

Documentación del Proyecto

# Tienda de Olivanders

Autores: José María Samos y Sebastià Adrover

Profesor: David Gelpi

Assignatura: Programació

<b>Metodologías y tecnologías utilizadas</b>	<b>3</b>
1.1 Tecnologías de desarrollo	3
1.2 Metodología de desarrollo	3
<b>Diagrama de componentes</b>	<b>3</b>
<b>Arquitectura del proyecto</b>	<b>4</b>
3.1 Objetos principales	4
3.2 Objetos excepción	4
3. 3 Ficheros de desarrollo web	5
<b>Resumen del desarrollo de la app</b>	<b>5</b>
4.1 Estructura de desarrollo	5
4.2 Diagrama de tiempo de desarrollo	5
4.3 Análisis de desarrollo	6

# Metodologías y tecnologías utilizadas

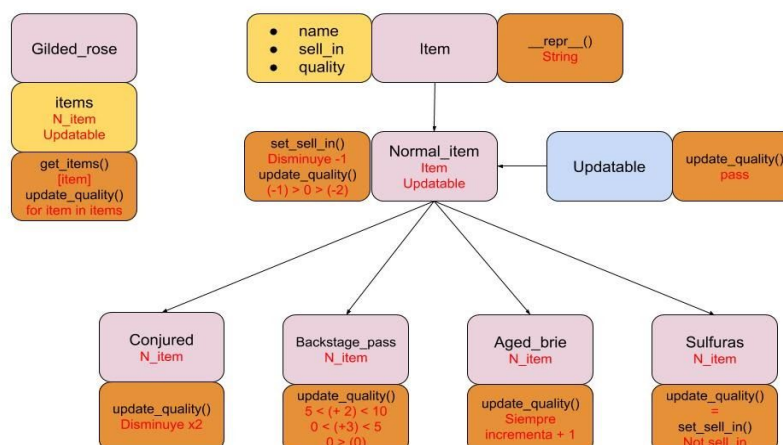
## 1.1 Tecnologías de desarrollo

- **Visual Studio Code**, es un editor de código fuente desarrollado por Microsoft para Windows , Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto,12 aunque la descarga oficial está bajo software propietario requiriendo tus datos de uso del programa legalmente
- **Flask**, es un framework minimalista escrito en Python que permite crear aplicaciones web rápidamente y con un mínimo número de líneas de código. Está basado en la especificación WSGI de Werkzeug y el motor de templates Jinja2 y tiene una licencia BSD.
- **Python**, es el lenguaje de programación utilizado para desarrollar el proyecto, se utiliza el módulo pytest para los casos test y comprobar que el programa funciona como se espera.

## 1.2 Metodología de desarrollo

- **Metodología cascada**, es un proceso secuencial, fácil de desarrollo en el que los pasos de desarrollo son vistos hacia abajo a través de las fases de análisis de las necesidades, el diseño, implantación, pruebas (validación), la integración, y mantenimiento.

## Diagrama de componentes



# Arquitectura del proyecto

La arquitectura del proyecto “Tienda de Olivanders” consta de nueve objetos, todos separados en distintos ficheros, sus casos test, un fichero de lógica y uno de desarrollo de aplicación web.

## 3.1 Objetos principales

- Item: es el objeto original y describe las propiedades básicas que tienen todos los objetos de la tienda; “name”, “sell\_in” (tiempo de vida) y “quality” (calidad).
- Updatable: es un objeto interfaz que obliga a “Normal\_item” que su “quality” deba actualizarse.
- Normal\_item: es el objeto heredero de la clase “Item” y “Updatable”. Este objeto define el comportamiento que deben tener las propiedades “sell\_in” y “quality” durante la existencia del producto por norma general; aunque hay excepciones. “Sell\_in” disminuye de uno en uno, a medida que “sell\_in” disminuye “quality” también. Cuando “sell\_in” es cero, “quality” disminuye el doble de rápido.

Luego por otra parte tenemos el objeto de la propia tienda:

- Gilded\_rose: su comportamiento consiste en añadir los items en una lista y asignar valores a “sell\_in” y “quality”.

## 3.2 Objetos excepción

Estos son los productos excepción de la “Tienda de Olivanders”, todos estos son herederos de la “Normal\_item”:

- Aged\_brie: su comportamiento sólo incrementa su calidad. Cuando su tiempo de vida es cero su calidad sigue aumentando.
- Sulfuras: este objeto se considera legendario y su calidad y caducidad no se altera.
- Backstage\_pass: objeto que a medida que su tiempo de vida va disminuyendo, su calidad va aumentando exponencialmente. Cuando su tiempo de vida es cero, su calidad pasa a ser cero también.
- Conjured: su comportamiento disminuye su calidad el doble de rápido que “Normal\_item”.

### 3. 3 Ficheros de desarrollo web

En cuanto al desarrollo de la aplicación web:

- Lógica: tiene dos funciones. Una se encarga de añadir items a la instancia de “Gilded\_rose” y la otra se encarga de generar el html de los items devueltos por dicha instancia.
- App: desarrolla la aplicación web mediante lo que le importamos de lógica.

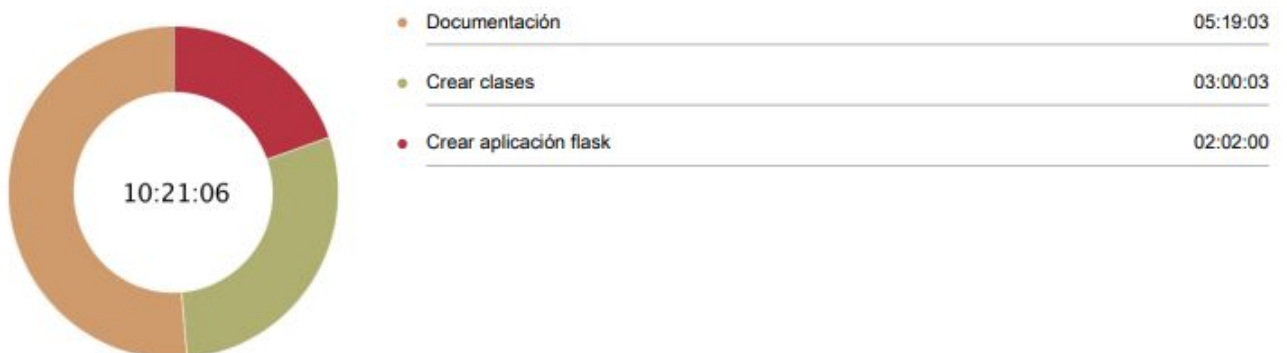
## Resumen del desarrollo de la app

### 4.1 Estructura de desarrollo

Durante el desarrollo del proyecto se han usado cuatro ramas:

- master: rama principal y en ella se desarrollaron los ficheros básicos de la arquitectura de del proyecto, es decir, “Gilded\_rose”, “Item”, “Normal\_item”, “Updatable”, “Aged\_brie”, “Sulfuras”, “Conjured” y “Backstage\_pass”.
- lógica: se ha desarrollado la lógica de la aplicación web, es decir el fichero “logica”.
- flask: se ha desarrollado el fichero que ejecuta flask, es decir “app”.
- documentación: en ella se ha realizado el “readme” del proyecto y se ha añadido la documentación específica del proyecto.

### 4.2 Diagrama de tiempo de desarrollo



### 4.3 Análisis de desarrollo

Una de las dificultades más notables durante el desarrollo de la aplicación ha sido la implementación del paradigma de programación orientada a objetos debido a la falta de tiempo durante el curso para poder practicar más con ello.

Por otra parte, otra de las dificultades ha sido el como usar “Flask” ya que resultaba ser una tecnología algo desconocida para nosotros.

Todas nuestras complicación durante el proyecto se han podido solucionar con una búsqueda de información sobre los temas comentados anteriormente.