

Aionno

**Capstone Studio
Harvard Extension School**

**Joshua Weiss
December 12, 2019**

Abstract: Aionno, a web app, aims to better connect teachers with meaningful and targeted strategies. The development process entailed prototyping, back-end coding, front-end coding, and testing. I document this process, and provide rationale for decisions along the way. At the culmination, I reflect on next steps and potential points for expansion in the user experience.

1. Introduction

1.1 Purpose and scope

When starting out in the classroom, I was prepared content-wise, but wholly unprepared strategy-wise. That is to say, I would be able to plan a lesson around a topic – for example, metaphors – but if anything went awry during the course of the lesson with regard to student behavior, I possessed no techniques with which to rectify the situation. While lists of interventions and strategies were available online, I did not fully understand how to select a strategy, nor how to apply it according to best practices.

Aionno aims to fix this. I will detail in this report how I designed this app to serve teachers, iterated and prototyped on an initial design, and ultimately coded a full web app. I will also lay out potential areas for extension.

1.2 Project Goal

The goal of the *Aionno* web app is to act as a “match-maker” service between student needs and teacher action. The interface is a straightforward, mobile-friendly search-and-filter-sequence. The results of a search point to extensive resources describing the strategy that can be used in the classroom, as well as best practices and guides to application. These strategies draw from learning science, cognitive psychology, sociology, and other areas related to behavior and achievement.

1.2 Learning Goals

In a general sense, my persona learning goals over the duration of this project were to develop my project management and technical skills. Project management-wise, I aimed to improve my capacity to roadmap a multi-stage project that includes various stakeholders, deliverables, and potential points of iteration. Technically, I improved my ability to design and execute an architecture that works well across devices and use cases.

1.3 Target Audience

The target audience is beginning-to-mid career teachers who work with students ages 11-16. The potential breadth of this audience is enormous; in the US alone, 75 million students are affected by the quality of interventions performed by their instructors.

Over the course of my capstone, I proposed to address the needs of 500 students through 5 teacher-partners. The teacher-partners would be based in California and Colombia, and are typically beginning teachers aged 22-32. I hope to expand this scope as my project develops beyond the capstone timeline.

1.4 User personas

By envisioning my target users, as well as some edge case users, I could better determine what features are worthwhile. I would also have a better focal point for future design decisions, as these personas would be revisited from time to time.

1.4.1 Persona 1 (Diagram 1.4.1): 1st year 8th grade English teacher in San Mateo, California, 23 years old, white female. Bachelor's degree, education major. Ambitious, compassionate, anxious. Has some strategies from undergraduate training but gets nervous about designing group work.

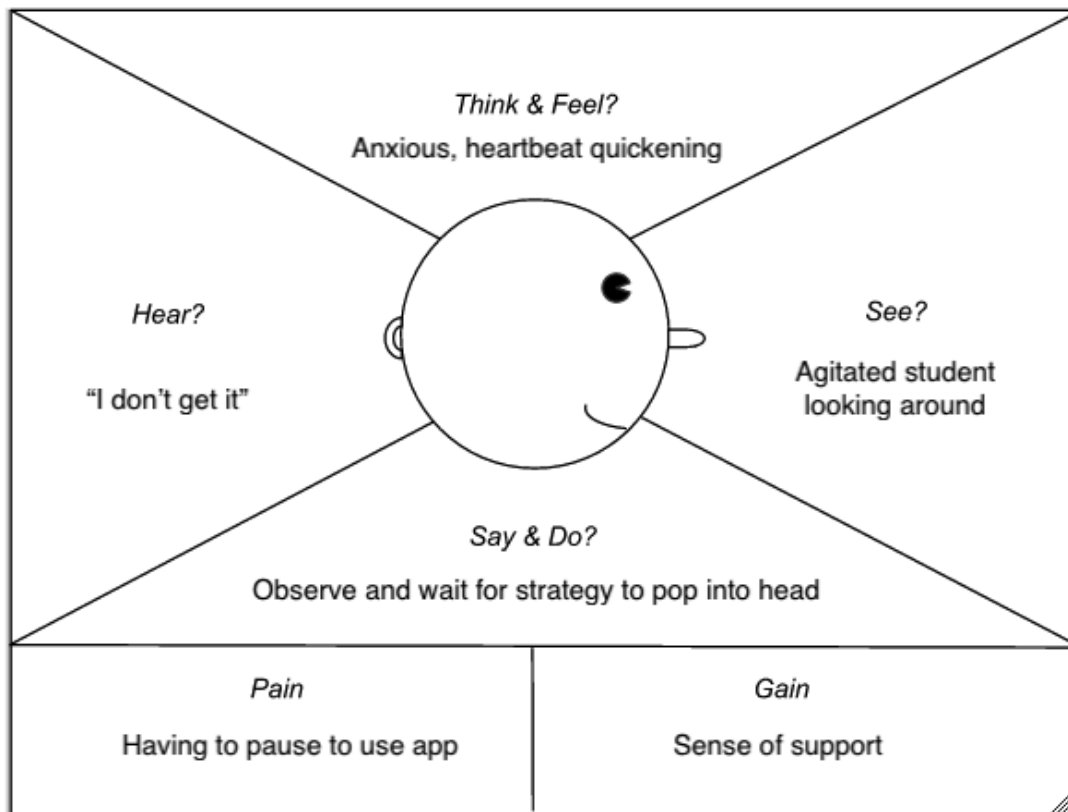


Diagram 1.4.1

1.4.2 Persona 2 (Diagram 1.4.2): 3rd year 9th grade history teacher in Bogota, Colombia. 26 years old, Latino male. Bachelor's degree, non-education major. Busy, analytical, social. Uses same strategies most days and wants to expand repertoire.

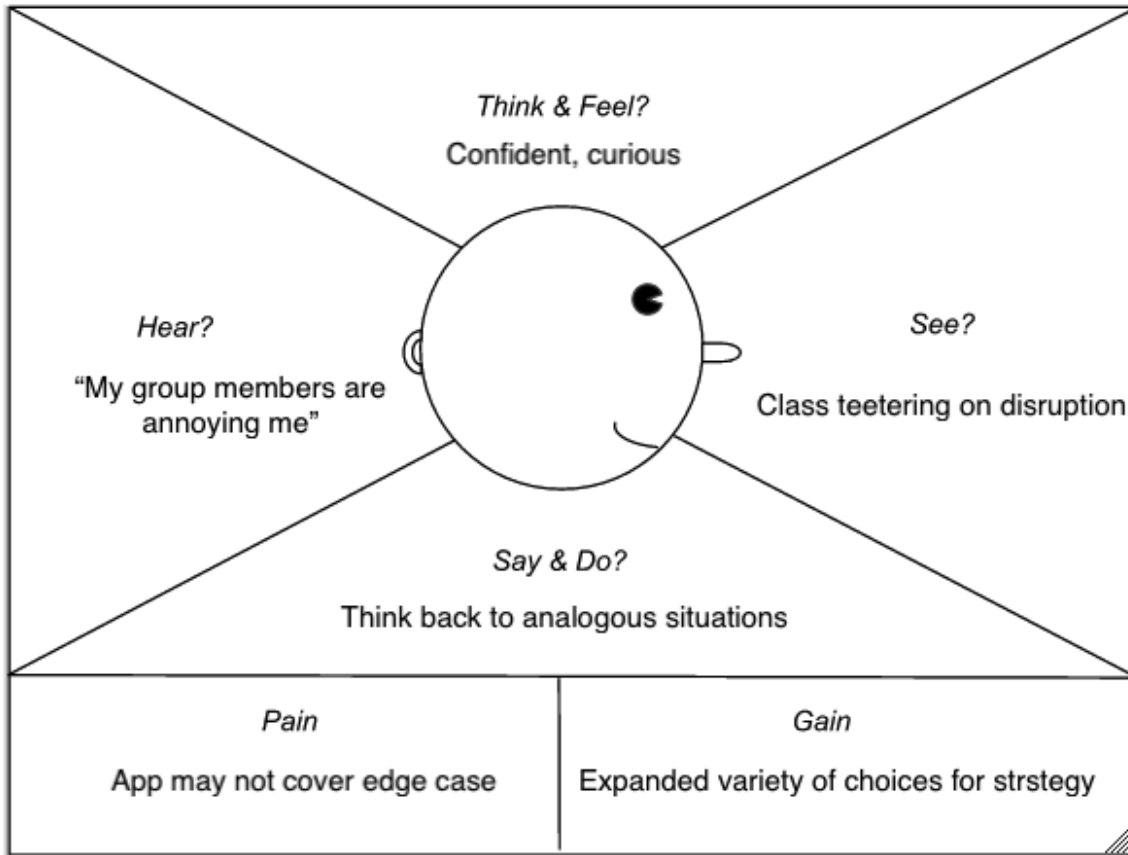


Diagram 1.4.2

1.4.3 Persona 3 (Diagram 1.4.3): 5rd year 7th grade ESL teacher in San Francisco, California. 32 years old, Asian female. Bachelor's degree, Master's degree in Education. Considerate, tired, driven. Just changed subject area from science to ESL and wants to extend old strategies into new content area.

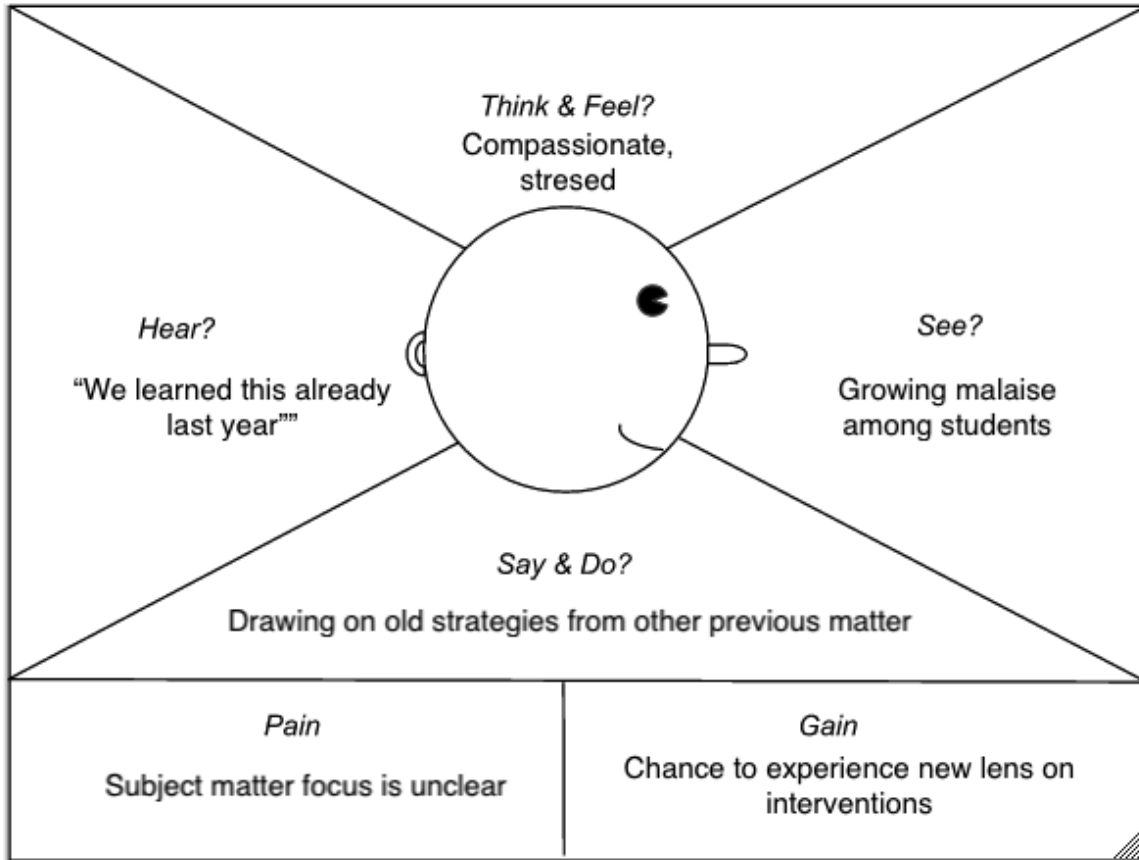


Diagram 1.4.3

1.5 Elevator Pitch

Teachers have a hard time understanding what students need. *Aionno* helps instructors deconstruct what is going on in their classroom, and then suggests strategies to address the situation. With *Aionno*, inexperienced teachers can target problems and apply strategies like a master teacher!

1.6 Metrics

Before building the technology, I mapped out metrics for user experience as well as technical benchmarks. I aimed for "Exceeds benchmark" in order to achieve maximal quality. The various qualitative and quantitative descriptions are laid out in Diagram 1.6.

Category	Approaches benchmark	Achieves benchmark	Exceeds benchmark
Interface design	Color and layout are accessible for most common devices; works	Color and layout are accessible for most devices; works on mobile, but not	Color and layout are universally accessible for all devices; optimized for

	on mobile, but not optimized (menu bar, type scaling, etc.); basic typography	optimized (menu bar, type scaling, etc.); functional typography	mobile; clean and inviting typography
Quality of code	Code is sometimes modular, not well commented, and infrequently demonstrates consideration for version control and hand-off	Code is often modular, well commented, and demonstrates consideration for version control and hand-off	Code is modular, descriptively commented, and demonstrates high consideration for version control and hand-off
Translation	User can find useful translation within 90 seconds	User can find useful translation within 60 seconds	User can find useful translation within 30 seconds
Strategy discovery	User can find useful strategy within 180 seconds	User can find useful strategy within 90 seconds	User can find useful strategy within 30 seconds
Strategy application	User can apply strategy with confidence within 60 minutes	User can apply strategy with confidence within 30 minutes	User can apply strategy with confidence within 15 minutes
Scale and access	Can be used by up to 100 people, 5 at once; can be used on any device; design does not accommodate for eventual porting to other languages (Spanish, Portuguese, etc.)	Can be used by up to 500 people, 20 at once; can be used on almost any device; design allows for eventual porting to other languages (Spanish, Portuguese, etc.), but with significant effort	Can be used by up to 1000 people, 50 at once; can be used on any device; design allows for eventual porting to other languages (Spanish, Portuguese, etc.)

Diagram 1.6

1.7 Life of the project beyond capstone

My project will continue to be iterated on as I develop more functionalities for broader and broader audiences. For example, it is scoped now for North American secondary school teachers, but it could well expand to assist Latin American secondary teachers or North American primary school teachers.

Professionally, there is no endgame. I don't believe it will help me get another job. I am primarily doing this out of my own desire to help struggling teachers, as I was once one myself.

My project will be utilized after the capstone and throughout this school year -- and hopefully into coming school years as it is iterated and integrated further into the

teaching workflow. With more trips to Latin America and Asia, I hope to disseminate the app further and with broader audiences, and then to iterate as users communicate what they'd like to see next.

2. Competitor review

To understand the current state of the art with strategy engines and similar search tools, I researched adjacent use cases on the market. Diagram 2.1 entails a break-down of that competitor review.

Name	Description	Dimension being compared	Advantages	Disadvantages
http://www.beesburg.com/edtools/glossary.html	List of instructional strategies posted on a website	Content	Exhaustive list of cognitive, emotional, social, and behavioral strategies and interventions	Only way to search is by alphabetical shortcut. Unless the user knows the name of the strategy, they will be unable to find what they want in a short amount of time.
http://www.kayak.com	Interactive website to select flights	Flow	Two-part selection process – basic parameters and drill-down filters	Busy interface that takes time to familiarize with
http://fcit.usf.edu/mathvids/strategies/completelist.html	List of instructional strategies posted on a website, accompanied by videos and implementation guidance	Content/experience	Video materials chunked into steps, lists of materials and complementary activities, extensive descriptions	Media is 20+ years old, poor site UX, not concise, language can be inaccessible to beginning teachers
https://www.plot-generator.org.uk/	Interface to generate story plots	Experience/flow	Narrowing down by genre then expansion by idea produces precise but diverse results	Exposing many possible ideas at once takes up a lot of screen real estate, which is not ideal for mobile
https://www.facinghistory.org/resources-library/teaching-strategies	Bank of teaching strategies chunked by type	Content/flow	Strategies can be searched by title or by type, making discovery much more efficient	Categories are very broad, with some types, such as “Image Analysis”, encapsulating 50+ strategies. Ordering is just alphabetical, instead of by relevance

Diagram 2.1

3. Technology requirements

Below, I identify the requisite technologies for my project. I express the rationale for each choice, the purpose, and some alternatives. I also describe in which Harvard Extension course I developed my proficiency with regard to the tool.

3.1 Node.js

- Description: back-end JavaScript framework
- Related courses or professional experience: E-31 Intro to Node.js
- Alternative technologies, technologies used by competitors: LAMP/MAMP, Ruby on Rails
- Reason for selecting this technology vs. its alternatives: flexibility with progressive web apps, personal familiarity, broad selection of libraries for web apps
- Use: Node.js will serve as the back-end, populating the page with new results based upon a user's search parameters

3.2 Client-side JavaScript

- Description: front-end JavaScript
- Related courses or professional experience: E-15 Screen-based and Physical Computing
- Alternative technologies, technologies used by competitors: Flash
- Reason for selecting this technology vs. its alternatives: broad adoption across browsers, personal familiarity, extensive support and documentation
- Use: UI and transitions in the browser

3.3 Sass

- Description: CSS compiler and productivity framework
- Related courses or professional experience: E-27 Modern and Mobile Front-end II
- Alternative technologies, technologies used by competitors: Less, Uikit
- Reason for selecting this technology vs. its alternatives: flexibility with new browser technologies, personal familiarity, broad selection of out-of-the-box functionalities
- Use: dynamically scaling across devices, visual accessibility, clean and modular code

3.4 Additional technologies

I will also be conducting significant user research in South America and North America to determine the query model through which users will access and discover new strategies. This will utilize adult learning theory, which was addressed in-depth in E-111, Adult Online Learning: Theory. I will also be designing the interface for intuitive and natural patterns, as well as common visual grammar in digital interfaces, concepts that were covered extensively in E-170, Information Systems Design.

4. Prototyping and planning

At the onset, I envisioned my project to be a web app that analyzes student behavior and provides instructors a “translation” of said behavior as well as strategies that apply to that interpretation. Upon entering the site, the user would see a title, a description, and instructions on how to use the dropdown menus (Diagram 4.1).

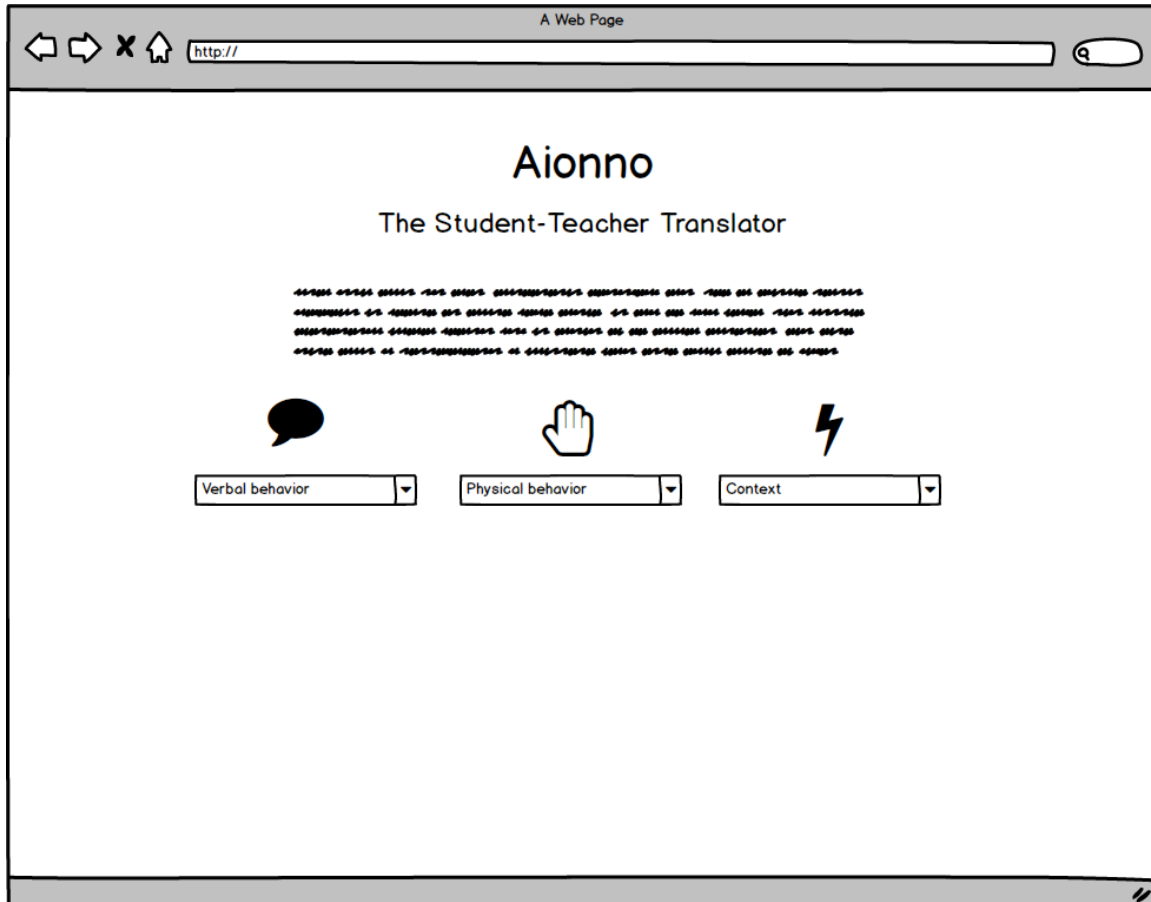


Diagram 4.1

Then, the user would select the dropdown options for “verbal behavior”, “physical behavior”, and “context” that best match their experience. These 3 attributes would combine to signify the instructor subjective experience in the classroom environment and their impression of the obstacle. (Diagram 4.2)

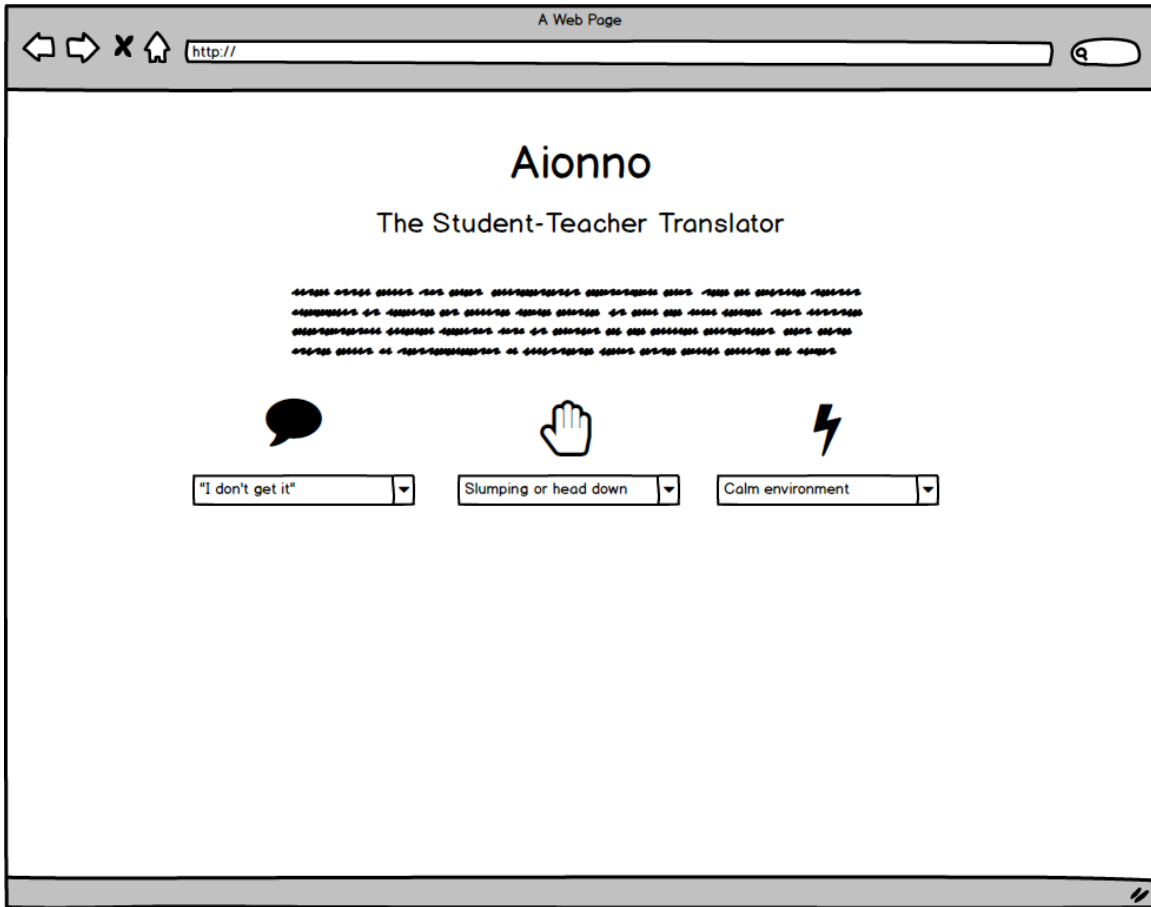


Diagram 4.2

After calibrating the dropdowns, the user would see several “translations” appear. The translations each list the most applicable strategies that could resolve the doubt or problem expressed by the student (Diagram 4.3). Each strategy card would consist of a summary and a link to an external resource page describing the strategy in more detail.

Aionno

The Student-Teacher Translator

Placeholder text for the main content area.



"I don't get it"

Slumping or head down

Calm environment

"This is beyond my grasp"

Strategy: chunking content

Placeholder text

Learn more here

Strategy: accessing background

Placeholder text

Learn more here

"I feel out of place"

Strategy: value statement

Placeholder text

Learn more here

Strategy: stereotype threat

Placeholder text

Learn more here

"I feel overwhelmed by my tasks"

Strategy: constructive worry

Placeholder text

Learn more here

Strategy: bullet journal

Placeholder text

Learn more here

Show me more



Diagram 4.3

The user could then peruse the strategies available to them, and either select “Show me more” to extend the search results, “Learn more here” to delve into strategy and implementation specifics, or select a new set of attributes to perform a novel search.

5. Instructional design brief

When planning the user experience, I mapped out an instructional design brief. The goal of the brief is to ascertain what the user will see and do, and to drill down on the psychological and behavioral effects. In Diagram 5.1, I discuss general concepts and overarching concepts. In Diagram 5.2, I discuss how understanding could manifest inside the app experience. In Diagrams 5.3-5.5, I describe the core app experience. In Diagram 5.6, I state pedagogical underpinnings.

<p>High-Level Overview. Briefly describe your learning experience, including the type of learning experience, intended audience, duration, etc. Use the Instructional Design Mad Lib to assist you.</p>	
<p>I'm designing a 2-minute just-in-time app for secondary teachers to discover classroom strategies. This is important because educators struggle to identify potential interventions that precisely address failing classroom circumstances. As a result of this experience, users will be better informed of applicable strategies and more confident in their capacity to administer their classrooms.</p>	
<p>Content Topic: Identify the content topic that will be explored. Though broad, this is often where instructional designers begin.</p>	<p>Big Idea: Keep in mind the misunderstanding or gap, and identify the big idea, a concept about this topic that is worth knowing and can be applied to other content/contexts. It provides a unifying and thoughtful way to focus the design of the project. The big idea should be expressed in a few words.</p>
<p>Classroom strategies</p>	<p>Why one strategy is more applicable than another</p>
<p>Misunderstanding or Gap: Think about the prior experiences, knowledge, and mindset of the learners. What might they misunderstand about this topic or what is a gap in their thinking/experience that prevents them from understanding this topic and/or big idea? This may be informed by any combination of research, observation, or interviewing.</p>	
<p>Users might think that there is one strategy for each circumstance. This is likely false; there are multiple interpretations and tools for each scenario. A full "menu" of possibilities is useful, as well as a sorted ranking for most likely to be useful.</p>	

Diagram 5.1

<p>6 Facets of Understanding: The 6 facets are a tool to help unpack what deep learning looks like. Not all facets are applicable for all projects; however, consider each one. What could a possible desired understanding or learning outcome be through the lens of each of the six facets? Share your notes here.</p>	
<p>Explanation: Really gets at explaining something in the learner's own words</p>	<p>Perspective: Gets at what it means to see the big picture or consider various points of view</p>
<p>Having to explain the circumstances in a classroom will force them to dissect, albeit subjectively, exactly what is and isn't</p>	

occurring, particularly at the point of intervention.	
<p>Interpretation: How to make sense of something</p> <p>Multiple “translations” of student expression provokes deeper interpretations into the cause and outcomes surrounding the scenario.</p>	<p>Empathy: Asks the learner to “walk in another’s shoes”</p> <p>Within the cycle of the app, users will have the opportunity to empathize with those in their classroom by closely identifying actions and responses and will thereby gain perspective into what all classroom agents are doing to contribute to the dysfunctional scenario that has prompted the use of a strategy.</p>
<p>Application: Matches knowledge to context</p> <p>As users interact with the interface, they will be gradually connecting classroom interactions and observations with potential solutions.</p>	<p>Self-Knowledge: Gets learners to think about their own thinking</p>

Diagram 5.2

<p>Why/Enduring Understanding(s): Frame your big idea as 1–2 understanding statements. The understanding statement is expressed as a full-sentence statement and represents an insight, inference, or conclusion about the big idea that learners should gain. Rather than the facts you want them to learn, the understanding statement looks to the meaning of the facts.</p>
<p>Learners will understand that each classroom situation has a potential strategy that can be applied to improve conveyance, exchange, collaboration, etc.</p>

Diagram 5.3

<p>Evidence of Understanding: How will you know that your learners have obtained the desired understanding? This is often thought of as assessment in formal learning environments. For self-paced and informal learning experiences, this may be more difficult to identify but try.</p>
<p>Users will apply the strategy and, in later iterations, rate how well the strategy mapped onto the scenario. (I do not plan on enabling this feature for the current iteration.)</p>

Diagram 5.4

<p>Learning Flow: What is the general flow of the learning experience? You may provide a bulleted high-level list, create a Journey Map (a timeline that graphically maps the experience), or other graphic organizer.</p>
<ol style="list-style-type: none"> 1. Enter app via web browser 2. Read description and purpose of site 3. Choose descriptions to explain classroom situation and student expression 4. View results of search manifested as list of possible student “translations” 5. Focus on which translation best fits their needs/expectations 6. Click on strategy to learn more about implementation 7. Implement strategy 8. Perform new search (as needed)

Diagram 5.5

<p>Learning Theory(ies): What learning theories—the way in which how people learn—will your learning experience draw upon? List them and make sure you research them further to see how they inform the approach you’ll take.</p>	<p>Pedagogies: What pedagogies—methods of how people teach—will your learning experience draw upon? List them and share why.</p>
<p>The app is largely based on behaviorist learning theory (Skinner) in that each “behavior” exhibited can be more or less classified and matched with another defined strategy that maps to it with regular success.</p>	<p>Connectivism is fundamental to the app. The experience collects and organizes the sum resources of the internet and provides a portal for initial discovery and recommendation. Users are encouraged to take this discovery point and extrapolate to new strategies and resources, which will be readily linked in the app itself.</p>
<p>Inspiration: Identify at least three other learning experiences/products that inspire your project, e.g, workshop, training, e-learning course, game, curriculum, museum exhibit, YouTube channel, etc. Be specific.</p>	
<p>First, I’m inspired by the discovery flow on kayak.com due to its rich feature set, intuitive dashboard, and clean layout. The rapid load times and graphical representations of recommendations is excellent UX-wise</p> <p>Second, I take inspiration from Khan Academy in their instructional flow and adaptivity to learner needs. It is clear they have done a lot of user testing and interviews to understand how online learning best operates for their user group.</p> <p>Third, I look to the design thinking workshop model, which emphasizes empathy, prototyping, and iteration with the users in mind (and present in the development process).</p>	

Diagram 5.6

6. Work plan and milestones

In order to manage the build process, and afford time for testing and QA, I created a full roadmap (Diagram 6.1) describing dates, tasks, and durations. For a high-quality version, please see the image [here](#).

TASK NAME	START DATE	END DATE	DURATION (DAYS)	July				August				September				October				November				December					
				1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
Complete User Research																													
Interview instructors	7/23	8/30	38																										
Ascertain the breadth of strategies	7/23	8/30	38																										
Determine input schema	8/17	9/10	24																										
Sketch initial UX with user input	8/20	9/10	21																										
Create mock-up																													
Create basic storyboard	9/10	9/20	10																										
Create clickable prototype	9/10	9/20	10																										
Code app back-end																													
Refamiliarize with frameworks (node.js, PWA)	9/20	10/1	8																										
Set up server-side technologies	9/25	10/20	18																										
Code app front-end																													
Refamiliarize with frameworks (Sass)	10/20	10/25	5																										
Set up client-side technologies	10/25	11/20	26																										
QA/User testing																													
Alpha testing	11/20	11/25	5																										
Beta testing	11/25	12/1	6																										

Diagram 6.1

(<https://drive.google.com/file/d/1Jt1q0RqrnnRyaDkymM9p3chatGTNIFJR/view?usp=sharing>)

Milestone #1: Complete user research

Date: September 10, 2019

Description: Ascertain the breadth of strategies (cognitive, social, emotional, etc.) that could be useful to instructors. Determine if input schema on app maps to how instructors conceptualize classroom behavior and intervention.

Milestone #2: Complete mid-fidelity mock-up

Date: September 20, 2019

Description: Create mid-fidelity, clickable mock-up (with software like inVision, Marvel, etc.) that concretizes structure of app

Milestone #3: Code app back-end

Date: October 20, 2019

Description: Develop server-side technologies to pass data and render pages

Milestone #4: Code app front-end

Date: November 20, 2019

Description: Develop client-side technologies to ensure app is responsive to mobile and is visually appealing

Milestone #5: QA/User testing

Date: December 1, 2019

Description: Test with alpha and beta groups to check for bugs. Perform user interviews.

7. Building the app

Starting in September, I began to construct the app. I first developed a rich, clickable prototype, then moved on to coding the back-end and front-end of the app. The narrative below describes the sequence of events.

7.1 September 2019

For the first few weeks, I focused on interface design, user feedback, and researching back-end architecture. Regarding interface design, I developed a clickable prototype using Marvel prototyping software. The purpose was to determine the minimal UI necessary to accomplish my goal of connecting instructors to classroom strategies. I first sketched out a few ideas, then reduced the designs to the bare bones: header, drop-downs, and cards. I did this in order to understand the core experience of my app, and to be able to hand this off to potential users for feedback. The design can be seen in Diagram 7.1.

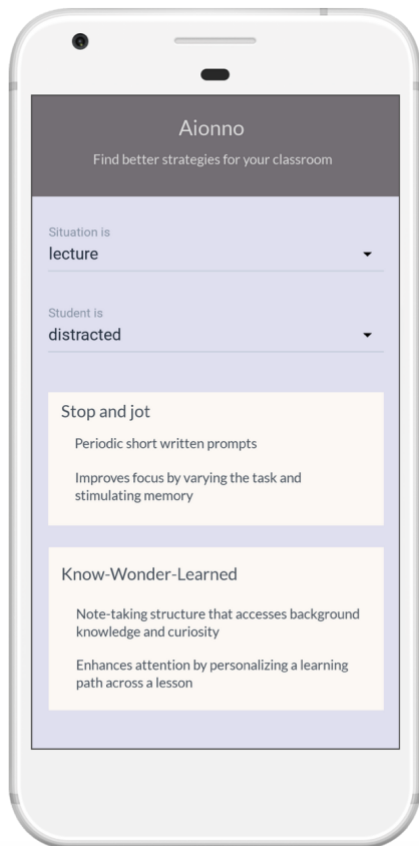


Diagram 7.1

I also spent time in the first month of development researching the most lightweight methods of app development. My goal was to consume as little data and bandwidth as possible, and eventually to deploy as an offline-optional app.

7.2 October 2019

In early October, my focus was to get my coding environment and project files in a workable condition. I gave myself a week to refamiliarize with Node.js and to spin up an environment. However, I found more and more that generating a whole new environment would not be a good use of time, particularly as the purpose of the project is not to re-learn web development, but rather to build on ideas and concepts I had already utilized. As a result, I recycled the basic server and folder structure of a previous project that used Node.js; this saved me significant time in generating error handlers and proofing the server for unintended consequences.

With that accomplished, I turned my attention to adding data and interactivity to my dropdown menus. While I had used Bootstrap drop downs before, I found them to be too rigid for what I was trying to do – specifically, grabbing a value. I re-familiarized with JQuery and employed the “on change” function to perform an event listener and then grab the value when the dropdown had changed. I then performed a test by having an alert box pop up whenever the value was grabbed, as one can see in Diagram 672.

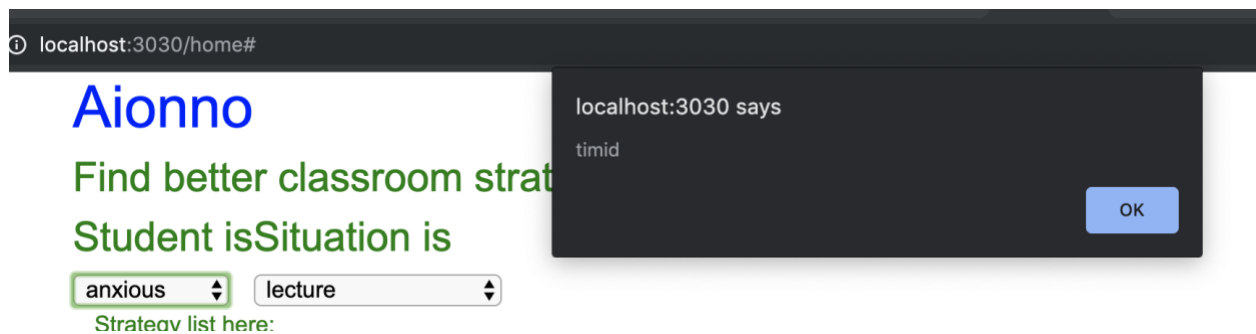


Diagram 7.2

With this accomplished, I moved on to populating the list of strategies based upon dropdown inputs. This is where I hit some real difficulties. First, I had to determine the best way to populate a series of tiles in the HTML. I thought about creating a series of tiles in the index.js file, but I felt that this was not a good use of the full power of the Pug templating engine. I did some research and discovered that pug has a for/each loop that can run over an array of objects, as well as functions (or “mixins”) that can accept parameters from an object.

So, I decided to make my data flow as follows: the drop-down menu would generate a new array of objects, which the router would serve to the Pug template; the Pug template would then read the array and generate a tile for each item. If the dropdown were triggered again with an “on change” event, then the array would be overwritten with a new list of objects, and those would pass to the HTML all over again.

The problem was that the data did not pass. I tried debugging by making an array in the router file itself (omitting the index.js file), but even then I could not get the array to pass to the HTML file. I spent a few days on the problem, stripping down the problem until I was left with the conclusion that nothing was passing to my HTML file at all via the router, which is very frustrating since I had done so successfully in a previous project. I asked coder friends to look at it, but none had experience with Node.js, so their ability to help was limited. Ultimately, I turned the app into a single-page app to avoid the routing issue and resolved to loop back to the routing issue once the base functionality on the main page was established.

I also worked on determining the proper data structure for the app. I initially created an array of objects that could live inside the app. I chose this structure because I had worked with similar JSON before, and I felt comfortable filtering it for information. I settled on this structure for now but knew that I might have to adjust my expectations at some point.

7.3 November 2019

My initial focus was developing the data structure behind the filtering mechanism. My major obstacle up to this point had really been determining how to encode the different data in a way that it can be separated out and searched for afterwards. I played around with different ways of organizing the data, and started out initially just trying to turn each strategy into a separate object, and then throw all of the separate objects into one enormous array. The problem with this, unfortunately, is that there were multiple attributes that could be attached to one single property – for example grade level could be 6, 7, 8, or 9. These all lived under one single category, which was called “grade level”.

So, I decided to rethink my data structure. Initially, I looked at different ways of developing separate tables for each possible combination of search factors. However, I consulted with some peers, and they suggested nesting an array within each of my properties within each of my objects. As a result, this would look like turning the previous example of grade 6, 7, 8, or 9 into one single array that lives within the property of “grade level”.

I also cleaned up the interface to be more user-friendly using UX principles like alignment and grouping, as evidenced in Diagram 7.3.

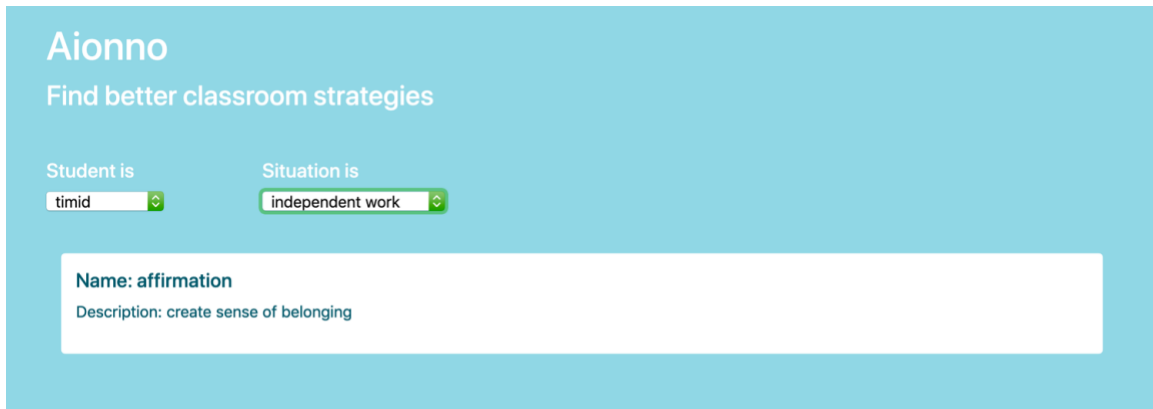


Diagram 7.3

Then, I focused on making the interface close enough to the final product that it would be testable. I had attempted previously to show my prototype to instructors, but the interface was so unwieldy and poorly formatted that the user couldn't even properly navigate the basic functionalities. To resolve this, I first set out to make the basic back-end work, specifically the basic filtering functions.

As such, I worked more on making the interface not only clean visually, but also more condensed in case users wanted a compact mobile display. The most obvious example of this is the accordion menu that I created on the right-hand side of the screen. Many filter items can now be accessed as separate sub containers inside of this accordion (see Diagram 7.4).

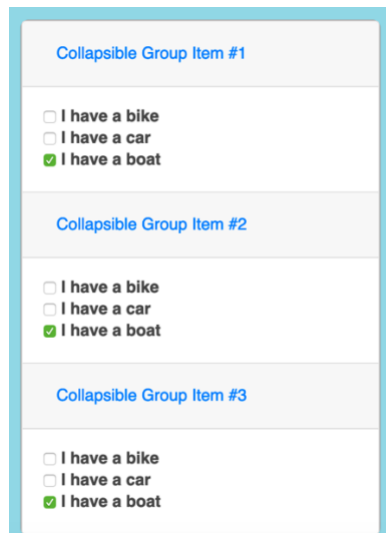


Diagram 7.4

This accordion is expanded on the initial page access so that even a first-time user is aware that there are many filters available to them.

The other part of the interface that I tried to clean up for easier navigation is the toolbar at the top and the spacing of the two drop-down menus. Regarding the navigation bar, I made sure that it was mobile-friendly, specifically using the familiar hamburger icon that would be comprehensible to most users, even if they were not very familiar with modern web design (see Diagram 7.5).

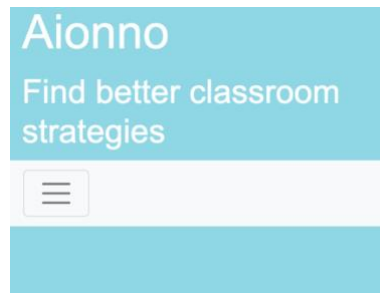


Diagram 7.5

The two drop-downs follow the same rule: even if the user is not used to modern web design, the user is still probably familiar with basic form elements like a drop-down. I made the two drop-downs to appear to come off the page using shadow properties so that there was a visual affordance that gave the user the idea that it was a clickable, button-type feature, as we see in Diagram 7.6.

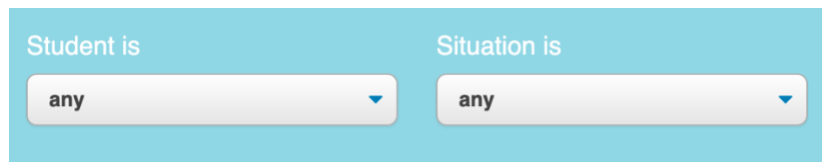


Diagram 7.6

The final UI element that I redesigned were the tiles for each strategy. Here I did two things. First, I gave a drop shadow to each tile so that each looks clickable, even on mobile. Second, I made the entire tile a single clickable field, instead of just having the text be clickable. Once the tile is clicked, the user is directed to a separate tab that gives in-depth resources and descriptions about that strategy.

My first goal was to figure out the side filter mechanism once and for all. Previously, I had hooked up the UI just within HTML and CSS. Now, I wanted to make the list of strategies populated or unpopulated based upon what side menu checkboxes have been ticked. The biggest obstacle for me to accomplish this was to determine the algorithm that would accommodate for two potential change events; either the side filter menu triggering the change event, or the drop-down top menu triggering the change event. This was tricky because of the way the information was grabbed, and the way the change event reads the target are both distinct in each of the inputs – that is to say, the

checkbox and the drop-down inputs. I got around this by externalizing a function that uses the show hide function in jQuery, but only after the list has been populated based upon the global variables for each of the dropdown menu items as well as any of the new checkbox items (see Diagram 7.7).

The diagram shows a user interface with a light blue background. On the left, there are two dropdown menus. The first is labeled "Student is" and has "any" selected. The second is labeled "Situation is" and has "lecture" selected. To the right of these is a sidebar with a white background. The sidebar has a header "Age of student" and three radio button options: "10-12", "13-15", and "16-18". Below the radio buttons is a section labeled "Content area".

Diagram 7.7

In addition, I also worked on putting the code on a server. Previously, I had been just hosting locally. I used my experience from the Node.js class at Harvard Extension, and utilized Digital Ocean, a service that I had hosted a note app on before. I followed some additional online documentation and was successful in posting this online, as seen in Diagram 7.8.

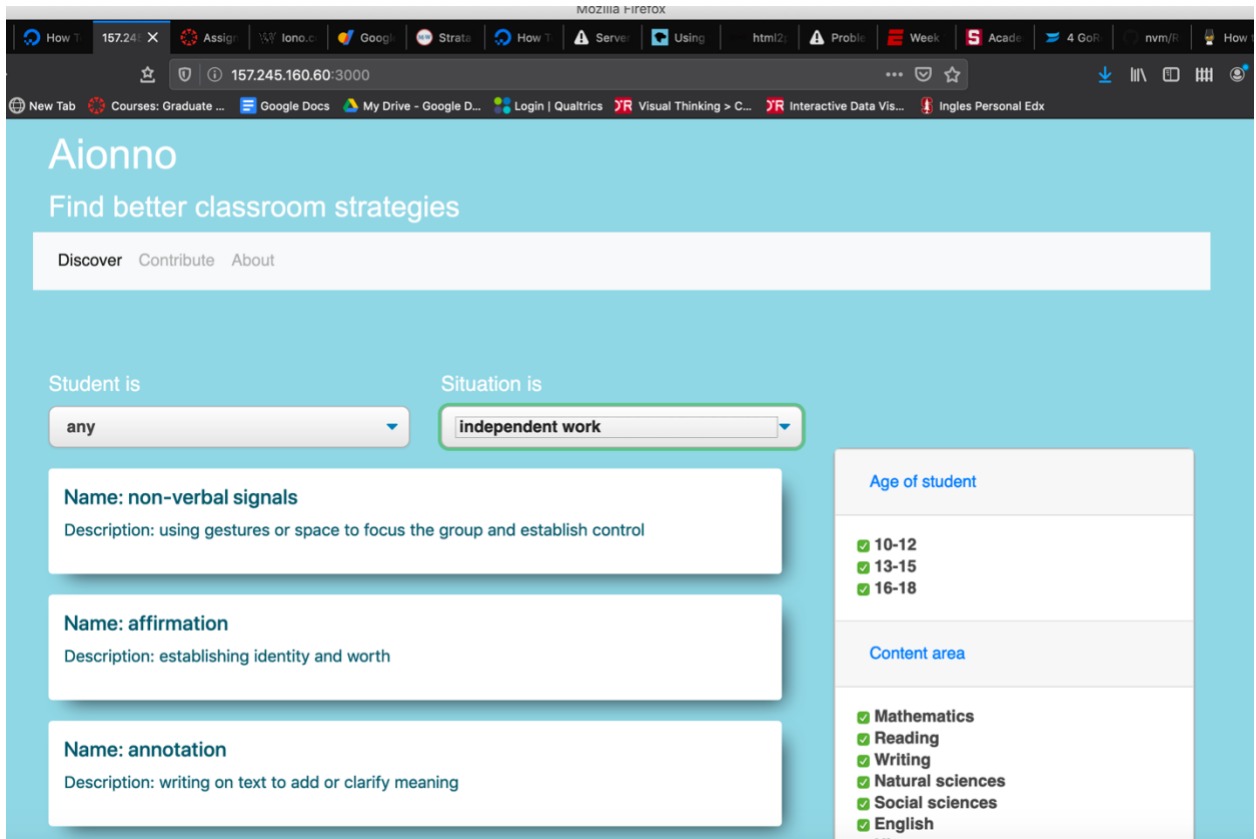


Diagram 7.8

I also performed some alpha testing with people at work and at home in order to determine if the interface was appropriate for these audiences. I determined that there needed to be some re-sequencing of the UI elements on mobile view, and also some of the typography and color needed to be changed to be accessible for everyone.

8. Conclusions

8.1 Re-purposing old code

The problems I encountered early in the process (September and October) were largely due to trying to generate an environment that could serve as a template. So, I resolved to change my tack; I would look for projects that could serve as jumping off points. With a workable set of code to start with, I could construct basic items like drop-down menus and get a feel for if the back-end was ready for dynamic templates. This was my plan, anyway.

I looked around for potential matches, but unfortunately the only sites I found that matched my desired functionality were proprietary. So, I instead started looking through past projects of mine and discovered an app that, while not beautiful, was functional and had a solid foundation for a multi-screen application. This project was built in the CSCI E-31 *Introduction to Web Applications Using Node.js* course. Fortunately, I documented my project well, so jumping back in took less time than looking at a new application from scratch. I took this as a “learning moment”; if open source code is not available, one should next give time to considering what pieces of formerly developed code could be incorporated in the name of expediency.

8.2 User testing

The problems I encountered later in my development (November and December) centered on perfecting the interface and polishing the user experience. With more testing, I rapidly unearthed non-intuitive patterns in my design, for example the placement of filters on desktop view vs. mobile view. I also got a better sense for how the user’s gaze tracked when first engaging with the interface. The navigation menu, an afterthought in my own mind, was a major focus for users. This compelled me to iterate further on the mobile navigation design, even condensing how filters in future versions could be more readily expandable and navigable on mobile devices.

If I were to perform more user testing, I would also take into greater consideration the professional development model an instructor is currently operating under. I found that broaching the topic of strategy use was tempered with the frameworks teachers were most familiar with, whether ADDIE, RTC, or even a home-grown model built by teachers in the school. While offering a clean interface is helpful, it is essential to build something organic to the current processes that the user operates under.

9. Next steps and future plans

While I am pleased with the initial look of the app, it is by no means finished. First, I plan on user testing with teachers in India and California further in order to get an idea of how it can be more navigable as well as how the information can be sorted and presented in a more intuitive way. For example, it is not clear to me how users would like to “collect” and preserve strategies for future reference. I’ll have to get users’ input first, and then I will try to code more pages into the app that have persisting data and database-like features for searching tagged or preferred strategies

I also would like to get a feel for if the interface is user-friendly enough for low-, mid-, and high-income countries and communities. I purposefully designed this interface to be very basic so as to not confuse less tech-savvy users. However, it is not clear to me if this erodes confidence, or if a more modern interface (e.g., Pinterest, Instagram) would be just as intuitive to users in low-income communities and low-income countries. My goal is to make the app maximally accessible, both visually and intellectually, to all users. Therefore, my inclination right now is to find the common denominator – namely, a very straightforward, text-driven interface.

10. Code

Github: <https://github.com/jaywise/Aionno>

Website: <http://157.245.160.60/>

11. References

- Anderman, Eric M., and Allison J. Young. "Motivation and Strategy Use in Science: Individual Differences and Classroom Effects." *Wiley Online Library*, John Wiley & Sons, Ltd, 19 Aug. 2006, onlinelibrary.wiley.com/doi/abs/10.1002/tea.3660310805.
- Borkowski, J., & Muthukrishna, N. (1992). Moving metacognition into the classroom: "Working models" and effective strategy teaching. In M. Pressley, K. R. Harris, & J. T. Guthrie (Eds.), *Promoting academic competence and literacy in school* (p. 477–501). Academic Press.
- Durlak, Joseph A., et al. "The Impact of Enhancing Students' Social and Emotional Learning: A Meta-Analysis of School-Based Universal Interventions." *Society for Research in Child Development*, John Wiley & Sons, Ltd (10.1111), 3 Feb. 2011, onlinelibrary.wiley.com/doi/full/10.1111/j.1467-8624.2010.01564.x.
- Fahner, Nancy, et al. "Instructional Strategies." *For the Teachers*, 17 May 2019, www.fortheteachers.org/instructional_strategies/.
- Garner, Ruth. "When Children and Adults Do Not Use Learning Strategies: Toward a Theory of Settings - Ruth Garner, 1990." *SAGE Journals*, 1990, journals.sagepub.com/doi/abs/10.3102/00346543060004517.
- Mcdaniel, Rhett. "Pedagogies & Strategies." *Vanderbilt University*, Vanderbilt University, 19 Nov. 2019, cft.vanderbilt.edu/teaching-guides/pedagogies-and-strategies/.
- Pressley, Michael. "PsycNET." *American Psychological Association*, American Psychological Association, 1990, psycnet.apa.org/record/1990-98396-000.
- Rosenshine, Barak, and Carla Meister. "The Use of Scaffolds for Teaching Higher-Level Cognitive Strategies." *Educational Leadership*, 30 Nov. 1991, eric.ed.gov/?id=EJ442788.
- Silberman, Mel. "Active Learning: 101 Strategies To Teach Any Subject." *ERIC*, Prentice-Hall, P.O. Box 11071, Des Moines, IA 50336-1071 (\$30.99)., 30 Nov. 1995, eric.ed.gov/?id=ED424243&xid=17259%2C15700021%2C15700124%2C15700186%2C15700190%2C15700201%2C15700237%2C15700242%2C15700248.
- Wolters, Christopher A. "The Relation between High School Students' Motivational Regulation and Their Use of Learning Strategies, Effort, and Classroom Performance." *Learning and Individual Differences*, JAI, 15 Feb. 2002, www.sciencedirect.com/science/article/pii/S1041608099800041.