

Αρχιτεκτονική της Εφαρμογής (3-Επιπέδων)

Ροή Εφαρμογής

1. Ο χρήστης επιλέγει προορισμό, αριθμό ατόμων και ημέρες
2. Ο χρήστης συνδέεται ή εγγράφεται
3. Το frontend στέλνει τα δεδομένα στο TravelPlanServlet.
4. Ο TravelPlanServlet επεξεργάζεται τα δεδομένα, καλεί τις μεθόδους για να δημιουργήσει το πρόγραμμα και το αποθηκεύει στη βάση δεδομένων.
5. Το servlet ανακτά το πρόγραμμα και τα σημεία ενδιαφέροντος.
6. Το frontend εμφανίζει το πρόγραμμα ταξιδιού μαζί με το χάρτη και τις πινέζες.

Χάρτης με Σημεία Ενδιαφέροντος

Ενσωμάτωση με Google Maps JavaScript API

1. Presentation Layer (Επίπεδο Παρουσίασης):

- **HTML/CSS/JSP:** Για την εμφάνιση της διεπαφής χρήστη, όπου οι χρήστες θα εισάγουν τις πληροφορίες τους (προορισμός, αριθμός ατόμων, ημέρες, προτιμήσεις) και θα βλέπουν το εξατομικευμένο πρόγραμμα ταξιδιού.

2. Business Logic Layer (Επίπεδο Λογικής Επιχείρησης):

- **Servlets:** Υλοποιούν τη λογική για την επεξεργασία των δεδομένων του χρήστη, τη δημιουργία και την ανάκτηση του προγράμματος ταξιδιού από τη βάση δεδομένων.

3. Data Access Layer (Επίπεδο Πρόσβασης Δεδομένων):

- **Βάση Δεδομένων (MySQL):** Αποθηκεύει τις πληροφορίες σχετικά με τους χρήστες, τα προγράμματα και τα σημεία ενδιαφέροντος.

Εφαρμογή

1) Διεπαφή Χρήστη:

- Φόρμα Εγγραφής/Σύνδεσης (Login/Signup).
- Φόρμα για την επιλογή προορισμού, αριθμού ατόμων, ημερών και προτιμήσεων.
- Παρουσίαση του προγράμματος ταξιδιού και του χάρτη με τα σημεία ενδιαφέροντος.

2. Servlets: Δημιουργία servlets που θα χειρίζονται τις αιτήσεις από τον πελάτη (frontend) και θα εκτελούν τη λογική της εφαρμογής.

- **UserServlet:** Διαχειρίζεται τις αιτήσεις για σύνδεση/εγγραφή χρηστών.
- **TravelPlanServlet:** Δημιουργεί το ταξιδιωτικό πρόγραμμα με βάση τις εισόδους του χρήστη.
- Επαλήθευση των στοιχείων σύνδεσης/εγγραφής.

- ο Δημιουργία και αποθήκευση του προγράμματος ταξιδιού στη βάση δεδομένων.
- ο Ανάκτηση δεδομένων από τη βάση δεδομένων για εμφάνιση.

3. DataBase

- **Βάση Δεδομένων:** Χρήση MySQL
 - ο **Πίνακες:**
 - Users: Για την αποθήκευση στοιχείων χρηστών.
 - TravelPlans: Για την αποθήκευση προγραμμάτων ταξιδιών.
 - Destinations: Για την αποθήκευση προορισμών και σημείων ενδιαφέροντος.

ένα servlet για το login /sign up

@WebServlet("/user")

```
public class UserServlet extends HttpServlet {

    protected void doPost(HttpServletRequest request, HttpServletResponse response)

        throws ServletException, IOException {

        String action = request.getParameter("action");

        if ("signup".equals(action)) {

        } else if ("login".equals(action)) {

        }

    }

}
```

Ένα για τη δημιουργία προγράμματος

@WebServlet("/travelPlan")

```
public class TravelPlanServlet extends HttpServlet {

    protected void doPost(HttpServletRequest request, HttpServletResponse response)

        throws ServletException, IOException {

        // Λήψη δεδομένων από τη φόρμα

        int userId = Integer.parseInt(request.getParameter("userId"));

        int destinationId = Integer.parseInt(request.getParameter("destinationId"));

        int numberOfPeople = Integer.parseInt(request.getParameter("numberOfPeople"));

        int days = Integer.parseInt(request.getParameter("days"));

        String preferences = request.getParameter("preferences"); // π.χ. "μουσεία, φαγητό"
```

```

List<Activity> activities = new ArrayList<>();
activities.add(new Activity("Επίσκεψη στο Μουσείο Μπενάκη", "09:00", "11:00"));
activities.add(new Activity("Φαγητό στο Εστιατόριο ", "11:00", "13:00"));
}
}

```

Ένα για την εμφάνιση προγράμματος

```

@WebServlet("/viewPlan")
public class ViewPlanServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        int planId = Integer.parseInt(request.getParameter("planId"));

        List<Activity> activities = getActivities(planId);

        PrintWriter out = response.getWriter();
        out.println("<h1>Following Travel Guide:</h1>");
        out.println("<ul>");
        for (Activity activity : activities) {
            out.println("<li>" + activity.getDescription() +
                " από " + activity.getStartTime() +
                " έως " + activity.getEndTime() + "</li>");
        }
        out.println("</ul>");
    }
    private List<Activity> getActivities(int planId) {
    }
}

```

2. Εισαγωγή Προτιμήσεων

Όταν ο χρήστης συμπληρώσει τη φόρμα με τις προτιμήσεις του, οι επιλογές αυτές θα πρέπει να αποθηκευτούν στον πίνακα Preferences. Για παράδειγμα, αν ο χρήστης επιλέξει "Μουσεία" και "Εστιατόρια":

```
String[] selectedCategories = request.getParameterValues("preferences"); // π.χ. {"1", "2"}
```

```
try (Connection conn = DriverManager.getConnection(url, username, password)) {  
    String sql = "INSERT INTO Preferences (UserID, CategoryID) VALUES (?, ?)";  
    PreparedStatement pstmt = conn.prepareStatement(sql);  
  
    for (String categoryId : selectedCategories) {  
        pstmt.setInt(1, userId); // userId από την τρέχουσα συνεδρία  
        pstmt.setInt(2, Integer.parseInt(categoryId));  
        pstmt.executeUpdate();  
    }  
}
```

3. Ανάκτηση Δραστηριοτήτων βάσει Προτιμήσεων

Αφού ο χρήστης έχει αποθηκεύσει τις προτιμήσεις του, χρησιμοποιούμε αυτές τις πληροφορίες για να ανακτήσουμε τις σχετικές δραστηριότητες:

```
public List<Activity> getCustomizedActivities(int userId) {  
    List<Activity> activities = new ArrayList<>();  
  
    String sql = "SELECT A.Name, A.StartTime, A.EndTime " +  
        "FROM Activities A " +  
        "JOIN Preferences P ON A.CategoryID = P.CategoryID " +  
        "WHERE P.UserID = ?";  
  
    try (Connection conn = DriverManager.getConnection(url, username, password);  
        PreparedStatement pstmt = conn.prepareStatement(sql)) {  
        pstmt.setInt(1, userId);  
  
        ResultSet rs = pstmt.executeQuery();
```

```

while (rs.next()) {
    String name = rs.getString("Name");
    Time startTime = rs.getTime("StartTime");
    Time endTime = rs.getTime("EndTime");
    activities.add(new Activity(name, startTime, endTime));
}
} catch (SQLException e) {
    e.printStackTrace();
}

return activities;
}

```

Πίνακες Βάσης Δεδομένων

- ☐ **Destinations:** Πίνακας με τους προορισμούς, περιλαμβάνοντας το όνομα, τη χώρα, περιγραφή και γεωγραφικές συντεταγμένες.
- ☐ **Activities:** Πίνακας με τις δραστηριότητες, που έχει σύνδεση με τους προορισμούς μέσω του DestinationID.
- ☐ **Users:** Πίνακας με τους χρήστες.
- ☐ **Preferences:** Πίνακας με τις προτιμήσεις των χρηστών.
- ☐ **Categories:** Πίνακας με τις κατηγορίες δραστηριοτήτων.

Αποθηκεύει τα στοιχεία των χρηστών.

```

CREATE TABLE Users (
    UserID INT AUTO_INCREMENT PRIMARY KEY,
    Username VARCHAR(255) NOT NULL,
    Password VARCHAR(255) NOT NULL,
    Email VARCHAR(255) NOT NULL
);

INSERT INTO Users (Username, Password) VALUES
('john_doe', 'password123'),
('jane_smith', 'mypassword'),
('alice_jones', 'securepass'),
('bob_brown', 'pass456');

```

Preferences: Αποθηκεύει τις προτιμήσεις του κάθε χρήστη.

```
CREATE TABLE Preferences (  
    PreferenceID INT PRIMARY KEY AUTO_INCREMENT,  
    UserID INT,  
    CategoryID INT,  
    FOREIGN KEY (UserID) REFERENCES Users(UserID)  
);  
  
INSERT INTO Preferences (UserID, CategoryID) VALUES  
(1, 1), -- john_doe prefers Museums  
(1, 2), -- john_doe prefers Restaurants  
(1, 3), -- john_doe prefers Nightlife  
(2, 1), -- jane_smith prefers Museums  
(2, 4), -- jane_smith prefers Cultural Shows  
(3, 5), -- alice_jones prefers Parks  
(3, 6), -- alice_jones prefers Shopping  
(4, 2), -- bob_brown prefers Restaurants  
(4, 3); -- bob_brown prefers Nightlife
```

Categories: Κατηγορίες δραστηριοτήτων (μουσεία, εστιατόρια, διασκέδαση)

```
CREATE TABLE Categories (  
    CategoryID INT PRIMARY KEY AUTO_INCREMENT,  
    CategoryName VARCHAR(50)  
);  
  
INSERT INTO Categories (CategoryName) VALUES  
( 'Museums'),  
( 'Restaurants'),  
( 'Nightlife'),  
( 'Cultural Shows'),  
( 'Shopping'),  
( 'Parks'),  
( 'Historical Sites'),  
( 'Beaches');
```

(αποθήκευση προορισμών)

```
CREATE TABLE Destinations
```

```
    DestinationID INT AUTO_INCREMENT PRIMARY KEY,
```

```
    Name VARCHAR(255) NOT NULL
```

```
);
```

```
INSERT INTO Destinations (Name, Country, Description, Latitude, Longitude) VALUES
```

```
('Athens', 'Greece', 'The capital city of Greece, known for its ancient history and landmarks like the Acropolis.', 37.9838, 23.7275),
```

```
('Rome', 'Italy', 'The capital city of Italy, famous for its nearly 3,000 years of globally influential art, architecture, and culture.', 41.9028, 12.4964),
```

```
('Paris', 'France', 'The capital city of France, renowned for its art, fashion, gastronomy, and culture.', 48.8566, 2.3522),
```

```
('Barcelona', 'Spain', 'A vibrant city known for its art and architecture, including the Sagrada Família and Park Güell.', 41.3851, 2.1734),
```

```
('Berlin', 'Germany', 'The capital city of Germany, known for its art scene, modern landmarks, and historical sites.', 52.5200, 13.4050),
```

```
('Amsterdam', 'Netherlands', 'The capital city of the Netherlands, famous for its canals, museums, and cycling culture.', 52.3676, 4.9041),
```

```
('Lisbon', 'Portugal', 'The capital city of Portugal, known for its hilly landscape, picturesque neighborhoods, and rich history.', 38.7223, -9.1393),
```

```
('Prague', 'Czech Republic', 'The capital city of the Czech Republic, famous for its old town square, historic architecture, and beautiful bridges.', 50.0755, 14.4378),
```

```
('Vienna', 'Austria', 'The capital city of Austria, renowned for its artistic and intellectual legacy, and home to famous composers.', 48.2082, 16.3738),
```

```
('Budapest', 'Hungary', 'The capital city of Hungary, known for its dramatic skyline, thermal baths, and historic sites.', 47.4979, 19.0402);
```

Activities: Δραστηριότητες με κατηγορίες.

```
CREATE TABLE Activities (
```

```
    ActivityID INT PRIMARY KEY AUTO_INCREMENT,
```

```
    Name VARCHAR(100),
```

```
    CategoryID INT,
```

```
    Location VARCHAR(100),
```

```
    Latitude DECIMAL(9, 6),
```

```

Longitude DECIMAL(9, 6),
Description TEXT,
StartTime TIME,
EndTime TIME,
Country VARCHAR(50),
FOREIGN KEY (CategoryID) REFERENCES Categories(CategoryID)
);

INSERT INTO Activities (Name, CategoryID, Location, Latitude, Longitude, Description, StartTime, EndTime, Country) VALUES

-- Athens, Greece

('Visit the Acropolis Museum', 1, 'Acropolis, Athens', 37.9683, 23.7288, 'An impressive collection of artifacts from the Acropolis.', '09:00:00', '11:00:00', 'Greece'),

('Lunch at Kalamaki', 2, 'Kalamaki, Athens', 37.9643, 23.7341, 'Enjoy traditional Greek souvlaki.', '11:30:00', '13:00:00', 'Greece'),

('Explore the Acropolis', 7, 'Acropolis, Athens', 37.9689, 23.7287, 'Visit the iconic ancient citadel.', '14:00:00', '16:00:00', 'Greece'),

('Stroll through the Ancient Agora', 7, 'Agora, Athens', 37.9735, 23.7259, 'Wander through the ruins of the ancient marketplace.', '16:30:00', '18:30:00', 'Greece'),

('Night out at Bouzoukia', 3, 'Bouzoukia, Athens', 37.9717, 23.7359, 'Experience Greek nightlife.', '20:00:00', '23:00:00', 'Greece'),

('Visit the National Archaeological Museum', 1, 'Museum, Athens', 37.9832, 23.7281, 'Home to a vast collection of Greek antiquities.', '09:00:00', '11:00:00', 'Greece'),

('Lunch at O Thanasis', 2, 'O Thanasis, Athens', 37.9741, 23.7260, 'Famous for its grilled meats and souvlaki.', '11:30:00', '13:00:00', 'Greece'),

('Explore Plaka neighborhood', 7, 'Plaka, Athens', 37.9644, 23.7333, 'Stroll through the picturesque old town.', '14:00:00', '16:00:00', 'Greece'),

('Visit Lycabettus Hill for sunset views', 7, 'Lycabettus, Athens', 37.9736, 23.7492, 'Take a cable car up for amazing views.', '16:30:00', '18:30:00', 'Greece'),

('Dinner at a rooftop restaurant', 2, 'Rooftop, Athens', 37.9840, 23.7275, 'Dine with a view of the Acropolis.', '20:00:00', '22:00:00', 'Greece'),

('Visit the Temple of Olympian Zeus', 7, 'Olympian Zeus, Athens', 37.9616, 23.7275, 'An ancient temple dedicated to Zeus.', '09:00:00', '11:00:00', 'Greece'),

('Lunch at Strofi', 2, 'Strofi, Athens', 37.9592, 23.7333, 'Enjoy Greek cuisine with a view.', '11:30:00', '13:00:00', 'Greece'),

```


('Walk in the National Garden', 5, 'National Garden, Athens', 37.9732, 23.7341, 'Relax in the beautiful gardens.', '14:00:00', '16:00:00', 'Greece'),

('Shopping in Ermou Street', 6, 'Ermou, Athens', 37.9733, 23.7331, 'Shop in the heart of Athens.', '16:30:00', '18:30:00', 'Greece'),

('Enjoy a cultural show', 4, 'Cultural Center, Athens', 37.9643, 23.7570, 'Experience traditional Greek music and dance.', '20:00:00', '22:00:00', 'Greece');

ΚΙ ΠΩΣ ΘΑ ΕΜΦΑΝΙΣΩ ΜΕΡΕΣ ΚΑΙ ΩΡΕΣ ?

1. Δημιουργία Πίνακα ActivitySchedule

Δημιουργία πίνακα που θα αποθηκεύει τις ώρες για κάθε δραστηριότητα.

```
CREATE TABLE ActivitySchedule (  
    ScheduleID INT PRIMARY KEY AUTO_INCREMENT,  
    ActivityID INT,  
    StartTime TIME,  
    EndTime TIME,  
    FOREIGN KEY (ActivityID) REFERENCES Activities(ActivityID)  
);
```

2. Εισαγωγή Δεδομένων στον Πίνακα ActivitySchedule

```
INSERT INTO ActivitySchedule (ActivityID, StartTime, EndTime) VALUES
```

(1, '09:00:00', '11:00:00'), -- Visit the Acropolis Museum

(2, '11:30:00', '13:00:00'), -- Lunch at Kalamaki

(3, '14:00:00', '16:00:00'), -- Explore the Acropolis

(4, '16:30:00', '18:30:00'), -- Stroll through the Ancient Agora

(5, '20:00:00', '23:00:00'), -- Night out at Bouzoukia

(6, '09:00:00', '11:00:00'), -- Visit the National Archaeological Museum

(7, '11:30:00', '13:00:00'), -- Lunch at O Thanasis

(8, '14:00:00', '16:00:00'), -- Explore Plaka neighborhood

(9, '16:30:00', '18:30:00'), -- Visit Lycabettus Hill for sunset views

(10, '20:00:00', '22:00:00'), -- Dinner at a rooftop restaurant

(11, '09:00:00', '11:00:00'), -- Visit the Temple of Olympian Zeus

(12, '11:30:00', '13:00:00'), -- Lunch at Strofi

(13, '14:00:00', '16:00:00'), -- Walk in the National Garden

(14, '16:30:00', '18:30:00'), -- Shopping in Ermou Street

(15, '20:00:00', '22:00:00'); -- Enjoy a cultural show

3. Εμφάνιση Δραστηριοτήτων με Ώρες στην Οθόνη

SELECT

a.Name AS ActivityName,

s.StartTime,

s.EndTime

FROM

Activities a

JOIN

ActivitySchedule s ON a.ActivityID = s.ActivityID

WHERE

a.DestinationID = 1;

Τα έτοιμα προγράμματα δεν λειτουργούν γιατί δεν μπορώ έτσι να κάνω αντιστοίχιση με τις προτιμήσεις

ΔΙΑΦΟΡΕΤΙΚΕΣ ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ ΓΙΑ ΚΑΘΕ ΜΕΡΑ

πίνακας που θα αποθηκεύει τις ημέρες του προγράμματος και θα συνδέει κάθε δραστηριότητα με μια συγκεκριμένη ημέρα.

1. Δημιουργία Πίνακα Itinerary

πίνακας Itinerary, ο οποίος θα συνδέει τις δραστηριότητες με τις ημέρες τους. Κάθε εγγραφή θα αναφέρεται σε μια συγκεκριμένη ημέρα για μια δραστηριότητα.

CREATE TABLE Itinerary (

ItineraryID INT PRIMARY KEY AUTO_INCREMENT,

ActivityID INT,

Day INT,

FOREIGN KEY (ActivityID) REFERENCES Activities(ActivityID)

);

2. Εισαγωγή Δεδομένων στον Πίνακα Itinerary

Μπορείς να προσθέσεις τις δραστηριότητες στο πρόγραμμα για 3 ημέρες. Κάθε δραστηριότητα θα αντιστοιχεί σε μια ημέρα.

INSERT INTO Itinerary (ActivityID, Day) VALUES

(1, 1), -- Day 1: Visit the Acropolis Museum

(2, 1), -- Day 1: Lunch at Kalamaki

(3, 1), -- Day 1: Explore the Acropolis

(4, 1), -- Day 1: Stroll through the Ancient Agora

(5, 1), -- Day 1: Night out at Bouzoukia

(6, 2), -- Day 2: Visit the National Archaeological Museum

(7, 2), -- Day 2: Lunch at O Thanasis

(8, 2), -- Day 2: Explore Plaka neighborhood

(9, 2), -- Day 2: Visit Lycabettus Hill for sunset views

(10, 2), -- Day 2: Dinner at a rooftop restaurant

(11, 3), -- Day 3: Visit the Temple of Olympian Zeus

(12, 3), -- Day 3: Lunch at Strofi

(13, 3), -- Day 3: Walk in the National Garden

(14, 3), -- Day 3: Shopping in Ermou Street

(15, 3); -- Day 3: Enjoy a cultural show

3. Ανάκτηση και Εμφάνιση Προγράμματος

Μπορείς να χρησιμοποιήσεις JOIN queries για να ανακτήσεις το πρόγραμμα με βάση τις ημέρες. Ακολουθεί ένα παράδειγμα query για να εμφανίσεις το πρόγραμμα για τρεις ημέρες:

SELECT

i.Day,

a.Name AS ActivityName,

s.StartTime,

s.EndTime

FROM

Itinerary i

JOIN

Activities a ON i.ActivityID = a.ActivityID

JOIN

ActivitySchedule s ON a.ActivityID = s.ActivityID

WHERE

a.DestinationID = 1 -- Εδώ βάζεις το ID του προορισμού

ORDER BY

i.Day, s.StartTime;

4. Παράδειγμα Εμφάνισης στην Οθόνη

Για να εμφανίσεις το πρόγραμμα σε HTML με βάση τα αποτελέσματα του query, μπορείς να κάνεις χρήση ενός πίνακα:

```
<table>

  <tr>

    <th>Ημέρα</th>

    <th>Δραστηριότητα</th>

    <th>Ωρα Έναρξης</th>

    <th>Ωρα Λήξης</th>

  </tr>

  <!-- Εδώ θα κάνεις loop στα αποτελέσματα από το SQL query -->

  <tr>

    <td>Day 1</td>

    <td>Visit the Acropolis Museum</td>

    <td>09:00</td>

    <td>11:00</td>

  </tr>

  <tr>

    <td>Day 1</td>

    <td>Lunch at Kalamaki</td>

    <td>11:30</td>

    <td>13:00</td>

  </tr>

  <!-- Συμπλήρωσε με άλλες δραστηριότητες -->

  <tr>

    <td>Day 2</td>

    <td>Visit the National Archaeological Museum</td>

    <td>09:00</td>

    <td>11:00</td>

  </tr>

  <!-- ... -->

</table>
```

ΠΡΟΓΡΑΜΜΑΤΑ

1. Athens, Greece

- **Day 1:**
 - **09:00 - 11:00:** Visit the Acropolis Museum
 - **11:30 - 13:00:** Lunch at "Kalamaki"
 - **14:00 - 16:00:** Explore the Acropolis
 - **16:30 - 18:30:** Stroll through the Ancient Agora
 - **20:00 - 23:00:** Night out at a Bouzoukia
- **Day 2:**
 - **09:00 - 11:00:** Visit the National Archaeological Museum
 - **11:30 - 13:00:** Lunch at "O Thanasis"
 - **14:00 - 16:00:** Explore Plaka neighborhood
 - **16:30 - 18:30:** Visit Lycabettus Hill for sunset views
 - **20:00 - 22:00:** Dinner at a rooftop restaurant
- **Day 3:**
 - **09:00 - 11:00:** Visit the Temple of Olympian Zeus
 - **11:30 - 13:00:** Lunch at "Strofi"
 - **14:00 - 16:00:** Walk in the National Garden
 - **16:30 - 18:30:** Shopping in Ermou Street
 - **20:00 - 22:00:** Enjoy a cultural show

2. Rome, Italy

- **Day 1:**
 - **09:00 - 11:00:** Visit the Vatican Museums
 - **11:30 - 13:00:** Lunch at "Trattoria Da Enzo"
 - **14:00 - 16:00:** Explore St. Peter's Basilica
 - **16:30 - 18:30:** Walk around Vatican City
 - **20:00 - 23:00:** Dinner and show at a local pizzeria
- **Day 2:**
 - **09:00 - 11:00:** Explore the Colosseum
 - **11:30 - 13:00:** Lunch at "La Taverna dei Fori Imperiali"
 - **14:00 - 16:00:** Visit the Roman Forum

- **16:30 - 18:30:** Walk through Palatine Hill
- **20:00 - 23:00:** Attend an opera performance
- **Day 3:**
 - **09:00 - 11:00:** Visit the Pantheon
 - **11:30 - 13:00:** Lunch at "Pizzeria Ai Marmi"
 - **14:00 - 16:00:** Explore Piazza Navona
 - **16:30 - 18:30:** Visit the Trevi Fountain
 - **20:00 - 22:00:** Enjoy gelato in Trastevere

3. Paris, France

- **Day 1:**
 - **09:00 - 11:00:** Visit the Louvre Museum
 - **11:30 - 13:00:** Lunch at "Le Relais de l'Entrecôte"
 - **14:00 - 16:00:** Explore the Tuileries Garden
 - **16:30 - 18:30:** Visit the Orangerie Museum
 - **20:00 - 23:00:** Seine River cruise with dinner
- **Day 2:**
 - **09:00 - 11:00:** Visit the Musée d'Orsay
 - **11:30 - 13:00:** Lunch at "Café de Flore"
 - **14:00 - 16:00:** Explore Montmartre and Sacré-Cœur
 - **16:30 - 18:30:** Visit the Basilica of the Sacred Heart
 - **20:00 - 22:00:** Attend a cabaret show at Moulin Rouge
- **Day 3:**
 - **09:00 - 11:00:** Explore the Marais district
 - **11:30 - 13:00:** Lunch at "Chez Janou"
 - **14:00 - 16:00:** Visit Notre-Dame Cathedral
 - **16:30 - 18:30:** Stroll along the Seine
 - **20:00 - 22:00:** Dinner at "Le Procope"

4. Barcelona, Spain

- **Day 1:**
 - **09:00 - 11:00:** Visit Sagrada Família
 - **11:30 - 13:00:** Lunch at "Cervecería Catalana"

- **14:00 - 16:00:** Explore Park Güell
- **16:30 - 18:30:** Visit Casa Batlló
- **20:00 - 23:00:** Flamenco show with dinner
- **Day 2:**
 - **09:00 - 11:00:** Visit the Gothic Quarter
 - **11:30 - 13:00:** Lunch at "Els Quatre Gats"
 - **14:00 - 16:00:** Stroll down La Rambla
 - **16:30 - 18:30:** Visit the Picasso Museum
 - **20:00 - 22:00:** Tapas tour
- **Day 3:**
 - **09:00 - 11:00:** Visit Montjuïc Castle
 - **11:30 - 13:00:** Lunch at "La Caseta del Migdia"
 - **14:00 - 16:00:** Explore the Olympic Park
 - **16:30 - 18:30:** Visit the Magic Fountain of Montjuïc
 - **20:00 - 22:00:** Dinner at a rooftop restaurant

5. London, England

- **Day 1:**
 - **09:00 - 11:00:** Visit the British Museum
 - **11:30 - 13:00:** Lunch at "Dishoom"
 - **14:00 - 16:00:** Explore Covent Garden
 - **16:30 - 18:30:** Visit the National Gallery
 - **20:00 - 23:00:** West End show
- **Day 2:**
 - **09:00 - 11:00:** Visit the Tower of London
 - **11:30 - 13:00:** Lunch at "The Ivy"
 - **14:00 - 16:00:** Walk across Tower Bridge
 - **16:30 - 18:30:** Visit the Tate Modern
 - **20:00 - 22:00:** Dinner at a pub
- **Day 3:**
 - **09:00 - 11:00:** Visit Buckingham Palace
 - **11:30 - 13:00:** Lunch at "The Goring"
 - **14:00 - 16:00:** Stroll in Hyde Park

- **16:30 - 18:30:** Explore Notting Hill
- **20:00 - 22:00:** Evening at Sky Garden

6. Amsterdam, Netherlands

- **Day 1:**

- **09:00 - 11:00:** Visit the Rijksmuseum
- **11:30 - 13:00:** Lunch at "The Pancake Bakery"
- **14:00 - 16:00:** Explore Vondelpark
- **16:30 - 18:30:** Visit the Van Gogh Museum
- **20:00 - 23:00:** Canal cruise with dinner

- **Day 2:**

- **09:00 - 11:00:** Visit Anne Frank House
- **11:30 - 13:00:** Lunch at "Cafe de Jaren"
- **14:00 - 16:00:** Explore the Jordaan district
- **16:30 - 18:30:** Visit the Flower Market
- **20:00 - 22:00:** Night out in the Red Light District

- **Day 3:**

- **09:00 - 11:00:** Visit the Amsterdam Museum
- **11:30 - 13:00:** Lunch at "De Silveren Spiegel"
- **14:00 - 16:00:** Explore the Hermitage Amsterdam
- **16:30 - 18:30:** Visit the NDSM Wharf
- **20:00 - 22:00:** Dinner in a traditional Dutch restaurant

7. Vienna, Austria

- **Day 1:**

- **09:00 - 11:00:** Visit the Kunsthistorisches Museum
- **11:30 - 13:00:** Lunch at "Figlmüller"
- **14:00 - 16:00:** Explore Schönbrunn Palace
- **16:30 - 18:30:** Walk in the palace gardens
- **20:00 - 22:00:** Attend a classical concert

- **Day 2:**

- **09:00 - 11:00:** Visit Belvedere Palace
- **11:30 - 13:00:** Lunch at "Cafe Central"
- **14:00 - 16:00:** Explore the Vienna State Opera

- **16:30 - 18:30:** Visit the Museum of Fine Arts
- **20:00 - 22:00:** Dinner at a local wine tavern
- **Day 3:**
 - **09:00 - 11:00:** Visit the Albertina Museum
 - **11:30 - 13:00:** Lunch at "Gasthaus Pöschl"
 - **14:00 - 16:00:** Stroll through the Stadtpark
 - **16:30 - 18:30:** Visit the Hofburg Palace
 - **20:00 - 22:00:** Enjoy an evening of wine tasting

8. Prague, Czech Republic

- **Day 1:**
 - **09:00 - 11:00:** Visit Prague Castle
 - **11:30 - 13:00:** Lunch at "Lokál"
 - **14:00 - 16:00:** Explore St. Vitus Cathedral
 - **16:30 - 18:30:** Walk across Charles Bridge
 - **20:00 - 23:00:** Enjoy a traditional Czech dinner
- **Day 2:**
 - **09:00 - 11:00:** Visit the Old Town Square
 - **11:30 - 13:00:** Lunch at "Cafe Louvre"
 - **14:00 - 16:00:** Explore the Astronomical Clock
 - **16:30 - 18:30:** Visit the Jewish Quarter
 - **20:00 - 22:00:** Attend a classical music concert
- **Day 3:**
 - **09:00 - 11:00:** Visit the National Museum
 - **11:30 - 13:00:** Lunch at "U Medvídků"
 - **14:00 - 16:00:** Explore Wenceslas Square
 - **16:30 - 18:30:** Visit Vyšehrad Fortress
 - **20:00 - 22:00:** Evening stroll along the Vltava River

9. Lisbon, Portugal

- **Day 1:**
 - **09:00 - 11:00:** Visit the Belém Tower
 - **11:30 - 13:00:** Lunch at "Pastéis de Belém"
 - **14:00 - 16:00:** Explore the Jerónimos Monastery

- **16:30 - 18:30:** Visit the MAAT Museum
- **20:00 - 23:00:** Fado music dinner show
- **Day 2:**
 - **09:00 - 11:00:** Visit the Alfama district
 - **11:30 - 13:00:** Lunch at "Bairro do Avillez"
 - **14:00 - 16:00:** Explore the Lisbon Cathedral
 - **16:30 - 18:30:** Ride Tram 28
 - **20:00 - 22:00:** Dinner at a rooftop restaurant
- **Day 3:**
 - **09:00 - 11:00:** Visit the National Tile Museum
 - **11:30 - 13:00:** Lunch at "Mercado da Ribeira"
 - **14:00 - 16:00:** Stroll through the Chiado district
 - **16:30 - 18:30:** Visit the Santa Justa Lift
 - **20:00 - 22:00:** Evening at a local bar

10. Copenhagen, Denmark

- **Day 1:**
 - **09:00 - 11:00:** Visit Tivoli Gardens
 - **11:30 - 13:00:** Lunch at "Aamanns"
 - **14:00 - 16:00:** Explore Nyhavn
 - **16:30 - 18:30:** Visit the National Museum
 - **20:00 - 23:00:** Dinner cruise on the canals
- **Day 2:**
 - **09:00 - 11:00:** Visit Rosenborg Castle
 - **11:30 - 13:00:** Lunch at "Café Norden"
 - **14:00 - 16:00:** Explore The Little Mermaid statue
 - **16:30 - 18:30:** Visit the Copenhagen Opera House
 - **20:00 - 22:00:** Attend a live music event
- **Day 3:**
 - **09:00 - 11:00:** Visit Christiania
 - **11:30 - 13:00:** Lunch at "Restaurant Radio"
 - **14:00 - 16:00:** Stroll through Frederiksberg Gardens
 - **16:30 - 18:30:** Visit the Round Tower

- **20:00 - 22:00:** Enjoy a beer tasting at a local brewery