



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

APLICACIÓN WEB DIDÁCTICA PARA FACILITAR A NIÑOS DE COLEGIO
A MEJORAR SU FLUIDEZ EN UN IDIOMA INTERACTUADO CON NIÑOS
DE OTROS PAÍSES 

AUTOR

Carlos Manuel Paz Jáuregui

AÑO

2017



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

APLICACIÓN WEB DIDÁCTICA PARA FACILITAR A NIÑOS DE COLEGIO
A MEJORAR SU FLUIDEZ EN UN IDIOMA INTERACTUADO CON NIÑOS
DE OTROS PAÍSES.

Trabajo de Titulación presentado en conformidad a los requisitos
establecidos para optar por el título de Ingeniero en Sistemas de
Computación e Informática

Profesor Guía

Ms. Juan José León Guerrero

Autor

Carlos Manuel Paz Jáuregui

Año

2017

DECLARACIÓN DEL PROFESOR GUÍA

— Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.

Juan José León Guerrero

Master Especializado en Ingeniería de Medios Numéricos

CI: 1707506760

DECLARACIÓN DEL PROFESOR CORRECTOR

— Declaro haber revisado este trabajo, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.

Tannia Jacqueline Álava Freire

Magister Administración Tecnológica

CI: 1706299169

DECLARACIÓN DE AUTORÍA DE EL ESTUDIANTE

—Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.

Carlos Manuel Paz Jáuregui

CI: 1716072127

AGRADECIMIENTO

Agradezco a todas las personas que han sido parte de mi desarrollo profesional, y a mis papás y hermanos que siempre me han dado un buen ejemplo.

Carlos Paz

RESUMEN

El proyecto incluye la explicación del desarrollo de un prototipo de aplicación web enfocada en alumnos desde 8º grado de Educación general básica hasta 3º curso de Bachillerato que busquen mejorar su comunicación oral en los idiomas inglés y español, la aplicación se encargará en dar herramientas para interactuar e intercambiar conocimientos de los idiomas. La aplicación utiliza un método educativo nocional-funcional para la enseñanza de idiomas. Se aplican técnicas y conceptos de gamificación para otorgar puntajes y evaluar el avance.

Contiene el proceso de desarrollo mediante una metodología de desarrollo ágil Scrum. Se utilizan librerías y frameworks Javascript como Node.js, React.js y Peer.js. Se hace uso de los servicios cloud de Firebase para manejar autenticación y base de datos en tiempo real que se acceden mediante llamadas al API.

Incluye el desarrollo de una plataforma de comunicación RTC, con videollamadas y envío de mensajes escritos utilizando PeerJS a través de las características WebRTC de los browsers.

ABSTRACT

This project includes the explanation of the development of a prototype of a web application focused on students from 8^o grade of Basic general education up to 3^o course of Baccalaureate who seek to improve their oral communication in English and Spanish, the application will be responsible for giving tools to interact and exchange knowledge of languages. The application uses a notional-functional educational method for language teaching. Gamification techniques and concepts are used to award scores and evaluate progress.

It contains the development process through an agile Scrum development methodology. Javascript libraries and frameworks such as Node.js, React.js and Peer.js are used. Firebase cloud services are used to handle authentication and real-time database access via API calls.

It includes the development of an RTC communication platform, with video calls and sending written messages using PeerJS through the WebRTC features of the browsers.

ÍNDICE

1. Capítulo I Introducción	1
1.1. Antecedentes	1
1.2. Alcance	2
1.3. Justificación.....	4
1.4. Objetivo General	4
1.5. Objetivos específicos.....	5
2. Capítulo II Estado del Arte	5
2.1. Metodologías utilizadas para el desarrollo.....	5
2.1.1. Scrum.....	5
2.1.2. Wireframes	6
2.1.3. Método nocional-funcional para la enseñanza de idiomas	7
2.1.4. Gamificación.....	8
2.2. Herramientas de Desarrollo.....	9
2.2.1. Node JS.....	9
2.2.2. Express JS	10
2.2.3. React JS.....	10
2.2.4. Peer JS.....	11
2.2.5. Firebase	12
3. Capítulo III Desarrollo del Proyecto	13
3.2. Diagrama de Clases	14
3.3. Diagrama de actividades	15
3.4. Casos de Uso	15
3.4.1. Descripción de Actores.....	16
3.5. Scrum	17
3.5.1. Sprint 1	19
3.5.1.1. Wireframes de aplicación.....	19
3.5.2. Sprint 2.....	21
3.5.2.1. Control de Versiones.....	21
3.5.2.2. Configuración base de la instalación	21

3.5.3. Sprint 3.....	24
3.5.3.1. Configuración Firebase	24
3.5.4. Sprint 4.....	27
3.5.4.1. Creación Interfaz Gráfica	27
3.5.4.2. Creación de Autenticación	29
3.5.5. Sprint 5.....	34
3.5.5.1. Configuración Peerjs.....	34
3.5.6. Sprint 6.....	35
3.5.6.1. Crear comunicación RTC	35
3.5.7. Sprint 7	37
3.5.7.1. Implementar Gamificación.....	37
3.5.8. Sprint 8.....	38
3.5.8.1. Crear Actividades.....	38
3.5.9. Sprint 9.....	40
3.5.9.1. Crear Formularios	40
4. Conclusiones y Recomendaciones	42
4.1. Conclusiones	42
4.2. Recomendaciones	43
Referencias.....	44
ANEXOS	47

1. Capítulo I Introducción

1.1. Antecedentes

En América Latina se tiene en promedio un nivel bajo en el nivel de aptitud de habla de inglés, existe un pequeño crecimiento en la mejora del idioma en algunos países latinoamericanos que se puede mejorar con los programas realizados por los gobiernos según dice el estudio del Índice del Nivel de Inglés EF “En general, el nivel de inglés en Latinoamérica es bajo y hay mucho lugar para mejorar. Casi todos los países de la región han lanzado programas ambiciosos para aumentar o mejorar la educación del idioma inglés. “(EF English Proficiency Index, 2017).

En un mundo cada vez más globalizado es necesario ver más allá de las fronteras y unirnos con gente de otros países, la mejor forma de romper esa barrera de la distancia es a través de la tecnología que nos permite comunicarnos en tiempo real con ayuda de videollamadas. El avance tecnológico nos permite utilizar nuevas herramientas de desarrollo que son más intuitivas y tienen un mejor desempeño. La velocidad de transferencia de Internet actual nos permite realizar videollamadas sin esfuerzo y con buena calidad.

Aprender un segundo idioma es esencial en la educación hoy en día, es una herramienta importante para el desarrollo de las personas y abre muchas puertas y oportunidades. En muchas ocasiones los idiomas son considerados como una carrera y siempre son de gran ayuda en el desarrollo profesional.

Actualmente la política de educación de los países incluye la enseñanza de un idioma adicional, y la mayoría de instituciones educativas ofrece cursos de idiomas. En Ecuador en el año 2014 con el acuerdo ministerial No. 052-14,

incluido en el Anexo 2, el Ministro de Educación dispuso: “la enseñanza de inglés, a partir del año lectivo 2016-2017, régimen Sierra y 2017-2018, régimen Costa sea obligatoria desde segundo grado de Educación General Básica hasta tercer curso de Bachillerato para todas las instituciones públicas, fiscomisionales y particulares del país.”(Freddy Peñafiel Larrea, p.3).

La metodología de enseñanza involucra libros que sirven para estudiar áreas como escritura y lectura. La mayor debilidad del aprendizaje de un idioma es la práctica, cuando una persona tiene la oportunidad de aprender el idioma en un entorno nativo, el aprendizaje es mucho más rápido.

Una técnica importante en las aplicaciones de educación es la gamificación que consiste en dar puntos y niveles por realizar ciertas actividades, esto motiva a los estudiantes a realizar actividades mezclando videojuegos con aprendizaje. Como dice Daniel Parente (2016, p.11) “En una sociedad digitalmente conectada y con acceso continuo a la información, capaz de sintetizar nuevos conocimientos, todo lo que no esté enfocado en mejorar su experiencia de uso resulta una realidad monótona y aburrida.”

1.2. Alcance

El alcance de este proyecto es el de crear un prototipo de aplicación web para alumnos desde 8º grado de educación general básica hasta 3º curso de Bachillerato en la que se podrá realizar una videollamada uno a uno con otros estudiantes situados en diferentes posiciones geográficas. Al momento de realizar las videollamadas los estudiantes realizarán diferentes actividades y juegos lúdicos que se deberán cumplir cooperativamente. Las actividades serán realizadas en dos idiomas, específicamente en inglés y español, de manera que los dos alumnos intercambien su conocimiento y aprendan los dos idiomas consecutivamente.

La metodología para el desarrollo que se utilizará es una metodología ágil Scrum, en la que se utilizarán Sprints para dividir el trabajo de desarrollo en intervalos de tiempo.

Las herramientas tecnológicas que se utilizarán para el desarrollo de la aplicación serán Node.js y Express.js del lado del servidor o back-end. React.js para manejar la interfaz de usuario o front-end. Peer.js para implementar la comunicación RTC (Real Time Communication) entre los navegadores con la que se obtendrán las videollamadas y la mensajería. Firebase con su servicio de autenticación y base de datos en tiempo real que funciona como DBAAS (Base de Datos como servicio).

Se planea utilizar una metodología de enseñanza de idiomas enfocada en un método nocional-funcional. Éste método se basa en nociones que se refieren a situaciones de la vida real como ir de compras, ir a una fiesta, etc. Y las funciones que dependen de la noción como saludar, preguntar precios, etc. El método nocional-funcional tiene una extensión con el método comunicativo que utiliza varias actividades como juegos, intercambio de idiomas, trabajo en parejas y juegos de simulación o *role playing* que serán útiles para el aprendizaje comunicativo de los estudiantes.

Se incluirán conceptos y elementos de gamificación en el desarrollo de la aplicación como los elementos PBL (*Points, Badges, Leaderboards*), de tal manera que los alumnos obtengan puntajes y medallas por realizar actividades y luego puedan tener acceso a un reporte de sus resultados.

1.3. Justificación

De acuerdo al ministerio de educación ecuatoriano se están tomando medidas para impulsar la enseñanza de idiomas con un curriculum de lengua extranjera realizado por el Ministerio de Educación en donde se tiene como objetivos principales los siguientes puntos:

- “Desarrollar la comprensión que los estudiantes tienen del mundo, de otras culturas y de la suya propia y su capacidad de comunicar sus puntos de vista a través de la Lengua extranjera.
- Desarrollar las habilidades personales, sociales e intelectuales necesarias para alcanzar su potencial y participar productivamente en un mundo cada vez más globalizado que opera en otras lenguas.” (Ministerio de Educación Ecuador, 2016).

De acuerdo a las medidas que está tomando el Ministerio de Educación ecuatoriano para fomentar la enseñanza de las lenguas extranjeras con el acuerdo ministerial No. 052-14 adjunto en el Anexo 2, pretendo apoyar a ésta iniciativa con una herramienta didáctica complementaria a la enseñanza actual de los colegios que se enfoca en desarrollar las habilidades sociales y comunicativas de los alumnos desde 8º grado de educación general básica hasta 3º curso de bachillerato.

1.4. Objetivo General

Desarrollar un prototipo de aplicación web didáctica para alumnos desde 8º grado de educación general básica hasta 3º curso de bachillerato utilizando una metodología nocional-funcional conjuntamente con la gamificación, que permita interactuar a través de actividades cooperativas en una sesión uno a uno desde cualquier parte del mundo.

1.5. Objetivos específicos

- Aplicar metodologías de desarrollo ágiles en el proceso de desarrollo de la aplicación
- Desarrollar un prototipo de plataforma de comunicación RTC que permita realizar videollamadas y enviar mensajes escritos.
- Desarrollar cuestionarios y registrar tiempo de uso para evaluar el avance en la plataforma.
- Generar reportes de puntajes y tiempo de uso

2. Capítulo II Estado del Arte

2.1. Metodologías utilizadas para el desarrollo

2.1.1. Scrum

Es una metodología ágil utilizada en el desarrollo de software que provee de conceptos y herramientas para facilitar el proceso de desarrollo. En ésta metodología pueden intervenir varias personas:

- **Product Owner:** es el encargado de escribir historias de usuario y encargarse del trabajo del equipo.
- **Scrum Master:** es el encargado de que se cumplan las reglas del proceso Scrum
- **Equipo Scrum:** son todas las personas encargadas de entregar un producto.

Parte importante de ésta metodología son las iteraciones de trabajo, se utiliza el término Sprint para definir el período de tiempo en el que se deben realizar

ciertas tareas, las tareas serán distribuidas entre los diferentes sprints. (Kalk, S., 2017).

En un proceso de Scrum hay diferentes tipos de reuniones que sirven para mantener al equipo organizado y sin bloqueos. Se deben definir las tareas a realizarse en una reunión al empezar cada Sprint que tiene el nombre de Sprint Planning. Se sugiere tener una reunión diaria para encontrar actividades bloqueadas y verificar avances en un Stand-up meeting. (Kalk, S., 2017).

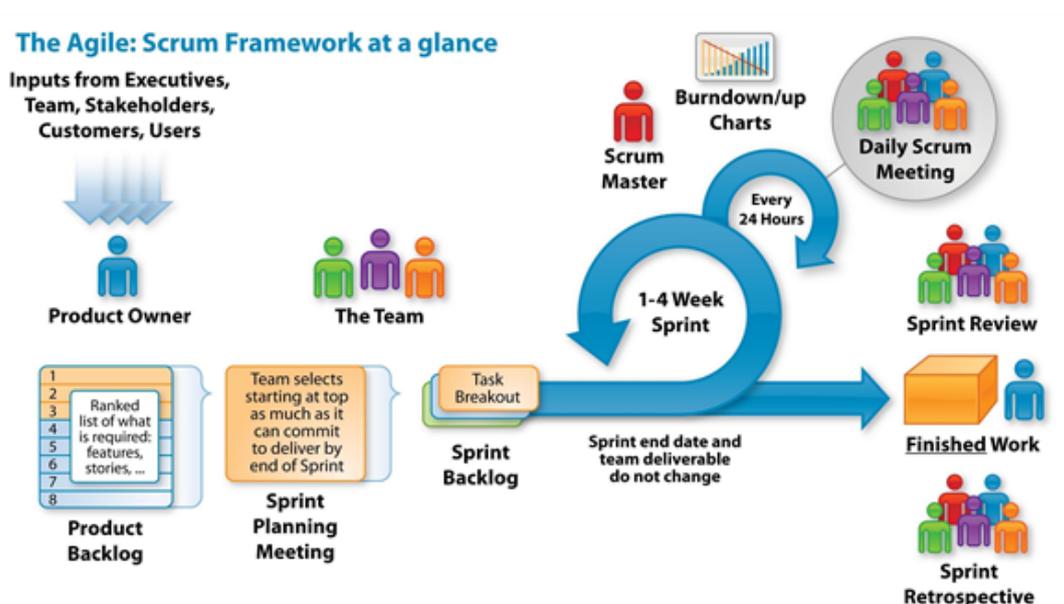


Figura 1. Flujo de Scrum con Roles, reuniones y etapas de la metodología. Tomado de Ness-Tech (2017).

2.1.2. Wireframes

El proceso de Wireframes consiste en construir un ejemplo de visualización de la página o aplicación web para poder determinar la estructura más conveniente antes de empezar la etapa de desarrollo. (Buckner, 2014)

Los wireframes buscan ordenar la información que se quiere representar, con todos los elementos necesarios incluyendo navegación, botones, imágenes, etc. Y al final se obtiene un plano o esquema que servirá de referencia para los desarrolladores y de donde pueden entender el funcionamiento y estructura de la aplicación web. (Terrera, G., 2017)

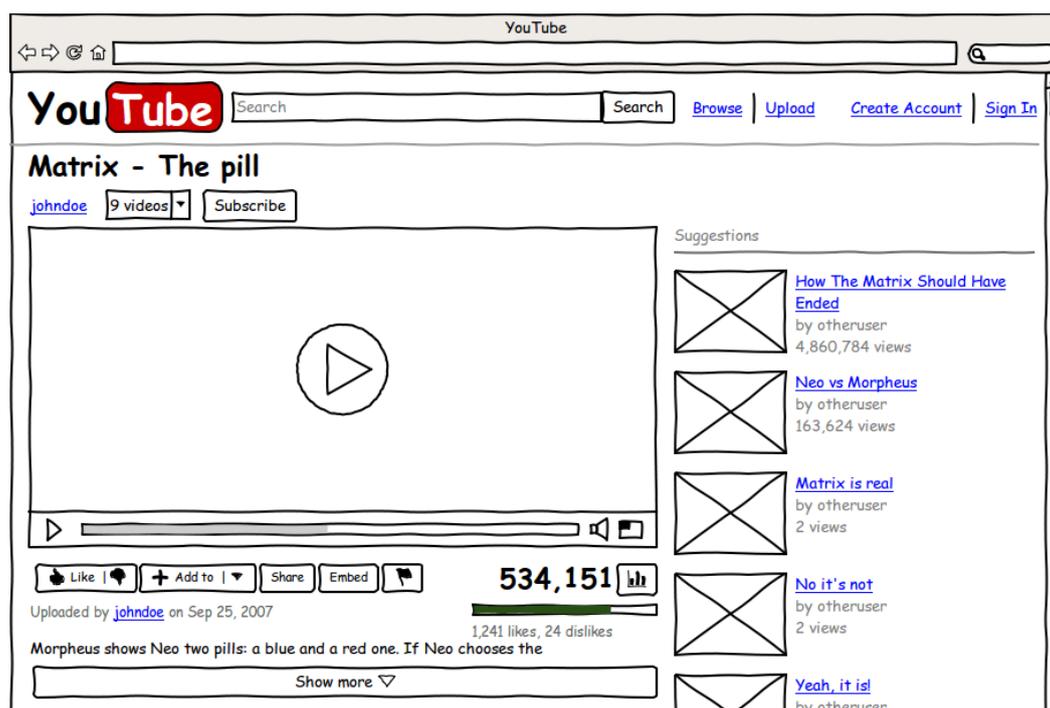


Figura 2. Ejemplo de Wireframe con la página de Youtube

Tomado de Testing Baires (2016)

2.1.3. Método nocional-funcional para la enseñanza de idiomas

El método nocional-funcional o también conocido como método comunicativo se refiere a un tipo de enseñanza de idiomas en el que se pone énfasis a categorías nocionales y funcionales. La categoría nocional tiene que ver con conceptos nocionales que son comunicados a través de la lengua como cantidades, tiempo y frecuencia, en cambio la categoría funcional se relaciona con los componentes pragmáticos de la lengua por ejemplo preguntar, presentarse y solicitar información. (CVC, 2017).

El método se basa en tres conceptos básicos para permitir una comunicación real:

- Vacío de información. Cada alumno necesita averiguar algo de su compañero, solo así podrá lograr cumplir con la tarea.
- Libertad de expresión. Cuando el alumno decide qué va a decir, cómo lo va a decir, en qué momento, etc.
- Retroalimentación. Los alumnos comprenden su avance mediante las reacciones verbales y no verbales de sus compañeros. (Gancedo, 2016, p.6).

Un buen ejemplo de metodología comunicativa son juegos de rol en los que los alumnos toman un papel para representar, de esta manera se pueden desenvolver en un ambiente diferente y a la vez cumplir con los conceptos básicos del método comunicativo. (Gancedo, 2016, p.12).

2.1.4. Gamificación

Gamificación es un término que se refiere a aplicar mecánicas de juegos a tareas de la vida cotidiana como estudiar. De esta manera se busca ludificar actividades no recreativas que se pueden considerar aburridas por los jóvenes. La gamificación crea estímulos y recompensas en las personas para mantenerlas motivadas utilizando elementos PBL que se traduce de sus siglas en inglés a Puntos, Medallas y Tabla de clasificación. (Gamificación educativa, 2017)

Como dice Freddy Valda Sanchez y Carlos Arteaga Rivero en su artículo sobre “la implementación de una estrategia de gamificación en una plataforma virtual de educación”, existen tres conceptos clave para la gamificación:

- “Dinámicas de juego: Corresponden a las motivaciones internas que tiene el ser humano para jugar, entre ellas se encuentran las emociones, la narrativa, el sentido de progreso, el reconocimiento, la recompensa, la cooperación, el altruismo, la competencia. Las dinámicas del juego responden a la pregunta ¿Por qué el ser humano desea participar de una actividad lúdica?
- Mecánicas de juego: Reglas y retos que propone el juego. Se pueden mencionar entre ellas la recolección de objetos, las comparativas y clasificaciones, los niveles, las respuestas (feedback), los premios, las transacciones, los turnos, los estados de juego de victoria, entre otros.
- Componentes de juego: Elementos base que dan estructura a un juego, podemos mencionar algunos elementos que son usados en distintos tipos de juegos: los logros, los avatares, las insignias, las luchas contra jefes, colecciones, combate, desbloqueo de contenidos, regalos, tablas de líderes, niveles, puntos, conquistas, gráficas sociales, equipos y mercancías virtuales.”(Freddy Valda Sanchez y Carlos Arteaga Rivero, 2015)

2.2. Herramientas de Desarrollo

2.2.1. Node JS

Es una plataforma que permite a los desarrolladores correr código Javascript del lado del servidor. Está construido en base al motor Javascript V8 de Google Chrome. (NodeJS, 2017)

Node.js contiene una librería de módulos básicos y de terceros que extienden la funcionalidad y capacidad de un servidor node, por ejemplo el módulo básico http le permite actuar como un servidor http, el módulo stream para manejar flujos de datos, etc. Los módulos de terceros necesitan ser instalados, para esto utilizamos NPM (Node package manager) que nos ayuda instalando, compilando

y actualizando los módulos, además maneja las dependencias (Kiesling M., 2012)

2.2.2. Express JS

“Express es una infraestructura de aplicaciones web Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles.”(Express, 2017). Es el marco de referencias estándar de servidor para Node.js.

Permite a las aplicaciones organizarlas en una arquitectura MVC del lado del servidor. Ofrece servicios para facilitar enrutamiento y llamadas a Middleware. (Express, 2017)

2.2.3. React JS

React es una librería escrita en JavaScript, facilita a los desarrolladores con la creación de componentes interactivos y a la vez reutilizables utilizados en la interfaz gráfica del usuario. Es desarrollada en Facebook y es ampliamente utilizado en aplicaciones como Instagram. Ésta librería no sólo es eficiente en el lado del cliente, se puede trabajar conjuntamente en el lado del servidor también. (React, 2017)

El principio de React se basa en la actualización del DOM cada vez que existe un cambio en la información de la página. Cuando realiza una actualización no recarga totalmente la página porque gastaría recursos y tiempo renderizando completamente la página. En cambio utiliza un término conocido como el DOM virtual que es una representación virtual de la página. Cuando se realiza un cambio en la información de la página el DOM virtual verifica mediante un algoritmo que partes de la página cambiaron y luego actualiza el DOM en base

a las diferencias logrando un cambio mínimo en el DOM para reflejar los cambios realizados. (React, 2017)

2.2.4. Peer JS

PeerJs es una librería Javascript que facilita la conexión entre usuarios mediante el uso de una colección de protocolos de comunicación llamado WebRTC que habilita a los navegadores con capacidades de comunicación en tiempo real (Real-Time Communications) mediante APIs Javascript. (PeerJS, 2017)

WebRTC permite a las aplicaciones del navegador realizar llamadas de voz, chat de vídeo y uso compartido de archivos P2P sin plugins. Y actualmente está siendo desarrollada por el W3C (World Wide Web Consortium). (WebRTC, 2016).

Peerjs provee de un API para conexiones peer-to-peer para comunicar a dos usuarios y realizar envíos de cualquier tipo de flujo de información como texto, imágenes, archivos, audio y video. (PeerJS, 2017)

WebRTC utiliza principalmente tres componentes que son getUserMedia para utilizar el micrófono y la cámara web, PeerConnection para establecer conexiones de audio y video, y DataChannels con el que se puede compartir información a través de P2P. (WebRTC, 2016).

PeerJs necesita de un servidor Peer que funciona como corredor (broker) y realizar la conexión entre los usuarios. Solamente se necesita de un ID de cada usuario para realizar una conexión P2P. La información que se transfiere entre usuarios no pasa por el servidor, la información se transfiere de usuario a usuario mediante el navegador. (PeerJS, 2017)

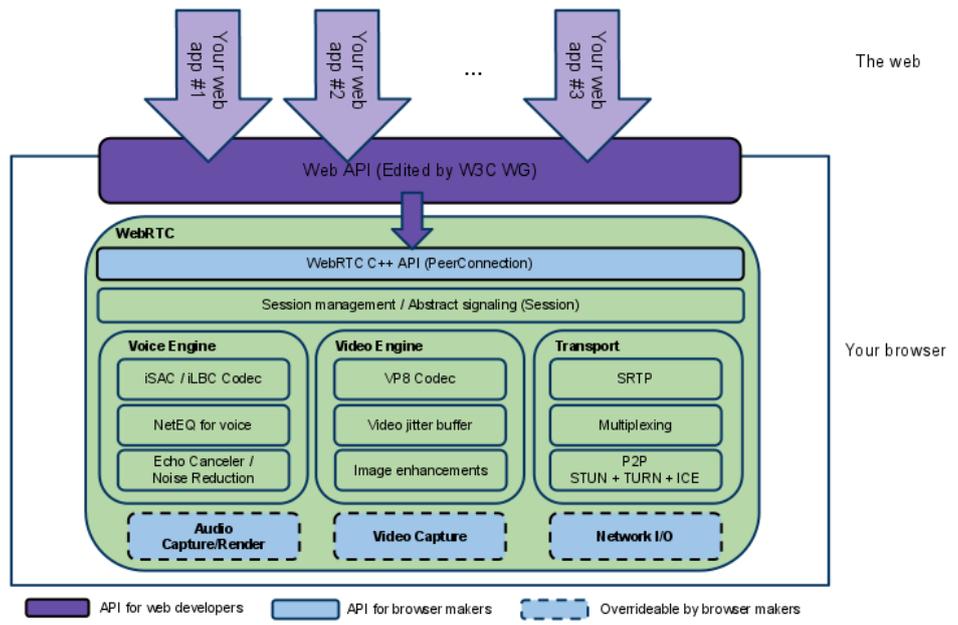


Figura 3. Arquitectura del componente WebRTC en los navegadores.

Tomado de WebRTC (2016)

2.2.5. Firebase

Firebase es un conjunto de herramienta para el desarrollo de aplicaciones web y móviles, facilita el proceso de desarrollo con herramientas como Base de datos en tiempo real, almacenamiento, autenticación e incluso información analítica de la aplicación. Éste servicio está bajo el desarrollo y la infraestructura de Google y es utilizada por unas de las aplicaciones más grandes como Shazam, Duolingo, The New York Times, The Economist. (Firebase, 2017)

La base de datos de tiempo real de Firebase funciona como una base de datos noSQL, ésta almacena objetos JSON. A diferencia de base de datos SQL no utiliza tablas o registros, la información se almacena en nodos con la estructura de JSON con un identificador asociado. (Tamada, 2017)

La autenticación con Firebase permite a los usuarios que se registren e inicien una sesión de manera fácil y mediante diferentes formas como con cuentas de Facebook, Google, Twitter, Github, o con un Email. (Firebase, 2017)

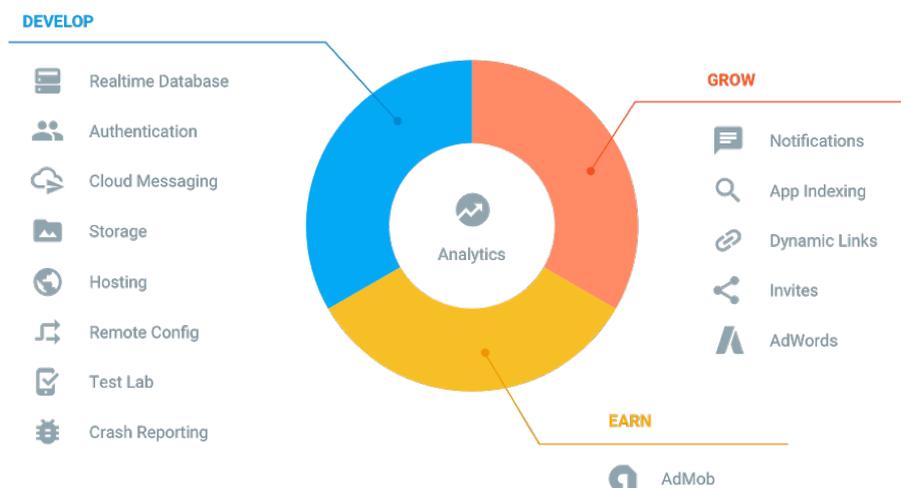


Figura 4. Componentes y Herramientas de Firebase.

Tomado de Tamada, (2017)

3. Capítulo III Desarrollo del Proyecto

3.1. Descripción del Proyecto

Los alumnos se podrán registrar en la aplicación para obtener un usuario y una contraseña con los que pueden acceder. Los usuarios registrados podrán entrar a un Lobby en donde encontrarán a otros alumnos conectados. Los alumnos se pueden poner en contacto mediante mensajes y videollamas uno a uno. Cuando empieza la videollamada aparecen un conjunto de actividades que los alumnos deberán cumplir cooperativamente. Cuando finalicen las actividades aparecerá un formulario en la pantalla en donde los alumnos deberán llenar unas preguntas y evaluar a su compañero. Éste formulario servirá para evaluar los resultados y

el progreso en las actividades, la evaluación de los compañeros determinará si los alumnos tienen un buen comportamiento al realizar las actividades.

3.2. Diagrama de Clases

La aplicación contiene 5 clases principales en su flujo, las clases de autenticación que son Login y Signup que manejan los campos de inicio de sesión de los usuarios como email y contraseña.

Luego están las clases de para la sección del Lobby, como se puede ver en la Figura 5 la clase Activity maneja los campos de la comunicación como un arreglo con los mensajes y un string de la dirección del stream de video remoto al que se conectan los usuarios.

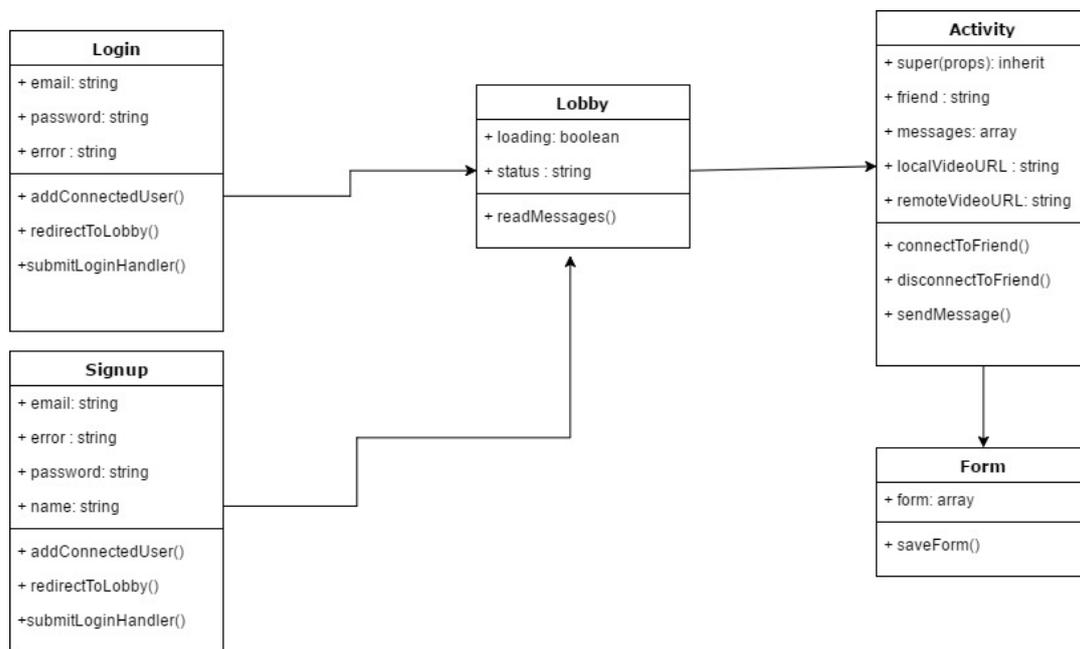


Figura 5. Diagrama de Clases

3.3. Diagrama de actividades

La Figura 6 describe las actividades principales en la aplicación y el flujo que deberían tener las mismas para obtener un resultado exitoso. Las acciones que se deben cumplir incluyen la autenticación (iniciar sesión y crear cuenta), luego de estas acciones se puede llegar al flujo de las actividades en donde se debe escoger un usuario, comunicarse con el usuario, realizar las actividades propuestas y por último llenar el formulario de retroalimentación.

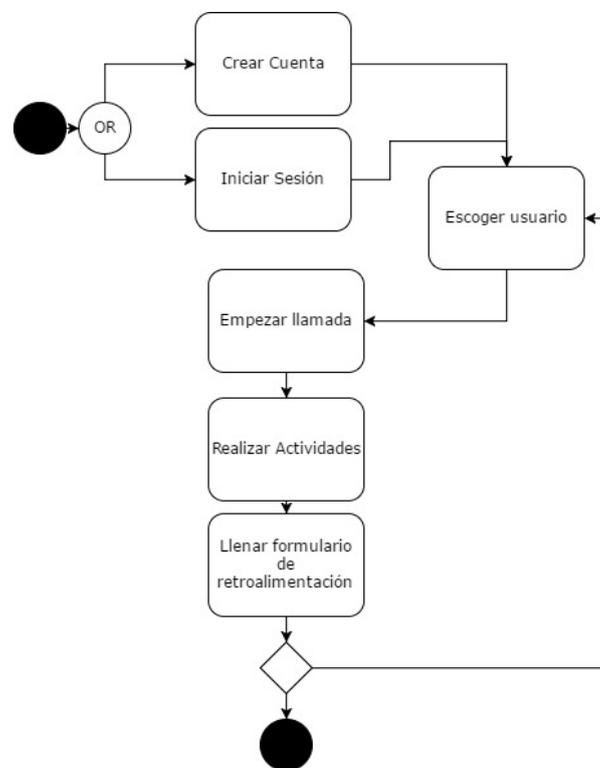


Figura 6. Diagrama de Actividades

3.4. Casos de Uso

En las especificaciones de la aplicación se definieron algunos casos de uso diferentes para los diferentes tipos de usuario que son el Estudiante y el Profesor. Entre éstas especificaciones se tomaron las más importantes y necesarias para

realizar éste prototipo. A continuación, se encuentra el gráfico de Casos de Uso en el que se definen las acciones que pueden realizar los usuarios dentro del sistema.

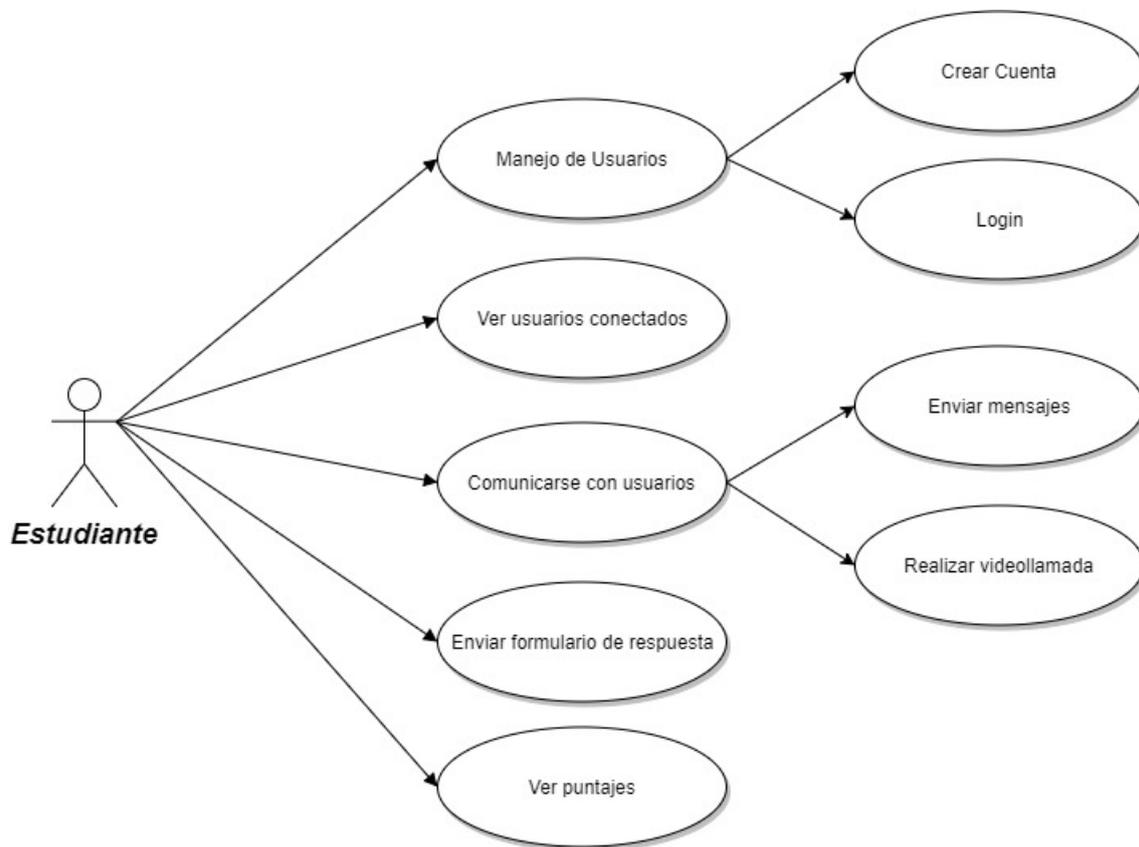


Figura 7. Diagrama de Casos de Uso

3.4.1. Descripción de Actores

En el siguiente cuadro se describe los diferentes actores de la aplicación.

Tabla 1

Descripción del actor Alumno

Actor	Alumno
Descripción	Alumnos de colegio desde 8º grado de Educación general básica hasta 3º curso de Bachillerato
Características	Puede interactuar con otros Alumnos en las actividades y llamadas
Relaciones	Interactúa con otros usuarios Alumno mediante llamadas y mensajes.

3.5. Scrum

En el desarrollo de éste proyecto utilizamos una metodología Scrum simplificada, en donde sólo intervienen dos personajes:

- Product Owner y Scrum Master, que será el profesor guía.
- Equipo Scrum, que será el desarrollador.

Se creó un Backlog de tareas que se deben realizar y se definió el número de Sprints a realizar. En éste caso realizamos Sprints de 2 semanas con Stand-up meetings cada dos días para verificar avances y tareas bloqueadas.

Tabla 2

Sprint Planning Scrum

Nombre de la Tarea	Duración	Comienzo	Fin	Sprint
Wireframes de aplicación	10d	02/20/17	03/03/17	Sprint 1
Creación servidor Express con Node.js	10d	03/06/17	03/17/17	Sprint 2
Instalación módulos Node.js	10d	03/06/17	03/17/17	Sprint 2
Instalación React y módulos	10d	03/06/17	03/17/17	Sprint 2
React Router Set-up	10d	03/06/17	03/17/17	Sprint 2
Instalar Firebase	10d	03/20/17	03/31/17	Sprint 3
Creación de Autenticación	10d	04/03/17	04/14/17	Sprint 4
Creación Interfaz gráfica	10d	04/03/17	04/14/17	Sprint 4
Instalar Peerjs	10d	04/17/17	04/28/17	Sprint 5
Crear Servidor Peerjs	10d	04/17/17	04/28/17	Sprint 5
Crear comunicación RTC	10d	05/01/17	05/12/17	Sprint 6
Implementar Gamificación	10d	05/15/17	05/26/17	Sprint 7
Crear Actividades	10d	05/29/17	06/09/17	Sprint 8
Crear Formularios	10d	06/12/17	06/23/17	Sprint 9

En la Tabla 2 podemos ver detallado un plan de Scrum con las tareas o Backlog a la izquierda y un cronograma de Sprints a la derecha. Se necesitaron 9 Sprints de 2 semanas empezando desde el 20 de febrero de 2017.

3.5.1. Sprint 1

3.5.1.1. Wireframes de aplicación

Como parte del proceso de Wireframe se crearon 3 diferentes pantallas que funcionarían como base para el desarrollo de la Interfaz Gráfica. La primera pantalla es la página que contiene los formularios de inicio de sesión y la bienvenida.

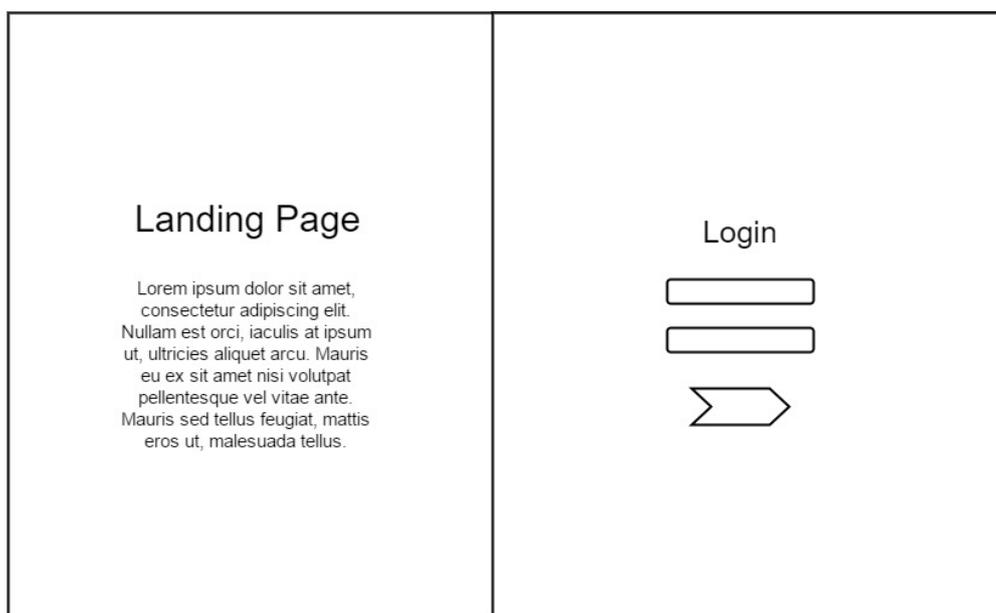


Figura 8. Wireframe de Página de Bienvenida

Luego de que el usuario haya iniciado sesión o se haya creado una cuenta será redirigido al Lobby, en ésta página podrán encontrar a otros alumnos conectados a la aplicación.

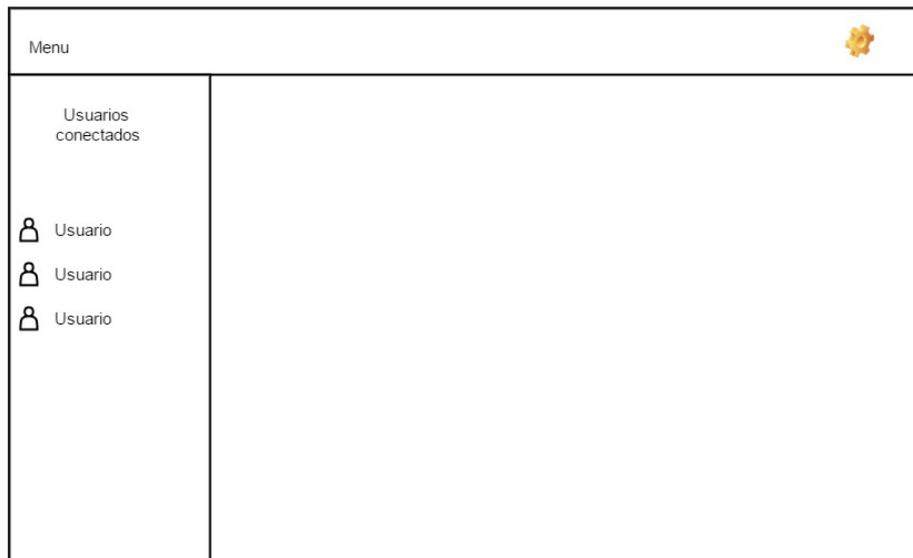


Figura 9. Wireframe de Lobby

Finalmente la página de la video llamada y actividades, como se puede ver en la Figura 10 a la derecha de la pantalla tendrán todos los componentes para poder comunicarse con el otro usuario con video y con mensajes.

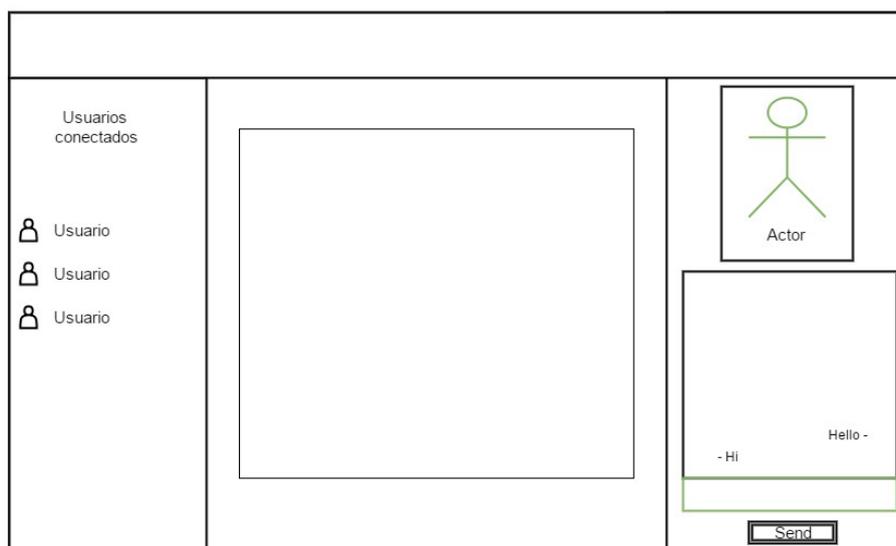


Figura 10. Wireframe de Actividad

3.5.2. Sprint 2

3.5.2.1. Control de Versiones

Para facilitar el versionamiento de la aplicación se utilizó Git, se utilizó un flujo de trabajo llamado Gitflow en donde se manejan dos ramas principales master y dev, cada feature se crea en su propia rama que luego será unida a la rama dev mediante un pull-request. Una vez que los cambios están completos se publican en la rama master.

Se puede descargar el código del repositorio mediante comandos git o accediendo al repositorio en Bitbucket a través de la web.

Código de desarrollo (DEV)

El código de la aplicación se encuentra en un repositorio git, se puede clonar los archivos con el comando de consola

```
git clone https://cmp92@bitbucket.org/cmp92/lang.git
```

Código ejecutable (PROD)

Se puede descargar el prototipo ejecutable desde

```
git clone https://cmp92@bitbucket.org/cmp92/langprod.git
```

3.5.2.2. Configuración base de la instalación

Para configurar el ambiente de la aplicación es necesario instalar unos requerimientos previos. En el desarrollo del prototipo se utilizó una máquina con el sistema operativo Ubuntu.

Como primer paso se instaló Node.js versión 6.11.1, luego utilizamos npm (Node Package Manager) para instalar React. Para crear un proyecto de React se deben ejecutar los siguientes comandos:

- “npm install -g create-react-app
- create-react-app mi-app”(Node, 2017)

Estos comandos crean la estructura base de la aplicación con los directorios y archivos como el package.json que maneja las dependencias de la aplicación.

En la Figura 11 se encuentra el código de configuración del servidor con Express en donde se utilizan módulos de Node como path, file system, https y express.

```
const path = require('path')
const fs = require('fs');
const https = require('https');
const express = require('express');
const app = express();

app.use(express.static(path.join(__dirname, 'build')));

const httpsServer = https.createServer({
  key: fs.readFileSync(path.join(__dirname, 'key.pem')),
  cert: fs.readFileSync(path.join(__dirname, 'cert.pem')),
  passphrase: '****'
}, app);

httpsServer.listen(8443);
```

Figura 11. Configuración de Servidor

Para correr y probar la ejecución de la aplicación podemos correr el comando *npm start* con el que la aplicación se compilará y correrá en el puerto especificado 8443 que es un puerto https.

El puerto https será necesario luego para la comunicación RTC con el servidor de Peerjs. La configuración del servidor https necesita un certificado SSL.

React-router son una colección de componentes de navegación para el front-end que permite a la aplicación enrutarse dinámicamente mientras la aplicación está renderizándose. Esto permite aprovechar las capacidades de React para renderizar solamente los componentes que cambian en la página.

```
const routes = {
  path: '/',
  component: Root,
  indexRoute: {
    component: Home
  },
  childRoutes: [
    {
      path: 'Home',
      component: Home
    },
    {
      path: 'Login',
      component: Login
    },
    {
      path: 'SignUp',
      component: SignUp
    },
    {
      path: 'Puntaje',
      component: Puntaje
    },
    {
      path: 'Lobby',
      component: Friends,
      childRoutes: [
        {
          path: ':uid',
          component: Friend,
        }
      ]
    }
  ]
};
```

Figura 12. Rutas creadas con React-Router

3.5.3. Sprint 3

3.5.3.1. Configuración Firebase

Firebase proporciona varios servicios, en el caso de ésta aplicación utilizaremos el manejo de Autenticación y la Base de Datos. Firebase permite utilizar varios tipos de cuentas como Facebook, Twitter, etc para manejar usuarios. Como se trata de estudiantes se definió la creación de usuarios solamente con un email.

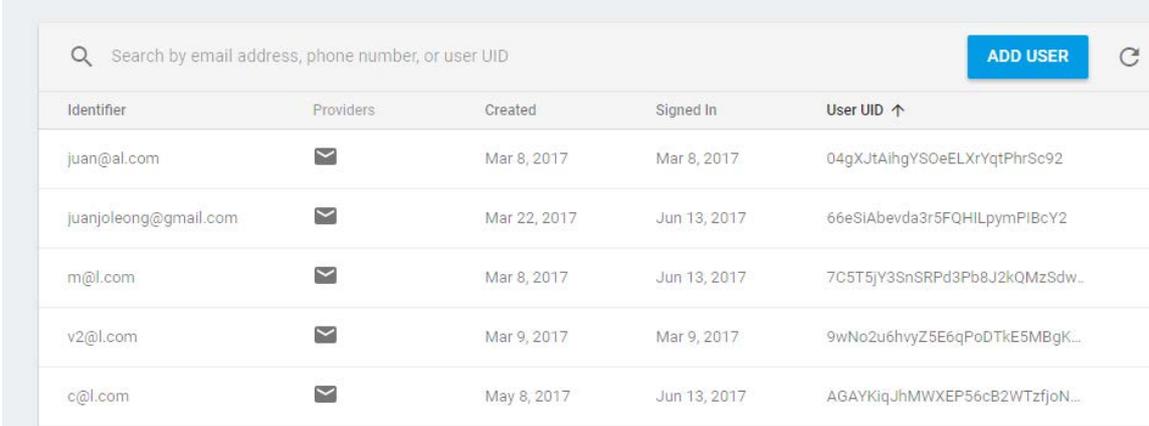
Firebase posee una interfaz gráfica en donde se puede ver la información almacenada. Se puede acceder a través de un browser en www.firebase.google.com.

Para hacer llamadas a Firebase desde la aplicación se debe añadir el siguiente código que es provisto desde la cuenta de Firebase, los valores deben cambiar con cada cuenta.

```
<script
src="https://www.gstatic.com/firebasejs/4.1.2/firebase.js">
</script>
<script>
  // Initialize Firebase
  var config = {
    apiKey: "AIzaSyApDfg7VP3P9lDOu9s*****",
    authDomain: "****.firebaseapp.com",
    databaseURL: "https://****.firebaseio.com",
    projectId: "****",
    storageBucket: "****.appspot.com",
    messagingSenderId: "91413342****"
  };
  firebase.initializeApp(config);
</script>
```

Figura 13. Código de configuración de Firebase

Firestore crea un registro de los usuarios y les asigna un ID, éste UID servirá luego para crear las conexiones con el PeerServer y realizar comunicaciones RTC.



Identifier	Providers	Created	Signed In	User UID ↑
juan@al.com	✉	Mar 8, 2017	Mar 8, 2017	04gXJtAihgYSOeELXrYqtPhrSc92
juanjoleong@gmail.com	✉	Mar 22, 2017	Jun 13, 2017	66eSiAbeveda3r5FQHILpymPIBcY2
m@l.com	✉	Mar 8, 2017	Jun 13, 2017	7C5T5jY3SnSRPd3Pb8J2kQMzSdw...
v2@l.com	✉	Mar 9, 2017	Mar 9, 2017	9wNo2u6hvyZ5E6qPoDTkE5MBgK...
c@l.com	✉	May 8, 2017	Jun 13, 2017	AGAYKiqJhMWXEP56cB2WTzfoN...

Figura 14. Registro de Usuarios con su UID en Firebase

Para almacenar información, Firebase utiliza una estructura JSON. En la Figura 15 podemos ver un ejemplo de registros en la Base de Datos, en este se encuentran los usuarios conectados a la aplicación y tienen como identificador el UID creado para los usuarios.



Figura 15. Almacenamiento en Base de Datos con JSON de información de usuarios

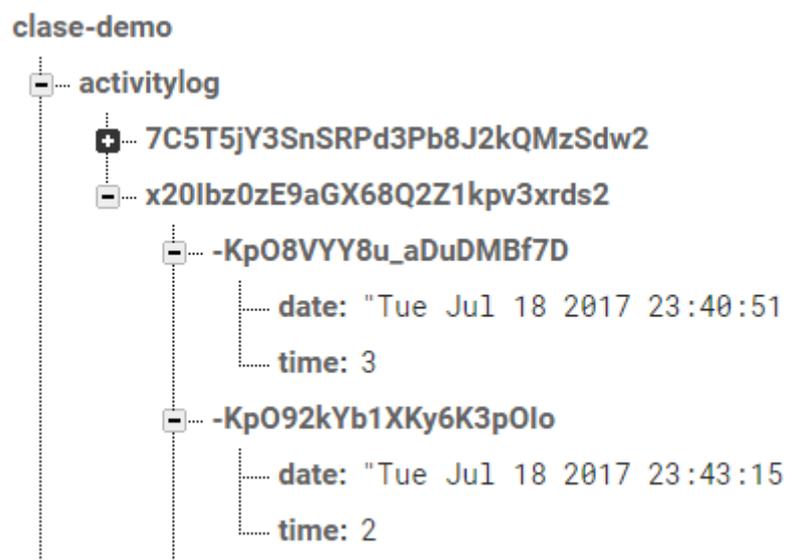


Figura 16. Almacenamiento en Base de Datos con JSON de información de actividades

En la Figura 16 podemos ver el almacenamiento del registro de actividades, empieza con el UID del usuario y seguido por un ID autogenerado por firebase

para cada actividad en donde se almacena una marca de la fecha con el tiempo de la actividad.

3.5.4. Sprint 4

3.5.4.1. Creación Interfaz Gráfica

La interfaz gráfica son todos los elementos con los que el usuario va a interactuar, necesita ser de fácil acceso y con un diseño intuitivo que sea adaptable a los diferentes dispositivos.

En el desarrollo de la interfaz gráfica se utilizó un framework de frontend llamado Bootstrap, este framework se enfoca en sitios mobile-first y utiliza diferentes clases en el HTML para crear los diferentes componentes y estilos CSS.

React tiene su propia versión de Bootstrap reconstruida específicamente para cumplir con los estándares del framework de Facebook y ayudar a los desarrolladores con librerías y componentes que se adaptan fácilmente a la sintaxis de JSX que utiliza React.

Luego de haber realizado los wireframes del sitio y de las pantallas podemos seguir con el armado de la página para que se vea reflejado lo planeado en los la aplicación web.

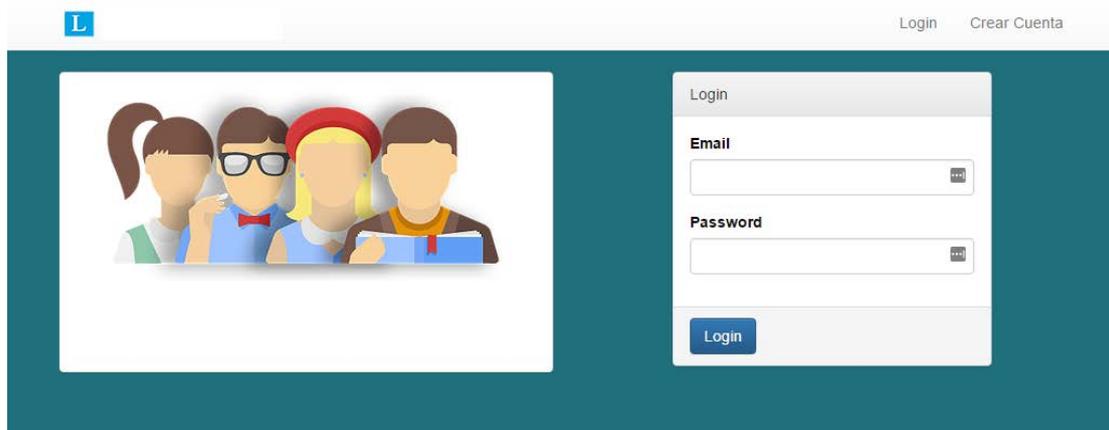


Figura 17. Desarrollo Interfaz Gráfica de Página de Bienvenida

Una vez que el usuario se haya logueado en la aplicación podrá acceder al Lobby para realizar las actividades. En la Figura 18 se puede ver la página como un usuario logueado en donde se presenta una imagen de avatar, con el nombre y un nivel dependiendo del puntaje.

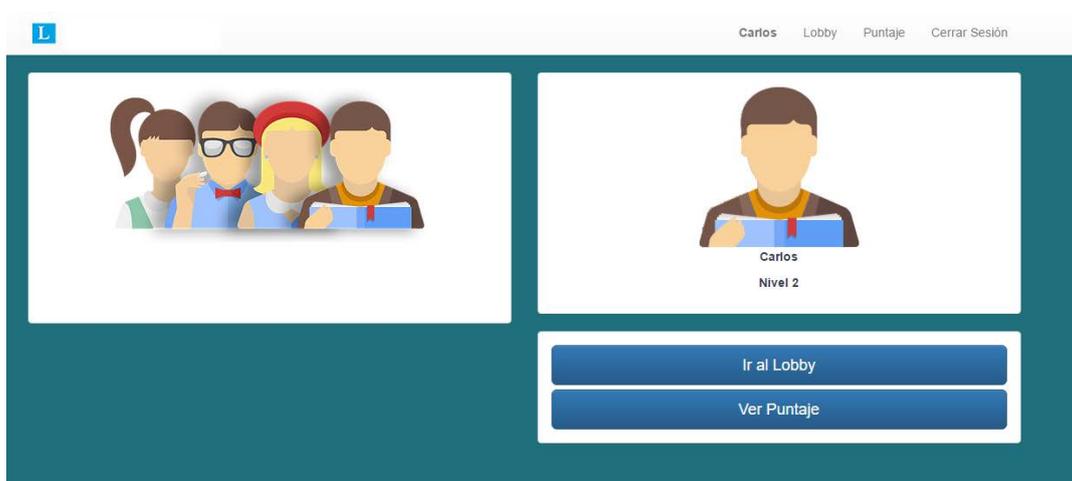


Figura 18. Desarrollo Interfaz Gráfica con Usuario Logueado

En la página del Lobby, en donde se puede encontrar la lista de Amigos, las actividades y el bloque de comunicación, se utilizar varios componentes de React-Bootstrap y HTML5.

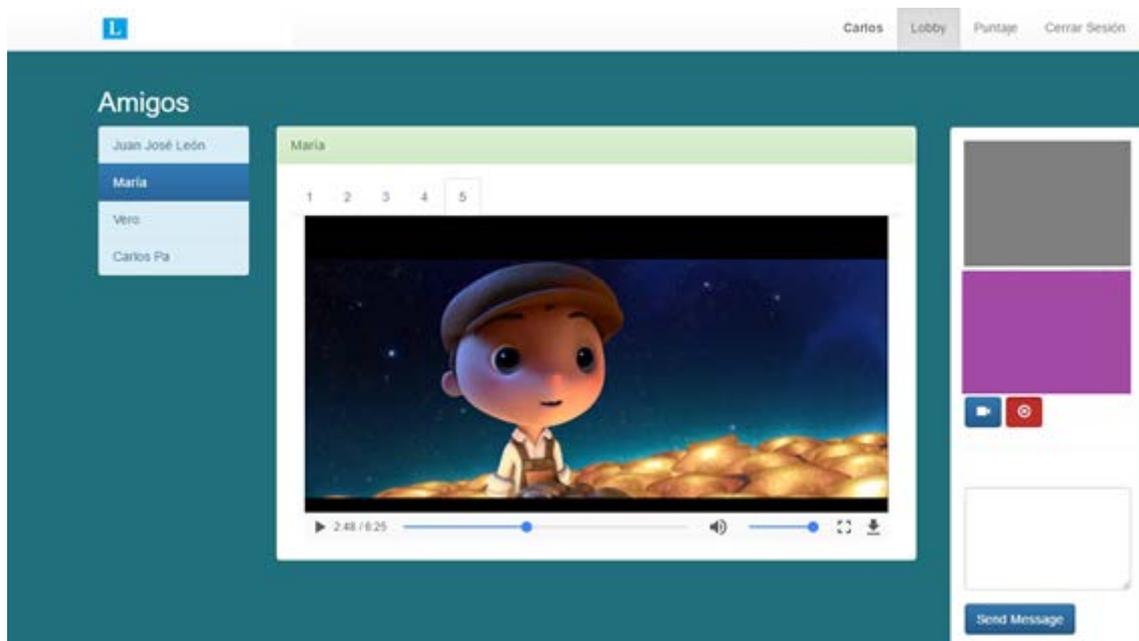


Figura 19. Desarrollo Interfaz Gráfica Lobby y Actividad

3.5.4.2. Creación de Autenticación

Con la autenticación en la aplicación se planeó resolver tres casos de uso que se explican a continuación

Tabla 3

Caso de Uso Crear Cuenta

Nombre:	Crear Cuenta
Actores:	Alumnos

Función:	Permitir a los alumnos crear una cuenta para ingresar a la aplicación
Descripción:	Los usuarios podrán crear una cuenta e ingresar su información de perfil

Tabla 4

Caso de Uso Login

Nombre:	Login
Actores:	Alumnos
Función:	Permitir a los usuarios entrar en la aplicación
Descripción:	Los usuarios pueden entrar a la aplicación con el email y contraseña de su cuenta.

Tabla 5

Caso de Uso Ver Usuarios Conectados

Nombre:	Ver usuarios conectados
Actores:	Alumnos
Función:	Ver una lista de alumnos conectados

Descripción:	Los alumnos podrán ver una lista de alumnos conectados con quienes podrán hablar
---------------------	--

Se creó un componente de autenticación que contiene los formularios y métodos para realizar inicio de sesión y creación de usuarios. Se debe ingresar los datos del alumno como Nombre, Email y una contraseña para poder crear una cuenta. El componente verifica las cuentas mediante el servicio de autenticación de Firebase.

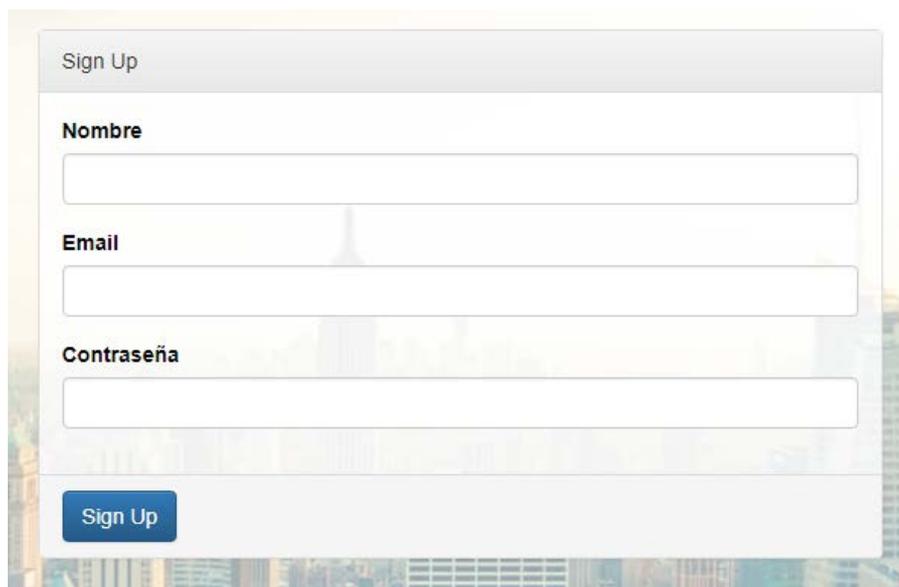
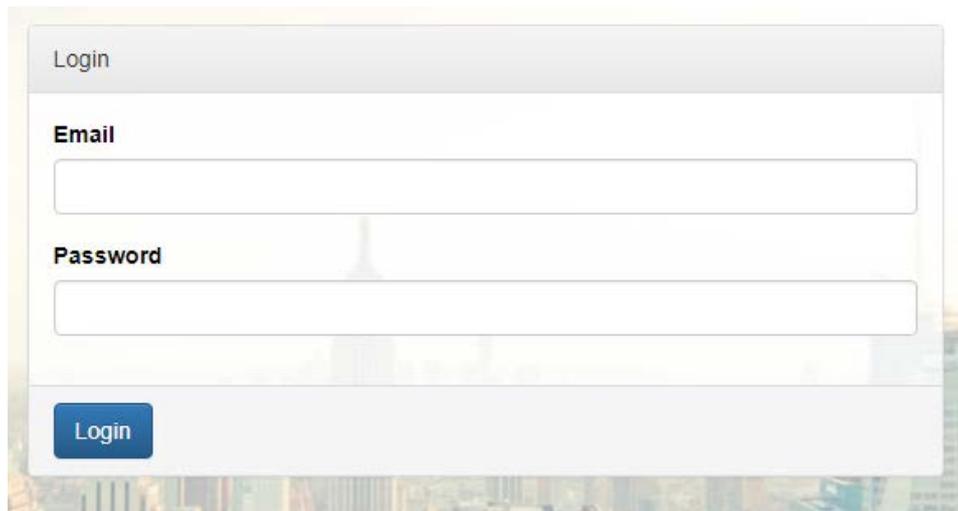
A screenshot of a web form titled "Sign Up". The form is set against a background image of a city skyline. It contains three input fields: "Nombre", "Email", and "Contraseña". Below the fields is a blue button labeled "Sign Up".

Figura 20. Creación de Usuario

Para iniciar sesión se necesita del email y la contraseña provistos anteriormente en la creación de cuenta.



The image shows a login form with a light gray header containing the word "Login". Below the header, there are two input fields: one labeled "Email" and one labeled "Password". At the bottom left of the form is a blue button with the text "Login". The form is set against a background image of a city skyline.

Figura 21. Pantalla de Login

Con la autenticación de Firebase podemos crear un registro en la base de datos para controlar cuando un usuario está conectado en la aplicación. En la base de datos generamos un registro con el uid del usuario, dentro de este se encuentra el nombre del usuario y un campo booleano `connected` que determina si el usuario está conectado o no. En la Figura 23 y 24 se muestran los registros en la base de datos y los resultados en la aplicación respectivamente.

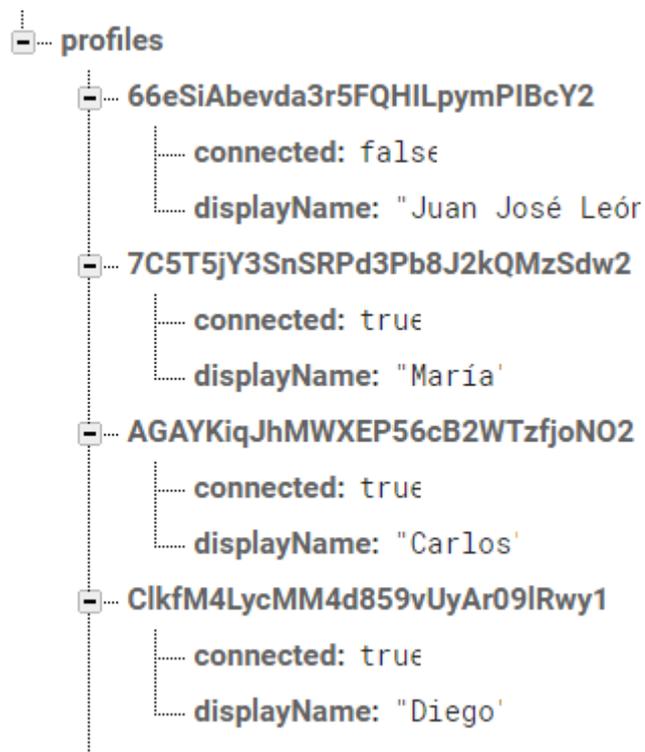


Figura 22. Registros en Firebase sobre usuarios conectados

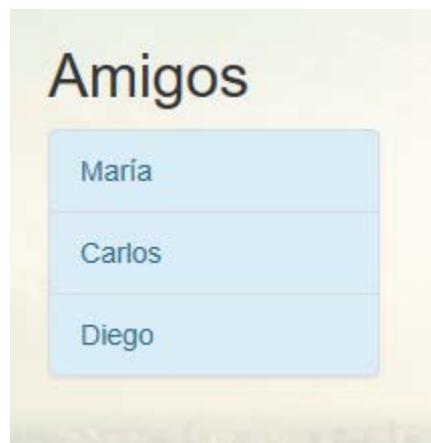


Figura 23. Componente de Lobby mostrando usuarios conectados

3.5.5. Sprint 5

3.5.5.1. Configuración Peerjs

El desarrollo de las comunicaciones Real Time Communication se realizó con PeerJS que facilita el desarrollo mediante métodos y funciones para el uso de las características WebRTC de los browsers en la aplicación.

PeerJS tiene un servicio gratuito de servidor cloud PeerServer que funciona con un API key pero solo permite realizar conexiones con protocolos http. Los navegadores han implementado seguridades que inhabilitan las comunicaciones de contenido mixto entre los diferentes protocolos http y https por ser una vulnerabilidad. En el caso de ésta aplicación fue necesario crear un servidor PeerServer propio para poder manejar llamadas por https.

La creación del servidor de Peerjs se puede lograr de varias maneras, en el Anexo 1 se encuentra la documentación para la instalación de este servidor. En el caso de este prototipo utilizamos la opción con Heroku. Heroku es una plataforma como servicio (PaaS) que nos permite correr aplicaciones en la nube.

En la documentación de Peerjs se encuentra un botón en el que crea automáticamente el servidor, en la administración de Heroku habilitamos nuestra aplicación y obtenemos una URL en donde está alojado nuestro servicio.

En el Sprint 6 se explicará más a fondo sobre la configuración necesaria dentro del prototipo.

3.5.6. Sprint 6

3.5.6.1. Crear comunicación RTC

En este Sprint se planeó resolver el caso de uso de la Comunicación entre los Alumnos como se describe en el siguiente cuadro.

Tabla 6

Caso de Uso Comunicarse con usuarios

Nombre:	Comunicarse con usuarios
Actores:	Alumnos
Función:	Permitir a los alumnos comunicarse con otros alumnos.
Descripción:	Permitir el envío de mensajes y realización de video llamadas para lograr una comunicación efectiva.

En la aplicación se debe referenciar al servidor Peer que se creó anteriormente con ayuda de Heroku. Para este prototipo Heroku nos asignó la URL <https://stark-meadow-76457.herokuapp.com/>. En la Figura 24 se muestran los parámetros utilizados para crear las conexiones RTC.

```
const options = {
  debug: 3,
  host: 'stark-meadow-76457.herokuapp.com',
  port: 443,
  secure: true
};
```

Figura 24. Configuración de Peerjs con Peerserver desplegado en Heroku

Al momento de crear la conexión RTC podemos recibir un objeto URL con el que podemos crear un stream del video remoto. En la función `connectToFriend()` detallado en la Figura 25 se demuestra cómo se maneja la conexión con peer, en donde el estado `remoteVideoUrl` creará el stream de video en el JSX detallado en la Figura 26.

```
connectToFriend() {
  const uid = this.props.params.uid;
  peer.onData(uid, (message) => {
    this.setState({
      messages: [...this.state.messages, message]
    })
  });
  peer.onCall(uid, (stream) => {
    this.setState({
      remoteVideoUrl: URL.createObjectURL(stream),
      callInProgress: true,
    });
    this.startTimer();
  });
  peer.connect(uid);
}
```

Figura 25. Función para recibir stream de video y mensajes

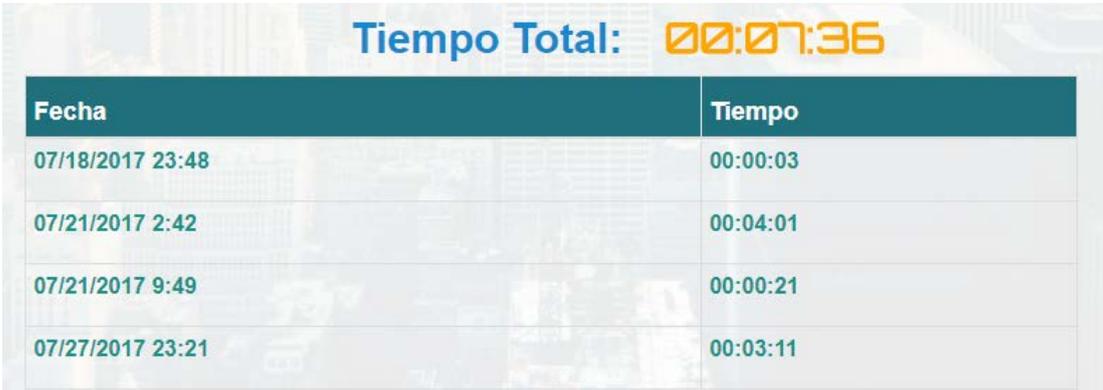
```
{
  this.state.remoteVideoUrl &&
  <video
    autoPlay
    src={this.state.remoteVideoUrl}></video>
}
{
  this.state.localVideoUrl &&
  <video
    muted="true"
    autoPlay
    src={this.state.localVideoUrl}>
  </video>
}
```

Figura 26. Configuración de Peerjs con Peerserver desplegado en Heroku

3.5.7. Sprint 7

3.5.7.1. Implementar Gamificación

En éste prototipo se implementó conceptos simples pero a la vez necesarios en la gamificación. Al momento de realizar una llamada existe un contador para medir el tiempo que duran las llamadas. Los alumnos subirán su nivel dependiendo del tiempo que se haya usado la plataforma.



Tiempo Total: 00:07:36	
Fecha	Tiempo
07/18/2017 23:48	00:00:03
07/21/2017 2:42	00:04:01
07/21/2017 9:49	00:00:21
07/27/2017 23:21	00:03:11

Figura 27. Componente de Puntaje (Reporte de Tiempo)

Para mostrar los resultados fue necesario crear una página con los resultados del alumno. La página de puntajes contiene toda la información correspondiente a la gamificación. En la Figura 27 se puede notar un reporte de las actividades realizadas y del tiempo total que se ha utilizado la plataforma. Existirá una sección en donde se puedan ver las medallas obtenidas y un gráfico con el feedback obtenido de otros compañeros, que por motivos del prototipo no estuvieron en el alcance.



Figura 28. Página de Puntaje

3.5.8. Sprint 8

3.5.8.1. Crear Actividades

Se crearon ejemplos de actividades que cumplan con la metodología nocial-funcional. Las actividades deberían fomentar una necesidad de comunicación con situaciones de la vida real.

Un ejemplo de actividad que se creó se basa en juego muy conocido que se llama Adivina Quien, en éste juego el objetivo es el de encontrar el personaje de tu compañero haciendo preguntas simples sobre las características de los personajes.



Figura 29. Juego Adivina el personaje

Otras actividades que se crearon requieren de intercambiar ideas, realizar entrevistas y comentar sobre lecturas y videos. Las actividades se deben realizar en inglés y español.

Carlos

1 2 3 4 5 Terminar

Mira el video y discute las preguntas con tu compañero.

¿Por que discuten el papá y el abuelo del niño?

¿Que consiguen barriendo la luna?

Figura 30. Actividades con videos

3.5.9. Sprint 9

3.5.9.1. Crear Formularios

Cuando la actividad se acaba o cuando alguno de los usuarios se desconecta aparecerá un formulario de retroalimentación en donde los usuarios deberán llenar unas preguntas sobre las actividades. Se deberá también calificar el comportamiento del compañero asignándole uno de las características disponibles.

Feedback

Califica el comportamiento de tu compañero

Amigable ▼

Evalúe su comprensión de la actividad

1 (Mala)

2

3

4

5 (Excelente)

Escribe un comentario de la Actividad

Figura 31. Formulario de Retroalimentación

Este formulario es necesario para cumplir con el concepto de Feedback de la gamificación en donde se recibe información del desempeño en la actividad. Los alumnos pueden calificar a su compañero escogiendo una característica con la que más se hayan destacado como amigable, cooperativo, entusiasta, etc.

4. Conclusiones y Recomendaciones

4.1. Conclusiones

Las metodologías ágiles contribuyen con un proceso de desarrollo continuo organizado. Scrum facilitó el manejo de tareas y la organización en el desarrollo de ésta aplicación asignando prioridades a las actividades importantes adaptándose a las necesidades de cada etapa de éste proceso.

Se pudo constatar el gran aporte de las tecnologías Javascript, facilitaron el proceso de desarrollo web con herramientas y módulos. Las tecnologías escogidas en este proyecto como React.js y Node.js son bastante nuevas y están ganando bastante popularidad por su eficiencia en los proyectos de gran escala.

La comunicación RTC con Peer.js logra conexiones entre browsers para enviar cualquier tipo de información. Las librerías de Peer.js ofrecen métodos para solucionar las comunicaciones RTC con facilidad.

En cuanto a la técnica de gamificación, se acopla correctamente en el medio didáctico porque motiva a los alumnos a seguir consiguiendo objetivos y conjuntamente con reportes de puntajes y tiempo de uso se obtiene una visión del avance. Los estímulos que genera con recompensas son una buena manera de mantener la motivación en el estudio.

Considero que esta aplicación facilitaría mucho el aprendizaje y mejoramiento de los idiomas, la tecnología como videollamadas y chat nos permiten reforzar nuestras habilidades comunicativas en un entorno que se acerca al nativo.

Se puede continuar el desarrollo de esta idea aumentando los idiomas con los que trabaja, de ésta manera se podrá conectar con alumnos de más países.

4.2. Recomendaciones

Se recomienda al momento de seleccionar la tecnología para desarrollar la aplicación, considerar versiones estables para evitar tener que cambiar la programación en medio de etapas de desarrollo. En lo posible la tecnología debe tener una buena documentación y además una comunidad activa de desarrolladores que aporten con su experiencia.

La aplicación debería utilizarse semanalmente con estudios complementarios del idioma de tal manera que se pueda mejorar todos los aspectos del idioma como lectura, escritura, audición y el hablar.

La metodología de Scrum no pudo ser aprovechada completamente en éste proyecto por las limitadas personas involucradas, aunque las herramientas que utiliza fueron muy útiles en la organización del proceso de desarrollo.

Referencias

Buckner, N. (2014). *Creating a wireframe*. Recuperado el 25 de mayo de 2017 de <http://nigelbuckner.com/downloads/handouts/web/Wireframing.pdf>

CVC. (2017). *Diccionario De Términos Clave De ELE*. Recuperado el 29 de mayo de 2017 de http://cvc.cervantes.es/ensenanza/biblioteca_ele/diccio_ele/indice.htm

EF Education First (2016). EF English proficiency index. Recuperado el 28 de junio de 2017 de <http://www.ef.com.ec/epi/regions/latin-america/>

Express (2017). Documentación. Recuperado el 5 de abril de 2017 de <http://expressjs.com/es>

Firebase (2017). Documentation. Recuperado el 25 de mayo de 2017 de <https://firebase.google.com/>

Fernando (2017). *Gamificación educativa*. Recuperado el 10 de julio de 2017, de <http://canaltic.com/blog/?p=1733>

Gancedo, M. (2016). *El desarrollo de la expresión oral en programas de Auxiliares de conversación en Martinica: una propuesta didáctica*. (Tesis de maestría). Universidad de Cantabria. Recuperado el 29 de mayo de 2017 de: <https://repositorio.unican.es/xmlui/bitstream/handle/10902/9838/TFM.MMV.pdf?sequence=1>.

- Houston, J., Bee, H. y Rimm, D. (2013). *Invitation to psychology*. Academic Press.
- Kalk, S. (2017). *Scrum-Roles*. Ness-tech.co.il. Recuperado el 20 de mayo de 2017 de <http://www.ness-tech.co.il/en/solutions/testing-qa-v-ness/Scrum-Roles>
- Kiessling, M. (2012). *The Node Beginners Book*. Recuperado el 25 de mayo de 2017 de <https://www.nodebeginner.org/>
- MINISTERIO DE EDUCACIÓN - Ecuador. (2016). Currículo lengua extranjera. Recuperado el 28 de junio de 2017 de <https://educacion.gob.ec/curriculo-lengua-extranjera/>
- Microsoft (2017). Crear modelos para la aplicación. Recuperado el 5 de abril de 2017 de <https://msdn.microsoft.com/es-ES/library/dd409436.aspx>
- NodeJS (2017). *Documentación*. Recuperado el 25 de mayo de 2017 de <https://nodejs.org/en/>
- W3Schools (2017). *NodeJS Introduction*. Recuperado el 25 de mayo de 2017 de https://www.w3schools.com/nodejs/nodejs_intro.asp
- Parente, D. *GAMIFICACION EN LA EDUCACIÓN*. Gamificación en aulas universitarias, (2016). Recuperado el 5 de abril de 2017 de http://incom.uab.cat/download/eBook_incomuab_gamificacion.pdf
- PeerJS (2017). *Documentation*. Recuperado el 5 de abril de 2017 de <http://peerjs.com/docs/>

- Peñafiel, F. (24 Marzo de 2014). Acuerdo No. 0052-14. Recuperado el 15 de julio de 2017 de <https://educacion.gob.ec/wp-content/uploads/downloads/2014/03/ACUERDO-052-14.pdf>
- React (2017). *Documentation*. Recuperado el 5 de abril de 2017 de <https://facebook.github.io/react/>
- Tamada, R. (2017). *Android Getting Started with Firebase - Login and Registration Authentication*. Recuperado el 25 de mayo de 2017 de: <http://www.androidhive.info/2016/06/android-getting-started-firebase-simple-login-registration-auth>
- Terrera, G. (2017). *Diferencias entre Sketch - Wireframe - Mockup - Prototipo - TestingBaires*. TestingBaires. Recuperado el 25 de mayo de 2017 de <http://testingbaires.com/diferencias-entre-sketch-wireframe-mockup-prototipo/>
- Valda, F. y Arteaga, C. (2017). *Diseño e implementación de una estrategia de gamificación en una plataforma virtual de educación*. *Scielo.org.bo*. Recuperado el 4 de julio de 2017 de http://www.scielo.org.bo/scielo.php?pid=S2071-081X2015000100006&script=sci_arttext&tlng=es
- Visual Paradigm (2016). *Wireframe Handbook*. Recuperado el 25 de mayo de 2017 de <https://www.visual-paradigm.com/learning/handbooks/agile-handbook/wireframe.jsp>
- WebRTC (2016). *Architecture*. Recuperado el 5 de abril de 2017 de <https://webrtc.org/architecture/#>

ANEXOS

Anexo 1: Instalación de Peer Server

PeerServer: A server for PeerJS

PeerServer helps broker connections between PeerJS clients. Data is not proxied through the server.

If you prefer to use a cloud hosted PeerServer instead of running your own, [sign up for a free API key here](#) or



Run PeerServer

Install the library:

```
$> npm install peer
```

Run the server:

```
$> peerjs --port 9000 --key peerjs
```

Or, create a custom server:

```
var PeerServer = require('peer').PeerServer;  
var server = PeerServer({port: 9000, path: '/myapp'});
```

Connecting to the server from PeerJS:

```
<script>  
  // No API key required when not using cloud server  
  var peer = new Peer('someid', {host: 'localhost', port: 9000, path:  
  '/myapp'});  
</script>
```

Using HTTPS: Simply pass in PEM-encoded certificate and key.

```
var fs = require('fs');  
var PeerServer = require('peer').PeerServer;  
  
var server = PeerServer({  
  port: 9000,
```

Anexo 2: Acuerdo No 52-14 del Ministerio de Educación en Ecuador

ACUERDA:

Expedir la siguiente NORMATIVA PARA LA IMPLEMENTACIÓN E INSTRUMENTALIZACIÓN DE LA DISPOSICIÓN TRANSITORIA CONTENIDA EN EL ACUERDO MINISTERIAL No. 041-14

Artículo 1.- DISPONER que la enseñanza de inglés, a partir del año lectivo 2016-2017, régimen Sierra y 2017-2018, régimen Costa sea obligatoria desde segundo grado de Educación General Básica hasta tercer curso de Bachillerato para todas las instituciones públicas, fiscomisionales y particulares del país.

Artículo 2.- DISPONER que las horas de Clubes establecidas en la malla curricular expedida mediante Acuerdo Ministerial 0041-14 del 11 de marzo de 2014 determinadas para los grados segundo a séptimo de Educación General Básica puedan ser usadas para la asignatura de inglés. En este caso, esta asignatura será evaluada tal como se determina en los artículo 193 y siguientes del Reglamento General de la Ley Orgánica de Educación Intercultural (LOEI).

Artículo 3.- DISPONER que para la implementación de la asignatura de inglés las instituciones educativas públicas incorporen a su planta docente, personal con calificación mínima de B2 de acuerdo con la escala del Marco Común Europeo de Referencia para las Lenguas, y pongan a disposición de los estudiantes los recursos pedagógicos necesarios para garantizar el adecuado aprendizaje de la lengua extranjera. Al inicio de cada año lectivo la institución educativa deberá remitir a la respectiva Dirección Distrital un informe con el detalle de los grados que han sido incorporados a la enseñanza del idioma inglés.

Artículo 4.- DISPONER que el Viceministerio de Educación coordine con el Instituto Nacional de Evaluación Educativa, la evaluación anual de los resultados del aprendizaje del idioma inglés en las instituciones educativas públicas, fiscomisionales y particulares, y que la

