



UNIVERSIDAD DE SONORA
"El Saber de mis Hijos hará mi Grandeza"

INGENIERIA EN SISTEMAS DE INFORMACION

Reporte de prácticas profesionales:

“Desarrollo de un Sistema para el manejo de Recursos Humanos, Ventas y Marketing por medio de la plataforma de colaboración Empresarial Microsoft SharePoint”

Empresa: Tiempo Development

Alumno: Juan Manuel Rodríguez Castro

Expediente: 206200822

09 de Abril del 2012

Índice de Contenido

Introducción	2
Descripción del área de la institución en la que se desarrollo la practica	3
ANTECEDENTES HISTÓRICOS	3
ASPECTOS GENERALES.....	3
POLÍTICAS Y REGLAS.....	4
Justificación del proyecto realizado.....	5
Objetivos del proyecto	5
Problemas planteados para resolverlos	6
Alcances y limitaciones en la solución de los problemas	6
Fundamento teórico de las herramientas y conocimientos aplicados	7
Procedimientos empleados y actividades desarrolladas	15
METODOLOGÍA Y/O ESTRATEGIAS DE SOLUCIÓN	15
PLANIFICACIÓN DEL PROYECTO DE ESTADÍA	19
DISEÑO DE LA SOLUCIÓN	21
APLICACIÓN DE LA SOLUCIÓN	23
Resultados obtenidos.....	27
Conclusiones y recomendaciones.....	28
Anexos	29
Bibliografía	30

Introducción

En el mundo actual, el ambiente empresarial cambia constantemente. Vemos que las empresas deben operar a un ritmo cada vez más apresurado y la toma de decisiones definitivas es un tema frecuente.

En la actualidad, las empresas no pueden darse el lujo de quedarse obsoletas ante la competencia. Deben buscar constantemente el cambio y la renovación que les impulse a seguir evolucionando para ofrecer a sus clientes el mejor servicio. Y una empresa no puede cumplir este objetivo si para empezar sus procesos internos no están optimizados.

Sharepoint ofrece una alternativa de administración que se adapta en gran medida al proceso de gestión de recursos humanos en la empresa Tiempo Development. La mejora en el proceso que ofrece la plataforma es muy significativa y representa para la compañía un ahorro en tiempos, espacio y costos.

Sin embargo, aun cuando la plataforma ofrece un gran número de funciones muy útiles para la compañía, es necesario desarrollar herramientas que la complementen y así, obtener un sistema que sobrepase las expectativas funcionales del cliente.

Descripción del área de la institución en la que se desarrollo la practica

Antecedentes históricos

Tiempo Development fue fundada en 2005 por su presidente y CEO Cliff Schertz. Ofrece outsourcing de desarrollo de software de alta calidad y servicios de pruebas de software a sus clientes en los Estados Unidos. Su centro de operaciones se encuentra en Phoenix, Arizona; y su principal centro de desarrollo se encuentra en Hermosillo, Sonora, México.

Tiempo Development combina la efectividad del desarrollo offshore con la cercanía con los Estados Unidos. La compañía opera en el mismo huso horario de sus clientes, de modo que sus desarrolladores pueden tener mejor comunicación en tiempo real con ellos. Sus equipos de trabajo utilizan metodologías de desarrollo ágil y procesos de calidad medible para reducir los ciclos de desarrollo.

El centro de desarrollo de Tiempo Development se encuentra en Hermosillo, una moderna ciudad ubicada en el centro del estado de Sonora, a 280 kilómetros de distancia de Phoenix (a una hora de vuelo) y en el mismo huso horario.

Aspectos Generales

Giro: Desarrollo de software

Razón Social: Software Associates of Mexico S de RL de CV

Dirección: Blvd. García Morales #545. Hermosillo, Sonora, México

Pagina Web: <http://www.tiempodevelopment.com>

Políticas y reglas

Asistencia

- Permisos para faltas o ausencias deberán solicitarse con anticipación.
- El horario exige cumplir con 9 horas de trabajo diarias, pudiendo cumplirlas entre las 7 y las 19 horas y con al menos media hora para comer, que deberá ser entre las 12 y las 14 horas. El personal debe asegurarse de encontrarse en las instalaciones de la empresa durante las horas principales de trabajo, comprendidas entre las 9 y las 12 horas, y entre 14 y 17 horas.

Sistemas

- Los equipos de cómputo pertenecen a la compañía. Por ningún motivo es permitido sacar de las instalaciones de la empresa equipo de computo o mobiliario.
- Las estaciones de trabajo no deben estar sin atención por más de diez minutos.
- Queda estrictamente prohibido cualquier tipo de descarga o stream que pueda disminuir el desempeño de la red interna.
- No se permite la instalación de programas “piratas” o con cracks.
- No se permite el uso de memorias USB o discos duros externos.

Justificación del proyecto realizado

Actualmente, el departamento de Recursos Humanos lleva un control manual de la información de todos los empleados. Para realizar esta tarea, se requiere de un gran esfuerzo por parte del administrador del departamento, lo cual consume mucho tiempo. Se requiere de una plataforma Web en la que el administrador del departamento pueda llevar de manera electrónica una gestión detallada y organizada y con la posibilidad de obtener reportes. De esta manera, se busca reducir tiempos y eliminar la desorganización que llega a presentarse.

La implementación de esta plataforma tendrá un impacto directo e inmediato en el área de gestión de recursos humanos, dado que, en la actualidad este proceso es obsoleto y carece de automatización.

La adopción de un sistema Web para la administración de recursos humanos representa un gran avance en cuanto a desarrollo interno, lo cual permite obtener nuevas prioridades laborales y mejora de otros procesos administrativos.

Objetivos del proyecto

Objetivo General del proyecto:

Personalizar una aplicación que permita al departamento de Recursos Humanos de la empresa Tiempo Development administrar información general de personal y generar reportes con esa información.

Objetivos Particulares del proyecto:

Realizar personalizaciones requeridas por el cliente al sistema Sharepoint para adaptarlo a sus necesidades.

Problemas planteados para resolverlos

- El proceso del manejo de los recursos humanos en la empresa involucra mucho trabajo y consumo de tiempo.
- No se tiene acceso a la información fácilmente.
- Se requiere que el sistema pueda ser accedido desde fuera de la red interna.

Alcances y limitaciones en la solución de los problemas

Para la solución de los problemas, Tiempo Development opto por elegir el sistema Microsoft SharePoint.

El proyecto consiste en la modificación de SharePoint y esta sujeto a las limitaciones del mismo; además, ciertas modificaciones y codificación que se llevara acabo dependen de permisos adicionales (acceso a red, servidores internos, información delicada).

También tenemos el factor limitante del tiempo, cada Sprint debe ser realizado en tiempo y forma, sino el proyecto podría verse afectado; adicionalmente, algunos permisos de acceso requieren ser aprobados.

El alcance que tiene el proyecto es de gran impacto para la empresa; manejando el registro de los empleados por medio de Microsoft Sharepoint, se pueden hacer modificaciones adicionales para llevar el control de horas laboradas de los empleados, del tiempo con exactitud laborando en ciertos proyectos o cierto nivel de desarrollador, etc.

Adicionalmente siendo SharePoint tan flexible, otras áreas de la empresa pueden utilizarlo para desempeñar y automatizar otras actividades propias su área.

Fundamento teórico de las herramientas y conocimientos aplicados

Recursos Humanos:

En la administración de empresas, se denomina recursos humanos (RRHH) al trabajo que aporta el conjunto de los empleados o colaboradores de esa organización. Pero lo más frecuente es llamar así a la función o gestión que se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización. Estas tareas las puede desempeñar una persona o departamento en concreto (los profesionales en Recursos Humanos) junto a los directivos de la organización.

El objetivo básico que persigue la función de Recursos Humanos con estas tareas es alinear las políticas de RRHH con la estrategia de la organización, lo que permitirá implantar la estrategia a través de las personas, quienes son consideradas como los únicos recursos vivos e inteligentes capaces de llevar al éxito organizacional y enfrentar los desafíos que hoy en día se percibe en la fuerte competencia mundial. Es imprescindible resaltar que no se administran personas ni recursos humanos, sino que se administra con las personas viéndolas como agentes activos y proactivos dotados de inteligencia, creatividad y habilidades intelectuales.

Generalmente la función de Recursos Humanos está compuesta por áreas tales como reclutamiento y selección, contratación, capacitación, inducción de personal y su permanencia en la empresa. Dependiendo de la empresa o institución donde la función de Recursos Humanos opere, pueden existir otros grupos que desempeñen distintas responsabilidades que pueden tener que ver con aspectos tales como la administración de la nómina de los empleados o el manejo de las relaciones con sindicatos, entre otros. Para poder ejecutar la estrategia de la organización es fundamental la administración de los Recursos humanos, para lo cual se deben considerar conceptos tales como la comunicación organizacional, el liderazgo, el trabajo en equipo, la negociación y la cultura organizacional.[1]

Microsoft .NET Framework:

.NET Framework es un componente integral de Windows que admite la compilación y la ejecución de la siguiente generación de aplicaciones y servicios Web XML. El diseño de .NET Framework está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.
- Ofrecer un entorno de ejecución de código que promueva la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.
- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan scripts o intérpretes de comandos.
- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en el Web.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.

.NET Framework contiene dos componentes principales: Common Language Runtime y la biblioteca de clases de .NET Framework. Common Language Runtime es el fundamento de .NET Framework. El motor en tiempo de ejecución se puede considerar como un agente que administra el código en tiempo de ejecución y proporciona servicios centrales, como la administración de memoria, la administración de subprocesos y la comunicación remota, al tiempo que aplica una seguridad estricta a los tipos y otras formas de especificación del código que promueven su seguridad y solidez. De hecho, el concepto de administración de código es un principio básico del motor en tiempo de ejecución. El código

destinado al motor en tiempo de ejecución se denomina código administrado, a diferencia del resto de código, que se conoce como código no administrado. La biblioteca de clases, el otro componente principal de .NET Framework, es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario (GUI) o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los formularios Web Forms y los servicios Web XML. [2]

Microsoft Sharepoint 2010:

SharePoint 2010 es la plataforma de colaboración empresarial que le permite incrementar la productividad y administrar los contenidos a través de la conocida interfaz de Office. Tanto si implementa SharePoint 2010 de forma local como si lo hace en servicios hospedados, los recursos integrados, mejorados por las tecnologías de búsqueda, le permitirán responder con rapidez frente a las cambiantes necesidades empresariales. Tome decisiones bien informadas e implemente soluciones personalizadas de forma rápida y segura para mejorar la colaboración en toda la empresa y con terceros. La consolidación de soluciones mediante SharePoint 2010 contribuye a ahorrar mediante la reducción de los gastos de mantenimiento y formación y a través del aumento de la productividad del departamento de TI. Todo gracias a una plataforma que ofrece un control total y además cumple con las normativas. [3]

Microsoft InfoPath 2010:

Microsoft InfoPath 2010 es una herramienta para crear formularios y recopilar datos que ayuda a las organizaciones a simplificar los procesos empresariales. InfoPath 2010 está pensada para usuarios profesionales avanzados y para programadores, según el tipo de solución basada en formularios que la organización necesite.

Los usuarios profesionales avanzados pueden usar InfoPath 2010 sin la necesidad de escribir código para diseñar formularios electrónicos complejos que permitan recopilar información de forma rápida y rentable.

Los programadores pueden crear formularios avanzados para procesos de negocio departamentales y empresariales (incluidas aplicaciones compuestas y secuencias de flujo de trabajo) con InfoPath 2010, Microsoft SharePoint Server 2010 y SharePoint Designer 2010, sin apenas recurrir al uso de código o no usándolo en absoluto. [4]

Microsoft C#:

C# es un lenguaje orientado a objetos elegante y con seguridad de tipos que permite a los desarrolladores compilar diversas aplicaciones sólidas y seguras que se ejecutan en .NET Framework. Se puede utilizar C# para crear aplicaciones cliente de Windows tradicionales, servicios Web XML, componentes distribuidos, aplicaciones cliente-servidor, aplicaciones de base de datos, y mucho, mucho más. Visual C# 2010 proporciona un editor de código avanzado, cómodos diseñadores de interfaz de usuario, depurador integrado y numerosas herramientas más para facilitar el desarrollo de aplicaciones basadas en la versión 4.0 del lenguaje C# y la versión 4 de .NET Framework.[5]

CSS:

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML2 (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>". [6]

AJAX:

Ajax, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado (scripting language) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y DocumentObjectModel (DOM). [7]

XAML:

XAML (acrónimo pronunciado xammel del inglés eXtensible Application Markup Language, Lenguaje Extensible de Formato para Aplicaciones en español) es el lenguaje de formato para la interfaz de usuario para la Base de Presentación de Windows (WPF por sus siglas en inglés) y Silverlight (wpf/e), el cual es uno de los

"pilares" de la interfaz de programación de aplicaciones .NET en su versión 3.0 (conocida con anterioridad con el nombre clave WinFX).

XAML es un lenguaje declarativo basado en XML, optimizado para describir gráficamente interfaces de usuarios visuales ricas desde el punto de vista gráfico, tales como las creadas por medio de Adobe Flash. XUL y UIML son otros ejemplos de lenguajes de interfaz basados en XML. SVG es un estándar de la organización W3C, el cual soporta gráficos, animaciones, audio y video integrados, eventos y comportamiento descrito por medio de escritura y puede ser utilizado como lenguaje de interfaz basado en XML.

En su uso típico, los archivos tipo XAML serían producidos por una herramienta de diseño visual, como Microsoft Visual Studio o Microsoft Expression. El XML resultante es interpretado en forma instantánea por un sub-sistema de despliegue de Windows Vista que reemplaza al GDI de las versiones anteriores de Windows. Los elementos de XAML se interconectan con objetos del Entorno Común de Ejecución para Lenguajes. Los atributos se conectan con propiedades o eventos de esos objetos.

XAML fue diseñado para soportar las clases y métodos de la plataforma de desarrollo .NET que tienen relación con la interacción con el usuario, en especial el despliegue en pantalla. El acrónimo XAML originalmente significaba Extensible Avalon Markup Language, Lenguaje Extensible de Formato de Avalon; habiendo sido Avalon el nombre clave original de la Base de Presentación de Windows, nombre que engloba a este grupo de clases de .NET.[8]

XML:

XML, siglas en inglés de eXtensible Markup Language ('lenguaje de marcas extensible'), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML). Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para

diferentes necesidades, de ahí que se le denomine metalenguaje.¹ Algunos de estos lenguajes que usan XML para su definición son XHTML, SVG, MathML.

XML no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil. [9]

JavaScript:

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, ³ basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador...⁴ aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML. [10]

Procedimientos empleados y actividades desarrolladas

Metodología y/o estrategias de solución

La empresa Tiempo Development basa sus actividades en las buenas prácticas de CMMI y Scrum, que consiste en un modelo de desarrollo ágil.

Siguiendo esta metodología, el proyecto consiste en varias iteraciones o Sprints, en los cuales se trabaja el análisis, diseño, codificación, pruebas e implementación. Cabe mencionar que este es un proyecto que, al momento de iniciar la estadía profesional, ya había comenzado.

Documentación

Son muchos los documentos que se generan y actualizan en cada uno de los proyectos realizados por Tiempo Development. Como se ha mencionado, la compañía maneja una metodología de desarrollo ágil basada en Scrum y apoyada en las buenas prácticas de CMMI, lo cual crea documentos por ambos modelos.

En la siguiente sección se enlistan y describen solo los documentos en los que el alumno estuvo involucrado, ya sea en su creación o actualización durante el desarrollo del proyecto.

- **Product Backlog**

Al iniciar el proyecto, se construye una lista de requerimientos priorizados proporcionados por el cliente, llamada Product Backlog. Algunas características de este documento son que contiene información detallada de cada uno de los requisitos del sistema, puede ser editado por cualquier miembro del equipo, contiene estimaciones de tiempo, entre otras.

- **Cronograma de actividades**

Se desarrolla un cronograma de actividades para cada iteración, detallando la duración de cada actividad, fechas de inicio y fin, los recursos asignados a cada tarea, etc.

- Casos de uso

Se desarrollan casos de uso de la aplicación para analizar los escenarios que pudieran presentarse al usar el sistema. Se muestra en un flujo descendente en texto los escenarios que podría provocar el usuario.

- Guía de migración de la aplicación

En este documento se plasman todos los cambios hechos al sistema original Sharepoint. Se detalla que archivos se modificaron para cubrir cada requerimiento. Se anotan también las adiciones necesarias a la base de datos para el correcto funcionamiento de las nuevas personalizaciones.

Cabe mencionar que este documento no forma parte de Scrum, sino que el equipo decidió que sería de mucha utilidad tener un documento que facilite la modificación de los requerimientos desarrollados al momento.

Juntas

Para la realización de este proyecto, y siguiendo las directivas que marca Scrum (y la metodología interna de Tiempo Development), se manejan principalmente los cuatro siguientes tipos de reuniones del equipo y con el cliente. Cada una tiene un propósito en particular y como se menciona, el cliente está presente en algunas de ellas.

- Junta de Estimación de Requerimientos

En esta junta se analizan los posibles tiempos que tomara el desarrollo de los requerimientos restantes en la lista de requerimientos entregada por el cliente (Product Backlog).

- Junta de planeación de Sprint

Se lleva a cabo una junta para planear el Sprint, en la que se señalan los requerimientos a desarrollar en esta iteración y los tiempos que tomara llevarlos a cabo.

- Junta de planeación del Sprint 1

En esta junta se analizan a detalle los requerimientos elegidos para ser desarrollados en esta iteración.

- Junta de planeación del Sprint 2

Se analiza el posible impacto de los requerimientos a realizar sobre el sistema actual. Se identifican los posibles riesgos que pueden presentarse al desarrollar estos requerimientos.

- Junta diaria y semanal

Se tiene una junta diaria en la que se da a conocer de manera breve el estado actual de la fase en desarrollo (ya sea actualización de la documentación, codificación, pruebas, etc.). Se tiene también una junta semanal para revisar el avance alcanzado en esa semana de trabajo. En estas juntas se requiere la asistencia del cliente.

- Junta de revisión del Sprint

Se tiene una junta al final del Sprint para revisar el trabajo completado y el que no pudo ser concluido y dar una demostración funcional al cliente.

- Junta retrospectiva

Se revisa y comenta qué salió bien y qué salió mal durante el Sprint, y qué podemos hacer diferente para evitar dichos problemas en la siguiente iteración.

Codificación

Idealmente, la codificación de los requerimientos se lleva a cabo solamente durante las fechas planeadas. Sin embargo, esta fase puede sufrir un retraso debido a imprevistos. Se usa un estándar interno de escritura de código que incluye buenas prácticas que involucran la correcta nomenclatura y definición de variables, funciones y clases, entre otros aspectos.

Pruebas e implementación

La implementación va de la mano con las etapas de prueba. En Tiempo Development hay 4 etapas en las que la aplicación es probada de varias maneras y en diferentes ambientes. Al final se obtiene una funcionalidad libre de errores y que puede ser entregada al cliente.

La primera implementación se hace hacia un ambiente de pruebas, al haber terminado y revisado el código. Es ahí donde el equipo de pruebas puede llevar a cabo el llamado Smoke Testing, una fase en la que se verifica únicamente que la aplicación no falle catastróficamente. En este caso, una aplicación web, se revisa que los links presentes en la página funcionen y se dirijan a donde deben, con el fin de confirmar que la o las nuevas funcionalidades no afectaron a las anteriores.

Si se presentan errores en esta fase, el equipo de desarrollo es notificado y corrige los errores. Al hacer las reparaciones, pone el sitio nuevamente en el ambiente de pruebas.

Al comprobar que el sistema no falla, se comienza la fase de pruebas funcionales, en las que se revisa el correcto funcionamiento de los requerimientos desarrollados. Los errores que aparecen en esta etapa son devueltos al equipo desarrollador para su reparación.

Al concluir las reparaciones, la aplicación es implementada una vez más en el ambiente de pruebas. Ahora el equipo de pruebas inicia las pruebas de regresión, donde se prueba que la corrección de errores fue satisfactoria y que estas reparaciones no afectaron al funcionamiento general de la aplicación.

Finalmente, cuando se ha revisado que la aplicación no presenta errores funcionales en el ambiente de pruebas, la aplicación esta lista para ser instalada en el ambiente de producción. Ya en este ambiente el equipo de pruebas realiza nuevamente Smoke Testing. Si llegan a aparecer errores (idealmente nulos y relacionados al ambiente, no al funcionamiento de la aplicación), el equipo de desarrollo los repara.

Al finalizar exitosamente las etapas de prueba, y una vez que el sistema ha sido implementado en el ambiente de producción de manera satisfactoria sin que presente errores, la aplicación esta lista para ser entregada al cliente y hacerle una demostración funcional.

Planificación del proyecto de estadía

Como se ha mencionado con anterioridad, en Tiempo Development se ha implementado un modelo de desarrollo ágil propio basado en Scrum, que a su vez establece ciclos de trabajo (llamados Sprints) que duran de 2 a 4 semanas, dependiendo de la magnitud de cada proyecto. Durante la realización de este proyecto de estadía, se espera que el estudiante tenga parte en al menos tres Sprints de desarrollo del proyecto.

Generalmente en cada Sprint se maneja la elaboración o actualización de documentos referentes al proyecto (casos de uso y guía de migración de la aplicación), codificación, corrección de errores e implementación.

Sprint 4

Para este Sprint se realizarán cinco actividades de las cuales, tres están relacionadas al puesto en el que se desempeña un empleado; una corresponde al registro de personas dependientes del empleado y la última apunta a la información de Infonavit de cada empleado.

Las actividades referentes al puesto del empleado requieren la adición de un lookup (Control de Sharepoint para una relación entre listas) para elegir entre diferentes tipos de contrato, la implementación de un control binario que indique si un empleado se encuentra actualmente activo o inactivo y el registro y presentación de los cambios realizados en este dato.

Pasando a las actividades referentes a las personas dependientes, se necesitará del mismo modo un control binario para mostrar si uno de ellos se encuentra actualmente asegurado por el empleado.

Por último, se requiere de un módulo para guardar y modificar la información de Infonavit (número y monto de crédito) de cada uno de los empleados.

Sprint 5

La actividad fuerte de este Sprint consistirá en la elaboración de un formulario de terminación de contrato del empleado. La funcionalidad contendrá varios campos que solicitan y presentan datos que corresponden a la fecha en que un empleado fue despedido, un campo que informa al administrador de Recursos Humanos si el empleado colaboró con la compañía y finalmente un campo para agregar comentarios a la terminación del empleado.

Además el cliente ha requerido modificar una funcionalidad agregada en el Sprint pasado. Se trata de agregar la posibilidad de modificar las fechas en que un empleado cambia de status entre Activo e Inactivo.

Sprint 6

Para este ciclo de trabajo se realizarán cuatro actividades, dos de ellas relacionadas con el salario del empleado y las restantes se relacionan con el contrato que tiene cada empleado.

Primeramente se necesitará crear un historial de cambios en el salario de cada empleado, agregando una lista similar a la añadida en Sprints pasados. El siguiente requerimiento irá muy de la mano con este, dado que se trata de agregar un campo para incluir comentarios en el momento de hacer cambios en el salario.

Después, se agregarán dos campos para almacenar las fechas en que inicia y termina el contrato de cada empleado.

Diseño de la solución

El proyecto de esta etapa consiste en realizar numerosas personalizaciones solicitadas por el cliente al ambiente de Sharepoint. La aplicación, llamada Sharepoint, es una aplicación web desarrollada por Microsoft, con una base de datos desarrollada en Microsoft SQL Server, un sistema de administración de bases de datos de nivel empresarial y múltiples subprocesos. El sistema corre en un servidor interno que contiene los módulos requeridos para el correcto funcionamiento del mismo (servidor IIS y Microsoft SQL Server 2008 R2 instalados).

Se le agregarán las siguientes funcionalidades a la aplicación (separados por Sprint):

Sprint 4

- Un campo en la sección de Job (Empleo) para elegir entre los diferentes tipos de contratos disponibles, por ejemplo, tiempo completo, medio tiempo, etc.

- Un campo CheckBox (selección) en la sección de Job (empleo) para diferenciar si el empleado se encuentra actualmente activo o inactivo.
- Agregar la funcionalidad para guardar los cambios de status activo e inactivo de los empleados.
- Un campo checkbox (selección) en la selección de Dependents para diferenciar si las personas dependientes del empleado se encuentran aseguradas por él.
- Agregar un nuevo apartado para almacenar y mostrar los datos referentes al crédito hipotecario de cada empleado.

Sprint 5

- Agregar una forma de terminación para cada empleado, que contiene los siguientes campos:
 - Fecha de terminación.
 - Campo de selección para indicar si el empleado es, o no, elegible para recontractación.
 - Total de meses que el empleado laboro en la empresa.
 - Comentarios.
- Agregar la función para poder cambiar las fechas en el historial de actividad/inactividad.

Sprint 6

- Agregar la funcionalidad para guardar los cambios hechos e el salario de cada empleado.
 - Este requisito contiene también un campo para agregar comentarios para justificar el cambio de salario.
- Agregar dos campos para indicar cuándo empezó el contrato de un empleado y cuando termina el mismo.

Aplicación de la solución

Tras haber realizado un análisis y un posterior diseño de los requerimientos a elaborar, las funcionalidades requeridas por el cliente fueron adaptadas al sistema de manera satisfactoria, dando así al cliente módulos funcionales en los que puede trabajar.

Entrada al sistema y su estructura

El sistema requiere de un usuario y contraseña para poder acceder a los registros de los empleados

Al acceder satisfactoriamente, el sistema muestra un menú en la parte superior con muchas opciones, como lo es la administración del sitio (ADMIN), la lista de empleados (PIM) y otras herramientas útiles.

En la parte media, el sistema presenta un buscador de empleados con varias opciones de búsqueda. Finalmente en la parte inferior, el sistema muestra de entrada una lista paginada con todos los usuarios registrados. Es en esta lista donde elegimos a un usuario para empezar a trabajar con él.

Al seleccionar a un empleado, el sistema muestra su información personal en la sección central. A la izquierda muestra un menú con todas las secciones que contienen la información del empleado, tales como información de su puesto (sección Job), información de su salario (sección Salary), entre otras más.

Requerimientos desarrollados

Como se describió en el diseño de la solución, se desarrollaron varios requerimientos del cliente durante tres Sprints.

Para acceder a ellos, es necesario estar identificado en la aplicación y seleccionar un empleado de la lista.

Sprint 4

- Agregar un menú desplegable (dropdown list) para seleccionar el tipo de contrato que el empleado tiene.
- Agregar un campo para saber si el empleado se encuentra Activo o Inactivo.
- Agregar la funcionalidad para guardar y mostrar los cambios hechos en el campo de Status activo o inactivo.

Estos cuatro requerimientos fueron agregados en la sección Empleo (Job) del sistema. Para entrar a esta sección, es necesario estar identificado en el sistema, elegir un empleado y dar clic en la sección Empleo (Job) del menú izquierdo.

- Agregar un campo para indicar si un dependiente está asegurado.

Este requerimiento fue incluido en la sección Dependents (Dependientes) del sistema. Para entrar a esta sección, es necesario estar identificado en el sistema, elegir un empleado y dar clic en la sección Dependents del menú izquierdo.

Sprint 5

- Agregar un formulario de terminación.
- Agregar fecha de término en la forma de término.
- Agregar un campo que indique si el empleado es elegible para recontractación.
- Agregar total de meses.
- Agregar comentarios en la forma de terminación.

La forma de terminación de los empleados aparece en la sección de Empleo (Job), al elegir el tipo de contrato Terminated (Terminado) del campo de selección agregado en el Sprint pasado. El nuevo formulario aparece debajo de la sección normal y contiene todos estos campos. Para entrar a esta sección, es necesario estar identificado en el sistema, elegir un empleado y dar clic en la sección Job (Empleo) del menú izquierdo.

- Agregar la función para poder cambiar las fechas en el historial de actividad/inactividad.

Este requerimiento consiste en una actualización al desarrollo en el Sprint pasado. Se trata de añadir la posibilidad de cambiar las fechas de los cambios registrados en el campo Activo o Inactivo, ya que el requerimiento anterior salva los cambios con la fecha actual. Esta función, al igual que la anterior, se encuentra en la sección de Job (Empleo). Se agregó una liga en cada uno de los cambios de status generados hasta el momento, que permite cambiar las fechas.

Sprint 6

- Agregar la funcionalidad para guardar y desplegar todos los cambios hechos en el campo de Salary (Salario), incluyendo las fechas en que se hicieron.
- Agregar un campo de comentarios para justificar el cambio de salario.

Estos dos requerimientos fueron incluidos en la sección de Salary (Salario). El primero es muy similar al realizado en el cuarto Sprint y consiste en almacenar los cambios hechos en el salario de cada empleado. El campo de comentarios va junto con este requerimiento y se ubican en la misma sección.

Este apartado presenta una lista de todos los cambios que se han hecho en el salario.

- Agregar un campo que indique la fecha en que inicio el contrato de un empleado.
- Agregar un campo que indique la fecha en que el contrato de un empleado debe terminar.

Estos dos requerimientos fueron agregados en la sección de Job (Empleo). Indican las fechas de inicio y fin del contrato del empleado.

Resultados obtenidos

Siendo una empresa líder dentro de la industria del desarrollo de software, Tiempo Development se rige bajo el modelo de calidad CMMI y sigue al modelo de desarrollo ágil Scrum. Esto implica que los productos realizados en la empresa deben tener un alto grado en su calidad.

De esta manera, cada proyecto cuenta con un equipo de pruebas, que realiza un análisis exhaustivo del funcionamiento de los nuevos requerimientos desarrollados. Este proceso le garantiza al cliente que el producto terminado tiene calidad y le da confianza en que el producto final habrá satisfecho sus necesidades al final de cada iteración.

El proyecto realizado durante la estadía no fue la excepción. El desarrollo estuvo todo el tiempo sujeto a la atención de un equipo de pruebas, así como también de asesoramiento por parte de un equipo de profesionales. El desarrollo de los requerimientos tuvo el resultado esperado, ya que el equipo de pruebas no solo fue capaz de aprobar el proyecto y considerarlo libre de errores, sino que pudo establecer que el proyecto cuenta con calidad y está dirigido a ella.

Con la personalización de Sharepoint, la empresa Tiempo Development se ha visto beneficiada en muchos aspectos, en gran medida por el mejoramiento del proceso del manejo de los recursos humanos. Esta implementación le ha permitido al cliente trabajar en los módulos disponibles y contar en todo momento con la información de los recursos laborando actualmente en la empresa, sin mencionar los útiles reportes que el sistema genera para obtener información detallada.

Cabe mencionar que en el área de desarrollo de productos internos se dio un gran salto durante el periodo de estadía, ya que se logro terminar e implementar un gran número de requerimientos establecidos por el cliente al inicio del proyecto.

Esto sin duda refleja un avance en el proyecto.

Conclusiones y recomendaciones

Conclusiones de ámbito personal

En la actualidad, el uso de las tecnologías de información en las empresas es de vital importancia para su desarrollo organizacional. La personalización de Microsoft SharePoint le permitirá al cliente expandir sus posibilidades en cuanto a colaboración empresarial, además de ayudar a mantener los registros de manera ordenada y centralizada en una plataforma web a la que se puede acceder desde cualquier terminal en la red interna y externa (con la debida configuración).

La implementación de esta plataforma permitirá al cliente reducir tiempos en la administración de los recursos humanos para poder ocuparlo en otras actividades. El avance que se obtuvo en el proyecto fue por demás satisfactorio. Se logro terminar un gran número de requerimientos en un periodo relativamente corto de tiempo.

Conclusiones de ámbito profesional

Fue un excelente trecho de aprendizaje el curso pasado, con un sinfín de conocimiento obtenido, lecciones aprendidas, nuevas personas conocidas y sin duda, todo lo aprendido en un ambiente de trabajo totalmente nuevo. Desde el horario de trabajo, hasta las políticas internas y los modelos de calidad, se puede decir con toda seguridad que me han dejado un gran crecimiento personal.

Estoy seguro que habrá mas alumnos que formen parte de una de las compañías de desarrollo de software más importantes del estado, con esfuerzo y constancia se pueden obtener muchos conocimientos y la posibilidad de laborar en esta empresa.

Anexos

Análisis FODA

Fortalezas y Oportunidades experimentadas:

Las Fortalezas y Oportunidades notorias, serian los conocimientos de la terminología relacionada a nuestra área, se nos enseñaron términos como: CMMI, SRUM, POO (encapsulamiento, sobrecarga, herencia, etc.).

Estos fueron muy importantes al momento de la selección de los practicantes.

Los conocimientos adquiridos con Java, indirectamente me fueron útiles, esto debido a que muchos comandos y lógica de este lenguaje también se manejan o tienen mucha similitud a los que tiene el lenguaje de programación C# (el utilizado durante el proyecto).

Debilidades y Amenazas experimentadas:

Durante este proyecto se hizo notoria la gran falta de práctica relacionada con la programación real que se hace en una empresa, y esto es muy importante y critico para futuras posiciones laborales.

Algunas de las cuales serian:

- Falta de práctica con problemas de desarrollo reales.
- Falta de práctica en trabajos en equipo y a gran escala.
- Falta de conocimientos en: .NET, SQL Server, IIS, Desarrollo Web (JavaScript, AJAX, etc.).
- Falta de conocimientos y práctica en Testing.
- Falta de conocimientos y práctica en “Debug” (método para encontrar errores).
- Falta de conocimientos y práctica con servidores.

Bibliografía

- [1] Wikipedia.org: Recursos Humanos. http://es.wikipedia.org/wiki/Recursos_humanos (9 de Abril del 2012).
- [2] Microsoft.com: Microsoft .NET framework. <http://msdn.microsoft.com/es-mx/library/zw4w595w.aspx> (9 de Abril del 2012).
- [3] Microsoft.com: Microsoft SharePoint 2010. <http://sharepoint.microsoft.com/es-mx/Paginas/default.aspx> (9 de Abril del 2012).
- [4] Microsoft.com: Microsoft InfoPath 2010. <http://office.microsoft.com/es-hn/infopath/> (9 de Abril del 2012).
- [5] Microsoft.com: Microsoft C#. <http://msdn.microsoft.com/library/z1zx9t92> (9 de Abril del 2012).
- [6] Wikipedia.org: CSS. http://es.wikipedia.org/wiki/Cascading_Style_Sheets (9 de Abril del 2012).
- [7] Wikipedia.org: AJAX. [http://en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming)) (9 de Abril del 2012).
- [8] Wikipedia.org: XAML. <http://es.wikipedia.org/wiki/XAML> (9 de Abril del 2012).
- [9] Wikipedia.org: XML. <http://es.wikipedia.org/wiki/XML> (9 de Abril del 2012).
- [10] Wikipedia.org: JavaScript. <http://es.wikipedia.org/wiki/JavaScript> (9 de Abril del 2012).