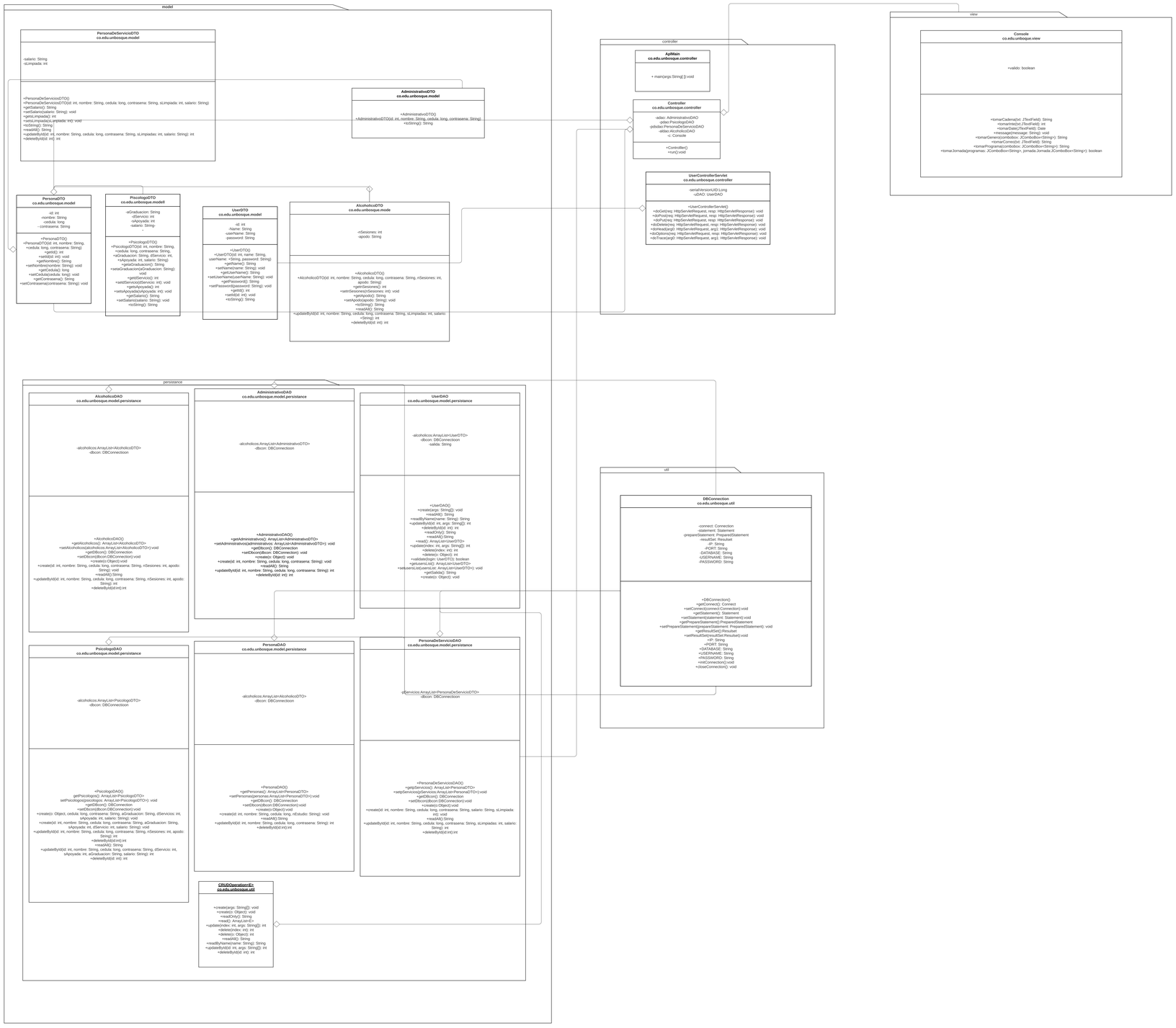
model
PersonaDeServiciosDTO
co.edu.unbosque.model
-salario: String
-sLimpiada: int
+PersonaDeServiciosDTO()
+PersonaDeServiciosDTO(d: int, nombre: String, cedula: long, contrasena: String, sLimpiada: int, salario: String)
+getSalario(): String
+setSalario(salario: String): void
+getsLimpiada(): int
+setsLimpiada(sLimpiada: int): void
+toString(): String
+readAll(): String
+updateById(d: int, nombre: String, cedula: long, contrasena: String, sLimpiadas: int, salario: String): int
+deleteById(d: int): int
AdministrativoDTO
co.edu.unbosque.model
+AdministrativoDTO()
+AdministrativoDTO(d: int, nombre: String, cedula: long, contrasena: String)
+toString(): String
PersonaDTO
co.edu.unbosque.model
-d: int
-nombre: String
-cedula: long
- contrasena: String
+PersonaDTO()
+PersonaDTO(d: int, nombre: String, +cedula: long, contrasena: String)
+getId(): long
+setId(pd: int): void
+getNombre(): String
+setNombre(nombre: String): void
+getCedula(): long
+setCedula(cedula: long): void
+getContrasena(): String
+setContrasena(contrasena: String): void
PsicologoDTO
co.edu.unbosque.model
-aGraduacion: String
-dServicio: int
-sApoyada: int
-salario: String
-
+PsicologoDTO()
+PsicologoDTO(d: int, nombre: String, +cedula: long, contrasena: String, +aGraduacion: String, dServicio: int, +sApoyada: int, salario: String)
+getaGraduacion(): String
+setaGraduacion(aGraduacion: String): void
+getdServicio(): int
+setdServicio(dServicio: int): void
+getsApoyada(): int
+setsApoyada(sApoyada: int): void
+getSalario(): String
+setSalario(salario: String): void
+toString(): String
UserDTO
co.edu.unbosque.model
-id: int
-Name: String
-userName: String
-password: String
+UserDTO()
+UserDTO(id: int, name: String, userName: +String, password: String)
+getName(): String
+setName(name: String): void
+getUserName(): String
+setUserName(userName: String): void
+getPassword(): String
+setPassword(password: String): void
+getId(): int
+setId(id: int): void
+toString(): String
AlcoholicosDTO
co.edu.unbosque.mode
+sSesiones: int
-apodo: String
+AlcoholicosDTO()
+AlcoholicosDTO(d: int, nombre: String, cedula: long, contrasena: String, nSesiones: int, apodo: String)
+getSesiones(): int
+setnSesiones(nSesiones: int): void
+getApodo(): String
+setApodo(apodo: String): void
+toString(): String
+readAll(): String
+updateById(d: int, nombre: String, cedula: long, contrasena: String, sLimpiadas: int, salario: +String): int
+deleteById(d: int): int
controller
ApiMain
co.edu.unbosque.controller
+ main(args.String[]):void
Controller
co.edu.unbosque.controller
-adao: AdministrativoDAO
-pdao:PsicologoDAO
-pdsdao:PersonaDeServiciosDAO
-alibro:AlcoholicosDAO
-c: Console
+Controller()
+run():void
UserControllerServlet
co.edu.unbosque.controller
-serialVersionUID:Long
-uDAO: UserDAO
+UserControllerServlet()
+doGet(req: HttpServletRequest, resp: HttpServletResponse): void
+doPost(req: HttpServletRequest, resp: HttpServletResponse): void
+doPut(req: HttpServletRequest, resp: HttpServletResponse): void
+doDelete(req: HttpServletRequest, resp: HttpServletResponse): void
+doHead(arg0: HttpServletRequest, arg1: HttpServletResponse): void
+doOptions(req: HttpServletRequest, resp: HttpServletResponse): void
+doTrace(arg0: HttpServletRequest, arg1: HttpServletResponse): void
view
Console
co.edu.unbosque.view
+valido: boolean
+tomarCadena(txt: JTextField): String
+tomarInte(txt:JTextField): int
+tomarDate(JTextField): Date
+message(message: String): void
+tomarGenero(combobox: JComboBox<String>): String
+tomarCorreo(tor: JTextField): String
+tomarProgramas(combobox: JComboBox<String>): String
+tomarJornada(programas: JComboBox<String>, jornada:Jornada:JComboBox<String>): boolean
util
DBConnection
co.edu.unbosque.util
-connect: Connection
-statement: Statement
-prepareStatement: PreparedStatement
-resultSet: Resultset
-URL: String
-PORT: String
-DATABASE: String
-USERNAME: String
-PASSWORD: String
+DBConnection()
+getConnect(): Connect
+setConnect(connect:Connection):void
+getStatement(): Statement
+setStatement(statement: Statement):void
+getPrepareStatement():PreparedStatement
+setPrepareStatement(prepareStatement: PreparedStatement): void
+getResultSet():Resultset
+setResultSet(resultSet:Resultset):void
+getURL: String
+PORT: String
+DATABASE: String
+USERNAME: String
+PASSWORD: String
+initConnection():void
+closeConnection(): void
persistance
AlcoholicosDAO
co.edu.unbosque.model.persistance
-alcoholicos:ArrayList<AlcoholicosDTO>
-dbcon: DBConnecticion
+AlcoholicosDAO()
+getAlcoholicos():ArrayList<AlcoholicosDTO>
+setAlcoholicos(alcoholicos:ArrayList<AlcoholicosDTO>):void
+getBlicon(): DBConnection
+setDbcon(dBcon:DBConnection):void
+create(o:Object):void
+create(id: int, nombre: String, cedula: long, contrasena: String, nSesiones: int, apodo: String): void
+readAll(): String
+updateById(id: int, nombre: String, cedula: long, contrasena: String, nSesiones: int, apodo: String): int
+deleteById(id:int):int
AdministrativoDAO
co.edu.unbosque.model.persistance
-alcoholicos:ArrayList<AdministrativoDTO>
-dbcon: DBConnecticion
+AdministrativoDAO()
+getAdministrativos():ArrayList<AdministrativoDTO>
+setAdministrativos(administrativos: ArrayList<AdministrativoDTO>): void
+getDbcon(): DBConnection
+setDbcon(dBcon: DBConnection): void
+create(o: Object): void
+create(id: int, nombre: String, cedula: long, contrasena: String): void
+readAll(): String
+updateById(id: int, nombre: String, cedula: long, contrasena: String): int
+deleteById(id: int): int
UserDAO
co.edu.unbosque.model.persistance
-alcoholicos:ArrayList<UserDTO>
-dbcon: DBConnecticion
-sAlda: String
+UserDAO()
+create(args: String[]): void
+readAll(): String
+readById(id: int): String
+updateById(id: int, args: String[]): int
+deleteById(id: int): int
+readOne(): String
+read(): ArrayList<UserDTO>
+update(index: int, args: String[]): int
+delete(index: int): int
+delete(o: Object): int
+validate(u: UserDTO): boolean
+getUsers(): ArrayList<UserDTO>
+setUsersList(usersList: ArrayList<UserDTO>): void
+getSalida(): String
+create(o: Object): void
PsicologoDAO
co.edu.unbosque.model.persistance
-alcoholicos:ArrayList<PsicologoDTO>
-dbcon: DBConnecticion
+PsicologoDAO()
getPsicologos():ArrayList<PsicologoDTO>
setPsicologos(psicologos: ArrayList<PsicologoDTO>): void
+getBlicon(): DBConnection
+setDbcon(dBcon:DBConnection):void
+create(o: Object, cedula: long, contrasena: String, aGraduacion: String, dServicios: int, sApoyada: int, salario: String): void
+create(id: int, nombre: String, cedula: long, contrasena: String, aGraduacion: String, sApoyada: int, dServicio: int, salario: String): void
+updateById(id: int, nombre: String, cedula: long, contrasena: String, nSesiones: int, apodo: String): int
+deleteById(id:int):int
+readAll(): String
+updateById(id: int, nombre: String, cedula: long, contrasena: String, dServicio: int, sApyada: int, aGraduacion: String, salario: String): int
+deleteById(id: int): int
PersonaDAO
co.edu.unbosque.model.persistance
-alcoholicos:ArrayList<AlcoholicosDTO>
-dbcon: DBConnecticion
+PersonaDAO()
+getPersonas(): ArrayList<PersonaDTO>
+setPersonas(personas:ArrayList<PersonaDTO>):void
+getDbcon(): DBConnection
+setDbcon(dBcon:DBConnection):void
+create(o:Object):void
+create(id: int, nombre: String, cedula: long, nEstudio: String): void
+readAll():String
+updateById(id: int, nombre: String, cedula: long, contrasena: String): int
+deleteById(id:int):int
PersonaDeServicioDAO
co.edu.unbosque.model.persistance
-dServicios:ArrayList<PersonaDeServicioDTO>
-dbcon: DBConnecticion
+PersonaDeServicioisDAO()
+getpServicios(): ArrayList<PersonaDTO>
+setpServicios(pServicios:ArrayList<PersonaDTO>):void
+getDbcon(): DBConnection
+setDbcon(dBcon:DBConnection):void
+create(o:Object):void
+create(id: int, nombre: String, cedula: long, contrasena: String, salario: String, sLimpiada: int): void
+readAll():String
+updateById(id: int, nombre: String, cedula: long, contrasena: String, sLimpiadas: int, salario: String): int
+deleteById(id:int):int
CRUDOperation<E>
co.edu.unbosque.util
+create(args: String[]): void
+create(o: Object): void
+readOne(): String
+read(): ArrayList<E>
+update(index: int, args: String[]): int
+delete(index: int): int
+delete(o: Object): int
+readAll(): String
+readByName(name: String): String
+updateById(id: int, args: String[]): int
+deleteById(id: int): int