

REFLEXIONES UNIDAD 2

ASHLEY SHADEN AGUILAR MORENO

1. Habilidades para Requerimientos

La ingeniería de requerimientos es el espectro amplio de tareas y técnicas que llevan a entender los requerimientos. Entre sus tareas principales se tienen a la indagación, la negociación y la validación haciendo selección de estas ya que ocurren en paralelo con otras y de cualquier forma todas las tareas se adaptan a las necesidades del proyecto; estas 3 son importantes y requieren de ciertas habilidades. Comenzando con la indagación esta es sumamente importante en el ciclo debido a que aquí se le pregunta al cliente, a los usuarios y a otras personas el objetivo para el sistema o producto por lo que se necesitarían de habilidades como las comunicativas, siguiendo con la negociación que en esta su importancia radica en que aquí se resuelven conflictos en cuanto a que los clientes y usuarios pidan más de lo que puede lograrse dado lo limitado de los recursos del negocio, se pide a clientes, usuarios y otros participantes que ordenen sus requerimientos según su prioridad y que después analicen los conflictos por lo que aquí se necesitarían habilidades de resolución de problemas y adaptabilidad, por último la validación es importante en el ciclo debido a que en esta se analiza la especificación a fin de garantizar que todos ellos han sido enunciados sin ambigüedades; que se detectaron y corrigieron las inconsistencias, las omisiones y los errores, y que los productos del trabajo se presentan conforme a los estándares establecidos para el proceso, el proyecto y el producto aquí la habilidad más importante es la de análisis y el mecanismo principal la revisión técnica.

2. CU vs US

Tanto la especificación de casos de uso y las historias de usuario representan las características que tiene que tener el sistema o el producto a elaborar, sin embargo las historias de usuario sirven para describir lo que el usuario desea ser capaz de hacer, en cambio los casos de uso describen detalladamente las interacciones del software con actores externos para satisfacer la necesidad de la funcionalidad por parte del usuario (el

para qué). Las historias de usuario están concebidas como un medio para fomentar la colaboración debido a que con el objetivo definido, el equipo puede colaborar para decidir cómo ofrecer un mejor servicio al usuario, mientras que los casos de uso se apoyan fuertemente en las especificaciones documentadas del sistema en lugar de colaboración tradicional de los métodos ágiles, si no existe un enfoque integrado en la colaboración, es posible que se ahonde en una especificación detallada en el caso de uso que se convierte en la fuente de registro en lugar de un mecanismo de conversaciones por lo que se puede decir que tienen un enfoque distinto. En este sentido se recomendaría usar las historias de usuario en un ambiente de colaboración como una herramienta de comunicación y la especificación de casos de uso especialmente en sistemas con más interacción hombre/máquina.

3. Políglotas

Para empezar hay personas que recomiendan el enfocarse a aprender los lenguajes de programación por separado y otras que recomiendan aprender algunos (obviamente no todos porque son muchos) al mismo tiempo, algo importante es entender que no todas las personas aprendemos de la misma forma, lo que a unas personas les puede ayudar a aprender de una manera rápida puede no servirles a otras y esto no tiene nada de malo, solo que tenemos que encontrar la manera que mejor se acople a nosotros. Los lenguajes tienen un fin en común que es comunicarnos con la computadora, sin embargo, no todos lo hacen de manera similar e incluso pueden hacerlo de formas muy distintas, de hecho algo importante antes de aprender diferentes lenguajes es el evitar suponer que son iguales o por ejemplo, cuando ya sabemos uno y queremos aprender otro debemos evitar suponer que el siguiente será igual o parecido al que ya sabemos para así evitar confundirnos o revolverlos, partiendo de esto la estrategia es empezar investigando y entendiendo bien conceptos relacionados que nos hagan entender mejor por qué se comunican, es decir, cual es el objetivo y de ahí conocer la lógica de programación, las funcionalidades y las diferentes sintaxis, por ejemplo, como ya sé que se comunican para crear algún programa que realice la suma de 3 números, pues bien, investigo como se realiza la función suma en tal lenguaje

y posteriormente anotó esas diferencias y/o similitudes con distintos lenguajes. Finalmente, teniendo en cuenta todo lo mencionado anteriormente lo más importante es el practicar, practicar y practicar con todos los lenguajes aprendidos y/o por aprender, ya que esta es la mejor forma de reforzar nuestros conocimientos.

4. Seguridad para IS

Según la página web de la compañía Thales Group “La idea detrás de la seguridad del software es crear software que sea seguro desde el principio sin tener que agregar elementos de seguridad adicionales para agregar capas adicionales de seguridad [...] La seguridad del software es fundamental porque un ataque de malware puede causar un daño extremo a cualquier pieza de software al tiempo que compromete la integridad, la autenticación y la disponibilidad. Si los programadores tienen esto en cuenta en la etapa de programación y no después, el daño puede detenerse antes de que comience.”

Ahora bien el plan de estudios propuesto por el grupo de Investigación ALARCOS - Instituto de Tecnologías y Sistemas de Información Dep. de Tecnologías y Sistemas de Información - Escuela Superior de Informática Universidad de Castilla - La Mancha cuenta con la asignatura de Seguridad de Sistemas de Software y según ellos su contenido consta de:

- Fundamentos de seguridad;
- Seguridad organizativa;
- Requisitos de seguridad;
- Seguridad en desarrollo de software;
- Seguridad de sistemas de información;
- Riesgos de seguridad;
- Servicios de seguridad;
- Gestión de seguridad;
- Certificación, normas y estándares para la seguridad.

Con todo esto podemos observar que es necesario como mínimo que los ingenieros de software conozcan, sepan y entiendan los fundamentos del porqué se estaría estudiando acerca de seguridad en ingeniería de software, la importancia y el por qué les serviría, además de que sean capaces de identificar, modelar e integrar los requisitos de seguridad del software en el proceso de su desarrollo, conocer las principales técnicas, mecanismos, servicios y los aspectos más importantes de seguridad del software, y conocer las normas, estándares y legislación más relevante sobre seguridad.