

Planificación y Gestión de Proyectos Informáticos



Tipo de documento: Sprint Retrospective Iteración 3

Nombre del proyecto: Adquisición de obras de arte en formato físico

Grupo: 2.12

Fecha: 26/11/2023

SPRINT RETROSPECTIVE

NOMBRE DEL PROYECTO:	Adquisición de obras de arte en formato físico (ArtUS)
CÓDIGO DEL PROYECTO:	2.12
DIRECTOR DEL PROYECTO:	Jesús Torres Valderrama
FECHA DE ELABORACIÓN:	01/12/2023

HISTORIAL DE VERSIONES			
FECHA Y HORA	N° DE VERSIÓN	DESCRIPCIÓN	ELABORADO POR
01/12/23 - 10:50	1.0	Primera versión del documento	Equipo de desarrollo
10/12/2023 - 12:33	2.0	Revisión del documento	Equipo de desarrollo

PROPÓSITO DEL SPRINT RETROSPECTIVE

El propósito principal del Sprint Retrospective es permitir que el equipo de desarrollo inspeccione y adapte sus procesos para mejorar continuamente su eficacia y colaboración de cara a las siguientes iteraciones que debe realizar para el proyecto.

Este documento proporciona un espacio dedicado para que el equipo reflexione sobre el Sprint recién finalizado. Se revisan los logros, los desafíos, las dificultades y los éxitos alcanzados durante el Sprint.

Asimismo, se identifican oportunidades para mejorar tanto los procesos de trabajo como la dinámica del equipo. Se buscan áreas donde se pueda optimizar el flujo de trabajo, la comunicación, la colaboración o cualquier aspecto que afecte la eficiencia y la calidad del trabajo. Todo esto se recoge en el plan de acción.

METODOLOGÍA SEGUIDA POR EL EQUIPO

SPRINT RETROSPECTIVE

Gestión de issues.

En este Sprint, se procedió de igual manera a las iteraciones anteriores, pues esta manera de trabajo funcionaba. Para ello se creó una tarea por elemento o conjunto de elementos del product backlog y se hizo uso de las tareas épicas para denominar aquellas que tenían una mayor carga de trabajo. A la hora de repartirlas, se intentó buscar que no hubiera dependencias entre tareas de distintos miembros en la medida de lo posible. Como principales puntos a mejorar en la última iteración:

- Repartición de tareas. El reparto ha vuelto a decrecer respecto al segundo sprint, ya que han habido dos miembros que se han llevado una gran carga de trabajo en comparación con otros integrantes.
- Estimaciones de las tareas. Se siguió usando la división de analistas y programadores dentro del equipo de desarrollo. Los analistas se encargaron de la documentación a la par que realizaron pequeñas mejoras en el código y cambios menores y los desarrolladores hicieron grandes cambios e importantes pero no tocaron documentación, consiguiendo equilibrar la carga de trabajo. Sin embargo, en este sprint, el trabajo de los programadores fue denso.
- Al haber menos personas tocando código, apenas han habido problemas de dependencia y si los ha habido, se han solucionado con comunicación.

Gestión de ramas.

En este Sprint se ha seguido usando la metodología GitFlow, y la gestión de ramas ha sido adecuada:

- Seguimiento adecuado de la política de ramas definida. Se ha trabajado en ramas de feature asociadas a issues que trataban de agregar funcionalidades al sistema, refactor para aquellas tareas de mejora de códigos o fix en caso de querer hacer un arreglo cuando ya se ha subido el código. Luego, se ha volcado todo el código en una rama develop que sirve de intermediaria entre las otras ramas y main.

Cabe destacar aparte, que la nomenclatura de las ramas ha sido correcta en todo momento, aunque al final, por prisas de finalizar el proyecto a tiempo, se ha roto un poco el uso de la metodología.

Seguimiento y control

Se ha mejorado la manera de proceder del seguimiento del proyecto para esta iteración siguiendo las recomendaciones proporcionadas por el patrocinador. Respecto al control de cronograma y costes se ha proseguido con la misma herramienta y se han ajustado bien los retrasos y

SPRINT RETROSPECTIVE

cambios que se han dado durante esta iteración, a pesar de que todavía el equipo no está acostumbrado al uso de MS Project. Asimismo, se ha generado el documento de Análisis del Valor Ganado para esta iteración y las anteriores.

REFLEXIÓN DEL SPRINT

LOGROS / ÉXITOS	DESAFÍOS / DIFICULTADES
<ul style="list-style-type: none">El equipo ha logrado realizar el trabajo requerido para la iteración.Lo realizado funciona correctamente y la interfaz se ve bonita.Los miembros han sabido organizarse y coordinarse correctamente con otros miembros.A pesar de haber tenido incidencias, el equipo ha logrado solventarlas en el menor tiempo posible.Satisfacción del cliente del producto mostrado a rasgos generales.Se ha mejorado considerablemente la comunicación con los miembros.	<ul style="list-style-type: none">Aparición de cambios a petición del cliente ya que no cumplían o con la calidad del proyecto o con algunos requisitos.Realizar el proyecto en presencia de días festivos y fines de semana.Arreglar y cambiar los estilos al proyecto entero en cuestión de días.Mayor carga de trabajo de programación para el equipo.

PLAN DE ACCIÓN

START	STOP	CONTINUE DOING
<ul style="list-style-type: none">Realizar un mejor reparto de las tareas entre los miembros para que sea más equitativo.	<ul style="list-style-type: none">Hacer crunching para terminar las tareas, cuando se podría haber dividido de mejor forma las tareas para tener menor carga de trabajo.	<ul style="list-style-type: none">Cumplimiento de GitFlow como política a seguir de commits, tareas, pull requests y ramas.Hacer una revisión de las distintas implementaciones antes de que estas se

SPRINT RETROSPECTIVE

<ul style="list-style-type: none">● Hacer tareas más atómicas, sin tanta carga de trabajo o realizar programación por pares.● Comprobación más exhaustiva de que el proyecto cumple con los requisitos recogidos y que tiene la calidad necesaria para evitar tener que realizar cambios de última hora.● Celebrar los éxitos del equipo para que ningún compañero se canse y acabe desinteresándose por el proyecto por desmotivación o cansancio.	<ul style="list-style-type: none">● Desentendimiento o desinterés del proyecto por parte de algunos miembros del equipo, ya que dificulta la comunicación y ocasiona un desajuste en la carga de trabajo.● Dar por sentado que algunos requisitos se cumplen sin haberlo comprobado claramente.	<p>acepten. Para aceptarla, debe haber al menos una valoración positiva.</p> <ul style="list-style-type: none">● Seguir trabajando de manera casi diaria y consistente.● Subdivisión de roles dentro del equipo de trabajo, en desarrolladores y documentadores.● Reuniones semanales con el equipo de dirección para determinar la calidad del proyecto y preguntar dudas.● Realizar solicitudes de cambio cuando haga falta realizar un cambio en el proyecto.● Mantener la comunicación sostenida que se ha tenido con los otros miembros.● Tener un seguimiento diario de los costes y del cronograma y documentar todos los cambios que sucedan cada día.
---	--	---