

DISEÑO

ESPINVEN MÓVIL

by Greenware

Caracas, 4 de febrero del 2022

Tabla de Contenido

Clasificación ISO 9126	3
Diagrama de la Arquitectura de la Aplicación	5
Arquitectura de la Aplicación	6
Diagramas de Componentes	7
Perfiles de Usuario	8
Elementos de Usabilidad	11
Principios de Diseño	12
Referencias Bibliográficas	14
Anexo	16

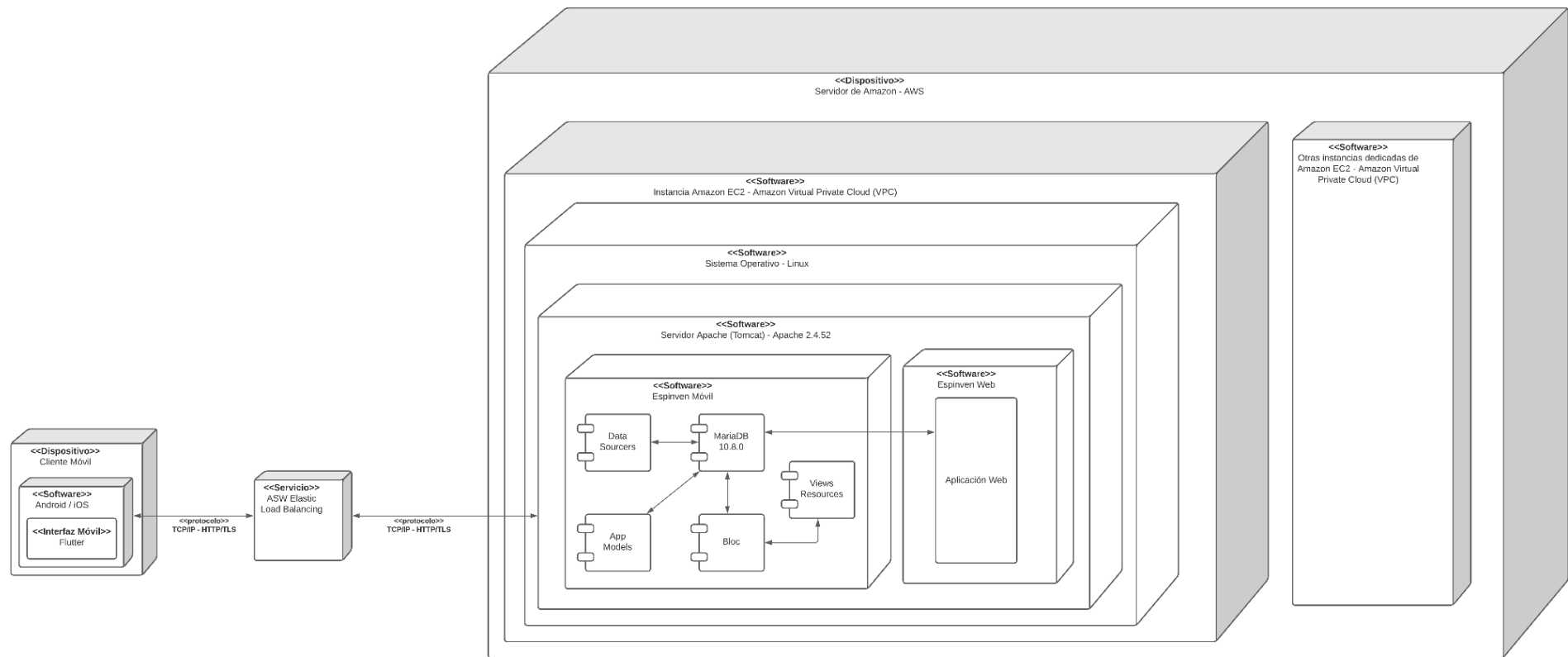
A continuación se presenta la clasificación de los requerimientos no funcionales de Espinven Móvil utilizando la norma ISO 9126:

Id PIECES	Requerimiento	F	C	FU	E	FM	P
REQ3	Diseñar e implementar una base de datos relacional unificada que almacene la información correspondiente a cada uno de los procesos.	X				X	
REQ4	Automatizar los procesos de gestión de cuentas, roles y controles de acceso: A. Para los dueños: Designación de gerentes. B. Para los gerentes: Suspensión, bloqueo y validación de los empleados.	X					
REQ5	Diseñar e implementar interfaces utilizando patrones y estándares de diseño con la finalidad de garantizar flujos y navegaciones similares entre plataformas. Patrones: <ul style="list-style-type: none"> ● Getting Input from Users - Forms and Controls. ● Making It Look Good - Visual Style and Aesthetics. Estándares: <ul style="list-style-type: none"> ● Estándar 3.3 de la WCAG 2.1 de W3C [4]. <ul style="list-style-type: none"> ○ 3.3.1 Error Identification. ○ 3.3.2 Labels or Instructions 	X		X			
REQ6	Configurar y desplegar las aplicaciones Espinven Movil y Espinven Web y base de datos MySQL (XAMPP) en servidores AWS (Servicios en la nube).		X		X	X	X
REQ7	Diseñar y dictar dos cursos de entrenamiento especializado para los usuarios finales de las aplicaciones tanto móvil como web.			X			
REQ8	Diseñar manuales de usuario con documentación referencial para cada una de las aplicaciones tanto móvil como web desplegadas en una wiki de forma online.			X		X	
REQ9	Integrar y comunicar la base de datos con la aplicación web y móvil.	X			X	X	X
REQ19	Diseñar e implementar una comunicación entre la APP con la WEB para enviar pedidos del menú.	X	X				X
REQ20	Diseñar e implementar los modelos lógicos y clases	X		X	X		

	funcionales del Back-End con su documentación técnica.						
REQ21	Diseñar e implementar arquitecturas de software basadas en el patrón BLOC con el objetivo de garantizar un menor tiempo de respuesta y proporcionar una mayor seguridad y extensibilidad. Este patrón utiliza el Framework Flutter y el lenguaje Dart como únicos recursos en cada una de sus 4 etapas.	X	X		X	X	X
REQ22	Configurar y desplegar la aplicación en las tiendas de Android y iOS (Play Store y App Store respectivamente).		X				X

Diagrama de Arquitectura de la Aplicación

Espinven móvil se fundamenta en una arquitectura cliente-servidor de tres capas e implementa el patrón de arquitectura de software Modelo–Vista–Modelo de Vista (MVVM) [1][2][3][4][5][6][12][13][14][15].



Arquitectura de la Aplicación

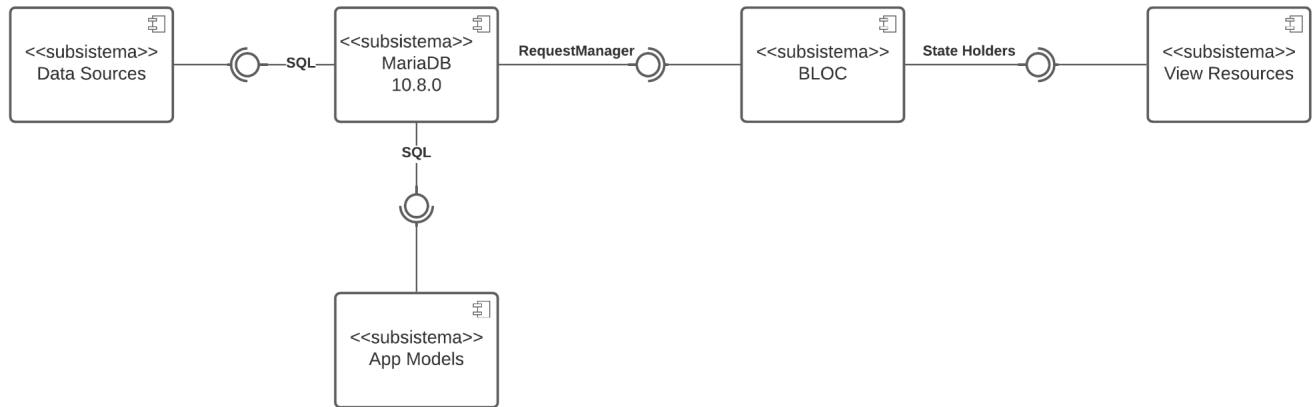
A continuación se presenta una breve explicación técnica de cada una de las capas o niveles que conforman a la arquitectura anteriormente presentada [1][2][3][4][5][6][12][13][14]:

- **Cliente:** Representa el encargado de recibir y enviar peticiones al servidor a través de distintos protocolos de red y por medio de las interfaces de usuario basadas en Flutter, el cual trabaja con interfaces nativas tanto en Android como en iOS, y así ejecutar la aplicación en los distintos dispositivos de los clientes con la finalidad de interactuar con el sistema. Por lo tanto los requerimientos técnicos mínimos necesarios para poder interactuar con el sistema se encuentran ligados a cualquier dispositivo compatible con las versiones operativas de Android e iOS actuales, en donde Flutter se encargará de la ejecución de la app en ambos ambientes.
- **AWS Elastic Load Balancing (Distribuidor de cargas):** Servicio preestablecido por Amazon para todas las instancias de servidores EC2, con la finalidad de distribuir automáticamente el tráfico de aplicaciones a múltiples destinos y dispositivos virtuales en una o más zonas de disponibilidad[6].
- **Servidor:** Conformado por los distintos mecanismos de seguridad y la instancia virtual dedicada del servidor (Amazon EC2) contratado a AWS. Amazon EC2 se apoya en las tecnologías de virtualización, permitiendo utilizar gran variedad de sistemas operativos a través de sus interfaces de servicios web (VPC), personalizarlos, gestionar permisos de acceso a la red y ejecutar tantos sistemas como sean requeridos. Cada instancia dedicada de los servidores Amazon EC2 poseen una capa de red virtual denominada como VPC o Amazon Virtual Private Cloud la cual brinda un control mucho más granular sobre la seguridad. Finalmente contamos con el servidor Apache [12], el cual se encuentra ejecutándose sobre la instancia de Linux del sistema operativo, este se encargará de recibir todas las solicitudes y peticiones provenientes del cliente, para luego redirigirlas a la aplicación correspondiente a través de la configuración de nombres de dominio o DNS de cada una. Apache es una colección de proyectos que incluyen un servidor web (Apache http) y un servidor de aplicaciones (tomcat), además de muchas otras bibliotecas y sistemas de middleware [13]. A continuación se resumen los recursos técnicos asociados a la instancia del servidor Amazon EC2 [14].

Hardware	Descripción
Procesador	4 Núcleos virtuales o CPU con arquitectura x86.
Memoria	16 gb
Almacenamiento	128 gb (EFS)
Sistema Operativo	Linux (Cualquier distribución)
Servidor	Apache server 2.4.52 (Tomcat)
Sistema de Virtualización	Amazon Virtual Private Cloud (VPC)
Interfaz Administrativa	Amazon Web Services (AWS)
Protocolos de Comunicación	Protocolos TCP/IP y HTTP/TLS.

Diagramas de Componentes

Para una mejor comprensión de la arquitectura de Espinven móvil y la implementación del patrón modelo vista vista-modelo (MVVM), a continuación se presenta un diagrama de componentes que representan las distintas interacciones entre los mismos con sus interfaces y servicios:



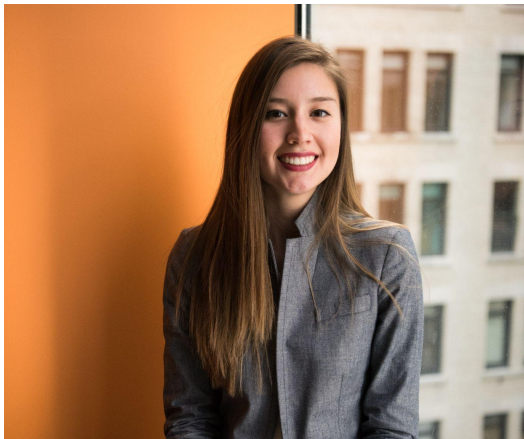
Espinven móvil se encuentra conformado por cinco componentes independientes que interactúan entre sí a través de tres interfaces. La primera interfaz es la “SQL”, constituida por un lenguaje estructurado de bajo nivel (SQL) y un conjunto de procedimientos particulares del manejador de bases de datos (MariaDB). La segunda interfaz es la de RequestManager, que es una clase compuesta por el conjunto de métodos y funciones que permiten el manejo y entrega de peticiones y sus respuestas. Y por último, tenemos la interfaz de los State Holders, que son contenedores de estados que contienen datos, los exponen a la IU y controlan la lógica [1][2][3][4][5][7].

Componente	Descripción
Views Resources	Views Resources es el componente que contempla toda la capa de UI y los diferentes elementos y estados que la componen para la creación de las vistas e interfaces de usuario[1].
BLOC	BLOC es el componente principal del patrón del mismo nombre, que ayuda a la gestión de los estados, operando la lógica del negocio y entregándola a la capa UI, y en el acceso a los datos desde un lugar central del proyecto y que a su vez, representa la capa ViewModel del patrón VMMV [4].
App Models	Los App Models es un componente que permite el empaquetamiento estructurado de los datos, tanto los de envío de peticiones, como los de respuesta de los mismos, sin importar si se está utilizando el Data Source o el repositorio externo.
Data Sources	Este componente le ofrece una ventaja al móvil, al permitir el almacenamiento interno o local del dispositivo de datos e información.
MariaDB	Es el componente del manejador de base de datos utilizado por Flutter a través del lenguaje SQL y se encarga de gestionar el repositorio y la base de datos[7].

Perfiles de Usuario

Jefe	
Datos demográficos	
Edad	Mayor de 25 años.
Género	Masculino o Femenino.
Formación	Requiere un nivel de formación de Pregrado o Técnico.
Experiencia	Requiere conocimientos básicos en el uso de aplicaciones móvil.
Requerimientos asociado	
IDs de los requerimientos PIECES	
REQ2	El perfil de usuario de tipo Jefe podrá consultar el histórico de las tareas de los empleados con sus estatus para evaluar el desempeño de los mismos, así como también podrá consultar los porcentajes de sustentabilidad del restaurante con sus históricos para medir la cantidad de ingredientes comprados, con respectos a los utilizados y desperdiciados parametrizados por fecha.
REQ4	El perfil de usuario de tipo Jefe podrá designar y modificar cargos de gerentes y proveedores.
Ambiente Físico	
Estos usuarios podrán acceder a la aplicación móvil a través de un dispositivo electrónico con SO Android o iOS y con acceso a internet desde su área de trabajo.	

A continuación se mostrará una breve descripción de un usuario que cumpla con estas características



Laura García es Administradora de Empresas que reside en Barcelona España. Nació en Barquisimeto, y a los 18 años se mudó a Caracas para estudiar su carrera de pregrado. Después de obtener su título universitario consiguió trabajo en una compañía en la ciudad de Barcelona, España; y allí conoció a su esposo, con el que tiene 3 hijos. Desde pequeña siempre tuvo interés por la cocina y la buena gastronomía, por lo que decidió abrir su propio restaurante, que ha llegado a convertirse en uno de los más reconocidos y exitosos de la ciudad de Barcelona.

Gerente	
Datos demográficos	
Edad	Mayor de 25 años.
Género	Masculino o Femenino.
Formación	Requiere un nivel de formación de Pregrado o Técnico.
Experiencia	Requiere conocimientos básicos en el uso de aplicaciones móvil.
Requerimientos asociado	
IDs de los requerimientos PIECES	
REQ1	El perfil de usuario de tipo gerente podrá administrar el inventario del restaurante a través de la consulta, eliminación o modificación de ingredientes, y será el encargado de su contabilización.
REQ2	El perfil de usuario de tipo gerente podrá revisar las estadísticas de desempeño de los empleados y niveles de sustentabilidad del restaurante en tiempo real.
REQ4	El perfil de usuario de tipo gerente podrá administrar la suspensión, bloqueo y validación de las cuentas de los empleados y proveedores.
Ambiente Físico	
Estos usuarios podrán acceder a la aplicación móvil a través de un dispositivo electrónico con SO Android o iOS y con acceso a internet desde su área de trabajo.	

A continuación se mostrará una breve descripción de un usuario que cumpla con estas características:



Xavier Soldevilla es un Magíster en Gerencia de Proyectos que reside en Barcelona, España. Nació en Valencia, España. A los 22 años se mudó a Barcelona para estudiar en la Universitat de Barcelona. Después de obtener su posgrado comenzó a trabajar en una franquicia de restaurantes de comida rápida y se certificó en manejo de alimentos y bebidas. Un tiempo después, un accionista de la franquicia le ofreció trabajo como gerente de un restaurante, y Xavier decidió aceptar la propuesta. Hoy en día se desempeña como gerente general de un reconocido restaurante en la ciudad de Barcelona.

Empleado Mesonero	
Datos demográficos	
Edad	Mayor de 18 años.
Género	Masculino o Femenino.
Formación	No requiere un grado de instrucción particular.
Experiencia	Requiere conocimientos básicos en atención al cliente y manejo de aplicaciones móviles.
Requerimientos asociado	
IDs de los requerimientos PIECES	
REQ18	El perfil de usuario de tipo empleado mesero podrá tomar, crear y gestionar pedidos en base al menú del restaurante.
Ambiente Físico	
Estos usuarios podrán acceder a la aplicación móvil a través de un dispositivo electrónico con SO Android o iOS y con acceso a internet desde su área de trabajo.	

A continuación se mostrará una breve descripción de un usuario que cumpla con estas características



Juan Maldonado es un estudiante de 8vo Semestre de Filosofía nacido en Montpellier, Francia. Tiene 20 años y desde hace casi 2 años trabaja como mesero para costear sus estudios y gastos básicos. Desde entonces, Juan no ha dejado de ganarse el cariño y el respeto de todos los comensales, ya que si hay algo que lo caracteriza a él es su encantadora sonrisa, su excelente servicio y sus habilidades de conversación, por lo cual la experiencia de los clientes ya no es solo degustar sus paladares, sino también sonreír y disfrutar un rato de la compañía de Juan.

En función del grupo de estándares y normas ISO/IEC 25000:2014 y las distintas características del SI, este contará con un conjunto de cualidades y atributos que se describen a continuación [8]:

- **Facilidad de aprendizaje:** Con la finalidad de cumplir con la característica de usabilidad *Facilidad de aprendizaje*, el SI contará con un conjunto de etiquetas descriptivas de las funcionalidades y resultados esperados de la aplicación tanto en formularios como acciones en el sistema. Tomando en cuenta los usuarios potenciales también se implementará una retroalimentación de los resultados de las transacciones realizadas, con la finalidad de maximizar el aprendizaje a cabalidad de las acciones en el sistema.
- **Facilidad de entendimiento:** Con la finalidad de cumplir con la característica de usabilidad *Facilidad de entendimiento* el SI contará con un tamaño de letra de 13px así como también colores de alto contraste, para así maximizar el entendimiento y la legibilidad de la aplicación. Además, se utilizarán dialogs para informar a los usuarios acerca de cualquier resultado de acción o modificación dentro de la aplicación. Finalmente, otras de las técnicas a utilizar para facilitar el entendimiento de la aplicación por parte de los usuarios, será el uso de íconos minimalistas representativos en los títulos, subtítulos y formularios con la finalidad de la identificación y comprensión visual de las palabras o frases asociadas.
- **Facilidad de uso:** Con la finalidad de cumplir con la característica de usabilidad *Facilidad de uso*, el SI contará con interfaces que faciliten la navegación a través de la aplicación de manera que el usuario pueda observar a simple vista aquellas herramientas que necesite utilizar, así como también se desplegarán mensajes de error o éxito con su explicación en caso de que se intente realizar una acción no permitida o válida. Otra de las formas en la que también se garantizará esta característica de usabilidad es a través de la creación de un sitio web o wiki de consulta, en donde se encontrará la documentación respectiva a la guía de usuario, con el objetivo de ayudar a los usuarios en caso de cualquier duda o problema al momento de utilizar el sistema.
- **Grado de atracción:** Con la finalidad de cumplir con la con la característica de usabilidad *Grado de atracción*, el SI contará con un diseño de interfaces estandarizado y uniforme tanto entre todas las interfaces de la aplicación web como móvil, implementando distintos patrones de diseño para los botones, ubicación de las acciones, títulos, subtítulos, estilos y colores así como también se mantendrá una organización y clasificación del contenido estándar entre las interfaces y aplicaciones con el objetivo maximizar su comprensión y minimizar el tiempo de búsqueda espacial o visual.

En función de estándares de diseño y las distintas características del SI, este contará con un conjunto de principios de diseño que se describen a continuación [9][10][11]:

- **Simplicidad:** Con la finalidad de garantizar la comprensión y generar una mayor probabilidad de recordar cómo se usa el sistema, se utilizará el principio de diseño *Simplicidad*, sintetizando y eliminando información que no sea necesaria.
- **Estructura:** Con la finalidad de garantizar el cumplimiento de las reglas de negocio en la automatización de los procesos, en el diseño de interfaces se tomarán en cuenta las secuencias de pasos detalladas en los diagramas de actividades. A continuación se presentan los principios y patrones de diseño a considerar para la aplicación:
 - **Principios orientados en patrones de “What Users Do”:** Entre los principios basados en los patrones de “What Users Do” [9] se utilizarán los siguientes:
 - **Spatial Memory:** Se basa en el principio de que las personas tienden a encontrar las cosas más fácilmente a través de la memoria espacial que por su nombre.
 - De igual forma que en los elementos de usabilidad, en el apartado de Grado de atracción una de las características que debe cumplir el SI corresponde a que todos los elementos de las aplicaciones deben colocarse en una ubicación uniforme y estándar, de manera que el usuario logre encontrarlos fácilmente.
 - **Streamlined Repetition:** Proporciona al usuario una manera fácil de repetir aquellas actividades que deben realizar de forma repetitiva.
 - Se incorporarán elementos que faciliten la navegación y señalización, al igual que también se agregará la automatización de ciertos procesos repetitivos para reducir la carga de trabajo.
 - **Principios orientados en patrones de “Organizing the Content: Information Architecture and Application Structure”:** Entre los principios basados en los patrones de “Organizing the Content: Information Architecture and Application Structure” [9] se utilizarán los siguientes:
 - **Dashboard:** Propone organizar las pantallas de datos en una sola página densa en información, actualizada periódicamente. Muestra a los usuarios información relevante y procesable.
 - El sistema contará con una página de inicio que contendrá un dashboard con las acciones más recientes o utilizadas en la aplicación, así como también el menú inferior que redirige a cualquier módulo.

- **Consistencia:** Con la finalidad de garantizar la consistencia y la uniformidad en la aplicación, se definirán los colores, fuentes, tamaños de fuentes, estilos, diseños de botones, notificaciones y mensajes para el diseño de las interfaces, de forma tal que sea unificado y estandarizado a lo largo de todas las interfaces de la aplicación tanto móvil como web.
- **Tolerancia:** Con la finalidad de minimizar los errores de uso, se proporcionarán ayudas que faciliten al usuario la ejecución de una tarea en la aplicación, así como también aquellos valores que sean predefinidos se mostrarán directamente en la aplicación para que los usuarios simplemente elijan aquel que deseen.
- **Accesibilidad:** Con la finalidad de garantizar la accesibilidad al sistema por parte de personas con discapacidades visuales, auditivas o motoras se implementarán métodos abreviados de teclados, se garantizará la compatibilidad con lectores de pantalla como VoiceOver, y se podrá customizar el tamaño de fuente de las interfaces en caso de que sea necesario.
- **Feedback:** A lo largo de toda la aplicación móvil, el usuario irá recibiendo por parte de esta última un feedback sobre sus acciones, que le permitan saber al usuario que el sistema está procesando correctamente las peticiones realizadas. Esto se puede encontrar por ejemplo, en cambios de color en la barra inferior del menú que indican en qué módulo de la app se encuentra el usuario.

Referencias Bibliográficas

- [1] Android Developers. (2021, 5 enero). *Guía de arquitectura de apps*, [en línea]. Recuperado 25 de enero de 2022, de <https://developer.android.com/jetpack/guide>
- [2] Mohite, J. (2021, 23 diciembre). *Flutter: MVVM Architecture*, [en línea]. Medium. Recuperado 25 de enero de 2022, de <https://medium.com/flutterworld/flutter-mvvm-architecture-f8bed2521958>
- [3] *MVVM en Android con Kotlin, LiveData y View Binding*. (2021, 22 abril). [en línea]. Curso Kotlin Para ANDROID. Recuperado 26 de enero de 2022, de <https://cursokotlin.com/mvvm-en-android-con-kotlin-livedata-y-view-binding-android-architecture-components/>
- [4] Vega, C. (2021, 9 diciembre). *Implementa Arquitectura a tu proyecto Flutter usando el patrón BLOC*, [en línea]. Medium. Recuperado 26 de enero de 2022, de <https://medium.com/comunidad-flutter/implementa-arquitectura-a-tu-proyecto-flutter-usando-el-patrn%C3%B3n-bloc-2cb031722166>
- [5] *Which protocols were used in mobile apps to connect to the back-end side?* (2015). [en línea]. Quora. Recuperado 26 de enero de 2022, de <https://www.quora.com/Which-protocols-were-used-in-mobile-apps-to-connect-to-the-back-end-side>
- [6] Amazon Web Services. (2022). *AWS | Elastic load balancing para la gestión de tráfico en aplicaciones*, [en línea]. Amazon Web Services, Inc. Recuperado 26 de enero de 2022, de <https://aws.amazon.com/es/elasticloadbalancing/>
- [7] *MariaDB Foundation*. (2022). MariaDB.Org, [en línea]. Recuperado 26 de enero de 2022, de <https://mariadb.org>
- [8] ISO. (2014, marzo). *ISO/IEC 25000:2014*, [en línea]. Recuperado 27 de enero de 2022, de <https://www.iso.org/standard/64764.html>
- [9] Tidwell, J. (2010). *Designing Interfaces: Patterns for Effective Interaction Design* (2nd ed.), [en línea]. O'Reilly Media. Recuperado 28 de enero de 2022, de <http://bedford-computing.co.uk/learning/wp-content/uploads/2016/07/Livro-Designing-Interfaces-2nd-Edition-2010.pdf>

- [10] Norman, D. (2013). *The Design of Everyday Things*, [en línea]. Adfo Books.. Recuperado 28 de enero de 2021, de <http://kowym.com/wp-content/uploads/2018/08/The-Design-of-Everyday-Things-Don-Norman.pdf>
- [11] Rosenfeld, L., Morville, P., & Arango, J. (2015). *Information Architecture for the World Wide Web* (4.a ed.), [en línea]. Van Duuren Media. Recuperado 28 de enero de 2021, de https://e-edu.nbu.bg/pluginfile.php/62325/mod_resource/content/1/Information_Architecture_For_The_Web_And_Beyond_Fourth_Edition.pdf
- [12] B., G. (2022, 1 febrero). ¿Qué es Apache? Descripción completa del servidor web Apache. Tutoriales Hostinger. [¿Qué es apache?](#)
- [13] Sorensen, M. (2020). Is Apache a web server or an application server? Quora. Recuperado 3 de febrero de 2022, de [Apache Server](#)
- [14] Tipos de instancias de Amazon EC2 - Amazon Web Services. (s. f.). Amazon Web Services, Inc. Recuperado 3 de febrero de 2022, de [Tipos de instancias](#)

A continuación se presenta una referencia al SRS de la clasificación de PIECES de los requerimientos tanto funcionales como no funcionales de Espinven Móvil:

Id	Requerimiento	P	I	E	C	E	S
REQ1	Automatizar los procesos de gestión y manejo de inventario.		X	X			
REQ2	<p>Diseñar e implementar módulo de estadísticas que incluya los siguientes reportes/métricas:</p> <p>A. Para los empleados: Total y porcentaje de tareas completadas, pendientes y no realizadas (Desempeño individual).</p> <p>B. Para los gerentes:</p> <p>a. Desempeño de los empleados: Cantidad de tareas con sus status parametrizado por empleado.</p> <p>b. Porcentaje de los niveles de sustentabilidad del restaurante: Cantidad de ingredientes comprados vs utilizados y desperdiciados.</p> <p>C. Para los dueños:</p> <p>a. Histórico del desempeño de los empleados: Cantidad de tareas con sus status, parametrizado por rango de fechas.</p> <p>b. Histórico de los niveles de sustentabilidad del restaurante: Porcentaje de los ingredientes comprados vs utilizados y desperdiciados en un intervalo de fecha.</p>		X	X			
REQ3	Diseñar e implementar una base de datos relacional unificada que almacene la información correspondiente a cada uno de los procesos.	X				X	X
REQ4	<p>Automatizar los procesos de gestión de cuentas, roles y controles de acceso:</p> <p>A. Para los dueños: Designación de gerentes.</p> <p>B. Para los gerentes: Suspensión, bloqueo y validación de los empleados.</p>				X		X
REQ5	<p>Diseñar e implementar interfaces utilizando patrones y estándares de diseño con la finalidad de garantizar flujos y navegaciones similares entre plataformas.</p> <p>Patrones:</p> <ul style="list-style-type: none"> Getting Input from Users - Forms and Controls. Making It Look Good - Visual Style and Aesthetics. <p>Estándares:</p> <ul style="list-style-type: none"> Estándar 3.3 de la WCAG 2.1 de W3C [19]. <ul style="list-style-type: none"> 3.3.1 Error Identification. 3.3.2 Labels or Instructions 						X

REQ6	Configurar y desplegar las aplicaciones Espinven Movil y Espinven Web y base de datos MySQL (XAMPP) en servidores AWS (Servicios en la nube).						X
REQ7	Diseñar y dictar dos cursos de entrenamiento especializado para los usuarios finales de las aplicaciones tanto móvil como web.						X
REQ8	Diseñar manuales de usuario con documentación referencial para cada una de las aplicaciones tanto móvil como web desplegadas en una wiki de forma online.						X
REQ9	Integrar y comunicar la base de datos con la aplicación web y móvil.						X
REQ18	Automatizar el proceso de creación de pedidos del menú.		X	X			
REQ19	Diseñar e implementar una comunicación entre la APP con la WEB para enviar pedidos del menú.						X
REQ20	Diseñar e implementar los modelos lógicos y clases funcionales del Back-End con su documentación técnica.						X
REQ21	Diseñar e implementar arquitecturas de software basadas en el patrón BLOC con el objetivo de garantizar un menor tiempo de respuesta y proporcionar una mayor seguridad y extensibilidad. Este patrón utiliza el Framework Flutter y el lenguaje Dart como únicos recursos en cada una de sus 4 etapas.	X				X	
REQ22	Configurar y desplegar la aplicación en las tiendas de Android y iOS (Play Store y App Store respectivamente).	X					X