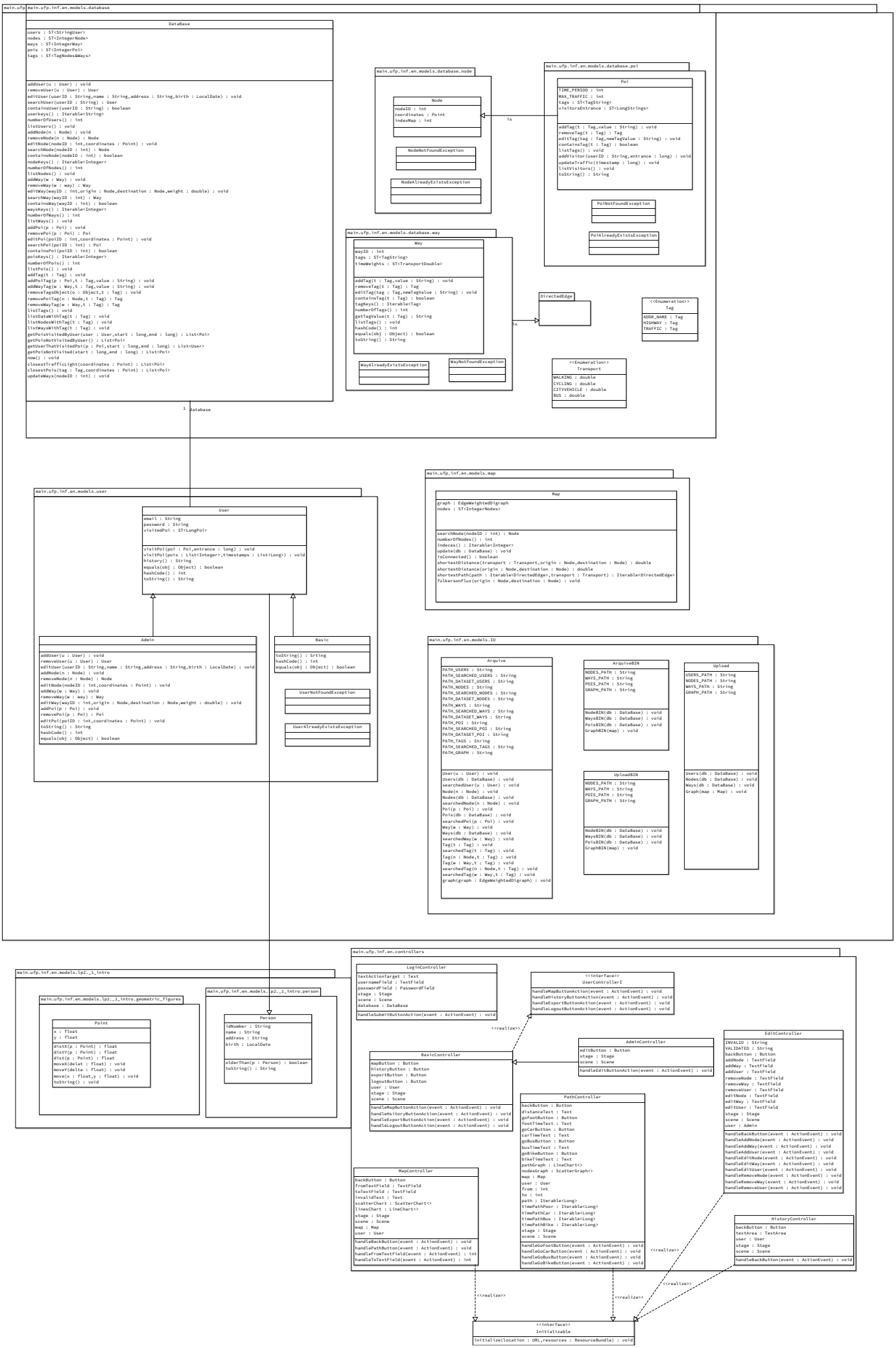
main.ufp.inf.en.models.database
DataBase
users : ST<StringUser>
nodes : ST<IntegerNode>
ways : ST<IntegerWay>
pois : ST<IntegerPoi>
tags : ST<TagNodesWays>
addUser(u : User) : void
removeUser(u : User) : User
editUser(userID : String,name : String,address : String,birth : LocalDate) : void
searchUser(userID : String) : User
containsUser(userID : String) : boolean
userkeys() : Iterable<String>
numberOfUsers() : int
listUsers() : void
addNode(n : Node) : void
removeNode(n : Node) : Node
editNode(nodeID : int,coordinates : Point) : void
searchNode(nodeID : int) : Node
containsNode(nodeID : int) : boolean
nodeKeys() : Iterable<Integer>
numberOfNodes() : int
listNodes() : void
addWay(w : Way) : void
removeWay(w : way) : Way
editWay(wayID : int,origin : Node,destination : Node,weight : double) : void
searchWay(wayID : int) : Way
containsWay(wayID : int) : boolean
waysKeys() : Iterable<Integer>
numberOfWays() : int
listWays() : void
addPoi(p : Poi) : void
removePoi(p : Poi) : Poi
editPoi(poiID : int,coordinates : Point) : void
searchPoi(poiID : int) : Poi
containsPoi(poiID : int) : boolean
poisKeys() : Iterable<Integer>
numberOfPois() : int
listPois() : void
addTag(t : Tag) : void
addPoiTag(p : Poi,t : Tag,value : String) : void
addWayTag(w : Way,t : Tag,value : String) : void
removeTagsObject(o : Object,t : Tag) : void
removePoiTag(n : Node,t : Tag) : Tag
removeWayTag(w : Way,t : Tag) : Tag
listTags() : void
listDataWithTag(t : Tag) : void
listNodesWithTag(t : Tag) : void
listWaysWithTag(t : Tag) : void
getPoisVisitedByUser(user : User,start : long,end : long) : List<Poi>
getPoisNotVisitedByUser() : List<Poi>
getUserThatVisitedPoi(p : Poi,start : long,end : long) : List<User>
getPoisNotVisited(start : long,end : long) : List<Poi>
now() : void
closestTrafficLight(coordinates : Point) : List<Poi>
closestPois(tag : Tag,coordinates : Point) : List<Poi>
updateWays(nodeID : int) : void
main.ufp.inf.en.models.database.node
Node
nodeID : int
coordinates : Point
indexMap : int
NodeNotFoundException
NodeAlreadyExistsException
is
main.ufp.inf.en.models.database.poi
Poi
TIME_PERIOD : int
MAX_TRAFFIC : int
tags : ST<TagString>
visitorsEntrance : ST<LongStrings>
addTag(t : Tag,value : String) : void
removeTag(t : Tag) : Tag
editTag(tag : Tag,newTagValue : String) : void
containsTag(t : Tag) : boolean
listTags() : void
addVisitor(userID : String,entrance : long) : void
updateTraffic(timestamp : long) : void
listVisitors() : void
toString() : String
PoiNotFoundException
PoiAlreadyExistsException
main.ufp.inf.en.models.database.way
Way
wayID : int
tags : ST<TagString>
timeWeights : ST<TransportDouble>
addTag(t : Tag,value : String) : void
removeTag(t : Tag) : Tag
editTag(tag : Tag,newTagValue : String) : void
containsTag(t : Tag) : boolean
tagKeys() : Iterable<Tag>
numberOfTags() : int
getTagValue(t : Tag) : String
listTags() : void
hashCode() : int
equals(obj : Object) : boolean
toString() : String
WayAlreadyExistsException
WayNotFoundException
is
DirectedEdge
<<Enumeration>>
Tag
ADDR_NAME : Tag
HIGHWAY : Tag
TRAFFIC : Tag
<<Enumeration>>
Transport
WALKING : double
CYCLING : double
CITYVEHICLE : double
BUS : double
1
database
main.ufp.inf.en.models.user
User
email : String
password : String
visitedPoi : ST<LongPoi>
visitPoi(poi : Poi,entrance : long) : void
visitPoi(pois : List<Integer>,timestamps : List<Long>) : void
history() : String
equals(obj : Object) : boolean
hashCode() : int
toString() : String
Admin
addUser(u : User) : void
removeUser(u : User) : User
editUser(userID : String,name : String,address : String,birth : LocalDate) : void
addNode(n : Node) : void
removeNode(n : Node) : Node
editNode(nodeID : int,coordinates : Point) : void
addWay(w : Way) : void
removeWay(w : way) : Way
editWay(wayID : int,origin : Node,destination : Node,weight : double) : void
addPoi(p : Poi) : void
removePoi(p : Poi) : Poi
editPoi(poiID : int,coordinates : Point) : void
toString() : String
hashCode() : int
equals(obj : Object) : boolean
Basic
toString() : Srting
hashCode() : int
equals(obj : Object) : boolean
UserNotFoundException
UserAlreadyExistsException
main.ufp.inf.en.models.map
Map
graph : EdgeWeightedDigraph
nodes : ST<IntegerNodes>
searchNode(nodeID : int) : Node
numberOfNodes() : int
indeces() : Iterable<Integer>
update(db : DataBase) : void
isConnected() : boolean
shortestDistance(transport : Transport,origin : Node,destination : Node) : double
shortestDistance(origin : Node,destination : Node) : double
shortestPath(path : Iterable<DirectedEdge>,transport : Transport) : Iterable<DirectedEdge>
fulkersonFlux(origin : Node,destination : Node) : void
main.ufp.inf.en.models.IO
Arquive
PATH_USERS : String
PATH_SEARCHED_USERS : String
PATH_DATASET_USERS : String
PATH_NODES : String
PATH_SEARCHED_NODES : String
PATH_DATASET_NODES : String
PATH_WAYS : String
PATH_SEARCHED_WAYS : String
PATH_DATASET_WAYS : String
PATH_POI : String
PATH_SEARCHED_POI : String
PATH_DATASET_POI : String
PATH_TAGS : String
PATH_SEARCHED_TAGS : String
PATH_GRAPH : String
User(u : User) : void
Users(db : DataBase) : void
searchedUser(u : User) : void
Node(n : Node) : void
Nodes(db : DataBase) : void
searchedNode(n : Node) : void
Poi(p : Poi) : void
Pois(db : DataBase) : void
searchedPoi(p : Poi) : void
Way(w : Way) : void
Ways(db : DataBase) : void
searchedWay(w : Way) : void
Tag(t : Tag) : void
searchedTag(t : Tag) : void
Tag(n : Node,t : Tag) : void
Tag(w : Way,t : Tag) : void
searchedTag(n : Node,t : Tag) : void
searchedTag(w : Way,t : Tag) : void
graph(graph : EdgeWeightedDigraph) : void
ArquiveBIN
NODES_PATH : String
WAYS_PATH : String
POIS_PATH : String
GRAPH_PATH : String
NodeBIN(db : DataBase) : void
WaysBIN(db : DataBase) : void
PoisBIN(db : DataBase) : void
GraphBIN(map) : void
UploadBIN
NODES_PATH : String
WAYS_PATH : String
POIS_PATH : String
GRAPH_PATH : String
NodeBIN(db : DataBase) : void
WaysBIN(db : DataBase) : void
PoisBIN(db : DataBase) : void
GraphBIN(map) : void
Upload
USERS_PATH : String
NODES_PATH : String
WAYS_PATH : String
GRAPH_PATH : String
Users(db : DataBase) : void
Nodes(db : DataBase) : void
Ways(db : DataBase) : void
Graph(map : Map) : void
main.ufp.inf.en.models.lp2._1_intro
main.ufp.inf.en.models.lp2._1_intro.geometric_figures
Point
x : float
y : float
distX(p : Point) : float
distY(p : Point) : float
dist(p : Point) : float
moveX(delat : float) : void
moveY(delta : float) : void
move(x : float,y : float) : void
toString() : void
main.ufp.inf.en.models.lp2._1_intro.person
Person
idNumber : String
name : String
address : String
birth : LocalDate
olderThan(p : Person) : boolean
toString() : String
main.ufp.inf.en.controllers
LoginController
textActionTarget : Text
usernameField : TextField
passwordField : PasswordField
stage : Stage
scene : Scene
database : DataBase
handleSubmitButtonAction(event : ActionEvent) : void
<<interface>>
UserControllerI
handleMapButtonAction(event : ActionEvent) : void
handleHistoryButtonAction(event : ActionEvent) : void
handleExportButtonAction(event : ActionEvent) : void
handleLogoutButtonAction(event : ActionEvent) : void
<<realize>>
BasicController
mapButton : Button
historyButton : Button
exportButton : Button
logoutButton : Button
user : User
stage : Stage
scene : Scene
handleMapButtonAction(event : ActionEvent) : void
handleHistoryButtonAction(event : ActionEvent) : void
handleExportButtonAction(event : ActionEvent) : void
handleLogoutButtonAction(event : ActionEvent) : void
AdminController
editButton : Button
stage : Stage
scene : Scene
handleEditButtonAction(event : ActionEvent) : void
EditController
INVALID : String
VALIDATED : String
backButton : Button
addNode : TextField
addWay : TextField
addUser : TextField
removeNode : TextField
removeWay : TextField
removeUser : TextField
editNode : TextField
editWay : TextField
editUser : TextField
stage : Stage
scene : Scene
user : Admin
handleBackButton(event : ActionEvent) : void
handleAddNode(event : ActionEvent) : void
handleAddWay(event : ActionEvent) : void
handleAddUser(event : ActionEvent) : void
handleEditNode(event : ActionEvent) : void
handleEditWay(event : ActionEvent) : void
handleEditUser(event : ActionEvent) : void
handleRemoveNode(event : ActionEvent) : void
handleRemoveWay(event : ActionEvent) : void
handleRemoveUser(event : ActionEvent) : void
PathController
backButton : Button
distanceText : Text
goFootButton : Button
footTimeText : Text
goCarButton : Button
carTimeText : Text
goBusButton : Button
busTimeText : Text
goBikeButton : Button
bikeTimeText : Text
pathGraph : LineChart<>
nodesGraph : ScatterGraph<>
map : Map
user : User
from : int
to : int
path : Iterable<Long>
timePathPoor : Iterable<Long>
timePathCar : Iterable<Long>
timePathBus : Iterable<Long>
timePathBike : Iterable<Long>
stage : Stage
scene : Scene
handleGoFootButton(event : ActionEvent) : void
handleGoCarButton(event : ActionEvent) : void
handleGoBusButton(event : ActionEvent) : void
handleGoBikeButton(event : ActionEvent) : void
<<realize>>
MapController
backButton : Button
fromTextField : TextField
toTextField : TextField
invalidText : Text
scatterChart : ScatterChart<>
lineChart : LineChart<>
stage : Stage
scene : Scene
map : Map
user : User
handleBackButton(event : ActionEvent) : void
handlePathButton(event : ActionEvent) : void
handleFromTextField(event : ActionEvent) : int
handleToTextField(event : ActionEvent) : int
HistoryController
backButton : Button
textArea : TextArea
user : User
stage : Stage
scene : Scene
handleBackButton(event : ActionEvent) : void
<<realize>>
<<realize>>
<<realize>>
<<interface>>
Initializable
initialize(location : URL,resources : ResourceBundle) : void