Key	Type of note	Summary	Description
NA-1	Data quality and standards	The X-tee services of the solutions to be created must comply with the requirements set out in the RIA X-tee manual.	
NA-2	Data quality and standards	The standard software supplied with the solution must comply with RIK requirements.	If the software solution is not specified in the procurement, the latest stable versions of the software listed below must be used for the specific needs. Server operating systems: 1) Linux RedHat Enterprise/Rocky 2) Windows Databases: 1) Microsoft SQL 2) Postgre SQL 3) MariaDB Web server: 1) Nginx 2) Microsoft IIS 3) Apache As an application server: 1) Tomcat Programming language: 1) C 2) Java 3) Python If commercial software is supplied with the delivery, its licence must include at least 5 years of security updates.
NA-3	Data quality and standards	The application must be created in accordance with the requirements of the Estonian Information Security Standard.	The measures valid at the time of the procurement announcement shall be taken as a basis.
NA-4	Data quality and standards	The public sector shall develop software primarily as open source and publish the software under a free licence in accordance with the licence requirements.	Exceptions to this requirement may only be made if otherwise provided by law or in other justified cases agreed with the contracting authority. The most common free software licences used in RIK are EUPL, GNU GPL, and MIT. The choice of licence depends on the needs and obligations and must be agreed on with the contracting authority. The licence terms and conditions that must be complied with when using the licence are as follows: In the case of EUPL, the following is required:

			1) a copyright notice in the header (Copyright © <year> <author's name>, followed by the statement "Licensed under EUPL". In the case of GNU GPL, the following is required: 1) copyright notice in the header (Copyright © <year> <author's name>); 2) a notice in the licence terms regarding the exclusion of warranty for the work. For MIT, the following is also required: 1) a copyright notice in the header (Copyright © <year> <author's name>) together with the notice specified in the licence; 2) a notice regarding the exclusion of warranty for the work as specified in the licence terms. For more details, please refer to the licence terms and conditions of the selected licence type. The licence terms of the selected licence shall be presented in one or both of the following ways: 1) The LICENCE file must be made public in the repository together with the software code; 2) the text of the licence terms and conditions in the header of each file; 3) by including a link in the header to the location where the licence conditions can be found.
NA-5	Architecture	Components must be such that their end of life (EOL) is not known to be less than 2 years.	Exceptions must be agreed to separately with the Strategy Team.
NA-6	Architecture	The components and topology of information systems must be agreed with RIK before the start of actual development.	Coordination is carried out by the Strategy Team (chief architect).
NA-7	Architecture	The application server must allow operation on a separate server from the database server.	
NA-8	Architecture	The application must be able to be moved between different domains and domain sites without reprogramming.	The solution must not contain any compiled absolute URLs.
NA-9	Architecture	External interfaces must be kept to a minimum.	Interfaces must be configurable. Error conditions for external interfaces must be handled. The system must function without non-business-critical interfaces.

NA-10	Architecture	Data exchange with databases belonging to the state information system and between databases belonging to the state information system takes place through the state information system data	Clause 43 (9) 5) of the Public Information Act. If an X-road query is performed by a person, the query header must contain the authenticated user data.
NA-11	Architecture	Application environments must use the endpoints of the corresponding X-tee security servers.	Development against development endpoint, etc.
NA-12	Architecture	The application must be configurable from a single location without the need for compilation.	If necessary, the configuration can be automatically copied without modification from a central location to another location (e.g. copying from the values file in the helm chart to several configmapi files). Logging settings can be stored separately from the application configuration file in an additional configuration file (e.g. Log4net).
NA-13	Architecture	The installation of the compiled application must take place within a reasonable time.	A reasonable time is up to 1 minute.
NA-14	Architecture	The application must be 64-bit.	
NA-15	Architecture	The database and application must use UTF-8 encoding.	This requirement applies to Oracle and Postgre databases. Exceptions will be agreed on separately with the Strategy Team. For example, UTF-16.
NA-16	Architecture	Files must be catalogued by year > month > date.	A more detailed solution must be agreed on with the Strategy Team.
NA-17	Architecture	The general programming paradigm is object-oriented.	Exceptions must be agreed on separately with the Strategy Team.
NA-18	Architecture	All foreign keys in base tables must be indexed.	Indexes and other measures are used to improve database performance. Foreign keys must also be used when referencing from one database to another.
NA-19	Architecture	Query variables (Parameter Binding) must be used.	When calling SQL queries from outside the database, query variables must be used to avoid SQL cache fragmentation.
NA-20	Architecture	Database tables must have a technical primary key.	The name must be "ID".
NA-21	Architecture	Files and file indexes must be replicable to another server space.	The location and logic of file storage shall be as agreed.
NA-22	Architecture	There must be an appropriate management interface for performing management operations.	The aim is to reduce the number of operations performed directly in the database.
NA-23	Architecture	The database must support cold and hot backup to another server room.	Services that prevent database mirroring (e.g. "failstream") must not be used.
NA-24	Architecture	The sorting rules must comply with the Estonian alphabet.	Case-insensitive, accent-sensitive sorting must be used.

NA-25	Architecture	The RIK e-mail server must be used.	The sending of messages and the template must be configurable. If the e-mail server does not accept messages, they must be resent once the e-mail service is restored.
NA-26	Architecture	The names of configuration parameters must be meaningful.	For example: X-tee Security Server, not XTTS or reference number, not vk_seb, etc.
NA-27	Architecture	The front-end and back-end systems must be clearly separated in terms of architecture.	Front-end and back-end systems must be separately installable and configurable.
NA-28	Architecture	Configuration files must be protected files by default.	For example, IIS: *.config, *.resources Apache: *.conf, .htaccess. If there are several of these, the developer must list them separately in the configuration file list.
NA-29	Architecture	The end user only sees the files they are supposed to see.	
NA-30	Architecture	The configuration must not contain duplicate parameters.	All parameters should be described only once in the configuration. For example, the following is not allowed: **connectionString**= "Server=myServerAddress;Database=myDataBase;User Id=myUsername;Password=myPassword;" **_cString**= "Server=myServerAddress;Database=myDataBase;User Id=myUsername;Password=myPassword;"
NA-31	Architecture	Applications must be highly available.	The applications we develop and the off-the-shelf products we use must be highly available. *Sticky sessions* are not recommended; the need for them must be discussed separately.
NA-32	Architecture	System integrations must be hidden from the client application.	For example, the client application must not access the database and X-tee directly.
NA-33	Architecture	Environment-specific variables must be configurable.	For example, WSDL must not contain references to development servers.
NA-34	Architecture	Windows service names must be configurable.	
NA-35	Architecture	The database must not implement the business logic of the application.	The database may only perform technical operations on incoming data. Except for background work. For example, calculating rights or generating unique keys.
NA-36	Architecture	The database must be convertible to MS-SQL standards.	Platform-specific solutions must not be used. Exceptions must be agreed on separately with the strategy team.

NA-37	Architecture	The application must use an OpenID-based authentication solution approved by RIK for authentication.	Authentication methods must be configurable. It must also be possible to specify in the application configuration whether OCSP or revocation list-based authentication is used for ID card authentication.
NA-38	Architecture	The length of the uniform resource identifier (URI) must not exceed the maximum value supported by any browser supported by the IS.	Max URI < 2000.
NA-39	Architecture	The application service description must be structured in such a way that it supports service versioning.	The value "any" for possible complexType versions in the service description
NA-40	Architecture	The application's operational base must be archivable.	This is usually done in parts, for example, legally expired procedures that are no longer visible to users.
NA-41	Architecture	Licences supplied with the application must be valid for at least 5 years.	In the case of EU projects, the validity period must be based on EU or RIA requirements.
NA-42	Architecture	The application must be divided into logical technical parts.	The logical division is based on business needs, administration and development. The technical parts must be separately installable.
NA-43	Architecture	File conversions must be performed using services approved by RIK.	
NA-44	Architecture	The user must not have access to technical information about the system.	For example, stack traces, technical logs, full file names, technologies and frameworks used, and their versions.
NA-45	Architecture	The application must be stateless.	Must work with a load balancer, does not use sticky sessions, SSL offload.
NA-46	Architecture	Application files must have read-execute access.	In the case of containers, the developer is responsible; otherwise, the administrator is responsible.
NA-47	Architecture	Windows servers must run on Windows Core servers.	
NA-48	Architecture	The most appropriate data type must be specified as the column type in the database.	Refer to the database engine documentation. The use of (max) types is prohibited unless justified and necessary.
NA-49	Architecture	A technical specification must be published for application machine interfaces.	For example, SOAP WSDL and REST OpenApi Swagger must not be published. The technical specification must include: endpoint description, endpoint description exclusion. Does not apply to out of the box products.
NA-50	Architecture	When using Active Directory (AD) authentication, the application must use valid standards.	SAML2.0 (Security Assertion Markup Language) and ADFS (Active Directory Federation Services).

NA-51	Architecture	When authenticating with an ID card, the client-side web application communicates with the web application only for the purpose of verification.	Further code execution and processing of possible error situations must take place strictly on the client side.
			The validity and authenticity of the client authentication certificate is checked to the maximum extent possible by the web server or proxy.
			The authentication certificate must contain the following information, as required by the web server or proxy:
			Apache: SSLVerifyClient require;
			NGINX: ssl_verify_client on;
			Pulse TM: ssl.requireCert();
			HAProxy: verify required;
			Tomcat: clientAuth="true";
			etc.
			If a request is made to a URL that requires a client authentication certificate and the server responds with an error, the client application must display the correct error message in Estonian.

NA-52	Architecture	The use and selection of robot traps must be coordinated with RIK.	It is recommended to avoid robot blocks and solve potential problems using load balancers and IP-based rules.
NA-53	Architecture	Inter-application communication must be machine-processable.	In general, we use REST, SOAP(X-tee) or message queues. Other cases should be discussed separately with the Strategy Team.
NA-54	Architecture	Application server endpoints must be limited and documented.	The application only responds to permitted HTTP methods; all others return an HTTP 405 error.
NA-55	Architecture	The application server must validate and, if possible, verify email addresses.	Must comply with RFC5322 and/or RFC6854 standards. Where possible, the system must verify the user's email addresses.
NA-56	Architecture	The application must have a minimum CSP header to ensure functionality.	Exceptions must be carefully considered. Low value "self" Additional information: https://csp-evaluator.withgoogle.com/ https://www.hardenize.com/dashboards https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Sec-WebSocket-Accept
NA-57	Information security	Communication between the client and server must use the TLS protocol.	Latest or valid version.
NA-58	Information security	When storing data on the client's computer, use the browser vault.	An exception is the language selection in multilingual systems.
NA-59	Information security	When a session ends or expires, the user must not be able to refresh or reuse the session.	

NA-60	Information security	Database entries with integrity security class 2 or 3 must be versioned.	When versioning, old records must be retained in their original form. The technical fields of a new record must include: the user who created the record and the time of creation. A record that has been declared invalid must include: the person who modified the record and the time of modification/deletion.
NA-61	Information security	Live data shall not be used for testing.	Synthetic, generated data is used for testing. Exceptions must be agreed separately with the Strategy Team.
NA-62	Information security	Application database accounts must have minimum rights.	The application does not use *schema* accounts. The rights required for accounts must be described in the application installation manual.
NA-63	Information security	End users of the application must have rights defined through roles (RBAC).	Use KeyCloak. AD OU must not be assigned a role.
NA-64	Information security	Only agreed and documented authentication procedures must be used to access the application and database.	
NA-65	Information security	Passwords must be stored securely.	Encryption must be in CBC, CRT or similar mode. ECB mode must not be used. For salt+pepper, follow the OWASP guidelines at https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html
NA-68	Information security	All public web applications must use web firewalls.	For IIS, use URL scan, for Apache, use modsecurity or a similar tool. Any restrictions must be agreed with the client during the detailed analysis based on the client's needs and requirements. The whitelisting principle must be used, not blacklisting. In addition, a web firewall (CloudFlare) is used.
NA-69	Information security	After successful login, the application must display the time of the previous successful login.	EU-direktiiv https://www.etsi.org/deliver/etsi_en/301500_301599/301549/03.02.01_60/en_301549v030201p.pdf
NA-70	Information security	Applications with a user interface must display the application version on the home page.	This applies to applications developed in-house. The opposite requirement applies to application servers and databases: NA-44.
NA-71	Information security	Sessions must be terminated on the server side and all applications must have a configurable user session timeout.	The time must be changeable along with other configuration parameters. If no requests have been received from the client within a specified time, the session must be terminated on the server's initiative.

			For more information, see https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html
NA-72	Information security	All input data must be encoded, filtered and validated.	This includes forms, machine-to-machine interfaces, and web services. This must be done before any business logic is executed.
NA-73	Information security	Forms sent by web-based applications must contain a hidden unique token that is checked when the form is received.	The purpose is to prevent CSRF attacks.
NA-74	Information security	Encryption and hashing must be based on the RIA study.	Refer to the latest document in this list: https://www.id.ee/artikkel/kruptograafiliste-algoritmide-elutsukli-uuringud-2/
NA-75	Information security	Session identifiers must not be copyable from the URL.	https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html
NA-76	Information security	Configuration files can only be changed by administrators.	If the application has a developed admin interface, changes can be made there. Other exceptions should be discussed with the strategy team.
NA-77	Information security	When authenticating with a client certificate (e.g. ID card) on the application side, the application must accept the certificate in the HTTP header.	When authenticating with a client certificate, the system should accept the client certificate in the agreed header, and the header name should be changeable in the application settings. This requirement arises from the GLB-level management of client certificate trust lists. The certificate is accepted by the client's web browser and forwarded to GLB. The GLB determines its suitability for the agreed header. The GLB forwards the already verified certificate to the system in the agreed header. The same applies to authentication with any client certificate, and the system must not accept certificates from any other location.
NA-78	Information security	All applications requiring digital signing must use the RIK digital signing service.	
NA-79	Information security	OWASP best practices must be followed to ensure application security.	The web application must pass an OWASP ASVS-based test without any problems. The initial external security testing shall be ordered at the expense of the contracting authority. If the test reveals errors caused by the developer's actions or inaction, the client may demand compensation from the developer for OWASP follow-up tests. Additional information is available at https://github.com/OWASP/ASVS

NA-80	Architecture	The application must be optimised for operation in the production environment.	The production environment application must not contain any unnecessary components. NT. Debug log, deviations from business logic necessary for testing. It must be possible to deliver the application to a freely selected environment (development, testing) without repackaging.
NA-81	Information security	When using time stamps, the solution approved by RIK is preferred.	Information about approved solutions is provided by the OPS team.
NA-82	Information security	Applications must support the use of SSO.	The session must be stateless
NA-83	Information security	The security class assigned to the system must be fulfilled in accordance with paragraph 10 of the Regulation "Cyber security requirements for network and information systems".	https://www.riigiteataja.ee/akt/113122022030?leia Kehtiv Before development begins, the requirements arising from the security class must be coordinated with the Information Security and IS Management Team.
NA-86	Information security	The thick client must encrypt temporary files containing sensitive/confidential data and delete them when they are closed.	If the thick client uses temporary files, their periodic deletion must be ensured to avoid overloading the user's computer. The aim is to ensure that no information that should not remain on the user's computer is left behind when the application is closed.
NA-87	Information security	The application must delete all temporary files from the server immediately when they are no longer in use.	The temporary file folder must be configurable.
NA-88	Information security	The application may only accept session keys issued by itself.	
NA-89	Information security	Files uploaded to the application must be filtered, validated and scanned for viruses.	The file type and extension must be checked for compatibility. The upper limit for file uploads must be specified in the analysis. Filtering – whitelisting, including size Validation - business logic compliance
NA-90	Information security	The file uploaded to the application must be saved with a unique generated name.	The original name of the file must be saved in the database.
NA-91	Information security	The application must not allow itself to be used within an iframe.	Use the CSP header. https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Sec-Fetch-Mode https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options

NA-92	Information security	Session cookies must have security flags and the prefix __Host.	The session cookie must include the Secure, HttpOnly and SameSite flags. The cookie name must be prefixed with "__Host-". Additional information: 1)https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie 2) https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/06-Session_Management_Testing/02-Testing_for_Cookies_Attributes 3) https://owasp.org/www-community/HttpOnly 4) https://owasp.org/www-community/controls/SecureCookieAttribute 5) https://owasp.org/www-community/SameSite#:~:text=SameSite%20prevents%20the%20browser%20from,none%20%2C%20lax%20%2C%20or%20strict%20.
NA-93	Source	All transferable versions of the application must be tested before being transferred to the client.	The test plan and scope must be agreed with the client during development.
NA-94	Logging and monitoring	Critical events – session start and end and role changes must be logged separately in a security log table.	For external applications, the user's IP address must be logged. Session key values, private keys, user passwords and other information that allows the user to be identified or that poses a threat to data protection must not be stored in the log.
NA-95	Logging and monitoring	The application must log all requests exchanged with external systems (including those moving through X-tee services) in the data exchange log.	The best tool for this is Graylog, the second choice is a database, and the third choice is file logging. It must be possible to enable or disable logging for each external system. The log must be structured so that queries and responses are in separate files. The location of the log file must be configurable by the administrator without recompiling the application.
NA-98	Logging and monitoring	The application must log all technical errors that occur in the application.	The log must contain at least the time (timestamp), error code, error content (component, stack trace, traceback, etc.), the entire HTTP query and, if possible, the user ID. Logging must be configurable without restarting the application.

NA-99	Logging and monitoring	When logging to a file system, logs must be catalogued, have a recognised file extension and be rotated.	It must be possible to enable or disable logging for external systems. No more than 10,000 files may be created in the file system folder. It must be ensured that each application server can write logs to its own log file if necessary. Logging must be configurable without restarting the application.
NA-100	Logging and monitoring	Application logs must be in a uniform format, machine-readable and complete.	Log fields that can be manipulated by the end user (IP, user agent, URL) must be stored in the log in an encoded and cleaned form. Each log entry must have a unique identifier for the query. If the value of a parameter is empty, it must be marked in the log with the replacement value "-". The date and time format of the log must comply with the ISO 8601 standard and be in the Estonian time zone.
NA-101	Logging and monitoring	Log tables must be archivable from the operational database.	As the table grows, it must be possible to keep old entries, for example by month or year, in separate tables or in another database. The mechanism must also work for encrypted logs.
NA-102	Logging and monitoring	If the application uses or offers external services, records must be kept of these.	A statistics module or OpenTelemetry must be used.
NA-103	Source	The system must undergo performance testing before going into production.	A detailed description must be agreed upon during the detailed analysis. Based on this -&gt; https://dok.rik.ee/pages/viewpage.action?pageId=244417515
NA-104	Logging and monitoring	The application must have a machine-readable status page.	
NA-106	Source	All comments must be justified.	Comments in the source code must be: 1. The code must be written in such a way that it is readable without comments. Comments are intended for complex and/or areas requiring adjustment and/or further work (the latter should be marked TODO). 2. Up-to-date – comments and code must correspond with each other. 3. Clear and unambiguous. 4. Correctly written - grammar and sentence structure must be correct. 5. Database files and other codes on the same basis

NA-107	Source	Names in the code must be meaningful and provide clear information about their purpose.	The project scope must include agreed naming rules.
NA-108	Source	Constants used in the code must be defined as variables.	This does not include classifiers. These must be in the base.
NA-109	Source	Data types defined in the code must be in the nominative singular. All data arrays must be named in the nominative plural.	For example, "Person", "Procedure", etc. Special characters must not be used in databases. Additional information: 1) C# - https://learn.microsoft.com/en-us/dotnet/standard/design-guidelines/general-naming-conventions 2) JAVA - https://www.oracle.com/java/technologies/javase/codeconventions-namingconventions.html 3) Python - https://peps.python.org/pep-0008/
NA-110	Source	Foreign keys contained in data tables must be linked by name to the table and field to which they refer.	For example, if there are tables 'Person' and 'Car', then the relationship 'person_car' would be: Person.ID=Car.Person_ID
NA-111	Source	The lengths of database fields must be based on the requirements described in the analysis.	Keep in mind that 1 byte may not equal 1 character, for example, special characters.
NA-115	Source	We use SonarQube rules to validate the code.	Trivy scanning must also be performed together with SonarQube.
NA-118	Source	The source code must not contain any unused parts.	
NA-120	Source	Business terms in the code base must be in Estonian.	The business analysis and code must reflect each other. The names in the database must also be in Estonian.
NA-121	Data quality and standards	When processing address data, follow the regulation "Address Data System."	Use ADS. Link to the regulation https://www.riigiteataja.ee/akt/128122024043
NA-123	Information security	The user's rights and role must be checked for each transaction.	OWASP requirements -&gt; "Validate the Permissions on Every Request" https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html#validate-the-permissions-on-every-request
NA-124	Data quality and standards	When handling areas of activity, follow the regulation "Classification System."	Use the RIK EMTAK service. https://www.riigiteataja.ee/akt/12910889
NA-126	Information security	Error messages issued by the system must not contain system information.	For example: "A technical error has occurred. Query ID: XXXXX". The actual error message must be retained in the log!
NA-127	Document	All application documentation must be written in Estonian.	Third-party components are an exception.
NA-128	Document	The documentation must comply with the requirements of the RIK documentation plan.	
NA-129	Document	Each new version is accompanied by a description of the changes.	Changelog or release notes.

NA-130	Versioning	All software packages provided for testing, training or implementation of the application must be located in the RIK code repository.	The developer is granted the necessary rights in the RIK code repository for this purpose.
NA-133	Versioning	The RIK ticket management environment must be used.	JIRA is used. External parties are included.
NA-137	Versioning	Third-party libraries must be located in the RIK repository.	
NA-138	Versioning	The content of database scripts must be verifiable.	The administrator must be able to verify the content of the scripts.