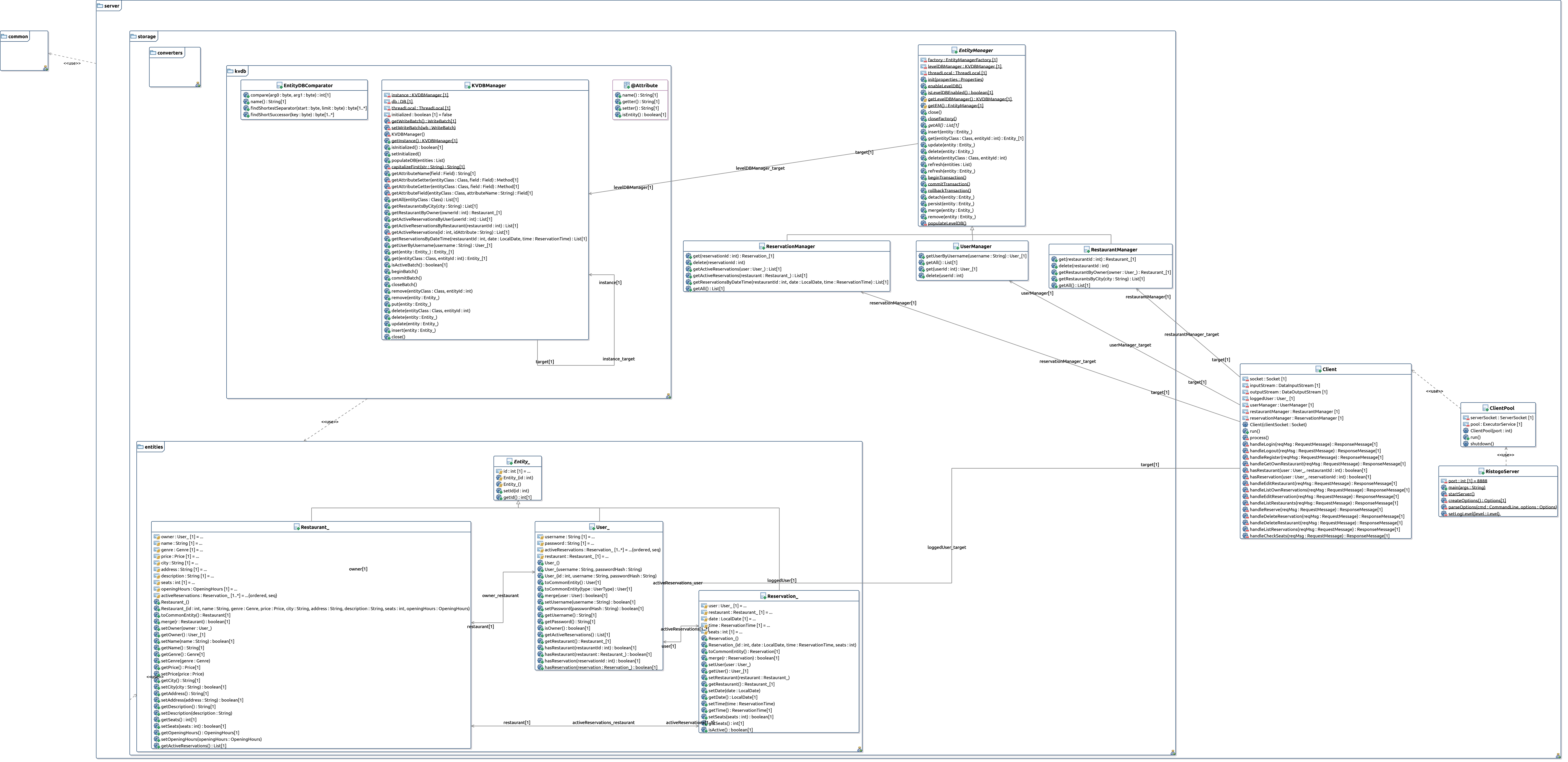
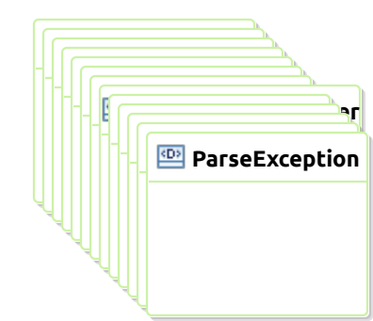


server
common
<<use>>
storage
converters
kvdb
EntityDBComparator
compare(arg0 : byte, arg1 : byte) : int[1]
name() : String[1]
findShortestSeparator(start : byte, limit : byte) : byte[1..*]
findShortSuccessor(key : byte) : byte[1..*]
KVDBManager
instance : KVDBManager [1]
db : DB [1]
threadLocal : ThreadLocal [1]
initialized : boolean [1] = false
getWriteBatch() : WriteBatch[1]
setWriteBatch(wb : WriteBatch)
KVDBManager()
getInstance() : KVDBManager[1]
isInitialized() : boolean[1]
setInitialized()
populateDB(entities : List)
capitalizeFirst(str : String) : String[1]
getAttributeName(field : Field) : String[1]
getAttributeSetter(entityClass : Class, field : Field) : Method[1]
getAttributeGetter(entityClass : Class, field : Field) : Method[1]
getAttributeField(entityClass : Class, attributeName : String) : Field[1]
getAll(entityClass : Class) : List[1]
getRestaurantsByCity(city : String) : List[1]
getRestaurantByOwner(ownerId : int) : Restaurant_[1]
getActiveReservationsByUser(userId : int) : List[1]
getActiveReservationsByRestaurant(restaurantId : int) : List[1]
getActiveReservations(id : int, idAttribute : String) : List[1]
getReservationsByDateTime(restaurantId : int, date : LocalDate, time : ReservationTime) : List[1]
getUserByUsername(username : String) : User_[1]
get(entity : Entity_) : Entity_[1]
get(entityClass : Class, entityId : int) : Entity_[1]
isActiveBatch() : boolean[1]
beginBatch()
commitBatch()
closeBatch()
remove(entityClass : Class, entityId : int)
remove(entity : Entity_)
put(entity : Entity_)
delete(entityClass : Class, entityId : int)
delete(entity : Entity_)
update(entity : Entity_)
insert(entity : Entity_)
close()
@Attribute
name() : String[1]
getter() : String[1]
setter() : String[1]
isEntity() : boolean[1]
instance[1]
target[1]
instance_target
EntityManager
factory : EntityManagerFactory [1]
levelDBManager : KVDBManager [1]
threadLocal : ThreadLocal [1]
init(properties : Properties)
enableLevelDB()
isLevelDBEnabled() : boolean[1]
getLevelDBManager() : KVDBManager[1]
getEM() : EntityManager[1]
close()
closeFactory()
getAll() : List[1]
insert(entity : Entity_)
get(entityClass : Class, entityId : int) : Entity_[1]
update(entity : Entity_)
delete(entity : Entity_)
delete(entityClass : Class, entityId : int)
refresh(entities : List)
refresh(entity : Entity_)
beginTransaction()
commitTransaction()
rollbackTransaction()
detach(entity : Entity_)
persist(entity : Entity_)
merge(entity : Entity_)
remove(entity : Entity_)
populateLevelDB()
target[1]
levelDBManager_target
levelDBManager[1]
ReservationManager
get(reservationId : int) : Reservation_[1]
delete(reservationId : int)
getActiveReservations(user : User_) : List[1]
getActiveReservations(restaurant : Restaurant_) : List[1]
getReservationsByDateTime(restaurantId : int, date : LocalDate, time : ReservationTime) : List[1]
getAll() : List[1]
UserManager
getUserByUsername(username : String) : User_[1]
getAll() : List[1]
get(userId : int) : User_[1]
delete(userId : int)
RestaurantManager
get(restaurantId : int) : Restaurant_[1]
delete(restaurantId : int)
getRestaurantByOwner(owner : User_) : Restaurant_[1]
getRestaurantsByCity(city : String) : List[1]
getAll() : List[1]
reservationManager[1]
userManager[1]
restaurantManager[1]
restaurantManager_target
userManager_target
reservationManager_target
target[1]
target[1]
target[1]
<<use>>
entities
Entity_
id : int [1] = ...
Entity_(id : int)
Entity_()
setId(id : int)
getId() : int[1]
Restaurant_
owner : User_ [1] = ...
name : String [1] = ...
genre : Genre [1] = ...
price : Price [1] = ...
city : String [1] = ...
address : String [1] = ...
description : String [1] = ...
seats : int [1] = ...
openingHours : OpeningHours [1] = ...
activeReservations : Reservation_ [1..*] = ...{ordered, seq}
Restaurant_()
Restaurant_(id : int, name : String, genre : Genre, price : Price, city : String, address : String, description : String, seats : int, openingHours : OpeningHours)
toCommonEntity() : Restaurant[1]
merge(r : Restaurant) : boolean[1]
setOwner(owner : User_)
getOwner() : User_[1]
setName(name : String) : boolean[1]
getName() : String[1]
getGenre() : Genre[1]
setGenre(genre : Genre)
getPrice() : Price[1]
setPrice(price : Price)
getCity() : String[1]
setCity(city : String) : boolean[1]
getAddress() : String[1]
setAddress(address : String) : boolean[1]
getDescription() : String[1]
setDescription(description : String)
getSeats() : int[1]
setSeats(seats : int) : boolean[1]
getOpeningHours() : OpeningHours[1]
setOpeningHours(openingHours : OpeningHours)
getActiveReservations() : List[1]
<<use>>
owner[1]
owner_restaurant
restaurant[1]
User_
username : String [1] = ...
password : String [1] = ...
activeReservations : Reservation_ [1..*] = ...{ordered, seq}
restaurant : Restaurant_ [1] = ...
User_()
User_(username : String, passwordHash : String)
User_(id : int, username : String, passwordHash : String)
toCommonEntity() : User[1]
toCommonEntity(type : UserType) : User[1]
merge(user : User) : boolean[1]
setUsername(username : String) : boolean[1]
setPassword(passwordHash : String) : boolean[1]
getUsername() : String[1]
getPassword() : String[1]
isOwner() : boolean[1]
getActiveReservations() : List[1]
getRestaurant() : Restaurant_[1]
hasRestaurant(restaurantId : int) : boolean[1]
hasRestaurant(restaurant : Restaurant_) : boolean[1]
hasReservation(reservationId : int) : boolean[1]
hasReservation(reservation : Reservation_) : boolean[1]
activeReservations_user
loggedUser[1]
activeReservations[1..*]
user[1]
Reservation_
user : User_ [1] = ...
restaurant : Restaurant_ [1] = ...
date : LocalDate [1] = ...
time : ReservationTime [1] = ...
seats : int [1] = ...
Reservation_()
Reservation_(id : int, date : LocalDate, time : ReservationTime, seats : int)
toCommonEntity() : Reservation[1]
merge(r : Reservation) : boolean[1]
setUser(user : User_)
getUser() : User_[1]
setRestaurant(restaurant : Restaurant_)
getRestaurant() : Restaurant_[1]
setDate(date : LocalDate)
getDate() : LocalDate[1]
setTime(time : ReservationTime)
getTime() : ReservationTime[1]
setSeats(seats : int) : boolean[1]
getSeats() : int[1]
isActive() : boolean[1]
restaurant[1]
activeReservations_restaurant
activeReservations[1..*]
loggedUser_target
target[1]
Client
socket : Socket [1]
inputStream : DataInputStream [1]
outputStream : DataOutputStream [1]
loggedUser : User_ [1]
userManager : UserManager [1]
restaurantManager : RestaurantManager [1]
reservationManager : ReservationManager [1]
Client(clientSocket : Socket)
run()
process()
handleLogin(reqMsg : RequestMessage) : ResponseMessage[1]
handleLogout(reqMsg : RequestMessage) : ResponseMessage[1]
handleRegister(reqMsg : RequestMessage) : ResponseMessage[1]
handleGetOwnRestaurant(reqMsg : RequestMessage) : ResponseMessage[1]
hasRestaurant(user : User_, restaurantId : int) : boolean[1]
hasReservation(user : User_, reservationId : int) : boolean[1]
handleEditRestaurant(reqMsg : RequestMessage) : ResponseMessage[1]
handleListOwnReservations(reqMsg : RequestMessage) : ResponseMessage[1]
handleEditReservation(reqMsg : RequestMessage) : ResponseMessage[1]
handleListRestaurants(reqMsg : RequestMessage) : ResponseMessage[1]
handleReserve(reqMsg : RequestMessage) : ResponseMessage[1]
handleDeleteReservation(reqMsg : RequestMessage) : ResponseMessage[1]
handleDeleteRestaurant(reqMsg : RequestMessage) : ResponseMessage[1]
handleListReservations(reqMsg : RequestMessage) : ResponseMessage[1]
handleCheckSeats(reqMsg : RequestMessage) : ResponseMessage[1]
<<use>>
ClientPool
serverSocket : ServerSocket [1]
pool : ExecutorService [1]
ClientPool(port : int)
run()
shutdown()
<<use>>
RistogoServer
port : int [1] = 8888
main(args : String)
startServer()
createOptions() : Options[1]
parseOptions(cmd : CommandLine, options : Options)
setLogLevel(level : Level)



ParseException