model

<<interface>> User

+ getEmail(): String
+ getName() : String
+ getPassword(): String
+ toString()

Employee

- name : String
- availabilities []: Availability

+ Employee (name, availabilities [])
+ getAvaillabilities() : Availability []
+ getName() : String
+ setAvailabilities(Availability [])
+ toString()

Business

- name : String
- desc : String
- address : String
- phone : String
- staff[ ] : Employee
- avBookings[] : Booking
- owner : Owner

+ Business (name, desc, address, phone, staff, avBookings, owner)
+ getAddress() : String
+ getAvBookings() : Booking[]
+ getDesc() : String
+ getName() : String
+ getPhone() : String
+ getStaff() : Employee []
+ getOwner () : Owner
+ displayBookings()
+ toString(): String

AbstractUser Implements User

- name: String
- email: String
- password: String

+ AbstractUser (name, email, password, bookings)
+ getEmail() : String
+ getName(): String
+ getPassword(): String
+ toString() : String

Owner extends AbstractUser

- business: Business

+ Owner (name, email, password)
+ Owner (name, email, password, bookings)
+ toString()
+ getBusiness() : Business
+ setBusiness(business)
+ viewBookings()

Customer extends AbstractUser

- address: String
- phone: String
- bookings: Booking[]

+ Customer (name, email, address, phone, password,)
+ Customer (name, email, address, phone, password, bookings)
+ getAddress() : String
+ getPhone () : String
+ toString()
+ getBookings() : Booking[]
+ setBookings(bookings[]) : boolean
+ addBooking(Booking) : boolean
+ cancelBooking(booking) : boolean
+ viewBookings()

Availability

+ date: String
+ time: String

+ Availability(date, time)
+ getDate() : String
+ getTime(): String
+ setDate(date)
+ setTime(time)
+ toString();

Booking

- slot : Availability
- staff : String
- business [] : Business
- status : String

+ Booking (slot, staff, status)
+ getSlot : Availability
+ getStaff() : String
+ getBusiness() : Business
+ setBusiness(business)
+ getStatus(): String
+ setStatus(String)
+ toString();

Tests

Login/RegisterTest

MenuTest

FileTest

BookingTest

Employee Test

Exceptions

RegistrationNonUniqueException

MenuInputException

RegistrationValidationException

CredentialsInvalidException

Controller

UserAuth

- utils: Utilities
- activeUser : User

+ UserAuth (Utilities utils)
+ authUser(String email, String password)
+ registerUser(name, email, password, address, phone)
+ validateEmail(email): booloean
+ validatePhone(phone): boolean
+ validateAddress(address): boolean
+ getActiveUser(): User
+ setActiveUser(User activeUser)
+ registerEmployee(String, Business): boolean
+ registerOwner(String, String, String): boolean

CustomerController

- appframe: Appframe
- logger: Logger

+ addBooking(Booking)
+ dashboard()
+ removeBooking(Booking)
+ CustomerController(AppFrame)

OwnerController

- appframe: Appframe
- logger: Logger

+ addAvBooking()
+ addAvBooking(String, String, String, String)
+ addEmployee()
+ addEmployee(String, Object)
+ checkEmail(String, Object)
+ createCustBooking(Customer, Business, Booking)
+ dashboard()
+ deleteBooking(Booking)
+ getBusinessNames() List<String>
+ getBusinessNames(User) List<String>
+ getEmployees(Object) List<String>
+ logout()
+ makeCustBooking()
+ newBusiness()
+ registerBusiness(String, String, String, String)
+ reloadDashboard()
+ removeBooking(Object)
+ OwnerController(AppFrame)

Registry

- userAuth: UserAuth
- utils: Utilities

+ getUserAuth(): UserAuth
+ getUtils(): Utilities
- Regisrty()

UserController

- appFrame: Appframe
- activeBusiness: Business
- logger: Logger

+ UserController(AppFrame)
+ loadLoginScreen()
+ loadRegisterScreen()
+ reloadWelcomeScreen()
+ login (String, String)
+ register (String, String, String, String) boolean
+ logout()
+ getActiveBusiness(): Business
+ setActiveBusiness(Business)

View Style

Gui Componenet

View

App

Utilities

- customerInfoFileName : String
- businessInfoFileName : String
- filePath : String
- splitChar : String
- businesses[] : Business
- customers[] : User
- owners: Owners
- logger: Logger

+ Utilities()
+ Utilities (filePath, customerInfoFileName,
businessInfoFileName, splitChar)
+ getBusinesses () : Business []
+ getCustomers() : Customer []
+ readBusinessData() : int
+ readCustomerData() : int
+ readData() : int
+ setCustomers() : int
+ addCustomers() : int
+ writeBusinessData() : int
+ writeCustomerData() : int
+ writeData() : int
+ searchCustomer(String): User
+ searchEmployees(String, List<Employee>): Employee
+ setCustomers(List<User>)
+ silentSave()
+ writeBusinessData(List<Business>): int
+ writeCustomerData(): int
+ writeData(List<Business>, List<User>): int

View Style

AppStyle

- mainBackgroundColor: Color
- mainForegroundColor: Color
- mainFontColor: Color
- successColor: Color
- warningColor: Color
- boldLargeFont: Font
- boldMedFont: Font
- margin: Border

Factory

BookingFactory

- bookingString: String
- length: String

+ bookingButton(booking): AvButton
+ bookingPane(booking): JTextPane
+ bookingString(booking): String
+ TestBookingPanel(): JTextPane

Gui Componenet

ABSMenuBar

- login: JMenItem
- register: JMenItem
- logout: JMenItem

+ toggleButton(String, Booolean): void
+ ABSMenuBar(AooFrame)

AvButton

- serialVersionUID: Long
- booking: booking

+ AvButton(String)
+ getBooking(): booking
+ setBooking(booking): void

BsButton

- seriealVersionUID: Long
- business: business

+ BsButton(String, Button)
+ getBusiness(): business
+ setBusiness(business): void

BusInfoDialog

- serialVersionUID:Long

+ BusInfoDialog(appFrame)

StatusBar

- serialVersionUID:Long

+ StatusBar(appFrame)

View

AppFrame

- content: Jpanel
- ownerDashbaord:Jpanel
- serialVersionUID: Long
- welcomePanel: JPanel

+ getOwnerDashboard(): JPanel
+ getWelcomePanel(): JPanel
+ setDialogueLocation(JFrame, JDialog):JPanel
+ setOwnerDashboard(Jpanel): void
+ AppFrame()
+ gettContent(): JPanel

LoginPanel

- serialVersionUID: Long
- status: JLabel

+ getStatus(): JLabel
+ LoginPanel()

WelcomePanel

- serialVersionUID: Long

+ welcomePanel()

Menu

- activeBusiness: Business
- sc: Scanner
- utils: Utilities
- userAuth: userAuth

+ menu(Scanner, Utilities, UserAuth)
+ menu(Utilities, UserAuth)
+ addBooking(): void
+ bookingOptions(): int
+ businessInfo(): int
+ businessSelect(): void
+ cancleBooking():void
+ customerBooking(): int
+ getActiveBusiness(): Business
+ loginMenu(): User
+ mainMenu(): int
+ myBookings(): int
+ ownerDashboard(): int
+ regsiterMenu(): User

RegisterPanel

- serialVersionUID: Long
- status: Jlabel

+ getStatus():Jlabel
+ registerPanel()

App

ABS

+ main()

ConsoleApp

- menu: Menu
- status
- userAuth: UserAuth
- utils: Utilities

+ businessInfo()
+ businessSelect()
+ customerDash()
+ exit()
+ login()
+ main()
+ mainMenu()
+ ownerDash()
+ register()
+ viewAvailableBookings(0
+ viewMyBookings()

1

leApp

0