QUEZON CITY UNIVERSITY

COLLEGE OF COMPUTER STUDIES

Computer Science | Information Technology | Information Systems

Airline Ticket Booking and Management System

Group 7:

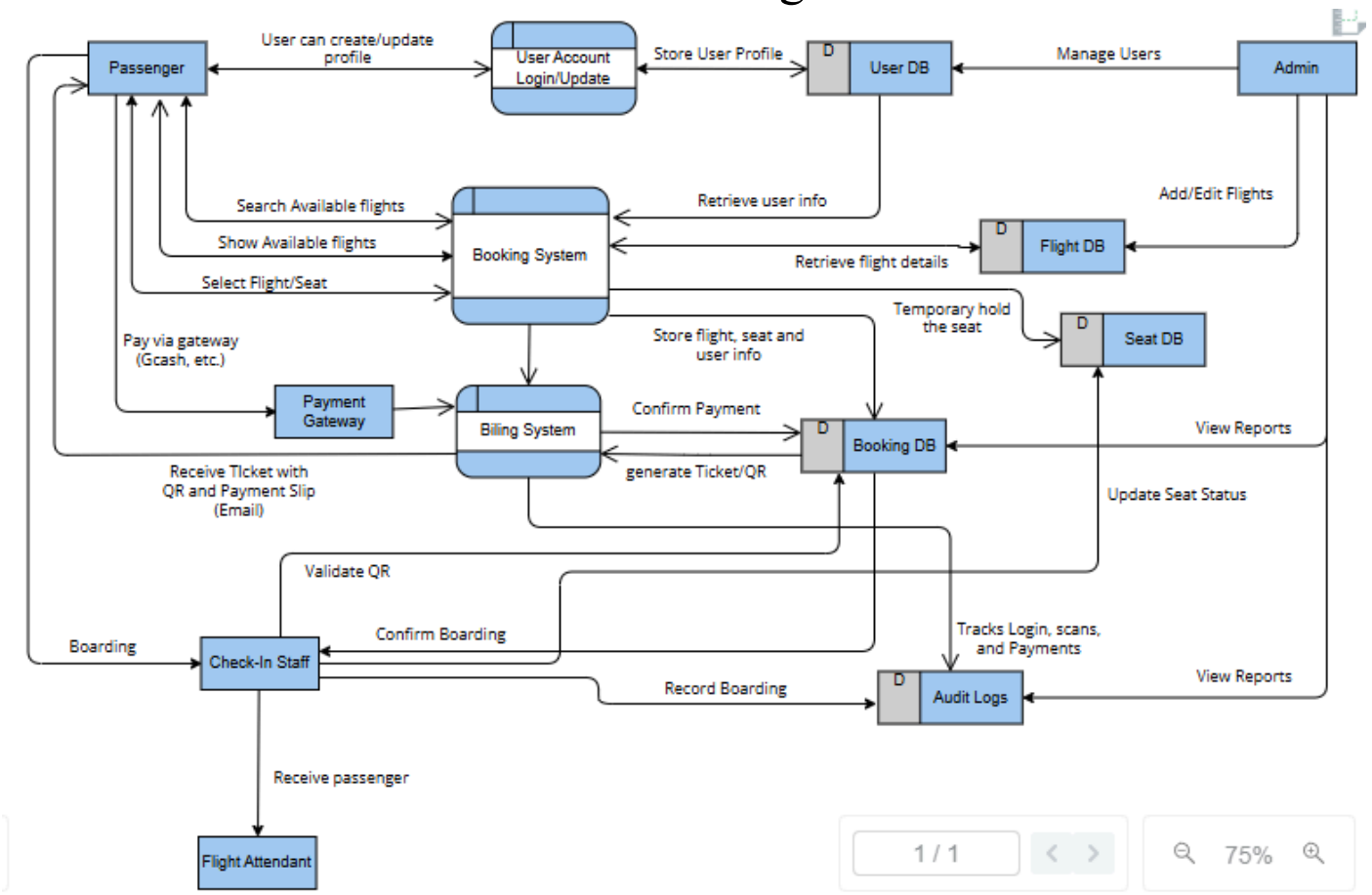
Padilla, Adrian B.
Pelegrino, Gerimel M.
Quirap, Kurt Lawrenze B.
Seña, Jerald U.
Sid-ay, Mark John P.

Database design

Objectives/ features

- **User Management:** Register, log in, and manage user profiles securely.
- **Flight Search and Booking:** Search for flights based on various criteria, book tickets securely, and make payments through compliant gateways.
- **QR Code Verification:** Generate and scan unique QR codes for each booked ticket for efficient verification at check-in or boarding gates.
- **Admin Panel:** Manage flight listings, user information, and system configurations efficiently.
- **Responsive Design:** Enjoy a seamless experience across different devices, including desktops, laptops, tablets, and smartphones.

Data Flow Diagram



Data Dictionary

ofbsphp

admin

Column	Type	Null	Default	Comments
admin_id *(Primary)*	int(11)	No		
admin_uname	varchar(20)	No		
admin_email	varchar(50)	No		
admin_pwd	varchar(100)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	admin_id	0	A	No	

airline

Column	Type	Null	Default	Comments
airline_id *(Primary)*	int(11)	No		
name	varchar(20)	No		
seats	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	airline_id	11	A	No	

cities

Column	Type	Null	Default	Comments
city	varchar(20)	No		

No index defined!

feedback

Column	Type	Null	Default	Comments
feed_id *(Primary)*	int(11)	No		
email	varchar(50)	No		
q1	varchar(250)	No		
q2	varchar(20)	No		
q3	varchar(250)	No		
rate	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	feed_id	0	A	No	

flight

Column	Type	Null	Default	Comments
flight_id *(Primary)*	int(11)	No		
admin_id	int(11)	No		
arrivale	datetime	No		
departure	datetime	No		
Destination	varchar(20)	No		
source	varchar(20)	No		
airline	varchar(20)	No		
Seats	varchar(110)	No		
duration	varchar(20)	No		
Price	int(11)	No		
status	varchar(6)	Yes	*NULL*	
issue	varchar(50)	Yes	*NULL*	
last_seat	varchar(5)	Yes		
bus_seats	int(11)	Yes	20	
last_bus_seat	varchar(5)	Yes		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	flight_id	21	A	No	
admin_id	BTREE	No	No	admin_id	2	A	No	

passenger_profile

Column	Type	Null	Default	Comments
passenger_id *(Primary)*	int(11)	No		
user_id	int(11)	No		
flight_id	int(11)	No		
mobile	varchar(110)	No		
dob	datetime	No		
f_name	varchar(20)	Yes	*NULL*	
m_name	varchar(20)	Yes	*NULL*	
l_name	varchar(20)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	passenger_id	6	A	No	
user_id	BTREE	No	No	user_id	6	A	No	
flight_id	BTREE	No	No	flight_id	6	A	No	

payment

Column	Type	Null	Default	Comments
card_no *(Primary)*	varchar(16)	No		
user_id	int(11)	No		
flight_id	int(11)	No		
expire_date	varchar(5)	Yes	*NULL*	
amount	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	card_no	6	A	No	
user_id	BTREE	No	No	user_id	6	A	No	
flight_id	BTREE	No	No	flight_id	6	A	No	

pwdreset

Column	Type	Null	Default	Comments
pwd_reset_id *(Primary)*	int(11)	No		
pwd_reset_email	varchar(50)	No		
pwd_reset_selector	varchar(80)	No		
pwd_reset_token	varchar(120)	No		
pwd_reset_expires	varchar(20)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pwd_reset_id	0	A	No	

ticket

Column	Type	Null	Default	Comments
ticket_id *(Primary)*	int(11)	No		
passenger_id	int(11)	No		
flight_id	int(11)	No		
user_id	int(11)	No		
seat_no	varchar(10)	No		
cost	int(11)	No		
class	varchar(3)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	ticket_id	6	A	No	
user_id	BTREE	No	No	user_id	6	A	No	
flight_id	BTREE	No	No	flight_id	6	A	No	

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
passenger_id	BTREE	No	No	passenger_id	6	A	No	

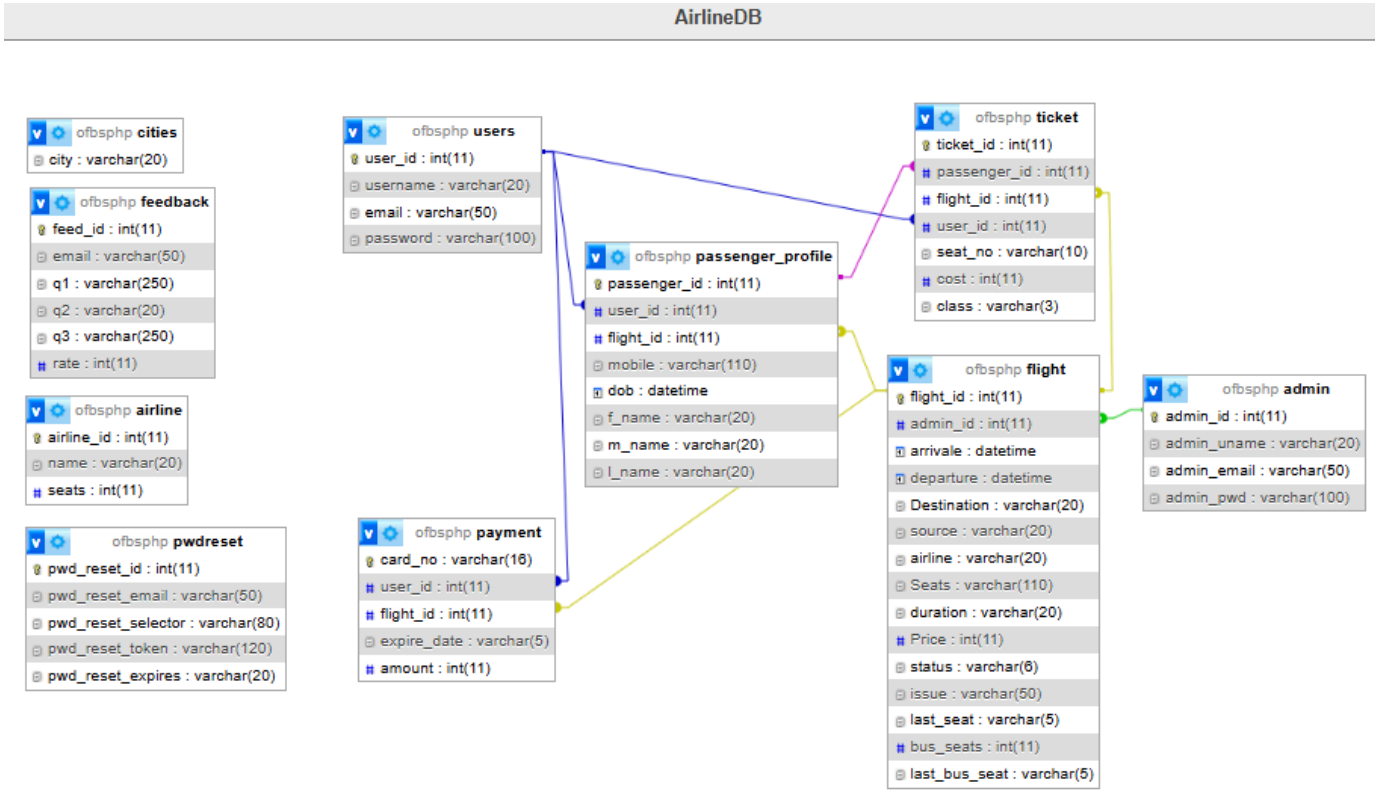
users

Column	Type	Null	Default	Comments
user_id *(Primary)*	int(11)	No		
username	varchar(20)	No		
email	varchar(50)	No		
password	varchar(100)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	user_id	4	A	No	

Schema



SiteMap

Site Map: Airline Ticket Booking System

1. Home

|-- User Account Login/Update
| ↳ Create/Update Profile
|-- Search Available Flights
| ↳ Show Available Flights
| ↳ Select Flight/Seat

|-- Payment Gateway
| ↳ Pay via Gateway (Gcash, etc.)
|-- Billing System
  ↳ Confirm Payment
  ↳ Generate Ticket/QR

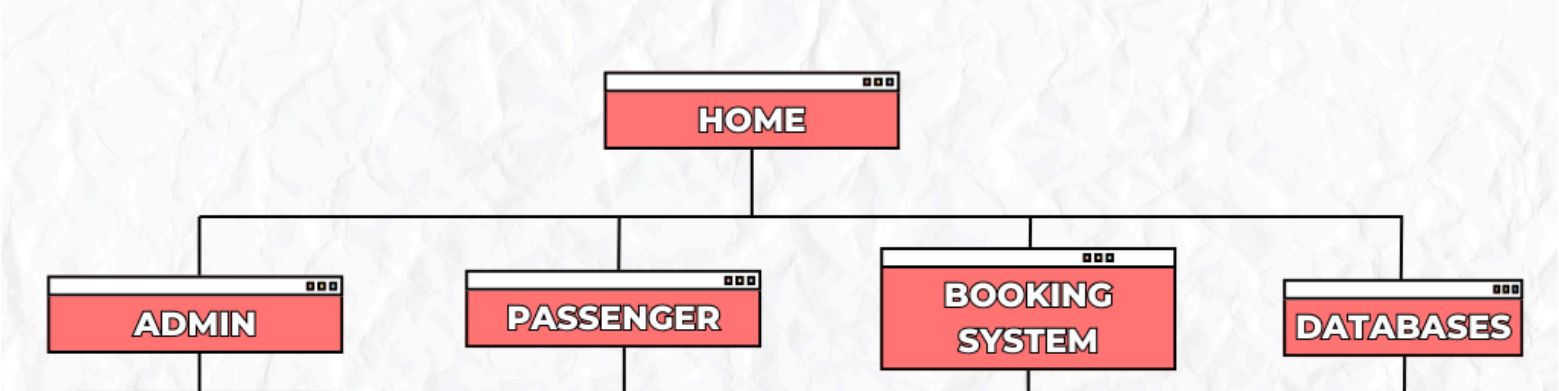
2. Check-in
|-- Check-in Staff
| ↳ Validate QR
| ↳ Confirm Boarding
| ↳ Record Boarding
|-- Flight Attendant
  ↳ Receive Passenger

3. Admin Panel
↳ Manage Users
↳ Add/Edit Flights
↳ View Reports

4. Booking System
↳ Retrieve User Info
↳ Retrieve Flight Details
↳ Store Flight, Seat, and User Info
↳ Temporary Hold the Seat
↳ Update Seat Status

5. Databases
↳ User DB
↳ Flight DB
↳ Seat DB
↳ Booking DB
↳ Audit Logs
  ↳ Tracks Login, Scans, and Payments
  ↳ View Reports

VTOC (Visual Table of Contents)



QUEZON CITY UNIVERSITY

COLLEGE OF COMPUTER STUDIES

Computer Science | Information Technology | Information Systems

1. Passenger

- User Account Login/Update → Store User Profile
- Booking System → Search Available Flights → Show Available Flights → Select Flight/Seat
- Payment Gateway → Pay via Gateway (Gcash, etc.)
- Billing System → Confirm Payment → Generate Ticket/QR
- Check-in Staff → Validate QR → Confirm Boarding → Record Boarding
- Flight Attendant → Receive Passenger

2. Admin

- Manage Users
- Add/Edit Flights
- View Reports

3. Booking System

- Retrieve User Info
- Retrieve Flight Details
- Store Flight, Seat, and User Info
- Temporary Hold the Seat
- Update Seat Status

4. Databases

- User DB
- Flight DB
- Seat DB
- Booking DB
- Audit Logs
    - Tracks Login, Scans, and Payments
    - View Reports

Programming language design

Components	Specification	Description
1.OS	Windows	Windows OS, developed by Microsoft, is the most widely used operating system for personal computers. It's a graphical operating system (GUI) that provides a user-friendly interface for interacting with a computer, allowing users to perform tasks like browsing the internet, managing files, and running software. We use the Windows Operating System, Because most of the website and programs are mainly usable in windows or in a laptop or computer devices. And the website we have created are mostly used and created in a Windows System.
2.Programming Language	PHP	PHP is a widely used, open-source, server-side scripting language primarily used for web development. It can be embedded within HTML, making it ideal for creating dynamic web pages and applications. PHP can also be used for command-line scripting and desktop applications. We decided to use the PHP Programming language, because it is more flexible and we are more familiar with using it to create a website. It is also easy to navigate and easier to create a dynamic web pages, which is the same to our website
3.Backend	XAMPP PHPMyAdmin	XAMPP is primarily used as a free, open-source local web server for development and testing purposes, particularly for web applications that utilize PHP and other web technologies. It essentially provides a simulated online environment on your computer, allowing you to build, test, and debug websites and applications without the need for a live server. For our Backend, we used XAMPP because we are more familiar and we know how to navigate and use it. It is also easier to connect to a website as its database. It is also more easier to debug and the web servers available are easier to utilized, without needing a live server.
4.Frontend	PHP, HTML, CSS, Javascript	We used the PHP, for frontend to properly navigate the servers easier and cleaner scripting process in creating the server for our website, PHP offers us more wide codes and scripts to use to further enhance our websites specially for the server. HTML means HyperText Markup Language, is the foundation of the World Wide Web. It's the standard language for creating web pages and structured content using tags and elements. HTML

		describes the structure and content of a web page, telling browsers how to display it. HTML provided us in creating the base and the whole body of our code, it provided us to create the structure of our frontend design. CSS or Cascading Style sheets, provided us the power to add color and design to our airline website. Also giving us our own unique design and style to make our website more engaging and unique to other websites. JavaScript is a versatile, high-level, and dynamic programming language primarily used to add interactivity to websites. It is an essential technology for front-end web development, allowing developers to create dynamic and engaging user interfaces. However, its capabilities extend beyond the client-side, enabling server-side development with Node.js and the creation of mobile applications, desktop applications, and even games Javascript also helps our website become more accessible and make the whole website interactive to provide the users full interactive and accessible experience in our website.