

Librería en Pypi

Para este proyecto vais a montar una librería con todo lo aprendido durante el bootcamp. Esta librería la subiréis al [principal repositorio de paquetes de Python, Pypi](#), para compartirla y que cualquiera pueda descargarse el paquete.

Objetivos

1. Repasar todo lo visto en el bootcamp.
2. Publicar una librería que pueda utilizar cualquiera, y además os de visibilidad.
3. Montar una herramienta que os sirva en el trabajo y aumente vuestra productividad.
4. Tener un proyecto en común con toda la clase que podáis seguir alimentando con lo aprendido cada uno en sus futuros trabajos.

Requisitos

1. **Documentación.** La librería tiene que estar correctamente documentada, con sus doctstrings y todas las descripciones necesarias para la página de PyPI, y el GitHub donde esté alojada.
2. **Control de versiones:** Hay que utilizar GitHub para subir el paquete a la plataforma y posteriormente indicarle a PyPI en qué repo se encuentra.
3. **Presentación:** Se realizará una presentación a los profes, una semana después. Para ello tendréis que hacer una demo y contar lo que habéis hecho.
4. **Trello:** Es recomendable utilizar una herramienta de gestión de proyectos como Trello.
5. **IDE:** Es obligatorio trabajar con un IDE como Pycharm o VSCode.
6. **Testing:** Cada participante tiene que ser desarrollador y tester. Desarrollará mínimo 3 funciones de la librería y **testará otras 3 de sus compañeros**. Para mejorar la calidad del testeo, el alumno no es el que realiza el testing de sus funcionalidades, sino sus compañeros. En la realización del testing el alumno dispondrá únicamente de la documentación generada para esa función y tendrá que ser capaz de usarla correctamente. Intenta abarcar todos los casos posibles en el testing.

Opcional: se recomienda realizar un testing profesional con librerías como “pytest” o con la función “asset” de Python.

Requisitos técnicos

Las funcionalidades desarrolladas en la librería deberían cubrir los siguientes temas:

7. **Programación con Python:** listas, bucles, clases...
8. **Lectura/escritura de fuentes de datos:** automatización, búsqueda de archivos, encodings, mandar mails...
9. **Limpieza de datos:** funciones de exploración y limpieza de datos en numpy/pandas: por ejemplo, reportes de análisis exploratorio, tratamiento de time series, de texto, automatización.
10. **Visualización.**

11. **Machine Learning:** automatización/ayuda de algunas tomas de decisiones, entrenamiento del modelo.
12. **Documentación & coding:** facilidad para encontrar código, sugerencias, códigos hechos...

Estos apartados son orientativos. Podéis añadir los que consideréis, mientras vayan a resultar de utilidad.

Además de estos requisitos técnicos, lo interesante es compartir aquellas funciones y códigos que hayáis tenido que desarrollar durante el bootcamp, y penséis que resultarían útiles para el resto de la clase/comunidad.

Desarrollo

El desarrollo del proyecto se compondrá de las siguientes fases:

1. Definición de equipos, roles y líderes de equipo
2. Definición de requisitos de la librería
3. Brainstorming de aportaciones de cada alumno. Para cada funcionalidad plantearos: ¿esto lo usaría yo?
4. Reparto de tareas y asignación de funciones y testing
5. Testing
6. Puesta en producción
7. Testeo en producción
8. Documentación
9. Presentación

Entrega y presentación

La entrega constará de dos enlaces, que se enviarán a los profesores.

1. Enlace al paquete de Pypi
2. Enlace a GitHub con el código fuente

Se realizará una presentación a la clase, con una demo, mostrando todas las funcionalidades que tiene la librería.

Equipos

No se asignarán equipos como tal, puesto que hasta que no empecéis a pensar en la librería, no sabréis quién puede aportar más en cada parte, por lo que los equipos los elegiréis vosotros.

Lo único con lo que contaréis es con los siguientes coordinadores:

1. Manager GitHub + Subida del paquete
2. Manager limpieza de datos
3. Manager visualización
4. Manager machine learning
5. Manager del proyecto (gestión y asignación de tareas)

Los alumnos con tales responsabilidades también desarrollarán código, por lo que deberían trabajar en las funcionalidades de la librería menos pesadas.

¡Suerte!

Bibliografía

Para montar el paquete en Pypi: <https://medium.com/@joel.barmettler/how-to-upload-your-python-package-to-pypi-65edc5fe9c56>

Testing en Python: <https://realpython.com/python-testing/>

Guia completa para documentar en Python: <https://realpython.com/documenting-python-code/>