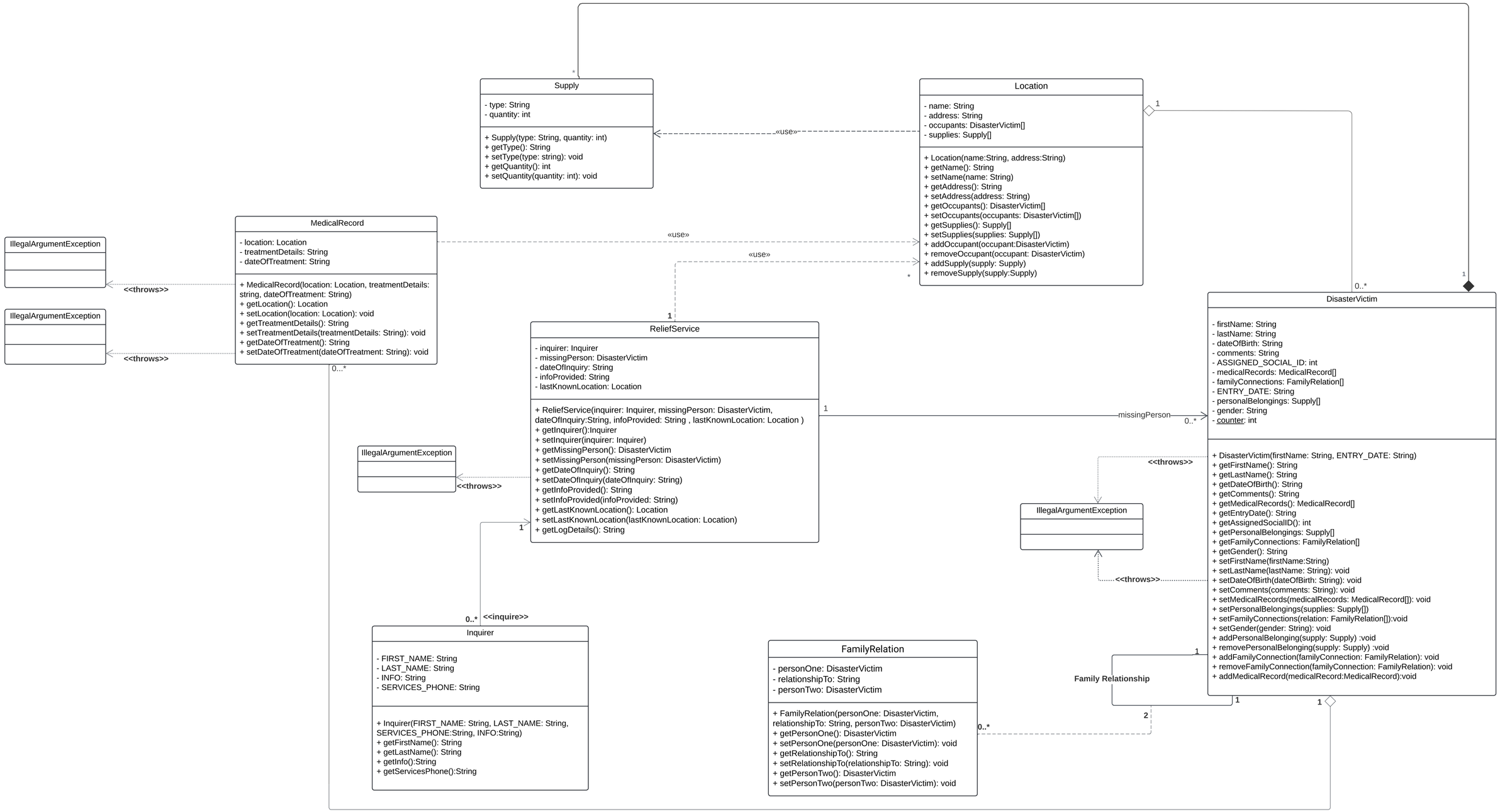
Supply
- type: String
- quantity: int
+ Supply(type: String, quantity: int)
+ getType(): String
+ setType(type: string): void
+ getQuantity(): int
+ setQuantity(quantity: int): void
Location
- name: String
- address: String
- occupants: DisasterVictim[]
- supplies: Supply[]
+ Location(name:String, address:String)
+ getName(): String
+ setName(name: String)
+ getAddress(): String
+ setAddress(address: String)
+ getOccupants(): DisasterVictim[]
+ setOccupants(occupants: DisasterVictim[])
+ getSupplies(): Supply[]
+ setSupplies(supplies: Supply[])
+ addOccupant(occupant:DisasterVictim)
+ removeOccupant(occupant: DisasterVictim)
+ addSupply(supply: Supply)
+ removeSupply(supply:Supply)
«use»
MedicalRecord
- location: Location
- treatmentDetails: String
- dateOfTreatment: String
+ MedicalRecord(location: Location, treatmentDetails: string, dateOfTreatment: String)
+ getLocation(): Location
+ setLocation(location: Location): void
+ getTreatmentDetails(): String
+ setTreatmentDetails(treatmentDetails: String): void
+ getDateOfTreatment(): String
+ setDateOfTreatment(dateOfTreatment: String): void
IllegalArgumentException
<<throws>>
IllegalArgumentException
<<throws>>
«use»
«use»
ReliefService
- inquirer: Inquirer
- missingPerson: DisasterVictim
- dateOfInquiry: String
- infoProvided: String
- lastKnownLocation: Location
+ ReliefService(inquirer: Inquirer, missingPerson: DisasterVictim, dateOfInquiry:String, infoProvided: String , lastKnownLocation: Location)
+ getInquirer():Inquirer
+ setInquirer(inquirer: Inquirer)
+ getMissingPerson(): DisasterVictim
+ setMissingPerson(missingPerson: DisasterVictim)
+ getDateOfInquiry(): String
+ setDateOfInquiry(dateOfInquiry: String)
+ getInfoProvided(): String
+ setInfoProvided(infoProvided: String)
+ getLastKnownLocation(): Location
+ setLastKnownLocation(lastKnownLocation: Location)
+ getLogDetails(): String
missingPerson
DisasterVictim
- firstName: String
- lastName: String
- dateOfBirth: String
- comments: String
- ASSIGNED_SOCIAL_ID: int
- medicalRecords: MedicalRecord[]
- familyConnections: FamilyRelation[]
- ENTRY_DATE: String
- personalBelongings: Supply[]
- gender: String
- counter: int
+ DisasterVictim(firstName: String, ENTRY_DATE: String)
+ getFirstName(): String
+ getLastName(): String
+ getDateOfBirth(): String
+ getComments(): String
+ getMedicalRecords(): MedicalRecord[]
+ getEntryDate(): String
+ getAssignedSocialID(): int
+ getPersonalBelongings: Supply[]
+ getFamilyConnections: FamilyRelation[]
+ getGender(): String
+ setFirstName(firstName:String)
+ setLastName(lastName: String): void
+ setDateOfBirth(dateOfBirth: String): void
+ setComments(comments: String): void
+ setMedicalRecords(medicalRecords: MedicalRecord[]): void
+ setPersonalBelongings(supplies: Supply[])
+ setFamilyConnections(relation: FamilyRelation[]):void
+ setGender(gender: String): void
+ addPersonalBelonging(supply: Supply) :void
+ removePersonalBelonging(supply: Supply) :void
+ addFamilyConnection(familyConnection: FamilyRelation): void
+ removeFamilyConnection(familyConnection: FamilyRelation): void
+ addMedicalRecord(medicalRecord:MedicalRecord):void
<<throws>>
IllegalArgumentException
<<throws>>
IllegalArgumentException
<<throws>>
<<inquire>>
Inquirer
- FIRST_NAME: String
- LAST_NAME: String
- INFO: String
- SERVICES_PHONE: String
+ Inquirer(FIRST_NAME: String, LAST_NAME: String, SERVICES_PHONE:String, INFO:String)
+ getFirstName(): String
+ getLastName(): String
+ getInfo():String
+ getServicesPhone():String
FamilyRelation
- personOne: DisasterVictim
- relationshipTo: String
- personTwo: DisasterVictim
+ FamilyRelation(personOne: DisasterVictim, relationshipTo: String, personTwo: DisasterVictim)
+ getPersonOne(): DisasterVictim
+ setPersonOne(personOne: DisasterVictim): void
+ getRelationshipTo(): String
+ setRelationshipTo(relationshipTo: String): void
+ getPersonTwo(): DisasterVictim
+ setPersonTwo(personTwo: DisasterVictim): void
Family Relationship
1
0..*
1
2
1
1
0..*
1
0..*
1
0..*
1
0..*
1
1