

Evaluación de Calidad de Servicio

Contract for each method

Versión 1.0

10/05/2013
Santo Domingo
República Dominicana

Historia del Documento		
Versión No.	Fecha, dd/mm/aaaa	Autor(es)
1.0	10/05/2013	Aneudy Labour

Responsables		
Nombre	Titulo	Medios para contacto
Felix Ferreira	Patrocinador	ferreiras54@yahoo.es; Móvil xxx-xxx-xxxx
Aneudy Labour	Autor	aneudylab@gmail.com; Móvil 829-441-8800

Contract Name (method):	+IniciarSesion(nombreUsr, clave)
Class:	ControladorSesiones
Use Case:	CU001: Iniciar Sesión
Responsibilities:	inicia una sesión con los datos del usuario que tenga la combinación de nombre de usuario y clave introducidas
Exceptions:	Si la combinación de nombre de usuario y clave no existe presenta un mensaje de error
Preconditions:	Debe existir un usuario con la combinación de nombre de usuario y clave introducidas en el sistema
Postconditions:	Si el usuario existe en una instancia de usuario fue creada.

Contract Name (method):	+LeerUsuario(unUsr)
Class:	UsuarioDA
Use Case:	CU001: Iniciar Sesión
Responsibilities:	Leer un usuario de la base de datos a partir de su clave y nombre de usuario
Exceptions:	Si el usuario no existe se se destruye la instancia pasada como argumento
Preconditions:	Debe existir un usuario con la combinación de nombre de usuario y clave introducidas en el sistema
Postconditions:	Si el usuario existe en una instancia de usuario fue creada.

Contract Name (method):	+CrearNuevaPlantilla()
Class:	ControladorPlantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Crear una nueva plantilla de evaluación
Exceptions:	
Preconditions:	la sesión abierta en el sistema debe ser un usuario con el rol de administrador
Postconditions:	Una nueva instancia de plantilla fue creada.

Contract Name (method):	+ValidarEsAdmin(Usuario) : Boolean
Class:	ControladorPlantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Valida que el rol de un usuario sea administrador
Exceptions:	Si se pasa una instancia de usuario null retnará false
Preconditions:	La instancia de usuario pasada no debe ser null
Postconditions:	Si el rol de usuario es administrador, true fue retornado. De lo contrario false.

Contract Name (method):	+CrearPuntoEvaluacion(descripcion)
Class:	ControladorPlantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Insertar un punto de evaluación en la plantilla de evaluación que se está creando
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla.
Postconditions:	Una nueva instancia de punto de evaluación fue creada en la plantilla en proceso de creación.

Contract Name (method):	+CrearPuntoEvaluacion(descripcion)
Class:	ControladorPlantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Insertar un punto de evaluación en la plantilla de evaluación que se está creando
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla.
Postconditions:	Una nueva instancia de punto de evaluación fue creada en la plantilla en proceso de creación.

Contract Name (method):	+GuardarPlantilla() : int
Class:	ControladorPlantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Guardar la plantilla de evaluación con sus respectivos puntos de evaluación y devolver el id generado para la misma.
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla.
Postconditions:	La plantilla de evaluación fue guardada.

Contract Name (method):	+GuardarPunto(idPlantilla)
Class:	PuntoEvaluacion
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Guardar un punto de evaluación para la plantilla creada.
Exceptions:	
Preconditions:	El id de la plantilla debe existir en la base de datos
Postconditions:	El punto de evaluación fue guardado

Contract Name (method):	+AgregarPunto(descripcion)
Class:	Plantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Agrega un punto de evaluación a una plantilla de evaluación.
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla de evaluación.
Postconditions:	una instancia de punto de evaluación fue creada.

Contract Name (method):	+AgregarPunto(descripcion)
Class:	Plantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Agrega un punto de evaluación a una plantilla de evaluación.
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla de evaluación.
Postconditions:	una instancia de punto de evaluación fue creada.

Contract Name (method):	+Guardar()
Class:	Plantilla
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Guardar la plantiilla de evaluación en la base de datos
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla.
Postconditions:	La plantilla de evaluación fue guardada.

Contract Name (method):	+GuardarPunto(Punto, idPlantilla)
Class:	PuntoEvaluacionDA
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Guardar un punto de evaluación en la base de datos.
Exceptions:	
Preconditions:	El id de la plantilla debe existir en la base de datos
Postconditions:	El punto de evaluación fue guardado

Contract Name (method):	+Guardar(Plantilla)
Class:	PlantillaDA
Use Case:	CU002: Crear Plantilla de Evaluación
Responsibilities:	Guardar la plantiilla de evaluación en la base de datos
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla.
Postconditions:	La plantilla de evaluación fue guardada.

Contract Name (method):	+VerRepresentantes() : List<Representante>
Class:	ControladorEvaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	recupera la lista de representantes que se reportan al supervisor que tiene la sesión activa en el sistema
Exceptions:	
Preconditions:	el usuario activo debe ser un supervisor y tener personal que se reporte a él
Postconditions:	la lista de representantes que se reportan al supervisor fue retornada

Contract Name (method):	+CrearEvaluacion(RepresentanteID) : Evaluacion
Class:	ControladorEvaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	Crea una instancia de evaluación para evaluar a un representante.
Exceptions:	
Preconditions:	el representante evaluado se debe reportar al supervisor que genera la evaluación.
Postconditions:	una nueva instancia de evaluación fue creada.

Contract Name (method):	+EvaluarPunto(idPunto, cumplePunto)
Class:	ControladorEvaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	Crea una respuesta que contesta si cumple o no un punto de evaluación.
Exceptions:	
Preconditions:	Debe existir la instancia de la evaluación
Postconditions:	Una instancia de respuesta fue creada

Contract Name (method):	+ValidarEsSupervisor(Usr) : boolean
Class:	ControladorEvaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	Valida que el rol de un usuario sea supervisor
Exceptions:	Si se pasa una instancia de usuario null retnará false
Preconditions:	La instancia de usuario pasada no debe ser null
Postconditions:	Si el rol de usuario es supervisor true fue retornado. De lo contrario false.

Contract Name (method):	+GuardarEvaluacion()
Class:	ControladorEvaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	Guardar la evaluación con sus respectivas respuestas.
Exceptions:	
Preconditions:	Debe existir la instancia de evaluación
Postconditions:	La evaluación fue guardada.

Contract Name (method):	+getListaRepresentantes() : List<Representante>
Class:	Supervisor
Use Case:	CU003: Evaluar Representante
Responsibilities:	recupera la lista de representantes que se reportan al supervisor que tiene la sesión activa en el sistema
Exceptions:	
Preconditions:	El supervisor debe tener personal que se reporte a él
Postconditions:	la lista de representantes que se reportan al supervisor fue retornada

Contract Name (method):	+BuscarRepresentantes(supervisor) : List<Representante>
Class:	RepresentanteDA
Use Case:	CU003: Evaluar Representante
Responsibilities:	recupera la lista de representantes que se reportan al supervisor que tiene la sesión activa en el sistema
Exceptions:	
Preconditions:	El supervisor debe tener personal que se reporte a él
Postconditions:	la lista de representantes que se reportan al supervisor fue retornada

Contract Name (method):	+AgregarPunto(idPunto, descripcion)
Class:	Plantilla
Use Case:	CU003: Evaluar Representante
Responsibilities:	Agrega un punto de evaluación a una plantilla de evaluación.
Exceptions:	
Preconditions:	Debe existir una instancia de plantilla de evaluación.
Postconditions:	una instancia de punto de evaluación fue creada.

Contract Name (method):	+BuscarPlantillaActiva() : Plantilla
Class:	PlantillaDA
Use Case:	CU003: Evaluar Representante
Responsibilities:	Buscar la plantilla activa en la base de datos
Exceptions:	
Preconditions:	Debe existir por lo menos un registro de plantilla en la bd
Postconditions:	una instancia de plantilla fue creada

Contract Name (method):	+BuscarPuntosEvaluacion(Plantilla)
Class:	PuntoEvaluacionDA
Use Case:	CU003: Evaluar Representante
Responsibilities:	Buscar los puntos de evaluación que se corresponden con la plantilla esepificada
Exceptions:	
Preconditions:	El id de la plantilla buscada debe existir en la base de datos
Postconditions:	una instancia de punto de evaluación fue creada por cada registro recuperado.

Contract Name (method):	+AgregarRespuesta(idPto, cumple)
Class:	Evaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	Crea una respuesta que contesta si cumple o no un punto de evaluación.
Exceptions:	
Preconditions:	Debe existir la instancia de la evaluación
Postconditions:	Una instancia de respuesta fue creada

Contract Name (method):	+Guardar()
Class:	Evaluacion
Use Case:	CU003: Evaluar Representante
Responsibilities:	Guardar la evaluación con sus respectivas respuestas.
Exceptions:	
Preconditions:	Debe existir la instancia de evaluación
Postconditions:	La evaluación fue guardada.

Contract Name (method):	+Guardar(Evaluacion)
Class:	EvaluacionDA
Use Case:	CU003: Evaluar Representante
Responsibilities:	Guardar la evaluación en la bd
Exceptions:	
Preconditions:	Debe existir la instancia de evaluación
Postconditions:	La evaluación fue guardada.

Contract Name (method):	+Guardar(idEvaluacion, Respuesta)
Class:	RespuestaDA
Use Case:	CU003: Evaluar Representante
Responsibilities:	Guardar la respuesta en la bd
Exceptions:	
Preconditions:	el id de la evaluación debe existir en la bd
Postconditions:	La evaluación fue guardada.

Contract Name (method):	+GuardarRespuesta(EvaluacionID)
Class:	Respuesta
Use Case:	CU003: Evaluar Representante
Responsibilities:	Guardar la respuesta en la bd
Exceptions:	
Preconditions:	el id de la evaluación debe existir en la bd
Postconditions:	La respuesta fue guardada.

Contract Name (method):	+VerEvaluacionesPendientes() : List<Evaluacion>
Class:	ControladorEvaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	recupera la lista de evaluaciones pendientes por revisar que tiene el representante
Exceptions:	
Preconditions:	el usuario activo debe ser un representante y tener evaluacions pendientes por revisar
Postconditions:	la lista de pendientes por revisar fue retornada

Contract Name (method):	+ValidarEsRepresentante(Usuario) : boolean
Class:	ControladorEvaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Validar que el rol de un usuario sea representante
Exceptions:	Si se pasa una instancia de usuario null retnará false
Preconditions:	La instancia de usuario pasada no debe ser null
Postconditions:	Si el rol de usuario es representante true fue retornado. De lo contrario false.

Contract Name (method):	+RevisarEvaluacion(evaluacionID)
Class:	ControladorEvaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Buscar los detalles de la evaluación seleccionada en la bd
Exceptions:	
Preconditions:	el id de la evaluación seleccionada debe existir
Postconditions:	Una instancia de evaluación fue creada

Contract Name (method):	+ReclamarRespuesta(puntoID, comentario)
Class:	ControladorEvaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	reclamar una respuesta y agregarle un comentario con la razón de la reclamación
Exceptions:	
Preconditions:	Debe existir una respuesta para el punto de evaluación que se reclama
Postconditions:	una respuesta fue actualizada

Contract Name (method):	+TerminarRevision()
Class:	ControladorEvaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Asignar el estado de reclamada a la evaluación
Exceptions:	
Preconditions:	Debe existir una instancia de evaluación
Postconditions:	una evaluación fue actualizada.

Contract Name (method):	+BuscarPlantilla(Evaluacion)
Class:	PlantillaDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Busca la plantilla que le corresponde a la evaluación seleccionada y se la asigna
Exceptions:	
Preconditions:	Deben existir la evaluación y la plantilla de esta
Postconditions:	una instancia de plantilla fue creada

Contract Name (method):	+ReclamarRespuesta(idPunto, comentario)
Class:	Evaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	reclamar una respuesta y agregarle un comentario con la razón de la reclamación
Exceptions:	
Preconditions:	Debe existir una respuesta para el punto de evaluación que se reclama
Postconditions:	una respuesta fue actualizada

Contract Name (method):	+ActualizarEstado()
Class:	Evaluacion
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Verificar si la evaluación tiene respuestas reclamadas para actualizar su estado a reclamada o terminada según sea el caso.
Exceptions:	
Preconditions:	Debe existir la instancia de la evaluación a actualizar.
Postconditions:	El estado de la evaluación fue actualizado.

Contract Name (method):	+ObtenerPuntosEvaluacion(Plantilla)
Class:	PuntoEvaluacionDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Recuperar los puntos de evaluación de una plantilla de evaluación
Exceptions:	
Preconditions:	Debe existir la plantilla en la bd
Postconditions:	Una instancia de punto de evaluación fue creada por cada registro recuperado.

Contract Name (method):	+BuscarRespuestas(Evaluacion)
Class:	RespuestaDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Recuperar los puntos las respuestas de una evaluación
Exceptions:	
Preconditions:	Debe existir la evaluación en la bd
Postconditions:	Una instancia de respuesta fue creada por cada registro recuperado.

Contract Name (method):	+ReclamarRespuesta(evalID, puntoID, comentario)
Class:	RespuestaDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Actualizar el estado de una respuesta en la bd a reclamada y su respectivo comentario
Exceptions:	
Preconditions:	Debe existir la respuesta en la bd
Postconditions:	Una instancia de respuesta fue creada por cada registro recuperado.

Contract Name (method):	+VerEvaluacionesPendientes() : List<Evaluacion>
Class:	EvalaucionDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	recupera la lista de evaluaciones pendientes por revisar que tiene el representante
Exceptions:	
Preconditions:	el usuario activo debe ser un representante y tener evaluacions pendientes por revisar
Postconditions:	la lista de pendientes por revisar fue retornada

Contract Name (method):	+BuscarEvaluacion(EvaluacionID) : Evaluacion
Class:	EvalaucionDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Buscar los detalles de la evaluación seleccionada en la bd
Exceptions:	
Preconditions:	el id de la evaluación seleccionada debe existir
Postconditions:	Una instancia de evaluación fue creada

Contract Name (method):	+ActualizarEstado(Evaluacion)
Class:	EvalaucionDA
Use Case:	CU004: Revisar Evaluación
Responsibilities:	Actualiza el estado de la evaluación en la bd.
Exceptions:	
Preconditions:	La evaluación a actuaizar debe existir.
Postconditions:	El estado de la evaluación fue actualizado.

Contract Name (method):	+VerEvaluacionesReclamadas() : List<Evaluacion>
Class:	ControladorEvaluacion
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	recupera la lista de evaluaciones reclamadas por por los respresentantes que se reportan al supervisor que tiene la sesión activa
Exceptions:	
Preconditions:	el usuario activo debe ser un supervisor y tener personal que se reporte a él.
Postconditions:	la lista de evaluaciones reclamadas

Contract Name (method):	+RevisarEvaluacion(idEval)
Class:	ControladorEvaluacion
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Buscar los detalles de la evaluación seleccionada en la bd
Exceptions:	
Preconditions:	el id de la evaluación seleccionada debe existir
Postconditions:	Una instancia de evaluación fue creada

Contract Name (method):	+ResponderReclamacion(puntoID, procede)
Class:	ControladorEvaluacion
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Actualiza el estado del punto de evaluación especificado según proceda o no la reclamación
Exceptions:	
Preconditions:	Debe existir una respuesta para el punto de evaluación reclamado.
Postconditions:	Si la reclamación procede fue actualizada la respuesta a lo opuesto que se había especificado. Si no procede se mantiene como está. En ambos casos la respuesta fue actualizada a no estar reclamada.

Contract Name (method):	+TerminarActualizacion()
Class:	ControladorEvaluacion
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Actualiza el estado de la evaluación a finalizada
Exceptions:	
Preconditions:	la actualización debe existir en la bd.
Postconditions:	El estado de la evaluación fue actualizado a finalizada.

Contract Name (method):	+ResponderReclamacion(puntoID, procede)
Class:	Evaluacion
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Actualiza el estado del punto de evaluación especificado según proceda o no la reclamación
Exceptions:	
Preconditions:	Debe existir una respuesta para el punto de evaluación reclamado.
Postconditions:	Si la reclamación procede fue actualizada la respuesta a lo opuesto que se había especificado. Si no procede se mantiene como está. En ambos casos la respuesta fue actualizada a no estar reclamada.

Contract Name (method):	+FinalizarEvaluacion()
Class:	Evaluacion
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Actualiza el estado de la evaluación a finalizada
Exceptions:	
Preconditions:	la actualización debe existir en la bd.
Postconditions:	El estado de la evaluación fue actualizado a finalizada.

Contract Name (method):	+ResponderReclamacion(procede)
Class:	Respuesta
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Actualiza el estado del punto de evaluación especificado según proceda o no la reclamación
Exceptions:	
Preconditions:	Debe existir una respuesta para el punto de evaluación reclamado.
Postconditions:	Si la reclamación procede fue actualizada la respuesta a lo opuesto que se había especificado. Si no procede se mantiene como está. En ambos casos la respuesta fue actualizada a no estar reclamada.

Contract Name (method):	+ResponderReclamacion(evalID, puntoID, cumple)
Class:	RespuestaDA
Use Case:	CU005: Actualizar Evaluación
Responsibilities:	Actualiza el estado del punto de evaluación especificado según proceda o no la reclamación
Exceptions:	
Preconditions:	Debe existir una respuesta para el punto de evaluación reclamado.
Postconditions:	Si la reclamación procede fue actualizada la respuesta a lo opuesto que se había especificado. Si no procede se mantiene como está. En ambos casos la respuesta fue actualizada a no estar reclamada.

Contract Name (method):	+CrearUsuario(nombre, rol, clave, nombreUsuario, idSupervisor)
Class:	controladorSesiones
Use Case:	CU006: Crear Usuario
Responsibilities:	Crear un usuario en la base de datos con la información introducida por el administrador.
Exceptions:	
Preconditions:	el usuario activo debe ser un administrador
Postconditions:	Una instancia de usuario fue creada.

Contract Name (method):	+Guardar()
Class:	Usuario
Use Case:	CU006: Crear Usuario
Responsibilities:	Crear un usuario en la base de datos con la información introducida por el administrador.
Exceptions:	
Preconditions:	el usuario activo debe ser un administrador
Postconditions:	El usuario fue guardado en la base de datos

Contract Name (method):	+CrearUsuario(nombre, rol, clave, nombreUsuario, idSupervisor)
Class:	Administrador
Use Case:	CU006: Crear Usuario
Responsibilities:	Crear un usuario en la base de datos con la información introducida por el administrador.
Exceptions:	
Preconditions:	el usuario activo debe ser un administrador
Postconditions:	Una instancia de usuario fue creada.

