**Relations of UML classes:**

Volunteer(voID, name, birthday, email, address, city\_id, travelReadinessTime)

Application(id, request\_id, volunteer\_id, modified, is\_valid) + (is\_accepted by a Transaction)

Request(reqID, title(=interestArea), beneficiary\_id, number\_of\_volunteers, priority, startDate, endDate, registerbyDate)

Skill(skillName, description)

InterestArea(interestName)

Beneficiary (benID, name, address, city\_id)

City(cityID, name, geolocation)

**Relations for associations:**

skill\_assignment(skillName, volID)

volunteer\_range(voID, cityID)

request\_location(request\_id, cityID)

interest\_assignment(voID, interestArea)

request\_skill(reqID, skillName, minimumNeedofppl, ImportanceValue)

All the functional dependencies below are non-trivial functional dependencies; meaning X -> Y where Y are not subsets of X

VoID -> name, birthday, email, address, travelReadinessTime

AppID -> reqID, voID, lastModifiedTS, valid

ReqID -> benID, interestArea, volNumNeeded, priority, startDate, endDate, registerbyDate

SkillName -> description

benID -> name, address

cityID -> name, geolocation

skillName, reqID -> minimumNeedofppl, ImportanceValue

The trivial dependencies are:

skillName,volID

voID, cityID

cityID, reqID

voID, interestArea

InterestArea ->

VoID -> name, birthday, email, address, travelReadinessTime

AppID -> **reqID**, **voID**, lastModifiedTS, valid, **benID**, interestArea, volNumNeeded, priority, startDate, endDate, registerbyDate, name, birthday, email, address, travelReadinessTime, name, address

ReqID -> benID, interestArea, volNumNeeded, priority, startDate, endDate, registerbyDate

benID -> name, address

SkillName -> description

cityID -> name, geolocation

skillName, reqID -> minimumNeedofppl, ImportanceValue

the dependent is strictly not a subset of the determinant