Tabla de contenido

[1. FLUJO DE TRABAJO: CAPTURA DE REQUISITOS 3](#_Toc487668723)

[1.1. DESCRIPCION DEL PROBLEMA 3](#_Toc487668724)

[1.2. REQUISITOS NO FUNCIONALES 3](#_Toc487668725)

[1.3. REQUISITOS FUNCIONALES 4](#_Toc487668726)

[1.3.1. INDENTIFICACION DE ACTORES Y CASOS DE USOS 4](#_Toc487668727)

[1.3.2. DETALLAR CASOS DE USOS 5](#_Toc487668728)

[1.3.3. MODELO GENERAL DE CASO DE USOS 28](#_Toc487668729)

[2. FLUJO DE TRABAJO: ANALISIS 29](#_Toc487668730)

[2.1. ANALISIS DE LA ARQUITECTURA 29](#_Toc487668731)

[2.1.1. IDENTIFICACION DE MODULOS 29](#_Toc487668732)

[2.1.2. DIAGRAMA DE COLABORACION 30](#_Toc487668733)

[3. FLUJO DE TRABAJO: DISEÑO 39](#_Toc487668734)

[3.1. DISEÑO DE LA ARQUITECTURA 39](#_Toc487668735)

[3.1.1. DISEÑO DE LA ARQUITECTURA FISICA 39](#_Toc487668736)

[3.1.2. DISEÑO DE LA ARQUITECTURA LOGICA 39](#_Toc487668737)

[3.2. DISEÑO DE LA INTERFAZ 40](#_Toc487668738)

[3.2.1. CU01: Gestión de personal. 40](#_Toc487668739)

[3.2.2. CU02: Gestionar equipos pesados 41](#_Toc487668740)

[3.2.3. CU03: Gestionar orden de trabajo 42](#_Toc487668741)

[3.2.4. CU04: Gestionar polígono de obra. 42](#_Toc487668742)

[3.2.5. CU05: Gestionar actividad de obra 43](#_Toc487668743)

[3.2.6. CU06: Gestionar ítem obra 43](#_Toc487668744)

[3.2.7. CU07: Gestionar usuarios de sistema 44](#_Toc487668745)

[3.2.8. CU08: Administrar gestiones 45](#_Toc487668746)

[3.2.9. CU09 Estadísticas de visitas 45](#_Toc487668747)

[3.3. DISEÑO DE LA BASE DE DATOS 45](#_Toc487668748)

[3.3.1. DISEÑO CONCEPTUAL 46](#_Toc487668749)

[3.3.2. DISEÑO LOGICO 48](#_Toc487668750)

[3.3.3. DISEÑO FISICO 49](#_Toc487668751)

[Script sql 50](#_Toc487668752)

[3.4. DISEÑO DE DETALLE PROCEDIMENTAL 63](#_Toc487668753)

[3.4.1. DIAGRAMA DE CLASE DINAMICA Y DIAGRAMA DE SECUENCIA 63](#_Toc487668754)

[4. FLUJO DE TRABAJO IMPLEMENTACION. 99](#_Toc487668755)

[4.1. Arquitectura de la Implementación: 99](#_Toc487668756)

[4.2. Diagrama de despliegue 99](#_Toc487668757)

[5. FLUJO DE TRABAJO PRUEBAS 100](#_Toc487668758)

[CU01: Gestionar personal 100](#_Toc487668759)

[CU02: Gestionar Equipo 101](#_Toc487668760)

[CU04: Gestionar Poligono 102](#_Toc487668761)

[ANEXOS 103](#_Toc487668762)

# FLUJO DE TRABAJO: CAPTURA DE REQUISITOS

## DESCRIPCION DEL PROBLEMA

La empresa dedicada al servicio de desmonte con maquinaria pesada.

Que cuenta con distintos campamentos de trabajos en el campo, tiene la necesidad de realizar el control de sus maquinarias y personal.

Cuenta con formularios de **control de avance diario de equipo y formularios para la transferencia del personal y maquinarias entre campamentos** para poder llevar el control de la producción de cada equipo y el personal.

## REQUISITOS NO FUNCIONALES

**Calidad:**

* El sistema debe poseer una interfaz sencilla para minimizar posibles errores de los usuarios Ejemplo: mensajes, entendible, etc. (**Usabilidad**).
* El software debe estar fuertemente respaldado por la documentación, para facilitar al momento de hacer cambios en el sistema (**Mantenimiento**).
* El sistema debe poseer la capacidad de permitir el desarrollo de nuevas funcionalidades (**Portabilidad**).
* El rendimiento del sistema debe ser minimizado para facilitar el manejo de las transacciones y procesamiento de datos. (**Eficiencia**).
* El software debe mostrar la información y vistas respectivas dependiendo de la sesión del usuario conectado(**seguridad**).

**Restricción:**

* Para la visualización correcta debe de usarse Firefox 53.0 o google chrome 57.0
* Debe de contar con un computador con conexión a internet.

## REQUISITOS FUNCIONALES

### INDENTIFICACION DE ACTORES Y CASOS DE USOS

* **Identificación de actores:**

****

**Administrador:** Encargado de administrar el sistema.

**Almacenero:** Es el encargado de utilizar el sistema de producción.

* **Identificación de casos de usos.**

En esta sección identificaremos los casos de uso del sistema de producción

CU01: Gestionar Personal.

CU02: Gestionar Equipo pesado.

CU03: Gestionar Orden de trabajo.

CU04: Gestionar polígonos de obra.

CU05: Gestionar actividad de obra.

CU06: Gestionar ítem obra (actividades productivas).

CU07: Gestionar usuarios de sistema.

CU08: Gestionar gestiones de periodo.

CU09: Estadísticas.

### DETALLAR CASOS DE USOS

#### CU01: Gestionar personal

##### **Diagrama de caso de uso**

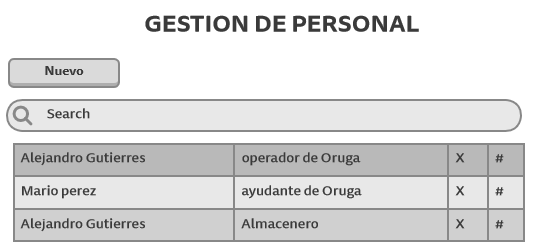


##### **Descripción de caso de uso:**

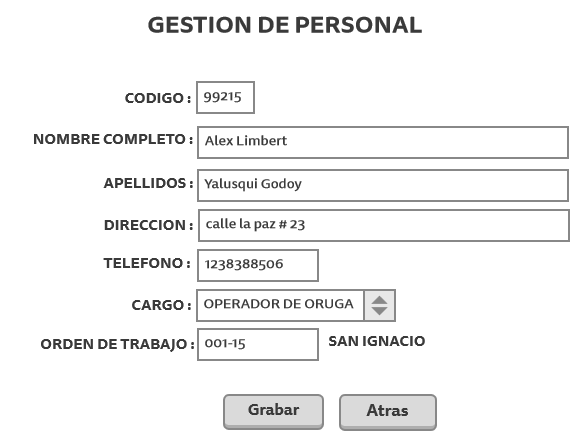
|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar personal** | | | |
| **ID :** | **CU1.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 22-abril-2017 | **Fecha de última modificación:** | 22-abril-2017 |
| **Actores Primarios:** | Administrador | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear un personal donde se introduce los datos personales del empleado como nombre completo, dirección, teléfono, carnet de identidad, cargo de empleado y orden de trabajo donde está contratado.  Además podemos modificar y eliminar un registro ya creado. | | |
| **Precondiciones:** | * El administrador debe de estar conectado al sistema. * Debe de tenerse creado y activada una gestión. * El administrador debe de tener creado por lo menos una orden de trabajo antes de gestionar un empleado. | | |
| **Postcondiciones:** | * Los cambios realizados en el personal actualizaran los datos del personal en la orden de trabajo. | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el administrador elige en el menú de la interfaz de usuario “Gestionar personal”. 2. El sistema mostrará la interfaz para gestionar personal con una lista paginada de los empleados disponibles incluyendo los campos: código, nombre completo, cargo de empleado,…, etc. con las opciones de modificar, eliminar por cada empleado. Además despliega las opciones de búsqueda avanzada, crear personal. 3. Si el administrador elige una opción del personal    1. Si el administrador elige la opción de modificar un empleado       1. El sistema visualiza la interfaz de edición de datos del personal incluyendo los campos: código, nombre completo, cargo de empleado,…, etc. Y la opción de grabar o retornar.       2. El administrador edita cualquiera de los campos listados como editables.       3. El administrador elige la opción de grabar.       4. El sistema verifica la validez de los datos a grabar.       5. El sistema graba en la base de datos los datos modificados e informa al administrador que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos los campos actualizados del personal.    2. Si el administrador elige la opción eliminar empleado       1. El sistema pide confirmación al administrador       2. Si el administrador confirma la eliminación          1. El sistema elimina el personal          2. El sistema visualiza el mensaje de que la eliminación fue exitosa.       3. Caso contrario          1. El sistema retorna a la vista detallada del personal seleccionado.    3. Si el administrador elige realizar una búsqueda avanzada 4. Si el Administrador elige Crear personal 5. El sistema visualiza la interfaz para ingresar los datos del personal como: nombre completo, cargo de empleado,…, etc. Y la opción de grabar o retornar 6. El administrador ingresa todos los datos requeridos del personal. 7. El administrador elige la opción de grabar. 8. El sistema verifica la validez de los datos a grabar. 9. El sistema graba en la base de datos los datos ingresados e informa al administrador que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos campos ingresados del personal. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 2. del flujo principal, si no existe ningún personal registrado.    1. El sistema visualizará solo las entradas de búsqueda, pero no visualizará ningún tipo de datos de personal. 2. En el punto 3.1.4, si los campos nombre completo,…, etc. son vacíos    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 3. En el punto 4.4., si los campos de nombre completo, cargo de empleado, etc. Están vacíos.   3.1. El sistema informa que campo o campos están vacíos y no  procede a grabar los datos. | | |

##### **Prototipo de interfaz**

La primera interfaz de usuario cuando el usuario selecciona el menú de gestión de personal. Las x: representa eliminar y # modificar personal.



La interfaz de nuevo, modificar personal:



#### CU02: Gestionar Equipo pesado

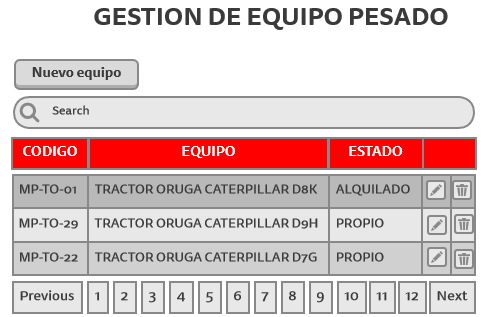
##### **Diagrama de caso de uso**



##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar Equipo Pesado** | | | |
| **ID :** | **CU02.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 22-abril-2017 | **Fecha de última modificación:** | 22-abril-2017 |
| **Actores Primarios:** | Administrador | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear un equipo pesado donde se introduce los datos del equipo introduciendo el tipo de equipo, modelo, fecha de ingreso, orden de trabajo y una descripción.  Además podemos modificar y eliminar un registro ya creado. | | |
| **Precondiciones:** | * El administrador debe de estar conectado al sistema. * Debe de tenerse creado y activada una gestión. * El administrador debe de tener creado por lo menos una orden de trabajo antes de gestionar un equipo. * El sistema carga los modelos y tipos de equipos antes de visualizar la interfaz. | | |
| **Postcondiciones:** | * Los cambios realizados en el equipo actualizaran los datos de control en el sistema de producción. | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el administrador elige en el menú de la interfaz de usuario “Gestionar Equipos pesados”. 2. El sistema mostrará la interfaz (fig. 3.3.2.1) que consta de una lista paginada de los equipos pesados disponibles, mostrando los campos: código, nombre de equipo, tipo de contrato. con las opciones de modificar, eliminar por cada equipo en la lista. Además despliega las opciones de búsqueda avanzada y nuevo equipo. 3. Si el administrador elige una opción del equipo pesado:    1. Si el administrador elige la opción de modificar un equipo       1. El sistema visualiza la interfaz (fig. 3.3.2.2) de edición de datos del equipo mostrando los campos: tipo de equipo, modelo, tipo de contrato, fecha de ingreso, orden de trabajo y descripción. Además las opciones de grabar o retornar.       2. El administrador edita cualquiera de los campos listados como editables.       3. El administrador elige la opción de grabar.       4. El sistema verifica la validez de los datos a grabar.       5. El sistema graba en la base de datos los datos modificados e informa al administrador que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos los campos actualizados del equipo pesado.    2. Si el administrador elige la opción de eliminar equipo       1. El sistema pide confirmación al administrador       2. Si el administrador confirma la eliminación          1. El sistema elimina el equipo pesado          2. El sistema visualiza el mensaje de que la eliminación fue exitosa.          3. El sistema actualiza sus datos de control interno.       3. Caso contrario          1. El sistema retorna a la vista de listado de equipo (fig. 3.3.2.1).    3. Si el administrador elige realizar una búsqueda avanzada   3.3.1. Introduce en el campo de búsqueda su criterio a buscar.  3.3.2. Se desplegará en la lista de equipos solo aquellos que cumplan el criterio de búsqueda.   1. Si el Administrador elige Nuevo equipo 2. El sistema visualiza la interfaz (fig. 3.3.2.2) para ingresar los datos del equipo como tipo de equipo, modelo, tipo de contrato, fecha de ingreso, orden de trabajo y descripción de equipo.   Además de las opción de grabar y retornar   1. El administrador ingresa todos los datos requeridos del equipo 2. El administrador elige la opción de grabar. 3. El sistema verifica la validez de los datos a grabar. 4. El sistema graba en la base de datos los datos ingresados e informa al administrador que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos campos ingresados del equipo pesado. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 2. del flujo principal, si no existe ningún personal registrado.    1. El sistema visualizará solo las entradas de búsqueda, pero no visualizará ningún tipo de datos de equipo. 2. En el punto 3.1.4, si los campos descripción, orden de trabajo, tipo de equipo y modelo son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 3. En el punto 4.4. si los campos descripción, orden de trabajo, tipo de equipo y modelo están vacíos. 4. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

##### **Prototipo de interfaz**



****

#### CU03: Gestionar Orden de trabajo

##### **Diagrama de caso de uso**



##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar Orden de trabajo** | | | |
| **ID :** | **CU03.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril-2017 |
| **Actores Primarios:** | Administrador | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear una orden de trabajo donde ingresamos los datos correspondientes a código de orden de trabajo, nombre, supervisor, área estimada y estado de orden de trabajo.  Además podemos modificar, eliminar, buscar un registro ya creado anteriormente. | | |
| **Precondiciones:** | * El administrador debe de estar conectado al sistema. * Debe de tenerse creada y activada una gestión. | | |
| **Postcondiciones:** |  | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el administrador elige en el menú de la interfaz de usuario “Gestionar orden de trabajos”. 2. El sistema mostrará la interfaz (fig. 3.3.5.1) que consta de una lista paginada de los órdenes de trabajo disponibles, mostrando los campos: código de orden, nombre y estado de orden. con las opciones de modificar, eliminar por cada orden de trabajo en la lista desplegada. Además despliega las opciones de búsqueda avanzada y nuevo equipo. 3. Si el administrador elige una de las opciones de la orden de trabajo:    1. Si el administrador elige la opción de modificar una orden       1. El sistema visualiza la interfaz (fig. 3.3.5.2) de edición de datos de la orden de trabajo mostrando los campos: código de orden de trabajo, nombre, supervisor, área estimada, estado de orden de trabajo.   Además las opciones de grabar o retornar.   * + 1. El administrador edita cualquiera de los campos listados como editables.     2. El administrador elige la opción de grabar.     3. El sistema verifica la validez de los datos a grabar.     4. El sistema graba en la base de datos los datos modificados e informa al almacenero que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos los campos actualizados de la orden de trabajo.   1. Si el almacenero elige la opción de eliminar actividad      1. El sistema pide confirmación al administrador      2. Si el administrador confirma la eliminación         1. El sistema elimina la orden de trabajo         2. El sistema visualiza el mensaje de que la eliminación fue exitosa.      3. Caso contrario         1. El sistema retorna a la vista de listado de equipo (fig. 3.3.5.1).   2. Si el administrador elige realizar una búsqueda avanzada   3.3.1. Introduce en el campo de búsqueda su criterio a buscar.  3.3.2. Se desplegará en la lista de órdenes de trabajo solo aquellos que cumplan el criterio de búsqueda.   1. Si el administrador elige Nueva obra 2. El sistema visualiza la interfaz (fig. 3.3.5.2) para ingresar los datos de la orden de trabajo como código, nombre, supervisor, área estimada, estado de orden de trabajo.   Además de las opción de grabar y retornar   1. El administrador ingresa todos los datos requeridos de la orden de trabajo. 2. Si el administrador elige la opción de grabar. 3. El sistema verifica la validez de los datos a grabar. 4. El sistema graba en la base de datos los datos ingresados e informa al administrador que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos los campos ingresados de la orden de trabajo. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 2. del flujo principal, si no existe ninguna orden de trabajo registrada.    1. El sistema visualizará solo las entradas de búsqueda, pero no visualizará ninguna orden de trabajo. 2. En el punto 3.1.4, si los campos código, nombre, estado de orden. Son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 3. En el punto 4.4. si los campos, código, nombre y estado de orden de trabajo. Son vacíos, o no cumplen alguna política de negocio. 4. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

##### **Prototipo de interfaz**



****

#### CU04: Gestionar polígonos de obra.

##### **Diagrama de caso de uso**



##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar polígonos de obra** | | | |
| **ID :** | **CU04.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril- 2017 |
| **Actores Primarios:** | Administrador | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear un polígono donde ingresamos los datos correspondientes a orden de trabajo, código y descripción del polígono.  Además podemos modificar, eliminar, ya creado anteriormente. | | |
| **Precondiciones:** | * El usuario debe de estar conectado al sistema. * Debe de tenerse creada y activada una gestión. * El administrador debe de tener creado por lo menos una orden de trabajo antes de crear un ítem obra. | | |
| **Postcondiciones:** |  | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el usuario elige en el menú de la interfaz de usuario “Gestionar polígonos”. 2. El sistema mostrará la interfaz que consta de una lista paginada de polígonos disponibles, mostrando los campos: orden de trabajo, código y descripción del polígono. con las opciones de modificar, eliminar por cada polígono en la lista desplegada. 3. Si el administrador elige una de las opciones del poligono:    1. Si el administrador elige la opción de modificar un poligono       1. El sistema visualiza la interfaz de edición de datos de poligono mostrando los campos: orden de trabajo, código y descripción del poligono.   Además las opciones de guardar y retornar.   * + 1. El usuario edita cualquiera de los campos listados como editables.     2. El usuario elige la opción de grabar.     3. El sistema verifica la validez de los datos a grabar.     4. El sistema graba en la base de datos los datos modificados.   1. Si el almacenero elige la opción de eliminar poligono      1. El sistema pide confirmación al usuario      2. Si el almacenero confirma la eliminación         1. El sistema elimina el poligono         2. El sistema muestra el nuevo listado de poligonos.      3. Caso contrario         1. El sistema retorna a la vista de listado de poligonos  1. Si el almacenero elige Nuevo poligono 2. El sistema visualiza la interfaz para ingresar los datos del poligono como orden de trabajo, código y descripción.   Además de las opción de grabar y retornar   1. El almacenero ingresa todos los datos requeridos del poligono 2. Si el almacenero elige la opción de grabar. 3. El sistema verifica la validez de los datos a grabar. 4. El sistema graba en la base de datos los datos ingresados. Y visualizando la lista de poligonos. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 3.1.4, si los campos orden de trabajo, codigo, descripción. Son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 2. En el punto 4.4. si los campos orden de trabajo, código o descripción. Son vacíos, o no cumplen alguna política de negocio. 3. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

#### CU05: Gestionar actividades de obra.

##### **Diagrama de caso de uso**



##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar actividad de obra** | | | |
| **ID :** | **CU05.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril- 2017 |
| **Actores Primarios:** | Usuario | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear una actividad donde ingresamos los datos correspondientes a código y descripción de la actividad.  Además podemos modificar, eliminar, ya creado anteriormente. | | |
| **Precondiciones:** | * El usuario debe de estar conectado al sistema. * El usuario debe de estar autenticado correctamente. | | |
| **Postcondiciones:** |  | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el usuario elige en el menú de la interfaz de usuario “Gestionar Actividades”. 2. El sistema mostrará la interfaz que consta de una lista paginada de actividades disponibles, mostrando los campos: codigo y descripción del actividad. con las opciones de modificar, eliminar por cada actividad en la lista desplegada. 3. Si el administrador elige una de las opciones de la actividad:    1. Si el administrador elige la opción de modificar una actividad       1. El sistema visualiza la interfaz de edición de datos de actividad mostrando los campos: código y descripción de la actividad.   Además las opciones de guardar y retornar.   * + 1. El usuario edita cualquiera de los campos listados como editables.     2. El usuario elige la opción de guardar.     3. El sistema verifica la validez de los datos a grabar.     4. El sistema graba en la base de datos los datos modificados.   1. Si el administrador elige la opción de eliminar actividad      1. El sistema pide confirmación al usuario      2. Si el administrador confirma la eliminación         1. El sistema elimina la actividad.         2. El sistema muestra el nuevo listado de actividades.      3. Caso contrario         1. El sistema retorna a la vista de listado de actividades.  1. Si el almacenero elige Nueva actividad. 2. El sistema visualiza la interfaz para ingresar los datos de la actividad como código y descripción.   Además de las opción de grabar y retornar   1. El administrador ingresa todos los datos requeridos de la actividad. 2. Si el administrador elige la opción de guardar. 3. El sistema verifica la validez de los datos a guardar. 4. El sistema graba en la base de datos los datos ingresados. Y visualizando la lista de actividades. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 3.1.4, si los campos codigo, descripción. Son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 2. En el punto 4.4. si los campos código o descripción. Son vacíos, o no cumplen alguna política de negocio. 3. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

#### CU06: Gestionar ítem obra.

##### **Diagrama de caso de uso**

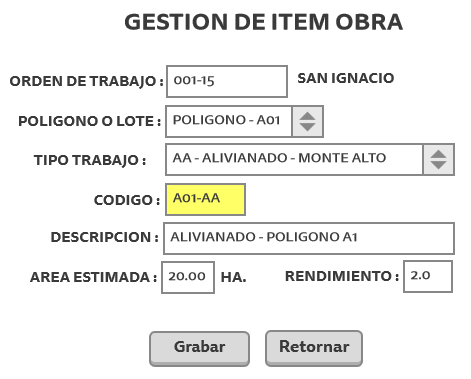


##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar Ítem obra** | | | |
| **ID :** | **CU06.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril- 2017 |
| **Actores Primarios:** | Almacenero | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear una actividad productiva donde ingresamos los datos correspondiente a orden de trabajo, polígono o lote, tipo de trabajo, descripción del ítem obra, área estimada del lote y el rendimiento.  Además podemos modificar, eliminar, buscar un registro ya creado anteriormente. | | |
| **Precondiciones:** | * El almacenero debe de estar conectado al sistema. * Debe de tenerse creada y activada una gestión. * El almacenero debe de tener creado por lo menos una orden de trabajo antes de crear un ítem obra. * El sistema carga los polígonos y tipo de trabajo antes de visualizar la interfaz. | | |
| **Postcondiciones:** | * Los cambios realizados en el equipo actualizaran los datos de control en el sistema de producción. | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el almacenero elige en el menú de la interfaz de usuario “Gestionar ítems obras”. 2. El sistema mostrará la interfaz (fig. 3.3.3.1) que consta de una lista paginada de los ítems obras disponibles, mostrando los campos: orden de trabajo, código y descripción de ítem obra. con las opciones de modificar, eliminar por cada ítem obra en la lista desplegada. Además despliega las opciones de búsqueda avanzada y nuevo equipo. 3. Si el almacenero elige una de las opciones del ítem obra:    1. Si el almacenero elige la opción de modificar un ítem obra       1. El sistema visualiza la interfaz (fig. 3.3.3.2) de edición de datos del ítem obra mostrando los campos: orden de trabajo, polígono, tipo de trabajo, descripción, área estimada y rendimiento.   Además las opciones de grabar o retornar.   * + 1. El almacenero edita cualquiera de los campos listados como editables.     2. El almacenero elige la opción de grabar.     3. El sistema verifica la validez de los datos a grabar.     4. El sistema graba en la base de datos los datos modificados e informa al almacenero que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos los campos actualizados del ítem obra.   1. Si el almacenero elige la opción de eliminar ítem obra      1. El sistema pide confirmación al almacenero      2. Si el almacenero confirma la eliminación         1. El sistema elimina el ítem obra         2. El sistema visualiza el mensaje de que la eliminación fue exitosa.         3. El sistema actualiza sus datos de control interno.      3. Caso contrario         1. El sistema retorna a la vista de listado de equipo (fig. 3.3.3.1).   2. Si el almacenero elige realizar una búsqueda avanzada   3.3.1. Introduce en el campo de búsqueda su criterio a buscar.  3.3.2. Se desplegará en la lista de ítems obras solo aquellos que cumplan el criterio de búsqueda.   1. Si el almacenero elige Nuevo ítem obra 2. El sistema visualiza la interfaz (fig. 3.3.3.2) para ingresar los datos del ítem obra como orden de trabajo, polígono, tipo de trabajo, descripción, área estimada.   Además de las opción de grabar y retornar   1. El almacenero ingresa todos los datos requeridos del ítem obra 2. Si el almacenero elige la opción de grabar. 3. El sistema verifica la validez de los datos a grabar. 4. El sistema graba en la base de datos los datos ingresados e informa al almacenero que la operación se realizó exitosamente mediante un mensaje. Y visualizando en detalle todos campos ingresados del ítem obra. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 2. del flujo principal, si no existe ningún ítem obra registrado.    1. El sistema visualizará solo las entradas de búsqueda, pero no visualizará ningún ítem obra. 2. En el punto 3.1.4, si los campos orden de trabajo, polígono, tipo de trabajo, descripción, etc. Son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 3. En el punto 4.4. si los campos orden de trabajo, polígono, tipo de trabajo, descripción, etc. Son vacíos, o no cumplen alguna política de negocio. 4. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

##### **Prototipo de interfaz**



****

#### CU07: Gestionar usuarios de sistemas.

##### **Diagrama de caso de uso**



##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Gestionar usuarios de sistema** | | | |
| **ID :** | **CU07.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril- 2017 |
| **Actores Primarios:** | Usuario | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear un usuario para el sistema donde ingresamos los datos correspondientes a nombre, apellidos, email, grupo, etc. del usuario.  Además podemos modificar, eliminar, ya creado anteriormente. | | |
| **Precondiciones:** | * El usuario debe de estar conectado al sistema. * El usuario debe de estar autenticado correctamente. | | |
| **Postcondiciones:** |  | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el usuario elige en el menú de la interfaz de usuario “Gestionar Usuarios de sistema”. 2. El sistema mostrará la interfaz que consta de una lista paginada de actividades disponibles, mostrando los campos: nombre de usuario y nombre completo, email y grupo. con las opciones de modificar, eliminar por cada usuario en la lista desplegada. 3. Si el administrador elige una de las opciones del usuario:    1. Si el administrador elige la opción de modificar un usuario       1. El sistema visualiza la interfaz de edición de datos de usuario mostrando los campos: nombre, apellidos, grupo.   Además las opciones de guardar y retornar.   * + 1. El usuario edita cualquiera de los campos listados como editables.     2. El usuario elige la opción de guardar.     3. El sistema verifica la validez de los datos a grabar.     4. El sistema graba en la base de datos los datos modificados.   1. Si el administrador elige la opción de eliminar usuario      1. El sistema pide confirmación al usuario      2. Si el administrador confirma la eliminación         1. El sistema elimina el usuario seleccionado.         2. El sistema muestra el nuevo listado de usuarios.      3. Caso contrario         1. El sistema retorna a la vista de listado de usuarios.  1. Si el almacenero elige Nuevo usuario. 2. El sistema visualiza la interfaz para ingresar los datos de usuario para ingresar el nombre, apellidos, email, etc.   Además de las opción de grabar y retornar   1. El administrador ingresa todos los datos requeridos del usuario. 2. Si el administrador elige la opción de guardar. 3. El sistema verifica la validez de los datos a guardar. 4. El sistema graba en la base de datos los datos ingresados. Y visualizando la lista de actividades. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 3.1.4, si los campos nombre, apellidos. Son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 2. En el punto 4.4. si los campos nombre, apellidos, email. Son vacíos, o no cumplen alguna política de negocio. 3. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

#### CU08: Gestionar gestiones de periodo.

##### **Diagrama de caso de uso**



##### **Descripción de caso de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Administrador de gestiones de periodo** | | | |
| **ID :** | **CU08.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril- 2017 |
| **Actores Primarios:** | Usuario | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite crear una gestión para el sistema donde ingresamos los datos correspondientes a codigo, fecha inicial y final.  Además podemos modificar, eliminar, ya creado anteriormente. | | |
| **Precondiciones:** | * El usuario debe de estar conectado al sistema. * El usuario debe de estar autenticado correctamente. | | |
| **Postcondiciones:** |  | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el usuario elige en el menú de la interfaz de usuario “Administrar gestiones”. 2. El sistema mostrará la interfaz que consta de una lista paginada de gestiones disponibles, mostrando los campos: código, fecha inicial y final. con las opciones de modificar, eliminar por cada usuario en la lista desplegada. 3. Si el administrador elige una de las opciones del gestión:    1. Si el administrador elige la opción de modificar una gestión       1. El sistema visualiza la interfaz de edición de datos de usuario mostrando los campos: código, fecha inicial y final.   Además las opciones de guardar y retornar.   * + 1. El usuario edita cualquiera de los campos listados como editables.     2. El usuario elige la opción de guardar.     3. El sistema verifica la validez de los datos a grabar.     4. El sistema graba en la base de datos los datos modificados.   1. Si el administrador elige la opción de eliminar gestion      1. El sistema pide confirmación al usuario      2. Si el administrador confirma la eliminación         1. El sistema elimina la gestión seleccionada.         2. El sistema muestra el nuevo listado de gestiones.      3. Caso contrario         1. El sistema retorna a la vista de listado de gestiones.  1. Si el almacenero elige Nueva gestión. 2. El sistema visualiza la interfaz para ingresar los datos de gestión para ingresar código y fecha inicial y final, etc.   Además de las opción de grabar y retornar   1. El administrador ingresa todos los datos requeridos del usuario. 2. Si el administrador elige la opción de guardar. 3. El sistema verifica la validez de los datos a guardar. 4. El sistema graba en la base de datos los datos ingresados. Y visualizando la lista de gestiones. | | |
| **Flujo(s) alternativo(s):** | 1. En el punto 3.1.4, si el código. Son vacíos.    1. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. 2. En el punto 4.4. si los campos codigo. Son vacíos, o no cumplen alguna política de negocio. 3. El sistema informa que campo o campos están vacíos y no procede a grabar los cambios. | | |

#### CU09: Estadística.

##### **Diagrama de caso de uso.**



##### **Descripción de caso de uso.**

|  |  |  |  |
| --- | --- | --- | --- |
| **CASO DE USO: Estadística** | | | |
| **ID :** | **CU09.** | | |
| **Creado por:** | Alex Limbert Yalusqui Godoy | **Última modificación por:** | Alex Limbert Yalusqui Godoy |
| **Fecha de creación:** | 21-abril-2017 | **Fecha de última modificación:** | 21-abril- 2017 |
| **Actores Primarios:** | Usuario | | |
| **Actores secundarios:** |  | | |
| **Descripción:** | Permite visualizar los números de accesos a las páginas del sistema. | | |
| **Precondiciones:** | * El usuario debe de estar conectado al sistema. * El usuario debe de estar autenticado correctamente. | | |
| **Postcondiciones:** |  | | |
| **Flujo Principal:** | 1. El caso de uso inicia cuando el usuario elige en el menú de la interfaz de usuario “Estadísticas de visitas”. 2. El sistema mostrará una generara una imagen con las estadísticas de acceso al sistema. | | |
| **Flujo(s) alternativo(s):** |  | | |

### MODELO GENERAL DE CASO DE USOS



# FLUJO DE TRABAJO: ANALISIS

## ANALISIS DE LA ARQUITECTURA

### IDENTIFICACION DE MODULOS

#### MODULO DE ADMINISTRACION



#### MODULO DE ALMACEN



### DIAGRAMA DE COLABORACION

#### CU01: Gestionar personal



#### CU02: Gestionar equipo pesado



#### CU03: Gestionar orden de trabajo



#### CU04: Gestionar polígono de obra



#### CU05: Gestionar actividad de obra



#### CU06: Gestionar ítem obra (actividad productiva)



#### CU07: Gestionar usuarios de sistema



#### CU08: Administración de gestiones



#### CU09: Estadísticas de visitas



# FLUJO DE TRABAJO: DISEÑO

## DISEÑO DE LA ARQUITECTURA

### DISEÑO DE LA ARQUITECTURA FISICA



### DISEÑO DE LA ARQUITECTURA LOGICA

#### 3.1.2.1. Módulo de administración



#### 3.1.2.2. Módulo de almacén



## DISEÑO DE LA INTERFAZ

### CU01: Gestión de personal.

#### Lista de personal

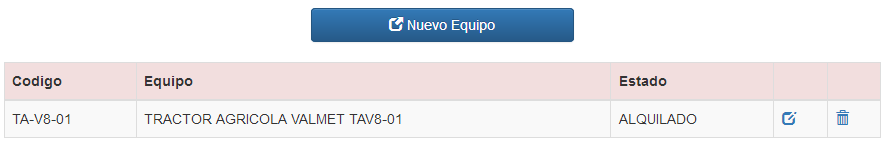


#### Vista de personal



### 3.2.2. CU02: Gestionar equipos pesados

#### Lista de equipos pesados



#### Vista de equipo pesados



### 3.2.3. CU03: Gestionar orden de trabajo

#### Lista de órdenes de trabajo

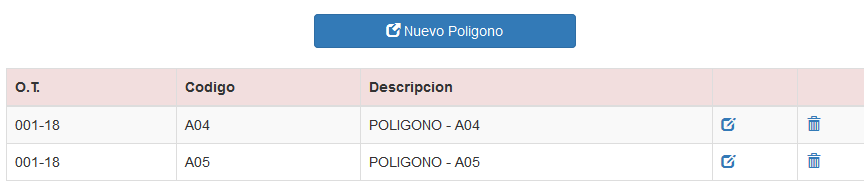


#### Vista de órdenes de trabajo



### 3.2.4. CU04: Gestionar polígono de obra.

#### Lista de poligonos

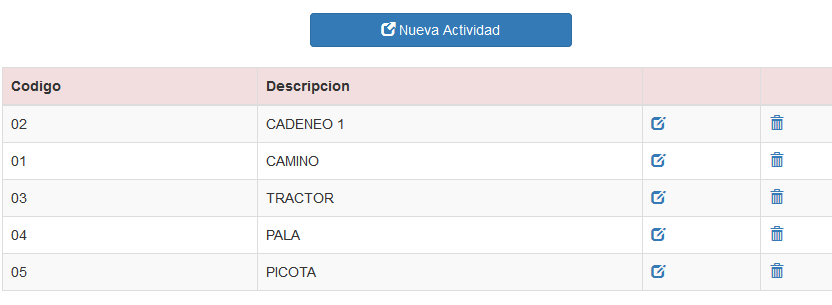


#### Vista de poligonos de obra

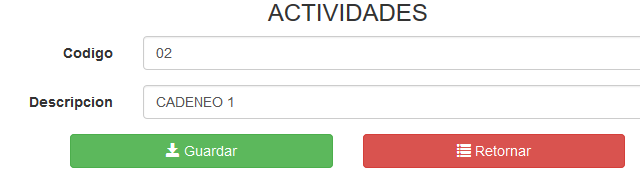


### 3.2.5. CU05: Gestionar actividad de obra

#### Lista de actividad de obras

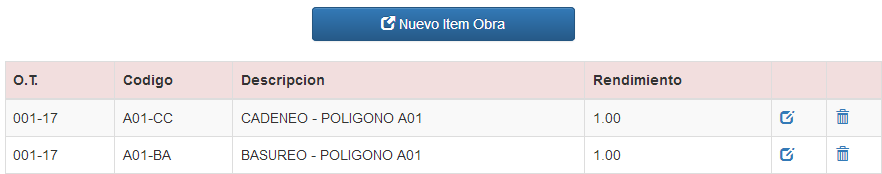


#### Vista de actividades de obra



### 3.2.6. CU06: Gestionar ítem obra

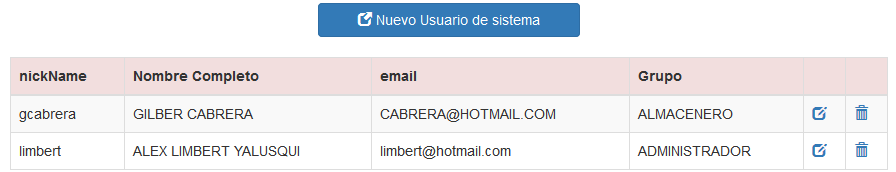
#### Lista de ítem’s obras



#### Vista de item’s obras

### 3.2.7. CU07: Gestionar usuarios de sistema

#### Lista de usuarios de sistema

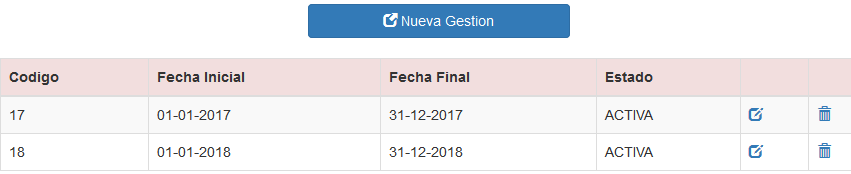


#### Vista de usuarios de sistema



### 3.2.8. CU08: Administrar gestiones

#### Lista de gestiones

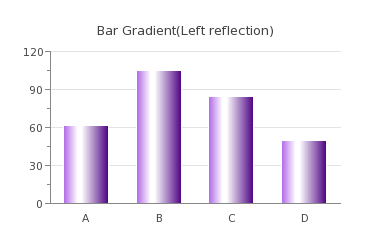


#### Vista de gestión



### 3.2.9. CU09 Estadísticas de visitas

#### Vista de estadística



## DISEÑO DE LA BASE DE DATOS

### DISEÑO CONCEPTUAL



### DISEÑO LOGICO

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| sppersonal | | | | | | | | | | |
| pkpersonal | fechaIngreso | nombreComp | apellidos | direccion | telefono | ci | fechaNac | email | fkcargo | fkordentrabajo |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| spequipo | | | | | | | |
| pkEquipo | fkcodigo | fktipocontrato | fechaingreso | descripcion | fkordentrabajo |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| spordentrabajo | | | | | | | |
| pkordentrabajo | codigo | nombre | supervisor | Areaestimada | estado | data | Fkgestion |

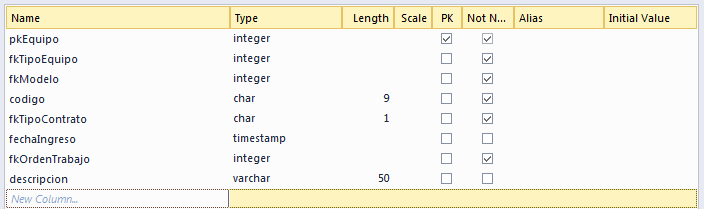
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| spitemobra | | | | | | | |
| pkitemobra | codigo | Descripcion | areatrab | rendimiento | fkordentrabajo | fkpoligono | fkactividad |

|  |  |  |  |
| --- | --- | --- | --- |
| Sppoligono | | | |
| pkpoligono | codigo | Descripcion | fkordentrabajo |

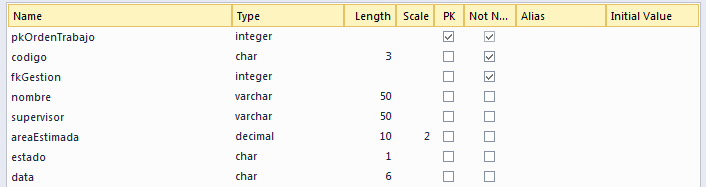
|  |  |  |
| --- | --- | --- |
| Spactividad | | |
| Pkactividad | codigo | Descripcion | |

### DISEÑO FISICO

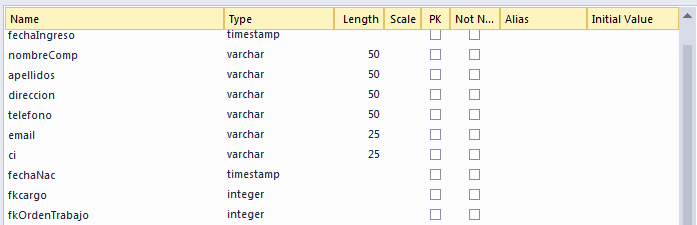
**spequipo**



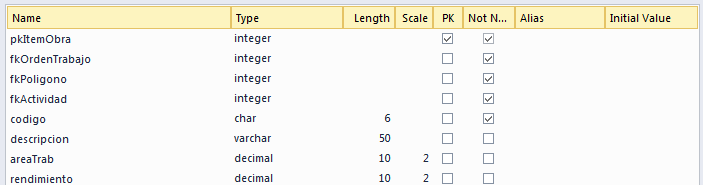
**spOrdenTrabajo**

****

**spPersonal**

****

**spItemObra**

****

### Script sql

DROP SEQUENCE IF EXISTS spactividad\_pkactividad\_seq;

DROP SEQUENCE IF EXISTS spcargo\_pkcargo\_seq;

DROP SEQUENCE IF EXISTS speqmodelo\_pkeqmodelo\_seq;

DROP SEQUENCE IF EXISTS speqtipo\_pkeqtipo\_seq;

DROP SEQUENCE IF EXISTS spequipo\_pkequipo\_seq;

DROP SEQUENCE IF EXISTS spgestion\_pkgestion\_seq;

DROP SEQUENCE IF EXISTS spimproductiva\_pkimproductiva\_seq;

DROP SEQUENCE IF EXISTS spitemobra\_pkitemobra\_seq;

DROP SEQUENCE IF EXISTS spordentrabajo\_pkordentrabajo\_seq;

DROP SEQUENCE IF EXISTS sppersonal\_pkpersonal\_seq;

DROP SEQUENCE IF EXISTS sppoligono\_pkpoligono\_seq;

DROP SEQUENCE IF EXISTS spturno\_pkturno\_seq;

/\* Drop Tables \*/

DROP TABLE IF EXISTS spActividad CASCADE;

DROP TABLE IF EXISTS spCargo CASCADE;

DROP TABLE IF EXISTS spEqModelo CASCADE;

DROP TABLE IF EXISTS spEqTipo CASCADE;

DROP TABLE IF EXISTS spEquipo CASCADE;

DROP TABLE IF EXISTS spGestion CASCADE;

DROP TABLE IF EXISTS spImproductiva CASCADE;

DROP TABLE IF EXISTS spItemObra CASCADE;

DROP TABLE IF EXISTS spOrdenTrabajo CASCADE;

DROP TABLE IF EXISTS spParteDiario CASCADE;

DROP TABLE IF EXISTS spPersonal CASCADE;

DROP TABLE IF EXISTS spPoligono CASCADE;

DROP TABLE IF EXISTS spTransfEquipo CASCADE;

DROP TABLE IF EXISTS spTransfPersonal CASCADE;

DROP TABLE IF EXISTS spTurno CASCADE;

/\* Create Tables \*/

CREATE TABLE spActividad

(

pkActividad integer NOT NULL DEFAULT NEXTVAL(('"spactividad\_pkactividad\_seq"'::text)::regclass),

fkOrdenTrabajo integer NOT NULL,

codigo char(2) NOT NULL,

descripcion varchar(50)

);

CREATE TABLE spCargo

(

pkCargo integer NOT NULL DEFAULT NEXTVAL(('"spcargo\_pkcargo\_seq"'::text)::regclass),

codigo char(3) NOT NULL,

descripcion varchar(50) NOT NULL

);

CREATE TABLE spDetImproductiva

(

pkDetImproductiva integer NOT NULL DEFAULT NEXTVAL(('"spdetimproductiva\_pkdetimproductiva\_seq"'::text)::regclass),

fkParteDiario integer NOT NULL,

nroItem integer,

fkImproductiva integer NOT NULL,

hrs decimal(10,2)

);

CREATE TABLE spDetProductiva

(

pkDetParteDiario integer NOT NULL DEFAULT NEXTVAL(('"spdetproductiva\_pkdetpartediario\_seq"'::text)::regclass),

fkParteDiario integer NOT NULL,

nroItem smallint NOT NULL,

fkItemObra integer NOT NULL,

grupo smallint NOT NULL,

hmtoIni decimal(10,2),

hmtoFin decimal(10,2),

area decimal(10,2)

);

CREATE TABLE spDetTransfEquipo

(

pkDetTransfEquipo integer NOT NULL DEFAULT NEXTVAL(('"spdettransfequipo\_pkdettransfequipo\_seq"'::text)::regclass),

fkTransfEquipo integer NOT NULL,

item integer,

fkEquipo integer NOT NULL,

observacion varchar(50),

hmto decimal(10,2)

);

CREATE TABLE spDetTransfPersonal

(

pkDetTransfPersonal integer NOT NULL DEFAULT NEXTVAL(('"spdettransfpersonal\_pkdettransfpersonal\_seq"'::text)::regclass),

fkTransfPersonal integer NOT NULL,

item integer,

fkPersonal integer NOT NULL,

observacion varchar(50)

);

CREATE TABLE spEqModelo

(

pkEqModelo integer NOT NULL DEFAULT NEXTVAL(('"speqmodelo\_pkeqmodelo\_seq"'::text)::regclass),

fkEqTipo integer NOT NULL,

codigo char(2) NOT NULL,

descripcion varchar(25)

);

CREATE TABLE spEqTipo

(

pkEqTipo integer NOT NULL DEFAULT NEXTVAL(('"speqtipo\_pkeqtipo\_seq"'::text)::regclass),

codigo char(2) NOT NULL,

descripcion varchar(25)

);

CREATE TABLE spEquipo

(

pkEquipo integer NOT NULL DEFAULT NEXTVAL(('"spequipo\_pkequipo\_seq"'::text)::regclass),

fkTipoEquipo integer NOT NULL,

fkModelo integer NOT NULL,

codigo char(9) NOT NULL,

fkTipoContrato char(1) NOT NULL,

fechaIngreso date,

fkOrdenTrabajo integer NOT NULL,

descripcion varchar(50)

);

CREATE TABLE spGestion

(

pkGestion integer NOT NULL DEFAULT NEXTVAL(('"spgestion\_pkgestion\_seq"'::text)::regclass),

codigo char(2) NOT NULL,

fechaIni date,

fechaFin date,

estado char(1)

);

CREATE TABLE spImproductiva

(

pkImproductiva integer NOT NULL DEFAULT NEXTVAL(('"spimproductiva\_pkimproductiva\_seq"'::text)::regclass),

codigo char(4),

descripcion varchar(50)

);

CREATE TABLE spItemObra

(

pkItemObra integer NOT NULL DEFAULT NEXTVAL(('"spitemobra\_pkitemobra\_seq"'::text)::regclass),

fkOrdenTrabajo integer NOT NULL,

fkPoligono integer NOT NULL,

fkActividad integer NOT NULL,

codigo char(6) NOT NULL,

descripcion varchar(50),

areaTrab decimal(10,2),

rendimiento decimal(10,2)

);

CREATE TABLE spOrdenTrabajo

(

pkOrdenTrabajo integer NOT NULL DEFAULT NEXTVAL(('"spordentrabajo\_pkordentrabajo\_seq"'::text)::regclass),

codigo char(3) NOT NULL,

fkGestion integer NOT NULL,

nombre varchar(50),

supervisor varchar(50),

areaEstimada decimal(10,2),

estado char(1),

data char(6)

);

CREATE TABLE spParteDiario

(

pkParteDiario integer NOT NULL DEFAULT NEXTVAL(('"sppartediario\_pkpartediario\_seq"'::text)::regclass),

fkOrdenTrabajo integer NOT NULL,

fecha date NOT NULL,

fkEquipo integer NOT NULL,

fkTurno integer NOT NULL,

fkOperador integer NOT NULL,

fkAyudante integer NOT NULL,

observacion varchar(50)

);

CREATE TABLE spPersonal

(

pkPersonal integer NOT NULL DEFAULT NEXTVAL(('"sppersonal\_pkpersonal\_seq"'::text)::regclass),

fechaIngreso date,

nombreComp varchar(50),

apellidos varchar(50),

direccion varchar(50),

telefono varchar(50),

ci varchar(25),

fechaNac date,

fkcargo integer,

fkOrdenTrabajo integer,

email varchar(25)

);

CREATE TABLE spPoligono

(

pkPoligono integer NOT NULL DEFAULT NEXTVAL(('"sppoligono\_pkpoligono\_seq"'::text)::regclass),

fkOrdenTrabajo integer NOT NULL,

codigo char(3) NOT NULL,

descripcion varchar(50)

);

CREATE TABLE spTransfEquipo

(

pkTransfEquipo integer NOT NULL DEFAULT NEXTVAL(('"sptransfequipo\_pktransfequipo\_seq"'::text)::regclass),

codigo char(4) NOT NULL,

fkGestion integer NOT NULL,

fecha date,

fkOrdenOrigen integer NOT NULL,

fkOrdenDestino integer NOT NULL,

observacion varchar(50),

data char(7) NOT NULL,

estado char(1) NOT NULL

);

CREATE TABLE spTransfPersonal

(

pkTransfPersonal integer NOT NULL DEFAULT NEXTVAL(('"sptransfpersonal\_pktransfpersonal\_seq"'::text)::regclass),

codigo char(4) NOT NULL,

fkGestion integer NOT NULL,

fecha date,

fkOrdenOrigen integer NOT NULL,

fkOrdenDestino integer NOT NULL,

observacion varchar(50),

data char(7),

estado char(1)

);

CREATE TABLE spTurno

(

pkTurno integer NOT NULL DEFAULT NEXTVAL(('"spturno\_pkturno\_seq"'::text)::regclass),

codigo varchar(3),

descripcion varchar(25),

fkOrdenTrabajo integer

);

/\* Create Primary Keys, Indexes, Uniques, Checks \*/

ALTER TABLE spActividad ADD CONSTRAINT PK\_spActividad

PRIMARY KEY (pkActividad);

CREATE INDEX IXFK\_spActividad\_spOrdenTrabajo ON spActividad (fkOrdenTrabajo ASC);

ALTER TABLE spCargo ADD CONSTRAINT PK\_spCargo

PRIMARY KEY (pkCargo);

ALTER TABLE spDetImproductiva ADD CONSTRAINT PK\_Table1

PRIMARY KEY (pkDetImproductiva);

CREATE INDEX IXFK\_spDetImproductiva\_spImproductiva ON spDetImproductiva (fkImproductiva ASC);

CREATE INDEX IXFK\_spDetImproductiva\_spParteDiario ON spDetImproductiva (fkParteDiario ASC);

ALTER TABLE spDetProductiva ADD CONSTRAINT PK\_spDetParteDiario

PRIMARY KEY (pkDetParteDiario);

CREATE INDEX IXFK\_spDetProductiva\_spItemObra ON spDetProductiva (fkItemObra ASC);

CREATE INDEX IXFK\_spDetProductiva\_spParteDiario ON spDetProductiva (fkParteDiario ASC);

ALTER TABLE spDetTransfEquipo ADD CONSTRAINT PK\_spDetTransfEquipo

PRIMARY KEY (pkDetTransfEquipo);

CREATE INDEX IXFK\_spDetTransfEquipo\_spEquipo ON spDetTransfEquipo (fkEquipo ASC);

CREATE INDEX IXFK\_spDetTransfEquipo\_spTransfEquipo ON spDetTransfEquipo (fkTransfEquipo ASC);

ALTER TABLE spDetTransfPersonal ADD CONSTRAINT PK\_spDetTransfPersonal

PRIMARY KEY (pkDetTransfPersonal);

CREATE INDEX IXFK\_spDetTransfPersonal\_spPersonal ON spDetTransfPersonal (fkPersonal ASC);

CREATE INDEX IXFK\_spDetTransfPersonal\_spTransfPersonal ON spDetTransfPersonal (fkTransfPersonal ASC);

ALTER TABLE spEqModelo ADD CONSTRAINT PK\_spEqModelo

PRIMARY KEY (pkEqModelo);

CREATE INDEX IXFK\_spEqModelo\_spEqTipo ON spEqModelo (fkEqTipo ASC);

ALTER TABLE spEqTipo ADD CONSTRAINT PK\_spEqTipo

PRIMARY KEY (pkEqTipo);

ALTER TABLE spEquipo ADD CONSTRAINT PK\_spEquipo

PRIMARY KEY (pkEquipo);

CREATE INDEX IXFK\_spEquipo\_spEqModelo ON spEquipo (fkModelo ASC);

CREATE INDEX IXFK\_spEquipo\_spEqTipo ON spEquipo (fkTipoEquipo ASC);

CREATE INDEX IXFK\_spEquipo\_spOrdenTrabajo ON spEquipo (fkOrdenTrabajo ASC);

ALTER TABLE spGestion ADD CONSTRAINT PK\_spGestion

PRIMARY KEY (pkGestion);

ALTER TABLE spImproductiva ADD CONSTRAINT PK\_spImproductiva

PRIMARY KEY (pkImproductiva);

ALTER TABLE spItemObra ADD CONSTRAINT PK\_spItemObra

PRIMARY KEY (pkItemObra);

CREATE INDEX IXFK\_spItemObra\_spActividad ON spItemObra (fkActividad ASC);

CREATE INDEX IXFK\_spItemObra\_spOrdenTrabajo ON spItemObra (fkOrdenTrabajo ASC);

CREATE INDEX IXFK\_spItemObra\_spPoligono ON spItemObra (fkPoligono ASC);

ALTER TABLE spOrdenTrabajo ADD CONSTRAINT PK\_spOrdenTrabajo

PRIMARY KEY (pkOrdenTrabajo);

CREATE INDEX IXFK\_spOrdenTrabajo\_spGestion ON spOrdenTrabajo (fkGestion ASC);

ALTER TABLE spParteDiario ADD CONSTRAINT PK\_ParteDiario

PRIMARY KEY (pkParteDiario);

CREATE INDEX IXFK\_spParteDiario\_spEquipo ON spParteDiario (fkEquipo ASC);

CREATE INDEX IXFK\_spParteDiario\_spOrdenTrabajo ON spParteDiario (fkOrdenTrabajo ASC);

CREATE INDEX IXFK\_spParteDiario\_spPersonal ON spParteDiario (fkOperador ASC);

CREATE INDEX IXFK\_spParteDiario\_spPersonal\_02 ON spParteDiario (fkAyudante ASC);

CREATE INDEX IXFK\_spParteDiario\_spTurno ON spParteDiario (fkTurno ASC);

ALTER TABLE spPersonal ADD CONSTRAINT PK\_spPersonal

PRIMARY KEY (pkPersonal);

CREATE INDEX IXFK\_spPersonal\_spCargo ON spPersonal (fkcargo ASC);

CREATE INDEX IXFK\_spPersonal\_spOrdenTrabajo ON spPersonal (fkOrdenTrabajo ASC);

ALTER TABLE spPoligono ADD CONSTRAINT PK\_spPoligono

PRIMARY KEY (pkPoligono);

CREATE INDEX IXFK\_spPoligono\_spOrdenTrabajo ON spPoligono (fkOrdenTrabajo ASC);

ALTER TABLE spTransfEquipo ADD CONSTRAINT PK\_spTransfEquipo

PRIMARY KEY (pkTransfEquipo);

CREATE INDEX IXFK\_spTransfEquipo\_spGestion ON spTransfEquipo (fkGestion ASC);

CREATE INDEX IXFK\_spTransfEquipo\_spOrdenTrabajo ON spTransfEquipo (fkOrdenOrigen ASC);

CREATE INDEX IXFK\_spTransfEquipo\_spOrdenTrabajo\_02 ON spTransfEquipo (fkOrdenDestino ASC);

ALTER TABLE spTransfPersonal ADD CONSTRAINT PK\_spTransfPersonal

PRIMARY KEY (pkTransfPersonal);

CREATE INDEX IXFK\_spTransfPersonal\_spGestion ON spTransfPersonal (fkGestion ASC);

CREATE INDEX IXFK\_spTransfPersonal\_spOrdenTrabajo ON spTransfPersonal (fkOrdenOrigen ASC);

CREATE INDEX IXFK\_spTransfPersonal\_spOrdenTrabajo\_02 ON spTransfPersonal (fkOrdenDestino ASC);

ALTER TABLE spTurno ADD CONSTRAINT PK\_spTurno

PRIMARY KEY (pkTurno);

CREATE INDEX IXFK\_spTurno\_spOrdenTrabajo ON spTurno (fkOrdenTrabajo ASC);

/\* Create Foreign Key Constraints \*/

ALTER TABLE spActividad ADD CONSTRAINT FK\_spActividad\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetImproductiva ADD CONSTRAINT FK\_spDetImproductiva\_spImproductiva

FOREIGN KEY (fkImproductiva) REFERENCES spImproductiva (pkImproductiva) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetImproductiva ADD CONSTRAINT FK\_spDetImproductiva\_spParteDiario

FOREIGN KEY (fkParteDiario) REFERENCES spParteDiario (pkParteDiario) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetProductiva ADD CONSTRAINT FK\_spDetProductiva\_spItemObra

FOREIGN KEY (fkItemObra) REFERENCES spItemObra (pkItemObra) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetProductiva ADD CONSTRAINT FK\_spDetProductiva\_spParteDiario

FOREIGN KEY (fkParteDiario) REFERENCES spParteDiario (pkParteDiario) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetTransfEquipo ADD CONSTRAINT FK\_spDetTransfEquipo\_spEquipo

FOREIGN KEY (fkEquipo) REFERENCES spEquipo (pkEquipo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetTransfEquipo ADD CONSTRAINT FK\_spDetTransfEquipo\_spTransfEquipo

FOREIGN KEY (fkTransfEquipo) REFERENCES spTransfEquipo (pkTransfEquipo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetTransfPersonal ADD CONSTRAINT FK\_spDetTransfPersonal\_spPersonal

FOREIGN KEY (fkPersonal) REFERENCES spPersonal (pkPersonal) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spDetTransfPersonal ADD CONSTRAINT FK\_spDetTransfPersonal\_spTransfPersonal

FOREIGN KEY (fkTransfPersonal) REFERENCES spTransfPersonal (pkTransfPersonal) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spEqModelo ADD CONSTRAINT FK\_spEqModelo\_spEqTipo

FOREIGN KEY (fkEqTipo) REFERENCES spEqTipo (pkEqTipo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spEquipo ADD CONSTRAINT FK\_spEquipo\_spEqModelo

FOREIGN KEY (fkModelo) REFERENCES spEqModelo (pkEqModelo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spEquipo ADD CONSTRAINT FK\_spEquipo\_spEqTipo

FOREIGN KEY (fkTipoEquipo) REFERENCES spEqTipo (pkEqTipo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spEquipo ADD CONSTRAINT FK\_spEquipo\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spItemObra ADD CONSTRAINT FK\_spItemObra\_spActividad

FOREIGN KEY (fkActividad) REFERENCES spActividad (pkActividad) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spItemObra ADD CONSTRAINT FK\_spItemObra\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spItemObra ADD CONSTRAINT FK\_spItemObra\_spPoligono

FOREIGN KEY (fkPoligono) REFERENCES spPoligono (pkPoligono) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spOrdenTrabajo ADD CONSTRAINT FK\_spOrdenTrabajo\_spGestion

FOREIGN KEY (fkGestion) REFERENCES spGestion (pkGestion) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spParteDiario ADD CONSTRAINT FK\_spParteDiario\_spEquipo

FOREIGN KEY (fkEquipo) REFERENCES spEquipo (pkEquipo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spParteDiario ADD CONSTRAINT FK\_spParteDiario\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spParteDiario ADD CONSTRAINT FK\_spParteDiario\_spPersonal

FOREIGN KEY (fkOperador) REFERENCES spPersonal (pkPersonal) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spParteDiario ADD CONSTRAINT FK\_spParteDiario\_spPersonal\_02

FOREIGN KEY (fkAyudante) REFERENCES spPersonal (pkPersonal) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spParteDiario ADD CONSTRAINT FK\_spParteDiario\_spTurno

FOREIGN KEY (fkTurno) REFERENCES spTurno (pkTurno) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spPersonal ADD CONSTRAINT FK\_spPersonal\_spCargo

FOREIGN KEY (fkcargo) REFERENCES spCargo (pkCargo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spPersonal ADD CONSTRAINT FK\_spPersonal\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spPoligono ADD CONSTRAINT FK\_spPoligono\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spTransfEquipo ADD CONSTRAINT FK\_spTransfEquipo\_spGestion

FOREIGN KEY (fkGestion) REFERENCES spGestion (pkGestion) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spTransfEquipo ADD CONSTRAINT FK\_spTransfEquipo\_spOrdenTrabajo\_02

FOREIGN KEY (fkOrdenDestino) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spTransfPersonal ADD CONSTRAINT FK\_spTransfPersonal\_spGestion

FOREIGN KEY (fkGestion) REFERENCES spGestion (pkGestion) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spTransfPersonal ADD CONSTRAINT FK\_spTransfPersonal\_spOrdenTrabajo

FOREIGN KEY (fkOrdenOrigen) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spTransfPersonal ADD CONSTRAINT FK\_spTransfPersonal\_spOrdenTrabajo\_02

FOREIGN KEY (fkOrdenDestino) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

ALTER TABLE spTurno ADD CONSTRAINT FK\_spTurno\_spOrdenTrabajo

FOREIGN KEY (fkOrdenTrabajo) REFERENCES spOrdenTrabajo (pkOrdenTrabajo) ON DELETE No Action ON UPDATE No Action;

/\* Create Table Comments, Sequences for Autonumber Columns \*/

CREATE SEQUENCE spactividad\_pkactividad\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spcargo\_pkcargo\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spdetimproductiva\_pkdetimproductiva\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spdetproductiva\_pkdetpartediario\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spdettransfequipo\_pkdettransfequipo\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spdettransfpersonal\_pkdettransfpersonal\_seq INCREMENT 1 START 1;

CREATE SEQUENCE speqmodelo\_pkeqmodelo\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spitemobra\_pkitemobra\_seq INCREMENT 1 START 1;

CREATE SEQUENCE spordentrabajo\_pkordentrabajo\_seq INCREMENT 1 START 1;

CREATE SEQUENCE sppartediario\_pkpartediario\_seq INCREMENT 1 START 1;

CREATE SEQUENCE sppersonal\_pkpersonal\_seq INCREMENT 1 START 1;

CREATE SEQUENCE sppoligono\_pkpoligono\_seq INCREMENT 1 START 1;

## DISEÑO DE DETALLE PROCEDIMENTAL

### 3.4.1. DIAGRAMA DE CLASE DINAMICA Y DIAGRAMA DE SECUENCIA

#### CU01: Gestionar personal



#### CU02: Gestionar equipos pesados





#### CU03: Gestionar orden de trabajos





#### CU04: Gestionar polígonos de obra





#### CU05: Gestión de actividades de obra





#### CU06: Gestionar ítem obra



#### CU07: Gestionar usuarios de sistema





#### CU08: Administración de gestiones





#### CU09: Estadísticas de visitas



# FLUJO DE TRABAJO IMPLEMENTACION.

De forma más específica el propósito de la implementación es; planificar las integraciones de sistema necesarias en cada iteración, distribuir el sistema asignando componentes ejecutables a nodos en el diagrama de despliegue, implementar las clases y subsistemas encontrados durante el diseño, probar los componentes individualmente, y a continuación integrarlos.

## Arquitectura de la Implementación:

El propósito de la implementación de la arquitectura es esbozar el modelo de implementación y su arquitectura mediante; la identificación de componentes significativos arquitectónicamente, la asignación de componentes a los nodos en las configuraciones de redes relevantes.

## Diagrama de despliegue



# FLUJO DE TRABAJO PRUEBAS

## CU01: Gestionar personal



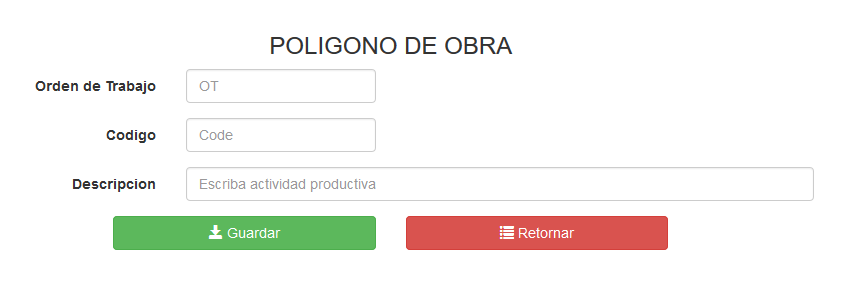
|  |  |
| --- | --- |
| **Entrada** | Guarda los datos personales |
| **Salida** | Guarda los datos ingresados al sistema. |
| **Condiciones** | 1.- Si previamente selecciono la opcion de editar mostrara esta pantalla  2.- Si presiono en el botón de nuevo personal mostrara esta venrtana |
| **Procedimientos** | 1.- ingresa la fecha.  2.- enseguida teclea el nombre y apellido de la persona.  3.- adiciona la dirección, correo electrónico, teléfono y ci.  4.- Selecciona el cargo de la lista desplegable cargo  5.- a continuación ingresa la fecha de nacimiento  6.- Ingresa del codigo de la orden de trabajo. |

## CU02: Gestionar Equipo



|  |  |
| --- | --- |
| **Entrada** | Guarda los datos de los Equipos |
| **Salida** | Guarda los datos ingresados al sistema. |
| **Condiciones** | 1.- Si previamente selecciono la opcion de editar mostrara esta pantalla  2.- Si presiono en el botón de nuevo personal mostrara esta venrtana |
| **Procedimientos** | 1.- selecciona tipo de equipo.  2.- selecciona modelo equipo  3.- ingresa el codigo  4.- selecciona tipo de contrato  5.- Ingresa la orden de trabajo |

## CU04: Gestionar Poligono



|  |  |
| --- | --- |
| **Entrada** | Guarda los datos de los Poligonos |
| **Salida** | Guarda los datos ingresados al sistema. |
| **Condiciones** | 1.- Si previamente selecciono la opcion de editar mostrara esta pantalla  2.- Si presiono en el botón de nuevo personal mostrara esta venrtana |
| **Procedimientos** | 1.- selecciona orden de trabajo.  2.- selecciona código  3.- detallar una descripción |

# ANEXOS

**[UMLREF]** Grady Booch, James Rumbaugh, Ivar Jacobson, *“Lenguaje Unificado de Modelado – Manual de Referencia”.*

Ediciónprimera en 1999, última edición 2001Madrid-España, editorial Addison Wesley Iberoamericana

**[UMLPRO]** Grady Booch, James Rumbaugh, Ivar Jacobson, *“El Proceso Unificado de Desarrollo de Software.*

Ediciónprimera en 1999, última edición 2001Madrid-España, editorial Addison Wesley Iberoamericana.

**[PRE4E]** Roger S. Pressman*, “Ingeniera del Software un enfoque práctico”.*

Quinta edición en 2002, Madrid-España, editorial McGraw Hill, título original: ”Software engineering a practitioner’s approach”, Traducido por: Rafael Ojeda Martín, Isabel Morales Jareño, Virgilio YAgüe Galaup, Salvador Sánchez Alonso.

**[CRAIGL99]** Craig Larman “UML y Patrones” Prentice Hall – Mexico 99