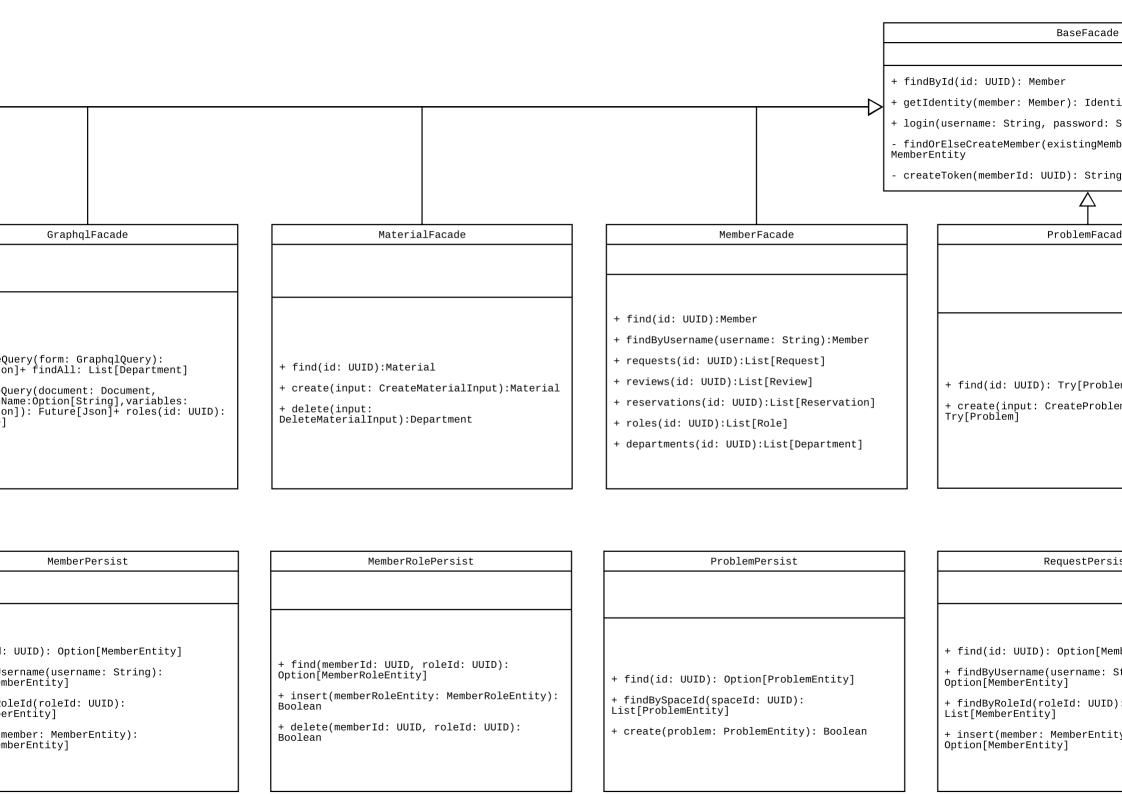
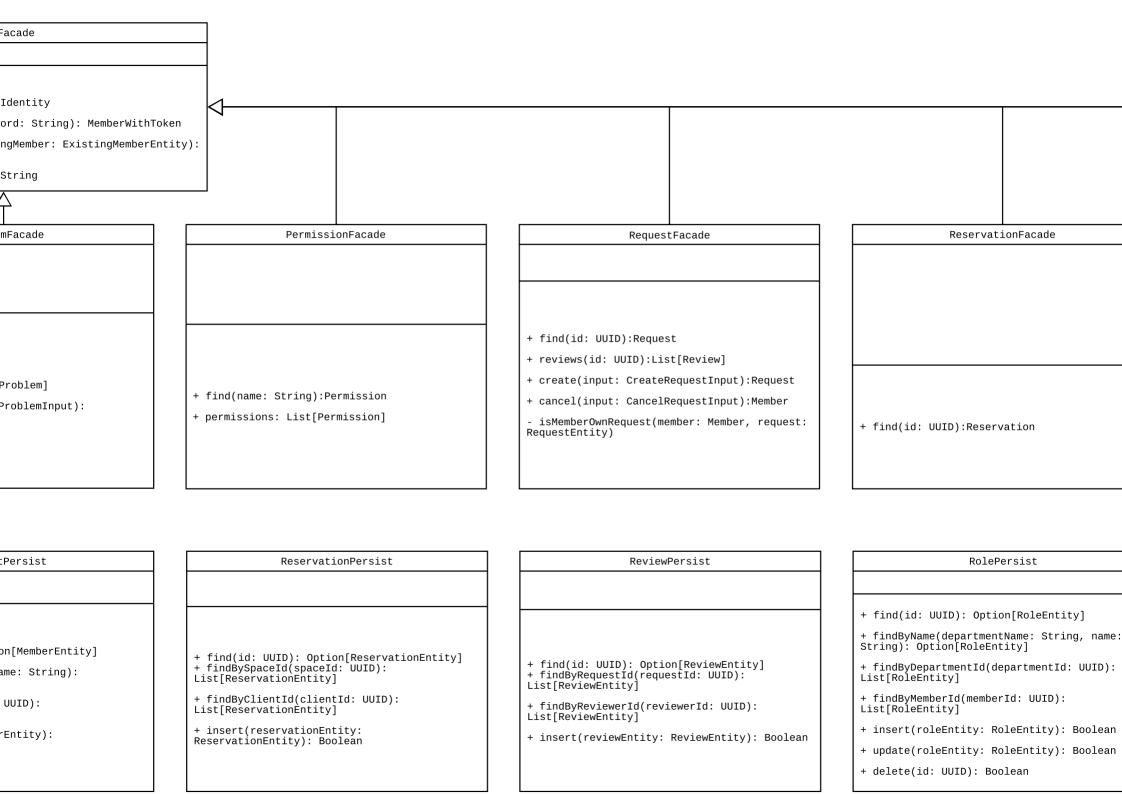


Boolean





RoleFacade ReviewFacade SpaceFacade + find(id: UUID): Try[Space] + find(id: UUID): Trv[Review] + findByName(department: String, name: + find(id: UUID): Try[Role] String):Space + create(input: CreateReviewInput):Review + findByName(department: String, + searchByName(name: String):List[Space] name: String): Roľe - updateRequestStatus(requestId: UUID): Boolean + findAll: List[Space] + members(id: UUID): List[Member] - updateReservation(requestId: UUID): Boolean + requests(space: Space): List[Request] + create(input: CreateRoleInput):Role - findRequest(requestId: UUID):RequestEntity + reservations(space: Space): + update(input: UpdateRoleInput):Role List[Reservation] - createReview(reviewEntity: ReviewEntity): ReviewEntity + delete(input: DeleteRoleInput):Department + problems(space: Space):List[Problem] - createReservations(requestEntity: + assign(input: AssignRoleInput):Role + create(input: CreateSpaceInput):Space RequestEntity): Boolean + revoke(input: RevokeRoleInput):Role + update(input: UpdateSpaceInput):Space - requiredApproval: Int - isSpaceNameValid(name: String):Boolean SearchPersist SearchPersist

name: UID): olean

olean

+ space(query: String, department: Option[String], tags: List[String], capacity: Option[Int]): List[SpaceEntity]

```
SearchPersist

+ find(id: UUID): Option[SpaceEntity]
+ searchByName(name: String):
List[SpaceEntity]
+ findByName(departmentName: String,
spaceName: String): Option[SpaceEntity]
+ findByDepartmentId(departmentId: UUID):
List[SpaceEntity]
+ findAll: List[SpaceEntity]
+ insert(space: SpaceEntity): Boolean
+ update(space: SpaceEntity): Boolean
+ delete(id: UUID): Boolean
```