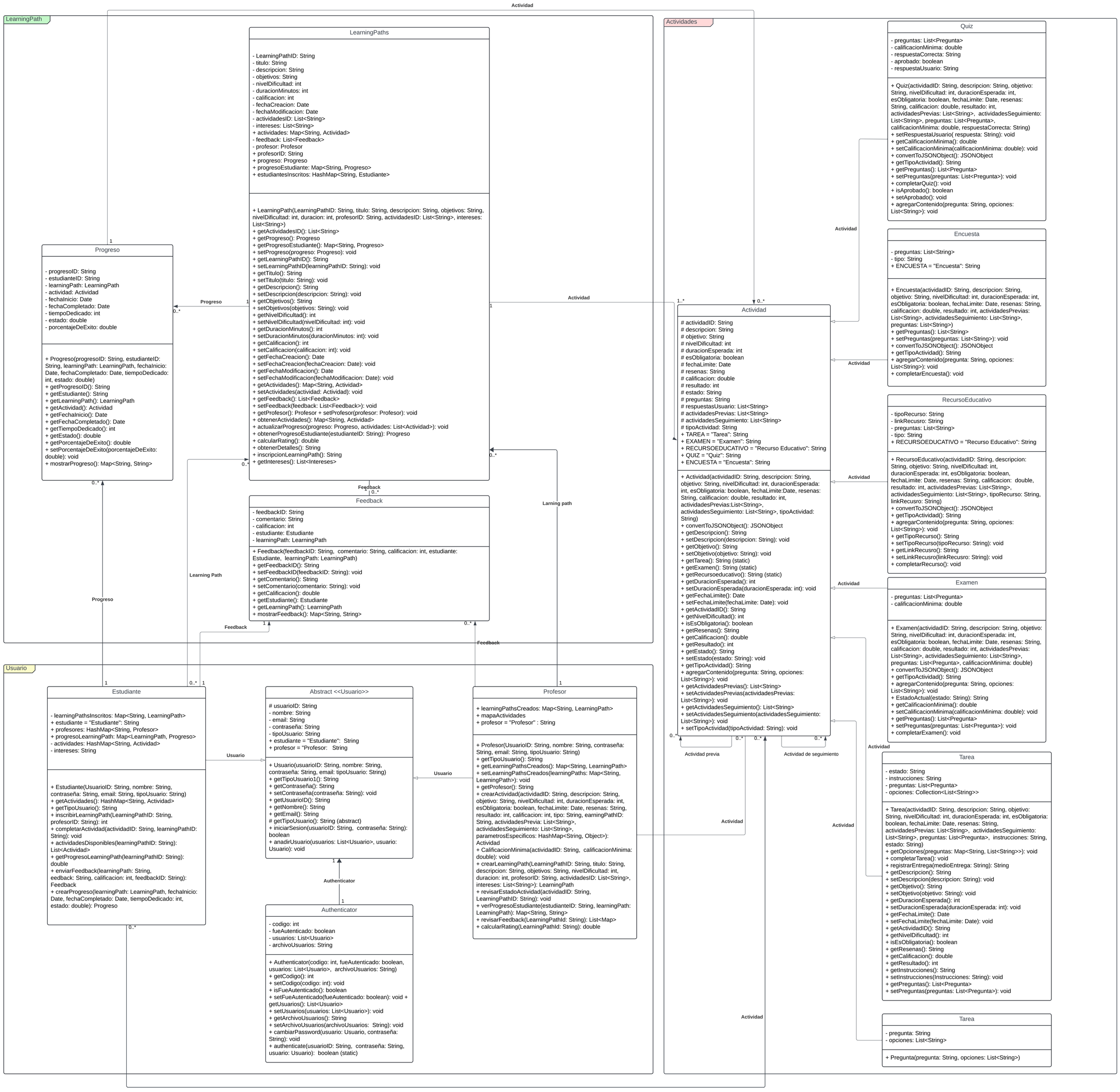
LearningPath
LearningPaths
- LearningPathID: String
- titulo: String
- descripcion: String
- objetivos: String
- nivelDificultad: int
- duracionMinutos: int
- calificacion: int
- fechaCreacion: Date
- fechaModificacion: Date
- actividadesID: List<String>
- intereses: List<String>
+ actividades: Map<String, Actividad>
- feedback: List<Feedback>
- profesor: Profesor
+ profesorID: String
+ progreso: Progreso
+ progresoEstudiante: Map<String, Progreso>
+ estudiantesInscritos: HashMap<String, Estudiante>
+ LearningPath(LearningPathID: String, titulo: String, descripcion: String, objetivos: String, nivelDificultad: int, duracion: int, profesorID: String, actividadesID: List<String>, intereses: List<String>)
+ getActividadesID(): List<String>
+ getProgreso(): Progreso
+ getProgresoEstudiante(): Map<String, Progreso>
+ setProgreso(progreso: Progreso): void
+ getLearningPathID(): String
+ setLearningPathID(learningPathID: String): void
+ getTitulo(): String
+ setTitulo(titulo: String): void
+ getDescripcion(): String
+ setDescripcion(descripcion: String): void
+ getObjetivos(): String
+ setObjetivos(objetivos: String): void
+ getNivelDificultad(): int
+ setNivelDificultad(nivelDificultad: int): void
+ getDuracionMinutos(): int
+ setDuracionMinutos(duracionMinutos: int): void
+ getCalificacion(): int
+ setCalificacion(calificacion: int): void
+ getFechaCreacion(): Date
+ setFechaCreacion(fechaCreacion: Date): void
+ getFechaModificacion(): Date
+ setFechaModificacion(fechaModificacion: Date): void
+ getActividades(): Map<String, Actividad>
+ setActividades(actividad: Actividad): void
+ getFeedback(): List<Feedback>
+ setFeedback(feedback: List<Feedback>): void
+ getProfesor(): Profesor + setProfesor(profesor: Profesor): void
+ obtenerActividades(): Map<String, Actividad>
+ actualizarProgreso(progreso: Progreso, actividades: List<Actividad>): void
+ obtenerProgresoEstudiante(estudianteID: String): Progreso
+ calcularRating(): double
+ obtenerDetalles(): String
+ inscripcionLearningPath(): String
+ getIntereses(): List<Intereses>
Progreso
- progresoID: String
- estudianteID: String
- learningPath: LearningPath
- actividad: Actividad
- fechaInicio: Date
- fechaCompletado: Date
- tiempoDedicado: int
- estado: double
- porcentajeDeExito: double
+ Progreso(progresoID: String, estudianteID: String, learningPath: LearningPath, fechaInicio: Date, fechaCompletado: Date, tiempoDedicado: int, estado: double)
+ getProgresoID(): String
+ getEstudiante(): String
+ getLearningPath(): LearningPath
+ getActividad(): Actividad
+ getFechaInicio(): Date
+ getFechaCompletado(): Date
+ getTiempoDedicado(): int
+ getEstado(): double
+ getPorcentajeDeExito(): double
+ setPorcentajeDeExito(porcentajeDeExito: double): void
+ mostrarProgreso(): Map<String, String>
Feedback
- feedbackID: String
- comentario: String
- calificacion: int
- estudiante: Estudiante
- learningPath: LearningPath
+ Feedback(feedbackID: String, comentario: String, calificacion: int, estudiante: Estudiante, learningPath: LearningPath)
+ getFeedbackID(): String
+ setFeedbackID(feedbackID: String): void
+ getComentario(): String
+ setComentario(comentario: String): void
+ getCalificacion(): double
+ getEstudiante(): Estudiante
+ getLearningPath(): LearningPath
+ mostrarFeedback(): Map<String, String>
Actividad
Progreso
Feedback
Learning Path
Larning path
Usuario
Estudiante
- learningPathsInscritos: Map<String, LearningPath>
+ estudiante = "Estudiante": String
+ profesores: HashMap<String, Profesor>
+ progresoLearningPath: Map<LearningPath, Progreso>
- actividades: HashMap<String, Actividad>
- intereses: String
+ Estudiante(UsuarioID: String, nombre: String, contraseña: String, email: String, tipoUsuario: String)
+ getActividades(): HashMap<String, Actividad>
+ getTipoUsuario(): String
+ inscribirLearningPath(LearningPathID: String, profesorID: String): int
+ completarActividad(actividadID: String, learningPathID: String): void
+ actividadesDisponibles(learningPathID: String): List<Actividad>
+ getProgresoLearningPath(learningPathID: String): double
+ enviarFeedback(learningPath: String, eedback: String, calificacion: int, feedbackID: String): Feedback
+ crearProgreso(learningPath: LearningPath, fechaInicio: Date, fechaCompletado: Date, tiempoDedicado: int, estado: double): Progreso
Abstract <<Usuario>>
usuarioID: String
- nombre: String
- email: String
- contraseña: String
- tipoUsuario: String
+ estudiante = "Estudiante": String
+ profesor = "Profesor": String
+ Usuario(usuarioID: String, nombre: String, contraseña: String, email: tipoUsuario: String)
+ getTipoUsuario1(): String
+ getContraseña(): String
+ setContraseña(contraseña: String): void
+ getUsuarioID(): String
+ getNombre(): String
+ getEmail(): String
# getTipoUsuario(): String (abstract)
+ iniciarSesion(usuarioID: String, contraseña: String): boolean
+ anadirUsuario(usuarios: List<Usuario>, usuario: Usuario): void
Authenticator
- codigo: int
- fueAutenticado: boolean
- usuarios: List<Usuario>
- archivoUsuarios: String
+ Authenticator(codigo: int, fueAutenticado: boolean, usuarios: List<Usuario>, archivoUsuarios: String)
+ getCodigo(): int
+ setCodigo(codigo: int): void
+ isFueAutenticado(): boolean
+ setFueAutenticado(fueAutenticado: boolean): void + getUsuarios(): List<Usuario>
+ setUsuarios(usuarios: List<Usuario>): void
+ getArchivoUsuarios(): String
+ setArchivoUsuarios(archivoUsuarios: String): void
+ cambiarPassword(usuario: Usuario, contraseña: String): void
+ authenticate(usuarioID: String, contraseña: String, usuario: Usuario): boolean (static)
Profesor
+ learningPathsCreados: Map<String, LearningPath>
+ mapaActividades
+ profesor = "Profesor" : String
+ Profesor(UsuarioID: String, nombre: String, contraseña: String, email: String, tipoUsuario: String)
+ getTipoUsuario(): String
+ getLearningPathsCreados(): Map<String, LearningPath>
+ setLearningPathsCreados(learningPaths: Map<String, LearningPath>): void
+ getProfesor(): String
+ crearActividad(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite: Date, resenas: String, resultado: int, calificacion: int, tipo: String, earningPathID: String, actividadesPrevia: List<String>, actividadesSeguimiento: List<String>, parametrosEspecificos: HashMap<String, Object>): Actividad
+ CalificacionMinima(actividadID: String, calificacionMinima: double): void
+ crearLearningPath(LearningPathID: String, titulo: String, descripcion: String, objetivos: String, nivelDificultad: int, duracion: int, profesorID: String, actividadesID: List<String>, intereses: List<String>): LearningPath
+ revisarEstadoActividad(actividadID: String, LearningPathID: String): void
+ verProgresoEstudiante(estudianteID: String, learningPath: LearningPath): Map<String, String>
+ revisarFeedback(LearningPathId: String): List<Map>
+ calcularRating(LearningPathId: String): double
Actividades
Actividad
- actividadID: String
- descripcion: String
- objetivo: String
- nivelDificultad: int
- duracionEsperada: int
- esObligatoria: boolean
- fechaLimite: Date
- resenas: String
- calificacion: double
- resultado: int
- estado: String
- preguntas: String
- respuestasUsuario: List<String>
- actividadesPrevias: List<String>
- actividadesSeguimiento: List<String>
- tipoActividad: String
+ TAREA = "Tarea": String
+ EXAMEN = "Examen": String
+ RECURSOEDUCATIVO = "Recurso Educativo": String
+ QUIZ = "Quiz": String
+ ENCUESTA = "Encuesta": String
+ Actividad(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite:Date, resenas: String, calificacion: double, resultado: int, actividadesPrevias:List<String>, actividadesSeguimiento: List<String>, tipoActividad: String)
+ convertToJSONObject(): JSONObject
+ getDescripcion(): String
+ setDescripcion(descripcion: String): void
+ getObjetivo(): String
+ setObjetivo(objetivo: String): void
+ getTarea(): String (static)
+ getExamen(): String (static)
+ getRecursoeducativo(): String (static)
+ getDuracionEsperada(): int
+ setDuracionEsperada(duracionEsperada: int): void
+ getFechaLimite(): Date
+ setFechaLimite(fechaLimite: Date): void
+ getActividadID(): String
+ getNivelDificultad(): int
+ isEsObligatoria(): boolean
+ getResenas(): String
+ getCalificacion(): double
+ getResultado(): int
+ getEstado(): String
+ setEstado(estado: String): void
+ getTipoActividad(): String
+ agregarContenido(pregunta: String, opciones: List<String>): void
+ getActividadesPrevias(): List<String>
+ setActividadesPrevias(actividadesPrevias: List<String>): void
+ getActividadesSeguimiento(): List<String>
+ setActividadesSeguimiento(actividadesSeguimiento: List<String>): void
+ setTipoActividad(tipoActividad: String): void
Actividad previa
Actividad de seguimiento
Quiz
- preguntas: List<Pregunta>
- calificacionMinima: double
- respuestaCorrecta: String
- aprobado: boolean
- respuestaUsuario: String
+ Quiz(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite: Date, resenas: String, calificacion: double, resultado: int, actividadesPrevias: List<String>, actividadesSeguimiento: List<String>, preguntas: List<Pregunta>, calificacionMinima: double, respuestaCorrecta: String)
+ setRespuestaUsuario(respuesta: String): void
+ getCalificacionMinima(): double
+ setCalificacionMinima(calificacionMinima: double): void
+ convertToJSONObject(): JSONObject
+ getTipoActividad(): String
+ getPreguntas(): List<Pregunta>
+ setPreguntas(preguntas: List<Pregunta>): void
+ completarQuiz(): void
+ isAprobado(): boolean
+ setAprobado(): void
+ agregarContenido(pregunta: String, opciones: List<String>): void
Encuesta
- preguntas: List<String>
- tipo: String
+ ENCUESTA = "Encuesta": String
+ Encuesta(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite: Date, resenas: String, calificacion: double, resultado: int, actividadesPrevias: List<String>, actividadesSeguimiento: List<String>, preguntas: List<String>)
+ getPreguntas(): List<String>
+ setPreguntas(preguntas: List<String>): void
+ convertToJSONObject(): JSONObject
+ getTipoActividad(): String
+ agregarContenido(pregunta: String, opciones: List<String>): void
+ completarEncuesta(): void
RecursoEducativo
- tipoRecurso: String
- linkRecusro: String
- preguntas: List<String>
- tipo: String
+ RECURSOEDUCATIVO = "Recurso Educativo": String
+ RecursoEducativo(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite: Date, resenas: String, calificacion: double, resultado: int, actividadesPrevias: List<String>, actividadesSeguimiento: List<String>, tipoRecurso: String, linkRecusro: String)
+ convertToJSONObject(): JSONObject
+ getTipoActividad(): String
+ agregarContenido(pregunta: String, opciones: List<String>): void
+ getTipoRecurso(): String
+ setTipoRecurso(tipoRecurso: String): void
+ getLinkRecusro(): String
+ setLinkRecusro(linkRecusro: String): void
+ completarRecurso(): void
Examen
- preguntas: List<Pregunta>
- calificacionMinima: double
+ Examen(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite: Date, resenas: String, calificacion: double, resultado: int, actividadesPrevias: List<String>, actividadesSeguimiento: List<String>, preguntas: List<Pregunta>, calificacionMinima: double)
+ convertToJSONObject(): JSONObject
+ getTipoActividad(): String
+ agregarContenido(pregunta: String, opciones: List<String>): void
+ EstadoActual(estado: String): String
+ getCalificacionMinima(): double
+ setCalificacionMinima(calificacionMinima: double): void
+ getPreguntas(): List<Pregunta>
+ setPreguntas(preguntas: List<Pregunta>): void
+ completarExamen(): void
Tarea
- estado: String
- instrucciones: String
- preguntas: List<Pregunta>
- opciones: Collection<List<String>>
+ Tarea(actividadID: String, descripcion: String, objetivo: String, nivelDificultad: int, duracionEsperada: int, esObligatoria: boolean, fechaLimite: Date, resenas: String, actividadesPrevias: List<String>, actividadesSeguimiento: List<String>, preguntas: List<Pregunta>, instrucciones: String, estado: String)
+ getOpciones(preguntas: Map<String, List<String>>): void
+ completarTarea(): void
+ registrarEntrega(medioEntrega: String): String
+ getDescripcion(): String
+ setDescripcion(descripcion: String): void
+ getObjetivo(): String
+ setObjetivo(objetivo: String): void
+ getDuracionEsperada(): int
+ setDuracionEsperada(duracionEsperada: int): void
+ getFechaLimite(): Date
+ setFechaLimite(fechaLimite: Date): void
+ getActividadID(): String
+ getNivelDificultad(): int
+ isEsObligatoria(): boolean
+ getResenas(): String
+ getCalificacion(): double
+ getResultado(): int
+ getInstrucciones(): String
+ setInstrucciones(Instrucciones: String): void
+ getPreguntas(): List<Pregunta>
+ setPreguntas(preguntas: List<Pregunta>): void
Tarea
- pregunta: String
- opciones: List<String>
+ Pregunta(pregunta: String, opciones: List<String>)
Usuario
Authenticator
1
0..*