Aplicación web de videoconferencia para prevenir el robo de información a estudiantes de todos los niveles educativos.

Videoconferencing web application to prevent information theft for students of all educational levels.

Joel Ruben Regalado Romero* (1).
Universidad DaVinci, CDMX, México.
joelromaro@gmail.com.

*corresponding author.

Artículo recibido en noviembre 05, 2021; aceptado en marzo 30, 2022.

Resumen.

El trabajo actual se desarrolla una aplicación web, la cual varios usuarios podrán conectarse simultáneamente por videoconferencia, la página puede usarse como medio de comunicación escolar, para el desarrollo se utilizó código abierto. El problema sobre el cual se desarrolló este artículo radica en el aumento de la inseguridad en tiempos de pandemia por COVID-19 al utilizar aplicaciones comerciales. Con este desarrollo se asegura la nula información compartida en internet y el óptimo funcionamiento de las reuniones.

Palabras clave: Aplicación web, ciberataques, seguridad, videoconferencia.

Abstract.

The current work develops a web application, which several users can connect simultaneously by videoconference, the page can be used as a means of school communication, open source was used for the development. The problem on which this article was developed lies in the increase in insecurity in times of the COVID-19 pandemic when using commercial applications. With this development, zero information is shared on the Internet and the optimal functioning of the meetings is ensured.

Keywords: Cyberattacks, security, videoconference, web application.

1. Introducción.

El Panorama de Amenazas en América Latina 2021 de *Kaspersky*, del informe anual realizado por el equipo de investigación y análisis, revela un aumento del 24% en ciberataques en la región durante los primeros ocho meses del año, en comparación con el mismo periodo en 2020. La conclusión de los especialistas es clara, la seguridad de las tecnologías para el trabajo remoto y educación a distancia debe ser prioridad y la piratería, tanto en dispositivos personales como profesionales, debe ser erradicada. En este contexto como se puede observar en la figura 1, Brasil lidera la región con más de 1,390 intentos de infección por minuto, seguido de México (299 por minuto); Perú (96 por minuto), Ecuador (89 por minuto) y Colombia (87 por minuto). (Diazgranados, 2021).

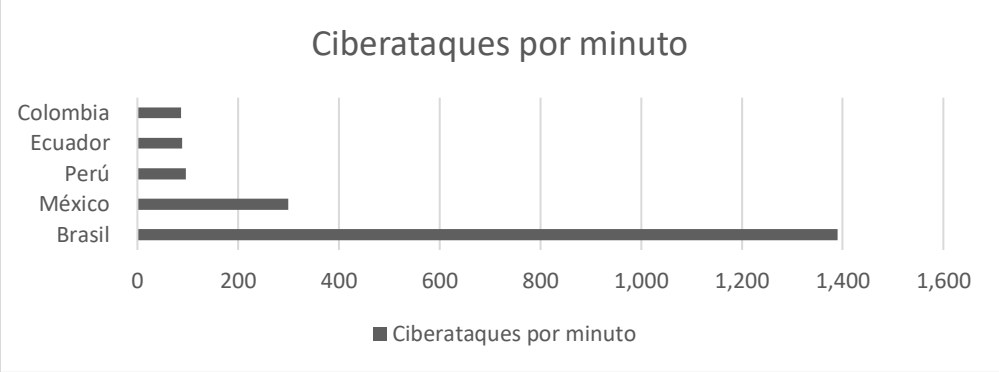

Figura 1. Ciberataques por minuto en América latina. Fuente: (Diazgranados, 2021).

A raíz de la pandemia por la *COVID-19* desde marzo de 2020 y las nuevas variantes como el *Ómicron* que han impactado a inicios del 2022, el Sistema Educativo Nacional se ha enfrentado a un desafío sin precedentes, implementando clases a distancia ante el cierre temporal de las escuelas, impactando todos los siclos escolares hasta el actual, aumentando el uso de las tecnologías de la información, como lo muestra la encuesta para la medición del impacto *COVID-19* en la educación, se identifica la población de estudiantes de 32.9 millones de alumnos de entre 3 a 29 años que se inscribieron en el siclo escolar 2020-2021 *(INEGI, 2021)*.

Por la educación en línea y el trabajo en casa, las plataformas de videoconferencia tuvieron un repunte histórico, y entre éstas, la que más relevancia ha tomado desde altos niveles de la política mundial hasta las vidas cotidianas de millones de estudiantes y trabajadores ha sido *Zoom* (El tiempo, 2020).

A través de una amplia variedad de servicios, aplicaciones, plataformas y entornos virtuales que brindan enormes oportunidades para el aprendizaje, a la vez que promueven la socialización y desarrollo de niñas, niños y adolescentes (NNA), así como de jóvenes estudiantes. Factores como la mayor dependencia de las comunicaciones y tecnologías de la información, el uso de múltiples soluciones digitales en el ámbito educativo, el mayor tiempo en línea, entre otros, incrementan la exposición de NNA, jóvenes estudiantes y docentes a amenazas y riesgos en línea. En este contexto, expertos en ciberseguridad advierten un entorno propicio para que prosperen los cibercriminales y que, tanto los miembros de la comunidad educativa como las instituciones académicas, se encuentren mayormente expuestos a múltiples amenazas de ciberseguridad (Secretaría de comunicaciones y transportes, 2020), como lo muestra la figura 2, el 48.3% de alumnos pasan de entre 3 a 5 horas dedicado a clases y actividades escolares, todos ellos utilizando una herramienta tecnológica. (INEGI, 2021).

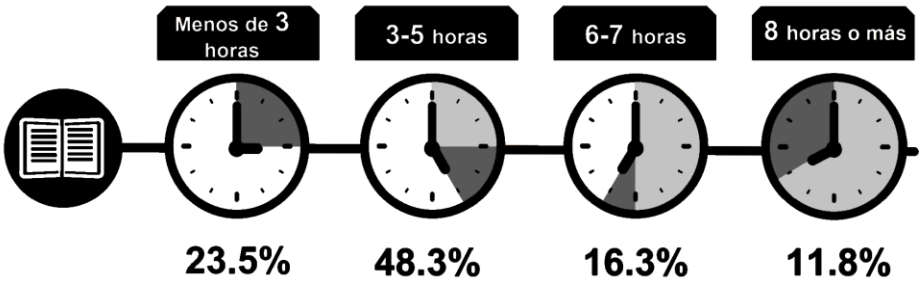

Figura 2. Distribución porcentual de la población de 3 a 29 años inscrita en el ciclo escolar 2020–2021 por tiempo dedicado a clases y actividades escolares por día. Fuente: *(INEGI, 2021)*.

La demanda de servicios de videoconferencias para la impartición de clases en línea, o bien, para sesiones de estudio, especialmente a partir de la emergencia sanitaria generada por COVID-19, se ha incrementado considerablemente. Las teleconferencias se han convertido en una herramienta indispensable para desarrollar las actividades de educación en

línea, dar continuidad a asuntos laborales, la vida cotidiana y la comunicación con familiares y amigos. Lo novedoso de estos servicios para muchos usuarios y la aparición de algunas vulnerabilidades en ciertas plataformas, supone para los ciberdelincuentes la oportunidad para el acceso no autorizado a información, robo de credenciales y acceso a los distintos recursos del dispositivo (como micrófono, cámara, etc.) (Secretaría de comunicaciones y transportes, 2020).

Un grupo de investigadores de seguridad descubrió que Zoom presenta una vulnerabilidad de día cero que los ciberdelincuentes podrían utilizar para lanzar ataques de ejecución remota de código, es decir, propagar malware capaz de tomar el control de los equipos. (El Universal, 2021). Por parte de Microsoft Teams, se han descubierto un total de cuatro vulnerabilidades con la introducción de la nueva característica de previsualización de enlaces. Estas fueron reportadas a Microsoft en Marzo de 2021, y hasta la fecha solo ha sido solucionada una de ellas, Las cuatro vulnerabilidades encontradas son del tipo *server-side request forgery* (SSRF) y un fallo de *spoofing* de previsualización de la URL, tanto en la web como en la aplicación de escritorio, y, en el caso de los usuarios de Android, se puede revelar su IP y realizar un ataque de denegación de servicio (*DoS*). (García, 2022). Por lo anterior, es necesario promover la adecuada protección de los usuarios para evitar incidentes al usar estos servicios, tomando en consideración las siguientes recomendaciones:

- Informarse sobre las políticas de privacidad y las medidas de seguridad que implementa el servicio que se desea utilizar.
- Descargar e instalar la aplicación correspondiente desde la página web oficial del desarrollador o desde las tiendas oficiales de apps.
- Mantener actualizada la aplicación que se utilice, pues es a través de este proceso que se puede asegurar que las vulnerabilidades detectadas y corregidas por el desarrollador se están implementando.

Al organizar una teleconferencia se recomienda tener en cuenta:

- En el caso de reuniones privadas, compartir el enlace directamente con los participantes, haciendo uso de las funciones de compartición de las propias aplicaciones, y evitando el uso de redes sociales o canales de comunicación abiertos que podrían promover accesos no deseados.
- Proteger la conferencia con una contraseña robusta, para restringir el acceso a ésta a personas no autorizadas.
- Poner a los asistentes en sala de espera, si la plataforma permite dicha funcionalidad. Previa verificación, se podrá aceptar su ingreso, de ser el caso.

Los ciberdelincuentes han aprovechado el auge de las plataformas de videoconferencia para cometer sus ataques. Por ello se tuvo la idea de desarrollar una aplicación web de videoconferencia, la cual sea privada y solo tengan acceso las personas autorizadas, y utilizando herramientas de código abierto.

En el mundo del desarrollo de sistemas web existen diferentes herramientas creadas por organizaciones, las cuales pueden ser usadas de manera gratuita, una de esas herramientas es *WebRTC* la cual es una tecnología de código abierto y opera en *Mozilla*, *Google* y *Opera* desde hace un tiempo, nos permite realizar una comunicación web en tiempo real, realizar llamadas de video desde el navegador, sin necesidad de utilizar ningún tipo de *plugin*, simplemente usando *HTML5* y una *API* basada en *JavaScript*. Con *WebRTC*, puede agregar capacidades de comunicación en tiempo real a su aplicación que funciona sobre un estándar abierto. Admite video, voz y datos genéricos que se envían entre pares, lo que permite a los desarrolladores crear potentes soluciones de comunicación de voz y video (WebRTC, 2021).

ASP.NET SignalR es una biblioteca para desarrolladores de *ASP.NET* que simplifica el proceso de agregar funcionalidad web en tiempo real a las aplicaciones. La funcionalidad web en tiempo real es la capacidad de hacer que el código del servidor envíe contenido a los clientes conectados instantáneamente a medida que esté disponible, en lugar de que el servidor espere a que un cliente solicite nuevos datos (Microsoft, 2021).

2. Métodos.

Metodología de desarrollo web.

Para el desarrollo de esta aplicación se utilizó el marco de trabajo *Scrum*, la cual permite la óptima gestión de los proyectos, este marco de trabajo todos los miembros del equipo pueden ver trabajando los demás miembros, el objetivo para el que están trabajando y el progreso alcanzado. Scrum es un marco de trabajo liviano que ayuda a las personas, equipos y organizaciones a generar valor a través de soluciones adaptativas para problemas complejos. Scrum emplea un enfoque iterativo e incremental para optimizar la previsibilidad y controlar el riesgo. Scrum involucra a grupos de personas que colectivamente tienen todas las habilidades y experiencia para hacer el trabajo y compartir o adquirir dichas habilidades según sea necesario.

Iterativo e incremental.

La contribución técnica detrás de la mejora iterativa es desarrollar un sistema de software incrementalmente, permitiendo al desarrollador aprovechar lo que se va aprendiendo durante el desarrollo de las versiones tempranas, incrementales y entregables del sistema. El aprendizaje viene tanto del desarrollo como del uso del sistema, donde sea posible. Los pasos clave en el proceso consisten en empezar con una implementación sencilla de un subconjunto de los requisitos del *software* y mejorar iterativamente la evolución secuencial de versiones hasta que el sistema está implementado. En cada iteración, se hacen modificaciones en el diseño a la misma vez que añadimos nuevas funcionalidades (V. Basili y J. Turner, 1975).

En la ceremonia de *Scrum Spring Planining* se realizó la planeación del proyecto, se llegó a la conclusión que en un Spring de un mes el proyecto debería de estar terminado, se realizó la asignación de las tareas, la prioridad y la dificultad de la tarea utilizando la serie *Fibonacci*, todo quedó plasmado en el *Sprint Backlog* como se muestra en la figura 3.

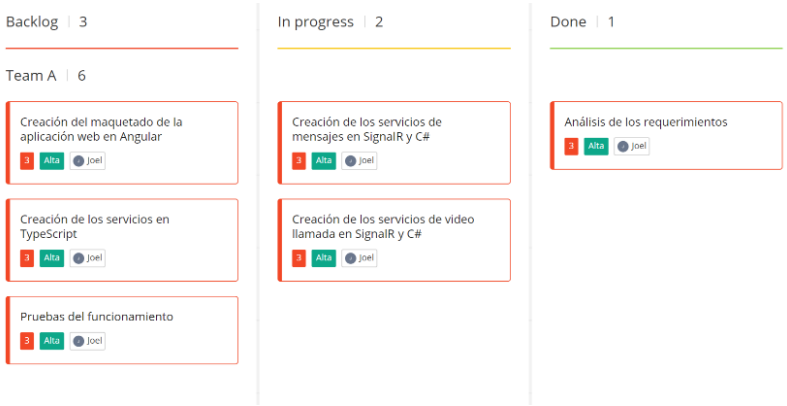

Figura 3. Sprint Backlog del proyecto.

Herramientas.

Las herramientas mencionadas a continuación corresponden a la etapa de Scrum Spring Planining:

- Microsoft Visual Studio Code. Herramienta que permite desarrollar, probar y desplegar código en diferentes lenguajes de programación.
- Microsoft Visual Studio 2019. Herramienta que permite desarrollar aplicaciones, sitios web, aplicaciones WPF, aplicaciones web, servicios web, *apps* de *Windows*, etc.

3. Desarrollo.

En la semana 1 se realizó el análisis de los requerimientos donde se creó el diagrama de contexto y el prototipo de la aplicación, se eligió el algoritmo de cifrado para la seguridad de los datos de entrada y salida y se creó el proyecto en Angular.

Diagrama de contexto.

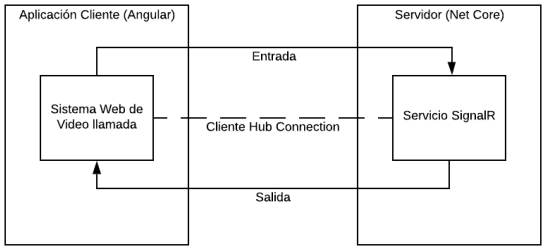

Figura 4. Diagrama de Contexto. Sistema de videoconferencia.

La figura 4 describe el flujo del funcionamiento de la aplicación web de videoconferencia, en primera instancia el cliente realiza la petición al servidor, si la conexión fue exitosa el servidor responde al cliente y se abre la conexión por *Sockets*. Cuando un nuevo usuario se conecta realiza la misma opción.

Diagrama de infraestructura de seguridad.

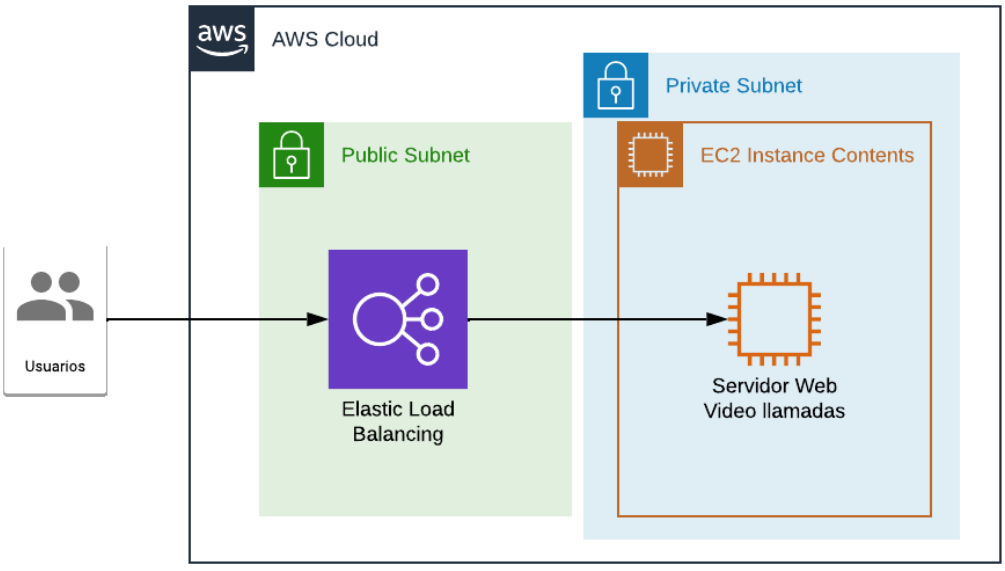

Figura 5. Diagrama de infraestructura de seguridad.

La figura 5 describe el flujo del diagrama de infraestructura de seguridad, el proyecto se podrá publicar en la nube, en caso se utilizó AWS (Amazon Web Services). Con AWS cuenta con la infraestructura global más segura de planta, todos los datos que fluyen en la red global de AWS y que interconectan los centros de datos y regiones se cifran de

manera automática en la capa física antes dejar nuestras instalaciones protegidas. También existen capas adicionales de cifrado. Por ejemplo, todo el tráfico de las interconexiones entre regiones de las VPC y las conexiones TLS del cliente o entre servicios. (AWS, s.f.).

Prototipo de la aplicación.

Figura 6. Prototipo de la aplicación videoconferencia.

Como se muestra en la figura 6 se creó el prototipo de la aplicación la cual cuenta con una caja de texto donde el alumno ingresará su nombre, cámara donde se podrá ver los alumnos conectados y la parte de los mensajes donde cada alumno podrá escribir y ver los mensajes enviados.

Creación de la aplicación.

Para el desarrollo web de esta aplicación, se utilizó para la parte de la vista *Angular* 10, *Bootstrap* y *CSS*, para la parte de la lógica se utilizó código abierto de *WebRTC* y *SignalR*. Para la creación del proyecto se necesita tener instalado *NodeJs* y después ejecutar el comando en consola como administrador: ng new videoconferenciaEscolar.

Algoritmo de cifrado.

Para el desarrollo de la aplicación se implementar el cifrado y descifrado *AES* para el envío y recepción de los mensajes enviados por los alumnos, *AES* es un estándar de cifrado avanzado el cual el descifrado de los datos cifrados solo es posible cuando conoce la contraseña correcta, se necesita instalar un archivo *crypto.js*, como se muestra en el código 1 se utilizará un método el cual va a cifrar y descifrar los mensajes.

Código 1; signal-r.service.ts. Método de cifrado y descifrado de mensajes.

```
ConvertirTexto( conversion:string ) {
   if (conversion=== 'encriptar') {
      this.conversionEncryptOutput = CryptoJS.AES.encrypt(this.plainText.trim(),
         this.encPassword.trim()).toString();
   }
   else {
      this.conversionDecryptOutput = CryptoJS.AES.decrypt(this.encryptText.trim(),
         this.decPassword.trim()).toString(CryptoJS.enc.Utf8);
   }
}
```

Desarrollo de la aplicación.

En la semana 2 y 3 se realizó la programación del proyecto el cual consta de 2 módulos importantes, se utilizó el diagrama de contexto el cual consta del código que programará la vista del alumno y la parte de la lógica de programación. El código 7 nos muestra el código en C# donde implementamos SignalR para la comunicación por sockets, la correcta gestión de los usuarios conectados, el envío y recepción de los mensajes.

Código 2; WebRtcHub.cs. Código en C# para la comunicación con SignalR.

```
this._hubConnection.on('receiveSignal', async (user: IUser, signal: string) => {
   await this.newSignal(user, signal);
});
this._hubConnection.on('MessageReceived', (data: any) => {
   this.messageReceived.emit(JSON.parse(data));
});

public mandarMensaje(userName: string, mensaje: string): void {
   this._hubConnection.invoke('NewMessage', userName, mensaje);
   delete this._connections[partnerClientId];
}
```

La figura 7 nos muestra la estructura del proyecto y parte del código fuente de la aplicación.

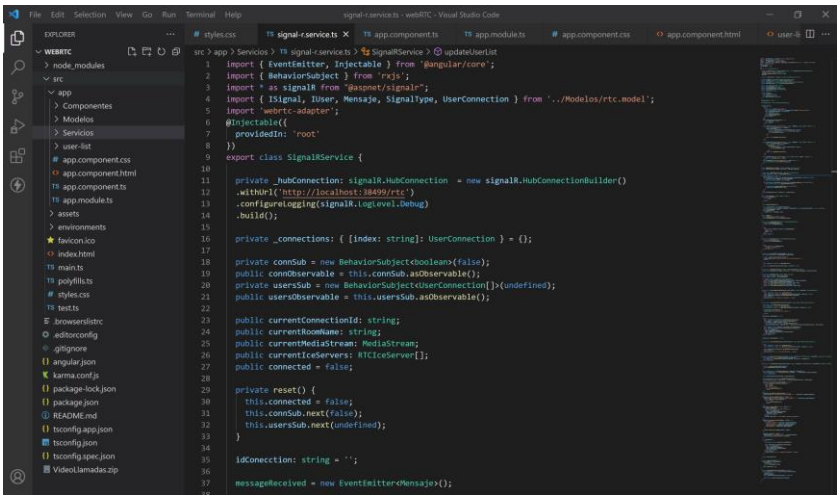

Figura 7. Código fuente de la aplicación web videoconferencia.

Resultados.

En la figura 8 se muestra la página web de inicio donde los alumnos podrán ingresar su nombre de usuario, automáticamente comenzara la sincronización con los alumnos y la conexión al servidor.

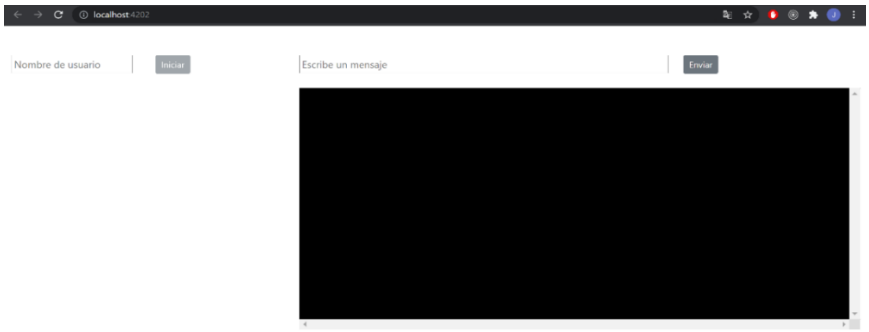

Figura 8. Página Web de inicio.

En la figura 9 se muestra la conexión exitosa al servidor, el alumno podrá mandar sus mensajes, escuchar y ver a los alumnos conectados.

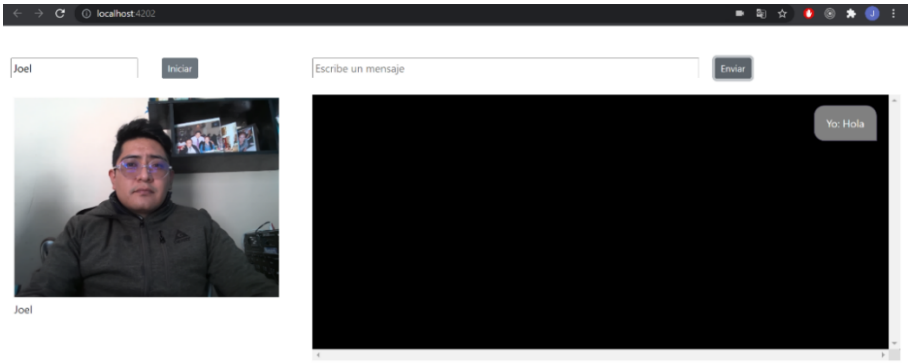

Figura 9. Primer alumno conectado**.**

En la figura 10 se muestra la comunicación entre dos o más alumnos y también el envío y recepción de mensajes de forma exitosa.

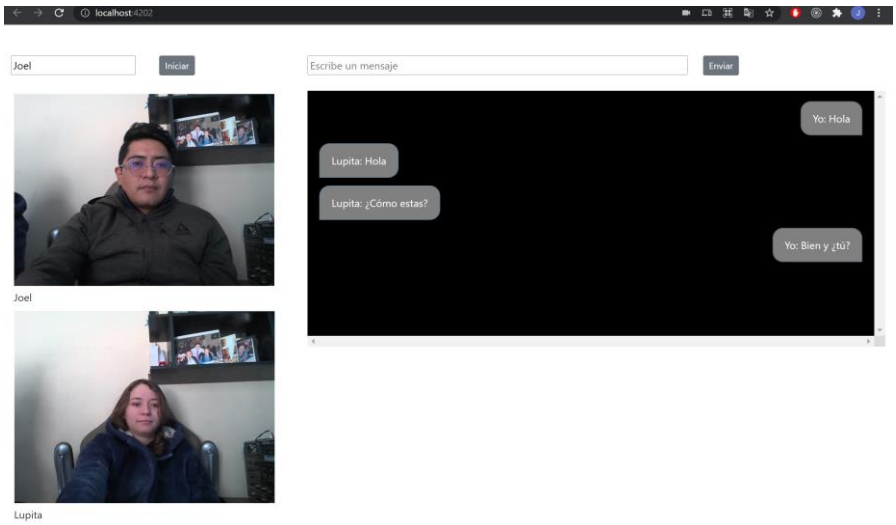

Figura 10. Dos o más alumnos conectados.

Conclusiones.

La aplicación web desarrollada en este trabajo no utiliza ninguna información personal de los alumnos, los ciberdelincuentes no podrán tener acceso a los mensajes compartidos por el *chat* ya que la información es cifrada, la aplicación web puede ser alojada en un servidor de una institución educativa o en la nube. Con todas las características antes mencionadas esta aplicación es muy segura y se evitará el robo de información.

El funcionamiento del proyecto es óptimo y se puede realizar mejoras como agregar un administrador que pueda silenciar micrófonos, apagar encender la cámara y una acción que pueda desconectar a los usuarios. El marco de trabajo *Scrum* permitió la óptima gestión de las tareas a realizar, el proyecto fue terminado en tiempo y forma establecido en la planeación del *Spring*. La metodología Iterativo e incremental permitió tener código funcional al término de cada *Sprint*.

Agradecimientos.

Se agradece a la universidad DaVinci y las autoridades que trabajan en la maestría en sistemas computacionales por todo el apoyo brindado en el desarrollo del presente proyecto. Se agradece al CONACYT por la beca otorgada por todos los semestres de la maestría en sistemas computaciones.

Referencias Bibliográficas.

AWS (s.f.). *Seguridad en la nube de AWS*. Obtenido de https://aws.amazon.com/es/security/

Diazgranados, H. (2021). Ciberataques en América Latina crecen un 24% durante los primeros ocho meses de 2021. Obtenido de https://latam.kaspersky.com/blog/ciberataques-en-america-latina-crecen-un-24-durante-los-primeros-ocho-meses-de-2021/22718/

El tiempo (3 de abril de 2020). *Eric Yuan: la historia del increíble ascenso del creador de Zoom.* Obtenido de https://www.eltiempo.com/tecnosfera/novedades-tecnologia/zoom-quien-es-el-multimillonario-fundador-de-plataforma-de-videollamadas-480754

El Universal (4 de Abril de 2021). *Descubren vulnerabilidad en Zoom que podría ponerte en peligro*. Obtenido de https://www.eluniversal.com.mx/techbit/zoom-descubren-vulnerabilidad-en-que-podria-ponerte-en-peligro

García, R. (10 de Enero de 2022). *Detectadas múltiples vulnerabilidades en Microsoft Teams*. Obtenido de https://unaaldia.hispasec.com/2022/01/detectadas-multiples-vulnerabilidades-en-microsoft-teams.html#:~:text=Se%20han%20descubierto%20una%20serie,incluso%20acceder%20a%20servicios%20internos.

INEGI. (2021). Encuesta para la Medición del Impacto COVID-19 en la Educación (ECOVID-ED). Obtenido de https://www.inegi.org.mx/contenidos/investigacion/ecovided/2020/doc/ecovid_ed_2020_presentacion_resultados.pdf

Microsoft. (2021). Introduction to SignalR. Obtenido de: https://docs.microsoft.com/en-us/aspnet/signalr/overview/getting-started/introduction-to-signalr

Secretaría de comunicaciones y transportes. **(Agosto de 2020).** Guía de ciberseguridad para el uso de redes y dispositivos de telecomunicaciones en apoyo a la educación. Obtenido de https://www.gob.mx/cms/uploads/attachment/file/570011/10082020_Gui_a_de_ciberseguridad_en_apoyo_a_la_educacio_n_-_VF_para_publicar.pdf

V. Basili y J. Turner (1975). "Iterative Enhancement: A Practical Technique for Software Development," IEEE Trans. Software Eng. pp. 390-396

WebRTC. (2021). Comunicación en tiempo real para la web. Obtenido de: https://webrtc.org/

Información de los autores.

Joel Ruben Regalado Romero, culminó la maestría en sistemas computacionales por la universidad DaVinci, cuenta con las certificaciones en Scrum Master y Scrum Developer por Certiprof, actualmente se desempeña como desarrollador web en la empresa Grupo Salinas desde el 2019.