

EIMS – Elektron Inventory Management System



CONTEXTO



Industria

Requisitos de Software

Requisitos de Usuario

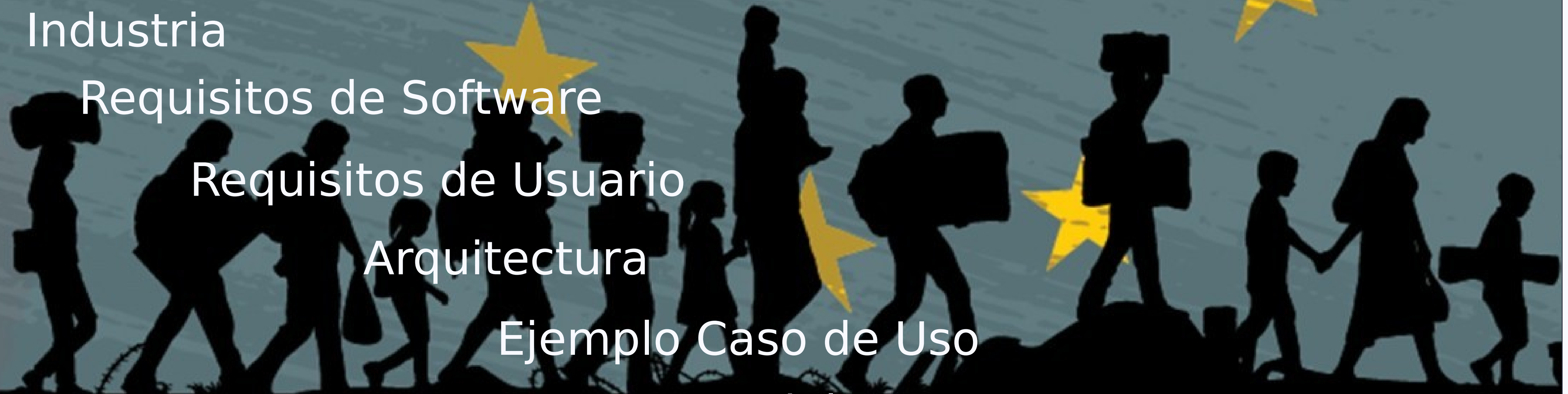
Arquitectura

Ejemplo Caso de Uso

Modelos E/R

Diagrama BPMN

Plan de Pruebas



Elektron Ingeniería Ltda.

Es una empresa que se dedica a prestar servicios de montaje eléctrico e instrumentación industrial y comercial con permanencia en la zona centro sur del país



Elektron tiene la necesidad de mejorar su proceso de inventario (herramientas, elementos de protección persona, suministros y materiales)



Especificación de Requisitos del sistema

Requisitos funcionales:

- Gestión de inventarios
- Gestión de ingreso de ítems
- Gestión de retiro de ítems
- Gestión de consultas
- Gestión de proveedores

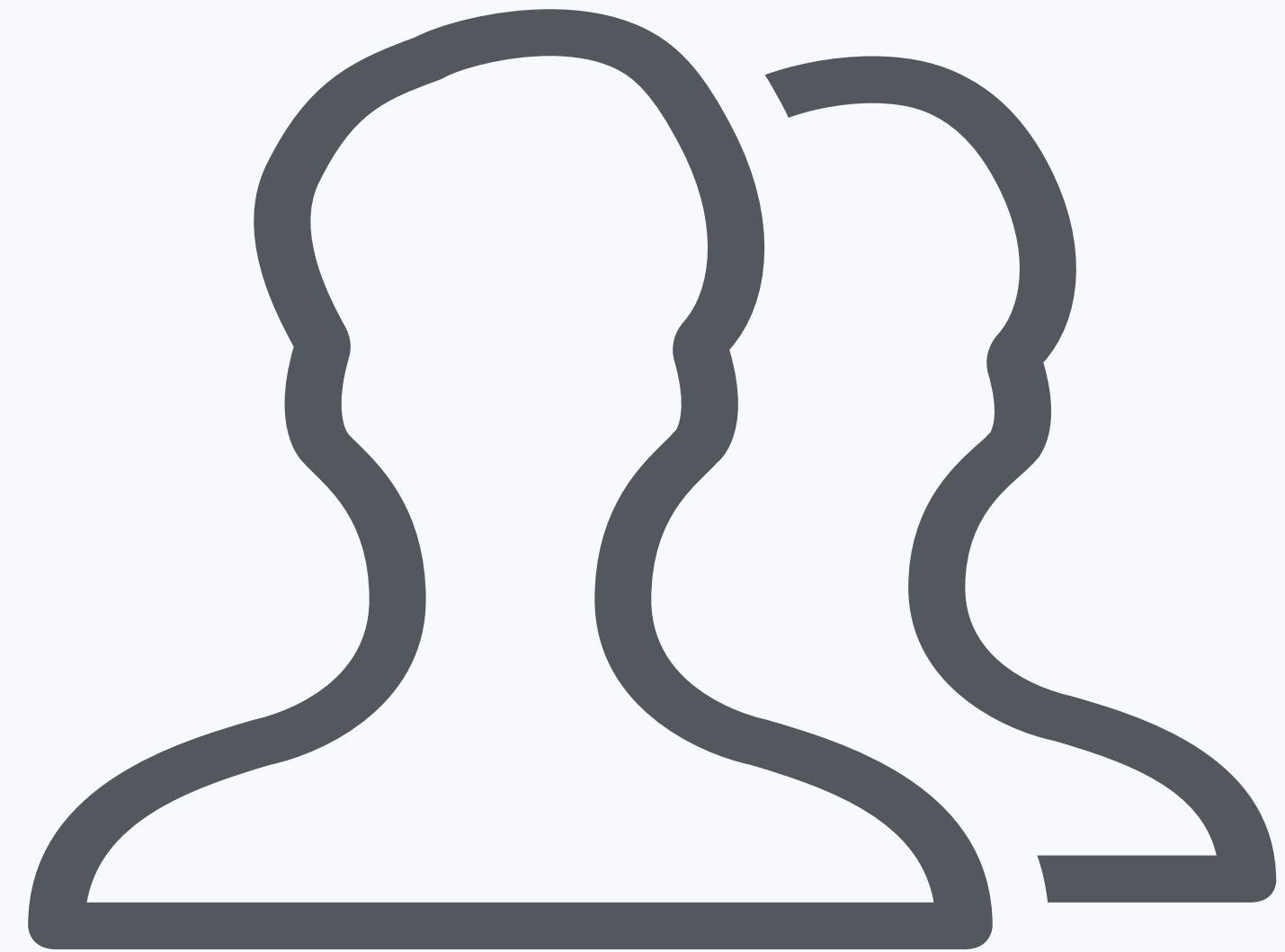
Requisitos no funcionales:

- Requisitos interfaces externo
- Requisitos de rendimiento
- Requisitos de desarrollo
- Requisitos tecnológicos
- Requisitos del producto: seguridad, protección y usabilidad



Especificación requisitos de usuario

- Gestión del almacén
- Gestión de pedidos
- Recepción de mercancías
- Salida de inventario



Arquitectura usada para el sistema

MVC para aplicación web. Las tecnologías que se emplearán en el desarrollo de este proyecto son:

- 1.HTML5
- 2.CSS y CSS3
- 3.PHP7
- 4.MySql 5.7
- 5.Framework laravel 5.5
- 6.Jquery
- 7.JavaScript



Modelo Caso de uso: Préstamos

Caso de uso 2.1 - EIMS

Caso de Uso: Creación de préstamos de elementos de inventario

Actores: Administrador

Precondición: CU1

Propósito: Permitir al usuario con perfil de administrador poder crear y hacer seguimiento a préstamos de artículos de inventarios vinculados a los trabajadores.

Resumen: Este caso de uso comienza cuando el administrador necesita otorgar un elemento del inventario a un determinado trabajador con la salvedad de que este elemento no es consumible, por lo que requiere un retorno controlado a bodega. El administrador encargado de la salida de estos elementos, crea un registro vinculando al trabajador y la herramienta a una determinada fecha de retorno.

Curso Normal de los Eventos:

Administrador	Sistema
1.- Este caso de uso comienza cuando el administrador (una vez autenticado) se dirige al apartado "items/préstamos" en el sistema EIMS.	2.- Sistema retorna la vista con el formulario de creación y seguimiento de préstamos.
3.- El administrador busca al trabajador que solicita o requiere del préstamo.	4.- Sistema despliega una lista de los trabajadores disponibles actualmente registrados y validados.
5.- Administrador busca el ítem que es requerido para el préstamo.	6.- Sistema despliega la lista de ítems actualmente registrados en la base de datos que no se encuentran en préstamo.
7.- Administrador confirma el ingreso y selección de datos anterior.	8.- Sistema notifica del éxito en la creación del préstamo y recarga la pagina para mostrar el nuevo registro en la tabla respectiva

Curso Alternativo de los Eventos (Form failed):

	4.1.- El ítem solicitado no se encuentra disponible para el préstamo en este momento (esta vinculado a otro usuario)
3.2.- Administrador cierra sesión.	

Modelo Caso de uso: Préstamos

Contratos

Contratos 1:

Responsabilidad:

getView(admin/items/loans/track)

Despliega la vista "administrar prestamos" que contiene el formulario para crear prestamos y la tabla de seguimiento de estos.

Precondición:

- This.Auth() != null //Tener sesión activa.
- Tipo de usuario es "Admin".

Contratos 2:

Responsabilidad:

select(user)

Selecciona un usuario de un combobox que despliega todos los registros actuales.

Precondición:

- This.Auth() != null //Tener sesión activa.

Postcondición:

- select id="u_id value = user.id" //select queda con la id del usuario seleccionado.

Contratos 3:

Responsabilidad:

select(item)

Selecciona un items de un combobox que despliega todos los registros actuales.

Precondición:

- This.Auth() != null //Tener sesión activa.

Postcondición:

- select id="i_id value = item.id" //select queda con la id del item seleccionado.

Contratos 4:

Responsabilidad:

select(startL,endL)

Selecciona un fecha de inicio y termino para el préstamo.

Precondición:

- This.Auth() != null //Tener sesión activa.

Postcondición:

- input id="startL" value = startL" //input queda con la fecha de inicio.
- input id="endL" value = endL" //input queda con la fecha de termino.

Contrato 5:

Responsabilidad:

postAddLoan(Request \$request)

Se encarga de almacenar los datos recolectados para el préstamo.

Precondición:

- This.Auth() != null //Tener sesión activa.
- select id="u_id value = user.id" //select queda con la id del usuario seleccionado.
- select id="i_id value = item.id" //select queda con la id del item seleccionado.
- input id="startL" value = startL" //input queda con la fecha de inicio.
- input id="endL" value = endL" //input queda con la fecha de termino

Postcondición:

```
$loan = new Loan();
$loan->item_id = $request->i_id;
$loan->user_id = $request->u_id;
$loan->startL = $request->startL_submit;
$loan->endL = $request->endL_submit;
$loan->save();
$itemL = Item::where('id', $request->i_id)->first();
$itemL->disp = 'loan';
$itemL->save();
```

- M = "Success message".

Contrato 6:

Responsabilidad:

logout();

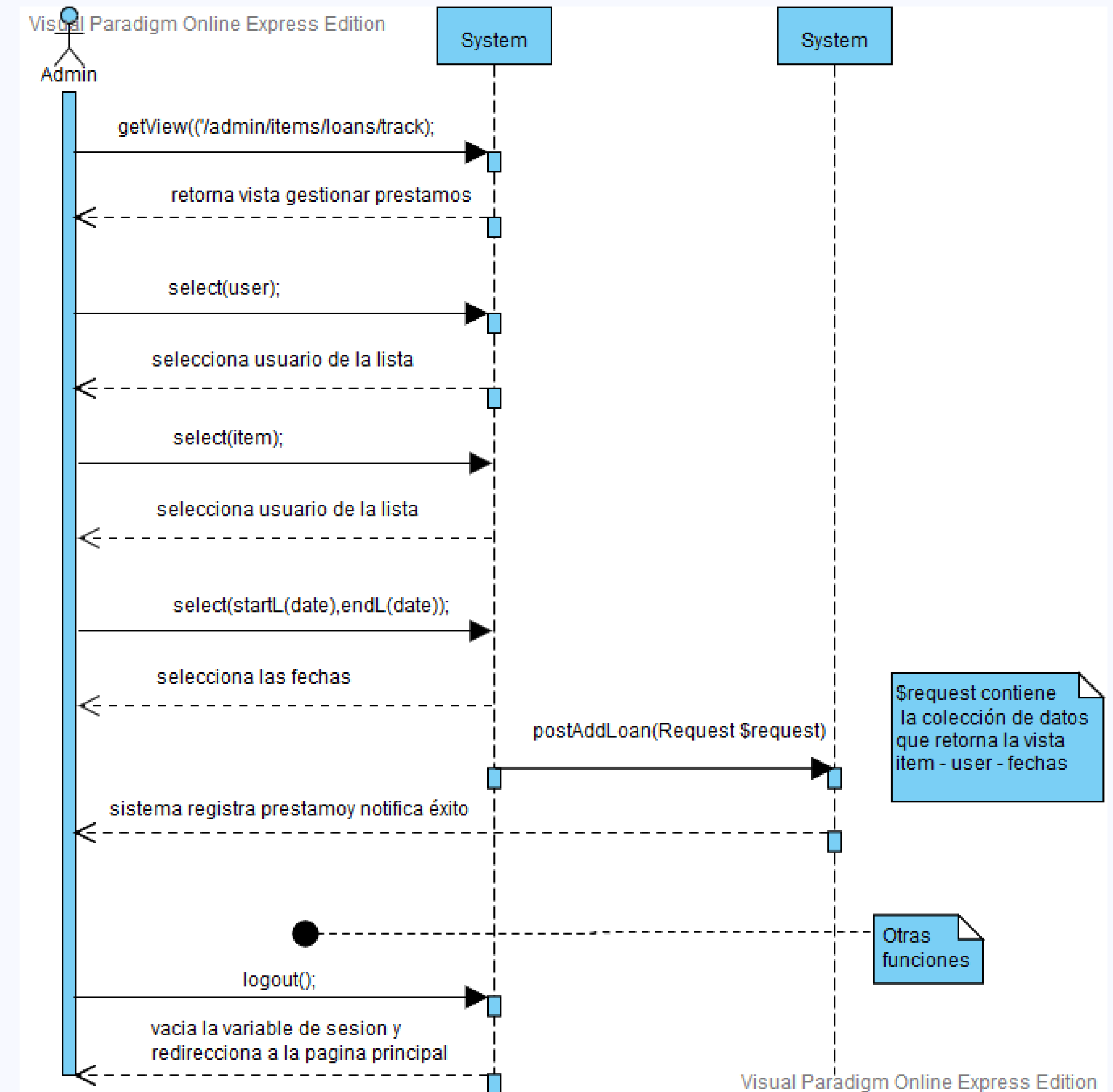
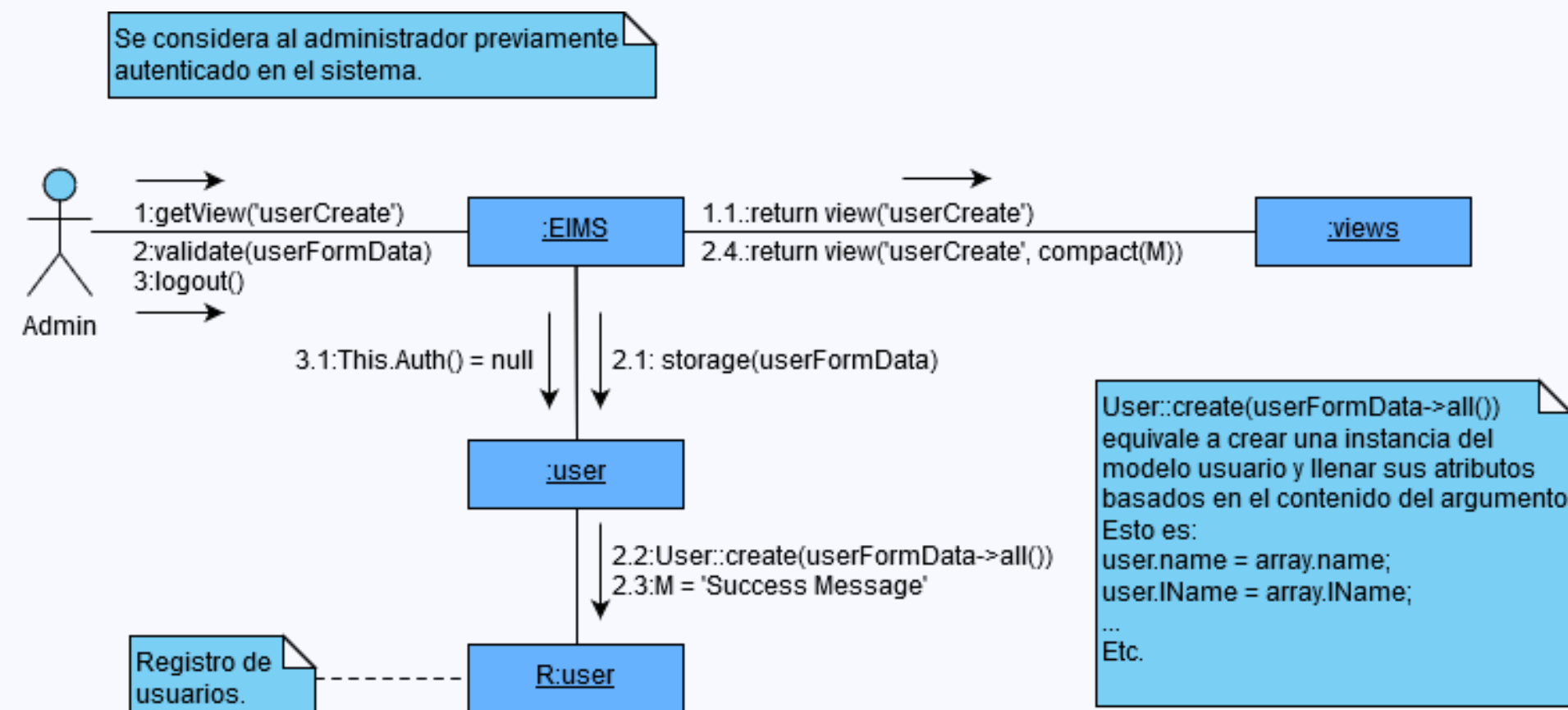
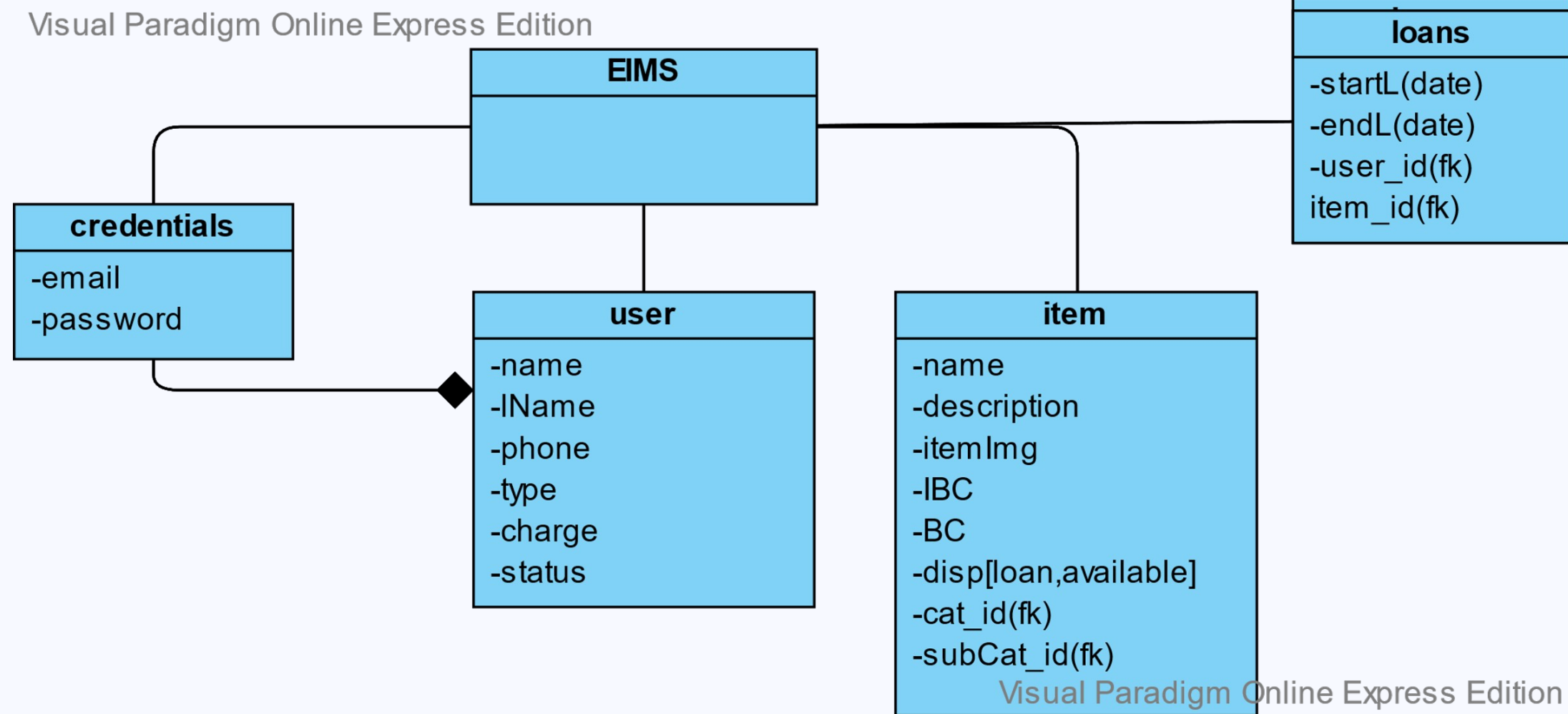
Se encarga de terminar la sesión activa del usuario y redireccionar a la página principal.

Precondición:

- This.Auth() != null //Tener sesión activa.

Postcondición: - This.Auth() = null.

Modelo Caso de uso: Préstamos



Modelo E/R

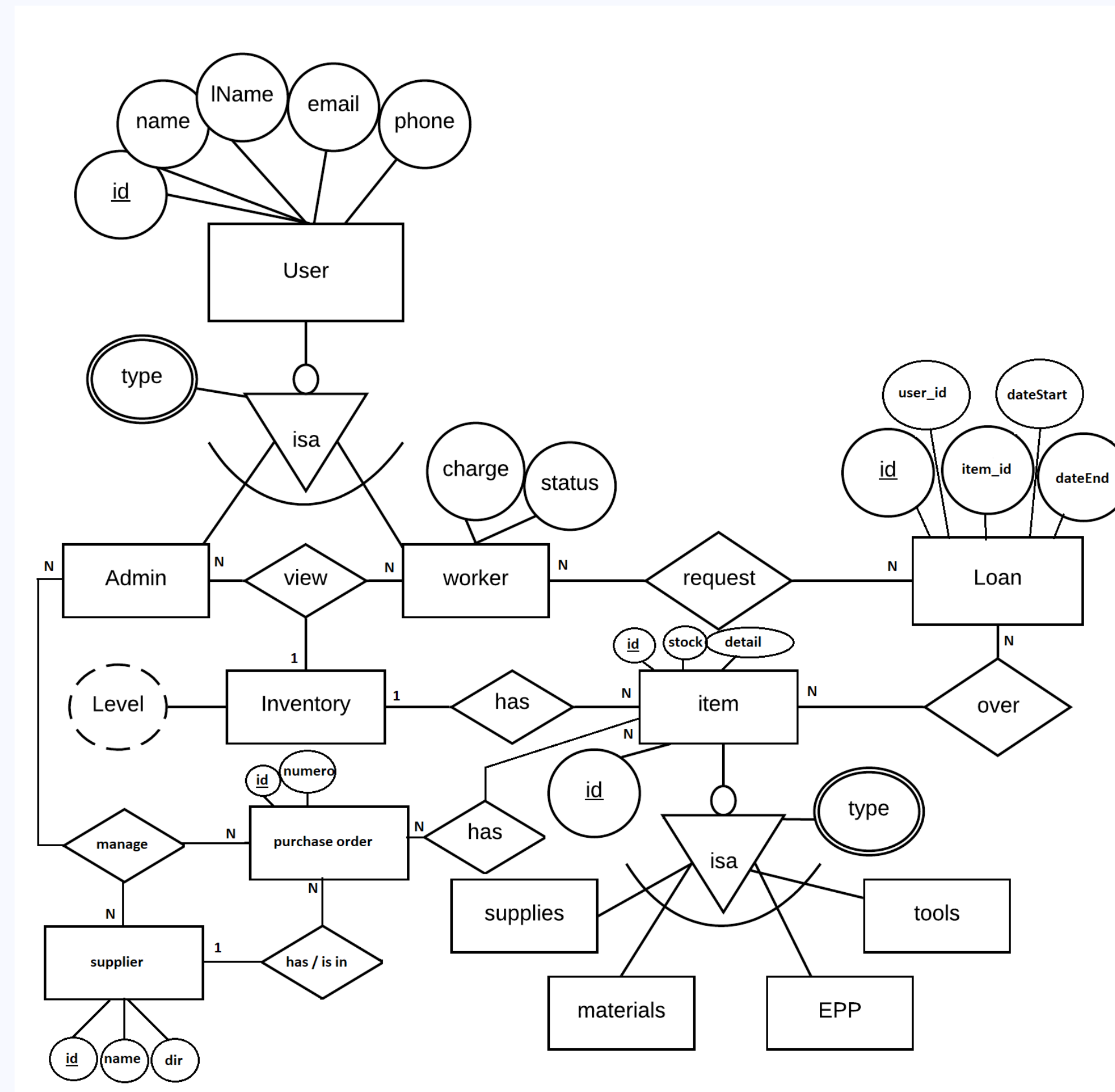
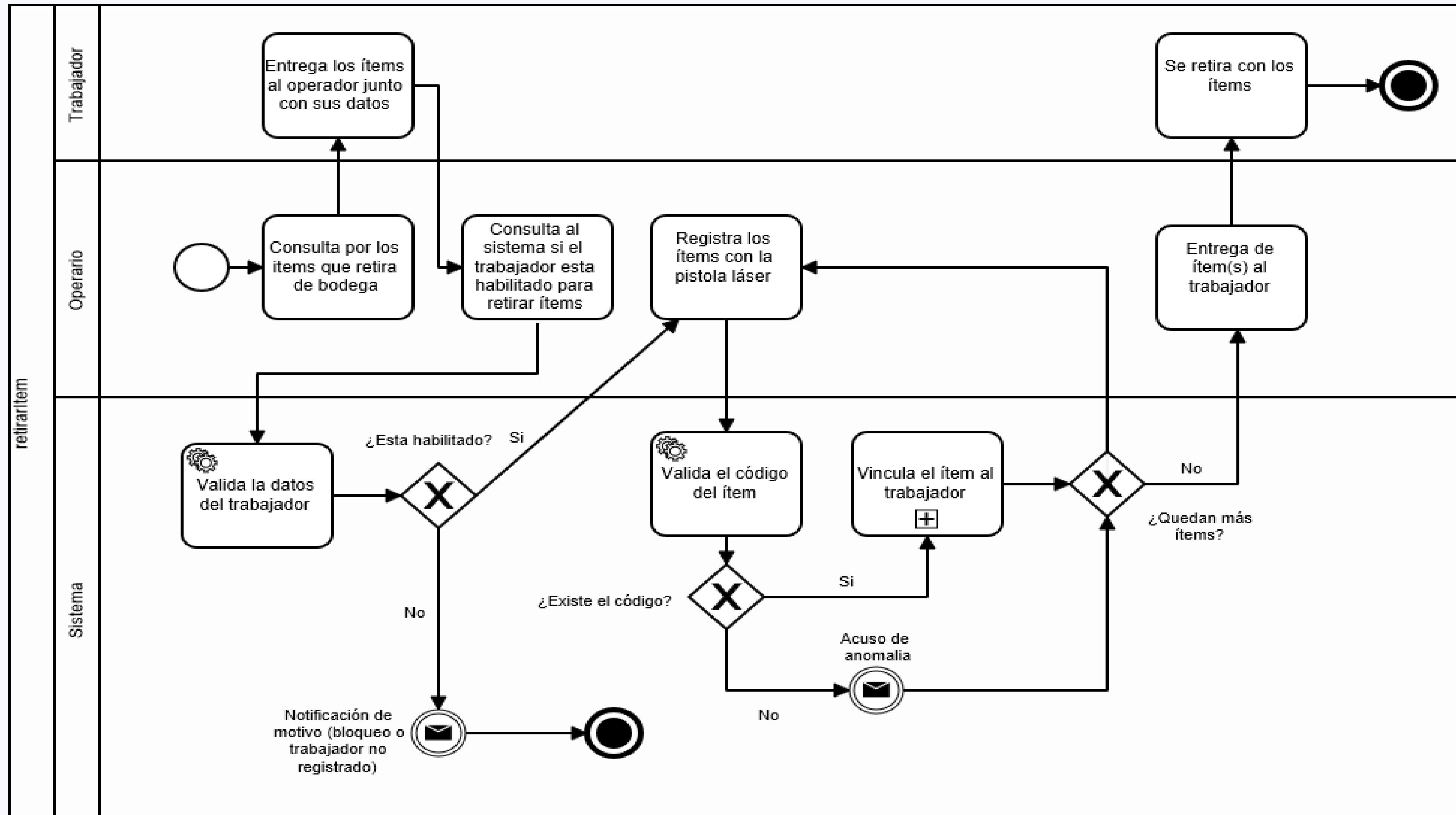


Diagrama BPMN: Préstamos



Plan de Pruebas

Basado en la estructura que necesita un plan de pruebas, se usaron 2 fases para los diseños:

- Funcionales del Sistema
- Técnico del sistema

Donde su estructura está basada éstos principales ejes:

- Pruebas Desarrollo
- Pruebas Versión
- Pruebas Usuario

Diseño Funcional del Sistema

Define las estrategias que abordan el plan:

1.2 ESCENARIOS Y ENTORNOS DE PRUEBAS		
NÚM.	CONTEXTO DE LAS PRUEBAS	PRIORIDAD
1	Detallar la configuración de equipos y otros dispositivos necesaria para hacer la prueba	media
2	Entregar resultados en las distintas etapas de pruebas a realizar, con historias de usuario en sistema, conforme al ciclo de vida.	media
NÚM.	MODOS DE ACCESO A LA APLICACIÓN Y ESCENARIOS DE USO QUE DEBEN SER PROBADOS	PRIORIDAD
1	Aplicación debe probar su portabilidad, definiendo su uso web en equipo escritorio y móvil	media
2	Aplicación debe demostrar la protección, definiendo la vista transparente de la información	Alta
3	Aplicación debe ser segura, donde perfiles definan que tipo de información acceden	Alta
NÚM.	OTROS SISTEMAS QUE DEBEN ESTAR DISPONIBLES / ACCESIBLES PARA LAS PRUEBAS	PRIORIDAD
1	Servicio de datos en almacenamiento deben estar constantemente operativos	Alta
2	Servicio de inventario para la disposición de insumos en la empresa	Alta
3	Servicio de proveedores que permita un seguimiento en los flujos de inventario	Alta
4	Servicio de solicitudes para órdenes de compra en el inventario	Alta
5	Servicio de préstamos en artículos del inventario para trabajadores de la empresa	Alta
NÚM.	OTROS PREREQUISITOS PARA PODER HACER LAS PRUEBAS	PRIORIDAD
1	La existencia de productos en el inventario debe ser coherente con los del sistema	alta
2	Selección de usuarios para pruebas de sistemas y aceptación.	Media
NÚM.	JUEGOS DE DATOS NECESARIOS PARA HACER LAS PRUEBAS	PRIORIDAD
1	Base de datos relacional para contener información de credenciales, productos, proveedores, órdenes de compra y préstamos	Alta
2	Proyección de gráficos como herramienta para los niveles de inventario	Media

Ejemplo Diseño Técnico del Sistema

1.3 CATÁLOGO DE CASOS DE PRUEBAS DE ACEPTACIÓN			
Nº	FUNCIÓN	NC	ESPECIFICACIÓN DE LAS PRUEBAS
1	Prueba desarrollo		
		1	Descripción: Acceder a la Aplicación. Requisitos probados: Usuarios actualmente trabajan en la empresa. Objetivo: Verificar el correcto acceso de la aplicación usando credenciales registradas. Pruebas a realizar: T-1. Comprobar que un usuario autorizado tiene acceso a la aplicación. T-2. Comprobar que un usuario autorizado con un password incorrecto no accede. T-3. Comprobar que un usuario inexistente no tiene acceso. T-4. Comprobar que un usuario inexistente con password incorrecto no accede. Comprobar Precondiciones: Existencia de credenciales de usuarios válidas en base de datos

2.1 CATÁLOGO DE TAREAS DE PRUEBAS FUNCIONALES				
Nº	FUNCIÓN	NC	NT	ESPECIFICACIÓN DE LAS PRUEBAS
1	Prueba desarrollo			
		1		Descripción: Acceder a la aplicación. Requisitos probados: <i>Usuarios trabajan actualmente en la empresa</i> Objetivo: Verificar el correcto acceso de la aplicación usando credenciales registradas. Precondiciones: Sistema de login y credenciales debe estar operativo Procedimiento de prueba: Se realizan secuencialmente todas las tareas de prueba validando el resultado tras cada una de ellas. Automatización de la Prueba: o PHPJunit Laravel: LoginAccess.php
			T1	Prueba a realizar: Comprobar que un usuario autorizado tiene acceso a la aplicación. Estimulo (acciones a realizar): Ingresar con credenciales correctas Respuesta esperada: Ingresa a página principal. Comando: vendor/bin/phpunit tests/Feature/LoginAccess --filter testLoginCorrectoUsuario
			T2	Prueba a realizar: Comprobar que un usuario autorizado con un password incorrecto no accede. Estimulo (acciones a realizar): Ingreso de credenciales con otro password Respuesta esperada: Permanecer en login y notificación del error Comando: vendor/bin/phpunit tests/Feature/LoginAccess --filter testLoginIncorrectoUsuario
			T3	Prueba a realizar: Comprobar que un usuario inexistente no tiene acceso. Estimulo (acciones a realizar): Ingreso con usuario no válido y password de alguna credencial válida. Respuesta esperada: Permanecer en login y notificación de error. Comando: vendor/bin/phpunit tests/Feature/LoginAccess --filter testLoginUsuarioRegistrado
			T4	Prueba a realizar: Comprobar que un usuario inexistente con password incorrecto no accede. Estimulo (acciones a realizar): Ingreso con credencial incorrecta. Respuesta esperada: Permanecer en login y notificación de error. Comando: vendor/bin/phpunit tests/Feature/LoginAccess --filter testLoginUsuarioNoRegistrado

A microphone with a face. The microphone is silver and has a black grille. It has a face drawn on it with black lines. The face has two large, round eyes, a small nose, and a wide, open mouth. The microphone is holding a small, rectangular object in its right hand. The background is a solid blue color.

Gracias por su atención.

Preguntas?