

**UNIVERSIDAD SIMÓN BOLÍVAR**

**Decanato de Estudios Profesionales**

**Departamento de Computación y Tecnología de la Información**

**CI-3715 Ingeniería del Software I**

**Septiembre-Diciembre 2017**

Informe Técnico

Tarea 3:

Programación por pares, desarrollo dirigido por pruebas (TDD).

Profesor: Alfonso Reinoza

Integrantes:

Rafael Cisneros 13-11156

Miguel Canedo 13-10214

Sartenejas, 19 de Octubre de 2017

Índice general

1. Introducción 3
2. Trabajo en equipo 4

2.1. Horas de trabajo 4

2.2. Programación por pares 4

2.3. Experiencia con las herramientas de apoyo 5

1. Análisis del dominio 6
2. Conclusiones 7
3. Apéndice: Django 8

2

Capítulo 1

Introducción.

Este documento tiene la finalidad de informar sobre el desarrollo de la tarea 3 por parte de nuestro grupo de trabajo.

El objetivo principal de esta asignación es ejercitarse en el uso de herramientas como Eclipse, PyUnit y Git, así como poner en práctica el conocimiento teórico sobre programación por pares y desarrollo dirigido por casos de prueba.

Para ello se debe realizar una estructura de datos que represente una billetera electrónica, con sus distintos atributos como lo son nombre, apellido y CI del dueño de la billetera, los créditos, débitos y el saldo de esta. Además se deben implementar estructuras de datos para representar los débitos y los créditos.

En el capítulo 2 describimos como fue nuestro trabajo en equipo, cuantas horas nos tomó realizar la asignación, como dividimos el trabajo y como fue nuestra experiencia en el manejo de las herramientas. En el capítulo 3 realizamos el análisis de los posibles datos del problema, y analizamos algunos casos bordes. Finalmente el capítulo 4 presenta las conclusiones del trabajo.

3

Capítulo 2

Trabajo en equipo.

En este capítulo se describe un poco como fue nuestra experiencia de trabajo en equipo: como dividimos el trabajo, como fue nuestra experiencia con las herramientas de trabajo, cuanto tiempo nos dedicamos a realizar la asignación.

2.1. Horas de trabajo

Para la realización exitosa de este trabajo fueron necesarias aproximadamente un total de 4 horas de trabajo por parte de cada integrante, durante 2 días continuos para cada miembro del equipo.

La asignación se realizó considerablemente más rápido que la asignación anterior, ya que tenemos un mayor conocimiento de las herramientas con las que trabajamos.

2.2 Programación por pares

El comienzo del trabajo fue realizado por Rafael Cisneros, quien proporciono las primeras pruebas para las estructuras de datos a realizar. Luego Miguel Canedo proporcionó las bases de las clases utilizadas en el programa, y se realizó la primera prueba con PyUnit, dando un resultado negativo ya que aún no se añade ningún código a las clases. Luego Miguel procedió a escribir los procedimientos necesarios en las estructuras creadas, hasta que se aprobaron las primeras pruebas. A continuación Rafael completo los casos de pruebas iniciales.

4

La realización del informe fue empezada por Rafael Cisneros, realizando la portada, índice y los capítulos 1 y parte del 2 del informe, luego Miguel Canedo se encargó de culminar los capítulos restantes y ajustar los últimos detalles para la entrega de la asignación.

2.3. Experiencia con las herramientas de apoyo

La realización de esta asignación fue bastante rápida ya que teníamos una experiencia previa en el desarrollo de programas con las herramientas utilizadas, esto nos facilitó mucho la realización de la asignación

5

Capítulo 3

Análisis del dominio

El problema planteado depende de 4 datos: los montos de los créditos, los montos de los débitos, el pin de la billetera y el nombre del dueño.

Para el caso de los créditos los montos deben ser no negativos, y para los débitos, además de ser no negativos, deben ser mayores al saldo actual de la billetera, de lo contrario la transacción no se realiza.

Para el caso del PIN, este debe ser un número real no negativo, no es considerado el caso en que las billeteras deben tener un PIN único.

Para el caso de los nombres, son aceptados nombres que contengan cualquier carácter correspondiente al castellano.

Algunos casos frontera considerados son aquellos en los que el saldo final de la billetera es cero o el nombre del dueño de la billetera contiene caracteres especiales. Como casos esquina consideramos casos en los que se incluyen nombres con caracteres especiales y saldo final cero, o con el número de pin erróneo y nombre con caracteres especiales, o el pin erróneo con saldo final cero. Para los casos con malicia realizamos test donde se realizan varias recargas y varios consumo

6

Capítulo 4

Conclusiones

Para finalizar se puede decir que este trabajo tuvo como objetivo principal, la familiarización por parte de los miembros del equipo con las herramientas de trabajo PyUnit, Egit y Github; además de aprender a elaborar casos de prueba frontera, esquinas y con malicia para probar casos críticos del problema con el software desarrollado y como secundario ejercitar nuestras habilidades como programadores y por así decirlo retomar la carrera universitaria después de haber experimentado un largo periodo de tiempo lejos de esta

Para ello debimos resolver un problema de calcular el precio de un servicio prestado, según una tarifa dada para días de semana y fines de semana, y una cantidad de horas de trabajo calculada a partir de un intervalo de tiempo dado en fechas y horas. Para luego realizar una serie de pruebas las cuales seguidamente se subirían a Github.

Dicho problema se resolvió creando dos clases. La clase Tarifa que consistía en un objeto en el que se pudiera almacenar la información de dos tipos de tarifas. Y la clase Servicio con contenía tanto la función principal para calcular precio, como las funciones pertinente para calcular el tiempo y verificar las condiciones expuestas en el enunciado del problema.

Seguidamente estas mismas se sometieron a prueba mediante un Testsuit el cual creamos siguiendo los pasos para la creación de casos de pruebas de fronteras, esquinas y malicia. Con lo que nos permitió corregir varios errores.

Y finalmente todos estos archivos fueron subidos a Github usando la perspectiva Egit proporcionada en Eclipse

7

Es importante destacar que lo mas complicado en este trabajo se vio reflejado en el momento de trabajar con Egit mediante Eclipse puesto que al ser una herramienta nueva con tantas opciones se hace fácil perder o cometer errores mientras se trabaja en esta dificultando así el manejo del mismo

Como recomendaciones podríamos dejar en claro que se tiene que dar un nivel alto de prioridad al entendimiento de Egit mediante eclipse puesto que si bien es cierto que saber utilizar Github puede ser una tarea normal , también es cierto que interactuar con este mediante Eclipse puede ser complicado si no se tiene una clara visión de como funciona este.

8