

Urbee Platform - Client Proposal

Urbee Platform



Urbee Logo

Software Development Proposal & Agreement

Prepared For: [Client Name]

Prepared By: [Your Company Name]

Date: February 10, 2026

Proposal Valid Until: March 10, 2026

Document Version: 2.0 - Production Ready

Brand Colors: #fdb82e (Primary) | • #2c3e50 (Text) | #27ae60 (Success)

Status: Fully deployed and performance-tested in production

Table of Contents

1. [Executive Summary](#)
2. [Project Overview](#)
3. [Scope of Work](#)
4. [Deliverables](#)

5. [Technology Stack](#)
 6. [Development Cost Analysis](#)
 7. [Investment & Payment Terms](#)
 8. [Project Timeline](#)
 9. [Ongoing Costs & Maintenance](#)
 10. [Terms & Conditions](#)
-

1. Executive Summary

We are pleased to present this proposal for the development of **ParkingSpots** — a comprehensive peer-to-peer parking rental marketplace platform. This solution will enable property owners to monetize their unused parking spaces while providing users with a seamless mobile experience to find, book, and pay for parking.

Value Proposition

	Benefit	Impact
Production-Ready		Fully deployed, tested at 1,666+ RPS with 95% cache hit rate
High Performance		<1ms average response time, supporting 1,500-2,500 concurrent users
Enterprise Infrastructure		PostgreSQL 14, Redis 6.x, 12-worker architecture
New Revenue Stream		Earn from every transaction on the platform
Market Opportunity		\$5B+ parking marketplace industry
Full Ownership		You own 100% of the production-tested codebase

Investment Options

Choose the payment structure that fits your business:

Option	Upfront Fee	Revenue Share	Recommended
Option 1	€20,000	20%	Balanced
Option 2	€15,000	25%	Lower initial cost
Option 3	€25,000	15%	Lower ongoing share

2. Project Overview

2.1 The Problem

- Urban parking is scarce and expensive
- Private parking spaces sit unused for hours daily
- No easy way to connect parking owners with drivers
- Existing solutions are limited to commercial parking only

2.2 The Solution

ParkingSpots is a two-sided marketplace that:

1. **For Parking Owners:** Provides tools to list, manage, and earn from their parking spaces

2. **For Drivers:** Offers a mobile app to discover, book, and pay for convenient parking

2.3 Target Users

User Type	Description	Needs
Property Owners	Homeowners, businesses, churches, schools	Monetize unused parking
Daily Commuters	Office workers, students	Reliable daily parking
Event Attendees	Sports fans, concertgoers	Parking near venues
Urban Drivers	City residents, visitors	Affordable short-term parking

3. Scope of Work

3.1 Included Features

User Management

- User registration & authentication
- Profile management
- Role-based access (Renter, Owner, Admin)
- Password reset & security

Parking Spot Management

- Create, edit, delete listings
- Photo upload support
- Pricing configuration (hourly/daily/monthly)
- Availability scheduling
- Amenity metadata (EV charging, covered, etc.)

Search & Discovery

- Location-based search with map view
- Advanced filters (price, type, amenities)
- Distance calculations
- Real-time availability display

Booking System

- Instant booking flow
- Price calculation engine
- Booking management (upcoming, active, past)
- Check-in/check-out tracking
- Cancellation handling

Payment Processing

- Secure payment via Stripe
- Automatic owner payouts
- Refund processing
- Transaction history

Reviews & Ratings

- 5-star rating system
- Written reviews
- Owner response capability
- Rating aggregation

3.2 Exclusions

The following are not included in this proposal:

- Custom admin dashboard (can be added for additional fee)
- SMS/Push notification service integration
- Customer support chat system
- Marketing website/landing pages
- App Store submission fees
- Ongoing server hosting costs
- Third-party service subscription fees

4. Deliverables

4.1 Complete Deliverables List

#	Deliverable	Description
1	Backend API	Complete FastAPI server with all endpoints
2	Database Schema	PostgreSQL database with all tables
3	Mobile App (iOS)	React Native app ready for App Store
4	Mobile App (Android)	React Native app ready for Play Store
5	API Documentation	Full Swagger/OpenAPI documentation
6	Technical Documentation	Architecture and deployment guides
7	Source Code	Complete, commented source code
8	Deployment Guide	Step-by-step deployment instructions

4.2 Source Code Components

Delivered Package:

```
└── backend/
    ├── app/
    │   ├── api/                      # REST API endpoints
    │   ├── core/                     # Configuration & security
    │   ├── db/                       # Database configuration
    │   ├── models/                  # Data models (6 models)
    │   └── schemas/                # API schemas
    └── requirements.txt            # Python dependencies
    └── Dockerfile                  # Container configuration

└── mobile/
    ├── src/
    │   ├── screens/                 # 6+ app screens
    │   ├── components/              # Reusable UI components
    │   ├── services/                # API integration layer
    │   ├── stores/                  # State management
    │   └── navigation/              # App navigation
    └── App.tsx                     # Application entry
    └── package.json                # Node dependencies

└── documentation/
    └── README.md                  # All Documentation
```

```
└─ TECHNICAL_DOCUMENTATION.md  
    └─ API_REFERENCE.md
```

5. Technology Stack

5.1 Technology Choices

Layer	Technology	Industry Standard	Rationale
Backend	Python + FastAPI	Yes	High performance, auto-documentation
Database	PostgreSQL	Yes	Reliable, scalable, feature-rich
Mobile	React Native	Yes	Cross-platform, native performance
Payments	Stripe	Yes	Industry leader, marketplace support
Maps	Google Maps	Yes	Best coverage, reliable
Hosting	AWS/GCP	Yes	Enterprise-grade infrastructure

5.2 Architecture Benefits

- Scalable:** Handles 1,500-2,500 concurrent users (tested in production)
- High Performance:** 1,666+ RPS with <1ms average response time
- Efficient:** 95% Redis cache hit rate reduces database load
- Reliable:** Multi-worker architecture with automatic failover
- Secure:** Industry-standard encryption and authentication
- Maintainable:** Clean code architecture for easy updates
- Cross-Platform:** Single codebase for iOS and Android

5.3 Production Performance Metrics

Infrastructure: - **Backend:** 12 worker processes (FastAPI + Uvicorn) - **Database:** PostgreSQL 14 (300 max connections, 3GB shared buffers) - **Cache:** Redis 6.x with hiredis parser - **Background:** Separate worker for automated booking management

Benchmark Results:

Metric	Single Worker Production (12 Workers)		Improvement
Requests/Second	515 RPS	1,666+ RPS	3.2x
Avg Response Time	1-2ms	<1ms	2x faster
Cache Hit Rate	94%	95%	Optimized
Concurrent Users	~500	1,500-2,500	4x capacity

Key Features: - Auto-checkout expired bookings (every 60 seconds) - Auto-start bookings at scheduled time - Real-time monitoring dashboard (`./monitor.sh`) - Comprehensive load testing suite (`./load_test.sh`) - Systemd service integration for production - Row-level locking prevents double-booking - Connection pooling optimized for scale

6. Development Cost Analysis

6.1 Market Rate Comparison

The following represents typical market pricing for developing a complete parking marketplace platform like ParkingSpots:

Component Breakdown

Component	Market Price
Backend API Development	\$40,000
• FastAPI server with async architecture	
• PostgreSQL database design & implementation	
• JWT authentication & security	
• User management system	
• Parking spots CRUD & search (location-based)	
• Instant booking system	
• Stripe payment integration	
• Automated payout system	
• Reviews & ratings system	
• Real-time availability (WebSocket/Redis)	
• API documentation	
Mobile App Development	\$48,000
• React Native for iOS & Android	
• Full navigation architecture	
• Authentication & user management	
• Interactive map with search	
• Advanced filtering system	
• Instant booking flow	
• Stripe payment UI integration	
• User profiles & settings	
• Reviews & ratings interface	
• State management (Zustand)	
• Push notifications setup	
UI/UX Design	\$8,000
• User interface design for all screens	
• User experience optimization	
• Brand identity elements	
Technical Documentation	\$3,000
• Complete technical documentation	
• API reference documentation	
• Deployment guides	
DevOps & Deployment	\$5,000
• Docker containerization	
• Database setup & optimization	
• CI/CD pipeline configuration	
• Environment setup	
Quality Assurance	\$8,000
• Comprehensive testing (unit, integration)	
• Bug fixes & optimization	
• Cross-platform testing	
• Performance optimization	

6.2 Total Market Value

Category	Cost
Backend API Development	\$40,000

Mobile App Development	\$48,000
UI/UX Design	\$8,000
Technical Documentation	\$3,000
DevOps & Deployment	\$5,000
Quality Assurance	\$8,000
Total Market Value	\$118,000

6.3 Your Investment Options

Choose the payment structure that best fits your business model:

Option 1: Balanced Partnership

- **Upfront Fee:** €20,000
- **Revenue Share:** 20% ongoing
- **Your Savings:** €98,000 (83% vs market)
- **Best For:** Balanced risk/reward with moderate upfront investment

Option 2: Lower Initial Investment

- **Upfront Fee:** €15,000
- **Revenue Share:** 25% ongoing
- **Your Savings:** €103,000 (87% vs market)
- **Best For:** Minimizing initial capital while sharing more growth upside

Option 3: Higher Upfront, Lower Share

- **Upfront Fee:** €25,000
- **Revenue Share:** 15% ongoing
- **Your Savings:** €93,000 (79% vs market)
- **Best For:** Maximizing long-term profitability with higher initial investment

Note: All options include the complete platform with identical features and quality. The significantly reduced development fee is offset by the revenue sharing agreement, creating a win-win partnership where both parties benefit from the platform's success.

7. Investment & Payment Terms

7.1 Fee Structure Options

Select one of the following investment structures:

Option 1: €20,000 + 20% Revenue Share

A. Development Fee: €20,000

Milestone	Percentage Amount	Due
Project Kickoff	30%	€6,000 Upon signing (February 2026)
Backend Completion	30%	€6,000 End of March 2026
Mobile App Completion	30%	€6,000 End of May 2026
Final Delivery	10%	€2,000 June 2026
Total	100%	€20,000

B. Revenue Share: 20% of Net Platform Revenue

Option 2: €15,000 + 25% Revenue Share

A. Development Fee: €15,000

Milestone	Percentage	Amount	Due
Project Kickoff	30%	€4,500	Upon signing (February 2026)
Backend Completion	30%	€4,500	End of March 2026
Mobile App Completion	30%	€4,500	End of May 2026
Final Delivery	10%	€1,500	June 2026
Total	100%	€15,000	

B. Revenue Share: 25% of Net Platform Revenue

Option 3: €25,000 + 15% Revenue Share

A. Development Fee: €25,000

Milestone	Percentage	Amount	Due
Project Kickoff	30%	€7,500	Upon signing (February 2026)
Backend Completion	30%	€7,500	End of March 2026
Mobile App Completion	30%	€7,500	End of May 2026
Final Delivery	10%	€2,500	June 2026
Total	100%	€25,000	

B. Revenue Share: 15% of Net Platform Revenue

Definition of Net Platform Revenue (applies to all options): - Total booking revenue collected through the platform - MINUS payment processor fees (typically 2.9% + \$0.30) - MINUS refunds issued - MINUS chargebacks

Revenue Share Comparison Examples:

Monthly Gross Revenue	Net Revenue	Option 1 (20%)	Option 2 (25%)	Option 3 (15%)
€10,000	€9,700	€1,940	€2,425	€1,455
€50,000	€48,500	€9,700	€12,125	€7,275
€100,000	€97,000	€19,400	€24,250	€14,550
€500,000	€485,000	€97,000	€121,250	€72,750

7.2 Revenue Share Terms

- Reporting Period:** Monthly, ending on the last day of each calendar month
- Payment Due:** Within 15 days of month end
- Reporting:** Client provides monthly revenue report with transaction details
- Audit Rights:** Developer may audit records once per year with 30 days notice
- Minimum Term:** Revenue share applies for 5 years from launch date
- Buyout Option:** Client may terminate revenue share with a one-time payment equal to 24 months of average revenue share (based on trailing 12 months)

8. Project Timeline

Project Start: February 2026

Final Delivery: June 2026

Total Duration: 16 weeks

8.1 Development Schedule

Weeks 1-4 (February 2026): Backend Foundation & Development

- └── Environment configuration
- └── Database schema implementation
- └── Core authentication system
- └── User management API
- └── Parking spots CRUD API
- └── Search & filtering system
- └── Booking system implementation
- └── Payment integration (Stripe)

Weeks 5-8 (March 2026): Backend Completion & Mobile Setup

- └── Automated payout system
- └── Reviews & ratings system
- └── API documentation & testing
- └── Mobile project setup
- └── Navigation architecture
- └── Authentication screens

Weeks 9-12 (April-May 2026): Mobile App Development

- └── Home screen with interactive map
- └── Search & listing screens
- └── Booking flow implementation
- └── Payment integration
- └── User profile & settings
- └── Reviews system UI

Weeks 13-16 (May-June 2026): Testing, Refinement & Delivery

- └── Integration testing (backend + mobile)
- └── Cross-platform testing (iOS & Android)
- └── Bug fixes & optimization
- └── Performance tuning
- └── Documentation finalization
- └── Deployment setup
- └── Final delivery & handoff

8.2 Milestone Schedule

Milestone	Timeline	Deliverable
M1: Kickoff	February 2026	Project start, requirements confirmed
M2: Backend Complete	End of March 2026	All backend APIs functional & tested
M3: Mobile Alpha	End of April 2026	Core mobile screens complete
M4: Mobile Complete	End of May 2026	All mobile features implemented
M5: Final Delivery	June 2026	Fully tested, documented, delivered

9. Ongoing Costs & Maintenance

9.1 Third-Party Service Costs (Client Responsibility)

Service	Provider	Estimated Monthly Cost
Cloud Hosting	AWS/GCP/DigitalOcean	\$50 - \$500*
Database	Managed PostgreSQL	€15 - €100*

Service	Description	Cost Range
Redis Cache	Managed Redis	\$15 - \$50*
Payment Processing	Stripe	2.9% + \$0.30 per transaction
Maps API	Google Maps	\$200/month (after free tier)
Email Service	SendGrid/AWS SES	\$15 - \$50*
Push Notifications	Firebase	Free tier usually sufficient
SSL Certificate	Let's Encrypt	Free
Domain Name	Any registrar	\$12 - \$50/year

*Costs scale with usage

9.2 Estimated Monthly Operating Costs

Stage	Users	Bookings/Month	Est. Hosting	Est. APIs	Total
Launch	< 1,000	< 500	\$100	\$50	~\$150/mo
Growth	1,000 - 10,000	500 - 5,000	\$300	\$200	~\$500/mo
Scale	10,000 - 100,000	5,000 - 50,000	\$1,000	\$500	~\$1,500/mo
Enterprise	100,000+	50,000+	\$3,000+	\$1,000+	~\$4,000+/mo

9.3 Optional Maintenance Package

Package	Monthly Fee	Includes
Basic	\$500/mo	Bug fixes, security patches
Standard	\$1,500/mo	Basic + minor feature updates, hosting management
Premium	\$3,500/mo	Standard + priority support, major updates, 24/7 monitoring

10. Terms & Conditions

10.1 Intellectual Property

- Source Code Ownership:** Upon full payment of the development fee, Client owns 100% of the delivered source code.
- Revenue Share Rights:** The revenue share agreement grants Developer a financial interest in platform revenue, not ownership of the code or business.
- Third-Party Components:** Open-source libraries and frameworks remain under their respective licenses.
- Residual Rights:** Developer retains no rights to use Client's specific implementation, branding, or user data.

10.2 Warranties

- Functional Warranty:** Developer warrants the software will function as described in this proposal for 90 days post-delivery.
- Bug Fixes:** Developer will fix any bugs reported within the warranty period at

no additional cost.

3. **Exclusions:** Warranty does not cover issues caused by:

- Third-party service failures
- Client modifications to source code
- Hosting environment issues
- Misuse of the platform

10.3 Confidentiality

Both parties agree to keep confidential: - Business strategies and plans - Technical implementations - Financial information - User data and metrics

10.4 Limitation of Liability

Developer's total liability shall not exceed the development fee paid (per selected option: €15,000, €20,000, or €25,000). Developer is not liable for: - Lost profits or revenue - Business interruption - Third-party claims - Indirect or consequential damages

10.5 Revenue Share Compliance

1. Client agrees to maintain accurate financial records
2. Client will provide monthly revenue reports by the 5th of each month
3. Late payments incur 1.5% monthly interest
4. 3 consecutive missed payments constitute breach

10.6 Termination

By Client: - May terminate development with 14 days notice - Fees paid for completed milestones are non-refundable - Revenue share continues on any delivered components used

By Developer: - May terminate for non-payment (30 days overdue) - Will deliver all completed work upon termination

Revenue Share Termination: - 5-year minimum term - After 5 years: Client may terminate with 90 days notice - Buyout option available (see Section 7.2)

Acceptance

By signing below, both parties agree to the terms outlined in this proposal.

Client

Company Name: _____

Authorized Signatory: _____

Title: _____

Signature: _____

Date: _____

Developer

Company Name: _____

Authorized Signatory: _____

Title: _____

Signature: _____

Date: _____

Appendix A: Feature Specifications

Detailed technical specifications available in TECHNICAL_DOCUMENTATION.md

Appendix B: API Endpoint List

Module	Endpoints	Description
Authentication	4	Register, login, refresh, password reset
Users	5	Profile CRUD, settings
Parking Spots	8	Listings CRUD, search, availability
Bookings	8	Booking CRUD, pricing, status
Reviews	7	Reviews CRUD, ratings
Payments	8	Payments, refunds, payouts
Total	40	Complete REST API

Questions? Contact [Your Email] or [Your Phone]

This document constitutes a binding agreement upon signature by both parties.