

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>SmartPark Innovations - Parking Spot Booking</title>
<script src="https://cdn.tailwindcss.com"></script>
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Pacifico&display=swap" rel="stylesheet">
<link href="https://cdn.jsdelivr.net/npm/remixicon@4.5.0/fonts/remixicon.css" rel="stylesheet">
<script>
tailwind.config = {
theme: {
extend: {
colors: {
primary: '#0EA5E9',
secondary: '#64748B'
},
borderRadius: {
'none': '0px',
'sm': '4px',
DEFAULT: '8px',
'md': '12px',
'lg': '16px',
'xl': '20px',
'2xl': '24px',
'3xl': '32px',
'full': '9999px',
'button': '8px'
}
}
}
}
}
}
}
</script>
<style>
:where([class^="ri-"]):before { content: "\f3c2"; }
input[type="number"]::-webkit-inner-spin-button,
input[type="number"]::-webkit-outer-spin-button {
-webkit-appearance: none;
margin: 0;
}
.parking-spot {
```

```
transition: all 0.3s ease;
}

.parking-spot:hover {
transform: scale(1.05);
}
</style>
</head>
<body class="bg-gray-50 min-h-screen">
<header class="bg-white shadow-sm">
<div class="max-w-7xl mx-auto px-4 py-4 flex items-center justify-between">
  <div class="w-10 h-10 flex items-center justify-center">
    <a href="index.html"><i class="ri-arrow-left-line text-2xl text-primary"></i></a>
  </div>
<div class="flex items-center space-x-2">
<span class="text-2xl font-['Pacifico'] text-primary">SmartPark Innovations</span>
</div>
</div>
</header>
<main class="max-w-7xl mx-auto px-4 py-8">
<div class="grid grid-cols-1 lg:grid-cols-2 gap-8">
<div class="bg-white rounded shadow-sm p-6">
<h2 class="text-xl font-semibold mb-6">Select Parking Spot</h2>
<div class="grid grid-cols-5 gap-4 mb-6">
<div id="parkingSpots" class="col-span-5 grid grid-cols-5 gap-4">
</div>
</div>
<div class="flex items-center space-x-4 mt-4">
<div class="flex items-center">
<div class="w-4 h-4 bg-green-500 rounded mr-2"></div>
<span class="text-sm">Available</span>
</div>
<div class="flex items-center">
<div class="w-4 h-4 bg-red-500 rounded mr-2"></div>
<span class="text-sm">Occupied</span>
</div>
<div class="flex items-center">
<div class="w-4 h-4 bg-primary rounded mr-2"></div>
<span class="text-sm">Selected</span>
</div>
</div>
</div>
</div>
<div class="bg-white rounded shadow-sm p-6">
<h2 class="text-xl font-semibold mb-6">Booking Details</h2>
<form id="bookingForm" class="space-y-6">
```

License Plate Number

Enter license plate number

Vehicle Type

Select vehicle type

Sedan

SUV

Van

Motorcycle

Vehicle Model

Enter vehicle model

Entry Time

Exit Time

Parking Fee:

\$0.00

```
<div>
<label class="block text-sm font-medium text-gray-700 mb-2">Payment Method</label>
<div class="grid grid-cols-4 gap-4">
<button type="button" class="payment-method active flex flex-col items-center justify-center p-4 border rounded-
button hover:border-primary cursor-pointer" data-method="credit">
<i class="ri-bank-card-line text-2xl mb-2"></i>
<span class="text-sm">Credit Card</span>
</button>
<button type="button" class="payment-method flex flex-col items-center justify-center p-4 border rounded-button
hover:border-primary cursor-pointer" data-method="debit">
<i class="ri-bank-card-2-line text-2xl mb-2"></i>
<span class="text-sm">Debit Card</span>
</button>
<button type="button" class="payment-method flex flex-col items-center justify-center p-4 border rounded-button
hover:border-primary cursor-pointer" data-method="alipay">
<i class="ri-alipay-fill text-2xl mb-2"></i>
<span class="text-sm">Alipay</span>
</button>
<button type="button" class="payment-method flex flex-col items-center justify-center p-4 border rounded-button
hover:border-primary cursor-pointer" data-method="wechat">
<i class="ri-wechat-pay-fill text-2xl mb-2"></i>
<span class="text-sm">WeChat Pay</span>
</button>
</div>
</div>
<div class="flex items-start">
<input type="checkbox" id="terms" required class="mt-1">
<label for="terms" class="ml-2 text-sm text-gray-600">I agree to the <a href="#" class="text-primary
hover:underline">Terms and Conditions</a></label>
</div>
<button type="submit" class="w-full bg-primary text-white py-3 rounded-button hover:bg-primary/90 transition-
colors !rounded-button whitespace-nowrap cursor-pointer">
Confirm Booking
</button>
</form>
</div>
</div>
</main>
<div id="receiptModal" class="fixed inset-0 bg-black bg-opacity-50 hidden items-center justify-center">
<div class="bg-white rounded-lg p-8 max-w-2xl w-full mx-4">
<div class="flex justify-between items-center mb-6">
<h3 class="text-2xl font-semibold">Booking Receipt</h3>
<button id="closeReceipt" class="text-gray-500 hover:text-gray-700">
<i class="ri-close-line text-2xl"></i>
</div>
</div>
</div>
```

```
</button>
</div>
<div id="receiptContent" class="space-y-4">
</div>
<div class="flex justify-end space-x-4 mt-8">
<button class="px-6 py-2 border border-gray-300 rounded-button hover:bg-gray-50 !rounded-button whitespace-
nowrap cursor-pointer" onclick="window.print()">
<i class="ri-printer-line mr-2"></i>Print
</button>
<button class="px-6 py-2 bg-primary text-white rounded-button hover:bg-primary/90 !rounded-button whitespace-
nowrap cursor-pointer" onclick="downloadReceipt()">
<i class="ri-download-line mr-2"></i>Download
</button>
</div>
</div>
</div>
<script>
const mockParkingSpots = Array(25).fill(null).map((_, index) => ({
id: index + 1,
available: Math.random() > 0.3
}));
let selectedSpot = null;
let selectedPaymentMethod = 'credit';
function renderParkingSpots() {
const container = document.getElementById('parkingSpots');
container.innerHTML = mockParkingSpots.map(spot => `
<div class="parking-spot cursor-pointer aspect-square rounded flex items-center justify-center text-white font-
medium ${
spot.available ? 'bg-green-500 hover:bg-green-600' : 'bg-red-500'
} ${selectedSpot === spot.id ? '!bg-primary' : ""}
${spot.available ? `onclick="selectSpot(${spot.id})"` : ""}
data-spot-id="${spot.id}">
${spot.id}
</div>
`).join("");
}
function selectSpot(id) {
selectedSpot = selectedSpot === id ? null : id;
renderParkingSpots();
}
document.querySelectorAll('.payment-method').forEach(btn => {
btn.addEventListener('click', () => {
document.querySelectorAll('.payment-method').forEach(b => b.classList.remove('border-primary'));
btn.classList.add('border-primary');
```

```
selectedPaymentMethod = btn.dataset.method;
});
});
function calculateParkingFee() {
const entryTime = new Date(document.getElementById('entryTime').value);
const exitTime = new Date(document.getElementById('exitTime').value);
const hours = Math.ceil((exitTime - entryTime) / (1000 * 60 * 60));
const rate = 2.00;
return (hours * rate).toFixed(2);
}
document.getElementById('entryTime').addEventListener('change', updateFee);
document.getElementById('exitTime').addEventListener('change', updateFee);
function updateFee() {
const fee = calculateParkingFee();
document.getElementById('parkingFee').textContent = `$$${fee}`;
}
document.getElementById('bookingForm').addEventListener('submit', (e) => {
e.preventDefault();
if (!selectedSpot) {
alert('Please select a parking spot');
return;
}
const bookingData = {
spotId: selectedSpot,
licensePlate: document.getElementById('licensePlate').value,
vehicleType: document.getElementById('vehicleType').value,
vehicleModel: document.getElementById('vehicleModel').value,
entryTime: document.getElementById('entryTime').value,
exitTime: document.getElementById('exitTime').value,
fee: calculateParkingFee(),
paymentMethod: selectedPaymentMethod,
bookingRef: 'BP' + Math.random().toString(36).substr(2, 9).toUpperCase()
};
showReceipt(bookingData);
});
function showReceipt(data) {
const receiptContent = document.getElementById('receiptContent');
receiptContent.innerHTML = `
<div class="border-b pb-4">
<div class="text-center mb-4">
<div class="text-2xl font-[Pacifico] text-primary mb-2">SmartPark Innovations</div>
<div class="text-sm text-gray-500"></div>
</div>
<div class="flex justify-between text-sm">
```

```
<span>Booking Reference:</span>
<span class="font-semibold">${data.bookingRef}</span>
</div>
</div>
<div class="space-y-3 py-4">
<div class="flex justify-between">
<span>Parking Spot:</span>
<span class="font-semibold">#${data.spotId}</span>
</div>
<div class="flex justify-between">
<span>License Plate:</span>
<span class="font-semibold">${data.licensePlate}</span>
</div>
<div class="flex justify-between">
<span>Vehicle:</span>
<span class="font-semibold">${data.vehicleType} - ${data.vehicleModel}</span>
</div>
<div class="flex justify-between">
<span>Entry Time:</span>
<span class="font-semibold">${(new Date(data.entryTime).toLocaleString())}</span>
</div>
<div class="flex justify-between">
<span>Exit Time:</span>
<span class="font-semibold">${(new Date(data.exitTime).toLocaleString())}</span>
</div>
<div class="flex justify-between">
<span>Payment Method:</span>
<span class="font-semibold">${data.paymentMethod.charAt(0).toUpperCase() + data.paymentMethod.slice(1)}
</span>
</div>
<div class="flex justify-between text-lg font-semibold">
<span>Total Amount:</span>
<span>$$${data.fee}</span>
</div>
</div>
<div class="border-t pt-4 text-center">
<div class="text-sm text-gray-500 mb-2">Scan QR Code for Parking Access</div>
<div class="w-32 h-32 mx-auto bg-gray-200 flex items-center justify-center">
<i class="ri-qr-code-line text-4xl text-gray-400"></i>
</div>
</div>
`;
document.getElementById('receiptModal').style.display = 'flex';
}
```

```
document.getElementById('closeReceipt').addEventListener('click', () => {
document.getElementById('receiptModal').style.display = 'none';
});
function downloadReceipt() {
const element = document.getElementById('receiptModal');
const opt = {
margin: 1,
filename: 'parking-receipts.pdf',
image: { type: 'jpeg', quality: 0.98 },
html2canvas: { scale: 2 },
jsPDF: { unit: 'in', format: 'letter', orientation: 'portrait' }
};
html2pdf().set(opt).from(element).save();
}
renderParkingSpots();
const now = new Date();
document.getElementById('entryTime').value = now.toISOString().slice(0, 16);
now.setHours(now.getHours() + 2);
document.getElementById('exitTime').value = now.toISOString().slice(0, 16);
updateFee();
</script>
</body>
</html>
```