

Ejercicios Solemne 2

Marcelo A. Yáñez
1er Semestre, 2023

1. Ya sabemos que todo beneficio tiene un costo. Analice cuales son los costos involucrados (considerando los atributos de calidad) al decidir incorporar en el diseño de un sistema cada uno de los siguientes requerimientos no funcionales:

- (a) Confiabilidad
- (b) Soportabilidad
- (c) Verificabilidad

Respuesta:

- (a) Al incorporar confiabilidad se tiene como costo:

- Verificabilidad
Para añadir verificabilidad a un sistema confiable, se deben realizar no sólo pruebas en el sistema en sí, se deben realizar pruebas en los sistemas de resiliencia a catástrofes y corrección de errores.
- Integrabilidad
Para garantizar un nivel de confiabilidad, los servicios externos que se utilicen deben tener el mismo nivel de confiabilidad, por lo tanto es más difícil integrar servicios.
- Performance
Por latencia de comunicación entre componentes redundantes

- (b) Al incorporar soportabilidad se tiene como costo:

- Performance
El incorporar soportabilidad requiere de herramientas de diagnóstico (por ejemplo logs), por lo que se deben alocar recursos de hardware (CPU y Almacenamiento) para el funcionamiento de estas herramientas.

- (c) Al incorporar verificabilidad se tiene como costo:

- Mantenibilidad
Dado que se deben probar todos los cambios que se realicen en el sistema, se vuelve más difícil extender la funcionalidad de este. Para cada nueva funcionalidad, se debe añadir también un test de pruebas.
- Integrabilidad
Para integrar un sistema externo se debe añadir un conjunto de pruebas con el fin de verificar el correcto funcionamiento del sistema. Por lo tanto para un cambio en los servicios utilizados, se debe volver a desarrollar este conjunto de pruebas.
- Portabilidad
Si se quiere hacer portable el sistema, se deben añadir pruebas adicionales para cada uno de los sistemas objetivos (SO + arquitectura). Por lo tanto, la portabilidad tiene una complejidad adicional para un sistema verificable.

2. Analice las siguientes frases, justificando sus comentarios:

- (a) Ya sabemos que el patrón **MVC** está estructurado en tres capas:
 - La capa de presentación (V)
 - La capa de datos (M)
 - La capa de procesos (C).
- (b) En este esquema de separación de funciones, (V) no tiene acceso a los datos, lo que satisface el requerimiento de seguridad, ya que los datos están en una capa distinta a la capa en la que están los usuarios.
- (c) Sin embargo, ello afecta al rendimiento del sistema, puesto que los requerimientos de acceso a los datos deben pasar por las tres capas.
- (d) Sin embargo, ayuda a la mantenibilidad, dado que un error que se produzca va a quedar circunscrito a la capa en la que ocurre.

Respuesta:

- (a) La afirmación es incorrecta. **MVC** está estructurado en tres componentes donde:
 - La componente de datos (M) contiene el modelo de datos del sistema.
 - La componente de presentación (V) presenta los datos del modelo al usuario.
 - La componente de procesos (C) es la que se encarga de controlar los cambios sobre el modelo.

Estos componentes no corresponden a capas.

- (b) Si bien (V) no tiene acceso directo a los datos, esta se comunica con (C) para acceder a los datos de (M). Por lo tanto, el requerimiento de seguridad solo se cumple si la capa de controladores tiene mecanismos para garantizar la seguridad.
- (c) El rendimiento del sistema efectivamente se ve afectado por la separación de las tres capas, debido a la latencia de la comunicación entre cada capa.
- (d) Si bien es cierto que ayuda a la mantenibilidad debido a la separación lógica de las capas, y se tiene que los errores quedan circunscritos en la capa en la que ocurren. El aislamiento de los errores contribuye a la soportabilidad, no a la mantenibilidad.

3. Analice las siguientes frases, justificando sus comentarios.

- (a) El desarrollo de un sistema se inicia con el análisis de los requerimientos funcionales que debe satisfacer el sistema.
- (b) Una vez obtenidos estos, el arquitecto de software procede a elegir el modelo de arquitectura en el que se hará el diseño y desarrollo de este.
- (c) Finalmente, basado en el modelo arquitectónico escogido, procede a definir cuales son los requerimientos no funcionales que deberán ser considerados.

Respuesta:

- (a) Es correcto que el desarrollo de un sistema se inicia con el análisis de requerimientos funcionales a satisfacer, puesto que estos requerimientos son parte del contexto del sistema.
- (b) Luego de levantar los requerimientos funcionales, se deben analizar que atributos de calidad son requeridos (requerimientos no funcionales) y el ambiente operacional en el que se ejecutará el sistema.

- (c) Esta afirmación es incorrecta puesto que se elige la arquitectura en base al contexto (requerimientos funcionales y no funcionales), por lo que los requerimientos no funcionales siempre se definen antes de escoger la arquitectura.
-

4. El gerente general de la empresa en la que usted trabaja se considera a si mismo un experto en Arquitectura de Software e insiste en que el sistema de Atención de Clientes que le está solicitando implementar sea desarrollado utilizando el modelo de Capas. Este nuevo sistema debe contemplar los siguientes requerimientos funcionales:

- Registro de Clientes (CRUD)
- Mantención de Cuentas de los clientes
- Control de Pedidos Online
- Seguimiento de Despacho de Productos
- Facturación y Cobranza

Que propondría usted, seguir lo que se le impone o utilizar otro modelo arquitectónico y cual? Justifique su decisión.

Respuesta:

Debido a que varios de los requerimientos funcionales que se pide implementar deben interactuar con el CRUD de clientes, y los que no necesitan otros datos de la BDD, es difícil dividir el sistema en distintas capas. Por lo tanto, se propone utilizar otro patrón interactivo.

En este caso, se puede utilizar MVC para desarrollar el sistema debido a que es más sencillo separar las partes de cada requerimiento que corresponden a controladores y a la lógica del negocio (modelo) en comparación a la complejidad de separar todos los procesos en distintas capas.

5. Una pastelería de barrio quiere ampliar la cobertura de atención de clientes abriéndose al comercio electrónico. Para ello, requiere el desarrollo de un sistema que cubra los siguientes requerimientos funcionales:

- Recepción de pedidos online
- Disponibilidad de diversos medios de pago (webpay, mercadoPago, transferencias, etc.)
- Organización de los envíos a domicilio
- Gestión de los repartidores
- Estadísticas de venta, diaria, semanal, mensual, anual

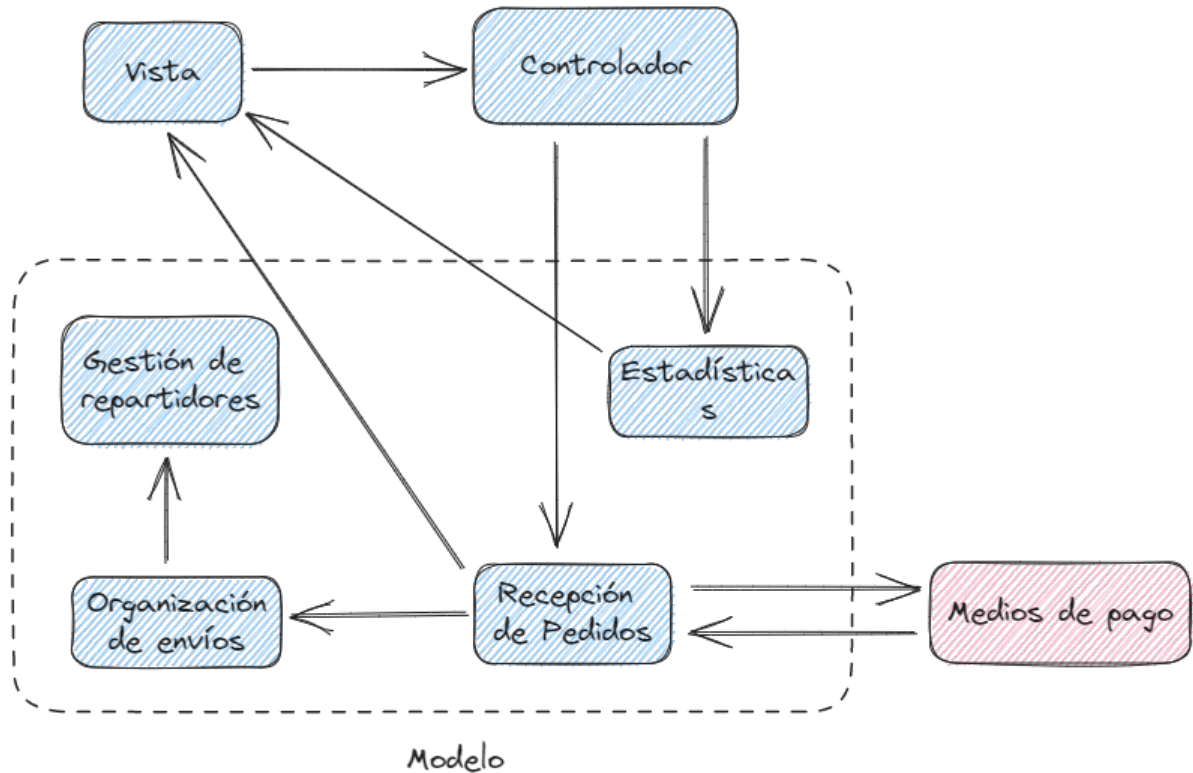
Se pide lo siguiente:

- (a) Justificar la elección del patrón de arquitectura a usar para el desarrollo del sistema
- (b) Diagrama global de la solución indicando como se va a satisfacer la funcionalidad requerida
- (c) Adicionalmente a Confiabilidad y Rendimiento, que deben estar considerados de todas maneras, describa dos requerimientos no funcionales a implementar en el sistema.

Respuesta:

- (a) Debido a que un comercio electrónico es un sistema altamente interactivo, se elige utilizar MVC como patrón de arquitectura. Los requerimientos funcionales,

(b) Diagrama de la solución



(c) Requerimientos funcionales a implementar:

- Integrabilidad: Debido a que se debe interactuar con distintos servicios de pago, es necesario que el sistema sea integrable.
- Mantenibilidad: Debido a que el negocio puede variar los productos que ofrece, es necesario que el sistema permita una fácil modificación y extensión de las funcionalidades.

6. Una Institución Financiera requiere un sistema para llevar el control del otorgamiento de préstamos a sus clientes. El proceso comienza con una solicitud en la que se especifica los datos del cliente, el monto solicitado y el plazo. El Comité de Créditos analiza la solicitud y puede aceptarla o rechazarla. En caso de rechazo, el cliente puede volver a solicitarlo entregando mas antecedentes que avalen su petición. En caso de aceptación, la solicitud pasa al departamento Legal en donde se redacta el contrato del préstamo. A continuación, el cliente debe firmar el contrato con su firma electrónica aceptando las condiciones que ahí se indican. Finalmente, el contrato es enviado al departamento de Finanzas para que proceda a transferir el monto del préstamo a la cuenta corriente del banco del cliente.

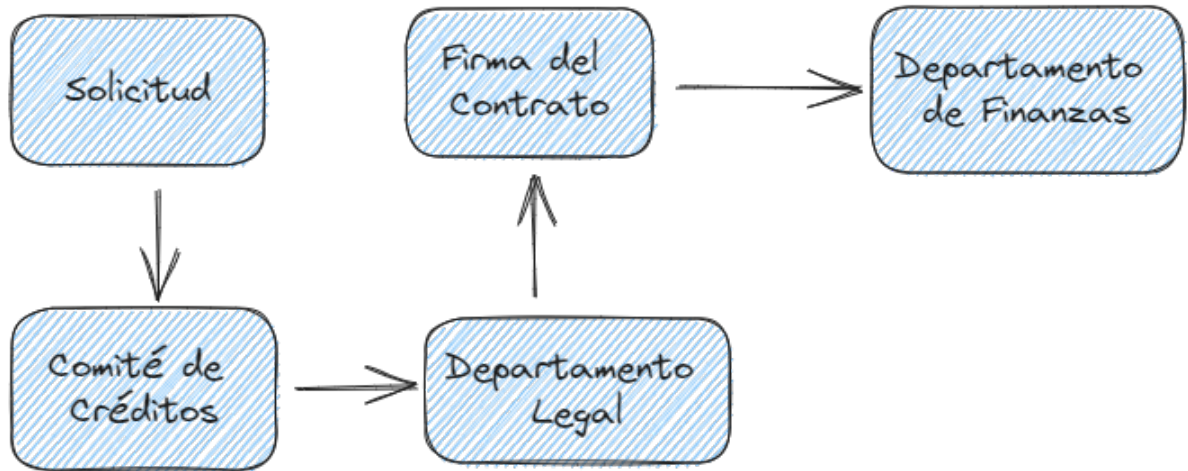
Se pide lo siguiente:

- Justificar la elección del patrón de arquitectura a usar para el desarrollo del sistema
- Diagrama global de la solución indicando como se va a satisfacer la funcionalidad requerida
- Adicionalmente a Seguridad y Confiabilidad, que deben estar considerados de todas maneras, describa que hay que hacer para implementar otros dos requerimientos no funcionales.

Respuesta:

- Debido a que el proceso es básicamente lineal, el patrón arquitectónico que más se ajusta a la problemática descrita es el patrón de Tubos y Filtros.

(b) Diagrama de la solución



(c) Requerimientos no funcionales:

- Soportabilidad: Se requiere de herramientas de diagnóstico para poder determinar en caso de un fallo, cual es el componente donde ocurre dicho fallo. Para implementar esto se puede almacenar logs para cada uno de los procesos del pipeline.
- Escalabilidad: Si la institución financiera es lo suficientemente grande, un solo pipeline de procesos no daría abasto a la cantidad de solicitudes. Por lo tanto se considera la escalabilidad. El sistema puede implementar escalabilidad horizontal, replicando el pipeline en su totalidad y utilizando un balanceador de carga entre pipelines.