Software Requirements Specifications

For

ONLINE TOUR RESERVATION SYSTEM

Submitted By:

Jyotirmaya Ojha

130905114

Sailesh Sriram

130905126

CONTENTS

1.1 Purpose :

This SRS Document contains the complete software requirements for the Tour Reservation System and describes the design decisions, architectural design and the detailed design needed to implement the system. It provides the visibility in the design and provides information needed for software support. This online Tour Reservation Booking System provides a faster and a reliable method for tourists to do their bookings for the intended tour.

1.2 Scope:

1. User stores his information at the time of registration on the software.
2. Store data of the tour packages provided by the agency.
3. Store personal information and choices made by the newly registered user.
4. Maintain personal information of all the registered users on the software.
5. Keep information of the famous places of a location as well as the historic details of that location.
6. Provide options to the user to pay for a particular package.
7. Generate online receipts of the transactions performed by the user.

1.3 References:

1. **IEEE Std 830-1998,IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer**

2.1 Product Perspective:

Online Tour Reservation System would provide a safe and secure way to the customers to choose and book their tour packages. The system would provide information about all the historic places relevant to the tour package.

2.1.2 User Interfaces

1) Login Screen - This window would ask the user to enter the required username and password. After entering data in both of these fields the user is supposed to click on submit button. If the user is new to the software, he/she should be given the option to register on the website.

2) Registration Form – If the user is new to the software, he/she would be required to fill the required credentials in the form.

List of DO's and DON'Ts:

1) The user should not refresh the page once the online payment page is arrived in the ticket booking system.
2) The user should enter the correct information to the best of her/his knowledge.

2.1.3 Hardware Interfaces

The most recent version of Google Chrome (52.0.2743.116) is advised for opening the website once the software redirects the user to the specified page.

2.1.4 Operations

The user should perform the online payment within 10-15 mins of initiating the transaction. The page would be automatically refreshed if the user fails to perform the required actions within the specified time frame.

2.2 Product Functions:

Online Tour Reservation System will provide an easy, efficient and transparent method to perform bookings for the available tour packages. The customers who are already registered on the software would be provided with an option to make changes in their existing bookings and in their personal information. The customers who are new to the software would be asked to register on the software. Separate log-in and password would be provided to the user for future transactions .The customer would be provided with an option to pay using credit card, debit card or through cheque. System would check for validity of the staff and once payment is done, valid staff would make the ticket reservation system.

2.3 User Characteristics:

The users of this system are customers who want to explore different tour packages made available by the agency as well as the people who want to book or make any changes in the existing bookings done by them. The users are assumed to have basic knowledge of computers.

The administrators of the system should know the basic architecture of the system. They should be able to issue tickets to all the valid users who have bought a package.

2.4 Constraints:

1. Correct information of all the customers must be entered in the database after verification.
2. Administrators should have a proper user account to log in to the software.
3. Administrators should be able to access the system database easily and interact with the software with no hassle.
4. Customers should be provided with unique log-in ID and password and should be able to make any bookings without any hassle.
5. Updated information should be consistent.

2.5 Dependencies and Assumptions:

1. Software should be able to cater to the needs of multiple users.
2. Users must be familiar with English as the language.
3. Users should have sufficient knowledge of computers.

3.0 Specific Requirements:

Functional Requirements:

R1: Request available tour packages

Input: credentials provided by the user.

Process: check if user is existing or not and prepare list of available tour packages.

Output: Display list of available tour packages.

R2: Register new user

Input: personal details provided by the user

Process: Store data of the user and provide username and password

Output: Display success or failure of the request for registration

R3: Display the available tour packages and deals

Input: locations , budget, number of travellers, number of days

Process: Compute best deals for given input

Output: Best deals for the user in ranked order

R4: Provide the itinerary to the user

Input: selected package by the user

Process: prepare the itinerary using the options provided by the user

Output: display the itinerary

R5: Request for reservation of Tour

Input: unique ID, password and selected tour package's ID

Process: store personal details and choices made by the customer and update the database.

Output: Display success or failure for the request.

R6: Update personal details of the customer

Input: personal details provided by the user

Process: Restructure the database on the basis of details provided by customer

Output: Display the updated personal details of the customer.

R7: Cancel reservation

Input: username, password and Tour package's ID

Process: find the respective reservation which was made by the user and update the table.

Output: Display success or failure for the cancellation request.

R8: Book tour package

Input: choice of mode of payment

Process: Ask for the required credentials by the user and redirect to payment gateway

Output: Display success or failure of transaction successful and give transaction ID along with online receipt of transaction.

R9: Issue ticket to customer

Input: transaction ID by the user

Process: Check the validity of the transaction ID and issue ticket on the basis of the verification.

Output: Issue ticket if process successful and ask for transaction ID again if transaction ID found to be invalid.

R10: Modify reservations

Input: transaction ID, reservation ID, password and username

Process: Find the required reservation on the basis of the input provided by the user

Output: Display the reservation in editable format

Event table:

Serial Number	Event	Trigger	Source	Use Case	Response	Destination
1	the user wants to get registered	store the data	user	signing-in	update the database and provide computer generated username and password	user
2	the user wants to log-in	check if the user has entered correct credentials	user	logging-in	display the appropiat e message	user
3	the user selects from available tour packages	store the selected choice made by the user	user	select package	notify user about choices being stored by the system	user

4	the user wants to modify his package	store the modifications made by the user	user	modify package	display modified package	user
5	the user wants to modify his personal details	update as required	user	update personal details	display modified personal details	user
6	the user wants to make payments	ask user about the mode of payment	user	make payment	store data and give transaction -id to the user	user
7	the admin wants to check for validity of staff	the admin checks if payment has been done	user	perform validation	check if the staff is a valid employee and if valid ask staff to make valid Ticket Reservation System.	admin
8	user wants to cancel his/her reservations	take customer to his home page	user	cancel reservation	display the remaining reservations	user

Use - Case Template :

1) Use Case ID: 0

   Use Case Name: signing-in

   Scenario: the user wants to get registered to the Reservation System

   Brief Description: The user is new to the System and is now to be provided with user--name and password

   Primary Actor: user

   Pre-Condition: NA

   Post-Condition: the user gets registered and is provided with System generated username and password

   flow of Events: 1) The user enters his personal details to get registered

   Exceptions: NA

2) Use Case ID: 1

   Use Case Name: logging-in

   Scenario: the user wants to log into the System

   Brief Description: The user has got some credentials and now logs into the System using his username and password. If log-in successful , the user is led to home page otherwise he is asked to enter details again

   Primary Actor: user

   Pre-Condition: NA

   Post-Condition: the user is navigated to his home page

   flow of Events: 1) The user enters his username and password to log-in

   Exceptions: The user is told if incorrect username or password has been entered

3) Use Case ID: 2

Use Case Name: select package

Scenario: the user selects the desired package

Brief Description: The user is given the choices which he has to select from and now he selects some desired package

Primary Actor: user

Pre-Condition: the package should be available on the system

Post-Condition: the table is updated and notifies user about the choices being stored

flow of Events: 1) The user checks the available Tour Packages

2) The user selects the Tour Packages he is interested in

3) The user is notified about the selected choices

Exceptions: NA

4) Use Case ID: 3

Use Case Name: modify package

Scenario: The user has selected some package/packages and now wants to made some modifications in his/her choices

Brief Description: The user has already selected some packages and now he/she wants to modify his/her packages. The user is shown changes which have been made to his choices

Primary Actor: user

Pre-Condition: The user has selected some packages earlier

Post-Condition: The table is updated and the user is shown the modified selections which he has made

flow of Events: 1) The user selects some Tour packages

2) The user now wants to make some modifications

Exceptions: NA

5) Use Case ID: 4

Use Case Name: update personal details

Scenario: The user had entered some personal details and now wants to make modifications

Brief Description: The user has already entered personal details and now he/she wants to modify his/her personal details. The user is shown changes which have been made to his details

Primary Actor: user

Pre-Condition: The user had entered some personal information earlier

Post-Condition: The table is updated and the user is shown the modified personal information

flow of Events: 1) The user entered some personal information

2) The user now wants to make some modifications regarding his personal information

Exceptions: NA

6) Use Case ID: 5

Use Case Name: make payment

Scenario: The user has selected some tour package and now wants to make payment

Brief Description: The user has now selected some Tout Package and now he/she proceeds forward to make the payment. He/She will be provided with the available options

Primary Actor: user

Pre-Condition: The user has selected some packages earlier

Post-Condition: The user is given options regarding mode of payment

flow of Events: 1) The user selects some Tour packages

2) The user now wants to payment

3) The user is shown the available options for payment

Exceptions: NA

7) Use Case ID: 6

Use Case Name: perform validation

Scenario: The user has selected some packages , has proceeded to make payments

and now admin wants to validate the staff who will issue the Ticket

Brief Description: The admin needs to check the validity of the Staff who would finally issue the ticket to the user

Primary Actor: admin

Pre-Condition: The user has made payment

Post-Condition: The admin validates the staff who would issue the ticket

flow of Events: 1) The user selects some Tour packages

2) The user now wants to payment

3) The user is shown the available options for payment

4) The user makes the payment and admin validates the staff

Exceptions: NA

8) Use Case ID: 7

Use Case Name: cancel reservation

Scenario: The user has selected some packages , has proceeded to make payments

and now admin wants to validate the staff who will issue the Ticket

but now he wants to cancel his reservation

Brief Description: The user wants to cancels his reservations which he has made . At the end of this operation the user is shown remain reservations if any

Primary Actor: user

Pre-Condition: The user has made the reservation

Post-Condition: The reservation would be cancelled

flow of Events: 1) The user selects some Tour packages

2) The user now wants to payment

3) The user is shown the available options for payment

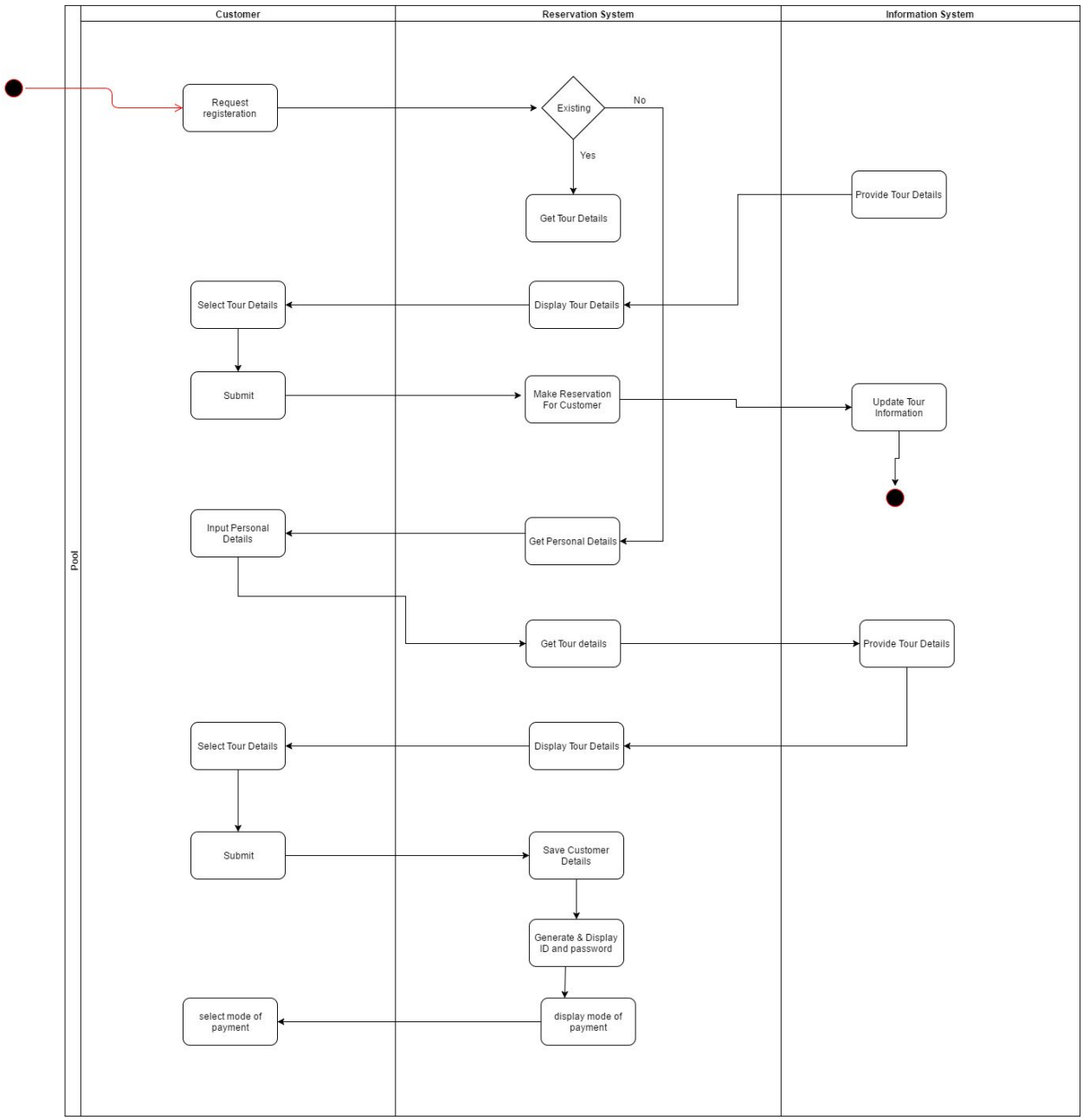
4) The user makes the payment and admin validates the staff

5) The reservation is made

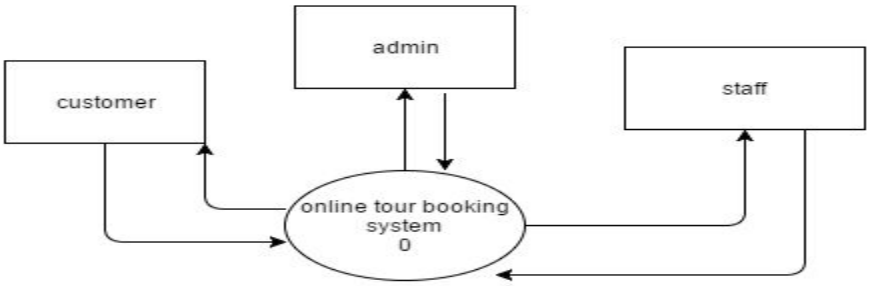
6) The user now cancels his reservation

Exceptions: NA

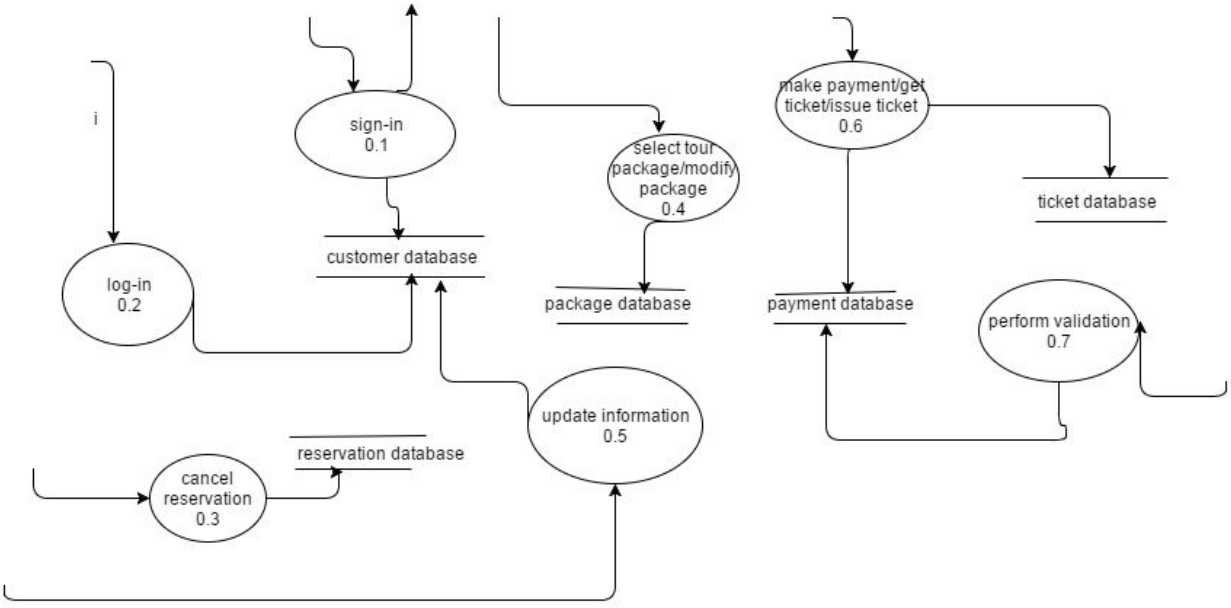
Activity Diagram:



Data Flow Diagram:



DFD 0



DFD 1

Data Dictionary 1:

A : Login Credentials: password + username

Username: String

Password: String

Sign-up: customer name + customer phone number

Customer name: String

Customer phone number: long int

Message: String

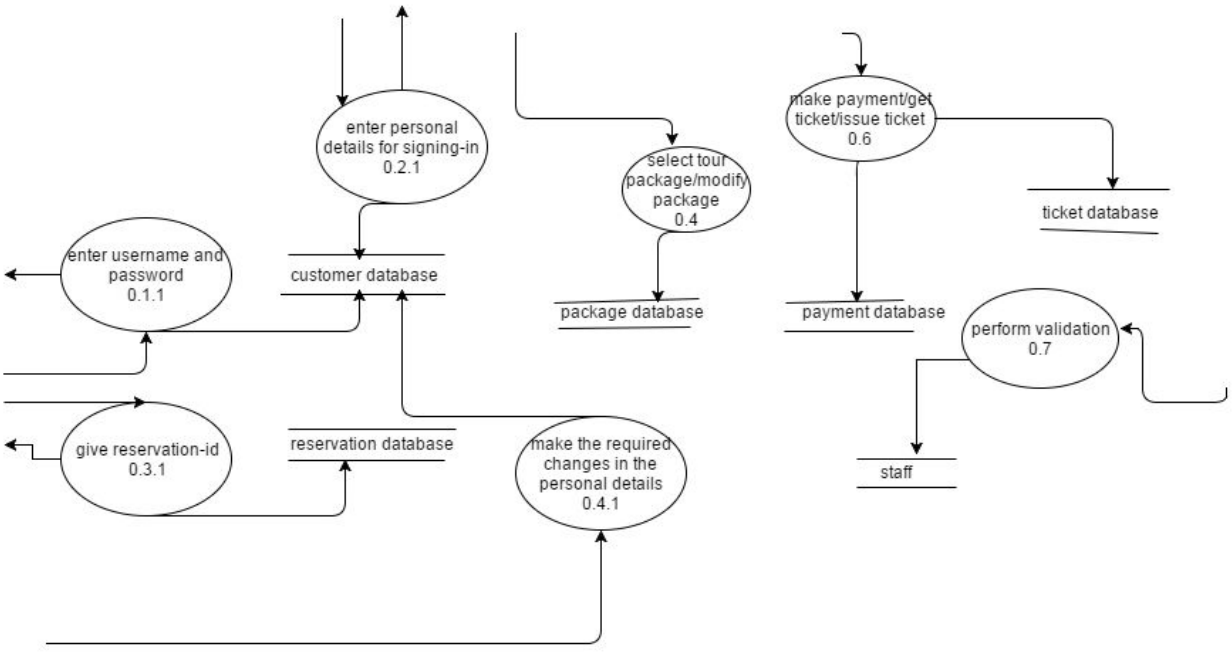
Clicked package: customer name + customer ID + package ID

Info to be updated: reservation ID

Payment details: reservation ID + payment type

Staff name : String

Reservation iD: Long int



DFD 2

Data Dictionary 1:

Login Credentials: password + username

Username: String

Password: String

Sign-up: customer name + customer phone number

Customer name: String

Customer phone number: long int

Message: String

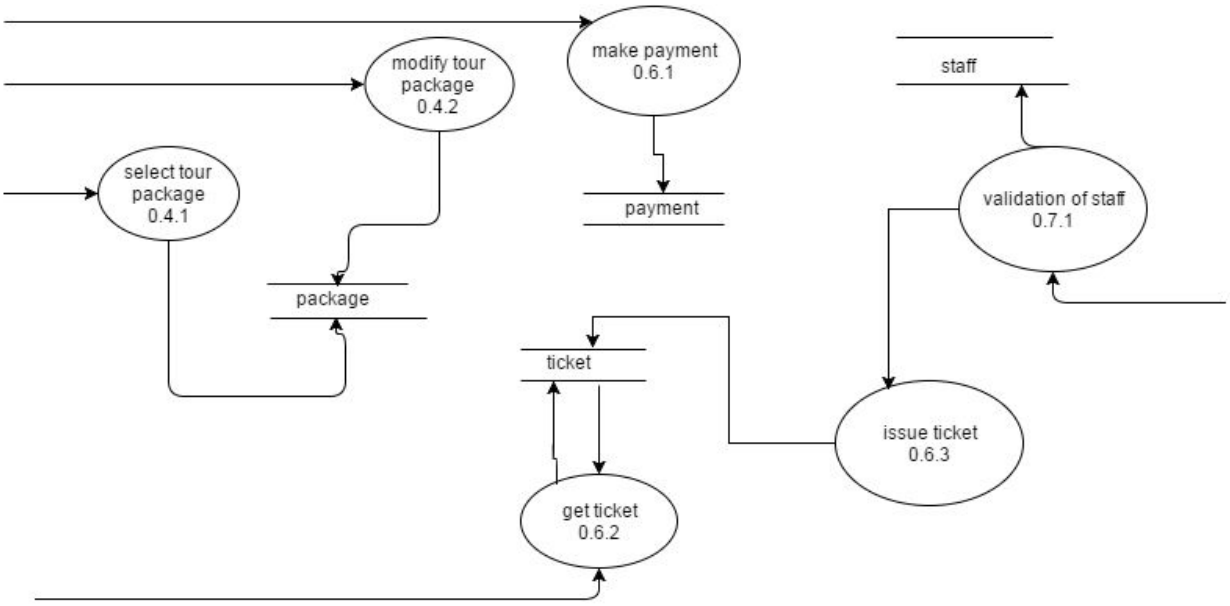
Clicked package: customer name + customer ID + package ID

Info to be updated: reservation ID

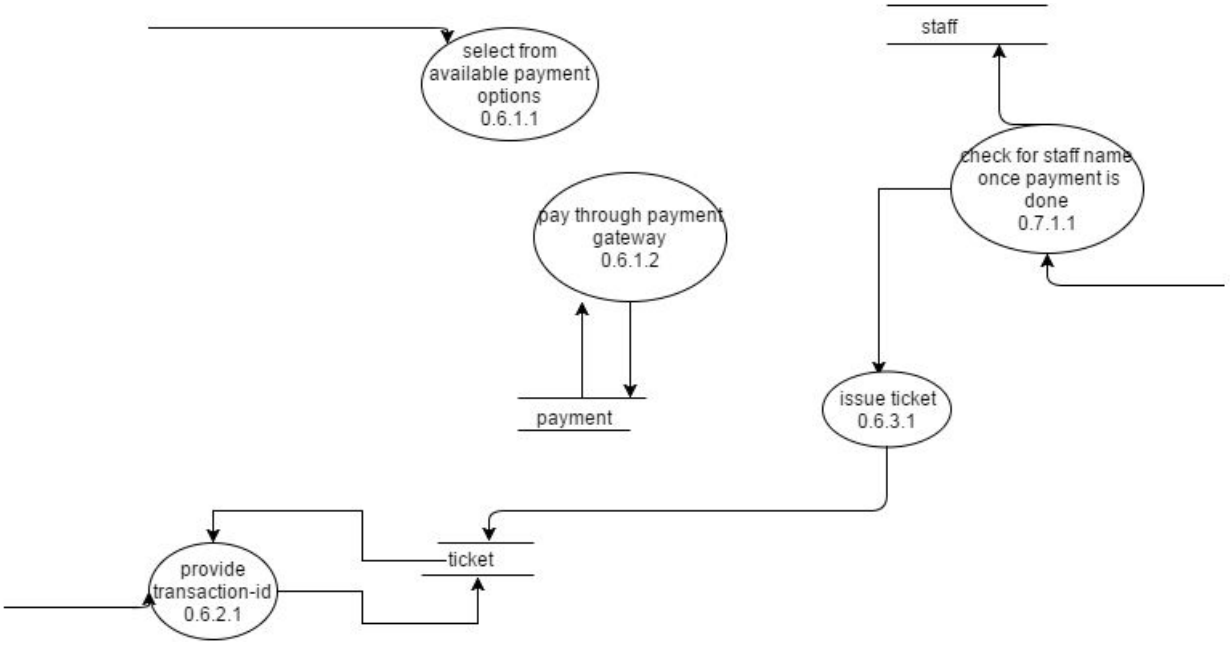
Payment details: reservation ID + payment type

Staff name : String

Reservation iD: Long int

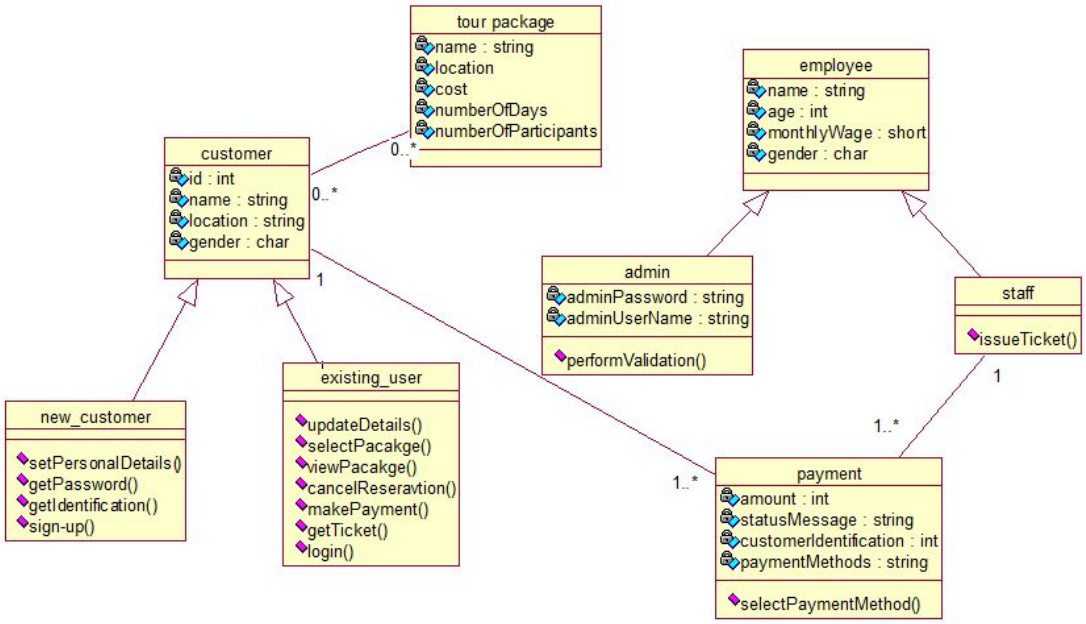

make payment
0.6.1
modify tour
package
0.4.2
staff
select tour
package
0.4.1
payment
validation of staff
0.7.1
package
ticket
issue ticket
0.6.3
get ticket
0.6.2

DFD 3


staff
select from
available payment
options
0.6.1.1
check for staff name
once payment is
done
0.7.1.1
pay through payment
gateway
0.6.1.2
payment
issue ticket
0.6.3.1
ticket
provide
transaction-id
0.6.2.1

DFD 4

Class Diagram:



Domain Class Diagram