



INTRODUCCIÓN A LA INGENIERÍA DE SOFTWARE

SEGUNDO CUATRIMESTRE 2022

SEGUNDA ENTREGA: INGENIERÍA DE SOFTWARE -
PROYECTOS DE SOFTWARE (REENTREGA)

INTEGRANTES DEL PROYECTO:

- ANA PAZ BAUSER (142037)
- MARCOS SIMÓN FERNÁNDEZ (142203)
- MARCOS GOIZUETA (142071)



1) Identifiquen a los participantes del proceso de desarrollo de software. ¿Por qué creen que el éxito de un proyecto depende mucho de los participantes que intervienen en el desarrollo de requerimientos?

Los participantes que forman parte del proceso de desarrollo de software del multicotizador son los siguientes:

- Como cliente, detectamos a la organización de productores de seguros, ya que estos solicitan el software a diseñar (generan la necesidad).
- Como usuarios logramos identificar a las personas que conforman dicha organización de empresas aseguradoras, ya que estos utilizarán el cotizador, para así proveer de su servicio a los clientes.
- Como desarrolladores, logramos identificar a la empresa de desarrollo llamada "Drubbit" la cual fue seleccionada para diseñar el multicotizador de seguros, en la misma podemos identificar un conjunto de participantes: en primer lugar los analistas de requerimientos, los cuales determinan las especificaciones solicitadas por los productores de seguros (clientes), un grupo de diseñadores, los cuales se encargan de describir qué hace el sistema, programadores, encargados de implementar en código lo descrito por los diseñadores, los testers, los cuales verifican que todo lo realizado por los programadores fue implementado de manera adecuada, y finalmente los capacitadores, quienes entrenan al cliente en el uso del multicotizador.

Creemos que el éxito de un proyecto depende mucho de los participantes que lo integran, ya que es totalmente necesario para el avance del proyecto, un diseñador de requerimientos que sea capaz de interpretar cada una de las funcionalidades y especificaciones dadas por el cliente para luego, implementar en conjunto con los desarrolladores el diseño del software requerido, y finalmente los testers, quienes verificarán que todo lo implementado, esté libre de errores. El participante de mayor importancia es el analista de requerimientos debido a que es el pilar fundamental de todo este proceso el cual comienza por el anteriormente nombrado. Si comete un error, éste provocará una falencia dentro del proyecto, pues los desarrolladores implementarán en código una idea provista por el analista que no coincidirá con la del cliente. Además, lo que hace, casi no es verificado por otros integrantes, a diferencia por ejemplo, de los programadores ya que los testers verifican en profundidad lo desarrollado por estos como fue anteriormente mencionado.

2) Sugieran un modelo de proceso para el ciclo de vida del producto en cuestión.



Desde nuestra perspectiva consideramos que el proceso de modelado evolutivo en espiral sería el proceso más conveniente para el ciclo de vida del multicotizador, ya que comienza con algunos primeros prototipos que son fundamentales en este caso debido a que los requerimientos no están bien definidos. Luego mediante cada iteración se logran ajustar los detalles que afectan la calidad de nuestro software, logrando así, a través de repetidas iteraciones, un proyecto bien estructurado, el cual estará en condiciones finalmente de ser el sistema implementado para su uso como multicotizador. Este modelo es adaptable durante todo el ciclo de vida del sistema, lo que consideramos muy valioso, ya que durante su desarrollo si se necesitan cambiar y/o agregar funciones del cotizador es de gran utilidad que su implementación no requiera grandes cambios ni procesos engorrosos, además productos de este estilo una vez finalizados aún así suelen requerir actualizarse con frecuencia.

3) Ingeniería de requerimientos

- i. Especifiquen al menos tres requerimientos funcionales y dos requerimientos no funcionales (o restricciones) que se hayan planteado.
- ii. Distingan los requerimientos identificados en el punto anterior en:
 - 1. Críticos, deben ser absolutamente satisfechos
 - 2. Deseables, pero no indispensables
 - 3. Posibles, pero que podrían eliminarse Justifiquen en cada caso por qué los distinguen de esa forma

i) Como requerimientos funcionales:

El usuario deberá poder modificar el estado de negociación: describe la interacción entre una función del multicotizador y el usuario.

Cotizar un seguro de autos con varias compañías al mismo tiempo y representar los resultados mediante una tabla comparativa.

El usuario solicitará una cotización y el software le proveerá mediante una función el resultado de la misma mediante la tabla.

Logueo de los productores/agentes comerciales: describe la interacción entre una función del multicotizador y el usuario.

Como requerimientos no funcionales:



El presupuesto debe ser aprobado por la organización: este requerimiento limita a la solución ya que si el presupuesto no es aprobado este no se puede llevar a cabo.

Se espera la entrega del producto a los productores en marzo de 2023: si el producto no es entregado cumpliendo con el tiempo solicitado, esta deja de ser una solución.

ii)

1_ Cómo requerimientos críticos identificamos:

- **El presupuesto debe ser aprobado por la organización:** ya que no hay ambigüedad o más de un punto de vista, si no es aprobado el multicotizador no será usado por la empresa.
- **Se espera la entrega del producto a los productores en marzo de 2023:** si no se cumple con la fecha pactada no aceptarán el producto, es indispensable.
- **Logueo de los productores/agentes comerciales:** si el sistema no permite a los usuarios loguearse no podrán hacer uso del multicotizador.

2_ Como requerimientos deseables identificamos:

- **El usuario deberá poder modificar el estado de negociación:** esto es beneficioso para todos los usuarios que estén interactuando en la negociación pero esta podría ser llevada a cabo si este requerimiento no se cumpliera, el producto no perdería su función.
- **Visualización de la tabla por parte del cliente mediante formato pdf, o vía mail:** la visualización de la tabla en más de un formato no es algo indispensable, solo con uno también se podría llevar a cabo la tarea deseada.

3_ No distinguimos ningún requerimiento como posible pero que podría eliminarse.

4) Expresión de requerimientos

i. Identifiquen los posibles estados de una cotización. ¿Qué método de representación del comportamiento de un sistema consideran más adecuado para ilustrar los estados y eventos que ocasionan los cambios de estado?



ii. Dependiendo de determinadas condiciones relacionadas al perfil del cliente y al vehículo se le podrá aplicar al cliente un descuento adicional a la cotización.

Por ejemplo, si tiene otro seguro contratado en la misma compañía se aplica un 20% de descuento, si es un cliente nuevo se aplicará un 10%, y si el auto tiene una valuación fiscal que supera los 5 millones se descuenta un 5% a la cotización. ¿Qué método de representación considera más adecuado para describir el comportamiento del sistema asociado a la aplicación de descuentos?

En cada inciso indiquen si se trata de una descripción dinámica o estática justificando la respuesta.

i) Consideramos que el método de representación del comportamiento más conveniente a la hora de representar los posibles estados de cotización es mediante el diagrama de transición, debido a que permite mostrar los estados en los que la cotización se encuentra. Primero hay un estado inicial donde aún no se ha realizado ninguna acción, luego se solicita que la cotización haya sido enviada al cliente (estado: lista de espera), luego si el cliente acepta alguna de las ofertas realizadas (estado: confirmada) el diagrama consulta si se envió una orden de emisión de póliza a la compañía seleccionada, y finalmente si la cotización fue rechazada o está vencida (estado: cancelada). Este contiene descripciones dinámicas, pues el diagrama acciona luego de las acciones denotadas anteriormente, además para que se cumplan determinados estados, primero deben acontecer estados anteriores, por lo tanto se pueden denominar como funciones.

ii) Teniendo en cuenta el comportamiento del sistema asociado a la aplicación de descuentos, creemos conveniente representarlos mediante el diagrama de tablas de decisión, ya que en la zona izquierda podemos representar las condiciones que afectarán el valor de la cotización (condiciones: si se tiene otro seguro contratado de la misma compañía 20% de descuento, si es un cliente nuevo se aplicará un 10% de descuento, si el valor tiene una valuación fiscal que supera los \$5.000.000, se descuenta un 5% de la cotización). En la zona de la derecha se definirán distintas reglas, las cuales se encargarán de mostrar todos los casos posibles de cotización luego de evaluar las distintas condiciones. Este diagrama contiene descripciones



dinámicas, ya que luego de aplicarse determinada condición, este afectará directamente al resultado.

5) Identifiquen al menos tres casos de pruebas a realizar. Para cada caso analice qué visibilidad deberá tener el testeador de los objetos a probar e indique el nombre con el que se conocen esas pruebas.

- Como última instancia del proyecto es necesarios realizar casos de prueba para poder verificar si nuestro producto tiene la calidad que buscamos, y que todas las funcionalidades respondan como se desea. No es posible detectar sólo con casos de prueba todos los errores del producto pero sí una gran parte de ellos.

Como casos de pruebas significativos elegimos:

- 1) El primer caso de prueba consiste en pedir que la tabla de cotizaciones sea enviada al mail, se espera que llegue al mail correcto sumado a que la tabla sea mostrada correctamente, y que además en el pdf pueda descargarse correctamente la tabla, la cual deberá cumplir el formato establecido con todas las cotizaciones solicitadas. Esto se prueba ya que es una de las principales funciones del multicotizador, por lo que su adecuada implementación es fundamental. El testeador deberá tener visibilidad de caja blanca ya que sabe exáctamente como es la estructura de la cotización, y cómo quiere que sea enviada.
- 2) El segundo caso de prueba sería el ingreso de varios usuarios a la vez al multicotizador y se espera que el mismo no colapse. Este caso se lleva a cabo ya que si el sistema no resiste el ingreso masivo de usuarios perderá la gran ventaja que otorga a la empresa, es decir la gran velocidad para realizar múltiples cotizaciones simultáneamente. Su correcta implementación es indispensable. El testeador deberá tener una visibilidad de caja negra ya que se “alimenta la caja negra” ingresando usuarios hasta que el sistema colapse o cumpla adecuadamente con la resistencia esperada.



-
- 3) El tercer caso de prueba probaría la funcionalidad que permite utilizar el multicotizador desde varios dispositivos (tablets, celulares, notebooks), se estima que funcione correctamente en todos, lo que abarca que el formato del interfaz gráfico del usuario se adapte adecuadamente, y las funciones puedan ser utilizadas sin errores. Este caso de prueba se lleva a cabo ya que si algo de lo anteriormente mencionado no se cumple la utilización del multicotizador se verá reducida a una menor cantidad de dispositivos. El testeador deberá tener visibilidad de caja blanca ya que tendrá que comprobar el correcto funcionamiento del multicotizador, verificando todos los caminos posibles dentro del mismo.

6) Análisis del proyecto:

i. Estimen el tipo y cantidad de recursos necesarios para llevar adelante el proyecto. Expliquen qué técnicas o herramientas usaron, o en qué se basaron, para llegar a los resultados presentados.

ii. Identifiquen al menos dos riesgos del proyecto indicando probabilidad estimada de ocurrencia e impacto.

iii. Especifiquen al menos tres requerimientos de calidad relevantes.

i) Los recursos necesarios para llevar adelante el proyecto son los siguientes:

- **Dinero:** presupuesto propuesto por la empresa de software. Para la realización del multicotizador, el gasto rondará los 5 millones de pesos, teniendo en cuenta el tiempo que se requerirá para llevar a cabo el proyecto, y el valor del costo promedio de desarrollo de un software personalizado que ronda entre los \$40000 y \$50000 dólares. Con esto debemos tener en cuenta el monto de 3,5 millones de pesos que la empresa “Drubbit” nos cobrará por el diseño del mismo, que se calcula con el tamaño del software, el costo por función y la cantidad de empleados (de la que hablaremos luego) que se ve involucrada en su diseño y desarrollo.



-
- **Recurso humano:** formado por los integrantes de la empresa de desarrollo de software llamada “Drubbit”, y el personal de la empresa que espera obtener el multicotizador para capacitarse en su uso y finalmente emplearlo en sus tareas. Deben ser muchos para reducir la duración del tiempo de desarrollo. Las habilidades que deben presentar los integrantes de la empresa de desarrollo son: tener la capacidad de desarrollar un software de calidad, testearlo de forma adecuada, y poder capacitar a los clientes del multicotizador. Contando desarrolladores de backend, desarrolladores de frontend, un QA (analista de calidad), un gestor de proyectos, y los anteriormente nombrados se estiman unas 25 personas. Se las necesita durante todo el proceso de desarrollo el cual finaliza en marzo de 2023.
 - **Equipamiento:** todos los dispositivos electrónicos en los que se hará uso del multicotizador, deben cumplir los requisitos necesarios para poder llevar a cabo esto (por ejemplo tener el sistema operativo adecuado). La cantidad debe ser suficiente para cubrir a todos los empleados que estén disponibles para usarlo en el día a día en un futuro. Se estima comenzar con unos 10 equipos nuevos. La cantidad irá aumentando con el tiempo pero comienza con esta pequeña cifra ya que la empresa antes hacía las cotizaciones de forma manual y deben adaptarse e ir creciendo de forma progresiva.

Se utilizó la técnica de Delphi de banda ancha en la que cada miembro del equipo de desarrollo estimó una duración pesimista, otra optimista, y la más probable, basándose en los conocimientos que tienen sobre su tarea específica. Luego de varias estimaciones se obtuvo un promedio.

ii) Riesgos del proyecto:

-El primer riesgo, podemos identificarlo como un riesgo técnico, donde existe la posibilidad de que el producto no haya sido testeado correctamente por los empleados y que alguna empresa que compita en el mercado o un grupo de personas con un fin desconocido pueda intervenir en los servidores del multicotizador, provocando que este no pueda utilizarse hasta que el problema se resuelva, también incluyendo el robo de información de los usuarios logueados. La probabilidad estimada es del 30% ya que será



previsto y se desarrollarán sistemas difíciles de penetrar. El impacto sería de tipo catastrófico (1).

- En segundo lugar podemos identificar un riesgo empresarial, donde el gobierno nos imponga un aumento salarial para los empleados, lo que nos significará un considerable aumento para el presupuesto ya enviado a los clientes, y lo que nos generará un decremento en las ganancias. La posibilidad estimada es de un 60%, debido a la inestabilidad económica de nuestro país y a los frecuentes cambios en las políticas impositivas. Estimamos que el impacto sería de tipo marginal (3).

iii) Requerimientos de calidad:

- "El multicotizador debe garantizar la visualización del mismo, mediante distintas plataformas". Consideramos que este requerimiento cumple con las condiciones especificadas en la PORTABILIDAD (pues se transfiere el software a otro ambiente de hardware).

- "Los productores y/o agentes comerciales, deberán loguearse en el sistema para poder multi-cotizar y administrar sus propias cotizaciones". Este requerimiento hace referencia a la INTEGRIDAD (pues se controla el acceso de personas no autorizadas) .

- "El multicotizador debe permitir cotizar un seguro de auto con varias compañías al mismo tiempo". Podemos asociar este requerimiento con la CONFIABILIDAD ,pues se espera que se cumpla con su función y con la precisión requerida.



Firma de cada integrante del trabajo:

Marcos Goizueta:_____

Ana Paz Bauser:_____

Marcos Simón Fernández:_____