

Service Definition Document: SaaS Application Development

Version: 1.0

Date: 28/01/2026

Provider: SRC UK Ltd

1. Executive Summary

This document defines the scope, methodology, and operational framework of the Software as a Service (SaaS) Development service provided by **SRC UK Ltd**. Our service is designed to deliver scalable, secure, and high-performance cloud-native applications tailored to specific business requirements. This definition serves as the baseline for all engagements and Statements of Work (SOW).

2. Service Overview

SRC UK Ltd provides end-to-end SaaS development, focusing on building multi-tenant architectures that allow clients to license software to end-users via the cloud. Our service covers the entire software development lifecycle (SDLC), from initial conceptualization and architecture design to deployment, monitoring, and ongoing maintenance.

Key Service Features

- **Cloud-Native Architecture:** Design of microservices or modular monoliths optimized for AWS, Azure, or Google Cloud.
- **Multi-Tenancy:** Secure logical separation of data and resources for multiple end-users within a single instance.
- **Scalability:** Auto-scaling infrastructure to handle fluctuating user loads.
- **Security Compliance:** Adherence to industry standards (e.g., SOC2, GDPR, HIPAA where applicable).

3. Scope of Services

Our development process is divided into five distinct phases.

Phase 1: Discovery & Strategy

We define the Minimum Viable Product (MVP) or full product roadmap.

- Requirement gathering and stakeholder interviews.
- Technical feasibility analysis.
- Creation of the Software Architecture Document (SAD).

Phase 2: UX/UI Design

We ensure the application is intuitive and accessible.

- Wireframing and user flow mapping.
- High-fidelity prototyping.
- Design system creation for consistent branding.

Phase 3: Core Development

The construction of the platform using modern technology stacks.

- **Frontend:** [e.g., React, Vue, Angular] implementation.
- **Backend:** [e.g., Node.js, Python, .NET] API development.
- **Database:** Design of relational (SQL) or non-relational (NoSQL) schemas.
- **API Integration:** Integration with third-party services

Phase 4: Quality Assurance (QA)

Rigorous testing to ensure stability.

- Automated unit and integration testing.
- Performance and load testing.
- Security penetration testing (vulnerability scanning).

Phase 5: Deployment & DevOps

- CI/CD pipeline setup for automated deployment.
- Infrastructure as Code (IaC) implementation (e.g., Terraform).
- Production environment setup and cutover.

4. Service Methodology

We utilize an **Agile Scrum** methodology to ensure transparency and adaptability.

- **Sprints:** Work is delivered in 2-week cycles.
- **Demos:** Functional software is demonstrated to stakeholders at the end of every sprint.
- **Backlog Management:** Priorities are reviewed and adjusted bi-weekly based on business value.

5. Support & Maintenance

Post-launch, we offer tiered support packages to ensure platform reliability.

Feature	Silver (Standard)	Gold (Business)	Platinum (Enterprise)
Help Desk Hours	9-5 Business Days	24/5	24/7
Response Time (Critical)	< 4 Hours	< 2 Hours	< 1 Hour
Uptime Guarantee	99.5%	99.9%	99.99%
Maintenance	Monthly Patching	Weekly Patching	Real-time Monitoring

6. Roles and Responsibilities

Provider (SRC UK Ltd)

- Provide qualified resources (Developers, PMs, QAs).
- Manage the technical infrastructure and code repositories.
- Report on progress, blockers, and budget burn-rate weekly.

Client

- Designate a Product Owner for decision-making, or one can be provided on behalf of SRC UK.
- Provide timely feedback on sprint deliverables (within 48 hours).
- Grant access to necessary third-party accounts (domain registrars, legacy systems).

7. Exclusions & Assumptions

Unless explicitly stated in a specific Statement of Work (SOW), the following are excluded from the standard service:

- Content creation (copywriting, video production).
- End-user marketing and customer acquisition.
- Hardware procurement.
- License fees for third-party SaaS tools used within the product (e.g., paid APIs).

Assumption: The client has legal ownership or rights to all data and intellectual property provided for the development process.