|  |  |  |  |
| --- | --- | --- | --- |
| Sigla Asignatura | DWY4101 | Nombre de la Asignatura | DESARROLLO DE APLICACIONES WEB Y MOBILE |
| Nombre del Recurso Didáctico | 2.1.9 Pruebas unitarias en Mis Perris | | |
| Unidad de Aprendizaje N° 2 | Programando del lado del servidor | | |
| Unidades de Competencia | Desarrolla aplicaciones utilizando patrones de diseño y buenas prácticas de programación que soporten los requerimientos de la organización (3) | | |

# Objetivos:

Utilizar Django unittest y Pytest Coverage en la implementación de casos de prueba unitarias para las funcionalidades de lado del servidor desarrolladas anteriormente.

# Instrucciones:

Siguiendo con el plan de modernización de la Fundación Mis Perris, en esta actividad nos enfocaremos en asegurar la calidad del sistema mediante la construcción, ejecución y análisis de pruebas unitarias automatizadas.

La Fundación ha solicitado, que la cobertura de las pruebas llegue al 100% por lo que, en equipo, deberán construir casos de prueba con al menos 2 casos de borde para cada uno para todos los métodos realizados en las funcionalidades que hasta ahora tiene el Sitio.

Recuerda, que las funcionalidades que actualmente tiene el Sitio son: registro de usuarios y mantenedor de rescatados.

**Observaciones:**

* Utiliza Django unnittest para la construcción y ejecución de los casos de prueba
* Elaboren un informe HTML del resultado de las pruebas que pueda ser revisado fácilmente en el repositorio (Puede ser un archivo README)
* No olvides respaldar tu trabajo en tu repositorio remoto en un nuevo Branch

# Material Complementario:

|  |  |
| --- | --- |
| **Descripción** | **Enlace** |
| Documentación Oficial Pytest - Django | <https://pytest-django.readthedocs.io/en/latest/> |
| Recomendaciones para Pytest en Django, PyCon 2017 | <https://pyvideo.org/pycon-es-2017/pytest-recomendaciones-paquetes-basicos-para-testing-en-python-y-django.html> |
| Desarrollo Dirigido por Pruebas (TDD) en Django, en inglés. | <https://www.youtube.com/watch?v=41ek3VNx_6Q>  código usado en el video:  <https://github.com/matthewbortolin/django-testing> |
| Pytest Coverage, permite generar reportes sobre la cobertura de nuestras pruebas sobre el código. | <https://pypi.org/project/pytest-cov/>  <https://github.com/pytest-dev/pytest-cov> |
| Documentación y ejemplos del módulo estándar “unittest” | <https://docs.djangoproject.com/es/2.0/topics/testing/>  <https://www.youtube.com/watch?v=1Lfv5tUGsn8>  <https://www.obeythetestinggoat.com/book/chapter_02_unittest.html> |

# Forma de Trabajo:

En parejas

# Tiempo estimado:

4.5 horas