



UNIVERSIDAD TECNOLÓGICA EQUINOCCIAL

**FACULTAD DE CIENCIAS DE LA INGENIERÍA E
INDUSTRIAS**

**CARRERA DE INGENIERÍA INFORMÁTICA Y
CIENCIAS DE LA COMPUTACIÓN**

**DISEÑO E IMPLEMENTACIÓN DE UN SOFTWARE QUE
PERMITA EVALUAR EN BASE A PERFILES, HABILIDADES Y
CUALIDADES A LOS EQUIPOS DE DESARROLLO DE
SOFTWARE DE LA EMPRESA VIMEWORKS UBICADA EN LA
CIUDAD DE QUITO.**

**TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN**

LUIS SEBASTIÁN ARAUJO PARRA

DIRECTOR: ING. MOSCOSO ZEA OSWALDO VICENTE

Quito, Agosto, 2016

© Universidad Tecnológica Equinoccial. 2016
Reservados todos los derechos de reproducción

FORMULARIO DE REGISTRO BIBLIOGRÁFICO PROYECTO DE TITULACIÓN

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD:	1723723092
APELLIDO Y NOMBRES:	Araujo Parra Luis Sebastián
DIRECCIÓN:	Av. Naciones Unidas y 10 de Agosto
EMAIL:	sebasaraujo23@gmail.com
TELÉFONO FIJO:	022253174
TELÉFONO MOVIL:	0995842467

DATOS DE LA OBRA	
TÍTULO:	DISEÑO E IMPLEMENTACIÓN DE UN SOFTWARE QUE PERMITA EVALUAR EN BASE A PERFILES, HABILIDADES Y CUALIDADES A LOS EQUIPOS DE DESARROLLO DE SOFTWARE DE LA EMPRESA VIMEWORKS UBICADA EN LA CIUDAD DE QUITO
AUTOR O AUTORES:	Luis Sebastián Araujo Parra
FECHA DE ENTREGA DEL PROYECTO DE TITULACIÓN:	20 de Mayo de 2016
DIRECTOR DEL PROYECTO DE TITULACIÓN:	Ing. Moscoso Zea Oswaldo Vicente
PROGRAMA	PREGRADO <input checked="" type="checkbox"/> POSGRADO <input type="checkbox"/>
TÍTULO POR EL QUE OPTA:	
RESUMEN: Mínimo 250 palabras	El presente estudio se enfocó en el diseño e implementación de un software

para la evaluación de equipos de desarrollo, basados en habilidades, características, esfuerzo, etc. de cada miembro que forma parte de dichos equipos.

Esto es posible ya que se elaboró un diagrama de evaluación mediante preguntas divididas por secciones, estas secciones tienen un peso que ayuda a calcular un resultado general del empleado, teniendo en cuenta que los empleados son evaluados únicamente por miembros de su mismo equipo.

Para el desarrollo de este sistema se implementó la metodología Scrum, ya que según un análisis que lo podremos observar en los capítulos siguientes se determinó que es la metodología ideal para el desarrollo de este proyecto.

La ejecución y culminación del proyecto permitió reafirmar que Scrum se puede aplicar en proyectos de software de mediano tamaño, en entornos cambiantes, con grupos de trabajo pequeños los cuales involucran permanentemente al dueño del producto.

Se pudo obtener un software funcional de gran ayuda para la empresa Vimeworks, permitiendo a la administración tener un mayor control de sus equipos de desarrollo durante la ejecución de un proyecto, y de esta manera realizar una repartición de bonos o incentivos más “justa” con respecto al esfuerzo que realizó cada miembro del equipo.

PALABRAS CLAVES:

- Implementación de Software
- Evaluación de equipos de desarrollo
- Scrum
- Ejecución de proyecto

ABSTRACT:

This study focused on the design and development of a software implementation for evaluating development teams based on skills, characteristics, effort, etc. of each member forming part of such team.

This was possible because an evaluation chart was drawn by questions divided into sections. These sections had weights that helped to calculate an overall result of the employee, considering that employees are judged solely by members of the same team.

For the development of this system, the Scrum methodology was used, An analysis was made to select this methodology. This analysis is shown in the following chapters. It shows that SCRUM is ideal for the development of this project.

The execution and completion of this project reaffirmed that Scrum is a good choice which can be applied to software projects of medium size, in changing environments, with small workgroups who permanently involve the product owner.

A functional software was obtained to supports Vimeworks company. This

KEYWORDS	software allows management to have greater control of their development teams during project implementation, and thus make a more fairly distribution of bonuses and incentives with respect to the effort that was realized every by team member
	<ul style="list-style-type: none">• software implementation• evaluating development teams• Scrum• Project implementation

Se autoriza la publicación de este Proyecto de Titulación en el Repositorio Digital de la Institución.

f: 

ARAUJO PARRA LUIS SEBASTIÁN

1723723092

DECLARACIÓN Y AUTORIZACIÓN

Yo, **ARAUJO PARRA LUIS SEBASTIÁN**, CI 1723723092 autor/a del proyecto titulado: **Diseño e implementación de un software que permita evaluar en base a perfiles, habilidades y cualidades a los equipos de desarrollo de software de la empresa VIMEWORKS ubicada en la ciudad de Quito** previo a la obtención del título de **INGENIERO EN INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN** en la Universidad Tecnológica Equinoccial.

1. Declaro tener pleno conocimiento de la obligación que tienen las Instituciones de Educación Superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
2. Autorizo a la BIBLIOTECA de la Universidad Tecnológica Equinoccial a tener una copia del referido trabajo de graduación con el propósito de generar un Repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Quito, 5 de Junio de 2016

f: _____

ARAUJO PARRA LUIS SEBASTIÁN

1723723092



Quito, 9 de Junio de 2016

CARTA DE AUTORIZACIÓN

Yo, **MAURICIO XAVIER SANTACRUZ ESCOBAR** con cédula de identidad N° 0501604771 en calidad de Gerente General de VIMEWORKS CIA. LTDA. autorizo a **SEBASTIÁN ARAUJO**, realizar la investigación para la elaboración de su proyecto de titulación "DISEÑO E IMPLEMENTACIÓN DE UN SOFTWARE QUE PERMITA EVALUAR EN BASE A PERFILES, HABILIDADES Y CUALIDADES A LOS EQUIPOS DE DESARROLLO DE SOFTWARE DE LA EMPRESA VIMEWORKS CIA. LTDA. UBICADA EN LA CIUDAD DE QUITO", basada en la información proporcionada por la compañía.

f: _____

MAURICIO SANTACRUZ E.

C.I. 0501604771

DECLARACIÓN

Yo **Luis Sebastián Araujo Parra**, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Tecnológica Equinoccial puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.



Luis Sebastián Araujo Parra

C.I. 1723723092

CERTIFICACIÓN

Certifico que el presente trabajo que lleva por título “**Diseño de la implementación de un software que permita evaluar en base a perfiles, habilidades y cualidades a los equipos de desarrollo de software de la empresa VIMEWORKS, ubicada en la ciudad de Quito.**”, que, para aspirar al título de **Ingeniero en Informática y Ciencias de la Computación** fue desarrollado por **Sebastián Araujo**, bajo mi dirección y supervisión, en la Facultad de Ciencias de la Ingeniería e Industrias; y cumple con las condiciones requeridas por el reglamento de Trabajos de Titulación artículos 19, 27 y 28.



Oswaldo Moscoso

DIRECTOR DEL TRABAJO

C.I. 1709629651

INDICE DE CONTENIDOS

	PÁGINA
RESUMEN	xvii
ABSTRACT	xviii
INTRODUCCIÓN	1
1. INTRODUCCIÓN	1
MARCO TEORICO	4
2. MARCO TEORICO	5
2.1 SISTEMAS DE INFORMACIÓN	5
2.2 DEFINICIÓN DE APLICACIONES WEB	5
2.2.1 QUÉ ES UNA APLICACIÓN WEB	5
2.2.2 INTERFAZ.....	5
2.3 ESTRUCTURA DE LAS APLICACIONES WEB.....	6
2.4 HERRAMIENTAS Y PLATAFORMA DE DESARROLLO	7
2.4.1 SERVIDOR DE BASE DE DATOS.....	7
2.4.1.1 MySQL.....	7
2.4.2 JAVA	7
2.4.2.1 Java Enterprise Edition.....	9
2.4.3 SERVIDOR DE APLICACIONES	9
2.5 EVALUACIÓN DE DESEMPEÑO LABORAL	10
2.5.1 MÉTODO DE AUTOEVALUACIÓN	10
2.5.2 MÉTODO DE COMISIÓN DE EVALUACIÓN	11
2.5.3 MÉTODO DE ESCALA GRÁFICA	11
2.5.4 MÉTODO DE ELECCIÓN FORZADA.....	11
2.5.5 MÉTODO DE INVESTIGACIÓN DE CAMPO	12
2.6 ANÁLISIS DE METODOLOGÍA DE SOFTWARE.....	12
2.7 METODOLOGÍAS ÁGILES.....	13
2.7.1 EL MANIFIESTO ÁGIL.....	14
2.7.2 SCRUM	15
2.7.2.1 Beneficios	16
METODOLOGÍA	19
3. METODOLOGÍA.....	20

3.1	METODOS DE ESTUDIO.....	20
3.2	METODOLOGÍA DE EVALUACIÓN SISTEMA VIMEWORKS	20
3.3	SCRUM	21
3.3.1	ROLES	21
3.3.1.1	El dueño del Producto (Product Owner)	22
3.3.1.2	Equipo de Desarrollo	23
3.3.1.3	Scrum Master	24
3.3.2	EVENTOS SCRUM.....	24
3.3.2.1	Pila de Requerimientos.....	24
3.3.2.2	Product Backlog.....	24
3.3.2.3	El Sprint	25
3.3.2.4	Planificación.....	26
3.3.2.5	Sprint Backlog.....	27
3.3.2.6	Daily Scrum	27
3.3.2.7	Estimaciones	28
3.3.2.8	Builds continuos.....	28
3.3.2.9	Revisión de Sprint.....	28
3.4	HERRAMIENTAS	29
3.4.1	TÉCNICAS DE RELEVAMIENTO	29
3.4.2	CASOS DE USO	29
3.4.3	NETBEANS.....	29
3.4.4	POWER DESIGNER.....	30
3.4.5	I-REPORT.....	30
	ANÁLISIS DE RESULTADOS.....	31
4.1	ANÁLISIS DE RESULTADOS	32
4.1.1	PLANIFICACIÓN	32
4.1.1.1	Análisis de Procesos	32
4.1.2	ALCANCE DEL SOFTWARE.....	36
4.1.3	CONFORMACIÓN DEL EQUIPO DE TRABAJO.....	36
4.1.4	DEFINICIÓN DEL BACKLOG DEL PRODUCTO	37
4.1.5	DISEÑO.....	38
4.1.5.1	Modelo de Datos	38

4.1.5.2 Arquitectura General	39
4.1.6 SPRINT 1 “MÓDULO DE EVALUACIÓN”	40
4.1.6.1 Planificación	40
4.1.6.2 Análisis	42
4.1.6.3 Diseño	43
4.1.6.4 Arquitectura	44
4.1.6.5 Construcción y Pruebas	44
4.1.7 SPRINT 2 “MÓDULO DE REPORTES”	46
4.1.7.1 Planificación	46
4.1.7.2 Análisis	47
4.1.7.3 Arquitectura	48
4.1.7.4 Construcción y Pruebas	49
4.1.8 SPRINT 3 “MÓDULO DE ADMINISTRACIÓN”	50
4.1.8.1 Planificación	50
4.1.8.2 Análisis	52
4.1.8.3 Diseño	53
4.1.8.4 Arquitectura	54
4.1.8.5 Construcción y Pruebas	55
CONCLUSIONES Y RECOMENDACIONES	57
5.1 CONCLUSIONES Y RECOMENDACIONES	58
5.1.1 CONCLUSIONES	58
5.1.2 RECOMENDACIONES	59
BIBLIOGRAFIA	60
ANEXOS	61

ÍNDICE DE TABLAS

	PÁGINA
Tabla 1. Diferencias entre metodologías ágiles y no ágiles	13
Tabla 2. Grupos de Scrum	21
Tabla 3. Simbología utilizada en los diagramas	34
Tabla 4. Requerimientos Funcionales / No funcionales. Sprint 0	35
Tabla 5. Equipo de Trabajo y Roles	37
Tabla 6. Backlog del Producto	37
Tabla 7. Sprint 1	41
Tabla 8. Historias de usuario Sprint 1.	41
Tabla 9. Equipo de Trabajo Sprint 1	42
Tabla 10. Actores del sistema	42
Tabla 11. Historias Sprint 1	45
Tabla 12. Sprint 2	46
Tabla 13. Historias de usuario Sprint 2.	47
Tabla 14. Equipo de Trabajo Sprint 1	47
Tabla 15. Actores del sistema	48
Tabla 16. Historias Sprint 2	49
Tabla 17. Sprint 3	51
Tabla 18. Historias de usuario Sprint 3.	51
Tabla 19. Equipo de Trabajo Sprint 3	52
Tabla 20. Actores del sistema	52
Tabla 21. Historias Sprint 3	55

ÍNDICE DE FIGURAS

	PÁGINAS
Figura 1. Modelo Scrum (Martel, 2015)	16
Figura 2. Roles de Scrum (Sutherland, 2015)	22
Figura 3. Sprint (Alaimo, 2015)	25
Figura 4. Builds Continuos (Martel, 2015)	28
Figura 5. Diagrama de Proceso	33
Figura 6. Escala de importancia definida por el Product Owner	38
Figura 7. Modelo de Datos del Sistema	39
Figura 8. Arquitectura MVC (Martel, 2015)	40
Figura 9. Casos de uso – Evaluación	43
Figura 10. Diagrama Conceptual Modulo Evaluación	43
Figura 11. Casos de Uso – Reportes	48
Figura 12. Casos de Uso - Administración	53
Figura 13. Diagrama Conceptual Modulo Administración	54

ÍNDICE DE ANEXOS

	PÁGINAS
ANEXO # 1 Selección de compañero para evaluar – Modulo de Evaluación	62
ANEXO # 2 Evaluación – Módulo de Evaluación	62
ANEXO # 3 Gestión de Reportes – Módulo Reportes	65
ANEXO # 4 Ingreso – Módulo Administración	66
ANEXO # 5 CRUD Proyectos – Módulo Administración	67
ANEXO # 6 CRUD Empleados – Módulo Administración	68
ANEXO # 7 Preguntas de Evaluación	69

NOMENCLATURA O GLOSARIO

AJAX	Acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas
Ley de Pareto	El 20% del esfuerzo proporciona el 80% del valor

RESUMEN

El presente estudio se enfocó en el diseño e implementación de un software para la evaluación de equipos de desarrollo, basados en habilidades, características, esfuerzo, etc. de cada miembro que forma parte de dichos equipos.

Esto es posible ya que se elaboró un diagrama de evaluación mediante preguntas divididas por secciones, estas secciones tienen un peso que ayuda a calcular un resultado general del desempeño del empleado, teniendo en cuenta que los empleados son evaluados únicamente por miembros de su mismo equipo.

Para el desarrollo de este sistema se implementó la metodología Scrum, ya que según un análisis que lo podremos observar en el presente trabajo, se determinó que es la metodología ideal para el desarrollo de este proyecto.

La ejecución y culminación del proyecto permitió reafirmar que Scrum se puede aplicar en proyectos de software de mediano tamaño, en entornos cambiantes, con grupos de trabajo pequeños los cuales involucran permanentemente al dueño del producto.

Se pudo obtener un software funcional de gran ayuda para la empresa Vimeworks, permitiendo a la administración tener un mayor control de sus equipos de desarrollo durante la ejecución de un proyecto, y de esta manera realizar una repartición de bonos o incentivos más “justa” con respecto al esfuerzo que realizó cada miembro del equipo.

ABSTRACT

This study focused on the design and development of a software implementation for evaluating development teams based on skills, characteristics, effort, etc. of each member forming part of such team.

This was possible because an evaluation chart was drawn by questions divided into sections. These sections had weights that helped to calculate an overall result of the employee, considering that employees are judged solely by members of the same team.

For the development of this system, the Scrum methodology was used, An analysis was made to select this methodology. This analysis is shown in the following chapters. It shows that SCRUM is ideal for the development of this project.

The execution and completion of this project reaffirmed that Scrum is a good choice which can be applied to software projects of medium size, in changing environments, with small workgroups who permanently involve the product owner.

A functional software was obtained to supports Vimeworks company. This software allows management to have greater control of their development teams during project implementation, and thus make a more fairly distribution of bonuses and incentives with respect to the effort that was realized every by team member.

INTRODUCCIÓN

1. INTRODUCCIÓN

El giro de negocio de la empresa VIMEWORKS, se caracteriza por conformar periódicamente equipos de desarrollo de software, los cuales están integrados por profesionales, quienes cuentan con diversas cualidades, capacidades y aportan al cumplimiento del proyecto asignado. Esta diversidad en los equipos de trabajo tiene como objetivo generar mejores resultados a la hora de desarrollar un proyecto.

Sin embargo, existe la posibilidad que en la ejecución del proyecto, el personal que forma parte del mismo no aporte de manera equitativa en su desarrollo e impida obtener un resultado ideal.

Además cuando el proyecto llega a su fin, la empresa obtiene una utilidad real, de la cual un porcentaje es repartido a los integrantes del equipo de desarrollo, en una manera equitativa, siendo esto injusto al momento de analizar su aporte en todo el proyecto.

El objetivo general del presente trabajo es diseñar la implementación de un software que permita evaluar en base a perfiles, habilidades y cualidades a los equipos de desarrollo de software de la empresa VIMEWORKS, ubicada en la ciudad de Quito.

Los objetivos específicos son los siguientes:

- Analizar la estructura, organización y desempeño actual de los equipos de desarrollo en los diferentes proyectos que se desarrollan en la empresa VIMEWORKS.
- Medir el porcentaje de aportación de un desarrollador en la ejecución de un proyecto
- Dividir equitativamente el porcentaje de utilidades asignadas al equipo de desarrollo, según el desempeño mostrado en el desarrollo del proyecto

El avance tecnológico en la actualidad ha generado que las organizaciones se vean en la necesidad de implementar una automatización de sus procesos de evaluación a sus empleados, para lo cual se necesita diseñar y desarrollar software a la medida de la empresa Vimeworks.

El desempeño de sus empleados constituye un factor clave en el desarrollo general de la empresa. Y las evaluaciones de desempeño representan herramientas muy poderosas para asegurarse de que el personal se comprometa y produzca más.

Para el desarrollo de software existen varias metodologías de desarrollo, sin embargo en la actualidad y por su gran impacto en la calidad, el tiempo de ejecución, la flexibilidad, etc., Scrum se ha convertido en la metodología más utilizada. Esta metodología sugiere la organización de equipos de desarrollo conformados por profesionales con altos conocimientos en el desarrollo de software.

Scrum permite la formación de equipos multidisciplinarios, ayudando a que todos sus miembros colaboren con un mismo fin, mejorando la comunicación entre ellos y de esta manera tener un mejor desempeño en el proyecto.

Los proyectos pueden ser muy variantes, ya que las necesidades y requerimientos del cliente están propensos a sufrir muchos cambios, por este motivo es necesario enfocarse en obtener un alto índice de eficiencia en la capacidad del equipo en entregar de una manera inmediata los requerimientos emergentes.

Por lo tanto, Vimeworks al diseñar e implementar un software que permita evaluar en base a perfiles, habilidades y cualidades a los equipos de desarrollo, podrá evidenciar la calidad de profesionales que conforman su organización, adicional obtendrá beneficios como: un mejor control de la implicación del personal en cada proyecto, establecer parámetros de efectividad a los integrantes de cada grupo de trabajo, reconocer el esfuerzo y la pro actividad de los profesionales en cada desafío que se presenta día a día, identificar las reacciones que presentan el personal en proyectos con un índice de dificultad alto.

La investigación se realizará en la ciudad de Quito, en la empresa VIMEWORKS, al equipo de desarrollo de software, mismo que está conformado por 24 profesionales. El software resultado de este trabajo permitirá evaluar a los equipos de desarrollo. El área de Recursos Humanos de la empresa proporcionará la información requerida, es decir los perfiles profesionales, los cuales están conformados por habilidades y cualidades que

tendrán diferente valoración (porcentaje de representación) al momento del proceso de evaluación.

El software estará compuesto por 3 módulos los cuales son: Módulo de Administración, Módulo de Evaluación, Módulo de Reportes.

MARCO TEORICO

2. MARCO TEORICO

2.1 SISTEMAS DE INFORMACIÓN

Se puede definir a un sistema de información de forma muy general como un conjunto de elementos tecnológicos que trabajan como uno, para de esta manera realizar una actividad determinada dentro de una empresa(Vieites Gómez Álvaro, 2010). Como su nombre lo especifica, estos sistemas son encargados de manipular la información (procesar, transportar y distribuir) referente a la actividad para la cual fueron desarrollados y así esta pueda ser interpretada o manipulada por el talento humano de la empresa.

2.2 DEFINICIÓN DE APLICACIONES WEB

2.2.1 QUÉ ES UNA APLICACIÓN WEB

Para una aplicación web no se necesita un servicio externo, ni instalar en un computador de escritorio, ya que los datos y los archivos almacenados y procesados para el trabajo dentro de una aplicación web son procesados dentro de la misma.

“Toda la información se encuentra de forma permanente en servidores de internet y nos envían a nuestros dispositivos o equipos los datos que requerimos en ese momento, mostrando una copia temporal en nuestro equipo.

En cualquier momento, lugar y desde cualquier dispositivo podemos acceder a este servicio, sólo necesitamos una conexión a internet y nuestros datos de acceso, que por lo general son el nombre de usuario y contraseña. (GCFAprendeLibre, 2015)”

2.2.2 INTERFAZ

La interfaz es un componente principal en las aplicaciones web ya que en ella interactúa el cliente con el servidor, sin embargo tiene limitaciones en

comparación con las aplicaciones de escritorio, como por ejemplo dibujar en la pantalla.

Para poder ofrecer al usuario una experiencia más interactiva se utilizan lenguajes interpretados, con lo cual se evita recargar la página por cada petición realizada por el usuario, en la actualidad existen varias tecnologías que ayudan a este inconveniente, como por ejemplo *AJAX*.

2.3 ESTRUCTURA DE LAS APLICACIONES WEB

En la actualidad existen muchas variaciones de la estructura de una aplicación web, sin embargo normalmente se encuentra estructurada mediante 3 capas, las cuales son:

El modelo vista controlador es un patrón que define la organización independiente del Modelo (Objetos del negocio), la Vista (interfaz de usuario) y el controlador (controlador de la lógica del negocio).

Para entender cómo funciona el patrón Modelo Vista Controlador, se debe entender la división a través del conjunto de estos tres elementos y como estos componentes se comunican unos con los otros además con otras vistas y controladores externos al modelo principal. Para ello, es importante saber que el controlador interpreta las entradas del usuario (tanto teclado como el ratón), enviando el mensaje de acción al modelo y a la vista para que se proceda con los cambios que se consideren adecuados.

El modelo, la vista y el controlador deben comunicarse de una manera estable los unos con los otros, de manera que sea coherente con las iteraciones que el usuario realizará. Como es lógico la comunicación entre la vista y el controlador es bastante básica pues están diseñados para operar juntos, pero los modelos se comunican de una manera diferente, un poco más sutil.

El modelo no tiene ninguna responsabilidad para comunicar los cambios a la vista porque ocurren solo por orden del usuario, por lo que esta función la llevará a cabo el controlador ya que será el que interprete las ordenes de este usuario debido a que solo debe comunicar que algo ha cambiado (EcuRed, 2016).

2.4 HERRAMIENTAS Y PLATAFORMA DE DESARROLLO

2.4.1 SERVIDOR DE BASE DE DATOS

Una base de datos es una entidad cuya finalidad es almacenar datos de manera estructurada, con la menor redundancia posible, además debe permitir a varias aplicaciones y usuarios acceder y utilizar estos datos (Turner, 2014).

Los usuarios pueden visualizar, ingresar o actualizar los datos, en concordancia con los derechos de acceso que se les haya otorgado.

2.4.1.1 MySQL

“MySQL es un sistema de base de datos utilizada en la web, la cual se ejecuta en un servidor y compila en un sin número de plataformas, siendo ideal tanto para aplicaciones pequeñas como grandes, ya que es muy rápido, fiable y fácil de usar, además utiliza SQL estándar y es código libre apoyado por Oracle Corporation” (W3School, 2016).

Los datos en una base de datos MySQL se almacenan en tablas. Una tabla es una colección de datos relacionados, y se compone de columnas y filas.

2.4.2 JAVA

Esta sección se basa de acuerdo a lo que manifiesta el autor Oscar Belmonte Fernández en su publicación Introducción al lenguaje de programación Java (Fernández, 2010)

El corazón de Java es la capacidad de ser un lenguaje compilado e interpretado. De esta manera todo programa realizado en Java debe de ser compilado y como resultado se obtiene bytcodes, los cuales son interpretados por una máquina virtual, de este modo se consigue la independencia de la máquina.

Java es un lenguaje orientado a objetos de propósito general. Aunque Java en sus principios fue conocido como un lenguaje de programación de applets que se ejecuta en el entorno de un navegador web, con el pasar de los años

se convirtió en un lenguaje robusto y en la actualidad se puede utilizar Java para construir cualquier tipo de proyecto.

La seguridad fue tomada en cuenta como un punto fundamental en el diseño de Java, por este motivo existen varios niveles de seguridad, desde el ámbito del programador, hasta el ámbito de la ejecución en la máquina virtual.

Con respecto al programador, Java realiza comprobación estricta de tipos durante la compilación, evitando con ello problemas tales como el desbordamiento de la pila. Pero, es durante la ejecución donde se encuentra el método adecuado según el tipo de la clase receptora del mensaje; aunque siempre es posible forzar un enlace estático declarando un método como final.

Todas las instancias de una clase se crean con el operador `new()`, de manera que un recolector de basura se encarga de liberar la memoria ocupada por los objetos que ya no están referenciados. La máquina virtual de Java gestiona la memoria dinámicamente.

Una fuente común de errores en programación proviene del uso de punteros. En Java se han eliminado los punteros, el acceso a las instancias de clase se hace a través de referencias.

Además, es importante que el desarrollador sea capaz de tratar todas las posibles excepciones que la aplicación produzca en tiempo de ejecución, por este motivo Java define varios procedimientos a seguir para tratar estos errores.

Java también posee mecanismos para garantizar la seguridad durante la ejecución comprobando, antes de ejecutar código, que este no viola ninguna restricción de seguridad del sistema donde se va a ejecutar.

También cuenta con un cargador de clases, de modo que todas las clases cargadas a través de la red tienen su propio espacio de nombres para no interferir con las clases locales.

Otra de las principales características de este lenguaje de programación es que se puede realizar una codificación concurrente sin la necesidad de utilizar una biblioteca externa.

Finalmente, Java posee un gestor de seguridad con el que poder restringir el acceso a los recursos del sistema.

2.4.2.1 Java Enterprise Edition

Java EE es una plataforma diseñada para desarrollar y ejecutar aplicaciones web en el lenguaje de programación Java, está diseñado principalmente para software modular que se ejecuta sobre un servidor de aplicaciones.

Tiene varias especificaciones de API, tales como JDBC, e-mail, Servicios Web, etc., también existen configuraciones únicas de componentes tales como Enterprise JavaBeans, Java Server Pages y varias tecnologías web, esto permite desarrollar aplicaciones portables entre plataformas y escalable, a la vez de poder integrarse con tecnologías anteriores.

Otros beneficios añadidos son, por ejemplo, que el servidor de aplicaciones puede manejar transacciones, la seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados, significando que los desarrolladores pueden concentrarse más en la lógica de negocio de los componentes en lugar de en tareas de mantenimiento de bajo nivel (PCWorld, 2011).

2.4.3 SERVIDOR DE APLICACIONES

El concepto de servidor de aplicaciones se encuentra relacionado con el concepto de sistema distribuido, ya que permite mejorar aspectos fundamentales en una aplicación tales como la alta disponibilidad, la escalabilidad y el mantenimiento.

La alta disponibilidad se refiere a que un sistema debe funcionar las 24 horas del día, los 365 días del año, esto es posible gracias a técnicas de balanceo de carga y de recuperación ante fallos.

“La escalabilidad es la capacidad que tiene un sistema al momento de incrementar la carga de trabajo, siendo capaces de mejorar el consumo de los recursos y el mantenimiento tiene que ver con la versatilidad a la hora de actualizar, depurar fallos y mantener un sistema” (EcuRed, 2016).

El estándar J2EE permite el desarrollo de aplicaciones de empresa de una manera sencilla y eficiente. Una aplicación desarrollada con las tecnologías J2EE permite ser desplegada en cualquier servidor de aplicaciones o servidor web que cumpla con el estándar. Un servidor de aplicaciones es una implementación de la especificación J2EE.

Proporciona servicios que soportan la ejecución y disponibilidad de las aplicaciones desplegadas. Es el corazón de un gran sistema distribuido.

Como se ha comentado, un servidor de aplicaciones es una implementación de la especificación J2EE. Existen diversas implementaciones, cada una con sus propias características que la pueden hacer más atractiva en el desarrollo de un determinado sistema.

2.5 EVALUACIÓN DE DESEMPEÑO LABORAL

El esfuerzo que desempeña un individuo en el entorno laboral está dirigido, en una parte por sus capacidades y habilidades, además de la perspectiva que tenga del papel que tiene que desempeñar en la organización.

“Una evaluación del desempeño es un concepto dinámico, ya que los empleados son siempre evaluados con cierta continuidad, sea formal o informalmente, es una técnica de la dirección imprescindible en la actividad administrativa, es un medio a través del cual es posible localizar problemas de supervisión del personal, de integración del empleado en la organización o al cargo que ocupa en la actualidad, de desacuerdos, etc. En otros términos la evaluación del desempeño puede ayudar a determinar la falta de desarrollo de una política de recursos humanos adecuada a las necesidades de la organización.” (Gestiopolis, 2002).

2.5.1 MÉTODO DE AUTOEVALUACIÓN

En varias organizaciones se utiliza este método, el cual consiste en la auto evaluación por parte de los empleados, es decir mediante una serie de preguntas las cuales tienen asignada un peso que representa la importancia se pretende obtener una medición cuantitativa del desempeño laboral del empleado.

Este método es basado en grupos que se encuentran compuestos por personas de buen nivel cultural y de un alto nivel intelectual, ya que se necesita poder despojarse del subjetivismo y distorsiones de índole personal (Rrhh-web, 2015).

2.5.2 MÉTODO DE COMISIÓN DE EVALUACIÓN

Este método consiste en asignar una comisión especializada y constituida únicamente para la evaluación de desempeño del personal, estando formada por miembros de diferentes áreas de la organización, los cuales son especializados en el liderazgo de personas.

Por lo general, un grupo de esta comisión participa de todas las evaluaciones y su papel es mantener un equilibrio de los juicios y de respetar los patrones de evaluación, mientras que el otro grupo está encargado de los juicios de los empleados que se encuentran ligados a sus áreas de desempeño (Urdaneta, 2000).

2.5.3 MÉTODO DE ESCALA GRÁFICA

“Se trata de un método que evalúa el desempeño de las personas mediante factores de evaluación previamente definidos y graduados. Este método utiliza un formulario de doble entrada en el cual las líneas horizontales representan los factores de evaluación del desempeño, en tanto que las columnas (sentido vertical) representan los grados de variación de tales factores.” (Gestiopolis, 2002).

En las empresas se utiliza este método atribuyendo puntos, con el fin de poder cuantificar los resultados y de esta manera facilitar la comparación del desempeño de los empleados, los factores son ponderados de acuerdo a la importancia en la evaluación, teniendo en cuenta los objetivos de la empresa.

2.5.4 MÉTODO DE ELECCIÓN FORZADA

“Consiste en evaluar el desempeño de los individuos mediante frases descriptivas de determinadas alternativas de tipos de desempeño individual. El evaluador debe escoger forzosamente entre las frases solo una o las dos que más se aplican al desempeño del empleado evaluado. Esto explica la denominación de elección forzada” (Gestiopolis, 2002).

En este método se cuenta con un supervisor, el cual está en la capacidad de seleccionar las frases que posiblemente suman puntos, de esta manera queda a consideración de una sola persona la evaluación y pudiendo tener una influencia personal en el resultado de dicha evaluación.

2.5.5 MÉTODO DE INVESTIGACIÓN DE CAMPO

“Es desarrollado con base en entrevistas de un especialista en evaluación con el superior inmediato, mediante el cual se verifica y evalúa el desempeño de sus subordinados, determinándose las causas, los orígenes y los motivos de tal desempeño, por medio del análisis de hechos y situaciones. Es un método de evaluación más amplio que permite además de un diagnóstico del desempeño del empleado, la posibilidad de planear junto con el superior inmediato su desarrollo en el cargo y en la organización.” (Gestiopolis, 2002).

2.6 ANÁLISIS DE METODOLOGÍA DE SOFTWARE

Existen dos tipos de metodologías para el desarrollo de software, aquellas metodologías que presentan mayor énfasis en la planificación y en el control del proyecto, teniendo especificaciones precisas de los requerimientos y el modelado del sistema, reciben el apelativo de Metodologías Tradicionales, que también se las puede denominar Metodologías Pesadas.

El otro grupo de metodologías denominadas Metodologías Ágiles, se orientan más hacia una fuerte planificación durante todo el proceso de desarrollo, principalmente están dirigidas a proyectos que se desarrollan en un periodo corto, ya que son metodologías ligeras y no exigen mucha documentación.

En la Tabla 1 se puede observar las principales diferencias que existen entre los dos grandes grupos de metodologías.

Tabla 1. Diferencias entre metodologías ágiles y no ágiles

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

El presente proyecto se basa en el desarrollo de un sistema de evaluación de equipos de desarrollo, el cual debe ser implementado en un periodo corto, ya que para la empresa es indispensable evaluar a sus empleados. Por lo tanto para la implementación del software se utilizara una metodología ágil.

2.7 METODOLOGÍAS ÁGILES

El desarrollo ágil de software hace referencia a métodos de ingeniería del software que tienen como base el desarrollo iterativo e incremental, estas metodologías son muy flexibles debido a que nos encontramos en un mundo en el cual nos exponemos a cambios recurrentes.

Siempre hay que tomar en cuenta que la última tendencia que se presenta en la actualidad, el día de mañana puede quedar obsoleta, por tal motivo existen

las metodologías ágiles donde los requisitos y soluciones se encuentran en una evolución constante mediante la colaboración de grupos auto organizados y equipos multidisciplinarios.

The Agile Alliance, una organización sin ánimos de lucro, la cual se dedica a promover los conceptos relacionados con el desarrollo ágil y de esta forma ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida de estas metodologías es el Manifiesto Ágil, el cual es un documento que resume la filosofía ágil.

2.7.1 EL MANIFIESTO ÁGIL

Según el manifiesto (ISSI, 2003) se valora:

- **Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.** La gente es el principal factor de éxito de un proyecto de software. Es mucho más importante poder construir un buen equipo que construir un entorno, debido a esto muchas veces se comete el error de construir primero un buen entorno y esperar que el equipo se adapte.
- **Desarrollar software que funcione más que conseguir una buena documentación.** No se debe producir documentos a menos que sea necesario de forma inmediata para tomar una decisión importante.
- **La colaboración con el cliente más que la negociación de un contrato.** Debe existir una interacción constante entre el cliente y el equipo de desarrollo.
- **Responder a los cambios más que seguir estrictamente el plan.** La planificación no debe ser estricta sino flexible y abierta.

Los valores anteriores derivaron los doce principios del manifiesto, los cuales son características que diferencian un proceso ágil de uno tradicional.

- La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor*
- Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva*
- Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.*
- La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.*

- v. *Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.*
- vi. *El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.*
- vii. *El software que funciona es la medida principal de progreso.*
- viii. *Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.*
- ix. *La atención continua a la calidad técnica y al buen diseño mejora la agilidad.*
- x. *La simplicidad es esencial*
- xi. *Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.*
- xii. *En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.*

2.7.2 SCRUM

Fue desarrollado por Ken Schwaber, Jeff Sutherland y Mike Beedle. Durante los últimos 10 años ha tenido una gran aceptación en el desarrollo de software, siendo perfecto para proyectos con un cambio rápido de requerimientos.

Scrum se basa en priorizar las funcionalidades y de esta manera poder construir primero la parte con mayor valor para el cliente, además se centra en los principios de inspección continua, adaptación, auto-gestión e innovación.

Está diseñada principalmente para entornos que trabajan con requerimientos que están propensos al cambio y que requieren rapidez y flexibilidad, es una metodología de desarrollo muy simple, sin embargo requiere trabajo duro ya que no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto.

En Scrum el desarrollo se realiza mediante iteraciones llamadas Sprints, con una duración variable de acuerdo al equipo de desarrollo, es recomendable no tener Sprints muy largos, después de cada Sprint se obtiene un incremento ejecutable del producto, el cual se lo muestra al cliente

Estas iteraciones son la base del desarrollo ágil, y Scrum gestiona su evolución mediante reuniones breves, diarias en las que todo el equipo revisa

el trabajo realizado el día anterior y el previsto para el día siguiente (Scrum.org, 2013).

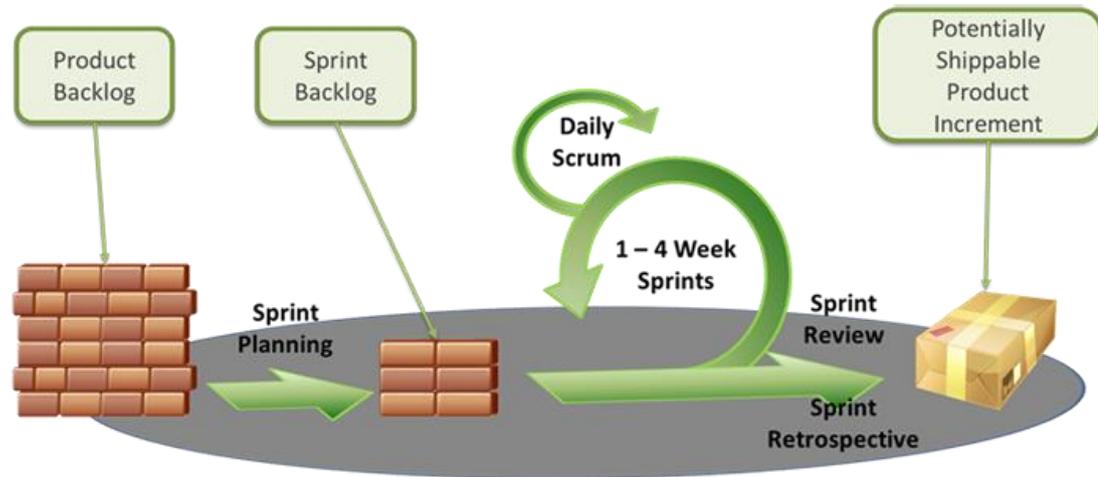


Figura 1. Modelo Scrum (Martel, 2015)

Como muestra la Figura 1, Scrum no se trata únicamente de un método para el desarrollo de software, ya que puede ser aplicado en cualquier contexto, en donde un grupo de personas con diferentes cualidades se juntan para cumplir con un objetivo común.

2.7.2.1 Beneficios

- **Gestión regular de las expectativas del cliente:** El cliente es el que puede establecer y proporcionar un valor a las historias de usuario, el equipo es el que estima el esfuerzo necesario y con esta información el Product Owner establece su prioridad.

Al finalizar cada Sprint el Product Owner comprueba que tan efectivo ha sido y que requerimientos se han cumplido e inmediatamente trasmite el feedback al equipo, esto ayuda mucho al equipo ya que desde un inicio se tiene la posibilidad de tomar decisiones correctivas a partir de los resultados obtenidos.

- **Flexibilidad a cambios:** El cliente es capaz de cambiar de dirección el proyecto debido a cambios en sus prioridades, diferentes cambios en el ámbito del mercado, de la funcionalidad ya desarrollada que ha permitido

entender de una mejor manera el producto, de la velocidad de desarrollo del equipo, etc.

Los cambios de prioridades en el proyecto son naturales, ya que en cada nuevo Sprint se re planifica todo el proyecto y las historias de usuario en las que trabaja el equipo de desarrollo.

Se minimiza la probabilidad de que existan grandes cambios durante la ejecución del proyecto, ya que los requisitos se completan en función al valor que aportan al cliente.

- **Resultados anticipados:** Es un gran beneficio ya que el cliente tiene una funcionalidad importante que puede ser utilizada sin haber finalizado el proyecto por completo.

“Siguiendo la ley de Pareto el cliente puede empezar antes a recuperar su inversión, comenzando a utilizar un producto al que sólo le faltan características poco relevantes, de esta manera se puede sacar un producto al mercado antes que su competidor, puede hacer frente a urgencias o nuevas peticiones del cliente, etc.”(Agiles, 2016).

- **Mayor calidad del software:** El trabajar en varios requerimientos específicos en tiempos cortos, ayuda a la obtención de un software de calidad.

Las personas trabajan más enfocadas y de manera más eficiente cuando hay una fecha límite a corto plazo para entregar un resultado al que se ha comprometido.

La calidad del software es asegurada ya que el cliente puede observar el producto funcional y creciendo a su necesidad.

- **Mayor productividad:** Los miembros del equipo tienen un trabajo sincronizado diariamente, ayudándose entre sí a resolver problemas diarios, los cuales pueden retrasar el cumplimiento de sus objetivos, esto es posible gracias al Daily Scrum.

Cuando es posible utilizar tu creatividad para resolver los problemas y tienes autoridad para organizar tu trabajo, te sientes más motivado, y esto es lo que hace Scrum con su equipo.

- **Maximiza el retorno de la inversión:** Se produce software únicamente con las funciones que aportan mayor valor de negocio, además siempre que el costo de desarrollo es mayor al beneficio pendiente, el cliente puede finalizar el proyecto.

- **Predicciones de tiempos:** Con un equipo formado se puede definir fácilmente su velocidad por sprint y de esta manera se puede estimar fácilmente cuando se podrá obtener una funcionalidad que se encuentra en el Backlog.

- **Reducción de riesgos:** Se despejan los riesgos de manera anticipada ya que las funcionalidades se encuentran priorizadas por valor para el cliente y de esta manera se obtienen la funcionalidad más importante al inicio del proyecto.

Desde el inicio del proyecto el equipo de desarrollo tiene la obligación de manejar los problemas que se presenten al entregar un sprint, ya que esto permite identificar los riesgos, y se pueden tratar de una manera anticipada.

Si hay que equivocarse o fallar, mejor hacerlo lo antes posible, el feedback temprano permite ahorrar esfuerzo y tiempo en errores técnicos.

“La cantidad de riesgo a que se enfrenta el equipo está limitada a los requerimientos que se puede desarrollar en una iteración, además la complejidad y riesgos del proyecto se dividen de manera natural en iteraciones”(Agiles, 2016).

METODOLOGÍA

3. METODOLOGÍA

Se realizará una investigación de tipo exploratorio, debido a que se recopilará la información necesaria directamente del campo en el que se desarrolla Vimeworks. A su vez, descriptivo, ya que se analizará a los grupos de nuestro objeto de estudio. Y finalmente, sistemático para poder proponer un sistema que sea consecuente con los objetivos y necesidades del sistema.

3.1 METODOS DE ESTUDIO

Para la realización de este trabajo, se utilizará varios métodos de investigación, debido a que existen diferentes actividades de investigación que se tendrá que llevar a cabo:

Método inductivo: Partiendo del desconocimiento de la importancia de un sistema de evaluación de desempeño del empleado, se llegará a diseñar e implementar un sistema de evaluación de desempeño del empleado para la empresa Vimeworks, el mismo que ayude a una mejor distribución de bonos a sus empleados.

Método deductivo: Será la herramienta que ayude al estudio del método de evaluación de desempeño, para establecer acciones y poder desarrollar y modelar y diseñar un sistema que cumpla todas las necesidades de la empresa.

Método de análisis: Será utilizado para analizar cada una de las partes posibles e identificar las partes que interesan a la investigación.

Método estadístico: Se recopilará, elaborara e interpretará datos numéricos o estadísticos, que respalden la veracidad de los datos manejados en el estudio, y poder establecer conclusiones y recomendaciones acorde a los resultados.

3.2 METODOLOGÍA DE EVALUACIÓN SISTEMA VIMEWORKS

El desarrollo del sistema de evaluación para la empresa Vimeworks, tuvo como base el método de auto evaluación, ya que como política de la empresa

la opinión del empleado es fundamental, por tal motivo este método de evaluación es el ideal.

Se llevó a cabo varias variaciones en el método para que acoja las necesidades de evaluación de la empresa, como por ejemplo se empleará la evaluación cruzada, la cual consiste en evaluar a todo el equipo de desarrollo y viceversa.

Las preguntas de la evaluación fueron definidas por parte del departamento de recursos humanos de la empresa, utilizando un criterio basado en el entorno laboral que se vive en dicha organización, dichas preguntas se encuentran descritas en el Anexo 7 al final del presente documento.

3.3 SCRUM

De acuerdo a las políticas de la empresa Vimeworks se eligió SCRUM ya que es la metodología de desarrollo usada para sus proyectos. Esto se debe a que es una metodología ideal para proyectos de mediana duración.

3.3.1 ROLES

Scrum define a sus miembros de acuerdo a una típica fabula agilista: *“un cerdo y una gallina discutían qué nombre ponerle al nuevo restaurante; la gallina propuso “Jamón y Huevos”. “No gracias”, respondió el cerdo, “Yo estaré comprometido mientras tu solo implicada”.*

Como se muestra en la Tabla 2 Scrum diferencia claramente a estos dos grupos determinando que quienes se encuentran comprometidos con el proyecto tienen la autoridad suficiente para poder lograr el éxito y quienes no tienen tanta responsabilidad no puedan producir interferencias innecesarias.

Tabla 2. Grupos de Scrum

Comprometidos con el Proyecto	Implicados en el Proyecto
<ul style="list-style-type: none">• Dueño del Producto• Equipo• SCRUM Master	<ul style="list-style-type: none">• Marketing• Comercial• Etc.

Como lo muestra la Figura 2 Scrum tiene una estructura muy simple, y definiendo que todas las responsabilidades del proyecto se reparten en 3 roles.

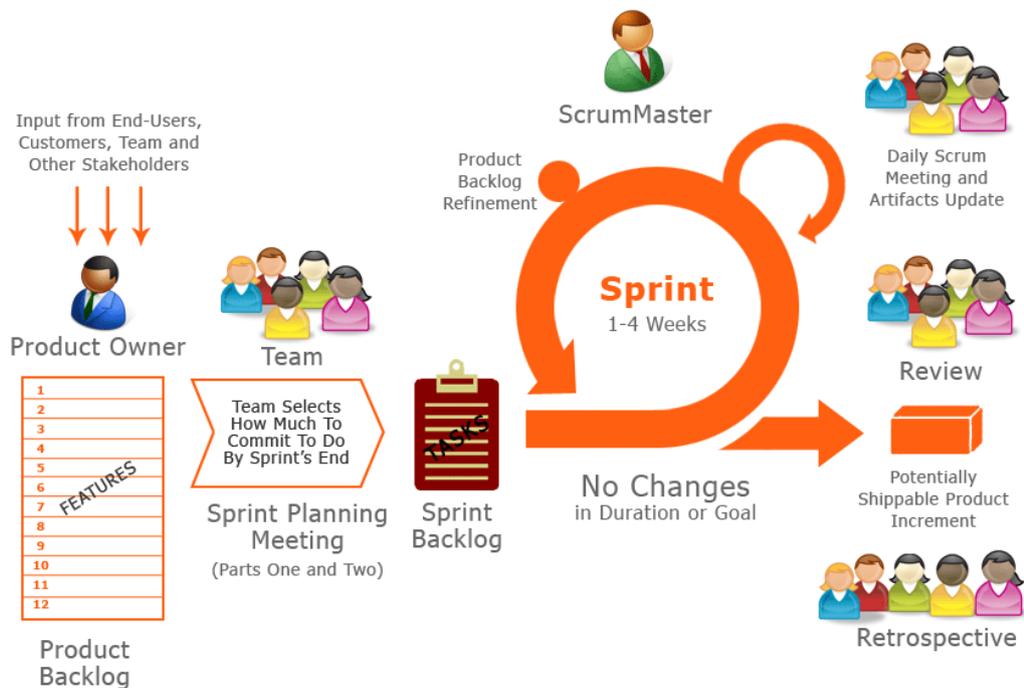


Figura 2. Roles de Scrum (Sutherland, 2015)

Estos roles fueron definidos junto con el personal administrativo de la empresa Vimeworks, teniendo en cuenta las habilidades de cada integrante del equipo, las personas y sus roles se encuentran definidos en las secciones siguientes.

3.3.1.1 El dueño del Producto (Product Owner)

El Product Owner es el que representa a todos los interesados en el producto final, siendo el responsable de definir y obtener un máximo valor del producto y del desempeño del equipo de desarrollo, también tiene el control y visibilidad de la lista de acumulación o lista de retraso del producto (Product Backlog¹).

¹ Lista de historias de usuario

Sus responsabilidades son:

- Financiación del proyecto
- Requisitos del Sistema
- Retorno de la inversión del proyecto
- Lanzamiento del proyecto

El Product Owner está en la capacidad de delegar o realizar el trabajo, sin embargo es el responsable del trabajo, teniendo en cuenta que es una sola persona, pero puede representar las necesidades de todo un comité en el Product Backlog, sin embargo si quieren cambiar la prioridad de los elementos del Product Backlog tiene que ser mediante el Product Owner.

3.3.1.2 Equipo de Desarrollo

El equipo de desarrollo consiste en un grupo de profesionales, responsables de transformar el Backlog de la iteración en un incremento de la funcionalidad del software, solo los miembros del equipo de desarrollo pueden participar de la creación del incremento.

Un equipo de desarrollo Scrum no puede tener más de 20 miembros, siendo 10 personas lo ideal, sin embargo esto puede diferir de acuerdo al tamaño del proyecto, si existen más personas se recomienda dividir en grupos, definiendo un equipo central el cual se encarga de la coordinación, las pruebas cruzadas y la rotación de los miembros.

Los equipos de desarrollo tienen las siguientes características:

- *“Son auto organizados, nadie indica al equipo de desarrollo como convertir elementos del Product Backlog en incrementos de funcionalidad potencialmente desplegados*
- *En el equipo Scrum todos son desarrolladores, sin importar cuales sean sus funciones en ese momento.*
- *Los miembros individuales del equipo de desarrollo pueden tener habilidades especializadas y áreas en las que estén más enfocados, pero la responsabilidad recae en el equipo de desarrollo como un todo”(Schwaber & Sutherland, 2013).*

3.3.1.3 Scrum Master

El Scrum Master es el responsable de asegurar que Scrum es entendido y adoptado, además hace esto asegurándose de que el Equipo Scrum trabaja ajustándose a la teoría, prácticas y reglas de Scrum.

Es un líder que está al servicio del equipo, de esta manera ayuda a las personas externas al equipo a entender que interacciones con el equipo pueden ser de ayuda y cuales no aportan ningún valor, además ayuda a modificar las iteraciones para maximizar el valor creado por el equipo (Schwaber & Sutherland, 2013).

3.3.2 EVENTOS SCRUM

En Scrum existen eventos predefinidos, con el objetivo de regularizar y minimizar la necesidad de tener una reunión fuera de los términos generales, todos los eventos tienen una duración máxima predefinida.

El Sprint se lo observa como un contenedor de eventos, siendo cada uno de dichos eventos los que constituyen una oportunidad para adaptar e inspeccionar cualquier aspecto.

3.3.2.1 Pila de Requerimientos

Se define como una buena práctica en los modelos ágiles, y consiste en hacer lo que el cliente realmente quiere.

Esta es la primera actividad de un proyecto a realizarse con Scrum, ya que en ella se arma una lista exhaustiva de requerimientos fundamentales para el sistema, posteriormente se realiza una evaluación a estos requerimientos, descartando los innecesarios, por este motivo se necesita un representante con una gran capacidad de decisión ya que definirá las prioridades.

3.3.2.2 Product Backlog

Una vez priorizados los requerimientos es posible armar el Product Backlog, que no es más que una forma de registrar y organizar el trabajo pendiente para el desarrollo del producto.

Siendo un documento dinámico que incorpora de una manera constante las necesidades del sistema, por lo tanto nunca llega a ser definitivo teniendo duración durante todo el ciclo de vida del proyecto y manejado por el Product Owner.

3.3.2.3 El Sprint

Scrum tiene como base el control empírico del proceso, ya que se tiene una capacidad de predicción vaga, teniendo una incertidumbre alta, debido a que el proceso puede ser demasiado largo o complejo para poder ser modelado y definido.

Por lo tanto, para poder tener una adaptación rápida a situaciones no previstas se crea una disciplina de inspección frecuente, que permite enfrentar los problemas de una manera dinámica.

Como muestra la Figura 3 un Sprint, es un bloque de tiempo de un mes o menos durante el cual se crea un incremento de producto "Terminado", de esta manera convirtiéndose en el corazón de Scrum, dividiendo al desarrollo en un conjunto de pequeñas "carreras".



Figura 3. Sprint (Alaimo, 2015)

En un Sprint se tiene que tener en cuenta lo siguiente:

- La duración máxima es 30 días, sin embargo lo ideal es 15 días.
- Es imposible modificar el trabajo que se ha acordado en el Backlog durante el Sprint.
- Solo es posible cambiar el curso del Sprint abortándolo, y esto sucede únicamente cuando el Scrum Master decide que no es viable la realización por una de las siguientes razones:

- La tecnología utilizada no funciona
- El giro del negocio ha cambiado
- El equipo ha tenido interferencias

3.3.2.4 Planificación

“El trabajo a realizar durante el Sprint se planifica en la Reunión de Planificación de Sprint, este plan se crea en conjunto con todo el equipo scrum completo, además tiene un máximo de duración de ocho horas para un Sprint de un mes”(Schwaber & Sutherland, 2013).

El objetivo de esta reunión es responder a las siguientes preguntas:

- ¿Qué puede entregarse en el incremento resultante del Sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el incremento?

En una planificación se debe tener en cuenta los siguientes aspectos:

- Una determinación general del alcance.
- Una estimación de alto nivel, la cual se realiza durante la etapa de concepción del proyecto.
- Esfuerzo necesario a labores de soporte o de preparación de ambientes.
- Esfuerzo necesario para las reuniones diarias, de planificación y reunión.
- Requerimientos de infraestructura.
- Habilidades dentro del equipo.
- Restricciones presentes en el equipo, relacionadas con conocimiento del negocio, tecnología, etc.

El rol del Scrum Master durante la planificación es:

- Dirigir
- Facilitar acuerdos entre el equipo y el Product Owner.
- Registrar problemas y riesgos detectados.
- Registrar las asignaciones, tareas y estimaciones.
- Iniciar el Backlog del Sprint.

3.3.2.5 Sprint Backlog

Son las tareas determinadas para el equipo de desarrollo que se tienen que cumplir durante el Sprint.

Estas tareas se convierten en un incremento del producto al finalizar el sprint, siendo estimadas en una duración de entre 1 y 20 horas de trabajo, si una tarea excede el tiempo tiene que ser dividida en sub-tareas.

3.3.2.6 Daily Scrum

“El Scrum diario es una reunión de 15 minutos, en la cual el equipo de desarrollo sincroniza sus actividades y crea un plan para las siguientes 24 horas, ya que se lleva a cabo inspeccionando el trabajo avanzado desde el último Daily Scrum y realizando una proyección del trabajo que podría completarse antes del siguiente”(Schwaber & Sutherland, 2013).

El Daily Scrum se realiza a la misma hora y en el mismo lugar todos los días, durante la reunión cada integrante del equipo responde las siguientes preguntas:

- ¿Qué hice ayer que ayudó al equipo a lograr el objetivo?
- ¿Qué haré hoy para ayudar al equipo a lograr el objetivo?
- ¿Veó algún impedimento que evite que el equipo o yo logremos el objetivo?

Esta reunión ayuda al equipo a evaluar el progreso hacia el objetivo y como se proyecta el equipo para la conclusión de los elementos del Product Backlog seleccionados, además optimiza las posibilidades de llegar a cumplir el objetivo trazado.

El Scrum Master se asegura de que el equipo tenga la reunión, sin embargo el equipo es el responsable de dirigir el Daily Scrum, también el Scrum Master enseña al equipo para que el Daily Scrum dure los 15 minutos predefinidos.

Los Daily Scrum mejoran la comunicación, elimina la necesidad de mantener otras reuniones, identifican y eliminan los impedimentos que se puedan presentar con respecto al desarrollo y mejora el nivel de conocimiento sobre el producto de cada miembro del equipo, esta reunión es clave para el desarrollo del proyecto.

3.3.2.7 Estimaciones

Una estimación se realiza por una sola vez al momento de la planificación, sin embargo las tareas se re-estiman todos los días, registrando los cambios definidos en el Sprint Backlog, se la puede realizar antes o después del Scrum Daily.

3.3.2.8 Builds continuos

Esta técnica es muy utilizada para realizar pruebas básicas del desarrollo del producto y consisten en:

- Los programadores desarrollan según el Sprint Backlog, notificando al integrador cuando se haya finalizado.
- El integrador toma el código y lo integra con el resto del código.
- Se compila el software y se realiza una prueba superficial para validar que la nueva versión se encuentre funcional y estable
- Si se encuentra problemas el código retorna al desarrollador.
- Si no existen errores se notifica al equipo que existe una nueva versión estable del producto.

Estas pruebas se consideran como un ciclo de vida lo cual se puede observar en la Figura 4.



Figura 4. Builds Continuos (Martel, 2015)

3.3.2.9 Revisión de Sprint

Al final del Sprint se lleva a cabo una revisión para revisar el incremento y adaptar el Product Back Log si fuera necesario, durante esta revisión todo el equipo colabora con lo que se hizo en el sprint, teniendo como base esta información se analiza como poder optimizar el valor, es decir la finalidad de esta reunión es tener una retroalimentación del equipo.

A esta reunión asisten el Scrum Master, el equipo de desarrollo, el Product Owner y todas las personas involucradas en el proyecto, teniendo como duración 4 horas, en las cuales se presenta las nuevas funcionalidades desarrolladas durante el Sprint.

3.4 HERRAMIENTAS

3.4.1 TÉCNICAS DE RELEVAMIENTO

Para la ejecución del presente proyecto se utilizará la observación, las reuniones con el cliente y los usuarios funcionales como principal técnica de relevamiento

3.4.2 CASOS DE USO

Los casos de uso son un método práctico para definir los requerimientos, ayudan a definir qué es lo que el sistema tiene que hacer desde el punto de vista del usuario, por este motivo no se puede considerar que nos brindan un modo claro y preciso de la comunicación con el cliente.

Es importante mencionar que los casos de uso no tienen como fin representar el diseño y describir los elementos internos de un sistema, su fin es mejorar y facilitar la comunicación con los futuros usuarios del sistema, además de determinar las características necesarias que tendrá el sistema, es decir describen que tiene que hacer el sistema pero no el cómo.

Se utilizó la herramienta UML para diseñar los casos de uso que ayudaron a entender la lógica del negocio y las principales acciones a ejecutar en el sistema de los usuarios.

3.4.3 NETBEANS

“Netbeans IDE es un entorno de desarrollo integrado (IDE), modular, de base estándar (normalizado), escrito en el lenguaje de programación Java. El proyecto Netbeans consiste en un IDE de código abierto y una plataforma de aplicación, las cuales pueden ser usadas como una estructura de soporte general (framework) para compilar cualquier tipo de aplicación.”(Netbeans.org, 2016)

Netbeans permite a los desarrolladores crear aplicaciones web, así como servicios web en cualquier entorno, soportando Java como lenguaje de programación, por este motivo se ha elegido esta herramienta para el desarrollo del proyecto.

3.4.4 POWER DESIGNER

Esta herramienta ayudó a diagramar la base de datos, tanto el modelo conceptual, como el modelo físico, generando de una manera más eficiente y automatizada la base de datos.

3.4.5 I-REPORT

Esta herramienta permitió generar reportes para mostrar la información y los resultados de la evaluación de desempeño de los empleados de una manera más amigable para el usuario y que de esta manera pueda tomar decisiones estratégicas para la empresa.

ANÁLISIS DE RESULTADOS

4.1 ANALISIS DE RESULTADOS

4.1.1 PLANIFICACIÓN

De acuerdo a la metodología de desarrollo definida en el capítulo anterior, la primera fase del proceso de desarrollo es la planificación, esta etapa al igual que el resto de etapas serán analizadas durante cada Sprint, sin embargo es necesario realizar una planificación inicial definiendo de una manera macro todas las iteraciones y sus objetivos, esta planificación global se la define como Sprint 0.

Antes de realizar una definición del número y objetivos de los Sprints a desarrollarse en el presente proyecto, es necesario realizar un análisis preliminar de los procesos del negocio, definir los alcances del presente proyecto y realizar una estimación de tiempos y recursos.

4.1.1.1 Análisis de Procesos

En los capítulos anteriores ya se definió el alcance general del proyecto, sin embargo es necesario realizar un alcance funcional modular del sistema que se implementará en la ejecución del proyecto.

Para poder definir los alcances funcionales, se analizarán los procesos definidos internamente por la empresa Vimeworks.

- **Macro Proceso – Vimeworks:** Al obtener un nuevo proyecto de desarrollo, se gestiona si existe un equipo disponible o si es necesario construir un nuevo equipo de desarrollo, teniendo en cuenta que los líderes de proyectos son los que seleccionan su propio equipo de desarrollo, los cuales tienen que encajar en la naturaleza del nuevo proyecto.

Se tiene que tomar en cuenta que el equipo de desarrollo puede ir variando durante la ejecución del proyecto, ya que en algún momento se puede necesitar un especialista en alguna área específica para cumplir con un requerimiento y este debe ingresar al equipo, también puede existir que miembros del equipo tengan que separarse del equipo.

Al finalizar el proyecto y entregar el producto final al cliente el equipo de desarrollo es recompensado con un porcentaje de las ganancias que será dividido para todos los miembros (los que siguen formando parte y los que alguna vez formaron parte del equipo), sin embargo no todos los miembros realizaron el mismo esfuerzo para la finalización del proyecto,

por tal motivo se requiere un software que evalúe el desempeño de cada miembro del equipo, este proceso se puede observar en la **Figura 5**, la simbología del diagrama se encuentra descrito en la **Tabla 3**.

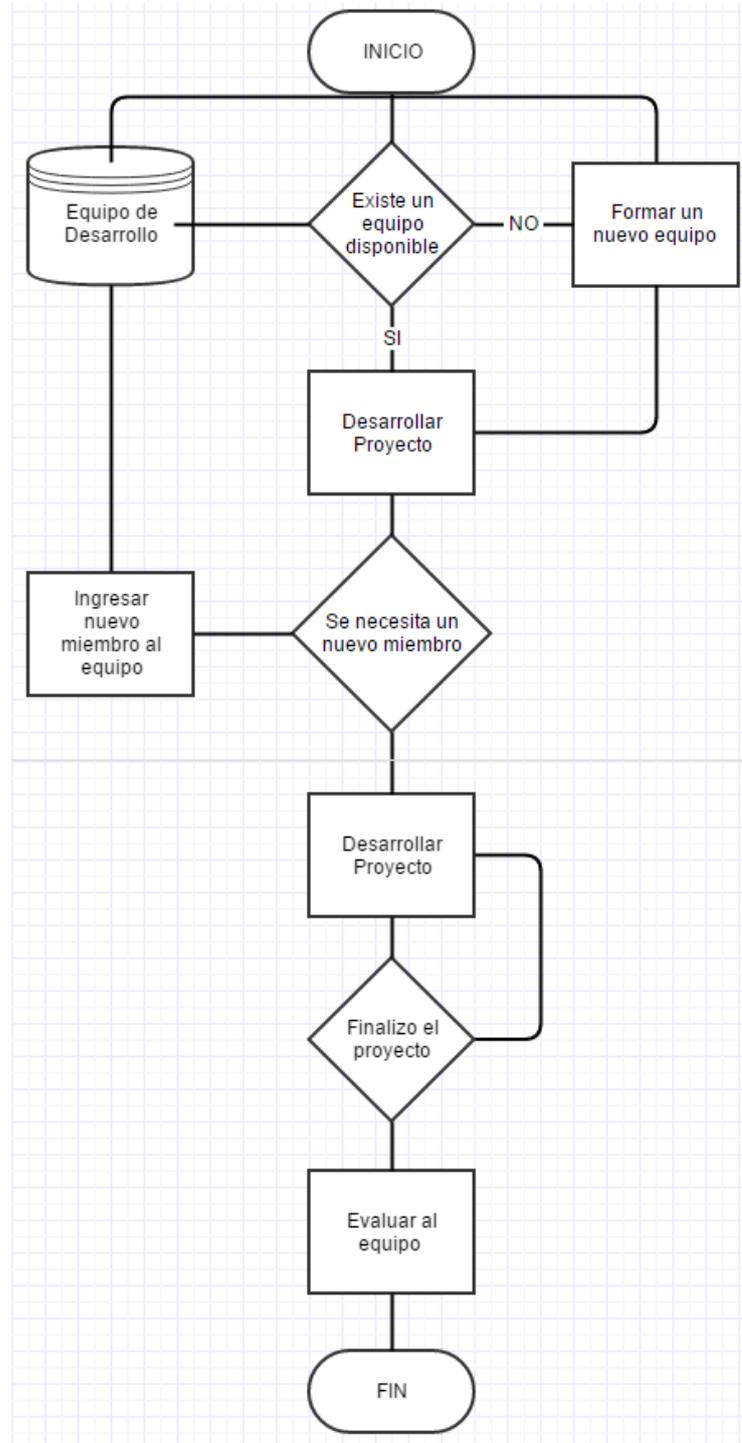
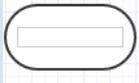
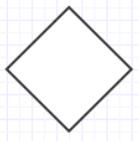
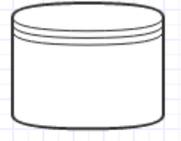


Figura 5. Diagrama de Proceso

Tabla 3. Simbología utilizada en los diagramas

Nombre	Símbolo	Función
Terminal		Representa el inicio y el fin de un programa, también puede representar una parada o interrupción programada.
Proceso		Cualquier tipo de operación que pueda originar cambio de valor, formato o posición de la información
Decisión		Indica operaciones lógicas o de comparación de datos.
Datos almacenados		Representa un grupo de datos, generalmente en una base de datos.

Del proceso definido anteriormente y del respectivo diagrama de flujo, se han podido identificar los requerimientos funcionales y no funcionales con los cuales debe cumplir el sistema a implementar y se encuentra descritos en la Tabla 4.

Tabla 4. Requerimientos Funcionales / No funcionales. Sprint 0

Modulo	Requisitos funcionales	Requisitos no funcionales
1.1		El sistema brindara seguridad para ser utilizado por usuarios que pertenezcan a la organización, con diferentes niveles de acceso
1.2	El sistema permitirá registrar a los usuarios del sistema por parte de un usuario administrador.	
1.3	El sistema permitirá registrar los proyectos de la empresa.	
1.4	El sistema permitirá registrar las secciones para la evaluación	
1.5	El sistema permitirá registrar las preguntas por sección para la evaluación	
1.6	El sistema permitirá asignar los desarrolladores a proyectos y de esta manera formar un equipo de desarrollo	Se debe tomar en cuenta que los empleados tienen un rol y solo los desarrolladores pueden ser asignados a un proyecto
2.1	El sistema permitirá evaluar a tus compañeros de equipo en un proyecto	No se puede modificar las respuestas una vez finalizada la sección
2.2	El usuario solo podrá evaluar a miembros de su equipo	
2.3	El sistema no permitirá realizar una segunda evaluación a un miembro evaluado con anterioridad en el mismo proyecto.	
3	El sistema permitirá obtener los siguientes reportes	
3.1	Reportes de evaluación por persona	
3.2	Reporte de evaluación por proyecto, detallado	
3.3	Reporte final de evaluación	

Nota: La lista de requerimientos ha sido socializada con los dueños del producto.

También se ha podido gestionar tentativamente los módulos a los que pertenecerán las funcionalidades, los módulos identificados durante el análisis son los siguientes:

- 1.- M. Administración
- 2.- M. Evaluación
- 3.- M. Reportes

4.1.2 ALCANCE DEL SOFTWARE

De acuerdo al análisis de los procesos y las funcionalidades definidas anteriormente, el software a desarrollarse tendrá los siguientes alcances.

- **Módulo de administración (1 en tabla 4)** permitirá:
 - Crear, eliminar, modificar los usuarios, proyectos, secciones y preguntas para la evaluación.
- **Módulo de Evaluación (2 en tabla 4)** permitirá:
 - Evaluar a los integrantes del equipo de desarrollo pertenecientes al mismo proyecto.
- **Módulo de Reportes (3 en tabla 4)** permitirá:
 - Consultar y generar los reportes, los cuales nos permitirán dar seguimiento del proceso.

4.1.3 CONFORMACIÓN DEL EQUIPO DE TRABAJO

Es necesario conocer el equipo humano con el cual se cuenta para el desarrollo del proyecto, el equipo de trabajo para el desarrollo del sistema estará conformado según lo descrito en la Tabla 5.

Tabla 5. Equipo de Trabajo y Roles

ROL	PERSONA	AREA
Product Owner	Alejandra Guerrero	Jefa Administrativa Financiera Vimeworks.
Scrum Master	Sebastián Araujo	Desarrollo - Outsourcing
Team	Mauricio Santacruz	Gerencia – Vimeworks
	Luis Falcones	Líder Proyectos - Vimeworks
	Sebastián Araujo	Desarrollo - Outsourcing

Nota: Sebastián Araujo participa en más de un rol, lo cual está permitido en la metodología, estos roles fueron definidos en la etapa anterior (Sprint 0).

4.1.4 DEFINICIÓN DEL BACKLOG DEL PRODUCTO

El Backlog del producto contiene toda la funcionalidad necesaria para el producto final, y de acuerdo a la metodología se ha definido el Backlog del producto para el presente proyecto, identificando las funcionalidades y priorizando cada una de ellas, realizando una estimación de tiempo necesario para su implementación.

El Backlog del producto del presente proyecto se encuentra definido en la Tabla 6:

Tabla 6. Backlog del Producto

ID	Nombre	Importancia	Tiempo estimado	Comentarios
1	Módulo de Administración	600	3 semanas	Funcionalidad de administración de recursos
2	Módulo de Evaluación	1000	3 semanas	Funcionalidad para evaluar a los integrantes del equipo
3	Módulo de Reportes	900	2 semanas	Funcionalidad para reflejar la información obtenida de las evaluaciones

Nota: La importancia está estimada de acuerdo a las necesidades del Product Owner y se encuentra cuantificada en una escala del 0 al 1000 siendo 1000 el más importante, como se muestra en la Figura 6.

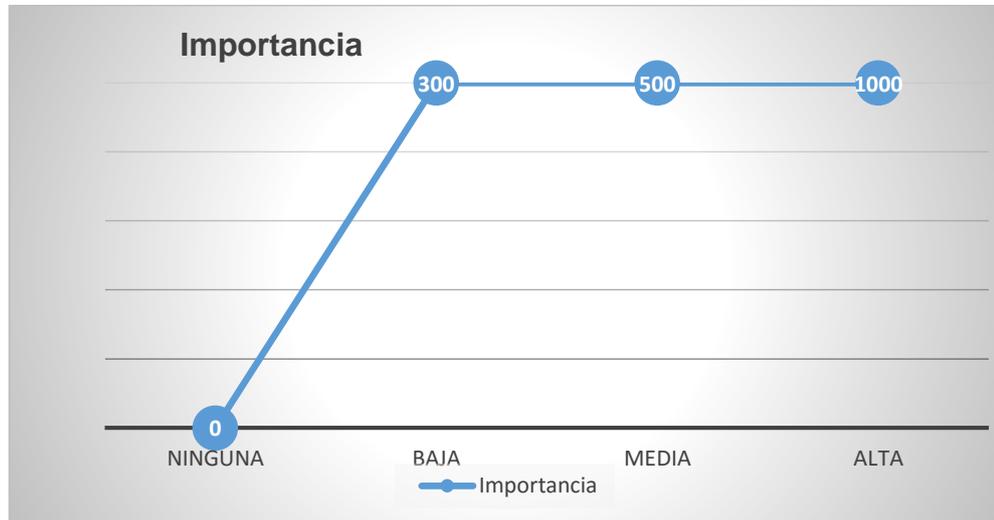


Figura 6. Escala de importancia definida por el Product Owner

4.1.5 DISEÑO

4.1.5.1 Modelo de Datos

En esta etapa se definirá en forma general, el modelo de datos el cual será implementado en los siguientes Sprints, este modelo mostrado en la Figura 7 servirá de guía para la implementación de cada módulo.

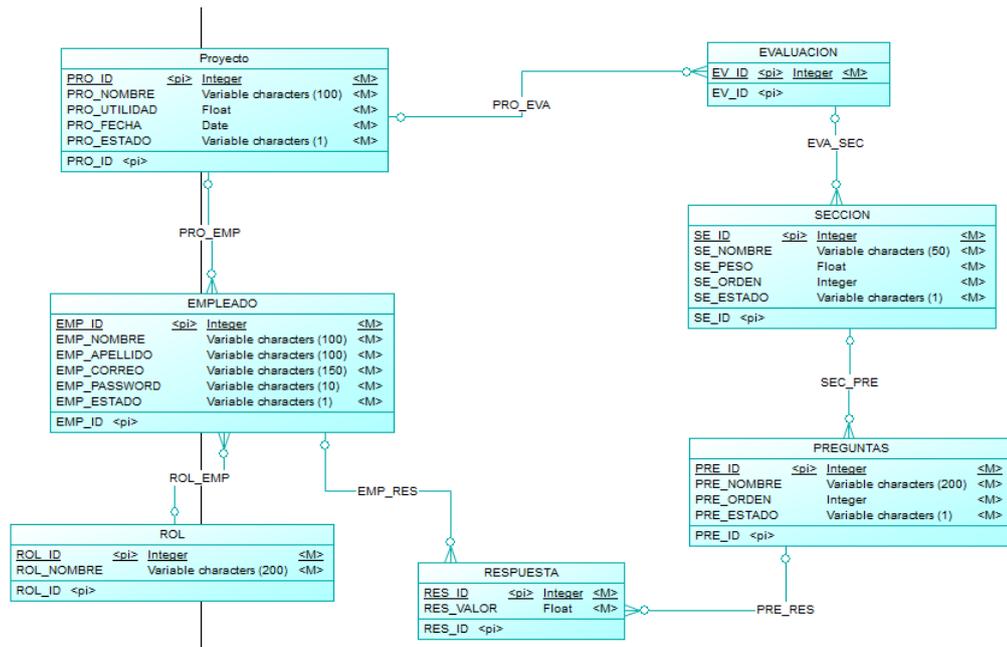


Figura 7. Modelo de Datos del Sistema

4.1.5.2 Arquitectura General

Para la implementación del sistema de Evaluación se utiliza un modelo de 3 capas denominado Modelo-Vista-Controlador (Figura 8)

- **Capa Modelo.-** En esta capa se define la lógica del negocio, la base de datos se encuentra en esta capa.
- **Capa Vista.-** En esta capa se define la interacción del usuario con la aplicación, en el sistema serán las páginas .jsf repartidas por todo el proyecto.
- **Capa Controlador.-** Esta capa es un bloque de código la cual se encarga de relacionar la capa modelo con la capa vista, siendo efectiva cuando el cliente realiza una interacción en la vista, tanto enviando datos al modelo como realizando una petición de datos.

Arquitectura MVC Básica

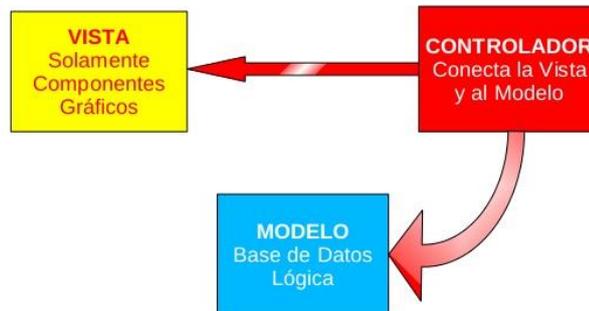


Figura 8. Arquitectura MVC (Martel, 2015)

4.1.6 SPRINT 1 “MÓDULO DE EVALUACIÓN”

El primer sprint tiene como objetivo implementar las funcionalidades requeridas para la evaluación de los miembros de un equipo de desarrollo por parte de un empleado, esto consiste en: poder responder una serie de preguntas para cada integrante del equipo respectivamente.

4.1.6.1 Planificación

Para la planificación del Sprint 1, se mantuvo una reunión con el Product Owner en la cual se analizó el proceso y las funcionalidades necesarias para este módulo.

De la Tabla 4, la cual fue obtenida durante el Sprint 0, se identificaron las funcionalidades que corresponden al módulo de evaluación y que ayudan a cumplir el objetivo del Sprint 1, las cuales son descritas en la Tabla 7.

Tabla 7. Sprint 1

Requisitos Funcionales	Requisitos no funcionales
El sistema permitirá evaluar a tus compañeros de equipo en un proyecto	No se puede modificar las respuestas una vez finalizada la sección
El usuario solo podrá evaluar a miembros de su equipo	
El sistema no permitirá realizar una segunda evaluación a un miembro evaluado con anterioridad en el mismo proyecto.	

De acuerdo a las funcionalidades identificadas para el módulo de evaluación se identifican las historias de usuario para el Sprint 1, las cuales son descritas en la Tabla 8.

Tabla 8. Historias de usuario Sprint 1.

ID	Historia de Usuario	Importancia Product Owner	Importancia Técnica	Descripción
1	Visualización de miembros del equipo	500	1000	Consiste en poder observar las personas que se encuentran en tu equipo de desarrollo
2	Selección de persona a evaluar	800	1000	Consiste en poder seleccionar un compañero de equipo que aún no hayas evaluado
3	Visualización de preguntas por sección	800	800	El sistema debe mostrar las preguntas agrupadas por sección
4	Almacenamiento de respuestas	1000	1000	El sistema debe almacenar las respuestas del usuario

Nota: La importancia técnica está basada en las necesidades no funcionales desde el punto de vista del usuario, pero que son necesarias para el funcionamiento correcto desde el punto de vista técnico.

El equipo de trabajo definido para la implementación de este Sprint se describe en la Tabla 9:

Tabla 9. Equipo de Trabajo Sprint 1

Rol		Persona	Descripción / Tarea
Product Owner		Alejandra Guerrero	Administración del proyecto desde la perspectiva del negocio.
Scrum Master		Sebastián Araujo	Asegurar que el proceso Scrum se lleve con normalidad
Team	Codificación	Sebastián Araujo	Codificación de las historias de usuario
	Pruebas	Luis Falcones	Pruebas funcionales

4.1.6.2 Análisis

Después de identificar las historias de usuario, se identifican los actores y se diagraman los casos de usos para la implementación del Sprint 1.

Actores del sistema

En la Tabla 10 se puede observar los actores que participan en los casos de usos identificados para el Sprint 1.

Tabla 10. Actores del sistema

Actor	Descripción
Usuario Desarrollador	Este tipo de usuario puede evaluar a sus compañeros de equipo.

Diagramas de caso de uso módulo de evaluación

En la Figura 9 se puede observar el caso de uso del módulo de evaluación

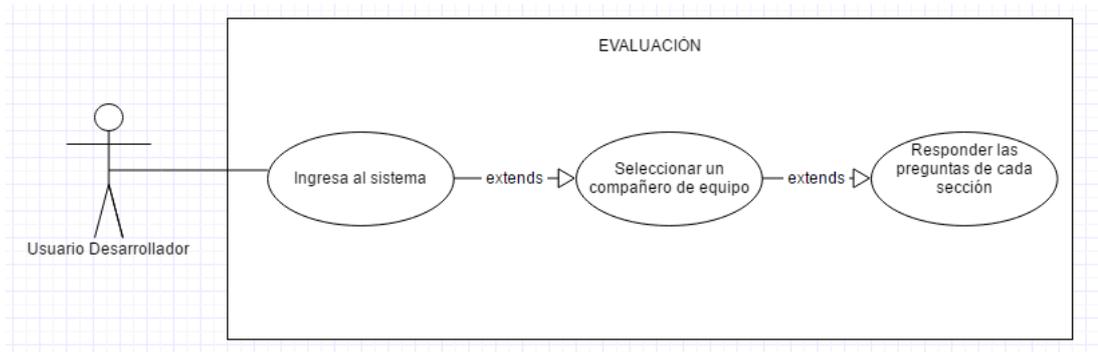


Figura 9. Casos de uso – Evaluación

4.1.6.3 Diseño

De acuerdo al análisis de los casos de uso, se determina la necesidad de utilizar el modelo de datos descrito en la siguiente ilustración, el mismo que será implementado como tablas en la base de datos y como entidades del negocio en la codificación, como se puede observar en la Figura 10.

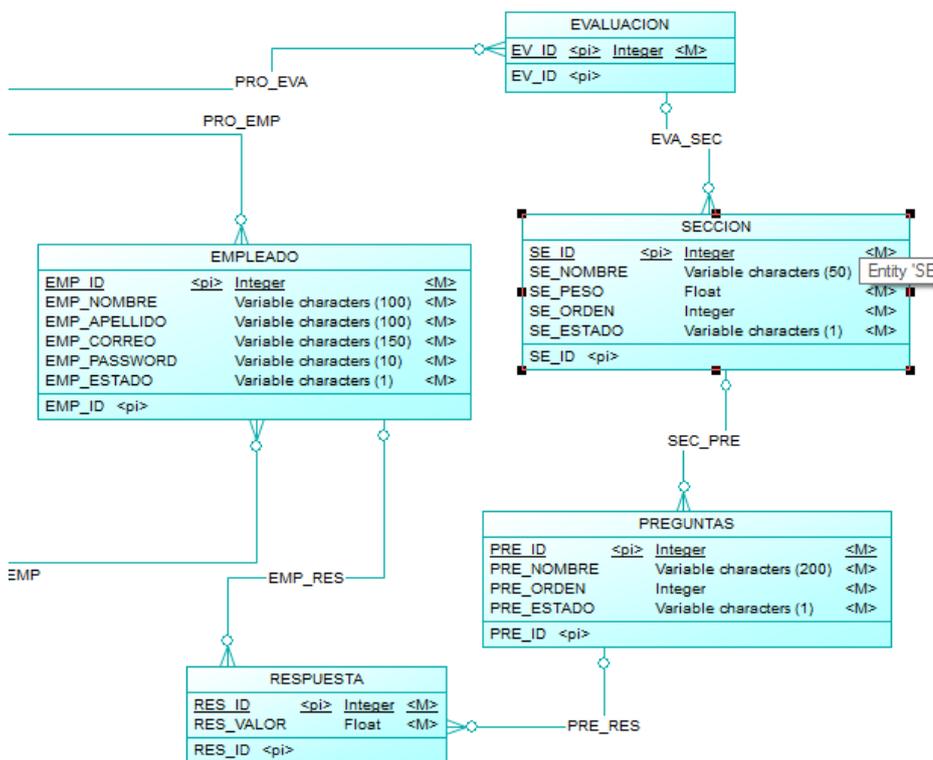


Figura 10. Diagrama Conceptual Modulo Evaluación

4.1.6.4 Arquitectura

De la arquitectura que se planteó en un inicio (Figura 8), la codificación para este módulo será de la siguiente manera:

Modelo

- Entidades que representan las tablas en la base de datos
- Servicios (Facade) los cuales tienen las consultas a la base de datos y la lógica del negocio

Vista

- Lista de Empleados para evaluar
- Evaluación

Controlador

- Gestión de empleados a ser evaluados
- Gestión de evaluación de un empleado

4.1.6.5 Construcción y Pruebas

Backlog del Sprint

Con las historias de usuario ya definidas, se deben implementar en el primer Sprint, por lo tanto es necesario definir el Backlog del Sprint, el cual permitirá implementar las funcionalidades determinadas en la etapa de análisis, el Backlog del Sprint 1 se muestra en la Tabla 11:

Tabla 11. Historias Sprint 1

Historia ID	Tarea #	Nombre Historia	Asignación
1		Visualización de miembros del equipo	
	1	Componentes para la conexión a la Base de Datos	Sebastián Araujo
	2	Configuración para la conexión a la Base de Datos	Sebastián Araujo
	3	Creación GUI	Sebastián Araujo
	4	Creación de controlador y lógica de negocio	Sebastián Araujo
	5	Pruebas Funcionales	Luis Falcones
2		Selección de persona a evaluar	
	1	Creación de controlador y lógica de negocio	Sebastián Araujo
	2	Pruebas Funcionales	Luis Falcones
3		Visualización de Preguntas por sección	
	1	Creación de GUI	Sebastián Araujo
	2	Creación de controlador y lógica de negocio	Sebastián Araujo
	3	Pruebas Funcionales	Luis Falcones
4		Almacenamiento de respuestas	
	1	Creación de controlador y lógica de negocio	Sebastián Araujo
	2	Pruebas Funcionales	Luis Falcones

Pruebas

Se realizaron pruebas de funcionalidad con el responsable de desarrollo en la empresa Vimeworks, las cuales consistieron en validar el proceso y las funcionalidades desarrolladas en el Sprint, y que dichas funcionalidades concuerden con el caso de uso y las necesidades del cliente.

En las pruebas se pudieron corregir errores de definición del requerimiento, además se pidieron mejoras en las funcionalidades del módulo actual, sin

embargo estas funcionalidades se reflejan como historias de usuario en el siguiente Sprint.

El resultado de las pruebas se muestra en el ANEXO 1 al final del presente trabajo.

4.1.7 SPRINT 2 “MÓDULO DE REPORTES”

El segundo sprint tiene como objetivo implementar las funcionalidades requeridas para la generación de reportes por parte del usuario administrador, esto consiste en: poder seleccionar el reporte requerido y visualizarlo.

4.1.7.1 Planificación

Para la planificación del Sprint 2, se mantuvo una reunión con el Product Owner en la cual se analizó el proceso y las funcionalidades necesarias para este módulo.

De la tabla 4, la cual fue obtenida durante el Sprint 0, se identificaron las funcionalidades que corresponden al módulo de reportes y que ayudan a cumplir el objetivo del Sprint 2, las cuales son descritas en la Tabla 12.

Tabla 12. Sprint 2

Requisitos Funcionales	Requisitos no funcionales
El sistema permitirá obtener un reporte de evaluación por persona	
El sistema permitirá obtener un reporte de evaluación por proyecto detallado	
El sistema permitirá obtener un reporte final de evaluación por proyecto	

De acuerdo a las funcionalidades identificadas para el módulo de reportes se identifican las historias de usuario para el Sprint 2, las cuales son descritas en la Tabla 13.

Tabla 13. Historias de usuario Sprint 2.

ID	Historia de Usuario	Importancia Product Owner	Importancia Técnica	Descripción
1	Visualización de proyectos activos	500	1000	Consiste en poder observar los proyectos que se encuentran activos
2	Visualización de miembros de proyectos activos	500	1000	Consiste en poder observar los miembros de proyectos que se encuentran activos
3	Generación de reporte	1000	1000	El sistema debe permitir generar el reporte seleccionado

Nota: La importancia técnica está basada en las necesidades no funcionales desde el punto de vista del usuario, pero que son necesarias para el funcionamiento correcto desde el punto de vista técnico.

El equipo de trabajo definido para la implementación de este Sprint se describe en la Tabla 14:

Tabla 14. Equipo de Trabajo Sprint 1

Rol		Persona	Descripción / Tarea
Product Owner		Alejandra Guerrero	Administración del proyecto desde la perspectiva del negocio.
Scrum Master		Sebastián Araujo	Asegurar que el proceso Scrum se lleve con normalidad
Team	Codificación	Sebastián Araujo	Codificación de las historias de usuario
	Pruebas	Luis Falcones	Pruebas funcionales

4.1.7.2 Análisis

Después de identificar las historias de usuario, se identifican los actores y se diagraman los casos de usos para la implementación del Sprint 2.

Actores del sistema

En la Tabla 15 podemos observar los actores que participan en los casos de usos identificados para el Sprint 2.

Tabla 15. Actores del sistema

Actor	Descripción
Usuario Administrador	Este tipo de usuario puede administrar el sistema y generar reportes

Diagramas de caso de uso módulo de evaluación

El la **Figura 11** se puede observar el caso de uso del módulo de reportes

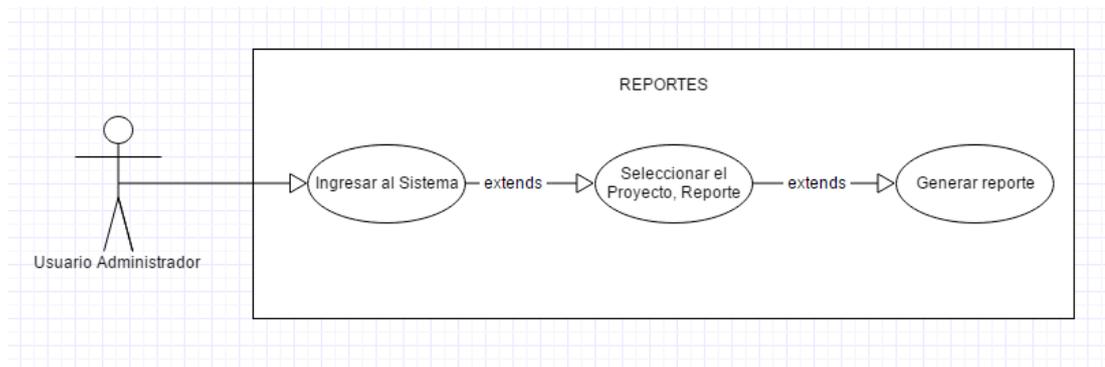


Figura 11. Casos de Uso – Reportes

4.1.7.3 Arquitectura

De la arquitectura que se planteó en un inicio (Figura 8. Arquitectura MVC), la codificación para este módulo será de la siguiente manera:

Modelo

- Entidades que representan las tablas en la base de datos
- Servicios (Facade) los cuales tienen las consultas a la base de datos y la lógica del negocio

Vista

- Selección de Proyecto, tipo reporte y empleado
- Reporte

Controlador

- Gestión de reportes

4.1.7.4 Construcción y Pruebas

Backlog del Sprint

Con las historias de usuario ya definidas, se deben implementar en el segundo Sprint, por lo tanto es necesario definir el Backlog del Sprint, el cual permitirá implementar las funcionalidades determinadas en la etapa de análisis, el Backlog del Sprint 2 se muestra en la Tabla 16:

Tabla 16. Historias Sprint 2

Historia ID	Tarea #	Nombre Historia	Asignación
1		Visualización de proyectos activos	
	1	Creación GUI	Sebastián Araujo
	2	Creación de controlador y lógica de negocio	Sebastián Araujo
	3	Pruebas Funcionales	Luis Falcones
2		Visualización de miembros de proyectos activos	
	1	Creación de controlador y lógica de negocio	Sebastián Araujo
	2	Pruebas Funcionales	Luis Falcones
3		Generación de reporte	
	1	Creación de reporte	Sebastián Araujo
	2	Creación de controlador y lógica de negocio	Sebastián Araujo
	3	Pruebas Funcionales	Luis Falcones

Pruebas

Se realizaron pruebas de funcionalidad con el responsable de desarrollo en la empresa Vimeworks, las cuales consistieron en validar el proceso y las funcionalidades desarrolladas en el Sprint, y que dichas funcionalidades concuerden con el caso de uso y las necesidades del cliente.

En las pruebas se pudieron corregir errores de definición del requerimiento, además se pidieron mejoras en las funcionalidades del módulo actual, sin embargo estas funcionalidades se reflejan como historias de usuario en el siguiente Sprint.

El resultado de las pruebas se muestra en el ANEXO 2 al final del presente trabajo.

4.1.8 SPRINT 3 “MÓDULO DE ADMINISTRACIÓN”

El tercer sprint tiene como objetivo implementar las funcionalidades requeridas para la administración del sistema, esto consiste en: poder crear y modificar: proyectos, secciones, preguntas y usuarios

4.1.8.1 Planificación

Para la planificación del Sprint 3, se mantuvo una reunión con el Product Owner en la cual se analizó el proceso y las funcionalidades necesarias para este módulo.

De la tabla 4, la cual fue obtenida durante el Sprint 0, se identificaron las funcionalidades que corresponden al módulo de evaluación y que ayudan a cumplir el objetivo del Sprint 3, las cuales son descritas en la **Tabla 17**.

Tabla 17. Sprint 3

Requisitos Funcionales	Requisitos no funcionales
El sistema permitirá registrar a los usuarios por parte de un usuario administrador	
El sistema permitirá registrar los proyectos de la empresa	
El sistema permitirá registrar las secciones para la evaluación	
El sistema permitirá registrar las preguntas por sección para la evaluación	
El sistema permitirá asignar los desarrolladores a proyectos y de esta manera formar un equipo de desarrollo	Se debe tomar en cuenta que los empleados tienen un rol y solo los desarrolladores pueden ser asignados a un proyecto

De acuerdo a las funcionalidades identificadas para el módulo de evaluación se identifican las historias de usuario para el Sprint 3, las cuales son descritas en la **Tabla 18**.

Tabla 18. Historias de usuario Sprint 3.

ID	Historia de Usuario	Importancia Product Owner	Importancia Técnica	Descripción
1	CRUD de usuarios	500	500	Consiste en poder crear y modificar usuarios
2	CRUD de proyectos	500	500	Consiste en poder crear y modificar proyectos
3	CRUD de secciones	500	500	Consiste en poder crear y modificar proyectos
4	CRUD de preguntas	500	500	Consiste en poder crear y modificar preguntas
5	Asignación de empleados a un proyecto	500	500	Consiste en poder asignar empleados a los proyectos activos

Nota: La importancia técnica está basada en las necesidades no funcionales desde el punto de vista del usuario, pero que son necesarias para el funcionamiento correcto desde el punto de vista técnico.

El equipo de trabajo definido para la implementación de este Sprint se describe en la Tabla 19 :

Tabla 19. Equipo de Trabajo Sprint 3

Rol		Persona	Descripción / Tarea
Product Owner		Alejandra Guerrero	Administración del proyecto desde la perspectiva del negocio.
Scrum Master		Sebastián Araujo	Asegurar que el proceso Scrum se lleve con normalidad
Team	Codificación	Sebastián Araujo	Codificación de las historias de usuario
	Pruebas	Luis Falcones	Pruebas funcionales

4.1.8.2 Análisis

Después de identificar las historias de usuario, se identifican los actores y se diagraman los casos de usos para la implementación del Sprint 3.

Actores del sistema

En la Tabla 20 podemos observar los actores que participan en los casos de usos identificados para el Sprint 3.

Tabla 20. Actores del sistema

Actor	Descripción
Usuario Administrador	Este tipo de usuario puede administrar el sistema

Diagramas de caso de uso módulo de evaluación

En la Figura 12 se puede observar los casos de uso del módulo de administración

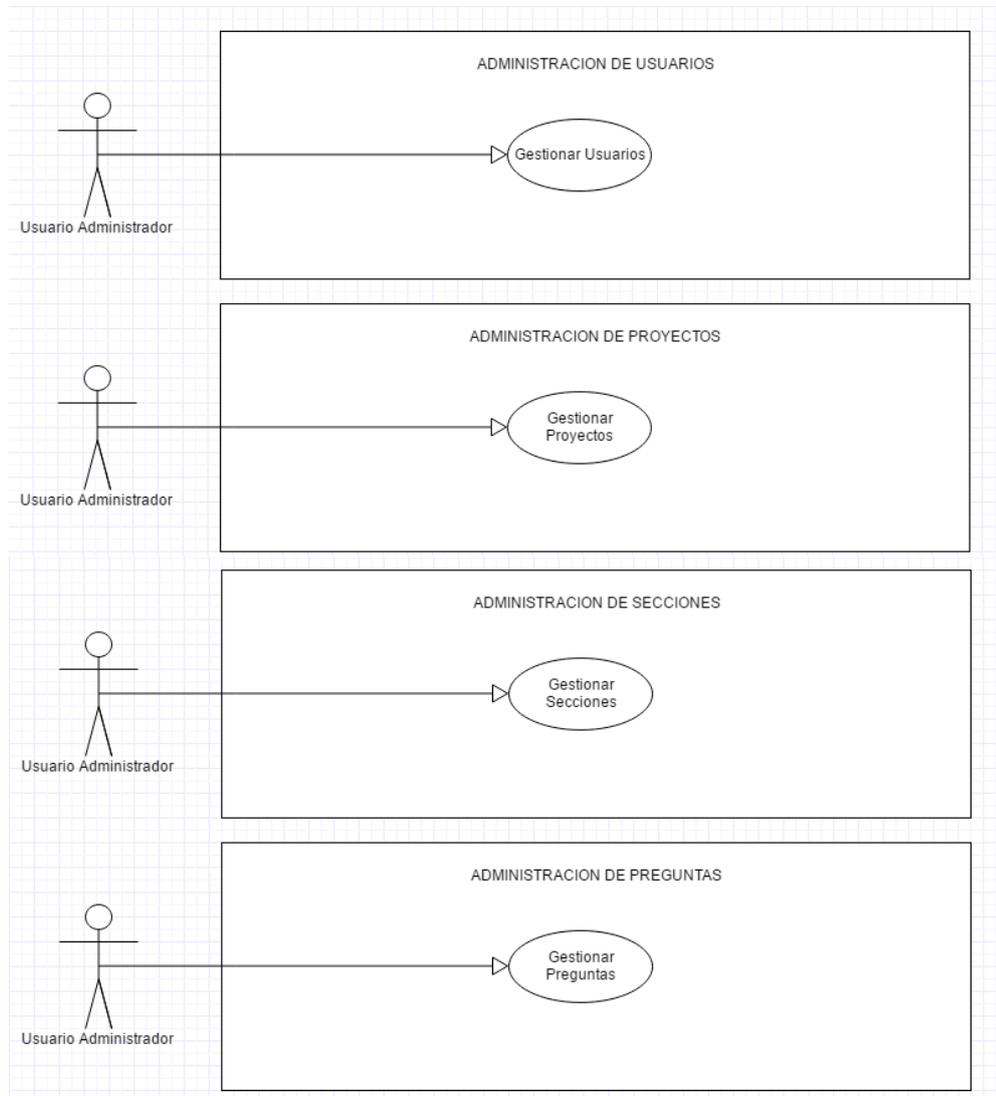


Figura 12. Casos de Uso - Administración

4.1.8.3 Diseño

De acuerdo al análisis de los casos de uso, se determina la necesidad de utilizar el modelo de datos descrito en la siguiente ilustración, las mismas que serán implementadas como tablas en la base de datos y como entidades del negocio en la codificación, como lo podemos observar en la Figura 13.

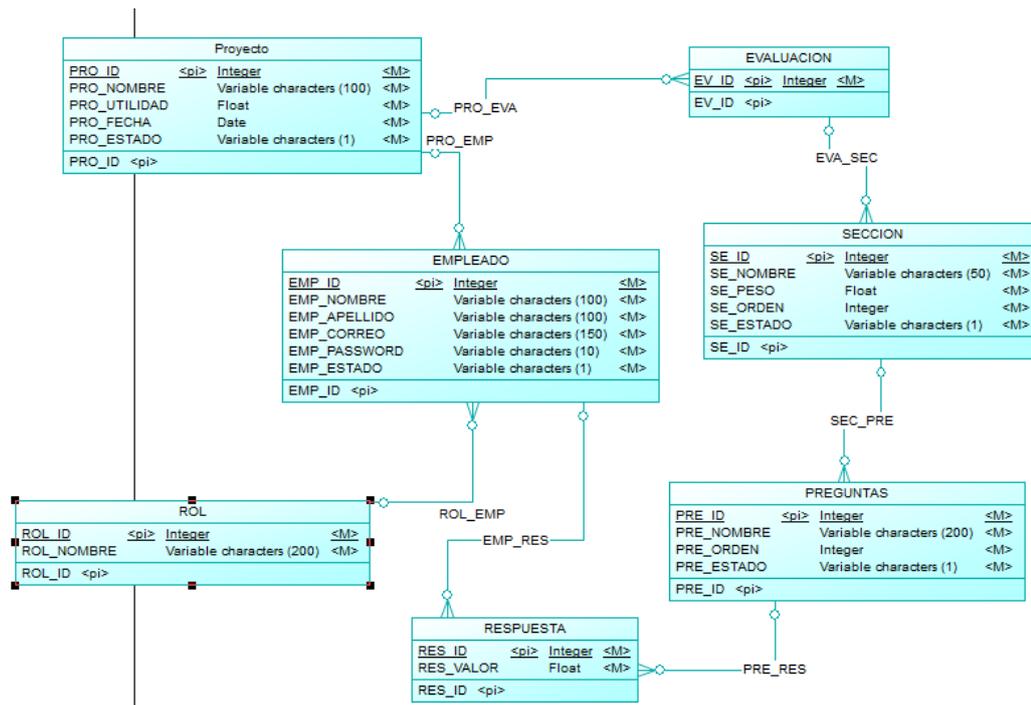


Figura 13. Diagrama Conceptual Modulo Administración

4.1.8.4 Arquitectura

De la arquitectura que se planteó en un inicio (Figura 8), la codificación para este módulo será de la siguiente manera:

Modelo

- Entidades que representan las tablas en la base de datos
- Servicios (Facade) los cuales tienen las consultas a la base de datos y la lógica del negocio

Vista

- CRUD de usuarios
- CRUD de proyectos
- CRUD de secciones
- CRUD de preguntas

Controlador

- Gestión de usuarios
- Gestión de proyectos

- Gestión de secciones
- Gestión de preguntas

4.1.8.5 Construcción y Pruebas

Backlog del Sprint

Con las historias de usuario ya definidas, se deben implementar en el tercer Sprint, por lo tanto es necesario definir el Backlog del Sprint, el cual permitirá implementar las funcionalidades determinadas en la etapa de análisis, el Backlog del Sprint 3 se muestra en la Tabla 21:

Tabla 21. Historias Sprint 3

Historia ID	Tarea #	Nombre Historia	Asignación
1		CRUD de usuarios	
	1	Creación GUI	Sebastián Araujo
	2	Creación de controlador y lógica de negocio	Sebastián Araujo
2	3	Pruebas Funcionales	Luis Falcones
		CRUD de proyectos	
	1	Creación GUI	Sebastián Araujo
3	2	Creación de controlador y lógica de negocio	Sebastián Araujo
	3	Pruebas Funcionales	Luis Falcones
		CRUD de secciones	
4	1	Creación de GUI	Sebastián Araujo
	2	Creación de controlador y lógica de negocio	Sebastián Araujo
	3	Pruebas Funcionales	Luis Falcones
5		CRUD de preguntas	
	1	Creación GUI	Sebastián Araujo
	2	Creación de controlador y lógica de negocio	Sebastián Araujo
6	3	Pruebas Funcionales	Luis Falcones
		Asignación de empleados a un proyecto	
	1	Creación de controlador y lógica de negocio	Sebastián Araujo
7	2	Pruebas Funcionales	Luis Falcones

Pruebas

Se realizaron pruebas de funcionalidad con el responsable de desarrollo en la empresa Vimeworks, las cuales consistieron en validar el proceso y las funcionalidades desarrolladas en el Sprint, y que dichas funcionalidades concuerden con el caso de uso y las necesidades del cliente.

En las pruebas se pudieron corregir errores de definición del requerimiento, además se pidieron mejoras en las funcionalidades del módulo actual.

El resultado de las pruebas se muestra en el ANEXO 3 al final del presente trabajo.

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES Y RECOMENDACIONES

5.1.1 CONCLUSIONES

- El sistema fue desarrollado en el tiempo preestablecido sin mayores complicaciones que lo retrasen.
- El sistema de evaluación permite a la empresa tener una mejor distribución de los beneficios extras al finalizar un proyecto, ya que se puede medir el porcentaje de aportación de los miembros del equipo de desarrollo involucrado en el proyecto.
- Scrum permitió realizar con facilidad los cambios que se presentaron en el transcurso del desarrollo del sistema, así como mantener un control de calidad con el usuario después de cada iteración.
- El utilizar tecnologías tales como JSF, EJB, JPA, para el desarrollo del sistema fue muy útil ya que se pudo obtener un diseño amigable y flexible.
- La permanente participación del Product Owner tuvo un aporte muy importante, ya que después de cada iteración se realizaban presentaciones que permitían identificar cambios a realizarse en la siguiente iteración.

5.1.2 RECOMENDACIONES

- Se recomienda realizar una capacitación a los usuarios del sistema con el fin de que sea utilizado de una manera eficiente.
- Es recomendable que el departamento de recursos humanos mantenga una constante revisión del sistema, ya que esto permitirá la creación de nuevas preguntas y secciones para la evaluación de los equipos de desarrollo, y mantendrá una evaluación dinámica
- Es importante la capacitación a los empleados para obtener una corrección de los aspectos en los cuales se encuentran fallando y de esta manera poder obtener un nivel de excelencia al desarrollar los proyectos.
- Se recomienda ampliar la evaluación al equipo administrativo, ya que el sistema en la actualidad se encuentra limitado únicamente para evaluar el desempeño de los desarrolladores, sin embargo el equipo administrativo también forma parte de la organización y cumple un papel importante en el desarrollo de un proyecto.
- Se recomienda mantener un registro histórico de la evolución de las evaluaciones a los equipos de desarrollo ya que con esto se puede evidenciar en cuanto pudo aportar el sistema para los objetivos de la empresa.
- Es recomendable en un futuro generar un módulo en el sistema que permita de acuerdo al desempeño de los empleados generar un aumento a su salario o un ascenso.

BIBLIOGRAFIA

- Agiles, P. (2016). Beneficios de Scrum. Recuperado a partir de <https://proyectosagiles.org/beneficios-de-scrum/>
- Alaimo, M. (2015). *Proyectos Ágiles con Scrum*.
- EcuRed. (2016). Servidor de Aplicaciones. Recuperado a partir de http://www.ecured.cu/Servidor_de_Aplicaciones
- Fernández, O. B. (2010). Introducción al lenguaje de programación Java.
- GCFAprendeLibre. (2015). ¿Qué es una aplicación web?
- Gestiopolis. (2002). Métodos de evaluación del desempeño laboral. Recuperado a partir de <http://www.gestiopolis.com/metodos-de-evaluacion-del-desempeno-laboral/>
- ISSI, G. (2003). Metodologías Ágiles en el Desarrollo de Software.
- Martel, A. (2015). *Gestión Práctica de Proyectos con Scrum*.
- Netbeans.org. (2016). Información NetBeans. Recuperado a partir de https://netbeans.org/community/releases/61/index_es.html
- PCWorld. (2011). Java Enterprise Edition 7.
- Rrhh-web. (2015). La evaluación de desempeño en la administración de los recursos humanos. Recuperado a partir de <http://www.rrhh-web.com/evaluaciondepuesto3.html>
- Schwaber, K., & Sutherland, J. (2013). La Guía de Scrum.
- Scrum.org. (2013). Guía Scrum.
- Sutherland, J. (2015). *SCRUM EL NUEVO Y REVOLUCIONARIO MODELO ORGANIZATIVO QUE CAMBIARÁ TU VIDA*. Planeta.
- Turner, S. (2014). Una introducción a servidores de bases de datos.
- Urdaneta, M. P. (2000). La Evaluación de desempeño y la gestión de RRHH.
- Vieites Gómez Álvaro. (2010). *Sistemas de Información Herramientas Prácticas para la Gestión Empresarial*.
- W3School. (2016). MySQL Database. Recuperado a partir de http://www.w3schools.com/php/php_mysql_intro.asp

ANEXOS

ANEXO # 1

Selección de compañero para evaluar – Modulo de Evaluación

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

Nombre	Apellido	Opcion
XAVIER	SOTOMAYOR	Seleccionar
ADRIANA	SALAZAR	Seleccionar

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

ANEXO # 2

Evaluación – Módulo de Evaluación

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

CAPACIDAD TECNICA	Conoce bien los lenguajes de programación	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
TRABAJO EN EQUIPO	Propone soluciones eficientes	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
INTEGRACION Y PARTICIPACION	Aprende nuevos lenguajes	<input type="radio"/> 1 <input checked="" type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5
TIEMPO DE PARTICIPACION	Utiliza nuevas herramientas	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
COLABORACION AL ADMINISTRATIVO	Participa en temas de innovación	<input type="radio"/> 1 <input type="radio"/> 2 <input checked="" type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5

[Siguiente](#)

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

CAPACIDAD TECNICA

TRABAJO EN EQUIPO

INTEGRACION Y PARTICIPACION

TIEMPO DE PARTICIPACION

COLABORACION AL ADMINISTRATIVO

Comparte su conocimiento

1 2 3 4 5

Se comunica asertivamente

1 2 3 4 5

Pone a discusión sus decisiones

1 2 3 4 5

Deja que los demás hablen

1 2 3 4 5

Asiste en las tareas a sus compañeros

1 2 3 4 5

Siguiente

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

CAPACIDAD TECNICA

TRABAJO EN EQUIPO

INTEGRACION Y PARTICIPACION

TIEMPO DE PARTICIPACION

COLABORACION AL ADMINISTRATIVO

Asiste a eventos de integración de la compañía

1 2 3 4 5

Asiste a eventos de emprendimiento

1 2 3 4 5

Muestra interés en colaborar en proyectos externos

1 2 3 4 5

Propone eventos y/o trabajos para el aprendizaje

1 2 3 4 5

Asiste puntualmente a capacitaciones

1 2 3 4 5

Siguiente

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

- CAPACIDAD TECNICA
- TRABAJO EN EQUIPO
- INTEGRACION Y PARTICIPACION
- TIEMPO DE PARTICIPACION
- COLABORACION AL ADMINISTRATIVO

Tiempo de participación en el proyecto

1 2 3 4 5

[Siguiente](#)

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

- CAPACIDAD TECNICA
- TRABAJO EN EQUIPO
- INTEGRACION Y PARTICIPACION
- TIEMPO DE PARTICIPACION
- COLABORACION AL ADMINISTRATIVO

Colaboración en entrega de información y facilitación de trámites.

1 2 3 4 5

[Finalizar](#)

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

INTEGRANTE DAVID

PESO SECCION
10.0 TIEMPO DE PARTICIPACION 0.45

	SEBASTIAN	XAVIER	Promedio
Tiempo de participación en el proyecto	5.0	4.0	4.5

ANEXO # 4

Ingreso – Módulo Administración

Sistema de Evaluación Vimeworks

[i Informacion](#) [Home](#)

Ingreso Empleado

[Olvido su contraseña? Presione aqui.](#)

Powered by Vimeworks

Copyright © 2013 Vimeworks S.A. – Todos los derechos reservados

ANEXO # 5

CRUD Proyectos – Módulo Administración

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

Administración de Proyectos [+ Crear](#)

(1 of 1) [1](#)

Código	Nombre	Editar
1	MINISTERIO	Seleccionar
2	Senplades	Seleccionar
3	Prueba	Seleccionar

(1 of 1) [1](#)

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

Administración de Proyectos [+ Crear](#)

(1 of 1) [1](#)

Código	Nombre	Editar
1	MINISTERIO	Seleccionar
2	Senplades	Seleccionar
3	Prueba	Seleccionar

(1 of 1) [1](#)

Creacion / Edicion

Código

Nombre

Utilidad

Fecha

Estado Activo Inactivo

[Crear](#) [Cancelar](#)

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

ANEXO # 6

CRUD Empleados – Módulo Administración

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

Administración de Empleados [+ Crear](#)

(1 of 1) [1](#)

Código	Nombre	Proyecto	Editar
1	SEBASTIAN ARAUJO	MINISTERIO	Seleccionar
2	DAVID RUBIO	MINISTERIO	Seleccionar
3	XAVIER SOTOMAYOR	MINISTERIO	Seleccionar
4	ADRIANA SALAZAR	MINISTERIO	Seleccionar

(1 of 1) [1](#)

Powered by Vimeworks

Copyright© 2013 Vimeworks S.A. - Todos los derechos reservados

Sistema de Evaluación Vimeworks

[i Informacion](#) [Salir](#)

Administración de Empleados [+ Crear](#)

(1 of 1) [1](#)

Código	Nombre	Proyecto	Editar
1	SEBASTIAN ARAUJO	MINISTERIO	Seleccionar
2	DAVID RUBIO	MINISTERIO	Seleccionar
3	XAVIER SOTOMAYOR	MINISTERIO	Seleccionar
4	ADRIANA SALAZAR	MINISTERIO	Seleccionar

(1 of 1) [1](#)

Creacion / Edicion

Código

Nombre

Apellido

Correo

Password

Proyecto

Estado Activo Inactivo

[Crear](#) [Cancelar](#)

ANEXO # 7

Preguntas de Evaluación

PESO		SECCION		
5,27	30%	CAPACIDAD TECNICA		
			No.	Pregunta
			1	Conoce bien los lenguajes de programación
			2	Propone soluciones eficientes
			3	Aprende nuevos lenguajes
			4	Utiliza nuevas herramientas
			5	Participa en temas de innovación
5,01	30%	TRABAJO EN EQUIPO		
			No.	Pregunta
			1	Comparte su conocimiento
			2	Se comunica asertivamente
			3	Pone a discusión sus decisiones
			4	Deja que los demás hablen

			5	Asiste en las tareas a sus compañeros
3,71	20%	INTEGRACION Y PARTICIPACION		
			No.	Pregunta
			1	Asiste a eventos de integración de la compañía
			2	Asiste a eventos de emprendimiento
			3	Muestra interés en colaborar en proyectos externos
			4	Propone eventos y/o trabajos para el aprendizaje
			5	Asiste puntualmente a capacitaciones
10,00	10%	TIEMPO DE PARTICIPACION		Aquí solo es el tiempo en el que participaron en el proyecto
0,50	10%	COLABORACION AL ADMINISTRATIVO		Colaboración en entrega de información y facilitación de trámites.