

**INFORME DE AVANCE FINAL
(PROYECTO DE DISEÑO DE BASES DE DATOS)**

CORTÉS HERNÁNDEZ SERGIO ANDRÉS	258006
LANCHEROS PINTO CARLOS DUVAN	258021
RODRÍGUEZ PORTELA JOHAN DAVID	258145
ROZO BAUTISTA JEISSON DAVID	257974
VERGARA VARGAS JEISSON ANDRÉS	258064

Grupo 22

Repositorio:
<https://github.com/PorcupineSS/psp-swing>
(Porcupine Software Portal)

Página oficial del proyecto:
<http://javergarav.wix.com/dbd2013>

ISMAEL CASTAÑEDA FUENTES
Profesor Titular, Universidad Nacional
Diseño de Bases de Datos

UNIVERSIDAD NACIONAL DE COLOMBIA
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS E INDUSTRIAL
BOGOTÁ D.C.
03 de Julio de 2013

1. INTRODUCCION

Para poder desarrollar de manera adecuada un producto de software para una empresa, es necesario conocer a fondo el funcionamiento de esta. Los aspectos a considerar en este proceso son la estructura organizacional, el entorno de la organización y sus procesos. También es fundamental plantear la forma en que va a trabajar el equipo de desarrollo, desde asignaciones de responsabilidades, cronogramas de actividades y la elección de las herramientas adecuadas para el desarrollo del producto solución.

2. LA EMPRESA

A. Elección

Se eligió a *Porcupine Security Solutions* como entorno de trabajo debido al contexto que manejan. Es un área importante del mercado a nivel nacional y mundial que aportaría un entendimiento esencial frente a este campo vanguardista en el desarrollo de software. Además, nos da una oportunidad de mirar cómo se actualizan procesos que se llevaban a mano y se van a poner al día.

B. Misión

Porcupine Security Solutions se destaca por comienzos pequeños y sueños de grandeza. Una búsqueda de innovación y competencia que sea beneficiosa para la evolución de las prestaciones de servicios que traigan como consecuencia mejor calidad para el usuario. Especializadas en principio en modalidades tipo fija, consultorías, asesorías escoltas a ambos, personas y vehículos, acompañadas por tecnologías de punta que nos permite dar un servicio confiable, garantizado para todos los sectores de la población, sea industrial residencial o comercial.

C. Visión

La empresa *Porcupine Security Solutions* está encaminada hacia un futuro de triunfos que la posicionen para el año 2015 como una de las mejores empresas de prestación de servicios de vigilancia en el sector privado tanto público. Dicha reputación ganada a través de certificaciones de buenas prácticas internacionales demostrará al mercado competitivo nuestra convicción firme para lograr nuestros objetivos, alimentados siempre por el deseo de dar un servicio de calidad y de utilidad pertinente a nuestros queridos usuarios.

D. Objetivo

Aportar una selección de personal, dotados con experiencia y conocimientos profesionales específicos con el fin de prevenir a través de las capacidades físicas y mentales del personal, las amenazas que puedan afectar el ejercicio regular de los derechos legítimos sobre la integridad física o los bienes materiales de un usuario.

E. ESTRUCTURA ORGANIZACIONAL



3. SOBRE EL PROYECTO

El proyecto a desarrollar se trata de un software para el manejo de algunos de los procesos de la empresa *Porcupine Security Solutions*. La aplicación se conocerá como **Porcupine Software Portal**.

4. Metodología de trabajo

Metodología Scrum

Scrum es un proceso ágil para desarrollar software que fue aplicado por primera vez por Ken Schwaber y Jeff Sutherland., quienes lo documentaron en detalle en el libro *Agile Software Development with Scrum*. Esta metodología centra su atención en las actividades de Gerencia y no especifica prácticas de Ingeniería. Fomenta el surgimiento de equipos autodirigidos cooperativos y aplica inspecciones frecuentes como mecanismo de control.

Scrum parte de la base de que los procesos definidos funcionan bien sólo si las entradas están perfectamente definidas y el ruido, ambigüedad o cambio es muy pequeño. Por lo tanto, resulta ideal para proyectos con requerimientos inestables, ya que fomenta el surgimiento de los mismos.

El ciclo de vida definido por Scrum es incremental iterativo y se caracteriza por ser muy adaptable.

Principales características:

- Equipos auto-dirigidos.
- Utiliza reglas para crear un entorno ágil de administración de proyectos.
- No prescribe prácticas específicas de ingeniería.
- Los requerimientos se capturan como ítems de la lista Product Backlog.
- El producto se construye en una serie de Sprints con una duración de 15 a 30 días.

Roles y Responsabilidades:

Scrum Master

Es un rol de administración que debe asegurar que el proyecto se está llevando a cabo de acuerdo con las prácticas, valores y reglas de Scrum y que todo funciona según lo planeado. Su principal trabajo es remover impedimentos y reducir riesgos del producto. Este papel es desempeñado por Duvan Lancheros.

Product Owner

Es el responsable del proyecto, administra, controla y comunica la Backlog List. Es el responsable de encontrar la visión del producto y reflejarla en la Backlog List, el responsable es Jeisson Vergara.

Scrum Team

Es el equipo del proyecto que tiene la autoridad para decidir cómo organizarse para cumplir con los objetivos de un Sprint. Sus tareas son:

Effort Estimation (Estimar Esfuerzo), crear el Sprint Backlog, revisar la Product Backlog List y sugerir obstáculos que deban ser removidos para cumplir con los ítems que aparecen.

El equipo es conformado normalmente por 8 personas, pero en este caso nuestro equipo de trabajo es conformado por 5 personas cada una especializada en algún

elemento que conforma los objetivos a cumplir, por ejemplo: Programadores, Diseñadores de Interfaz de usuario, etc. La dedicación de los miembros del equipo debería ser full-time, pero debido a que no se puede lograr esto debido a que nuestra disponibilidad de tiempo es limitada tendremos algunas excepciones. La membresía solo puede cambiar entre sprints (no durante).

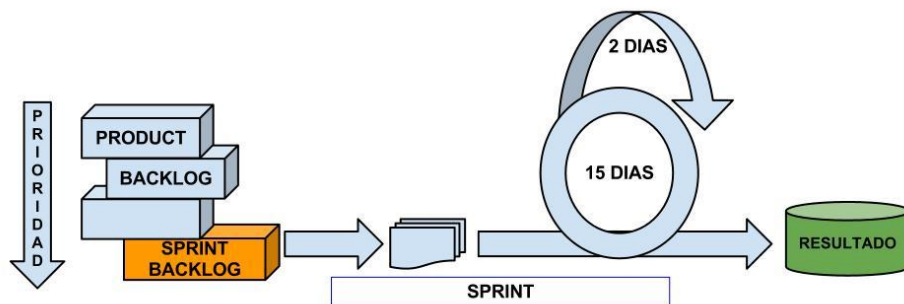
Customer

El cliente participa en las tareas que involucran la lista Product Backlog, en nuestro caso este papel es desempeñado por el Docente.

Management

Es el responsable de tomar las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto. Participa en la selección de objetivos y requerimientos y en la selección del Scrum Owner. Tiene la responsabilidad de controlar el progreso y trabaja junto con el Scrum Master en la reducción de la Product Backlog List, el encargado de este papel es Sergio Cortés.

Esquema General Usado por la Metodología:



5. Requerimientos y validación

El desglose de los requerimientos para este caso de estudio puede apreciarse en el documento anexo “Especificación de Requerimientos”. Se siguió el estándar de la IEEE (830) para describir los criterios de aceptación del software.

La validación del modelo se hizo consultando con compañeros de la carrera que han desarrollado modelos similares y complementando el nuestro.

6. Herramientas

A. Frameworks

- Java (Swing, JSF y otros Frameworks de vista).
- Como unidad de persistencia se utilizará EclipseLink.

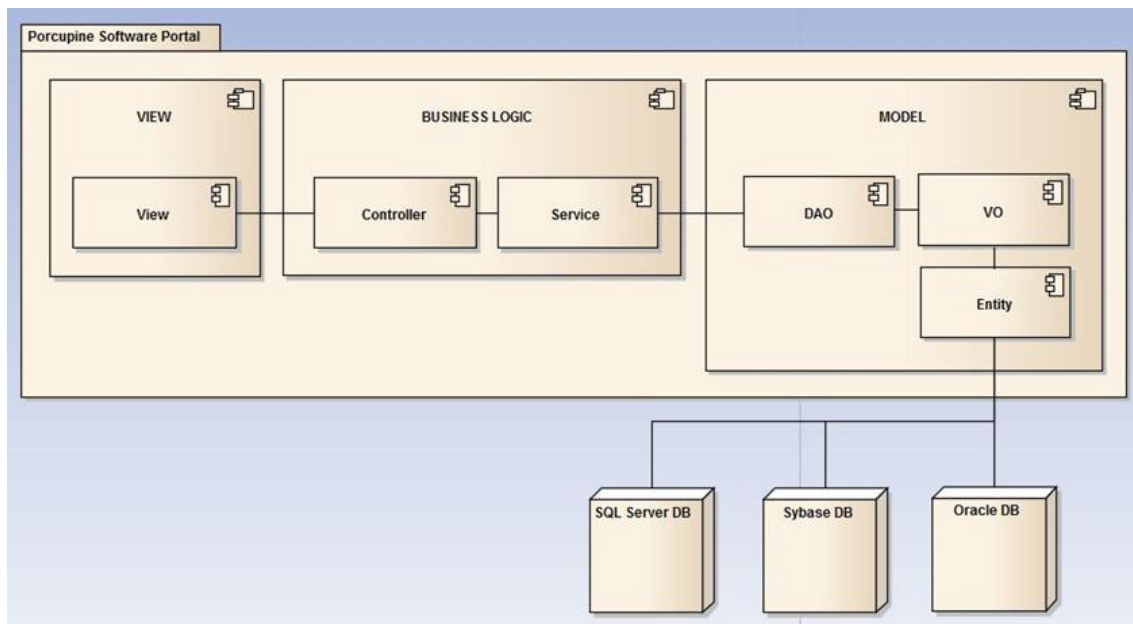
B. Lenguajes de Programación

- Para desarrollar nuestro producto se ha elegido el lenguaje de programación Java, debido a que es el lenguaje que cumplen todas las exigencias necesarias para nuestra aplicación.

C. Utilidades

- En nuestro equipo de trabajo estaremos utilizando distintos tipos de utilidades, que satisfacen nuestros distintos tipos de necesidades:
- *Herramienta de diseño:* PowerDesigner.
- *Herramientas para almacenamiento e intercambio de archivos:* Google Drive, Dropbox.
- *Formas de comunicación:* Hangouts de Google+, Grupo de Facebook, Correos electrónicos (institucional y personal).
- *IDE:* NetBeans.
- *Sistemas de Gestión de Bases de Datos:* Oracle SQL Developer, Microsoft SQL Server, Sybase.
- *Repositorio:* GIT.

7. Descripción de la Arquitectura



A. VIEW

View: capa que permite la interacción entre el Usuario y todas las funcionalidades por la aplicación Porcupine Software Portal.

B. BUSINESS LOGIC

Controller: capa que recibe las solicitudes del usuario, invoca procesamiento en los componentes del modelo, determina cual vista debe ser desplegada, define el flujo de la aplicación y realiza el mapeo entre las acciones del usuario y la modificación del modelo.

Service: capa que se encarga de administrar las transacciones provenientes del Controller. Se comunica con el Modelo.

C. MODEL

Dado el énfasis que se dio a avanzar el modelo de bases de datos, no se dedican muchos recursos al adelantamiento de la aplicación cliente. El objetivo concerniente a esta parte consistía en hacer la conexión con la base de datos, hacer una ingeniería inversa de las tablas en ella y convertirlas a código java. Se construyeron los objetos de valor que permitirán la comunicación de la información a través del modelo a varias capas manejado en la aplicación. Luego se construyeron las capas de acceso a datos y de servicios que sirven para tener un buen flujo de la información. Se comenzó con el diseño de las interfaces y de los controladores de conexión con las capas más bajas de la aplicación.

Hasta el momento de la presentación se logró diseñar 7 interfaces, pero solo 2 han sido probadas y hace falta más trabajo antes de la próxima entrega.

- **Login:** Interfaz que permite el acceso a un empleado al sistema de información. Aunque no está implementado aún, la funcionalidad después de validar al usuario consistirá en dirigirlo a la pantalla de acción que le corresponda según su rol.
- **Seleccionar el motor base de datos:** Diseñado sólo para propósitos de prueba, este selector permite elegir en qué motor se harán las llamadas y las persistencias a la base de datos.
- **Agregar implemento:** Permite agregar un implemento de seguridad al inventario de la empresa
- **Crear empleado:** Permite enrolar un empleado
- **Encontrar persona:** Permite hacer una búsqueda en los registros por alguna persona en particular.
- **Crear cliente:** Permite enrolar un cliente
- **Encontrar cliente:** Permite hacer una búsqueda en los registros por algún cliente en particular.
- **Gestión de recursos humanos:** Menú principal del rol de director de recursos humanos
- **Coordinación Técnica y Tecnológica:** Menú principal del rol del coordinador técnico y tecnológico
- **Director Comercial y Gerencia:** Menu principal de los roles de Director Comercial y de Subgerente
- **Consulta de bitacora:** Se pueden ver los cambios que han ocurrido en la base de datos.

La generación de gráficas de ayuda en toma de decisión no ha sido implementada. Se ha discutido las posibilidades de utilizar los datos en este modelo para darle al cliente funcionalidad adicional, y se nos ha ocurrido lo siguiente:

- Elección del trabajador del mes (Grafica con cantidad de contratos en un mes)
- Evaluación de calidad de implementos (Conceptos de actualización en los implementos)

10. Diseño de la navegación

Id Proceso: 01

Nombre del Proceso: Logearse



Id Proceso: 02

Nombre del Proceso: Crear Empleado



Id Proceso: 03

Nombre del Proceso: Buscar Persona



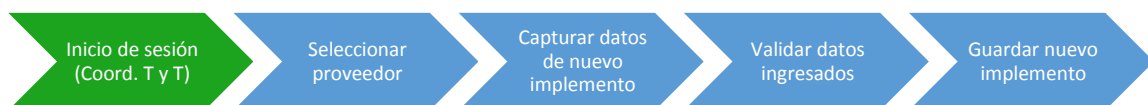
Id Proceso: 04

Nombre del Proceso: Realizar informe de Recursos Humanos



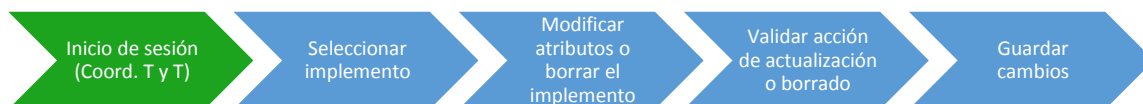
Id Proceso: 05

Nombre del Proceso: Registrar Implemento



Id Proceso: 06

Nombre del Proceso: Actualizar y Borrar Implemento



Id Proceso: 07

Nombre del Proceso: Realizar Control de Inventario



Id Proceso: 08

Nombre del Proceso: Asignar Implemento a Empleado Temporal



Id Proceso: 09

Nombre del Proceso: Despojar Implemento de Empleado Temporal



Id Proceso: 10

Nombre del Proceso: Exponer Comunicado



Id Proceso: 11

Nombre del Proceso: Consultar Contrato



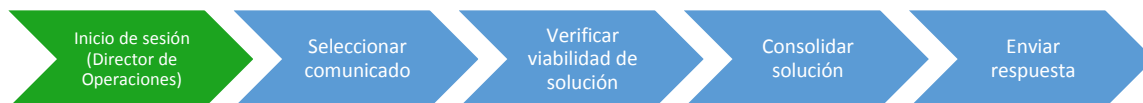
Id Proceso: 12

Nombre del Proceso: Asignar Contrato



Id Proceso: 13

Nombre del Proceso: Responder Comunicado



Id Proceso: 14

Nombre del Proceso: Consultar Empleado Temporal



Id Proceso: 15

Nombre del Proceso: Consultar Inventario



Id Proceso: 16

Nombre del Proceso: Consultar Contrato



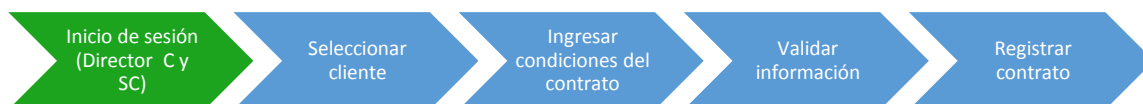
Id Proceso: 17

Nombre del Proceso: Registrar Cliente



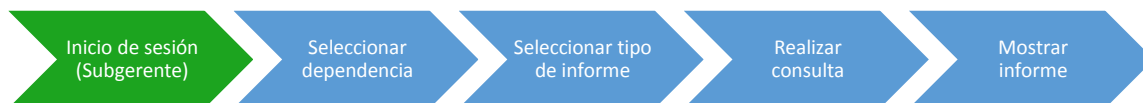
Id Proceso: 18

Nombre del Proceso: Registrar Contrato con Cliente



Id Proceso: 19

Nombre del Proceso: Consultar Informe



Desarrollo e implementación

ESPECIFICACIONES BD

Se ha trabajado principalmente en tres items para poder cumplir con los tres aspectos de integridad, seguridad y privacidad.

REGLAS

- Los teléfonos celulares de los empleados, clientes y proveedores deben ser de operadores válidos en Colombia, es decir, deben comenzar por 300, 301, 311, 312, 313, 314, 315, 316, 317, 318, 320, 321.
- El precio unitario de los Implementos de Seguridad no debe ser negativo.
- La cantidad de los Implementos de Seguridad no debe ser negativa.
- La cantidad de personal de un Contrato no puede ser negativa.
- El costo mensual de un Contrato no puede ser negativo.
- La fecha de un Comunicado no puede ser posterior a la fecha actual.
- La cantidad de implementos Asignados a un Empleado Temporal no puede ser negativa.
- La fecha de una asignación de implementos no puede ser posterior a la fecha actual.
- Un Empleado Temporal tiene o NO tiene contrato, pero ambas condiciones no pueden darse simultáneamente. En otras palabras, su estado de contrato solo puede tener los valores activo o inactivo (true o false).
- Un Empleado Temporal NO puede tener más de un Contrato asignado al mismo tiempo.

- Los empleados temporales sólo pueden ser del tipo Escolta o Vigilante.
- Los únicos que pueden asignar contratos son los Coordinadores de contrato.
- Cuando se actualiza un implemento (se añadan más a la Bodega), la cantidad de implementos Agregados no puede ser negativa.

TRIGGERS

Los triggers implementados hasta el momento son los que se requieren para el uso de la bitácora de seguimiento, se ha decidido realizar tres disparadores para cada tabla del sistema, cada disparador corresponde a insert, delete y update respectivamente, también se decidió por estandarizar los nombres de los triggers, a continuación se lista el nombre de la tabla y el nombre de sus triggers asociados.

La notación para los triggers es TR_(Tipo de Trigger)_(Nombre de tabla simplificado)), donde el tipo de Trigger corresponde con: INS para “ON INSERT”, DEL para “ON DELETE” y “UPD” para ON UPDATE.

El listado de triggers generados para cada motor es el siguiente:

- **EMPLEADOS:** TR_INS_EMPLEADOS, TR_DEL_EMPLEADOS y TR_UPD_EMPLEADOS.
- **EMP_PLANTA:** TR_INS_EMPLANTA, TR_DEL_EMPLANTA y TR_UPD_EMPLANTA.
- **EM_TEMP:** TR_INS_EM_TEMP, TR_DEL_EM_TEMP y TR_UPD_EM_TEMP.
- **EM_TIENE_TELS:** TR_INS_EMTELS, TR_DEL_EMTELS y TR_UPD_EMTELS.
- **ACTUALIZACION:** TR_DEL_ACTUAL, TR_INS_ACTUAL y TR_UPD_ACTUAL.
- **COMUNICADO:** TR_INS_COMUN, TR_DEL_COMUN y TR_UPD_COMUN.
- **ASIGNACION_C:** TR_DEL_ASIGC, TR_INS_ASIGC y TR_UPD_ASIGC.
- **CLIENTE:** TR_INS_CLIEN, TR_DEL_CLIEN y TR_UPD_CLIEN.
- **CONTRATO:** TR_INS_CONT, TR_DEL_CONT y TR_UPD_CONT.
- **IMPL_SEGURIDAD:** TR_INS_IMSEG, TR_DEL_IMSEG y TR_UPD_IMSEG.
- **PROVEEDOR:** TR_INS_PROV, TR_DEL_PROV y TR_UPD_PROV.
- **ASIG_IMPL:** TR_INS_ASMP, TR_DEL_ASMP y TR_UPD_ASMP.
- **ACTU_IMPL:** TR_INS_ACIMP, TR_DEL_ACIMP y TR_UPD_ACIMP.

VISTAS

Para la creación de vistas y procedimientos lo primero que se consideró fue el acceso a las entidades por medio de Vistas CRUD. Posteriormente se hizo énfasis en las necesidades de usuario y los requerimientos. A continuación se listan las vistas y procedimientos identificados:

- **Tabla: BITACORA_SEG:**

- i. Todos los registros en la bitácora de seguimiento.
Nombre de vista asociada: **V_BITACORA.**
- ii. Dada una fecha indicar los registros que se realizaron en la bitácora que corresponden a esta fecha.
Nombre del procedimiento asociado: **P2_BITACORA.**
- iii. Dada una tabla indicar los registros que se realizaron en la bitácora que corresponden a esta tabla.
Nombre del procedimiento asociado: **P1_BITACORA.**
- iv. Dada una operación indicar los registros que se realizaron en la bitácora que corresponden a esta operación.
Nombre del procedimiento asociado: **P3_BITACORA**

- **Tabla: EMPLEADOS:**

- i. **Vista para operaciones CRUD**
Nombre de vista asociada: **V_EMPL**
- ii. Dada la cédula de un empleado temporal indicar datos personales (nombre completo, sueldo actual, fecha de ingreso, tiempo que lleva vinculado en la empresa en meses), contratos en los que ha estado vinculado incluyendo el actual e implementos asignados.
- iii. Dada la cédula de un empleado de planta indicar datos personales (nombre completo, edad, teléfonos, sueldo actual, fecha de ingreso, tiempo que lleva vinculado en la empresa en meses),
- iv. Empleados temporales sin contrato asignado actualmente.
- v. Dado un cargo perteneciente a la empresa, indicar los datos personales de los empleados de planta con este cargo.

- **Tabla: COMUNICADO:**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_COMUN.**

- ii. Solicitudes, quejas y reclamos junto a su remitente.

- iii. Solicitudes, quejas y reclamos junto a su remitente, sin respuesta.

- **Tabla: CLIENTE:**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_CLIENTE.**

- ii. Todos los clientes registrados en la empresa junto a sus datos más relevantes (Nombre del cliente, dirección, y tiempo de vinculación en la empresa).

- iii. Dado el nombre de un cliente indicar los contratos que éste ha firmado con la empresa y el contrato actual si lo tiene.

- **Tabla: ASIGNACION C (asignación de contrato)**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_ASIGC.**

- ii. Asignaciones de empleados temporales. (Incluyendo fecha de la asignación, código de contrato al cual se le asigna el empleado temporal, empleado de planta que asigna al empleado temporal y empleado temporal asignado.

- b. Tabla: CONTRATO**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_CONT.**

- ii. Dado el código de un contrato indicar las características más relevantes de este. (Código del contrato, Cliente que solicita el

contrato, Ubicación del contrato, Teléfonos, Tipo de contrato, Fecha de registro, Fecha de inicio, Costo mensual, Tipo de personal solicitado y Empleados temporales asignados si es que ya se le han asignado).

iii. Contratos sin empleados temporales asignados.

- **Tabla: IMPL SEGURIDAD**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_IMSEG.**

- ii. Implementos de seguridad registrados en la empresa junto a sus características más relevantes (Nombre del implemento, Nombre del proveedor, Precio unitario , Cantidad actual registrada, Cantidad actual asignada, Cantidad actual sin asignar, Descripción, Fecha de registro).

- iii. Dado el nombre del implemento indicar la cantidad de implementos disponible para asignar.

- **Tabla: PROVEEDOR**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_PROV.**

- ii. Proveedores registrados en la empresa junto con sus características más relevantes (Nombre del proveedor, Dirección, Tiempo que lleva vinculado en la empresa.)

- iii. Dado el nombre de un proveedor indicar los productos que éste distribuye en nuestra empresa junto con sus precios.

- **Tabla: ASIG IMPL**

- i. **Vista para operaciones CRUD.**

- Nombre de vista asociada: **V_AIMPL.**

- ii. Asignaciones de implementos de seguridad. (Incluyendo fecha de la asignación, empleado de planta que asigna el implemento y empleado temporal al cual se le es asignado el implemento).

- **Tabla: ACTU IMPL**
 - i. **Vista para operaciones CRUD.**
Nombre de vista asociada: **V_ACIMPL.**
 - ii. Actualizaciones del inventario de los implementos de seguridad.

Procedimientos:

Para el manejo de concurrencias y transacciones se crearon los siguientes procedimientos, estos procedimientos están enfocados únicamente al componente de manejo de implementos

- **ADICIONAR_IMPLEMENTO (ID_I IN NUMBER, CANT_AD IN NUMBER):**
Este procedimiento sirve para aumentar la cantidad de un implemento específico con identificador ID_I un número CANT_AD de elementos.
- **ASIGNAR_IMPLEMENTO (ID_I IN NUMBER, ID_EM_TEM IN NUMBER, ID_COOR IN NUMBER, CANT_ASIG IN NUMBER):** Este procedimiento sirve para asignar una cantidad CANT_ASIG de implemento de identificación ID_I a un empleado temporal identificado por ID_EM_TEM. Todo esto realizado por un coordinador tecnológico que tiene el id ID_COOR.
- **DESPOJAR_IMPLEMENTO (ID_I IN NUMBER, ID_EM_TEM IN NUMBER, ID_COOR IN NUMBER, CANT_DESP IN NUMBER):** Este procedimiento sirve para quitar una cantidad CANT_ASIG de implemento de identificación ID_I a un empleado temporal identificado por ID_EM_TEM. Todo esto realizado por un coordinador tecnológico que tiene el id ID_COOR.
- **P1_BITACORA (TABLA IN VARCHAR2, RESPUESTA IN OUT INT):**
Devuelve las columnas de la bitácora en que se hicieron operaciones sobre la tabla TABLA.
- **P2_BITACORA (FECHA IN DATE, RESPUESTA IN OUT INT):** Devuelve las columnas de la bitácora en que se hicieron operaciones en la fecha FECHA.

- **P3_BITACORA (OPERACION IN VARCHAR2, RESPUESTA IN OUT INT):**
Devuelve las columnas de la bitácora en que se hizo un tipo de operación específica OPERACIÓN. Las opciones posibles son: INSERT, UPDATE, DELETE, CREATE.

11. Conclusiones

- Para generar un modelo adecuado para un sistema de información, es necesario conocer a fondo la estructura organizacional de la empresa, así como las interacciones entre sus elementos.
- El hecho de conocer en su totalidad las tareas que hacen los trabajadores, permite categorizarlos de tal modo que se generen estereotipos con funciones definidas, y esto genera gran facilidad a la hora de crear requisitos funcionales.
- En el transcurso de un proyecto puede haber cambios en la estructura del equipo de trabajo (puede haber personas que cancelen para el caso del curso), por lo tanto, se hace necesario considerar las variaciones definiendo un alcance concreto.
- Dadas las variaciones de disponibilidad de tiempo en el ambiente del grupo de trabajo (en la universidad se generan problemas que tienden a modificar el calendario de trabajo), la metodología de desarrollo fue lo más adaptable posible, llegándose a concluir que la mejor opción para el caso presente es el uso de metodologías ágiles (como SCRUM).
- A la hora de generar un modelo lógico, es necesario tener control de asociaciones cíclicas para evitar problemas en la generación de un modelo físico.
- La salida de miembros del grupo de trabajo hizo reducir la dedicación que se podía dar al proyecto.
- El uso del repositorio ha demostrado ser de vital importancia para el desarrollo de proyectos en cual intervienen múltiples personas debido a la dinámica de trabajo que permite, en la que los avances pueden hacerse independientemente de la geo localización de las personas y del tiempo que cada integrante tenga disponible para el proyecto.
- El uso de repositorios en GitHub permiten mantener un registro gráfico de la efectividad por persona dentro del proyecto y lograr así, de ser necesario, tener razones de justificación para los procesos de gerencia dentro del equipo de desarrollo.
- A pesar de que SQL es considerado un lenguaje estandarizado, se hace notable la diferencia de la implementación dependiendo del motor, más que todo a la hora de implementar triggers y procedimientos.
- Es importante manejar un estándar al momento de nombrar los diferentes objetos creados en la base de datos, esto facilita la identificación de cada elemento por los miembros del grupo, como también da la facilidad de reutilizar código al momento

de desarrollar la aplicación, pues los objetos tendrán el mismo nombre independientemente del motor usado.

- Es de vital importancia que todo lo realizado para poder cumplir con el comportamiento deseado de la base de datos sea hecho directamente en la base de datos, con eso se puede garantizar la integridad, privacidad y seguridad se cumplan independientemente del método de acceso (Usuarios, aplicativos, otro sistema).