	UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA			
	Documento FORMATO HOJA DE RESUMEN PARA TRABAJO DE GRADO	Código F-AC-DBL-007	Fecha 10-04-2012	Revisión A
	Dependencia DIVISIÓN DE BIBLIOTECA	Aprobado SUBDIRECTOR ACADEMICO		Pág. 1(136)

RESUMEN – TRABAJO DE GRADO

AUTORES	DANILO ANDRES ASCANIO LOBO		
FACULTAD	DE INGENIERIAS		
PLAN DE ESTUDIOS	INGENIERIA DE SISTEMAS		
DIRECTOR	EDWIN BARRIENTOS AVENDAÑO		
TÍTULO DE LA TESIS	DESARROLLO DEL APLICATIVO WEB MYBC DE LA EMPRESA TMSOFT UTILIZANDO VUE.JS Y EXPRESS.JS		
RESUMEN (70 palabras aproximadamente)			
<p>EL PRESENTE DOCUMENTO CONTIENE EL DESARROLLO DE UN APLICATIVO WEB PARA LA EMPRESA TMSOFT S.A.S QUE LE PERMITIRA GESTIONAR LAS LLAMADAS, COMO LO ES REALIZAR, TERMINAR Y GRABAR LLAMADAS, CREAR EDITAR Y ELIMINAR NOTAS Y CONTACTOS, DONDE PUEDE TAMBIEN REALIZAR VIDEOS LLAMADAS POR MEDIO DE UNA INTERFAZ TELEFONICA DESDE UNA PAGINA WED POR MEDIO DE DOS FRAMEWORK COMO LO ES VUE.JS PARA EL FRONTEND Y EXPRESS.JS PARA EL BACKEND.</p>			
CARACTERÍSTICAS			
PÁGINAS: 64	PLANOS: 0	ILUSTRACIONES:	CD-ROM:



Vía Acolsure, Sede el Algodonal, Ocaña, Colombia - Código postal: 546552
Línea gratuita nacional: 01 8000 121 022 - PBX: (+57) (7) 569 00 88 - Fax: Ext. 104
info@ufpso.edu.co - www.ufpso.edu.co

**DESARROLLO DEL APLICATIVO WEB MYBC DE LA EMPRESA TMSOFT S.A.S
UTILIZANDO VUE.JS Y EXPRESS.JS**

AUTOR:

DANILO ANDRES ASCANIO LOBO

**Trabajo de Grado presentado como requisito para optar al título de Ingeniero de
Sistemas**

Director

Ing. EDWIN BARRIENTOS AVENDAÑO

UNIVERSIDAD FRANCISCO DE PAULA

SANTANDER FACULDADE DE INGENIERIAS

INGENIERIA DE SISTEMAS

Ocaña, Colombia

Abril, 2021

Índice

Índice	5
Resumen.	10
Introducción.....	11
1. Desarrollo de un Aplicativo web MYCB de la empresa TMSOF S.A.S utilizando Vue.js y Express.js.....	1
1.1 Tmsoft.....	1
1.1.1 Misión.....	1
1.1.2 Visión.....	1
1.1.3 Objetivos de la empresa.....	1
1.1.3.1 Objetivo General.....	1
1.1.3.2 Objetivos Específicos.	2
1.1.4 Estructura organizacional.....	2
1.1.5 Dependencia y/o proyecto asignado.	2
1.2 Diagnóstico inicial de la dependencia asignada	3
1.2.1 Planteamiento del problema.	4
1.3 Objetivos de la pasantía	5
1.3.1 General.....	5
1.3.2 Específicos.....	5
1.4 Actividades a desarrollar.....	5
2. Enfoques referenciales.....	7
2.1 Enfoque conceptual	7
2.1.1 Aplicativo Web.....	7
2.1.2 Node.js.....	7
2.1.3 Npm	8
2.1.4 Express.js.....	8
2.1.5 Vue.js.....	8
2.1.6 BootstrapVue	9
2.1.7 Sass	9
2.1.8 Adobe Xd.....	10

2.1.9 Github.....	10
2.1.10 Telefonía Ip.....	10
2.2 Enfoque legal.....	11
3.Informe de Cumplimiento del Trabajo.....	13
3.1 Presentación de Resultados.....	13
Objetivo 1. Definir requerimientos del sistema de acuerdo las necesidades de la empresa.	13
3.1.1 Planificación del Sprint.....	13
3.1.1.1. Roles de Scrum.....	13
3.1.1.1 Historias de usuario épicas.....	13
3.1.1.2 Product Backlog.....	14
3.1.1.3 Product Backlog del Proyecto.....	15
3.1.1.4 Requerimientos no funcionales.....	18
3.1.1.5 Sprint Backlog.....	18
3.1.2 Objetivo 2. Diseñar las interfaces de usuario de acuerdo a estándares de usabilidad y ergonomía para mejorar la experiencia de usuario mediante el framework vue.js....	25
3.1.2.1 Maquetación y diseño del aplicativo web Mybc.....	25
3.1.2.2 Evaluación del diseño por parte de coordinador de Desarrollo.....	32
3.1.2.3 Corrección del diseño.....	32
3.1.3 Objetivo 3. Realizar las pruebas necesarias para verificar la calidad de las diferentes funcionalidades del aplicativo soportado bajo el framework express.js.....	33
3.1.3.1 Codificación de los componentes para el aplicativo MYBC.....	33
3.1.3.2 Evaluación del api por parte del coordinador de Desarrollo.....	45
3.1.3.3 Corrección en el diseño del api.....	52
3.1.3.4 Realización de informe final.....	52
4 .Diagnóstico Final.....	53
5. Conclusiones.....	54
6.Recomendaciones.....	56
7. Referencias.....	57

Lista de figuras

Figura 1 Estructura Organizacional Tmsoft. Fuente: Autor de la pasantía.	2
Figura 2. Loguin de Mybc para el acceso a la aplicación. Fuente: Autor de la pasantía.	25
Figura 3. Interfaz principal (Phone) del aplicativo web. Fuente: Autor de la pasantía.	26
Figura 4. Interfaz Contactos (Phone-1) listado contactos. Fuente: Autor de la pasantía.	26
Figura 5. la interfaz Dider (Phone-2) Cargar archivos Fuente: Autor de la pasantía.	27
Figura 6. Interfaz Dider (Phone-3) Lanzar llamadas. Fuente: Autor de la pasantía.	27
Figura 7. Interfaz Notas (Phone-4). Fuente: Autor de la pasantía.	28
Figura 8. Interfaz Notas Usuario (Notas). Fuente: Autor de la pasantía.	28
Figura 9. Interfaz estado de llamadas (Phone-5). Fuente: Autor de la pasantía.	29
Figura 10. Interfaz Conferencia (Conferencia). Fuente: Autor de la pasantía.	29
Figura 11. Interfaz sala de conferencia (Conferencia-1). Fuente: Autor de la pasantía.	30
Figura 12. Interfaz Registro de llamadas (CDR). Fuente: Autor de la pasantía.	30
Figura 13. Interfaz Registro de Contactos Fuente: Autor de la pasantía.	31
Figura 14 . Interfaz Registro de voicemail (Buzón, maibox). Fuente: Autor de la pasantía.	31
Figura 15. Interfaz Siguieme (Buzón, maibox). Fuente: Autor de la pasantía.	32
Figura 16. Arquitectura del aplicativo web Mybc. Fuente: autor de la pasantía.	33
Figura 17 .Estructura del Frontend. Fuente. Autor de la pasantía.	37
Figura 18. Archivo de configuración api del aplicativo web. Fuente: Autor de la pasantía.	38
Figura 19. Archivos de configuración de la carpeta de Components. Fuente: Autor de la pasantía.	39
Figura 20. Archivos de configuración de la carpeta de locales. Fuente: Autor de la pasantía.	39
Figura 21. Archivos de configuración de la carpeta de locales. Fuente: Autor de la pasantía.	40
Figura 22. Archivos de configuración de la carpeta router. Fuente: Autor de la pasantía.	40
Figura 23. Archivos de configuración de la carpeta de views. Fuente: Autor de la pasantía.	40
Figura 24. Estructura del backend del aplicativo Mybc. Fuente. Autor de la pasantía.	41
Figura 25. Configuración del archivo. env, y modelo mysql para conexión a la base de datos. Fuente: Autor de la pasantía.	42
Figura 26. Configuración del archivo de Mybc para la autenticación. Fuente: Autor de la pasantía.	42
Figura 27. Configuración del archivo mybcController. Fuente: Autor de la pasantía.	43
Figura 28. Archivo de configuración del modelo Mybc del aplicativo web. Fuente: Autor de la pasantía.	43
Figura 29. Archivo de configuración router del aplicativo web Mybc. Fuente: Autor de la pasantía.	44
Figura 30. Configuración del archivo principal app.js. Fuente: autor de la pasantía.	44
Figura 31. Prueba de la interfaz de inicio de sección al aplicativo web desde el servidor. Fuente: Autor de la pasantía.	45
Figura 32. Interfaz de inicio con el usuario y contraseña correspondiente para el acceso al aplicativo. Fuente: Autor de la pasantía.	46
Figura 33. Interfaz principal después de inicio de sección. Fuente: Autor de la pasantía.	46
Figura 34. Realización de una llamada a un teléfono móvil. Fuente: Autor de la pasantía.	47
Figura 35. Llamada establecida con el teléfono móvil. Fuente: Autor de la pasantía.	47
Figura 36. Interfaz Registro de Contactos funcional desde el servidor Fuente: Autor de la pasantía.	48
Figura 37. Interfaz funcional de Conferencia (Conferencia). Fuente: Autor de la pasantía.	48
Figura 38. Inicio de la Interfaz sala de conferencia (Conferencia-1). Fuente: Autor de la pasantía.	49
Figura 39. Interfaz sala de conferencia después de iniciada la conferencia (Conferencia-1). Fuente: Autor de la pasantía.	49
Figura 40. Interfaz Siguieme implementada (Buzón, maibox). Fuente: Autor de la pasantía.	50
Figura 41. Interfaz Registro de voicemail implantada (Buzón, maibox). Fuente: autor de la pasantía.	50
Figura 42. Interfaz Registro de llamadas implementada (CDR). Fuente: autor de la pasantía.	51

Figura 43. Interfaz para el reinicio de logue después de un tiempo de no ser utilizado el aplicativo. 51

Lista de tablas

Tabla 1. Matriz DOFA.....	3
Tabla 2.Actividades a Desarrollar.....	6
Tabla 3.Roles de Scrum	13
Tabla 4. Historia de Usuario épica HUE01.....	13
Tabla 5.Historia de Usuario épica HUE02.....	14
Tabla 6.Historia de Usuario épica HUE03.....	14
Tabla 7.Historia de Usuario épica HUE04.....	14
Tabla 8. Product Backlog del Proyecto.....	15
Tabla 9. Sprint Backlog del sprint 1	19
Tabla 10.Sprint Backlog del Sprint 2.....	20
Tabla 11.Sprint Backlog del Sprint 3.....	21
Tabla 12.Sprint Backlog del Sprint 4.....	23
Tabla 13.Sprint Backlog del Sprint 5.....	24

Resumen.

El proyecto que se llevó a cabo en la empresa TMSOFT S.A.S , fundamentalmente en desarrollar un nuevo aplicativo web MYCB que pueda sustituir el actual, Un aplicativo que ya para la actualidad se encuentra obsoleto y presentado muchos inconvenientes a los usuarios que lo están utilizando, lo que hizo necesario el desarrollo y reestructuración del nuevo aplicativo web MYCB , tomando como base nuevas tecnológicas (framework , lenguajes de programación y plugins) para su construcción, posibilitando aún más la implantación de muchas más funciones que se requieren para satisfacer la demanda de los usuarios en la actualidad, para ello se evaluó la tecnología a utilizar, para este caso se escogió vue.js y bootstrapVue para el desarrollo del frontend y node.js express.js para el desarrollo del backend.

Introducción.

En la actualidad todo lo relacionado con las aplicaciones web está en un constante evolución y actualización, por lo que aparecen nuevas tecnologías que nos brindan tanto mejoras visuales que mejoran drásticamente la experiencia de sus usuarios en la utilización de los aplicativos como de su rendimiento en general.

Todas estas tecnologías nuevas conllevan un sin número inmenso de posibilidades, todo esto hace que los desarrolladores de aplicaciones web piensen en otras soluciones a las dificultades que presentan sus proyectos, por eso en la empresa TMSOF se presentó la necesidad de desarrollar y rediseñar el aplicativo con que cuenta la empresa llamado MYBC, un aplicativo que fue constituido y desarrollado con un tecnología que para su momento estaba a la vanguardia pero que en la actualidad tanto el rendimiento, la parte visual y la usabilidad del aplicativo esta presentado mucho dificultades e inconvenientes a los usuarios finales como a la empresa.

Este proyecto se base en el desarrollar el aplicativo web MYBC con las nuevas tecnologías como lo es la implementación de node.js y express.js en el backend, y de veu.js y bootstrapVue en el desarrollo del frontend donde el aplicativo mejorara tanto en el rendimiento, la parte visual y en la usabilidad y la implementación de nuevas funciones que tendrá el usuario final.

1. Desarrollo de un Aplicativo web MYCB de la empresa

TMSOF S.A.S utilizando Vue.js y Express.js

1.1 Tmsoft

La empresa TMSOFT S.A.S. es una organización de carácter privado, dedicada al Desarrollo de Software y Especialista en Telecomunicaciones cuya prioridad es ofrecer soluciones de tipo informática al sector empresarial, diseñadas para mejorar la productividad y reducir costos, cuenta actualmente con sedes en Bogotá, Bucaramanga y una en Ocaña que recientemente empieza. Cuenta con servicios de profesionales expertos en tecnología, datacenters, soporte técnico, servidores dedicados y networking.

1.1.1 Misión.

Ofrecer soluciones integrales en tecnología y telecomunicaciones, adaptados a las necesidades que exige el mercado, con un equipo joven y dinámico altamente calificado, trabajando con un concepto nuevo de productos y servicios, brindándole a nuestros clientes una propuesta económica encaminada a lograr sus objetivos y estrategias empresariales.

1.1.2 Visión.

Ser una empresa sólida, reconocida y competitiva en el país, que ofrece al sector corporativo soluciones estándar y a la medida de cada operación, logrando incursionar en el mercado extranjero en el área de las telecomunicaciones.

1.1.3 Objetivos de la empresa.

1.1.3.1 Objetivo General.

Cumplir con los requisitos y necesidades de nuestros clientes, ofreciendo productos y

servicios a la medida de cada operación, a través de nuestro Sistema de Calidad mediante un proceso de Mejora Continua.

1.1.3.2 Objetivos Específicos.

- Proponer a nuestros clientes, una amplia gama de productos y servicios acorde a las necesidades que se tengan. Brindar un recurso humano competitivo y comprometido, basado en una selección rigurosa del personal.
- Ofrecer garantía a nuestros clientes por los servicios y/o productos suministrados durante el tiempo pactado, respaldados en la calidad y compromiso.
- Realizar mantenimiento a los equipos y herramientas para garantizar su óptimo estado en ejecución de contratos.
- Aumentar el número de clientes para nuestra empresa MYBC.
- Mejorar continuamente la eficacia del Sistema de Gestión de la Calida.

1.1.4 Estructura organizacional.



Figura 1 Estructura Organizacional Tmssoft. Fuente: Autor de la pasantía.

1.1.5 Dependencia y/o proyecto asignado.

La dependencia de desarrollo de software es un área de la empresa TMSOF S.A.S en la cual se lleva cabo el diseño de la interface (UI/UX) y funcionalidades del lado del servidor (API, Base de datos), con los cuales empresa busca brindar soluciones optimas y sencillas para satisfacer las necesidades de sus clientes.

1.2 Diagnóstico inicial de la dependencia asignada

Tabla 1.

Matriz DOFA

	Fortalezas	Debilidades
Diagnóstico inicial	1. Ambiente caracterizado por la colaboración mutua y el aprendizaje constante. 2. Asignación del trabajo al recurso humano acorde con las fortalezas que se muestran. 3. Infraestructura propia. 4. Producto innovador. 5. Cuenta con servidores y networking.	1. No existe documentación acerca de los errores presentados al momento del Desarrollo. 2. No cuentan con un Sistema de Información que permita la Distribución Automática de Llama
Oportunidades	Estrategia FO	Estrategia DO
1. Es una de las empresas de Desarrollo de Software en la región que ofrece este tipo de servicio, con estas tecnologías. 2. El aprendizaje en distintas áreas de aplicación de la carrera. 3. Muchas empresas tienen la necesidad de los Sistemas de Información. 4. Aumento considerable	1. Aprovechamiento al máximo de cada una de las facultades y habilidades del personal. 2. Mejorar de manera escalable la infraestructura y tecnología para innovar con productos. 3. Adquirir nuevos profesionales y doneos para el fortalecimiento de las áreas de desarrollo y marketing	1. Fortalecer la documentación, como método de aprendizaje más rápido. 2. Implementación de metodologías ágiles para desarrollo de proyectos.

desarrollo de Software a nivel nacional e internacional.

Amenazas	Estrategia FA	Estrategia DA
1.La herramienta de desarrollo usada presenta una curva de aprendizaje demasiado pronunciada, con lo que el desarrollo, al inicio, suele ser un poco lento.	1.Mejora el tiempo de aprendizaje con la implementación de nuevas estrategias teniendo en cuenta las nuevas tecnologías.	1.Documentar acerca de los inconvenientes presentados, así en futuras ocasiones, será mucho más fácil resolverlos.
2.Surgimientos de nuevas Empresas dedicadas al desarrollo de aplicativos webs	2.Implementacion de nuevas estrategias de márketing para exponer la empresa a nivel nacional e internacional	2.General de manera eficiente un sistema de información para distribuir automáticamente las llamadas

1.2.1 Planteamiento del problema.

La empresa TMSOFT, cuenta con un aplicativo web MYBC que en el momento de su contricción fue hecha con el framework GWT el cual permite administrar de una forma básica todo lo correspondiente llamadas como lo es el poder generar una llamada o varias llamadas a un usuario o un determinado grupo específico de usuarios. gestionar toda la información de dichas llamadas, el aplicativo web MYBC ya en la actualidad tener muchas falencias y dificultades para todos sus usuarios ya que la mayoría de ellos reportan que el aplicativo presentas los siguientes problemas:

- AL enviar los eventos de gran tamaño en el servidor llegaba el momento donde se detenía, dejaba de funcionar lo que produce una denegación de servicio como de web como de web Job (servicios telefónicos).
- El diseño del aplicativo en el frontend hace que los usuarios tengan problemas de usabilidad, el diseño hace que sea el usuario se demore en comprender y entender de

manera rápida el funcionamiento del aplicativo debido que es emborrazo y complicado de manejar.

- El rendimiento del aplicativo a trascurso del tiempo ha disminuido ya que se demora en realizar sus funcionalidades debido a que el aplicativo ha tenido más usuarios.

Por tanto, la empresa TMSOFT quiere mejorar el aplicativo web para mejorar su rendimiento, usabilidad, confiabilidad y dar soporte a muchos más usuarios buscando que sea un desarrollo escalable, que cuente con una interfaz agradable y más sencillo para el uso de sus usuarios.

1.3 Objetivos de la pasantía

1.3.1 General.

Desarrollar un aplicativo web MYBC de la empresa TMSOFT utilizando vue.js y express.js.

1.3.2 Específicos.

- Definir requerimientos del sistema de acuerdo a las necesidades de la empresa para el desarrollo de la aplicación.
- Diseñar las interfaces de usuario de acuerdo a estándares de usabilidad y ergonomía para mejorar la experiencia de usuario mediante el framework vue.js.
- Realizar las pruebas necesarias para verificar la calidad de las diferentes funcionalidades del aplicativo soportado bajo el framework express.js.

1.4 Actividades a desarrollar.

Tabla 2.

Actividades a Desarrollar

Objetivo General	Objetivo Específico	Actividades Desarrollar
	Definir requerimientos del sistema de acuerdo a las necesidades de la empresa	1. Recolección y análisis de la información para determinar requerimientos del software
Desarrollo del aplicativo web MYBC de la empresa TMSOF utilizando vue.js y express.js.	Diseñar las interfaces de usuario de acuerdo a estándares de usabilidad y ergonomía para mejorar la experiencia del usuario mediante el framework vue.js.	1.Maquetación de interfaces. 2.Evaluación de diseño por parte de coordinador de Desarrollo. Corrección de diseño
	Realizar las pruebas necesarias para verificar la calidad de las diferentes funcionalidades del aplicativo soportado bajo el framework express.js.	1.Codificación de los componentes para el aplicativo MYBC. 2.Evaluación del api por parte del coordinador de Desarrollo. 3.Corrección en el diseño del api. 4.Realización de informe final

2. Enfoques referenciales.

2.1 Enfoque conceptual

En el desarrollo del proyecto es necesario tener en consideración los siguientes conceptos y herramientas utilizadas para el desarrollo del proyecto, dará a conocer las tecnologías que fueron implementadas.

2.1.1 Aplicativo Web

Las aplicaciones Web reciben este nombre porque se ejecutan en la Internet. Es decir que los datos o los archivos en los que trabajas son procesados y almacenados dentro de la Web. Estas aplicaciones, por lo general, no necesitan ser instaladas en tu computador. El concepto de Aplicaciones Web está relacionado con el almacenamiento en la nube. Toda la información se guarda de forma permanente en grandes servidores de Internet y se envían a los dispositivos o equipos los datos requeridos en ese momento, quedando una copia temporal dentro de los equipos (Cordoba Vasquez, 2017).

2.1.2 Node.js

Node es un entorno de ejecución orientado a eventos asíncronos de JavaScript diseñado para construir aplicaciones de red escalables. Como las principales características de Node.js tenemos que permite desarrollar aplicaciones de red escalables, impulsada por eventos asíncronos, si no hay trabajo por hacer Node.js permanece dormido hasta que haya una conexión para que se active una llamada al servicio, todo esto es posible ya que se usa JavaScript del lado de servidor. (Haro, Guarda, Peñaherrera, & Quiña, 2019)

Node.js Es muy utilizado en los servidores debido que es JavaScript por su naturaleza asíncrona, es decir, que una petición es tratada cuando llegue totalmente la petición, cuando se

termine de utilizar el recurso, básicamente cuando se termine de ejecutar los eventos, El objetivo de Node.js es permitir escribir aplicaciones de web, de red y aplicaciones de real-time de una manera que soporte concurrencias muy altas por eso la naturaleza asíncrona, node.js delega peticiones a los procesos alternos.

2.1.3 Npm

Es un gestor de paquetes de código abierto escrito en JavaScript y para JavaScript. Es el gestor por defecto del entorno de ejecución node.js. Consiste en una línea de comandos que interactúa con un registro remoto y online para consumir y distribuir módulos de JavaScript presentes en ese registro y administrar las dependencias locales del proyecto. (Martínez Fernandez, Sanz, & Vado, 2020)

2.1.4 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. Con una gran variedad de métodos de utilidad HTTP y middleware a su disposición, crear una API robusta es rápido y fácil. (OpenJS, 2017)

2.1.5 Vue.js

Es un marco progresivo para la creación de interfaces de usuario. A diferencia de otros marcos monolíticos, Vue está diseñado desde cero para ser adoptable incrementalmente. La biblioteca principal se centra únicamente en la capa de vista y es fácil de recoger e integrar con otras bibliotecas o proyectos existentes. (vue.js, 2020)

Se basa en el patrón “modelo-vista-modelo de vista” para desacoplar la lógica de la interfaz. Además, posee un renderizado declarativo, es reactivo, permite el uso de “templates” y

cuenta con componentes y directivas para facilitar la programación. También cuenta con bibliotecas propias y oficiales como Router, para manejar las rutas de forma sencilla o Vuex , que en resumen sirve para manejar estados de la aplicación, pudiendo transmitirlos entre componentes. (Martínez Fernández, Sanz, & Vado, 2020).

Uso de Vue.js Framework para la construcción de aplicaciones de manera reactiva, en donde los elementos visuales reaccionan a través de los datos dinámicamente sin necesidad de realizar un renderizado de los elementos nuevamente, esto permite que la información pueda perdurar cierto tiempo. Vue.js también maneja el concepto de Single Page Application (Aplicación de una sola página), esta característica de Vue.js permite que nuestros webs renderizan mucho más rápido, creando una experiencia de usuario muy agradable. (Lopez Padilla & Arrieta Morelo, 2020)

2.1.6 BootstrapVue

Es una biblioteca de código abierto especializada en el diseño de aplicaciones web. Se encuentra escrito en HTML, CSS, JavaScript, Less y Saas. Contiene multitud de plantillas de diseño con sus componentes propios para usarlas y personalizarlas en pocas líneas de código. BootstrapVue es la adaptación de Bootstrap para Vue.js. Posee las mismas funcionalidades, pero adaptadas al framework, por lo que es requerido disponer de PortalVue, que no es más que un componente usado para renderizar otros componentes en cualquier parte del DOM. (Martínez Fernández, Sanz, & Vado, 2020)

2.1.7 Sass

Es un preprocesador CSS que facilita la creación y el mantenimiento de hojas de estilos. Ofrece la posibilidad de crear variables, funciones o la inclusión de otras hojas de estilo Sass, por lo que normalmente, las hojas de estilo Sass suelen ser más limpias y fácil de entender

(y por tanto de mantener) que hojas de estilo convencionales. Cuando son incluidas en la web, es necesario hacer una transformación a hoja de estilos CSS. (Carreño Villalba , 2016).

2.1.8 Adobe Xd

Software Adobe XD para el diseño y creación de prototipos para la experiencia de usuarios tanto para dispositivos móviles como web, Adobe XD que permite además simular interacciones y animaciones entre las pantallas de la aplicación, facilitando la percepción del cliente de cómo la aplicación se vería y como funcionaría una vez desarrollada. (Cáceres Sánchez, 2019).

2.1.9 Github.

Es uno de los sistemas de control de versiones más famosos que hay disponibles, permitiendo que todos los miembros de los distintos proyectos trabajen en paralelo de una forma eficaz. Para funcionar, GitHub hace uso del sistema de control de versiones de Git. Cuenta con herramientas tales como ramificaciones del proyecto, interfaz gráfica, posibilidad de dar marcha atrás a los cambios realizados o estadísticas y datos relevantes para los colaboradores de los repositorios. (Martínez Fernadez, Sanz, & Vado, 2020)

2.1.10 Telefonía Ip

La Telefonía sobre Internet (cuyo acrónimo es ToIP) hace posible que, entre otros servicios, la señal de voz viaje a través de Internet empleando el protocolo IP. Esto significa que se envía la señal de voz en forma digital, en paquetes de datos, en lugar de enviarla en forma analógica a través del par de hilos telefónicos utilizados en la telefonía convencional. Asociado a este servicio van otros como buzón de voz, enlace con correo electrónico, fax, etc. (Castro Navarro, 2017).

2.2 Enfoque legal

2.2.1 Ley 1341 de 2009 – MinTIC

TITULO I, disposiciones generales, capítulo I - PRINCIPIOS GENERALES

Art. 2 Principios orientadores

Prioridad al acceso y uso de las Tecnologías de la Información y las Comunicaciones. El Estado y en general todos los agentes del sector de / las Tecnologías de la Información y las Comunicaciones deberán colaborar, dentro del marco de sus obligaciones, para priorizar el acceso y uso a las Tecnologías de la Información y las Comunicaciones en la producción de bienes y servicios, en condiciones no discrimina. (MinTIC, 2021).

- **Protección de los derechos de los usuarios.** El Estado velará por la adecuada protección de los derechos de los usuarios de las Tecnologías de la Información y de las Comunicaciones, así como por el cumplimiento de los derechos y deberes derivados del Habeas Data, asociados a la prestación del servicio. Para tal efecto, los proveedores y/u operadores directos deberán prestar sus servicios a precios de mercado y utilidad razonable, en los niveles de calidad establecidos en los títulos habilitantes o, en su defecto, dentro de los rangos que certifiquen las entidades competentes e idóneas en la materia y con información clara, transparente, necesaria, veraz y anterior, simultanea y de todas maneras oportuna para que los usuarios tomen sus decisiones. (MinTIC, 2021)
- **Promoción de la Inversión.** Todos los proveedores de redes y servicios de telecomunicaciones tendrán igualdad de oportunidades para acceder al uso del espectro y contribuirán al Fondo de Tecnologías de la Información y las Comunicaciones. (MinTIC, 2021).
- El mencionado artículo de la Ley 1341 de 2009 específicamente dice al respecto de las tarifas de estos servicios que:

Artículo 23.- Regulación de precios de los servicios de telecomunicaciones. Los proveedores de redes y servicios de telecomunicaciones podrán fijar libremente los precios al usuario. La Comisión de Regulación de Comunicaciones sólo podrá regular estos precios cuando no haya suficiente competencia, se presente una falla de mercado o cuando la calidad de los servicios ofrecidos no se ajuste a los niveles exigidos, lo anterior mediante el cumplimiento de los procedimientos establecidos por la presente Ley. (Rodríguez Bernal, 2018).

La seguridad informática nace a partir de la necesidad de asegurar los recursos de los sistemas de información de una compañía, y se puede definir como la disciplina que se enfoca en mantener la integridad, confidencialidad y disponibilidad de la información. En ese sentido cada país debe crear sus mecanismos cuanto a la ciberseguridad. Colombia es uno de los países líderes en Latinoamérica a través de su política nacional en ciberseguridad y ciberdefensa, además cuenta con la ley 1273 de 2009 que establece " la protección de la información y de los datos" y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones (Bravo León, 2019)

3. Informe de Cumplimiento del Trabajo

3.1 Presentación de Resultados

Objetivo 1. Definir requerimientos del sistema de acuerdo las necesidades de la empresa.

Actividad N°1.

Recolección y análisis de la información para determinar requerimientos del software.

Se determina las actividades a realizar en base a las pautas de la metodología ágil SCRUM

3.1.1 Planificación del Sprint.

3.1.1.1. Roles de Scrum.

Tabla 3.

Roles de Scrum

<u>Roles de Scrum</u>	
Product Owner	Jefe de la empresa: Alberto Cabrales
Scrum Master	Coordinador de la pasantía: Daniel Jimenez
Equipo de desarrollo	Pasante: Danilo Ascanio

3.1.1.1 Historias de usuario épicas

Tomando en cuenta los requerimos, se muestra las siguientes historias de usuario épicas.

Tabla 4.

Historia de Usuario épica HUE01

<u>Historias de usuario épicas</u>	<u>Código: HUE01</u>
Título:	Análisis de las tecnologías y estructura del aplicativo web
Descripción:	Estudiar y analizar las características a favor como las que pueda tener en contra de las tecnologías a implantar e identificar la mejor arquitectita para el aplicativo

Tabla 5.

Historia de Usuario épica HUE02

<u>Historias de usuario épicas</u>	<u>Código: HUE02</u>
Título: Desarrollo del backend del aplicativo web	
Descripción: Implantación del framework para el backend, crear los controladores, crear los modelos, crear las rutas, definir la seguridad, crear los modelos de la base de datos, la conexión a la base datos, realización de la lógica para realizar llamadas.	

Tabla 6.

Historia de Usuario épica HUE03

<u>Historias de usuario épicas</u>	<u>Código: HUE03</u>
Título: Desarrollo del frontend del aplicativo web	
Descripción: Diseño e implantación de las interfaces del aplicativo web para el correcto funcionamiento del aplicativo, creación de componentes, creación del plugins para la generación de todas las funciones correspondientes a las llamadas.	

Tabla 7.

Historia de Usuario épica HUE04

<u>Historias de usuario épicas</u>	<u>Código: HUE04</u>
Título: Pruebas y correcciones	
Descripción: Pruebas de rendimiento y uso, se realizarán pruebas del aplicativo mediante la aplicación Postman para verificar el correcto funcionamiento del api rest ful del aplicativo y se realizara pruebas para medio de usuarios para verificar el uso simple del aplicativo, Correcciones de cada interfaces y funcionamiento de ser necesario.	

3.1.1.2 Product Backlog.

Para gestionar de manera más eficiente las historias épicas se desarrolló una lista de las historias de usuario que se realizado En el desarrollo del aplicativo web Mybc. El Product Backlog es un listado de los requerimientos funcionales, un listado de todas las tareas a

desarrollar durante la ejecución del aplicativo de acuerdo con lo que se desea cumplir y con proyección a la realización de Sprints.

3.1.1.3 Product Backlog del Proyecto

Tabla 8.

Product Backlog del Proyecto

<u>Historia</u> <u>Épica</u>	<u>Id</u>	<u>Historia de Usuario</u>	<u>Esfuerzo</u>	<u>Importanci</u> <u>a</u>	<u>Tiemp</u> <u>o en</u> <u>Días</u>	<u>Sp</u> <u>rin</u> <u>t</u>
HUE01	HUE01-01	Análisis de las tecnologías para el desarrollo	Alta	100	2	
	HUE01-02	Implementación de la arquitectura del aplicativo	Alta	100	3	1
	HUE01-03	Creación del servicio de rest para la lógica de acceso al aplicativo web.	Media	90	6	
	HUE02-01	Desarrollar el plugin para la realización de las llamadas	Alta	100	7	
HUE02	HUE02-02	Creación del servicio rest para el registro de llamadas del aplicativo	Media	90	6	
	HUE02-03	Creación del servicio rest para el registro de contactos del aplicativo	Media	90	5	
	HUE02-04	Creación del servicio rest para el registro de Voicemail del	Media	90	5	

		aplicativo				
	HUE02-05	Creación del servicio rest para el registro de Sigüeme del aplicativo	Media	90	5	2
	HUE03-01	Diseñar e implementar la interfaz de Loguin para el acceso al aplicativo web	Media	90	5	
	HUE03-02	Diseñar e implementar la interfaz principal (Phone) del aplicativo web	Media	90	5	
HUE03	HUE03-03	Diseñar e implementar la interfaz Contactos (Phone-1) listado contactos de llamadas del aplicativo web	Media	90	5	3
	HUE03-04	Diseñar e implementar la interfaz Dialer (Phone-2) Cargar archivos del aplicativo web	Media	90	5	
	HUE03-05	Diseñar e implementar la interfaz Dialer (Phone-3) Lanzar llamadas del aplicativo web.	Media	90	5	
	HUE03-06	Diseñar e implementar la interfaz Notas (Phone-4) del	Media	90	5	

		aplicativo web.				
HUE03-07	Diseñar e implementar la interfaz Notas Usuario (Notas) del aplicativo web.	Media	90	5		
HUE03-08	Diseñar e implementar la interfaz estado de llamadas (Phone-5) del aplicativo web.	Media	90	6		
HUE03-09	Diseñar e implementar la interfaz Conferencia (Conferencia)del aplicativo web.	Media	90	6	4	
HUE03-10	Diseñar e implementar la interfaz sala de conferencia (Conferencia-1) del aplicativo web.	Media	90	6		
HUE03-11	Diseñar e implementar la interfaz Registro de llamadas (CDR)del aplicativo	Media	90	6		
HUE03-12	Diseñar e implementar la interfaz Registro de Contactos (Contactos)del aplicativo	Media	90	6		
HUE03-13	Diseñar e implementar la interfaz Registro de voicemail (Buzón,	Media	90	6		

		maibox)del aplicativo			5
	HUE03 -14	Diseñar e implementar la interfaz Sigueme (Buzón, maibox)del aplicativo	Media	90	6
	HUE04 -01	Realización de pruebas con la aplicación Potsman	alta	100	6
HUE04	HUE04 -02	Corrección de error del aplicativo	Alta	100	6
	HUE04 -03	Construcción del informe final	bajo	50	4

3.1.1.4 Requerimientos no funcionales.

Seguridad de los datos: El aplicativo web se maneja la seguridad de aplicativo, los datos de los usuarios serán manipulados únicamente por administrador.

Funcionalidad: Se refiere a que el aplicativo web satisfaga los requerimientos definidos.

Usabilidad: Se refiere a que el aplicativo web en la operación y manejo del aplicativo sea de manera fácil, para que el usuario aprenda rápido.

Confiabilidad: El aplicativo web será tolerante a fallos y que el usuario pueda acceder al aplicativo en el momento que lo desee.

3.1.1.5 Sprint Backlog

Se especifican las tareas asignadas a cada historia de usuario para el Sprint

Tabla 9.

Sprint Backlog del sprint 1

<u>Sprint</u>	<u>Historia de Usuario</u>	<u>Tarea</u>
1	HUE01-01	<ul style="list-style-type: none"> • Estudiar los Framework que hay en la actualidad y que pueden ser utilizados y más viables para el aplicativo web. • Definir los Framework que se utilizaran para el backend y para el frontend. • Implementar las tecnologías escogidas para el desarrollo del aplicativo.
	HUE01-02	<ul style="list-style-type: none"> • Definir la arquitectura más apropiada para el aplicativo web • Definir las tecnologías para conexión para la comunicación entre el backend y el frontend. • Implementación de la arquitectura para el aplicativo. • Establecer la conexión del backend con la base de datos y la del backend con la frontend del aplicativo
	HUE01-03	<ul style="list-style-type: none"> • Crear la validación para el ingreso de usuario mediante un middleware que verifique el usuario esta registrado en el aplicativo VOICE de la empresa, verificar las contraseña para dar el acceso o no a al aplicativo • Generar un toquen para mejorar la seguridad del aplicativo. • Determinar que el toque se vuelva inactivo en un tiempo determinado y se pida de nuevo el inicio de sesión.

Se especifican las tareas asignadas a cada historia de usuario para el Sprint 2.

Tabla 10.

Sprint Backlog del Sprint 2

<u>Sprint</u>	<u>Historia de Usuario</u>	<u>Tarea</u>
	HUE02-01	Desarrollar el plugin para la realización de las llamadas donde tenga todas las características de un teléfono móvil actual
	HUE02-02	Crear los archivos de configuración del modelo, del controlador y de la lista de rutas para que el usuario puede consultar los contactos que tiene y las acciones realizadas con cada uno de ellos.
	HUE02-03	Crear los archivos de configuración del modelo, del controlador y de la lista de rutas para la consulta de contactos.
	HUE02-04	<ul style="list-style-type: none"> • Crear los archivos de configuración del modelo, del controlador y de la lista de rutas para la consulta de contactos. • Crear los archivos de configuración del modelo, del controlador y de la lista de rutas para la consulta de los audios. • Crear los archivos de configuración del modelo, del controlador y de la lista de rutas para la consulta de las extensiones que contengan llamadas de los usuarios.
2	HUE02-05	<ul style="list-style-type: none"> • Crear los archivos de configuración del modelo, del controlador y de la lista de rutas para editar crear y listar de numero telefónicos de los contactos.
	HUE03-01	<ul style="list-style-type: none"> • Diseñar la interfaz de Loguin del aplicativo web • Codificación de la interfaz con el framework vue.js • Implementar la interfaz diseñada. • Crear los archivos del api para la conexión de la frontend con el backend para la validación de los datos que el usuario registre en los campos de usuario y clave. • Crear la acción que permita Notificar al usuario mediante un mensaje que el toquen ya no es válido
	HUE03-02	<ul style="list-style-type: none"> • Diseñar la interfaz principal (Phone) • Codificar la interfaz con su menú lateral, cabecera y sección principal • Implementar la interfaz diseñada. • Menú de contactos pase a la interfaz Contactos (Phone-1) cuando el usuario realice un clic en el icono de contactos.

-
- Menú de Dialer pase a la interfaz Dialer (Phone-2) Cargar archivos cuando el usuario realice un clic en el icono de Dialer.
 - Menú de Notas pase la interfaz Notas (Phone-4) Cargar las notas cuando el usuario realice un clic en el icono de Notas.
 - La interfaz del teléfono al momento de oprimir las teclas se escuche un sonido. Presente estados en el transcurso de las acciones de las llamadas.
-

Se especifican las tareas asignadas a cada historia de usuario para el Sprint 3.

Tabla 11.

Sprint Backlog del Sprint 3

<u>Sprint</u>	<u>Historia de Usuario</u>	<u>Tarea</u>
	HUE03-03	<ul style="list-style-type: none"> • Diseñar la interfaz Contactos (Phone-1). • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • La interfaz del teléfono al momento de oprimir las teclas se escuche un sonido. Presente estados en el transcurso de las acciones de las llamadas. • Listar los datos de los contactos. • En el listado de los contactos permitir la opción de realizar llamadas. • Regresar a la interface Phone si el usuario lo decida.
	HUE03-04	<ul style="list-style-type: none"> • Diseñar la interfaz Dialer (Phone-2) Cargar archivos del aplicativo web • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • Crear la opción de cargar contactos de un archivo de Excel. • Regresar a la interface Phone si el usuario lo decida. • La interfaz del teléfono al momento de oprimir las teclas se escuche un sonido. Presente estados en el transcurso de las acciones de las llamadas.

3

HUE03-05

-
- Diseñar la interfaz Dialer (Phone-3) Lanzar llamadas del aplicativo web
 - Codificar la interfaz con su menú lateral, cabecera y sección principal.
 - Implementar la interfaz diseñada.
 - Listar los contactos con su nombre y número telefónico que se carga del archivo Excel.
 - Poder lanzar la llamada a los contactos existentes.
 - La interfaz del teléfono al momento de oprimir las teclas se escuche un sonido. Presente estados en el transcurso de las acciones de las llamadas.
 - Regresar a la interface Phone si el usuario lo decida.

HUE03-06

-
- Diseñar la interfaz Notas (Phone-4) del aplicativo web
 - Codificar la interfaz con su menú lateral, cabecera y sección principal.
 - Implementar la interfaz diseñada.
 - Crear, eliminar y copiar notas con su respectiva fecha de edición.
 - Generará una nueva nota cada vez que el usuario realice un clic en el icono de notas.
 - La interfaz del teléfono al momento de oprimir las teclas se escuche un sonido. Presente estados en el transcurso de las acciones de las llamadas.
 - Regresar a la interface Phone si el usuario lo decida.

HUE03-07

-
- Diseñar interfaz Notas Usuario (Notas) del aplicativo web.
 - Codificar la interfaz con su menú lateral, cabecera y sección principal.
 - Implementar la interfaz diseñada.
 - Crear, eliminar y copiar notas con su respectiva fecha de edición.
 - Generará una nueva nota cada vez que el usuario realice un clic en el icono de notas.
 - Regresar a la interface Phone si el usuario lo decida.
-

Se especifican las tareas asignadas a cada historia de usuario para el Sprint 4.

Tabla 12.

Sprint Backlog del Sprint 4

<u>Sprint</u>	<u>Historia de Usuario</u>	<u>Tarea</u>
	HUE03-08	<ul style="list-style-type: none"> • Diseñar la interfaz estado de llamadas (Phone-5) del aplicativo web. • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada • La interfaz del teléfono con las funciones básicas de un teléfono y al momento de oprimir las teclas se escuche un sonido. Presente estados en el transcurso de las acciones de las llamadas. • Realizar y cancelar una llamada.
4	HUE03-09	<ul style="list-style-type: none"> • Diseñar la interfaz Conferencia (Conferencia)del aplicativo web. • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • Al crear el botón de inicio conferencia al darle clic pasara la reunión.
	HUE03-10	<ul style="list-style-type: none"> • Diseñar la interfaz sala de conferencia (Conferencia-1) del aplicativo web. • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Crear el plugin para realizar la conferencia.
	HUE03-11	<ul style="list-style-type: none"> • Diseñar la interfaz Registro de llamadas (CDR)del aplicativo • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada • Listar los contactos que se realizo llamadas estado de la mismas, si tiene notas relacionadas, duración y fecha y poder realizar llamadas.

HUE03-12	<ul style="list-style-type: none"> • Diseñar la interfaz Registro de Contactos (Contactos)del aplicativo. • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • Listar los contactos con sus nombres y numero telefónicos, con su respectiva nota y poder realizar llamadas
----------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Se especifican las tareas asignadas a cada historia de usuario para el Sprint 5.

Tabla 13.

Sprint Backlog del Sprint 5

Sprint	Historia de Usuario	Tarea
5	HUE03-13	<ul style="list-style-type: none"> • Diseñar la interfaz Registro de voicemail (Buzón, maibox) del aplicativo • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • Activar o desactivar los audíos del registro • Listar las llamadas con su respectivo audio, fecha también realizar llamadas.
	HUE03-14	<ul style="list-style-type: none"> • Diseñar la interfaz Sigüeme (Buzón, maibox) del aplicativo. • Codificar la interfaz con su menú lateral, cabecera y sección principal. • Implementar la interfaz diseñada. • Generar cinco opciones de Sigüeme para el aplicativo web • Activar o desactivar la opción de poder generar editar eliminar los Sigüeme del aplicativo.

HUE04-01	<ul style="list-style-type: none"> • Realización de pruebas con la aplicación Potsman. • Verificar el correcto funcionamiento del aplicativo del api rest del aplicativo. • Realizar llamadas para el correcto funcionamiento del plugin del teléfono.
HUE04-02	<hr/> <ul style="list-style-type: none"> • Corrección de error del aplicativo.
HUE04-03	<hr/> <ul style="list-style-type: none"> • Construcción del informe final

3.1.2 Objetivo 2. Diseñar las interfaces de usuario de acuerdo a estándares de usabilidad y ergonomía para mejorar la experiencia de usuario mediante el framework vue.js.

Actividades:

- Maquetación de interfaces.
- Evaluación del diseño por parte de coordinador de Desarrollo
- Corrección del diseño.

3.1.2.1 Maquetación y diseño del aplicativo web Mybc

Maquetación y diseño de la interfaz acceso al aplicativo web Mybc.



Figura 2. Loguin de Mybc para el acceso a la aplicación. Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz principal (Phone).

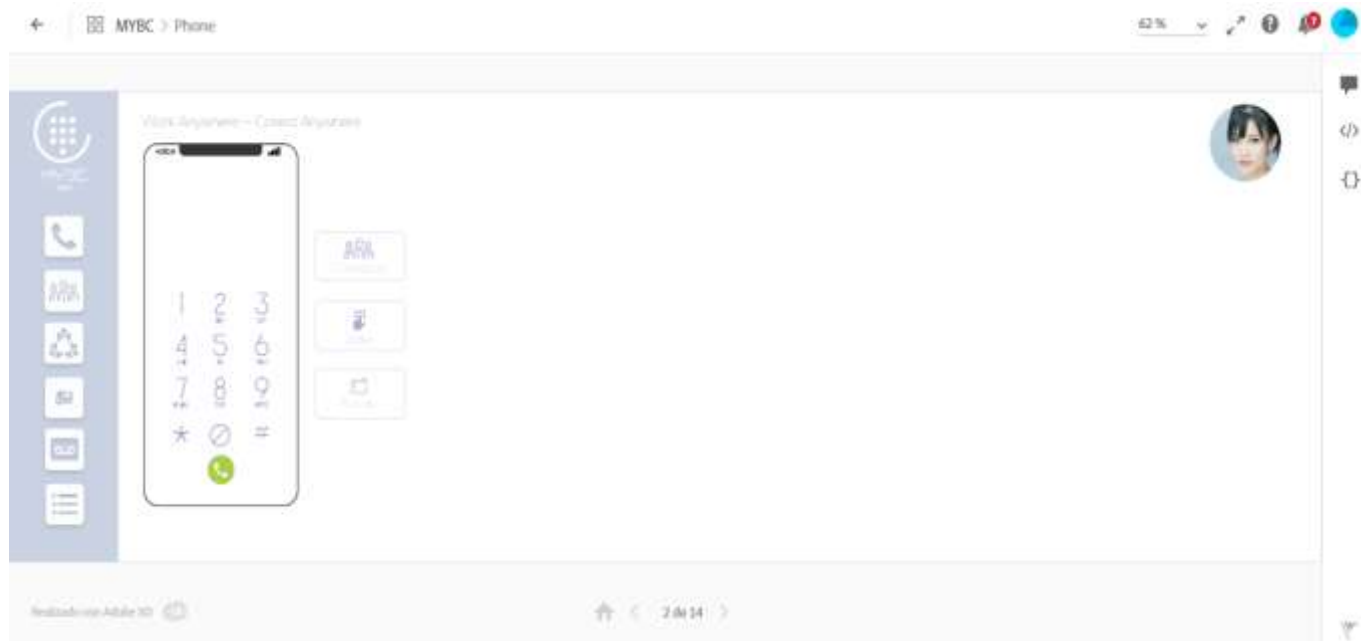


Figura 3. Interfaz principal (Phone) del aplicativo web. Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Contactos (Phone-1) listado contactos.



Figura 4. Interfaz Contactos (Phone-1) listado contactos. Fuente: Autor de la pasantía.

Maquetación y diseño interfaz Dider (Phone-2) Cargar archivos.



Figura 5. la interfaz Dider (Phone-2) Cargar archivos Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Dider (Phone-3) Lanzar llamadas.

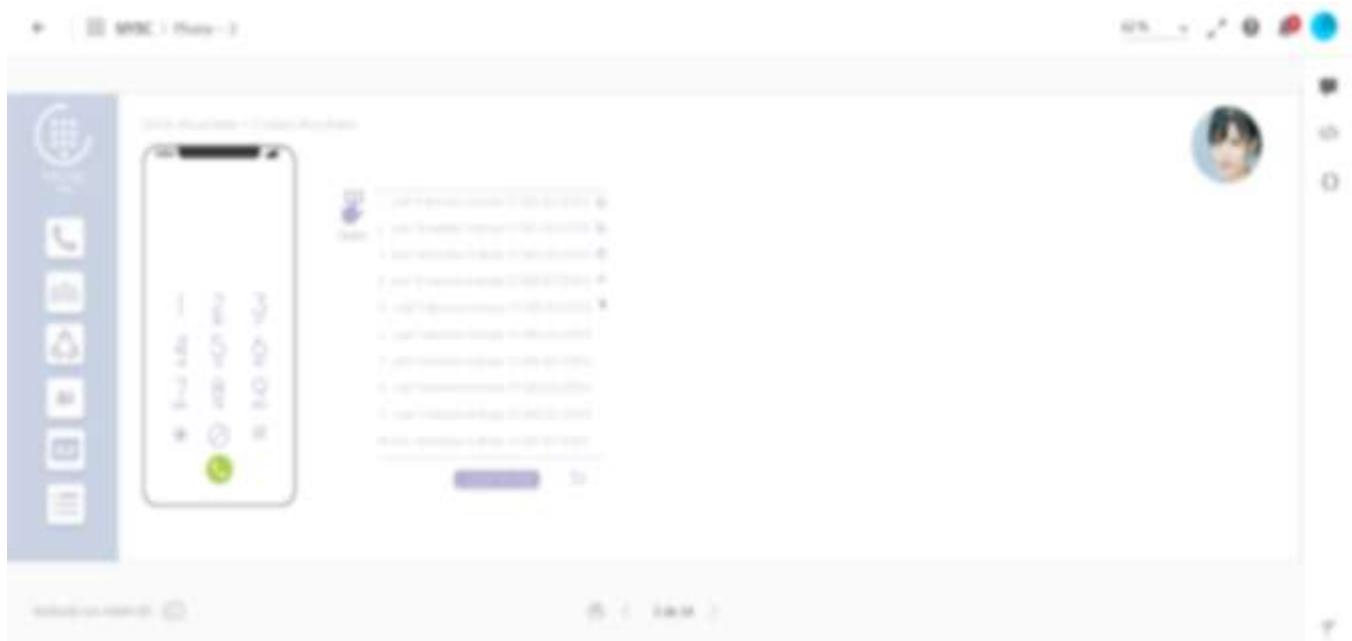


Figura 6. Interfaz Dider (Phone-3) Lanzar llamadas. Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Notas (Phone-4).



Figura 7. Interfaz Notas (Phone-4). Fuente: Autor de la pasantía.

Maquetación y diseño interfaz Notas Usuario (Notas).



Figura 8. Interfaz Notas Usuario (Notas). Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz estado de llamadas (Phone-5).

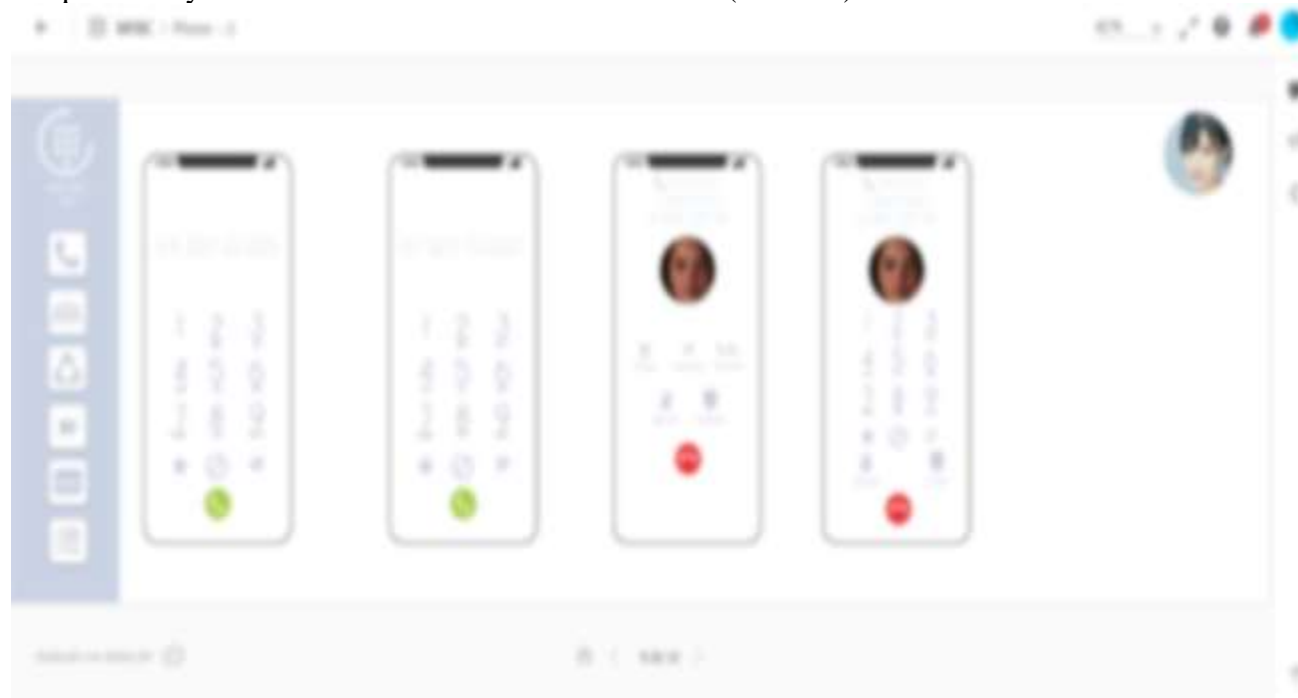


Figura 9. Interfaz estado de llamadas (Phone-5). Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Conferencia (Conferencia).



Figura 10. Interfaz Conferencia (Conferencia). Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz sala de conferencia (Conferencia-1).

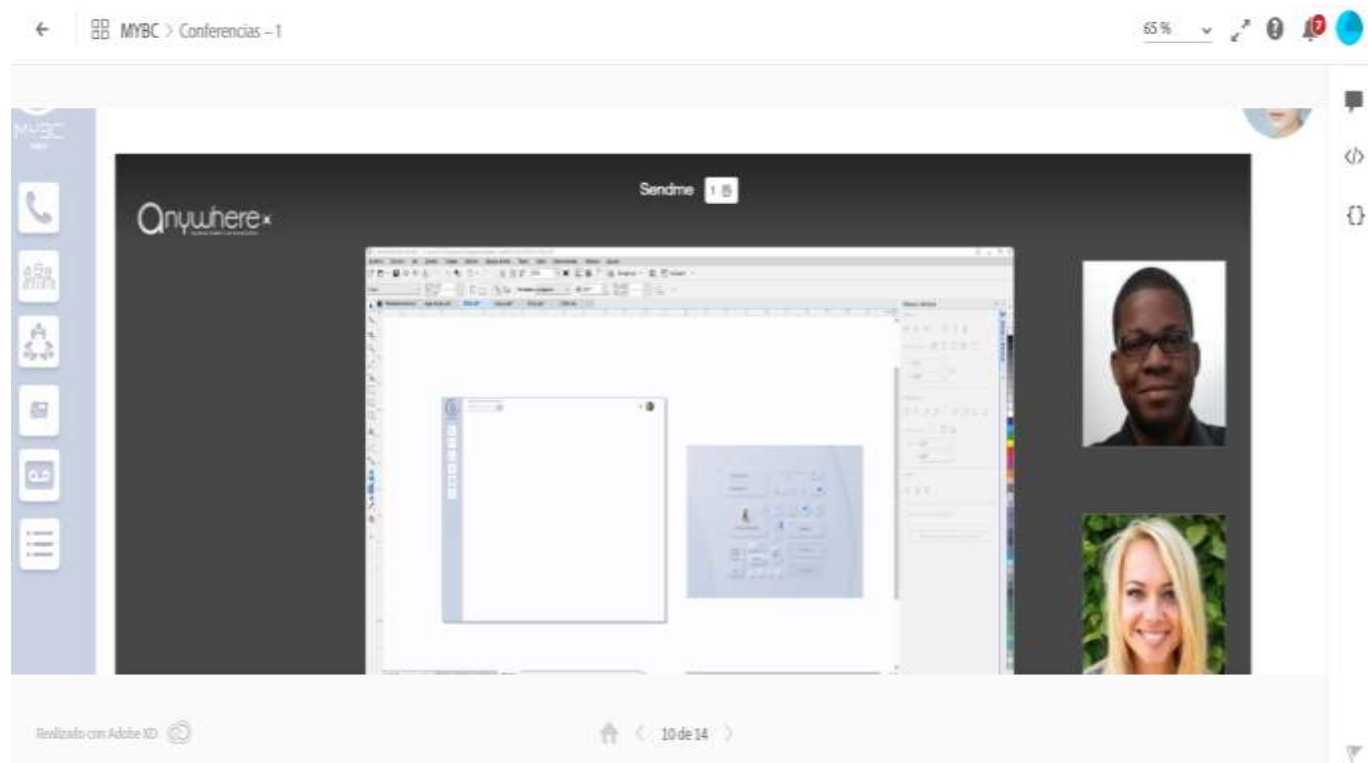


Figura 11. Interfaz sala de conferencia (Conferencia-1). Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Registro de llamadas (CDR).



Figura 12. Interfaz Registro de llamadas (CDR). Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Registro de Contactos.



Figura 13. Interfaz Registro de Contactos Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Registro de voicemail (Buzón, maibox).



Figura 14. Interfaz Registro de voicemail (Buzón, maibox). Fuente: Autor de la pasantía.

Maquetación y diseño de la interfaz Siguieme (Buzón, maibox).

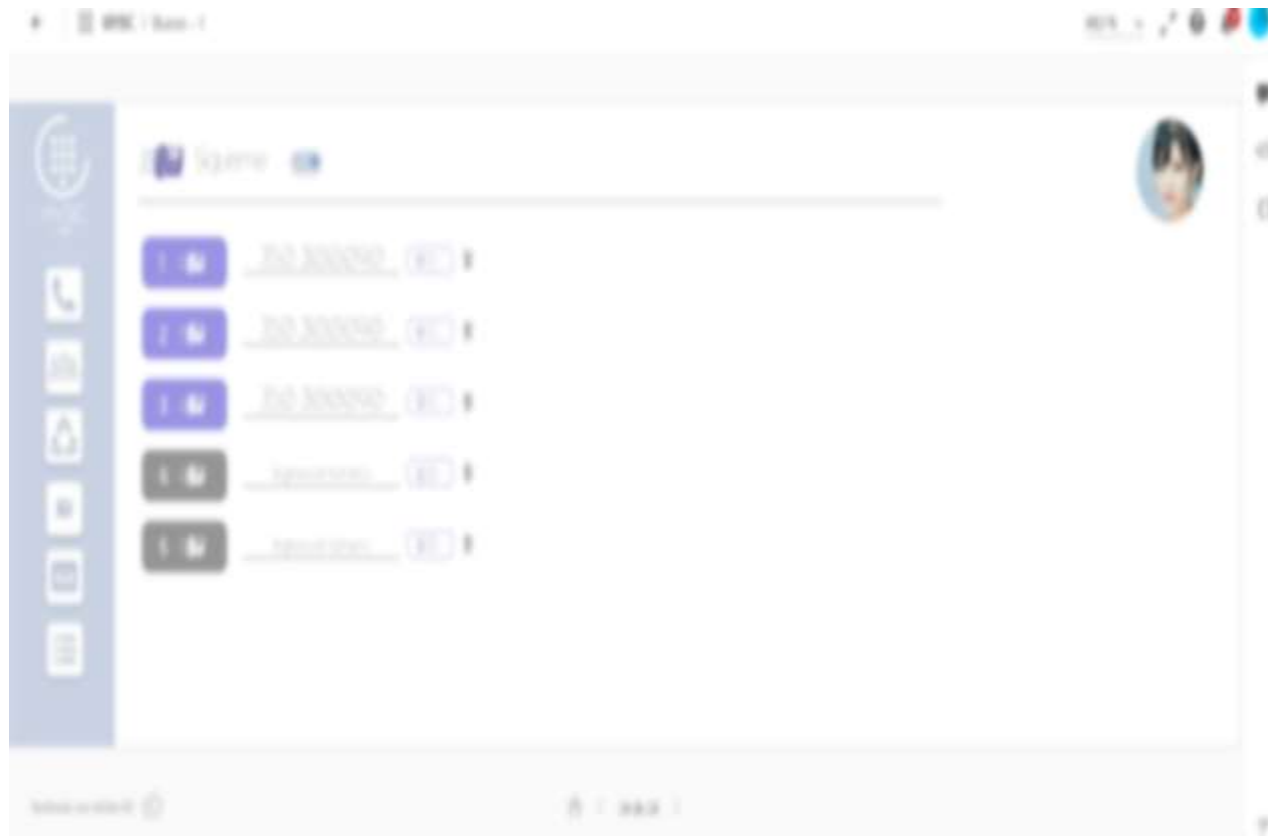


Figura 15. Interfaz Siguieme (Buzón, maibox). Fuente: Autor de la pasantía.

3.1.2.2 Evaluación del diseño por parte de coordinador de Desarrollo.

Se entregó la nueva maquetación y diseño del aplicativo web Mybc para la evaluación por parte del coordinador de la pasantía. Consistió en presentar cada una de las nuevas interfaces del aplicativo web por medio de reuniones con el coordinador donde se expuso la debida corrección.

3.1.2.3 Corrección del diseño.

Esta actividad consistió en revisar todas las indicaciones que el coordinador de la pasantía entregó en dichas reuniones y por medio de un feedback hecho por el coordinador que también se tomó como base para el desarrollo de las respectivas correcciones de la maquetación y diseño del aplicativo web.

3.1.3 Objetivo 3. Realizar las pruebas necesarias para verificar la calidad de las diferentes funcionalidades del aplicativo soportado bajo el framework express.js.

Actividades

- Codificación de los componentes para el aplicativo MYBC.
- Evaluación del api por parte del coordinador de Desarrollo.
- Corrección en el diseño del api.
- Realización de informe final

3.1.3.1 Codificación de los componentes para el aplicativo MYBC.

Arquitectura del Aplicativo web Mybc.

El aplicativo web Mybc se construyó en base a una arquitectura de cliente/servidor, esta arquitectura se divide en tres capas como son:

- La capa de presentación (Frontend).
- La capa de procesos (Backend).
- La capa datos (Base datos).

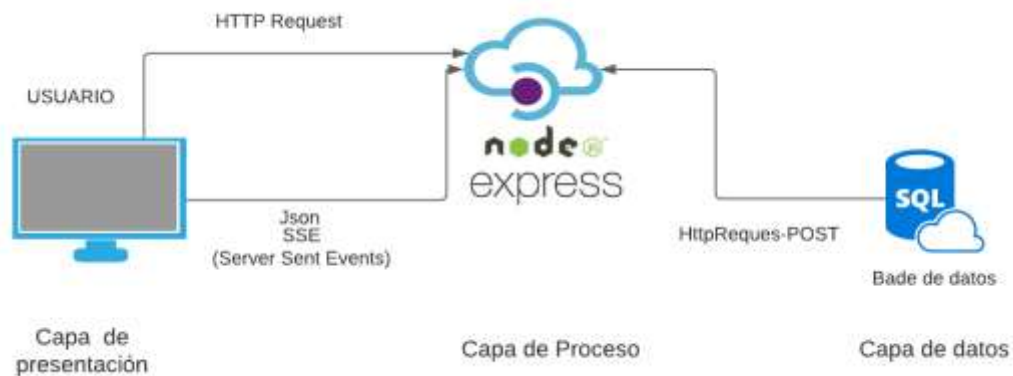


Figura 16. Arquitectura del aplicativo web Mybc. Fuente: autor de la pasantía.

La capa de presentación (Frontend): Esta capa se encarga de que el usuario pueda visualizar

todas las interfaces del aplicativo web, esto para que los usuarios interactúen con el sistema de manera más amigable con el sistema. Para este aplicativo se desarrolló con framework vue.js ya que es flexible para el desarrollo de las interfaces.

La capa de proceso (Base datos): Esta capa es donde se codifica toda la lógica del aplicativo web, la capa de proceso se relaciona con la capa de presentación cada vez que sea necesario para darle soluciones a las peticiones del usuario, su desarrollo se realizó con el framework de Express.js permite la construcción de la lógica del aplicativo de manera más sencilla y con menos consumo de recursos.

La capa datos (Base datos): En esta capa se almacenan los registros, datos en información más importante que después será trata para el funcionamiento del aplicativo web, el gestor de base datos garantiza la creación, actualización eliminación de dichos datos. MySQL es el gestor de base de datos que se utilizó para este proyecto

Estructura del Frontend.

EL frontend es la parte visual del aplicativo es decir donde se muestra los datos provenientes de un servidor para que sean manipulados por los usuarios por medio de elementos interactivos. En el proyecto la creación del Frontend se utilizó la metodología de desarrollo de interfaces basado en componentes, es decir construir elementos reiterativos de una aplicación para poder ser usados en una nueva instancia del desarrollo. También permite arreglar o hacer modificaciones a futuro.

En el desarrollo de un aplicativo es fundamental hacer un prediseño y maquetado de cómo se quieren los elementos visuales acordes a los requerimientos del aplicativo para esto se

empleamos la herramienta xd.adobe para este proyecto.

La interfaz de usuario está definida en principalmente en 3 partes

- Una cabecera.
- Un menú lateral situado a la izquierdo de la pantalla.
- Una sección principal donde se muestra el funcionamiento del aplicativo.

A continuación, se detallarán alguna de las características que componen cada una de las partes:

Cabecera

Situado en la parte superior de la pantalla. Un tamaño necesario para contener un título y componente del perfil del usuario.

- Imagen: ubicada en la parte izquierda, donde se representa la sección donde se encuentran actualmente el usuario.
- Título de la sección actual: Donde se observa el nombre de la sección en la que se encuentra en ese momento el usuario.
- Perfil del usuario: ubicado en la parte derecha del encabezado, permite visualizar la imagen personalizada por usuario y un botón de salir de la aplicación.

Menú lateral.

Muestra un listado de todas las secciones que cuenta el aplicativo web, es un menú que te direccionan a las diferentes sección o vistas del aplicativo.

Sección principal.

Situado por debajo de la cabecera, y la derecha del menú lateral, es una zona de la interfaz donde se pueden encontrar el plugin de un teléfono funcional, creación, edición y eliminar de elementos, un crud de información del usuario, es la zona donde el usuario todas las actividades necesarias para el fin que fue desarrollado el aplicativo.

Como se puede ver en figura 17, de todos los archivos que componen la estructura del frontend los más importantes son:

- **Mybc/node_modules:** Contiene los archivos
- **mybc/public:** Esta carpeta se utiliza para almacenar ficheros estáticos que no serán procesados por el framework.
- **mybc/src:** Es donde se contiene todo el cuerpo del proyecto. Desde las imágenes, las vistas hasta las traducciones
- **mybc/src/api:** Es la carpeta donde se encuentran los métodos y rutas para la conexión del frontend con el backend. Para que el aplicativo fuera full Api.
- **mybc/src/assets:** Es la carpeta donde se almacenan imágenes, audios, sonidos.
- **mybc/src/components:** Donde se almacena los componentes que se utiliza en las vistas
- **mybc/src/imagenes:** En la carpeta donde se guarda las imágenes para el componente Wphone.vue
- **src/libs:** donde se almacena los archivos de audios.js, eventBus.js, SIPml-api.js y útil.js.
- **mybc/src/locales:** la carpeta contiene los archivos js de traducciones.
- **mybc/src/plugins:** Se utiliza para almacenar las diferentes traducciones del sistema.
- **mybc/src/router:** Está compuesto por los archivos js que se emplean para manejar las mybc/rutas de la aplicación.
- **mybc/src/store:** Se usa para almacenar la configuración para emplear Vuex.
- **mybc/src/utills:** Se encuentra el request.js de la aplicación
- **mybc/src/views:** Se almacenan los archivos de todas las vistas con la extensión vue.
- **mybc/src/App.vue:** Es el archivo principal de la aplicación.
- **mybc/src/main.js:** Se trata del fichero principal que arranca el proyecto Vue.

- **mybc/browserlistrc:** almacena la configuración api REST que proporciona acceso a la base de datos.
- **mybc/eslintrc.js:** Es el archivo que se emplea para configurar ESLint.
- **mybc/.gitignore :** Fichero que indica los archivos que Git debe ignorar al hacer el versionado.
- **Mybc/package-lock.json:** Fichero histórico de versionado de apoyo para el package.json
- **mybc/package.json:** Contiene las dependencias del proyecto.
- **Mybc/readme.md:** Fichero Markdown donde se documenta información sobre el proyecto.
- **Mybc/vue.config.js:** Fichero general de configuración del proyecto Vue.js

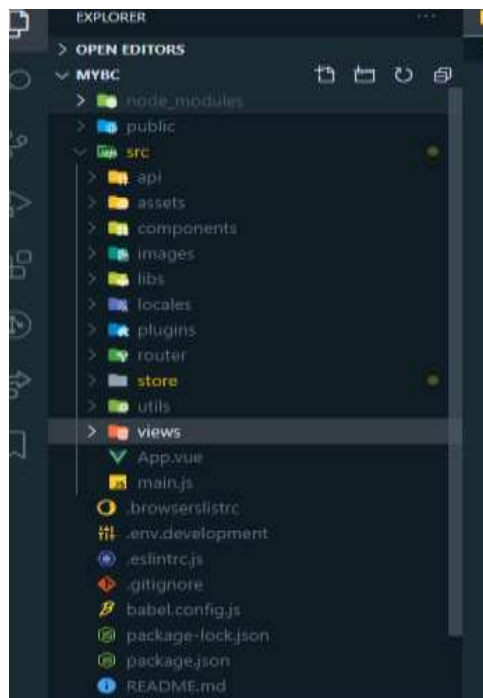


Figura 17 .Estructura del Frontend. Fuente. Autor de la pasantía.

A continuación, se evidencia las configuraciones de dichos archivos del frontend del aplicativo web.

Carpeta api.

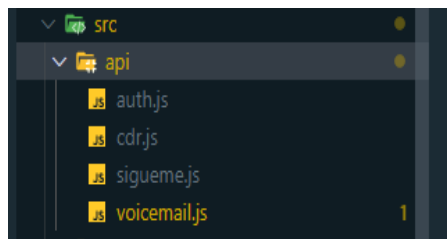


Figura 18. Archivo de configuración api del aplicativo web. Fuente: Autor de la pasantía.

Como se puede observar en la figura 17 se presenta los archivos que componen la carpeta api del aplicativo web:

- Auth.js: Archivo de configuración donde se encuentran las rutas de autenticación y métodos correspondientes para conectarse con el backend y poder darle el permiso o no al usuario para el acceso al aplicativo.
- Cdr.js: Archivo de configuración donde están las rutas y métodos para poder acceder desde frontend a los atributos de contactos según lo requeridos por los métodos creados en el backend para que cumplan con la petición que se realizaron.
- Sigume.js: Archivo de configuración donde se encuentran las rutas y método para hacer posible conexión desde frontend con el backend y así cumplir con los requerimientos pedidos como lo son la consulta, la edición y eliminación de la interfaz Sigume.
- Voicemail.js: Archivos de configuración donde se encuentran las rutas y métodos para poder tener la conexión entre el frontend y el backend y así poder realizar el listado de las llamadas realizadas con sus respectivos audios y fechas de ejecución.

Carpeta Components.

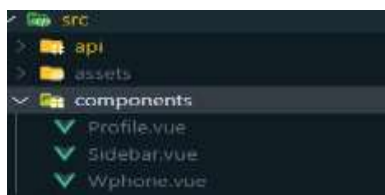


Figura 19. Archivos de configuración de la carpeta de Components. Fuente: Autor de la pasantía.

Como se puede observar en la imagen anterior, la figura 18 se tiene la carpeta de componentes donde se encuentran los archivos de configuración de las interfaces. El desarrollar el aplicativo web en la parte del frontend se construyó por medio de componentes puesto que se facilita la codificación y creación de todas las interfaces del aplicativo.

- Profile.vue: En este archivo de configuración se encuentra la codificación del avatar del usuario y donde se puede salir de forma general del aplicativo web.
- Sidebar.vue: En este archivo de configuración se encuentra la codificación del menú lateral donde se puede mover por las diferentes interfaces del aplicativo web.
- Wphone.vue: En este archivo de configuración se está toda la codificación de la interfaz del teléfono.

Carpetas locales



Figura 20. Archivos de configuración de la carpeta de locales. Fuente: Autor de la pasantía.

En la figura 20 se puede observar los archivos de configuración que contiene la carpeta local en cargada de la contracción al inglés como al español de todos los textos que contiene el aplicativo web.

- **En.json:** Archivo de configuración de para el idioma inglés.

- **Es.json:** Archivo de configuración del idioma español.

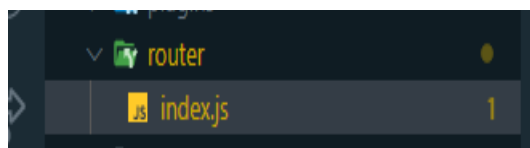
Carpeta plugins



*Figura 21.*Archivos de configuración de la carpeta de locales. Fuente: Autor de la pasantía

Como se puede observar en la figura anterior, la figura 21 muestra los archivos de configuración de la carpeta plugin donde se encuentre el plugins de configuración para que se puede realizar la traducción de los textos.

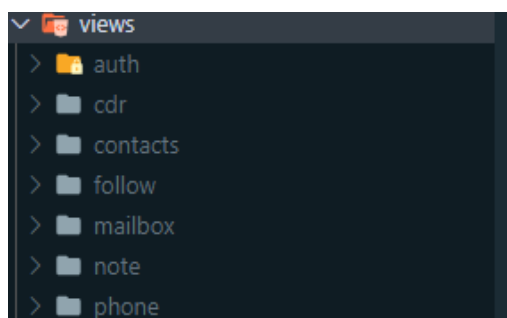
Carpeta router



*Figura 22.*Archivos de configuración de la carpeta router. Fuente: Autor de la pasantía

En la figura 22 se evidencia el archivo de configuración index.js que se encarga de las rutas que se emplean en el aplicativo web, para moverse al usuario de una interfaz a otra.

Carpeta views.



*Figura 23.*Archivos de configuración de la carpeta de views. Fuente: Autor de la pasantía.

Estructura backend del aplicativo web Mybc.

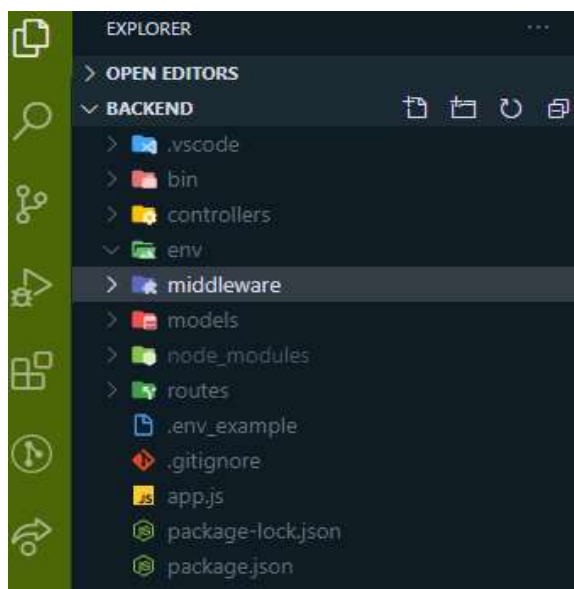


Figura 24. Estructura del backend del aplicativo Mybc. Fuente. Autor de la pasantía.

Controllers: Se realiza las operaciones y acciones del aplicativo web.

Middleware: Este archivo nos permite o niega el acceso a nuestro aplicativo web.

Models: Este archivo me permite generar los modelos y esquemas de la base de datos.

Routes: Este archivo me permite establecer las rutas, siendo las que define nuestro aplicativo web en ellas está la lógica de conexión entre el backend y el frontend del aplicativo web.

App: Este archivo me permite generar el servidor web con Node.js, se almacena la configuración de express, donde ejecutamos las rutas después de crearlas.

Package-lock.json: En este archivo se encuentran los fichero histórico de versionado de apoyo para el package.json.

Package.json: En este archivo se encuentra la configuración principal del backend del aplicativo web por ejemplos las dependencias.

Conexión con la base de datos.

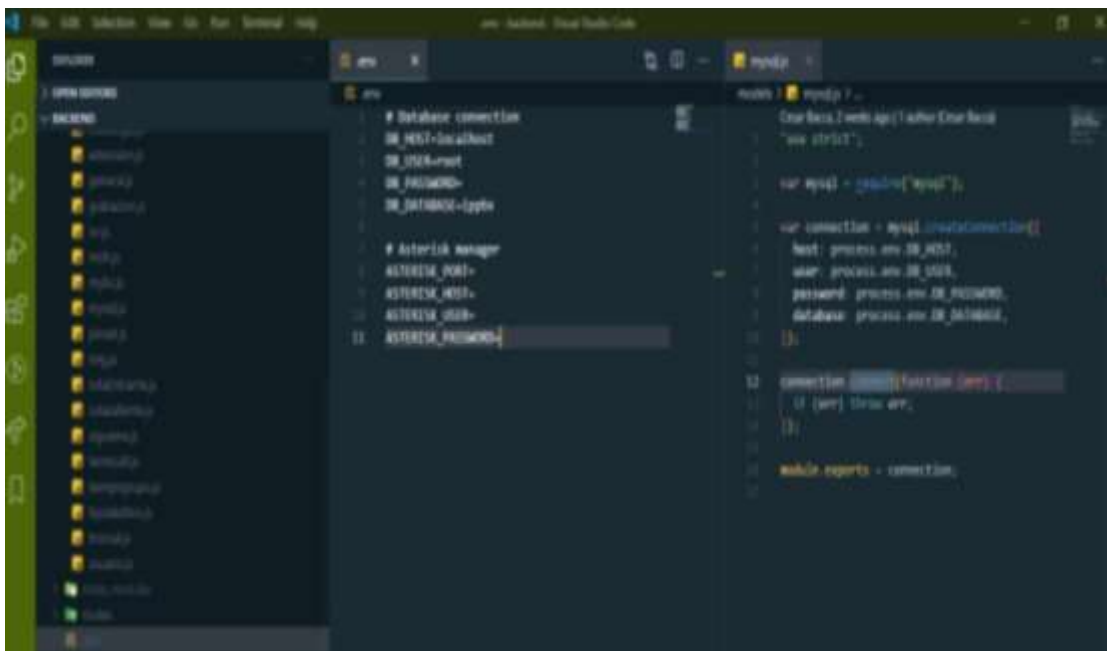


Figura 25. Configuración del archivo .env, y modelo mysql para conexión a la base de datos. Fuente: Autor de la pasantía.

Configuración del archivo de autenticación

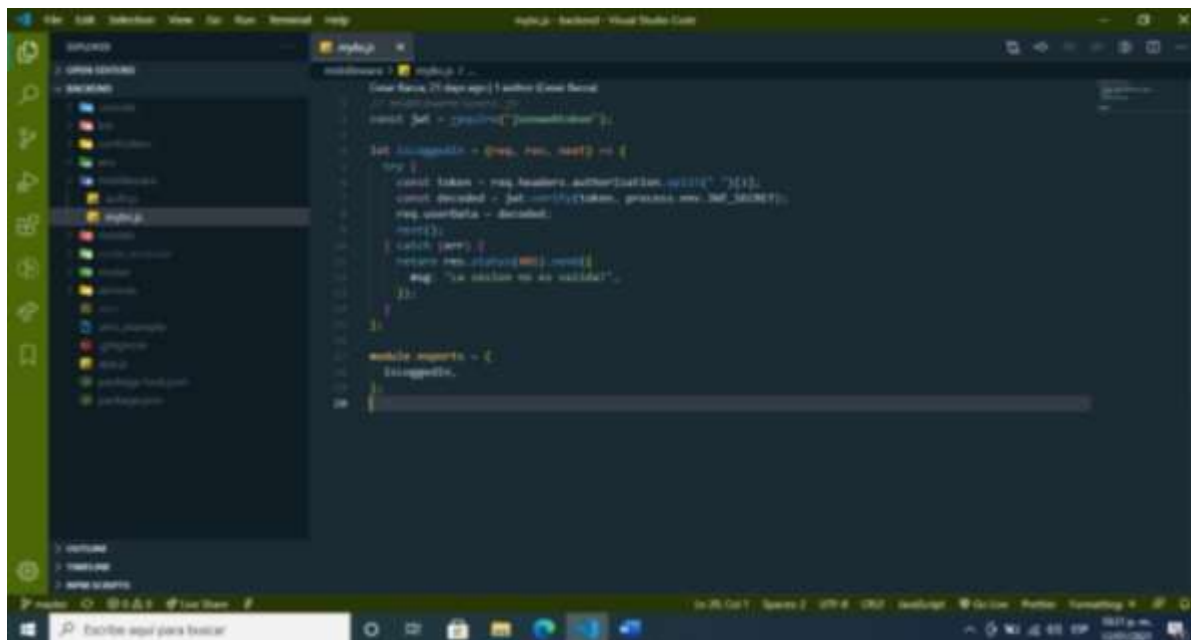


Figura 26. Configuración del archivo de Mybc para la autenticación. Fuente: Autor de la pasantía.

Configuración del archivo MybcController del controlador del aplicativo web.

```

var Mybc = require("../Modelo/mybc.js");

exports.cdr = function (req, res) {
  Mybc.cdr(function (err, cdr) {
    if (err) res.send(err);
    res.send(cdr);
  });
};

exports.followe = function (req, res) {
  var extension = req.query.extension;
  Mybc.getfollowe(extension, function (err, followe) {
    if (err) res.send(err);
    res.send(followe);
  });
};

exports.votomail = function (req, res) {
  var extension = req.query.extension;
  Mybc.getvotomail(extension, function (err, votomail) {
    if (err) res.send(err);
    res.send(votomail);
  });
};

exports.create_followe = function (req, res) {
  const followe = req.body;

  Mybc.createfollowe(followe, function (err, followe) {
    if (err) res.send(err);
    res.send(followe);
  });
};

```

Figura 27. Configuración del archivo mybcController. Fuente: Autor de la pasantía.

Modelo del aplicativo web Mybc.

```

var Mybc = require("../Modelo/mybc.js");

exports.cdr = function (req, res) {
  Mybc.cdr(function (err, cdr) {
    if (err) res.send(err);
    res.send(cdr);
  });
};

exports.followe = function (req, res) {
  var extension = req.query.extension;
  Mybc.getfollowe(extension, function (err, followe) {
    if (err) res.send(err);
    res.send(followe);
  });
};

exports.votomail = function (req, res) {
  var extension = req.query.extension;
  Mybc.getvotomail(extension, function (err, votomail) {
    if (err) res.send(err);
    res.send(votomail);
  });
};

exports.create_followe = function (req, res) {
  const followe = req.body;

  Mybc.createfollowe(followe, function (err, followe) {
    if (err) res.send(err);
    res.send(followe);
  });
};

```

Figura 28. Archivo de configuración del modelo Mybc del aplicativo web. Fuente: Autor de la pasantía.

Configuración del archivo de rutas Routes

```

mybc.js
routes > mybc.js > ...
1  const express = require("express");
2  const router = express.Router();
3
4  const Mybc = require("../controllers/MybcController");
5
6  router.route("/cdr").get(Mybc.cdr);
7
8  router.route("/followme").get(Mybc.followme);
9
10 router.route("/followme").post(Mybc.create_followme);
11
12 router.route("/followme/:id").put(Mybc.update_followme);
13
14 router.route("/voicemail").get(Mybc.voicemail);
15
16 router.route("/voicemail/:id").put(Mybc.update_voicemail);
17
18 router.route("/check").get(Mybc.check);
19
20 module.exports = router;
21

```

Figura 29. Archivo de configuración router del aplicativo web Mybc. Fuente: Autor de la pasantía.

Configuración del archivo principal

```

app.js
app > ...
1  var createError = require("http-errors");
2  const helmet = require("helmet");
3  var express = require("express");
4  var cookieParser = require("cookie-parser");
5  var fileupload = require("express-fileupload");
6  var compression = require("compression");
7  var osu = require("node-os-utils");
8  const { exec } = require("child_process");
9  const S3 = require("express-s3");
10 const s3 = new S3();
11
12
13 const path = require("path");
14 require("dotenv").config({ path: path.join(__dirname, ".env") });
15
16 var am = new require("asterisk-manager")({
17   process.env.ASTERISK_PORT,
18   process.env.ASTERISK_HOST,
19   process.env.ASTERISK_USER,
20   process.env.ASTERISK_PASSWORD,
21   true
22 });
23
24 const bodyParser = require("body-parser");
25 const authMiddleware = require("../middlewares/auth");
26 const mybcMiddleware = require("../middlewares/mybc");
27
28 //const corsMiddleware = require("../middlewares/cors");
29
30 var logger = require("morgan");
31 var cors = require("cors");
32
33 //var bodyParser = require("body-parser");
34 //var multer = require("multer");

```

Figura 30. Configuración del archivo principal app.js. Fuente: autor de la pasantía.

3.1.3.2 Evaluación del api por parte del coordinador de Desarrollo.

Prueba de la interfaz Loguin de Mybc para el acceso a la aplicación ya montado en el servidor designado por la empresa al aplicativo web Mybc.

Loguin o inicio de sesión

La vista desarrollada es la de acceso a la aplicación, que permite o no el acceso del usuario dependiendo si tiene las credenciales necesarias.

- **El usuario:** Debidamente registrado en los de más aplicativos de la empresa como es el voice donde se gestiona una Extensión que es el usuario admitido para el inicio de sección
- **La contraseña:** Es generada por el usuario en aplicativo voice, cuando el usuario quiere enlazar una extensión y con el aplicativo Mybc en el aplicativo voice.

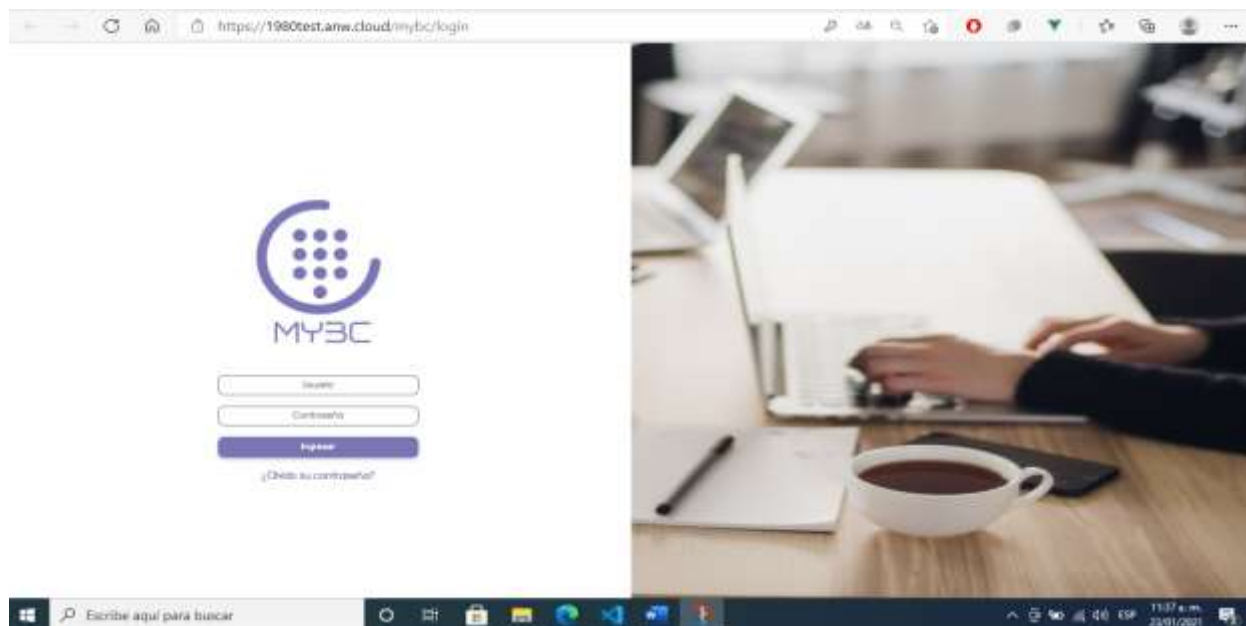


Figura 31. Prueba de la interfaz de inicio de sesión al aplicativo web desde el servidor. Fuente: Autor de la pasantía.

Comprobación de que se permita o no el acceso al aplicativo por medio de las credenciales que tenga el usuario. Usuario y contraseña diligenciados.

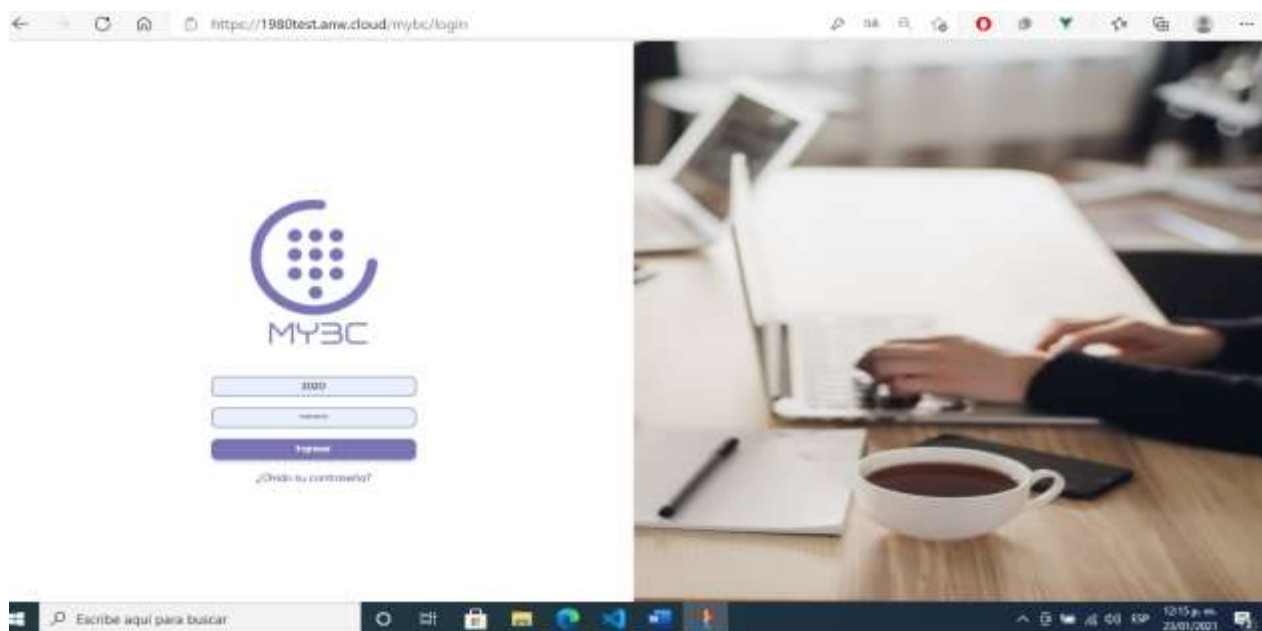


Figura 32. Interfaz de inicio con el usuario y contraseña correspondiente para el acceso al aplicativo. Fuente: Autor de la pasantía.

Acceso permitido al aplicativo mediante las credenciales correctas. Donde se puede realizar las llamadas como si se tratase de un teléfono móvil posee las mismas características.



Figura 33. Interfaz principal después de inicio de sesión. Fuente: Autor de la pasantía.

Realización de una llamada telefónica a un teléfono móvil.

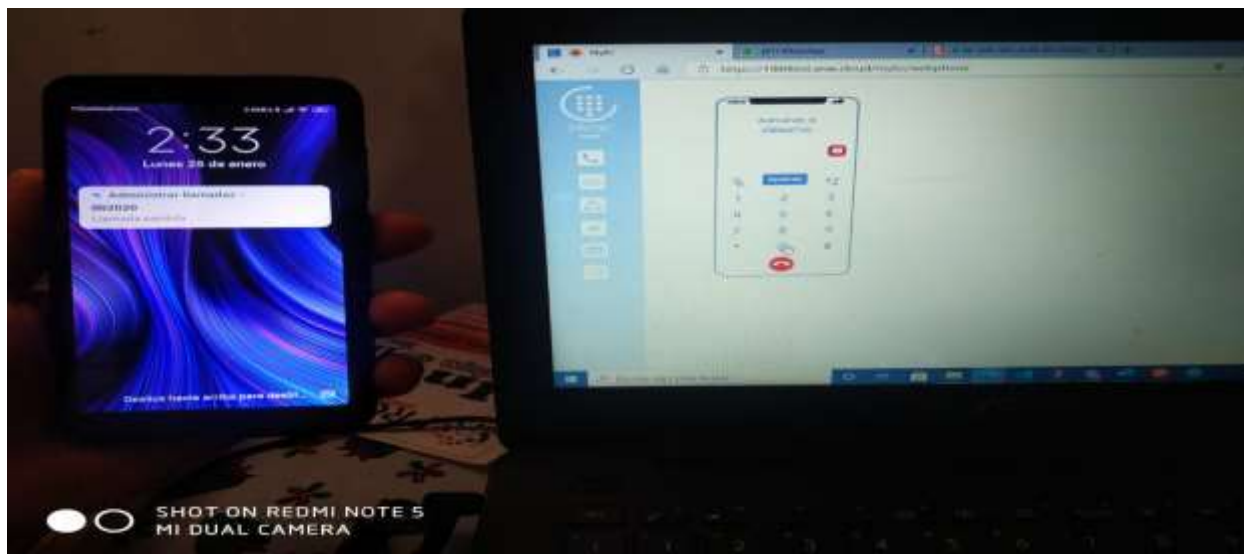


Figura 34. Realización de una llamada a un teléfono móvil. Fuente: Autor de la pasantía.

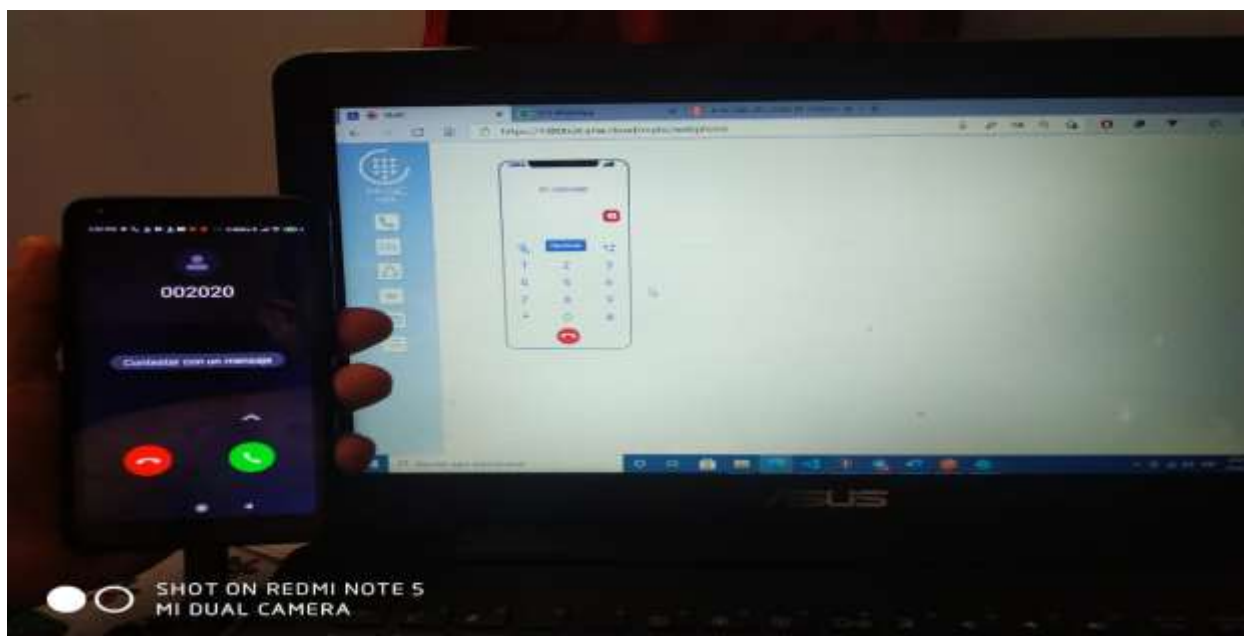


Figura 35. Llamada establecida con el teléfono móvil. Fuente: Autor de la pasantía.

Interfaz de lista de contactos del aplicativo web Mybc trayendo desde la base datos todos los contactos que tengo el usuario.



Figura 36. Interfaz Registro de Contactos funcional desde el servidor Fuente: Autor de la pasantía.

Interfaz de conferencia en el servidor de la empresa funcional.



Figura 37. Interfaz funcional de Conferencia (Conferencia). Fuente: Autor de la pasantía.

Inicio de la conferencia en el servidor del aplicativo web Mybc donde se puede compartir la conferencia y poder iniciar la conferencia

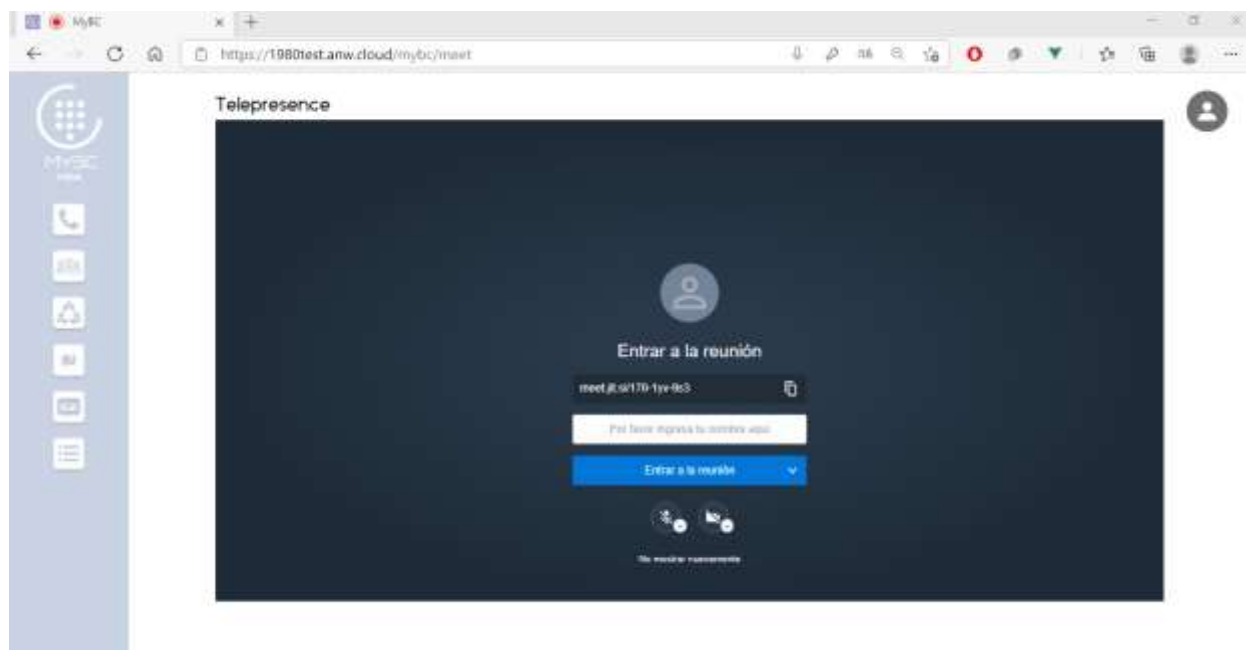


Figura 38. Inicio de la Interfaz sala de conferencia (Conferencia-1). Fuente: Autor de la pasantía.

Interfaz después de dar inicio a la conferencia.

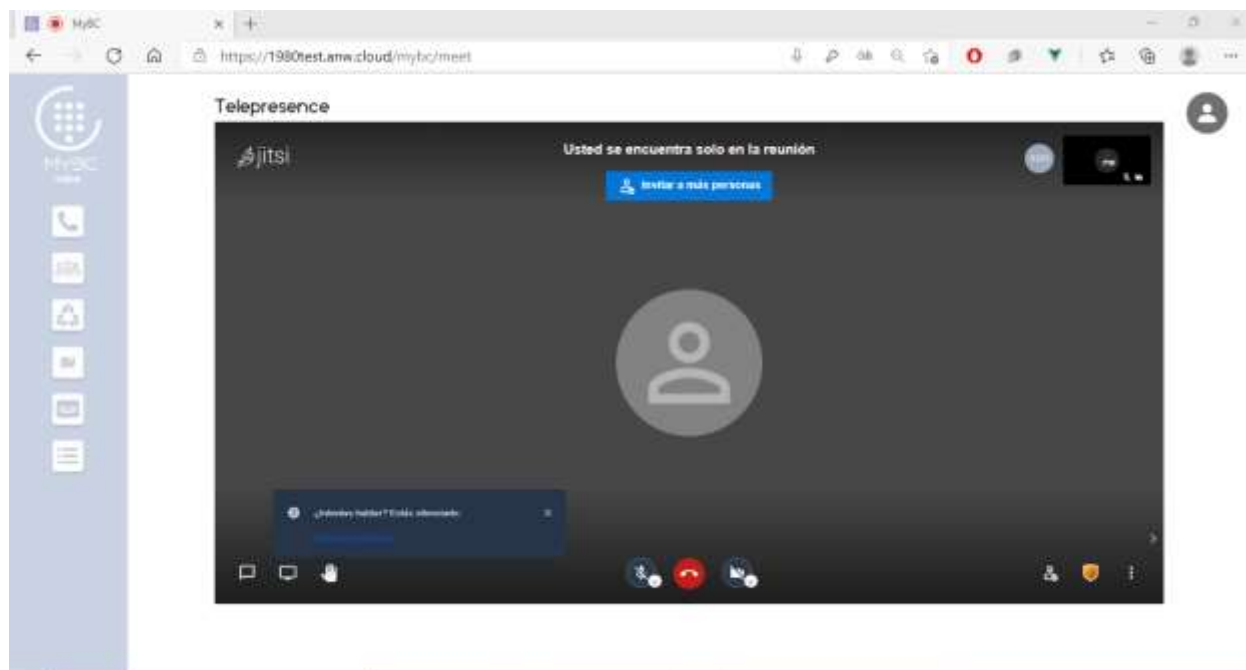


Figura 39. Interfaz sala de conferencia después de iniciada la conferencia (Conferencia-1). Fuente: Autor de la pasantía.

Interfaz del Sigüeme (Buzón, maibox) ya implementada y funcionando en el servidor de la empresa.

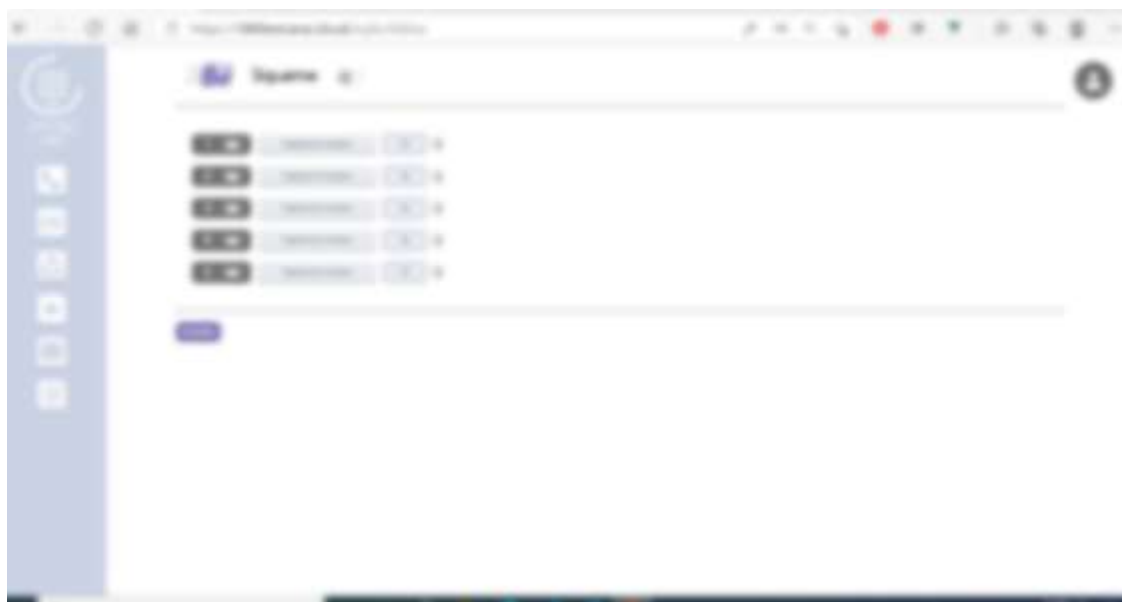


Figura 40. Interfaz Sigüeme implementada (Buzón, maibox). Fuente: Autor de la pasantía.

Interfaz de Registro de voicemail (Buzón, maibox) implementada en el servidor del Mybc.



Figura 41. Interfaz Registro de voicemail implantada (Buzón, maibox). Fuente: autor de la pasantía.

Interfaz Registro de llamadas (CDR) implantada en el servidor de Mybc de la empresa.



The screenshot displays a web application interface for 'Registro de llamadas'. It features a table with columns for 'Origen', 'Destino', 'Duración', 'Estado', and 'Fecha'. The table contains several rows of data, with some cells containing red and green status indicators. A vertical sidebar on the left contains navigation icons, and a user profile icon is visible in the top right corner.

Origen	Destino	Duración	Estado	Fecha
100	100	00:01	OK	2023-10-27 10:00
100	100	00:01	OK	2023-10-27 10:01
100	100	00:01	OK	2023-10-27 10:02
100	100	00:01	OK	2023-10-27 10:03
100	100	00:01	OK	2023-10-27 10:04
100	100	00:01	OK	2023-10-27 10:05
100	100	00:01	OK	2023-10-27 10:06
100	100	00:01	OK	2023-10-27 10:07
100	100	00:01	OK	2023-10-27 10:08
100	100	00:01	OK	2023-10-27 10:09
100	100	00:01	OK	2023-10-27 10:10

Figura 42. Interfaz Registro de llamadas implementada (CDR). Fuente: autor de la pasantía.

Cuando el usuario tiene un tiempo en la aplicativo y no está en uso el aplicativo lo devuelve a la interfaz de inicio para que nuevamente realice el proceso de inicio de sección para renovar el token y tener el acceso de nuevo al aplicativo.

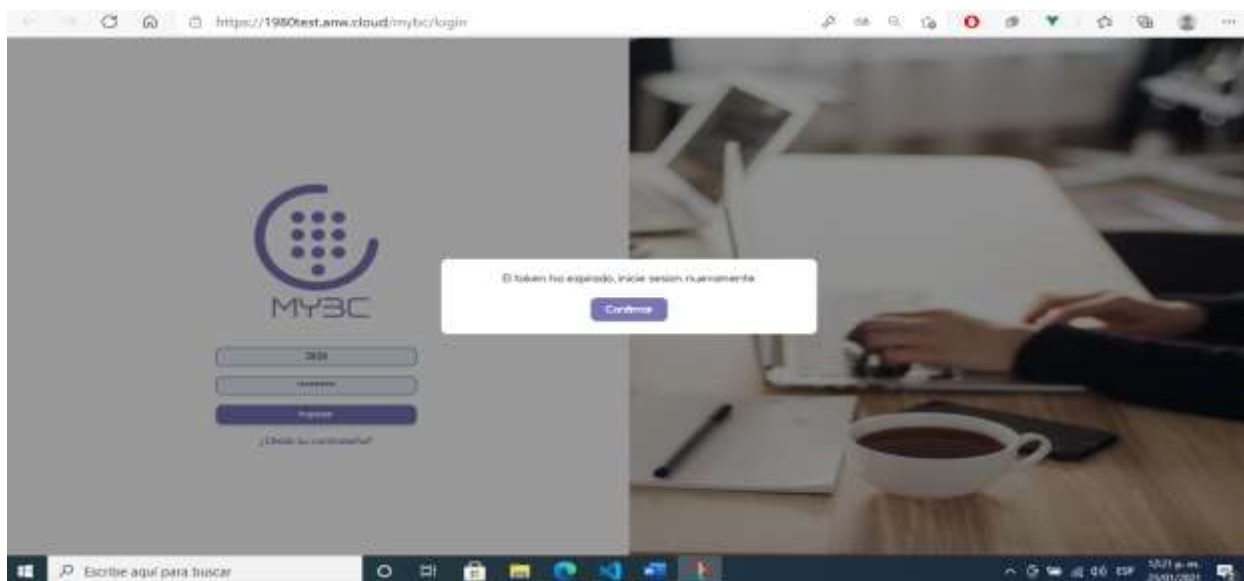


Figura 43. Interfaz para el reinicio de logue después de un tiempo de no ser utilizado el aplicativo.

3.1.3.3 Corrección en el diseño del api.

En esta actividad se desarrolló en base a las correcciones hechas por el coordinador y el jefe de la empresa en cada entrega, en la entrega del frontend, en la entrega del backend y en la entrega de aplicativo funcionando corrigiendo cada error que se presentó.

3.1.3.4 Realización de informe final

Se realiza el informe de entrega final que se entrega al jefe de la empresa.

Por último, es claro aclarar que Normas de seguridad y privacidad de la empresa no se puede presentar todas las actividades y desarrollo del aplicativo.

4 .Diagnóstico Final

Este proyecto se basa en el desarrollo del aplicativo web Mybc para mejorar el rendimiento, usabilidad y visualización que presentaba el antiguo aplicativo, teniendo en cuenta las requerimientos obtenidos al inicio del proyecto y solucionado cada uno de ellos se puede evidenciar que gracias a este nuevo desarrollo del aplicativo web Mybc que se cuenta con un mejor rendimiento ya que las acciones y funciones del aplicativo que puede realizar los usuario se procesan más rápido y con menos recursos, los usuarios encuentra las interface más agradable para el uso del aplicativo, por este motivo el usuarios del aplicativo se le facilitara el sus del mismo aplicativo web.

Todo esto conlleva a que la empresa pueda darles soporte a las necesidades de cada uno de sus usuarios de una manera más rápida, eficiente, confiable y segura. Gracias a las mejoras realizadas se logró satisfacer en gran parte las necesarias presentada por los usuarios del aplicativo web Mybc a la empresa de MTSOFT. Le permitirá aumentar el número de usuarios al que le podría ofrecer los servicios del aplicativo web Mybc y con una alta calidad.

5. Conclusiones

Al llevar a cabo el desarrollo de proyecto que constaba en la creación de un nuevo aplicativo web Mybc, para satisfacer las necesidades y errores expuestos por los usuarios en el antiguo aplicativo, se produjo la ejecución de este proyecto con el inicio de la identificación de los requerimientos del nuevo aplicativo tanto el backend y frontend, como la realización de la diseño y la implementación de las nuevas interfaces y la creación de la lógica necesario para el funcionamiento del aplicativo podemos concluir que en este trabajo de pasantías se logró desarrollar y aplicar los conocimientos adquiridos durante el tiempo de estudio en la carrera de Ingeniería de Sistemas donde se pueden evidenciar en los siguientes puntos.

- El desarrollo del backend y del frontend del aplicativo se realizó mediante la implementación de metodología Scrum, donde se utilizó 5 sprints, permitiendo adaptar los procesos del desarrollo a las necesidades del usuario de una mejor manera, una revisión continua de las tareas a realizar, manteniendo una idea clara de avance del proyecto en cada sprint.
- El desarrollo del frontend se realizó mediante el framework vue.js y en el backend con framework express.js ambos basados en JavaScript permiten la implantación de los conocimientos adquiridos relacionados con JavaScript en el transcurso del desarrollo de la carrera.
- El manejo y entendimiento de la base de datos que cuenta la empresa para el aplicativo web Mybc se pudo comprender y entender cada una de sus entidades y relaciones entre ellas, fue gracias al conocimiento que se obtuvo en el transcurso de la carrera de ingeniería de sistemas relacionados con bases de datos relacionales.

La utilización de nuevas aplicaciones en el desarrollo del nuevo aplicativo web Mybc como lo es express.js, vue.js, node.js y MySQL hace que el proceso de desarrollo sea más sencillo, ya sea en la etapa de desarrollo como después en la ejecución del aplicativo ya que se mantiene más organizado el código, se optimiza el código para poder reutilizar y no se pierda tiempo si en un llegado momento es necesario la implantación a futuro de una nueva funcionalidad.

La forma en que fue codificado e implantado el aplicativo web Mybc es decir los estándares de desarrollo que fueron empleando para su creación, lo podrá manipular y entender el código cualquier otro desarrollador web que tengo conocimientos de las herramientas utilizadas.

Gracias a las pruebas realizadas, se puede concluir que el aplicativo web Mybc ha cumplido con todos los requerimientos estableciendo para si aceptación.

6.Recomendaciones

Se recomienda seguir programación estructurada tomando en cuenta patrones de diseño y patrones arquitectónicos ya que el software obtendrá claridad, calidad y menor tiempo de desarrollo y en su futuro simplificar revisiones en caso de reestructuración.

Se recomiendo el fortalecimiento del área de desarrollo de la empresa. La cual mejorar el proceso de análisis, desarrollo, implementación, el diagnostico de errores y su respectiva corrección para que no se pierda tiempo que en las fases finales del proyecto serán necesarios para finalizar su desarrollo.

Se recomiendo.

En caso de implantar nuevas funcionalidades al aplicativo web, se recomiendo seguir los mismo estándares desarrollo.

7. Referencias.

- Bravo León, M. R. (2019). Diseño e Implementación de una Infraestructura de Red de Datos para el Caso de Estudio de la Empresa XYZ a Partir de un Entorno Virtualizado. *epository.unad.edu.co*, 29-30.
- Cáceres Sánchez, J. S. (2019). Diseño de prototipo de aplicación para visualización de productos en realidad aumentada. *Bachelor's thesis*, 12-13.
- Carreño Villalba , R. (2016). Aplicación de administración con Angular, Node y Express para una aplicación Django. *uma.esriuma*, 8.
- Castro Navarro, M. A. (2017). . PROPUESTA PARA EL MEJORAMIENTO ORGANIZACIONAL Y COMERCIAL DE LA EMPRESA TMSOFT SAS. 26-27. Obtenido de <https://sic.us.es/servicios/infraestructurascomunicaciones-hw-y-sw/telefonía-sobre-internet-toip>
- Cordoba Vasquez, V. A. (2017). REDISEÑO DE LA PAGINA WEB DE LA EMISORA INSTITUCIONAL DE LA UFM STEREO 95.2 DE LA UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA. *Repositoria institucional ufps*, 10-11.
- Haro, E., Guarda, T., Peñaherrera, A. O., & Quiña, G. N. (2019). Desarrollo backend para aplicaciones web, Servicios Web Restful: Node.js Vs Spring Boot . *Iberica de Sistemas e Tecnologías de Informáticas* , 309-321.
- Lopez Padilla , N., & Arrieta Morelo, Á. M. (2020). Desarrollo de un aplicativo web app progresivo para la divulgación de contenidos institucionales. *académicos y culturales de la Universidad de Córdoba*, 46-50.
- Martínez Fernández, A., Sanz, J., & Vado, A. D. (2020). Desarrollo de una herramienta de creación de contenidos de ficción interactiva.
- MinTIC. (03 de 01 de 2021). *Ministerio de Tecnologías de la Información y las Comunicaciones*. Obtenido de Ministerio de Tecnologías de la Información y las Comunicaciones:

https://mintic.gov.co/portal/604/articles-8580_PDF_Ley_1341.pdf

OpenJS, F. (2017). *Express*. Obtenido de Express: <https://expressjs.com/>

Rodriguez Bernal, M. C. (2018). Visión regulatoria para la implementación de los servicios de voz en Colombia por medio de redes inalámbricas de área local. *Repositorio Institucional Univercidad Santo tomas Bogota*, 47-48.

vue.js. (2020). *Vue.js*. Obtenido de vue.js: <https://vuejs.org/v2/guide/>

