Annex 6

6. Core principles

Core principles target all layers of software and infrastructure components, aimed to ensure scalability, reusability, interoperability, and modularity of developed solutions. Key components to be considered are as follows:

1. **Intuitive design and usability**
2. **Secure by design**
    a. Security
    b. Incident management
    c. Disaster recovery
3. **Maintainability, modularity, and reusability**
4. **Interoperability**
5. **Efficiency**
6. **Codebase**
    a. Version control
    b. Documentation
    c. Branching strategy
    d. Implementation of CI/CD pipelines
7. **Testing**
8. **Future proofing**

6.1.1 Intuitive design and usability

General requirements: In general, the user experience design process shall follow a design thinking approach, consisting of the following five phases:

1. Empathizing with the users - learning about the users
2. Defining the problem - identifying the users' needs
3. Ideating - generating ideas for design
4. Prototyping - turning ideas into concrete examples
5. Testing - evaluating the design.

An essential condition for achieving best possible results is to ensure quick and fast prototyping and testing capabilities, the results of which will be required to be approved by the CPC.

UX related minimum requirements: User interface of the application shall be designed in a way that will be self-explanatory, without requiring any special training. Minimum requirements include:

- **Usability metrics:** Software usability metrics shall be defined, measured and changes performed to address any low performing metrics.
- **Consistency**: Design of the user interface shall be consistent in its all components.
- **Navigation:** Content should be displayed in a clear, unambiguous, easy to read manner and must ensure easy navigation.

- **Responsive design:** The interface should be responsive for ensuring the best possible experience on both desktop and mobile platforms.
- **Fonts:** Fonts used shall comply with 8-bit Unicode Transformation Format (UTF-8) encoding. Font size should ensure convenience of text perception with the minimum permissible resolution.
- **Language selection:** It should be possible to change the interface language at any time. Armenian and English languages shall be the minimum supported options.
- **Accessibility:** The system shall be developed following the principles and guidelines of Web Content Accessibility Guidelines (WCAG 2.1). A conformance level of AA will be required.
- **Information format:** Date formats, names, calendars, and similar basic information must match the standards presently used by other state information systems.
- **Measurement units:** The metric system should be used for presenting measurement units.
- **Website content:** Standardized schemas, such as those available on schema.org, shall be used for developing structured data, which will be added to the content of each page using JSON-LD (JavaScript Object Notation for Linked Data). This shall be done to ensure search engine optimization capabilities, making it easier to search for website content, including published reports.

6.1.2 Secure by design

General requirements: The software application shall consider security and privacy of data and users to be of highest priority and ensure that all relevant security tactics and patterns are implemented.

Security controls shall be designed with the information security triad, *confidentiality*, *security*, and *availability*, in mind, and with a clear understanding of the value of assets which need protection.

All code shall be free of viruses, malware, or backdoors. The code shall not contain tracking pixels or any tracking technology that is not specifically documented. Applicable secure coding practices shall be exercised, including, but not limited to:

- Input validation
- Keeping the design as simple as possible
- Compiling code using the highest warning levels and addressing those warnings wherever reasonably possible. Static and dynamic analysis tools shall be used to detect and eliminate potential security flaws.
- Considering access decisions based on permissions (whitelisting), and adhering to the principle of least privilege
- Data sanitization
- Embedding security in the architecture, such as by designing the software in a way to implement and enforce security policies
- Adopting secure coding standards
- Practicing defense in depth, by implementing multiple layers of protection
- Adopting effective quality assurance mechanisms.

Security planning, incident management, disaster recovery and risk management procedures shall be established in accordance with best practices and well-known standards, such as the latest version of ISO/IEC 27002, ITIL, CIS ver. 7.1. Relevant security controls shall be selected, implemented, and maintained.

Security related minimum technical requirements: As follows are a set of security related minimum requirements needed to be implemented:

- **System interaction protocol:** HTTPS protocol (TLS 1.3) should be used for all communication and any kind of user interactions with the system and other information environments.
- **Authentication:** Multifactor authentication mechanisms should be enforced using a combination of at least two of the following:
    - Username / password,
    - ID card issued by the RA Police/Mobile ID Card
    - One Time Password (OTP).

  Passwords must be stored and encrypted using Argon2 hash function (or similar), reducing the risks resulting from potential breaches of the password database.
- **Authorization:** The system should implement Role Based Access Control (RBAC) ensuring actions are restricted unless they are explicitly assigned to the user. Users shall be able to self-register, change/reset passwords, and deactivate accounts. A set of predefined roles shall be included based on the organizational chart of the CPC. There should be the possibility of creating new roles or modifying existing ones by adding/excluding permissions by authorized users only and operations which shall undergo a supervisory approval stage shall be identified and designed accordingly. The authorization engine shall be designed in a flexible way, allowing assignment of supervisory approval stages, as necessary, without requiring a vendor`s intervention.
- **Protection of sensitive data:** Sensitive data, including personally identifiable information (PII) shall be stored and encrypted using AES-256 or equivalent algorithm. The system shall ensure the protection of personal data in accordance with applicable rules and the requirements of the RA Law on Protection of Personal Data.
- **Data security:** The system shall have proper measures implemented and be tested against the latest OWASP Top Ten Web Application Security Risks and latest CWE Top 25.
- **Data integrity:** The system shall not have any right to modify the data inserted by users, and that shall be safeguarded using hash functions, which will be calculated for each form as soon as that form is submitted by the user. Any further modifications by the CPC, or any other entity, shall go through proper approval stages (based on the CPC's internal business processes) and shall record a complete and easy to analyze audit log.
- **Logs:** The system shall support different logging levels for all nodes and components which should be stored in the event log file, or a database. A well-known logging framework suitable for such applications shall be used to ensure a consistent approach. Any unauthorized attempt to edit data shall be logged with further possibility to be subjected to audit. Logs of users' interactions with the system, system internal interactions, system interactions with the underlying platform and external systems all shall be recorded, maintained and be easily

retrievable/exportable by applying filters in a format which will be easy to analyze. It is essential for all cases that an accurate, complete audit trail be recorded and retrievable, as needed. Integrity of all recorded audit trails shall be guaranteed by eliminating the possibility to delete/modify them by any user (e.g. through the storage of such trails in a separate, secure database which is managed by other personnel).

- **User session:** Minimum and maximum session duration must be set in minutes and shall be modifiable.
- **Data integrity:** Tools shall be implemented to enable checking database, and files' integrity. The system shall guarantee data integrity, non-repudiation, and accessibility and prevent any altering, damage, and unauthorized access to the system data. Data entered into the system may not have been edited, damaged, or deleted without authorization.
- **Availability:** The system shall guarantee full data storage on predefined, configurable periods, which shall not exceed Recovery Point Objective (RPO) and Recovery Time Objective (RTO), as defined by the CPC.

To ensure continuous development and maintain a secure environment, it is necessary to consider a regular information system audit, which will include assessment of effectiveness and efficiency of designed and implemented controls and penetration tests.

6.1.3 Maintainability, modularity, and reusability

Overall maintainability shall be ensured by considering the following concepts: modularity, understandability, changeability, testability, reusability. The core requirement here is to follow architecture and design best practices which will allow adding future building blocks (modules) on the developed application with minimum effort, and without affecting the system`s overall complexity and performance.

To ensure maintainability, it is essential to also follow best practices in source code, API, software use and system installation documentations and administration guides.

6.1.4. Interoperability

The current system being used by the CPC, which has connection to several data points (details provided in annex 4), does not utilize the full potential of available data in state databases. To address this and in aim of utilizing the whole potential of data already stored in different state databases, it is vital for the new system to be connected to the RA Government Interoperability Platform (GIP), and other databases which are not currently a part of GIP.

GIP is a centralized data store enabling data sharing from various state databases (full list of databases currently connected to the platform can be found in annex 5, and the CATIS catalogue which is a lists majority of state databases is presented in annex 6), which will serve as a data extraction point for the CPC`s system.

The system`s overall architecture and implementation shall be compliant with RA Government`s interoperability requirements1. This is required to ensure compatibility of the system with the GIP, so that it will be capable of being connected to the government interoperability platform (GIP) and extract data from connected databases and enable two-way data sharing if/when necessary.

Corresponding adaptors (described in section 6.7.) should be developed for interconnection with the GIP and for all the external databases, which are not connected to the GIP, but are necessary for the operations of the system.

6.1.5. Efficiency

The system shall ensure efficient use of computing resources (memory, cache, CPU, disk space) and network connections, either deployed in the cloud, or on premise. In this aspect, data access performance, data management, and compliance with best practices in coding for the used database management system shall also be considered.

6.1.6. Codebase

Version control: A distributed version-control system (recommended DVCS is Git) should be used for tracking changes in source code during software development.

Documentation: Source code shall use consistent, standard, well-known coding convention/s depending on programming language/s used.

All code shall be well-documented and written according to well-accepted style guides and should contain inline documentation (self-explanatory, well-structured comments), with suggestive names for variables.

Branching strategy: The branching strategy should be chosen in a way that will ensure easy collaboration among internal/external stakeholders, and shall ensure:

- Availability of releasable code disregarding the features under development, allowing quick releases and prototyping. This should also be supported by implementation of branching policies, ensuring any merges to master (release) branch will trigger notifications also to the CPC responsible person (more details in change management section).
- It should provide possibility to switch to an older releasable version if necessary (e.g. identification of a critical bug in the latest released version).

CI/CD pipelines: Continuous integration / Continuous delivery pipelines should be used to automate the software delivery process, minimize manual errors, provide standardized development feedback loops, and enable fast product iterations. In such an approach, infrastructure shall also be maintained as code (IaC), enabling management of IT infrastructure using configuration files which also follow similar versioning and branching strategies as deployed for application source code.

[1] RA Government Decree, N 1093-Ն, 31 August 2015 – Link: https://www.arlis.am/DocumentView.aspx?docid=128039

Escrow: To further manage software maintenance and mitigate the risk of abandonment, or orphaning of software, it is necessary to have a source code escrow agreement, enabling the CPC to own the source code if, and due to any reason, the vendor is unable to continue its operations.

All code used in the software should be free from any third-party claims and shall be owned solely and exclusively by the CPC.

6.1.7. Testing

Unit, integration, and system tests: Standardized, well-known testing frameworks shall be utilized to ensure better maintainability of the codebase and relatively effortless identification of potential bugs and errors. It is essential that important parts of the code be covered by unit tests. Meanwhile, the overall coverage shall not be less than 90% for unit tests, 80% for integration tests and 70% for system tests.

Tests for covering business cases: All business cases supported by the application should have corresponding test cases developed, clearly indicating inputs and outcomes. Acceptance testing of software shall be performed based on developed cases, in cooperation with the vendor.

Acceptance testing (performed by the CPC with the vendor’s support) shall also include performance and stress testing, which will be performed in cooperation with the vendor.

All tests, including GUI and API related tests, shall be automated, whenever possible, using well-known, standard test automation software.

The vendor`s access to live/real data shall be limited according to the CPC’s requirements and shall be agreed on in advance.

6.1.8. Future proofing

The following approaches shall be considered in aim of future proofing the overall application, and extending its life:

- Use of established, well known patterns and frameworks - Only robust, well-known, industry standard and accepted components, platforms, frameworks, and programming tools should be used.
- Abstraction of external dependencies and maintaining of vendor’s agnostic approach.
- Change management strategy – Policies shall be developed and enforced for managing changes in source code, infrastructure, and application configuration.
- Monitoring and logging strategy – Monitoring tools shall enable verification that the system is functioning as expected, and if not, informative notifications shall be generated.
- Support and maintenance – A support framework shall be developed to ensure proper and secure maintenance and enhancement of the developed application. A minimum support period of 3 years shall be considered with the opportunity to be

prolonged, as necessary. Support and maintenance shall also consider the system's infrastructure components, including corresponding hardware/software, and licenses. The vendor shall be capable of providing 24/7 support services to the CPC, as needed, and based on the severity of identified issues, such issues should be resolved within an agreed-on timeframe.

6.2. Performance requirements

The system shall be capable of managing at least 5,000,000 submissions per annum, and support at least 20,000 concurrent sessions. The system shall be able to analyze (plausibility check) at least 25,000 gift registration submissions in 1 hour, and the growth rate of data shall be considered to maintain the performance needs for at least next 5 years (20% average annual growth). The system should be easily scalable to support more submissions and concurrent sessions and enable more processing power, if necessary.

Scalable storage solutions shall be considered to support the growing storage needs of the system. This shall be a Storage Area Network (SAN) arranged with an acceptable redundancy level (e.g. RAID 6) which will provide hybrid storage devices, such as fast performing (read/write) storage (e.g. solid-state drives) for live data, hard disks for data which is accessed less often and tape backups for archived data. The system shall have access to 50 TB storage, which again should be able to be easily scaled, whenever necessary.

Performance of web pages shall be measured by well-known tools such as Google PageSpeed Insights (scored not less than 95% both for desktop and mobile). And the overall system shall guarantee 99.9% uptime.

6.3. Data migration requirements

It should be possible to import the data from the legacy system into the new gift registration submission system. Controls shall be implemented in the process to ensure end-to-end data integrity and completeness during the whole process.

6.4. Architecture requirements

The system shall be designed based on microservices architecture, which will offer enhanced scalability and flexibility. It should follow all well-known principles of such architecture and conform to the REST architectural style.

6.5. Deployment requirements

Software components (services) should be packaged up with all dependencies, enabling the application to run reliably on any infrastructure, while remaining isolated from its running environment. Software should be cloud deployment ready, but also easily deployable on premise (local servers).

For better scaling and management, a container orchestration solution such as Kubernetes should be used. All infrastructure-related configuration shall be provided as code (IaS), and infrastructure monitoring solutions shall also be deployed. Infrastructure security benchmarks shall be defined based on best practices, such as those provided by the Center of Information Security (CIS), Docker Benchmark, and CIS Kubernetes Benchmark.

6.6. Documentation requirements

The following documentation shall be submitted for review and acceptance:

- System documentation, providing an overview of the underlying technology
- Requirements documentation, including business rules, use cases and user stories
- Software architecture documentation, including designed APIs, and diagrammatic representation of the overall system and underlying infrastructure
- Maintenance documentation, describing limitations and known problems within the system and implemented solutions; dependencies between system components shall also be presented in this document
- User manuals, including an end user manual, system installation and administration guides.
- Network topology of all layers, including physical and logical, firewalls, IPS/IDS systems, hardware interconnections.

Documentation shall be provided both in a searchable text format (e.g. PDF, Word, etc.) and on an open source, web-based repository, such as wiki.js, which will enable easier search, maintenance, and use. API documentation should be designed and delivered using frameworks which will reduce efforts for future maintenance and development (e.g. Swagger).

6.7. External database connectors

External databases to be connected to the system shall be identified, data quality assessed, and types of data to be extracted identified by the vendor, and related connector modules, which will be based on a predefined and easy to customize mapping logic, shall be developed. Such modules should be developed in a way that will enable connecting all types of widely used database technologies for digital services in Armenia, and other systems which will be connected to the CPC's system. Data mapping from external systems should be easily manageable, maintainable, and customizable by the CPC.

These connector modules will be developed in compliance with GIP requirements. This will enable reusability of the modules and reduce the costs of development, if/when any directly connected database becomes a part of the GIP. Such connectors shall consider one-way data retrieval by default and enable two-way data exchange, only if considered necessary by the CPC.

Declaration forms should be preloaded with the data which is already available in state registries and allow the declarant to edit/add data, if needed - the interconnection with external databases being the core enabler of the system's auto fill feature.

Standard fields such as country, car makes/models, etc. should also be loaded from constantly updated sources, which contain quality data.

Such external data sources shall include, not only the state databases, both connected and disconnected from the government's interoperability platform, but also resources such as investigative journalism sources, social networks, WikiLeaks, news websites (used by media monitoring module) for data scraping and processing.

All such external data sources shall be specified, and data collection and storage approaches defined by the vendor and approved by the CPC. To support the vendor`s efforts, a comprehensive list of databases available from each state agency, with general details regarding type of information stored in each, will be provided.

Such connectors should be developed in a way that will enable adding new sources, as needed, with no or minimum modification required.

While identification of the exhaustive list of external databases necessary to interconnect with the CPC's system shall be done (and containing data quality assessed and verified) by the vendor, as follows is presented a set of databases which are either already connected to the CPC, or the GIP (details can be found in annexes 4 and 5) and are considered to be of high priority for enabling overall functionality of the system:

#	Name	Enabler for
1	RA Ministry of Justice - Population registry	Extraction of personal information which will enhance analytical capabilities.
2	RA Police - Population registry	Extraction of personal information which will enhance analytical capabilities.
3	RA State Revenue Committee - Customs registry	Analyzing interests/assets/expenditure based on assets/goods imported.
4	RA State Revenue Committee - Tax registry	Analyzing interests/assets/income based on declared incomes.

6.8. Data retention

All the gift declarations and related information to the submitters shall be stored indefinitely, allowing the CPC to archive the data whenever needed. In case of termination of the position, the gift registration form remains published in the register for three years and is then archived. Archived forms should be restored if a person holds a public position after three years. Deletion, modification, or archival of any data related to the declarants, filled forms, performed, and finalized audits, issued decisions, and any other records, shall not be possible by a single user, and whenever attempted, an easy to track and verify audit trail shall be preserved. The data shall be maintained linked to the submitter and his/her history in the system. Implemented solutions shall ensure that software performance will not be affected as data volume increases.

Automated backup of the full data, both on internal and remote storage (disaster recovery site) shall be made possible based on defined, easy to modify periods.

Audit trails and logs shall not be modifiable (edited or deleted) by any single user, including super users (administrators). All the logs (including audit trails) should be stored in a separate database, independently maintained.

Infrastructure necessary for data retention (Storage Area Network) will be designed and provided by the vendor.

6.9. Data storage and structure

The aim of the CPC is to be able to apply data driven decision-making to corruption prevention. Thus, the new gift registry system should ensure that collected data is stored and structured in an effective manner to supplement further analytical and monitoring activities (e.g. OLAP). It should be flexible enough to integrate new developments and expansion of data whenever required by the CPC. It should also provide fast and effective extraction of various data sets, therefore enabling efficient analytic performance for the CPC. Data storage and structuring should meet enhanced requirements for security and protection of privacy and confidentiality.

6.10. Access management and user roles

Role Based Access Control (RBAC) shall be implemented. Resource types should be identified, and granular permissions (view, modify, create, delete) will be defined. Resources are elements to which privileges should be assigned. Examples of such resources for the gift registry system are gift registration forms, privileges management, logging configuration, etc. Each resource shall also have a confidentiality level assigned to it, which will define what clearance level will be required to view that resource. The clearance level, per user type, can be assigned separately (users with same role can have different clearance levels). Recommended confidentiality levels are:

- Confidential (top confidentiality level)
- Restricted
- Internal use
- Public (everyone can see the information).

The application administrator should be able to either assign predefined roles or create custom roles by grouping various permissions on resource types.
Predefined user types shall include:

- *Public officials and Public Servants*: There can be different types of individuals who will need to have a different type of dashboard and gift declaration forms to fill;

- *CPC staff*: CPC staff will be able to log into the system and access their workplaces, based on their assigned roles. The system shall provide the flexibility to reflect the evolving organizational structure of the CPC, with corresponding business functionalities regarding declaration processes.
- *Public access*: Access to the CPC's website, published gift declarations and the public API shall be provided without any authorization requirements.

Remote access requirements shall be identified also based on user roles. System functionalities which shall be made available from a remote workplace shall be configurable for each user role and individual user, if necessary. All other functionality, which is not specifically allowed to be accessed remotely, should not be possible to use.

6.11. Machine readable forms

It should also be possible for the users to export forms in machine readable format, making it possible to upload into the system, passing through the recognition engine, which will convert the data in the form into text filling in corresponding fields of the database.

6.12. System components

6.12.1 Frontend (User Interface)

Each user shall have a responsive dashboard and corresponding working space to perform tasks based on their assigned role and privileges (details are included in sections 3, 4.4, 5.4.1, 5.5, and 5.6 and main user groups are mentioned in section 6.10). Users shall be provided with all the relevant reporting, visualization, data extraction, analysis, and other supporting tools, which will enable them to undertake their tasks. A content management system (CMS) like solution should be provided for dashboard and workplace customization, allowing users to add/remove tools on their dashboards (e.g. data visualization capabilities such as those provided by Tableau).

To provide additional flexibility and accommodate changes in legislation, it shall be possible to design forms used for gift declarations interactively, using predefined question types such as[2]:

- Multiple choice
- True/False
- Numerical (includes integers and floating-point numbers)
- Drop down fields (where answer options are filled either manually, or by connection to external sources, e.g. country names)
- Calendar
- Descriptive (free form text).

It should be possible to select required fields.

6.12.2 APIs

API Gateway: This is the main gateway serving as the bridge between the public interface (frontend) and individual microservices. It should be responsible to take all calls from clients, route to appropriate microservice, composition, and protocol translation.

Public API: Public access to the gift registry submission data shall be made possible through a publicly available API, which will allow online data extraction from the system in searchable formats, which will enable further analysis and feeding to external analytics

[2] A more comprehensive list of possible question types with corresponding examples can be found here: https://www.questionpro.com/article/types-of-questions-question-types.html

systems (e.g. M.S Excel, JSON, CSV, XML, etc.). CPC staff should be able mark the gift registry submission fields and any other data which are considered to contain personally identifiable information (PII) and exclude them from public access. Data export filters such as years of service, PO grade, etc. should be implemented to increase the export process efficiency, considering that the data volume will increase considerably over time.

Interoperability API: This is the adaptor to serve for data import/export from/to the government interoperability platform. Such adaptors are required for all external databases, which will serve as data sources for the gift registry submission system.

6.12.3. Databases

Technologies for each storage/database should be chosen to provide the best performance, based on the type of data stored and transaction characteristics. Data should be retained indefinitely, without compromising the system's performance as volume grows. Archival/backup/restoration capabilities shall be easily enabled, and timeframes and paths should be configurable.

All databases should be hardened, designed, and developed based on the performance and security best practices of the database technology vendor and wider industry.

6.13. Hardware/Software requirements

6.13.1 Frontend

Frontend is considered as the user's interaction interface with the system. The system shall provide a responsive web interface and mobile app for end users to interact with it. The interface should easily adapt to the different standard screen sizes and work flawlessly with the most used web browsers, including but not limited to Google Chrome, Mozilla Firefox, Safari, Opera, Microsoft edge and Internet Explorer.

Frontend should follow all security best practices. The vendor will be required to provide the minimum requirements for client systems (end user computers, CPU, RAM, storage, graphic card, mobile phone specifications, etc.) and browser version compatibility information, as necessary.

6.13.2. Backend

Backend (business logic and all the non-user facing components) shall be designed following micro services architecture.

The system should be capable of utilizing the hardware resources as efficiently as possible, thus reducing the cost and need for higher-end components and configuration. It must run in a virtualized environment (using virtual machines).

Infrastructure, on which the backend will be deployed, should be capable of supporting implementation of private cloud, be easily scalable, as needed, and to increase security and reduce licensing fees, it is recommended to use platforms well known for better built in security measures (e.g. Unix/Linux). Hardware and the overlaying platform should be hardened based on the vendor's recommendations and industry best practices.

The vendor will be required to provide a minimum hardware (physical components, VM configuration, storage, network requirements etc.) and software configuration for the host system, that will be needed to meet the performance and security requirements mentioned in this document.

6.13.4. Security

The vendor shall also consider acquiring, installing, and configuring security components such as hardware/software firewalls, intrusion detection/prevention systems (IPS/IDS) and anti-virus software.

6.14. Project Evaluation and Oversight

The CPC recognizes that evaluation and oversight of the project to create this new E-Platform will require certain expert qualifications and experience and cannot be provided by the CPC itself. While the overall responsibility will remain with the CPC, the professional and technical oversight and evaluation of project implementation will be outsourced to independent organization(s) or person(s) with subject matter expertise to ensure all the business requirements provided in the ToR are consistently realized and that the final outcome will be fit-for-purpose. Further, this subject matter expert(s) will help ensure that the project goals can be implemented in phases according to a sequence that maximizes cost-effectiveness.

6.15. Project phasing

Priorities of system functionalities are grouped into phases as shown below. The vendor will be required to coordinate with the CPC in addressing the timing of phases, taking into consideration both functional, nonfunctional and hardware infrastructure requirements.

Phase 1:

- ***Work Plan:*** Development of a mutually agreed work plan by vendor including but not limited to**:**
    - Identifying all the data points which will enable the implementation of defined features;
    - Provide a report about the quality and structure of data stored in all identified data points, issues, and recommendations to address interconnection issues with the CPCs system, and possible alternative solutions aimed not to compromise system functionality due to potential low quality of data;
- ***Prototypes:*** The vendor shall demonstrate simplified version of the designed functionalities within the system for CPC to ensure mutual understanding around how the system will work. A prototype for the core functionalities of the system will be presented to CPC.

- ***User experience (UX) design documentation:*** UX design documentation shall be provided according to the approaches defined in this document. This shall also include wireframes of the interfaces. The UX design package shall be approved by the CPC.
- ***User interface (UI) wireframes:*** User interface design documentation based on the wireframes approved in the phase.
- ***Project portfolio risk analysis:*** All potential issues, gaps, risks, and technical constraints which might result in difficulties in implementing required functionalities and reaching the defined goals shall be identified by the vendor and be reviewed in detail. A mitigation plan/solution for each identified issue shall be proposed by the vendor.
- ***Hardware infrastructure specification:*** Vendor shall provide technical specification of the required IT and network infrastructure and develop proposals / recommendations for architecture and related infrastructure of the Solution, based on the principles defined in the section 6 of this document.
- ***Testing strategy:*** Test strategy based on the acceptance criteria and functional specification which shall be approved by CPC. Document test strategy and test analysis in connection to the acceptance criteria and functional specification.

Phase 2:

- ***Technical/Architectural design specifications***: Detailed specifications of the solution including interaction interfaces and diagrams (Data Flow Diagrams), Use Cases, data retrieval and validation scenarios.
- ***Authentication:*** Declarants shall be able to register using a strong password, email address, their social security number as their username, and an OTP made available via SMS. RBAC shall be implemented, resources identified, and classification levels applied. *Defined in section 3.2. and 6.*
- ***Secure portal and profile:*** User profile shall be manually filled by the declarant. *Defined in section 3.2. 1..*
- ***Notifications:*** System shall be able to send email notifications and alerts on predefined timelines. *Defined in section 5.4.5.15.*
- ***Filling forms:*** All the functionality *described in section 3* should be implemented except autocomplete functionality. Secure portal shall be implemented allowing declarants to easily fill in the required forms.
- ***Automated screening and verification system for submissions:*** Formal check, publication and submission compliance check functionality should be fully deployed. *Defined in sections 5.1 and 5.2.3.*
- ***Case management dashboard:*** Will be implemented according to requirements *defined in section 5.4.5.13.*
- ***Automated publication system:*** Will be implemented according to requirements *defined in section 5.5.*
- ***CPC Workspace:*** Overall workspace design and integration shall be implemented as *defined in sections 5.4 and 5.6* except cross checking capabilities with external databases
- ***Focal points portal:*** Portal for updating declarants' roster by focal points (HR departments) as *defined in section 4.4.1.*

- ***Banking sector portal:*** Portal for collecting data from RA Banks, where responsible personnel from banks will login and fill out corresponding forms. *As defined in section 4.4.2.*

Phase 3:

- ***Authentication:*** Authentication based on e-ID, m-ID, and integration of third-party authentication apps for generating OTP. *Defined in section 3.2 and 6.*
- ***Secure portal and profile:*** State databases shall be integrated, and data extraction should be performed so that the declarant`s profile will be auto filled. *Defined in section 3.2.1.*
- ***Notifications:*** Declarant shall be able to select following options for receiving notifications via automated audio calls using provided phone number, text messages (SMS), text messages using messaging services (e.g. WhatsApp, Viber, Telegram, Signal, Facebook Messenger). *Defined in section 5.4.5.15.*
- ***Filling forms:*** State databases should be integrated with the system and auto complete feature should be activated, which will also enable enhanced field verification.
- ***Automated screening and verification system for submissions:*** Plausibility check should be deployed as *defined in section 5.5*, with all the corresponding risk modelling and analysis capabilities. Also, formal check features shall be enhanced as external databases will be integrated into the system by now.
- ***Risk management dashboard:*** Will be implemented according to requirements *defined in section 5.2.3,* with reduced functionality without AI and machine learning capabilities.
- ***CPC Workspace:*** Cross checking capabilities with external databases shall be implemented along with dashboard customization features.
- ***Reporting:*** Should be fully deployed according to *description provided in section 5.4.5.4.*
- ***Media monitoring:*** Should be fully deployed according to *description provided in section 5.4.4.10.*
- ***Communications module:*** Should be fully deployed according to *description provided in section 5.4.4.11.*

Phase 4:

- ***Risk management dashboard:*** AI and machine learning capabilities will be added to the system as *defined in section 5.2.3.*
- ***Authentication:*** Adding biometric factors for multifactor authentication including face, voice, fingerprint recognition.
- ***Mobile app:*** Should be implemented and fully functional for iOS, Android, and iPad OS.
- ***Automated screening and verification system for submissions:*** Full audit management dashboard should be developed. *Defined in section 5.2.3.3.*
- ***Maintenance dashboard:*** Should be fully deployed according to *description provided in section 5.4.8.2.*

6.16. Project management

Efficient resource planning, allocation, and progress control during each phase, as well as quality control and monitoring and evaluation of the deliverables shall be considered throughout the project life cycle. To achieve this, the establishment of a project management office will be necessary, where all stakeholders will be represented, and which will be considered as having ultimate responsibility (with the CPC) for archiving project goals. The vendor will be responsible for project management, as well as for execution of activities and a related work plan, mutually agreed with the CPC. The jointly agreed work plan will include the necessary activities and deliverables' review and coordination process, within an agreed timeframe.

The vendor will be required to follow a well-known project management framework (e.g., PMI PMBOK, PRINCE2, etc.) throughout the project's lifecycle, which will ensure timely project implementation.

A project manager, assigned by the vendor, shall have the authority and responsibility to coordinate and implement the project so that the objectives are met in accordance with the requirements defined in this document, and the timeline mutually agreed in the work plan. S/He should ensure effective communication with all stakeholders, through progress reports (and other forms of communication, as specified by the CPC), in frequencies defined by the CPC. The vendor shall ensure timely resolution of identified issues in any phase of the project and provide a proposal describing the issue and mechanisms suggested for its escalation/resolution.