Javier López de la Nieta Luján
Daniel Varela Sánchez
Guillermo Martin-Coello Juárez

javier.lopezdelanieta@estudiante.uam.es
danielo.varelas@estudiante.uam.es
guillermo.martin-coello@estudiante.uam.es

PADSOF:DELIEVERABLE 3(PART 1)

Application: **Theaception**

Javier López de la Nieta Luján
Daniel Varela Sánchez
Guillermo Martin-Coello Juárez

javier.lopezdelanieta@estudiante.uam.es
danielo.varelas@estudiante.uam.es
guillermo.martin-coello@estudiante.uam.es

TRACEABILITY MATRIX

	searchEvent	LogIn	Register	bookTicket	checkBooking	cancelBooking	payTicketWithCard	selectTicket	buyPassWithCard	purchaseBooked	getTickets	subscribeToWaitlist	unsusbscribe to Waitlist	cancelPerformance
Search events	x													
Log in		x												
Register			x											
Book seat				x	x	x								
Purchase seat							x							
Choose seat								x						
Purchase annual pass									x					
Purchase cycle pass									x					
pay reservations										x				
See tickets											x			
Apply to a waitlist												x		
Leave the waitlist													x	
Cancel performance														x
Create event														

	createPerformance	postponePerformance	cancelPerformance	createStanding	createSitting	createFragmented	initializePrices	setPrice	setOccupancy	createCycle	maintainSeat	setTTB	setTTP	getStatistics	updateStatistics	cypher	deCypher	generateCode	save	load
Create performance	x																			
Postpone performance		x																		
Cancel event			x																	
Set zones				x	x	x														
Set event prices							x	x												
Set event capacity									x											
Create cycles										x										
Enable and disable seats											x									
Limit booking time												x								
Limit payment time after booking													x							
See occupancy statistics														x						
See revenue statistics															x					
Cyfrated passwords																x	x			
Unique code por the tickets																		x		
Persistence																			x	x

Javier López de la Nieta Luján
Daniel Varela Sánchez
Guillermo Martin-Coello Juárez

javier.lopezdelanieta@estudiante.uam.es
danielo.varelas@estudiante.uam.es
guillermo.martin-coello@estudiante.uam.es

CLASS DIAGRAM

FragmentedZone
+addSubzone(subzone:Zone):void
+removeSubZone(zone:Zone):void
+getAttendees():int
+getNMaxPeople():int
+getNPosPeople():int
+isOccupied(zone:Zone,seat:Seat):boolean
+isOccupied(zone:Zone):boolean
+occupy(zone:Zone):void
+cloneZone():Zone

StandingZone
- nMaxPeople: int
- nPeople: int
+getAttendees():int
+getNPosPeople():void
+isOccupied():boolean
+reduceNPeople(n:int):void
+occupy():void
+free():void
+cloneZone():Zone

SittingZone
- nRows: int
- nColumns: int
- occupiedSeats:double
+chooseOccupancy(p: int): void
+getAttendees():int
+getNMaxPeople():int
+getNPosPeople():int
+resetSeatsAvailability():void
+fullfilMatrix():void
+isAvailable(row:int,column:int):bo
+occupy(Seat seat):void
+free(seat:Seat):void
+isOccupied(Seat seat):boolean
+cloneZone():Zone

Seat
-occupied:bool
-nRow: int
-nColumn: int
-available: boolean
+ mantainSeat(d: Date, notes: String): void
+ habilitateSeat(): void

ManteincanceSeat
-endDate:LocalDate
-notes: String
+maintainedSeatVigent():boolean

ZoneStatistic
-all:double
-oc:double
-max:int
+updateStatisticsFromPerformance(event:Event,performance:Performance,zone:Zone):void
+updateStatisticsFromEvent(event:Event,zone:Zone):void
+updateStatistics(List<Event>,zone:Zone):void

«abstract» Statistic
-name: String
-occupancy: double
-profits:double
+calculateProfit(zone:Zone,p:Performance):d

PerformanceStatistic
+updateStatistics(event:Event):void

EventStatistic
+updateStatistics(event:Event):void

«abstract» Zone
-name:String
-occupancy:int
+getAttendees():int
+getNPosPeople():int
+getNMaxPeople():int
+cloneZone():Zone

TheaterApp
- ttb: int
- ttp: int
+payTicketWithCard(c:Customer,e:Event,t:Ticket,n:String):boolean
+payPassWithCard(c:Customer,p:Pass,card:String):boolean
+logIn(username:String,password:String):boolean
+register(username:String,password:String):boolean
+payWithPass(c:Customer,p:Pass,e:Event,t:Ticket):boolean
+searchEvent(type:int):ArrayList<Event>
+createDanceEvent(t:String,a:String,di:String,de:String,da:String,o:String):void
+createPlayEvent(t:String,a:String,di:String,de:String,de:String,):void
+createConcertEvent(t:String,a:String,di:String,de:String,mp:String,o:String):void
+cancelEvent(event:Event):void
+createPerformance(event:Event,date:LocalDateTime):void
+cancelPerformance(performance:Performance):void
+createCycle(title:String):void
+createCyclePass(name:String,discount:int,zone:Zone,cycle:Zone):void
+createAnnualPass(c:Customer,zone:Zone):void
+bookTicket(c:Customer,t:Ticket):void

Cycle
+addEvent(e: Event):void
+calculateSumPrice(zoneZone):double

Pass
-arleadyAttended:ArrayList<Event>
+getIfHasAlreadyAttended():boolean

AnnualPass
-price:int
-startDate:Date
-endDate:Date
+checkIfValidAnnualPass():boolean

CyclePass
-name:String
-totalPrice:int
-discountPertcentage: int
-canGoTo:ArrayList<Event>
+calcuclateDiscountedPrice():double

Performance
-LocalDateTime: Date
-availableSeats: boolean
+postponePerformance(date: LocalDateTime): void
+checkOccupied(zone:Zone,seat:Seat):boolean
+checkOccupied(zone:Zone):boolean
+selectTicket(zone: Zone, seat: Seat): Ticket
+selectTicketWithString(zone:String,nRow:int, nCol:int):Ticket
+selectTicketWithString(zone:String):Ticket
+cancelPerformanceTicket(t:Ticket):void
+checkIfFull():boolean

Price
-price:int

«abstract» Event
-title:String
-author:String
-director:String
-description:String
+cancelPerformance(performance: Performance): void
+addPerformance(performance: Performance):void
+addPrice(p: Price):void
+createPerformance(date: LocalDateTime, zones: ArrayList<Event>):void
+checkIfPerformanceExist(date LocalDateTime):boolean

«abstract» RegisteredUser
-username: String
-password: String
+checkCredentials():boolean
+cypher(password: String):String

TheaterPlay
-actors: String

Concert
-conductor: String
-music pieces: String
-orchestra: String

Dance
-conductor: String
-dancer: String
-orchestra: String

Ticket
-code:int
-purchased: boolean
-booked:Boolean
+freeTicket():void
+generateCode():int
+generateTicketPDF():void

Customer
+addTicket(ticket: Ticket):void
+removeTicket(ticket: Ticket):void
+addPass(pass: Pass):void
+removePass(tpass: Pass):void

Manager

WaitList
+subscribeToWaitList(customer: Customer): voi
+unsubscribeToWaitList(customer: Customer):

TicketPDF

Booked
-timeToPay: LocalDateTime
+checkOutOfTime():boolean

zone • dad • subzones 0..* • statistics • seats • seat in • maintenance seat • zones • users • events • passes • customers passes • performance • performance seats • zone of the ticket • zone • from cycle • from events 1..* • annual pass price • attended events • Performances • from a • available tickets • if booked • users waiting • customer tickets