

```
// Şehir Dedektifi - Ana Uygulama
const app = {
  map: null,
  userPoints: 0,
  discoveredStories: new Set(),
  badges: [
    { id: 'first', name: 'İlk Keşif', icon: '🎯', unlocked: false, requirement: 1 },
    { id: 'five', name: '5 Hikaye', icon: '⭐', unlocked: false, requirement: 5 },
    { id: 'ten', name: '10 Hikaye', icon: '💎', unlocked: false, requirement: 10 },
    { id: 'expert', name: 'Bölge Uzmanı', icon: '🎓', unlocked: false, requirement: 20 },
    { id: 'legend', name: 'Şehir Efsanesi', icon: '👑', unlocked: false, requirement: 50 }
  ],

  // İstanbul'daki gizli hikaye noktaları
  stories: [
    {
      id: 1,
      title: "Yerebatan Sarnıcı'nın Medusa Sırrı",
      location: "Sultanahmet, Fatih",
      lat: 41.0084,
      lng: 28.9780,
      difficulty: "medium",
      image: "https://images.unsplash.com/photo-1541432901042-2d8bd64b4a9b?w=600&h=400",
      story: "Bizans İmparatoru Justinianus'un 532 yılında inşa ettirdiği bu sarnıca, Medusa'nın başının saklandığı söylenir.",
      quiz: {
        question: "Yerebatan Sarnıcı kaç yılında inşa edildi?",
        options: ["532", "632", "432", "732"],
        correct: 0
      },
      points: 25
    },
    {
      id: 2,
      title: "Pera Palas'ın 411 Numaralı Oda",
      location: "Tepebaşı, Beyoğlu",
      lat: 41.0317,
      lng: 28.9753,
      difficulty: "hard",
      image: "https://images.unsplash.com/photo-1566073771259-6a8506099945?w=600&h=400",
      story: "Ağatha Christie 1926-1932 yılları arasında her İstanbul ziyaretinde bu otelde kaldı.",
      quiz: {
        question: "Ağatha Christie hangi ünlü romanını bu otelde yazarken yazdı?",
        options: ["Nil'de Cinayet", "Doğu Ekspresinde Cinayet", "On Küçük Hırsız"],
        correct: 1
      },
      points: 35
    }
  ]
}
```

```
    },
    points: 40
  },
  {
    id: 3,
    title: "İstiklal'in Gizli Pasajları",
    location: "İstiklal Caddesi, Beyoğlu",
    lat: 41.0368,
    lng: 28.9784,
    difficulty: "easy",
    image: "https://images.unsplash.com/photo-1524231757912-21f4fe3a7200?",
    story: "İstiklal Caddesi'nde yürürken hiç fark ettiniz mi? Caddenin h",
    quiz: {
      question: "İstiklal Caddesi'ndeki pasajlar ne zaman inşa edildi?",
      options: ["1700'ler", "1800'ler", "1900'ler", "2000'ler"],
      correct: 1
    },
    points: 15
  },
  {
    id: 4,
    title: "Galata Kulesi'nin Hapishane Dönemi",
    location: "Galata, Beyoğlu",
    lat: 41.0256,
    lng: 28.9741,
    difficulty: "medium",
    image: "https://images.unsplash.com/photo-1527838832700-5059252407fa?",
    story: "Bugün romantik bir seyir terası olan Galata Kulesi'nin karanl",
    quiz: {
      question: "Hezarfen Ahmet Çelebi Galata Kulesi'nden nereye uçtu?",
      options: ["Karaköy", "Eminönü", "Üsküdar", "Kadıköy"],
      correct: 2
    },
    points: 30
  },
  {
    id: 5,
    title: "Moda İskelesi'nin Aşk Mektupları",
    location: "Moda, Kadıköy",
    lat: 40.9838,
    lng: 29.0253,
    difficulty: "easy",
    image: "https://images.unsplash.com/photo-1578894381163-e72c17f2d45f?",
    story: "1950'lerde Moda İskelesi, ayrı düşmüş sevgililerin buluşma no",
    quiz: {
      question: "İskelenin gizli posta kutusunda kaç mektup bulundu?",
      options: ["50'den fazla", "100'den fazla", "200'den fazla", "300'
```

```
        correct: 2
    },
    points: 20
},
{
    id: 6,
    title: "Haydarpaşa Garı'nın Hayalet Treni",
    location: "Haydarpaşa, Kadıköy",
    lat: 40.9934,
    lng: 29.0183,
    difficulty: "hard",
    image: "https://images.unsplash.com/photo-1474487548417-781cb71495f3?
    story: "1906 yılında Alman mühendisler tarafından inşa edilen Haydarp
    quiz: {
        question: "Haydarpaşa Garı hangi ülkenin mühendisleri tarafından
        options: ["Fransa", "İngiltere", "Almanya", "İtalya"],
        correct: 2
    },
    points: 45
},
{
    id: 7,
    title: "Çiçek Pasajı'nın Büyük Yangını",
    location: "Beyoğlu",
    lat: 41.0359,
    lng: 28.9785,
    difficulty: "medium",
    image: "https://images.unsplash.com/photo-1414235077428-338989a2e8c0?
    story: "1876'da inşa edilen Cité de Péra, İstanbul'un en görkemli pas
    quiz: {
        question: "Çiçek Pasajı'nın orijinal adı neydi?",
        options: ["Cité de Péra", "Passage de Fleurs", "Beyoğlu Pasajı",
        correct: 0
    },
    points: 25
},
{
    id: 8,
    title: "Balat'taki Gizli Havra",
    location: "Balat, Fatih",
    lat: 41.0266,
    lng: 28.9485,
    difficulty: "hard",
    image: "https://images.unsplash.com/photo-1609137144813-7d9921338f24?
    story: "Balat'ın dar sokaklarında 500 yıllık bir sır gizli. Ahrida Si
    quiz: {
        question: "İspanyol Yahudileri İstanbul'a hangi yılda geldi?",
```

```
        options: ["1392", "1492", "1592", "1692"],
        correct: 1
    },
    points: 50
},
{
    id: 9,
    title: "Topkapı Sarayı'nın Gizli Tünelleri",
    location: "Sultanahmet, Fatih",
    lat: 41.0115,
    lng: 28.9833,
    difficulty: "hard",
    image: "https://images.unsplash.com/photo-1583020899958-9011dbfe7f7d?",
    story: "Topkapı Sarayı'nın altında 3 km uzunluğunda gizli tünel ağı v",
    quiz: {
        question: "Topkapı Sarayı'ndaki gizli tünellerin toplam uzunluğu",
        options: ["1 km", "2 km", "3 km", "5 km"],
        correct: 2
    },
    points: 45
},
{
    id: 10,
    title: "Kız Kulesi'nin Gerçek Hikayesi",
    location: "Salacak, Üsküdar",
    lat: 41.0211,
    lng: 29.0040,
    difficulty: "easy",
    image: "https://images.unsplash.com/photo-1567269475957-0fea5f5f4c5e?",
    story: "Herkes yılan efsanesini bilir. Ama Kız Kulesi'nin gerçek hika",
    quiz: {
        question: "Kız Kulesi ilk kez kim tarafından inşa edildi?",
        options: ["Romalılar", "Atinalılar", "Bizanslılar", "Osmanlılar"],
        correct: 1
    },
    points: 15
}
],
```

```
leaderboard: [
    { name: "Ayşe K.", discoveries: 18, points: 620 },
    { name: "Mehmet A.", discoveries: 15, points: 515 },
    { name: "Zeynep Y.", discoveries: 14, points: 490 },
    { name: "Burak S.", discoveries: 12, points: 425 },
    { name: "Elif D.", discoveries: 10, points: 360 },
    { name: "Can T.", discoveries: 8, points: 285 },
    { name: "Selin M.", discoveries: 7, points: 245 },
```

```

    { name: "Emre B.", discoveries: 5, points: 180 },
  ],

  init() {
    this.initMap();
    this.loadUserData();
    this.renderBadges();
    this.renderLeaderboard();
    this.updateStats();
  },

  initMap() {
    // İstanbul'un merkezi
    this.map = L.map('map').setView([41.0082, 28.9784], 12);

    L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
      attribution: '© OpenStreetMap contributors',
      maxZoom: 19
    }).addTo(this.map);

    // Hikayeleri haritaya ekle
    this.stories.forEach(story => {
      this.addMarker(story);
    });

    // Kullanıcı konumunu göster
    if (navigator.geolocation) {
      navigator.geolocation.getCurrentPosition((position) => {
        const userLat = position.coords.latitude;
        const userLng = position.coords.longitude;

        L.marker([userLat, userLng], {
          icon: L.divIcon({
            className: 'user-location',
            html: '<div style="width:20px;height:20px;background:#667'
            iconSize: [20, 20]
          })
        }).addTo(this.map).bindPopup('Buradasınız!');

        // Haritayı kullanıcı konumuna yakınlaştır
        this.map.setView([userLat, userLng], 14);
      });
    }
  },

  addMarker(story) {
    const difficultyClass = story.difficulty === 'easy' ? 'marker-easy' :

```

```

        story.difficulty === 'medium' ? 'marker-medium' :

const marker = L.marker([story.lat, story.lng], {
  icon: L.divIcon({
    className: 'custom-marker',
    html: ``,
    iconSize: [40, 40],
    iconAnchor: [20, 40]
  })
}).addTo(this.map);

marker.on('click', () => {
  this.showStory(story);
});

marker.bindPopup(`
  <strong>${story.title}</strong><br>
  <small>${story.location}</small><br>
  <small>Zorluk: ${story.difficulty === 'easy' ? '🟢 Kolay' : story.difficulty === 'medium' ? '🟡 Orta' : '🔴 Zor'}</small>
`);
},

showStory(story) {
  const modal = document.getElementById('storyModal');
  const content = document.getElementById('storyContent');

  const isDiscovered = this.discoveredStories.has(story.id);
  const difficultyText = story.difficulty === 'easy' ? '🟢 Kolay' :
    story.difficulty === 'medium' ? '🟡 Orta' : '🔴 Zor';

  content.innerHTML = `
    
    <div class="story-difficulty">${difficultyText} • ${story.points} Puan</div>
    <h1 class="story-title">${story.title}</h1>
    <div class="story-location">📍 ${story.location}</div>
    <div class="story-text">${story.story}</div>

    ${!isDiscovered ? `
      <div class="quiz-section" id="quizSection">
        <div class="quiz-title">🍀 Bilgi Testini Tamamla ve Puanı Kazan</div>
        <div class="quiz-question">${story.quiz.question}</div>
        <div class="quiz-options" id="quizOptions">
          ${story.quiz.options.map((option, index) => `
            <div class="quiz-option" onclick="app.checkAnswer('${story.id}', ${index})">${option}</div>
          `).join('')}
        </div>
      </div>
    ` : ''}
  `;
}

```

```

        `: `
        <div class="reward-section">
            <div class="reward-icon">✅</div>
            <div class="reward-text">Bu hikayeyi zaten keşfettin!</div>
        </div>
    `}
    `;

    modal.classList.add('active');
},

checkAnswer(storyId, selectedIndex, correctIndex) {
    const options = document.querySelectorAll('.quiz-option');

    if (selectedIndex === correctIndex) {
        options[selectedIndex].classList.add('correct');

        setTimeout(() => {
            const story = this.stories.find(s => s.id === storyId);
            this.discoveredStories.add(storyId);
            this.userPoints += story.points;
            this.saveUserData();
            this.updateStats();
            this.checkBadges();

            document.getElementById('quizSection').innerHTML = `
                <div class="reward-section">
                    <div class="reward-icon">🎉</div>
                    <div class="reward-text">Tebrikler! +${story.points} Puan
                </div>
            `;
        }, 1000);
    } else {
        options[selectedIndex].classList.add('wrong');
        options[correctIndex].classList.add('correct');

        setTimeout(() => {
            document.getElementById('quizSection').innerHTML = `
                <div style="text-align:center;padding:20px;color:white;">
                    <div style="font-size:40px;margin-bottom:10px;">😞</div>
                    <div style="font-size:18px;">Yanlış cevap! Tekrar dene.</div>
                </div>
            `;

            setTimeout(() => {
                this.closeStory();
            }, 2000);
        });
    }
}

```

```

    }, 1500);
  }
},

checkBadges() {
  const discoveredCount = this.discoveredStories.size;
  let newBadges = false;

  this.badges.forEach(badge => {
    if (!badge.unlocked && discoveredCount >= badge.requirement) {
      badge.unlocked = true;
      newBadges = true;
      this.showBadgeNotification(badge);
    }
  });

  if (newBadges) {
    this.renderBadges();
  }
},

showBadgeNotification(badge) {
  // Simple notification for new badge
  const notification = document.createElement('div');
  notification.style.cssText = `
    position: fixed;
    top: 100px;
    left: 50%;
    transform: translateX(-50%);
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    color: white;
    padding: 20px 30px;
    border-radius: 16px;
    box-shadow: 0 10px 40px rgba(0,0,0,0.3);
    z-index: 3000;
    animation: slideDown 0.5s;
    text-align: center;
  `;
  notification.innerHTML = `
    <div style="font-size:48px;margin-bottom:10px;">${badge.icon}</div>
    <div style="font-size:20px;font-weight:700;">Yeni Rozet Kazandın!</div>
    <div style="font-size:16px;margin-top:5px;">${badge.name}</div>
  `;
  document.body.appendChild(notification);

  setTimeout(() => {
    notification.remove();
  }, 3000);
}

```



```

    }, 3000);
  },

  closeStory() {
    document.getElementById('storyModal').classList.remove('active');
  },

  updateStats() {
    document.getElementById('userPoints').textContent = this.userPoints;
    document.getElementById('userBadges').textContent = this.badges.filter(b

    // Update profile stats
    document.getElementById('profileStories').textContent = this.discoveredSt
    document.getElementById('profilePoints').textContent = this.userPoints;
    document.getElementById('profileBadgesCount').textContent = this.badges.f
  },

  renderBadges() {
    const grid = document.getElementById('badgesGrid');
    grid.innerHTML = this.badges.map(badge => `
      <div class="badge ${badge.unlocked ? 'unlocked' : ''}">
        <div class="badge-icon">${badge.icon}</div>
        <div class="badge-name">${badge.name}</div>
      </div>
    `).join('');
  },

  renderLeaderboard() {
    const list = document.getElementById('leaderboardList');
    list.innerHTML = this.leaderboard.map((user, index) => `
      <div class="leaderboard-item">
        <div class="leaderboard-rank">${index === 0 ? '🏆' : index === 1
        <div class="leaderboard-info">
          <div class="leaderboard-name">${user.name}</div>
          <div class="leaderboard-discoveries">${user.discoveries} hika
        </div>
        <div class="leaderboard-points">${user.points}</div>
      </div>
    `).join('');
  },

  saveUserData() {
    localStorage.setItem('userPoints', this.userPoints);
    localStorage.setItem('discoveredStories', JSON.stringify([...this.discover
    localStorage.setItem('badges', JSON.stringify(this.badges));
  },

```

```

loadUserData() {
  const savedPoints = localStorage.getItem('userPoints');
  const savedStories = localStorage.getItem('discoveredStories');
  const savedBadges = localStorage.getItem('badges');

  if (savedPoints) this.userPoints = parseInt(savedPoints);
  if (savedStories) this.discoveredStories = new Set(JSON.parse(savedStories));
  if (savedBadges) this.badges = JSON.parse(savedBadges);
}
};

// Navigation functions
function showMap() {
  document.getElementById('profileSection').classList.remove('active');
  document.getElementById('leaderboardSection').classList.remove('active');

  document.querySelectorAll('.nav-item').forEach(item => item.classList.remove('active'));
  document.querySelectorAll('.nav-item')[0].classList.add('active');
}

function showProfile() {
  document.getElementById('profileSection').classList.add('active');
  document.getElementById('leaderboardSection').classList.remove('active');

  document.querySelectorAll('.nav-item').forEach(item => item.classList.remove('active'));
  document.querySelectorAll('.nav-item')[1].classList.add('active');
}

function showLeaderboard() {
  document.getElementById('leaderboardSection').classList.add('active');
  document.getElementById('profileSection').classList.remove('active');

  document.querySelectorAll('.nav-item').forEach(item => item.classList.remove('active'));
  document.querySelectorAll('.nav-item')[2].classList.add('active');
}

function closeStory() {
  app.closeStory();
}

// Initialize app when page loads
window.addEventListener('DOMContentLoaded', () => {
  app.init();
});

```