



Universidad Simón Bolívar

Dpto. Ingeniería de computación

Laboratorio de Ingeniería de Software I

Tarea 2

Profesor: Franco Nori González.

OBJETIVO GENERAL

1. Aprender a usar herramientas básicas de apoyo a la Ingeniería de Software.
2. Ejercitar la técnica de análisis de esquinas, fronteras, normales y malicia para elaborar y programar una suite de pruebas unitarias.
3. Practicar TDD.

ACTIVIDADES ESPECÍFICAS A REALIZAR

- 1) Programar utilizando las siguientes herramientas de apoyo del curso: un lenguaje de programación orientado a objetos (Python 3.x), un *IDE* (PyCharm/VSCode) y una herramienta de control de versiones (Git).
- 2) Aplicar la técnica de análisis de fronteras para elaborar una suite de casos de pruebas unitarias en PyUnit y usando Git para gestionar sus versiones.
- 3) Enriquecer los casos de prueba anteriores con algunos casos de prueba que ejerciten esquinas y otros casos “maliciosos”.
- 4) Trabajar la tarea en pareja.

ENUNCIADO

Desarrolle en Python 3.x y pruebe con PyUnit en su *IDE*, usando el enfoque *TDD* (dirigido por casos de prueba), una función que devuelva como resultado si una persona cumple con los requisitos legales para recibir una pensión de vejez del Instituto Venezolano de los Seguros Sociales (IVSS).

Sabemos que, a tal fin, en Venezuela, la principal referencia legal es el artículo 27 de la Ley del Seguro Social (Decreto N° 6266,31 de julio 2008), el cual establece:

La asegurada o el asegurado, después de haber cumplido 60 años de edad si es varón o 55 si es mujer, tiene derecho a una pensión de vejez siempre que tenga acreditadas un mínimo de 750 semanas cotizadas.

Además, la ley también establece en el artículo 28:

La asegurada o el asegurado que realice actividades en medios insalubres o capaces de producir una vejez prematura, tiene derecho a una pensión por vejez a una edad más temprana a la que se refiere el artículo anterior y en la forma en que lo determine el Reglamento.

Este reglamento establece en el *Artículo 162*:

La pensión de vejez podrá concederse antes del cumplimiento de las edades señaladas en el artículo 27 de la Ley del Seguro Social, cuando el asegurado realice actividades en medios insalubres o capaces de producir vejez prematura, circunstancias éstas que deberán ser comprobadas por el Instituto.

A los efectos establecidos en este artículo, mediante reglamentación especial dictada por el Instituto y aprobada por el Ministerio del Trabajo, la edad límite podrá ser rebajada hasta en un (1) año por cada cuatro (4) trabajados en las condiciones antes dichas. Tal reducción no podrá exceder de cinco (5) años.

Reforma Parcial del Reglamento General de la Ley del Seguro Social
Decreto N° 8.992 del 24 de abril 2012

Debe aprovechar las funciones de librería de Python para manejar fechas. La función debe dar la respuesta a la fecha de hoy, donde hoy es el día en que se ejecute.

Distribuya la labor de programación de la función y de los casos de prueba con su pareja y utilicen Git para guardar una versión nueva de su suite cada vez que alguno de los miembros culmine un caso de prueba nuevo. Recuerde ejecutar los casos previos para cerciorarse de que no se haya roto nada. Puede organizar las ramas de su repositorio de la manera que considere más conveniente, pero incluya al menos una rama master y una rama por programador, de modo que puedan practicar operaciones sencillas de commit, *merge*, *pull* y *push*. Una vez culminada la suite creada con la técnica de análisis de fronteras, agreguen algunos casos con esquinas y por malicia.

ESPECIFICACIONES DE LA ENTREGA

La entrega consistirá de los siguientes productos:

- 1) Acceso al repositorio GitHub/GitLab (o cualquier otro servicio Git) de su grupo de dos para que los profesores podamos clonar su repositorio y revisar su código fuente, compilarlo y ejecutar las suites correspondientes de prueba y evaluar el grafo generado en GitHub.
- 2) En el repositorio debe poderse ver con claridad que han aplicado TDD. Usen los comentarios a los commits.
- 3) Un informe que incluya el número de horas que trabajó cada miembro del equipo (recomiendo la herramienta Clockify para hacer tracking de horas trabajadas), cómo se

dividieron las labores, un breve resumen de sus experiencias con las herramientas de apoyo y el modelo de ramas que adoptaron para el control de versiones.

- 4) El informe debe incluir como un apéndice, la descripción de cada caso de prueba en el formato:
- Objetivo de la prueba.
 - Entradas.
 - Valores esperados.

Nota: Aquí debe verse claramente que siguieron TDD.

- 5) Los informes deben seguir los "Lineamientos sobre cómo escribir informes técnicos" de la profesora Soraya Abad. (https://www.u-cursos.cl/ingenieria/2010/1/MI3130/1/material_docente/bajar?id_material=274509)

Todos los productos se deben enviar vía email a mi correo 08-10801@usb.ve a más tardar el jueves a la 6 pm.

Bibliografía recomendada: Test Driven Development With Python, Harry Percival. O'Reilly.