**TÀI LIỆU ĐẶC TẢ KĨ THUẬT**

**CHUONG TRÌNH KIỂM SOÁT VÉ GA TÀU TỰ ĐỘNG**

NHÓM 1:

* HOÀNG GIA BẢO
* PHẠM NGỌC BÁ
* NGUYỄN QUANG ANH
* LÊ ĐỨC ANH

MÔ HÌNH 3 LỚP:

**CONTROLLER LAYER**

HasValidTimeController.java: (abstract)

* **public** **void** getInStation(String stselect): *Ghi lại các thông tin của thẻ và nhà ga vào trong bảng passing\_history khi hành khách đi vào*

*@param stselect id của nhà ga khi đi vào .*

* **public** **void** getOutStation(String stselect): *Tính toán phí của chuyến đi và ghi lại các thông tin của thẻ, nhà ga, phí của chuyến đi vào trong bảng passing\_history khi hành khách đi ra.*

*@param stselect id của nhà ga khi đi ra .*

1. CardController.java

* public boolean checkBalance (String id): *Kiểm tra số dư trong thẻ có lớn hơn số dư tối thiểu hay không.*

*@return true nếu không ít hơn, false nếu ngược lại.*

2. Ticket24Controller.java

* **public** **boolean** checkTimeValidity(): *Kiểm tra vé 24h còn thời hạn sử dụng không.*

*@return Trả về true nếu còn hạn, false nếu ngược lại*

3. TicketOwController.java

* **public** **boolean** checkEnterStation(String enterst): *Kiểm tra nhà ga muốn vào có nằm ngoài 2 nhà ga ghi trên vé hay không.*

*@param enterst Id nhà ga muốn vào.*

*@return Trả về true nếu không nằm ngoài, false nếu ngược lại*

* **public** **boolean** checkStatus(): *Kiểm tra trạng thái sử dụng của ticket one-way*

*@return Trả về true nếu vẫn còn trạng thái sử dụng, false nếu ngược lại*

**DAO LAYER**

1. CardsDAO.java

* **public** **static** **void** updateCard(CardsDTO card): *Cập nhật thông tin thẻ vào database*

*@param card object CardsDTO chứa thông tin của card*

* **public** **static** CardsDTO getCardById(String id): *Lấy về bản ghi card theo giá trị id truyền vào*

*@param id Định danh của card*

*@return Trả về 1 object CardsDTO*

2.ListTicketDAO.java

* **public** **static** ListTicketDTO getTicketType(String id):*Lấy thông tin vé từ bảng list\_ticket*

*@param id id vé*

3. PassHistoryDAO.java

* **public** **static** List<PassHistoryDTO> getAll(): *Lấy về danh sách lịch sử thông hành qua các nhà ga*

*@return Trả về danh sách lịch sử thông hành*

* **public** **static** PassHistoryDTO getInfoBySql(String sql): *Truy vấn Select tới bản passing\_history*

*@param sql Câu lệnh truy vấn Select*

*@return Trả về 1 object PassHistoryDTO*

* **public** **static** PassHistoryDTO getInfoById(String id): *Lấy 1 bản ghi Lịch sử thông hành qua Id của vé/ticket đã được sử dụng*

*@param id*

*@return Trả về 1 object PassHistoryDTO*

* **public** **static** PassHistoryDTO getInfoByPassId(**int** pass\_id): *Lấy 1 bản ghi Lịch sử thông hành qua Id của nó*

*@param id*

*@return Trả về 1 object PassHistoryDTO*

* **public** **static** PassHistoryDTO getInfo(String id, String getin\_point, Timestamp getin\_time): *Lấy 1 bản ghi Lịch sử thôg qua Id của vé/card đã đi qua, điểm vào, và thời gian vào.*

*@param id Id vé/thẻ*

*@param getin\_point Nhà ga đã vào*

*@param getin\_time Thời gian vào*

*@return Trả về 1 object PassHistoryDTO*

* **public** **static** **void** updatePassHistoryById(PassHistoryDTO ph): *Cập nhật lại Lịch sử thông hành trong CSDL*

*@param ph 1 object PassHistoryDTO*

* **public** **static** **void** insertPassHistory(PassHistoryDTO ph): *Chèn thêm vào Lịch sử thông hành trong CSDL*

*@param ph 1 object PassHistoryDTO*

4. StationDAO

* **public** **static** List<StationDTO> getAll() : *Lấy danh sách cách nhà ga lưu vào 1 List*

*@return Trả về danh sách nhà ga lưu dạng List(StationDTO)*

* **public** **static** StationDTO getStationById(String id): *Lấy thông tin nhà ga từ database theo id của nó.*

*@param id Id nhà ga*

*@return Trả về 1 object StationDTO*

5. Ticket24hDAO

* **public** **static** Ticket24hDTO getTk24ById(String id): *Lấy thông tin ticket 24h trong database theo id*

*@param id Id vé 24h*

*@return Trả về 1 object Ticket24hDTO*

* **public** **static** **void** updateTk24(Ticket24hDTO tk24): *Cập nhật thông tin 1 ticket 24h vào database*

*@param tk24 Object Ticket24hDTO*

*6.TicketOnewayDAO*

* ***public******static*** *TicketOnewayDTO getTkowById(String id): Lấy về thông tin vé 1 chiều từ Database theo id truyền vào*

*@param id Id ticket one-way*

*@return Trả về 1 object TicketOnewayDTO*

* ***public******static******void*** *updateTkow(TicketOnewayDTO tkow): Cập nhật thông tin 1 bản ghi vé 1 chiều vào database*

*@param tkow Object TicketOnewayDTO*

**DTO LAYER**

1.CardsDTO.java

* **public** String getCard\_id():

*@return the card\_id*

* **public** **void** setCard\_id(String card\_id)
* **public** String getOwner\_id()

*@return the owner\_id*

* **public** **float** getBalance()

*@return the Balance*

* **public** Timestamp getRelease\_time()

*@return the release\_time*

* **public** Timestamp getRelease\_time()
* **public** **void** setRelease\_time(Timestamp release\_time)
* **public** **int** getLast\_pass()
* **public** **void** setLast\_pass(**int** last\_pass)

2. [ListTicketDTO.java](https://bitbucket.org/phamba/isd.vn.20191-01/src/master/MiniProjectItss/src/dto/ListTicketDTO.java)

3. [PassHistoryDTO.java](https://bitbucket.org/phamba/isd.vn.20191-01/src/master/MiniProjectItss/src/dto/PassHistoryDTO.java)

4.[StationDTO.java](https://bitbucket.org/phamba/isd.vn.20191-01/src/master/MiniProjectItss/src/dto/StationDTO.java)

5.[Ticket24hDTO.java](https://bitbucket.org/phamba/isd.vn.20191-01/src/master/MiniProjectItss/src/dto/Ticket24hDTO.java)

6. [TicketOnewayDTO.java](https://bitbucket.org/phamba/isd.vn.20191-01/src/master/MiniProjectItss/src/dto/TicketOnewayDTO.java)

**GUI**

*Screen.java*

* ***private******static******boolean*** *isAvailableAction(String input, String pattern): Kiểm tra xâu mà người dùng nhập có đúng định dạng hay không*

*@param input xâu mà người dùng nhập vào.*

*@param pattern định dạng mà người dùng muốn so sánh.*

*@return trả về true nếu xâu đúng định dạng và ngược lại*

* ***private******static******void*** *printStationSelectionScreen(): Phương thức in ra màn hình console list các nhà ga*
* ***private******static******void*** *printTicketList(): In ra màn hình console danh sách các mã pseudo code.*

*Message.java*

* ***public******static******void*** *printOpenMess(String type, String id,* ***float*** *balance): In ra thông thông báo mở cổng, và các thông tin của ticket one-way hay card đã được sử dụng để đi qua.*

*@param type Loại vé hay card*

*@param id Mã định danh của vé/card*

*@param balance Giá vé/số dư của thẻ*

*@exception InterruptedException Ngoại lệ cho gate*

* ***public******static******void*** *printOpenMess(String type, String id, Timestamp valid): In ra thông thông báo mở cổng, và các thông tin của ticket 24h đã được sử dụng để đi qua.*

*@param type Nên truyền vào chuỗi "Ticket 24h"*

*@param id Mã định danh của vé/card*

*@param valid Thời hạn của vé 24h*

*@exception InterruptedException Ngoại lệ cho gate*

* ***public******static******void*** *printErrorMess(String type, String id,* ***float*** *balance,* ***float*** *fare): In ra thông thông báo lỗi khi giá mua ticket one-way hoặc số dư trong card nhỏ hơn giá hành trình.*

*@param type Loại vé hay card*

*@param id Mã định danh của vé/card*

*@param balance Giá vé/số dư của thẻ*

*@param fare Giá hành trình*

* ***public******static******void*** *printErrorMess(String type, String id, Timestamp valid, Timestamp getin\_time): In ra thông thông báo mở cổng, và các thông tin của ticket 24h đã được sử dụng để đi qua*

*@param type Nên truyền vào chuỗi "Ticket 24h"*

*@param id Mã định danh của vé 24h*

*@param valid Thời hạn của vé 24h*

*@param getin\_time Thời gian vào cổng*

* ***public******static******void*** *printErrorMessCard(String id,* ***float*** *balance): In ra thông thông báo lỗi khi số dư trong thẻ nhỏ hơn 2.5 euros.*

*@param id Mã định danh của card*

*@param balance Số dư trong thẻ*

* ***public******static******void*** *printErrorMessTkow(): In ra thông thông báo lỗi khi người dùng nhập vào mã ticket one-way đã được sử dụng.*
* ***public******static******void*** *printErrorMessTkow2(String id,* ***float*** *price, String st1, String st2, String st): In ra thông thông báo lỗi khi ticket one-way được sử dụng để vào hoặc ra một nhà ga nằm ngoài phạm vi 2 nhà ga được ghi trên vé.*

*@param id Mã định danh của vé/card*

*@param price Giá vé*

*@param st1 Nhà ga thứ 1 được ghi trên vé*

*@param st2 Nhà ga thứ 2 được ghi trên vé*

*@param st Nhà ga mà vé được sử dụng để vào*

* ***public******static******void*** *pressAnyKeyToContinue(): Press any key to continue*