

Análisis y diseño Proyecto RAD

[Proyecto RAD]
Fecha: *[03/04/2020]*

Tabla de contenido

Historial de Versiones	3
Información del proyecto.....	3
Sistemas y niveles de contexto según usuario.....	4
Descripción del dominio de la función	6
Dominio de la función: Agregar Trabajador.....	6
Dominio de la función: Buscar Trabajador.....	10
Dominio de la función: Editar Trabajador.....	14
Dominio de la función: Eliminar Trabajador.....	18
Dominio de la función: Visualizar Registro.....	21
Dominio de la función: Visualizar Trabajador.....	25
Dominio de la función: Ingresar Registro.....	28
Dominio de la función: Iniciar Sesión.....	31
Dominio de la función: Generar Registro.....	34

Historial de Versiones

Fecha	Versión	Autor	Organización	Descripción
04.10.19	1.0	Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 1 Proyecto RAD
11.10.19	2.0	Christian Matzner Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 2 Proyecto RAD
25.11.19	3.0	Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 3 Proyecto RAD
13.12.19	4.0	Christian Matzner Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 4 Proyecto RAD
22.12.19	5.0	Christian Matzner	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 5 Proyecto RAD
27.02.20	6.0	Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 6 Proyecto RAD
13.03.20	7.0	Christian Matzner Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 7 Proyecto RAD
25.03.20	8.0	Christian Matzner Manuel Jaramillo	Henry Fehrmann, Sebastián Molina, Christian Matzner, Manuel Jaramillo	Análisis y diseño Sprint 8 Proyecto RAD

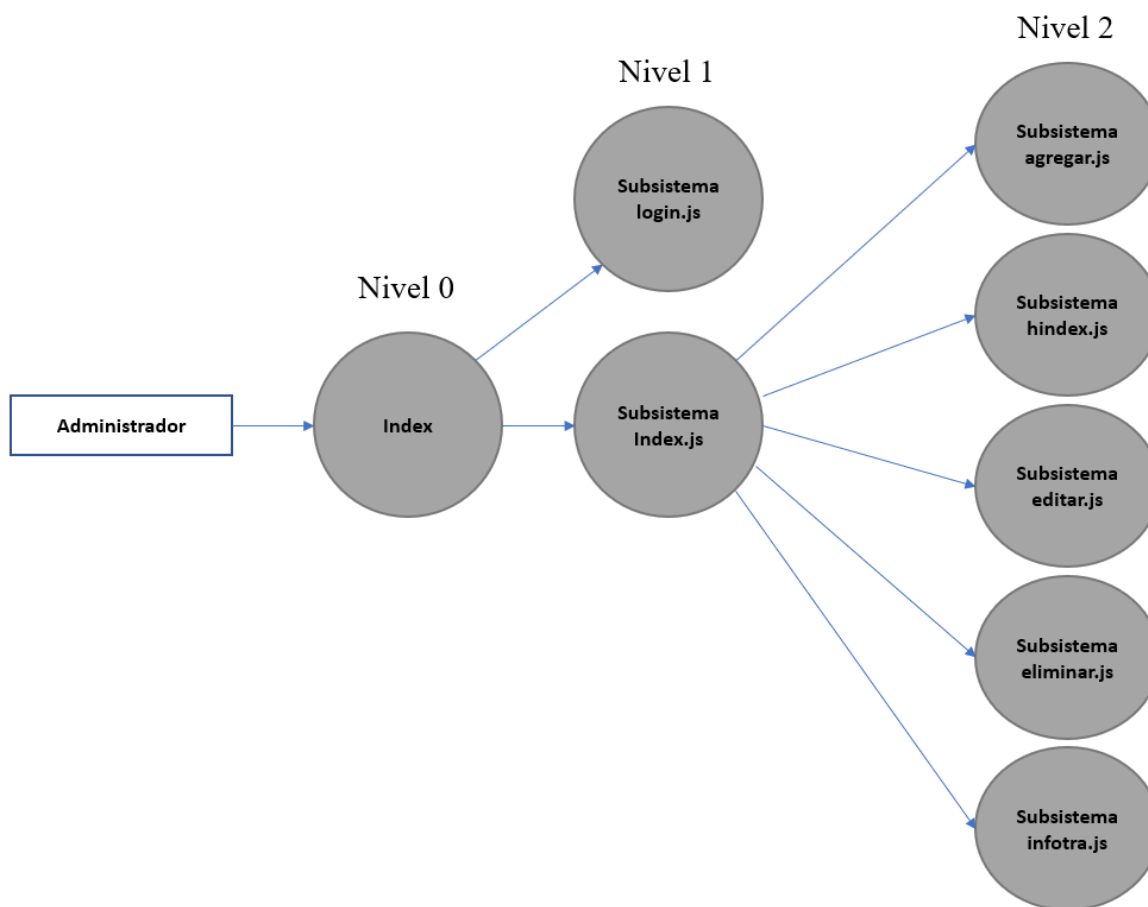
Información del Proyecto

Empresa / Organización	Empresa Lechera Santa Fé Ltda.
Proyecto	RAD (Registro de Asistencia Digital)
Fecha de preparación	29 de Agosto del 2019
Cliente	Jorge Matzner
Patrocinador principal	Jorge Matzner
Gerente / Líder de Proyecto	Christian Matzner
Gerente / Líder de Desarrollo de Software	Sebastian Molina

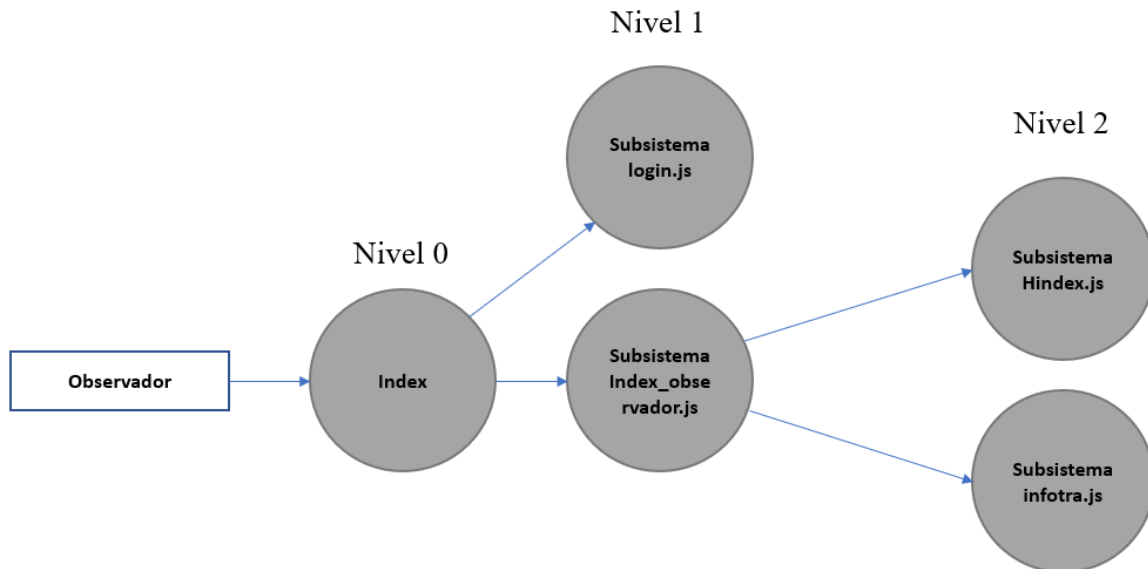
Sistemas y niveles de contexto según usuario

En este apartado se presentan los Sistemas a los que se puede acceder dependiendo del usuario que esté haciendo uso del Software, además de los niveles de contexto a los que pertenecen cada uno en caso de que se trate de subsistemas.

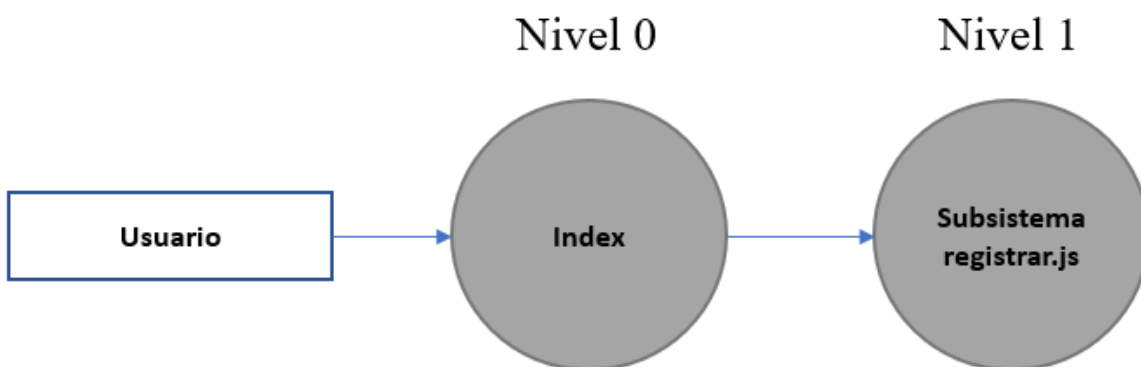
[Administrador]



[Observador]



[Usuario]



Descripción del dominio de la función

En este apartado se presenta la funcionalidad del sistema que satisface los objetivos, requisitos y restricciones expresadas en el apartado anterior, descripción detallada del problema. Para ello se hará uso de los Diagramas de Flujo de Datos.

[Agregar Trabajador]

El Sistema debe permitir agregar un nuevo empleado si es que se contrata nuevo personal.

Nivel 0 de contexto

Entidades

-Entidad externa **administrador**: Representa a una persona que se encarga de la labor de ingresar la información de nuevos trabajadores.

-Entidad externa **pantalla del ordenador**: Representa la pantalla de cualquier terminal de ordenador con el que se haya accedido a nuestro sistema, allí se irá mostrando información de interés.

Proceso

-**Index**: El Sistema se encarga de validar que la información que haya sido ingresada por el administrador sea correcta, en cuanto a formato.

Flujo de datos

-Flujo de **administrador**: Contiene datos acerca de los usuarios a ingresar por parte del administrador, para ser agregados al sistema.

-Flujo de **pantalla externa**: Representa la *información* emitida por el sistema para validar el ingreso.

-Flujo de datos **flujo_formulario**: Solicita y envía los datos que tienen que ser agregados a la base de datos.



Nivel 1 de contexto

Subsistemas

-Subsistema **index.js**: Este subsistema se encarga de gestionar las peticiones del administrador acerca de las solicitudes de ingresos de nuevos trabajadores.

Flujo de datos

- Flujo de datos **salida_administrador**: Representan los datos que se le mostrarán al administrador, en este caso, corresponde a el botón que le permite agregar un nuevo trabajador.



Nivel 2 de contexto

Subsistemas

- Subsistema **agregar.js**: Se encarga de generar el formulario que el administrador u observador debe rellenar.

Procesos:

-Proceso **salir**: Regresa al subsistema principal “index”.

-Proceso **validator.js**: Le muestra al usuario, por medio de mensajes, que el formato que ingresó en la casilla de algún dato no corresponde al solicitado.

-Proceso **agrega_trab_formulario**: Genera un arreglo (const) en base a los datos que se hayan ingresado.

Flujo de datos

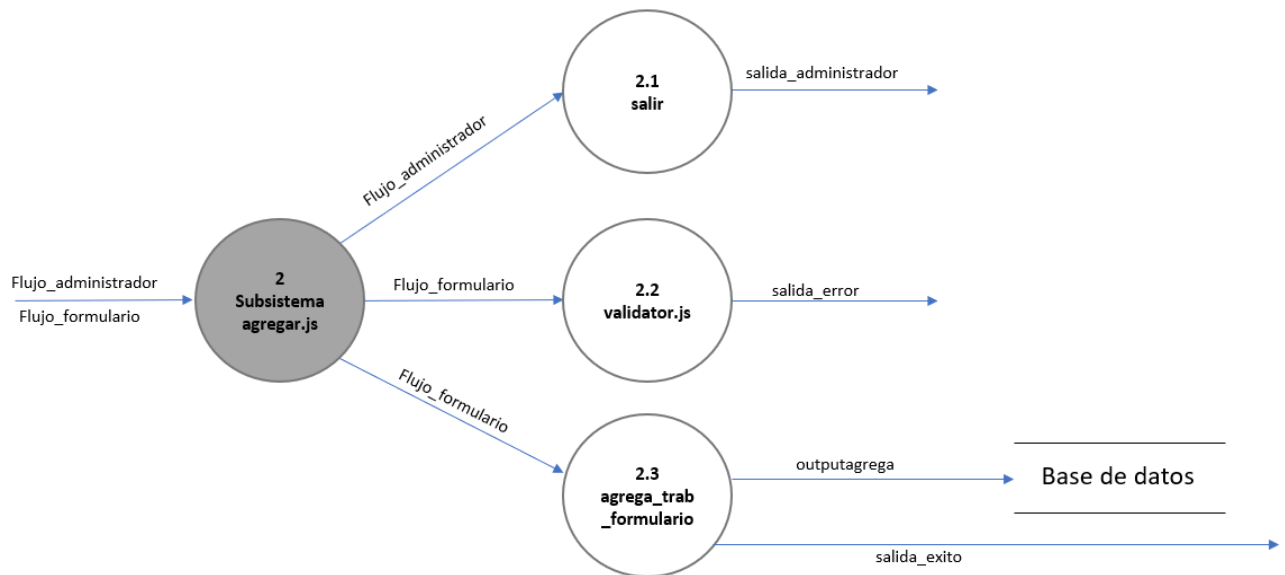
- Flujo de datos **salida_error**: Representa los mensajes de error en caso de que se hayan ingresado datos incorrectos.

- Flujo de datos **salida_exito**: Representa un mensaje que informará el éxito de la operación de agregar un nuevo trabajador

- Flujo de datos **outputagrega**: Representa los datos que se generaron en el proceso agrega_trab_formulario, los cuales son convertidos en un objeto y almacenados en la base de datos.

Almacén

- Almacén **Base de Datos**: Representa la base de datos del Sistema, que será modificada al ingresar un nuevo trabajador por el proceso agrega_trab_formulario.



[Buscar Trabajador]

El Sistema debe permitir la búsqueda de alguno de los trabajadores de la base de datos por medio del ingreso de información sobre este.

Nivel 0 de contexto

Entidades

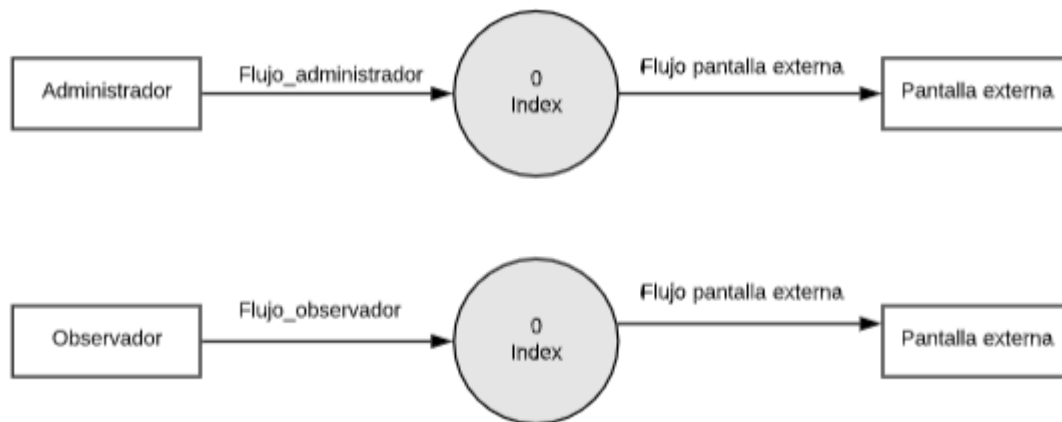
- Entidad externa **administrador**: Representa a una persona que se encarga de ingresar los datos correspondientes a los del empleado que se vaya a buscar.
- Entidad externa **pantalla del ordenador**: Se muestra la información de interés.
- Entidad externa **observador**: Representa a una persona que se encarga de ingresar los datos correspondientes a los del empleado que se vaya a buscar.

Proceso

- Index**: Gestiona la búsqueda del administrador u observador, por medio de la implementación de filtros, mostrando solamente la información que corresponda a lo que el administrador ingresó en el campo de búsqueda.

Flujo de datos

- Flujo de **administrador**: Contiene información relacionada con la búsqueda del trabajador deseado.
- Flujo de **observador**: Contiene información relacionada con la búsqueda del trabajador deseado.
- Flujo de **pantalla externa**: Representa en pantalla el resultado de la búsqueda.



Nivel 1 de contexto

Subsistemas

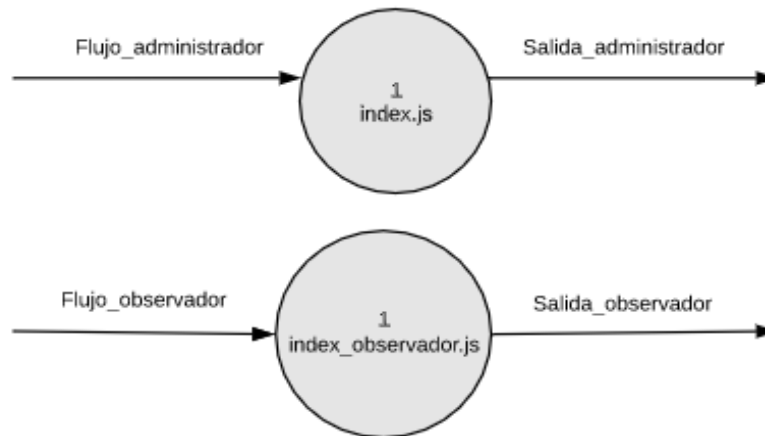
-Subsistema ***index.js***: Este subsistema se encarga de gestionar las peticiones del administrador acerca de la búsqueda de un trabajador en los registros existentes.

-Subsistema ***index_observador.js***: Este subsistema se encarga de gestionar las peticiones del observador acerca de la búsqueda de un trabajador en los registros existentes.

Flujo de datos

- Flujo de datos ***salida_administrador***: Representan los datos que se le mostrarán al administrador, en este caso, corresponde al o los usuarios que cumplan con las características de la información descrita por el administrador en su búsqueda.

- Flujo de datos ***salida_observador***: Representan los datos que se le mostrarán al observador, en este caso, corresponde al o los usuarios que cumplan con las características de la información descrita por el administrador en su búsqueda.



Nivel 2 de contexto

Subsistemas

-Subsistema **hindex.js**: Este subsistema se encarga de gestionar las actividades realizadas cuando hay cambios en el estado de la barra de búsqueda.

Procesos:

-Proceso **filtrar**: Cada vez que se ingresa o borra un carácter desde la barra de búsqueda este proceso se encarga de realizar un filtrado en la base de datos considerando el campo completo, mostrando en pantalla todos los trabajadores registrados que coincidan con las búsquedas.

Flujo de datos

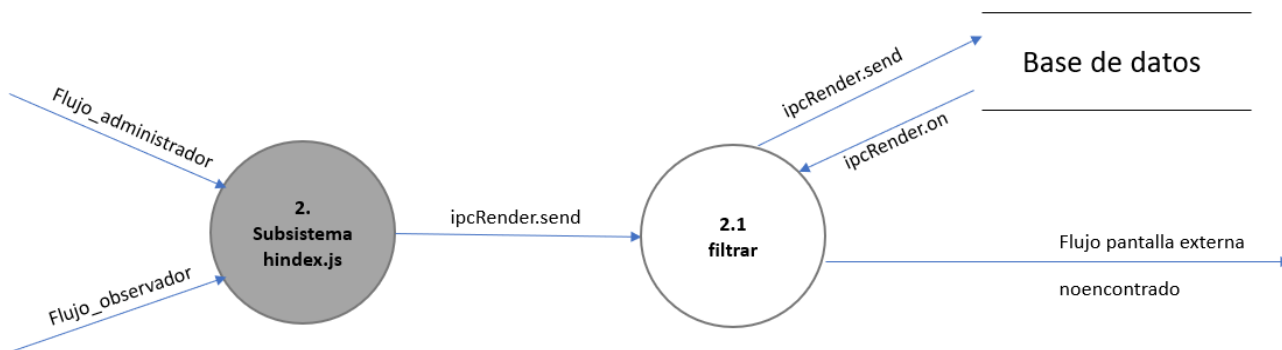
- Flujo de datos **ipcRender.send**: Representa el valor obtenido en el campo de búsqueda, que se utilizará para realizar la búsqueda en la base de datos.

- Flujo de datos **ipcRender.on**: Representa un arreglo de trabajadores proveniente de la base de datos que han coincidido de alguna forma con la búsqueda realizada.

- Flujo de datos **noencontrado**: Representa el mensaje que aparecerá en pantalla en caso de que no se haya registrado ninguna coincidencia con la búsqueda realizada.

Almacén

- Almacén **Base de Datos:** Representa la base de datos del Sistema, en donde se realizarán las consultas pertinentes dependiendo de lo ingresado en el campo de búsqueda.



[Editar Trabajador]

El sistema debe permitir al administrador editar la información del empleado si se requiere.

Nivel 0 de contexto

Entidades

- Entidad externa **administrador**: Representa a la persona que hará uso del botón “editar” y además sobrescribirá la información del usuario que se muestre en el formulario.
- Entidad externa **pantalla del ordenador**: Muestra la información del usuario a editar en forma de formulario.

Proceso

- Index**: Gestiona la solicitud de edición del administrador, mostrando en pantalla un formulario editable, que contiene la información del trabajador seleccionado.

Flujo de datos

- Flujo de **administrador**: Envía el formulario al sistema para que actualice los datos del trabajador editado
- Flujo de **pantalla externa**: Muestra en pantalla la confirmación de la actualización de los datos
- Flujo de datos **flujo_formulario**: Solicita y envía los datos que tienen que ser modificados en la base de datos.



Nivel 1 de contexto

Subsistemas

-Subsistema ***index.js***: Este subsistema se encarga de gestionar las peticiones del administrador acerca de la edición de la información de algún trabajador.

Flujo de datos

- Flujo de datos ***salida_administrador***: Representan los datos que se le mostrarán al administrador, en este caso, corresponde los datos, mostrados en forma de formulario, correspondientes al trabajador en cuestión, permitiendo sobrescribir estos.



Nivel 2 de contexto

Subsistemas

-Subsistema ***editar.js***: Este subsistema se encarga de las tareas correspondientes a la solicitud del administrador de editar un trabajador.

Proceso

-Proceso ***validator.js***: Le muestra al usuario, por medio de mensajes, que el formato que ingresó en la casilla de algún dato no corresponde al solicitado.

- Proceso ***val_rut.js***: Este proceso se encarga de verificar si el rut ingresado es válido, en base a un algoritmo interno.

- Proceso ***sacardatos(id)***: Este proceso se encarga de buscar en la base de datos, la información correspondiente al trabajador seleccionado y la plasma en un formulario que el administrador puede editar.

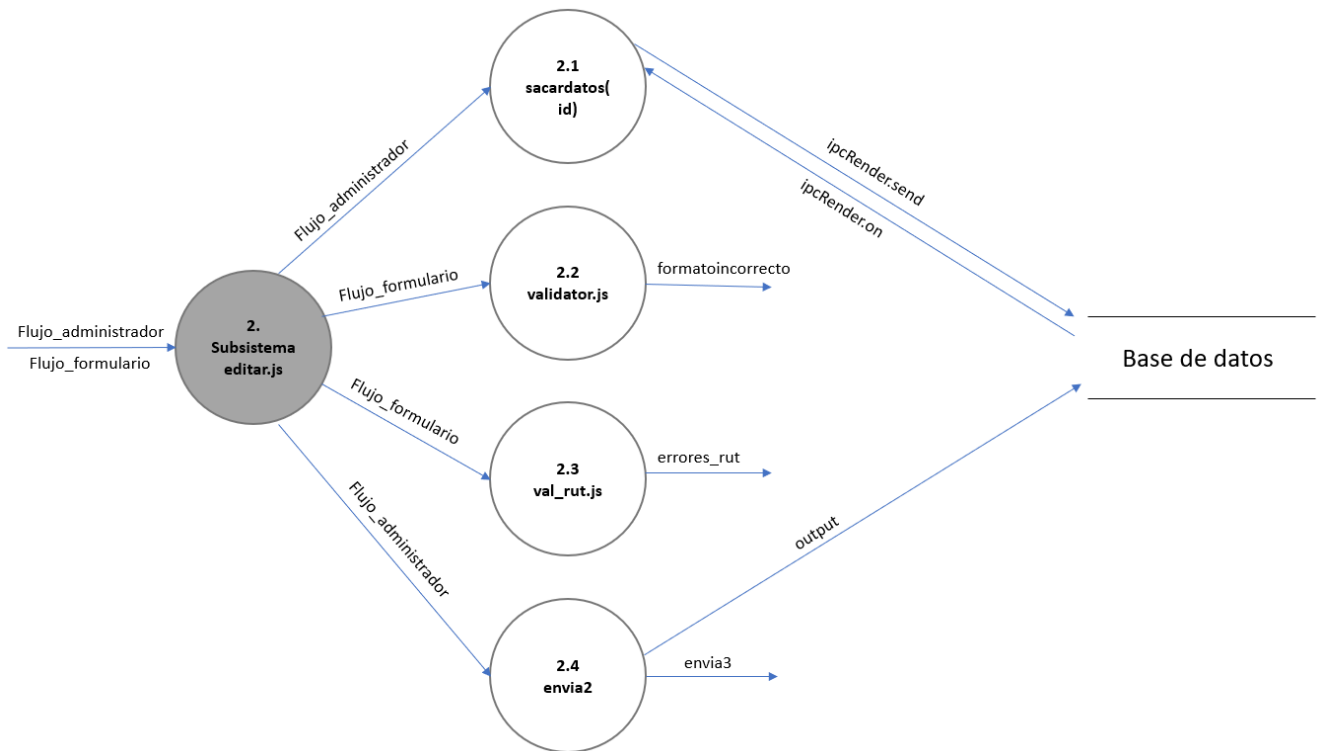
- Proceso **envia2**: Este proceso se encarga de verificar que todos los datos ingresados sean válidos, crea un arreglo con estos y lo sobrescribe en la base de datos, en la sección correspondiente según el id del trabajador.

Flujo de datos

- Flujo de datos **errores_rut**: Muestra un mensaje en pantalla acerca de que el rut ingresado es incorrecto.
- Flujo de datos **formatoincorrecto**: Representa el mensaje que se mostrará en pantalla cuando el formato de los datos ingresados no corresponda al del campo seleccionado.
- Flujo de datos **output**: Representa los valores del formulario que se enviarán a la base de datos.
- Flujo de datos **envia3**: Muestra un mensaje en pantalla confirmando que se han actualizado correctamente los datos del trabajador.
- Flujo de datos **ipcRender.send**: Representa el id del trabajador a editar, el cual será buscado en la base de datos.
- Flujo de datos **ipcRender.on**: Representa un arreglo con los datos del trabajador seleccionado, obtenido de la base de datos, la cual se mostrará en pantalla para ser modificado.

Almacén

- Almacén **Base de Datos**: Representa la base de datos del Sistema, en donde se realizarán la búsqueda de la información del trabajador y se sobrescribirá una vez finalizado el proceso.



[Eliminar Trabajador]

El sistema debe permitir eliminar un empleado si este deja de trabajar en la empresa.

Nivel 0 de contexto

Entidades

-Entidad externa **administrador**: Representa a una persona que tiene la facultad de eliminar a un usuario.

-Entidad externa **pantalla del ordenador**: Muestra en pantalla la confirmación de que el usuario fue eliminado correctamente o no.

Proceso

-**Index**: Solicita una confirmación de la operación al administrador, para confirmar que no se trata de un “click accidental”, luego al ser eliminado un trabajador de manera, el sistema informará al usuario, por medio de un mensaje mostrado en pantalla, el éxito de la operación.

Flujo de datos

-Flujo de **administrador**: Contiene la petición de eliminar a un usuario especificado.

-Flujo de **pantalla externa**: Representa la *información* emitida por el sistema para validar la eliminación y confirmar esta.



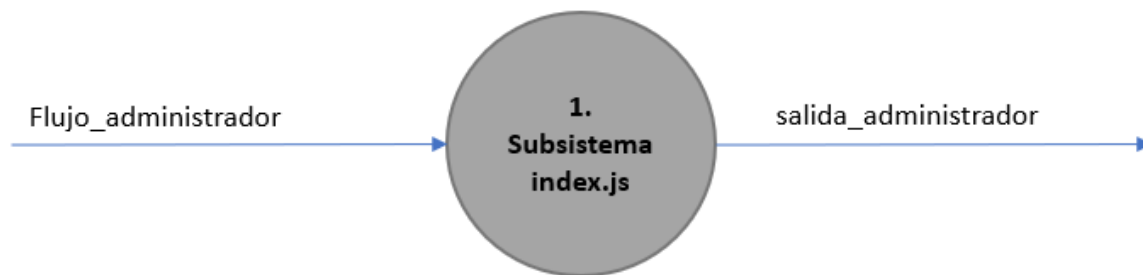
Nivel 1 de contexto

Subsistemas

-Subsistema ***index.js***: Este subsistema se encarga de gestionar las peticiones del administrador acerca de la eliminación de un trabajador.

Flujo de datos

- Flujo de datos ***salida_administrador***: Representan los datos que se le mostrarán al administrador, en este caso, una solicitud de confirmación de eliminación, además de un mensaje mostrando el éxito de la operación.



Nivel 2 de contexto

Subsistemas

-Subsistema ***eliminar.js***: Este subsistema se encarga de mostrar una pantalla similar a la de *index.js*, mostrando un buscador, una lista de trabajadores con una cantidad de datos asociados más reducida y un botón asociado a cada uno, que le dará al administrador la opción de eliminar al trabajador.

Proceso

-***enviarid***: Una vez que se haya presionado el botón eliminar, para consultarle al administrador si está seguro de realizar la operación.

Flujo de datos

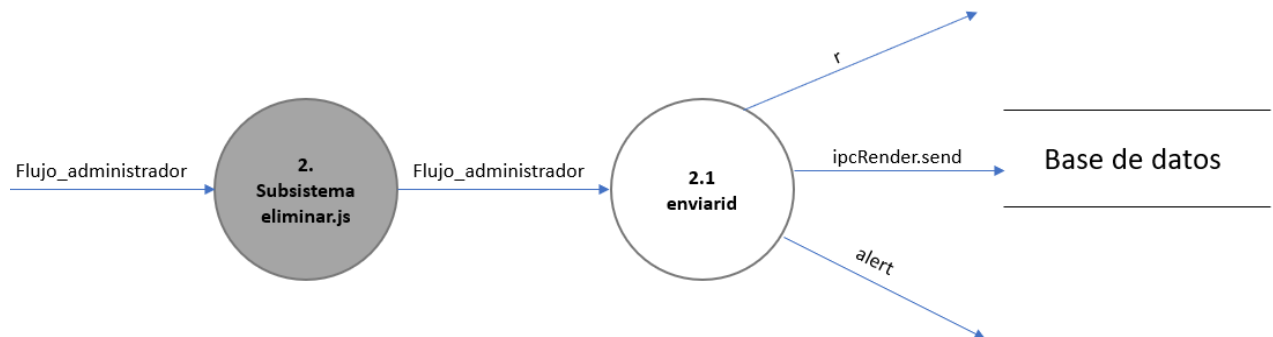
- Flujo de datos ***r***: Representa una casilla que se mostrará en pantalla donde se le dará la opción al administrador de confirmar o cancelar la operación.

- Flujo de datos ***ipcRender.send***: Representa la información en forma de id que se enviará a la base de datos y modificará el atributo "eliminado" del trabajador seleccionado por un "1", con esto se ocultará este de las búsquedas.

- Flujo de datos **alert**: Muestra un mensaje en pantalla confirmando la eliminación del trabajador.

Almacén

- Almacén **Base de Datos**: Representa la base de datos del Sistema, en donde se modificará el atributo “eliminado” del trabajador.



[Visualizar Registro]

El Sistema debe mostrar, por medio de interfaz gráfica, una tabla con todos los trabajadores y los horarios respectivos que han cumplido, pudiendo seleccionar uno y ver el historial personal del trabajador.

Nivel 0 de contexto

Entidades

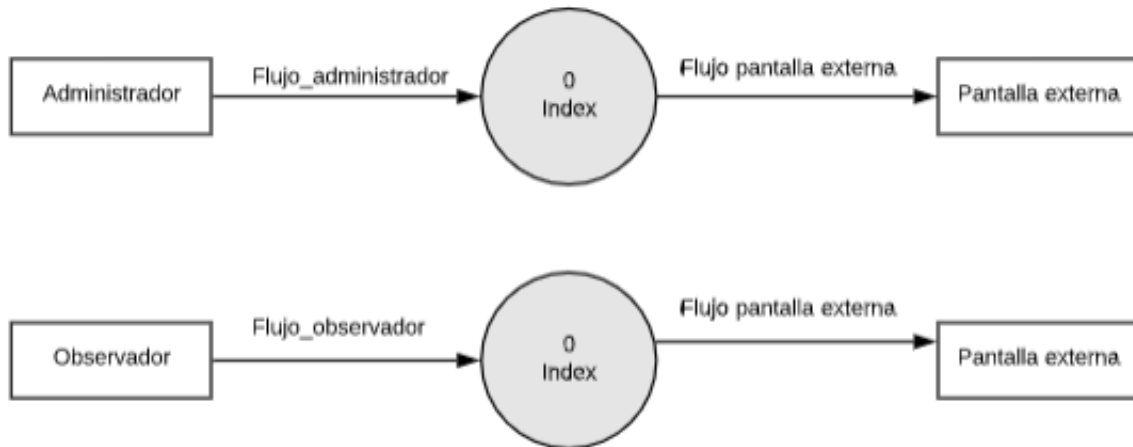
- Entidad externa **administrador**: Representa a la persona que visualiza la tabla con la información de los trabajadores existentes en el sistema.
- Entidad externa **pantalla del ordenador**: Muestra en pantalla las tablas solicitadas.
- Entidad externa **observador**: Representa a la persona que visualiza la tabla con la información de los trabajadores existentes en el sistema.

Proceso

- Index**: El Sistema despliega en pantalla la información correspondiente a la solicitud de visualización del administrador u observador, registrada por medio del buscador.

Flujo de datos

- Flujo de **administrador**: Representa la información del usuario que el administrador desea visualizar y que es enviada al sistema.
- Flujo de **pantalla externa**: Representa la información emitida por el sistema, correspondiente a la solicitud del administrador u observador.
- Flujo de **observador**: Representa la información del usuario que el observador desea visualizar y que es enviada al sistema.



Nivel 1 de contexto

Subsistemas

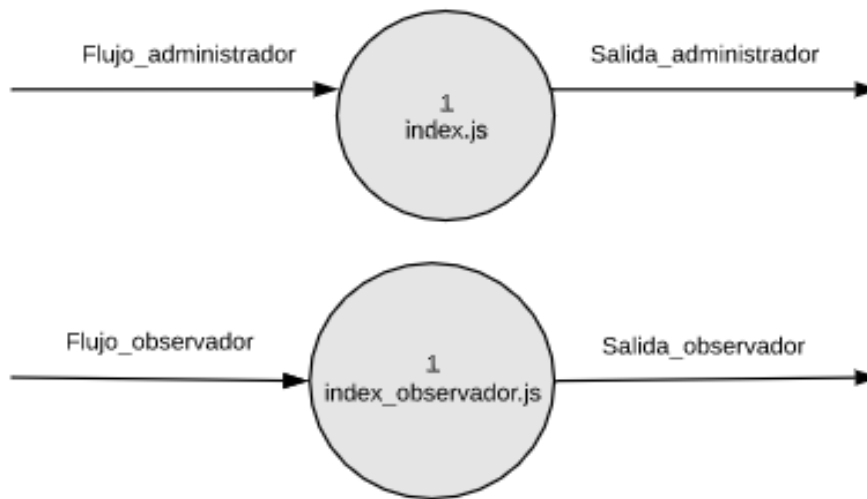
-Subsistema ***index.js***: Este subsistema se encarga de gestionar las peticiones del administrador sobre visualizar varios trabajadores.

-Subsistema ***index_observador.js***: Este subsistema se encarga de gestionar las peticiones del observador sobre visualizar varios trabajadores.

Flujo de datos

- Flujo de datos ***salida_administrador***: Representan los datos que se le mostrarán al administrador, en este caso, tablas con información resumida sobre varios trabajadores.

- Flujo de datos ***salida_observador***: Representan los datos que se le mostrarán al observador, en este caso, tablas con información resumida sobre varios trabajadores.



Nivel 2 de contexto

Subsistemas

-Subsistema ***Hindex.js***: Este subsistema se encarga de realizar las consultas a la base de datos con el objetivo de visualizar todos los registros y ordenarlos en forma de tabla.

Procesos

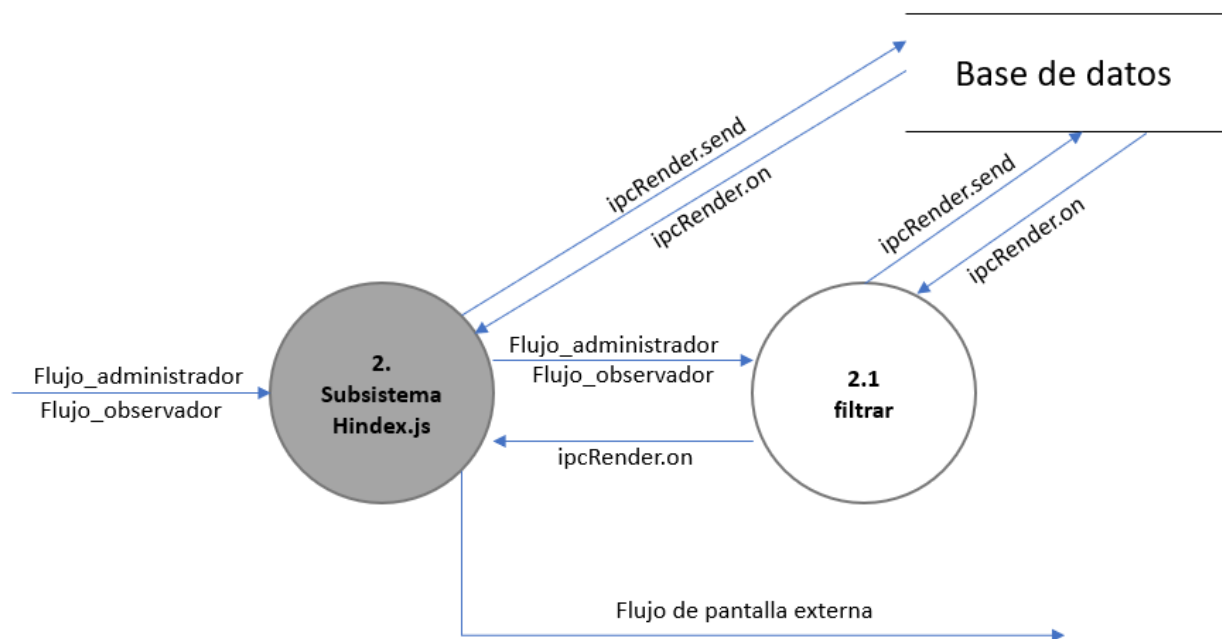
-Proceso ***filtrar***: Cada vez que se ingresa o borra un carácter desde la barra de búsqueda este proceso se encarga de realizar un filtrado en la base de datos considerando el campo completo, mostrando en pantalla todos los trabajadores registrados que coincidan con las búsqueda.

Flujo de datos

- Flujo de datos ***ipcRender.send***: Representa lo que se buscará en la base de datos.
- Flujo de datos ***ipcRender.on***: Representa la totalidad de los trabajadores que coincidan en algún campo con la búsqueda que se haya realizado; o se mostrará la totalidad de trabajadores si no se ha realizado búsqueda.

Almacén

- Almacén **Base de Datos**: Representa la base de datos del Sistema, en donde se buscarán los trabajadores que se mostrarán en pantalla



[Visualizar Trabajador]

El Sistema debe mostrar, por medio de interfaz gráfica, la información detallada sobre uno de los empleados registrado.

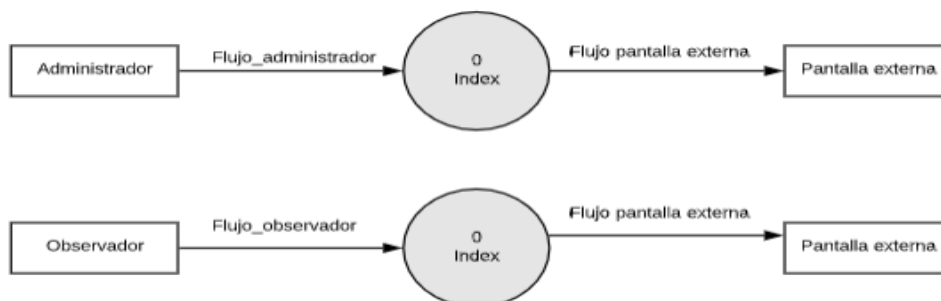
Nivel 0 de contexto

Entidades

- Entidad externa **administrador**: Representa a la persona que aprieta el botón que permite visualizar para ver un trabajador en específico y todos sus detalles.
- Entidad externa **observador**: Representa a la persona que aprieta el botón que permite visualizar para ver un trabajador en específico y todos sus detalles.
- Entidad externa **pantalla del ordenador**: Muestra en pantalla la información del trabajador solicitada.

Flujo de datos

- Flujo de **administrador**: Contiene la solicitud de mostrar el trabajador seleccionado.
- Flujo de **pantalla externa**: Representa la información, enviada por el sistema, del trabajador, en específico, solicitado, mostrado en forma de dos tablas (una con la información del trabajador y otra con sus periodos de trabajo).
- Flujo de **observador**: Contiene la solicitud de mostrar el trabajador seleccionado



Nivel 1 de contexto

Subsistemas

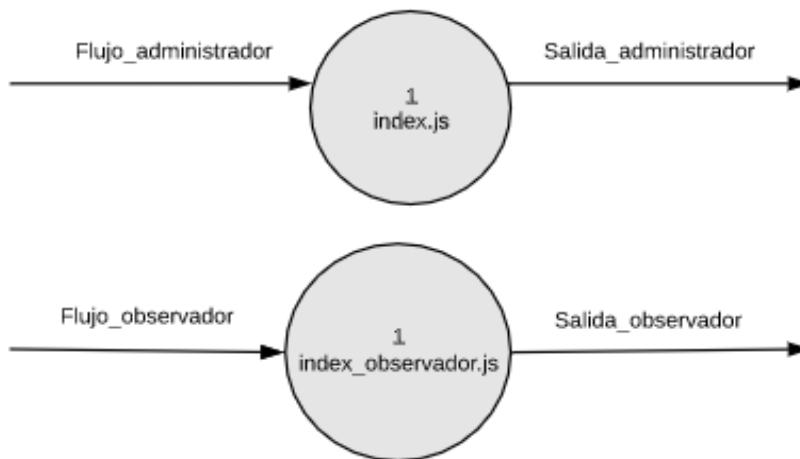
-Subsistema ***index.js***: Este subsistema se encarga de gestionar las peticiones del administrador sobre visualizar un trabajador en específico.

-Subsistema ***index_observador.js***: Este subsistema se encarga de gestionar las peticiones del observador sobre visualizar un trabajador en específico.

Flujo de datos

- Flujo de datos ***salida_administrador***: Representan los datos que se le mostrarán al administrador, en este caso, corresponde a una sección con la información detallada de algún trabajador seleccionado.

- Flujo de datos ***salida_observador***: Representan los datos que se le mostrarán al observador, en este caso, corresponde a una sección con la información detallada de algún trabajador seleccionado.



Nivel 2 de contexto

Subsistemas

-Subsistema ***infotra.js***: Este subsistema se encarga de realizar la búsqueda del trabajador seleccionado en la base de datos, utilizando su id para esto, para posteriormente generar una tabla más detallada que la mostrada en index.js con los

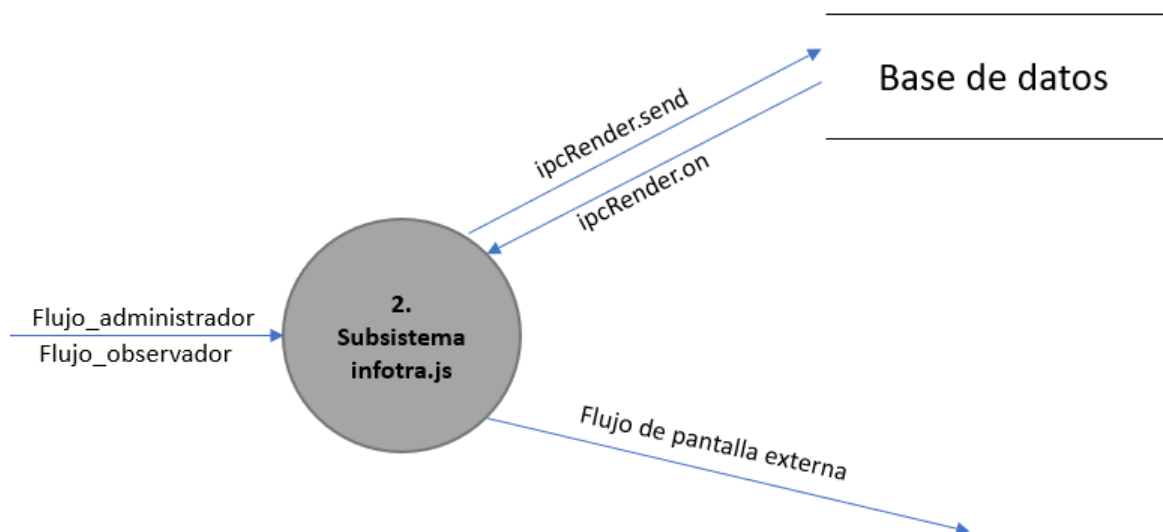
datos obtenidos. Además genera una tabla adicional con todos los periodos que le correspondan al trabajador.

Flujo de datos

- Flujo de datos ***ipcRender.send***: Representa al id del trabajador seleccionado, con el que se realizará la búsqueda en la base de datos.
- Flujo de datos ***ipcRender.on***: Representa un arreglo con todas las filas correspondientes a los distintos periodos de trabajo del trabajador, obtenidos en la base de datos.

Almacén

- Almacén ***Base de Datos***: Representa la base de datos del Sistema, en donde se buscarán los trabajadores que se mostrarán en pantalla



[Ingresar Registro]

El Sistema debe consultar si es que la huella digital del trabajador corresponde a un funcionario dentro de la base de datos e ingresará automáticamente su horario de entrada/salida.

Nivel 0 de contexto

Entidades

- Entidad externa **usuario**: El usuario ingresa su rut en el sistema para validar su ingreso al trabajo (Huella digital aún no implementada).
- Entidad externa **pantalla del ordenador**: Representa la información necesaria para que el usuario conozca los pasos a seguir para ingresar su rut.

Proceso

- Index**: El Sistema se encuentra alerta esperando el ingreso del rut del usuario. Cuando este es ingresado, el sistema busca la información del usuario y almacena un nuevo periodo de trabajo para este, con la hora actual del sistema.

Flujo de datos

- Flujo de **usuario**: Representa el rut del usuario.
- Flujo de **pantalla externa**: Representa la información enviada por el sistema correspondiente a la validación del rut.



Nivel 1 de contexto

Subsistemas

-Subsistema **registrar.js** Este subsistema se encarga de gestionar el ingreso de información correspondiente al usuario que haya ingresado su rut (Sistema con la huella digital aún no implementado) revisando que este se encuentre registrado en la base de datos. En caso de éxito se ingresa un nuevo periodo en la base de datos, registrando este con la hora del sistema al momento que se hizo uso.

Procesos

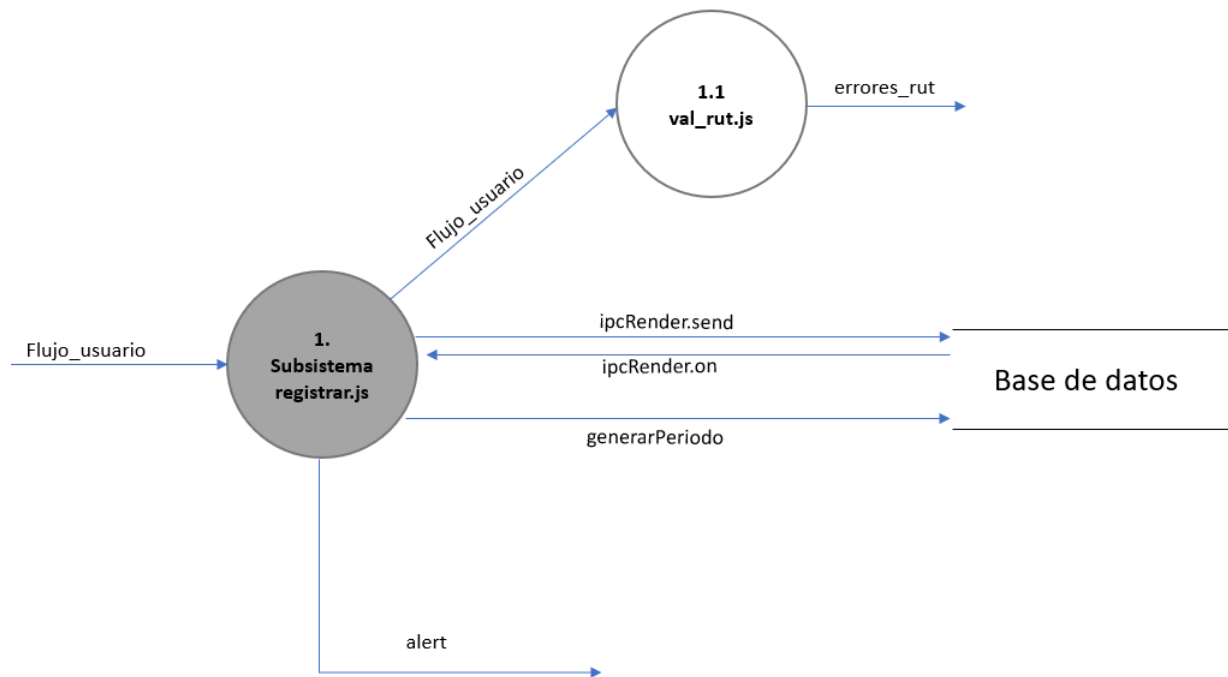
- Proceso **val_rut.js**: Este proceso valida por medio de un algoritmo si es que el rut ingresado es válido.

Flujo de datos

- Flujo de datos **alert**: Muestra en pantalla que se ha ingresado un nuevo registro.
- Flujo de datos **ipcRender.send**: Representa al rut del trabajador que se buscará en la base de datos.
- Flujo de datos **ipcRender.on**: Representa un booleano que nos indicará si el trabajador estaba registrado en la base de datos.
- Flujo de datos **errores_rut**: Muestra un mensaje en pantalla acerca de que el rut ingresado es incorrecto.
- Flujo de datos **generarPeriodo**: Representa el nuevo periodo del usuario que se agregará en la base de datos.

Almacén

- Almacén **Base de Datos**: Representa la base de datos del Sistema, en donde se verificará si el rut se encuentra registrado, y además se almacenará el nuevo periodo del trabajador correspondiente



[Inicio de sesión]

El sistema pide usuario y contraseña al administrador para dar ingreso a una sección del programa exclusiva para este.

Nivel 0 de contexto

Entidades

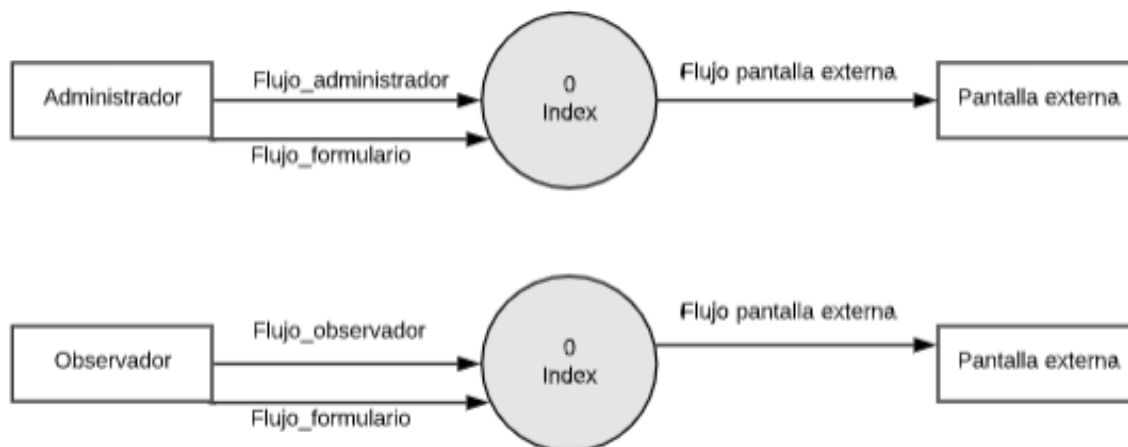
- Entidad externa **administrador**: Representa a la persona que desea ingresar a una sección del programa de uso exclusiva para él, por medio de su identificación.
- Entidad externa **observador**: Representa a la persona que desea ingresar a una sección del programa de uso exclusiva para él, por medio de su identificación.
- Entidad externa **pantalla del ordenador**: Muestra en pantalla las casillas a rellenar en el formulario para el inicio de sesión.

Proceso

- Index**: El Sistema solicita la información al administrador u observador por medio de casillas, donde debe rellenar la información sobre su identificación. Una vez rellenos y enviados, el Sistema debe validar que estos datos corresponden a un ente previamente registrado, y permitirle el acceso a determinadas funciones en caso de que este se valide.

Flujo de datos

- Flujo de **administrador**: Contiene los caracteres ingresados por el administrador en el formulario.
- Flujo de **observador**: Contiene los caracteres ingresados por el observador en el formulario.
- Flujo de **pantalla externa**: En caso de que el formulario esté correcto, muestra la sección exclusiva para el administrador u observador, en caso de que no, muestra un mensaje de error.
- Flujo de datos **flujo_formulario**: Solicita y envía datos que deben ser revisados y validados en la base de datos.



Nivel 1 de contexto

Subsistemas

- Subsistema **login.js**: Este subsistema se encarga de gestionar las solicitudes de inicio de sesión por parte de administradores u observadores.

Procesos

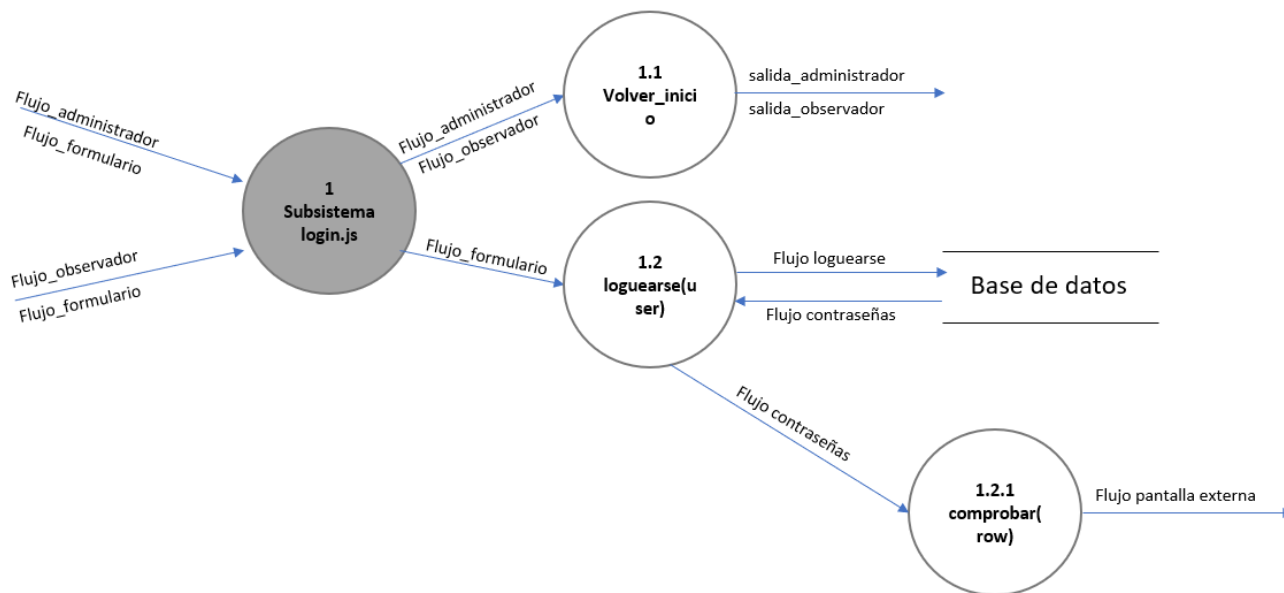
- Proceso **Volver_inicio**: Cierra la ventana de iniciar sesión y regresa a la pantalla principal.
- Proceso **loguearse(user)**: Busca en la base de datos posibles contraseñas que concuerden con el “user” ingresado, retornando las filas en la base de datos que correspondan.
- Proceso **comprobar(row)**: Comprueba si la contraseña ingresada corresponde a algunas de las que se obtuvieron desde la base de datos.

Flujo de datos

- Flujo **loguearse**: Representa el texto de usuario que se buscará en la base de datos por parte del proceso loguearse.
- Flujo **contraseñas**: Representa las contraseñas encontradas en la base de datos que corresponden al usuario buscado.

Almacén

- Almacén **Base de datos**: Representa a la base de datos del Sistema, que será consultado por medio del proceso loguearse.



[Generar Registro]

El sistema es capaz generar un documento en formato pdf con la información de un trabajador solicitado.

Nivel 0 de contexto

Entidades

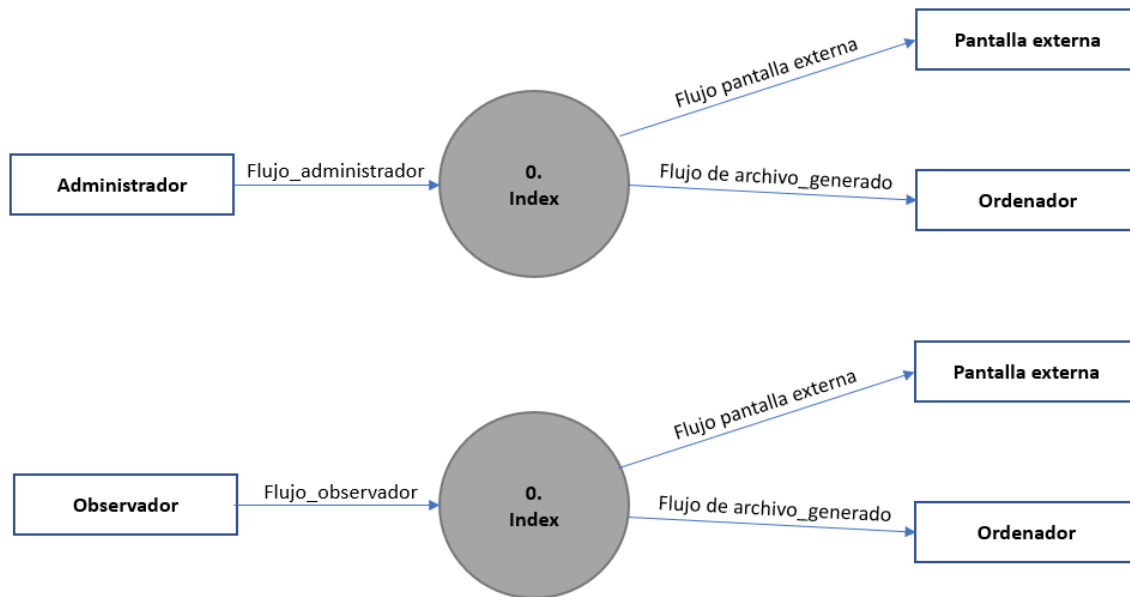
- Entidad externa **administrador**: Representa a la persona que desea contar con el registro de uno de los trabajadores en formato pdf.
- Entidad externa **observador**: Representa a la persona que desea contar con el registro de uno de los trabajadores en formato pdf.
- Entidad externa **pantalla del ordenador**: Muestra la información correspondiente a la solicitud de la búsqueda de una ubicación destino para el archivo pdf.
- Entidad externa **ordenador**: Representa a la ubicación en la que se almacenará el archivo pdf.

Proceso

- Sistema**: El Sistema genera el archivo pdf en base a la información del usuario seleccionado y realiza una consulta al administrador u observador sobre la ubicación de destino que tendrá este archivo.

Flujo de datos

- Flujo de **administrador**: Representa la solicitud del administrador de generar un archivo pdf.
- Flujo de **observador**: Representa la solicitud del observador de generar un archivo pdf.
- Flujo de **archivo_generado**: Representa la información que tendrá el archivo que se generará en la computadora.



Nivel 1 de contexto

Subsistemas

- Subsistema **infotra.js**: Este subsistema se encarga de la gestión de solicitudes de generación de registros.

Procesos

-Proceso **geninforme**: Este proceso se encarga de generar un pdf mediante la lectura de html correspondiente al trabajador que se haya seleccionado previamente en infotra.js.

Flujo de datos

-Flujo **envia_html**: Corresponde al archivo html que se utilizará para la generación del pdf.

