Requerimiento Funcional	**Nombre de la Clase**	**Nombre del método**
ReqFunc001 Register Users	Main	Register user()
	Controller	registerUser(userName : string, password : string) : string
		registerUser(userName : string, password : string, name : string, email : string, phoneNumber : string) : string
		registerUser(userName : string, password : string, name : string, email : string, phoneNumber : string, area : string, position : string) : string
	User	User(userName : string, password : string) : void
	Visitor	User(userName : string, password : string) : void
	Collector	Collector(name : string, email : string, phoneNumber : string)
	Investigator	Investigator(name : string, email : string, phoneNumber : string, area : string, position : string) : void
ReqFunc002 Manage Projects	Main	manageProjects()
	Controller	registerProject(typePillar : model.PillarType, name : string Id : string, description : string, initialDate : calendar, finalDate : calendar, state : boolean) : string
		modifyProject(userName : string, password : string, pillarName : string, projectId : string, newName : string, newDescription : string, newInitialDate : calendar, newFinalDate : calendar, newState : boolean) : string
		searchProject(pillarName : string, projectName : string) : string
		deleteProject(pillarName : string, idProject : string) : string
		searchUser(usernName:string): User
	User	searchPillar(namePillar) : string
		instanceof(userName):string
		searchProject(projectName : string) : Porject
	Pillar	searchProject(projectId : string) : Project

		addproject(proyect : model.Project) : void
		showProjects() : String
		getType() : string
	Project	Project(name : string, Id : string, description : string, initialDate : calendar, finalDate : string, state : boolean) : void
		setName(name : string) : void
		setId(Id : string) : void
		setDescription(description : string) : void
		setInitialDate(initialDate : calendar) : void
		setFinalDate(finalDate : calendar) : void
		setState(state : boolean) : void
		toString() : string
ReqFunc003 Manage Evidence	Main	manageEvidence() : void
	Controller	registerEvidenceToProject(userName : string, password : string, pillarName : string, projectId : string, evidenceName : string, description : string, review : string) : string
		registerEvidenceToProject(userName : string, password : string, pillarName : string, projectId : string, evidenceName : string, description : string, fileType : TypeFile, file : string) : string
		modifyEvidence(userName : string, password : string, pillarName : string, projectId : string, evidenceName : string, newName : string, newDescription : string, newReview : string) : string
		modifyEvidience(userName : string, password : string, pillarName : string, projectId : string, evidenceName : string, newName : string, newDescription : string, newFile : string) : string
		desactivateEvidence(pillarName : string, projectId : string, evidenceName : string) : string
		registerReviewToProject(pillarName : string, projectId : string, keyPointName : string, evidenceName : string, description : string review : string) : string
	User	searchPillar(namePillar) : string
		instanceof(userName) : string

		searchProject(projectName : string) : Porject
	Project	registerEvidence(evidence : Evidence) : string
		searchEvidence(nameEvidence : string) : Evidence
	Evidence	Evidencia(name : string, description : string, registeraDate : calendar) : void
		setName(name : string) : void
		setDescription(description : string) : void
		setRegisterDate(registerDate : calendar) : void
		getName() : string
		getDescription() : string
		getRegisterDate() : calendar
		toString()
	Review	Review(review : string, state : boolean) : void
		setReview(review : string) : void
		setState(state : boolean) : void
		setAttribute(attribute) : void
		getReview() : string
		getState() : boolean
		getAttribute()
	File	File(fileType : FileType) : void
ReqFunc004 Manage Points of Interest	Main	managePointOfInterest() : void
	Conroller	registerKeyPoint(userName : string, password : string, pillarName : string, projectId : string, qRCode : strin, name : string, x : string, y : string) : string
		modifyKeyPoint(userName : string, password : string, pillarName : string, projectId : string, newQRCode : strin, newName : string, newX : string, newY : string) : string
		consultKeyPoint(pillarName : string, projectId : string) : string
		deleteKeyPoint(userName : string, password : string, pillarName : string, projectId : string, keyPointName) : string

		registerObservation(userName : string, password : string, projectName : string, keyPointName : string, observation : string) : string
		reportKeyPoint(userName : string, passWord : string, pillarName : string, keyPointName : string) : string
	User	searchPillar(namePillar) : string
	Pillar	searchProject(projectId : string) : Project
	Project	searchKeyPoint(nameKeyPoint : String) : model.KeyPoint
		registerKeyPoint(keyPoint : model.KeyPoint) : string
		getId() : string
	KeyPoint	KeyPoint(name : string, QRCode : string, x : string, y : string)
		toString() : string
		getAttribute()
		setAttribute(attribute) : void
		getX() : string
		setX(x : string) : void
		getY() : string
		setY(y : string) : void
		getName() : string
		setName(name : string) : void
		getQRCode() : string
		setQRCode(QRCode : string) : void
		addObservation(observation : string) : void