

Proposal for Web App

https://spanish-lab-client.vercel.app

May 1, 2023

Makayla Broyles Cole Milne Geri Lynn Viallon makaylabroyles1@gmail.com colemilne54@gmail.com geriviallon@gmail.com

Abstract

In language classrooms all over the world, students spend most of their time learning vocabulary lists and grammar rules. Despite knowing words and grammar, many remain ineffective communicators in their second language. There are many language apps available, some of them excellent. But most of these are gamified ways of learning vocabulary and grammar. Our Interactive Spanish app does not aim to replace these ways of learning, but rather offers a complimentary tool where students must be active participants, creating their own learning experience.

Our app offers students a fun way to build a narrative. It encourages them to use what they already know, visually reinforces their ideas, and gives them opportunities to create and interact in Spanish. By offering structured choices, an entire class can use the app to vote on what happens in the story, making the choices that move the story forward. The AI image generator is another effective aspect of the experience. Since each image is generated for that particular story, memory is reinforced. There is a sense of accomplishment in creating something unique. We want to give teachers a platform that leverages interaction, immediate feedback, and personal creation. We are hoping that this technology will help humanize the language classroom. It can also be used by individual learners, on their own or to complement their classroom experience.

Abstract, update

Our vision stayed very stable throughout the entire process. The hope of a fun app that would help Spanish teachers have an interactive activity to the class using visuals was our purpose all throughout.

Introduction

The National Academy of Engineering, in 2008, named 14 Grand Engineering Challenges facing 21st century humanity. Advancing personalized learning is one of these challenges[5]. More recently, the challenges that schools, students and families faced during covid-realated school shutdowns have only made the need for high quality online resources more acute.

Although skilled teachers are the single most important factor for effective learning to take place, education resources simply don't allow for this level of personalization without the use of technology. With the help of tech tools, teachers can provide an experience for students where success and growth is the norm. The experience of both struggle and success is foundational to successful learning.

As to the choice of a Spanish learning app, 13% of the US population are native speakers of Spanish. This makes it the country with the second largest Spanish-speaking population in the world [7]. Living in a country with these demographics, it would seem beneficial for everyone to have at least basic Spanish skills. The third challenge we hope to take a small bite out of is the educational technology gap, particularly in the language classroom. Spanish Lab is meant to be extremely easy for teachers and students to implement.

We are creating an interactive Spanish classroom application to help teachers and students create and participate in a more hands-on atmosphere when learning Spanish in the classroom. It will give teachers and students an application that leverages immediate feedback, picture memorization, and a fun way to learn Spanish!

What makes SpanishLab different?

There are many language learning apps available. DuoLingo is by far the most popular, with over 100 million downloads on the Google Play Store[2]. After DuoLingo, the other most popular language apps are geared toward learning English, since most of the world learns English as a second language. Babbel, Busuu, and Rosetta Stone offer many languages and have around 10 million downloads each (Google Play Store). Finally, Kahoot is an educational quiz app which has influenced the design of SpanishLab, so I will mention its functionality as well. What surprises me most is how similar these different apps are. All are divided very linearly into lessons, along grammar and vocabulary themes. It is quite similar to reading a language textbook. Of course there are many advantages over a textbook: you get immediate feedback, there is audio content, you can respond using voice. I would call these apps interactive textbooks. They are gamified enough to make it feel somewhat like a game. They also show you your progress and remind you to keep to your studying goals.

Our app is not a broad Spanish Learning app, but rather is destined to be a fun, creative tool to help learners. It doesn't intend to explicitly teach particular grammatical or vocabulary items, though the teacher could certainly structure it to do so.

Kahoot is an educational app where teachers can build a quiz, and students can login and answer questions. Our app will use aspects of this, but instead of responding to questions with right/wrong answers, students will be voting on what they want to happen in the story.

What makes Spanish Lab different, Update

Spanish Lab's aim is still a classroom tool that adds some fun and imagination to learning. It would work as a filler activity for spare moments in the classroom. It can be used to complement a story-based lesson. It could be used to add more interactivity to a traditional lesson.

Pedagogical Basis

The inspiration for this idea came from a local Spanish teacher, Kristen Wolf, who regularly uses TPRS methodology in her classroom. Teaching Proficiency through Reading and Storytelling is proving to be an effective tool in the modern language classroom. A typical TPRS lesson follows 3 basic steps[1]:

- Show. The teacher shows students the meaning of a few language structures. She uses gestures, drawings, comprehensible input, cognates, and translation if necessary.
- Ask. The teacher asks questions, drawing responses from the students, forming a narrative together. She may propose options and let the class vote, or may gather ideas and choose among them.
 - The teacher can ask Who? What? Why? questions to deepen the narrative and give personality to the story.
 - Students might act out the story.
- 3. *Read.* Using the story students just created, they can read elements of it, or the entire story. This will consolidate understanding. It is also helpful to use another processing modality to strengthen memory and deepen learning.

Teachers find this method to be highly engaging for their students[4]. Studies have shown that TPRS boosts student engagement, autonomy and enthusiasm.

Our hope is that the app would be a way to complement a TPRS lesson. It could also

"Co-creating stories with the teacher results in a heightened sense of personal ability and belongingness to the group." be useful to teachers wanting to test out TPRS methodology in a simple way. And our hope would be that teachers with no exposure to the

TPRS method would also find it a useful tool in their classroom.

Our Team

We are a diverse group and each of us have different reasons that this project is a motivating one. Makayla is experienced in microservice development, which she has been doing at Zoot since last summer. Her people skills make her skilled at figuring out what customers want and need. Cole is skilled at both marketing and web development. He has an entrepreneurial spirit and runs his own SEO/marketing business. Geri, before deciding to study Computer Science, spent most of her time abroad teaching English and French. She was already pursuing gamification to make the process more engaging to students.

For our project, Geri will be doing most of the backend, Makayla the frontend, and Cole will be floating between the two.

Roles update

Makayla: frontend styling

Cole: full-stack

Geri: secretary, cheerleader

COLE MILNE

(916) 934-3287 | <u>colemilne54@gmail.com</u> | 614 South 15th Ave, Bozeman, MT 59715

EDUCATION

Montana State University, Bozeman Bachelor of Science in Computer Science (in-progress) 2019- Spr. 2023

Montana State University, Bozeman Minor in Data Science (in-progress) 2019- Spring 2023

AWARDS

- Dean's List (3.5+ GPA) Fall 2019, Spring 2020, Fall 2020, Spring 2021
- President's List (4.0 GPA) Fall 2021
- GPA 3.76
- Western Interstate Commission for Higher Education's Western Undergraduate Exchange 2019-2023

Student Organizations

Sigma Alpha Epsilon Fraternity

- New Member President Fall 2021
 Recruitment Committee Member Spring 2022
- FSL New Member Award Nominee Spring 2022
- Philanthropy Chairman Spring 2022
- Community Service/ Volunteering Chairman Spring 2022
- Eminent Recorder Spring 2022

RELATED EXPERIENCE

The Milne Empire FOUNDER & CEO

I run a digital marketing agency that helps grow business online. We achieve this task primarily through web design and development, along with increasing organic traffic through search engine optimization (SEO).

Forex League Junior Developer Fall 2019

During the Fall semester of 2019 I worked for Forex League in which I helped develop their new app along with generating business ideas to help the company's growth. The technologies I used were Ionic, JavaScript (TypeScript), CSS (SASS), and HTML. Every week I would meet with the CEO and share my progress and any prototypes I had developed. I also developed a successful sales funnel that is still used today.

Defense Distributed

Full Stack Developer -Summer-Fall 2021

During the Summer of 2021 and into the Fall semester, I had an internship with Defense Distributed as a (MEAN) full-stack developer. I created both the frontend and backend of web applications.



MAKAYLA BROYLES

MICROSERVICES DEVELOPER

ABOUT ME

As I grew up I was always intrigued by technology due to all of the capabilities technology allows in daily human life. As I grew older, I realized that a career in technology would push me to be creative and initiative to bring more capabilities to the world. This has inspired me to pursue a computer science degree and work towards developing software that is impactful and helps people and companies.

CONTACTS

+1 (406) 200-3249

makaylabroyles@icloud.com

A Bozeman, MT

EDUCATION

September 2020 - December 2023 Computer Science Montana State University

September 2020 - December 2023 Business Finance Minor Montana State University

SKILLS

- Communication
- Attention to Detail
- Problem-Solving

LANGUAGES

- Java
- C/C++
- HTML/CSS
- JavaScript
- SQL

WORK EXPERIENCE

Microservices Developer

Zoot Enterprises | Bozeman, MT | June 2022- Present

- Undertake development and maintenance on Zoot's microservices libraries
- · Build and maintain data provider APIs
- Full stack development for a demo of Zoot's capabilities

Microservices Internship

Zoot Enterprises | Bozeman MT | September 2021 - May 2022

- Service platforms throughout Zoot's tool
- Undertake in the development of client specific web screens

۲

۲

۲

User Experience Internship

Zoot Enterprises | Bozeman, MT | May 2021 - August 2021

- Test and evaluate web pages for accessibility using tools such as NVDA
- Update platforms with compliance to WCAG Guidelines and Web Development standards.
- Engage in the design of prototyping diagrams as well as develop personas

Coffee Barista

Coffee Mill Espresso & More | Billings, MT | November 2017 - July 2020

- Managed morning rush of over 30 customers daily with efficient, cheerful customer service.
- Enthusiastically engaged with customers during busy periods and kept mood light and fun.

Zoot Enterprises | Bozema

Geri Lynn Viallon

Ed Tech Desiger

contact

1408 ½ S Willson Ave., Bozeman, MT 59715

geriviallon@gmail.com

406 209 5531

goals

During my time as a language teacher, I was able to see how introducing technology in the classroom was truly motivating for students. I believe that learning and mastery are basic human drives. In the right circumstances, learning can be highly effective, pleasurable and affirming. Technology can support this kind of learning by lowering economic and affective barriers. The result would be education that is both more cost effective and more closely targets student needs.

skills

- Unity
- Blender
- Java
- Python
- C
- French
- Spanish

education

MSU

2019 – present Senior in Computer Science

GPA:3.66

MSU

2014

BA in Liberal Studies, capstone focusing on resilience.

experience

Knick Machining, Bozeman. Admin

2018-2019

I kept the workflow smooth through via many channels: entering orders, ensuring their progress, and mediating customer expectations. I also filled holes in production such as passivating and delivering parts.

Online French Tutor

2015-2017

I created highly tailored French lessons. Complete beginners up to intermediate speakers. Emphasis on communication.

English Teacher, IUT (France) and private lessons

2009-2016

I introduced students to a wide variety of English content and structured lessons that maximized intrinsic motivation. Turned extensive knowledge *about* English into a usable skill through creative projects.

Previously

I pursued my interests in language and education in numerous ways: studying abroad, working at Louisiana School for the Deaf, and teaching English in Xian, China.

Work Schedule

We will use Agile as our lifecycle approach. Weekly checkpoints will allow us to keep motivated, and to have access to each other's ideas and skills as we complete our tasks on our own time. We may need to meet more often as the deadline gets closer, especially once we are able to do some user testing. That may make our iteration loop shorter, and we can add meetings if necessary.

Schedule - 2nd Semester



Schedule, update

We did more work in our own corner than originally intended. Because of this, in

person meetings were often need-based rather than weekly. We did weekly

progress checks and met when we were stuck or needed to consult.

Proposal Statement

The SpanishLab application is an interactive classroom Spanish story generator. It helps teachers bring an interactive component to the classroom. The application will feature 5 different theme options- based on skills that the teacher will be able to share with her class. Students will all connect to the classroom story with a code and participate in making decisions that will impact the final story. This allows for all students to be participating and creates a platform where students will learn Spanish through pictures and story-telling instead of right/wrong vocabulary lessons. There will be a final story generated after all questions have been answered that provides the teacher and students the opportunity to look through the story and translate any words that they may be confused on. The following requirements have been determined in order to acquire this functionality.

Non-functional Requirements:

- 1. The login functionality shall be secure.
- 2. The application shall be quick to receive answers from the other devices.
- 3. Photo generation should be quick and make sense to the students.
- 4. The application should be compatible with all devices.
- 5. The application should be easy to maintain and add features.

Non-functional Requirements, update

We have met these requirements.

Functional Requirements:

- 1. The website will keep track of the story based on student selections.
- 2. The website shall have premade story templates that the teacher will be able to choose for an interactive class.
- 3. All questions and answers for a given story shall make sense together.
- 4. The application will generate pictures with an AI Picture Generator in order to help students learn Spanish.
- 5. The application shall have translations features built into the final story generated in order to help students understand the story they have created.

Functional Requirements, update

2. There is a single template at this point.

5. The app does not do translation (and some teachers would argue that it is

better this way.)

All other requirements have been met.

Interface Requirements:

- 1. The website will have two different views student and teacher view.
- 2. It will be accessible across mobile devices and computers.
- 3. It will be readable and follow best practices for design, layout, and color.
- 4. It will be simple and easy to use for new users.

Interface Requirements, update

There are teacher and student views. The teacher view belongs to anyone who

creates a story.

The interface is functional, quick, intuitive, and accessible across devices.

On the following pages a prototype of the GUI has been attached. This shows the basic layout of the application and the basic functionality that both the student and the teacher will see. Notice that the mobile phone pictures are to represent the students view and the desktop top size prototype is to represent the teacher view of the application.

Spanish	Spanjsh	Spanjsh
Log In Username Password Submit	Enter Code Submit	Donde esta el león? With Con Nueva York Con nostros en Africa

Spanish	
Log In	
Username Password	
Submit	









Click to Translate Words:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Arcu odio ut sem nulla pharetra diam sit. Congue quisque egestas diam in arcu cursus. Integer malesuada nunc vel risus. Odio tempor orci dapibus ultrices in iaculis nunc sed. Quam vulputate dignissim suspendisse in est. Non arcu risus quis varius quam quisque id diam. Purus gravida quis blandit turpis cursus in hac habitasse platea. Mauris pharetra et ultrices neque



LOG IN

Click to Translate Words:

Translation

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Arcu odio ut sem nulla pharetra diam sit. Congue quisque egestas diam in arcu cursus. Integer malesuada nunc vel risus. Odio tempor orci dapibus ultrices in iaculis nunc sed. Quam vulputate dignissim suspendisse in est. Non arcu risus quis varius quam quisque id diam. Purus gravida quis blandit turpis cursus in hac habitasse platea. Mauris pharetra et ultrices neque

Frontend update

While our final user interface doesn't look exactly like the images above, the most important features are the same. The color scheme is still blue tones. The logo is the same. We did not include the translation feature. The multiple choice questions became more linear, and do not have images. Adding images is a goal for Next Steps. Making items vertically linear was important for readability. Verticality also allows any number of choices without impacting comprehensibility.

See section, Walkthrough, for final screenshots.

Use Cases

As for our use cases, we will have two main actors/ users that will be interacting with the client of our web application. There will be a minor actor who is able to view the final story. The flow of our use case will follow as such:

The teacher has different application permissions that the students have. The teacher will create the basis of the story that will be played through, and will initiate the beginning of the game/ storytelling. Students will then launch an instance of the game (one instance for each student) to vote on and/ or create story elements as it develops. By the end of the playthrough, an entire story will be created and from that our DeepAI Text-To-Image Generator API will display an image based on that story the class collectively wrote.



Development Tools

Our project will follow best practices for current web standards. We will be using a MERN stack for development which includes MongoDB, Express, React.js, and Node.js. We find that with our experience in our jobs and in the web development class this will be the easiest for us to implement. We will also be using a couple APIs including a Google Translate API and an AI Image Generator API. These will be used to easily translate language for our users and contribute to our story with AI generated pictures.

We will be using GitHub for version control and collaboration across multiple people. We have decided that we can all use our own IDE since we will be ultimately using GitHub for collaboration. We will also be using AWS web services for our server and deployment side of our project.

Methodology

Design Patterns and UML Diagrams

In our research we found there to be 3 common categories of design patterns: Creational Design Patterns, Structural Design Patterns, and Behavioral Design Patterns.

Creational design patterns are defined as such by netsolutions.com, "creational design pattern deals with object creation and initialization, providing guidance about which objects are created for a given situation. These design patterns are used to increase flexibility and to reuse existing code [6]."

Amongst these creational design patterns is the factory pattern. The factory pattern is one of the most commonly used of the creational design patterns, especially when developing in Java. In our use case of designing the SpanishLab we will use the factory pattern for user creation/ registration. The factory pattern will work as follows for our design:

- 1. Create a User interface
- 2. Create concrete classes implementing the User interface
- 3. Define a factory class UserFactory
- 4. Create demo class RegistrationPage
 - a. Uses UserFactory to get a User object

b. Sends information (Teacher or Student) to UserFactory to create the right type of object

In Unified Modeling Language (UML) this will look like:

User Registration Factory



Now we have a pattern for how we create users and how we assign different user types/ permissions depending on if the user is a teacher or student. What we need to do next is to implement a design pattern for how we create instances of a game. As a teacher will only be creating one game at a time another creational design pattern will be used, the Singleton design pattern. The Singleton design pattern is used when a class is responsible for the "creation, initialization, access, and enforcement" of a single instance object [2]. This is useful because it is designed to *only* allow for a single instance of an object to exist. In the context of our app the SpanishLab, this would mean that a teacher can not accidently make extra game instances.

In UML, this design pattern would look as follows:



Component Diagram (Broken up to fit on page):

React.js Front-End



Node.js Backend:



MongoDB Database:



UML update

The biggest change is that we did not use a database. SocketIO ended up being an extremely useful library. It facilitated many of the functions and meant that we no longer needed a database. See the following section, *Design Trade-offs update*, for further discussion about this choice.

The Singleton pattern as well is built into the functionality of Socket IO.

Design Trade-offs and Decisions

When we first sat down to figure out our deliverables we met with a Spanish teacher, Kristen, at a local elementary middle school, with whom Geri is friends. As we were ambitious in the beginning because this was our first step in actual work, and because Kristen was the original ideator of the SpanishLab, we initially set very lofty goals of what we wanted the app to accomplish.

We quickly realized we were in over our heads when we sat down with all of these grandiose ideas and tried to accomplish parts of this proposal. It took us a bit of

time for us to come to the conclusion that if we were getting overwhelmed writing a proposal about our web app, then actually programming it will be nearly impossible for us to accomplish given our experience and time frame.

We zoomed out a bit and tried to figure out what our main goals were. We wanted a way for teachers to teach their students Spanish in a fun, interactive way through custom storytelling. Getting a succinct overview of what we wanted allowed us to narrow back in on our deliverables and end up with a proposal of a web application that will be realistic for us to implement come next spring semester.

Design Trade-offs update

SocketIO

SocketIO was our big discovery that made many other parts of this project simpler. SocketIO is a TCP-based connection. It is bidirectional, real time, and event-driven responses [9]. SocketIO uses web Sockets, but many of the more difficult bits of web Sockets are abstracted away.:

- Unlike webSockets, it can get through a firewall or a proxy.
- SocketIO has built-in reconnecting.
- There are *rooms*, to facilitate sending different messages to different

groups.

• And generally makes it easier across devices and browsers.[11]

This led us to the surprising conclusion that we no longer needed a database. SocketIO creates the group, and all necessary information is shared between members of the group for as long as the connection is maintained. It also meant we no longer needed accounts or logins. A teacher creates a room by going to *create classroom* and creating a room code. The students then gather there. This means the set uptime for beginning an activity is short. The other benefit is that students can do this activity independently as well.

I think the only downside to this is that each story is ephemeral. Once everyone in the room exits, it disappears. Of course this could be fixed with a database in a future version.

Simplicity

Our original version was much more complex, as things often are in the dreaming phase. But when our dreams were confronted with implementation, we decided we would rather have an app that would take roughly 10 seconds to learn to use, for students and teachers.

Expected Results

I am imagining a classroom where students are excited because they know that today is story day. We want this tool to be an engaging way for teachers and students to collaborate in the classroom. I hope that the process itself will go so smoothly and the technology will be nearly invisible. We hope that students go home and show their families the stories they made in class.

Did we meet Expectations?

We believe that we have met the most important goals: it is both *easy* and *enjoyable* to use. Since most language classrooms tend to be low-tech, this ease of use is extremely important. Without it, there would be no chance of adoption. Smoothly tallying the votes is another invisible feature where its quick functioning is vital, and is working in our favor for adoption.

Although the framework is in place for a bigger story to be told, that functionality isn't complete at this point.

If we could do it over again...

- Prototype early. Low-fi prototypes are very important for having a clear vision of what we want to do.
 - Doing user testing on low-fi prototypes
- Better time management of how long things will take.
- More communication, not being afraid of conflict
- Choosing technologies that group members were more familiar with

Unresolved issues perro sushi!

A type//paniel-bed-dient vereelappi//room/pup/ A type//room/pup//ro	<text></text>		
<page-header><page-header><image/><text><text><text><list-item><list-item><list-item><list-item><list-item><list-item><list-item><section-header><section-header></section-header></section-header></list-item></list-item></list-item></list-item></list-item></list-item></list-item></text></text></text></page-header></page-header>	← → C O A https://spanish-lab-client.vercel.app/#/room/purple		☆ ♡ ৬ 0 원 ≡
Prompt: ¿Qué está comiendo el animal? hamburguesa biza burnio Comma francer Popular Answer: burnio burnio burnio burnio 	Room: purple	panjsh	LOGIN
Prompt: ¿Que esta comiendo el animal?			
 Instructuyees sushi pizza burrito Colomit Answer 	Prompt: ¿Que esta comiendo el animal?	Popular Answer: burrito	
pizza burnto Submit Answer Next Prompt	• sushi	• burrito	
Submit Answer	o pizza O burrito	• perro	
Next Prompt	Submit Answer		
Next Prompt			
Next Prompt			
			Next Prompt



Chat GPT entered the world in the middle of our project. Would we have designed our app differently if we had started one year later? (And is language learning still a meaningful subject?)

A large systematic review of the uses of AI in education showed that the most common use of AI in higher education is in the language classroom[10]. In education worldwide, AI is used for reading, writing, and vocabulary acquisition. It is also used for "content management" to offer more personalized learning options according to students' actual needs.

In a greatly expanded version of our app, if we wanted to continue its development, we could certainly feature AI generated stories that fit students' needs. The story would be tailored to the vocabulary and sentence structures each student had acquired and what they still needed to practice.

Next Steps

If we were to pursue Spanish Lab, here would be some of the most important things to carry out, from a development perspective:

- Increasing the number of story templates.
- Giving teachers the possibility of making their own story and adding multiple choice items of their choosing.
- Image generation for the multiple choice elements.
- Ability to save and share stories
- And of course, user testing every step of the way.

Looking at these a bit more closely, Here is what the development of each of these features would look like in more detail:

More story templates

If the teacher wanted a quick activity in class, they could just quickly choose a template and wouldn't need to have anything prepared beforehand.

Making and personalizing stories

There is a "Create Story" option in the menu. It would allow teachers to deviate from the templates and tailor the story to their lesson. It could pre-teach, reinforce, expand, or extend the main points of the lesson. At this point, this functionality isn't complete, we have just this pattern the teacher would follow to complete the story/questionnaire.

Image generation for the multiple choice elements.

As teachers typed in the options for multiple choice elements, they should have the option to add an image. It is important to have this choice because if it was a vocabulary item that students hadn't learned yet, this is a good way to introduce the word.

To further simplify things for teachers, a few images from Google image search could pop up as each multiple choice item was entered. This would make structuring the story a much simpler process for teachers, with the result that they would likely use it more frequently in their classroom.

Ability to save and share stories

One proven memory technique is spaced repetition[8]. The ability to save stories and share them would make the app stronger from a pedagogical viewpoint. Saving stories would give students a chance to review what they learned. If they also talked about it, that would serve

More testing

To be ready to deploy, this app would need more formal testing. We have only done informal beta testing within our group and with a larger group of people.

Learning Reflections

All three of us agreed that this process was an enormous learning process. However, the two meanings of "I learned a lot" apply: both '*it was a helpful experience*' AND 'I felt unprepared'.

None of us felt proficient enough in github to use it well. Of course we can pull and clone. But when it came to using it collaboratively, we all wished we had more practice, so we could revert/push/branch with confidence.

Working collaboratively was another skill we felt unpracticed in. Sure, we had

all done partner projects where each person works to their strengths. But none of us felt experienced in a truly collaborative project where so many decisions needed to be made. Perhaps this is due to our first couple of years being so deeply affected by COVID protocols, but all of us are more comfortable coding on our own and not having to integrate it with anyone else's code. Thus, both hard and soft collaboration skills felt unpracticed.

Additionally the web App classes we have taken did not go into nearly enough depth to build a full app such as this one.

Walkthrough

To begin, the teacher clicks *Enter Classroom* in the menu. Once the code is created, this allows others to enter into the same room. There is no special account for the teacher. The leader of this story is simply the one who created this unique code.



The students will then choose *Enter Classroom* from the hamburger menu and type in the classroom code. Here we can see the layout on both iPad and phone.

LOGIN		~ V	° p	return	? ·	, in the second
h		greens	8 9 i 0	k l	: ! m ;	.?123
Sppan	Join Room	green	5 6 7 t y u	g (, b n	
		"gree"	4 r		v	
Join a room	gree	5 ∂ 🖪	e 3	d s	- + = X C V	Ŷ

Voting

As we see below, there are 2 votes for *perro* and one vote for oso. On the

teachers page we see the votes as they arrive in real time, and the most popular

answer is shown.



More Voting

Once the teacher clicks on Next Prompt, everyone can move forward to the

next question.

$\leftarrow \rightarrow \mathbf{G}$	O A https://spanish-lab-client.vercel.app/#/room/green		☆ ♡ ৬ 0 원 ≡
Room: gree	۱	Spanjsh	LOGIN
Prompt: ¿Que hamburgues sushi pizza burrito	é está comiendo el animal?	Popular Answer: sushi All Answers: • pizza • sushi • sushi • sushi	Prompt: ¿Qué estă comiendo el animal? • tranburguesa • austri • pizza • bizza • bizza
	C A spariah-lab-client/vecet.apg/#/hoom/pree	n û 🛎 🗏 PRÊSK LOGN	Popular III O K
	● piza ● burnto ■ piza ● burnto ■ Popular Answer: sushi All Answer:		

Final Round

Room: green	parish	Room: green
Prompt: ¿Dónde está comiendo el animal? • en la playa • en la escuela • en el lago • en el lago • mortañas • en el lago • mortañas • en el lago • mortañas • en el ago • mortañas • en el ago • mortañas • en el ago • mortañas • en el ago • mortañas • en el ago	Popular Answer: en las montañas All Answers: • en las montañas animal? *animal?	Prompt: 2Dónde está comiendo el animal? • en la playa • en la socuela • en el lago Popular Motivar en II • • • • • • • • • • • • • • • • • •

Image Generation

And the students' choices generate a unique image from openai, which is

displayed to the screen!



Bibliography

- 1. D'Argenio, Linda. "What Is TPRS? A Look at the Language Acquisition Strategy That's Growing in Popularity." *BridgeUniverse* – TEFL Blog, News, Tips & Resources, 28 Nov. 2022, <u>https://bridge.edu/tefl/blog/tprs/</u>.
- 2. "Design Patterns and Refactoring." SourceMaking, https://sourcemaking.com/design_patterns/singleton.
- 3. "Duolingo: Language Lessons Apps on Google Play." Google, Google, <u>https://play.google.com/store/apps/details?id=com.duolingo&gl=US</u>.
- 4. Liam Printer (2021) Student perceptions on the motivational pull of Teaching Proficiency through Reading and Storytelling (TPRS): a self-determination theory perspective, The Language Learning Journal, 49:3, 288-301, DOI: 10.1080/09571736.2019.1566397

- 5. "NAE Grand Challenges for Engineers." Grand Challenges 14 Grand Challenges for Engineering, National Academy of Engineering, http://www.engineeringchallenges.org/challenges.aspx.
- Singla, L. (2022, September 16). What's a software design pattern? (+7 most popular patterns). Insights Web and Mobile Development Services and Solutions. Retrieved November 10, 2022, from https://www.netsolutions.com/insights/software-design-pattern/#1-creational-design-patterns
- 7. Thompson, Sonia. "The U.S. Has the Second-Largest Population of Spanish Speakers-How to Equip Your Brand to Serve Them." Forbes, Forbes Magazine, 9 Nov. 2022, <u>https://www.forbes.com/sites/soniathompson/2021/05/27/the-us-has-the-second-largest-population-of-spanish-speakers-how-to-equip-your-brand-to-serve-them/?sh=568237793aee.</u>

Additional references:

- 8. G Branwen, Spaced repetition. Retrieved April 29, 2023 from <u>https://www.gwern.net/Spaced%20repetition</u>. (2016).
- 9. Introduction. SocketIO RSS. (2023, March 27). Retrieved April 29, 2023, from <u>https://socket.io/docs/v3</u>.
- Crompton, Helen, and Diane Burke. "Artificial Intelligence in Higher Education: The State of the Field." International Journal of Educational Technology in Higher Education 20, no. 1 (2023): 22–. <u>https://doi.org/10.1186/s41239-023-00392-8</u>.
- 11. Bunch, Robert. (2023, April). Socketio (with websockets) the details. updated April 2023. Udemy. Retrieved April 29, 2023, from https://www.udemy.com/course/socketio-with-websockets-the-details/

Appendix

- 1. <u>spanish-lab-client/public/index.html</u> **U**
- 2. <u>SpanishLab/client/src/App.css</u>
- 3. SpanishLab/client/src/pages/JoinRoom.js ひ
- 4. SpanishLab/client/src/Room.js
- 5. <u>SpanishLab/client/src/generateImage.js</u>
- 6. <u>SpanishLab/client/src/index.js</u>
- 7. SpanishLab/client/src/openai-api.js
- 8. <u>spanish-lab-server/server.js</u>
- 9. SpanishLab/server/controllers/openaiControllers.js ひ
- 10. SpanishLab/server/routes/openaiRoutes.js 신

spanish-lab-client/public/index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
 <meta name="viewport" content="width=device-width, initial-scale=1" />
 <meta name="theme-color" content="#000000" />
  <meta
      name="SpanishLab: A fill in the blank style story to help you learn Spanish."
  />
 k rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
 <!--
      manifest.json provides metadata used when your web app is installed on a
      user's mobile device or desktop. See
https://developers.google.com/web/fundamentals/web-app-manifest/
  -->
 <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
 <!--
      Notice the use of %PUBLIC_URL% in the tags above.
      It will be replaced with the URL of the `public` folder during the build.
      Only files inside the `public` folder can be referenced from the HTML.
      Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
```

```
work correctly both with client-side routing and a non-root public URL.
      Learn how to configure a non-root public URL by running `npm run build`.
  -->
 <title>SpanishLab</title>
</head>
<body>
<noscript>You need to enable JavaScript to run this app.</noscript>
<div id="root"></div>
<!--
 This HTML file is a template.
 If you open it directly in the browser, you will see an empty page.
 You can add webfonts, meta tags, or analytics to this file.
 The build step will place the bundled scripts into the <body> tag.
 To begin the development, run `npm start` or `yarn start`.
 To create a production bundle, use `npm run build` or `yarn build`.
-->
</body>
</html>
```

```
SpanishLab/client/src/App.css (back to appendix)
```

```
.App {
  text-align: center;
}
.App-logo {
  height: 40vmin;
  pointer-events: none;
}
@media (prefers-reduced-motion: no-preference) {
  .App-logo {
      animation: App-logo-spin infinite 20s linear;
  }
}
.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
```

```
color: white;
}
.App-link {
  color: #61dafb;
}
@keyframes App-logo-spin {
   from {
     transform: rotate(0deg);
   }
   to {
       transform: rotate(360deg);
   }
}
```

SpanishLab/client/src/pages/JoinRoom.js

(back to appendix)

```
import React, { useState } from 'react';
import { useNavigate } from 'react-router-dom';
function JoinRoom() {
 const [roomId, setRoomId] = useState('');
 let navigate = useNavigate();
 const handleSubmit = (e) => {
      e.preventDefault();
      navigate(`/room/${roomId}`);
 };
  return (
      <div className="standardPage">
      <h1>Join a room</h1>
      <form className ="displayBlock" onSubmit={handleSubmit}>
      <input
      type="text"
      placeholder="Room ID"
      value={roomId}
      onChange={(e) => setRoomId(e.target.value)}
      required
      />
      <button className="primaryButton" type="submit">Join Room</button>
      </form>
      </div>
 );
}
```

export default JoinRoom;

```
SpanishLab/client/src/Room.js
                                         (back to appendix)
import React, { useState, useEffect } from 'react';
import { useParams } from 'react-router-dom';
import io from 'socket.io-client';
import ImageComponent from './generateImage';
const socket = io('http://localhost:3001');
const popularAnswers = [];
function Room() {
      const { roomId } = useParams();
const [answer, setAnswer] = useState('');
      const [answers, setAnswers] = useState([]);
      const [isTeacher, setIsTeacher] = useState(false);
      const [promptIndex, setPromptIndex] = useState(0);
      const [selectedAnswer, setSelectedAnswer] = useState('');
      const [hasAnswered, setHasAnswered] = useState(false);
      const [showImage, setShowImage] = useState(false);
      const [finalPrompt, setFinalPrompt] = useState('');
      const Questions = [
      { "question": "What animal is our story about?", "options": ["Dog", "Cat",
"Lion", "Bear"] },
       { "question": "What is the animal eating?", "options": ["Hamburger", "Sushi",
"Pizza", "Burrito"] },
      { "question": "Where is the animal eating?", "options": ["On the beach", "At
School", "In The Mountains", "At the Lake"] },
var isDone = false;
      var displayEnd = (promptIndex+1 === Questions.length) ? true : false;
      useEffect(() => {
      socket.emit('joinRoom', roomId);
      socket.on('receiveAnswer', (receivedAnswer) => {
             setAnswers((prevAnswers) => [...prevAnswers, receivedAnswer]);
      });
      socket.on('teacher', () => {
             setIsTeacher(true);
      });
      socket.on('nextPrompt', () => {
```

```
setPromptIndex((prevIndex) => prevIndex + 1);
             setAnswers([]);
             setHasAnswered(false);
      });
      return () => {
             socket.off('receiveAnswer');
             socket.off('teacher');
             socket.off('nextPrompt');
      };
      }, [roomId]);
      const handleNextPrompt = () => {
      popularAnswers.push(popularAnswer());
      console.log(promptIndex + " " + Questions.length)
      if(promptIndex+1 === Questions.length) {
             isDone = true;
      } else {
             socket.emit('nextPrompt', roomId);
      }
      };
      const handleEndQuiz = () => {
      popularAnswers.push(popularAnswer());
      setFinalPrompt(`A cartoon ${popularAnswers[0]} eating a ${popularAnswers[1]}
${popularAnswers[2]}`);
      setShowImage(true);
      }
      const handleSubmit = (e) => {
      e.preventDefault();
      socket.emit('newAnswer', { roomId, answer: selectedAnswer });
      setSelectedAnswer('');
      setHasAnswered(true);
      };
      const popularAnswer = () => {
      const counts = {};
      let maxCount = 0;
      let maxValue = null;
      answers.forEach((a) => {
             counts[a] = (counts[a] || 0) + 1;
             if (counts[a] > maxCount) {
             maxCount = counts[a];
             maxValue = a;
             }
```

```
});
return maxValue;
};
return (
<div>
{showImage ? <ImageComponent prompt={finalPrompt} /> :
<div>
      <h1>Room: {roomId}</h1>
      <h2>Prompt: {Questions[promptIndex].question}</h2>
      <form onSubmit={handleSubmit}>
       {Questions[promptIndex].options.map((choice, index) => (
             <div key={index}>
                    <input
                    key={`prompt-${promptIndex}-choice-${index}`}
                    type="radio"
                    id={`choice-${index}`}
                    name="answer"
                    value={choice}
                    onChange={(e) => setSelectedAnswer(e.target.value)}
                    required
                    />
                    <label htmlFor={`choice-${index}`}>{choice}</label>
             </div>
      ))}
      <button type="submit" disabled={hasAnswered}>Submit Answer</button>
      </form>
      {isTeacher && (
      <div>
      {displayEnd ?
             <button type="button" onClick={handleEndQuiz}>
                    End Quiz
             </button> :
             <button type="button" onClick={handleNextPrompt}>
                    Next Prompt
             </button>
      }
      </div>
      )}
      <h2>Popular Answer: {popularAnswer()}</h2>
      <h3>All Answers:</h3>
      {answers.map((answer, index) => (
             key={index}>{answer}
      ))}
```

```
</div>
}
</div>
);
}
```

```
export default Room;
```

```
SpanishLab/client/src/generateImage.js (back to appendix)
import React, { useState } from 'react';
import { generateImage } from './openai-api';
function ImageComponent({ prompt }) {
  const [imageUrl, setImageUrl] = useState('');
  const [isLoading, setIsLoading] = useState(false);
  const fetchImage = async () => {
      setIsLoading(true);
      const imageData = await generateImage(prompt);
      if (imageData && imageData.data) {
      setImageUrl(imageData.data);
      } else {
      console.error('Failed to generate image');
      }
      setIsLoading(false);
  };
  return (
      <div>
      <h3>{prompt}</h3>
      {isLoading && Loading...}
      {imageUrl && <img src={imageUrl} alt="Generated Image From Prompt"/>}
      <br/>
      <button onClick={fetchImage}>Generate Image</button>
      </div>
 );
}
export default ImageComponent;
```

```
SpanishLab/client/src/index.js (back to appendix)
```

SpanishLab/client/src/openai-api.js (back to appendix)

import axios from 'axios'; const API_URL = 'http://localhost:3001'; // Replace with your server URL in production const api = axios.create({ baseURL: API_URL, }); export const generateImage = async (prompt) => { try { const response = await api.post('/openai/generate-image', { prompt: prompt, }); console.log(prompt); console.log(response.data); return response.data; } catch (error) { console.error('Error generating image:', error); return null; } };

```
SpanishLab/server/controllers/openaiControllers.js
const { Configuration, OpenAIApi } = require("openai")
```

```
const configuration = new Configuration({
      apiKey: process.env.OPENAI_API_KEY,
});
const openai = new OpenAIApi(configuration);
const generateImage = async (req, res) => {
      console.log(req.body);
      const { prompt } = req.body;
      try {
      const response = await openai.createImage({
             prompt: prompt,
             // n: 1, n = 1 is default
             size: "512x512",
      });
      const imageUrl = response.data.data[0].url
      res.status(200).json({
             success: true,
             data: imageUrl
      });
      } catch (error) {
      if (error.response) {
             console.log(error.response.status);
             console.log(error.response.data);
      } else {
             console.log(error.message);
      }
      res.status(400).json({
             success: false,
             error: 'The image could not be generated'
      });
      }
};
module.exports = { generateImage };
```

```
spanish-lab-server/routes/openaiRoutes.js (back to appendix)
const express = require('express');
const { generateImage } = require('../controllers/openaiControllers');
const router = express.Router();
```

```
router.post('/generate-image', generateImage);
module.exports = router;
spanish-lab-server/server.js (back to appendix)
const express = require('express');
const app = express();
const { createServer } = require('http');
// Enable CORS for all routes
app.use(cors(corsOptions));
app.use(express.json());
app.use(express.static('public'));
const io = new Server(httpServer, {
      cors: {
      origin: ['https://spanish-lab-client.vercel.app', 'http://localhost:3000'],
       }
});
      if (isNewRoom) {
             room.teacher = socket.id;
             socket.emit('teacher');
       }
       room.users.add(socket.id);
       rooms.set(roomId, room);
       socket.join(roomId);
       console.log(`User (${socket.id}) joined room: ${roomId}`);
       console.log(`Teacher: ${room.teacher}`);
      });
       socket.on('newAnswer', (data) => {
       const room = rooms.get(data.roomId);
      if (room) {
      console.log(data.answer);
             room.answers.push(data.answer);
             io.to(data.roomId).emit('receiveAnswer', data.answer);
       }
       });
       socket.on('nextPrompt', (roomId) => {
```

```
const room = rooms.get(roomId);
      if (room && socket.id === room.teacher) {
             room.answers = [];
             io.to(roomId).emit('nextPrompt');
      }
      });
      socket.on('disconnect', () => {
      console.log('user disconnected');
      for (const [roomId, room] of rooms.entries()) {
             room.users.delete(socket.id);
             if (room.teacher === socket.id) {
             room.teacher = room.users.values().next().value;
             }
             if (room.users.size === 0) {
             rooms.delete(roomId);
             }
      }
      });
});
const PORT = process.env.PORT || 3001;
httpServer.listen(PORT, () => {
      console.log(`listening on *:${PORT}`);
});
```