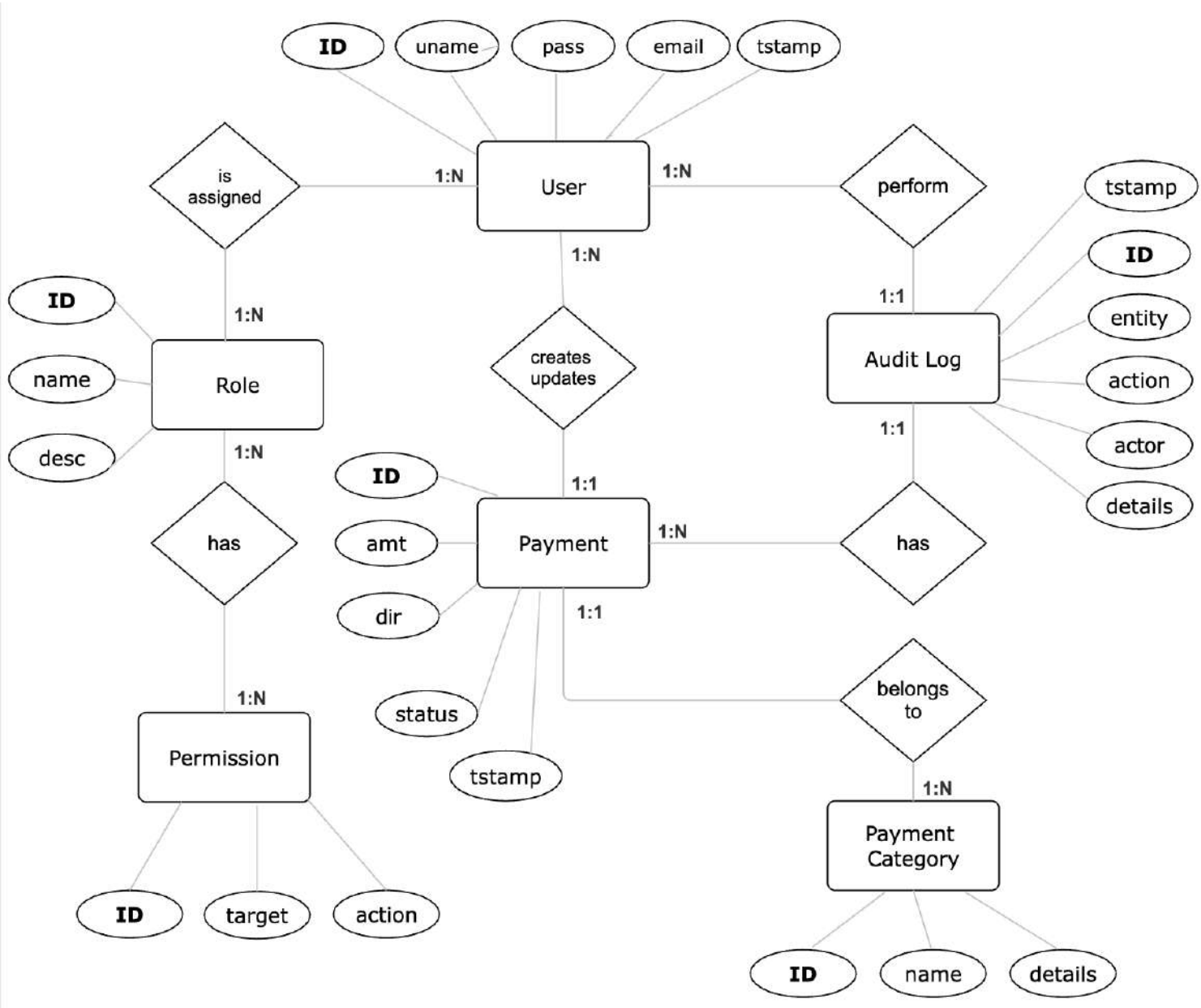
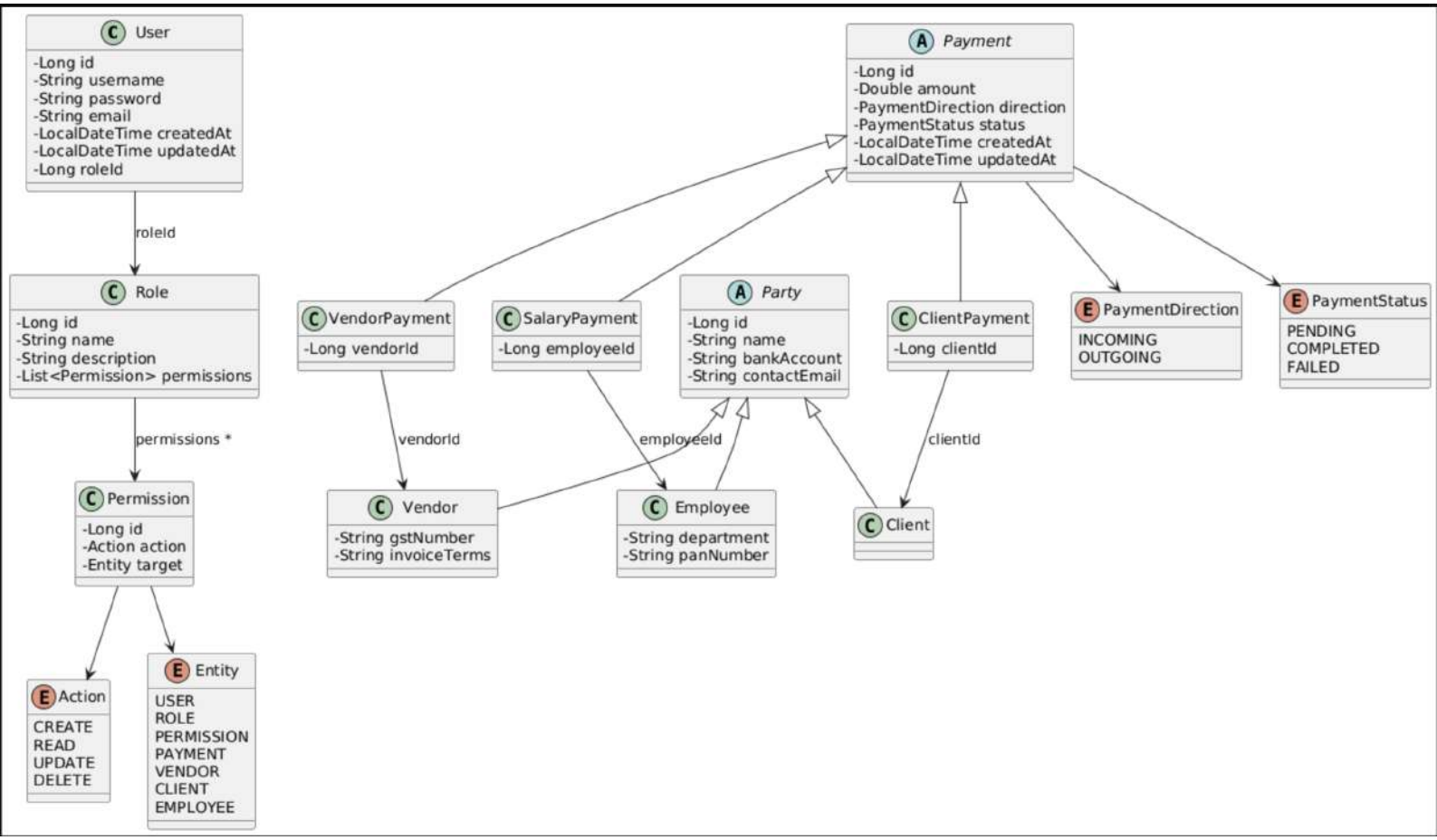
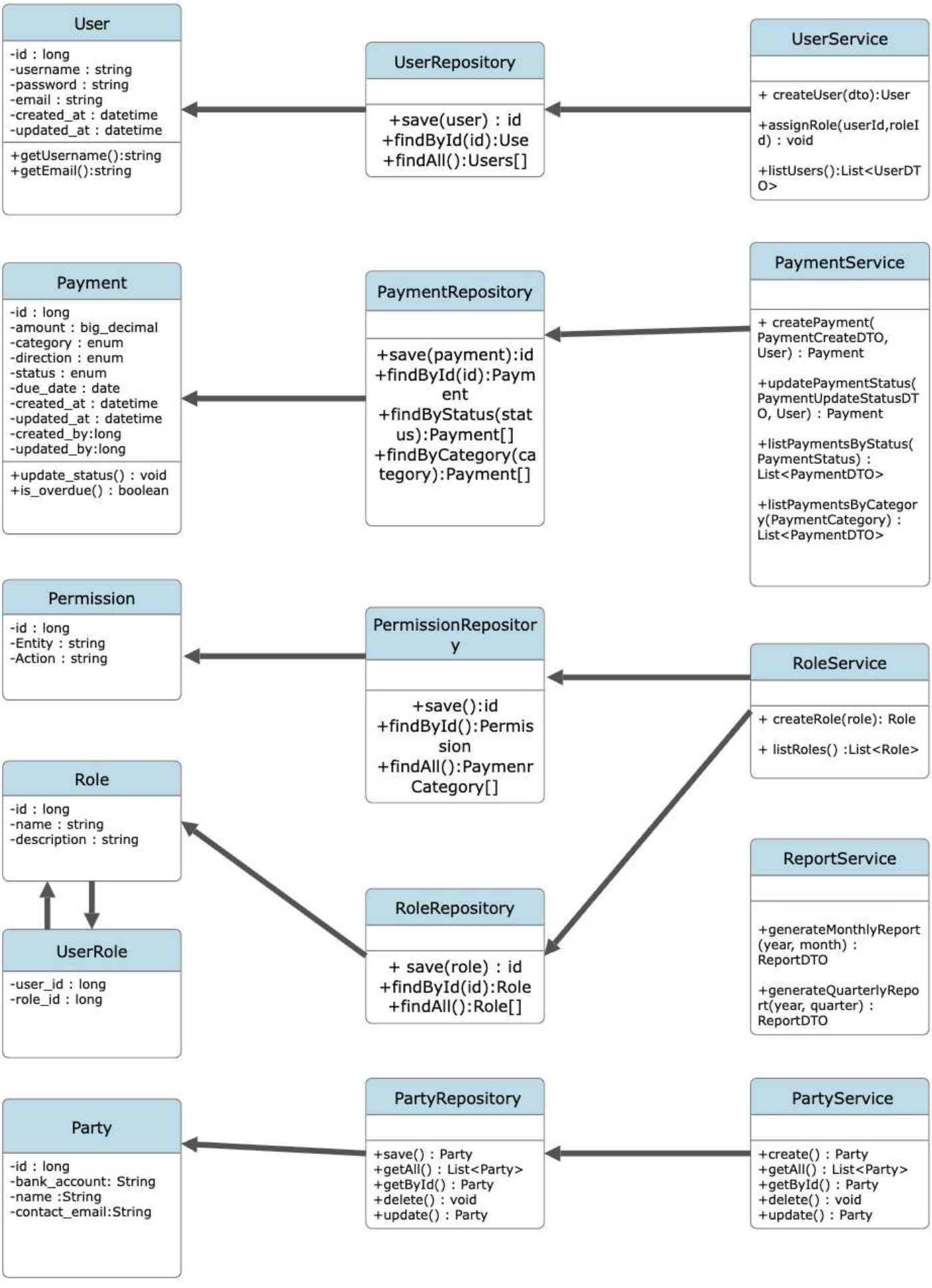
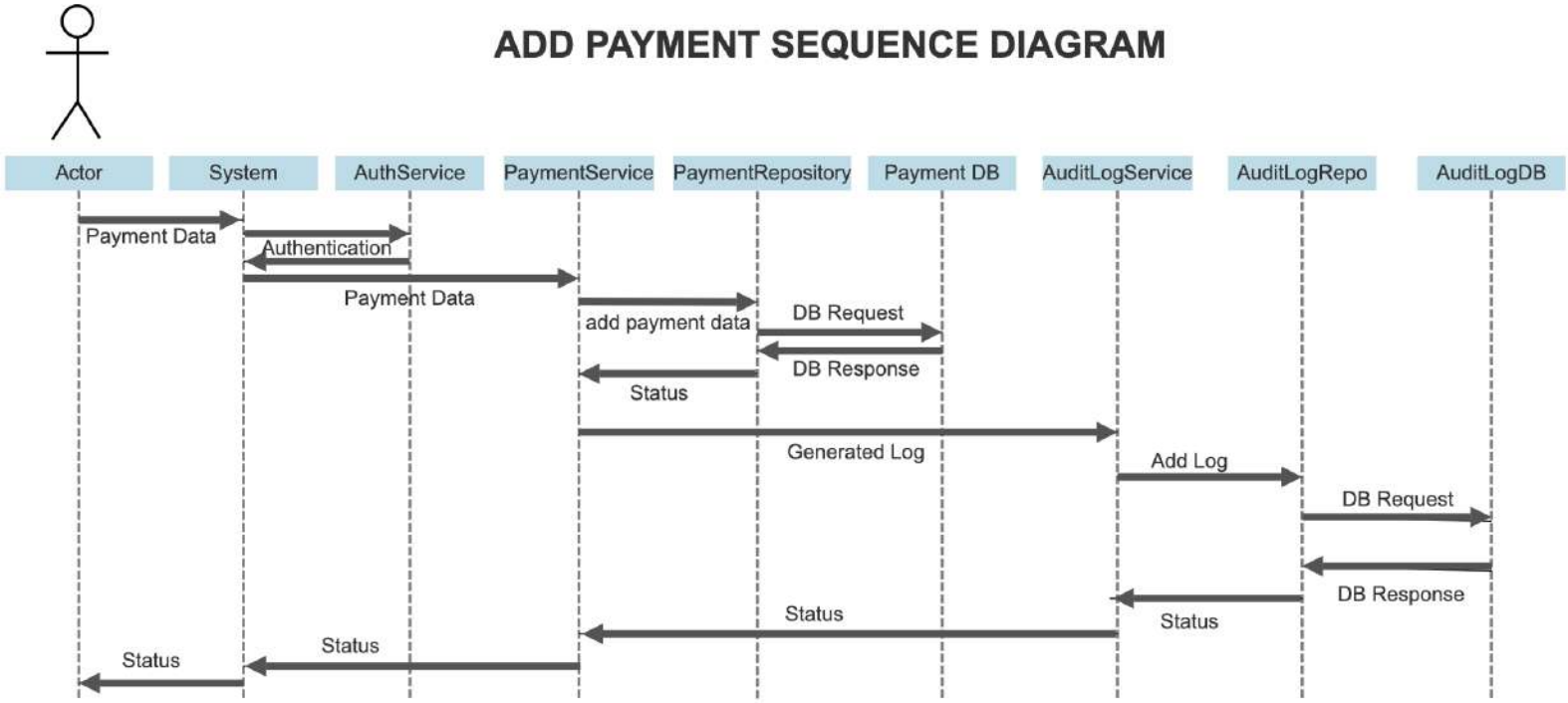

ID
uname
pass
email
tstamp
is assigned
1:N
User
1:N
perform
tstamp
ID
entity
action
actor
details
1:N
1:N
1:1
ID
name
desc
Role
creates updates
Audit Log
1:N
1:1
ID
amt
dir
Payment
1:N
has
has
1:1
1:N
status
tstamp
belongs to
1:N
Permission
Payment Category
ID
target
action
ID
name
details

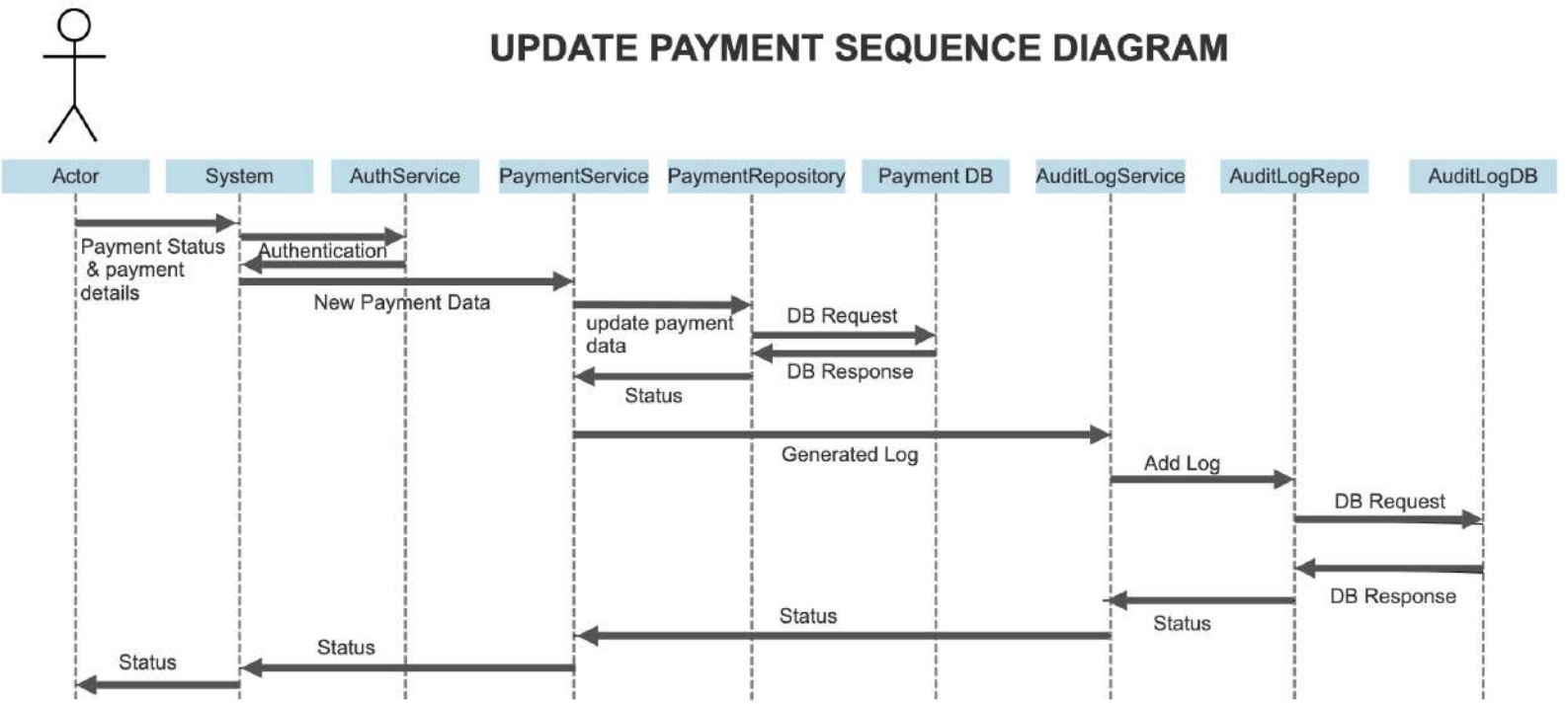
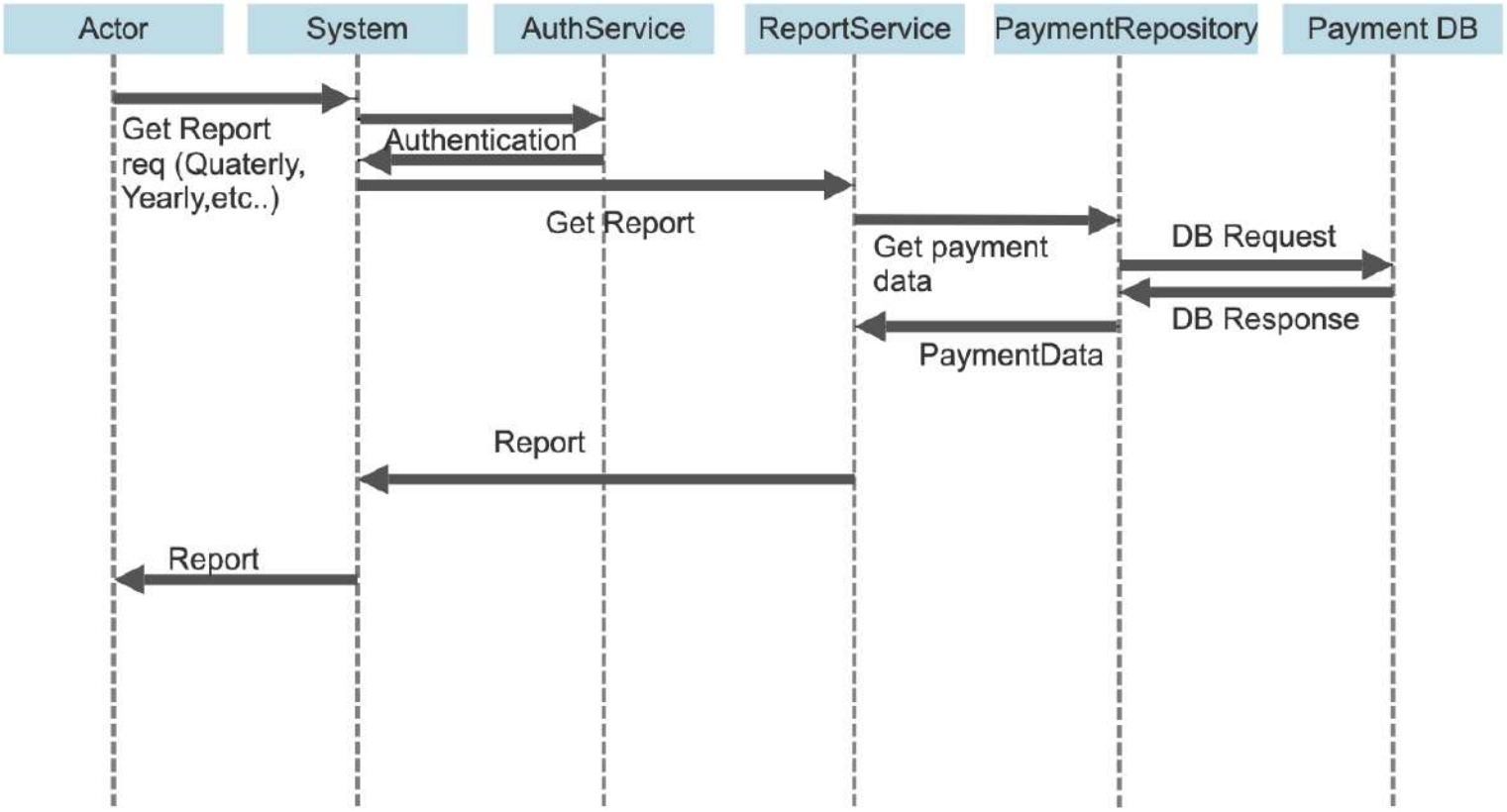

User
-Long id
-String username
-String password
-String email
-LocalDateTime createdAt
-LocalDateTime updatedAt
-Long roleId
roleId
Role
-Long id
-String name
-String description
-List<Permission> permissions
permissions *
Permission
-Long id
-Action action
-Entity target
Action
CREATE
READ
UPDATE
DELETE
Entity
USER
ROLE
PERMISSION
PAYMENT
VENDOR
CLIENT
EMPLOYEE
Payment
-Long id
-Double amount
-PaymentDirection direction
-PaymentStatus status
-LocalDateTime createdAt
-LocalDateTime updatedAt
VendorPayment
-Long vendorId
SalaryPayment
-Long employeeId
Party
-Long id
-String name
-String bankAccount
-String contactEmail
ClientPayment
-Long clientId
PaymentDirection
INCOMING
OUTGOING
PaymentStatus
PENDING
COMPLETED
FAILED
vendorId
employeeId
clientId
Vendor
-String gstNumber
-String invoiceTerms
Employee
-String department
-String panNumber
Client


User
-id : long
-username : string
-password : string
-email : string
-created_at : datetime
-updated_at : datetime
+getUsername():string
+getEmail():string
UserRepository
+save(user) : id
+findById(id):Use
+findAll():Users[]
UserService
+ createUser(dto):User
+assignRole(userId,roleId) : void
+listUsers():List<UserDTO>
Payment
-id : long
-amount : big_decimal
-category : enum
-direction : enum
-status : enum
-due_date : date
-created_at : datetime
-updated_at : datetime
-created_by:long
-updated_by:long
+update_status() : void
+is_overdue() : boolean
PaymentRepository
+save(payment):id
+findById(id):Payment
+findByStatus(status):Payment[]
+findByCategory(category):Payment[]
PaymentService
+ createPayment(PaymentCreateDTO, User) : Payment
+updatePaymentStatus(PaymentUpdateStatusDTO, User) : Payment
+listPaymentsByStatus(PaymentStatus) : List<PaymentDTO>
+listPaymentsByCategory(PaymentCategory) : List<PaymentDTO>
Permission
-id : long
-Entity : string
-Action : string
PermissionRepository
+save():id
+findById():Permission
+findAll():PaymenrCategory[]
RoleService
+ createRole(role): Role
+ listRoles() :List<Role>
Role
-id : long
-name : string
-description : string
UserRole
-user_id : long
-role_id : long
RoleRepository
+ save(role) : id
+findById(id):Role
+findAll():Role[]
ReportService
+generateMonthlyReport(year, month) : ReportDTO
+generateQuarterlyReport(year, quarter) : ReportDTO
Party
-id : long
-bank_account: String
-name :String
-contact_email:String
PartyRepository
+save() : Party
+getAll() : List<Party>
+getById() : Party
+delete() : void
+update() : Party
PartyService
+create() : Party
+getAll() : List<Party>
+getById() : Party
+delete() : void
+update() : Party

ADD PAYMENT SEQUENCE DIAGRAM



UPDATE PAYMENT SEQUENCE DIAGRAM



GENERATE REPORT SEQUENCE DIAGRAM