core (Server)

commandProcessor: CommandProcessor ioManager: myio.IOManager selector: String 'Selector? serverSocketChannel: String 'ServerSocketChannel? clientReadStates: String 'Map<SocketChannel, ObjectReaderState> clientWriteQueues: String 'Map<SocketChannel, ArrayDeque<ByteBuffer>> requestProcessingPool: String 'ForkJoinPool responsePreparationPool: String 'ForkJoinPool running: Boolean startServer(port: Int) handleAccept() handleRead() handleWrite() cleanupClient() adminConsoleLoop() shutdownServerInternals() **C** CommandProcessor ioManagerForLogging: myio.IOManager commandsList: String 'Map<String, CommandInterface> userDAO: db.UserDAO vehicleService: VehicleService activeUserSessions: String 'Map<SocketChannel, User> processCommand(request: common.Request): common.Response getCommandDescriptors(): List<common.CommandDescriptor> vehicleDAO: db.VehicleDAO vehiclesCache: String 'Map<Int, Vehicle> ' ConcurrentHashMap addVehicle(vehicle: model.Vehicle, userId: Int): model.Vehicle? updateVehicleById(id: Int, vehicle: model.Vehicle, userId: Int?): Boolean removeVehicle(id: Int, userId: Int): Boolean getVehicleById(id: Int): model.Vehicle? getAll(): List<model.Vehicle> clearUserVehicles(userId: Int) size(): Int size(): Int isEmpty(): Boolean getMax(characteristic: String): model.Vehicle? getMin(characteristic: String): model.Vehicle? filterByCharacteristic(characteristic: String, arg: String): List<model.Vehicle> findByCharacteristic(characteristic: String, arg: String): model.Vehicle? deleteByNumber(number: Int, userId: Int): Boolean

 getConnection(): String 'java.sql.Connection
 initializeDatabase()
 addUser(username: String, password: String): model.User?
 findUserByUsername(username: String): model.User?
 verifiPassword(user: model.User, password: String): Boolean vehicle: model.Vehicle?username: String?password: String? main() ' Or equivalent entry point C VehicleDAO addVehicle(vehicle: model.Vehicle, userld: Int): model.Vehicle?
 updateVehicle(vehicle: model.Vehicle): Boolean
 deleteVehicle(vehicleld: Int, userld: Int): Boolean
 getVehicleById(vehicleld: Int): model.Vehicle?
 getAllVehicles(): List<model.Vehicle>

Server Application Class Diagram (Kotlin - Simplified Types)

