

MODELO DE CALIDAD MONOLÍTICO

ASTAIZA CASTRO URIEL ARMANDO

MOYA CALLEJAS BRIAN STEVEN

QUIROGA SILVA LUIS ALFONSO

FICHA: 1803184

SERVICIO NACIONAL DE APRENDIZAJE -SENA-
ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN -ADSI-
OCTAVO TRIMESTRE
INSTRUCTOR: DILINGER ARAGÓN PEREA

2021

TABLA DE CONTENIDO

1.	INTRODUCCIÓN	4
2.	PARÁMETROS Y ESPECIFICACIONES DEL MODELO A LA MEDIDA MONOLÍTICO	5
2.1	DEFINICIÓN ARQUITECTURA MONOLÍTICA	5
2.2	FACTORES ESENCIALES	5
2.3	CARACTERÍSTICAS	5
2.4	VENTAJAS	6
2.4.1	Fácil de desarrollar	6
2.4.2	Fácil de escalar	6
2.4.3	Pocos puntos de fallo	6
2.4.4	Autónomo	6
2.4.5	Fácil de probar	6
2.5	DESVENTAJAS	7
2.5.1	Anclado a un Stack tecnológico	7
2.5.2	Escalado Monolítico	7
2.5.3	El tamaño sí importa	7
2.5.4	Versión tras versión	7
2.5.5	Si falla, falla todo	7
2.5.6	Es fácil perder el rumbo	7

2.5.7	Puede ser abrumador	8
2.6	PASOS A IMPLEMENTAR.....	8
2.7	RECOMENDACIONES GENERALES	8
3.	CONCLUSIÓN	9
4.	CIBERGRAFÍA	10

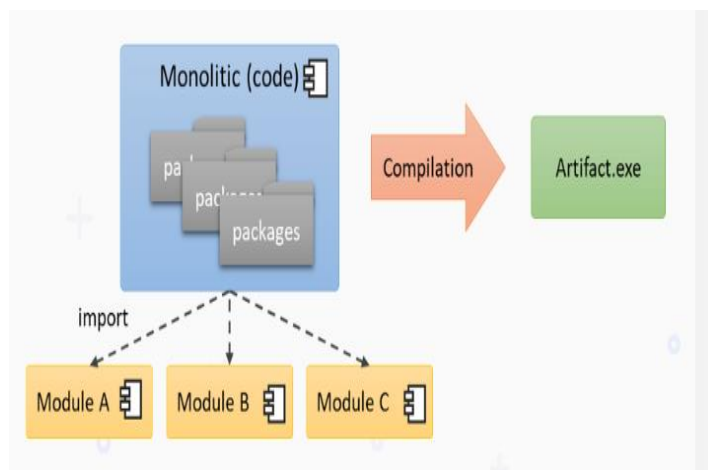
1. INTRODUCCIÓN

El modelo de calidad que se va a implementar es el “*Modelo a la Medida Monolítico*”, ya que con él se empleará una marca única, que consiste en crear una aplicación autosuficiente que contenga absolutamente toda la funcionalidad necesaria para realizar la tarea para la cual fue diseñada, sin contar con dependencias externas que complementen su funcionalidad. Además, se desea recalcar, que se empleará el nombre como único nombre unificador para futuros proyectos.

2. PARÁMETROS Y ESPECIFICACIONES DEL MODELO A LA MEDIDA

MONOLÍTICO

2.1 DEFINICIÓN ARQUITECTURA MONOLÍTICA



Consiste en crear una aplicación autosuficiente que contenga absolutamente toda la funcionalidad necesaria para realizar la tarea para la cual fue diseñada, sin contar con dependencias externas que complementen su funcionalidad. En este sentido, sus componentes trabajan juntos, compartiendo los mismos recursos y memoria. En pocas palabras,

una aplicación monolítica es una unidad cohesiva de código.

2.2 FACTORES ESENCIALES

Según David Aaker, un modelo monolítico potencia tres (3) factores esenciales para el desarrollo de una marca:

- **Claridad**, porque la marca madre o principal garantiza tranquilidad a la hora de comprar, y da valor y calidad a los productos.
- **Sinergias**, brindando experiencia, publicidad y calidad al producto ofrecido.
- **Apalancamiento**, permitiendo la ampliación y divulgación de la marca.

2.3 CARACTERÍSTICAS

Las características se pueden convertir en ventajas o desventajas solo cuando analizamos la problemática que vamos a resolver, mientras tanto, son solo características:

- Son aplicaciones autosuficientes (no requieren de nada para funcionar).
- Realizan de punta a punta todas las operaciones para terminar una tarea.
- Son por lo general aplicaciones grandes, un que no es un requisito.
- Son por lo general silos de datos privados, es decir, cada instalación administra si propia base de datos.

- Todo el sistema corre sobre una solo plataforma.

2.4 VENTAJAS

2.4.1 Fácil de desarrollar

Debido a que solo existe un componente, es muy fácil para un equipo pequeño de desarrollo iniciar un nuevo proyecto y ponerlo en producción rápidamente.

2.4.2 Fácil de escalar

Solo es necesario instalar la aplicación en varios servidores y ponerlo detrás de un balanceador de carga.

2.4.3 Pocos puntos de fallo

El hecho de no depender de nadie más, mitiga gran parte de los errores de comunicación, red, integraciones, etc. Prácticamente los errores que pueden salir son por algún bug del programador, pero no por factores ajenos.

2.4.4 Autónomo

Las aplicaciones Monolíticas se caracterizan por ser totalmente autónomas (auto suficientes), lo que les permite funcionar independientemente del resto de aplicaciones.

Performance: Las aplicaciones Monolíticas son significativamente más rápidas debido que todo el procesamiento lo realizan localmente y no requieren consumir procesos distribuidos para completar una tarea.

2.4.5 Fácil de probar

Debido a que es una sola unidad de código, toda la funcionalidad está disponible desde el inicio de la aplicación, por lo que es posible realizar todas las pruebas necesarias sin depender de nada más.

2.5 DESVENTAJAS

2.5.1 Anclado a un Stack tecnológico

Debido a que todo el software es una sola pieza, implica que utilicemos el mismo Stack tecnológico para absolutamente todo, lo que impide que aprovechemos todas las tecnologías disponibles.

2.5.2 Escalado Monolítico

Escalar una aplicación Monolítica implica escalar absolutamente toda la aplicación, gastando recursos para funcionalidad que quizás no necesita ser escalada (en el estilo de Microservicios analizaremos como solucionar esto).

2.5.3 El tamaño sí importa

Sin albur, las aplicaciones Monolíticas son fácil de operar con equipo pequeños, pero a medida que la aplicación crece y con ello el equipo de desarrollo, se vuelve cada vez más complicado dividir el trabajo sin afectar funcionalidad que otro miembro del equipo también está moviendo.

2.5.4 Versión tras versión

Cualquier mínimo cambio en la aplicación implicará realizar una compilación del todo el artefacto y con ello una nueva versión que tendrá que ser administrada.

2.5.5 Si falla, falla todo

A menos que tengamos alta disponibilidad, si la aplicación Monolítica falla, falla todo el sistema, quedando totalmente inoperable.

2.5.6 Es fácil perder el rumbo

Por la naturaleza de tener todo en un mismo módulo es fácil caer en malas prácticas de programación, separación de responsabilidades y organización de las clases del código.

2.5.7 Puede ser abrumador

En proyectos muy grandes, puede ser abrumador para un nuevo programador hacer un cambio en el sistema.

2.6 PASOS A IMPLEMENTAR

Para poder implementar este tipo de modelo, lo ideal sería el tener otros nombres de productos previos que hagan sinergia con el nombre de nuestro sistema que es **API**. Para ello haríamos un paso a paso que se aplicará cuando se genere un producto que posea características similares. Para ello tendremos en cuenta:

1. Inicialmente, se debe crear otro sistema como punto de partida
2. A continuación, se toma el nombre del producto madre, del cual se desprenderían una o varias palabras de la nueva funcionalidad que vaya a tener dicho producto.
3. Finalmente, sería lanzado al mercado para posterior uso de los clientes

2.7 RECOMENDACIONES GENERALES

Con respecto a este punto, podemos decir que las recomendaciones empezarían a ser viables desde el momento en que se cree y ponga en funcionamiento el sistema madre, y se haya creado uno o varios sistemas alternos.

3. CONCLUSIÓN

Como se pudo observar el sistema monolítico escogido, permitirá al usuario del mismo, el poder interactuar de manera activa y directa al nuevo sistema.

4. CIBERGRAFÍA

<https://knoow.net/es/cieeconcom/gestion/modelo-monolitico/>

<https://reactiveprogramming.io/blog/es/estilos-arquitectonicos/monolitico>