to understand if and how specific course objectives are useful in downstream classes or jobs.

Solution

A web-based platform for former students to give feedback on skills gained in a class and how those skills are being used in downstream classes and jobs. Teachers can use this feedback to see how to revise their courses, and students can use it to see whether a course is valuable for their short and long-term goals.

Tasks & Final Interface Scenarios

Task 1: Teachers browse feedback (easy)

A teacher who logs into our platform can browse feedback for the classes they teach and curate the feedback they receive to revise their classes.

Our needfinding interviews with educators at Stanford and other institutions revealed that they valued feedback from former students who could say how a specific skill in the class was useful for them later on. There was a definite need for educators to ensure learning objectives were being met and were useful downstream.
Also, our interviewees expressed that they would often seek out feedback from former students, but there is currently no organized way for doing so.

In this task, a teacher can do the following activities:

- Find a class they teach - Our system will be integrated with existing teaching records which will allow the platform to match teachers to their classes. (Fig. 1)

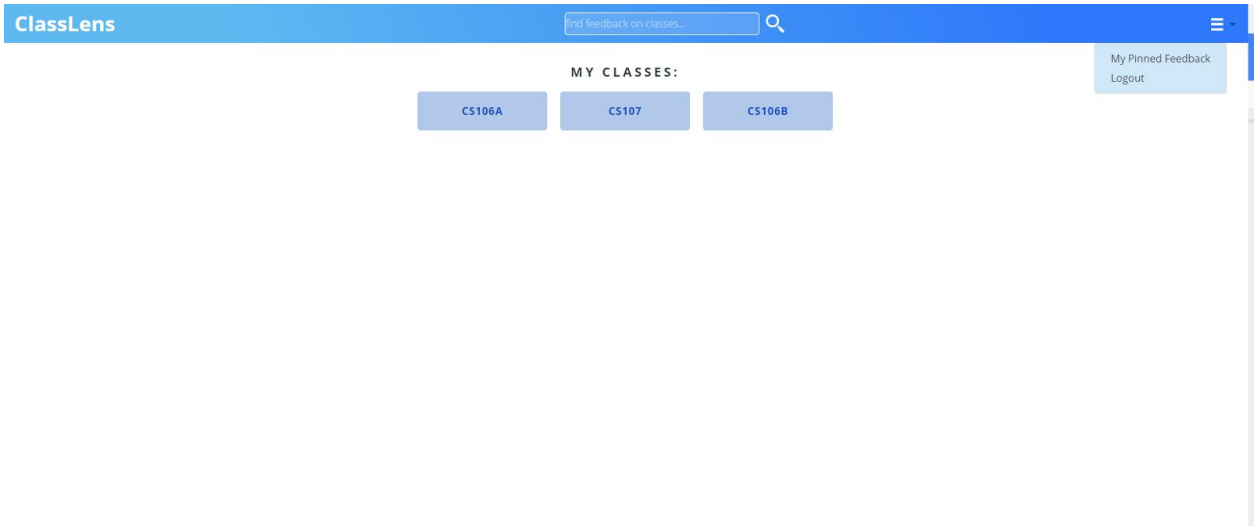


Fig. 1: Find classes you teach

- View their class feedback (Fig. 2)

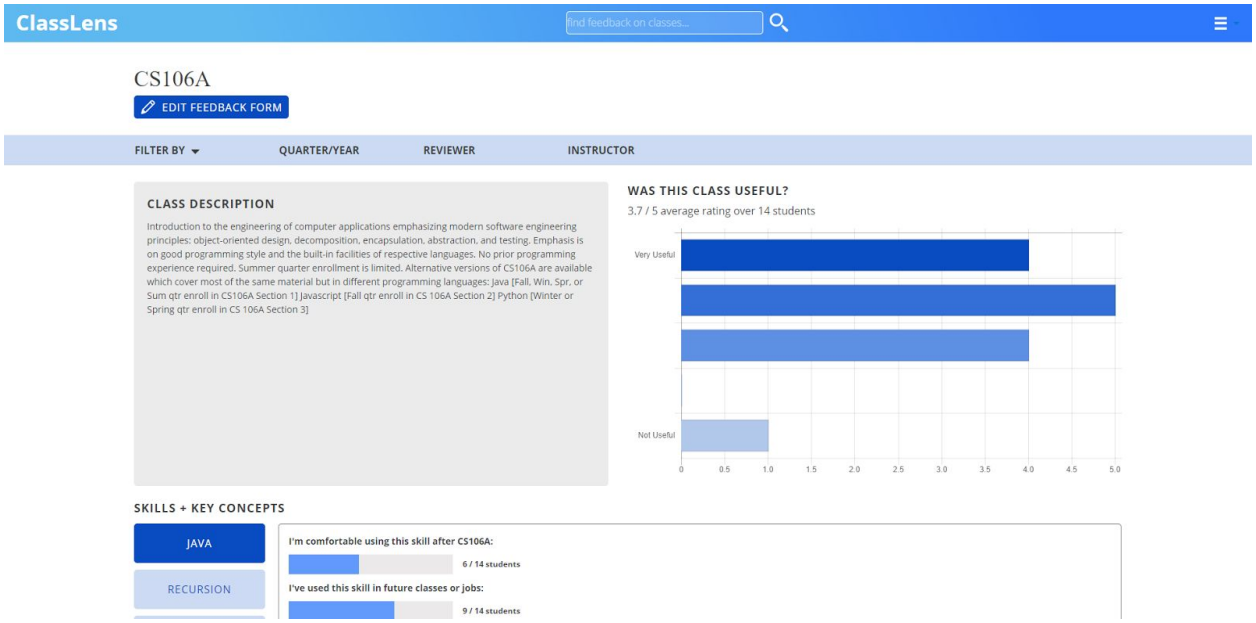


Fig 2. Class Feedback

- Pin useful feedback to a board according to the nature of the feedback (positive, negative, or neutral) (Fig. 3,4)

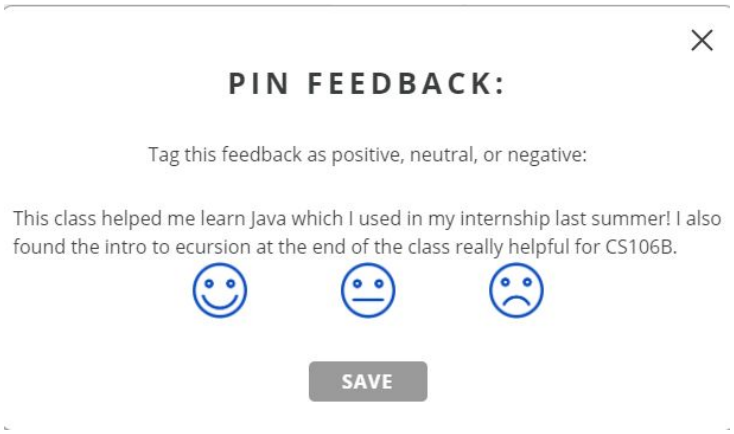


Fig. 3: Pin Feedback

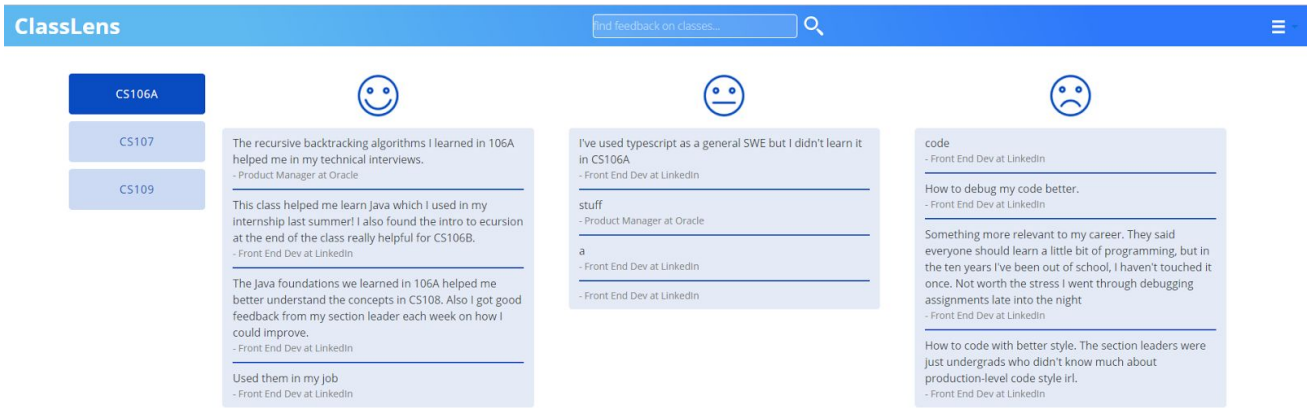


Fig. 4: Pinned Feedback List

- Edit and preview the feedback form sent out to former students of the classes they teach (Fig. 5,6)

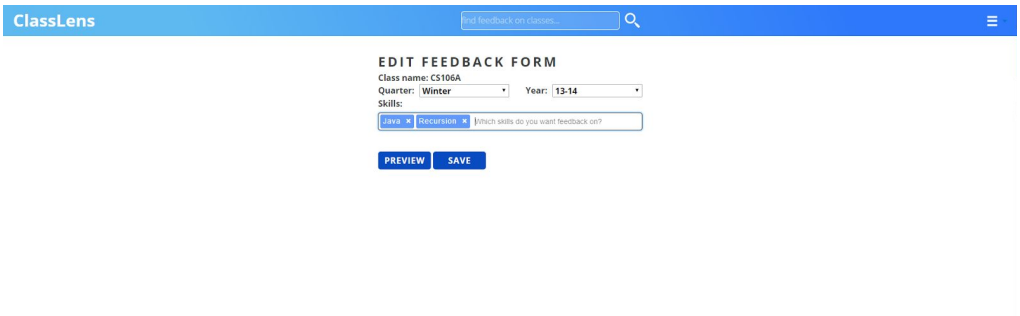


Fig 5: Edit Feedback Form

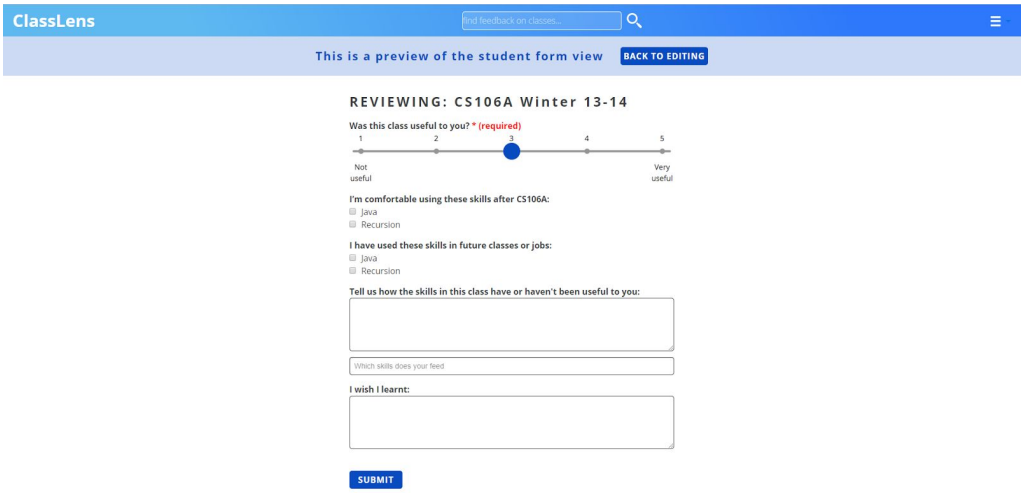


Fig. 6: Preview Feedback Form

Task 2: Former students give feedback on classes (medium)

Former students can log into the platform and review the class they took in the past. We use former students to refer to Stanford alumni as well as current students who took the course in a previous iteration.

Our needfinding interviews showed that both current students and teachers want confidence that learning objectives are being met and are useful for students later on. Feedback taken immediately at the end of the class, which is how Stanford course reviews are typically done today, is insufficient for meeting this need. Hence, we decided to include this task, providing an organized, simple way for former students to give feedback on classes they have taken.

In this task, a former student can do the following activities:

- Find the course they had taken in the past (Fig. 7)

Fig. 7: Course Review Hompage

- Fill out a review form for the class (Fig. 8)

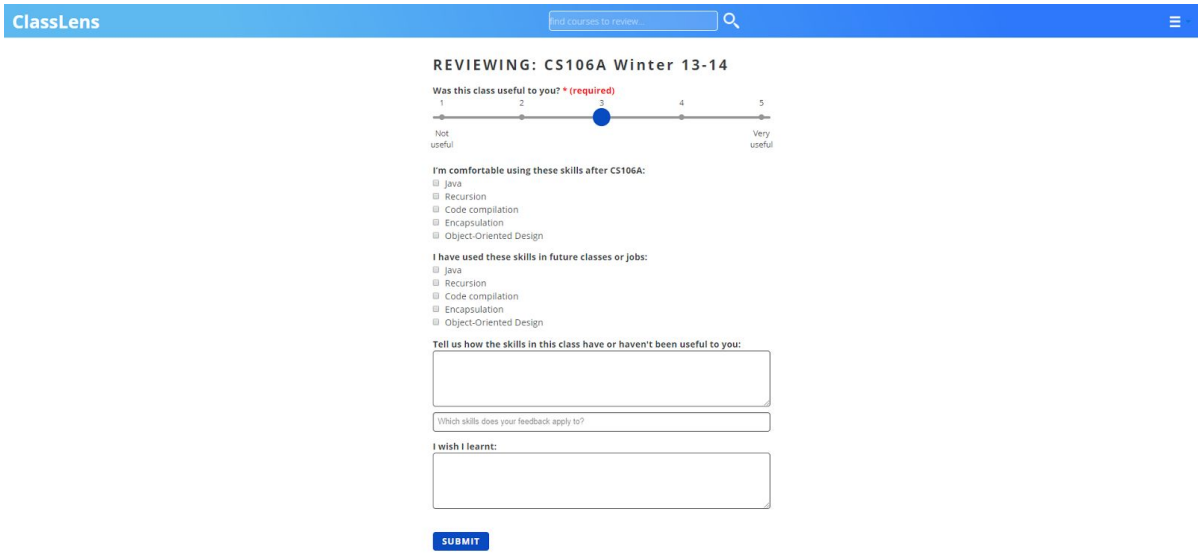


Fig. 8: Feedback Form

Task 3: Students explore careers and classes (complex)

Current students can access our platform and explore feedback given by former students on any class. They can also explore the skills taught in courses and see how they map to various careers they might be considering using the “Explore Graph” feature.

As of now, current students find it difficult to gauge whether a course is useful for future classes or jobs just from the class feedback they read. Building this confidence in course objectives can be achieved by communicating feedback from former students to current students. This the reason behind including this task in our final product.

In this task, a former student can do the following activities:

- Navigate to a career, skill, or class using the career graph (Fig. 9)

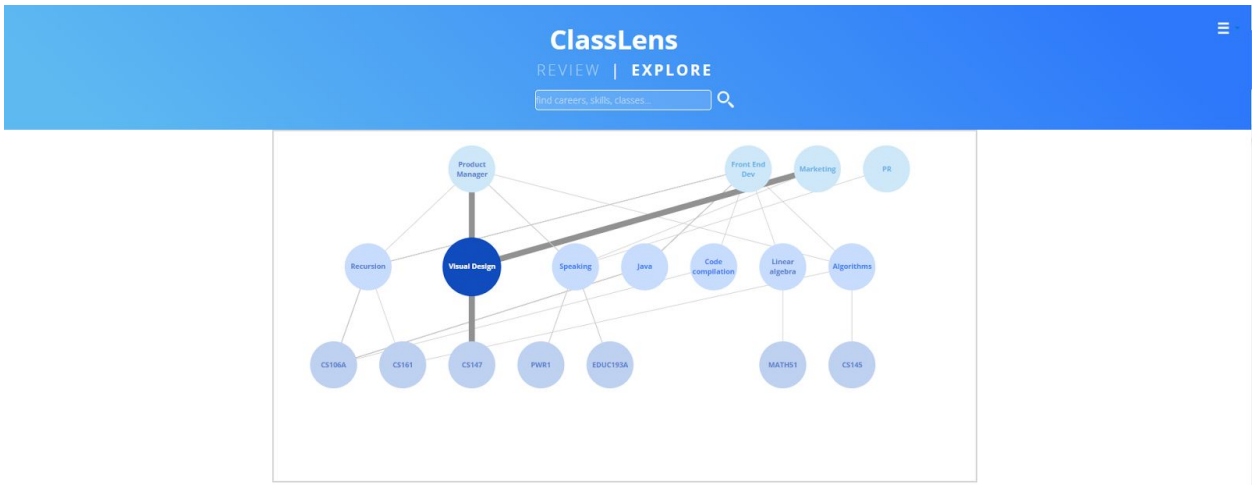


Fig. 9: Explore Page

- Browse feedback for a class they are interested in (Fig. 10)

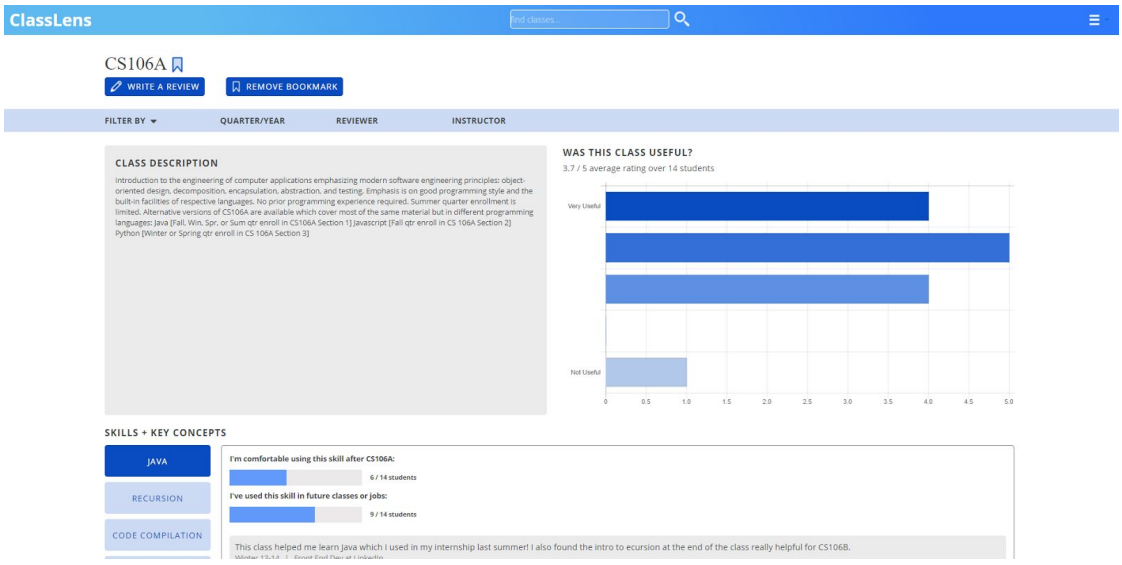


Fig. 10: Course page

- Bookmark a class they might be planning to enroll in (Fig. 11)

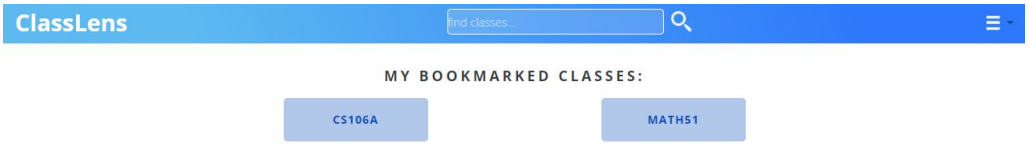
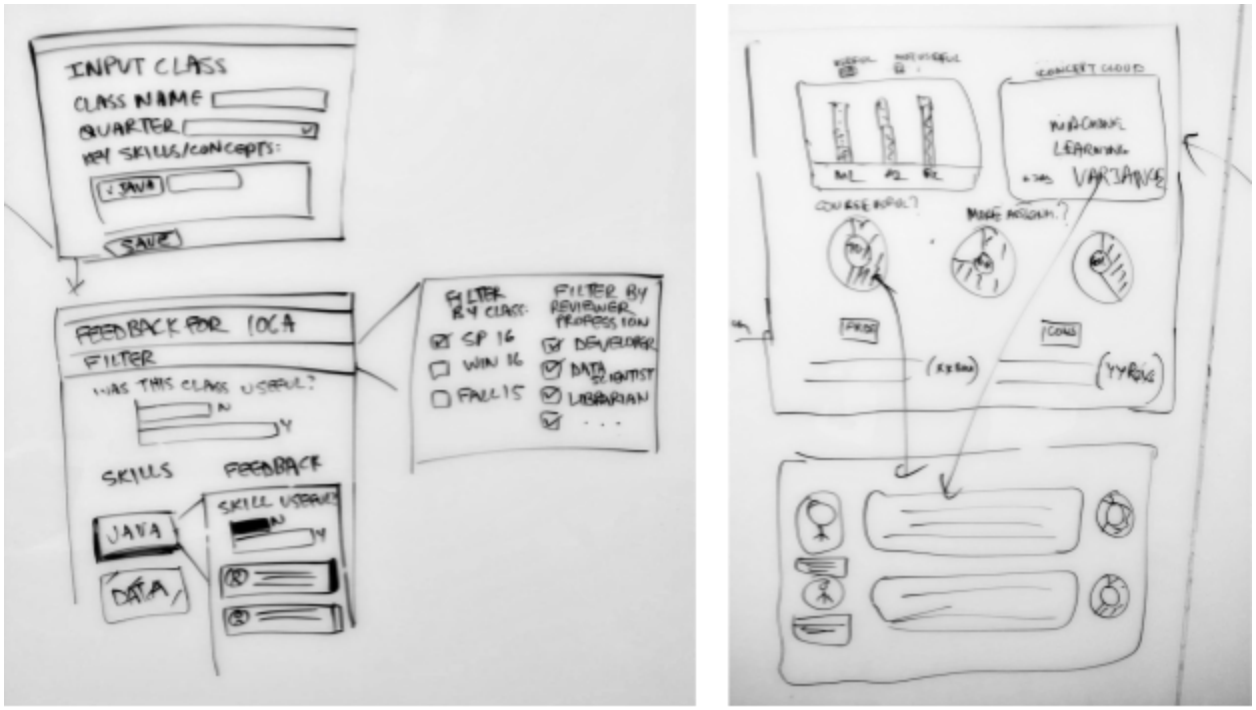


Fig. 11: Bookmarked class

Design Evolution

Concept Sketches



Chosen Rejected

Fig. 12: Chosen vs. Rejected Sketch

We created several initial sketches to explore how we could meet the user need in the best way possible (Fig. 12). Out of all our sketches, we chose a skill-based feedback layout. There were three reasons for this -

- Skill focused: Our need finding showed that professors and students want to see the feedback tied to skills/lessons they learn in class. Needfinding results showed that teachers wanted to see feedback categorised according to skill learnt.
- Needfinding indicated that users would like to see some general information about the course as well as the ability to easily deep dive into the feedback.
- This layout required less interactions to view the feedback.

Low-fi Prototype

During our low-fi prototyping, we conceptualized the basic layout and flow for our 3 tasks. We allow teachers to organize feedback in a more useful form for them individually by adding features like pinning feedback and the ability to edit the feedback form sent to former students. (Fig. 13)

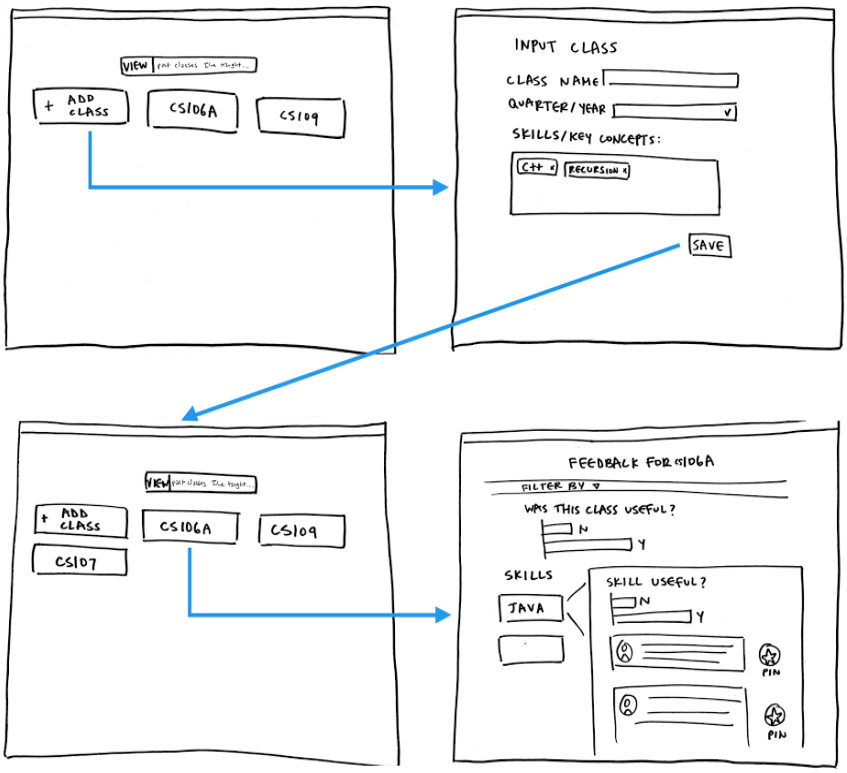


Fig. 13: Teacher Storyboard

For former students, we tried to keep the feedback form experience as seamless and easy as possible by keeping the form to just few questions on a single page. (Fig. 14)

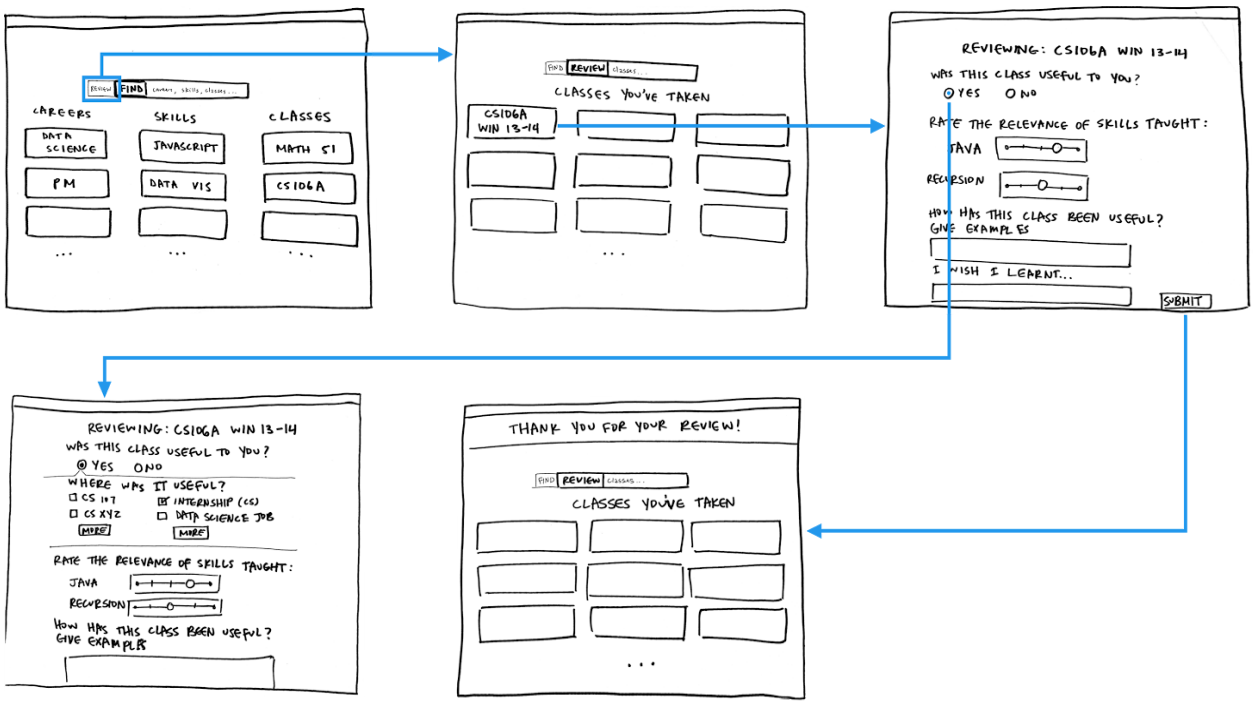


Fig. 14: Review Storyboard

Exploring the connection among career, skills, and classes was made possible using the explore graph. In addition, we inferred that students might use our platform for planning their coursework, and thus we added a feature for students to bookmark classes so they could quickly reference them in this process. (Fig. 15,16)

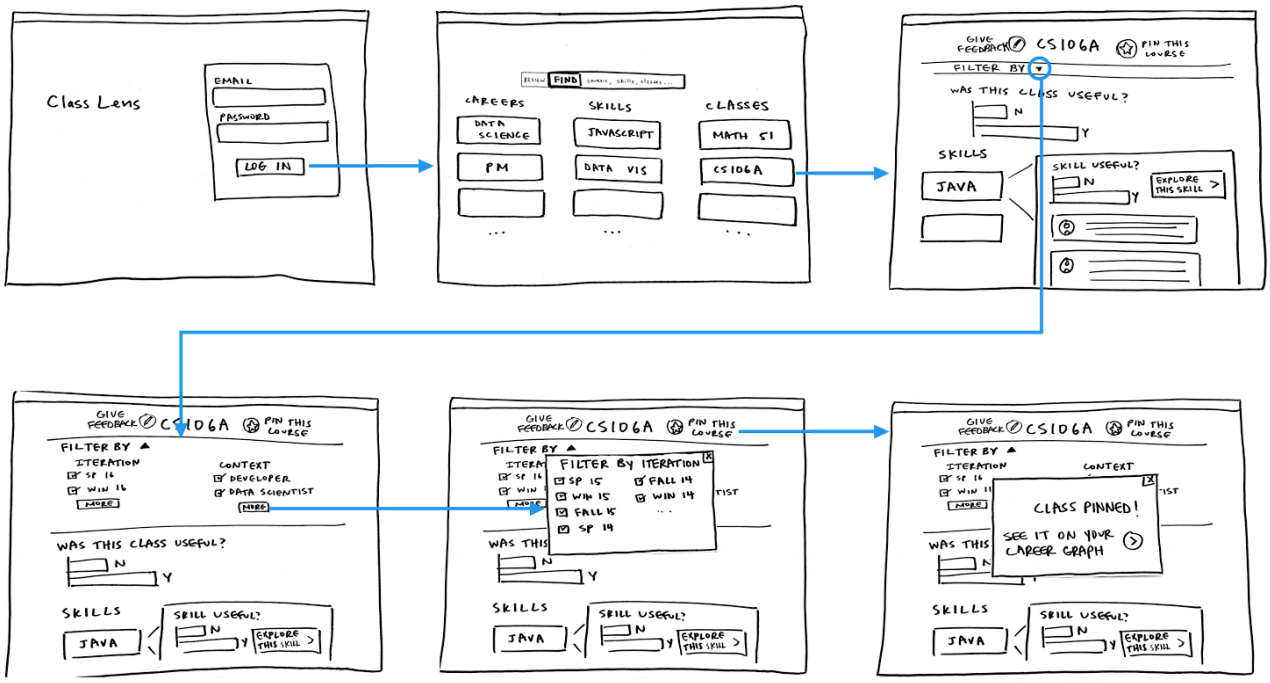


Fig. 15: Pinning a class

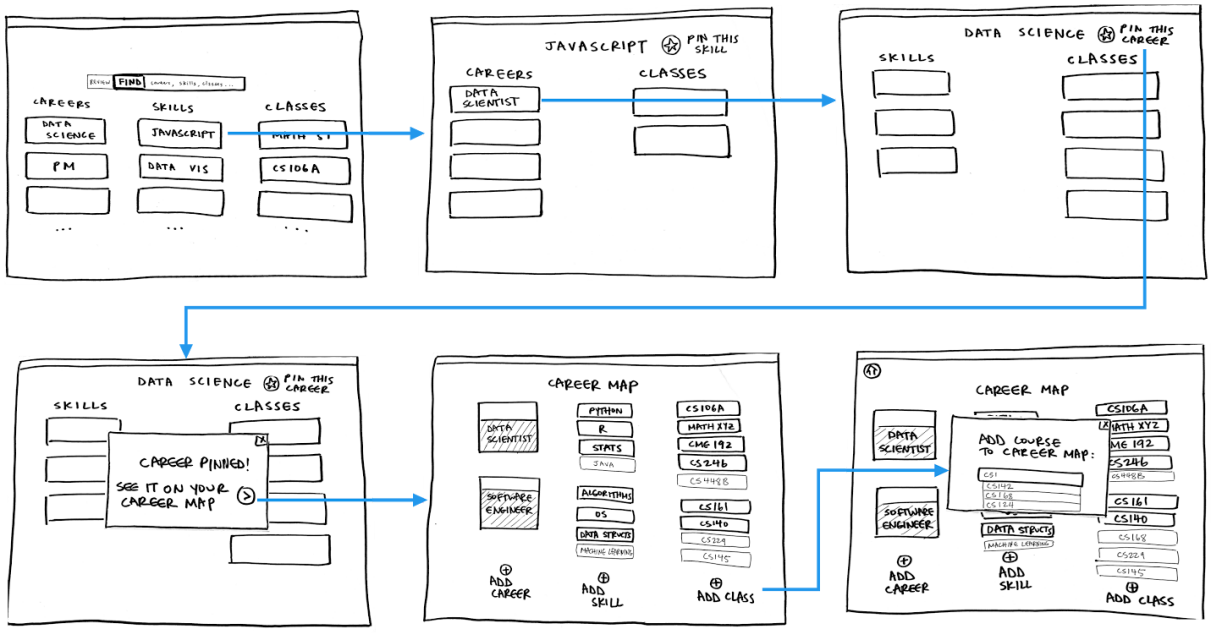


Fig. 16: Adding a career to map storyboard

Design Changes and Medium-Fi Prototype

Based on feedback from our usability tests with participants using our low-fi prototype, we revised our design and used the revision in our medium-fi prototype. There were four major categories of those changes.

New Homepage Design

Initially, our career map page lacked a sense of hierarchy when moving between career, skills, or classes. Also, the “review” button was almost impossible to find. A few participants even felt that the career map could be merged with the homepage of the explore section. (Fig. 17)

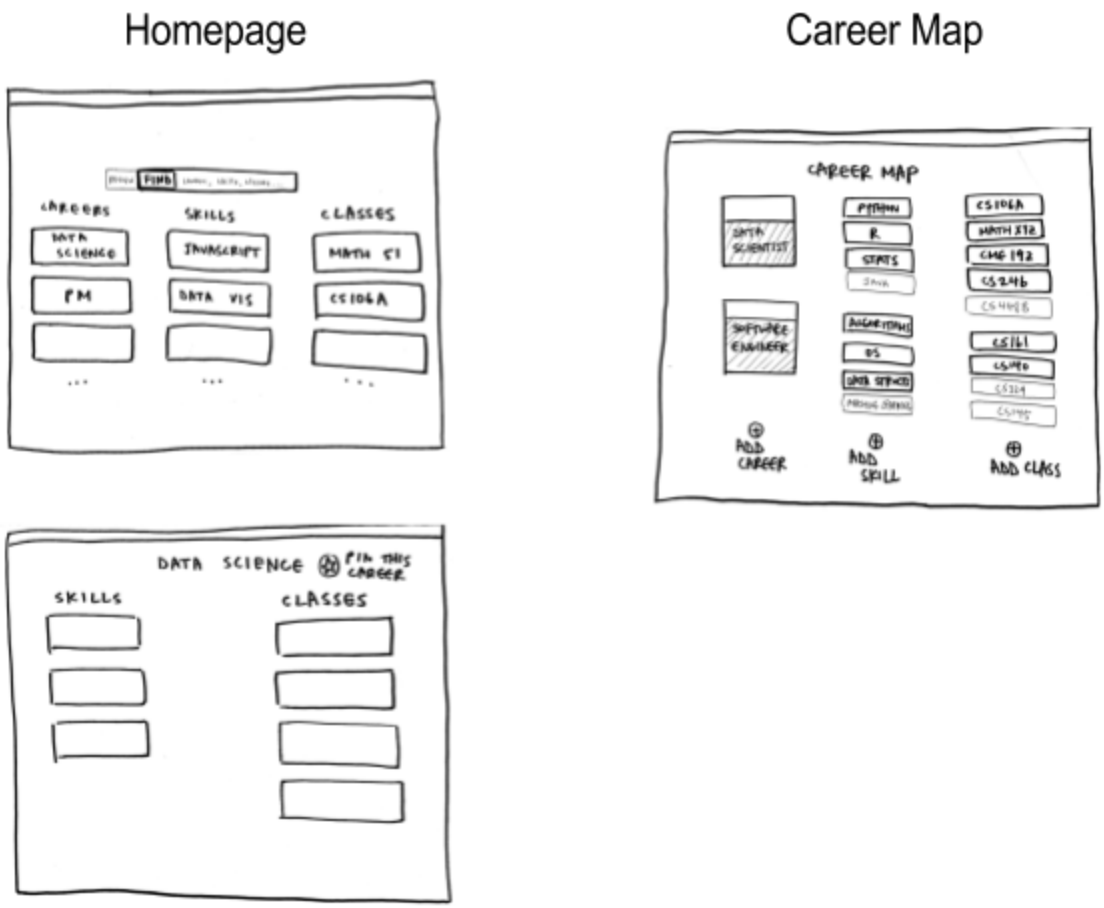


Fig. 17: Old Homepage

We adopted all of these suggestions and unified the homepage and career map into one page with a connected graph visualization. We believe improving the explore experience is critical for current students to gain value out of our platform. Thus, we made sure there weren't any redundant screens or clicks anywhere. Additionally, the vertical layout clearly indicates the hierarchy of how careers, skills, and classes are linked. (Fig. 18)

We also clearly separated the "Explore" and "Review" tabs on the student homepage, allowing students to easily discover these tasks and toggle between them. (Fig. 18)

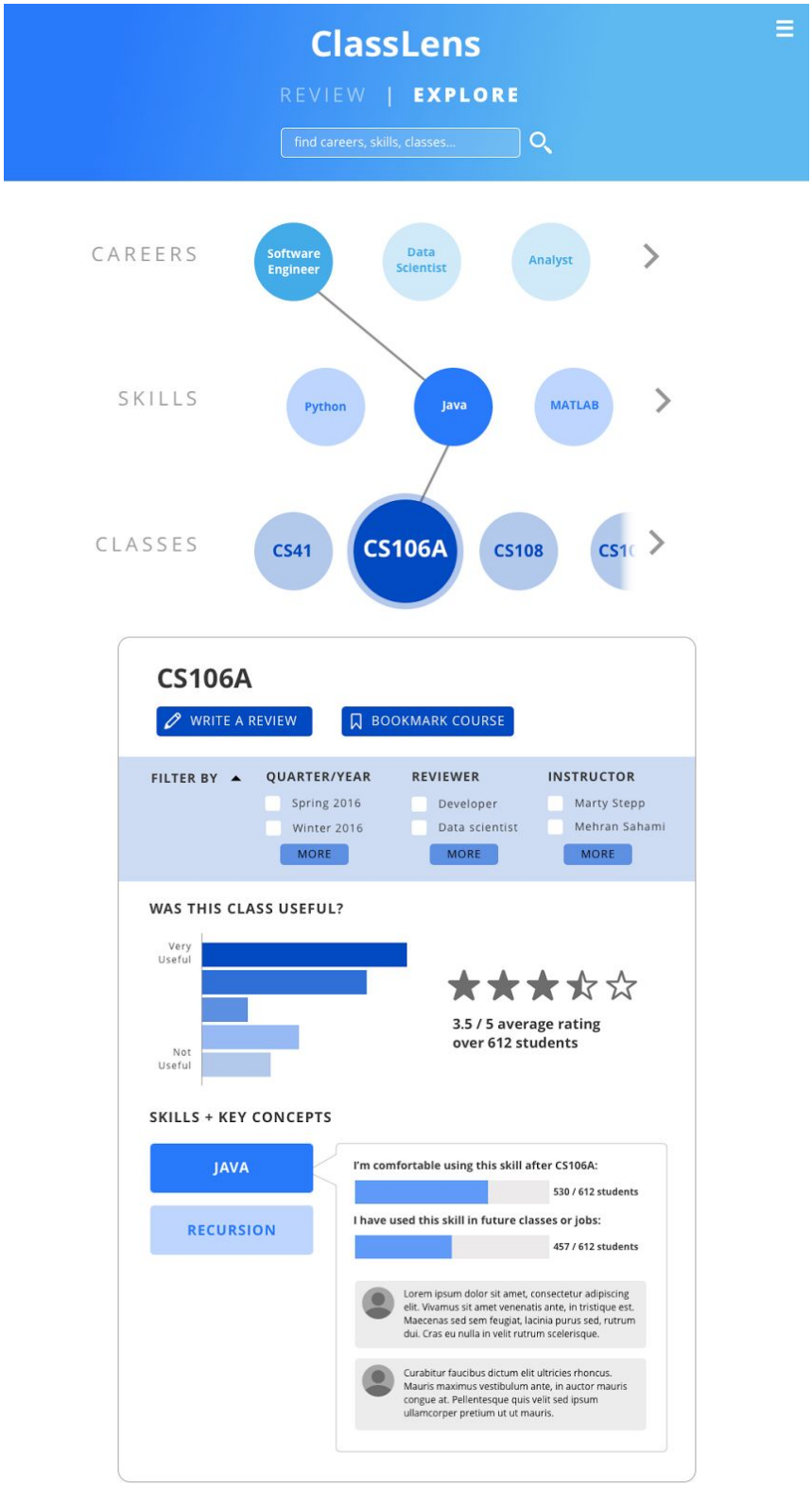


Fig. 18: New Homepage

Though people enjoyed the interaction of pinning skills and careers, we found that it took away from the core focus of the task which is exploring courses. We realized that the idea of pinning is actually a fourth, separate task of “students plan courses/careers”. We decided we wanted to focus on our original three tasks and save the pinning and planning process as a possible add-on feature.

Students will still be allowed to save courses they’re interested in to a simple list of bookmarked courses, but the intention is for the actual course/career planning to take place outside of our application, using our app as a tool. In later versions of our product, course planning could be added as a fully-fledged feature.

Personalized Reviews Homepage

Earlier, the review section used by former students simply contained a list of all the courses they had taken. We realized this view may be overwhelming because you are immediately presented with a lot of choices. (Fig. 19)

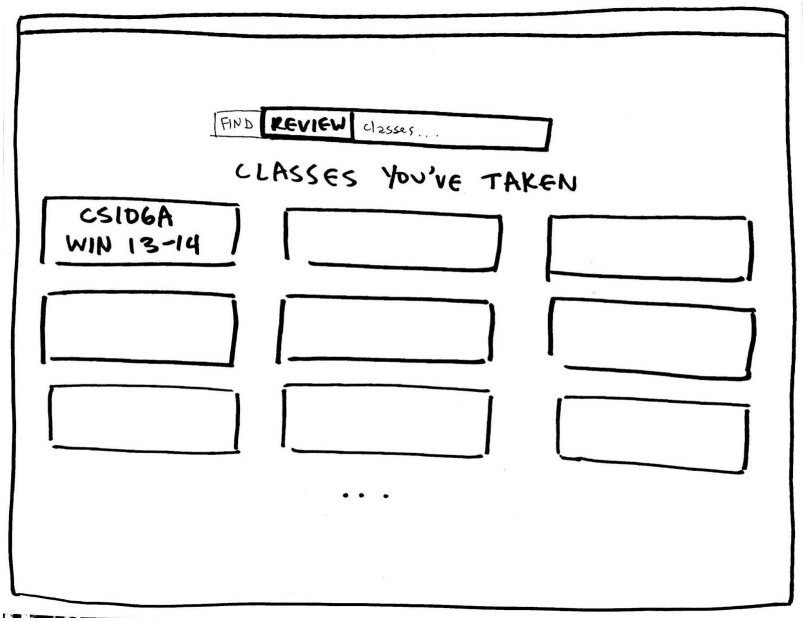


Fig. 19: Old classes taken page

In our medium-fi prototype, we decide to revise the homepage and give the user suggestions for what courses to review (for example, based on popular demand from other students). The homepage now contains several top suggestions as well as a way to view and search from all of their past courses. (Fig. 20)

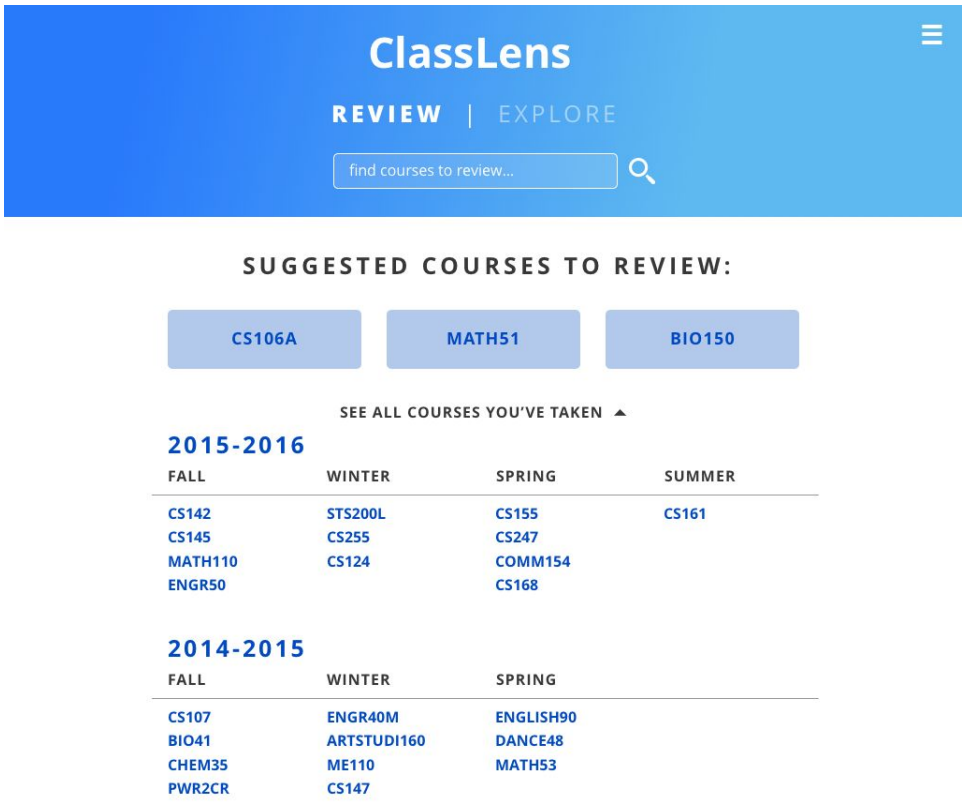


Fig. 20: New classes taken page

Intuitive skill-based feedback form

In our low-fi prototype, the feedback form had simple yes/no questions and ratings for each skill. However, this could lead to issues in filling out the form and understanding the results as the multiple choice questions had no frame of reference or context. For example, there is no standardized meaning for what it means to rate a skill as 4 out of 5 in terms of "usefulness." Thus, we removed the likert scales for individual skills. (Fig. 21)

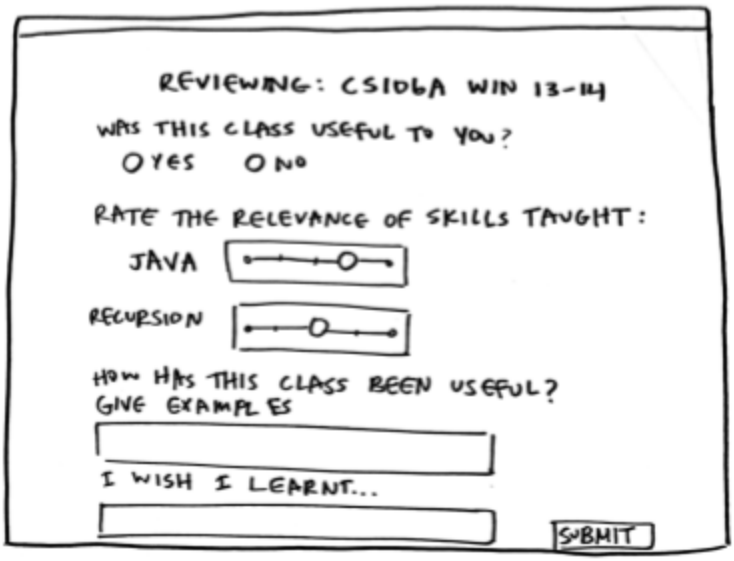


Fig. 21: Old Give Feedback page

However, our participants did feel that a Likert scale for whether a class was useful or not would be more meaningful than a yes/no type question, so we incorporated that change. We also gave the user the ability to tag their written response to particular skills. (Fig. 22)

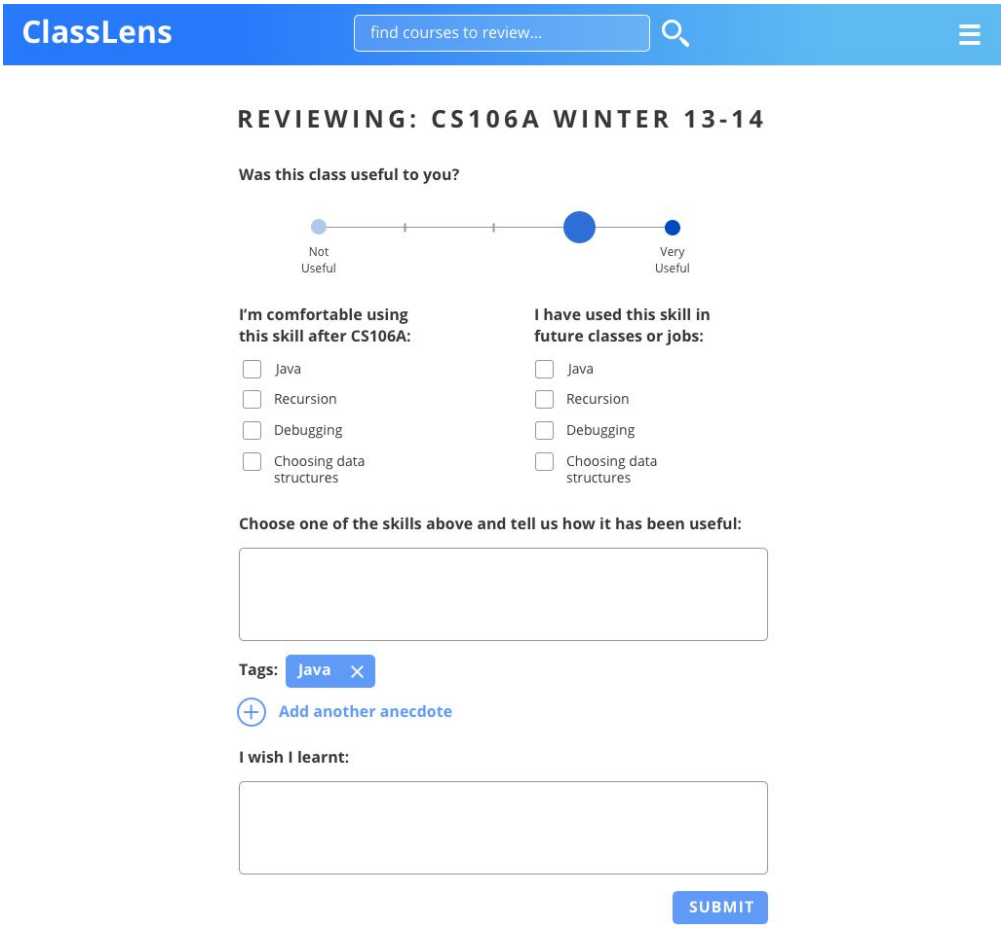


Fig. 22: New Give Feedback Page

Teacher view with better filters and pinning action

Based on our low-fi prototype testing study, we noted that there was a need to pin feedback to multiple labels, based on the nature of feedback. Specifically, instructors who were saving feedback wanted to be able to categorize it based on whether it was positive or negative feedback so that they can easily sort through and find feedback corresponding to changes that need to be made to their courses. (Fig. 23)

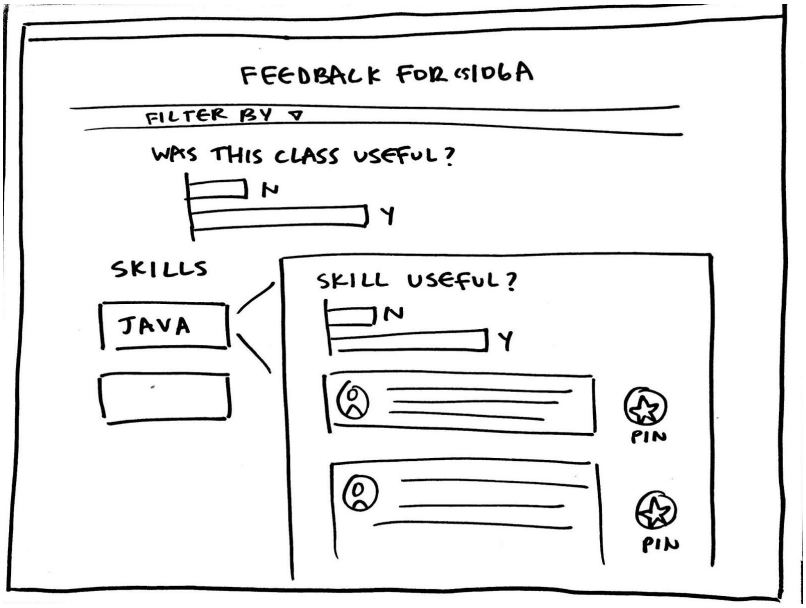


Fig. 23: Old Feedback page for Teachers

Thus, we decided to change the 1-button pin action to a 2-button pin action for tagging positive and negative feedback. We also added a mechanism to filter feedback based on multiple categories. This gave additional flexibility to the teacher who uses this platform. (Fig. 24)

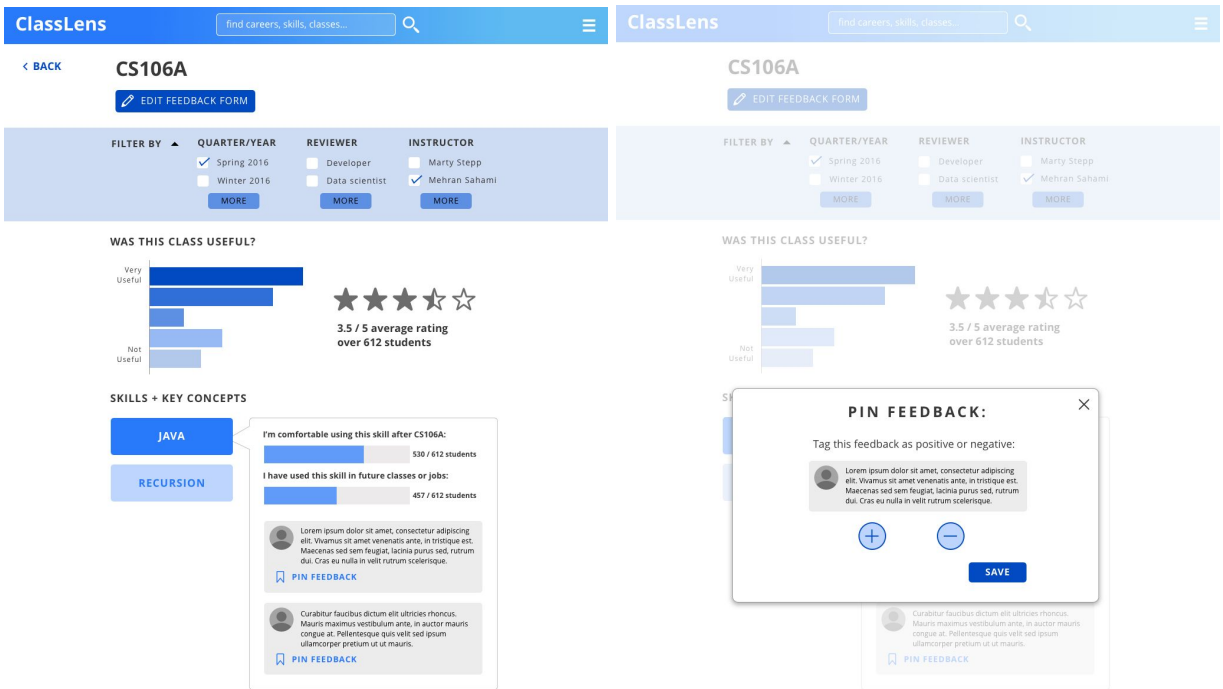


Fig. 24: New Feedback page for Teachers

Major Usability Problems Addressed

After creating a medium-fi prototype, our product was evaluated against various heuristics. This led to multiple changes in the product which have been summarized below.

Changes in the “Write a Review” page

1. There was no indication of required fields in the feedback form.
   Fix: We marked the required fields with an asterix and the text “required” in red color
2. For the user, it was unclear that word “tag” refers to skills.
   Fix: We agreed with this observation. We renamed this part as - “Which skill does your feedback apply to?”
3. The word “anecdote” is quite unclear to the average user.
   Fix: We decided to remove the “add another anecdote” feature entirely. We felt that instead of writing multiple experiences pertaining to each skill, it would be easier to write in one box and tag it with various skills.
4. When a user clicked on the submit button, it led to the review page and not the last visited page.
   Fix: We have fixed this in the high fi prototype as this was an issue with the medium-fi prototype tool we had adopted (Fig. 25,26)

Fig. 25: Before Changes - Review Page

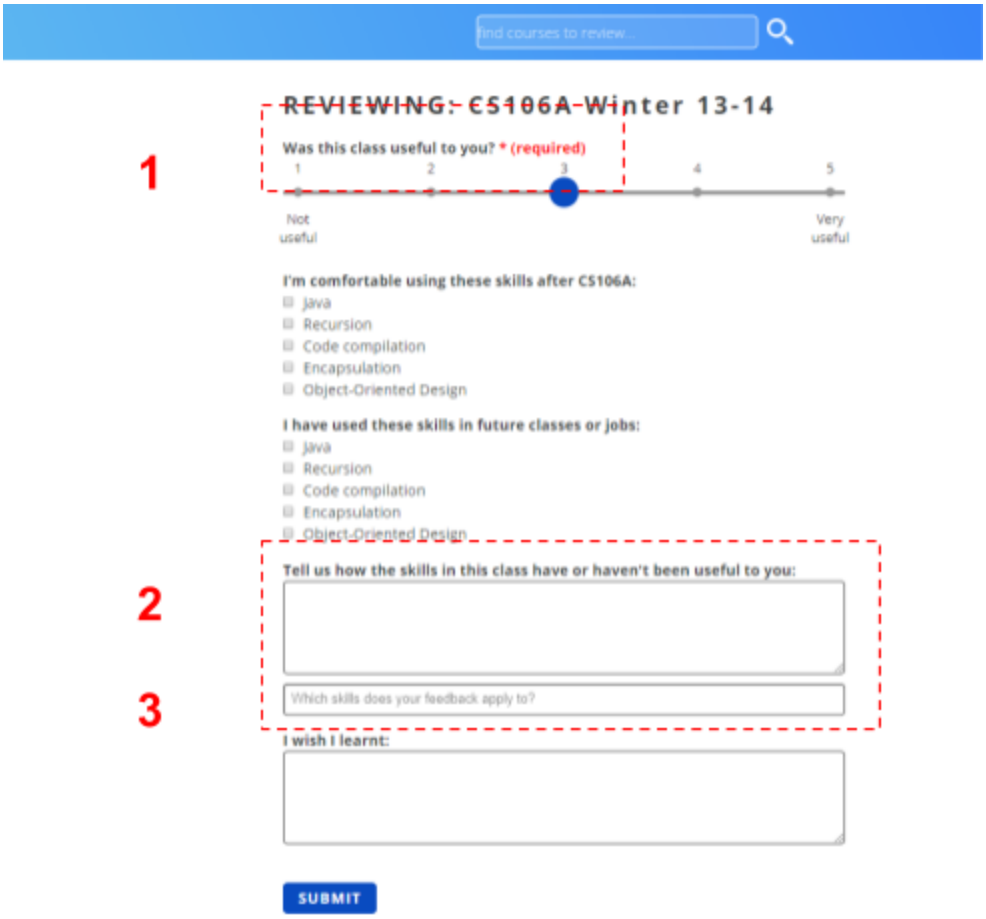
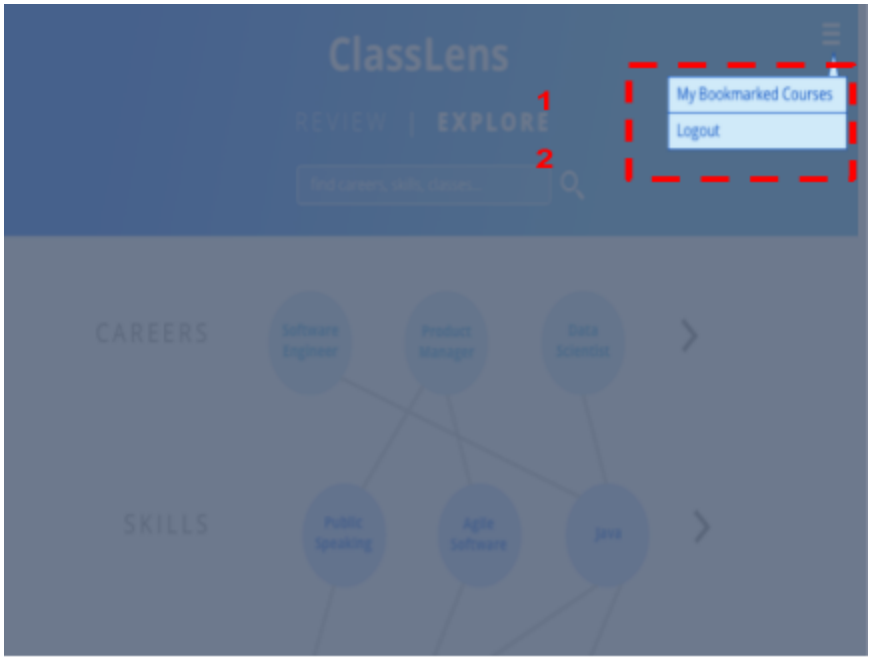


Fig. 26: After Changes - Feedback Form

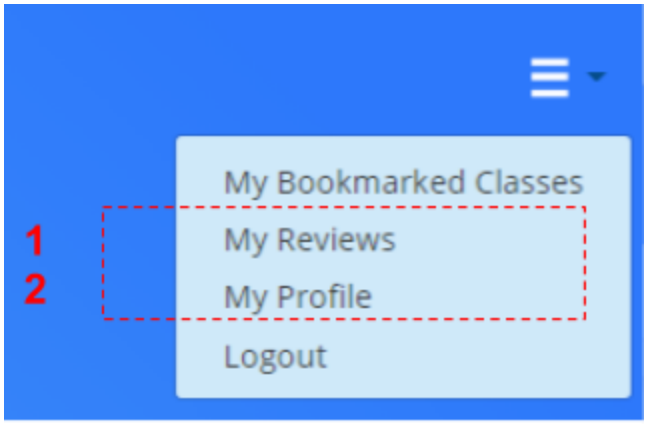
Changes in the top right menu bar

- There was no way in the medium-fi prototype to read what reviews a user may have written.
  Fix: We have added a “My reviews” page to the top right menu bar. This was the most convenient place where a user would intuitively go to in order to check his past reviews.
- There was no way for a user to see their current career
  Fix: We have added a “My Profile” page which can be visited by clicking on the link in the top right menu bar. As of now, the profile page hasn’t been made editable because this wasn’t core to our top three tasks.(Fig. 27,28)



Before

Fig. 27: Before Changes - Menu



After

Fig. 28: After Changes - Menu

Changes in the pinning feedback interaction

- The status of the pinned feedback doesn't change after pinning
  Fix: We decided to remove the icon for pin feedback as it wasn't central to the task. After a user pins the feedback, the text changes to "Remove Pin". This is much simpler for a user to follow.
- In the pin feedback popup, "+" and "-" buttons don't make sense for the user. Specifically, "+" and "-" are often used to represent add and remove so it was confusing that we used them to mean positive and negative.
  Fix: We changed the positive and negative buttons to happy, sad and neutral smileys. This will make the idea of pinning feedback according to its nature easily understandable to the user.
- In our medium-fi prototype, a feedback could be pinned without choosing a polarity (positive or negative).
  Fix: We decide to grey out the save button until a polarity has been chosen. This will make sure feedback isn't inadvertently pinned without choosing a polarity. For added flexibility, we also included a neutral polarity.
- There is no way to edit the pinned feedback list
  Fix: We provide that functionality on the class page so that prototype users know it can be done. The actual product would have standard organizational features (e.g., ability to reorder and sort through feedback, removing feedback) implemented in the pinned feedback list page, but these details were not core to our task, so we didn't implement it for the prototype. (Fig. 29,30)

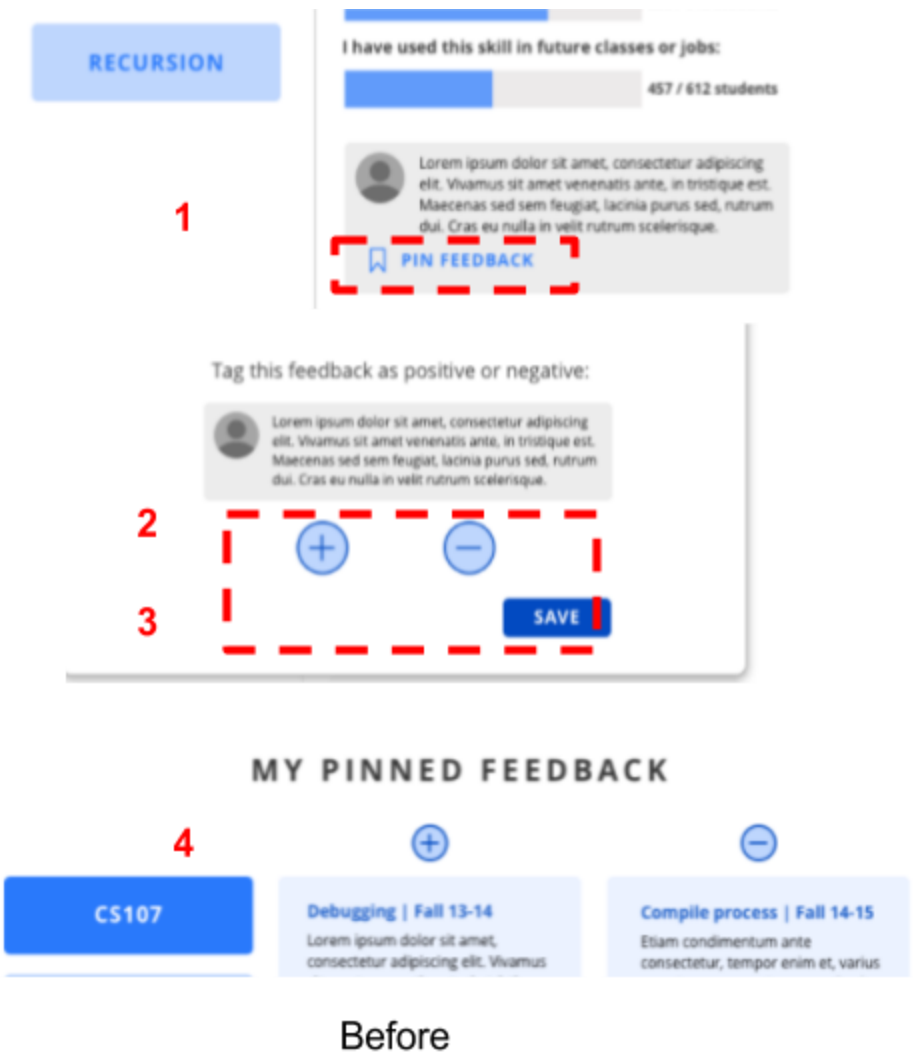


Fig. 29: Before Changes - Pinned Feedback

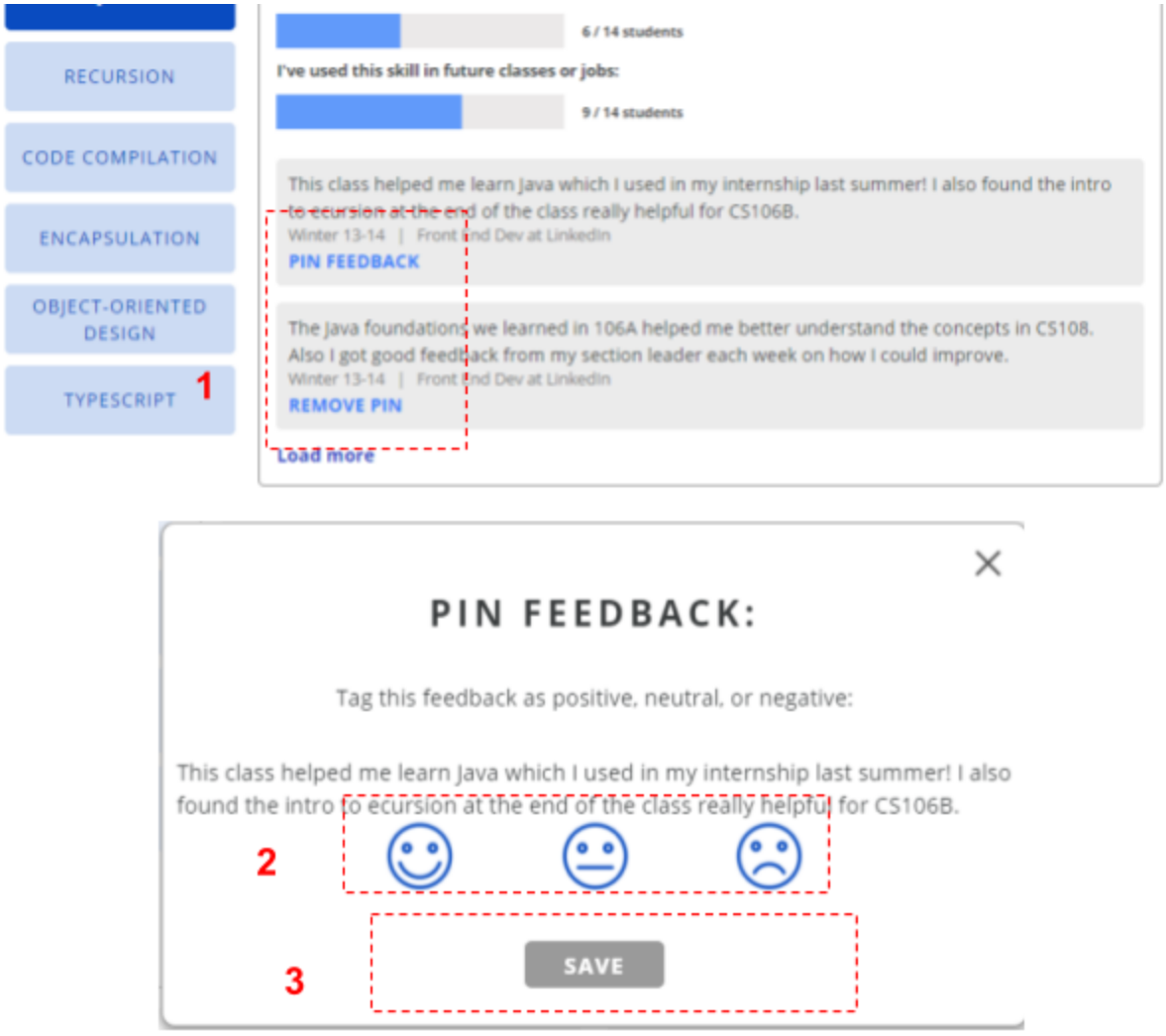


Fig. 30: After Changes - Pinned Feedback

Subsequent Changes

We also added other changes to the website to resolve other heuristic violations:

- Menu Icon in the top right corner was really small initially and could have been overlooked.
  Fix: We decide to increase the size of the icon and make it more visible.
- The color of the likert scale bubbles were confusing initially. We redesigned it to make sure that there was no confusion regarding its use.
- We have also corrected grammatical confusions (This vs. These and Review vs. Feedback) in the feedback form.

Prototype Implementation

Tools

We built the prototype using node.js for the server, d3.js for the graph visualization, Angular for the front-end framework, various front end libraries to support specific functionalities, and wrote and fetched JSON files locally to simulate a database.

How the tools helped

Using JSON files instead of an actual database framework was extremely quick to setup and use right away. Angular's two-way data binding simplified programming.

How the tools did not help

Angular had some specific quirks that had to be worked through, which consumed some time.

Not having a database framework meant that common database functionality, such as joins, indexing, and integrity constraints had to be simulated through our own code. Moreover, all the data in the JSON files had to be loaded into memory at the beginning of server startup to ensure that I/O would not become a bottleneck in our application. This is feasible for a prototype, but definitely not for a product that could have much more data. Also, although data was written to disk on every modification, user data could still be lost in a crash or data could be left in an inconsistent state: our prototype lacks concept of transactions or crash recovery.

D3.js didn't have pre-built graph layouts that exactly met our requirements, so a lot of manipulation had to be done in order to approximate the graph we wanted. In particular, the D3's cluster layout was the closest to what we wanted, but didn't support multi-parent nodes, so we let d3's cluster layout determine the node positions and then wrote additional code to determine the edges.

Wizard of Oz

The prototype is not actually synced with Axess and does not constrain students to only review classes that they have taken and teachers to editing the feedback form for classes they have taught.

Hard-coded Data

Class reviews, course information, and skills are hard-coded in the sense that we did not source them from actual users. However, they are dynamic in that user prototype input is persistent and will be displayed e.g., leaving a review for a class will result in that review being displayed on the class page

What is missing and what might you add in the future?

In the future, we would use an actual database framework, which would make information retrieval and storage more efficient and would be necessary for larger amounts of data. We'd negotiate with Stanford to sync with Axess and obtain student / teacher information so we can see what classes students have taken and what courses teachers have taught. We'd also fully flesh out personalization, provide more filters on class pages, implement account creation, refine the explore graph, and vet reviews with site moderators and NLP.

Summary

For students and teachers, knowing that the learning objectives of a course are useful and being met is very important. However, this issue is not addressed enough by existing feedback mechanisms, where course evaluation forms are filled out immediately at the end of each quarter, are a pain to read and write, and the feedback given is more about factors like teaching style/workload/educator personality than the **skills** learned in the class.

Our vision is that Classlens can be used very effectively by educational institutions around the world to provide much better feedback to their students and the teaching community. We also believe that our product has universal appeal and can be deployed in diverse situations, institutions or geographies.

Through this quarter, we have tried to make sure that our product is capable of meeting the needs of the various stakeholders involved and is highly functional apart from being beautiful to look at in the conventional sense. We would like to thank the teaching team for providing us with this amazing experience of starting from scratch to building a functional, high-fi prototype in one quarter.