

PLAN DE GESTIÓN DE LOS REQUISITOS

NOMBRE DEL PROYECTO:	HouseRent
CÓDIGO DEL PROYECTO:	PGPI-G2.11
DIRECTOR DEL PROYECTO:	Alejandro Pérez Vázquez Juan Carlos Ramírez López Carmen Ruiz Porcel Sergio Santiago Sánchez María Vico Martín
FECHA DE ELABORACIÓN:	16 / 10 / 2023

HISTORIAL DE VERSIONES			
FECHA Y HORA	Nº DE VERSIÓN	DESCRIPCIÓN	ELABORADO POR
16 / 10 / 2023 23:00	1.0	Se realiza la primera edición del documento, incluyendo la información que se cree pertinente.	Alejandro Pérez Vázquez Juan Carlos Ramírez López Carmen Ruiz Porcel Sergio Santiago Sánchez María Vico Martín
18 / 10 / 2023 16:00	1.1	Complemento de la información ya detallada	Alejandro Pérez Vázquez Juan Carlos Ramírez López Carmen Ruiz Porcel Sergio Santiago Sánchez María Vico Martín
19 / 10 / 2023 20:00	1.2	Refinamiento de la gestión de la configuración.	Alejandro Pérez Vázquez Juan Carlos Ramírez López Carmen Ruiz Porcel Sergio Santiago Sánchez María Vico Martín

PROPÓSITO DEL PLAN DE GESTIÓN DE REQUISITOS

¿Cuál es el objetivo de este documento?

El propósito del presente documento es tratar de garantizar la correcta y eficaz recopilación y validación de los requisitos, así como el procedimiento por el cual categorizan y priorizan los requisitos. Obteniendo de este modo un procedimiento donde recogemos, clasificamos, priorizamos y validamos los requisitos en base a una serie de protocolos y directrices que garanticen el mejor resultado.

PLAN DE GESTIÓN DE LOS REQUISITOS

RECOPILACIÓN DE REQUISITOS

Para la recopilación de los requisitos el equipo deberá reunirse con el cliente / patrocinador, con el fin de preguntarle acerca del sistema que desea obtener. Una vez el cliente describa como es la aplicación / web que desea, el equipo deberá de preguntar sobre aquellos aspectos que hayan podido dotarse con algo más de ambigüedad. De este modo, mediante las historias de usuario, descritas por el patrocinador, y las preguntas realizadas por el equipo podremos recabar la mayoría de requisitos. Si bien, en una fase posterior del proceso pueden verse ligeramente modificados, siempre y cuando se acuerde previamente.

ANÁLISIS DE REQUISITOS

Se realizará una puesta en común para analizar los requisitos, poder obtener su nivel de abstracción más bajo. A su vez, se deberá de llevar a cabo una planificación mediante Planning poker, logrando así un consenso. Por tanto, los elementos propios de la EDT se describirán en dicha lluvia de ideas. En caso de duda o conflictos con los requisitos se deberá comunicar al cliente para que este pueda aportar algo de claridad sobre los mismos. Es importante mencionar que a la par que recabamos información sobre los requisitos y estimamos el tiempo de desarrollo, debemos de aplicar un nivel de importancia, pues para el cliente no todos los requisitos son iguales, siempre hay algunos con mayor prioridad que otros.

CLASIFICACIÓN DE REQUISITOS

Con el fin de poder clarificar la naturaleza de cada requisito por desarrollar, estos se deben de clasificar en: funcionales, no funcionales y de calidad. Si bien los de calidad son también no funcionales, le damos cierta importancia porque los atribuimos a alguna característica que el cliente nos ha indicado que será distintiva para el, no por ello más importante.

DOCUMENTACIÓN DE REQUISITOS

Se deberán de incluir las posibles opciones, solución seleccionada y conflictos que se hayan encontrado en aquellos requisitos de complejidad media-alta.

PRIORIZACIÓN DE REQUISITOS

Los requisitos los clasificaremos dependiendo de su importancia para que el sistema funcione correctamente y/o el nivel de importancia para el cliente, pues aunque un requisito no sea crucial para el correcto funcionamiento del sistema si lo puede ser para la satisfacción del cliente. De este modo, clasificaremos los requisitos según resulten de prioridad: muy alta, alta, media, moderada, baja

MÉTRICAS DE REQUISITOS

Se tendrá en cuenta que los requisitos se describen de forma sencilla y detallada, de tal forma que puedan ser entendidos por todos los miembros del equipo, evitando problemas de comprensión. Debemos de asegurar la no conflictividad entre requisitos. Los requisitos se trasladarán en a issue / ticket para mantener a los desarrolladores informados al respecto del estado del proyecto. Como el proyecto avanza y puede verse modificado, se llevará a cabo un registro de los cambios en los requisitos, como prioridad, descripción, añadido o eliminado.

PLAN DE GESTIÓN DE LOS REQUISITOS

VALIDACIÓN DE REQUISITOS

Se proveerá de la aplicación tanto al cliente, como a una serie de usuarios de prueba, con el fin de comprobar la usabilidad y la intuitividad del sistema. Así mismo se realizan alquileres ficticios para comprobar el correcto funcionamiento. No obstante, el sistema también deberá contar con una batería de test que puedan ser ejecutados en cualquier momento, para comprobar el funcionamiento de al menos las partes más importante.

TRAZABILIDAD DE REQUISITOS

Desde la primera concepción de los requisitos se llevará a cabo un estudio para desglosar todos ellos en requisitos de menor nivel, es decir, desde la descripción del requisito por parte del cliente pretendemos alcanzar todos los requisitos a un nivel de abstracción mínimo.

INFORME DE REQUISITOS

Todo avance en los requisitos debe ser documentado, de tal forma que quede registrado el estado actual, los posibles problemas que hayan surgido, además de en casos donde se considere un requisito complejo, deberán de ser documentadas las distintas opciones posibles junto con qué opción ha sido seleccionada y su razón.

GESTIÓN DE LA CONFIGURACIÓN

En el presente proyecto se evaluará cada versión siguiendo el esquema X.Y.Z, donde:

- **X:** Hace referencia a un cambio de considerable envergadura, como una nueva función antes no presente en el sistema.
- **Y:** Indica los cambios pequeños, funciones minimalistas.
- **Z:** Indicativo de aquellas versiones que suponen la corrección de algún tipo de error o fallo.

Asimismo, aplicaremos la estrategia de ramas conocida como **git flow**, la cual se basa en una rama **Main/Master**, seguida de una **Develop** y una rama por cada **feature**, es importante destacar que toda documentación deberá formar parte primero de la rama **HOUS-DOCS**, la cual, además de **develop**, es la única que tiene permitido verter sus cambios a **Main**. Para poder trasladar los cambios de una rama de feature a develop será necesaria la aprobación de un segundo miembro del equipo, mientras que para verter los cambios de Develop a Master será necesaria la aprobación de dos miembros extra, es decir, el solicitante y dos compañeros más. Es importante mencionar que no se puede realizar ningún cambio directamente en Máster, salvo que se corresponda con una **HotFix**, es decir, la solución de un error o fallo grave.




El proyecto en producción, desplegado, se corresponderá con el código presente en la rama Master, siendo necesario un despliegue de la misma cada vez que se realice un cambio.

La gestión de los tickets / issues se realizará con Jira, por lo que los commits deberán seguir la estructura **<CODE>-<X>:<ACTION>**, donde:

- **CODE:** Representa el código del proyecto en Jira.
- **X:** Representa el número asociado al ticket.
- **ACTION:** Indica el cambio realizado.

PLAN DE GESTIÓN DE LOS REQUISITOS

APROBACIÓN

Nombre	Cargo	Firma	Fecha
Jesús Torres Valdegrama	Patrocinador		19 / 10 / 2023
Alejandro Pérez Vázquez	Director del Proyecto, Desarrollador		19 / 10 / 2023
Juan Carlos Ramírez López	Director del Proyecto, Desarrollador		19 / 10 / 2023
Carmen Ruiz Porcel	Director del Proyecto, Desarrollador		19 / 10 / 2023
Sergio Santiago Sánchez	Director del Proyecto, Desarrollador		19 / 10 / 2023
María Vico Martín	Director del Proyecto, Desarrollador		19 / 10 / 2023