

**PROTOTIPO BASADO EN REDES DE SENSORES INALÁMBRICAS (WSN)
PARA EL APOYO DE CULTIVOS EXPERIMENTALES EN LA GRANJA DE
LA UNIVERSIDAD DE LOS LLANOS – SENVARA (Sensores de Variables
Ambientales)**

MODALIDAD: ESTUDIANTE PARTICIPANTE EN INVESTIGACIÓN (EPI)

NÉSTOR STIVEN TOLEDO BURGOS

**UNIVERSIDAD DE LOS LLANOS
FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA
PROGRAMA INGENIERÍA DE SISTEMAS
VILLAVICENCIO- COLOMBIA**

2017

**PROTOTIPO BASADO EN REDES DE SENSORES INALÁMBRICAS (WSN)
PARA EL APOYO DE CULTIVOS EXPERIMENTALES EN LA GRANJA DE
LA UNIVERSIDAD DE LOS LLANOS – SENVARA (Sensores de Variables
Ambientales)**

**Trabajo presentado como requisito parcial para optar título de
Ingeniero de Sistemas**

Directora:

OLGA LUCERO VEGA MÁRQUEZ

MSc en Ingeniería, área Sistemas y Computación

**UNIVERSIDAD DE LOS LLANOS
FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA
PROGRAMA INGENIERÍA DE SISTEMAS
VILLAVICENCIO- COLOMBIA**

2017

NOTA ACEPTACIÓN

MsC. Ingeniería de Sistemas y Computación

OLGA LUCERO VEGA MÁRQUEZ

Director / Jurado

Fecha

Quiero dedicarle primeramente este logro a Dios que me otorgó la vida. A mis padres y hermanos por su apoyo incondicional, sus consejos que fueron indispensables para afrontar este arduo camino. A los docentes que compartieron sus conocimientos y saberes para hacer de nosotros unos profesionales íntegros, finalmente, agradecer a mis compañeros y demás familia que siempre estuvieron ahí apoyándome.

AGRADECIMIENTOS

Agradezco primeramente a mi madre Diana Lucia Burgos Martínez por su sabiduría aplicada en los buenos consejos que han hecho de mí una mejor persona, por su incansable interés y dedicación, por ser pieza fundamental en mi educación y formación como ser humano integro.

A mí padre Reinaldo Toledo Becerra quien fue mi apoyo fundamental a la hora de iniciar este largo camino, persona que siempre fue incondicional, apoyándome y brindándome su cariño y comprensión, gracias a ello he culminado una de mis grandes metas, la cual es el inicio de un mejor futuro para todos.

Al centro de investigación CiEnTiC de la Universidad de los Llanos y su excelente equipo de trabajo, por permitirme desarrollar este proyecto bajo su coordinación y apoyo, a la ingeniera Olga Lucero Vega, por su paciencia, dedicación y disciplina, por compartir sus conocimientos y buenos consejos.

Finalmente a todos los que de una u otra manera aportaron en la elaboración y culminación de este trabajo, compañeros, familiares y amigos.

¡Gracias!.

Contenido

1	RESUMEN.....	8
2	ABSTRACT	8
3	INTRODUCCIÓN.....	9
4	DESCRIPCIÓN DEL PROBLEMA	10
5	JUSTIFICACIÓN.....	11
6	OBJETIVOS	14
6.1	Objetivo general	14
6.2	Objetivos Específicos	14
7	MARCO TEÓRICO	14
7.1	Estructura de una red de sensores	15
7.2	Tecnologías Disponibles.....	17
7.2.1	JAVA EE (Java Enterprise Edition)	17
7.2.2	JSF (Java Server Faces)	18
7.2.3	HTML 5.....	20
7.2.4	Base de datos.....	21
7.2.5	Procces Dashboard	21
7.2.6	JavaScript.....	21
7.2.7	CSS.....	22
7.2.8	Bootstrap	22
7.2.9	XAMMP	22
7.2.10	JPA (Java Persistence Api)	22
7.2.11	MAVEN.....	24
7.2.12	JBOSS.....	26
8	METODOLOGÍA.....	29
8.1	PSP-TSP.....	30
8.2	Niveles en PSP-TSP.....	31
8.3	Nivel a utilizar en este proyecto.	32
8.4	Fases de la metodología.....	33
8.4.1	Fase de Planificación.....	33
8.4.2	Fase de Desarrollo.	34
8.4.3	Fase de PostMortem	34

9	APLICACIÓN DE LA METODOLOGÍA	35
9.1	RESULTADOS	35
9.1.1	Glosario	35
9.1.2	Modelo conceptual	36
9.1.3	Modelo de Procesos BPMN	36
9.1.4	Levantamiento de requerimientos	37
9.1.5	Arquitectura de la solución.....	44
9.1.6	Diseño de la interfaz	46
9.2	Diseño de solución	47
9.2.1	SenVarA.....	47
9.2.2	SenVarA Web.....	48
9.3	Desarrollos	53
9.3.1	SenVarA.....	53
9.3.2	SenVarA Web.....	56
10	RESULTADOS.....	59
11	CONCLUSIONES	59
12	TRABAJO FUTURO.....	60
13	Bibliografía	60

1 RESUMEN

En este proyecto se desarrolló un prototipo que integra las Redes de Sensores Inalámbricos (Wireless Sensor Networks – WSN, por sus siglas en inglés) y el software orientado a la web. Se construyó el sistema de información SenVarA (Sensores de Variables Ambientales), que facilita el manejo de información de las variables ambientales de un cultivo agrícola, de forma que se podrá consultar el valor de cada parámetro monitoreado desde cualquier lugar con acceso a internet. Este sistema permite la recolección de los valores de las variables de manera sistemática y almacenamiento de las mismas para posterior procesamiento y análisis. SenVarA facilita la administración de los datos tomados por los sensores convirtiéndolos en información que ayuda a la toma de decisiones en el manejo del cultivo.

2 ABSTRACT

In this project a prototype has been developed that integrates Wireless Sensor Networks (WSN) and web-oriented software. (Sensors of Environmental Variables), which facilitates the handling of information on the environmental variables of an agricultural crop, so that you can see the value of each parameter monitored from anywhere with Internet access. This system allows the collection of the values of the variables in a systematic way and the storage of the same for the later processing and the analysis. SenVarA facilitates the management of the data taken by the sensors, making them the information that helps decision making in crop management.

3 INTRODUCCIÓN

La universidad de los llanos cuenta con una granja experimental conformada por diversos lotes que son utilizados para el desarrollo de proyectos que buscan mejorar la producción y manejo de los cultivos agrícolas. Para el monitoreo de los cultivos se requiere de personal y diversos dispositivos para medir las variables ambientales y luego almacenar sus valores en un archivo de datos (Excel, Word, entre otros); generalmente estas mediciones se realizan esporádicamente. Esta labor es necesaria para analizar la información obtenida en las diversas mediciones y tomar decisiones que favorezcan la productividad del cultivo. Este proyecto se realizó con el fin de automatizar este proceso, de manera que se reduzca el trabajo por parte del hombre, se obtengan mayor cantidad de muestras en determinado tiempo, se reduzca el error de medida en cada muestra (ya que se omitirá el error humano) y se pueda monitorear el cultivo en tiempo real.

Con el fin de mejorar la toma de datos y procesamiento de información de las variables ambientales que afectan a un cultivo agrícola, se desarrolló una aplicación que junto con una red inalámbricas de sensores (WSN) obtiene las variables ambientales de un cultivo agrícola y permite la visualización y almacenamiento sus valores de manera remota en dispositivos conectados a internet. Las redes de sensores inalámbricas conocidas como WSN (por sus siglas en ingles Wireless Sensor Networks) tienen como finalidad la monitorización y control de los fenómenos que ocurren en el mundo físico [1]. Las WSN son un conjunto de pequeños dispositivos llamados nodos, interconectados entre sí de forma inalámbrica. Estos nodos son capaces de captar, procesar y transmitir información extraída de áreas físicas de observación, al resto de los nodos de la red. Estas redes son consideradas una potente línea de investigación debido a las diferentes aplicaciones en las que podemos encontrarlas hoy día, por ejemplo: monitorización de un hábitat (para determinar la población y comportamiento de animales y plantas), detección de incendios, terremotos o inundaciones, sensorización de edificios “inteligentes”,

control de tráfico, agricultura de precisión y asistencia militar o civil [2]. Además el monitoreo remoto cubre un amplio rango de aplicaciones donde los sistemas inalámbricos pueden complementar sistemas de cable reduciendo costos de cableado y permitiendo nuevos tipos de aplicaciones de medición [3].

En los próximos capítulos se explica de qué manera este proyecto pretende mejorar el proceso de toma de datos de las variables que afectan a los cultivos agrícolas utilizando la tecnología WSN y la aplicación SenVarA. A continuación se da una descripción del trabajo en la que se evidencia el contexto en el cual se va a desarrollar el proyecto, la problemática y la solución que se planteó y desarrolló; luego los objetivos propuestos, seguidamente se describe el procedimiento que se llevó a cabo para el desarrollo del sistema, los resultados obtenidos y finalmente las conclusiones.

4 DESCRIPCIÓN DEL PROBLEMA

Cuando se inició el proyecto, la Universidad de los Llanos contaba con 17 programas de pregrado, 16 especializaciones, 2 Maestrías y 1 Doctorado, que permiten ofrecer servicios educativos con gran cobertura regional y nacional y aproximadamente 5000 estudiantes y 450 docentes [4]. Para el desarrollo de las actividades de investigación en temas de agricultura y prácticas de la UNILLANOS en los pregrados, especialmente en Ingeniería Agronómica y Licenciatura en Producción Agropecuaria se tiene a disposición de docentes y alumnos la Granja Experimental, conformada por varios lotes dispuestos para el desarrollo de proyectos de tipo experimental, como son los cultivos.

Los diferentes proyectos de investigación que se llevan a cabo en cuanto a desarrollo de cultivos tradicionales como piña, hortalizas y otros, requieren que el personal a cargo realice observaciones en campo, tomando mediciones de parámetros como temperatura, humedad, Ph, niveles freáticos, vibraciones y otros factores ambientales, lo que implica un esfuerzo y tiempo del personal para obtener la medida, registrar las bitácoras en papel y alimentar hojas de

cálculo en Excel (forma esporádica), en las cuales se realiza el procesamiento de la información, que finalmente permite tomar las decisiones del caso.

Una vez realizados los análisis de la información en la coordinación de granja o en el sitio donde se tengan los equipos de cómputo, el investigador procede a aplicar medidas correctivas en el cultivo como eliminar excedentes de agua, ejecutar procedimientos de riego, aplicar abonos o nutrientes en el campo, y otros insumos necesarios para mantener en buena forma los cultivos objeto de investigación.

La anterior situación descrita evidencia una carencia de mecanismos que permitan automatizar la administración de la información necesaria para el análisis de los cultivos; de manera que se hace necesario un sistema que optimice la toma de decisiones apoyado en datos en tiempo real, la organización de la información que permita reducir los tiempos en la búsqueda y recuperación de datos estadísticos y los costos que acarrea para la UNILLANOS la duplicidad de información, la impresión de documentos y el almacenamiento de los mismos, así como la mejor administración de insumos requeridos en los cultivos señalados [5].

Para automatizar el proceso de toma de datos de las variables ambientales que afectan a los cultivos agrícolas se diseñó y desarrolló un prototipo conformado por una red de sensores inalámbricos, un sistema que captura los valores de las variables tomadas por los sensores y los almacena en una base de datos y finalmente un sistema web, el cual muestra estas variables de una forma clara y ordenada; la combinación de estos sistemas facilitará la toma de decisiones en el cultivo donde se instalen los sensores.

5 JUSTIFICACIÓN

La economía actual supone la generalización del proceso globalizador y aplicación masiva de nuevas tecnologías, las grandes empresas saben que la clave del éxito descansa en buena medida en la explotación del saber hacer

tecnológico, de la potenciación de las marcas y del desarrollo y comercialización de nuevos productos y servicios. Los cambios en el entorno hacen obsoletas las ventajas competitivas tradicionales y fuerzan a la empresa a renovarse para asegurarse no sólo su rentabilidad presente sino su supervivencia futura. En cualquier caso, la clave de la incorporación de las TICs a la empresa ha radicado en alcanzar incrementos significativos de productividad [6].

Las Tecnologías de la Información y la Comunicación han transformado la manera de trabajar y gestionar recursos. Las TIC son un elemento clave para hacer que el trabajo sea más productivo, agilizando las comunicaciones, sustentando el trabajo en equipo, gestionando las existencias, realizando análisis financieros, y promocionando diversos productos en el mercado.

Bien utilizadas, las TIC permiten a las empresas producir más cantidad, más rápido, de mejor calidad, y en menos tiempo.

La automatización y sistematización de los procesos a nivel industrial y agroindustrial, permite incorporar herramientas consideradas indispensables para la toma de decisiones basadas en la información recopilada y procesada. Esto permite reemplazar a operadores humanos en tareas que requieren precisión y son de carácter rutinario. La automatización influye en la reducción de costos, la disminución de tiempos de recolección de información y procesamiento, la realización de tareas complejas y en la toma de decisiones [7]. Con la automatización de la recolección de datos y parámetros desde un cultivo experimental se reducen los costos asociados a la contratación de personal y materiales ya que la integración de tecnología en los procesos agrícolas brindan la posibilidad de adaptar los procesos a los diferentes entornos, racionalizar los elementos y materia prima utilizada, y un aporte en la conservación ecológica [8].

La tecnología de redes inalámbricas de sensores (Wireless Sensor Networks WSN), es una de las más recientes en desarrollo, y como tecnología emergente, tiene la posibilidad de integrarse en varias áreas de aplicación y plantea una nueva forma de comunicación en redes Adhoc, las cuales son redes en las que no se tiene un nodo central, sino que todos los dispositivos

están en igualdad de condiciones, sin requerir de la clásica configuración de infraestructura [9]. En diferentes investigaciones realizadas al respecto en Instituciones Universitarias a nivel nacional e internacional [10], se han propuesto desarrollos en la incorporación de elementos de tecnología como lo es las redes de sensores inalámbricos (WSN) para la planeación y mantenimiento de un cultivo, dado que en el entorno rural se adolece de infraestructura de comunicaciones, se requiere un despliegue fácil de la implementación, un bajo consumo de energía y las redes de sensores son una adecuada elección tomando en cuenta estos aspectos. En concordancia con lo expuesto anteriormente, en la propuesta de investigación surge la pregunta ¿Será posible generar un prototipo basado en la tecnología de redes de sensores inalámbricos (WSN) y apoyado por un sistema de información web para la granja experimental de UNILLANOS como apoyo en el monitoreo de proyectos de investigación en cultivos? La implementación de un proyecto prototipo que utilice la tecnología WSN en conjunto con un Sistema de Información WEB, realizando la aplicación en la Granja experimental y que incorpora la agricultura de precisión, está enmarcada entre las tecnologías de punta y que tendrán el mayor desarrollo en los próximos años, tanto por el uso de dispositivos móviles en la vida cotidiana como el uso de nuevas tecnologías inalámbricas de acceso como el RFID, WSN [11]. El desarrollo del presente proyecto permitirá a la UNILLANOS contar con un Prototipo basado en WSN y apoyado con un sistema de información Web para la administración de datos en tiempo real sobre investigaciones en cultivos experimentales, en cultivos comerciales que permitan tomar decisiones y tener un archivo histórico de tendencias.

Así mismo en el Plan Gestión Institucional, se encuentra la estrategia 1, Gestión eficaz y eficiente de la administración académica, con el programa planteado Sistema de Información y Comunicación, como el soporte técnico en la toma de decisiones [12].

6 OBJETIVOS

6.1 Objetivo general

Desarrollar un prototipo basado en redes de sensores inalámbricos y tecnologías web como apoyo al monitoreo de variables ambientales en un cultivo.

6.2 Objetivos Específicos

- Implementar el prototipo de redes de sensores inalámbricos (WSN)
- Diseñar una aplicación que sea capaz de almacenar las variables ambientales en una base de datos MySQL.
- Diseñar e implementar un prototipo de sistema de información Web para la vigilancia de las variables en cultivos experimentales, para la utilización del usuario final.

7 MARCO TEÓRICO

Durante la última década se han desarrollado un sinnúmero de dispositivos de comunicación que permiten el envío de información tanto por vía cableada (Wired) como inalámbrica (Wireless), apoyados en el modelo de red de infraestructura. La característica de un modelo de red basado en infraestructura, es el uso de un elemento central de comunicaciones que recoge y encamina la información generada por los clientes, en este modelo los hosts no se comunican directamente entre sí, con las implicaciones que esto conlleva [9].

En los últimos años los dispositivos de comunicación han reducido su tamaño y han incrementado su capacidad de transmisión, en este contexto se encuentran los sensores que se pueden utilizar para el monitoreo y control [13].

Las redes de sensores (Wireless Network Sensor WSN) están constituidas por lo general de varios equipos de cómputo con dispositivos de transmisión de datos que se comunican entre sí generando una red Adhoc [13], con alimentación propia y un sensor que se utiliza para determinada tarea, de esta forma cada equipo (nodo) cumple la función específica y actúa como repetidor y enlace inalámbrico hasta un equipo (nodo) central que recoge los datos generados [9].

La función de cada nodo es cumplir como un elemento de forma similar de una red en infraestructura y permitir la retransmisión de datos de nodos lejanos hasta el centro de coordinación. La operación de los nodos permite involucrar desde varios a cientos de nodos recogiendo información de sensores para control de parámetros como temperatura, vibración, presión, movimiento. Estos sensores pueden estar fijos o ser móviles [14].

De esta forma una red de sensores inalámbricos está formada por varios, cientos o miles de ellos para la transmisión de diferentes datos y parámetros, estos nodos son autónomos unos de otros y están distribuidos geográficamente en un área determinada con el propósito de monitorear, con cierta capacidad de cómputo y almacenamiento y principalmente la comunicación sin requerir cableado [13].

En los nodos sensores se integran dispositivos para medir luz, temperatura, presión, humedad, etc. Debido al tamaño de estos el consumo de energía es mínimo y el costo de los equipos es reducidos, generalmente las aplicaciones de este tipo de redes son muy puntuales.

7.1 Estructura de una red de sensores

Estructura de una Red de Sensores Básicamente la estructura general de una red de sensores se compone de: [15].

- Hardware: En esta categoría se encuentra el Nodo mota y el Nodo enlace. El nodo denominado “mota” está formado por un dispositivo transmisor, un procesador, una memoria, una batería y dos o tres sensores. De esta forma la mota tiene la posibilidad de procesar datos de alguna forma, almacenar los datos propios o actuar como un buffer de paso para los datos de otras motas y

la capacidad de enviar esta información hacia otras motas o hacia el nodo central. Un elemento importante en el nodo mota es el procesador quien es el encargado de interpretar y procesar los datos para transmitirlos a otra estación. Existen diversas implementaciones de procesadores basados en tecnologías como FPGA, Microprocesadores, Microcontroladores, ARM7, Atmel AVR, Intel XCALE, PIC. La alimentación de la mota es otro elemento importante en su estructura, el consumo de energía viene dado por la carga en sensores, las comunicaciones y procesamiento, implementándose principalmente con baterías de NiCd, NiZn, Nimh, Litio-Ion.

Los sensores en la nodo mota están constituidos por elementos que pueden codificar señales físicas y transformarlas en señales eléctricas para que puedan ser transmitidas hacia los nodos centrales o la puerta de enlace.

Las comunicaciones en la mota se realizan en la banda de 433 Mhz a 2.4 Ghz, las cuales son de libre uso y acceso en el territorio Colombiano, y entre las técnicas para transmitir se encuentra la radio frecuencia (RF), vía laser o infrarrojos. Actualmente el estándar 802.15.4 ZigBee, es el tipo más general utilizado para las WSN [16].

El nodo enlace (Gateway), es el encargado de hacer la conexión de la WSN hacia una red de infraestructura tradicional, de tal forma que se pueda recoger los datos obtenidos en la WSN hacia una aplicación o Base de Datos. En términos generales este nodo tiene unas características de mayor capacidad respecto a las motas, como es capacidad de procesamiento, almacenamiento, memoria, capacidad de comunicación y posee una fuente de energía constante [14].

- Software: En la estructura de la mota se encuentra el Sistema operativo utilizado en el nodo y el lenguaje de programación utilizado. El sistema operativo de la mota es el encargado de realizar operaciones de procesamiento de la información, y que permiten controlar las principales funciones del micro controlador y comunicaciones de la mota [17].

7.2 Tecnologías Disponibles

7.2.1 JAVA EE (Java Enterprise Edition)

Es una plataforma de programación propiedad de Oracle Corporation [18], conocida como Java Empresarial, que define un estándar para el desarrollo y ejecución de aplicaciones de software distribuidas, basadas en componentes que corren sobre un servidor de aplicaciones, utilizando un modelo de múltiples capas orientado a la construcción de sistemas empresariales, añadiendo las capacidades necesarias para proveer una plataforma Java completa, estable, segura y rápida, reduciendo significativamente el costo y complejidad. [19]

Una aplicación empresarial Java EE típica presenta una solución concreta basada en la arquitectura que muestra la Ilustración 1:

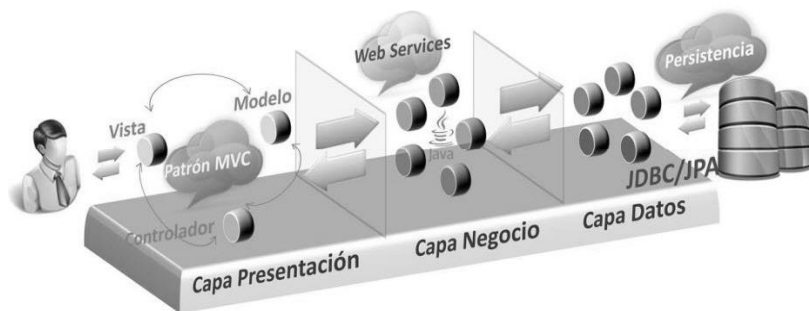


Ilustración 1: Arquitectura Multicapas Java EE [18]

Como se ve en la **Ilustración 1**, los componentes se despliegan dentro de lo que se denominan capas. Las capas son elementos que proporcionan un entorno entre los componentes y su funcionalidad.

La plataforma era conocida como Java 2 Platform Enterprise Edition o J2EE hasta la versión 1.4, a partir de la cual se definió sencillamente Java Platform Enterprise Edition o Java EE. A lo largo del tiempo fue extendiendo sus servicios y funcionalidad con el desarrollo de diferentes versiones:

- J2EE 1.2 (12 de Diciembre 1999)
- J2EE 1.3 (24 de Septiembre de 2001)
- J2EE 1.4 (11 de Noviembre de 20013)
- Java EE 5 (11 de Mayo de 2006)
- Java EE 6 (10 de Diciembre de 2009)
- Java EE 7 (12 de Junio de 2013)

- Java EE 8 (JSRs aprobada el 22 Sep, 2014, La liberación final se espera para finales del 2017)

Este proyecto será desarrollado con la última versión Java EE 7 que puede ser descargada de:

<http://www.oracle.com/technetwork/java/javaee/downloads/index.html>

Java EE se construye sobre la base del estándar del lenguaje Java (Java SE) [20], añadiendo funcionalidades y servicios necesarios para crear una aplicación empresarial con cumple con los atributos de escalabilidad, portabilidad e integración con tecnologías anteriores. Además el servidor de aplicaciones permite manejar transacciones, seguridad, concurrencia y administración de los componentes.

A continuación se presentan los componentes utilizados en este proyecto y las API asociados a dichos servicios. Todas las APIs y su especificación en detalle se pueden consultar en el siguiente enlace: <http://docs.oracle.com/javaee/7/api/overview-summary.html>.

7.2.2 JSF (Java Server Faces)

El api JSF (*javax.faces.**) es un framework MVC (Modelo-Vista Controlador) que proporciona un entorno para simplificar el desarrollo de interfaces de usuario en aplicaciones Java EE RIA (*Rich Internet Applications*), proporcionando un conjunto de componentes en forma de etiquetas definidas en páginas XHTML mediante el framework Facelets. Facelets se define en la especificación 2 de JSF como un elemento fundamental de JSF que proporciona características de plantillas y de creación de componentes compuestos.

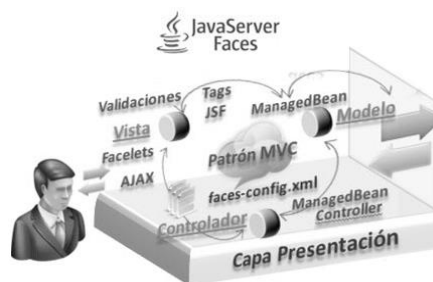


Ilustración 2: API Java Server Faces (Capa de Presentación) [20]

JSF utiliza las páginas Facelets como vista, objetos Javabeen como modelos y métodos de esos objetos como controladores. En la **Ilustración 2** se puede apreciar cómo se implementa este patrón. El servlet FacesServlet realiza toda la tediosa tarea de procesar las peticiones HTTP, obtener los datos de entrada, validarlos y convertirlos, colocarlos en los objetos del modelo, invocar las acciones del controlador y renderizar la respuesta utilizando el árbol de componentes. La forma en que estos componentes están distribuidos y se comunican, se aprecia en la **Ilustración 3**.

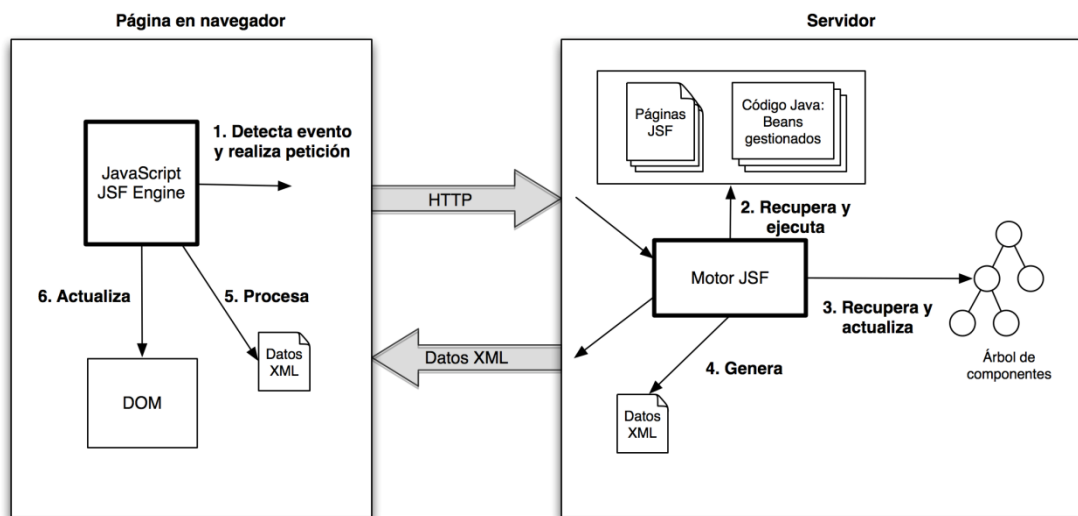


Ilustración 3: Proceso de actualización de una página JSF [21]

JSF proporciona las siguientes características destacables:

- Definición de las interfaces de usuario mediante vistas que agrupan componentes gráficos.
- Conexión de los componentes gráficos con los datos de la aplicación mediante los denominados beans gestionados.
- Conversión de datos y validación automática de la entrada del usuario.
- Navegación entre vistas.
- Extensiones para desarrollar diferentes componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entradas, definir filtros de navegación de las páginas y dar soporte para internacionalización y accesibilidad, por mencionar algunas de las más relevantes.

- Modelo estándar de comunicación AJAX entre la vista y el servidor.
- Aspectos de seguridad como compatibilidad de SSL¹ que previene inyección de código malicioso XSS². [21]

7.2.3 HTML 5

Es la última versión de la norma que define HTML(HyperText Markup Language). El término representa dos conceptos diferentes, una nueva versión del *lenguaje* HTML, con nuevos elementos, atributos y comportamientos, y un conjunto más amplio de tecnologías que permite crear sitios web y aplicaciones más diversas y de gran alcance. [22]

Principales características:

- **Semántica:** permite describir con mayor precisión cuál es su contenido.
- **Conectividad:** permite comunicarse con el servidor de formas nuevas e innovadoras, como por ejemplo Web Sockets³.
- **Desconectado y almacenamiento:** permite a páginas web almacenar datos localmente, en el lado del cliente y opera fuera de línea de manera más eficiente (el cache de la aplicación)
- **Multimedia:** permite la manipulación eficiente de material multimedia con la utilización de etiquetas como <audio> y <video>
- **Gráficos en 2D/3D:** permite una gama mucho más amplia de opciones de presentación, como por ejemplo el elemento <canvas> para dibujar todo tipo de gráficos.
- **Rendimiento e integración:** proporciona una mayor optimización de la velocidad y un mejor uso del hardware del equipo
- **Dispositivos de acceso:** admite el uso de varios dispositivos de entrada y salida (webcam, pantallas táctiles, etc.)
- **Styling:** deja a los desarrolladores escribir temas más sofisticados.

¹ SSL (Secure Sockets Layer) Son protocolos criptográficos que proporcionan conexiones seguras por una red, comúnmente internet

² XSS (Cross-site scripting) Es un tipo de inseguridad informática típico de aplicaciones web, que permite inyectar código HTML y JavaScript en las vistas de usuario

³ Permite crear una conexión permanente entre la página y el servidor e intercambiar datos no HTML a través de ese medio

7.2.4 Base de datos

Una base de datos es una colección de archivos relacionados que permite el manejo de la información de alguna compañía. Cada uno de dichos archivos puede ser visto como una colección de registros y cada registro está compuesto de una colección de campos. Cada uno de los campos de cada registro permite llevar información de algún atributo de una entidad del mundo real. [23]

Una base de datos es administrada a través de un gestor de base de datos, que es el grupo de herramientas con las que se puede crear la base de datos, gestionar las tablas, los roles, registros y relaciones entre éstos. El gestor de bases de datos que se utilizará en este proyecto es MySQL.

7.2.5 Procces Dasboard

El Software Process Dashboard es una iniciativa de código abierto para crear una PSP (SM) / TSP (SM) herramienta de apoyo. Creemos que el PSP y TSP son tecnologías notables que pueden cambiar la faz de la industria del software, y compartimos el celo del SEI para promover su uso generalizado. Creemos que una poderosa herramienta de apoyo a libre disposición podría ayudar a eliminar una de las barreras más importantes para la adopción PSP / TSP. Por lo tanto, nuestro objetivo es desarrollar una herramienta de clase mundial en el marco del modelo de código abierto, y distribuirlo libremente a cualquier persona que utilice la PSP y / o TSP. Creemos que esto es lo menos que podemos hacer para agradecer a la SEI para el desarrollo y la distribución de estos procesos notables. [23]

7.2.6 JavaScript

JavaScript es un lenguaje de programación interpretado con una sintaxis semejante a la del lenguaje Java, que permite a los desarrolladores crear acciones en los sitios web y definir interactividades con el usuario, no requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos. [24]

7.2.7 CSS

Es un lenguaje de hojas de estilos creado para definir el aspecto o presentación de los sitios web definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web enriquecidas. [25]

7.2.8 Bootstrap

Un entorno de desarrollo con una serie de recursos que simplifican el desarrollo de un proyecto web, Bootstrap fue creado y desarrollado por desarrolladores de twitter para unificar sus criterios de trabajo internos, el resultado fue tan bueno que decidieron compartirlo bajo licencia Open Source, la acogida del proyecto fue tan buena que al final la comunidad de bootstrap en Git Hub es de las más numerosas. [26]

7.2.9 XAMMP

Es una distribución para instalar de forma fácil y de manera local software para el uso en servidores, en este caso MySQL, PHP y Perl, adicional instala de manera predeterminada una plataforma apache para lograr emular proyectos de manera local y en cualquier sistema operativo.

7.2.10 JPA (Java Persistence Api)

El api JPA (*javax.persistence.**) es el estándar creado como parte de la plataforma Java EE para la capa de persistencia. Es una especificación basada fundamentalmente en las experiencias que los distintos fabricantes han tenido con los framework ORM como Hibernate, Toplink y JDO, por esta razón todos estos soportan el uso del api JPA.

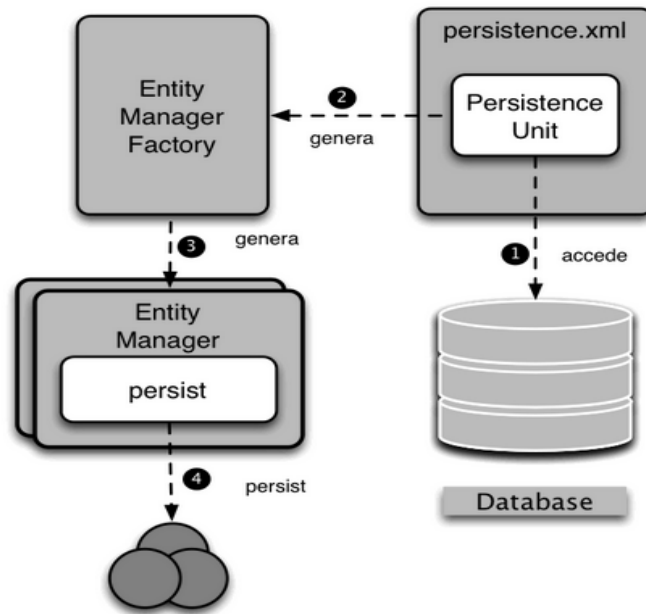


Ilustración 4: Relación componentes JPA [29]

La **Ilustración 4** muestra la distribución de la especificación JPA, que es una abstracción del Api JDBC para realizar operaciones sobre la base de datos realizando una conversión entre las entidades Java y los registros en las tablas por medio de un mapeo objeto-relacional (ORM), que es una técnica de programación para comunicar datos entre bases de datos relacionales y lenguajes de programación orientados a objetos, como Java. El ORM está basado en POJO (Plain Old Java Objects) y se realiza mediante simples anotaciones en las propias clases de entidad, lo que quiere decir que el diseño y uso del modelo de datos (el dominio) en nuestra aplicación se hará con instancias (objetos) Java convencionales que representan un registro(fila) en la base de datos.

A continuación se explican cuáles son los componentes fundamentales de la especificación JPA:

- **Persistence.xml:** archivo que contiene todos los parámetros necesarios para conectarnos a un servidor de base de datos (local / JTA⁴) y define el conjunto de entidades que vamos a gestionar.

⁴ JTA (Java Transaction API) proporciona una interfaz estándar para la realización de transacciones entre el manejador y un servidor de aplicaciones

- **Persistence:** clase que se encarga de leer el archivo persistence.xml y seleccionar una unidad de persistencia concreta con la cual quedaremos enlazados a una base de datos.
- **EntityManagerFactory:** clase que tiene rol de fábrica se encarga de crear Entity Managers que serán los que en última instancia gestionen la persistencia de los distintos objetos contra la base de datos
- **EntityManager:** clase fundamental de JPA que se encarga de todas las operaciones que afectan los objetos que deseamos persistir (seleccionar, insertar, actualizar y borrar objetos). Se encarga además de mantener el estado de los objetos que tenemos en memoria con los cambios que en ellos se han producido.
- **EntityTransaction:** clase que genera una abstracción sobre el concepto general de transacción a nivel de base de datos. [27]

Características:

- Abstrae el proveedor de persistencias (El gestor de la base de datos)
- Elimina la necesidad de usar código SQL, solo se realizan consultas sobre los objetos. A su vez tiene su propio lenguaje de consultas JPQL para consultas más complejas
- Puede crear relaciones ManyToOne, OneToOne, OneToMany y ManyToMany entre diferentes entidades solo con anotaciones.
- Controla la transaccionalidad (Utilizando JTA⁴)

7.2.11 MAVEN

Maven es una herramienta open source de línea de comandos para gestionar el ciclo de vida de un proyecto de software, desde la creación del proyecto en un lenguaje específico, hasta la generación de los binarios o ejecutables que puedan distribuirse con el proyecto para la producción.

Para ello, en Maven se definen diferentes ciclos de construcción (build) del software con una serie de etapas diferenciadas. Por ejemplo el ciclo por defecto tiene las etapas de:



Ilustración 5: Logo maven tomado de [30]

- **Validación** (validate): Validar que el proyecto es correcto.
- **Compilación** (compile).
- **Test** (test): Probar el código fuente usando un framework de pruebas unitarias.
- **Empaquetar** (package): Empaquetar el código compilado y transformarlo en algún formato tipo .jar o .war.
- **Pruebas de integración** (integration-test): Procesar y desplegar el código en algún entorno donde se puedan ejecutar las pruebas de integración.
- **Verificar**: que el código empaquetado es válido y cumple los criterios de calidad (verify).
- **Instalar**: el código empaquetado en el repositorio local de Maven, para usarlo como dependencia de otros proyectos (install).
- **Desplegar**: el código a un entorno (deploy).
- Para poder llevar a cabo alguna de estas fases anteriormente mencionadas en nuestro proyecto, tan solo se tiene que ejecutar el comando mvn y el nombre de la fase.

Por otra parte, con Maven la gestión de dependencias entre módulos y distintas versiones de librerías se hace muy sencilla. En este caso, solo se tiene que indicar los módulos que componen el proyecto, o qué librerías utiliza el software que estamos desarrollando en un archivo de configuración de Maven del proyecto llamado POM el cual se encarga de realizar las descargas automáticamente de un repositorio Maven central donde se encuentran la mayoría de librerías que se utilizan en los desarrollos de software.

A continuación se describen algunos de los componentes de Maven:

- **Pom:** un pom (project object model) es la unidad de trabajo básica de Maven. Se trata de un archivo XML que contiene información y detalles de configuración usados por Maven para construir nuestro proyecto.
- **Arquetipo:** generar la estructura de ficheros y directorios necesarios para nuestro proyecto, gestionando las librerías que le indiquemos así como las dependencias transitivas que se generen con otras librerías, los desarrolladores lo utilizan como base para escribir y organizar el código de la aplicación.
- **Plugin:** es un componente de software que agrega funcionalidad. En el caso de Maven cada plugin agrega nueva lógica al sistema. Los plugins tienen varios goals.
- **Goal:** una tarea específica que se ejecutará de manera autónoma.
- **Repositorio:** es una colección de artefactos de proyectos almacenados en una estructura y formato que pueden ser fácilmente entendidos por Maven.
- **Ciclo de vida:** es una secuencia organizada de fases que se encargarán de ejecutar unas determinadas tareas (goals). [28]

7.2.12 JBOSS

Jboss es un servidor de aplicaciones basado en el estándar JavaEE de código abierto implementado en java que puede ser utilizado en cualquier sistema operativo en el que esté disponible la máquina virtual de Java.



Ilustración 6: Logo Jboss tomado de [32]

Jboss es una plataforma de middleware⁵ para el desarrollo y despliegue de software basado en componentes, ideal para sistemas distribuidos por su extensibilidad y modularidad. Funciona como contenedor de aplicaciones que proporciona servicios de aplicación a las computadoras cliente, generalmente gestiona la mayor parte de las funciones de lógica de negocio y de acceso a los datos de la aplicación. Los principales beneficios de la aplicación de la tecnología jboss son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones, frente a la tradicional estructura en dos capas de un servidor web, un servidor de aplicaciones proporciona una estructura en tres capas que permite estructurar nuestro sistema de forma más eficiente.

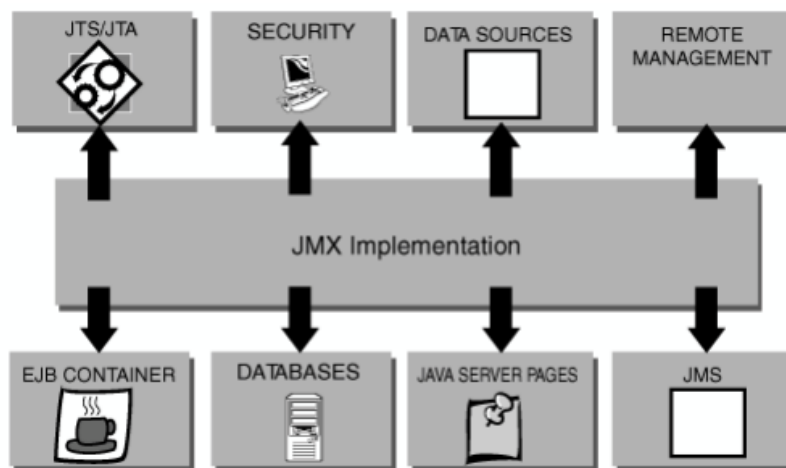


Ilustración 7: Integración de los módulos de Jboss usando JXM [33]

El servidor de aplicaciones JBoss está dividido en módulos e implementado usando plugins basados en componentes, esto es posible gracias a que su microkernel que está basado en JMX (Java Management Extension), que es la tecnología que define la arquitectura de gestión, la API (Application Programming Interface), los patrones de diseño, y los servicios para la monitorización/administración de aplicaciones basadas en Java.

El éxito de las aplicaciones de código abierto de JavaEE es que usen JXM. JXM es una buena herramienta para la integración de software ya que provee una columna vertebral que permite integrar módulos, contenedores y plug-ins. La ¡Error! No se encuentra el origen de la referencia. muestra cómo se integran l

⁵ Software que reside entre una aplicación y el sistema operativo para realizar interacciones entre ellos

os diferentes módulos y el rol de JMX. El grado de modularidad beneficia directamente a los desarrolladores puesto que permite conectar o desconectar componentes de acuerdo a las necesidades. Por ejemplo, si ya no es necesario manejar EJB en la aplicación, simplemente se “desconecta” esta característica y se elimina del servidor. Otro ejemplo es la libertad de poder descargar todo el mapeo de la base de datos objeto relacional dentro del contenedor del servidor.

Algunos de los servicios ofrecidos:

- **Generación de HTML:** incorporar generación dinámica de contenido (HTML, XHTML, XML, etc.), para enviar al cliente.
- **Trabajo con bases de datos:** soporte hibernate que contiene objetos que facilitan el acceso a bases de datos, ocupándose de gestionar las conexiones y proporcionando un acceso uniforme. Otros objetos se encargan de la gestión de transacciones englobando diversas sentencias y ocupándose de los commit o rollback.
- **Funcionamiento multiproceso o multihilo:** el servidor es el responsable de tener funcionando un número de hilos o procesos que atiendan a distintas peticiones.
- **Sesiones:** Jboss provee de persistencia a los datos del usuario mediante objetos de sesión (session). Elimina la necesidad de incluir código en las aplicaciones para diferenciar las peticiones de distintos usuarios.
- **Lógica de negocio:** Soporta la especificación EJB 3.0 donde la lógica de negocio propia de cada aplicación es encapsulada en componentes. A cada uno de ellos se le asignan mecanismos propios de seguridad, gestión de transacciones. etc.
- **Seguridad:** posee características de seguridad que den soporte a aplicaciones seguras. Los clientes deben autenticarse contra al servidor, y este es el responsable de darles acceso a sus diferentes componentes, como puede ser una base de datos.

Un servidor Jboss se puede configurar en diferentes modos:

- **Minimal:** Servicios mínimos para que funcione Jboss

- **Default:** Tiene los servicios usados más frecuentemente
- **All:** Arranca todos los servicios disponibles
- **Standar:** Configuración de servicios certificado JavaEE
- **Web:** Configuración ligera de contenedor web

8 METODOLOGÍA

Este trabajo sigue los lineamientos del marco de referencia para la investigación en sistemas de información propuesto por Hevner A y otros [29], que hace parte de la investigación en las ciencias de lo artificial, tal como lo plantea Herbert A Simon [30]. Consecuentemente, la metodología general a aplicar es la de la Investigación del Diseño descrita por Vaishnavi y Kuechler [31].

La aplicación de la Metodología de la Investigación de Diseño (DSRM por sus siglas en inglés Design Science Research Metodology) se realizará siguiendo el modelo de proceso sugerido por Peffers y otros [32], que se muestra en la **Ilustración 8**.

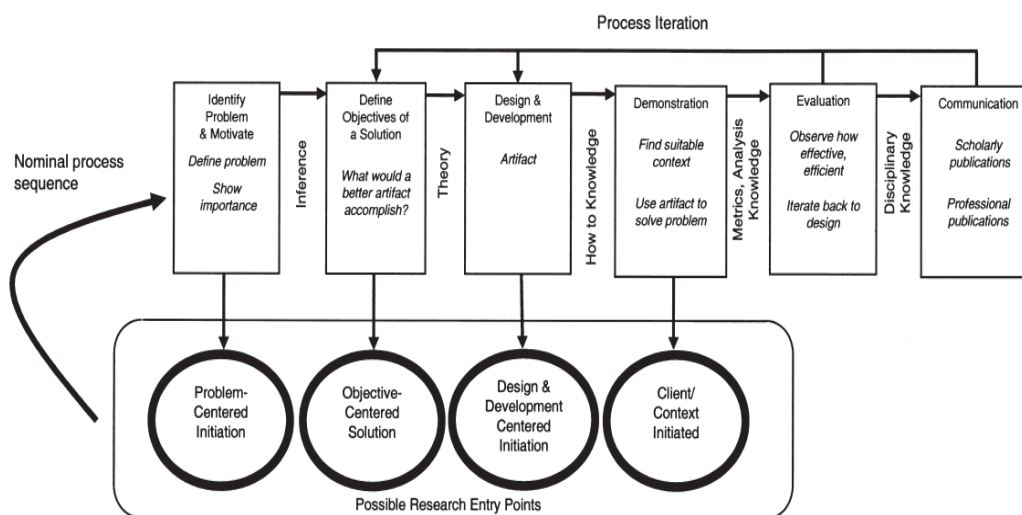


Ilustración 8. Proceso de aplicación de la metodología de DSRM

DSRM abarca el proceso de desarrollo de software desde las etapas de identificación y definición del problema, hasta la publicación profesional de

resultados, incluyendo un proceso de diseño y desarrollo de artefactos para el que se usará PSP, ya que provee los mecanismos necesarios para la construcción de los artefactos de software.

8.1 PSP-TSP.

El proceso de desarrollo de este proyecto se llevara a cabo usando la metodología PSP-TSP, que entrega a los ingenieros un marco de referencia de disciplina personal para mejorar su trabajo y realizarlo con alta calidad, con el propósito de ayudarlos a aprender y practicar aquellos métodos para producir software que son más efectivos para ellos. Entendiendo como principio fundamental que con un proceso de calidad los productos derivados de éste serán también de calidad. [33]

PSP - Personal Software Process - (Proceso de desarrollo de Software en forma Personal) es un esquema de trabajo definido y medible que ayuda a los desarrolladores en la planificación y medición de su trabajo, generando productos de alta calidad. La metodología señala cómo administrar la calidad del producto de software y cómo formular metas alcanzables. Con los datos que se registran en diversos formularios, el desarrollador puede predecir en forma más acertada el tamaño del software (medidas en LOC- Lines Of Code- líneas de código) y al conocer su productividad (LOC/hora) puede estimar en forma más precisa el tiempo que tardará en su desarrollo y, por consiguiente, justificar la planificación de su proyecto.

Los scripts de proceso definen los pasos de cada parte del proceso, los logs y formularios proveen templates para registrar y almacenar datos, y los standards guían a los desarrolladores a mientras hacen el trabajo **Ilustración 9.**

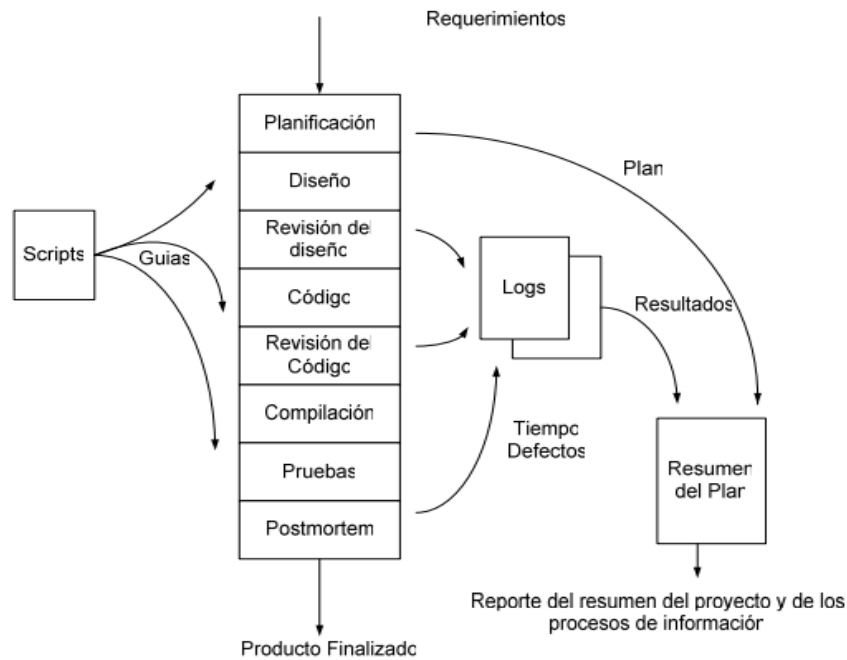


Ilustración 9. Flujo del proceso PSP por [34].

La metodología Personal Software Process (PSP) es creada por el Instituto de Ingeniería de software (SEI) de la Universidad Carnegie Mellon y consiste en un conjunto de métodos que muestran cómo planificar, medir y administrar el trabajo individual de un profesional miembro de un equipo de desarrollo de software [33]. La metodología se compone de siete versiones, cada una introduce nuevos elementos que son registrados en diversos formularios.

8.2 Niveles en PSP-TSP

PSP 0. En el primer nivel, los desarrolladores registran el tiempo de desarrollo y los errores durante cada fase de desarrollo (planeación, diseño, codificación, compilación y prueba). Estas medidas se utilizan en el análisis y planificación del proceso y como base para mejorar estimaciones futuras.

PSP 0.1. Este nivel agrega medidas para calcular el tamaño real del código y un registro donde se consignan propuestas para el mejoramiento del proceso (formulario usado para registrar los problemas encontrados durante el proceso de desarrollo del software y las propuestas para su solución).

PSP 1. En este nivel se introduce el método PROBE (PROxy-Based Estimating) el cual usa datos históricos para estimar el tamaño del código

empleando una regresión simple. Este método también permite determinar la precisión de la estimación basándose en la aplicación de la distribución t-student con 70% y 90% de intervalos de predicción. Adicionalmente, en este nivel se agrega un formulario para registrar un informe de la planificación de la etapa prueba.

PSP 1.1. Este nivel agrega estimación de recursos y planeación de las tareas para el desarrollo del proyecto. Por lo general, es difícil llevar control sobre el avance de los proyectos debido a que las tareas se realizan en un orden diferente a lo planificado. La planeación de las tareas permite, además de organizar el trabajo, identificar la importancia o peso de las tareas (lo que en diagramas de PERT se conoce como actividades críticas), para que el atraso en alguna de ellas no signifique un retraso en la duración total del proyecto.

PSP 2. En este nivel se introduce al ciclo de desarrollo las etapas de revisión de la fase de diseño y revisión de la fase de codificación, como también, cálculos de la eficiencia en la remoción de errores (defectos). El desarrollador deberá realizar una lista personal para chequear el trabajo realizado (checklist) en diseño y codificación, tomando como antecedente las anotaciones efectuadas en el formulario de registro de errores (defectos).

PSP 2.1. Este nivel enseña técnicas de especificación de diseño (especificación funcional, especificación de estado y especificación lógica) y formas de prevenir errores (defectos), como por ejemplo el escenario operacional.

PSP 3. En este último nivel o versión de PSP se cubren técnicas de verificación de diseño. [35].

8.3 Nivel a utilizar en este proyecto.

En este proyecto se utilizará la metodología en su nivel PSP 0, debido a la poca experiencia de los desarrolladores en el uso de la tecnología y a la ausencia de registros históricos de desempeño (tiempo, errores, tamaño) como los requeridos por los niveles siguientes. Tal como lo establece PSP 0 solo se tendrá en cuenta el tiempo de desarrollo como métrica del proceso **Ilustración 10.**



Ilustración 10. Estructura y Flujo de los guiones de proceso

8.4 Fases de la metodología.

A continuación se ilustra, a través de los scripts o guías de la metodología, las diferentes fases a abarcar en el desarrollo de los artefactos de software:

8.4.1 Fase de Planificación.

PSP0 Guión de la Planificación

Propósito	Guiar el proceso del plan del PSP	
Criterios de entrada	<ul style="list-style-type: none"> - Descripción del problema - Formulario de Resumen del Plan de Proyecto - Log de registro del tiempo 	
Paso	Actividades	Descripción
1	Requisitos del Programa	<ul style="list-style-type: none"> - Producir u obtener una declaración de requisitos del programa. - Asegurar que los requisitos son claros y no ambiguos - Resolver cualquier pregunta.
2	Estimación del Recurso	<ul style="list-style-type: none"> - Hacer la mejor estimación del tiempo requerido para desarrollar este programa - Ingresar los datos del tiempo planificado en el Formulario de Resumen del Plan del Proyecto
Criterios de Salida	<ul style="list-style-type: none"> - Declaración de requisitos documentados - El formulario de Resumen del Plan de Proyecto completado con los datos de tiempo de desarrollo estimado - Log de registro del Tiempo completado 	

Ilustración 11. Planificación PSP-TSP [36]

8.4.2 Fase de Desarrollo.

PSP0 Guión de Desarrollo

Propósito	Para guiar el desarrollo de pequeños programas	
Criterios de Entrada	<ul style="list-style-type: none"> - Declaración de requisitos - Formulario resumen del Plan del Proyecto con el tiempo estimado para el desarrollo del programa - Logs de registro de Tiempo y Defectos - Estándar del tipo de defectos 	
Paso	Actividades	Descripción
1	Diseño	<ul style="list-style-type: none"> - Revisar los requisitos y producir un diseño para lograrlos. - Registrar en el log de registro de Defectos cualquier defecto de requisitos encontrado - Registrar el tiempo en el log de registro de Tiempos.
2	Código	<ul style="list-style-type: none"> - Implementar el diseño. - Registrar en el log de Registro de Defectos cualquier defecto de requisitos o de diseño encontrados. - Registrar el tiempo en el log de Registro de Tiempo.
3	Compilación	<ul style="list-style-type: none"> - Compilar el programa hasta que se encuentre libre de errores - Reparar todos los defectos encontrados. - Registrar los defectos en el log de Registro de Defectos - Registrar el tiempo en el log de Registro de Tiempos.
4	Prueba	<ul style="list-style-type: none"> - Probar hasta que las pruebas se ejecuten sin errores - Reparar todos los defectos encontrados. - Registrar los defectos en el log de Registro de Defectos - Registrar el tiempo en el log de Registro de Tiempos.
Criterios de Salida	<ul style="list-style-type: none"> - Un programa probado a profundidad - Logs de Registro de Defectos y de Tiempos completado. 	

Ilustración 12. Desarrollo PSP-TSP [36]

8.4.3 Fase de PostMortem

PSP0 Guión del Postmortem

Propósito	Guiar el proceso Postmortem del PSP	
Criterios de Entrada	<ul style="list-style-type: none"> - Descripción del problema y declaración de requisitos - Formulario del resumen del Plan del proyecto con los datos del tiempo de desarrollo - Logs de registro de Defectos y Tiempos completados - Un programa probado y ejecutado 	
Paso	Actividades	Descripción
1	Registro de Defectos	<ul style="list-style-type: none"> - Revisar el Resumen del Plan de Proyecto para verificar que todos los defectos encontrados en cada fase fueron registrados - Registrar cualquier defecto omitido.
2	Consistencia de los datos defectuosos	<ul style="list-style-type: none"> - Revisar que los datos de cada defecto en el log de Registro de Defectos sea preciso y completo. - Verificar que el número de defectos introducidos y removidos por fase son razonables y correctos. - Corregir cualquier defecto omitido o incorrecto.
3	Tiempo	<ul style="list-style-type: none"> - Revisar la completitud del log de Registro de Tiempos por errores u omisiones. - Corregir cualquier dato de tiempo incompleto u omitido.
Criterios de Salida	<ul style="list-style-type: none"> - Programa probado completamente - Formulario del Resumen del Plan del Proyecto completado - Logs de Registro de Defectos y de Tiempos completado. 	

Ilustración 13. Postmortem PSP-TSP [36]

9 APLICACIÓN DE LA METODOLOGÍA

Para el desarrollo del proyecto en gran medida se procuró seguir paso a paso las fases explicadas anteriormente, Se desarrollaron procedimientos diseñados, tareas medidas, se estimaron costos y tiempos con la herramienta Process Dashboard que es una iniciativa de código abierto para facilitar el trabajo de la metodología de desarrollo implementada PSP.

9.1 RESULTADOS

Como resultado, después de aplicar la metodología se construyeron diversos artefactos entre estos:

Modelo del negocio

9.1.1 Glosario

El documento glosario es una representación de algunos de los términos manejados a lo largo del desarrollo de la aplicación, que tiene como propósito definir con exactitud y sin ambigüedad la terminología manejada de tal modo que se cree un lenguaje común entre el cliente y el desarrollador, a continuación podemos ver en la Ilustración 14 algunos de los conceptos tratados, para ver en detalle todas las definiciones ver el documento glosario anexo.

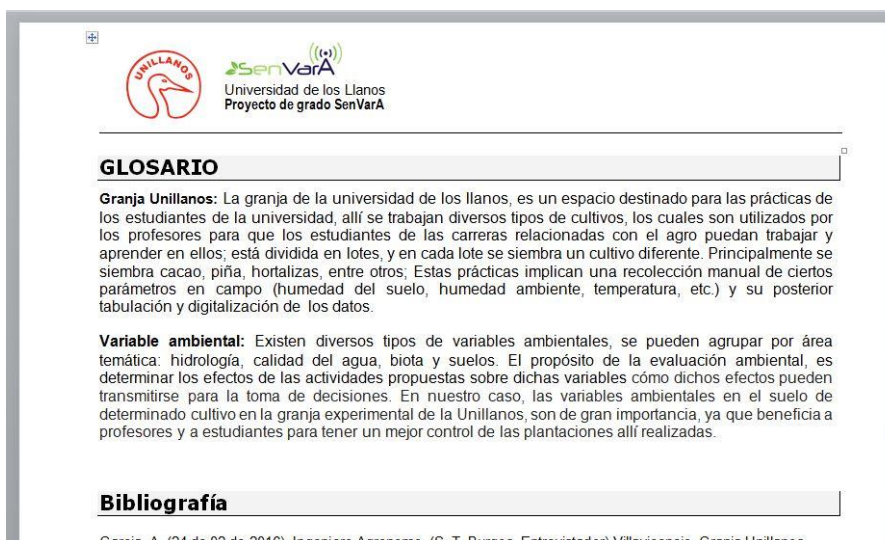


Ilustración 14: Imagen tomada del documento glosario anexo

9.1.2 Modelo conceptual

El siguiente artefacto generado fue el diagrama conceptual en el cual se organizó y representó gráficamente la jerarquía y relación que existe entre la Universidad de los Llanos y su granja, los principales actores y cómo interactúan unos con otros; la Ilustración 15 muestra parte del diagrama, el cual se puede ampliar en detalle en el diagrama conceptual anexo.

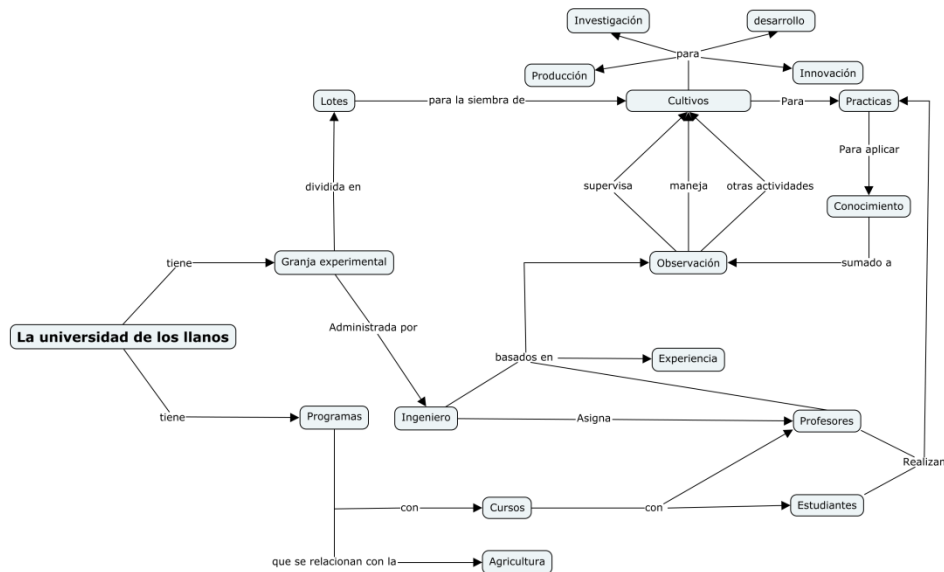


Ilustración 15: Diagrama Conceptual General

Procesos de negocio AS-IS

9.1.3 Modelo de Procesos BPMN

Tenemos el modelo BPMN (Notación para el Modelo de Procesos de Negocio) Como artefacto fundamental para la representación del modelo de dominio, en el que se describe el flujo de actividades que se ejecutan en el proceso y los participantes involucrados.

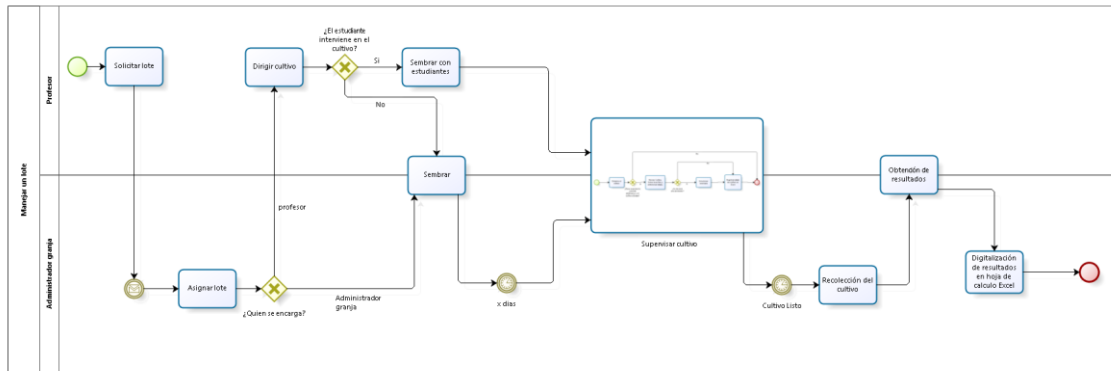
Usuarios involucrados, en la **ilustración 16** se pueden identificar los siguientes actores que interactúan en el proceso:

- I. Administrador Granja
- II. Profesor

- Fases identificadas

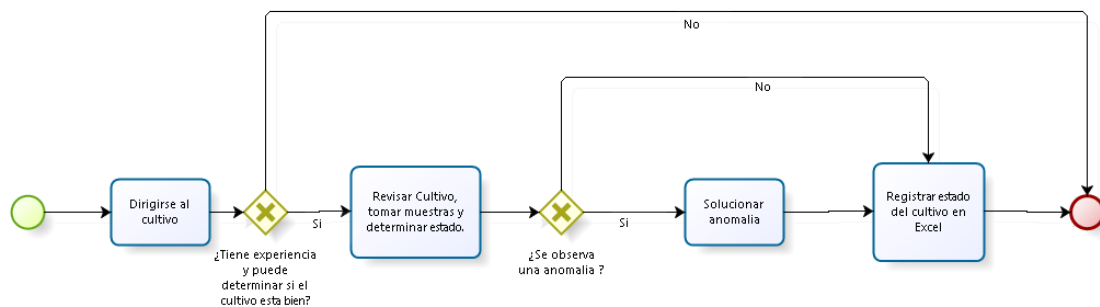
En la ilustración 17 se muestra el flujo de actividades del proceso en el cual solo se encontró una fase la cual se denominó.

- Supervisar cultivo



Powered by
bizagi
Modeler

Ilustración 16: Modelo BPMN del proceso de manejo de la granja de la universidad de los llanos



Powered by
bizagi
Modeler

Ilustración 17: Actividades susceptibles de ser sistematizadas. "Supervisar Cultivo"

9.1.4 Levantamiento de requerimientos

9.1.4.1 Requerimientos Funcionales

Los casos de uso representan los pasos o actividades que deben realizar los actores para llevar a cabo los procesos en el sistema de información, definiéndose así la funcionalidad del sistema. Estos requerimientos se establecen identificando los actores y las acciones.

- **Actores**

Los actores son las agrupaciones de los diferentes usuarios del sistema en roles, fueron descubiertos en la fase de definición de objetivos.

En el modelado de los requerimientos funcionales deben identificarse inicialmente los actores, que para SenVarA únicamente es el Usuario:

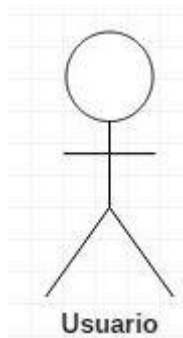


Ilustración 18: Usuario de SenVarA – Web

- **Caso de uso**

A continuación se presenta el caso de uso general, que busca describir el sistema, el cual le permite al usuario generar las gráficas, tablas y además generar reportes.

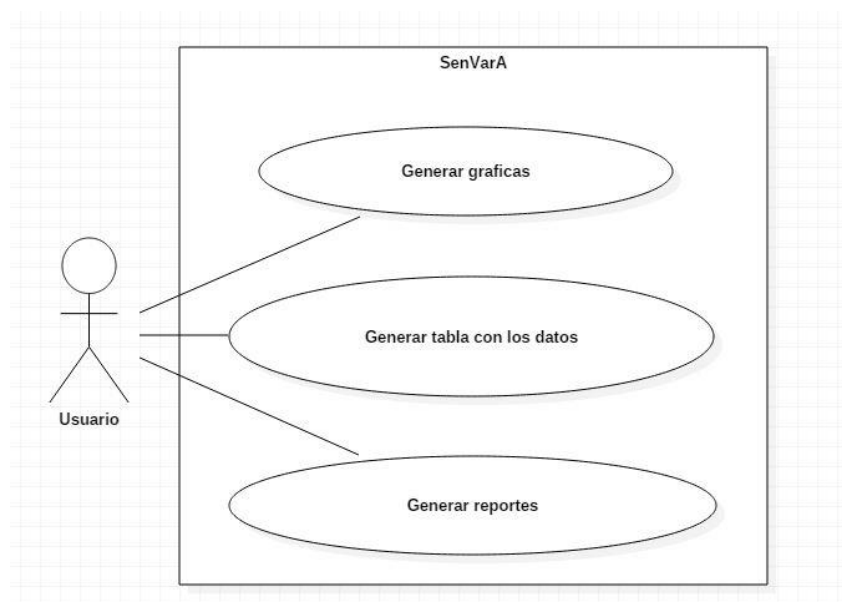


Ilustración 19: Caso de uso general

9.1.4.2 Especificación de caso de uso

9.1.4.2.1 Generar graficas

Proyecto	SenVarA											
Modulo	Principal											
No	1											
Nombre	CD-CdU1-Generar-Graficas											
Descripción	Cuando un usuario digita la información necesaria para obtener una vista de la información mediante gráficas.											
Actores	Primario : Usuario											
Fase	Levantamiento											
Guión	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. Ingresar a Sistema de Información</td> <td></td> </tr> <tr> <td>2. Ingresar los rangos de fechas a consultar información.</td> <td></td> </tr> <tr> <td></td> <td>3. Validar que las fechas ingresadas estén correctas.</td> </tr> <tr> <td></td> <td>4. Generar graficas de los valores obtenidos a través de los sensores.</td> </tr> </tbody> </table>		Actor	Sistema	1. Ingresar a Sistema de Información		2. Ingresar los rangos de fechas a consultar información.			3. Validar que las fechas ingresadas estén correctas.		4. Generar graficas de los valores obtenidos a través de los sensores.
Actor	Sistema											
1. Ingresar a Sistema de Información												
2. Ingresar los rangos de fechas a consultar información.												
	3. Validar que las fechas ingresadas estén correctas.											
	4. Generar graficas de los valores obtenidos a través de los sensores.											
Excepciones	5. Cuando la fecha inicial es superior a la fecha final (en los rangos de fechas solicitados)											
	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td>5.1 Muestra mensaje "Fecha inicial superior a la final, inténtelo de</td> </tr> </tbody> </table>		Actor	Sistema		5.1 Muestra mensaje "Fecha inicial superior a la final, inténtelo de						
Actor	Sistema											
	5.1 Muestra mensaje "Fecha inicial superior a la final, inténtelo de											

		nuevo”
		5.2. Termina
	6. Cuando no se ingresa un rango de fechas.	
	Actor	Sistema
		6.1 Muestra mensaje “Se debe ingresar un rango de fecha, por favor seleccione la fecha inicial y la fecha final”
		6.2. Termina
	7. Cuando no hay datos en el rango de fechas.	
	Actor	Sistema
		7.1 Muestra mensaje “No hay datos en ese rango de fechas”
		7.2. Termina
Casos de uso relacionados		
Documentos relacionados		
Otros Requerimientos	Debe funcionar en Internet Explorer 7 o superior, Firefox 3.5 o superior. Se recomienda el uso del navegador Google Chrome para mejor visualización de la aplicación.	
Prototipo de Pantallas		
Control de Cambios		

Versión	Autores	Fecha	Comentario
1	Stiven Toledo	Septiembre/2016	Versión Inicial

9.1.4.2.2 Generar tabla con todos los datos.

Proyecto	SenVarA										
Modulo	Principal										
No	1										
Nombre	CD-CdU1-Generar-tabla-datos										
Descripción	Cuando un usuario ingresa en la pestaña de reportes, automáticamente se llena la tabla con todos los datos existentes en la base de datos, esto para verificar y generar reportes de la misma.										
Actores	Primario : Usuario										
Fase	Levantamiento										
Guión	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td>1. Ingresar a Sistema de Información</td> <td></td> </tr> <tr> <td>2. Ingresar a la pestaña "generar reporte".</td> <td></td> </tr> <tr> <td></td> <td>3. Generar tabla con los datos almacenados en la base de datos.</td> </tr> </tbody> </table>			Actor	Sistema	1. Ingresar a Sistema de Información		2. Ingresar a la pestaña "generar reporte".			3. Generar tabla con los datos almacenados en la base de datos.
Actor	Sistema										
1. Ingresar a Sistema de Información											
2. Ingresar a la pestaña "generar reporte".											
	3. Generar tabla con los datos almacenados en la base de datos.										
Excepciones	5. que la base de datos este sin datos.										
	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>			Actor	Sistema						
Actor	Sistema										

		5.1 muestra una tabla sin datos.	
		5.2. Termina	
Casos de uso relacionados			
Documentos relacionados			
Otros Requerimientos	Debe funcionar en Internet Explorer 7 o superior, Firefox 3.5 o superior. Se recomienda el uso del navegador Google Chrome para mejor visualización de la aplicación.		
Prototipo de Pantallas			
Control de Cambios			
Versión	Autores	Fecha	Comentario
1	Stiven Toledo	Septiembre/2016	Versión Inicial

9.1.4.2.3 Generar reportes.

Proyecto	SenVarA
Modulo	Principal
No	1
Nombre	CD-CdU1-Generar-reportes
Descripción	Cuando un usuario ingresa en la pestaña de reportes, automáticamente se llena la tabla con todos los datos existentes en la base de datos, en ese momento se cargan 3 botones para poder descargar el contenido en formatos distintos (PDF, XLS y CSV).

Actores	Primario : Usuario	
Fase	Levantamiento	
Guión	Actor	Sistema
	1. Ingresar a Sistema de Información	
	2. Ingresa a la pestaña “generar reporte”.	
		3. Generar tabla con los datos almacenados en la base de datos.
	4. Presionar el botón que indica el formato según el tipo de reporte que desee generar	
Excepciones	5. que la base de datos este sin datos.	
	Actor	Sistema
		5.1 genera archivos sin datos.
		5.2. Termina
Casos de uso relacionados		
Documentos relacionados		
Otros Requerimientos	Debe funcionar en Internet Explorer 7 o superior, Firefox 3.5 o superior. Se recomienda el uso del navegador Google Chrome para mejor visualización de la aplicación.	
Prototipo de Pantallas		
Control de Cambios		

Versión	Autores	Fecha	Comentario
1	Stiven Toledo	Septiembre/2016	Versión Inicial

9.1.4.3 No funcionales

- **Restricción de conectividad:**

El acceso al sitio de consulta de variables ambientales se debe hacer a través de internet.

- **Restricción de usabilidad:**

El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.

La aplicación web debe poseer un diseño “Responsive” a fin de garantizar la adecuada visualización en múltiples computadores personales, dispositivos tableta y teléfonos inteligentes.

El sistema debe poseer interfaces gráficas bien formadas.

9.1.5 Arquitectura de la solución

El Sistema de Información SenVarA estará basado una arquitectura cliente – servidor de tres capas. La idea de desarrollo en tres capas consiste en distribuir los diferentes componentes del sistema de acuerdo a su funcionalidad, en por lo menos tres grupos: la interfaz de usuario, las reglas del negocio y los modelos de datos. Esta separación es la razón fundamental para la utilización de la tecnología de JavaEE.

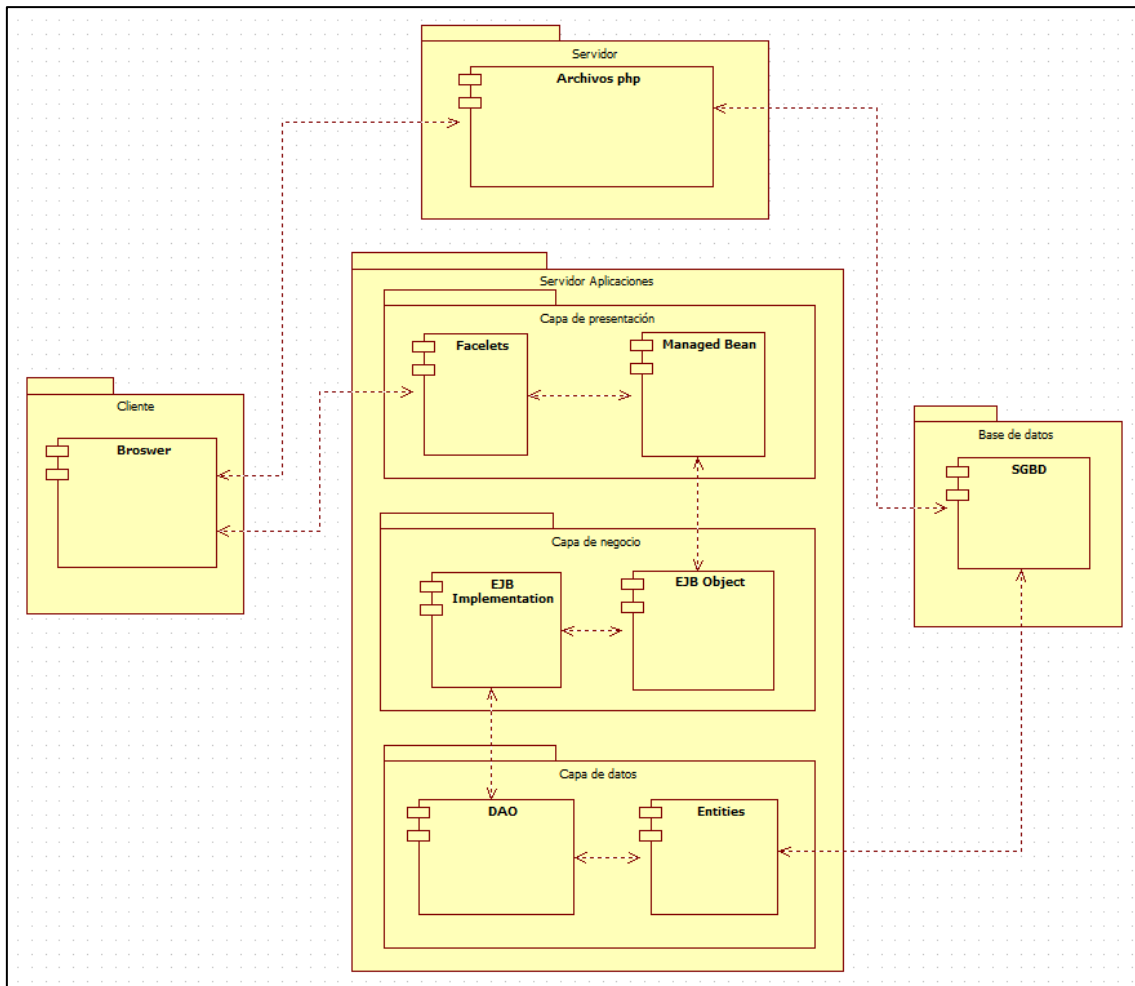


Ilustración 20: Arquitectura de la aplicación

La **Ilustración 21** representa la arquitectura propuesta en la cual se pueden observar los siguientes componentes:

- **FACELETS:** son paginas XHTML usadas por el framework JSF para representar sus vistas, éstas utilizan bloques HTML como componentes del marco, apoyados por clases java
- **MANAGED BEAN:** son las clases java usadas como modelos o controladores para los componentes de la interfaz de usuario.
- **EJB OBJECT:** son clases tipo interfaz las cuales declaran los métodos de instancia públicos, encargados de realizar las operaciones, sin su implementación.
- **EJB IMPLEMENTATION:** Son las clases donde se encuentra la lógica de negocio, implementan todos los métodos especificados por la interfaz

- **DAO:** Son las clases que abstraen los métodos de manipulación de la persistencia como todas las actividades CRUD (créate, read, update y delete).
- **ENTITIES:** Son las clases que representan una tabla del modelo de datos relacional y cada instancia u objeto corresponde a un registro en esta tabla.

9.1.6 Diseño de la interfaz

Con las funcionalidades establecidas para el sistema en los casos de uso se realizaron los siguientes prototipos de interfaz.

La ilustración 22 muestra la interfaz principal de la aplicación, se determinó que no era necesario implementar más de una interfaz, ya que en la principal se puede mostrar toda la información requerida para esta solución.

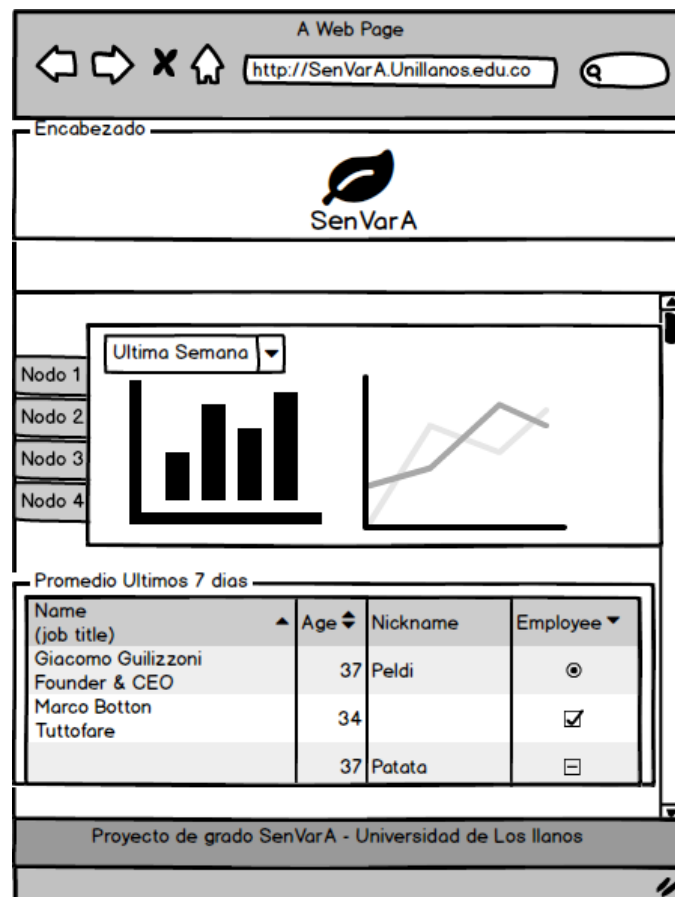


Ilustración 21: Mockup interfaz gráfica Senvara – Web.

9.2 Diseño de solución

Para este proyecto se estimó que es necesario diseñar 2 aplicaciones: una encargada de recolectar las variables ambientales y almacenarlas en una base de datos MySQL la cual se denominó SenVarA, la otra aplicación que se requiere diseñar, es un sistema de información web, el cual debe mostrar la información almacenada en la base de datos, en una forma clara y ordenada, ya sea por medio de la utilización de graficas o tablas, esta se denominó SenVarA Web.

A continuación se explica brevemente el diseño de cada una de las soluciones.

9.2.1 SenVarA

Se requiere una aplicación que almacene en una base de datos MySQL los datos obtenidos por un puerto del servidor, los cuales son obtenidos a través de sensores implantados en un terreno de prueba, para ello se diseñó una solución utilizando un lenguaje de la plataforma .NET denominado C# (C Sharp). [37]

.NET Framework es un entorno multilenguaje para generar, implementar y ejecutar servicios Web XML y aplicaciones Web. Consta de tres partes principales [38]:

- Common Language Runtime
- Clases de programación unificada
- ASP.NET

Para compilar el código fuente de C#, se utilizó el Entorno de desarrollo integrado (IDE) de Visual Studio 2015.

Microsoft C# es un lenguaje de programación diseñado para crear un amplio número de aplicaciones empresariales que se ejecutan en .NET Framework. Supone una evolución de Microsoft C y Microsoft C++; es sencillo, moderno, proporciona seguridad de tipos y está orientado a objetos. El código creado mediante C# se compila como código administrado, lo cual significa que se beneficia de los servicios de Common Language Runtime. Estos servicios incluyen interoperabilidad

entre lenguajes, recolección de elementos no utilizados, mejora de la seguridad y mayor compatibilidad entre versiones [38].

La aplicación desarrollada en Visual C#, tiene la tarea de recibir la información por los puertos del computador y almacenarla en una base de datos MySQL.

9.2.2 SenVarA Web

Para el diseño y desarrollo se aplicó la metodología PSP. Inicialmente y como línea base para futuras estimaciones y planes se utilizó el primer nivel PSP0, que tiene como objetivo inculcar practicas disciplinadas, siguiendo los diferentes scripts del proceso proporcionados por Watts S. Humphrey [34]. Este autor nos enseña la importancia del registro de datos, haciendo uso de los diferentes formularios y estándares, en este primer nivel nos encargamos básicamente de medir el tiempo invertido y los defectos inyectados y removidos en cada una de las fases.

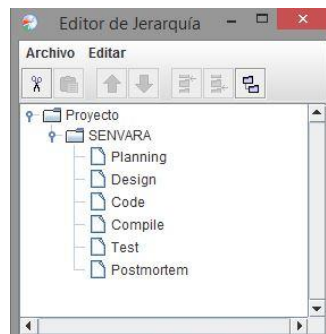


Ilustración 22: Creación del Proyecto en Process Dashboard

Para desarrollar el proyecto y seguir el proceso planteado en PSP0 haremos uso de la herramienta Process Dashboard [39] la cual nos brinda soporte para el proceso de PSP/TSP, esta es una herramienta desarrollada en Java y con licencia Open Source, que integra todos los elementos fundamentales (Scripts, Formas, Métricas y Estándares) y de una forma muy fluida y ágil.

Para dar inicio creamos al proyecto con la herramienta como lo muestra la **ilustración 19**. El nombre utilizado para nuestro proyecto es “**SenVarA**” que representa las iniciales de **Sensores de Variables Ambientales** esto para no extender demasiado el campo nombre, cabe recordar que PSP tiene por definición tres fases (Planeación, Desarrollo

y PostMorten), a su vez PSP0 expande la fase de desarrollo en Diseño, Codificación y Pruebas, es por esto que al momento de crear el proyecto la herramienta inmediatamente nos crea la estructura de carpetas con las diferentes fases.

Para aplicar PSP es necesario seguir las actividades propuestas en el Script, con Process Dashboard podemos visualizar los Scripts que necesitemos en cualquier momento, para el caso de PSP 0 esta es nuestra guía general de proceso.

PSP0 Process Script	
Purpose	To guide the development of module-level programs
Entry Criteria	<ul style="list-style-type: none"> • Problem description • PSP0 Project Plan Summary form • Time and Defect Recording logs • Defect Type standard • Stopwatch (optional)
Step	Activities
	Description
1	Planning
	<ul style="list-style-type: none"> • Produce or obtain a requirements statement. • Estimate the required development time. • Enter the plan data in the Project Plan Summary form. • Complete the Time Recording log.
2	Development
	<ul style="list-style-type: none"> • Design the program. • Implement the design. • Compile the program, and fix and log all defects found. • Test the program, and fix and log all defects found. • Complete the Time Recording log.
3	Postmortem
	<ul style="list-style-type: none"> • Complete the Project Plan Summary form with actual time, defect, and size data.
Exit Criteria	<ul style="list-style-type: none"> • A thoroughly tested program • Completed Project Plan Summary form with estimated and actual data • Completed Time and Defect Recording logs

Adapted from "PSP Materials," copyright © 2006 Carnegie Mellon University. Used by permission.

Ilustración 23: PSP0 Process Script

Este Script es la guía que nos permite avanzar de manera ordenada y disciplinada en nuestro trabajo, las actividades establecidas son las siguientes:

9.2.2.1 Criterios de Entrada

La primera actividad es verificar que los criterios de entrada se cumplan, para nuestro caso los criterios se cumplen así:

9.2.2.1.1 Descripción del problema

En el numeral **4** se describió el problema planteado.

9.2.2.1.2 Formulario de resumen del proyecto (“PSP0 Project Plan Summary”)

/Proyecto/SENVARA

PSP0 Project Plan Summary

Project Owner:

Start Date:

Completion Date: Completed:

Keywords:

Language:


Time in Phase (min.)	Plan	Actual	To Date	To Date %
Planning		95:19	0:00	5%
Design		23:45	0:00	25%
Code		70:54	0:00	25%
Compile		2:19	0:00	2,5%
Test		40:59	0:00	12,5%
Postmortem		0:00	0:00	5%
Total	<input type="text" value="????"/>	233:16	0:00	

Defects Injected	Actual	To Date	To Date %
Before Development	0	0	0%
Planning	0	0	0%
Design	0	0,0	30,8%
Code	0	0,0	61,5%
Compile	0	0,0	7,69%
Test	0	0	0%
Total Development	0	0,0	

Defects Removed	Actual	To Date	To Date %
Planning	0	0	0%
Design	0	0	0%
Code	0	0,0	7,69%
Compile	0	0,0	15,4%
Test	0	0,0	7,69%

Ilustración 24: PSP0 Project Plan Summary. Tomada del DashBoard

9.2.2.1.3 Formulario de registro de tiempo(“PSP0 Time Recording Logs”)



The screenshot shows a web browser window with the title 'Bitácora de TiempoPara' and the URL 'localhost:2468/Proyecto/SENVARA//reports/timelog.class'. The main content is a table titled 'Bitácora de TiempoPara /Proyecto/SENVARA'. The table has six columns: 'Proyecto/Tarea', 'Fase', 'Tiempo Inicio', 'Delta', 'Interrupción', and 'Comentario'. It contains 20 rows of data, each representing a time recording entry.

Proyecto/Tarea	Fase	Tiempo Inicio	Delta	Interrupción	Comentario
Proyecto/SENVARA	Planning	feb 10, 2016 04:11:15 PM	2	0	
Proyecto/SENVARA	Code	feb 10, 2016 08:17:51 PM	330	0	
Proyecto/SENVARA	Code	feb 11, 2016 11:58:05 AM	86	0	
Proyecto/SENVARA	Code	feb 11, 2016 03:14:03 PM	93	0	
Proyecto/SENVARA	Planning	feb 17, 2016 01:28:42 AM	117	0	
Proyecto/SENVARA	Planning	feb 23, 2016 09:11:49 AM	69	0	
Proyecto/SENVARA	Planning	mar 02, 2016 08:29:36 AM	50	0	
Proyecto/SENVARA	Planning	mar 04, 2016 09:21:26 AM	83	0	
Proyecto/SENVARA	Planning	mar 15, 2016 11:58:13 AM	59	0	
Proyecto/SENVARA	Code	mar 15, 2016 12:57:13 PM	610	0	
Proyecto/SENVARA	Code	mar 21, 2016 09:07:51 PM	43	0	
Proyecto/SENVARA	Planning	mar 29, 2016 12:20:51 PM	680	0	
Proyecto/SENVARA	Planning	abr 11, 2016 11:44:44 PM	27	0	
Proyecto/SENVARA	Planning	abr 12, 2016 02:48:29 PM	579	0	
Proyecto/SENVARA	Planning	abr 26, 2016 01:53:50 PM	490	0	
Proyecto/SENVARA	Planning	may 03, 2016 04:08:39 PM	301	0	
Proyecto/SENVARA	Code	may 04, 2016 08:42:29 AM	72	0	
Proyecto/SENVARA	Design	may 09, 2016 08:11:07 PM	882	0	

Ilustración 25: PSP0 Time Recording Logs.Tomada del Dashboard

9.2.2.2 Modelo de datos

9.2.2.2.1 Modelo Entidad Relación

Una vez definida la arquitectura de la aplicación se procedió con el diseño del modelo de datos el cual consistió en buscar las entidades que describen los objetos que intervienen en el problema y las relaciones entre estas entidades. Las tablas o entidades identificadas para el funcionamiento del sistema y sus relaciones se pueden observar en el modelo Entidad – Relación representado en la **Ilustración 27**.

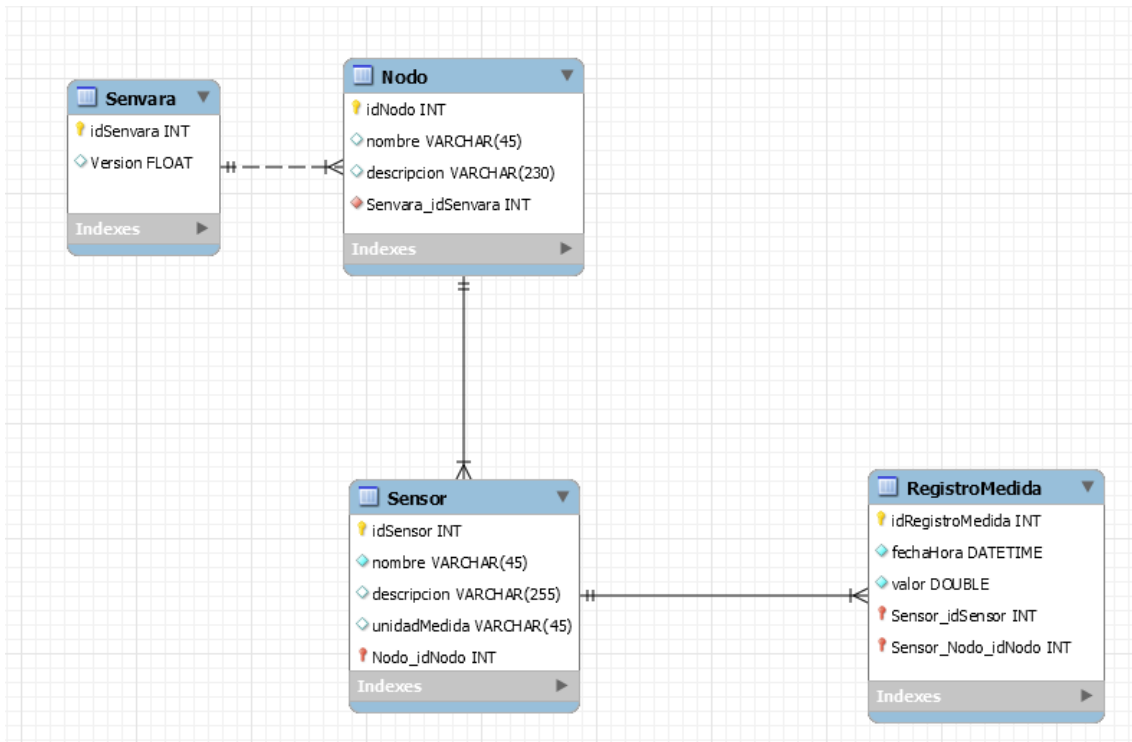
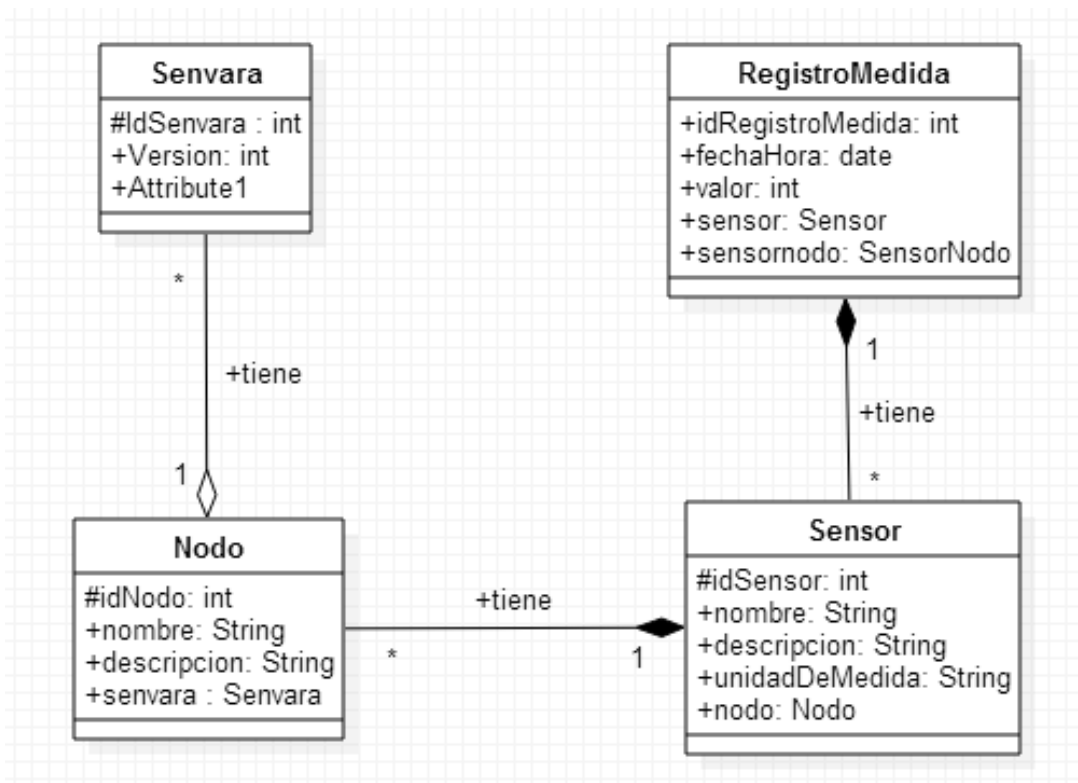
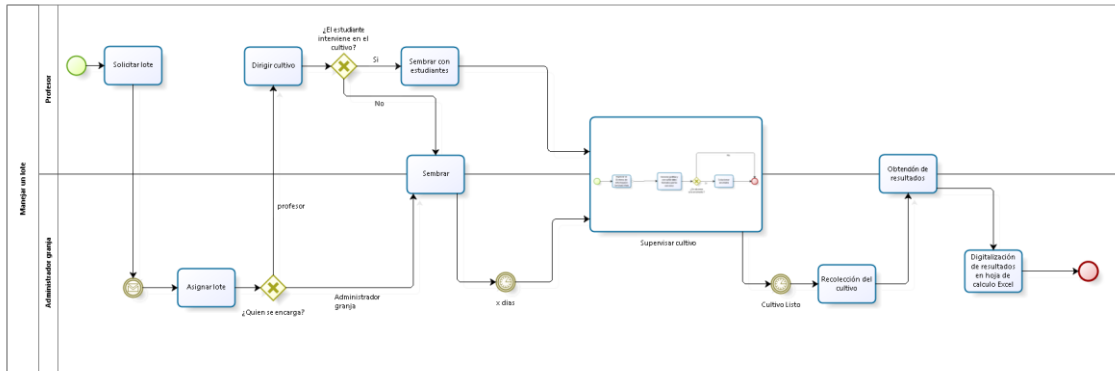


Ilustración 26: Modelo Entidad - Relación

9.2.2.3 Diagrama de clases

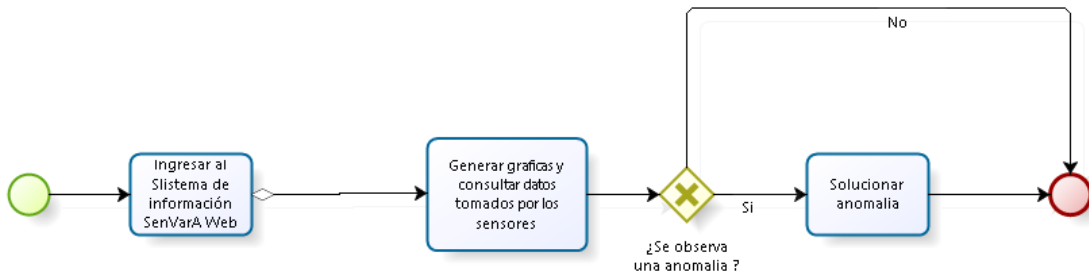


9.2.2.4 Modelo de procesos



Powered by
bizagi
Modeler

Ilustración 27: Modelo BPMN del proceso TO-BE



Powered by
bizagi
Modeler

Ilustración 28: Modelo BPMN del proceso "Supervisar Cultivo" TO-BE

9.3 Desarrollos

9.3.1 SenVarA

Para iniciar con el proyecto, los sensores se configuraron previamente utilizando tecnologías tales como el arduino y la Xbee-Pro; los cuales se utilizaron para capturar los valores que arrojan los sensores y por medio inalámbrico enviar esos datos a un dispositivo, el cual debe almacenar en una base de datos MySQL los valores capturados, para darle solución a esto, se desarrolló una pequeña aplicación en Visual Studio 2012 utilizando el lenguaje de

programación C#, la cual se encarga de recibir los valores enviados al puerto usb del computador y almacenarlos en la base de datos MySQL.



Ilustración 29: Pantallazo de la aplicación Senvara-App

Para explicar mejor el trabajo previo al diseño de la solución web, se realizó un documento el cual explica paso a paso el proceso de implementación de los sensores que se realizó para este proyecto, el cual se complementó con la información de SenvaraWeb y se dejó como Manual Técnico.

DESARROLLO DEL PROYECTO

A continuación se explica el proceso realizado en implementación del proyecto, el cual inicia con la configuración de los sensores, la exploración y la investigación de las herramientas como lo es Arduino y Xbee, que van a ser esenciales para el desarrollo del mismo.

A continuación se realiza una breve descripción de los elementos principales utilizados en la construcción de los Nodos.

Arduino Mega 2650

Arduino es un proyecto de código abierto que crea micro-controladores basados en kits para la construcción de dispositivos digitales y objetos interactivos que pueden detectar y controlar dispositivos físicos [2].



Imagen 1: Placa de arduino

Shield Arduino - Xbee

La shield Xbee ha sido creada en colaboración con Libelium [3] y sirve de intermediario entre la Xbee y la placa arduino, es de mucha utilidad ya que esto evita realizar un circuito aparte en la transición de información de la xbee a la placa arduino y viceversa [4].

Ilustración 30: Imagen tomada del documento Manual Técnico anexo

- Disposición de los sensores en un terreno de prueba.

Para comprobar que los sensores capturan variables en un terreno cultivable se requiere uno de prueba. Por facilidad de transporte y para este prototipo se determinó que el uso de unas Materas es lo más apropiado para esta labor, se usaron 4, en las cuales se implementaron los sensores con sus respectivos circuitos de captura y envió de información, como se muestra en la Ilustración 23.



Ilustración 31: Fotografía de la implementación de los sensores en terreno de prueba.

9.3.2 SenVarA Web

- Se realizaron diversas consultas en la base de datos poblada. Para verificar su correcto funcionamiento. Algunas de estas consultas son:

Consultas:

Consultar todos los Nombres de nodos, nombres de los sensores, valores y fechas.

```
SELECT n.nombre, s.nombre, r.valor, r.fechaHora
FROM nodo n, sensor s, registromedida r
WHERE n.idNodo=s.idNodo AND s.idSensor=r.idSensor
```

nombre	nombre	valor	fechaHora
Nodo 1	Temperatura_C	30	2017-07-18 12:02:47
Nodo 1	Temperatura_C	29	2017-07-18 13:05:20
Nodo 1	Temperatura_C	29	2017-07-18 14:08:40
Nodo 1	Temperatura_C	28	2017-07-18 15:11:27
Nodo 1	Temperatura_C	29	2017-07-18 16:14:06
Nodo 1	Temperatura_C	31	2017-07-18 17:17:15
Nodo 1	Temperatura_C	29	2017-07-18 18:20:49
Nodo 1	Temperatura_C	27	2017-07-18 19:22:36
Nodo 1	Temperatura_C	25	2017-07-18 20:25:54
Nodo 1	Temperatura_C	26	2017-07-18 21:29:13
Nodo 1	Temperatura_F	86	2017-07-18 12:02:47
Nodo 1	Temperatura_F	84	2017-07-18 13:05:20
Nodo 1	Temperatura_F	84	2017-07-18 14:08:40
Nodo 1	Temperatura_F	82	2017-07-18 15:11:27
Nodo 1	Temperatura_F	84	2017-07-18 16:14:06
Nodo 1	Temperatura_F	88	2017-07-18 17:17:15

Consultar Nombres de los sensores, valores y fechas del Nodo 1


```
SELECT n.nombre, s.nombre, r.valor, r.fechaHora FROM nodo n, sensor s, registromedida r WHERE n.idNodo=s.idNodo AND s.idSensor=r.idSensor AND n.nombre= "Nodo 1"
```

#PMAURL-3:tbl_sql.php?db=senvarabd&table=nodo&server=1&target=&token=2eada15b1f6f0226154fa8cb11af0b6

Servidor: 127.0.0.1 » Base de datos: senvarabd » Tabla: nodo

Examinar Estructura SQL Buscar Insertar Exportar Importar Privilegios Más

Mostrando filas 0 - 24 (total de 50, La consulta tardó 0.0019 segundos.)

```
SELECT n.nombre, s.nombre, r.valor, r.fechaHora FROM nodo n, sensor s, registromedida r WHERE n.idNodo=s.idNodo AND s.idSensor=r.idSensor AND n.nombre= "Nodo 1"
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Agregar comentario]

1 > >> | Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

nombre	nombre	valor	fechaHora
Nodo 1	Temperatura_C	30	2017-07-18 12:02:47
Nodo 1	Temperatura_C	29	2017-07-18 13:05:20
Nodo 1	Temperatura_C	29	2017-07-18 14:08:40
Nodo 1	Temperatura_C	28	2017-07-18 15:11:27
Nodo 1	Temperatura_C	29	2017-07-18 16:14:06
Nodo 1	Temperatura_C	31	2017-07-18 17:17:15
Nodo 1	Temperatura_C	29	2017-07-18 18:20:49
Nodo 1	Temperatura_C	27	2017-07-18 19:22:36
Nodo 1	Temperatura_C	25	2017-07-18 20:25:54
Nodo 1	Temperatura_C	26	2017-07-18 21:29:13
Nodo 1	Temperatura_F	86	2017-07-18 12:02:47
Nodo 1	Temperatura_F	84	2017-07-18 13:05:20
Nodo 1	Temperatura_F	84	2017-07-18 14:08:40

Consultar las temperaturas en °C

```
SELECT n.nombre, s.nombre, r.valor, r.fechaHora FROM nodo n, sensor s, registromedida r WHERE n.idNodo=s.idNodo AND s.idSensor=r.idSensor AND s.nombre= "Temperatura_C"
```

Servidor: 127.0.0.1 » Base de datos: senvarabd » Tabla: nodo

Examinar Estructura SQL Buscar Insertar Exportar Importar Privilegios Más

```
SELECT n.nombre, s.nombre, r.valor, r.fechaHora FROM nodo n, sensor s, registromedida r WHERE n.idNodo=s.idNodo AND s.idSensor=r.idSensor AND s.nombre="Temperatura_C"
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Ac

1 > >> Mostrar todo | Número de filas: 25 | Filtrar filas:

+ Opciones

nombre	nombre	valor	fechaHora
Nodo 1	Temperatura_C	30	2017-07-18 12:02:47
Nodo 1	Temperatura_C	29	2017-07-18 13:05:20
Nodo 1	Temperatura_C	29	2017-07-18 14:08:40
Nodo 1	Temperatura_C	28	2017-07-18 15:11:27
Nodo 1	Temperatura_C	29	2017-07-18 16:14:06
Nodo 1	Temperatura_C	31	2017-07-18 17:17:15
Nodo 1	Temperatura_C	29	2017-07-18 18:20:49
Nodo 1	Temperatura_C	27	2017-07-18 19:22:36
Nodo 1	Temperatura_C	25	2017-07-18 20:25:54
Nodo 1	Temperatura_C	26	2017-07-18 21:29:13
Nodo 2	Temperatura_C	22	2017-07-18 12:02:47
Nodo 2	Temperatura_C	23	2017-07-18 13:05:20
Nodo 2	Temperatura_C	24	2017-07-18 14:08:40
Nodo 2	Temperatura_C	25	2017-07-18 15:11:27
Nodo 2	Temperatura_C	26	2017-07-18 16:14:06

Insertar:

Insertar en el Nodo 2

```
INSERT INTO `senvarabd`.`registromedida` (`idRegistroMedida`, `fechaHora`, `valor`, `idSensor`, `Sensor_Nodo_idNodo`) VALUES (NULL, CURRENT_TIMESTAMP, '26', '2', '2'), (NULL, CURRENT_TIMESTAMP, '32', '6', '2'), (NULL, CURRENT_TIMESTAMP, '45', '10', '2'), (NULL, CURRENT_TIMESTAMP, '126', '14', '2'), (NULL, CURRENT_TIMESTAMP, '724', '18', '2');
```

```
INSERT INTO `senvarabd`.`registromedida` (`valor`, `idSensor`, `Sensor_Nodo_idNodo`) VALUES ('36', '2', '2'), ('52', '6', '2'), ('25', '10', '2'), ('26', '14', '2'), ('124', '18', '2')
```

Nodo 1

```
INSERT INTO `senvarabd`.`registromedida` (`valor`, `idSensor`, `Sensor_Nodo_idNodo`) VALUES ('22', '1', '1'), ('12', '5', '1'), ('15', '9', '1'), ('26', '13', '1'), ('524', '17', '1')
```

Nodo 3

```
INSERT INTO `senvarabd`.`registromedida` (`valor`, `idSensor`, `Sensor_Nodo_idNodo`) VALUES ('31', '3', '3'), ('72', '7', '3'), ('45', '11', '3'), ('36', '15', '3'), ('24', '19', '3')
```

Nodo 4

```
INSERT INTO `senvarabd`.`registromedida` (`valor`, `idSensor`, `Sensor  
Nodo_idNodo`) VALUES ('16', '4', '4'), ('22', '8', '4'), ('45', '12',  
'4'), ('2', '16', '4'), ('424', '20', '4')
```

10 RESULTADOS

- Prototipo de una red inalámbrica de sensores (WSN), conformada por cuatro nodos y un coordinador encargados de recolectar las variables ambientales y transmitirlas a la aplicación SenVarA.
- Software denominado SenVarA App desarrollado en C#, encargado de recibir los datos por puerto serial y almacenarlos de forma ordenada en una base de datos MySQL.
- Desarrollo de la aplicación SenVarA Web Utilizando JSF, en el IDE proporcionado por Eclipse. Esta aplicación se instaló en el servidor de la oficina del grupo de investigación DyATIC, para poder realizar las respectivas pruebas.

11 CONCLUSIONES

- Con el prototipo del sistema de información SenVarA se pudo observar que las personas encargadas de la administración de los cultivos se verán beneficiados con la implementación de este sistema, ya que con la automatización de este proceso se podrán monitorear los cultivos agrícolas para mejorar la producción, ahorrando tiempo y optimizando el análisis de la información para la aplicación de insumos o toma de decisiones para implementar medidas que mejoren la producción, lo cual es muy necesario y útil en un país productor como el nuestro.
- El realizar este trabajo de grado permitió poner en práctica todos los conocimientos adquiridos a lo largo de la formación profesional, desde investigación, formulación de un proyecto y desarrollo del mismo con lo cual se demuestra la calidad de los cursos correspondientes al programa de Ingeniería de Sistemas incluyendo algunos básicos de electrónica.
- Al implementar este proyecto se logra automatizar algunas de las labores realizadas en la toma y verificación de diversas variables ambientales, en los cultivos de la granja de la Unillanos.
- Se genera cultura acerca del manejo de las TIC para entidades educativas como la Universidad de los Llanos, en la que las personas tienen facilidad de acceder a los servicios, logrando así sacar el mayor provecho posible a las TIC con un fin altamente favorable para la Universidad de los Llanos.

12 TRABAJO FUTURO.

El proyecto de grado denominado “SenVarA (sensores de Variables Ambientales) es una solución a nivel “prototipo”, que permite determinar las principales actividades que se pueden realizar para ampliar las funcionalidades. A continuación se sugieren actividades que podrían realizarse como trabajo futuro:

1. Extender el modelo de datos agregando una entidad que describa las características de las variables tomadas (temperatura, humedad, etc), como por ejemplo: unidad de medida, precisión, etc.
2. Extender el modelo de datos para agregar al nodo su localización georeferenciada, de forma que se pueda observar en un mapa digital, dónde se encuentran localizados los nodos de sensores.
3. Evaluar el diseño de la base de datos para un cultivo real, teniendo en cuenta aspectos como la necesidad de almacenar gran cantidad de información.
4. Evolucionar el prototipo con una fuente de energía renovable
5. Un sistema de información que cuente con una amplia base de datos con variables ambientales tiene diversos usos en los cuales se puede enfocar, como trabajo futuro se propone ampliar a otros campos la implementación de este proyecto, por ejemplo, en la producción de pollos, donde es importante tener en cuenta la temperatura y humedad ambiente.
6. Implementar un método en el cual se almacene la información en el ordenador encargado de enviar la información, en caso de perder la conexión a la base de datos, para cuando se restablezca, no perder datos.

13 Bibliografía

- [1] Santamaría Bultrago F. Archila Cordoba D., "Estado del arte de las redes de sensores inalámbricos," *Revista digital TIA.2(1)*, pp. 4-14, 2013.
- [2] Díaz Rodríguez M. Isleydi ReyesR., "Desarrollo de un software basado en modelos para redes inalámbricas de sensores.," *Revista Académica de Tlatemoani.*, 2013.
- [3] National Instruments Corporation. (2009, Abril) <http://www.ni.com>. [Online]. <http://www.ni.com/white-paper/7142/es/>
- [4] Arango A, "Haciendo Camino.," , Villavicencio - Meta, Recuperado de <http://acreditacion.unillanos.edu.co/contenidos/HISTORIA.pdf>.
- [5] Montoya A, Aristizabal D, and Piedrahita D, "Monitoreo de variables medioambientales, mediante redes de sensores inalámbricas usando motes," 2008.
- [6] Hagendoorn. (2012) Documento en línea de Hagendoorn. [Online]. <http://fcbi.unillanos.edu.co/proyectos/Facultad/php/programas/sistemas/docs/plans.2011.pdf>
- [7] Garcia E and Flego F, "Redes de sensores inalámbricos," *ABB*, ISSN 1013-3135, N° 2, pp. 39-42, 2006.
- [8] Blackmore S, "Agricultura de precision," *Revista Nacional de Agricultura*. Vol N°949, pp. 20-48, 2007, Recuperado de <http://www.sac.org.co/Pages/Revista/949/agriculturaprecision.pdf>.
- [9] Lewis F, Magazine Smart Eviroments: Technologies, Protocols and Applications.
- [10] Gildardo L and Montoya N., Monitoreo y control remoto de variables de un invernadero., 2005.
- [11] Escolar S, Carretero J, Garcia F, Isaila F, and Fernandez J, Acabando con los desarrollos ad-hoc en Wireless Sensor Networks, Septiembre 2006, Recuperado de http://www.arcos.inf.uc3m.es/~sescolar/index_files/PUBLICACIONES/JornadasParalelismo2006.pdf.
- [12] Domínguez O, Plan de gestión institucional 2009 - 2012, Recuperado el 16 de 09 de 2011, de universidad de los llanos:

acreditación.unillanos.edu.co/contenido/pgi.pdf.

- [13] Chandrima K, "A Survey on Sensor Network," *International Journal of computer and information technology (IJCIT) Vol1*, pp. 76-87, Recuperado de http://www.uap-bd.edu/jcit_papers/vol-1_no-1/JCIT-100715.pdf.
- [14] Méndez Chavez D, Pena N, and Guerrero M, "Redes de sensores inalámbricas: estructura y aplicación," *Revista Innovación y Ciencia ISSN 0121-5140*, 2007.
- [15] Martínez R, Mere J, Pisón Ascacibar, and Marcos A, "Redes inalámbricas de sensores: Teoría y aplicación práctica," *Editorial Universidad de la Rioja*.
- [16] ZigBee Alliance. (2013) Zigbee Specifications. [Online]. <https://www.zigbee.org/Specifications/ZigBeeIP/Overview.aspx>
- [17] Weiss M, TINYOS., Diciembre 2003, Recuperado de <http://csce.unl.edu/~witty/f2004/csce489/WSN-20031201.pdf>.
- [18] Oracle. Oracle Corporation. [Online]. <http://www.oracle.com/index.html>
- [19] Oracle. (2013, Mayo) Pagina Oficial sobre Java Enterprise Edition. [Online]. <http://www.oracle.com/technetwork/java/javaee/overview/index.html>
- [20] Oracle. Java SE. [Online]. <http://www.oracle.com/technetwork/java/javase/overview/index.html>
- [21] ORACLE Corp. JavaServer Faces Technology. [Online]. <http://www.oracle.com/technetwork/java/javaee/jaserverfaces-139869.html>
- [22] W3C. (2014, Abril) HTML 5.1 Nightly. [Online]. <http://www.w3.org/html/wg/drafts/html/master/Overview.html>
- [23] Marco Antonio Cruz Chávez. (2010, Enero) Grid Morelos. [Online]. <http://www.gridmorelos.uaem.mx/~mcruez//cursos/miic/bd1.pdf>
- [24] (2007, July) Maestros del Web. [Online]. <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>
- [25] (2003, Nov.) Maestros del Web. [Online]. <http://www.maestrosdelweb.com/introcss/>

- [26] get bootstrap. [Online]. <http://getbootstrap.com/getting-started/>
- [27] Cecilio Alvarez Caules,, 2014, ch. 1.
- [28] Erick Camacho. JavaHispano Tutorial Introducción a Maven. [Online]. javahispano.org
- [29] A. Hevner, S. March, J. Park, and S. Ram, "Design Science Research in Information Systems," *Management Information Systems Quarterly*, vol. 28, no. 1, pp. 75-105, marzo 2004.
- [30] Herbert A Simon, *The Sciences of the Artificial*, 3rd ed. Cambridge, MA: MIT Press, 1996.
- [31] Vijay Vaishnavi and Bill Kuechler. (2004, enero) Association for Information Systems. [Online]. <http://desrist.org/design-research-in-information-systems/>
- [32] Ken Peppers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee, "A Design Science Research Methodology for Information Systems Research.," *Journal of Management Information Systems*, vol. 24, no. Issue 3, pp. 45-77, Winter 2007/2008.
- [33] INGENIARE, "Arquitectura orientada a servicios para software de apoyo para el proceso personal de software.," *Revista Chilena de Ingeniería*, vol. 19, no. 1, pp. 40-53, January-2011.
- [34] Watts S. Humphrey. Introduction to the Personal Software Process.
- [35] Universidad Autónoma de México - UAM. (2014, Junio) Facultad de Ingeniería. [Online]. http://prontus.uv.cl/pubacademica/pubprofesores/n/pubniemannkaren/site/artic/20070522/asocfile/ejercicio_word_vers_2003.doc
- [36] Universidad Carlos III de Madrid. (2012) OpenCourseWare - Universidad Carlos III de Madrid. [Online]. <http://ocw.uc3m.es/ingenieria-informatica/principios-de-ingenieria-informatica/quiones-ppsp0>
- [37] Donis Marshall, "Programación Seguridad de.NET ," *CodeGuru*, 2014.
- [38] Microsoft. (2016) msdn.microsoft.com. [Online]. [https://msdn.microsoft.com/es-es/library/aa287477\(v=vs.71\).aspx](https://msdn.microsoft.com/es-es/library/aa287477(v=vs.71).aspx)

- [39] the software process dashboard initiative. [Online]. <http://www.processdash.com/>
- [40] Charles Oppenheim, "CLOUD LAW AND CONTRACT NEGOTIATION.," *El Profesional de la Información*, vol. Vol. 21 Issue 5, no. 2, pp. 453-457, 2012. [Online]. http://www.elprofesionaldelainformacion.com/contenidos/2012/septiembre/02_esp.pdf
- [41] A. Hevner, S. March, J. Park, and S. Ram, "Design Science Research in Information Systems," *Management Information Systems Quarterly*, vol. 28, no. 1, pp. 75-105, marzo 2004.
- [42] Global Mentoring. (2015) Curso Java Enterprise Edition. [Online]. <http://globalmentoring.com.mx/curso-javaee/>
- [43] Global Mentoring. Explota tu creatividad con JavaServer Faces.
- [44] Universidad de Alicante. Introduccion JSF. [Online]. <http://www.ittech.ua.es/j2ee/publico/jsf-2012-13/sesion01-apuntes.html>
- [45] Apache Maven Project. Apache maven. [Online]. <http://maven.apache.org/>
- [46] CodeJobs. Jboss. [Online]. <http://www.codejobs.biz/es/blog/2013/09/30/que-es-jboss#sthash.Pvfol0i4.dpbs>
- [47] Universidad EAFIT. Repositorio institucional. [Online]. <https://repository.eafit.edu.co/bitstream/handle/10784/2763/6.%20MarcoTeorico.pdf?sequence=6&isAllowed=y>
- [48] Stiven Toledo, "Entrevista Ingeniero Adrian, encargado de la Granja," Universidad De Los Llanos, Villavicencio, 2016.
- [49] Montoya A, Aristizabal D, and Piedrahita D, "Monitoreo de variables medioambientales mediante redes de sensores inalámbricas usando motes.," 2008.
- [50] Universidad de los llanos. (2014, July) Maestría en Gestión Ambiental Sostenible. [Online]. <http://magestas.unillanos.edu.co/>
- [51] Universidad de los Llanos. (2010, Apr.) Unillanos. [Online]. <http://www.unillanos.edu.co/index.php/presentacion-idead>

- [52] Corporación universitaria Rafael Núñez. (2011, May) CERES CURN. [Online]. <http://www.curn.edu.co/es/ceres-curn.html>
- [53] Ministerio de Educación. (2009, July) Colombia aprende. [Online]. <http://www.mineducacion.gov.co/1621/w3-article-187077.html>
- [54] Universidad de los Llanos. (2009, Julio) Reforma estatuto general. [Online]. http://reformaestatutaria.unillanos.edu.co/as_04_2009_estauto%20gral.pdf
- [55] Universidad de los Llanos. (2009, Julio) Unillanos sitio web oficial. [Online]. http://www.unillanos.edu.co/index.php/centro-de-documentacion/doc_view/176-acta-no-013.html
- [56] Universidad de los Llanos. (2009, Julio) Universidad de los Llanos - Estatuto general. [Online]. http://www.unillanos.edu.co/docus/as_04_2009_estauto%20gral.pdf
- [57] Universidad de los Llanos. (2009, Mar.) Universidad de los Llanos - Programas. [Online]. <http://www.unillanos.edu.co/index.php/programas>
- [58] Congreso de la República. (1992, Dec.) Secretaría del Senado de Colombia. [Online]. http://www.secretariasenado.gov.co/senado/basedoc/ley_0030_1992_pr001.html
- [59] Universidad de los Llanos. (2010, Febrero) Acuerdo Superior Universitario N° 002 de 2010 - Artículo 4° - Párrafo 1. [Online]. <http://web.unillanos.edu.co/docus/Acuerdo%20Superior%20N%C2%B0%20002%20de%202010.pdf>
- [60] Senado de la República de Colombia. (1992, Diciembre) Ley 30 de 1992 - Artículo 70. [Online]. http://www.secretariasenado.gov.co/senado/basedoc/ley/1992/ley_0030_1992_pr_001.html
- [61] Universidad Abierta y a Distancia. (2008, Marzo) Datateca UNAD. [Online]. http://datateca.unad.edu.co/contenidos/434206/434206/leccin_23_curso_academic_o_definicion_y_tipos_de_cursos.html
- [62] Universidad de los Llanos. (2002, Abril) Centro de documentación - Universidad de los Llanos. [Online]. http://documentacion.unillanos.edu.co/index.php/centro-de-documentacion/doc_view/1183-acuerdo-07-de-2002.html

- [63] Rafael Sneider Cuartas Velandia and Olga Lucero Vega Márquez, "Desarrollo de un Prototipo de Integración de Moodle con los Sistemas de Información Académicos de la Universidad de los Llanos," Ingeniería de Sistemas, Universidad de los Llanos, Villavicencio, Tesis de grado /SIS/0244, 2014.
- [64] Universidad de Murcia Um. (2014, Abril) Informática aplicada a la Gestión Pública. [Online]. <http://www.um.es/docencia/barzana/IAGP/IAGP2-Ingenieria-software-introduccion.html>
- [65] Rafael Menéndez and Barzanallana Asensio. (2012, Agosto) Universidad de Murcia- Actualidad Informática. [Online]. <http://www.um.es/docencia/barzana/DIVULGACION/INFORMATICA/Historia-desarrollo-aplicaciones-web.html>
- [66] Sebastián Otero and Carla Maglione. (2013) Escritorio Alumnos- Conectar Igualdad, Competencias en TIC. [Online]. http://escritorioalumnos.educ.ar/datos/recursos/arquitectura_web_1.pdf
- [67] ORACLE. (2013, Junio) Java EE. [Online]. <http://www.oracle.com/technetwork/java/javaee/overview/index.html>
- [68] Php Group. (2002, Enero) Manual introductorio - PHP. [Online]. <http://php.net/manual/es/intro-what-is.php>
- [69] Alicia Cañellas Mayor, "LMS y LCMS: Funcionalidades y beneficios," *C&P Comunicación y Pedagogía, Especial LMS*, no. 1, pp. 263-264, 2006. [Online]. <http://www.centrocp.com/lms-y-lcms-funcionalidades-y-beneficios/>
- [70] Moodle. (2015, Abril) Moodle- Procesos en Moodle. [Online]. https://docs.moodle.org/all/es/Acerca_de_Moodle
- [71] Carnegie Mellon. (2011, Marzo) CMMI Institute. [Online]. <http://cmmiinstitute.com/about-cmmi-institute>
- [72] LARAVEL. Comunidad laravel. [Online]. <http://laraveles.com/docs/4.1/>
- [73] ORACLE. (2013, Junio) Java - Recursos de ayuda. [Online]. https://www.java.com/es/download/what-is_java.jsp
- [74] LARAVEL. Laravel. [Online]. <http://web.archive.org/web/20140920185439/http://laravel.com/docs/introduction>

- [75] LARAVEL. Laravel Comunidad en Español. [Online]. <http://laraveles.com/docs/5.0/eloquent#introduction>
- [76] LARAVEL. Laravel. [Online]. <https://laravel.com/docs/5.0/authentication>
- [77] (2001) php.net. [Online]. <http://php.net/manual/es/intro.oci8.php>
- [78] moodle. (2015, July) moodle desarrollo. [Online]. https://docs.moodle.org/dev/Web_services_API#services.php
- [79] getBootstrap. [Online]. <http://getbootstrap.com/>
- [80] Apache Friends. (2013, Aug.) Apache Friends: XAMPP. [Online]. <http://www.apachefriends.org/es/xampp.html>
- [81] The Software Process Dashboard Initiative. (2014, Oct.) The Software Process Dashboard. [Online]. <http://www.processdash.com/download>
- [82] Stephen R. Schach,,: Mc Graw Hill, 2011, ch. 2, p. 667.
- [83] (2013) Docuemntacion Apache. [Online]. <http://httpd.apache.org/docs/2.0/es/vhosts/>
- [84] php OCI8. PHP Documentation. [Online]. <http://php.net/manual/es/intro.oci8.php>