

Server Side

Client Side

partner-ear

partner-api

AuditTO

- id: Integer
- partnerCode: Integer
- operation: String
- timeStamp: LocalDateTime

+ AuditTO()
+ AuditTO(partnerCode: Integer, operation: String, timeStamp: LocalDateTime)

<<interface>> PartnerService

+ savePartner(partner: PartnerTO): PartnerTO
+ getPartner(partnerCode: Integer): PartnerTO
+ listAllPartners(): List<PartnerTO>
+ updatePartner(partner: PartnerTo): PartnerTO

<<interface>> Partner

+ savePartner(partner: PartnerTO): PartnerTO
+ getPartner(partnerCode: Integer): PartnerTO
+ listAllPartners(): List<PartnerTO>
+ updatePartner(partner: PartnerTo): PartnerTO

PartnerTO

- partnerCode: Integer
- name: String
- phone: String
- email: String
- rating: Integer

+ PartnerTO()
+ PartnerTO(partnerCode: Integer, name: String, phone: String, email: String, rating: Integer)

<<interface>> Audit

+ saveRegister(audit: AuditTO): void

<<interface>> AuditLocal

<<interface>> AuditRemote

<<interface>> PartnerLocal

<<interface>> PartnerRemote

partner-ejb

AuditBean

- em: EntityManager

+ saveRegister(audit: AuditTO): void

PartnerQueueSender

- log: Loggger
- queue: Queue
- connectionFactory: ConnectionFactory

+ sendObjectMessage(audit: AuditTo): void

Audit

- id: Integer
- partnerCode: Integer
- operation: String
- timeStamp: LocalDateTime

+ Audit()
+ Audit(partnerCode: Integer, operation: String, timeStamp: LocalDateTime)

PartnerBean

- log: Loggger
- queueSender: PartnerQueueSender

+ savePartner(partner: PartnerTO): PartnerTO
+ getPartner(partnerCode: Integer): PartnerTO
+ listAllPartners(): List<PartnerTO>
+ updatePartner(partner: PartnerTo): PartnerTO
- toCollection(partnerList: List<Partner>): List<partnerTO>

Partner

- partnerCode: Integer
- name: String
- phone: String
- email: String
- rating: Integer

+ Partner()
+ Partner(partnerCode: Integer, name: String, phone: String, email: String, rating: Integer)

partner-impl

PartnerServiceImpl

- partnerBean: PartnerLocal

+ savePartner(partner: PartnerTO): PartnerTO
+ getPartner(partnerCode: Integer): PartnerTO
+ listAllPartners(): List<PartnerTO>
+ updatePartner(partner: PartnerTo): PartnerTO

partner-web

RestApplication

+ getClasses(): Set

PartnerServlet

- log: Loggger

doGet(req: HttpServletRequest, resp: HttpServletResponse): void

partner-mdb

PartnerQueueMDB

- log: Loggger
- mdc: MessageDrivenContext
- auditBean: AuditLocal
- em: EntityManager

+ onMessage(message: Message): void

partner-app-client

PartnerClient

- REST_URI_POST: String = "http://localhost:8080/dm110-web/api/partners/new"
- REST_URI_GET_OR_UPDATE_ONE: String = "http://localhost:8080/dm110-web/api/partners/partner"
- REST_URI_GET_ALL: String = "http://localhost:8080/dm110-web/api/partners/all"

+ main(args: Array<String>): void
- createPartner(partner01: PartnerTO): PartnerTO
- getPartner(partnerCode: Integer): PartnerTO
- updatePartner(partner00: PartnerTO): PartnerTO
- getAllPartners(): List<PartnerTO>

PartnerEJBClient

+ main(args: Array<String>): void
- invokeStatelessBean(): void
- lookupStatelessPartner(): PartnerRemote
- createInitialContext(): Context