

It is expected, that the candidate will also create and maintain in a central repository all the documents, that shall be output of the project, as prescribed by the list of documents (according to templates in POxxx requirements) and as prescribed by this document.

Any documentation containing information of or about the following authorization classes must not be stored in repositories publicly or 3rd party accessible (this includes also data storage services such as Google Drive, Microsoft OneDrive, Drop Box etc.):

- "secret" or "top secret"
- "confidential"
- "protected"

where:

- secret or top secret = information of data or data being classified to the specified degree by national legislation, additional requirements for the subjects handling such data may apply, however this project does not foresee handling of any of such data
- confidential = information of or data, whose disclosure may directly reveal substantial information that might lead to any kind of abuse, gaining unjustified profit or predominance of one party above another one, this concerns also private (personal) data of physical persons including sensitive data about them. Data giving hint to-, forming or directly representing the meaning for whose control and record keeping the system is mainly designated are also considered confidential.
- protected = information being not publicly available, yet their knowledge gives or might give either direct or substantial indirect opportunity, facilitation or aid for gaining access to information with higher level of protection (e.g. confidential, secret or top-secret).
- non-public = information (or data) not belonging to either of the previous categories, however requiring protection and non-disclosure since they are not meant for publication (e.g. the scope of their intended addresses or other persons meant to deal with them is enumerable by a finite list of persons, contacts, groups, roles etc.). If not belonging into any of the previous categories, the "business know-how" and similar belong to this category.
- public = information or data belonging to neither of the previous categories, that can be or are being made available or accessible to anyone without preceding knowledge or designation of the scope of those subjects.

### 9.4.3 Other (project) requirements - Output and actions schedule milestones

In linked document:

During the execution of the contract, following milestone deadlines have to be met:

ID	Deadline	Deliverables present
M0	(date of contract signature)	Signed contract UKZUZ + contractor
M1	M0 + 7 months	<ol style="list-style-type: none"> <li>1. Acceptance protocol of system design and architecture signed by the representatives of UKZUZ and NFA (ENG/KAT)</li> <li>2. Analysis and system design document – in the latest version, ENG.</li> <li>3. Prototype / mock-ups (ENG)</li> </ol> <p><b>NOTE: The deadline for M1 is the latest one. It is expected, that at least a half of functionalities shall have an earlier (4 months!) deadline. The handover and the "fixation" or "freezing" of the requirements shall be done by a partial acceptance protocol. This deadline or number of "fixed" functionalities/requirements may however be postponed or altered contractually.</b></p>
M2	M1 + 12 months	<ol style="list-style-type: none"> <li>4. Acceptance protocol (ENG/KAT)</li> <li>5. Analysis and system design documentation – including all annexes in their respective languages.</li> </ol>

		<ul style="list-style-type: none"> <li>+ Filled test cases by UKZUZ and/or NFA (not every test case is necessarily to be evaluated by all of them)</li> <li>6. Manuals (KAT/ENG lang. where appropriate)</li> <li>7. Software <ul style="list-style-type: none"> <li>- installed in production environment as ready-to-manufacture version and the same in testing environment</li> <li>- commented and prepared source codes</li> </ul> </li> <li>8. Project documentation pack version to the date M2 minus 2 weeks.</li> <li>9. Training presence lists (signed by all attendees) – ENG/KAT language (may be postponed +3 months if necessary).</li> </ul> <p>NOTE: This acceptance is being done on-site in Tbilisi, Georgia with at least one representative of the contractor (authorized to fully represent the contractor at least in the entire scope of the project – including contractual agreements) present during the formal handover procedures (does not necessarily apply on testing).</p>
M3	M2 + 24 months	<ul style="list-style-type: none"> <li>10. Handover protocol</li> <li>11. Analysis and system design documentation – warranty claims resulting changes applied + changelog documents including all annexes in their respective languages.</li> <li>12. Software + Source codes with changes applied</li> <li>13. Manuals and other user documentation with updates (ENG/KAT)</li> <li>14. Test protocols relevant to the changes.</li> </ul>

Please note, that the abovementioned deadlines M1, and M2 reflect the **final date** of the acceptance procedure of the outputs, however some of the outputs (especially the installed software and “Analysis and system design document”) have to be present in their latest version for the respective milestone certain time earlier, as required by the testing and acceptance procedure defined within the Contract (that means at least 2 comments iterations, each in the duration of 10 working days/2 weeks at least, the mock-ups for at least one iteration of comment).

Based on the test results the delivery may be accepted, accepted with objection or not accepted at all. When a single test fails in such a manner, which makes the usage of the system or one of its use-cases or vital functionalities impossible, the system will not be accepted. However, in such a case (repeated acceptance) only that test, that have failed and have been corrected, shall be re-done. Those test results that are not influenced by the bug being corrected, may be reused from previous acceptance testing.

### 9.4.4 Other (project) requirements - Project outputs requirements (PO)

#### A note on the structure and list:

The meta-model diagram of documentation as well its descriptions are featuring the expected obligatory structure and minimum content of each of the items (minimum information given, e.g. elements and their relations used or minimum table-of-contents outline).

#### Note on the mandatory / optional items:

While an item in the following text or in the referred model may be labelled as "Optional" (or "OPT"), its inside textual description describing its outline may bear the tags "Mandatory". This is

no way contradictory, it only means, that "SHOULD the documentation item be used (if necessary) THAN (and only then) it shall follow at least those outlines or mandatory elements/links required in its description."

The contractor may append any other diagrams/models/notation recognized as necessary during the project.

### 9.4.4.1 Project outputs requirements (PO) - General

<b>PO000</b>	<b>PO000 Project outputs</b>
The structure and requirements for project outputs refer to the appropriate diagram "Delivery content (incl. metamodel) - Documentation structure"	
<b>PO001</b>	<b>PO001 Output forms - methodical</b>
Any diagrams or schemes being part of the project output in neither technological representation should be created: <ul style="list-style-type: none"> <li>Using a formal notation such as the preferred UML (1.1 and above) or BPMN, EPC or ArchiMate (version 3.x preferred).</li> </ul> OR <ul style="list-style-type: none"> <li>(especially for smaller/high-level schemas or schemas intended for end users) Using any notation, supposing all the elements used within the schema are covered in the appropriate legend.</li> </ul>	
<b>PO002</b>	<b>PO002 Output form - technological</b>
<p>The system documentation described as "<b>MODEL</b>" shall be delivered</p> <ul style="list-style-type: none"> <li>as a shared model (no important what technique, if a mailed/shared eapx/feap file, XMI imports/exports including those realized using the CVS integration or shared SEA cloud or database)</li> <li>as a model report (DOC(X) or PDF with reasonable size, split into multiple documents and reasonable comments period)</li> </ul> <p>with an exception of those marked in dark orange colour in the diagram (Data-logical diagram, Interfaces layout schemes/designs and Components breakdown and their functions), that are to be delivered as readable by non-it personnel (a specific SEA report template fulfilling this requirement is also acceptable).</p> <p>The system documentation described as "<b>USER DOCUMENTATION</b>", while it may be stored in the model as empty packages containing linked documents only, shall be delivered as:</p> <ul style="list-style-type: none"> <li>Human written and non-IT specialist readable Office format documents (PDF, DOCX, XLSX, PPTS, JPEG/PNG/BMPs or open software variants thereof).</li> </ul> <p>The documentation described as "<b>PROJECT DOCUMENTATION</b>", being stored separately, shall be delivered in those formats the contractor sees fit, if not able to export / preview in software without a licence or free of charge, the contractor has to provide at least 4 licenses to the contracting authority for the entire duration of the project including its warranty period.</p>	
<b>PO003</b>	<b>PO003 Trackback</b>
<p>For all SEA model elements applies:</p> <p>All these have to be traced back either to any meeting or other kind of dialogue within the analysis phase or to the original requirements within this Tender annex. Should any chapter/requirement of this annex prove as obsolete, this has to be mentioned explicitly in the "Analysis and system design document".</p>	
<b>PO004</b>	<b>PO004 Language</b>
<p>All project outputs shall be in English language, with exception:</p> <ul style="list-style-type: none"> <li>Software I/O on GUI as per UR004</li> <li>User and administration manuals, same multilingual rules apply as UR004</li> </ul>	

### 9.4.4.2 Project outputs requirements (PO) - Analysis and system design documentation

This document has to address following aims:

- The “central point” for capturing of requirements, architectural concepts and decisions
- The reference (may be auto-generated) for (public as well private/protected etc.) modules, classes/source files, methods, variables and data entities (persistently stored as well used inside the run-times)

It is required, that the “Analysis and system design document” exists in the following versions, e.g. is created and updated (maintained) at least for following stages of the project:

- Initial writing – the version is to be embodied by the accepted document after the analysis&design phase (= delivered by M1)
- Finalization for the project acceptance (delivered by M2) – this version should incorporate seamlessly all the changes done during the project implementation & testing and has to reflect exactly the state of the software/system as accepted during the acceptance phase.
- Finalized version + differential change lists (annexes) AND entirely updated document if any warranty corrections should be done (delivered by M3) – Documents recording the changes that have resulted from any warranty claims. Change lists should refer to relevant warranty claims or have their own acceptance clause, the updated document(s) (incl. source code) are to be delivered only as electronic version with all the factual changes resulting from changelogs have to be reflected.

The minimum content is as follows:

PO005	PO005 (updated until M1, M2, M3) Business input and requirements
<p>Consisting of (horizontal spec.):</p> <ul style="list-style-type: none"> <li>• Requirements catalogue, issues and the changed/added requirements (SEA "Custom diagram"),</li> <li>• and thereof resulting users/roles (UML or ArchiMate)</li> <li>• carrying out their respective use-cases and/or process(es) on the data, (UML/ArchiMate, BPMN where necessary)</li> <li>• enlisted as a full list of data entities existing throughout the entire system and their relations, including full list of their attributes (logical perspective, not physical) (UML class diagram).</li> </ul> <p>Optionally: data entities may be additionally represented multiple times with specification of variants of each used for persistence, data flow within the I/O of various interfaces etc. (for example: while a person may have the set of 20 attributes, this data entity may be represented not only as "general one" having 20 attributes, but also as a data entity of some say WS or GUI, where only 10 of the attributes are used).</p> <p>Minimum level-of-detail (LOD - vertical spec.):</p> <ul style="list-style-type: none"> <li>• All requirements/issues/changes independently on their respective LOD</li> <li>• user/roles independently on independently on their LOD, entire list of roles (and their permitted functionalities) must be present</li> <li>• data entities on logical level up to their atomic attributes, where the word "atomic" means "representable as some electronically recognized data type"</li> </ul>	
PO006	PO006 (updated until M1, M2, M3) System architecture design
<p>Consisting of (horizontal spec.):</p> <ul style="list-style-type: none"> <li>• Architecture (breakdown) of modules/components of the system and their respective functionality (ArchiMate - app layer (preferred) OR UML Component)</li> <li>• Interfaces design/layout of GUI, WSs, APIS containing: For webservices: methods list, their I/Os attributes and exceptions. For GUI: rich textual description of the functionalities, schemas up to the detail of data attributes (depicted as input fields labels and list/table headers) and functionalities, featuring at least one example representing also the graphical layout (template) for each type of functionality.</li> <li>• Detailed data documentation for each of the module (data structure in the database, data structure of I/O messages etc. - note: the diagram may contain only the DB persistence, the other may be and are encouraged to be part of the</li> </ul>	

respective documentation of WSeS, system classes (such as MVC - Model) etc.) (ERD / UML Data modelling diagrams)

- Deployment diagram (how the respective modules/components are deployed and where (ArchiMate technol. layer (preferred) OR UML deployment diagram)
- Test cases (UML testing)

Note: As one may reason from the text above, the architecture design documentation is required only in the form of a "static" perspective. However, a dynamic (process-oriented) documentation has to be added where the collaboration of multiple parts of the system may not be easily deduced. However, this dynamic view documentation does not have to be comprehensive (done for each of) for all the functions/functionalities/cases/processes intended within the system.

Minimum level-of-detail (LOD - vertical spec.):

- Architecture of modules/components of the system = a "module" is considered up to a component having the size of single I/O screen/report or its part devoted to a single or prevailing data entity, proposed minimum LODs:
  1. General overview of modules (= system and its main parts forming different binaries to be setup on different servers)
  - 2.(or up to 3.) Modules structure (what pages/templates/ function groups/WSeS/robots or automated agents triggered events etc. the part consists of)
  - OPT: 3.( or up to 4.) Inside parts structure (control sets/custom control/ custom sub-templates/WS methods) does each page/function group/WS consist of.
- their respective functionality = full list of functions that the part (on level 1.-2. above, optionally: 3.-4.) provides, the LOD of the functions is let up to the discretion of the contractor
- Data (physical representation) up to their atomic attributes, where the word "atomic" means "representable as some electronically recognized data type"
- Deployment diagram = components representable as artefacts, e.g. packages deployable on nodes, nodes and infrastructure describing a technologically configurable environment
- Test cases (UML testing) = at least in such a detail, that each test-case belongs to each function/requirement or small groups thereof.

### 9.4.4.3 Project outputs requirements (PO) - Manuals

As annexes to the Analysis and system design document, following system manuals are to be delivered as addition to the abovementioned documentation (each in all languages as indicated by the abbrev. in braces):

<b>PO007</b>	<b>PO007 (until M2, updated if necessary until M3) The user manual (ENG/KAT)</b>
Fully describing all of the functionalities and processes covered by the software and the ways of their usage designated for all roles of end-users, excl. user-side administrators.	
<b>PO008</b>	<b>PO008 (until M2, updated if necessary until M3) The user manual for administrator</b>
Fully describing all of the functionalities and processes covered by the software and the ways of their usage designated for the user-side administration tasks (such as for example the editing of registry or selection lists if not covered by a (manual for a) specific GUI).	
<b>PO009 (updated until M2, M3) Technological administrator and deployment manual</b>	<ul style="list-style-type: none"> <li>• Deployment and technological architecture manual, including               <ol style="list-style-type: none"> <li>1. full step-by-step manual for installation of the application (including basic schemas of application layout and setup possibilities, if applicable),</li> <li>2. the documentation of all the configurable variables in any configuration repositories (such as but not limited to: the Microsoft Windows registry, any text/xml/csv/tabbed/json and similar files, the database, configuration repositories of any kind in any proprietary software including the RDBMS being used, templates/reports and the available variables for the report templates configuration; if a 3rd party software is used, then the documentation may be covered by the documentation of the 3rd party software itself, however an addendum has to be made about how the configuration influences the delivered software) and</li> <li>3. Network architecture and ports used between the components (may use the</li> </ol> </li> </ul>

	<p>"Deployment architecture" diagram).</p> <p>4. Backup &amp; recovery plan containing:</p> <ul style="list-style-type: none"> <li>• Analysis of data assets (may be part of data models)</li> <li>• Architecture of backup solution(s) being employed</li> <li>• Data assets &lt;--&gt; backup solution(s) (which asset is backed up by what solution and with what frequency, type of backup etc.)</li> <li>• Steps to check the backups being done correctly (per typ of backup solution) or (should this require manual operation) the manual for performing these steps.</li> <li>• Steps to recover each data asset (per type of backup solution)</li> </ul>
<b>PO010</b>	<b>PO010 Developer documentation (workstation setup + licenses)</b>
	<ul style="list-style-type: none"> <li>• Development prerequisites needed for the code to be run (including 3rd party tools installers where possible or necessary from the point of view of the licensing terms) incl. processes and means of keeping track of changes and patching changes made during the warranty period. + Development environment setup guide (may include virtual machines images, services mock-ups etc., however these do not substitute the step-by-step guide) - this does not apply for those (parts of) developer seats corresponding to standard administration tools for SITIAgri platform.</li> <li>• Source codes files / configuration projects and resources structure overview and description (UML: any kind of structured overview of the files groups and their link to respective modules (architecture modules - LOD 1.)) + their inner logical program design structure (classes, object model)</li> </ul>
<b>PO011</b>	<b>PO011 API Developer documentation</b>
	<p>Public and internal API (each as separate document!) developer documentation, one for each type of an endpoint (SOAP/DLL/COM, header files, network endpoints etc.) describing:</p> <ol style="list-style-type: none"> <li>1. Methods exposed (including their link to relevant requirements and the processes it belongs to)</li> <li>2. Their I/O data and their attributes up to atomic items (primitive datatypes), including their allowable ranges and content types/forms; in case of numbers or lists influencing the flow of the handling of the data also with documentation on the specific values and the consequences on the process for each of them.</li> <li>3. Their exceptions and errors that can be thrown/returned, their possible reasons (as a short textual fault analysis guide) and possible solutions.</li> </ol>

### 9.4.4.4 Project outputs requirements (PO) - Project documentation

<b>PO012</b>	<b>PO012 (continuous updates) Project documentation</b>
	<p>The project (management) documentation shall consist of:</p> <ul style="list-style-type: none"> <li>• Contractual docs (e.g. the contract, its attachments, amendments and contractor proposal in the tender) NOTE: While the electronic copy of these document shall be accessible within the repository, the contractor does not have to provide it back to the awarding authority when delivering the delivery at any milestone.</li> <li>• Cooperation requirements plan (required role + schedule) - may be done also as a set of (outlook or other similar PIM) invitations with at least one month in advance</li> <li>• Repository of meeting minutes.</li> <li>• Project plan as per PO013 including schedule</li> </ul>
<b>PO013</b>	<b>PO013 (continuous updates) Project plan</b>
	<p>A project plan has to be created and further maintained by the candidate (contractor), containing at least the following information:</p> <p>A) An organizational a managerial structure of the project, consisting of:</p> <ul style="list-style-type: none"> <li>• Roles of project participants</li> <li>• and their mutual relations</li> <li>• and their competencies within the project and outside the project (towards the broader scope of stakeholders around the project).</li> <li>• The name of physical persons acting within these roles (employees or equivalent of the contracting authority as well the contractor).</li> </ul> <p>B) The project schedule (based on the required schedule as described in these terms and requirements), including the milestones of the project, that have been identified and set during the preliminary as well main analytic phases.</p>

It is required, that at least following phases and their items are scheduled within the schedule (the following list does not cover those phases/items, which arise or might arise from other requirements):

1. (phase) Analysis: (item) physical presence for analysis phase on-site
2. (phase) Analysis: (item) creation of the “Analysis and system design documentation”
3. (phase) Analysis: (item) creation and opponency of schemas/mock-ups or beta-version of the GUI of the final system
4. (phase) Analysis: (item) opponency & acceptance process of the “Analysis and system design document”
5. (phase) Analysis: (milestone) Acceptance of the “Analysis and system design document”
6. (phase) Testing: (item) physical presence for supporting the (acceptance) testing phase on-site
7. (phase) Testing: (item) Acceptance testing by the contracting authority
8. (any appropriate phase) training periods
9. (phase) Support: (item) Support (remote or on-site in Georgia, when needed) for warranty claims that might emerge during the production usage of the system within the warranty period agreed within the contract.

C) List and contents (and preferably templates) of relevant project documents

D) Project meeting types and schedules

E) Other project-dependent processes and their agendas including the required cooperation by the contracting authority

### 9.4.4.5 Project outputs requirements (PO) - Software

As annexes to the Analysis and system design document, following software are to be delivered:

PO014	PO014 (continually) Software - installed instance
<p>As a part of the delivery, the software has to be installed (and covered by all licenses required):</p> <ul style="list-style-type: none"> <li>• (until M1) Mock-ups /beta-version prototype in any remote-access capable environment provided by the contractor, based on the “Analysis and system design document” initial writing version</li> <li>• (until M2) Set up in designated testing/production environment as beta version / launch version.</li> <li>• (until M3) Code changes applied and delivered according to the warranty period changes. + testing protocols for those (regression) tests that are relevant to these changes.</li> </ul>	
PO015	PO015 (continually, optionally) Software - updates and patches
<p>It may be, that by recommendation, no 3<sup>rd</sup> party employees will have access to the production environment. In such case following documents and sources have to be provided for each patching/update of the production software (typically bugs correction during the warranty period), this also applies for client-side software if necessary:</p> <ul style="list-style-type: none"> <li>• Patch binaries/install packages in a form best ready for the installation with the smallest number of steps required (for example: source code is all right, if the installation means the copying of scripts into the web-app directory or compilation with heavy optimization for the target platform that cannot be done in the development environment or is required due to security reasons or considerations; however pre-compiled binaries for Copy&amp;Paste or registration or one-click install – such as but not limited to installation scripts/projects - are preferred)</li> <li>• Patch manual – step by step guide for installing the patch, including the editing of configuration files etc.</li> <li>• Patch tools – any tools necessary for the execution of the steps described in the manual</li> <li>• Client-side software only: Installation tool suitable for user-side install done by operator with little or no IT-related knowledge or suitable for remote/bulk deployment on the target platform.</li> </ul> <p>The delivery mentioned above may be omitted if agreed otherwise with the NFA during the analysis.</p>	
PO016	PO016 (updated until M2, M3) Software - source codes
<ul style="list-style-type: none"> <li>• As a part of the delivery, the entire source codes created or being used within the development of the software in any way, unless licensed separately as 3rd party closed binaries, have to be handed over during handover/acceptance (M2, M3).</li> </ul> <p>All source codes have to meet following requirements:</p> <ul style="list-style-type: none"> <li>• Commented in such a level of detail, that at least blocks of code performing complex functions are being described. The comments have to be exclusively in English language.</li> <li>• They all meet the same form of indenting, brackets usage etc.</li> </ul>	

- All variables (including any persistent data such as DB structure description elements and including runtime data such as method parameters or API attributes) have to be named in English language (when the variable name is representing any meaningful information, this does not apply for indexing variables, technological filed/stack/pointer counters etc.)
- This does not apply for those packages/scripts/objects and other entities of the systems, where such commenting is not possible or some of the concepts (such as variables, bracketing/indenting etc.) are not being used.
- Any changes made during the warranty period have to be handed over as well (as changelog, differential patches or similar), including one last handover of the entire (patched) code at the end of the warranty period or the whole set of patches once more.
  - During the warranty period of at least 2 years the contracting authority (or any 3<sup>rd</sup> party in contracted by them) is/are not entitled to make any changes of the software without providing the relevant differential patch being shared and accepted by the contractor by means as described and approved in the “Analysis and system design document” by the contractor.
  - The contractor is then required to patch his version of source codes (or similar packages/projects/configurations) in order to be able to provide warranty changes.
  - The contractor is not to be held liable for any bugs resulting from within these patches. For clarity the word “resulting” means in the context of this paragraph that either the patched lines/segments of code/configuration entities either throw an unexpected/unhandled error or they alter the data of the application in such a way, that any other lines/segments/configuration entities, even the original ones, throw an unexpected/unhandled error due to this concrete data value. “Data” means not only persistent ones (files, database etc.), but also the runtime data (variables, temporary files, calling parameters of functions/methods, network transported data etc.) within any part of the software application system.
  - The above also applies proportionally for various backups/exports etc. of in-system specifications of datasets, forms, commands/actions etc. as far as the underlying platform allows.

### 9.4.5 Other (project) requirements - Minimum project actions (tasks) requirements (PA)

<b>PA001</b>	<b>PA001 Project actions general</b>
The contractor shall ensure and perform all steps necessary to reach the goal and aim of the project - the delivery of the system as required by the requirements specified in this document. Above this, following mandatory steps cannot be omitted:	
<b>PA002</b>	<b>PA002 Analysis on-site</b>
It is required, that the contractor provides at least one physical presence on-site (Tbilisi, Georgie) for performing the process of analysis within the proposed scope.	
<b>PA003</b>	<b>PA003 Analysis in Czech Republic</b>
It is required, that the contractor performs at least one physical presence in Czech Republic (Prague) for performing the process of analysis within the proposed scope.	
<b>PA004</b>	<b>PA004 Acceptance testing on-site</b>
It is required, that the contractor provides at least one physical presence on-site (Tbilisi, Georgia) in order to support process of testing within the proposed scope.	
<b>PA005</b>	<b>PA005 User training on-site</b>
It is required, that the contractor performs an on-site training in English language for following target audience:	
<ul style="list-style-type: none"> <li>• 5 system administrators if at least 1 nominated by the Georgian side (IT administrators/developers as well data/user administrators of total max. 5 persons)</li> <li>• 20 users if at least 1 nominated by the Georgian side</li> </ul>	
in the total length of at least 2 days (1 day if no users or no administrators are nominated, 0 days if no users nor administrators are nominated), covering at least following themes:	
<u>IT administrators/developers:</u>	
<ul style="list-style-type: none"> <li>• Source code structure and blueprints/patterns used and employed</li> </ul>	

- System architecture and design (incl. DB entities and relations and key sets of namespaces/packages/classes)
  - Internal API of the application (incl. DB objects such as stored procedures, triggers etc.)
  - Application installation and setup parameters (configuration, registry, command line parameters)
  - Trainees questions
- (All the above-mentioned themes especially in such a manner, so that is explaining combination and influence between the various parts of the system.)

### Administrators – data/app:

- Explanation of the admin GUI
- Administrative tasks and the performing their of incl. the demonstration of the impacts in the user-side
- Trainees questions

### Users:

- The application GUI and generic tasks/procedures - handling all the processes as per functional requirements.
- The use-cases and the performing thereof; incl. the methodology and methodical remarks
- Advanced tasks accessible to the users but not necessary for everyday usage
- Demonstration of admin GUI and the impact on the user-side of the application

## 9.4.6 Other (project) requirements - Cooperation requirements (PC)

The candidate shall indicate in his proposal, what kind and range of cooperation he expects from the contracting authority. This should be done for the following parts of the project:

<b>PC001</b>	<b>PC001 Project management</b>
Cooperation needed for project management	
<b>PC002</b>	<b>PC002 System design, implementation, testing, acceptance</b>
Cooperation needed for system design, implementation, testing and acceptance	
<b>PC003</b>	<b>PC003 Production deploy &amp; operation</b>
Cooperation needed for system production run within the warranty period	
<b>PC004</b>	<b>PC004 Risks</b>
Cooperation needed for acceptance, mitigation, transfer or prevention of risks (in such case a risk analysis has to be performed, listing all threads possible, the likelihood of their occurrence, impact analysis and the proposed type of reaction).	
<b>PC005</b>	<b>PC005 Other cooperation</b>
Other cooperation needed incl. training	

## 9.4.7 Other (project) requirements - Risks to be addressed (RR)

Note: Issues and requirements in this chapter are here for:

- information to the candidate
- as cooperation advice, since the solutions used by 3rd party to address or solve these risks / issues may result into other requirements for the software. The chosen candidate has to cover these solutions as far as they can be fairly held liable within the contract.

<b>RR001</b>	<b>RR001 GE legislation</b>
Currently, the state of the legislation in Georgia does not provide any mandatory requirement for the relevant subjects to be registered in the phytosanitary register. The legislation has to ensure, that there is mandatory registration (either by solving RR002 or other means).	

<b>RR002</b>	<b>RR002 Ensuring data flow from justice register</b>
<p>Currently, the justice register:</p> <ul style="list-style-type: none"> <li>does not provide list of new "unknown" registrations (the ID of registration has to be known to obtain further details of the subject being registered)</li> <li>might overflow the phytosanitary register with subjects, that have applied for the business of handling of phytosanitary material only for precaution, but are not handling any material yet.</li> </ul> <p>The data from justice register have to be filtered and acquired in a reasonable and useful manner.</p>	
<b>RR003</b>	<b>RR003 Determining the type of registered subjects</b>
<p>The current legislation does not address the terms and conditions for determining what subjects shall have / will have the duty to be registered.</p> <p>The legislation has to provide this clearly and a simulation and/or an estimate has to be made for determining the number of business subjects registration numbers (no. of subject registered per given time frame). This also has impact on the RR004.</p>	
<b>RR004</b>	<b>RR004 Evaluation of number of users</b>
<p>The administration processes for the phytosanitary register are not entirely clear at the present moment of writing. The administration/government processes of the phytosanitary register have to be layed down clearly, especially to determine:</p> <ul style="list-style-type: none"> <li>the final number of registered subjects (per time frame, see RR003),</li> <li>the final number of users using the system</li> <li>and eventually some details concerning the deployment architecture (branch offices etc.).</li> </ul>	

## 9.5 Requirements catalogue - Tender proposal requirements - Cooperation requirements (PC)

(recapitulation of proposal requirements identical as specified in 9.4.6.)

The candidate shall indicate in his proposal, what kind and range of cooperation he expects from the contracting authority. This should be done for the following parts of the project:

<b>PC001</b>	<b>PC001 Project management</b>
Cooperation needed for project management	
<b>PC002</b>	<b>PC002 System design, implementation, testing, acceptance</b>
Cooperation needed for system design, implementation, testing and acceptance	
<b>PC003</b>	<b>PC003 Production deploy &amp; operation</b>
Cooperation needed for system production run within the warranty period	
<b>PC004</b>	<b>PC004 Risks</b>
Cooperation needed for acceptance, mitigation, transfer or prevention of risks (in such case a risk analysis has to be performed, listing all threads possible, the likelihood of their occurrence, impact analysis and the proposed type of reaction).	
<b>PC005</b>	<b>PC005 Other cooperation</b>
Other cooperation needed incl. training	

## 10. Requirements catalogue - Tender proposal requirements

The technical part of the contractor (tender contestant) proposal should contain:



	<b>Tender001 Preliminary architecture</b>
Preliminary architecture (as per PO006 - modules breakdown) (may be based on the diagrams being part of this document) specifying concrete platforms, techniques, concepts and 3rd party tools the contractor is expecting to use in order to deliver the various parts/modules of the solution.	
	<b>Tender002 Preliminary data architecture</b>
Preliminary data architecture (as per PO005 or PO006 - data documentation/entities) (may be done as non-contractually obligatory database data model) refining as observed and analysed from the data diagram and functional requirements.	
	<b>Tender003 Interface schemes</b>
3-5 schemas of selected interfaces (as per PO006 - Interfaces design) with statement what parts/characteristics are typical for the platform used and which may be customized as part of the project, featuring at least one (general) graphical layout example of either previously done applications or designed specifically for the contract presenting the proposed graphical design.	
	<b>Tender004 Project management and testing</b>
Project procedures and testing (internal) the contractor is intending to employ within the project.	
	<b>Tender005 Patching cooperation</b>
Project procedures the contractor is intending for the patching and cooperation with the developers during the warranty period (when both sides - the users, developers and the contractor - may alter the code).	
	<b>Tender006 Project plan</b>
Preliminary project plan (as per PO013) featuring yellow "TODO" in those places the contractor cannot be yet sure of and additionally including: 1. gaps and questions currently open from the point of view of the contractor, 2. cooperation requirements as per PCxxx (including preliminary risk analysis if applicable).	
	<b>Tender007 Preliminary HW requirements</b>
As per OR003 the required HW for deploying the solution has to be specified.	
	<b>(optional)Tender008 License pricing and maintenance/assurance</b>
If any additional licenses are to be acquired for the solution, the tender proposal has to enumerate: <ul style="list-style-type: none"><li>• the price for such a licence</li><li>• any kind for a maintenance / assurance for 2 years</li></ul>	