

PFM05. IMPLEMENTACIÓN

Sistemas de Gestión Empresarial

2º Curso // CFGS DAM // Informática y Comunicaciones

Alfredo Oltra

**Cicles
Formatius**

ÍNDEX

1	Introducción	4
2	Objetivo	4
3	Entrega	4
4	Consejos	6

Versión: 230914.2238

Licencia



Reconocimiento – NoComercial – CompartirIgual (by-nc-sa). No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

☑ Atención.

📖 Interesante.

PFM05. IMPLEMENTACIÓN

1 INTRODUCCIÓN

Llegamos a la parte más práctica de todo el proyecto, la **codificación**. Una vez ya hemos planificado con bastante detalle el módulo, es el momento de empezar con su implementación. Es, sin duda, la **fase más importante** de todo el PFM y la que centraliza gran parte de los conocimientos del módulo, por lo que su dedicación y, por lo tanto, su duración es más larga.

2 OBJETIVO

El objetivo de esta fase **implementar** y **hacer funcional** el **módulos** (o módulos) descritos en la fase 4.

3 ENTREGA

La **entrega** de esta fase consistirá en:

1. Un **documento en fomato *md*** indicando:
 - las **urls** de los **repos**,
 - el **orden** de **instalación** (por si un módulo depende de otro)
 - por **cada rol**, los **usuarios** y **contraseñas**, así como los **datos** de **autenticación** del **administrador**,
 - **listado** de **modulo.modelo** de **otros módulos** de los que **depende**.
 - si son necesarios, listado de **servicios externos**. En este caso será necesario indicar la **url** del fork de **odoodock** utilizado que incluya esos servicios.
2. **Video corto** a modo de **manual del usuario**, donde se pueda ver el **ciclo completo** de trabajo del módulo..
3. La **publicación** en el **remoto_alumno** de **github** del código de los módulos, **de** tal manera que:
 - los **módulos** sean **funcionales** con el código subido a **github**. Esto implica que si existen **dependencias** de otros **módulos** deben ser incluidas en el fichero **__manifest__.py**.
 - se definan **datos** de **demo** para poder **realizar pruebas**: **usuarios**, **productos**, **proyectos**, ... Físicamente esos datos deben estar dentro de la **carpeta data/demo** (¡ojo! no en **/demo**) y se **cargarán** desde el **apartado data** del **__manifest__.py**, **no** desde el apartado **demo**.



La ubicación de los datos de demo dentro de la carpeta data es por razones logísticas a la hora de corregir.

No hay que confundir los datos demo con los datos necesarios en la carga (datos cuasiestáticos), por ejemplo los datos departamentos.

- Acceso a la documentación del módulo vía *github pages*. Para ello la documentación se ubicará en la carpeta */docs*. Esa documentación debe incorporar, además de lo incluido en la fase 4, información sobre el proceso de instalación y configuración

Para realizar un control más sencillo de quien ha realizado la fase, será necesario entregar en la tarea PFM-F5 el documento indicado en el punto 1 y el video del punto 2. A partir de esa información el profesor descargará los proyectos alojados en *remoto_alumno* de *github*.



La evaluación de la fase se realizará clonando los repos del alumno, arrancando los contenedores (o clonando y arrancando el fork de *odoo dock* indicado por el alumno) e instalando los módulos. En el caso de que ese proceso no permita la comprobación de la fase, automáticamente se dará por no válida. Es por ello que, para evitar esta circunstancia, es conveniente que el alumno recree todo el proceso en su ordenador para comprobar que no da errores.

Recuerda que esta entrega cierra la segunda evaluación, por lo que la fecha límite indicada marca el límite para todas las entregas de todas las fases anteriores en el caso que desees ser evaluado en el curso. El caso contrario no implica el suspenso del módulo: puedes seguir desarrollando el proyecto hasta la ordinaria siempre que cumplas las condiciones comentadas en el documento de introducción del PFM y en la programación del curso.



La fecha de límite de entrega será el 21 de febrero de 2024 a las 23:55. El día 22 de febrero a las 0:05 se lanzará un script de clonación de todos los proyectos con relación a la cuenta asignada. Si el proyecto no es clonado, se dará como no entregado y se calificará con un 0.

4 CONSEJOS

- Ves paso a paso y prueba. No codifiques a lo loco esperando que todo funcione luego bien.
- “Commitea”. No pasa nada por hacer muchos *commits*. Y por supuesto semánticamente correcto y que aporte información par apposteriores búsquedas. Un *commit* que con un texto: “*fase casi acabada*” no aporta nada.
- Para cualquier duda sobre el enunciado o sobre posible información a incluir en la empresa puedes utilizar el foro *PFM-F5. Dudas implementación*