Creating a full-service reservation application requires a significant amount of code, and it's beyond the scope of a simple response.

Here's an outline of the major components and their corresponding code:

**1. Setting up a Web Framework:**

You can choose a web framework like Flask, Django, Ruby on Rails, or Node.js with Express.js for the backend. Below is an example using Flask:

from flask import Flask, render\_template, request, redirect, url\_for

app = Flask(\_\_name)

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**2. Database Setup:**

You'll need a database to store information about services, users, bookings, and other relevant data. SQLite or PostgreSQL is a good choice for a small to medium-sized application. In this example, we'll use SQLite:

import sqlite3

conn = sqlite3.connect('service\_booking.db')

cursor = conn.cursor()

**3. User Authentication:**

Implement user registration and login. You can use Flask-Login for this purpose in a Flask application.

from flask\_login import LoginManager, UserMixin, login\_user, login\_required, current\_user, logout\_user

login\_manager = LoginManager()

login\_manager.init\_app(app)

**4. Service Management:**

Implement functionality for adding, listing, and managing services. You'll need routes and views for these operations.

@app.route('/add\_service', methods=['POST'])

@login\_required

def add\_service():

@app.route('/services')

def list\_services():

@app.route('/service/<service\_id>')

def view\_service(service\_id):

**5. Booking System:**

Create routes and views to allow users to book services.

@app.route('/book/<service\_id>', methods=['POST'])

@login\_required

def book\_service(service\_id):

@app.route('/bookings')

@login\_required

def list\_bookings():

# List a user's bookings

# ...

**6. Frontend:**

Create HTML templates and use a frontend framework like React, Angular, or Vue.js for the user interface. These templates will integrate with your Flask application through routes and render data from the backend.

**7. Payment Processing:**

If service booking application involves payments, we must integrate a payment gateway like Stripe, PayPal, or Braintree. Each payment gateway has its own API and documentation for integration.

This is a high-level outline of what you'd need to build a service booking application. The actual code for each component can be quite extensive, and it's important to plan and structure your application well. Additionally, consider security, scalability, and user experience in your development process.