Class Model

«control»
Controller

«control»
HomePage
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
+initializeLibri(String type): void
+initializeRivista(String type): void
+initializeRicerca(String type): void
+persistence(String type): void

«control»
CercaInnesta
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+getLista(String type, String database): ObservableList<Racconto>

this class has not relationships with «entity», since it violate schema VORCCP

«control»
Visualizza
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+getID(): int
+getListLibri(String type): ObservableList<Libro>
+getListGiornali(String type): ObservableList<Giornale>
+getListRivista(String type): ObservableList<Rivista>

«control»
PagamentoCash
+controlla(String nome, String cognome, String via, String cap, String type): void
+getIdx(): String[]

«control»
Download
-LIBRO: String = libri (readOnly)
-GIORNALI: String = giornali (readOnly)
-RIVISTA: String = riviste (readOnly)
-ANTIQUATED: String = antiquated (readOnly)
-FILE: String = file (readOnly)
+acquistaLibro(String persistence): void
+acquistaGiornale(String persistence): void
+acquistaRivista(String persistence): void
+scaricaLibro(String type, String persistence): void

«control»
PagamentoCartaCreditto
-DATABASE: String
-FILE: String
-MEMORIA: String
+aggiungiCartaDiCk(String n, String c, String cod, java.sql.Date data, String ch): float prezzo, String persistenza): boolean
+listaUtenteCC(String nome): String persistenza): ObservableList<CartaCreditto>
+pagamentoCC(String nome, String database): void
+getIdx(): String
+formatNumero(String numero, String persistenza): observableList<CartaCreditto>

«control»
ArrestaPagamento
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+getPaRiva(String type): String
+getPagamento(String type): String
+cancellaPaRiva(String text, String string): boolean
+cancellaPagamento(String text, String string): String

«control»
Ricerca
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+listaLibri(String nome, String persistenza): ObservableList<Racconto>
+listaGiornali(String nome, String persistenza): ObservableList<Racconto>
+listaRivista(String nome, String persistenza): ObservableList<Racconto>

«control»
Acquista
-LIBRO: String = libri (readOnly)
-GIORNALI: String = giornali (readOnly)
-RIVISTA: String = riviste (readOnly)
-ANTIQUATED: String = antiquated (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+getNomeCodeShop(String string): String[]
+getPrezzo(String q, String string): float

«control»
CheckPagamentoData
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+pagamentoLibri(String type, Pagamento p): void
+pagamentoDiversiale(String type, Pagamento p): void
+pagamentoGiornale(String type, Pagamento p): void
+checkPagamentoDiversi(String nome, String type): void
+controllaPag(String d, String c, String cv): void

«control»
SongWagazin
-DATABASE: String = database (readOnly)
-FILE: String = file (readOnly)
-MEMORIA: String = memoria (readOnly)
+getWagon(String type): ObservableList<Wagazin>
+isOpen(String type, int id): boolean
+isAvailable(int type, int id): String

this class has not relationships with «entity», since violate schema VORCCP

«control»
SystemState
-idLibro: int
-idGiornale: int
-idRivista: int
-idQ: int
-isSearch: boolean
-isPaUp: boolean
-prezzo?: float
-quantità: int
-nomeCodeShop: String
-typeAssetNow: String
-instance: SystemState
+getIdLibro(): int
+setIdLibro(value: int)
+getIdGiornale(): int
+setIdGiornale(value: int)
+getIdRivista(): int
+setIdRivista(value: int)
+getType(): String
+setType(value: String)
+getIsSearch(): boolean
+setIsSearch(value: boolean)
+getIsPaUp(): boolean
+setIsPaUp(value: boolean)
+getPrezzo?(): float
+setPrezzo?(value: float)
+getQuantità(): int
+setQuantità(value: int)
+getNomeCodeShop(): String
+setNomeCodeShop(value: String)
+getTypeAssetNow(): String
+setTypeAssetNow(value: String)
+getInstance()

«entity»
Libri
-titolo: String
-categoria: String
-editore: String
-autore: String
-lingua: String
-categoria: String
-descrizione: String
-dataPubb: LocalDate
-recensione: String
-copie: int
-inCopie: int
-disponibilità: int
-prezzo: float
-id: int
-isNotGenerati: String[]
-setNotGenerati(): void
-getTitolo(): String
-setTitolo(value: String)
-getAutore(): String
-setAutore(value: String)
-getLingua(): String
-setLingua(value: String)
-getCategoria(): String
-setCategoria(value: String)
-getDescrizione(): String
-setDescrizione(value: String)
-getDataPubb(): LocalDate
-setDataPubb(value: LocalDate)
-getRecensione(): String
-setRecensione(value: String)
-getPaDigital(): int
-setPaDigital(value: int)
-getNoCopie(): int
-setNoCopie(value: int)
-getDisponibilità(): int
-setDisponibilità(value: int)
-getPrezzo(): float
-setPrezzo(value: float)
-getId(): int
-setId(value: int)
-getIdGiornale(): String
-setIdGiornale(value: String)
-getIdCodeCopy(): String
-setIdCodeCopy(value: String)
-getIsNotGenerati(): boolean
-setPaPdf(String name): void

«enumerative»
CategorieLibri
ADOLESCENTI, RAGAZZI
ARTE
CINEMA, FOTOGRAFIA
BIOGRAFIE
DIARI, MEMORIE
DIRITTO, ECONOMIA, AGENDE
DIRITTO
DIZIONARI, OPERE
ENCICLOPEDIE
FAMIGLIA
FANTASCIENZA, FANTASY
FUMETTI, MANGA
GIALLI, THRILLER
GIOCHI, HOBBY, GIOCHI
HUMOR
INFORMATICA
LETTERATURA, MEDIA
LETTERATURA, NARRATIVA
LIBRI BAMBINI
LIBRI SCOLASTICI
LIBRI UNIVERSITARI
LINGUA, DIZIONARI
LINGUISTICA, SCRITTURA
POLITICA
RELIGIONE
ROMANZI, RSA
SCIENZE
TECNOLOGIA, MEDICINA
ALTRO

«enumerative»
Categorie

«interface»
Racconto
+scaricaLibro(): void
+leggiLibri(): void

«interface»
Serializzabile

«implementation»

«entity»
CartaCreditto
-tipo: int
-numeroCarta: String
-data: double
-ammontare: double
-scadenza: java.sql.date
-prezzoIncassodover: float
-id: int
-types: String = fro CardCash Only,"DINERS","JCB","MASTER","VISA" (readOnly)
+getTipo(): int
+setTipo(value: int)
+getNumeroCC(): String
+setNumeroCC(value: String)
+getLimite(): double
+setLimite(value: double)
+getAmmontare(): double
+setAmmontare(value: double)
+getScadenza(): java.util.date
+setScadenza(value: java.util.date)
+getPrezzoIncassodover(): float
+setPrezzoIncassodover(value: float)
+getId(): String
+setId(value: String)

«entity»
Entity

«entity»
Futura
-nome: String
-cognome: String
-via: String
-cap: String
-ammontare: float
-id:Futura: int
+getNome(): String
+setNome(value: String)
+getCognome(): String
+setCognome(value: String)
+getVia(): String
+setVia(value: String)
+getCore(): String
+setCore(value: String)
+getAmmontare(): float
+setAmmontare(value: float)
+getIdFutura(): int
+setIdFutura(value: int)

«entity»
Negozio
-nome: String
-cognome: String
-a:bool: boolean
-isOpen: boolean
-id: int
+getNome(): String
+setNome(value: String)
+getNeg(): String
+setNeg(value: String)
+getIsOpen(): boolean
+setIsOpen(value: boolean)
+getIsAvailable(): boolean
+setIsAvailable(value: boolean)
+getId(): int
+setId(value: int)

«entity»
Pagamento
-idPag: int
-nome: String
-ammontare: float
-tipo: String
-isOggetto: String
+getIdPag(): int
+setIdPag(value: int)
+getNome(): String
+setNome(value: String)
+getAmmontare(): float
+setAmmontare(value: float)
+getTipo(): String
+setTipo(value: String)
+getIsOggetto(): String
+setIsOggetto(value: String)

«entity»
Giornale
-titolo: String
-categoria: String
-lingua: String
-editore: String
-dataPubb: LocalDate
-copieRimanenti: int
-disponibilità: int
-prezzo: float
-id: int
-isNotGenerati: String[]
-isTitolo: String = titoloGi (readOnly)
-TITOLOR: String = titoloGi (readOnly)
-DATEBTH: String = dsPubb (readOnly)
+getTitolo(): String
+setTitolo(value: String)
+getCategoria(): String
+setCategoria(value: String)
+getLingua(): String
+setLingua(value: String)
+getEditore(): String
+setEditore(value: String)
+getDataPubb(): LocalDate
+setDataPubb(value: LocalDate)
+getCopieRimanenti(): int
+setCopieRimanenti(value: int)
+getDisponibilità(): int
+setDisponibilità(value: int)
+getPrezzo(): float
+setPrezzo(value: float)
+getId(): int
+setId(value: int)
+getIsNotGenerati(): String
+setIsNotGenerati(value: String)
+isaPdf(): void

«implementation»

«entity»
Rivista
-titolo: String
-categoria: String
-editore: String
-lingua: String
-dataPubb: LocalDate
-descrizione: String
-datePubb: localDate
-copie: int
-prezzo: float
-copieRimor: int
-id: int
-isNotGenerati: String[]
-isTitolo: String = titoloRi (readOnly)
-TITOLOR: String = titoloRi (readOnly)
-DATEBTH: String = dsPubb (readOnly)
+getTitolo(): String
+setTitolo(value: String)
+getCategoria(): String
+setCategoria(value: String)
+getVolume(): String
+setVolume(value: String)
+getLingua(): String
+setLingua(value: String)
+getEditore(): String
+setEditore(value: String)
+getDescrizione(): String
+setDescrizione(value: String)
+getDataPubb(): localDate
+setDataPubb(value: localDate)
+getDisp(): int
+setDisp(value: int)
+getPrezzo(): float
+setPrezzo(value: float)
+getCopieRimor(): int
+setCopieRimor(value: int)
+getId(): int
+setId(value: int)
+getIsNotGenerati(): String
+setIsNotGenerati(value: String)
+isaPdf(): void

«enumerative»
CategorieRivista
SETTIMANALE
BISETTIMANALE
MENSILE
BIMESTRALE
TRIMESTRALE
ANNUALE
ESTIVO
INVERNALE
ARTISTICA
CINEMATOGRAFICA
GOSSIP
MODA, STILE
MILITARE
INFORMATICA

«entity»
Factory
-LIBRO: String = libri (readOnly)
-GIORNALI: String = giornali (readOnly)
-RIVISTA: String = riviste (readOnly)
+creaRaccolta(String titolo, String isbn, String editore, String autore, String lingua, String categoria): void
+creaRaccolta(String titolo, String isbn, int diqu, float prezzo, int id): String
+creaRaccolta(Frase[] [][]string typology2, String titolo, String isbn, String editore, String autore, String lingua, String categoria): void
+creaRaccolta(Frase[] [][]string typology2, int numFrag, int inCopie, int disp, float prezzo, int id): void
+creaRaccolta(Frase[] [][]string typology2, int numFrag, int inCopie, int disponibilità, float prezzo, int id): Racconto
+creaLibro(String[] info, LocalDate dataPubb, String recensione, String idea): String[]idCodeS: Racconto
+creaGiornale(String[] info, LocalDate dataPubb, int inCopie, int disponibilità, float prezzo, int id): Racconto
+creaRivista(String[] info, String descrizione, LocalDate dataPubb): int disp, float prezzo, int copieRim, int id): Racconto

MatchPersen
+proposta1(String titolo, String isbn, String editore, String autore, String lingua, String categoria): String
+proposta3(int numFrag, int inCopie, int disp, float prezzo, int id): String

«implementation»