

Metodología Eje 1

De la visión a los requisitos, de los requisitos al desarrollo

Lo primero que queremos aclarar al meternos de lleno en este eje temático de la materia es que no consideramos al mismo centrado en practicar la primera fase de tareas típicas de la elaboración del software bajo el conocido término "Relevamiento de información relacionada al problema". También necesitaremos girar un poco el centro de nuestros pensamientos para no enfocar, ahí en el medio del escenario, al usuario como guía de nuestros próximos pasos.

La captura de requisitos es un acto de descubrimiento, sin dudas connota el proceso difícil de determinar lo que hay que construir. Y en la bibliografía de referencia se vincula esa tarea con el artefacto caso de uso e incluso podremos llegar a leer que el caso de uso debe escribir acerca del qué hacer y no acerca del cómo hacerlo. Eso puede confundirnos acerca del nivel de profundidad que se requiere en este tipo de documentación y llevarnos a interpretar que puede carecer de nivel de detalle.

Si sumamos a estas ideas otras, que son parte también de la bibliografía, respecto a que la documentación que se plasme bajo el nombre "Captura de requisitos" *debe ser interpretada por nuestro cliente*; puede resultarnos un poco raro el enfoque que estamos trabajando en nuestro Eje 1.

Lo único que vamos a enfatizar es que los artefactos pueden servir a múltiples propósitos. Vamos a recomendar consultar la bibliografía para hacer crecer nuestras perspectivas, sin perder de vista el enfoque que damos al trabajo práctico en este Eje. La bibliografía nos inspiró hasta encontrar una metodología para aprender a hacer, y a sumar capas de diseño, agregando en la siguiente algo de lo que carecíamos en la capa anterior. Usamos un artefacto si creemos que sumarlo nos agregará al final, más dominio sobre el trabajo que estamos desarrollando, dentro del contexto que más propicie "el aprender a hacer".

Dicho esto, para tranquilidad de ustedes, trabajaremos en escenarios en donde el relevamiento está escrito y que lo que no esté escrito se puede suponer a criterio y sentido común. Ese sentido común se mide dentro del contexto dado por las verdades que ya se conocen del problema y de la solución y acorde a nuestro conocimiento de lo que puede automatizarse y lo que no. Viviremos este Eje concentrándonos en producir un diseño que pueda funcionar y cumplir las especificaciones que nos dieron por escrito. Necesitamos afianzar nuestros conocimientos técnicos para transitar esas etapas sin dificultades. Luego, en otra etapa, lidaremos con cómo relevar, con cómo escuchar al usuario. Haremos algunos simulacros, algunos juegos al respecto. Pero nuestro foco, ahora, está en dominar a fondo nuestros datos y nuestros procesos de modo que podamos lograr, lo antes posible una Visión del sistema que tenemos que construir. **No lo podremos hacer si no dominamos los detalles.** Más adelante, todo este proceso nos saldrá rápidamente, tal vez sin documentar tanto. Por ahora, la documentación que generaremos nos tiene que servir a

Metodología Eje 1

nosotros (equipo de desarrollo) para identificar el nivel de comprensión que estamos alcanzando acerca de lo que implica automatizar un requisito determinado.

Un detalle no menor: no pensemos que algún usuario va a poder dictarnos el diseño de lo que hay que programar. Eso dependerá del perfil de los clientes y usuarios que nuestros clientes nos pongan a disposición para colaborar con nosotros en este proceso. También dependerá de la experiencia y conocimiento que ellos tengan acerca del negocio. Por eso en esta etapa, aprendamos el cómo hacer lo que hay que hacer. Acotando lo que entendemos bajo el término cómo, ampliamos: ***qué datos se necesitan, los debo capturar por pantalla o los debo recuperar de lo almacenado, qué debo controlar, qué debo asegurar y calcular, de qué me tengo que hacer cargo en este algoritmo, qué debo grabar, lo puedo hacer o no lo puedo hacer, está completo mi circuito de información, todo lo que necesito leer en un caso de uso es ingresado y almacenado previamente en otro caso de uso.***

Caso de uso y sus características propicias

Uno de los aspectos que nos motiva a usar Casos de uso, como punto de partida, es la facilidad que nos presenta a la hora de bosquejar qué funcionalidad esperamos que presente el sistema que estamos construyendo.

Por un lado el Modelo de Casos alcanza directamente el concepto de Vista tan anhelada en UML. En este caso es una vista que referencia el comportamiento del sistema aportándonos, ya en una simple observación, dos condimentos imprescindibles para hacer viables la construcción realista de un software.

- El sistema debe disponer de su funcionalidad en “**opciones**” concretar que alguien o algo **inicia o decide ejecutar**
- La funcionalidad contenida en cada caso de uso debe ser **independiente** de la funcionalidad del resto de los casos de uso.
 - o (Los casos de uso NO cooperan “on line”, no dialogan. Más adelante veremos Estructuración de casos de usos, pero es un significado diferente. Justamente si forzamos la aparición de dicho significado en estadios tempranos, puede que se desvirtúe la esencia de la Vista que deseamos identificar)

Amplíemos un poquito, cada una de esas dos ideas

Metodología Eje 1

- ***Debe haber puntos concretos de disparo de la funcionalidad esperada, lo que nos revela de inmediato el concepto de "opciones del sistema"***
 - o Como ustedes ya lo vienen ensayando desde el año anterior esos "puntos" los concebimos primero imaginando qué tipo de usuario o rol (actor) está interesado en accionar esa funcionalidad.
 - o Y segundo racionalizando cuáles son todos los pasos que deben ir "juntos" bajo el mismo caso de uso para alcanzar el objetivo anhelado.
 - o Con esto dicho ya pasamos al segundo aspecto...
- ***Es clara la independencia que se busca entre la funcionalidad de un caso de uso y de otro.***
 - o Si bien es uno de los puntos que nos puede costar al principio lo vamos a entender fácilmente usando nuestros conocimientos de programación adquiridos en materias anteriores.
 - o El programa inicia porque alguien o algo "lo activó", esa parte ya está superada y entendida.
 - o Se producirá una serie de pasos internos algunos con cierta interacción con el iniciador otros no, y hasta que no podamos dejar grabado "algo consistente en el sistema" tendremos que seguir ejecutando pasos, buscando datos, pidiendo datos y así terminar el caso de uso de una manera coherente y elegante.
 - o En este comentario hemos dejado por sentado un principio fundamental de nuestra metodología: **partimos de un conjunto de datos persistentes que almacenó otro caso de uso e incluso otras instancias de un mismo caso de uso, lo que no sea "conocido" (o sea no haya sido "grabado antes") se lo tendremos que pedir al usuario y así estaremos en condiciones de modificar el estado inicial en el que encontramos nuestro sistema al iniciar el caso de uso.**
 - o *¿Todos los casos de uso cambian el estado del sistema?* No. Hay algunos que sólo toman los datos previamente almacenados, piden datos al usuario y producen resultados que no requieren ser almacenados pero que son de *valor para el usuario* y por eso los atendemos en un caso de uso.

La vista de Casos de uso o modelo de casos de usos, si bien apunta al comportamiento del sistema es conceptualmente estructural no dinámico, desde lo identificado en UML. Dado que no modela comportamiento del sistema dependiente del tiempo.

Lo que introduce nuestra metodología como necesidad es ir considerando aspectos dinámicos en el modelado desde estadíos tempranos. Esto es porque así nos obligaremos a poner foco en el análisis de los detalles, a no perder nunca de vista lo que se puede automatizar y en qué se requiere para poderlo concretar. En un tiempito estaremos en condiciones de pensar en todos estos aspectos mentalmente tal vez en una reunión con un cliente, donde tengamos que hacernos una primera idea de lo que nos están pidiendo. Pero en este eje

Metodología Eje 1

iremos desarrollando todos estos pasos con detalle y por escrito para depurar los conceptos y ordenar nuestro razonamiento.

Es decir comenzaremos a pensar la funcionalidad del sistema, con clara conciencia de que cada caso de uso materializará un cambio en el estado inicial en el que encontró dentro del sistema al iniciar. Al modelar "**midiendo los cambios producidos por los casos de uso**" introducimos un *primer aspecto de dinamismo*.

Por otro lado como hemos identificado que los casos de uso son independientes entre sí hemos tomado conciencia de que cada uno describe "un momento" en la vida del sistema. Este momento deber ser centrado en los intereses que busca el actor pero siendo concretos y profesionales debe estar identificando lo que es factible dejar almacenado en las entidades persistentes del sistema para que cuando otro caso de uso se ejecute "en otro momento posterior" pueda encontrar todo preparado para seguir adelante desde un punto cierto y previamente estipulado. Como hablamos de **casos de uso como eventos o momentos ordenados** que suceden en la vida de un sistema hemos indefectiblemente introducido otro aspecto dinámico. De todas las vistas, aspectos y objetivos entre los que puede ser útil definir la palabra sistema, ahora lo tratamos como un conjunto de interacciones entre actores y una o más instancias de los casos de uso definidos en el interior de los límites establecidos, que se producen en el tiempo.

Diagrama de Clases de entidad y Estado del Sistema

Para seguir reflexionando, es bueno que repasemos cómo hemos definido Estado del Sistema hasta el momento. Hemos expresado que el estado del sistema se refleja en las **entidades persistentes del sistema**. Hemos establecido que las mismas **cambian merced a lo que sucede dentro de los casos de uso**. Hemos declarado que nos interesa medir **cuál es el cambio que se produce**.

Bien, como ligado el concepto de Estado del Sistema a la propia definición de caso de uso necesitaremos, entonces, pensar esos Casos de uso a medida que pensamos en un Diagrama de Clases, sólo con las clases de estereotipo de análisis Entidad. Así podremos obtener una unidad de medida concreta para identificar los cambios de manera racional y fehaciente.

Expresaremos siempre el **estado del sistema en términos de Instancias de las Entidades que hemos modelado en nuestro Diagrama de clases de entidad**.

Recordemos que la pregunta a que deben responder siempre quienes estudian sistemas en cualquier área de la ingeniería es cómo identificar cuál es el conjunto de propiedades que me permiten describir al sistema. Sólo cuando se puedan definir valores específicos a todas esas propiedades podemos decir que el sistema se encuentra en un determinado estado. Por eso la relevancia de

Metodología Eje 1

tomar una posición de diseño al respecto de lo que consideramos estado del sistema. Esto nos permitirá observar diferentes momentos de equilibrio en el sistema con la posibilidad de sacar poderosas conclusiones.

Más dinamismo

Llegará un punto en donde necesitaremos validar las decisiones que vamos tomando en artefactos que sumen claridad, profundidad. En la mayoría de los casos esa profundidad o nivel de detalle, como ya lo expresamos anteriormente, será plasmada ya en artefactos por naturaleza dinámicos.

- Descripción del Flujo de Sucesos
 - Con su correspondiente Interfaz de Usuario
 - Notarán que usamos un lenguaje específico para documentar las secciones de Actor – Sistema con primitivas o acciones previamente codificadas para resolver la lógica en cada paso
 - Es importante estructurar la lógica siempre primero identificando los grandes pasos a resolver y con posterioridad en niveles de profundidad más detallados especificar la lógica de cada paso
- Diagrama de Transición de estados
 - Utilizamos diagramas de transición de estado alrededor de las entidades prioritarias que soportan en su definición el circuito de información en aspectos relevantes del problema.
 - Estas entidades se caracterizan por ser Estado Final de casos de uso prioritarios y por tener un atributo estado en su estructura
 - Lo interesante del enfoque empleado es que identificamos cuáles son los casos de uso que hacen posible el tránsito entre un estado y otro, con lo cual el artefacto nos permite revisar rápidamente si hemos construido adecuadamente el Estado inicial y Final de los casos de uso relacionados.

Conclusión

Confiamos en que, con estos elementos, estarán pudiendo empoderarse de su solución.

¡Que disfruten el proceso y el resultado!

Metodología Eje 1

Objetivos de la ejercitación

Lo que buscamos con la ejercitación es atravesar el proceso metodológico propuesto en una experiencia concreta. Si leen la guía de trabajo podrán identificar

Que el estudiante, junto con su grupo de trabajo y con la guía directa del docente pueda:

- Aprender en un ejemplo concreto el proceso de construcción del modelo de requisitos del sistema, encontrando actores y casos de uso.
- Entender la importancia que tiene lograr una especificación de requerimientos detallada, a través de la documentación adecuada de los casos de uso y ejercitarla en casos concretos.
- Integrar artefactos estudiados en materias anteriores, como apoyo a la actividad de especificación detallada del modelo de casos de uso (modelo del dominio, diseño de interfaz de usuario, descripción de flujo de sucesos a nivel de casos de uso, diagrama de transición de estados) y aplicarlos en casos concretos.
- Comprender el concepto de estado del sistema y aplicarlo en la construcción de los casos de uso. Logrando definir el circuito de información del sistema y cómo se modifica el estado mediante la ejecución ordenada de los casos de uso.
- Encontrar contextos adecuados de aplicación de los diferentes tipos de relaciones en un Diagrama de Clases (particularmente para el estereotipo de Entidad).
- Iniciarse en el desarrollo de criterios de comparación de diseños alternativos, distinguiendo aquellos que no resuelven el problema de aquellos que sí lo resuelven, encontrando ventajas, restricciones y desventajas de cada uno.

Primer meta

La primera meta concreta es poder confeccionar Diagrama de Clases y Modelo de Casos de Uso al mismo tiempo, por las razones que se plantearon en la guía metodológica. Para poder visualizar la conexión entre ambas introducimos un artefacto propio de la cátedra al que le llamamos Lista Descriptiva de los Casos de Uso, que debe confeccionarse usando la plantilla

Nombre del Caso de Uso

- Debe comenzar con un verbo en infinitivo, dando la idea de “acción”. Debe connotar preferentemente lo que el sistema se propone hacer y no tanto “lo que el usuario se propone hacer”. En casos concretos puede haber una diferencia relevante y podría quedarnos un nombre no apropiado si no cuidamos este detalle

Actor:

- El concepto lo conocen del año pasado. Sólo recuerden que, en algunas soluciones, pueden necesitar usar herencia de actores siempre en relación al concepto de roles bajo un sistema de seguridad (perfiles aditivos) y muchas veces para emprolijar las situaciones en las cuales “aparentemente” se mencionan varios iniciadores alternativos. También tengan presente que la funcionalidad puede ser iniciada por un actor humano o un actor sistema. Un ejemplo sería la forma en que se activan los servicios webs que tenemos disponibles para consumo por parte de sistemas externos. Recuerden también que si bien un caso de uso tiene que tener un único actor iniciador puede haber otros actores participando (sin ser iniciadores). Un ejemplo de esto, siguiendo con los servicios web, serían los diferentes sistemas con los que nos conectamos a través de WS.

Metodología Eje 1

Breve descripción:

- Tengamos en cuenta que el caso de uso “Nombre ya tiene” con los cual nunca debemos usar este apartado para repetir el nombre del caso de uso. Debemos aprovechar este apartado para esclarecer las particularidades del alcance que tendrá nuestro caso de uso. Uno de los aspectos que siempre es interesante puntualizar, si existen actores no iniciadores, aclarar en contexto de qué función aparece la colaboración. Imaginen que no se pide el desarrollo de la Descripción del Flujo de Sucesos a nivel de Casos de uso de cada caso de uso del Diagrama. Intenten sintetizar lo relevante respecto a qué se puede esperar del caso de uso (alcance) sin tener a disposición todos los pasos pertinentes para su ejecución. También tengan en cuenta que en este apartado ustedes le muestran a su jefe, compañeros, cliente el dominio que tienen en cuanto a la separación de un caso de uso del resto. Con lo cual tendrán ganas de identificar esas palabritas claves que hagan que quienes leemos el texto “no nos confundamos a este caso de uso con otro”, por ejemplo “o que no logremos capturar la profundidad y alcance de la propuesta concreta”

Prioridad (A/B/C):

- WOW este sí que es todo un tema. Si miramos el conjunto de casos de uso que quedó contenido en nuestro modelo definitivamente podríamos subclasificarlos de muchas formas diferentes. Debemos aprender a descubrir la relevancia que tienen algunos casos de uso frente a otros. ¿En base a qué? Principalmente en la complejidad y en la criticidad que representan dentro del circuito general de la información que hemos armado. Estaremos de acuerdo en que, identificar aquellos casos de uso que serán centrales dentro de la arquitectura base, será fundamental para tomar buenas decisiones de diseño en cuanto a lo que se desprenda de ese análisis. Esto quiere decir que, cuando llegue una situación crítica donde los recursos están peleados y tenemos que decidir entre un par de alternativas usaremos esta clasificación para decidir. La alternativa B favorece a un caso de uso de Prioridad A, pero no conviene tanto a uno de Prioridad B. Y la alternativa A es la que más conviene a un caso de prioridad C. Pero existe un conflicto de intereses y debemos elegir. Nuestro compromiso siempre deberá favorecer las prioridades establecidas. No siempre será fácil buscar un equilibrio general. Uno de los momentos donde más se emplea este análisis es al decidir sobre la navegabilidad y multiplicidad de las entidades que son Estado Final de los casos de uso.

Parámetros de entrada:

- ¿A través de qué atributos/elementos de información específicos se establece el diálogo entre actor y sistema? ¿Qué requiere el sistema conocer desde la IU para hacer su trabajo?
- No debiera ser necesario realizar el diseño de la Interfaz de usuario completa y detallada para poder identificar los parámetros de entrada. Es un aspecto que tiene que ver con la lógica y de nuestro dominio de los datos que hemos modelado. Pero, aunque no se haya solicitado, si te sientes más cómodo al principio, dibujando las pantallas completas es un recurso válido y respetable.

Precondiciones:

- ¿El caso de uso confía en (es decir “asume como válida”) alguna verdad que fue establecida por otro caso de uso que se ejecutó anteriormente?
 - Quiere decir que hay un cierto dato que está establecido a priori antes de iniciar el caso de uso, y en el interior del caso de uso no se cuestiona esa información, no se la valida y simplemente se la “acepta”

Metodología Eje 1

- Esta pregunta puede resultar contradictoria con el concepto que hemos enfatizado acerca de que los casos de usos deben ser independientes entre sí.
 - Justamente la Pre-condición, maneja una de las excepciones a ese concepto.
 - Por ahora vas a dejar este apartado en blanco, porque como dijimos anteriormente, no queremos desvirtuar algo esencial en el modelado de casos de uso que es entender que se trata de opciones desconectadas unas de otras. Eso es fundamental en esta etapa. Estudiaremos pre-condiciones en un rato.

Estado Inicial:

- Instancias creadas, relaciones establecidas entre entidades del modelo que establecen el punto de partida para poner en contexto la oportunidad de ejecución del caso de uso
- ¿Qué quedó reflejado en el sistema mediante la ejecución de un caso de su “anterior” que sea relevante para condicionar la ejecución de éste?
- NOTA: Advertir que en el interior del sistema el caso de uso deberá garantizar que el estado inicial requerido se cumple para poder permitir al usuario avanzar en la ejecución
 - Es decir, el caso de uso “no confía”. En este caso “duda” y tiene la responsabilidad de ASEGURAR antes de ejecutar.

Estado Final:

- Instancias creadas, atributos seteados y relaciones establecidas entre entidades, producto de la ejecución de este caso de uso
- ¿Qué cambió en el estado de sistema, qué se persistió, de dónde partirán los casos de uso que continúan describiendo el circuito de información?

Segunda meta

La segunda meta es poder llegar a fondo con las especificaciones y comprometernos con el qué y con el cómo hasta dominar los requisitos. Nada puede quedar librado al azar o la imaginación tiene que haber evidencia de que los hemos pensado y resuelto. El ideal es aplicar el método a todos los casos de uso de la solución. Por una cuestión de tiempos, lo hacemos con los que están mencionados en el práctico que son parte de la arquitectura base casos de uso de Prioridad A.

Descripción de flujo de sucesos

- Una vez estudiado el lenguaje a utilizar estudiaremos en los modelos cómo se emplea para resolver la lógica de los casos de uso

Interfaz de Usuario

- No encontrarán las interfaces de usuario resueltas en los modelos, deberán investigar algunos sitios webs para encontrar un estilo que les permita llegar a concebir objetos cohesivos y de fácil uso por parte del usuario y así lograr un dialogo amigable actor sistema. Buscamos que la IU sea integrada y que el tránsito hacia pantallas de nivel más profundo sea claro y con links de retornos establecidos. Es decir que se pueda navegar pero sin perder el hilo del caso de uso.

Diagramas de Transición de Estados

- Como dijimos anteriormente buscamos con este artefacto visualizar la conexión entre casos de uso y entidades. Es como representar gráficamente el estado inicial y final que ya documentamos para las entidades más relevantes del circuito.