

# Diseño de Software

Daniel Méndez - Felipe Rios - Carlos Martínez - Luis Cáceres

**Resumen**—En este informe se indican los detalles necesarios para poder avanzar en el desarrollo del proyecto, mostrando con diagramas y especificaciones escritas, el diseño que consideramos adecuado para resolver el problema planteado. Para este cometido se toma en cuenta el levantado de requerimientos que hicieron nuestros compañeros de trabajo cuando conversaron con el cliente.

## I. INTRODUCCIÓN

Al momento de comenzar el proceso de diseño hemos considerado los requerimientos del usuario, hemos caracterizado las principales entidades participantes en el software, el usuario (persona ejecutante del programa), los trabajadores y la compra de café. Para hacer más comprensivo aún el trabajo a realizar, se implementarán diagramas de flujo y casos de uso, con los cuales estandarizaremos el diseño a implementar más adelante, logrando comunicar de forma clara como se empleará el programa y los módulos necesarios a desarrollar. Ambos diagramas son excluyentes entre sí y tienen distintos enfoques, pero ambos buscan mostrar y definir el software que hemos diseñado.

## II. MARCO TEORICO

Para la realización del diseño, se tomó la decisión de usar diagramas de flujo. Un diagrama de flujo es una representación gráfica de un proceso, cada paso del proceso es representado por un símbolo diferente que contiene una breve descripción de la etapa. Los símbolos gráficos del proceso están unidos entre sí mediante flechas que indican la dirección del flujo. Estos símbolos están estandarizados según la norma ISO 5807, que establece el significado de cada figura.

Además, se utilizaron diagramas de casos de uso, los cuales constan de actores y relaciones a las diferentes opciones que tiene un usuario dentro del software, usando dibujos que describen brevemente el uso del algoritmo. Para esto se usó lenguaje unificado modelado, o UML, por sus siglas en inglés, *Unified Modeling Language*, que es el lenguaje gráfico más conocido y utilizado en la actualidad para visualizar, especificar, construir y documentar un sistema.

## III. ANÁLISIS Y DISEÑO DE SOLUCIÓN

### A. Requerimientos faltantes

A partir del informe entregado, nos dispusimos a realizar el diseño, pero topamos al llegar a los requerimientos no funcionales, los cuales nos estaban especificados, por lo que basados en experiencias anteriores en el desarrollo de proyectos como este, definimos unos nuevos. Éstos son:

- La información debe mostrarse en la terminal de cada usuario.
- Cada usuario debe tener acceso a indicar quienes faltan.



Figura 1. Diagrama de casos de uso.

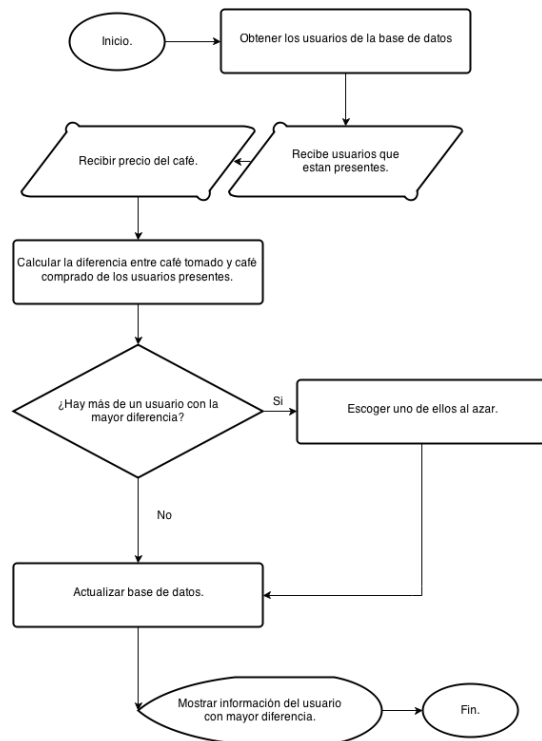
- Cada usuario debe tener acceso a ingresar el valor del café.
- Cada usuario puede agregar o eliminar a otros del sistema.

### B. Diagrama de casos de uso

El usuario tiene como objetivo principal, realizar la consulta de quien es el compañero de trabajo seleccionado para comprar café, este es el primer caso de uso, y se relaciona directamente con registrar una ausencia, dado que es necesario saber quienes están disponibles para realizar la compra. El programa responde mostrando el usuario que debe ir a comprar café, utilizando un algoritmo que se explica más adelante. También el usuario puede tener acceso al registro de gasto de todos los trabajadores. Si bien este caso no es funcional, permite supervisar la plataforma y asegurar su correcto funcionamiento. Por último, los usuarios pueden eliminar o agregar trabajadores del sistema e ingresar el precio del café que están comprando, tal como aparece en la figura 1.

### C. Diagrama de flujo

Tal como se especificaba en los requerimientos recibidos, y como podemos ver en la figura 2, lo primero que hacemos es obtener los datos guardados en la base de datos y compararlos con las personas que están presentes, que son ingresadas por el usuario junto con el precio del café. Luego de esto, debemos calcular la diferencia de precio de los cafés tomados y los cafés comprados para cada uno de los usuarios y buscar el mayor de estos valores. En un principio este será nuestro elegido para ir a comprar y en caso de que este número se repita,



**Figura 2.** Diagrama de flujo algoritmo selector.

se escoge uno de ellos al azar, siempre con las personas que están presentes. Para finalizar, se actualiza la base de datos y se muestra la elección en pantalla.

#### D. Discusión

El principal punto en el que nos vimos estancados, fue en la definición de los requerimientos no funcionales, dado que en el documento recibido no estaban claros. La definición de nuevos requerimientos nos trajo otro problema, la utilización de diferentes roles, como un administrador y usuarios. El problema al asignarle tareas únicas al administrador, era que existe la posibilidad de que el usuario asignado para este cargo no asista, dejando a sus colegas sin poder escoger a un comprador. Finalmente, y como se vio reflejado en el informe, solo definimos un tipo de rol. Otro de los puntos importantes fue la abstracción que debimos tomar a la hora de realizar los diferentes diagramas y evitar encasillarnos en un lenguaje específico.

## IV. CONCLUSIÓN