

Universidad Simón Bolívar
Dpto. de Computación y Tecnología de la Información
CI-3715 Laboratorio de Ingeniería de Software I
Enero-Marzo 2017

Tarea # 3

Estudiantes:

Edwar Yepez

Edgar Silva

Carnet: 12-10855

Carnet: 11-10968

Profesores:

Betzaida Romero

Alfonso Reinoza

Caracas, 2 de febrero de 2017

Introducción

La técnica de programación por pares es aquella que requiere a dos programadores que trabajen de manera conjunta en un sitio de trabajo realizando un proyecto de manera combinada. Cada miembro tiene su papel bien definido y se turnan estos roles. Mientras que uno escribe el código y las pruebas, el otro se mantiene observando y buscando nuevas ideas y captando errores que puedan ocurrir a lo largo de la ejecución del programa.

Estos roles tienen un nombre definido, aquel que se encarga de escribir el código se le llama conductor, y a la persona que observa y dirige se le hace llamar navegador. La presente tarea tiene como objetivo que los estudiantes ejerzan este estilo de programación ya que es muy útil tanto en lo académico como en el ámbito laboral en el que en el futuro nos desenvolveremos.

Desarrollo y Pruebas

Para implementar la billetera electrónica fue necesario desarrollar la clase `BilleteraElectronica`, así como los métodos `saldo()`, `recargar()` y `consumir()`. Todas las recargas y consumos se guardan en un arreglo dinámico que contiene dos tipos de estructuras según el caso, de recargas y consumos, con atributos específicos de la misma.

El trabajo fue dividido equitativamente entre los miembros del equipo: Edwar fue el conductor en todo el desarrollo de la implementación de la Billetera Electrónica, Edgar su navegador. Por otro lado, durante el desarrollo de las pruebas unitarias el conductor fue Edgar y el navegador Edwar.

El tiempo de desarrollo total del proyecto fue de aproximadamente 8 horas.

Pruebas:

Las pruebas representaron parte primordial en el desarrollo de la billetera electrónica dado el enfoque TDD que debía practicarse. Así pues, se construyeron cinco archivos de prueba:

- `testSaldoTarea3.py`: En este archivo se realizaron pruebas para el método `obtenerSaldo()`, el cual devuelve el saldo actual de la billetera.
- `testConsumirTarea3.py`: En este archivo de pruebas unitarias se sometió a pruebas el método `consumir()`, el cual permite realizar operaciones de consumo siempre y cuando el saldo de la billetera lo permitiera.
- `testRecargarTarea3.py`: Este archivo se encarga de testear el método `recargar()`, el cual permite “depositar” dinero en la billetera electrónica.
- `testCreditoTarea3.py`: Las pruebas realizadas en este código tienen como finalidad probar la estructura `Recarga()`, para así confirmar que todos los atributos de la misma no contengan basura o elementos inesperados.
- `testDebitoTarea3.py`: Este archivo tiene una finalidad similar al anterior, y es probar la correctitud de la estructura `Debito()`, para así descartar cualquier comportamiento anómalo.

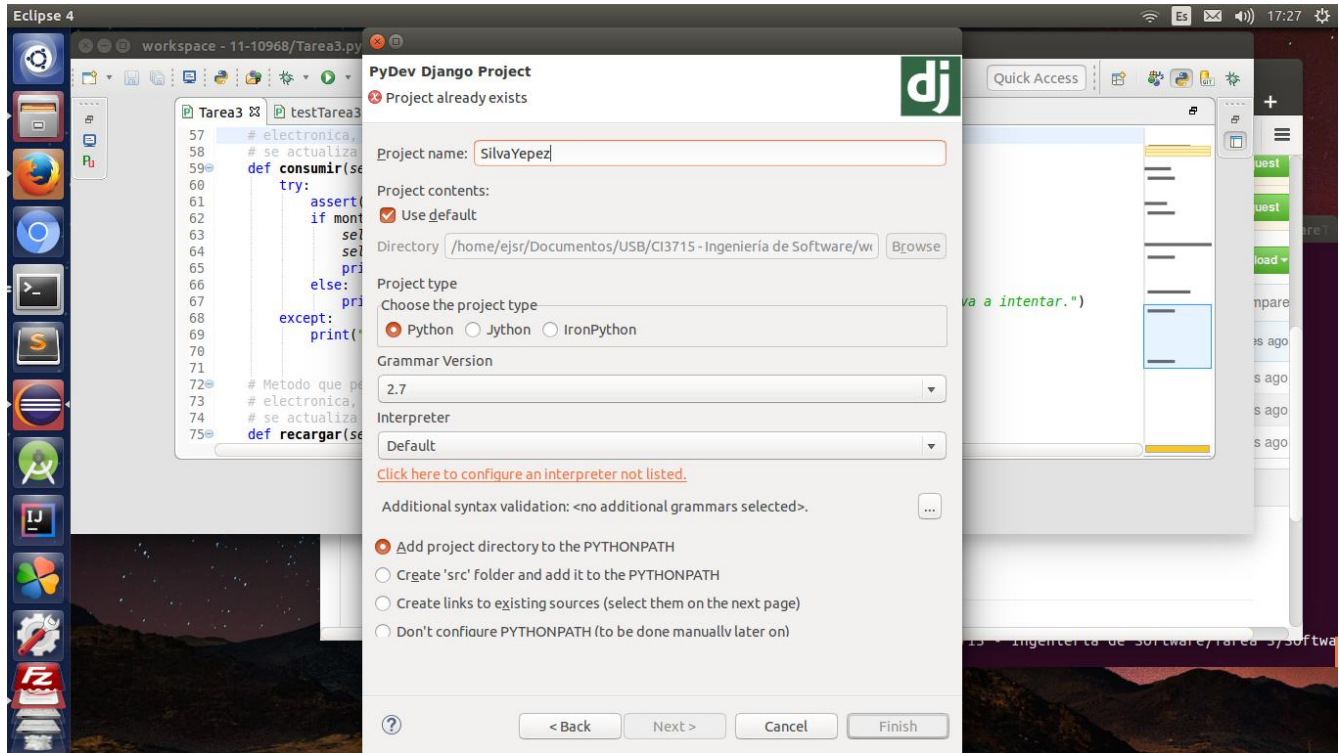
Herramientas:

En este proyecto nos apoyamos de las herramientas de control de versiones utilizando el repositorio de GitHub además del IDE Eclipse, a través de estas herramientas descubrimos las bondades de trabajar en equipo tanto de manera remota como presencial.

Utilizando estas herramientas de programación se nos facilita considerablemente el desarrollo de software ya que nos permite trabajar en un entorno que integra todas las herramientas que se utilizarán durante la realización del proyecto y además tener archivados todos los cambios importantes que se hagan, teniendo la posibilidad de volver a cierta versión del proyecto cuando sea necesario.

Apéndice-A

ScreenCapture



Conclusión

A traves de la realizacion del proyecto nos pudimos dar cuenta de los beneficios que trae la programacion por pares y la utilizacion de las pruebas unitarias, ademas de las herramientas recomendadas en el curso para manejo de versiones y para desarrollo de proyectos.

Entre los beneficios otorgados se encuentran el mantenimiento del proyecto en un buen estado de orden y el facil alcance de las distintas herramientas. Otra de las ventajas es la distribucion equitativa del trabajo y la posibilidad de que cada quien se desenvuelva un tiempo en los distintos roles abarcados en el estilo de programacion en pares.