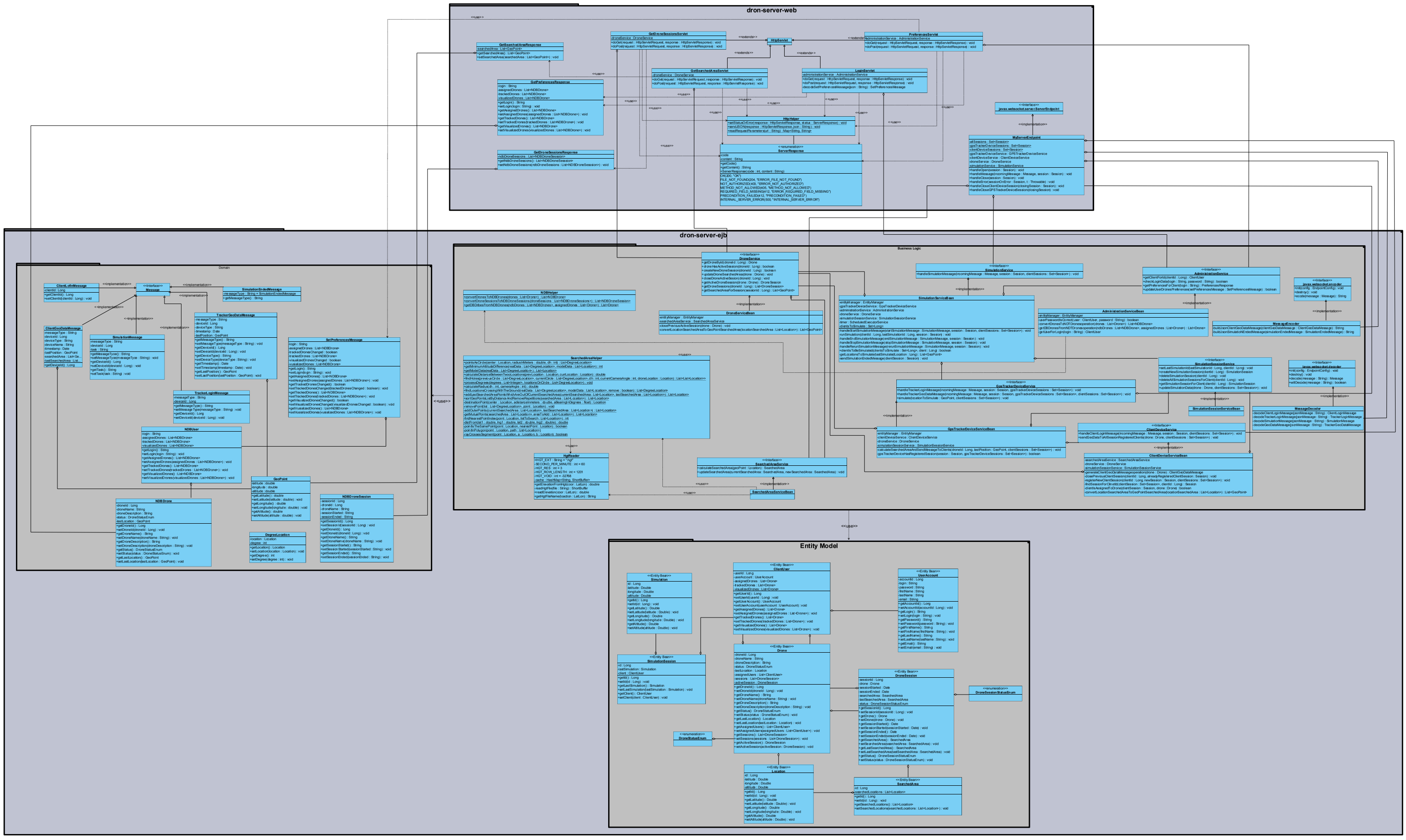
dron-server-web
GetDroneSessionsServlet
-droneService : DroneService
+doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+doPost(request : HttpServletRequest, response : HttpServletResponse) : void
HttpServlet
PreferencesServlet
-administrationService : AdministrationService
+doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+doPost(request : HttpServletRequest, response : HttpServletResponse) : void
GetSearchatAreaResponse
-searchedArea : List<GeoPoint>
+getSearchedArea() : List<GeoPoint>
+setSearchedArea(searchedArea : List<GeoPoint>) : void
GetSearchedAreaServlet
-droneService : DroneService
+doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+doPost(request : HttpServletRequest, response : HttpServletResponse) : void
LoginServlet
-administrationService : AdministrationService
+doGet(request : HttpServletRequest, response : HttpServletResponse) : void
+doPost(request : HttpServletRequest, response : HttpServletResponse) : void
-decodeSetPreferencesMessage(json : String) : SetPreferencesMessage
GetPreferencesResponse
-login : String
-assignedDrones : List<NDBDrone>
-trackedDrones : List<NDBDrone>
-visualizedDrones : List<NDBDrone>
+getLogin() : String
+setLogin(login : String) : void
+getAssignedDrones() : List<NDBDrone>
+setAssignedDrones(assignedDrones : List<NDBDrone>) : void
+getTrackedDrones() : List<NDBDrone>
+setTrackedDrones(trackedDrones : List<NDBDrone>) : void
+getVisualizedDrones() : List<NDBDrone>
+setVisualizedDrones(visualizedDrones : List<NDBDrone>) : void
HttpHelper
+sentStatusOrErrorResponse : HttpServletResponse, status : ServerResponse) : void
+sendJSONResponse : HttpServletResponse,json : String) : void
+readRequestParameterstuff : String) : Map<String, String>
<<interface>>
javax.websocket.server.ServerEndpoint
GetDroneSessionsResponse
-ndbDroneSessions : List<NDBDroneSession>
+getNdbDroneSessions() : List<NDBDroneSession>
+setNdbDroneSessions(ndbDroneSessions : List<NDBDroneSession>) : void
<<enumeration>>
ServerResponse
-code
-content : String
+getCode()
+getContent() : String
+ServerResponse(code : int, content : String)
OK(200, "OK")
FILE_NOT_FOUND(204, "ERROR_FILE_NOT_FOUND")
NOT_AUTHORIZED(3,49, "ERROR_NOT_AUTHORIZED")
METHOD_NOT_ALLOWED(405, "METHOD_NOT_ALLOWED")
REQUIRED_FIELD_MISSING(412, "ERROR_REQUIRED_FIELD_MISSING")
PRECONDITION_FAILED(412, "PRECONDITION_FAILED")
INTERNAL_SERVER_ERROR(500, "INTERNAL_SERVER_ERROR")
MyServeEndpoint
-allSessions : Set<Session>
-gpsTrackerDeviceSessions : Set<Session>
-clientDeviceSessions : Set<Session>
-gpsTrackerDeviceService : GPSTrackerDeviceService
-clientDeviceService : ClientDeviceService
-droneService : DroneService
-simulationService : SimulationService
+handleOpen(session : Session) : void
+handleMessage(incomingMessage : Message, session : Session) : void
+handleClose(session : Session) : void
+handleError(session:OnError : Session, t : Throwable) : void
+handleCloseClientDeviceSession(closingSession : Session) : void
+handleCloseGPSTrackerDeviceSession(closingSession) : void
dron-server-ejb
Business Logic
Domain
ClientLoginMessage
-clientId : Long
+getClientId() : Long
+setClientId(clientId : Long) : void
<<interface>>
Message
SimulationEndedMessage
-messageType : String = SimulationEndedMessage
+getMessageType() : String
ClientGeoDataMessage
-messageType : String
-deviceId : Long
-deviceType : String
-deviceNems : String
-timestamp : Date
-lastPosition : GeoPoint
-searchedArea : List<GeoPoint>
-lastSearchedArea : List<GeoPoint>
+getDeviceId() : Long
SimulationMessage
-messageType : String
-deviceId : Long
-task : String
+getMessageType() : String
+setMessageType(messageType : String) : void
+getDeviceId() : Long
+setDeviceId(deviceId : Long) : void
+getTask() : String
+setTask(task : String) : void
TrackerGeoDataMessage
-messageType : String
-deviceId : Long
-deviceType : String
-timestamp : Date
-lastPosition : GeoPoint
+getMessageType() : String
+setMessageType(messageType : String) : void
+getDeviceId() : Long
+setDeviceId(deviceId : Long) : void
+getDeviceType() : String
+setDeviceType(deviceType : String) : void
+getTimestamp() : Date
+setTimestamp(timestamp : Date) : void
+getLastPosition() : GeoPoint
+setLastPosition(lastPosition : GeoPoint) : void
SetPreferencesMessage
-login : String
-assignedDrones : List<NDBDrone>
-trackedDrones : List<NDBDrone>
-visualizedDrones : List<NDBDrone>
+getLogin() : String
+setLogin(login : String) : void
+getAssignedDrones() : List<NDBDrone>
+setAssignedDrones(assignedDrones : List<NDBDrone>) : void
+getTrackedDrones() : List<NDBDrone>
+setTrackedDrones(trackedDrones : List<NDBDrone>) : void
+getVisualizedDrones() : List<NDBDrone>
+setVisualizedDrones(visualizedDrones : List<NDBDrone>) : void
TrackerLoginMessage
-messageType : String
-deviceId : Long
+getMessageType() : String
+setMessageType(messageType : String) : void
+getDeviceId() : Long
+setDeviceId(deviceId : Long) : void
NDBUser
-login : String
-assignedDrones : List<NDBDrone>
-trackedDrones : List<NDBDrone>
-visualizedDrones : List<NDBDrone>
+getLogin() : String
+setLogin(login : String) : void
+getAssignedDrones() : List<NDBDrone>
+setAssignedDrones(assignedDrones : List<NDBDrone>) : void
+getTrackedDrones() : List<NDBDrone>
+setTrackedDrones(trackedDrones : List<NDBDrone>) : void
+getVisualizedDrones() : List<NDBDrone>
+setVisualizedDrones(visualizedDrones : List<NDBDrone>) : void
GeoPoint
-latitude : double
-longitude : double
-altitude : double
+getLatitude() : double
+setLatitude(latitude : double) : void
+getLongitude() : double
+setLongitude(longitude : double) : void
+getAltitude() : double
+setAltitude(altitude : double) : void
NDBDrone
-droneId : Long
-droneName : String
-droneDescription : String
-status : DroneStatusEnum
-lastLocation : GeoPoint
+getDroneId() : Long
+setDroneId(droneId : Long) : void
+getDroneName() : String
+setDroneName(droneName : String) : void
+getDroneDescription() : String
+setDroneDescription(droneDescription : String) : void
+getStatus() : DroneStatusEnum
+setStatus(status : DroneStatusEnum) : void
+getLastLocation() : GeoPoint
+setLastLocation(lastLocation : GeoPoint) : void
DegreeLocation
-location : Location
-degree : int
+getLocation() : Location
+setLocation(location : Location) : void
+getDegree() : int
+setDegree(degree : int) : void
NDBDroneSession
-sessionId : Long
-droneId : Long
-droneName : String
-sessionStarted : String
-sessionEnded : String
+getSessionId() : Long
+setSessionId(sessionId : Long) : void
+getDroneId() : Long
+setDroneId(droneId : Long) : void
+getDroneName() : String
+setDroneName(droneName : String) : void
+getSessionStarted() : String
+setSessionStarted(sessionStarted : String) : void
+getSessionEnded() : String
+setSessionEnded(sessionEnded : String) : void
<<interface>>
DroneService
+getDroneById(droneId : Long) : Drone
+droneHasActiveSession(droneId : Long) : boolean
+createNewDroneSession(droneId : Long) : boolean
+updateDroneSearchedArea(drone : Drone) : void
+doneDroneAddSelectedSession(droneId : Long) : void
+getActiveDroneSession(drone : Drone) : DroneSession
+getDroneSessions(droneId : Long) : List<DroneSession>
+getSearchedAreaForSession(sessionId : Long) : List<GeoPoint>
<<interface>>
SimulationService
+handleSimulationMessage(incomingMessage : Message, session : Session, clientSessions : Set<Session>) : void
<<interface>>
AdministrationService
+getClientForId(clientId : Long) : ClientUser
+checkLoginDialogin : String, password : String) : boolean
+getPreferencesForClient(login : String) : PreferencesResponse
+updateUserDronesPreferences(setPreferencesMessage : SetPreferenceMessage) : boolean
<<interface>>
javax.websocket.encoder
+init(config : EndpointConfig) : void
+destroy() : void
+encode(message : Message) : String
NDBHelper
+convertDroneSessionsToNDBDroneSessions(droneSessions : List<NDBDroneSession>) : List<NDBDroneSession>
+convertDroneSessionsToNDBDroneSessions(droneSessions : List<NDBDroneSession>, assignedDrones : List<Drone>) : List<Drone>
+getNDBforBestFromNDBDronesandNDBDrones : List<NDBDrone>, assignedDrones : List<Drone>) : List<Drone>
DroneServiceBean
-entityManager : EntityManager
-searchedAreaService : SearchedAreaService
-closePreviousActiveSessionIfExists(drone : Drone) : void
+convertLocationSearchedAreaToGeoPointSearchedArea(locationSearchedArea : List<Location>) : List<GeoPoint>
SimulationServiceBean
-entityManager : EntityManager
-gpsTrackerDeviceService : GpsTrackerDeviceService
-administrationService : AdministrationService
-droneService : DroneService
-simulationSessionService : SimulationSessionService
-timer : ScheduledExecutorService
-clientsToSimulate : SetLong>
+handleStartSimulationMessage(startSimulationMessage : SimulationMessage, session : Session, clientSessions : Set<Session>) : void
+runSimulation(clientId : Long, lastSimulationId : Long, session : Session) : void
+handleEndSimulationMessage(endSimulationMessage : SimulationMessage, session : Session) : void
+handleBodyupSimulationMessage(bodyupSimulationMessage : SimulationMessage, session : Session) : void
+handleReturnSimulationMessage(returnSimulationMessage : SimulationMessage, session : Session) : void
+clientsToBeSimulated(clientsToSimulate : SetLong>, client : Long) : boolean
-getLocationsToSimulate(lastSimulationId : Long) : List<GeoPoint>
-sendSimulationEndedMessage(clientSession : Session) : void
AdministrationServiceBean
-entityManager : EntityManager
-userPasswordsDomainUser : ClientUser, password : String) : boolean
-convertDronesToNDBDronesoperation(drones : List<Drone>) : List<NDBDrone>
-getNDBDronesFromNDBDronesoperation(ndbDrones : List<NDBDrone>, assignedDrones : List<Drone>) : List<Drone>
-getUserForLoginlogin : String) : ClientUser
MessageEncoder
-buildJsonClientGeoDataMessage(clientGeoDataMessage : ClientGeoDataMessage) : String
-buildJsonSimulationEndedMessage(simulationEndedMessage : SimulationEndedMessage) : String
SearchedAreaHelper
-pointsAroundcenter : Location, radiusInMeters : double, dh : int[] : List<DegreeLocation>
+getMinumumAbsoluteDifference(realData : List<DegreeLocation>, modelData : List<Location>) : int
+getNearestDatasetedData : List<DegreeLocation>) : List<Location>
+calculateDistanceBetweenTwoLocations(pointLocation : Location, curtLocation : Location) : double
+findNextexpireduCircle : List<DegreeLocation>, currentCircle : List<DegreeLocation>, dh : int, currentCameraAngle : int, droneLocation : Location) : List<List<Location>>
+processDegree(degrees : List<Integer>, locationOnCircle : List<DegreeLocation>) : void
+calculateRadius(dh : int, camerAngle : int) : double
+findLocationOnCreasingWithTheAroundvalueData : List<DegreeLocation>, modelData : List<Location>, remove : boolean) : List<DegreeLocation>
+addLastSearchedAreaPointsWithAreOutOfCurrentSearchedAreaaccurrentSearchedArea : List<Location>, lastSearchedArea : List<Location>) : List<Location>
+pointSeparationLeftyDistanceAndRemovePointsinsearchedArea : List4, oputation) : List<Location>
-destinationPoint(center : Location, adistancekinmeters : double, alfearangInDegrees : float) : Location
-removePoint(list : List<DegreeLocation>, point : Location) : void
-addOutsePoints(currentSearchedArea : List<Location>, lastSearchedArea : List<Location>) : List<Location>
-getMutualPoints(searchedArea : List<Location>, arrayToAdd : List<Location>) : List<Location>
-findNearestPointIndex(point : Location, listToSearch : List<Location>) : int
-distFromSet1 : double, log1 : double, lat2 : double, log2 : double) : double
-pointsTheSamePoint(point : Location, nearestPoint : Location) : boolean
-pointInPolygon(point : Location, path : List<Location>)
-clipCrossesSegment(point : Location, a : Location, b : Location) : boolean
HgtReader
-HGT_EXT : String = ".hgt"
-SECOND_PER_MINUTE : int = 60
-HGT_RES : int = 3
-HGT_ROW_LENGTH : int = 1201
-HGT_VOID : int = -32768
-cache : HashMap<String, ShortBuffer>
+getElevationFromHgt(coor : LatLon) : double
-readHgtFile(file : String) : ShortBuffer
+readElevation(coor : LatLon) : double
+getHgtFileName(loadlon : LatLon) : String
<<interface>>
SearchatAreaService
+calculateSearchedAreaUsingGePoint : Location) : SearchedArea
+updateSearchedArea(currentSearchedArea : SearchatArea, newSearchedArea : SearchedArea) : void
SerchedAreaServiceBean
GpsTrackerDeviceService
-entityManager : EntityManager
-clientDeviceService : ClientDeviceService
-droneService : DroneService
-simulationSessionService : SimulationSessionService
-calculateSearchedAreaAndSendMessageToClients(droneId : Long, lastPosition : GeoPoint, clientSessions : Set<Session>) : void
-gpsTrackerDeviceHasRegisteredSession(session : Session, gpsTrackerDeviceSessions : Set<Session>) : boolean
GpsTrackerDeviceServiceBean
-entityManager : EntityManager
-clientDeviceService : ClientDeviceService
-droneService : DroneService
-simulationSessionService : SimulationSessionService
-calculateSearchedAreaAndSendMessageToClients(droneId : Long, lastPosition : GeoPoint, clientSessions : Set<Session>) : void
-gpsTrackerDeviceHasRegisteredSession(session : Session, gpsTrackerDeviceSessions : Set<Session>) : boolean
<<interface>>
SimulationSessionService
+getLastSimulationId(lastSimulationId : Long, clientId : Long) : void
+createNewSimulationSession(clientId : Long) : SimulationSession
+deleteSimulationSession(clientId : Long) : void
+deleteAllSimulationSessionsForClient(clientId : Long) : void
+getSimulationSessionForClient(clientId : Long) : SimulationSession
+updateSimulationDataDrone : Drone, clientSessions : Set<Session>) : void
SimulationSessionServiceBean
<<interface>>
ClientDeviceService
+handleClientLoginMessage(incomingMessage : Message, session : Session, clientSessions : Set<Session>) : void
+sendGeoDataToAllSessionRegisteredClients(drone : Drone, clientSessions : Set<Session>) : void
ClientDeviceServiceBean
-searchedAreaService : SearchedAreaService
-droneService : DroneService
-simulationSessionService : SimulationSessionService
+generateClientGeoDataMessageoperation(drone : Drone) : ClientGeoDataMessage
-closePreviousClientSession(clientId : Long, alreadyRegisteredClientSession : Session) : void
-registerNewClientSession(clientId : Long, newSession : Session, clientSessions : Set<Session>) : void
-findSessionForClient(clientSession : Set<Session>, clientId : Long) : Session
-clientsAssignedToDrone(clientSession : Session, drone : Drone) : boolean
-convertLocationSearchedAreaToGeoPointSearchedArea(locationSearchedArea : List<Location>) : List<GeoPoint>
<<interface>>
javax.websocket.decoder
+init(config : EndpointConfig) : void
+destroy() : void
+decode(message : String) : Message
+willDecode(message : String) : boolean
MessageDecoder
-decodeClientLoginMessage(jsonMessage : String) : ClientLoginMessage
-decodeTrackerLoginMessage(jsonMessage : String) : TrackerLoginMessage
-decodeSimulationMessage(jsonMessage : String) : SimulationMessage
-decodeGeoDataMessage(jsonMessage : String) : TrackerGeoDataMessage
Entity Model
<<Entity Bean>>
Simulation
-id : Long
-lastSimulation : Simulation
-client : ClientUser
+getId() : Long
+setId(id : Long) : void
+getLastSimulation() : Simulation
+setLastSimulation(lastSimulation : Simulation) : void
+getClient() : ClientUser
+setClient(client : ClientUser) : void
<<Entity Bean>>
ClientUser
-userId : Long
-userAccount : UserAccount
-assignedDrones : List<Drone>
-trackedDrones : List<Drone>
-visualizedDrones : List<Drone>
+getUserId() : Long
+setUserId(userId : Long) : void
+getUserAccount() : UserAccount
+setUserAccount(userAccount : UserAccount) : void
+getAssignedDrones() : List<Drone>
+setAssignedDrones(assignedDrones : List<Drone>) : void
+getTrackedDrones() : List<Drone>
+setTrackedDrones(trackedDrones : List<Drone>) : void
+getVisualizedDrones() : List<Drone>
+setVisualizedDrones(visualizedDrones : List<Drone>) : void
<<Entity Bean>>
UserAccount
-accountId : Long
-login : String
-password : String
-firstName : String
-lastName : String
-email : String
+getAccountId() : Long
+setAccountId(accountId : Long) : void
+getLogin() : String
+setLogin(login : String) : void
+getPassword() : String
+setPassword(password : String) : void
+getFirstName() : String
+setFirstName(firstName : String) : void
+getLastName() : String
+setLastName(lastName : String) : void
+getEmail() : String
+setEmail(email : String) : void
<<Entity Bean>>
SimulationSession
-id : Long
-lastSimulation : Simulation
-client : ClientUser
+getId() : Long
+setId(id : Long) : void
+getLastSimulation() : Simulation
+setLastSimulation(lastSimulation : Simulation) : void
+getClient() : ClientUser
+setClient(client : ClientUser) : void
<<Entity Bean>>
Drone
-droneId : Long
-droneName : String
-droneDescription : String
-status : DroneStatusEnum
-lastLocation : Location
-assignedUsers : List<ClientUser>
-sessions : List<DroneSession>
-activeSession : DroneSession
+getDroneId() : Long
+setDroneId(droneId : Long) : void
+getDroneName() : String
+setDroneName(droneName : String) : void
+getDroneDescription() : String
+setDroneDescription(droneDescription : String) : void
+getStatus() : DroneStatusEnum
+setStatus(status : DroneStatusEnum) : void
+getLastLocation() : Location
+setLastLocation(lastLocation : Location) : void
+getAssignedUsers() : List<ClientUser>
+setAssignedUsers(assignedUsers : List<ClientUser>) : void
+getSessions() : List<DroneSession>
+setSessions(sessions : List<DroneSession>) : void
+getActiveSession() : DroneSession
+setActiveSession(activeSession : DroneSession) : void
<<Entity Bean>>
DroneSession
-sessionId : Long
-drone : Drone
-sessionStarted : Date
-sessionEnded : Date
-searchedArea : SearchedArea
-lastSearchedArea : SearchedArea
-status : DroneSessionStatusEnum
+getSessionId() : Long
+setSessionId(sessionId : Long) : void
+getDrone() : Drone
+setDrone(drone : Drone) : void
+getSessionStarted() : Date
+setSessionStarted(sessionStarted : Date) : void
+getSessionEnded() : Date
+setSessionEnded(sessionEnded : Date) : void
+getSearchedArea() : SearchedArea
+setSearchedArea(searchedArea : SearchedArea) : void
+getLastSearchedArea() : SearchedArea
+setLastSearchedArea(lastSearchedArea : SearchatArea) : void
+getStatus() : DroneSessionStatusEnum
+setStatus(status : DroneSessionStatusEnum) : void
<<enumeration>>
DroneSessionStatusEnum
<<enumeration>>
DroneStatusEnum
<<Entity Bean>>
Location
-id : Long
-latitude : Double
-longitude : Double
-altitude : Double
+getId() : Long
+setId(id : Long) : void
+getLatitude() : Double
+setLatitude(latitude : Double) : void
+getLongitude() : Double
+setLongitude(longitude : Double) : void
+getAltitude() : Double
+setAltitude(altitude : Double) : void
<<Entity Bean>>
SearchedArea
-id : Long
-searchedLocations : List<Location>
+getId() : Long
+setId(id : Long) : void
+getSearchedLocations() : List<Location>
+setSearchedLocations(searchedLocations : List<Location>) : void