Sistema de procesos de inventarios

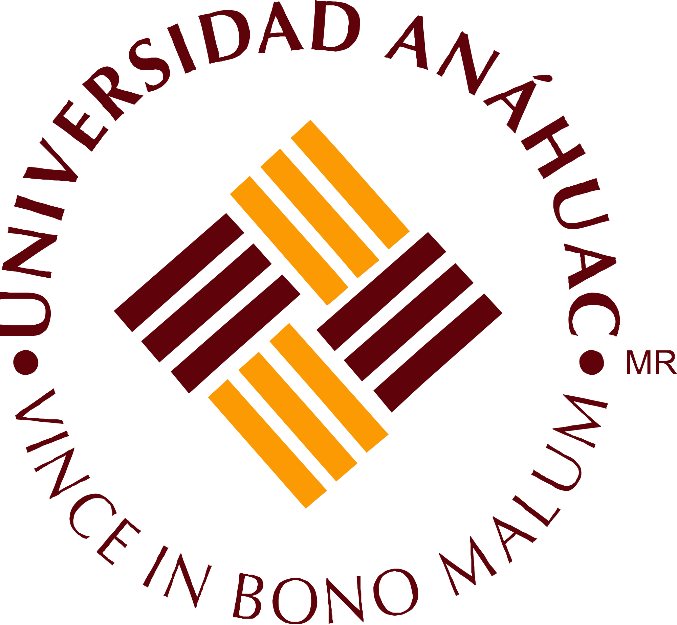
PROYECTO DE INGENIERÍA DE SOFTWARE

en la

UNIVERSIDAD ANÁHUAC OAXACA

durante el periodo enero – mayo de 2021

por Erick Guzmán Ramos, Jehiely Ruiz Ignacio y Jorge Domínguez Jiménez



DOCUMENTO DE REQUERIMIENTOS DE SOFTWARE (SRD)

VERSION 0-1-0

# A. Abstract

Este documento contiene los requisitos de software para el sistema de procesos de inventarios. Este proyecto es el proyecto final del curso de Ingeniería de Software en la Universidad Anáhuac Oaxaca. El sistema de procesos de inventarios es parte de un proyecto para la constructora COCOVA. En esta asignación consideramos el sistema de roles y la seguridad de la comunicación del sistema a construir. Proporcionaremos un diseño y una implementación de este diseño. Los requisitos de software se construyen a partir de los requisitos del usuario como se indica en el [URD]. El documento cumple con el Documento de requisitos de software (SRD) del Estándar de ingeniería de software, según lo establecido por la Agencia Espacial Europea [ESA].

**B. Tabla de contenido**

[A. Abstract 2](#_Toc63720359)

[C. Hoja de estado del documento 5](#_Toc63720360)

[1. Introducción 6](#_Toc63720361)

[1.1 Propósito 6](#_Toc63720362)

[1.2 Alcance 6](#_Toc63720363)

[1.3 Definiciones, acrónimos y abreviaturas. 6](#_Toc63720364)

[1.4 Referencias 7](#_Toc63720365)

[1.5 Visión general 7](#_Toc63720366)

[2. Descripciones generales 8](#_Toc63720367)

[2.1 Relación de proyectos actuales 8](#_Toc63720368)

[2.2 Relación con proyectos predecesores y sucesores 8](#_Toc63720369)

[2.3 Función y propósito 8](#_Toc63720370)

[2.4 Ambiente de desarrollo 8](#_Toc63720371)

[2.5 Relación con otros sistemas 8](#_Toc63720372)

[2.6 Limitaciones generales 8](#_Toc63720373)

[2.7 Descripción de modelo 9](#_Toc63720374)

[2.7.1 Modelo de alto nivel 9](#_Toc63720375)

[2.7.2 Clases de roles de sistema 9](#_Toc63720376)

[2.7.3 Clases cliente administrador 9](#_Toc63720377)

[2.7.4 Clases producto obra 10](#_Toc63720378)

[2.7.5 Diagrama completo 11](#_Toc63720379)

[3. Requerimientos específicos 12](#_Toc63720380)

[3.1 Requerimientos de funcionalidad 12](#_Toc63720381)

[3.1.1 Clases de Roles de sistema 12](#_Toc63720382)

[3.1.2 Clases producto obra 13](#_Toc63720383)

[3.2 Requerimientos de desempeño 14](#_Toc63720384)

[3.3 Requerimientos de interfaz 14](#_Toc63720385)

[3.4 Requerimientos operacionales (prototipo) 15](#_Toc63720386)

[3.5 Requerimientos de recursos 15](#_Toc63720387)

[3.6 Requerimientos de verificación 15](#_Toc63720388)

[3.7 Requerimientos de portabilidad 15](#_Toc63720389)

[3.8 Requerimientos de mantenimientos 15](#_Toc63720390)

[3.9 Requerimientos de confiabilidad 15](#_Toc63720391)

[3.10 Requerimientos de seguridad 15](#_Toc63720392)

[3.11 Requerimientos de documentos 15](#_Toc63720393)

[4. Matriz de trazabilidad de requerimientos 16](#_Toc63720394)

[4.1 Requerimientos de software a requerimientos de usuario 16](#_Toc63720395)

[E. Apéndice 1. PROTOTIPO A MODELO LÓGICO 18](#_Toc63720396)

[F. APÉNDICE 2: CASOS DE USO 19](#_Toc63720397)

[4.2.1 Vista de reportes del usuario final 19](#_Toc63720398)

[4.2.2 El usuario final da de alta productos de material 20](#_Toc63720399)

[4.2.3 El superusuario asigna roles a usuario 21](#_Toc63720400)

# C. Hoja de estado del documento

|  |  |
| --- | --- |
| Nombre del documento | Documento de requerimientos del sistema |
| Ubicación del documento | Archivos/SRD.docx |
| Autores | Erick Guzmán Ramos, Jehiely Ruiz Ignacio, Jorge Domínguez Jiménez |
| Versión | 0-1-0 |
| Estado | **Borrador** / **Aceptado Internamente** / Aprobado Condicionalmente / Aprobado |

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Fecha | Autores | Resumen |
| 0-0-1 | 02/02/2021 | Erick Guzmán Ramos, Jehiely Ruiz Ignacio, Jorge Domínguez Jiménez | Creación del documento |

# 1. Introducción

## Propósito

El propósito del Documento de requisitos de software (SRD por sus siglas en inglés) es describir formalmente la funcionalidad del software que se realizará en este proyecto. No describe de ninguna manera cómo se implementará esta funcionalidad.

## Alcance

El software que se va a crear se denomina Sistema de procesos de inventarios. El objetivo principal del sistema es gestionar la entrada y salida de material para obras de la constructora de tal manera que se tengan reportes precisos y actualizados del material disponible para nuevos proyectos. El almacenamiento de los datos del material deberá ser capturado por el residente a cargo y los reportes serán visualizados por el área administrativa.

## Definiciones, acrónimos y abreviaturas.

|  |  |
| --- | --- |
| SPI | El nombre del proyecto: Sistema de Procesos de Inventarios |
| Cliente | Un cliente del Sistema |
| Usuario Final | Persona que almacena / recupera datos en / del SPI |
| Cliente administrador | La persona que maneja el sistema en el lado del cliente |
| Documento | La entidad que cuya información es almacenada en SPI |
| Diagrama de secuencia | Diagrama que muestra la interacción entre dos objetos |
| Modelo lógico | Descripción simplificada del producto a realizar y la funcionalidad que se encuentra en los requisitos del usuario. |
| Orientado a objetos | Paradigma de programación |
| Rol de persona | Esta es la abstracción funcional de un usuario final en el sistema. |
| Rol | Una entidad que posee derechos posee documentos y administra otros roles. |
| RUD | Read, Update, Delete |
| SRD | Software Requirements Document |
| Derechos estáticos | El conjunto de derechos sobre documentos o roles que tiene un rol de persona o un rol, que serán heredados por los administradores de ese rol. |
| UAO | Universidad Anáhuac Oaxaca |
| UML | Unified Modeling Language |
| URD | User Requirements Document |

## Referencias

|  |  |
| --- | --- |
| [ESA] | ESA Software Engineering Standards, ESA PSS-05-02 Issue March 1995  ESA Board of Software Standardization and Control (BSSC) ISBN 0-13-106568-8 |
| [SCMP] | Software Configuration Management Plan, del proyecto SPI. SCMP/SCMP-0-0-1.\* |
| [URD] | User Requirements Document, del proyecto SPI.  URD/URD-0-0-1.\* |

## Visión general

En el capítulo dos se ofrece una descripción del SPI y su entorno. Contiene un modelo que describe el software. El capítulo tres contiene los requisitos de software. Este capítulo también contiene una descripción del prototipo. El cuarto capítulo presenta las matrices de trazabilidad, que vinculan los requisitos del usuario con los requisitos del software.

# Descripciones generales

## Relación de proyectos actuales

El Sistema de Procesos de Inventarios es el único proyecto que realizar durante el periodo, pero es importante resaltar que existe una vinculación de varias materias, tales como Minería de Datos, impartida a estudiantes de ingeniería industrial, Bases de datos Avanzadas e Ingeniería de software, ambas cursadas por estudiantes de ingeniería en sistemas y T.I.

## Relación con proyectos predecesores y sucesores

Este sistema es un proyecto que será el primero de la constructora COCOVA, por lo que queda a disposición de siguientes versiones del sistema.

## Función y propósito

Este documento traduce todos los requerimientos que se encuentran en el [URD] en requerimientos de software. Esto se hace analizando todos los requisitos de usuario e identificando dónde se utilizarían en el producto. A continuación, se realizará un modelo lógico, este proporciona una descripción simplificada del producto y contiene la funcionalidad que se encontró en los requerimientos del usuario. El modelo especifica las relaciones entre diferentes entidades del producto. En esta etapa, se debe evitar la terminología de implementación para garantizar que todas las opciones de implementación sean posibles. La implementación que se elegirá para el proyecto se discutirá a un alto nivel en la fase de Diseño Arquitectónico. Para obtener más información, consulte el documento de diseño arquitectónico [ADD] y el documento de diseño detallado [DDD]. Este documento también contiene una descripción del prototipo SDA. El prototipo ofrece una definición visual de la funcionalidad y la interfaz de usuario del sistema.

## Ambiente de desarrollo

El entorno en el que se ejecuta el sistema se describe generalmente en el [URD].

Las configuraciones del servidor cliente y del servidor de archivo, en el que ejecutaremos las pruebas, son:

• Sistema operativo Windows

Para que el sistema funcione correctamente, se requiere que el usuario final tenga un navegador web.

## Relación con otros sistemas

Este proyecto no tiene sistemas externos conectados. En el futuro podrían a ver sistemas externos conectados.

## Limitaciones generales

Las limitaciones generales del sistema están descritas en el [URD]

## Descripción de modelo

El modelo lógico se construirá como orientado a objetos, con lenguaje UML. El modelo lógico partirá directamente de los requerimientos anteriormente descritos. Las correcciones del modelo respecto a las funcionalidades del SPI pueden encontrarse en el apéndice 2 de este documento. Las relaciones mostradas entre las clases no están relacionadas con la implementación si se encuentran mejores soluciones. El modelo lógico es una herramienta que hace posible analizar posibles soluciones.

### Modelo de alto nivel

El modelo lógico es construido identificando las interfaces externas del sistema y por casa interfaz, considerando la funcionalidad deseada. Para el SPI pueden ser identificadas cinco\* interfaces correspondientes a los diferentes usuarios del sistema.

Las clases en el modelo lógico están directamente derivadas de los requerimientos de usuario. Cada clase representa una entidad encontrada en el [URD].

### Clases de roles de sistema

El sistema de roles es una de las partes más complejas del modelo, en esta sección se describen los roles y sus relaciones.

Cada usuario de la empresa tendrá un rol personal y estos serán generados únicamente por el super usuario administrador. Los roles pueden ser de administración, almacenistas o supervisores de obra.

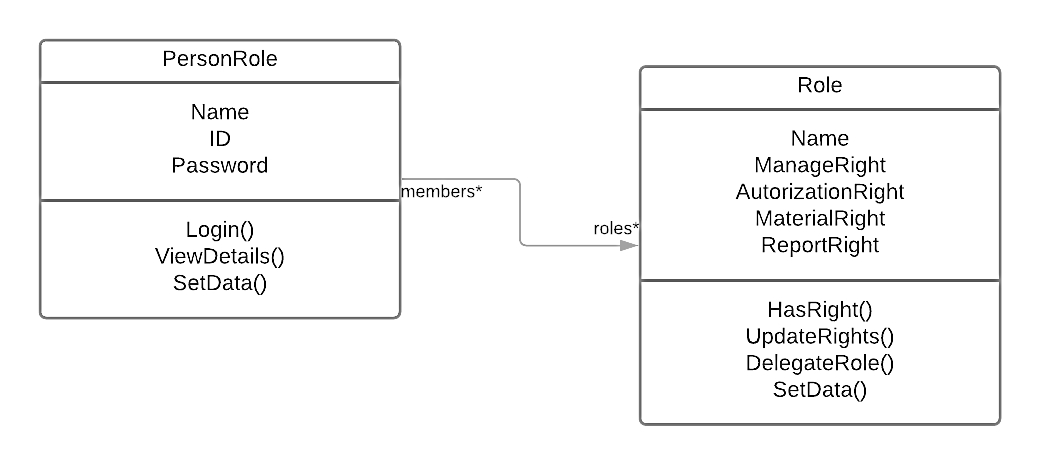


Figure Diagrama de roles

### Clases cliente administrador

El cliente administrador es quien controla la jerarquía de roles, es este usuario quien puede crear nuevos usuarios y eliminarlos también.

### Clases producto obra

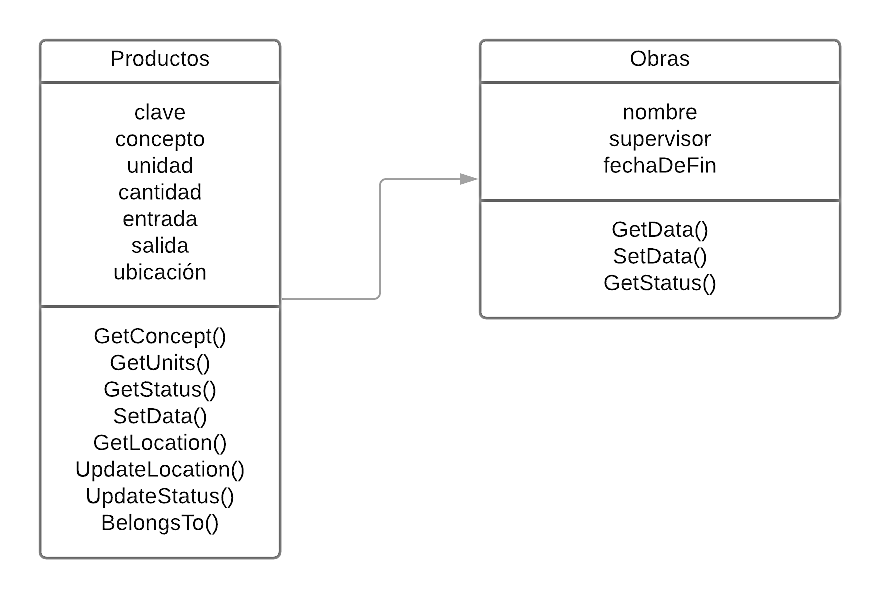


Figure Diagrama de productos – obras

Los productos que se tienen en inventario deben pertenecer a una obra que puede estar en proceso o terminada, la relación se genera también puesto que es el supervisor de obra quien lleva parte del proceso de inventarios.

### Diagrama completo

Diagrama

Descripción generada automáticamente

Figure Diagrama de clases completo

# 3. Requerimientos específicos

## 3.1 Requerimientos de funcionalidad

Estos requisitos son descritos usando la definición de las clases del modelo lógico. Todo requerimiento esta referenciado en el [URD].

### 3.1.1 Clases de Roles de sistema

**Person role**

Representa un empleado individual de la empresa

*Atributos*

* Name

[SRFUR001] Un nombre para identificarse

* ID

[SRFUR002] Una clave única que identifique al usuario en el sistema

* Password

[SRFUR003] Única contraseña para verificar la autenticidad del usuario

*Métodos*

* Login(password)

[SRFUR004] El login es usado por los usuarios finales para ingresar al sistema. Este método verifica la contraseña.

* ViewDetails()

[SRFUR005] Devuelve el nombre, rol y ID del usuario

* SetData(name, id, psswrd)

[SRFUR036] Agrega los datos del usuario

**Role**

Representa un rol al que un usuario pertenezca

*Atributos*

* Name

[SRFUR006] El nombre único del rol

* ManageRight

[SRFUR007] Determina si el rol tiene permisos de crear o eliminar usuarios

* AutorizationRight

[SRFUR008] Determina si el rol tiene permisos para autorizar salida o entrada de materiales

* MaterialRight

[SRFUR009] Determina si el rol tiene permisos para dar de alta materiales

* ReportRight

[SRFUR010] Determina si el rol tiene permisos para ver y crear reportes de inventarios

*Métodos*

* HasRight(right )

[SRFUR011] Verifica si el rol tiene un permiso específico

* UpdateRights(kind,value)

[SRFUR012] Actualiza los permisos que tiene el rol

* DelegateRole(PersonRole,timeout)

[SRFUR013] Este método crea la relación entre Role y Person Role

* SetData(name,mr,ar,matr,rr)

[SRFUR035] Agrega los datos del rol

### 3.1.2 Clases producto obra

**Producto**

Representa un producto del material utilizado en obras

*Atributos*

* Clave

[SRFUR014] Una clave única para identificar el producto

* Concepto

[SRFUR015] El nombre del producto

* Unidad

[SRFUR016]

* Cantidad

[SRFUR017] El número de piezas que hay del producto

* Entrada

[SRFUR018] Determina una entrada del producto

* Salida

[SRFUR019] Determina una salida del producto

* Ubicación

[SRFUR020] En dónde se encuentra el producto

*Métodos*

* GetConcept()

[SRFUR021] Devuelve el nombre del producto

* GetUnits()

[SRFUR022] Devuelve las unidades del producto

* GetStatus()

[SRFUR023] Devuelve el estatus del producto, ya sea si se encuentra como entrada de material o de salida

* GetLocation()

[SRFUR024] Devuelve la ubicación del producto

* UpdateLocation(location)

[SRFUR025] Actualiza el valor de la ubicación del producto

* UpdateStatus(status)

[SRFUR026] Actualiza el estatus del producto a entrada o a salida de material

* BelongsTo()

[SRFUR027] Genera la relación de a qué obra pertenece el producto

* SetData(id,concept,unit,cant,in,out,loc)

[SRFUR028] Agrega los datos de la obra

**Obra**

Representa un proyecto de obra de la empresa

*Atributos*

* Nombre

[SRFUR029] Nombre de la obra

* Supervisor

[SRFUR030] El usuario a cargo de la obra

* Finalización

[SRFUR031] Fecha de culminación de la obra

*Métodos*

* SetData(name, super, finalDate)

[SRFUR032] Agrega los datos de la obra

* GetData()

[SRFUR033] Devuelve todos los datos de la obra

* GetStatus()

[SRFUR034] Devuelve si la obra esta en proceso o ha sido finalizada

## 3.2 Requerimientos de desempeño

[SRNFR0013 El sistema deberá ser ágil y con datos del proceso de inventarios en tiempo real.

## 3.3 Requerimientos de interfaz

[SRNFR001] El servidor cliente corre en Windows\*

[SRNFR002] La interfaz del usuario final del SPI funciona en la última versión de Chrome \*

[SRNFR003] La interfaz del usuario es accesible únicamente mediante internet.

[SRNFR005] El SPI funcionará en los equipos de la empresa que cuyos navegadores cumplan con los requisitos de funcionamiento.

## 3.4 Requerimientos operacionales (prototipo)

Sin requerimientos

## 3.5 Requerimientos de recursos

[SRNFR004] El servidor cliente tiene la capacidad de almacenar a los empleados involucrados en el proceso de inventarios de la empresa.

## 3.6 Requerimientos de verificación

[SRNFR002] La interfaz del del usuario final del SPI funciona en la última versión de Chrome \*

[SRNFR005] El SPI funcionará en los equipos de la empresa que cuyos navegadores cumplan con los requisitos de funcionamiento.

## 3.7 Requerimientos de portabilidad

[SRNFR006] El sistema es accesible mediante dispositivos móviles.

[SRNFR007] La interfaz de usuario es por medio de una web responsiva.

## 3.8 Requerimientos de mantenimientos

[SRNFR008] El SPI tendrá que recibir mantenimiento y soporte.

[SRNFR009] El sistema será programado en PHP, HTML y CSS

## 3.9 Requerimientos de confiabilidad

Sin requerimientos

## 3.10 Requerimientos de seguridad

[SRNFR010] La conexión entre el servidor y el usuario debe ser segura

[SRNFR011] Los usuarios no pueden ingresar a las interfaces de otros usuarios.

[SRNFR012] El cliente administrador no puede ser borrado

## 3.11 Requerimientos de documentos

Sin requerimientos

# 4. Matriz de trazabilidad de requerimientos

## 4.1 Requerimientos de software a requerimientos de usuario

|  |  |
| --- | --- |
| Software requirement | User requirements |
| SRNFR001 | URCAR2, URCAR3, URCAR5 |
| SRNFR002 | URCAR3, URCAR4, URCAR5, URCAR7 |
| SRNFR003 | URCAR2, URCAR5, URCAR6, URCAR7 |
| SRNFR004 | URCAR0, URCAR6, URCAR8 |
| SRNFR005 | URCAR3, URCAR4, URCAR5 |
| SRNFR006 | URCAR2, URCAR4 |
| SRNFR007 | URCAR3, URCAR4, URCAR6, URCAR7 |
| SRNFR008 | URCAR14, URCAR10 |
| SRNFR009 | URCAR5, URCAR7, URCOR3, URCOR6 |
| SRNFR010 | URCOR2 |
| SRNFR011 | URCAR1, URCAR13 |
| SRNFR012 | URCAR8, URCAR14 |
| SRFUR001 | URCAR0, URCAR15 |
| SRFUR002 | URCAR15 |
| SRFUR003 | URCAR15 |
| SRFUR004 | URCAR5, URCAR13 |
| SRFUR005 | URCAR8, URCAR14, URCAR16 |
| SRFUR006 | URCAR1, URCAR9, URCAR13, URCAR16 |
| SRFUR007 | URCAR8 |
| SRFUR008 | URCAR1, URCAR9, URCAR13, URCAR17 |
| SRFUR009 | URCAR1, URCAR9, URCAR13, URCAR17 |
| SRFUR010 | URCAR1, URCAR9, URCAR13, URCAR18, URCAR19 |
| SRFUR011 | URCAR13, URCAR14, URCAR16, URCAR17, URCAR18, URCAR19 |
| SRFUR012 | URCAR9, URCAR14, URCAR16 |
| SRFUR013 | URCAR8, URCAR9, URCAR14 |
| SRFUR014 | URCAR12, URCAR25 |
| SRFUR015 | URCAR12, URCAR25 |
| SRFUR016 | URCAR12, URCAR26 |
| SRFUR017 | URCAR12, URCAR26 |
| SRFUR018 | URCAR17, URCAR27 |
| SRFUR019 | URCAR**17**, URCAR27 |
| SRFUR020 | URCAR12, URCAR27 |
| SRFUR021 | URCAR12, URCAR25 |
| SRFUR022 | URCAR12, URCAR26 |
| SRFUR023 | URCAR12, URCAR17, URCAR19, URCAR27 |
| SRFUR024 | URCAR12, URCAR17, URCAR19, URCAR27 |
| SRFUR025 | URCAR17, URCAR19, URCAR27 |
| SRFUR026 | URCAR17, URCAR18, URCAR19, URCAR27 |
| SRFUR027 | URCAR17, URCAR19, URCAR28 |
| SRFUR028 | URCAR18, URCAR19, URCAR28 |
| SRFUR029 | URCAR12, URCAR16, URCAR28 |
| SRFUR030 | URCAR18, URCAR14, URCAR28 |
| SRFUR031 | URCAR28 |
| SRFUR032 | URCAR28 |
| SRFUR033 | URCAR28 |
| SRFUR034 | URCAR28 |
| SRFUR035 | URCAR13, URCAR14, URCAR16 |
| SRFUR036 | URCAR8, URCAR9, URCAR14, URCAR15 |

Table Tabla de trazabilidad de requerimientos de software a requerimientos de usuario

# E. Apéndice 1. PROTOTIPO A MODELO LÓGICO

No aplica

# F. APÉNDICE 2: CASOS DE USO

Los casos de uso se usan para probar el modelo lógico, los casos que se muestran en este documento son:

• La vista de los reportes de un usuario final

• El usuario final da de alta productos de material

• El administrador asigna roles a usuario

### 4.2.1 Vista de reportes del usuario final

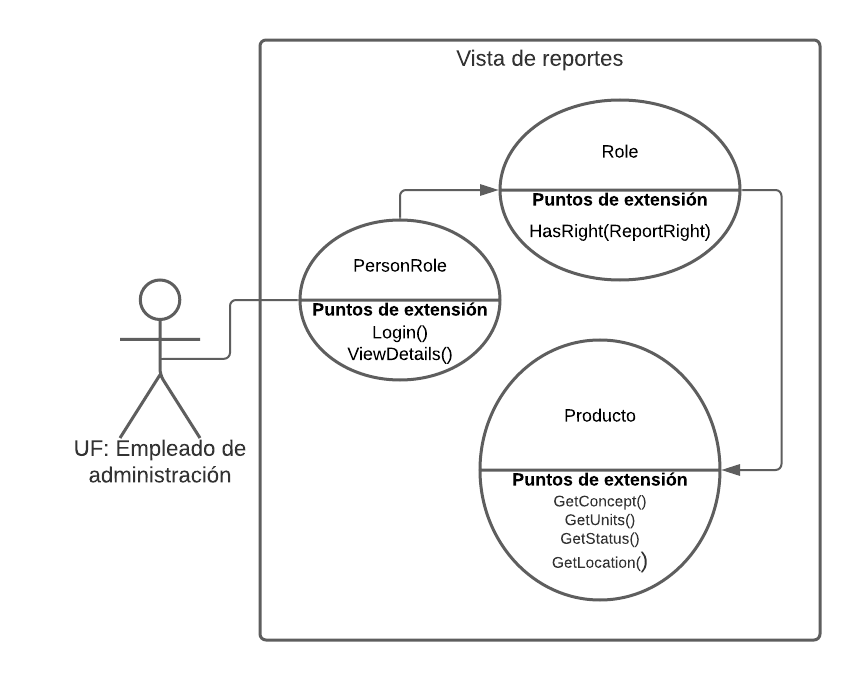


Figure Caso de uso 4.2.1

El empleado de administración que ingresa al sistema con el login() tiene asignado un rol que le permite ver los reportes de productos existentes que tiene la constructora, sus permisos pueden ser comprobados con HasRight(ReportRight), esto le genera la vista de los Productos donde puede observar el concepto del producto, las unidades, su estado y su ubicación por medio de Gets.

### 4.2.2 El usuario final da de alta productos de material

Diagrama

Descripción generada automáticamente

Figure Caso de uso 4.2.2

El empleado bodeguero tiene un usuario en el sistema cuyo rol le permite dar de alta materiales, es por esto que cuando ingresa al sistema por medio de login() tiene el acceso a poner datos en un producto (sus permisos pueden ser comprobados con HasRight(MaterialRight)) tales como su concepto, las unidades, cantidad, ubicación y la entrada de material.

### 4.2.3 El superusuario asigna roles a usuario

Diagrama

Descripción generada automáticamente

Figure Caso de uso 4.2.3

El usuario superusuario que es manejado por la directora de administración de la empresa ingresa en el sistema con login() y puesto que tiene los permisos para crear y borrar usuarios, sus permisos pueden ser comprobados con HasRight(ManageRight).El superusuario crea un nuevo usuario y le agrega los datos correspondientes así como el rol que va a tener en el sistema por medio de .