A designer can mull over complicated designs for months. Then suddenly the simple, elegant, beautiful solution occurs to him. When it happens to you, it feels as if God is talking! And maybe He is.

*—Leo Frankowski*

## Table of Contents

# Table of Figures

# Question 1

A brief (single page) high-level outline of the system you suggest, identifying high-level functionality, static data storage requirements (do not produce a detailed account of static data at this stage), and characterising the kinds of user of the system.

*Answer:*

## Overall description

The basic goal and objective is to develop a web based computerized system that will provide an easy, efficient and reliable way to manage data and processes for the Playhouse Theatre. The goal of the web system is to bring more flexibility and reliability to the system by automating it and incorporating the following features in order to solve the current problems:

- Providing an anytime anyplace booking, returns and updates service for the customer via web.
- To provide a proper web interface to agencies for retrieving banned customers lists, returning tickets and updating the booking orders. A better way to reconcile returned tickets and avoid any situations like double bookings.
- Effective and efficient way to handle the discounts on pricing system for the customers and remunerations for the staff.
- Provision of an automatic ticket issuer at box office which would help in getting rid of the long customer queues and thus provide a reliable way to count the number of people present in a performance. Also, providing an opportunity for various other agencies to resell tickets as the automatic ticket issuer will handle the printing of ticket solely.
- Minimizing the number of staff at the ticket box.
- By keeping customers updated with mailings lists.
- Introducing a better way to resolve waiting lists for bookings.
- Providing report generation facility for the sales & returns.
- Introducing the recalling tickets feature to keep the customers informed if any changes or delays have occurred for the performances they have bought tickets for.
- A reliable and efficient way of refunding tickets.
- An efficient way to change seating arrangements for each performance by maintaining a seating plan against every performance.

## System Features

1. Online ticket booking & returns and profile management for members that comprises of:
    - Making a booking/buying a ticket
    - Cancel booking
    - Update booking
    - Update profile information

- Update account information
- View account & profile details
- View booking details
- Browse and search for plays information
- Subscribe/unsubscribe to mailing lists

2. Online ticket booking & returns and profile management for agencies that comprises of:
   - Confirming booking orders (updating booking orders database)
   - Returning tickets
   - Viewing banned customers

3. Online browse and search options for play information for online visitors
4. Online Subscribing option to mailing lists for online visitors
5. Online box office booking to manage booking & returns for booking staff that includes:
   - Checking seat availability
   - Booking ticket
   - Invoicing
   - Reconciling returned tickets
   - Automatic ticket issuing
   - Checking booking details
   - Resolving waiting lists
   - Refunding.

6. Online System management to:
   - Maintain mailing list
   - Recalling tickets
   - Manage banned customers
   - Update seating plan as per performance & booking requirements
   - Manage discount patterns & ticket types
   - Manage price coding & ticket information
   - Report generation for ticket sales & returns
   - Maintain and update all database and users.

**Non-functional**

1. Importing the previous manual data into the database successfully
2. Daily automated back up for the sales & returns data
3. Login facility for every user
4. Online help including FAQs and 'help form' services.

<u>**System Users**</u>

1. **System Managers:** They would be responsible for maintaining system safety and security, daily back ups, report generation and any updates to the database like changing seating plans per performance, managing discount changes etc (as mentioned above in system management feature).

2. **Booking staff:** They are responsible for managing the ticket sales, returns and invoicing process.

3. **Customer:** The customers can access the booking system via web for making booking, canceling booking, updating customer info, checking booking details, browse for plays information and may subscribe or unsubscribe to the mailing list and other features.

4. **Agencies:** They can use the web interface for updating data for returning tickets, confirming booking orders, retrieve list of banned customers.

5. **Credit Card verifier:** This will be an external user/entity integrated to work with the system whose services would be required to verify the credit card of the customer.

**Static data storage requirements**

A relational database to store all the information pertaining to:

- Booking orders
- Returns
- Customers
- Agencies
- Performance
- Seating plan
- Ticket (ticket types to handle discounts and pricing codes)
- Venue
- Ticket requests (waiting lists)
- Mailing lists
- Issued tickets

## Question 2

Data flow diagrams relating to the activity of ***making a theatre booking***. You should present several, consistent, diagrams with increasing granularity, identifying: processes, inputs, outputs, data flows and static data stores.

*Answer:*

Assuming that the DFDs are required for the process of booking ticket/theatre booking as mentioned in the question so the following chain of DFDs have been detailed according to keeping in mind the booking ticket process chiefly.

**Customer**

Ticket details

ticket request

**Ticket Booking System**
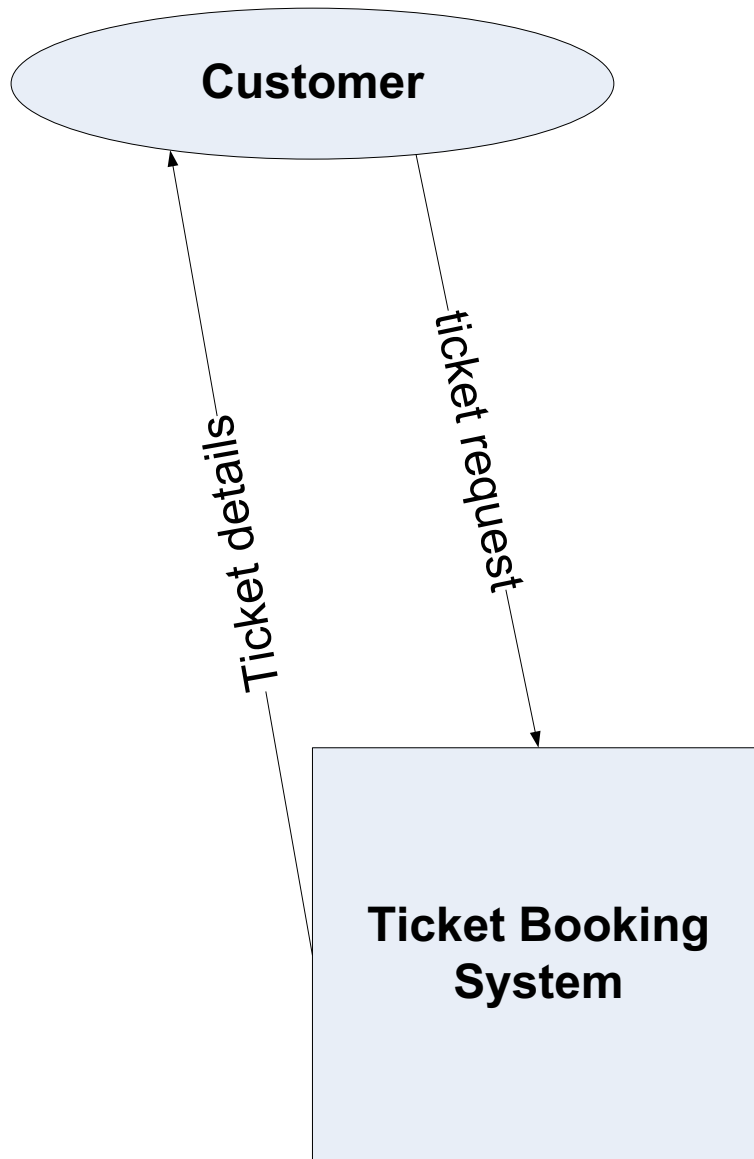
**Figure 1: Context Diagram – Ticket booking System**

The context diagram shows a customer requests for a ticket and if successful he is issued a ticket.
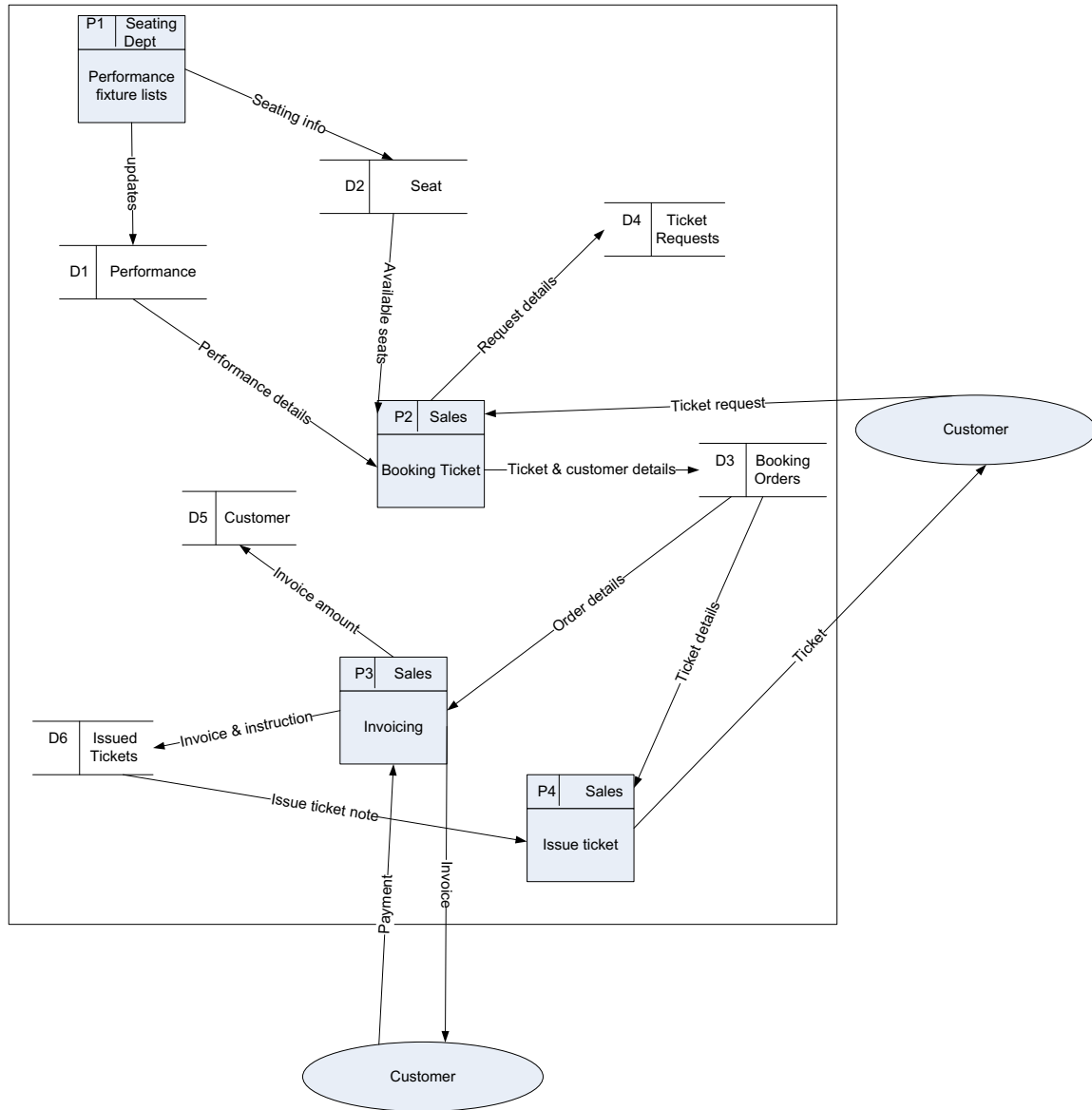
**Figure 2: Level 1 DFD - Ticket Booking System**

The above level 1 data flow diagram is for the *'ticket booking system'*. It is missing a few processes as the main focus in this question is the *activity of making a theatre booking*. Returns and cancellations haven't been handled.

The system managers (seating department) adjust the seat database on the basis of performance requirements.

A customer requests for booking ticket. The booking staff (sales dept) checks for available seats, inputs performance details and books the order. The invoice is given to the customer and the payment is made.

If no available seat is found then the request is added to the waiting list (ticket requests database). This waiting list is resolved on basis of returns of ticket from customers and agencies.

If customer is a member of the theatre then the invoice amount and order details are saved to his account as well. The ticket is then issued to the customer, but if he is making an online booking then he is given a ticket reference number.

The automated ticket issuer at the box office requires the customer to enter the ticket reference number and his password and the ticket is printed and collected by him. The security and issuing requirements basically depend on the business logic. I assume the ticket reference number and member password would be a good way to handle issuing ticket process for online bookings.

The booking process is analyzed more closely in the DFDs below:
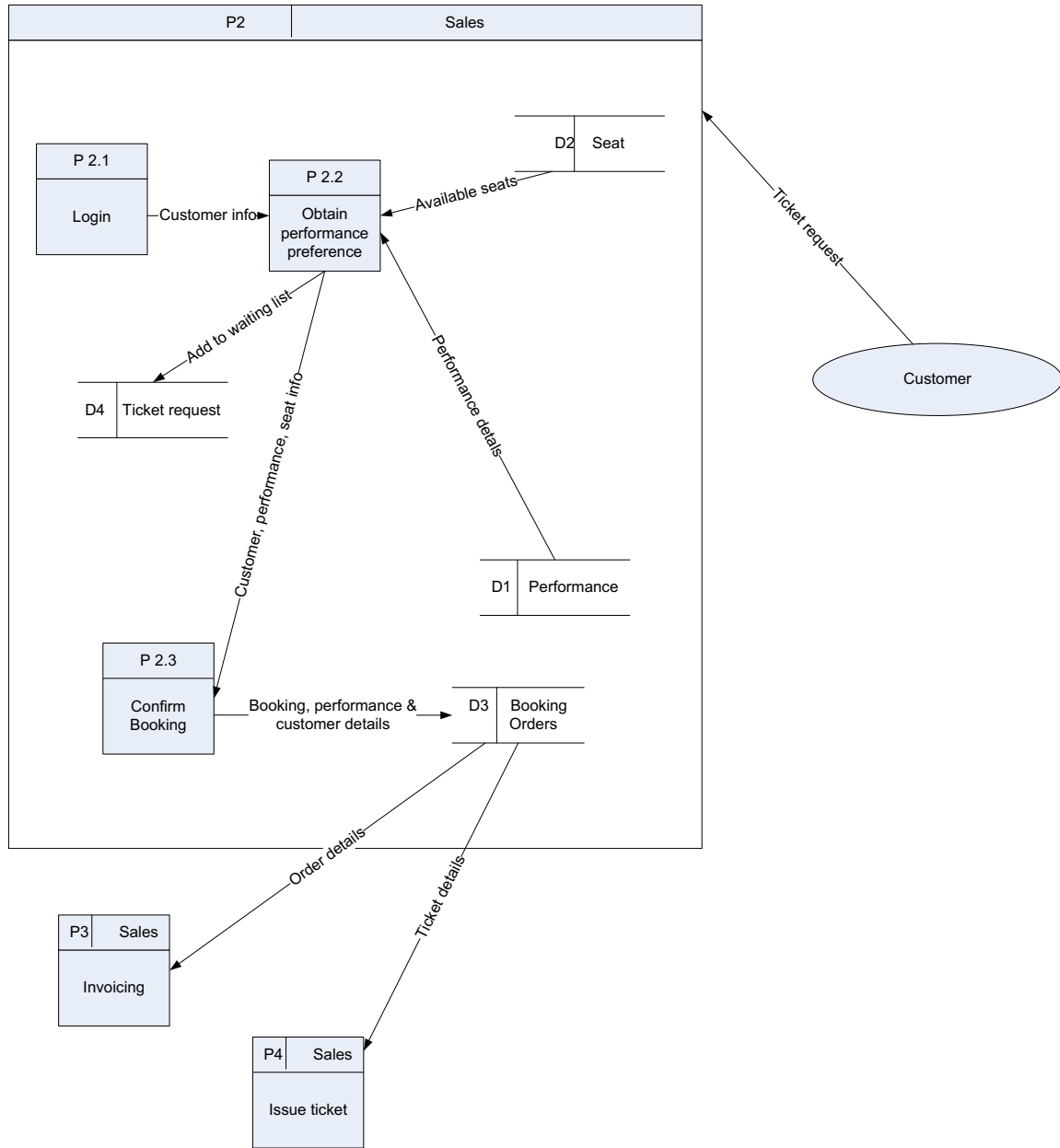
**Figure 3: Level 1 DFD - 'Booking ticket' process**

The above DFD takes a closer look of the '***booking ticket***' process. If the process is being done at box office, then the booking manager logins into the system, checks for the available seats, enters customer information, his preferred choice for performance and confirms booking.

In case of online booking, the member would login, select the performance details and make a booking.

In case the performance tickets are sold out, the customer is being added to the waiting list if he desires.
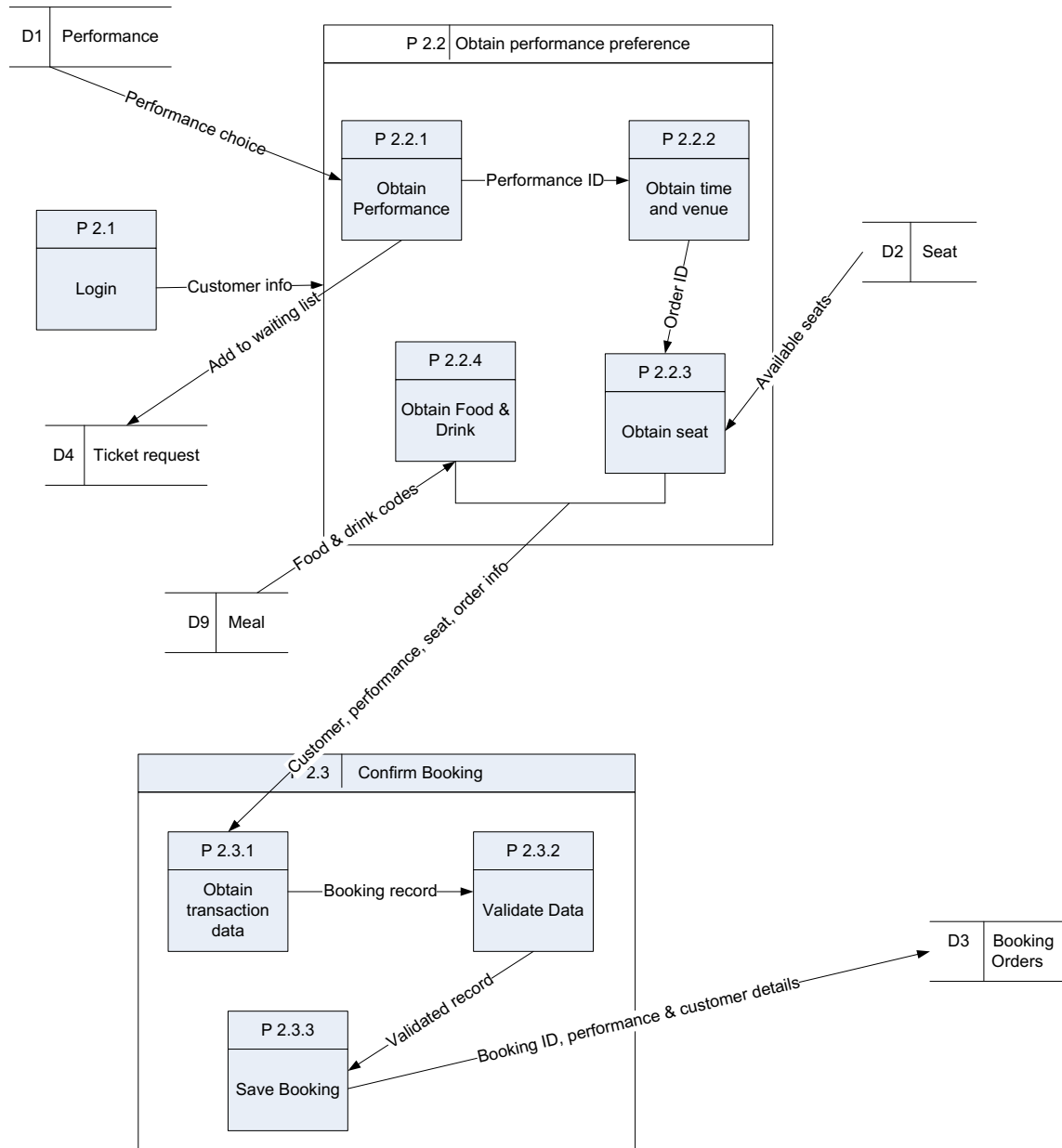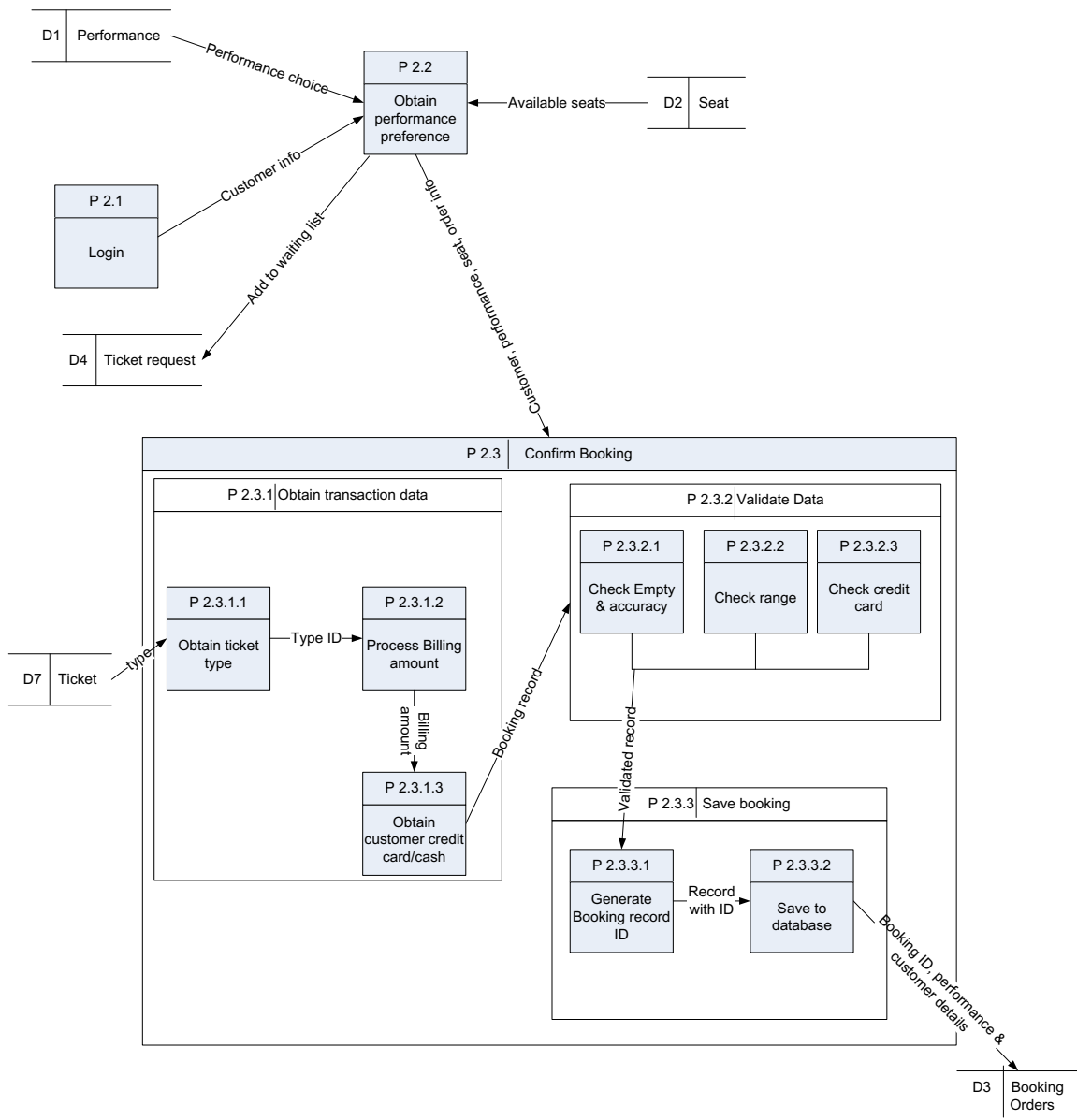
**Figure 4: Level 2 DFD - 'Booking ticket' process**

In the above DFD we closely investigate the processes of *'choosing performance preference'* and *'confirm booking'*.

### 'Choosing performance preference'

The performance is selected from the list and its time and date are obtained. The choice of seat is entered and any food and drinks orders are also entered. Once done the booking information is confirmed.

### 'Confirm Booking'

The process of 'confirm booking' involves the process of obtaining the booking record and then it is checked for correctness & accuracy. Once validated the booking information is written to the booking orders database.



**Figure 5: Level 3 DFD - 'Booking ticket' process**

In the above DFD, the following 3 processes involved in *'confirm booking'* are investigated deeply.

### *'Obtain transactional data'*

Once all the booking data is received it is time to calculate price and bill the customer. So the process of 'obtaining transaction data' involves getting the price info of the 'ticket type' that is selected, in short it gets the price code for the selected preferences of user. The 'ticket type' would handle all the discounts offered. Next the billing amount is calculated and the customer gives cash or credit card details.

### *'Validate Data'*

The booking information record is passed on to the process of 'validate data'. In the sub process of 'validate data' the booking record is checked for correctness of values entered. The values are checked to ensure that they are accurate, non- empty and fall with in range. The credit card is also verified via an external entity.

### *'Save booking'*

Once the data is validated the booking record is passed on to the 'save booking' process. Here the ID for the booking record is generated. Just a note, the booking record that was being passed on earlier was without an ID. The booking record with an ID is then inserted to the booking orders database and the booking process is hence complete.

Assuming that the booking procedure of agency is not a part of this system but the agencies would be provided with a web interface where they could update the booking orders, check for banned customers list, check available seats and return tickets.

# References

**[Pressman, 1997]**

Roger S. Pressman, "*Software Engineering – A Practitioner's Approach*", McGraw Hill 1997, p. 278.


**[Sommerville, 1992]**

I. Sommerville*, Software Engineering*, Addison- Wesley, 1992.