|  |  |
| --- | --- |
|  |  |
| OutfitForYou |  |
|  |  |
|  | Abril -- 2024Diseño Detallado de Software |
|  | Vivian Echeverri LondoñoHector Manuel Vanegas |

# Introducción

La intención de este documento es presentar, de manera concisa pero completa, las metodologías y enfoques que guiarán el desarrollo de los componentes, así como los patrones más relevantes a utilizar. Se delimitarán el alcance y contexto del proyecto, estableciendo definiciones clave para el diseño técnico, drivers, requerimientos funcionales significativos, restricciones y atributos de calidad. Además, se explorarán las vistas que conforman la arquitectura, junto con su modelo subyacente, para finalmente abordar los riesgos inherentes a esta estructura.

Desde una perspectiva más amplia este proyecto surge con la idea de abordar diversas problemáticas y necesidades presentes en la sociedad con relación al ámbito de la moda y la vestimenta. La moda, entendida como el “gusto colectivo y cambiante en lo relativo a prendas de vestir y complementos”, ha sido constante a lo largo de la historia humana, siendo mucho más que una mera tendencia pasajera. Por el contrario, ha sido una manifestación de libertad, un requerimiento social, y una expresión tangible de la cultura.

## Objetivo

El objetivo de este documento es establecer el diseño técnico para el proyecto de desarrollo de una aplicación destinada a la creación de outfits personalizados. Este diseño se fundamentará en los dominios de negocio, aplicación, tecnología y datos definidos en el framework de TOGAF. Se buscará proporcionar una estructura sólida y coherente que permita la integración eficiente de los diferentes aspectos del proyecto, desde los requerimientos del negocio hasta las consideraciones tecnológicas más avanzadas. En última instancia, el objetivo es garantizar que la aplicación resultante cumple con los estándares de calidad, seguridad y usabilidad requeridos, ofreciendo una experiencia óptima para los usuarios finales.

## Definiciones, Acrónimos y Abreviaturas

App: Abreviatura de “aplicación”, se refiere a un programa infromativo diseñado para cumplir una función específica en dispositivos móviles u ordenadores.

Outfit: Término en inglés que se refiere a un conjunto de prendas de vestir y accesorios combinados para crear un cinjunto armonioso.

Diseño Técnico: Proceso de definición detallada de la estructura, funcionamiento y aspecto visual de una aplicación o sistema informático.

Arquitectura basada en Microservicios: Que es un enfoque arquitectónico para desarrollar una aplicación como un conjunto de servicios pequeños e independientes, cada uno ejecutando un proceso único y específico.

Java: Es un lenguaje de programación interpertado, concurrente, orientado a objetos, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible.

Spring Boot: Proporciona un conjunto de herramientas para construir rápidamente aplicaciones, basándose en una infraestructura ligera que elimina la mayor parte del trabajo de configurar las aplicaciones basadas en Spring.

Requerimientos Funcional: Define el comportamiento interno del Software: cálculos, detalles técnicos, manipulación de datos y otras funcionalidades que definen como los casos de uso serán satisfechos.

Requerimientos No Funcional: Un requerimiento que especifica criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos.

Requerimiento Funcional Significativos: Describen las acciones que el sistema debe realizar, las operaciones que debe admitir o las caracteristicas que debe tener para satisfacer las necesidades de los usuarios finales. Implica que estos requerimientos son críticos o importantes para el funcionamiento y la utilidad del sistema. Pueden ser indentificados mediante la colaboración entre los stakeholders del proyecto, como los usuarios, los clientes, los analistas de negocio y los desarrolladores, quienes determinan qué funciones son esenciales para alcanzar los objetivos del proyecto.

## Alcance

El alcance de este proyecto consiste en el desarrollo de una aplicación móvil para la creación de outfits personalizados, con el objetivo de proporcionar a los usuarios una experiencia interactiva y personalizado en el ámbito de la moda y la vestimenta. La aplicación permitirá a los usuarios explorar, seleccionar y personalizar conjuntos de ropa t accesorios de acuerdo con sus preferencias de estilo y tendencias de moda actuales. El alcance del proyecto incluirá las siguientes funcionalidades principales:

* Funcionalidades Principales:

Creación de perfiles de usuario.

Búsqueda y navegación de prendas de vestir y accesorios.

Visualización de conjuntos de ropa recomendados.

Personalización de conjuntos según preferencias de estilo y ocasiones específicas.

Integración con redes sociales para compartir outfits y recibir feedback.

Funcionalidades de compra, incluyendo carrito de compras y proceso de pago.

* Gestión de Contenido:

Base de datos de prendas de vestir y accesorios, incluyendo imágenes, descripciones, precios y tallas disponibles.

Herramientas con servicios de administración de contenido para agregar, editar y eliminar productos.

* Integraciones Externas:

Integración con APIs de proveedores externos de moda para obtener información actualizada sobre productos, precios y disponibilidad de inventario.

Integración con servicios de pago en línea para facilitar transacciones seguras.

Integración con redes sociales para compartir outfits y acceder a datos de perfil de usuario.

* Experiencia del Usuario:

Interfaz de usuario intuitiva y atractiva.

Experiencia de navegación fluida y receptiva.

Personalización de la experiencia según las preferencias del usuario.

Funcionalidades de retroalimentación y soporte al cliente.

Arquitectura:

La arquitectura de la aplicación se basará en un enfoque de microservicios, que proporcionará una estructura técnica flexible y escalable para el desarrollo y la implementación del sistema. La arquitectura constará de los siguientes componentes principales:

* Fronted

Interfaz de usuario desarrollada utilizando tecnología web como HTML, CSS y JavaScript, posiblemente con un marco de trabajo como React.js o Angular Js.

Lógica del cliente para manejar la interacción del usuario, las solicitudes de servidor y la presentación de datos.

* Backend

Servidor de aplicaciones desarrollado utilizando tecnologías como Java Spring

Base de datos para almacenar información sobre usuarios, productos y transacciones

* Servicios de integración

API RESTful para facilitar la comunicación entre el front y el back, así como la integración con servicios externos.

Integración con APIs de proveedores externos de moda y servicios de pago en línea.

* Motor de recomendaciones

Utiliza algoritmos de recomendaciones para sugerir conjuntos personalizados basados en las preferencias del usuario y las tendencias de moda actuales.

* Seguridad

Implementación de medidas de seguridad como autenticación de usuarios, cifrado de datos y protección contra ataques.

* Infraestructura

Despliegue en la nube utilizando servicios de proveedores como AWS, Google Cloud y Microsoft Azure.

Contenedores Docker para facilitar el despliegue y la gestión en entornos de desarrollo y producción.

La arquitectura propuesta proporcionará una base sólida y escalable para el desarrollo de la aplicación de creación de outfits personalizados, garantizando una experiencia de usuario óptima y cumpliendo con los objetivos del proyecto.

# Contexto

## Fundamentos de la Solución

Los patrones de diseño sirven para dar soluciones comunes y eficientes a problemas de la arquitectura. Los patrones que se utilizan en esta arquitectura son los siguientes:

|  |  |
| --- | --- |
|  |  |
| Forma  Descripción generada automáticamente |  |

Ilustración 1. Patrones de Arquitectura

## Decisiones de Arquitectura

Se proponen los siguientes patrones que se utilizan en la etapa de Arquitectura:

|  |  |  |
| --- | --- | --- |
| Cod.  Decición | Nombre | Descripción |
| DA01 | Microservicios | Se implementa como patrón nuclear el estilo de arquitectura Microservicios, la cual se centra en componentes desacoplados y cada uno con su propia responsabilidad y sus recursos específicos, permitiendo realizar, cuando se requiera escalabilidad de recursos y generando una arquitectura modular permitirá modificar o agregar nuevas funcionalidades al sistema. |
| DA02 | Clean  Arquitecture | Se implementa como patrón complementario Clean Architecture (Hexagonal – Puertos y Adaptadores), con el fin de evitar el acoplamiento de nuestro dominio con elementos externos lo que producirá sistemas:  Independientes del framework y proveedor.  Testables.  Independientes de la UI  Independientes de la base de datos.  Independiente de agentes externos  Aumentar la tolerancia al cambio.  Reutilizables.  Mantenibles. |

## Decisiones de Diseño

Se proponen los siguientes principios que se utilizarán en la etapa de Diseño

|  |  |  |
| --- | --- | --- |
| Cod.  Decisión | Nombre | Descripción |
| DD01 | SRP | Single Responsability Principle o Principio de Responsabilidad Única |
| DD02 | OCP | Open/Closed Principle o Principio de Abierto/Cerrado |
| DD03 | LSP | Liskov Substitution Principle o Principio de Sustitución de Liskov |
| DD04 | ISP | Interface Segregation Principle o principio de Segregación de la Interfaz |
| DD05 | DIP | Dependency Inversión Principle o Principio de Inversión de Dependecia |

# Drivers de Arquitectura

## Objetivos de la Arquitectura de la solución

Los objetivos de la arquitectura de la solución son las metas y principios que guían el diseño y la implementación del sistema. Estos objetivos se centran en garantizar que la arquitectura satisfaga las necesidades del proyecto y cumpla con los requisitos de los stakeholders. Algunos de los objetivos comunes de la arquitectura de la solución para una aplicación de creación de outfits personalizados podrían incluir:

* Escalabilidad: La arquitectura debe ser capaz de crecer y adaptarse a medida que aumenta la cantidad de usuarios y la carga de trabajo, garantizando un rendimiento óptimo en todo momento.
* Flexibilidad: Debe ser flexible y modular, permitiendo la incorporación de nuevas funcionalidades y la integración con sistemas externos de manera sencilla y sin afectar a otras partes del sistema.
* Seguridad: Debe garantizar la seguridad de los datos del usuario, protegiendo la información personal y financiera contra accesos no autorizados y ataques maliciosos.
* Usabilidad: La arquitectura debe proporcionar una experiencia de usuario intuitiva y agradable, asegurando que la aplicación sea fácil de usar y accesible para usuarios de diferentes niveles de habilidad.
* Mantenibilidad: Debe ser fácil de mantener y actualizar, permitiendo la identificación y corrección rápida de errores, así como la implementación de mejoras y nuevas características de manera eficiente.
* Rendimiento: Debe garantizar un rendimiento rápido y eficiente, minimizando los tiempos de carga y respuesta para ofrecer una experiencia fluida y receptiva a los usuarios.
* Integración: Debe facilitar la integración con sistemas externos, como APIs de proveedores de moda y servicios de pago en línea, permitiendo una colaboración fluida y una experiencia de usuario mejorada.

Al definir y cumplir con estos objetivos, la arquitectura de la solución asegura que la aplicación de creación de outfits personalizados sea robusta, segura y altamente funcional, cumpliendo con las expectativas de los usuarios y los requisitos del negocio.

## Atributos de Calidad