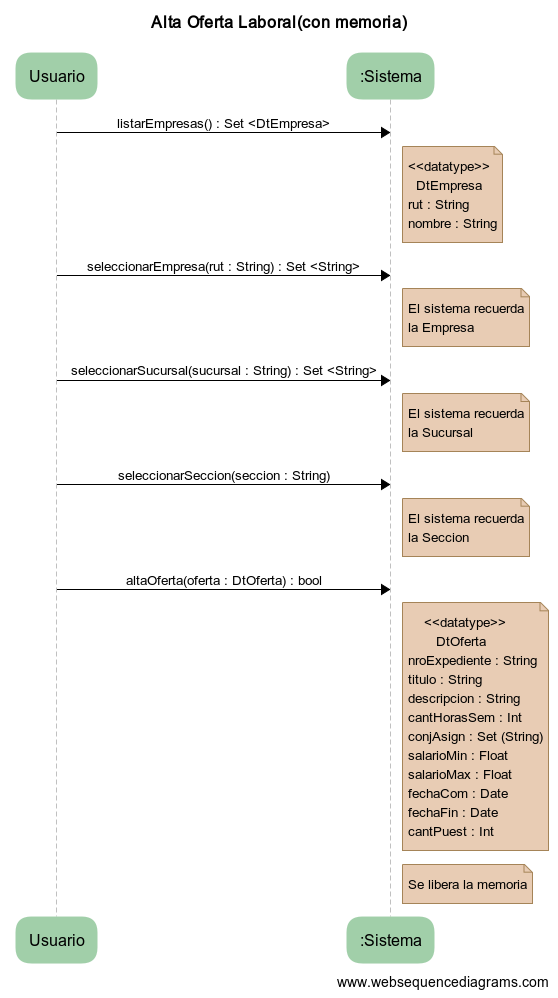
Parte 1) Diagramas de Secuencia de Sistema correspondientes a los casos de uso descritos

anteriormente y especificación de contratos para cada operación.



**listarEmpresas() : Set <DtEmpresa>**

Pre:

-Existe al menos una instancia de Empresa

Post:

-El sistema lista todas las instancias de Empresa e mostrando los datos e.nombre y

e.rut

**seleccionarEmpresa(rutEmpresa : String) : Set <String>**

Pre:

-Existe una instancia de Empresa e con e.rut = 'rutEmpresa'

Post:

-El sistema guarda una referencia a esta instancia de Empresa

-El sistema lista todas las instancias de Sucursal s que tengan un vínculo con la instancia de Empresa con e.rut = 'rutEmpresa' mostrando los datos s.nombre

**seleccionarSucursal(sucursal : String) : Set <String>**

Pre:

-Existe una instancia de Sucursal con el dato s.nombre = 'sucursal' con un vínculo

a la instancia de Empresa anterior

Post:

-El sistema guarda una referencia a esta instancia de Sucursal

-El sistema lista todas las instancias de Sección se que tengan un vínculo con la instancia de Sucursal con s.nombre = 'sucursal' mostrando los datos se.nombre

**seleccionarSeccion(seccion : String)**

Pre:

-Existe una instancia de Sección con el dato se.nombre = 'sección' con un vínculo

a la instancia de Sucursal anterior

Post:

-El sistema guarda una referencia a esta instancia de Sección

**altaOferta(oferta : DtOferta) : bool**

Pre

-Todos los valores de los campos del DtOferta 'oferta' deben ser distintos de null

-Ninguna instancia de Oferta Laboral o debe tener el valor 'nroExpediente' en el

campo o.numero

-’conjAsign’ debe tener al menos un código de instancia de Asignatura y todos deben ser códigos válidos de instancias existentes

-'cantHorasSem' debe ser mayor a 0

-'salMin' debe ser menor o igual a 'salMax' y ambos deben ser mayores a 0

-'fechaCom' debe ser anterior a 'fechaFin' y ambas deben ser mayores a la fecha actual

-'cantPuest' debe ser mayor a 0

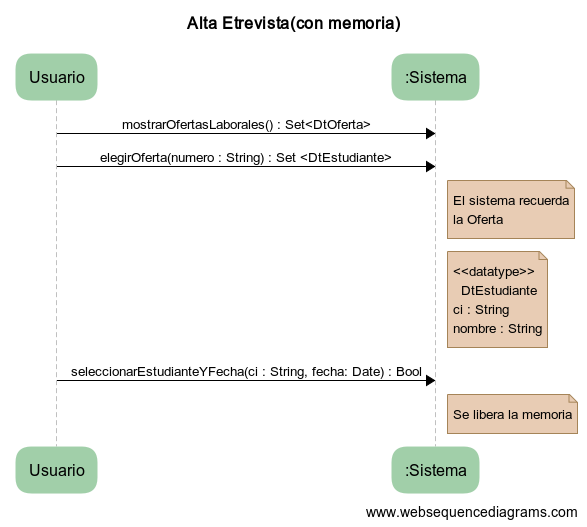
Post:

-El sistema devuelve un booleano siendo true : el alta se efectúa o false : hubo algún error al ingresar datos

-Se crea una instancia de Oferta Laboral asignandosele a los campos o.numero = 'nroExpediente', o.titulo = 'titulo', o.descripcion = 'descripcion', o.hora\_sem = 'cantHorasSem', o.salario\_min = 'salMin', salario\_max = 'salMax', o.fecha\_ini = 'fechaCom', o.fecha\_fin = 'fechaFin', o.puestos = 'cantPuest'

-Se crea una relación entre esta instancia y la respectiva de Sección

-Se crea relaciones entre esta instancia y las respectivas de Asignatura



**mostrarOfertasLaborales() : Set<DtOferta>**

Pre:

-Existe al menos una instancia de Oferta Laboral

Post:

-El sistema lista todas la instancias de Oferta Laboral mostrando un DtOferta

con los datos de cada una

**elegirOferta(numero : String) : Set <DtEstudiante>**

Pre:

-Existe una instancia de Oferta Laboral o con o.codigo = 'numero'

Post:

-El sistema guarda una referencia a esta instancia de Oferta Laboral

-El sistema lista todas las instancias de Estudiante e que tengan un vínculo con la instancia de Oferta Laboral con el campo o.titulo = 'numero' mostrando los campos e.ci y e.nombre

**seleccionarEstudianteYFecha(ci : string, fecha: Date) : bool**

Pre:

-Existe una instancia de Estudiante e con e.ci = 'ci'

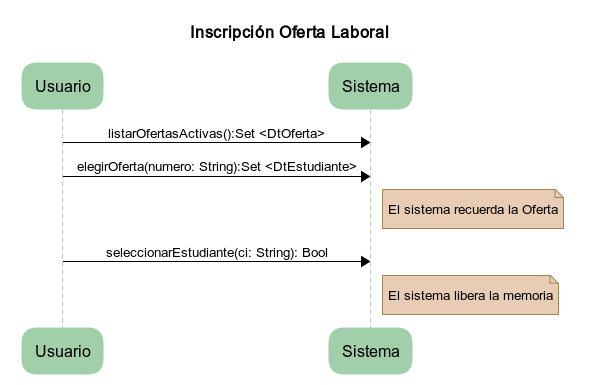
-'fecha' debe ser una fecha válida mayor a la fecha actual

Post:

-Se agrega el nuevo dato 'fecha' al campo "entrevista" en la instancia de Postulacion

que tenga un vínculo con la instancia de Oferta Laboral seleccionada y otro vínculo con la instancia de Estudiante con e.ci = 'ci'

-El sistema devuelve un booleano siendo true : el alta se efectúa o false : hubo algún error al ingresar datos



**listarOfertasActivas():Set<DtOferta>**

Pre:

- Existe al menos una instancia de Oferta Laboral o con o.fecha\_fin > CurrentDate().

Post:

- El sistema lista todas las posibles instancias de Oferta Laboral o con o.fecha\_ini <= CurrentDate() y con o.fecha\_fin > CurrentDate().

**elegirOferta(numero: String):Set < DtEstudiante >**

Pre:

- Debe haber una instancia de Oferta Laboral o con o.numero = ‘numero’.

Post:

- El sistema lista todas las posibles instancias de Estudiantes que no estén inscriptos.

**seleccionarEstudiante(ci: String): Bool**

Pre:

- Debe haber una instancia de Estudiante e con e.ci = ci.

- No debe existir un vínculo entre Estudiante e y Oferta Laboral o.

- La instancia de Estudiante e debe estar vinculada al menos a las instancias de Asignatura vinculadas a la instancia de Oferta Laboral o.

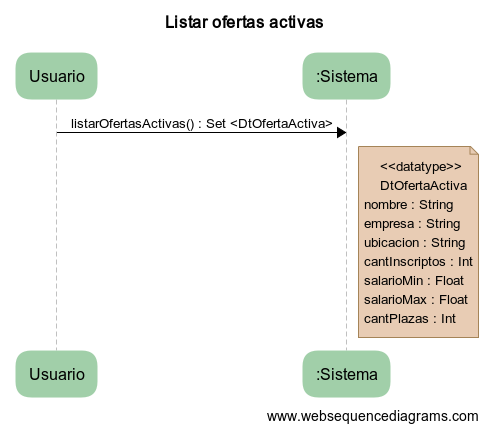
Post:

- Se crea una instancia de Postulacion p con la fecha actual.

- Se crea un vínculo entre la instancia de Postulacion p y la instancia de Oferta Laboral o.

- Se crea un vínculo entre la instancia de Postulacion p y la instancia de Estudiante e

-El sistema devuelve un booleano siendo true : la inscripción se efectúa o false : hubo algún error al ingresar datos.



**listarOfertas() : Set <DtOfertaActiva>**

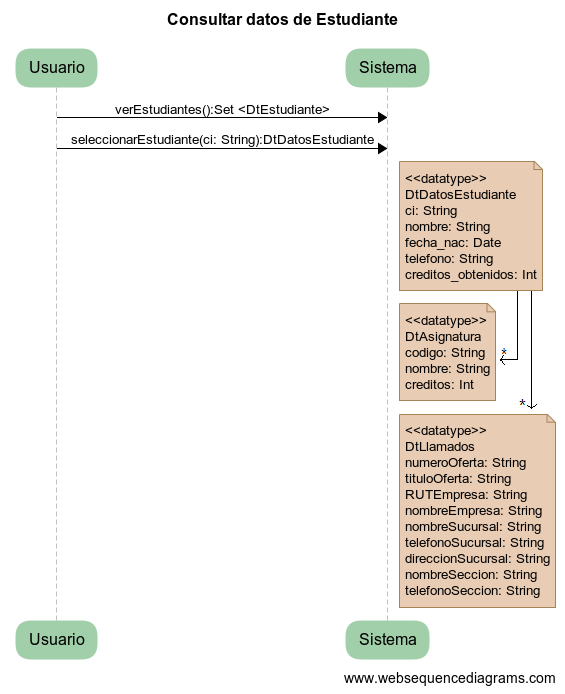
Pre:

-Debe existir al menos una instancia de Oferta Laboral o con o.fecha\_ini <= CurrentDate() y o.fecha\_fin >= CurrentDate()

Post:

-El sistema lista todas las instancias de Oferta Laboral o con o.fecha\_ini <= CurrentDate() y o.fecha\_fin >= CurrentDate() mostrando los datos o.nombre, o.salario\_min, o.salario\_max,

o.puestos, la cantidad de instancias de Estudiante con un vínculo a esta, el dato "direccion" de la instancia de Sucursal vinculada a la instancia de Seccion vinculada a esta, y el dato "nombre" de la instancia de Empresa vinculada a la instancia de Sucursal anterior.



**verEstudiantes():Set <DtEstudiante>**

Pre:

- Existe al menos una instancia de Estudiante en el sistema.

Post:

- El sistema lista todas las instancias de Estudiante e con e.ci = ‘ci’ en el sistema.

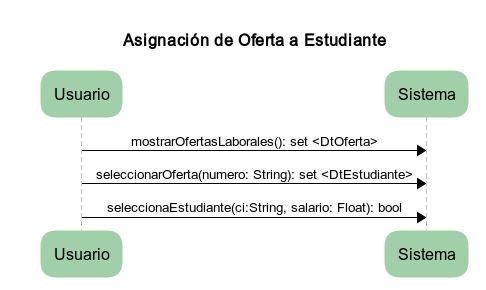
**seleccionarEstudiante(ci: String):DtDatosEstudiante**

Pre:

- Existe una instancia de Estudiante e con e.cedula = ci.

Post:

- El sistema muestra los datos de la instancia Estudiante e con e.ci = ‘ci’, la información de las instancias de Asignatura vinculadas a e, el campo “numero” de las instancias de Oferta Laboral vinculadas y los datos de las instancias de Seccion, Sucursal y Empresa asociados.



**mostrarOfertasLaborales(): set <DTOferta>**

Pre:

- Existe al menos una instancia de Oferta Laboral o.

Post:

- El sistema lista todas las posibles instancias de Ofertas Laborales.

**seleccionarOferta (numero: String): set <DtEstudiante>**

Pre:

- Existe por lo menos una instancia de Oferta Laboral o con o.numero = ‘numero’

Post:

- Se lista todas las instancias de Estudiante vinculada a la instancia Oferta Laboral o.

**seleccionaEstudiante(ci: String, salario: Float): bool**

Pre:

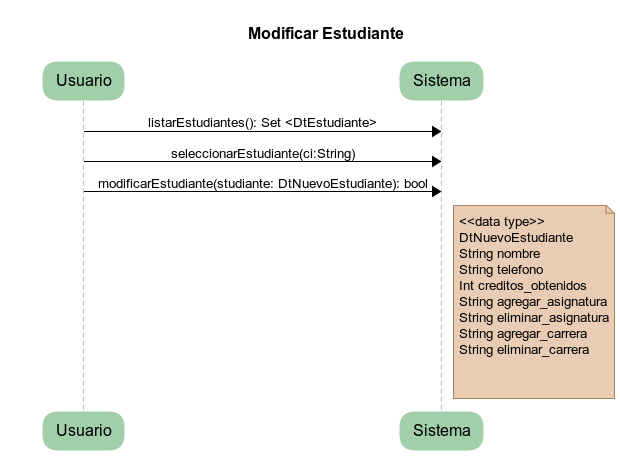
- Existe una instancia de Estudiante e con e.ci = 'ci’ comprendido en el listado anteriormente.

Post:

- Se crea una nueva instancia de Cargo Efectivo ce con ce.fecha= CurrentDate() y el atributo ce.salario = ‘salario’.

- Se crea un vínculo entre la instancia de Estudiante e y la de Cargo efectivo ce.

-El sistema devuelve un booleano siendo true: la inscripción se efectúa o false: hubo algún error al ingresar datos.



**listarEstudiantes():Set <DtEstudiante>**

Pre:

-Existe por lo menos una instancia de Estudiante.

Post:

- El sistema lista todas las posibles instancias de Estudiante.

**seleccionarEstudiante(ci:String)**

Pre:

-Existe una instancia de Estudiante e con e.ci = 'ci’.

Post:

-Guarda una referencia a la instancia de Estudiante e.

**modificarEstudiante(estudiante: DtNuevoEstudiante): bool**

Pre:

-Todos los valores de los campos del DtNuevoEstudiante 'estudiante' a ser modificados deben ser distintos de null.

- ‘creditos\_obtenidos’ debe ser igual a la suma de los créditos de las asignaturas vinculadas.

- ‘agregar\_asignatura’ debe ser el valor del campo “codigo” de una instancia de Asignatura existente y no vinculada a la instancia de Estudiante e.

- ‘eliminar\_asignatura’ debe ser el valor del campo “codigo” de una instancia de Asignatura existente y vinculada a la instancia de Estudiante e.

- ‘agregar\_carrera’ debe ser el campo “codigo” de una instancia de Carrera existente y no vinculada a la instancia de Estudiante e.

- ‘eliminar\_carrera’ debe ser el campo “codigo” de una instancia de Carrera existente y vinculada a la instancia de Estudiante e.

Post:

-En caso de ser e.nombre, e.telefono o creditos\_obtenidos alguno distinto de null se modifica el campo correspondiente en la instancia de Estudiante e.

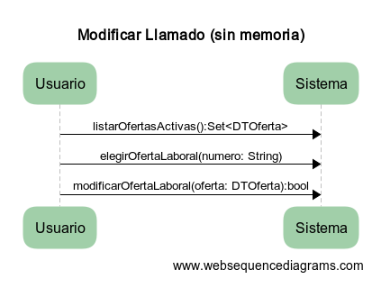
-En caso de ser ‘agregar\_carrera’ distinto de null, se crea un vínculo entre la instancia de Estudiante e y la instancia de Carrera c con c.codigo = ‘agregar\_carrera’.

-En caso de ser ‘eliminar\_carrera’ distinto de null, se desvincula la instancia Estudiante e de la instancia de Carrera c con c.codigo = ‘eliminar\_carrera’.

-En caso de ser ‘eliminar\_asignatura’ distinto de null, se desvincula la instancia de Estudiante e de la instancia de Asignatura a con a.codigo = ‘eliminar\_asignatura’.

-En caso de ser ‘agregar\_asignatura’ distinto de null, se crea un vínculo entre la instancia de Estudiante e y la instancia de Asignatura a con a.codigo = ‘agregar\_asignatura’.

-El sistema devuelve un booleano siendo true: la modificación se efectúa o false: hubo algún error al ingresar datos.



**listarOfertasActivas():Set<DTOferta>**

Pre:

-Debe existir al menos una instancia de Oferta Laboral

Post:

- El sistema lista todas las posibles instancias de Oferta Laboral o con o.fecha\_ini <= CurrentDate() y o.fecha\_fin >= CurrentDate()

**elegirOfertaLaboral(numero: String)**

Pre:

-Debe existir al menos una Oferta Laboral o con o.numero = ‘numero’.

Post:

-El sistema guarda una referencia de esta instancia de Oferta Laboral.

**modificarOfertaLaboral(oferta: DTOferta) : bool**

Pre:

-Todos los valores de los campos del DtOferta 'oferta' a ser modificados deben ser distintos de null.

- El campo ‘nroExpediente’ debe mantenerse igual.

-’conjAsign’ debe tener al menos un código de instancia de Asignatura y todos deben ser códigos válidos de instancias existentes.

-'cantHorasSem' debe ser mayor a 0.

-'salMin' debe ser menor o igual a 'salMax' y ambos deben ser mayores a 0.

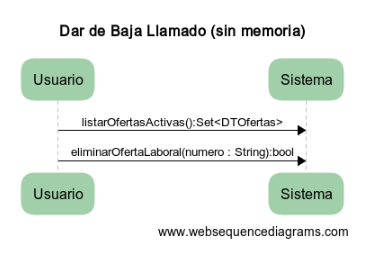
-'fechaCom' debe ser anterior a 'fechaFin' y ambas deben ser mayores a la fecha actual.

-'cantPuest' debe ser mayor a 0.

Post:

-Se cambia los datos de la instancia de Oferta Laboral o por los datos contenidos en el DtOferta ‘oferta’ distintos de null.

- El sistema devuelve un bool == True indicando que se realiza la modificación o bool == False indicando que hubo un error.



**listarOfertasActivas():Set<DTOferta>**

Pre:

- Debe existir al menos una instancia de Oferta Laboral

Post:

-El sistema devuelve al menos un DTOferta (o una colección de estos) con los datos de la/las oferta/s laboral/es activa/s.

**eliminarOfertaLaboral(numero: String):bool**

Pre:

-Debe existir la Oferta Laboral o con o.numero = ‘numero’

Post:

-El sistema elimina la Oferta Laboral con o y devuelve bool == True confirmando la eliminación o bool == False indicando que hubo un error