

controllers

ControllerAccount
-TAG : String = "ControllerAccount"
<<Property>> -user : User
+setUser(intent : Intent) : void

ControllerPath
-TAG : String = "ControllerPath"
+jsonToNaPath(naPathJSON : JSONObject) : NaPath

ControllerAddPath
-TAG : String = "ControllerAddPath"
+createPathWithGPX(name : String, difficulties : String, duration : int, disableAccessibility : boolean, description : String, uriGPX : Uri, userEmail : String, activity : Activity) : void
-GPXParser(path : Uri) : ArrayList<GeoPoint>
-getPathFromGPX(parsedGPX : Gpx) : ArrayList<GeoPoint>
-GPXTrackAdapter(GPXTracks : List<Track>) : ArrayList<GeoPoint>
-GPXRouteAdapter(GPXRoute : List<Route>) : ArrayList<GeoPoint>
-GPXWayPointAdapter(GPXTracks : List<WayPoint>) : ArrayList<GeoPoint>
-openInsertPointOfInterest(response : JSONObject, activity : Activity, path : NaPath, emailUser : String) : void
+addPathtoDatabase(naPath : NaPath, activity : Activity) : void
+createNaPath(name : String, difficulties : String, duration : int, disableAccessibility : boolean, description : String, path : ArrayList<GeoPoint>, userEmail : String, country : String, region : String, activity : Activity) : void
-pathJSONAdapter(path : ArrayList<GeoPoint>) : JSONArray
-difficultyEnumAdapter(difficulties : String) : DifficultyEnum
+addPointOfInterestFromExternalUser(sights : ArrayList<Sights>, naPath : NaPath, activity : Activity, emailUser : String) : void
+addPointOfInterestToDatabase(sights : ArrayList<Sights>, naPath : NaPath, activity : Activity, emailUser : String) : void
-openAddPhotoToPhoto(activity : Activity, naPath : NaPath, emailUser : String) : void
+checkInputDuration(durationEditText : EditText) : String
+checkInputName(namePath : String) : String
+nameAlreadyExist(naPaths : ArrayList<NaPath>, name : String) : boolean

ControllerSearch
+searchUserWith(data : String, users_result : ArrayList<User>) : ArrayList<User>
+searchPathWith(name : String, paths : ArrayList<NaPath>, naPathFilters : NaPathFilters) : ArrayList<NaPath>
-filterDisableAccess(paths : ArrayList<NaPath>, accessible : Boolean) : ArrayList<NaPath>
-filterDifficult(paths : ArrayList<NaPath>, difficult : DifficultyEnum) : ArrayList<NaPath>
-filterCountry(paths : ArrayList<NaPath>, country : String) : ArrayList<NaPath>
-filterRegion(paths : ArrayList<NaPath>, region : String) : ArrayList<NaPath>
-filterDuration(paths : ArrayList<NaPath>, minDuration : Integer) : ArrayList<NaPath>

ControllerRegistration
-TAG : String = "ControllerRegistration"
+checkName(name : String, regex : String) : String
+checkPassword(password : String, regex : String, confirmedPassword : String) : String