

Cuentas Claras

Retrospectiva cuentas claras grupo 48

Drop	Votes
Definir el flujo de trabajo de manera mas completa para que abarque la gran mayoría de todos los procesos necesarios durante el desarrollo- Daniel V	2
Asumir que una funcionalidad debe actuar de alguna forma en específico. (Andrea)	1
Entregar mas de lo definido en las historias de usuario -Daniel V	1
Llegar sin haber revisado el contenido de la semana. (Andrea)	0

Add	Votes
Hacer pruebas de sistema para validar que al final del incremento no se ha afectado alguna funcionalidad. (Andrea)	2
Realizar pausas activas para que el equipo contrarreste los efectos de largas sesiones de trabajo continuo -Daniel V	1
Implementar patrones de diseño en la aplicacion donde se vea que puedan ser usados -Daniel V	0
Añadir comentarios en el codigo para que los demas desarrolladores puedan entender rapidamente que hace una funcionalidad -Daniel V	0

Keep	Votes
Hacer pair Programming en las funcionalidades que lo requieran. (Andrea)	1
Uso de TDD para desarrollar sólo el código necesario para cumplir con lo requerido (Andrea)	1
Mantener los canales de comunicación de manera continua. (Andrea)	1
Repartir responsabilidades a principio de la iteración(Andrea)	1
Uso de TDD para desarrollar sólo el código necesario para cumplir con lo requerido. (Andrea y Daniel)	1
Mantener el trabajo en equipo buscando el alcanzar los objetivos. (Andrea y Daniel)	1
El uso de mensajes de commit descriptivos que ayudan a la hora de revisar el historial de cambios -Daniel V	0
Continuar con el trabajo en equipo.(Andrea)	0

Establecimiento de los acuerdos del flujo de trabajo al inicio de la iteración. (Andrea)	0
Uso de TDD para desarrollar sólo el código necesario para cumplir con lo requerido -Daniel V	0
El uso de integración continua para automatizar ciertas tareas -Daniel V	0
Mantener el trabajo en equipo buscando el alcanzar los objetivos. -Daniel V	0
Uso de herramientas adicionales que facilitan el desarrollo(Sqlite db browser, el debugger, etc...) -Daniel V	0

Improve	Votes
Evaluar o revisar todos los escenarios posibles al desarrollar una funcionalidad. (Andrea y Daniel)	2
Optimización del tiempo de trabajo con la idea de evitar largas sesiones de desarrollo que provocan efectos dañinos en el equipo -Daniel V	1
Hacer merge de main en develop , después de hacer la mezcla en main para evitar errores. (Andrea)	1
Actualización del diseño de la solución a medida que hay cambios en el desarrollo. (Andrea)	1
Evaluar todos los escenarios posibles al desarrollar una funcionalidad. (Andrea)	1
Comentar blockers apenas sucedan con el fin de avanzar mas rapidamente en el desarrollo del proyecto -Daniel V	1
Hacer refactoring del codigo mas seguido y asi mejorar la calidad del proyecto de manera continua -Daniel V	0
Revisar todos los escenarios posibles previo al desarrollo -Daniel V	0

Cuentas Claras

Acciones de mejora

Aspects	Votes
Definir el flujo de trabajo de manera mas completa para que abarque la gran mayoría de todos los procesos necesarios durante el desarrollo-D	2

Action Plan	Votes
Al principio de la iteración evaluar a detalle requerimientos de las actividades y agregarlos dentro de la definición del flujo de trabajo.	0
Revisar documentación y proyectos de ejemplo que sirvan para diseñar nuestro propio flujo de trabajo.	0
En caso de duda, preguntar a expertos.	0

Cuentas Claras

Acciones de mejora

Aspects	Votes
Hacer pruebas de sistema para validar que al final del incremento no se ha afectado alguna funcionalidad. (Andrea)	2

Action Plan	Votes
Tener casos de prueba preparados con los datos esperados para poder validar las funcionalidades.	0
Definir flujos de pruebas para las funcionalidades.	0
Hacer el ciclo de pruebas al finalizar el desarrollo y repetirlo cuanto se hagan las mezclas con otras ramas.	0
Tener definido el Done.	0