Sistema de gestión de farmacias

Lecciones Aprendidas

Versión 1.1

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Descripción | Autor |
| 14/09/2014 | 1.0 | Primera versión del documento. Hay partes del template sin editar, para que queden como guía al momento de agregar ítems. | Agustín Farías |
| 14/09/2014 | 1.1 | Se agrega información correspondiente a la Gestión de Calidad. | Sofía Palamarchuk |
|  |  |  |  |
|  |  |  |  |

Contenido

[1.Introducción 3](#__RefHeading___Toc176624154)

[2.Por Disciplinas 3](#__RefHeading___Toc176624155)

[2.1.Requerimientos 3](#__RefHeading___Toc176624156)

[2.2.Diseño 3](#__RefHeading___Toc176624157)

[2.3.Implementación 3](#__RefHeading___Toc176624158)

[2.4.Verificación 3](#__RefHeading___Toc176624159)

[2.5.Implantación 4](#__RefHeading___Toc176624160)

[2.6.Gestión de Proyecto 4](#__RefHeading___Toc176624161)

[2.7.Gestión de Configuración y Control de Cambios 4](#__RefHeading___Toc176624162)

[2.8.Gestión de Calidad 4](#__RefHeading___Toc176624163)

[2.9.Comunicación 4](#__RefHeading___Toc176624164)

[2.10.Formación y Entrenamiento 4](#__RefHeading___Toc176624165)

[3.Otras lecciones 4](#__RefHeading___Toc176624166)

[3.1.[Observaciones del Director] 4](#__RefHeading___Toc176624167)

[3.2.[otra lecc. 2] 4](#__RefHeading___Toc176624168)

1. Introducción

Lección Aprendida: Experiencia positiva o negativa obtenida durante la realización de alguna actividad. Se trata del registro de mejores prácticas, problemas recurrentes o experiencias exitosas, durante la implantación del proceso.

La siguiente lista de verificación es actualizada (agregando, modificando y/o eliminando items, cada 15 días en la reunión de equipo).

1. Por Disciplinas
   1. Requerimientos

Las reuniones con el cliente deben tener una hora de comienzo y fin, y lineamientos preestablecidos de los temas a tratar.

* 1. Diseño
* Dividir el sistema en componentes, donde cada uno cumple un rol específico en la realidad. De esta forma se logra que las operaciones en esos componentes estén relacionadas entre sí, logrando componentes cohesivos.
* No utilizar un único controlador para todo el sistema. Es conveniente que haya un controlador por subsistema. De esta forma cada controlador implementa únicamente la interfaz que provee el subsistema. Esto tiene como consecuencia controladores menos acoplados y más cohesivos.
* Diseñar la lógica de negocio sin tener en cuenta aspectos de seguridad. La capa de seguridad puede implementarse luego como una capa en un nivel superior que controla los permisos e invoca a las operaciones de la lógica de negocio. De esta forma se logra dividir mejor las responsabilidades y no acoplar a la lógica de negocio a un mecanismo de seguridad concreto.
* Definir un conjunto de clases común a todas las capas que sirvan para la comunicación entre ellas. De esta forma se logra un mecanismo uniforme de comunicación y no se requiere la transformación de un objeto a otro equivalente con el único fin de transmitirlo de una capa a otra.
* No usar siempre tipos de datos. La definición de tipos de datos es un objeto sin “lógica”, compuesto únicamente por atributos y sus métodos get() y set() correspondientes. Son utilizados para la comunicación entre capas. Generalmente al usar tipo de datos se tiene clases que representan entidades de la realidad y que encapsulan la lógica. Típicamente, tiene métodos del tipo getDatos() que devuelven los tipos de datos que los representan. Esto tiene como desventaja que se requiere implementar más clases y métodos y por lo general las clases de las entidades y las clases de los tipos de datos que los representan son muy similares. El único cuidado que se tiene que tener es que las clases que se utilizaran como tipos de datos no dependan de clases como controladores, etc. Solo pueden depender de clases que representen entidades de la realidad.
  1. Implementación

[El plazo límite para reportar las pruebas unitarias es...]

[No modificar los elementos de línea base sin gestión de cambios...]

[Atención al integrar modulo jjj con modulo mmm...]

* 1. Verificación

[...]

* 1. Implantación

[Verificar que el servidor de producción cuenta con...]

* 1. Gestión de Proyecto
* Organizar reuniones con temas marcados en tiempos acotados.
* Antes de una reunión, preparar los temas a tratar.
* En las reuniones de planificación, procurar que cada responsable de área conozca las tareas de la iteración y los entregables asociados por semana.
* Siempre poner en conocimiento general las planificaciones y discutirlas.
* Asignar tareas individuales en lugar de colectivas.
  1. Gestión de Configuración y Control de Cambios
* El uso de Github no requiere una estructura de carpetas.
* El ambiente controlado de desarrollo tuvo que haber sido creado desde antes que comience la implementación.
* No recaer en que todos conocen las herramientas, ayudarlos en lo que se pueda a que se adapten.
* Seguir de cerca los cambios que se realicen en el repositorio, ya que al principio hay más probabilidad de errores.
  1. Gestión de Calidad
* Planificar y comunicar a los responsables de cada área los entregables de la semana los días lunes o domingos de noche, así pueden organizar las tareas entre los integrantes del área.
* Definir todas las semanas una hora límite de entrega, dependiendo de la cantidad de documentos a entregar y la carga del trabajo de cada integrante.
* Seguir de cerca las reuniones con el cliente, para asegurarse de que los atributos de calidad más relevantes fueron identificados, tener claro cuáles son y definir las vías mediantes las cuales se validará y se hará seguimiento de cada uno.
* Tener comunicación continúa con los responsables de cada área para poder detectar de forma temprana posibles desviaciones y actuar de manera proactiva.
  1. Comunicación
* Por consultas técnicas dirigirse a Santiago Mera o Daniel Rubino.
* Por consultas del entorno del negocio dirigirse a Fabiana Pérez, Santiago Mera o Daniel Rubino.
  1. Formación y Entrenamiento

1. Otras lecciones
   1. [Observaciones del Director]

* [Para futuras reuniones cambiar...]
* [...]
  1. [otra lecc. 2]
* [...]