COMPSCI399 Project

Computer-based Spatial Skills Testing

Version:

1.0

Prepared for:

Burkhard C. Wünsche

Prepared by Team 34:

Project Lead

Ethan Lin

Programming

Zac Cleveland

Design

Daniel Gildea, Rawiri Mahue and Kayles Holmes-Albert

Front-end Developer

Joshua Scragg

Executive Summary

The goal of our project was to design and implement systems to help in the evaluation and education of spatial skills. Research [1][2] has shown a clear correlation between spatial intelligence and success in the fields of Science, technology, engineering, and mathematics (STEM). Studies have also shown that STEM education and research are key components to improving societies' productivity, competitiveness and well-being [3].

Our project had two main goals, to develop a web-based spatial skills test and to extend an existing Coderunner-based spatial skills test. There was also one optional goal, to develop a standalone game-based spatial skills test.

Our team has successfully developed a website that is able to test the four major aspects of spatial skills [6][Appendix 2]. The website is also able to provide analytical data to both the users who sit the tests and the instructors who assigned the tests. These features allow the participants to gauge their performance and improve upon them. The website also allows instructors to upload new questions and create new tests, this feature allows the expansion of the testing capabilities of the website as new tests are developed.

We have also successfully extended the Coderunner test's existing database. These new questions were produced through our team's research into various sources relating to spatial skills such as research and websites [Appendix 1]. These new questions were then integrated into the existing Coderunner's database.

However, due to time constraints, our team was unable to meet the optional goal of developing a standalone game-based spatial skills test.

Overall our project's two main goals were met. There are improvements that could be made to further enhance the website. Such as website speed optimization, extension of supported question types, increasing the robustness of the quiz function and extending the statistics generated by our website. Our website's code is well documented and available on Github and can be used for future development.

Regarding the future creation of the spatial skills tests and questions, we recommend collaboration with psychologists to produce new and updated versions of tests. The development of standalone games can also be integrated in ways such that tests can be fun to take.

Table of Contents

Introduction

Aims and objectives

The overall aim of the project was to design and implement systems to test users' spatial skills through tests of various formats, from these tests produced analysable data that could be used to further train and improve the participants' spatial skills. The individual objectives were divided into two main categories and one optional category.

Main Goal One: Develop a web-based spatial skills test

Target audience

The website was designed in a way that professors, teachers and instructors can utilise the website to test their students and produce useful and quantifiable data for analysis.

Scope

The initial requirements of the web based application was to have the ability to test the four main aspects of spatial skills [6][Appendix 2].

They are:

1. Spatial or visuospatial perception.
2. Mental rotation
3. Spatial visualisation
4. Spatial working memory

A very important component of the website was that it must have the ability to display animated objects that are timed in order to have the capacity to test Spatial working memory.

Throughout various client meetings our goals for the website became clearer. The website also needed to have the ability to add new questions, create new tests.The teachers needed the ability to manage their classes, assign quizzes, create quizzes and view their class' statistics.

Approaches

We used the following Waterfall approach to produce the desired website. First, we outlined what we wanted the website to be capable of, we then designed the UI by drawing out what we wanted the pages to look like. After this, we worked on developing the website. After creating all of the outlined features, we worked on getting the website hosted on AWS. Finally, we tested our website to ensure that all of the systems were working. We choose development tools that allow for rapid prototyping, such as Next.js and React, to ensure that we could complete our website in the required time.

Outcomes

We had completed a functional website with the following features. Our website has two main systems, one for instructors and one for students. Instructors have the ability to create and import questions, create and import quizzes, and create classes. Through the class system, they can manage their assignments to specify which students should take which quizzes. This system allows for setting due dates on assignments. Instructors can also assign quizzes anonymously to see how their class performs at a higher level, not an individual one. Students can see which quizzes they have been assigned, take the quizzes, and also look back at their previous attempts.

Main Goal Two: Extend an existing Coderunner-based spatial skills test

Target audience

Coderunner is the University of Auckland's existing online website based testing platform. It is used by faculties to perform tests and quizzes for the University's students.

Scope

Due to the inability for Coderunner to display animated content, it was not possible to implement tests for Spatial working memory. Therefore our focus for coderunner was to implement new questions for the other three aspects of spatial skills.

Approaches

Our team had looked through many online resources [Appendix 1] to produce new questions to be implemented into the existing Coderunner's questions database. Throughout our research, many of the modern versions of questions were copyrighted. Whereas the old original scholarly articles were either unavailable or had low fidelity due to their age.

We had drawn our own spatial skills test questions based on existing versions of questions available. We also made sure our questions' formats were consistent with current versions so as to not alter their difficulty. We also looked into non-copyrighted versions of questions to see if they could be used as is.

Outcomes

Using a combination of questions our team had drawn and questions from non-copyrighted sources, each of the three sections of spatial skills were populated with the following new questions [Appendix Coderunner Github].

1. Spatial or visuospatial perception: 11 new questions.
2. Mental rotation : 12 new questions.

3. Spatial visualisation: 10 new questions.

Optional Goal: Develop a standalone game-based spatial skills test.

Target audience

The target audience was to be everyone. It was to be designed in a way so that players did not feel like they were being tested.

Scope

The standalone game needed to make the players feel like they were not being tested. While still having the ability to produce quantifiable results that can be analysed as part of a test.

Approaches

Our team had looked into existing games with spatial skills components and looked at ways to combine multiple aspects of spatial skills into a single game. Documents were created for team members to input their brainstorm ideas.

Outcomes

Due to time constraints, our team had focused our efforts on the two main goals of the project. Unfortunately this meant that we were unable to develop a functional game before the deadline.

Background

“Since at least 1999 we have known what spatial intelligence is” [5], research has been happening in the field ever since. We specifically want to build a platform that is capable of allowing students to be tested in a consistent and user friendly environment, that allows our researchers to take tangible measurements. For example, say we wanted to run a study comparing how students will improve in a subject like COMPSCI 373 based on their spatial skills. There could be two groups, one where the students perform spatial skills training and tests throughout the semester and another who just take the class. Using these results we could try to find some causation to whether or not spatial skills training improves grades in relevant subjects.

There are many online spatial skills tests already available for example https://www.123test.com/spatial-reasoning-test/, https://www.aptitude-test.com/free-aptitude-test/spatial-ability/. But the issue with these is that they tend to be quite short and any decent test is locked behind a paywall. Our solution allows us to create our own tests, swap in and out questions, and extract user results with ease.

Although our platform is a new solution our questions are not, we have based our questions completely in line with already existing and accredited questions. For some questions copyright was an issue so we had to re-draw them to circumvent that. Also our spatial memory game the “Corsi block tapping game” [4] has been taken and been implemented into our platform as a question type allowing for a memory based spatial skills test.

Specification & Design

Our project consists of two deliverables. Firstly, we must extend an existing CodeRunner-based spatial skills test. This required the collection of high-quality spatial testing assets, modifying them to be unique, and then creating a CodeRunner quiz using our custom assets. Students will then be able to take these quizzes using the CodeRunner application.

Secondly, we must build a web-based spatial skills test. For our website, we decided to create an application allowing instructors to manage classes, assign quizzes, create quizzes, and view class statistics. We wanted the platform to be similar to that of Canvas, but with a more narrow scope and a focus on quizzing. While this took longer than if we had excluded the class management system, we decided it would set our solution apart from the others. Once the instructor has set up a class, the students can log in using teacher-set passcodes and take the quizzes. Their results are tracked, which provides statistics for the instructors. While our website is geared towards spatial skills testing, any quiz content can be tested for any subject. This flexibility allows for the future expansion of our website into a general quizzing website.

For the quizzes, our team had looked at various online sources and compiled them into a questions repository [Appendix 1]. Each member assigned to the Coderunner component of the project were allocated numerous sources to look through and determine their suitability for our purpose of creating new questions.

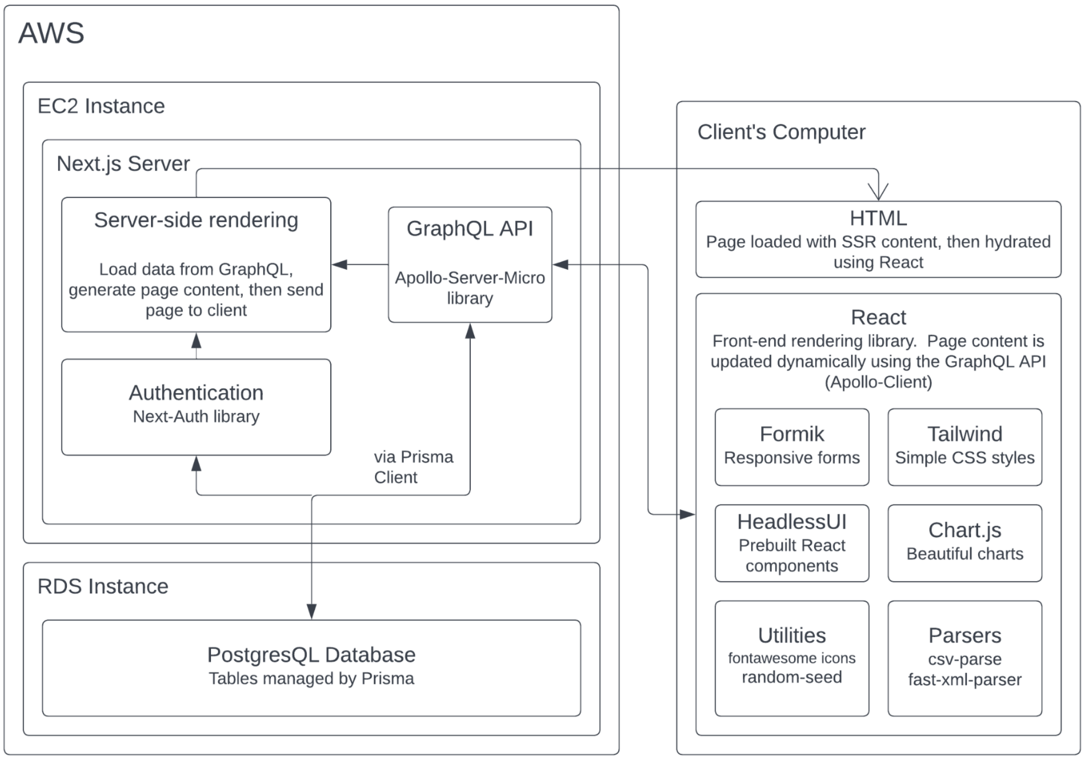
We used criteria such as image fidelity, accreditation, difficulty and copyright. For the questions that were deemed suitable but had copyright, our team had drawn our own versions inspired by the sources we looked into. We made sure our versions were unique, while also making sure they adhered to the same format as the sources. This was to make sure we did not inadvertently modify the questions' difficulty as that will affect the outcome of the tests. These new custom images were then implemented into the Coderunner database.

For the versions of questions where the sources had either no copyright or were free to use for educational purposes, we could use them as is and could implement some of them directly into the existing Coderunner database.

There were many choices for our backend server, such as Express, Next.js, AWS Amplify, ASP.NET, Ruby on Rails, Flask, etc. We decided on Next.js as our team members were most familiar with this platform. It also is designed for use with our front-end system, React. While there are many alternatives to React, such as Svelte, Angular.JS and Vue.js, again React was the front-end tool that our team was most familiar with. This made the combination an easy choice.

To allow anyone to use our website, we used an AWS EC2 server for hosting. This means that our website is available online for all to access. This ended up being the easiest solution for hosting, as an EC2 instance is a virtual computer on which we can run any code.

Architectural Overview



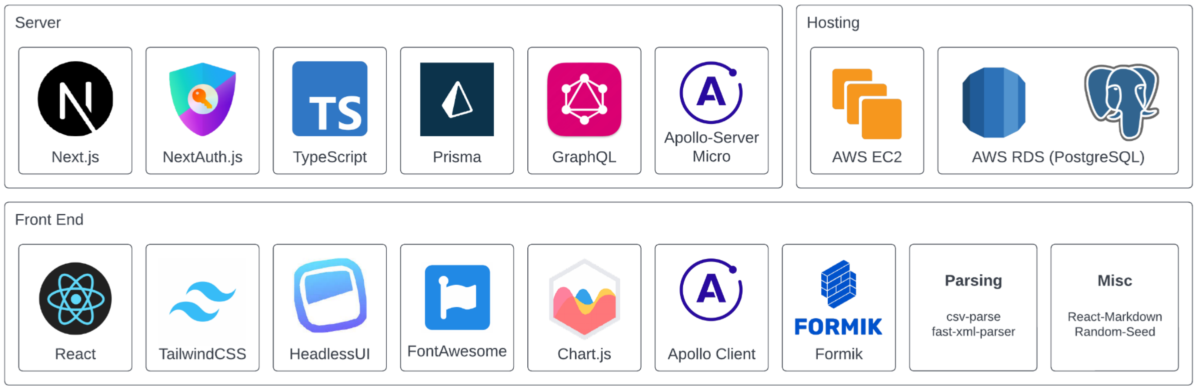
Our system contains three main components. The Next.js server component runs inside the AWS EC2 instance and serves our web pages to our users. The front end runs in the clients' browser and communicates with the server using GraphQL. We also have a database hosted with AWS RDS, which communicates with the server using Prisma client.

Importantly, the GraphQL server connects to our database instead of the client connecting. This allows us to perform database actions securely without giving the client's computer access to our database.

Implementation

How did you implement your design?

Technologies Used



We implemented our website's backend using Next.js to allow for a responsive user experience. We use Typescript Language to ensure that our code is type-safe, which prevents an entire class of errors from occurring in our code. We use Prisma to design our database schema, manage the database, and use Prisma Client to connect to our database from the backend. We use NextAuth for authentication, which allows us to easily add OAuth login (such as Google and GitHub) to our website. Finally, we use Apollo Server Micro as the GraphQL server to communicate from the server to the client.

On our front end, we use React for a responsive user experience, which goes hand-in-hand with Next.js. We used several React extensions. Tailwind CSS was used for the rapid styling of our components. We also used Tailwind's Headless UI library, which contains some useful prebuilt components. FontAwesome's React library was also used for icons. We used multiple parsing libraries on the front end, such as csv-parse and fast-xml-parser. Formik's React library was used to create forms with simple validation, form submission, and error handling systems. We used the Apollo Client to connect to our GraphQL server. We used React Markdown to display markdown formatted descriptions. We used Chart.js for our statistic charts, which enabled us to present our calculated data quickly. Finally, we used the random-seed library to allow our quizzes to have the same randomly generated content for all quiz users, as required.

Finally, to host our website, we used an AWS EC2 instance as well as an AWS RDS Database running PostgreSQL

During development, we decided to change our Server ↔ Client communication protocol from a REST API to a GraphQL API. This allowed us to write a more structured backend using the Apollo GraphQL libraries designed for React. Using GraphQL also allowed us to update the data on our page using React hooks dynamically. This made subsequent development much faster. However, this change took a week to research, design, and implement. The delay took us behind schedule and meant we ran out of time to implement some of the required features.

Another unforeseen problem that we encountered was difficulty hosting the server using AWS. We initially decided to use AWS Amplify, which Amazon stated to be compatible with Next.js, our server framework. However, when it came time to host our application, we discovered that Amplify supports Next.js version 11, and we built it using Next.js version 12. To account for this, we switched to AWS Elastic Beanstalk for our hosting. Once again, we had issues with Elastic Beanstalk. Our website would not build correctly on the platform. Finally, we decided to use an AWS Elastic Compute 2 instance. EC2 provides a private server that allows us to host the application the same way we hosted it for development. This solution ended up working well for us.

Results & Evaluation

Our overarching goal was to create a testing platform to allow researchers to conduct spatial skill studies in as easy a way as possible, both for the researchers and the subjects participating in the study. Considering this, I believe we have achieved our goal as we have created a platform to easily create, assign, and take spatial skills tests both from the point of view of a researcher and a subject.

Throughout development, we tested features as they were added; for example, when the class dashboard was added, there were a lot of fields that needed to be tested to ensure they were behaving as expected. One of those fields was the due date for assignments we ran into issues where the date selected would update to a month or so behind what they had initially selected. Also, in a final test we noticed that there were some security flaws we hadn’t noticed when developing the system.

In light of testing, we realised that although we have authentication, there are some routes that do not check ownership. For example, if someone has a user's assignment ID they would be able to change the grade attached to said ID; this was a massive oversight in our development and what I would say is our biggest weakness. For strengths on the other hand, the biggest is that we were able to do what we set out to do, create a platform to allow researchers to conduct spatial skills testing. More specifically, I would say that the actual quiz taking for us is a strength; we have implemented persistence so that even if a user closed the app, it would pick up where they left off, allowing for flexibility for the users.

Future Work

What are the ideas/features that you could not implement in your system?

While our website is, for the most part, complete, there are still a handful of features we could not implement before the deadline. The most important feature we did not implement was Moodle XML exporting. This would allow teachers to export the questions they have created into Code Runner. The added flexibility would ensure that teachers could re-use the questions they create inside our website on other platforms. Another feature that we did not implement was an accurate Corsi block tapping test. The Corsi test we implemented had the blocks aligned in a grid, unlike the standard Corsi test. Finally, the last feature we ran out of time for was the ability to edit existing questions on our platform. This would allow teachers to modify questions after they find out the question is too easy or too hard. One idea that we did not have time to explore was user testing. With comprehensive user testing, we could optimise our platform for the students we created it for. This would ensure that our user interface and experience are not confusing.

How can your system be enhanced in the future?

In the future, we could enhance our website by adding several features. Firstly, we could extend the statistics captured. Right now, we only track the percentage of students that have completed each quiz, their average grade, and a breakdown of grades for each quiz question. We could add more detailed statistics, such as which questions students spent the most time on, individual student statistics, and new types of graphs to allow teachers to understand their students better. Another feature we could implement was to include more question types. Since our quiz applet allows for interactive questions, I believe that we could build more to make our application stand out. Interactive questions are a unique point of our website. We could also implement more security features in our backend code. At the moment, all of the backend code checks to ensure that the user is logged in; however, it does not check if the user is modifying their resources, or someone else's. This is very insecure, so care will need to be taken to implement this in the future before the website is released to the public. Finally, we should refactor the code. By spending some time re-writing the code for our website, we could make the website much easier to maintain and expand upon.

Conclusion

To summarise, the overall aim of this project was to offer students and teachers a set of platforms to assess, analyse, and improve their spatial reasoning skills. With our platforms as a medium to improve users' spatial skills over time, we ultimately hope to provide students a unique way to increase their likelihood of success and achievement in STEM-based subjects, as well offer teachers tangible statistics to measure their students' progress in this endeavour. This aim was realised by implementing a CodeRunner-based and web-based spatial skills test which allows users to assess the four key aspects of spatial intelligence; namely, spatial perception, spatial visualisation, mental rotation, and visuospatial working memory. Via the successful deployment of a web-based platform where users can obtain quantifiable scores and hence easily compare and improve their spatial skills over time, as well as the successful extension of an existing CodeRunner-based spatial skills test, we believe that the overarching aims of this project were met. One key finding of this project was that testing certain aspects of spatial skills would prove to be a challenging task. For instance, established multichoice tests for the visuo-spatial perception category were virtually nonexistent online so it was difficult to move beyond trivial map reading questions in the design process. In terms of major outcomes, the web-based platform was successfully implemented up to the desired standard, and the CodeRunner questions were extended, however the optional game-based spatial skills test wasn't completed due to time constraints and pressure from other courses. Notwithstanding the fact that certain aspects of the project could be improved, the novel statistics-based features offered in our web-based platform more than exceed our initial expectations, and the challenges overcome to produce the final result are a testament to the perseverance of our programmers and developers.

References

[1] Duffy, Gavin & Farrell, Stephanie & Harding, Rachel & Behan, Avril & Raighne, Aaron & Howard, Robert & Nevin, Edmund & Bowe, Brian. (2015). The effects of spatial skills and spatial skills training on academic performance in STEM education.

[2] Young Christopher J., Levine Susan C., Mix Kelly S. The Connection Between Spatial and Mathematical Ability Across Development. IEEE Frontiers in Psychology., vol. 9, (2018). DOI: 10.3389/fpsyg.2018.00755.

[3] B. Freeman, S. Marginson , R. Tytler. An International View of STEM Education. IEEE STEM Education 2.0, pp. 350–363, Aug 2019, DOI:https://doi.org/10.1163/9789004405400_019

[4] Brunetti, Riccardo, et al. “eCorsi: implementation and testing of the Corsi block-tapping task for digital tablets.” *Frontiers*, 6 August 2014, https://www.frontiersin.org/articles/10.3389/fpsyg.2014.00939/full. Accessed 29 October 2022.

[5] Wikipedia. “Howard Gardner.” *Wikipedia*, https://en.wikipedia.org/wiki/Howard_Gardner. Accessed 29 October 2022.

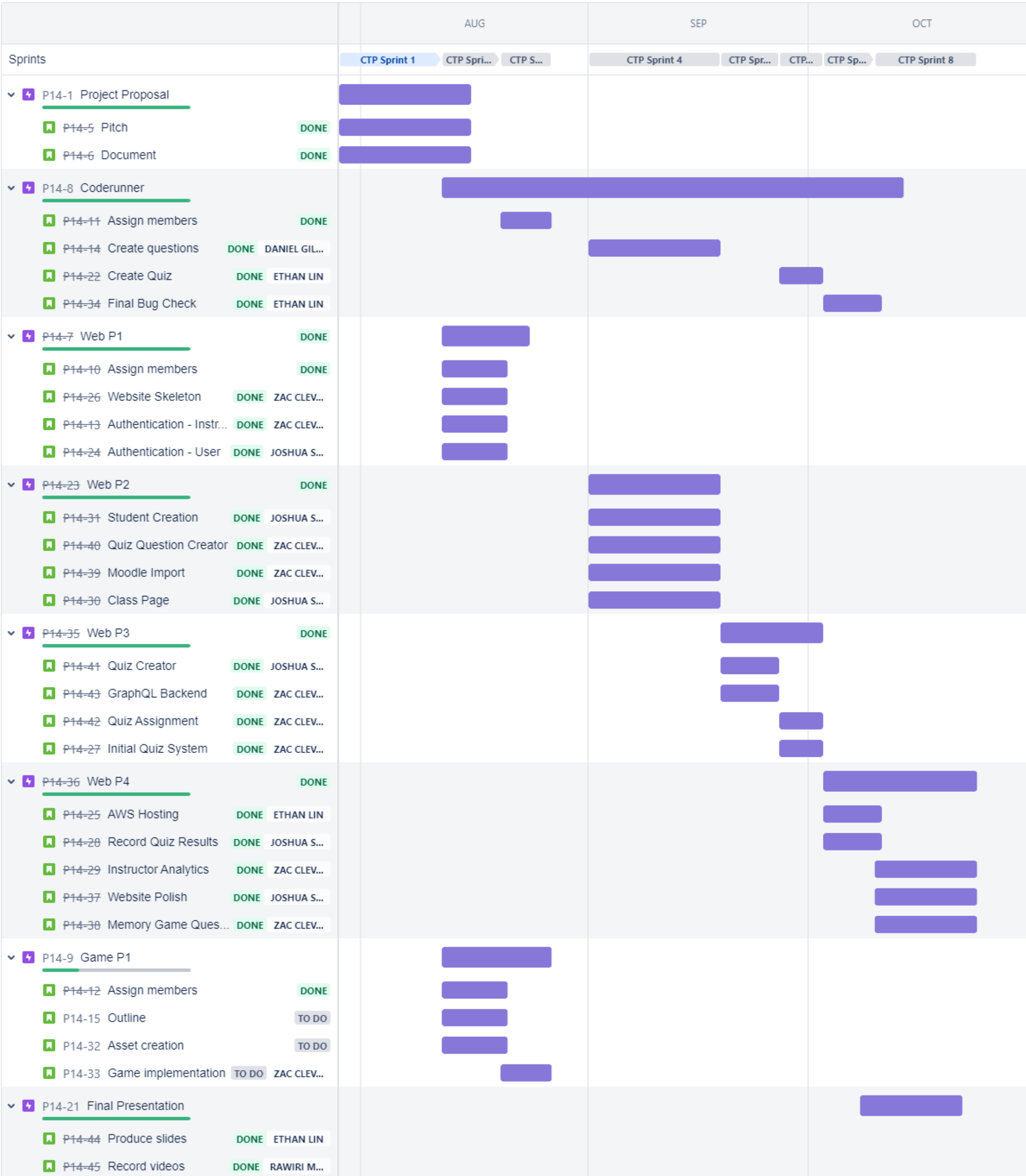
[6] Wikipedia. “Spatial Ability.” *Wikipedia,* https://en.wikipedia.org/wiki/Spatial_ability Accessed 30 October 2022

Declaration of Authorship

Section	Author(s)	Signature(s)
Executive Summary	Ethan Lin	
Table of Contents	Rawiri Mahue	
Introduction	Ethan Lin, Zac Cleveland	
Background	Joshua Scragg	Josh Scragg
Specification & Design	Zac Cleveland, Ethan Lin	
Implementation	Zac Cleveland	
Results & Evaluation	Joshua Scragg	Josh Scragg
Future Work	Zac Cleveland	
Conclusion	Rawiri Mahue, Daniel Gildea	Daniel Gildea
References	Ethan Lin, Joshua Scragg	Josh Scragg
Appendices	Ethan Lin	

Appendices

Figure 1: Jira Roadmap



Link to Team 34 Spatial Skills Test website:

https://spatialskills.team34.software/

Link to video:

https://drive.google.com/file/d/1QtZshXRVoZyvSGU5HHiQ39ouAGK4dmyA/view?usp=sharing

Link to Team 34 website Github repository:

https://github.com/uoa-compsci399-s2-2022/Website

Link to Team 34 Coderunner Github repository:

https://github.com/uoa-compsci399-s2-2022/Team34-CodeRunner

[Appendix 1]

Existing Spatial skills tests: Questions repository.

https://docs.google.com/spreadsheets/d/1jSuRuexDit7RhmNQmKbVwP3ASTcB8gt2_E7a_pU35S8/edit?usp=sharing

[Appendix 2]

Burkhard C. Wünsche. Spatial Intelligence and CompSci.

https://docs.google.com/presentation/d/1Ihj-jVbbXMVHUAgN_1SrH7dXis_fmW-_/edit?usp=sharing&ouid=101928513053171616985&rtpof=true&sd=true