Car Rental Management System

Management client:

Outlet, Employee, Partner and Category are pre loaded data in the singleton data initialization session bean.

We split our Management client into 3 modules:

1. Sales manager: perform CRUD operation to the rental rate.
2. Operation manager:

* perform CRUD operation for Models: In the case that there are still cars under this model, the model is set to disabled and cannot be deleted.
* perform CRUD operation for Cars: In the case that the car is in rental, the car cannot be deleted and it is set to disabled.
* View Transit Driver Dispatch Record:The system will check with the current date and retrieve a list of dispatch record.
* Assign Transit Driver, manager need to input the dispatch record id, system retrieve the driver that is in the current outlet and let the manager assign the employee to the dispatch record. the status of the dispatch record is changed from Unassigned to Assigned.
* Update Transit as completed: dispatch record is changed from Assigned to Completed. The outlet of the car is changed from the previous return outlet to the new pickup outlet.

1. Customer service executive:

EJB Timer:

Allocate Cars to current day reservation:

1. the timer will trigger every day at 2am
2. system retrieve the list of all reservations
3. from there we get the list of reservations on the current day.
4. a. if the reservation did not contain a specific model.

* retrieve a list of all cars regardless of models under the specific category.
* if the car has a reservation and it is not disabled, it has 4 possibilities:

1. the car is available and it is already in the pickup outlet. break

2. the car is on rental but the return outlet of this previous reservation will be at the new pickup outlet. break

3. the car is available but the return outlet of the previous reservation is != as the pickup outlet of new reservation. in this case, dispatch record is generated. break

4. the car is on rental and the return outlet of previous reservation is != as the pickup outlet of new reservation. in this case, dispatch record is also generated.

b. if the reservation have a specific model

* retrieve the list of cars under this model and category
* if the car has a reservation and it is not diabled, it has the 4 possibilities mentioned above.

Generation of dispatch record:

dispatch record is being generated during car allocation mentioned above. when dispatch record is generated. it is linked to a reservation record. After operation manager has assigned transit driver to the dispatch record, it will be associated with an employee.

Reservation Client:

pick up car: status of car is set to on rental, outlet is set to null, reservation has past is set to null such that the reservation cannot be cancelled.

return car: status of car is set to available, outlet is set to return outlet

Search Car:

1. customer input model/category or both, pickupdateTime, pickupOutlet(only can choose those which are opening at the indicated pickupdateTime, return date time, return outlet(only can choose those which are opening at the indicated return date time)

search car can be split into 2 parts:

1. we check whether there is an available rental rate for the input reservation.

* retrieve a list of rental rate under the specific category
* we cut the reservation into 24 hour blocks:
* 1st block start: pickupdatetime, end: pickupdatetime+24 hrs.
* 2nd block start: end of 1st block, end: start of 2nd block+24 hrs.
* Nth block start: end of (n-1)the block, end: start of nth block+24 hrs
* We check the first block if there is an existing rental rate overlaps with the start of this block, if there is, we add to a list of rental rate on that particular block. Then we calculate the minimum rental rate and sum it to the total rental rate.
* Increment by blocks until it finish checking (n-1)th block
* If there is any block’s starting does not match with any rental rate, exception is thrown.
* Lastly, we check the rental rate for the nth block, the nth block. But we change the end time of this block=>Start: (n-1)th end, end: return date time of the reservation.
* If there is existing rental rate that overlaps with the nth block start date. We add the rental rate to the list of rental rate on that particular block and find the minimum and then add to the total rental rate sum. Otherwise if no rental rate matches we throw an exception.

1. we check whether there is an available cars for the input reservation.

2a. if the customer input the category only

* list As= get all the reservation under this category
* list Bs = get all the available cars under this category
* check the reservations in As overlap with the new reservation, if they overlap, add them(a.getCars) to a new list Cs
* take away the C that is present in the Bs, the Bs are the available cars left for new reservation.

2b. if the customer did not input category but model or they input both category and model

* listBs = get all available cars under this model
* listAs = get all reservation under this model
* check the reservations in As overlap with the new reservation, if they overlap, add them(a.getCars) to a new list Cs
* take away the C that is present in the Bs, the Bs are the available cars left for new reservation.

Partner:

Log in as a partner

register for the customer

make reservation on behalf of the client