

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
	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 DEL EQUIPO DE DESARROLLO

SPRINT RETROSPECTIVE

Gestión de issues.

En este Sprint, se procedió de manera similar 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 mejorado respecto al anterior sprint, aunque este podría ser aún mejor buscando como objetivo que cada miembro del equipo de desarrollo pueda trabajar de forma totalmente independiente en sus tareas y por tanto ser más eficiente y rápido.
- Estimaciones de las tareas. Se ha mejorado la carga de trabajo para los distintos integrantes, en especial gracias a la designación de personas dedicadas a la documentación y las personas designadas a la programación. 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.

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.

Seguimiento

Se ha mejorado la manera de proceder del seguimiento del proyecto para esta iteración siguiendo las recomendaciones proporcionadas por el patrocinador. Añadir que todavía se presentan algunas dificultades de como realizar debido a ciertos cambios en el cronograma por retrasos en alguna actividad.

SPRINT RETROSPECTIVE

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.	<ul style="list-style-type: none">● Mantener una comunicación sostenida con los miembros.● Aparición de incidencias por despiste del equipo de desarrollo hacia los requisitos del proyecto.● Adaptación del equipo al proyecto debido a circunstancias externas al mismo.● Conflictos entre las diferentes ramas a la hora del desarrollo provocando un pequeño incremento en el tiempo invertido.

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 equitativo.● Fomentar una mejor comunicación entre los miembros del equipo y con el equipo de dirección.● Hacer tareas más atómicas, sin tanta carga de trabajo o realizar programación por pares.	<ul style="list-style-type: none">● Realizar incrementos del código sin tener en mente los requisitos del proyecto.● Realizar commits con una nomenclatura incorrecta.● Crear issues sin seguir las plantillas proporcionadas.	<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 acepten. Para aceptarla, debe haber al menos una valoración positiva.● Seguir trabajando de manera casi diaria y consistente.● Cada miembro se encarga de una tarea a la vez, es decir, una tarea en "In progress" por persona.

SPRINT RETROSPECTIVE

		<ul style="list-style-type: none">● Se crean las pull request asociadas inmediatamente después de que la tarea sea movida a la columna de "In review".● Mover las tareas dentro de GitHub para representar correctamente el estado en el que se encuentra cada una.● 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.
--	--	--