MIROSLAV TADEJ CONSULTING

Secure Development Policy

Building software securely, by design

OWASP Top 10 · Secure by Design · Aligned with ISO/IEC 27001:2022 technological controls

Document control

Document title	Secure Development Policy
Document owner	Miroslav Tadej, Principal Consultant
Organisation	Miroslav Tadej Consulting
Version	1.0
Status	Approved
Classification	Public — issued for tender and client assurance
Effective date	13/06/2026
Next review date	12/06/2027
Approved by	Miroslav Tadej, Principal Consultant

Contents

1. Purpose

This Secure Development Policy defines how Miroslav Tadej Consulting ("the Consultancy") builds, tests, and maintains software securely. Software development is central to the Consultancy's service, and security is therefore not an afterthought but a defining characteristic of how that software is produced. This policy translates the "Secure by Design" commitment in the Information Security Policy into concrete development practice.

2. Scope

This policy applies to all software the Consultancy designs, develops, tests, deploys, or maintains — for clients and for its own platforms — across the full development lifecycle, and to the Principal Consultant and any contractor who writes or reviews code under the Consultancy's direction. It covers source code, dependencies, build and deployment pipelines, development and production environments, and the handling of secrets and credentials.

3. Secure development principles

All development is governed by the following principles:

- **Secure by Design:** security requirements are considered from the start of every engagement and built into the design, not bolted on afterwards.
- **Least privilege:** every component, user, and process is granted the minimum access necessary.
- **Defence in depth:** security is layered so that the failure of one control does not lead to compromise.
- **Fail securely:** systems fail safely — access is denied by default unless explicitly permitted.
- **Never trust input:** all input is treated as untrusted and validated; all output is encoded for its context.
- **Keep it simple:** simple, well-understood designs are preferred over complex ones that are hard to secure.

4. Secure development lifecycle

Security activities are embedded at each stage of the development lifecycle:

4.1 Requirements and design

- security and data-protection requirements are captured alongside functional requirements;
- sensitive data flows and trust boundaries are identified, and high-risk flows (authentication, payments, privileged actions) are threat-modelled;
- critical state lifecycles are defined explicitly so that invalid transitions are prevented; and
- where processing is likely to be high-risk to individuals, a data-protection impact assessment is considered.

4.2 Implementation

- code is written to the secure coding standards in section 5 and to the OWASP guidance in section 6;
- secrets and credentials are never hard-coded or committed to source control; configuration is supplied through environment variables or a secret store; and
- a committed example configuration documents required variables without exposing real values.

4.3 Testing and review

- code is reviewed for security before release; dependencies are scanned for known vulnerabilities;
- automated tests cover critical and security-sensitive flows; and

- application security is verified against the OWASP Top 10 prior to deployment, with findings remediated and recorded.

4.4 Deployment and maintenance

- deployments run through a controlled, repeatable pipeline; database changes are applied through forward-only, reviewed migrations;
- production configuration is hardened and separated from development; and
- systems and dependencies are kept patched, and security monitoring is in place throughout the supported life of the software.

5. Secure coding standards

The Consultancy applies consistent secure-coding practices across its work, including:

- **Injection prevention:** all database access uses parameterised queries; user input is never concatenated into queries, commands, or markup.
- **Input & output handling:** all input is validated against expected type, format, and range on the server; output is encoded for its context to prevent cross-site scripting.
- **Authentication:** passwords are hashed with a strong, salted algorithm (bcrypt); session and token handling is secure, using short-lived access tokens with refresh-token rotation carried in httpOnly cookies.
- **Authorisation:** authorisation is enforced on the server for every protected action, at both the route and data layers, with object-level ownership checks.
- **Data integrity:** monetary values are stored as integers (minor units) to avoid rounding errors; multi-step financial operations are wrapped in transactions to preserve integrity.
- **Error handling:** errors are handled centrally; messages returned to users never expose stack traces, secrets, or internal detail.
- **Payments:** payment flows use a PCI-compliant provider so card data never reaches Consultancy or client servers; payment webhooks verify provider signatures.

6. OWASP Top 10 control mapping

The Consultancy maps its controls to the OWASP Top 10, the industry-standard list of the most critical web-application security risks. The following table summarises how each category is addressed:

OWASP Top 10 (2021)	How the Consultancy addresses it
A01 Broken Access Control	Role-based access control enforced at both route and data layers; deny-by-default; object-level ownership checks to prevent IDOR/BOLA; admin routes double-gated by authentication and role.
A02 Cryptographic Failures	TLS for all data in transit; encryption at rest; passwords hashed with bcrypt; secrets held in secure stores, never in source; no sensitive data in logs or URLs.
A03 Injection	Parameterised queries for all database access — no string concatenation; input validation and output encoding; ORM/query-builder used safely.
A04 Insecure Design	“Secure by Design” from the outset; threat modelling of sensitive flows; explicit state machines for critical lifecycles; least privilege as a default.
A05 Security Misconfiguration	Hardened configuration; security headers and content-security policy; CORS restricted to known origins; environments separated; no default credentials.
A06 Vulnerable & Outdated Components	Dependencies tracked and scanned with automated alerting; security updates prioritised by severity; unused dependencies removed.
A07 Identification & Authentication Failures	Strong authentication with MFA where supported; short-lived JWT access tokens with refresh-token rotation in httpOnly cookies; secure session handling; protection against brute force.
A08 Software & Data Integrity Failures	Trusted dependency sources; integrity of build and deploy pipeline; signed/verified artifacts where available; webhook signature verification.

OWASP Top 10 (2021)	How the Consultancy addresses it
A09 Security Logging & Monitoring Failures	Security-relevant events logged; centralised structured logging and error monitoring in production; alerting on anomalies; logs protected and retained.
A10 Server-Side Request Forgery (SSRF)	Validation and allow-listing of outbound requests; restriction of internal network access from application servers.

7. Dependency and supply-chain security

Third-party components are a significant source of risk. The Consultancy uses only dependencies that are necessary and from trusted sources, declares every dependency explicitly, and monitors them for known vulnerabilities through automated alerting. Security-relevant updates are assessed and applied promptly, prioritised by severity. The integrity of the build and deployment pipeline is protected, and unused or unmaintained dependencies are removed.

8. Secrets and credential management

Secrets — API keys, database credentials, tokens, and signing keys — are never committed to source control or embedded in code. They are supplied at runtime through environment configuration or a dedicated secret store, separated between environments, and rotated when staff or contractor access changes or when a compromise is suspected. Source repositories are scanned to detect accidental exposure of secrets, including in history.

9. Environment separation and access

Development, testing, and production environments are kept separate. Production data is not used in development or testing unless it has been anonymised. Access to production environments and code repositories is restricted, protected by multi-factor authentication, and reviewed periodically. Administrative actions in production are logged.

10. Vulnerability management and disclosure

Vulnerabilities identified through scanning, testing, monitoring, or external reports are recorded, assessed for severity, and remediated within a timeframe proportionate to the risk. Where the Consultancy maintains software for a client, identified vulnerabilities and their remediation are communicated to the client. The Consultancy supports responsible disclosure and provides a means for security issues to be reported and addressed.

11. Roles and responsibilities

The Principal Consultant is responsible for upholding this policy, performing secure design and review, and maintaining current secure-development competence — underpinned by formal cybersecurity qualification at NFQ Level 9. Any contractor who writes or reviews code on the Consultancy's behalf is required to comply with this policy and its standards before being granted access to code or environments.

12. Review

This policy is reviewed at least every twelve months and whenever there is a significant change in the Consultancy's technology stack, tooling, threat landscape, or the standards it follows (including updates to the OWASP Top 10). Reviews ensure the policy remains current and effective, and result in documented improvements.

13. Related documents

- Information Security Policy
- Data Protection Policy
- GDPR Compliance Statement

- Business Continuity & Disaster Recovery Plan
- Quality Management Statement

14. Approval

This Secure Development Policy is approved and issued by the Principal Consultant of Miroslav Tadej Consulting and takes effect from the effective date recorded in the document-control table above.

Name & role	Signature	Date
Miroslav Tadej Principal Consultant	Miroslav Tadej	13/06/2026