Tecnológico de Costa Rica

Customer Relationship Manager Reporte de auditoría de código

Version 1.0

Reporte de auditoría de código

Customer Relationship Manager

Índice

Información del documento	2
Descripción	2
Identificación del documento	2
Organización encargada	2
Historial de cambios	2
Introducción	3
Alcance	3
Referencias	3
Glosario	3
Desviaciones con respecto al estándar	3
Observaciones de la implementación	4
Recomendaciones v acciones a tomar	

Customer Relationship Manager

1. Información del documento

1.1. Descripción

El siguiente es un reporte que detalla el resultado de la revisión del código fuente del producto *Customer Relationship Manager*, referido de ahora en adelante como CRM.

Versión del software a revisar: 4.0.

1.2. Identificación del documento

CI102017-2

1.3. Organización encargada

Los siguientes recursos componen la organización encargada de llevar a cabo la inspección pertinente.

- Izcar Muñoz
- Isaac Campos
- Jeffrey Alvarado

1.4. Historial de cambios

Fecha	Versión	Descripción de los cambios	Autor
30/10/2017	1.0	Primera versión	Jeffrey Alvarado

Customer Relationship Manager

2. Introducción

2.1. Alcance

El propósito de este documento es proporcionar un informe resultante del proceso de reunión e inspección de código fuente llevado a cabo en el producto de software CRM. La revisión abarca los cambios y nuevas implementaciones desde el último reporte. Esto incluye los cambios en los conjuntos de funciones Manejo de usuarios y Manejo de contactos, además de la implementación del Seguimiento de ventas y las pruebas automatizadas.

2.2. Referencias

[DEP] Documento de estándar de programación

2.3. Glosario

Las siguientes abreviaciones son usadas en el presente documento:

CRM Customer Relationship Manager

MVC Model, view, controller

3. Desviaciones con respecto al estándar

- Algunos métodos tienen nombres que inician con mayúscula y otros con minúscula dentro de las clases.
- Hay algunos nombres poco significativos, principalmente en las clases
 GUIBuilder, PropuestaVenta y RegistroVentas (propuestaXContacto, verificarEstado, propuestaVentaContactoJS, etc).
- Faltan comentarios para explicar la expresión regular del método verificarCorreo de la clase Verificador.
- Algunos nombres de métodos son singulares y otros plurales, incluso dentro de la misma clase.

Customer Relationship Manager

 Hay comentarios con código residual, principalmente en los métodos Page_Load de cada página y en la clase ResgistroRespuestas en el método registrarRespuesta.

 Siguen existiendo nombres en español y otros en inglés, por ejemplo el método setCommandText de la clase Conexion.

4. Observaciones de la implementación

- Hay codigo muy parecido entre metodos que podría agruparse en uno mismo. Esto es muy evidente en los metodos propuestaVentaXcontactoJS de las clases ContactoEmpresa y ContactoPersona.
- Hay un error en el contenido del mensaje de error del método mostrarMensaje de la clase Verificador. El mensaje dice textualmente: "El servicio no está disponible en este, inténtelo de nuevo más tarde", cuando debería decir "El servicio no está disponible en este momento, inténtelo de nuevo más tarde".
- La implementación de la inyección de vista de ventas y propuestas por contacto es muy mejorable.
- Las clases de prueba, en la carpeta Test, y la clase FakeConexion, deberían de ser separadas del proyecto principal y probadas en un proyecto aparte dedicado únicamente a ejecutar pruebas del sistema.

5. Recomendaciones y acciones a tomar

- Alinear todos los nombres de métodos con el estándar, o sea, cambiarlos para que estos empiecen todos con letras minúsculas y sean singulares.
 Igualmente intentar hacer más significativos los métodos mencionados y escribir todos en un mismo idioma, ya sea español o inglés.
- Eliminar los comentarios con código residual para evitar confusiones al mantener el código.
- Tratar de reducir la cantidad de código donde sea posible. Esto se puede lograr parametrizando funciones como propuestaVentaXcontactoJS, propuestaVentaXcontacto,

 ventaXcontacto,

Reporte de auditoría de código

Customer Relationship Manager

crearVistaPropuestaVentaXcontacto y crearVistaVentaXcontacto. Todas estas funciones realizan cosas similares. Si se parametrizan para reutilizar su funcionalidad con otros mensajes, se puede reducir significativamente el código y sus respectivas pruebas.

- Corregir el mensaje del método mostrarMensaje de la clase Verificador.
- Aislar las pruebas y todo lo relacionado a estas en un proyecto distinto.
- Se sugiere cambiar el patrón de diseño. Debido a las ventajas que ofrece el framework, ASP.Net, se sugirió considerar la idea de reimaginar el sistema bajo las estructura MVC. Esto para ayudar a reducir la redundancia de código y aprovechar al máximo las funciones de la herramienta de desarrollo. También el cambio tendría la intención de reducir la cantidad de pruebas automatizadas necesarias para cubrir la mayoría del código.