**Document for creating a Bus Ticketing system:**

**Project Overview:**

Salesforce Bus Ticketing System reduce workloads for your staff and optimize customer service. These platforms can make sure that bookings are synced and the availability is updated with each booking processing.

This system includes various features which are mainly divided into Customization part and Configuration part.

Features:

1. The user can able to book the ticket from his/her mobile app feature of Salesforce and also he/she can select their seat as per the availability and simultaneously pay for it.
2. The user should see auto populated pay options.
3. System administrator should able to book any bus trip for the customers and also able to cancel it as per the customer’s requirement (NOTE: company policies should also take into considerations).
4. The customers can also share their feedback regarding their experience.

Before getting started, let’s discuss about the **overview of this Salesforce Bus Ticketing System**. This system is specially designed on cloud based platform which allow customers to place their booking at a time that is most convenient for them. Using an online booking system, they can make their booking at any time of day or night, without having to worry if it's within business hours or not.

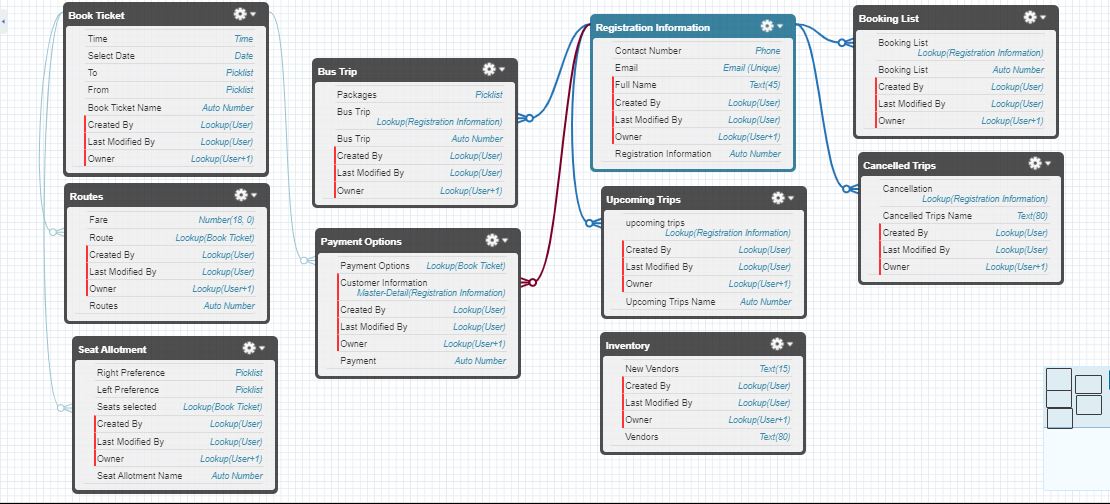
(NOTE: We’re considering only **one vendor** (who is connected with our system with their bus facility)for a time being. We can add additional requirements afterwards.)

**Data Model Overview:**

This overview will help knowing the data model of the Bus Ticketing System which consist of Objects, Fields and relationships between them. Below entity diagram will give you a glance regarding the overview:

**Entity Relationship Diagram:**

The ERD depicts all of the objects that play a significant role and are needed to understand the application requirements.



The above diagram shows the slight overview of objects and their relationships.

**Data Requirements:**

Temporally, we are creating this system between an end user and System Administrator.

Currently, all Users will be customers who can registered on bus ticket system or can view it.

On the other side, System Administrator will manage booking of bus trips and also cancellation of bus trips.

1. Book Ticket Object: Public Read/Write Sharing Model

This object represents that a customer can search for tickets in this section and can confirm it if he/she want to proceed further. They can select from where they want to start their journey and where to end. Also, they can select the date and times for specific travel.

1. Routes Object: This object is related to the Book Ticket Object. It shows which fare of the route selected.
2. Seat Allotment Object: This object is related to the Book Ticket Object. It shows the seats availability for specific route associated with it. Also this will provide preference to the customers between the seats they want to select.
3. Bus Trips Object: Public Read Only

This object is related to registration Information of the customer. Also, system Administrator can only make a use of this object for creating, managing the trips, managing the packages as per the customer’s requirement.

1. Payment options Object: Provide a payment gateway integrated with the salesforce. Also customers can select different preferences for paying for the booking.
2. Registration Information Object: This is not a compulsory registration for the customers. Customers still can view our app to check the availability or fare for the buses. Registration comes into scene when a customer wants to book a ticket further.

Moreover, the registration details of the customers will show related list object as well i.e., Upcoming trips, Cancelled bookings and list of bookings they have made until now.

1. Inventory object: This object is used for storing different routes, buses information, vendors who are connected with our company and availability of buses for booking trips.

(**Note**: Entity diagram might shows some fields but this documentation will give a detail view regarding process and its flow)

**User**: We have a created only one user which will act as a customer who can book tickets, view and edit his/her details.

**Profile**: The user which is created, he/she will assign with a profile giving access (Create, Update, Edit, Delete- CRUD) to some of the objects except Inventory, Bus Trips etc.

**# Let’s have a look at the flows of each objects in details:**

**START**

No **No**

**Step 1**: Able to search FROM-TO options, also can select date and time. Search button will redirect them to next page.

Can contact to support for booking Bus Trip or enquiry regarding packages of bus trips.

Able to view search options, can also check availability.

Just visiting website.

Want to book Bus Ticket?

**Yes**

**Step 4**: Customer will able to choose the payment options and then the custom button will redirect to payment gateway which is integrated with Salesforce.

**Step 3**: Seat selection options will be there depending upon the preference customer want and also availability.

**Step 2**: Selected routes will show the prices/Fare for specific booking per the availability.

**CONCLUDE**

**# if you are only going through Bus Ticket System App:**

Want to check registration details, booking details, upcoming trips, or Cancellation: Go to **Registration Information Object**

Upcoming Trips, Booking List, Cancellation details.

This object shows customer information.

#These are related Tab which will show other information.

**Logic Requirements:**

This section will provide the implemented and developed architecture which defines the described feature. These are implemented after constructing the app, objects and necessary Tabs and relationships.

1. **Booking Bus Ticket Properties:**

Let’s start considering that any user/customer can visit and book ticket. Therefore, if the customer/User come to visit the website, he/she didn’t go through registration process. He/she can view our few objects such as Book ticket, Seat allotment, Routes etc.

Now consider a scenario where the customer/user want to book Bus Ticket. For booking Ticket option, as we have created objects as per the flow described above the major thing is to keep in mind that what a user should see and can do with that data. There comes the role of Profile and permission sets i.e. visibility of Object Level Security and Field Level Security.

We will create a Visualforce Page to pass the parameter between these three pages simultaneously. Passing the parameter means that if the user have search for specific route and select a specific date and time which presents when he want to book the ticket will directly redirect them to the second page by clicking on command button created as Search Button which will create an action “{!gotosecondpage}”. This will also take into consideration the data which have been selected by the customer earlier, and help us to select the data from second and third page as well. Once we landed over the second page, the user will see the fare price for the selected route and he/she need to select the seat allotment preferences. After selecting the preferences they need to confirm details by clicking on Confirm command button which create a action “{!gotothirdpage}”. This third page will contain the payment preferences which contains a payment gateway that is integrated with the salesforce. After confirming all the details, the customer/user will receive confirmation message regarding the successful booking of their ticket.

This Visualforce page will be accessed through the salesforce mobile app as well. If the user logged in to their mobile app, they can fill the details with no hassle.

 To enable a Visualforce page for Salesforce1:

* From Setup, click Develop | Pages.
* Click Edit for the desired Visualforce page.
* Select Available for Salesforce mobile apps then click Save.

**# integrating any payment Gateway with salesforce:**

• First you will make a token call (https://login.salesforce.com/services/oauth2/token) with the Consumer Key & Consumer secret.

• Then you will receive Request Token which you will add it in the header i.e. [Authorization = Bearer + Access Token].

• Then you will make a REST API Call with Get method to the Endpoint URL - <https://ap15.salesforce.com/services/data/v48.0/sobjects/yourobjectname> . Replace ap15 with your salesforce Instance and 48.0 with your version number. Then you will get whatever you have requested. The Endpoint URL - https://ap15.salesforce.com/services/ remains same till /services, after that it changes depending upon the Requirement.

\*\*\*NOTE - Here you’re making GET Request (For getting any data from your Salesforce Org to your Third Party App- here POSTMAN).For POST Requests you will need to add Content-Type also in the Header

1. **System Administrator Properties:** Modify All data Sharing Modal

Manage the requests of Bus trips i.e. suggest best packages as per customer needs and also able to see the list of customers in an attachments.

Will send push notifications to customers regarding cancellation of their booking for Bus .trips

Manage and maintain the functionality of the whole org

Also will able to cancel bus trips as per the customers’ requests.

System

Administrator