



Alquifiestas: Sensacional Eventos
Tarea 1
Reflexión de Aplicación de Lean Software Development

Hugo Eduardo Rivas Fajardo - 22500
José Santiago Pereira Alvarado - 22318
Nancy Gabriela Mazariegos Molina - 22513
Giovanni Alejandro Santos Hernández - 22523
Mauricio Julio Rodrigo Lemus Guzmán- 22461
Alexis Mesías Flores - 22562

Tabla de reflexión:

Principio	Aplicación	Porcentaje	Mejoras
Eliminar desperdicios/restos	Para la aplicación de este principio en nuestro proyecto, nosotros tratamos de la mejor manera posible el poder eliminar carpetas innecesarias en el proyecto, como lo pueden ser las “_pycache_”, además tratamos de refactorizar nuestro código de manera que pudiéramos reducir código repetitivo y realizar las mismas funciones con bloques de códigos más pequeños y eficientes.	90%	<ul style="list-style-type: none"> ● Hacer una evaluación de todo el proyecto cada vez que se implementa algún cambio, para siempre revisar si está lo más abstracto posible. ● Asignar la tarea de eliminar desperdicios a uno o dos compañeros del grupo. ● No crear o implementar funcionalidades extras que el product owner no haya solicitado o que no se encuentren en el product o sprint backlog.
Amplificar el aprendizaje	Este principio es de los que más se aplicó debido a que cada integrante del grupo tiene distintos conocimientos sobre cómo se puede aplicar un lenguaje de programación conforme a la funcionalidad que se le quiera otorgar. Se práctico el continuo aprendizaje al investigar y utilizar librerías que parte del grupo no conocía pero que de una manera nos iba a beneficiar en el proyecto. En algún punto se decidió cuáles iban a ser	75%	<ul style="list-style-type: none"> ● Realizar reuniones para intercambiar y/o explicar las distintas tecnologías que se utilizan en el proyecto. ● Hacer una puesta en común de aspectos que nos podrían ayudar a mejorar el proyecto. ● Sesiones para resolución de dudas dentro del proyecto o en el proceso de este mismo. ● Una o dos veces a la semana realizar una revisión de código para verificar cambios y/o

	los lenguajes y tecnologías que se utilizaron para el proyecto para así tener definido y organizado todo lo que se va a hacer.		arreglos de este mismo.
Tomar decisiones lo más tarde posible	En el desarrollo de nuestro software para alquifiestas, hemos aplicado este principio al dejar para el final la toma de decisiones sobre el control de las fechas de entrega y recogida de los artículos. Esto nos ha permitido mantener la flexibilidad en el diseño y funcionalidad del sistema, adaptándonos a los cambios y requisitos del cliente a lo largo del desarrollo.	70%	<ul style="list-style-type: none"> ● Revisiones constantes y verificación del cumplimiento de los requisitos del cliente para corroborar que vamos en la línea de trabajo correcta. ● Desarrollar el proyecto a manera que si se quiere hacer un cambio de último momento no se tenga que cambiar todo el código. ● Crear prototipos o realizar pruebas tempranas a los usuarios para tener un feedback real.
Entregar lo antes posible	Para nosotros fue primordial poder hacer las entregas lo antes posible. Debido a que nos ayuda a poder darnos cuenta de algún error o problema que tengamos y poder solucionarlo sin ningún problema. Ya que a última hora puede ser un poco complicado.	75%	<ul style="list-style-type: none"> ● Hacer reuniones con avances para poder dar todos nuestra opinión sobre los avances de los demás. ● Estar en constante comunicación con los miembros del equipo. ● Realizar pruebas del código lo antes posible para así hacer la entrega completamente funcional.
Potenciar el equipo	Para el proyecto se	90%	<ul style="list-style-type: none"> ● Mejor organización

	<p>dividieron bastante bien las tareas en base a la complejidad y en el tiempo, las tareas también fueron divididas dependiendo de las aptitudes de cada integrante del grupo. La división de tareas fue elaborada de manera estratégica, el grupo se dividió en grupos diferentes ya que cada uno de estos tenían encargada una parte como los fue el backend y frontend. Dentro de estos dos grupos cada quien se encargaba de un aspecto del grupo al que corresponden.</p>		<p>de tareas para cada integrante.</p> <ul style="list-style-type: none"> ● Realizar sesiones para poder discutir de lo realizado y/o hacer puestas en común. ● Revisar la retroalimentación y discutir acerca de esta misma. ● Realizar una lluvia de ideas con el grupo ya sea de mejoras o cosas que se le puedan implementar al proyecto.
Crear la integridad	<p>Hemos aplicado este principio en el proyecto para verificar el funcionamiento correcto de las herramientas desarrolladas para garantizar su desempeño al product owner. Lo hemos realizado al realizar pruebas de usabilidad para asegurar que los componentes realizados funcionen en un entorno de trabajo, por ejemplo, modificar el inventario.</p>	70%	<ul style="list-style-type: none"> ● Generar pruebas automatizadas de extremo a extremo que verifiquen el funcionamiento de las historias de usuario realizadas. ● Crear pruebas de integración al momento de realizar cambios para verificar la funcionalidad. ● Realizar sesiones semanales para verificar la usabilidad y proponer mejoras.
Visualizar todo el conjunto	<p>La idea principal de tener una visualización correcta de todo el sistema, nos permite ver las necesidades, tanto las que se han solventado, como las</p>	100%	<ul style="list-style-type: none"> ● Diseños eficientes ● Maneras sencillas de acceder a todos los documentos de apoyo. ● Hilos de trabajo optimizados para el

	que no se han solventado. El conjunto conlleva las necesidades técnicas y específicas solicitadas por el PO, así como las necesidades que no se miran.		team work
--	--	--	-----------

Conclusión:

- Durante el desarrollo del sistema, desde la fase de planeación, se han implementado muchas características de la metodología LEAN. Hay una gran diferencia entre la planeación y la implementación. Durante la fase de planeación, se alcanzó un cierto porcentaje de implementación de LEAN, que se mantuvo a lo largo del desarrollo. Por lo tanto, se puede afirmar que el porcentaje de LEAN utilizado de manera adecuada a lo largo del proyecto es correcto. Aunque hay áreas que se pueden mejorar, en esencia, se ha implementado de manera adecuada.

Referencias:

- Vajda, J. (2010, 3 mayo). Lean Software development Principles [Diapositivas]. SlideShare. <https://es.slideshare.net/jpvajda/lean-software-development-principles>