

THE COMPLETE APP LEGAL & SECURITY CHECKLIST

15 Essential Checks Before You Launch Your App, SaaS, or AI Product

Protect your users, review your responsibilities, and identify critical gaps before your next deployment.

Prepared by SwiTech Solution | Website: <https://switech-solution.com> | Edition: 2026-202

7

PROMINENT DISCLAIMER: This checklist is an educational and operational reference resource intended for technology founders, product managers, and engineering teams. **It does not constitute legal advice, regulatory counsel, or a formal cybersecurity audit.** Completing this checklist does not create an attorney-client relationship. Legal requirements vary significantly based on your corporate domicile, user locations, data processing architectures, and vertical domain. Always consult with a qualified technology attorney and certified information security expert prior to commercial launch.

SECTION 1: Executive Scope & Legal Notice

Operational Scope

Launching a web application, mobile software, SaaS platform, or AI-powered API service exposes an enterprise to a complex matrix of regulatory and operational vulnerabilities. No standardized document, checklist, or automated scanner can guarantee 100% legal immunity or absolute defense against advanced cyber threats.

Applicable compliance regimes depend heavily on factors such as:

- **Jurisdictional Reach:** Operating across borders invokes conflicting regulatory standards. Serving users in the European Union triggers the EU General Data Protection Regulation (EU GDPR); operating within Switzerland subjects entities to the Swiss

Federal Act on Data Protection (Swiss FADP); serving US residents requires adherence to state-level regulations (e.g., CCPA/CPRA in California, VCDPA in Virginia) alongside federal sector-specific mandates.

- **Target Audience & Demographics:** Processing data belonging to minors (e.g., US COPPA, UK Age Appropriate Design Code) carries stringent affirmative consent rules and liability structures.
- **Data Classification:** Processing standard telemetry differs fundamentally from handling "Special Category Data" (biometrics, health metrics, political affiliations, trade union memberships, financial telemetry, or precise geolocation).
- **Architectural Footprint:** Self-hosted physical infrastructure, multi-tenant public cloud configurations (AWS, GCP, Azure), serverless paradigms, and third-party AI LLM API integrations present distinct data ownership, transit, and subprocessor risks.

Classification Legends

Legal & Operational Status Labels

To assist product and legal engineering teams, items throughout this manual are tagged with standard regulatory classifications:

- **[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW]:** Mandated by specific statutory frameworks (e.g., GDPR, FADP, CCPA). Failure to execute carries direct administrative fines or regulatory enforcement.
- **[RECOMMENDED PRACTICE]:** Industry-standard operational baseline. Reduces litigation exposure, improves software reliability, and satisfies institutional due diligence.
- **[VERIFY WITH A QUALIFIED PROFESSIONAL]:** Highly complex, variable, or context-sensitive item requiring explicit sign-off from specialized legal counsel or a certified cybersecurity auditor.

Risk Priority Classification Matrix

- **P1 (High Priority — Launch Blocker):** Critical legal non-compliance or high-severity security vulnerability. Operating without this check exposes the organization to immediate regulatory enforcement, statutory fines, or automated compromise. Must be resolved before opening access to production users.
 - **P2 (Important — Systemic / Post-MVP):** Mandatory infrastructure or contractual arrangement required for operational scaling and medium-term risk reduction. Must be resolved within 30–60 days of launch or immediately upon reaching initial commercial traction.
 - **P3 (Ongoing Maintenance & Audits):** Continuous governance processes, routine audits, and periodic operational reviews executed throughout the product lifecycle.
-

SECTION 2: The Master Checklist (Executive Summary)

ID	Checklist Area	Priority	Status	Operational Summary	Default Owner
01	Privacy Policy	P1		Comprehensive, clear disclosure of data practices and user rights	Legal / Product
02	Terms & Conditions	P1		Binding end-user contract defining usage, limits, and liability	Legal / Ops
03	Privacy Law Compliance	P1		Architectural alignment with GDPR, Swiss FADP, and US frameworks	Data Protection Officer
04	Cookie & Tracking Governance	P1		Banner consent, ePrivacy alignment, and server-side compliance	Lead Frontend Eng
05	Legal Basis for Processing	P1		Mapping operations to legitimate interest, consent, or contract	DPO / Legal
06	Marketing & Outreach Consent	P1		Opt-in verification, double opt-in logs, and suppression logic	Growth / Marketing
07	Data Subject Requests (DSR)	P1		Automated workflows for access, correction, and cascade deletion	Lead Backend Eng
08	Data Retention Schedules	P2		Storage minimization limits and automated purging pipelines	Data Engineer
09	Data Processing Agreements	P1		Signed DPAs, Subprocessor audits, and valid SCC frameworks	Legal / Procurement
10	Third-Party Transparency	P1		Exhaustive public disclosure of vendors, APIs, and host regions	Devops / Security
11	Application Security Baseline	P1		OWASP compliance, secrets isolation, AuthN/AuthZ, RLS, rate limits	CISO / Tech Lead
12	Incident Response & Breach Plan	P1		Defined containment, 72h/FADP alert procedures, and logging	SecOps / Legal

ID	Checklist Area	Priority	Status	Operational Summary	Default Owner
13	IP & Code Ownership	P1		Work-for-hire assignment, trademarking, AI IP liability review	Founder / IP Counsel
14	Open-Source Compliance	P2		Licensing audit (Permissive vs. Copyleft), SBOM generation	Lead Architect
15	Payment & Subscription Governance	P1		Chargeback mitigation, statutory cancellation, price transparency	Product / Finance

SECTION 3: Detailed Technical & Legal Checks

CHECK 01 — PRIVACY POLICY

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

A Privacy Policy is a mandatory, publicly accessible legal document that transparently details how an organization collects, processes, stores, shares, and protects personal data obtained from users.

2. Why It Matters

Global privacy statutes mandate transparent communication. Operating without an accessible, accurate, and up-to-date Privacy Policy exposes the entity to immediate regulatory fines, rejection from app stores (Apple App Store, Google Play), and revocation of API integrations (OAuth providers like Google, Meta, Microsoft).

3. Potential Risks If Neglected

- Class-action lawsuits and regulatory fines (GDPR fines up to 4% of global turnover or €20M).
- Immediate rejection or removal of application from the Apple App Store and Google Play Store.
- Termination of critical vendor accounts (e.g., Stripe, Google Cloud Platform, AWS).

4. Step-by-Step Review Checklist

- [] Document is published on a dedicated, publicly accessible URL (e.g., [/privacy](#)) without login barriers.
- [] Identifies the exact Legal Entity Name, physical registration address, and contact email.
- [] Explicitly enumerates all data attributes collected (e.g., name, email, IP address, device IDs, billing data, telemetry).
- [] Outlines the precise legal bases for processing each data category.
- [] Lists all third-party sub-processors, categories of recipients, and external service providers.
- [] Explains user rights clearly (Access, Erasure, Rectification, Portability, Restriction, Objection).
- [] Specifies data retention periods or criteria used to determine retention durations.
- [] Details international data transfer mechanisms (e.g., Standard Contractual Clauses).
- [] Provides direct contact details for your Data Protection Officer (DPO) or privacy contact person.

5. Practical Implementation Examples

- **SaaS Application:** Include an explicit hyperlink to [/privacy](#) on the registration interface, footer, and inside transaction confirmation emails.
- **Mobile Application:** Link the policy within the App Store Connect metadata, Google Play Console console, account creation screen, and settings panel.
- **AI Tooling:** Clearly state whether prompt text, uploaded documents, or user interactions are utilized for downstream AI model training or fine-tuning.

6. Common Mistakes & Anti-Patterns

- Copy-pasting a Privacy Policy from an unrelated business without modifying operational realities.
- Hiding the Privacy Policy link behind user authentication or within complex deep menus.
- Failing to update the policy when integrating new analytical, marketing, or infrastructure vendors.

7. Questions the Founder Must Answer

- "What personal data enters our database the exact millisecond a user interacts with our software?"
- "Do we feed user content into third-party AI models, and is that explicitly disclosed?"

8. Recommended Evidence or Documentation

- Dated archive or version control history (Git tracking) of all published Privacy Policy iterations.

- Timestamped audit logs recording user acknowledgment of the privacy terms during sign-up.

9. Priority & Suggested Next Action

- **Action:** Audit all production data ingestion points and publish an updated, accurate Privacy Policy at </privacy>.

10. Relevant Official Resources

- [European Commission — Data Protection for Business](#)
 - [Swiss FDPIC — Privacy Statements on the Internet](#)
-

CHECK 02 — TERMS & CONDITIONS

[RECOMMENDED PRACTICE] | Priority: P1

1. What It Means

Terms & Conditions (Terms of Service / Terms of Use) form the legal contract governing the relationship between the software provider and the end user. It establishes operational rules, payment obligations, intellectual property rights, acceptable use policies, and liability limitations.

2. Why It Matters

Without enforceable Terms, your company lacks legal protection against system abuse, content scraping, payment fraud, unexpected downtime claims, and indirect operational damages incurred by users.

3. Potential Risks If Neglected

- Unlimited financial and legal liability for platform outages, data loss, or system errors.
- Inability to ban abusive users, malicious bots, or unauthorized scrapers.
- Payment dispute vulnerabilities and elevated chargeback ratios.

4. Step-by-Step Review Checklist

- ☐ Defines legal entity identity, governing law, and exclusive venue for dispute resolution.
- ☐ Contains explicit Acceptable Use Policies (prohibiting reverse engineering, scraping, security circumvention, illegal activity).
- ☐ Includes explicit Disclaimer of Warranties ("AS IS" and "AS AVAILABLE") and Limitation of Liability clauses.
- ☐ Details billing terms, recurring payment rules, cancellation policies, and refund eligibility.

- [] Asserts full company ownership over software, branding, source code, and platform intellectual property.
- [] Specifies account termination rules and rights of service suspension for policy breaches.
- [] Enforces user agreement via an active, explicit click-wrap mechanism (e.g., checking an unchecked box).

5. Practical Implementation Examples

- **SaaS Application:** Force new registrants to actively check [x] I agree to the Terms of Service prior to enabling the "Create Account" submit button.
- **Mobile Application:** Display a blocking modal on first launch requiring explicit agreement to the Terms before accessing application features.
- **AI Tooling:** Include specific clauses on AI output ownership, disclaimers regarding model hallucinations, and explicit restrictions against prompt injection attacks.

6. Common Mistakes & Anti-Patterns

- Relying on "Browse-wrap" agreements (assuming users agree simply by browsing the website).
- Neglecting to state which geographical court holds legal jurisdiction over contract disputes.
- Writing unlimited liability allowances into standard user contracts.

7. Questions the Founder Must Answer

- "If our API fails for 48 hours and a customer loses commercial revenue, does our contract limit our liability to amounts paid to us?"
- "Who owns the output generated by our users when using our tools or AI capabilities?"

8. Recommended Evidence or Documentation

- Immutable database logs capturing user ID, timestamp, IP address, and exact version ID of the Terms agreed to upon signup.

9. Priority & Suggested Next Action

- **Action:** Implement explicit click-wrap enforcement on all signup forms and review liability caps with legal counsel.

10. Relevant Official Resources

- [EU Consumer Rights Directive Guidelines](#)

CHECK 03 — GDPR AND APPLICABLE PRIVACY LAWS

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Ensuring software architecture, operational processes, and storage locations comply with regional privacy regulations—specifically the European Union GDPR, the revised Swiss Federal Act on Data Protection (Swiss FADP), and applicable US state privacy acts (CCPA/CPRA).

2. Why It Matters

Privacy laws enforce extraterritorial jurisdiction: if you process personal data of individuals residing in the EU or Switzerland, you are subject to their statutory requirements regardless of where your corporate entity is headquartered.

3. Potential Risks If Neglected

- Severe statutory fines (up to 20 million EUR or 4% of worldwide annual turnover under GDPR; personal criminal fines for executives up to CHF 250,000 under revised Swiss FADP for intentional violations).
- Cross-border operational injunctions ordering the immediate halt of processing activities.

4. Step-by-Step Review Checklist

- ☐ Determine exact regulatory applicability based on target user base (EU GDPR, Swiss FADP, CCPA/CPRA).
- ☐ Identify whether processing "Special Category Data" occurs (requiring heightened security and explicit consent).
- ☐ Validate mechanisms for cross-border data transfers outside the EU/Switzerland (Standard Contractual Clauses, Data Privacy Framework).
- ☐ Maintain an internal Data Processing Inventory / Register of Processing Activities (ROPA).
- ☐ Appoint an EU/UK Representative or Data Protection Officer (DPO) if legally mandated.
- ☐ Verify architectural compliance with **Privacy by Design** and **Privacy by Default** principles.

5. Practical Implementation Examples

- **SaaS Application:** Restrict default telemetry options to opt-in, enforcing privacy-preserving settings upon account provision.
- **Mobile Application:** Prompt for permissions (location, camera, contacts) strictly in-context when features are triggered, rather than globally at install.
- **AI Tooling:** Provide mechanism to opt-out of model training pipelines while retaining application functionality.

6. Common Mistakes & Anti-Patterns

- Assuming that storing data on AWS servers located in Frankfurt or Zurich automatically makes an application "GDPR compliant."

- Conflating Swiss FADP requirements with EU GDPR; the Swiss FADP imposes direct criminal liability on individuals/management for specific intentional violations.
- Neglecting to account for US state-level privacy statutes when onboarding North American users.

7. Questions the Founder Must Answer

- "Do we have a legally valid transfer mechanism in place for data routed to US-based API services?"
- "Can we prove our default system settings prioritize user privacy without requiring manual user adjustments?"

8. Recommended Evidence or Documentation

- Completed Register of Processing Activities (ROPA).
- Formally executed Standard Contractual Clauses (SCCs) with non-EU/Swiss sub-processors.

9. Priority & Suggested Next Action

- **Action:** Map all data collection points and draft a formal ROPA document capturing all personal processing streams.

10. Relevant Official Resources

- [Swiss Federal Data Protection and Information Commissioner \(FDPIC\)](#)
- [European Data Protection Board \(EDPB\)](#)

CHECK 04 — COOKIE CONSENT AND TRACKING

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Adherence to ePrivacy directives and privacy laws governing the execution of non-essential tracking technologies (cookies, local storage, pixel tags, server-side fingerprinting, mobile advertising identifiers) on end-user devices.

2. Why It Matters

Non-essential cookies and tracking scripts are illegal to load prior to receiving explicit, freely given user consent in the EU, UK, and Switzerland. "Implied consent" (e.g., "By continuing to use this site you accept cookies") is non-compliant under current enforcement standards.

3. Potential Risks If Neglected

- Automated regulatory scanning, enforcement notices, and substantial financial penalties.

- Ad-account and analytics platform suspensions (e.g., Meta Pixel, Google Analytics account termination).

4. Step-by-Step Review Checklist

- [] Block ALL non-essential, marketing, and analytical scripts prior to explicit user opt-in.
- [] Implement a compliant Consent Management Platform (CMP) or script blocker.
- [] Ensure the consent banner offers equal visual weight for "Accept All" and "Reject All" options.
- [] Categorize cookies accurately (Strictly Necessary, Performance, Analytics, Marketing).
- [] Provide a persistent link in the web site footer allowing users to modify or withdraw cookie preferences at any time.
- [] Maintain proof of explicit consent logs stored without capturing unnecessary personal data.

5. Practical Implementation Examples

- **SaaS Application:** Utilize a script manager (e.g., Cookiebot, OneTrust, or custom open-source blocking logic) that delays execution of PostHog/Mixpanel scripts until the user clicks "Accept".
- **Mobile Application:** Integrate Apple's App Tracking Transparency (ATT) framework, showing standard permission dialogs prior to activating tracking SDKs.
- **AI Tooling:** Ensure analytics logging inside chat UI frameworks does not inadvertently mirror user prompt text into tracking platforms.

6. Common Mistakes & Anti-Patterns

- Pre-checking consent checkboxes on the CMP banner.
- Making the "Reject" button hidden within secondary configuration menus while placing "Accept" on the primary view.
- Utilizing server-side tracking (e.g., Facebook Conversions API) to bypass client-side consent choices.

7. Questions the Founder Must Answer

- "If a user lands on our marketing site and does not interact with the banner, do any tracking cookies fire?"
- "Can a user easily change their cookie preferences after closing the banner?"

8. Recommended Evidence or Documentation

- Script load audit logs verifying zero third-party cookie initialization prior to explicit consent execution.

9. Priority & Suggested Next Action

- **Action:** Execute an un-authenticated browser scan using developer tools to verify no analytics or ad scripts fire before consent.

10. Relevant Official Resources

- [EDPB Guidelines on Consent under Regulation 2016/679](#)
-

CHECK 05 — LEGAL BASIS FOR DATA PROCESSING

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Establishing, documenting, and maintaining an explicit legal basis under privacy laws (e.g., GDPR Article 6) for every unique processing activity carried out by your software infrastructure.

2. Why It Matters

Processing personal data without a clear, valid legal basis is inherently unlawful under modern privacy frameworks. You cannot collect user data simply because it "might be useful later."

3. Potential Risks If Neglected

- Invalidation of entire database collections, requiring mandatory data destruction.
- Direct fines for unlawful processing independent of security breaches or operational errors.

4. Step-by-Step Review Checklist

- [] Document the legal basis for every data pipeline:
 - **Contractual Necessity:** Required to deliver core software functionality.
 - **Explicit Consent:** Opt-in required for non-essential features, marketing, tracking.
 - **Legitimate Interest:** Internal system security, fraud detection (requires a formal Legitimate Interest Assessment).
 - **Legal Obligation:** Financial record retention, tax compliance.
- [] Ensure consent requests are kept distinct from acceptance of standard Terms & Conditions.
- [] Audit data fields to confirm strict alignment with the **Data Minimization** principle.

5. Practical Implementation Examples

- **SaaS Application:** Collecting payment information is justified by *Contractual Necessity*; sending weekly marketing newsletters requires *Explicit Consent*.

- **Mobile Application:** Requesting access to device Bluetooth for core hardware control is *Contractual*; using it for proximity marketing requires *Consent*.
- **AI Tooling:** Retaining API inputs for system safety auditing is justified by *Legitimate Interest*; using inputs to train general models requires explicit *Consent*.

6. Common Mistakes & Anti-Patterns

- Claiming "Legitimate Interest" for intrusive email marketing campaigns without performing a Legitimate Interest Assessment (LIA).
- Bundling consent for privacy-invasive practices inside mandatory Terms of Service agreements.

7. Questions the Founder Must Answer

- "If challenged by a data protection agency, can we produce a document explaining the exact legal ground for collecting user device telemetry?"

8. Recommended Evidence or Documentation

- Documented Legitimate Interest Assessments (LIA) for platform security and telemetry features.

9. Priority & Suggested Next Action

- **Action:** Complete a Data Processing Inventory mapping every form field to its specific legal basis.

10. Relevant Official Resources

- [ICO Guidance on Legal Bases for Processing](#)
-

CHECK 06 — MARKETING CONSENT

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

System architecture and operational workflows ensuring commercial communications (email marketing, SMS, push notifications, cold outreach) strictly comply with direct marketing and anti-spam legislation (e.g., ePrivacy, CAN-SPAM, CASL).

2. Why It Matters

Unsolicited marketing triggers severe monetary fines per infraction, destroys domain sender reputation, and leads to blacklisting by major email service providers (SendGrid, Mailgun, AWS SES).

3. Potential Risks If Neglected

- Escalating financial penalties per email sent (CAN-SPAM fines up to \$50,000+ per non-compliant email).
- Domain name suspension and blacklisting of corporate transactional email systems.

4. Step-by-Step Review Checklist

- ☐ Implement Double Opt-In (DOI) verification for marketing newsletter registrations where required.
- ☐ Maintain explicit separation between transactional emails (password resets, invoices) and promotional emails.
- ☐ Embed a functional, one-click "Unsubscribe" header and visible link in every promotional email.
- ☐ Maintain an automated, centralized suppression list ensuring unsubscribed users are never re-contacted.
- ☐ Verify B2B cold outreach compliance with local restrictions (e.g., strict restrictions in Switzerland and specific EU member states).

5. Practical Implementation Examples

- **SaaS Application:** Include separate, unchecked opt-in options for promotional content on signup: ☐ `Send me monthly product tips.`
- **Mobile Application:** Request OS-level push notification authorization strictly after explaining value, maintaining distinct settings for transactional updates vs. promotional offers.
- **AI Tooling:** Never auto-subscribe users signing up for free platform trials directly to paid product partner promotional emails.

6. Common Mistakes & Anti-Patterns

- Using transactional email infrastructures (e.g., Postmark) to send mass marketing promotions.
- Failing to honor unsubscribe requests within required timeframes (GDPR mandates immediate effect; CAN-SPAM within 10 business days).
- Pre-checking marketing agreement options during customer checkout flows.

7. Questions the Founder Must Answer

- "If a user unsubscribes from marketing emails, does our system guarantee they still receive password reset links while blocking future product newsletters?"

8. Recommended Evidence or Documentation

- Double Opt-In timestamp logs capturing user confirmation clicks, IP addresses, and email headers.

9. Priority & Suggested Next Action

- **Action:** Configure transactional and promotional email dispatch across entirely separate API keys, domains, or subdomains (e.g., `mail.domain.com` vs `marketing.domain.com`).

10. Relevant Official Resources

- [FTC CAN-SPAM Act Compliance Guide for Business](#)
-

CHECK 07 — DATA ACCESS AND DELETION

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Architectural workflows enabling users to exercise Data Subject Access Requests (DSAR), including requesting a copy of all personal data held, requesting corrections, and executing the "Right to be Forgotten" (complete account deletion).

2. Why It Matters

Data protection laws mandate that data subjects can access or delete their data within strict timeframes (GDPR: 30 days; FADP: 30 days). Failure to delete data across primary stores, secondary caches, and third-party tools creates direct regulatory non-compliance.

3. Potential Risks If Neglected

- Fines for ignoring statutory user rights requests.
- Residual data exposure during unexpected database breaches occurring long after a user requested account deletion.

4. Step-by-Step Review Checklist

- [] Build an automated or operational workflow to process account deletion within statutory limits (30 days).
- [] Implement cascade deletion across all relational tables, document stores, vector databases, and backup snapshots.
- [] Ensure third-party APIs (analytics, CRM, payment processors) receive downstream deletion signals.
- [] Provide exported user data in a structured, commonly used machine-readable format (JSON or CSV).
- [] Establish mechanisms to handle edge-case data assets (AI embedding stores, application logs, prompt histories).

5. Practical Implementation Examples

- **SaaS Application:** Provide a self-service "Delete Account" button within settings that executes relational foreign-key cascades across the user database.
- **Mobile Application:** Apple App Store guidelines **require** built-in account deletion functionality inside the application if account creation is supported.
- **AI Tooling:** Ensure user profile deletion triggers an explicit purge of high-dimensional vector embeddings and vector ID references stored in Pinecone/Weaviate/pgvector.

6. Common Mistakes & Anti-Patterns

- Performing "soft deletes" (setting `is_deleted = true`) while retaining personal identifiable information (PII) indefinitely without anonymization.
- Forgetting that log aggregators (Datadog, Papertrail) store PII captured in unformatted stack traces.

7. Questions the Founder Must Answer

- "When a user clicks 'Delete Account', does their email address remain in third-party error logging tools or AI vector index stores?"

8. Recommended Evidence or Documentation

- Automated script logs and ticketing records showing successful end-to-end execution of deletion routines within 30 days.

9. Priority & Suggested Next Action

- **Action:** Build and test a script verifying that initiating account deletion cascades through every primary database table, vector store, and third-party integration.

10. Relevant Official Resources

- [EDPB Guidelines on Data Subject Rights](#)

CHECK 08 — DATA RETENTION

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P2

1. What It Means

Establishing defined timelines for retaining personal data, enforcing the **Storage Limitation** principle, and automatically purging data once original operational purposes expire or statutory limits are reached.

2. Why It Matters

Retaining personal data indefinitely is illegal under GDPR and FADP. Old, unmaintained data increases storage costs and significantly expands legal liability during a data security breach.

3. Potential Risks If Neglected

- Mandatory administrative penalties for storage limitation non-compliance.
- Increased blast radius and severe breach liabilities from compromised legacy user archives.

4. Step-by-Step Review Checklist

- [] Create a formal Data Retention Policy specifying maximum holding periods for all categories of data.
- [] Align retention limits with local commercial retention requirements (e.g., Swiss Code of Obligations requires keeping financial records/invoices for 10 years).
- [] Implement automated retention-expiry jobs (e.g., auto-deleting system error logs after 30 to 90 days).
- [] Anonymize or aggregate analytical data retained for long-term platform reporting.

5. Practical Implementation Examples

- **SaaS Application:** Configure database retention policies to automatically drop unverified user accounts after 14 days.
- **Mobile Application:** Clear local caches and transient analytics payloads upon user logout.
- **AI Tooling:** Set strict time-to-live (TTL) limits (e.g., 30 days) on raw audio/text files submitted for inference processing.

6. Common Mistakes & Anti-Patterns

- Retaining full application database backups indefinitely in unencrypted S3 buckets.
- Storing sensitive transaction traces long after statutory legal retention obligations have elapsed.

7. Questions the Founder Must Answer

- "Do we have automated cron jobs or database lifecycle policies configured to destroy temporary file uploads and expired session tokens?"

8. Recommended Evidence or Documentation

- Formal Data Retention Schedule document detailing data categories, holding periods, and destruction mechanisms.

9. Priority & Suggested Next Action

- **Action:** Configure S3 bucket lifecycle rules and log management policies to automatically delete debug logs after 30–90 days.

10. Relevant Official Resources

- [ICO Guidance on Storage Limitation](#)

CHECK 09 — DATA PROCESSING AGREEMENTS (DPA)

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Executing legally binding contracts—Data Processing Agreements (DPAs)—between your legal entity (Data Controller) and any third-party tool, vendor, hosting provider, or API integrated into your stack (Data Processors).

2. Why It Matters

Under GDPR (Article 28) and Swiss FADP, utilizing third-party data processors without a formal DPA in place is an explicit legal violation. You remain responsible for security failures occurring within vendor systems unless proper contracts exist.

3. Potential Risks If Neglected

- Fines for contracting non-vetted third-party software processors.
- Total lack of legal recourse or liability recovery if a vendor suffers a major security breach affecting your user data.

4. Step-by-Step Review Checklist

- [] Identify all vendors processing personal data on your behalf (hosting providers, database services, transactional email platforms, analytics, AI APIs).
- [] Sign or digitally execute DPAs with every identified vendor.
- [] Verify that DPAs incorporate valid international data transfer mechanisms (EU/Swiss Standard Contractual Clauses - SCCs) for vendors based outside the EEA/Switzerland.
- [] Ensure vendor DPAs legally bind them to notify you of data breaches within tight timelines (e.g., 24 to 48 hours).
- [] Establish a vendor re-assessment policy for recurring annual security and compliance reviews.

5. Practical Implementation Examples

- **SaaS Application:** Execute AWS DPA, Stripe DPA, and Postmark DPA online within respective vendor administration dashboards.
- **Mobile Application:** Ensure mobile analytics SDK providers (e.g., Amplitude, AppsFlyer) offer signed DPAs covering mobile telemetry.
- **AI Tooling:** Sign enterprise data processing contracts with LLM providers (e.g., OpenAI, Anthropic) ensuring strict zero-data-retention options for enterprise user inputs.

6. Common Mistakes & Anti-Patterns

- Using standard consumer/self-serve terms for enterprise vendor tools handling production user PII.
- Assuming that accepting general Terms of Service automatically executes an underlying DPA.

7. Questions the Founder Must Answer

- "Do we have signed DPAs with Standard Contractual Clauses on file for every vendor that touches our user data?"

8. Recommended Evidence or Documentation

- A centralized compliance repository containing executed DPAs and signed Standard Contractual Clauses for all vendors.

9. Priority & Suggested Next Action

- **Action:** Download, sign, and archive DPAs for all primary cloud infrastructure and API providers used in production.

10. Relevant Official Resources

- [EU Standard Contractual Clauses \(SCCs\) Guidance](#)
-

CHECK 10 — THIRD-PARTY DISCLOSURES

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Maintaining total operational transparency by explicitly disclosing all third-party sub-processors, external APIs, hosting environments, and integration tools in your public-facing documentation.

2. Why It Matters

Users and business customers possess a statutory right to know who processes their data. Furthermore, claiming that an app is "100% Swiss Hosted" or "EU Cloud Secured" while passing payload data through non-EU cloud APIs constitutes false advertising and regulatory fraud.

3. Potential Risks If Neglected

- Regulatory enforcement for misrepresenting software data handling practices.
- Inability to pass enterprise procurement security reviews due to inaccurate vendor documentation.

4. Step-by-Step Review Checklist

- [] Publish a dedicated, accessible Subprocessor Directory page (e.g., /subprocessors).
- [] Disclose vendor name, precise service provided, processing location, and link to their privacy documentation.
- [] Implement a mechanism allowing enterprise customers to subscribe to notifications regarding sub-processor changes.
- [] Audit application code to verify no hidden external calls or undeclared SDKs exist.
- [] Debunk claims: Validate whether "Swiss Hosted" infrastructure truly avoids processing traffic through non-Swiss/EU API layers.

5. Practical Implementation Examples

- **SaaS Application:** Host a dynamic Subprocessor page detailing hosting regions (e.g., AWS Zurich), payment engine (Stripe US), and transactional email (Postmark US).
- **Mobile Application:** Update app store disclosure nutrition labels matching exact sub-processor data collection profiles.
- **AI Tooling:** Disclose explicit model infrastructure providers (e.g., Azure OpenAI Europe, AWS Bedrock Frankfurt, or self-hosted vLLM instances).

6. Common Mistakes & Anti-Patterns

- Advertising strict local data sovereignty while routing user inputs through US-based serverless proxy infrastructure.
- Forgetting to update sub-processor directories when switching production cloud services.

7. Questions the Founder Must Answer

- "Does our public sub-processor list accurately reflect every third-party service receiving application payload data today?"

8. Recommended Evidence or Documentation

- Public-facing Subprocessor Disclosure page with user notification subscription infrastructure.

9. Priority & Suggested Next Action

- **Action:** Audit all outbound network requests using application monitoring tooling to compile an accurate sub-processor list.

10. Relevant Official Resources

- [Swiss FDPIC Guidance on Transborder Data Flows](#)
-

CHECK 11 — DATA SECURITY

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Implementing technical and organizational measures (TOMs) to safeguard data confidentiality, integrity, and availability, anchored on proven cybersecurity standards like OWASP principles.

2. Why It Matters

Security breaches lead to catastrophic financial losses, legal liability, corporate reputation damage, and severe mandatory regulatory fines under security protection mandates (e.g., GDPR Art. 32).

3. Potential Risks If Neglected

- Total application compromise, database exposure, and data encryption by ransomware.
- Immediate loss of user trust, major legal claims, and severe statutory fines.

4. Step-by-Step Review Checklist

- [] **Secrets Management:** Enforce complete removal of hardcoded production secrets, API keys, and database credentials from code repositories.
- [] **Authentication & Authorization:** Enforce strong password policies, multi-factor authentication (MFA), robust session management, and strict access controls.
- [] **Row-Level Security (RLS):** Verify tenant isolation in database architectures to prevent cross-tenant data leaks.
- [] **Data Encryption:** Enforce TLS 1.3 for data in transit and AES-256 for data at rest.
- [] **Input Validation & Sanitization:** Defend against SQL Injection (SQLi), Cross-Site Scripting (XSS), and Remote Code Execution (RCE).
- [] **Rate Limiting:** Protect all public endpoints and sensitive APIs (e.g., `/login`, `/reset-password`) from brute-force and denial-of-service (DoS) attacks.
- [] **Dependency Scanning:** Conduct automated vulnerability scans across application packages and libraries.

5. Practical Implementation Examples

- **SaaS Application:** Mandate Row Level Security (RLS) policies inside Supabase/PostgreSQL verifying user identity token tenant parameters on every query.
- **Mobile Application:** Store access tokens securely within the iOS Keychain or Android Keystore, prohibiting local unencrypted storage.
- **AI Tooling:** Wrap prompt input fields with server-side validation and sanitization pipelines to stop basic prompt injection payloads.

6. Common Mistakes & Anti-Patterns

- Storing environment variable files (`.env`) in public or private Git repositories.

- Relying solely on client-side routing logic for authorization checks while leaving backend API endpoints exposed.
- Disabling CORS (Cross-Origin Resource Sharing) security mechanisms globally (`Access-Control-Allow-Origin: *`).

7. Questions the Founder Must Answer

- "If a malicious user modifies an ID parameter in an API request from `/api/user/101` to `/api/user/102`, will our backend block the unauthorized access?"

8. Recommended Evidence or Documentation

- Third-party penetration testing report, OWASP compliance review, and automated SAST/DAST security scan logs.

9. Priority & Suggested Next Action

- **Action:** Run an automated security scan (e.g., Snyk, GitHub Dependabot, OWASP ZAP) against all source code repositories.

10. Relevant Official Resources

- [OWASP Top Ten Security Risks](#)
 - [OWASP Application Security Verification Standard \(ASVS\)](#)
-

CHECK 12 — DATA BREACH PROCEDURE

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Developing, documenting, and testing a clear security incident response plan to identify, contain, evaluate, and report data breaches to regulators and affected individuals within strict statutory deadlines.

2. Why It Matters

When a security breach occurs, rapid containment and notification are critical. GDPR mandates notifying supervisory authorities within **72 hours** of becoming aware of a breach. The revised Swiss FADP mandates notifying the FDPIC **as soon as possible**.

3. Potential Risks If Neglected

- Compounded administrative fines specifically for failing to report a breach on time, distinct from the breach itself.
- Disastrous public relations crises and unmitigated secondary damage to compromised users.

4. Step-by-Step Review Checklist

- [] Establish a clear operational definition distinguishing a minor system bug/outage from a reportable Data Breach.
- [] Create a step-by-step Security Incident Containment Playbook.
- [] Designate an internal Incident Lead and assemble an incident team (Engineering, Legal, Communications).
- [] Prepare standard breach notification templates for regulatory authorities and affected users.
- [] Verify that system logging infrastructure captures sufficient forensic logs to determine breach scope.
- [] Perform a mock breach drill to test containment procedures and response speed.

5. Practical Implementation Examples

- **SaaS Application:** Maintain an incident alert channel integrated into PagerDuty that automatically notifies engineering leads upon anomalous bulk data export detection.
- **Mobile Application:** Build quick remote kill-switch mechanisms for compromised API keys embedded in mobile app versions.
- **AI Tooling:** Maintain exact timestamp records of prompt data traffic to determine whether compromised model memory logs contained PII.

6. Common Mistakes & Anti-Patterns

- Attempting to cover up a breach or delaying regulatory reporting past statutory deadlines to protect public relations.
- Overwriting forensic logs during emergency recovery efforts, destroying evidence needed to evaluate breach scope.

7. Questions the Founder Must Answer

- "If our database is compromised at 2:00 AM on a Saturday, who is alerted, and can we report to regulators within 72 hours?"

8. Recommended Evidence or Documentation

- Formally documented Incident Response Plan and completed Incident Communication Templates.

9. Priority & Suggested Next Action

- **Action:** Draft a 1-page Incident Response Protocol establishing immediate technical containment and notification steps.

10. Relevant Official Resources

- [EDPB Guidelines on Data Breach Notification](#)
-

CHECK 13 — INTELLECTUAL PROPERTY (IP)

[RECOMMENDED PRACTICE] | Priority: P1

1. What It Means

Securing, protecting, and formally documenting corporate ownership of all software codebases, designs, trademarks, brand assets, and proprietary technical algorithms.

2. Why It Matters

Without formal IP assignments from contractors, freelancers, agencies, or co-founders, your corporate entity may not actually own its core software. Unresolved IP ownership halts startup acquisitions, VC investment, and institutional financing rounds.

3. Potential Risks If Neglected

- Inability to close investment rounds due to failed legal due diligence.
- Costly copyright ownership lawsuits brought by former developers or contractors.

4. Step-by-Step Review Checklist

- [] Execute written IP Assignment Agreements (Work-for-Hire clauses) with all employees, freelancers, and external software agencies.
- [] Register company core trademarks (brand name, primary app logo) in target operating jurisdictions.
- [] Review font licenses, media assets, and commercial UI kits to verify full commercial authorization.
- [] Perform legal review of copyright and ownership rights for AI-assisted or AI-generated application code.
- [] Verify proprietary license terms for custom software features and code packages sold commercially.

5. Practical Implementation Examples

- **SaaS Application:** Ensure every developer contract contains explicit invention assignment language assigning all code copyright to the legal entity.
- **Mobile Application:** Audit custom graphic designs, app store screenshots, and UI audio assets for valid commercial licenses.
- **AI Tooling:** Verify that third-party generative asset pipelines do not utilize copyrighted source content that creates legal exposure.

6. Common Mistakes & Anti-Patterns

- Assuming paying an invoice to a freelance developer automatically transfers code copyright to your company without a signed agreement.
- Building core product features around assets downloaded under "Non-Commercial Use" licenses.

7. Questions the Founder Must Answer

- "Do we have a signed legal contract with every engineer who contributed code containing an explicit IP assignment clause?"

8. Recommended Evidence or Documentation

- Signed Founder IP Assignment Agreements and Contractor Work-for-Hire Contracts.

9. Priority & Suggested Next Action

- **Action:** Perform a audit of all historical code contributors to confirm fully executed IP assignment contracts exist for each contributor.

10. Relevant Official Resources

- [World Intellectual Property Organization \(WIPO\)](#)
 - [Swiss Federal Institute of Intellectual Property \(IPI\)](#)
-

CHECK 14 — OPEN-SOURCE LICENCES

[RECOMMENDED PRACTICE] | Priority: P2

1. What It Means

Auditing and managing all open-source software (OSS) packages, dependencies, and frameworks integrated into your codebase to ensure compliance with underlying open-source licenses.

2. Why It Matters

Integrating components governed by "Copyleft" licenses (e.g., GPL, AGPL) can legally force you to open-source your entire proprietary codebase. Conversely, permissive licenses (MIT, Apache 2.0, BSD) require proper copyright notices.

3. Potential Risks If Neglected

- Involuntary legal requirement to publish your proprietary source code under public open-source licenses.
- Copyright infringement lawsuits from open-source original authors and foundation maintainers.

4. Step-by-Step Review Checklist

- [] Distinguish between Permissive Licenses (MIT, Apache 2.0, BSD) and Strong Copyleft Licenses (GPL, AGPL).
- [] Scan all software dependencies for AGPL licenses used within server-side hosted environments.
- [] Generate a Software Bill of Materials (SBOM) listing all third-party dependencies.

- [] Maintain copyright attribution files for open-source libraries integrated into compiled client software binaries.
- [] Set up automated continuous integration tools to block installation of restrictive licenses.

5. Practical Implementation Examples

- **SaaS Application:** Configure GitHub Actions workflows using `license-checker` to fail builds if an AGPL package is imported.
- **Mobile Application:** Include an "Open-Source Licenses" attribution view within app settings listing all third-party library credits.
- **AI Tooling:** Verify licensing terms for open-weights AI models (e.g., Llama 3 commercial usage thresholds, Apache 2.0, strict non-commercial research limits).

6. Common Mistakes & Anti-Patterns

- Importing AGPL-licensed libraries into a commercial SaaS backend without recognizing it triggers full code disclosure requirements.
- Removing copyright headers and author attributions from imported open-source source files.

7. Questions the Founder Must Answer

- "Do we have any AGPL-licensed dependencies in our core SaaS repository?"

8. Recommended Evidence or Documentation

- Automated Software Bill of Materials (SBOM) report and license audit log.

9. Priority & Suggested Next Action

- **Action:** Run a license scan tool (e.g., FOSSA, `license-checker`) against your primary code repositories to identify license types.

10. Relevant Official Resources

- [Choose an Open Source License](#)
- [Open Source Initiative \(OSI\)](#)

CHECK 15 — PAYMENTS AND SUBSCRIPTIONS

[LEGAL REQUIREMENT – DEPENDS ON APPLICABLE LAW] | Priority: P1

1. What It Means

Structuring payment execution, subscription renewals, pricing displays, and cancellation mechanisms in compliance with legal consumer protection rules, card network guidelines (Visa/Mastercard), and gateway policies (Stripe).

2. Why It Matters

Regulators are heavily penalizing software companies using deceptive patterns ("Dark Patterns") to complicate subscription cancellations. Additionally, card network penalties and high chargeback ratios (>0.9%) will trigger payment processing account termination.

3. Potential Risks If Neglected

- Immediate termination of payment processing capabilities by gateways (Stripe, Adyen).
- Mandatory regulatory fines under consumer protection laws enforcement against deceptive subscription retention practices.

4. Step-by-Step Review Checklist

- [] Ensure transparent, clear pricing displays specifying currency, tax calculations (VAT/Sales Tax), and billing cycles.
- [] Implement simple, frictionless subscription cancellation mechanisms matching sign-up ease ("click-to-cancel").
- [] Comply with statutory consumer withdrawal rights ("cooling-off periods") where applicable, or obtain explicit consent waiving withdrawal rights upon immediate access.
- [] Provide automated email transaction receipts and pre-renewal notifications for annual plans.
- [] Deploy Strong Customer Authentication (SCA / 3D Secure) to satisfy anti-fraud and PSD2 regulations.

5. Practical Implementation Examples

- **SaaS Application:** Provide an immediate "Cancel Subscription" button within the billing dashboard that does not require contacting customer support.
- **Mobile Application:** Route all in-app digital subscriptions through Apple In-App Purchase or Google Play Billing systems according to platform mandates.
- **AI Tooling:** Clearly state token usage pricing scales, dynamic billing caps, and auto-refill triggers before charging user credit cards.

6. Common Mistakes & Anti-Patterns

- Forcing users to call phone support or submit manual contact forms to cancel a digital subscription activated online.
- Hiding recurring billing details in fine print during checkout.

7. Questions the Founder Must Answer

- "Can a user cancel their subscription in as few steps as it took them to sign up?"

8. Recommended Evidence or Documentation

- Documented checkout flow UI screenshots showing explicit pricing details, terms consent, and straightforward self-service cancellation paths.

9. Priority & Suggested Next Action

- **Action:** Audit the billing portal to confirm users can execute one-click subscription cancellations.

10. Relevant Official Resources

- [Stripe Legal & Compliance Guidance](#)
- [EU Consumer Rights Directive on Digital Content](#)

SECTION 4: The AI & "Vibe-Coding" Risk Review

"Before You Ship AI-Generated Code"

Modern development flows frequently leverage LLMs, AI agents, and IDE extensions (e.g., Cursor, Claude, Windsurf, GitHub Copilot, ChatGPT) alongside no-code/low-code builders. Rapid "vibe-coding" drastically accelerates development speed but regularly introduces hidden security vulnerabilities, structural flaws, and legal exposure.

Execute these **18 specific technical and legal checks** on all AI-assisted or auto-generated codebases prior to production release:

Risk Domain	Item Code	Verification Check Requirement	Priority	Status
Secrets Security ▾	AI-01	Verify no production API keys, secrets, or internal database URIs were embedded into AI prompts that could be used for external LLM model training.	P1	[] ▾
Secrets Security ▾	AI-02	Scan generated code to confirm zero hardcoded fallback credentials, test tokens, or private signing keys exist in source files.	P1	[] ▾
Data Privacy ▾	AI-03	Confirm zero user PII, customer payloads, or sensitive database records are routed to public AI APIs without an enterprise Zero-Data-Retention (ZDR) agreement.	P1	[] ▾

Risk Domain	Item Code	Verification Check Requirement	Priority	Status
Data Privacy ▾	AI-04	Verify vector database indexes (Pinecone, pgvector) enforce strict tenant isolation to prevent retrieval augmented generation (RAG) cross-tenant data leaks.	P1	[] ▾
Auth & Authorization ▾	AI-05	Audit AI-generated API route handlers to ensure explicit server-side authorization logic exists; verify endpoints do not rely solely on client-side routing.	P1	[] ▾
Auth & Authorization ▾	AI-06	Inspect database configuration scripts generated by AI to ensure Row Level Security (RLS) is explicitly enabled and thoroughly tested.	P1	[] ▾
Supply Chain Safety ▾	AI-07	Verify that every third-party package, dependency, and import recommended by AI models actually exists in public registries (prevent "Package Hallucination" attacks).	P1	[] ▾
Supply Chain Safety ▾	AI-08	Verify imported open-source dependencies suggested by AI adhere to permitted commercial license frameworks (e.g., no unintended AGPL imports).	P2	[] ▾
Input Validation ▾	AI-09	Confirm AI-written database interaction code utilizes parameterized queries (prepared statements) to prevent SQL injection vulnerabilities.	P1	[] ▾
Input Validation ▾	AI-10	Verify user input rendered in dynamic frontend views is properly sanitized to prevent Cross-Site Scripting (XSS).	P1	[] ▾

Risk Domain	Item Code	Verification Check Requirement	Priority	Status
Tool Calling / Agents ▾	AI-11	Restrict execution contexts for autonomous AI Agents or tool-calling routines; prohibit raw shell/system access without strict sandboxing.	P1	[] ▾
Tool Calling / Agents ▾	AI-12	Implement human-in-the-loop (HITL) authorization checks prior to allowing AI Agents to execute destructive operations (e.g., data deletion, fund transfers).	P1	[] ▾
Prompt Security ▾	AI-13	Implement input sanitization and system instruction framing to minimize vulnerability to Direct and Indirect Prompt Injection attacks.	P1	[] ▾
Prompt Security ▾	AI-14	Enforce output validation pipelines to prevent AI LLM outputs from rendering unvalidated raw HTML, Markdown, or executable scripts directly in browsers.	P1	[] ▾
Operational Reliability ▾	AI-15	Implement rate limiting, timeout controls, and backoff logic around all downstream AI model API connections to mitigate cost spikes and denial-of-service risks.	P1	[] ▾
Operational Reliability ▾	AI-16	Establish fallback error-handling logic for instances when third-party AI APIs experience elevated latency, context length errors, or service outages.	P2	[] ▾
IP & Copyright ▾	AI-17	Review copyright status and IP risks of generated code blocks; ensure code does not violate existing third-party open-source copyrights.	P2	[] ▾

Risk Domain	Item Code	Verification Check Requirement	Priority	Status
Model Disclaimers ▾	AI-18	Include clear UI disclaimers informing end users that AI outputs are machine-generated and require independent verification for accuracy.	P1	

SECTION 5: Pre-Launch Readiness Matrix

Use this fillable matrix to conduct your final pre-launch gatekeeping review.

Verification Item	Category	Priority	Status	Evidence / Location / Notes	Operational Owner	Target Date
01. Privacy Policy	Legal ▾	P1		Published at /privacy	Legal / Product ▾	Date
02. Terms & Conditions	Legal ▾	P1		Click-wrap enabled on signup	Legal / Ops ▾	Date
03. Privacy Law Alignment	Legal ▾	P1		ROPA documented	DPO ▾	Date
04. Cookie Governance	Legal/Tech ▾	P1		CMP active; scripts blocked prior to consent	Frontend Lead ▾	Date
05. Legal Basis Mapping	Legal ▾	P1		Data inventory completed	DPO / Legal ▾	Date
06. Marketing Consent	Legal/Ops ▾	P1		Double opt-in active; suppression list ready	Marketing Lead ▾	Date
07. Data Deletion (DSAR)	Legal/Tech ▾	P1		Self-service deletion & DB cascades verified	Backend Lead ▾	Date
08. Data Retention Schedule	Legal/Tech ▾	P2		Auto-purging DB cron jobs running	Data Eng ▾	Date
09. Signed DPAs	Legal ▾	P1		DPAs signed with cloud host & APIs	Legal / Procurement ▾	Date
10. Third-Party Disclosures	Legal/Tech ▾	P1		Subprocessor directory updated	DevOps ▾	Date

Verification Item	Category	Priority	Status	Evidence / Location / Notes	Operational Owner	Target Date
11. Application Security	Security ▾	P1	[] ▾	SAST scans passed; RLS enforced; no secrets in Git	Security Lead ▾	📅 Date
12. Incident Response Plan	Legal/Sec ▾	P1	[] ▾	Incident Response Protocol documented	SecOps / Legal ▾	📅 Date
13. Intellectual Property	Legal ▾	P1	[] ▾	IP assignment agreements signed for all code	Founder / Counsel ▾	📅 Date
14. Open-Source Licenses	Tech/Legal ▾	P2	[] ▾	SBOM generated; no AGPL violations	Lead Architect ▾	📅 Date
15. Payments & Cancel Flow	Legal/Ops ▾	P1	[] ▾	One-click cancel tested; clear billing text	Product / Finance ▾	📅 Date
AI Risk Check (18 Items)	Tech/Legal ▾	P1	[] ▾	Vibe-coding security audit completed	Lead Architect ▾	📅 Date

Final Launch Decision Governance Gate

- [] **NOT READY:** Unresolved P1 items remain. Production launch is **blocked**.
- [] **READY FOR REVIEW:** All P1 items resolved. Final legal and security sign-off pending.
- [] **READY FOR LAUNCH:** All P1 items verified; P2/P3 items scheduled. Production release approved.

SECTION 6: Learning Library & Resources

All references below connect to official regulatory bodies, security foundations, and legal standards frameworks:

- **European Commission Data Protection:**
https://commission.europa.eu/law/law-topic/data-protection/information-business-and-organisations_en
Official guidance on GDPR compliance, data protection principles, and business obligations. (Relates to: Checks 01, 03, 05, 07, 08, 09, 12)
- **Swiss Federal Data Protection and Information Commissioner (FDPIC):**
<https://www.edoeb.admin.ch/>

Official portal for Swiss data protection law, regulations, and administrative updates.
(Relates to: Checks 01, 03, 09, 10, 12)

- **FDPIC Privacy Statements Guidelines:**
<https://www.edoeb.admin.ch/en/privacy-statements-on-the-internet>
Official recommendations for drafting privacy policies under the Swiss FADP. (Relates to: Checks 01, 03, 10)
- **Choose an Open Source License:** <https://choosealicense.com/>
Non-technical guide to understanding open-source software license types and restrictions. (Relates to: Check 14)
- **Open Source Initiative (OSI):** <https://opensource.org/licenses>
Official body for open-source licensing definitions and standard license templates. (Relates to: Check 14)
- **OWASP Foundation:** <https://owasp.org/>
Open Web Application Security Project main repository for application security standards. (Relates to: Check 11)
- **OWASP Top 10 Web Application Security Risks:**
<https://owasp.org/www-project-top-ten/>
Standard awareness document for developers and web application security. (Relates to: Check 11)
- **OWASP Application Security Verification Standard (ASVS):**
<https://owasp.org/www-project-application-security-verification-standard/>
Detailed framework for testing web application technical security controls. (Relates to: Check 11)
- **GitHub Code Security Documentation:** <https://docs.github.com/en/code-security>
Documentation on automated secret scanning, Dependabot alerts, and static analysis. (Relates to: Checks 11, 14)
- **Open Source Security Foundation (OpenSSF):** <https://openssf.org/>
Cross-industry initiative focusing on open-source software supply chain security. (Relates to: Checks 11, 14)

SECTION 7: Practical Operational Templates

Template 1: Data Processing Inventory (ROPA Matrix)

Data Category	Purpose of Processing	Legal Basis	Recipient / Processors	Storage Location	Retention Period	Security Measures
User Account PII	User authentication & platform access	Contractual Necessity	AWS PostgreSQL	Zurich (EU-central-2)	Duration of active account + 30 days	Encrypted at rest (AES-256), RLS enabled

Data Category	Purpose of Processing	Legal Basis	Recipient / Processors	Storage Location	Retention Period	Security Measures
Billing Records	Payment processing & tax compliance	Legal Obligation	Stripe Inc.	United States (SCCs signed)	10 Years (Statutory tax law)	PCI-DSS Level 1 Gateway
Product Telemetry	Performance analytics & debugging	Explicit Consent	PostHog EU	Frankfurt (eu-central-1)	90 Days	Anonymized IP, transient cookies

Template 2: Third-Party Vendor Register						
Vendor Name	Service Provided	Data Shared	Processing Location	DPA Executed?	Transfer Mechanism	Security Verification
Amazon Web Services	Cloud Infrastructure	Full DB Payloads	Frankfurt / Zurich	YES	EU Standard Contractual Clauses	ISO 27001, SOC 2 Type II
Stripe, Inc.	Payment Gateway	Name, Billing, Card	United States	YES	Data Privacy Framework / SCCs	PCI-DSS Level 1 Compliant
OpenAI LLC	LLM API Inference	Anonymized Prompts	United States	YES	Zero-Data-Retention API Agreement	SOC 2 Type II Compliance

Template 3: Data Retention & Disposal Register						
Target Data Collection	Systems / Database	Holding Period	Expiry Trigger	Disposal Method	Responsible Role	
Unverified Accounts	Primary User DB	14 Days	Registration without verification	Automated database deletion cron job	Lead Backend Engineer	
Application Debug Logs	CloudWatch / Datadog	30 Days	Log creation date	Lifecycle automated purge policy	DevOps Lead	
Closed User Profiles	Core Relational DB	30 Days	User account deletion trigger	Hard delete cascade across tables	Lead Backend Engineer	

Template 4: Security Incident Response Log							
Incident ID	Discovery Timestamp	Incident Description	Classification Severity	Affected Data Scope	Regulatory Report Required?	Containment Status	Resolved Timestamp
INC-2028-01	YYYY-MM-DD HH:MM	Staging database port inadvertently left public	Medium (P2)	Zero production data impacted	NO	Firewall rule updated within 15 min	YYYY-MM-DD HH:MM

Incident ID	Discovery Timestamp	Incident Description	Classification Severity	Affected Data Scope	Regulatory Report Required?	Containment Status	Resolved Timestamp
INC-2026-02	YYYY-MM-DD HH:MM	S3 Bucket policy error exposing log archives	Critical (P1)	1,200 analytical access logs exposed	YES (FDPIIC/GDPR notified)	Bucket permissions reset; log files secured	YYYY-MM-DD HH:MM

--

Template 5: Open-Source Dependency & Licence Tracker					
Package / Library Name	Version	Open Source License	Copy/left Risk?	Usage Context	Approved for Production?
React	18.2.0	MIT License	NO	Client Frontend UI	YES
Pgvector	0.5.0	PostgreSQL License	NO	Vector Embeddings Store	YES
GPL-Parser-Lib	1.0.4	GNU General Public (GPL v3)	YES	Server Backend Logic	REJECTED (Replaced with ...)

--

Template 6: Pre-Launch Readiness Sign-Off Sheet

PRE-LAUNCH COMPLIANCE & SECURITY SIGN-OFF

Product / Platform Name:

Release Target Version:

Target Launch Date:

EXECUTIVE SIGN-OFFS:

1. Legal & Regulatory Compliance

Status: [] APPROVED [] REJECTED [] CONDITIONAL

Name & Title:

Signature: _____ Date:

2. Technical & Cybersecurity Lead

Status: [] APPROVED [] REJECTED [] CONDITIONAL

Name & Title:

Signature: _____ Date:

3. Founder / Chief Executive Officer

Status: ☐ APPROVED ☐ REJECTED ☐ CONDITIONAL

Name & Title:

Signature: _____ Date:

Operational Conditions / Mandatory Launch Blockers Reserved:

Template 7: Questions to Ask Your Tech/Corporate Lawyer

1. *"Based on our target user demographic and corporate jurisdiction, which privacy regulations (GDPR, Swiss FADP, CCPA) apply to our platform on Day 1?"*
2. *"Are our standard Limitation of Liability and Indemnification caps enforceable in our governing law jurisdiction for software outages?"*
3. *"Does our click-wrap registration mechanism legally bind users under recent case law?"*
4. *"Do our subprocessor agreements correctly utilize the latest European Commission Standard Contractual Clauses (SCCs) for transfers outside Europe?"*
5. *"Are our IP assignment contracts with external contractors fully binding under local labor and IP statutes?"*

Template 8: Questions to Ask Your Cybersecurity Consultant

1. *"Does our backend architecture enforce multi-tenant isolation at the database query layer (Row Level Security) preventing cross-tenant access?"*
2. *"Have we properly isolated and encrypted API keys, secrets, and database credentials outside source code repositories?"*
3. *"Are our public endpoints, authentication handlers, and payment integration routes properly rate-limited against brute-force attacks?"*

4. *"Does our vector database and RAG setup prevent prompt injection attacks and unauthorized data exposure?"*
 5. *"Can our infrastructure logging trace data access events effectively enough to conduct forensics following a security incident?"*
-

SECTION 8: The 30-Minute App Risk Audit

For busy founders and technical leads, complete this rapid diagnostic framework in 5-minute sprints:[00-05 min] Sprint 1: Data Ingestion & Consent Review

└─ Check browser DevTools network tab: Are tracking cookies/analytics firing before consent?

└─ Is the Privacy Policy link live, complete, and un-authenticated at /privacy?

[05-10 min] Sprint 2: Core Authentication & Authorization Audit

└─ Attempt to access protected API endpoints directly without authorization headers.

└─ Verify Row Level Security (RLS) or tenant isolation is active on core user database tables.

[10-15 min] Sprint 3: Code Repository & Secret Scanning

└─ Search repository git history for hardcoded passwords, AWS keys, or API credentials.

└─ Verify .env files and local configuration files are explicitly listed in .gitignore.

[15-20 min] Sprint 4: Vendor & Dependency Governance

└─ Review external third-party API dependencies: Are executed DPAs archived for all processors?

└─ Run a dependency scan to confirm no unvetted AGPL copyleft packages exist in the build.

[20-25 min] Sprint 5: Payment & Subscription Flow Validation

└─ Test the user subscription registration experience: Are pricing terms clear and transparent?

└─ Test cancellation mechanics: Can a user cancel their plan self-service within 2 clicks?

[25-30 min] Sprint 6: Incident Preparedness & Data Handling

└─ Can the engineering team execute complete user account deletion within 1 step?

└─ Is an emergency Incident Response contact list documented and accessible to technical leads?

The Five-Stage Action Cycle

Execute this operational loop continuously across product sprints:

\$\$\text{IDENTIFY} \rightarrow \text{REVIEW} \rightarrow \text{FIX} \rightarrow \text{DOCUMENT} \rightarrow \text{RECHECK}\$\$

1. **IDENTIFY:** Pinpoint new software integrations, tracking tags, database schemas, or AI APIs introduced in product updates.
2. **REVIEW:** Cross-reference new technical components against the 15 Essential Checks in this checklist.
3. **FIX:** Remediate identified gaps (e.g., add missing RLS rules, execute vendor DPAs, configure rate limits).
4. **DOCUMENT:** Update internal ROPA registers, subprocessor lists, and dependency tracking logs.
5. **RECHECK:** Re-run automated security scanners and manual verification workflows prior to production deployment.

Document Sign-Off & Solutions Engineering Notice

This technical and legal review resource is maintained by **SwiTech Solution**. We specialize in technical architecture, software engineering, and security governance for applications, SaaS platforms, and enterprise AI implementations.

For specialized assistance auditing your application architecture, executing security assessments, or designing scalable cloud infrastructure, contact our engineering team:

- **Official Portal:** <https://switech-solution.com>
- **Inquiries & Technical Reviews:** <https://switech-solution.com>
- **Edition:** 2026–2027 Comprehensive Release