Documento de Desarrollo

* Analice el proceso completo que siguió durante los tres proyectos (Proyecto 1, Proyecto 2 y Proyecto 3).

Síntesis: Se buscaba construir una aplicación para hacerle seguimiento a un proyecto realizado por una o por varias personas. Para conseguir esto existían tres principales componentes proyectos, actividades y participantes. Los proyectos tienen un nombre, una descripción, una lista de tipos, una lista de actividades, una lista de participantes, un dueño, fechas y un reporte. Las actividades tienen nombre, descripción, fechas de inicio y fin, un participante asignado, un tiempo de trabajo y un tipo asignado. Los participantes tienen un nombre y su correo. El reporte tiene que estar basado en actividades y participantes. Finalmente toca resaltar que toda la información debería ser persistente.

Proyecto 1, Primeros Pasos:

Fase 1: Análisis con diagrama de clases, requerimientos funcionales y restricciones

En esta fase tocaba hacer un diagrama inicial de clases muy básico, listar los requerimientos funcionales y las restricciones. Esta fase fue algo simple, debido a que apenas estábamos comenzando el proceso, no hubo alguna tarea particularmente difícil. Sin embargo, al principio dudamos como manejar un participante que es dueño vs uno corriente, en su momento decidimos que dueño y participante eran dos clases distintas y que cada función debería tener su variación para dueño o participante, cosa que no era acertada y luego cambio. Aparte de este punto, todo salió bien, se identificaron efectivamente los requerimientos funcionales con ayuda de las historias de usuario y las restricciones. Esta fase fue algo nuevo, ya que no acostumbrábamos a hacer un análisis antes de realizar los proyectos, pero sentimos que fue de utilidad para entender el problema.

Fase 2: Diseño detallado del proyecto

En esta fase tocaba hacer un diseño detallado del proyecto, esto incluía elementos como un diagrama de clases, un diagrama de clases de alto nivel, diagramas de secuencia y un documento de diseño con especificaciones de las clases, el diseño y las decisiones tomadas. Esta fase debido a que era iterativa, en un inicio presento varios problemas como que no sabíamos que estereotipos darles a las clases y 80% eran information holders o que nos faltaban algunas responsabilidades que creíamos implícitas, pero no lo eran. Además, presentamos un punto de inflexión al decidir cuáles serían las colaboraciones ya que estas decidirían el estilo de control de la aplicación, el nivel de acoplamiento y de cohesión. Al final nos decantamos por un estilo centralizado con Proyecto como centro y las relaciones derivando desde ahí, aunque tocaba tener cuidado con muchas relaciones y que tan adaptable sería el diseño a futuro. Con respecto a la anterior fase se añadieron dos cosas, primero decidimos unir participante y dueño en una sola clase, con un atributo de está definiendo si es un dueño o no y segundo se añadió una clase reporte para facilitar la creación de este (cosa que removida más adelante). Finalmente cabe resaltar que esta fase fue un poco extraña ya que solíamos hacer primero el código y luego ver como resulto su diseño no al revés, cosa que también nos daba miedo ya que inconscientemente nos podíamos estar cerrando a nuevos cambios si seguíamos el diseño estrictamente.

Fase 3: Implementación y Actualización

Esta fase que principalmente era hacer el código del proyecto, fue la que más problemas nos produjo hasta el momento, esto probablemente debido al desconocimiento del lenguaje y que este era el primer gran proyecto que realizábamos sobre java. Estos problemas iniciales con el desconocimiento del lenguaje se reflejaron al inicio con el manejo de los objetos y las clases principalmente, de estas últimas toca resaltar que no habíamos previsto el uso de clases auxiliares (como cronometro o file manager) que nos simplificaron y habilitaron ciertos procesos. Pero una vez superada esta primera barrera de desconocimiento el proceso fue más estable y preciso. Un primer obstáculo, fue decidir como almacenar las actividades, dudamos entre mapas y listas, al final decidimos ir con el primero y que el nombre fuera la llave, pero que adentro tuviera una lista de las actividades con ese nombre para poder solucionar ese dilema de actividades homónimas. En cuanto a requerimientos puntuales presentamos problemas con la asignación de actividades con el mismo nombre como mencionamos arriba y con cómo lograr hacer la persistencia. Para poder lograr esto último tuvimos que utilizar la clase auxiliar de file manager, aprender a escribir archivos .txt y luego leerlos y cargarlos en las clases, algo demandante para el nivel de conocimiento del lenguaje. Aparte de esto no presentamos mayores problemas y se logró hacer todos los requerimientos como se esperaba. En cuanto al diseño, aprendimos la importancia de tener un buen diseño (cosa que afortunadamente logramos) porque sirve mucho para tener una guía y un referente de lo que toca hacer, pero además también aprendimos que no hay que tener miedo a los cambios y que esto es un proceso que cada vez se va a ir perfeccionando más y más. Además, entendimos la importancia de tener un diagrama de clases, de alto nivel y de secuencia tanto para guiarnos como para que los externos pudieran entender nuestra solución.

Proyecto 2, Despegándose de la consola:

Síntesis: El objetivo principal de esta parte del proyecto, era utilizar las interfaces gráficas y despegarse de la consola. Esto servía tanto para aprender a manejar nuevos elementos del lenguaje como las ventanas, botones, diálogos, entre otros y para que la aplicación, tomará forma de aplicación y fuera mucho más amigable con el usuario y placentera de usar. Sentimos que esto es una parte muy fundamental y que contribuyo mucho a nuestro aprendizaje, además de que también ayuda a ver las relaciones en un programa de manera distinta.

Fase 1: Diseño de la Interfaz Gráfica

Esta fase consistía en hacer un diagrama o diseño, que reflejara como queríamos que se viera nuestra aplicación con una interfaz gráfica. En esta fase el mayor obstáculo fue encontrar un diseño de interfaz gráfica en el que todos estuviéramos de acuerdo, pero sobre todo que se acomodara a nuestro código de la mejor manera. Es decir que el proceso que teníamos en consola pasara de manera suave a un diseño con interfaz gráfica, para esto, en vez de hacer una traducción literal decidimos un enfoque por capaz, desde la más general a la más específica. Entonces lo primero que aparecía en la interfaz era crear o seleccionar un proyecto y un login para que el participante quede registrado, más adelante aparecía un menú mas general con la posibilidad de añadir participantes, añadir actividades, modificar actividades y generar un reporte. Adentro de cada una de estas opciones aparece la ventana correspondiente para realizar la acción. Aparte de esta duda inicial del enfoque para el diseño, no presentamos mayores problemas y realizamos nuestro diagrama.

Fase 2: Implementación de la Interfaz Grafica

Esta fase consistía en implementar el diseño de la interfaz gráfica que hicimos en la fase anterior. En esta parte del proyecto fue donde se presentaron los problemas, debido a que nos tocó enfrentarnos al uso de un framework nuevo como era SWING y adaptar el diseño nos a este. Un primer obstáculo, fue hacer que fuera estéticamente placentera, esto lo logramos sobre todo con el uso de una paleta de colores e imágenes de fondo, además de una ubicación adecuada de los botones y widgets. Un segundo obstáculo fue comprender las relaciones entre cada widget para poder utilizarlos apropiadamente, como por ejemplo cuando usar combobox o jtext y sobre todo poder extraer adecuadamente la información de estos elementos. El mayor obstáculo que enfrentamos fue hacer la gráfica para el reporte, sin embargo, llegamos a una solución adecuadamente compleja al hacer un pie chart con ayuda de JPaint, algo que vimos en un taller en clase. Sin embargo, no todo fue obstáculos, ya que gracias a nuestra buena implementación en el Proyecto 1 Fase 3, fue relativamente fácil hacer un traslado de las funciones a la interfaz grafica y hacer un diagrama de clases con nuestro estilo de control centralizado también fue relativamente fácil.

Proyecto 3: Cambios en el sistema

Síntesis: En esta fase del proyecto, se quería ver que tan flexible y modificable era la aplicación que habíamos construido hasta ahora. Para esto la nueva tarea era añadir un nuevo elemento a la creación de los proyectos, añadir una WBS que dentro de esta traía un elemento llamado paquetes con nombre y descripción y otro elemento llamado tareas que tenia nombre, descripción, actividad, tipo y fechas. Además de esto también tocaba generar nuevos reportes basándose en estos elementos añadidos.

Fase 1: Rediseño

En esta fase teníamos que adaptar nuestros diseños de clases, de alto nivel y de interfaz a los nuevos elementos y requerimientos introducidos en esta tercera parte. En este caso se nos presentó un obstáculo al pensar como manejar las tareas y paquetes, ya que los paquetes podrían tener otros paquetes por dentro, pensamos en muchas posibilidades, pero al final decidimos que ambos tuvieran una clase abstracta la cual extendían, de esta forma podían tener una lista de la clase abstracta y tanto paquetes como tareas serian incluidas. Un pequeño dilema fue como conectar los nuevos elementos las anteriores clases, pero siguiendo el estilo centralizado solo los conectamos a Proyecto y desde ahí se vinculaba a las otras clases. Aparte de esto no presentamos más problema que evitar muchas relaciones y dependencias (reducir el acoplamiento) y hacer que la interfaz grafica fuera eficiente y estética.

Fase 2: Implementación de los cambios

Esta fase consistía en implementar los cambios mencionados anteriormente, de acuerdo con los diseños realizados en la fase anterior. En este caso una cosa que se nos dificulto bastante fue la persistencia de la WBS, debido a que aparte de realizar la persistencia de este objeto también tocaba modificar las clases actuales de Proyecto y Actividad por lo cual había un cambio grande. Además de esto todo recaía sobre el trabajo manual de crear las nuevas interfaces gráficas, las nuevas clases, las relaciones actualizadas, los nuevos métodos, acoplar todo y sobre todo los nuevos reportes, que a pesar de no ser muy difícil si era un trabajo pesado y cansador. Aparte de esto no hubo gran dilema y gracias a las buenas implementaciones anteriores no hubo que hacer casi modificaciones sobre esta.