Modernization of Government Services

in the Republic of Moldova

Project ID No. P148537

Terms of Reference

CONSULTING SERVICES FOR THE DESIGN, DEVELOPMENT, CONFIGURATION AND DEPLOYMENT OF THE GOVERNMENT SHARED DOCUMENTS "MDOC"

I. Background

The Government of Moldova is determined to fundamentally change the way how public services are provided in Moldova through a variety of interventions for modernization of service delivery, which combat corruption, foster a customer care culture, enhance access, as well as increases efficiency in the Moldovan public administration.

From 2006 to 2013, Moldova modernized its civil service legislation and administrative processes under the Central Public Administration Reform (CPAR), supported by the World Bank's administered CPAR Multi-Donor Trust Fund.

In July 2016, the Government of Moldova approved the Public Administration Reform Strategy for 2016-20201, that kept the modernization of public services delivery process among its main objectives.

To achieve the stated objectives, the Government requested the World Bank's assistance for a PAR operation, that became effective in June 2018, called Modernization of Government Services Project (hereafter MGSP or the Project).

The design of the project takes into account the Government of Moldova's vision, stated in the Public Administration Reform Strategy 2016-2020 and makes extensive use of institutional and technological achievements of Governance e-Transformation Project (GeT) implemented by the Government of Moldova and World Bank in the period between November 2011- December 2016.

In 2021, the new Executive issued its governing Programme "Establishing Good Times for Moldova"2 and set modernization of at least 100 administrative services and access of 100% of active population to electronic public services as some if its objectives. The Government Action Plan 2021 – 20223 through its envisaged actions reconfirms the determination of the Government to modernize the administrative service delivery system by improving access to public services through various channels, their efficiency, reduction of unnecessary administrative burdens and cost of services for both beneficiaries and service providers, ensuring a stable level of quality of administrative services.

MGSP continues to play a very important role in achieving the high level objectives set up by the Government. The project aims to improve access, efficiency and quality of delivery of selected administrative services through the following components:

1. Administrative Service Modernization

The key activities under this component focus on re-engineering a group of government to citizen and government to business administrative services; piloting of one-stop-shops for public service

[1] http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=366209

[2] https://www.gov.md/sites/default/files/document/attachments/programul_de_activitate_al_guvernului_moldova_vremurilor_bune.pdf

[3] https://www.gov.md/sites/default/files/document/attachments/pag_2021-2022_ro.pdf

delivery in selected locations and rolling out at national level; increasing public awareness on and advocacy for administrative services, with a particular highlight on e-services.

2. Digital Platform and Services

The main objective of this component is to digitize selected re-engineered government services; complete and strengthen a common infrastructure and mechanisms for rapid deployment of ICT-enabled public services; introduce government wide IT Management and Cyber Security standards and procedures. The component finances the procurement of additional shared computing infrastructure elements, digitization of services needed to deliver Government services electronically, as well as the development of a learning management system to mainstream the new digital infrastructure and the modernized services within the government.

3. Service Delivery Model Implementation

The objective of this component is to ensure that the institutional capabilities of key government agencies are aligned with and support the new model of public services delivery.

4. Project Management

This component supports the Project Implementation Unit (PIU), based in the e-Governance Agency (eGA), and ensures the activity of the core e-Governance Agency team.

In this regard, this document specifies the requirements for the implementation of a centralized platform for storing and sharing documents related to the public services delivered by Moldovan public authorities.

MDoc implementation will allow to standardize the processes of sharing the results of the public services provided by the Moldovan public authorities through a digital platform, which will also be accessible to the public authorities that have not digitized their services yet.

The technological and organizational benefits of MDoc implementation are as follows:

- a centralized data store with all documents delivered during public service delivery;
- standardized process of sharing documents related to the result of public service delivery;
- digitization of document sharing process for the public authorities that do not have high-performance information solutions;
- reduced costs of public service delivery;
- encouraging electronic exchange of documents between the Moldovan public authorities;
- creating conditions for the implementation of software for the creation and management of electronic archives;
- an efficient mechanism for automated data exchange between the information systems MDoc will interact with;
- integration with governmental platform services (MPass, MSign, MNotify, MLog, MPower, MCabinet, MDelivery, MWallet, Semantic Catalogue, Open Data Portal);
- a single intuitive and ergonomic user interface;
- high-performance management, configuration and dynamic development facilities.

MDoc is intended to improve public services through the digital platforms of the Moldovan public authorities and reduce the use of paper documents by providing citizens with appropriate facilities.

II. Scope of Assignment

The Electronic Governance Agency (the Client) intends to engage an IT company (the Consultant) with a skilled and experienced team to develop the governmental MDoc service based on the Agile methodology.

MDoc will be implemented by a competitive and qualified team meeting the requirements specified in chapter 8 of these Terms of Reference. The team members proposed by bidders must be available full-time during the project activities (depending on the project stage, the team members assigned to work on the activities specific to that stage must work exclusively on MDoc and will not be assigned to other projects of the Consultant).

The estimated effort to design, develop and implement MDoc is 810 man-days for a team of 6 key experts (see Table 1, chapter 8 of these Terms of Reference). The implementation period is 12 calendar months, including 9 months for development, during which the Consultant will ensure the availability and involvement of the requested experts, and a three-month warranty period.

The estimates included in Table 1, chapter 8 are preliminary and depending on project progress the effort planned for certain categories of experts can be reallocated to others but will not exceed the total amount of 180 man-days.

III. Scope of work and Development approach

The scope of work of this assignment is to design, develop, configure, and deploy the information system as a fully functional product with all functionalities in place, according to the specifications iteratively defined by the Client (the indicative set of requirements is listed in Annex 1 Annex 2, and Annex 3) and following the development approach described below.

The development of the solution will follow agile iterative software development principles. Since there are many interpretations of agile software development and in order to avoid misunderstandings, this section provides key technology principles to be used in development of the solution.

Iterative development

In contrast to waterfall software development approach, the solution shall be developed in iterations named sprints. This means that the implementation of different functionalities will take place in phases with some modules being in production while others still being in development. The priorities of functionalities included in a sprint will be determined by the Client. Sprint duration will be determined by the Client together with the Consultant.

Agile development

The development shall follow agile principles by allowing change and flexibility in implementation. Client will maintain the master list of generic requirements for the solution– product backlog, which consists of ordered business and technical requirements as seen by the Client. Items in product backlog are ordered by the Client by their priorities. Client is free to manage the product backlog by adding new items to it, removing items and reordering them as he/she desires. At the beginning of each sprint, the topmost N items that fit into a sprint are taken, and a sprint backlog is built out of them. Items in sprint backlog are further detailed and distributed to developers. Sprint backlog is not changed during the sprint.

Working product in each iteration

Each sprint ends up in a working product which is presented to the Client for acceptance in the last day(s) of sprint. The working product shall meet the agreed criteria – Definition of Done (e.g. it must be fully functional, fully tested, accompanied with relevant unit tests, accompanied with relevant documentation where necessary, complete commented source code supplied etc.). Payments will be made upon successful delivery of working packages (one or more working products). In case the deliverables contain defects for reasons not imputable to the Client, the Consultant shall fix them without impacting the time schedule and at no additional costs, including

possible visits to Client site. Working products from different sprints can be combined into a release deployed in production at Client's discretion. Any incidents reported by the Client after the release, shall be solved by the Consultant according to the agreed Service Level Agreements (SLAs) as defined in Annex 2, p.10 Support and Warranty requirements.

To ensure that the development team is in position to deliver on time working products, a Client representative – typically named the Product Owner in agile methodologies – is permanently available to the team for answering eventual questions, thus not slowing down the implementation pace.

The Consultant will appoint a Scrum Master from the team of key or non-key experts for the entire duration of the project.

The Scrum Master will be responsible for the day-to-day liaison with the Client; s/he must ensure the internal coordination and guidance of the project experts and the project coordination with external counterparts.

The Scrum Master must also ensure the availability of suitable experts in accordance with the project planning documentation.

Client involvement

In contrast with commonly used waterfall model for procurement and implementation of information systems for the government, the Client designated person – Product Owner – will be heavily involved in the development process. The Product Owner will have three core responsibilities:

1. Maintenance of product backlog – the owner will maintain the product backlog up to date, so it reflects prioritized list of desired functionalities.
2. Answering to questions coming from developers – the owner will be at all time available to the development team for answering their eventual clarification questions, thus avoiding complex and formal communication within the project. This is essential to ensure the team has all the information on time to deliver a working product at the end of the sprint.
3. Acceptance of working packages – delivered working packages are presented to the Client for acceptance at the end of each sprint. The Client shall accept the working package or notify the Consultant of any defects during the following sprint.

Although it is not strictly necessary, the Product Owner may participate in team stand up meetings listening for progress and eventual blockers for an immediate reaction.

Product Owner also decides on product releases, as per release plan.

Also, as per the principles of Agile project management methodology, the Client will define the Product Vision Statement and Product Roadmap in order to track progress and to ensure the appropriate product development.

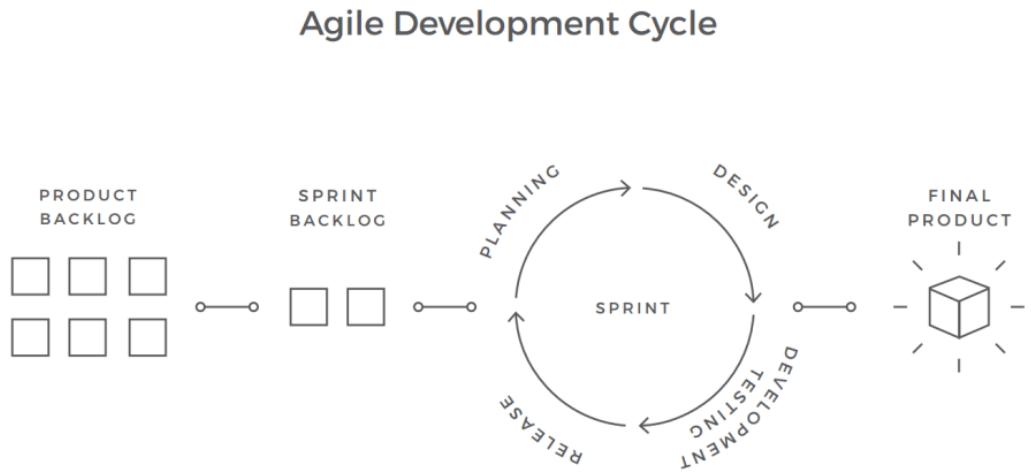

Figure 1. The indicative illustration of the Agile Development Cycle/Process.

Warranty

The Consultant shall provide 12 months of warranty for the developed solution. The warranty period starts after final release. During the warranty period the Consultant shall fix any identified defects.

The development and operations must be in compliance with the legal and regulatory documents listed in Annex 3.

Required technology stack

To preserve e-Government investments, the solution shall be developed using the latest versions of the following technology stack:

- Programming language is C#.
- ORM is Entity Framework Core.
- Web framework is ASP.NET MVC Core.
- RDBMS is Microsoft SQL Server.
- Container engine is Docker.
- Container orchestration is Kubernetes.
- Cache server and session store is SQL Server or Redis.

During the development process, the Consultant or the Client may propose use of additional components required for the development and proper functionality of the solution in production. Upon the Client's approval of such components, the costs for them shall be added through amendments to the contract.

IV. Expected Deliverables

The following deliverables will be provided by the Consultant during this assignment:

1. A fully functional information system with all functionalities developed and deployed according to the requirements defined by the Client during the assignment. The Consultant will deliver compilable and documented source code (including third-party tools and libraries, licenses, where applicable and automation scripts).
2. Technical and End-user documentation developed according to the Client's documentation requirements defined in Annex 2.
3. Training sessions and training materials developed according to the Client's training requirements defined in Annex 2.

Please note that any population with or migration of data is not part of this assignment.

V. Reporting Requirements

The following reports will be provided during the assignment:

a) Sprint Report, including release notes, breakdown and duration of tasks implemented during the sprint, velocity, issues and outstanding problems, proposed actions to be taken;

b) Next Sprint Backlog, including breakdown and estimated duration of tasks proposed to be implemented during the next sprint, resources that the Consultant expects to be provided by the Client and/or actions to be taken by the Client;

c) Training reports, submitted after each training session, including:

- Participants list;
- Training session agenda;
- Training materials (presentations, labs etc.);
- Trainees test results.

VI. Timing

The tasks defined under the current contract are estimated to be performed in 9 months for development and 3 months of warranty period. If new functionalities will be identified by the Client based on users' feedback and subject to satisfactory performance, the contract can be extended based on the same fee rates.

VII. Institutional arrangements

The **Client** is responsible for all administrative and procedural aspects, contract and financial management, including acceptance and payment of deliverables/reports expected under the Contract, general project responsibilities and efficient coordination with stakeholders.

A Product Owner will be appointed by the Client and will coordinate and decide on all issues related to the technical elements of the Contract. The Product Owner will issue the administrative notice on the start date of the implementation of the contract and other administrative duties.

The Client will provide the following:

- infrastructure resources for testing and production environments;
- code repository, issue tracking system, CI/CD environment, task management system via the Client's subscription in Azure DevOps. The Consultant shall not include Azure DevOps subscription in its financial proposal;
- Training facilities.

The **Consultant** will ensure that adequate working conditions (workspace/office premises for experts, office equipment, computers, communication facilities, etc.) and services are provided to the Consultant's staff during the lifetime of the project.

The Consultant will be responsible for the day-to-day management of the project team and availability of necessary resources.

The Consultant will organize the Kick-off meeting and initial GCSS Backlog at its premises. All Consultant's Key Experts as specified in the section defining the qualification requirements, shall participate in the Kick-off meeting and initial GCSS Backlog. The costs associated with the Client's presence at the Kick-off meeting will be covered by the Client and shall not be included in the Consultant's financial proposal.

The Consultant will ensure visits to the Client site to provide training to end users.

In case the deliverables contain defects and/or there are delays for reasons not imputable to the Client that may impact project outcome, the Consultant may be requested to visits to Client's site in order to solve the project issues.

The communication languages will be Romanian or English.

The Consultant shall work under the supervision of the appointed Product Owner and report to the Client's Chief Digital Officer.

VIII. Qualification Requirements

Consultant qualifications requirements

The Consultant shall furnish documentary evidence (including information about the completed contracts and contact information of clients from whom the references could be taken or whom the Client may, when necessary, visit to familiarize themselves with the systems put into operation by the Consultant) to demonstrate that it meets the following experience requirements:

1. Have been in operation for at least five (5) years with main part of its business being the development of information systems.
2. Experience in conducting projects similar size and complexity developing web applications proven by at least two (2) contracts with the development phase finalized in the last three (3) years. For ongoing projects, copies of acceptance documents of the entire software solution shall be provided.
3. Experience in software development using agile software development principles (as described in the scope of work and development approach section of the ToR) would be an asset. This shall be demonstrated by presenting the project methodology describing the role of the client.
4. Demonstrated experience using required technology stack would be an asset.

Staff qualifications requirements

The Consultant shall provide a team of the following key experts:

- *Key expert 1.* Business analyst/Team Leader/Scrum master;
- *Key expert 2.* Senior software developer;
- *Key expert 3.* Software developer;
- *Key expert 4.* Software developer;
- *Key expert 5.* Database administrator;
- *Key expert 6.* Software Tester and Trainer.

Each key expert must meet at least one the following requirements:

- Proven experience in web UI design and development using responsive frameworks, progressive web apps;
- Proven experience in database design, development, and optimization;
- Experience in systems' integration, API design and development using SOAP/REST;
- Experience with unit testing;
- Experience in DevOps practices;
- Experience in system analysis.

Per total the entire team of the proposed key experts must meet all the above requirements.

Offers which will not demonstrate that the team covers the above requirements may be subject of disqualification.

For proposed key experts the CVs need to be submitted, demonstrating the minimum qualifications requirements, as detailed below:

Key Expert 1. Business analyst/Team Leader/Scrum master:

The Business analyst/Team Leader/Scrum master shall oversee that all reporting obligations are fulfilled in a timely manner to a high-quality standard.

- university degree in Computer Science or another relevant domain;
- at least 7 years of experience in software development;
- at least 5 years of proven experience business analysis, team/project management with the application of Agile methodology, with at least 2 projects implemented in the last 3 years;
- at least 5 years of experience in software development using C#, Entity Framework, ASP.NET MVC, SQL Server and a dependency injection framework;
- experience with design and implementation of governmental platforms which is an asset;
- certifications in any technology from the required technology stack is an asset;
- ability to communicate in Romanian and English.

Key Expert 2. Senior software developer:

The Senior software developer shall oversee that all reporting obligations are fulfilled in a timely manner to a high-quality standard.

- university degree in Computer Science or another relevant domain;
- at least 7 years of experience in software development;
- participated in at least 2 software development projects in the last 3 years using Agile approach;
- at least 3 years of experience in software development using C#, Entity Framework, ASP.NET MVC, SQL Server and a dependency injection framework;
- certifications in any technology from the required technology stack is an asset;
- ability to communicate in Romanian or English.

Key Expert 3-4. Software developer:

- university degree in Computer Science or another relevant domain;
- at least 5 years of experience in software development;
- participated in at least 2 software development projects in the last 3 years using Agile approach;
- at least 3 years of experience in software development using C#, Entity Framework, ASP.NET MVC, SQL Server and a dependency injection framework;
- certifications in any technology from the required technology stack is an asset;
- ability to communicate in Romanian or English.

Key Expert 5. Data Base administrator:

- university degree in Computer Science or another relevant domain;
- at least 5 years of experience in data base design, data base administration and software development;
- participated in at least 2 software development projects in the last 3 years using Agile approach;
- at least 3 years of experience in software development using C#, Entity Framework, ASP.NET MVC, SQL Server and a dependency injection framework;
- certifications in any technology from the required technology stack is an asset;
- ability to communicate in Romanian or English.

Key Expert 6. Software Tester and Trainer:

- university degree in Computer Science or another relevant domain;
- at least 3 years of experience in data base design, data base administration and software development;
- proven experience in software testing analysis and design;
- proven experience in automated testing;
- proven experience in performance (load and stress) testing;
- proven experience in security testing;
- certification in testing or any technology from the required technology stack is an asset;
- experience in conducting training sessions for end users and ICT specialists;
- experience writing technical and end-user documentation;
- ability to communicate in Romanian and English.

IX. Estimated project effort

Table 1 contains a preliminary estimation of the effort required for designing, developing and implementing MDoc for all the requested categories of experts specified in chapter 7 of these Terms of References.

Table 1. Estimated effort for MDoc implementation

#	Expert competence	Man-days Input
1.	Business Analyst/ Team Leader/ Scrum Master	130
2.	Software Developer - FrontEnd	80
3.	Software Developer - BackEnd	180
4.	Software developer/ DevOps Engineer	60
5.	Developer/ Database Administrator	60
6.	Software Tester and Trainer	120

TOTAL	**810**

The Consultant's team shall not exceed the total estimated effort indicated in Table 1. Considering the Agile development methodology, if necessary, the Client can reallocate the workload among the experts involved in the MDoc development project, so that the effort that was not used by some experts can be distributed to others.

The payment will be made based on the team's actual effort agreed with the Client at the stage of planning the project SPRINTs and on the deliverables accepted by the Client.

Annex 1. Functional Requirements

This Annex describes MDoc functional requirements. The functional requirements specify what the information system should do. The Annex also describes the specific functions (actors) that define what functionalities MDoc should provide.

The functional requirements are defined by „Use Cases". Each use case is described as a set of functions to be used by user.

A1.1. MDoc Actors

The following actors will be involved in MDoc management and will act in different capacity:

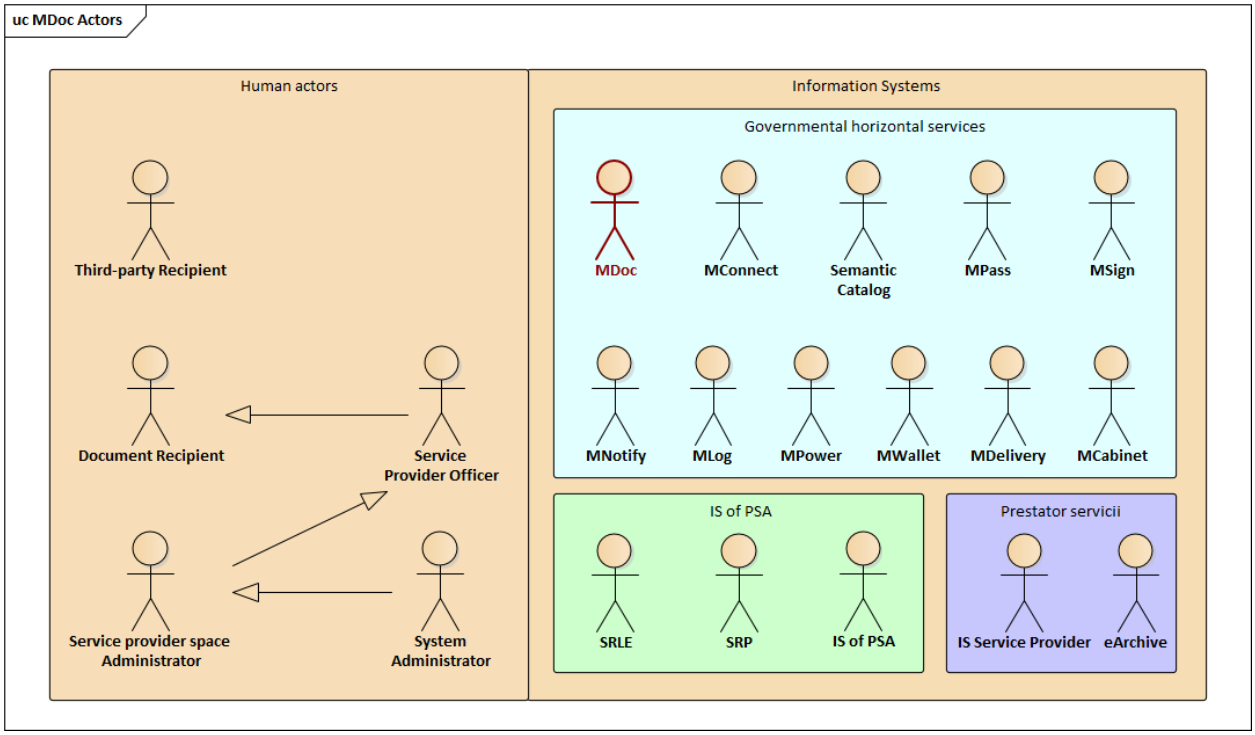

Figure 2. MDoc Actors

Document Recipient - human actor, who represents the recipients of documents prepared by public authorities and stored through MDoc (individual or business that requested the document) to be subsequently accessed and downloaded by Document Recipients or by public authorities delivering other services requested by Document Recipients.

Third-Party Recipient - human actor, who represents the individuals or businesses authorized by Document Recipient to access, view and download the requested documents through MDoc. Third-Party Recipients may also be the information systems of public authorities, which require access to specific documents in order to deliver certain public services to Document Recipients.

Service Provider Officer - human actor, employee of a public authority who uses MDoc to upload the documents requested by Document Recipients. This role will be used by the public authorities that do not have information systems allowing to send documents to MDoc using the interoperability mechanisms.

Service Provider's Space Administrator - human actor, employee of a public authority in charge of the configuration and administration of the functional and physical space reserved for the public authority in MDoc.

System Administrator –a category of users in charge of MDoc technical administration acting on behalf of the Client. Although the administration tasks in MDoc could be performed by other roles

(defined later), in this document the System Administrator will cover all MDoc technical administration tasks.

In the context of these Terms of Reference, the following information systems will need to interact so that MDoc achieves its purposes:

- **MDoc** - a governmental information solution intended for the implementation of a centralized mechanism for storing and sharing documents resulting from public service delivery.
- **MConnect** - a governmental interoperability platform allowing data exchange between the governmental information systems.
- **Semantic Catalogue** - a governmental platform representing a register of all semantic assets of the Republic of Moldova and rules and rights to access them.
- **MPass** - a platform service used for user authentication and access control to MDoc user interface.
- **MSign** - a platform service used for the application and validation of the electronic signature in MDoc business processes.
- **MNotify** - a platform service used as a mechanism for notification of MDoc authorized users or those seeking technical support.
- **MLog** - a platform service used to log and audit critical MDoc business events (highly important to the actors of the IS or with future legal effect).
- **MPower** - a register of use authorizations intended for validation of certain actions of MDoc authorized users.
- **MWallet** - a governmental service intended for storing and sharing documents issued by the Moldovan public authorities.
- **MDelivery** - a governmental service intended for physical delivery of the results of public or private service delivery to individuals and businesses.
- **MCabinet** - a governmental portal that could be alternatively used by Document Recipients to access all the documents issued for them by the Moldovan public authorities.
- **State Register of Population** - a state register containing official identity data of individuals residing in the Republic of Moldova.
- **State Register of Legal Entities** - a state register containing official registration data of the legal entities registered in the Republic of Moldova.
- **Service Provider's IS** - the information solution of a Moldovan public authority which, following the public service requests, supports production of requested documents, their storage in MDoc, where they can be accessed and downloaded by Document Recipients. PSA's information systems which digitize the workflows of the public services provided by PSA also belong to this category of actors.
- **eArchive (e-Archives)** - an information solution for archiving documents (related to the national, department archives or the archives of a public authority) that will be integrated with MDoc in order to take over documents and associated metadata for further filing and archiving the documents issued by the Moldovan public authorities.

A1.2. MDoc Use Cases

Figure 3 describes the main use cases implemented in MDoc. Each top-level use case corresponds to an Agile Epic that will be broken down into several User Stories (depending on the peculiarities of the use case).

UC01: Use Personal Cabinet

This is a key MDoc use case that provides functionalities specific to the Document Recipient's space in MCabinet which will offer user interface elements based on APIs exposed by MDoc. The Document Recipient's personal space will provide the following functionalities:

- **UC01.1: Access document**. This use case provides the functionalities that allow individuals or businesses to access the documents intended for them. The MCabinet will provide the functionality to view all the documents issued for the Document Recipient with document search/filter, view and download functions. The functionalities allowing to access documents will be available both in the Document Recipient's Personal Cabinet and through a URL (made available to third parties by Document Recipient directly or via a QR code).
- **UC01.2: Sign/countersign document**. This use case allows Document Recipients to electronically sign/countersign documents. Any document that requires to be electronically signed by Document Recipient, regardless of the issuing public authority, will be accessible through the Document Recipient's Personal Cabinet. This use case, besides the signing functionality, will also provide the functionality to configure the signatories (list of persons/authorities that must countersign the document).
- **UC01.3: Share document**. This use case provides to Document Recipients the functionality to share and store in MDoc the documents issued by Moldovan public authorities. The sharing option involves providing the Document Recipient with the functionalities of defining the document sharing parameters. A document can be shared through a URL that is accessible to everyone who receives it, through a QR code that contains the URL and by defining a limited number of recipients who receive the right to access the document. The URL can be sent both directly by the Document Recipient and by MDoc by email based on the parameters defined by the Document Recipient.
- **UC01.4: Send document feedback**. This use case provides to Document Recipients the functionality to send feedback on the received document. The Document Recipient can evaluate level of satisfaction with the quality of the received services, or send messages, comments in case of detected deficiencies.
- **UC01.5: Access notifications**. This use case provides to Document Recipients access to all the notifications on the traceability processes of their documents stored in MDoc or on the security procedures they have to comply with. MDoc will provide, where appropriate, the functionality to access the source of the notification-generating event (e.g. Document Recipient's document, the electronic signature application form etc.)

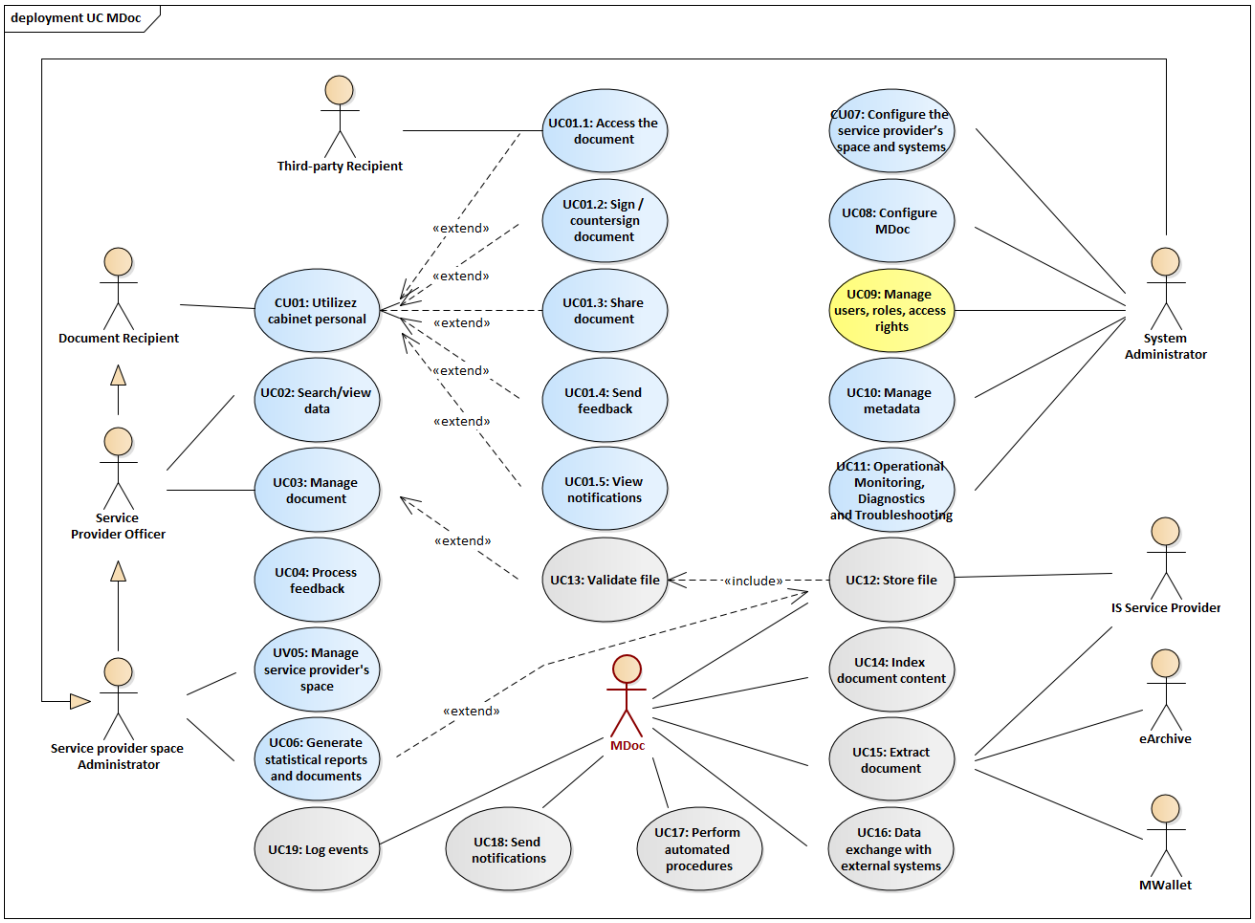

Figure 3. MDoc Use Cases

UC02: Search/View Data

This MDoc use case allows authorized users to explore the data store due to their roles in the information system and to their job-related duties.

MDoc will provide a data search mechanism using criteria such as:

- textual content of documents;
- document-related metadata;
- data related to authorized users;
- identification/registration data of Document Recipients;
- data related to public authorities;
- document status;
- etc.

SSI SPM will display the following search results:

- documents;
- document recipients;
- public authorities;
- authorized users;
- other specific targets.

For each result category MDoc will allow the following operations:

- for documents: view document; change document status; download document; edit document; define document recipients etc.;
- for document recipients: access document recipient profile; edit document recipient profile; view documents related to document recipients;
- for public authorities: access public authority profile; edit public authority profile; configure public authority's space in MDoc; view documents stored by public authority/shared with the public authority;
- for authorized users: access authorized user profile; edit authorized user profile; view documents prepared by authorized user.

MDoc will contain an indexed document search mechanism based on Apache Solr and will display the query results in order of their relevance.

UC03: Manage document

This complex use case will allow the Moldovan public authorities that do not have digitized workflows for public service delivery to access a user interface in order to store the documents related to the delivered public services in MDoc. The following document management functionalities will be put in place:

- filling out the form for uploading a document in MDoc (uploading document file and entry of its associated metadata);
- definition of recipients or actors who will countersign the document;
- modification of documents stored in MDoc (updating the document file and and/or related metadata);
- removal of a document stored in MDoc;
- other relevant functionalities identified during the business analysis stage.

Prior to activating the access of recipients to the document, the latter must be validated through UC13.

UC04: Process feedback

This use case provides the functionalities to view and process the Document Recipients' feedback on the quality of delivered public services. Besides the feedback indicating level of satisfaction, the public service providers will receive from Document Recipients through the feedback mechanism reports of errors detected in documents. These errors will further be fixed and the documents will be redone.

Based on the Document Recipients' feedback, MDoc must be able to rate the public service providers.

UC05: Manage service provider's space

This complex use case allows the Moldovan public authorities to manage their spaces in MDoc. The management of the public provider's space includes the following activities:

- management of credentials and access rights of users with Service Provider roles;
- configuration of the types of documents to be stored in MDoc and the access for third-party information systems to documents (if appropriate);
- configuration of templates of documents based on which printable versions of structured documents will be generated (in XML or JSON format);

- configuration of the metadata set additional to the mandatory ones defined in MDoc related to the documents to be stored in MDoc;
- other options for managing the space of the public service provider identified during business analysis stage.

UC06: Generate statistical reports and documents

This functionality is accessible to MDoc authorized users and enables generation of structured documents uploaded in MDoc (received in XML or JSON format) in printable format (PDF), as well as pre-defined and ad-hoc reports on the information content of the information system and the activity of authorized users.

The statistical reports generated by MDoc are useful for the analysis of the information basis of the information system and the performance of the authorized users in particular and their entities, allowing to extract specific performance indicators required for the analysis of document storage/use processes in MDoc.

The information system should integrate a solution for the configuration and generation of reports (report generator) to be reused also for the configuration and extraction of standardized documents specific to MDoc business processes.

UC07: Configure the service provider's space and systems

This is a complex use case which allows the MDoc System Administrator to configure the space reserved for public authorities for uploading the documents requested by individuals and businesses.

Configuration of this space involves defining the basic space parameters (e.g. public authority, information systems that will upload/download documents, the size limit of the space/stored documents, types of uploaded documents, validation schemes for uploaded files etc.) and the access credentials for the information systems and administrators of public authorities.

UC08: Configure MDoc

This use case provides the functionalities to configure the MDoc operating parameters. It should be noted that MDoc must be a configurable system and its adaptation to users' current needs and the legal framework should be made through the user interface without requiring an intervention in the program code, its compilation and repeated deployment of the information system.

System Administrator must be able to define at least the following configurations:

- jobs for automated procedures;
- access paths, values of the variables required for MDoc operation;
- workflows;
- templates of documents and reports;
- rules for file content, type and size validation;
- file retention periods (according to the types of documents and their sources);
- rules for indexing file content;
- integration parameters with external information systems;
- configurations related to the information security management system;
- other relevant configurations.

UC09: Manage users, roles, access rights

This is a use case outside MDoc that provides System Administrators with functionalities to manage roles and the associated rights to be further assigned to MDoc authorized users.

All the MDoc authorized users (Service Provider Clerk, Service Provider's Space Administrator, System Administrator or other roles defined during MDoc operation) will be managed through the platform service MPass. The access rights to the user interface and the database records will be configured for each role or explicitly for each user. These configurations will consider the specifics of user (e.g. the service provider's space to which the user has access).

UC10: Manage metadata

This is a MDoc use case that allows the management of MDoc's metadata system. The following metadata types are likely to require managing:

- Official national classifications (e.g. Classification of Organizational-Legal Forms, Classification of Property Types, Classification of Administrative-Territorial Units of Moldova, Classification of Moldovan Economy Activities etc.);
- Classifications and nomenclatures specific to the file storage and sharing business processes in MDoc;
- Interoperability classifications/nomenclatures (metadata set specific to data exchange with external information systems);
- MDoc's internal classifications/nomenclatures (e.g. user interface labels and messages in 3 languages).

All the metadata sets specific to MDoc will be identified at the business analysis stage. The metadata management mechanism will take into account the transfer of metadata values and the validity period of such values.

UC11: Operational monitoring, diagnostic and troubleshooting

This is a complex use case allowing MDoc administrative roles to access the functionalities to monitor MDoc operating parameters, diagnose and troubleshoot technical issues.

This use case will also provide functionalities to generate pre-defined and ad-hoc statistical reports on MDoc operating events. These reports are useful for the analysis of processes, informational basis of the information system, performance of authorized users and allow anticipating information security issues. Unlike UC06, UC11 is intended for the information audit processes aimed to support the information security mechanisms.

UC12: Store file

This is a complex use case that provides the API exposed by MDoc to the Service Provider's Information Systems (through MConnect) to upload in MDoc the files and associated metadata that resulted from public service delivery to Moldovan individuals or businesses.

Uploading a file involves going through a file validation procedure (through UC13) and, when files are provided in XML or JSON formats (through UC06), generating their printable equivalents (PDF).

UC14 will provide functionalities both for primary uploading of files and associated metadata and for updating metadata/replacing or removing files from MDoc.

UC13: Validate file

This is a complex use case which allows to validate all files before they are stored through MDoc. Validation involves several checks as follows:

- checking format of the file to be stored in MDoc for adequacy;

- checking size of the file to be stored in MDoc;
- checking the metadata related to the file to be stored in MDoc (whether all mandatory metadata have been entered, appropriate data are used, the value range is permissible etc.);
- checking the scheme used to produce the file to be stored in MDoc for accuracy (for the documents structured in XML and JSON format);
- other relevant criteria identified at the business analysis stage.

Only the files validated through UC13 will be stored in MDoc.

UC14: Index file

With this complex use case all files and associated metadata uploaded in MDoc will be indexed (to ensure a highly efficient document search procedure). The indexing solution may be Apache Solr.

Given the large number of files that will have to be stored in MDoc and that a large index therefore will be required, the index will contain only the files dating from a specific moment configured through UC08 (the old files will be regularly removed from the Index).

UC15: Extract file

This is a complex use case that provides the API exposed by MDoc to the Service Provider's Information Systems (through MConnect) to download the files one of the recipients of which is the public authority owning the Information System that provides services to citizens (when to provide a public service a public authority requests documents from other authorities).

A specific case of using UC15 are the digital archives that will access MDoc through UC15 for retrieving documents and associated metadata which must be filed and archived.

UC16: Exchange data with external systems

This use case will provide MDoc the functionalities to support the exchange of data with external information systems or to implement the functionalities provided by governmental platform services.

This data exchange refers to exposing or consumption of interfaces intended for mutual data exchange (receipt of data from external sources, sending data to external information systems and two-way data exchange).

Some of the integrations with external information systems (Service Provider's Information Systems, State Register of Population, State Register of Legal Entities, Semantic Catalogue, MCabinet, MDelivery, MWallet, e-Archives solutions) will be implemented through the interoperability platform MConnect. The platform services (MPass, MSign, MLog, MNotify, MPower) will be integrated directly through their APIs.

UC17: Perform automated procedures

This is a complex use case that supports automatic activation and operation of specific MDoc functionalities to ensure rational use of server resources, launch services specific to MDoc operation and supply data for authorized users when appropriate. It will be implemented through a configurable job manager used to configure all the automated procedures. Such procedures may include:

- Regular search index updating (removing expired files from the index);
- Archiving expired files (their removal from Document Recipients' personal cabinets;
- Generation of complex statistical reports requiring more time, which can be scheduled during the hours when MDoc is least used;

- Checking deadlines and automatic notification of relevant users required to take specific actions;
- Checking MDoc security issues which involves implementing automatic procedures for background monitoring of the work of authorized users. These procedures, based on user's behavior, will highlight suspicious activities (e.g. authentication at short intervals from geographically remote areas, etc.). MDoc will send notifications to System Administrator on these security alerts and in specific cases will perform automatic actions (e.g. block access).
- Creation of MDoc backups according to the continuity procedure put in place by the Client.

UC18: Send notifications

This use case supports the notification of Document Recipients and authorized MDoc users. All the notifications will be sent through the governmental notification services MNotify.

Authorized MDoc users will be able to access the notifications in their Personal Cabinets and will have direct access to the electronic document the business event of which has generated the notification.

MDoc will automatically generate and send to authorized users notifications related to any business event or traceability of business processes implemented in MDoc (e.g. receipt of a new document, document signing/countersigning, end of the document retention period etc.), as well as events related to the information security management system specific to MDoc.

UC19: Log Events

This use case will be used to log the business events generated by MDoc's functional components. Any event generated during the business processes implemented in MDoc will be logged and saved in the appropriate Database tables.

The logging mechanism will be based on the industry standards and good practices. The information system will provide the required functionalities to configure the business event logging strategy, including: the types of business events subjected to logging, logging calendar period (definite or indefinite) etc.

The critical or sensible business events will be additionally logged through the platform service MLog.

A1.3. MDoc Architecture

MDoc must provide a WEB interface accessible through a largely used Internet browser (MS Internet Exlorer/MS Edge, Mozila FireFox, Opera, Google Chrome or Safari). Functionally, this must be a reliable and scalable solution to support an increase both in the number of concurrent users and in the amount of managed information.

MDoc will be based on a multi-level service oriented architecture (which excludes the direct interaction of the application with the database) based on up-to-date WEB technologies. To ensure an adequate level of information security the information system must allow secure connections between the Client stations and the application server to guarantee the security of the information sent (using VPN connections and TLS/SSL sessions).

MDoc will be deployed and will operate on the government platform MCloud. MDoc must be designed, developed and implemented based on the architecture described in Figure 4.

The architecture shown in Figure 4 is high-level and indicative. It describes the main elements of MDoc and how they interact. At the stage of MDoc Concept development, the Consultant, jointly with the experts of the Client, will design a detailed architecture for MDoc. The Consultant will

start working on the development of MDoc only after the MDoc architecture solution is identified and approved.

As may be seen in Figure 4, the resource cooperation solution for ensuring proper operation of MDoc consists of 6 distinct types of nodes:

- **Client computers** from which authorized users will access MDoc functionalities (depending on their rights and roles).
- **ICT infrastructure of MDoc in MCloud** - ICT infrastructure of MDoc in the governmental cloud (MCloud) that will host MDoc.
- **Horizontal governmental services** - all the common horizontal governmental services MDoc will be integrated with in order to reuse certain platform functionalities (authentication, electronic signing, notification, logging, checking credentials, data exchange with external information systems etc.).
- **MConnect** – a governmental interoperability platform which will support data exchange between MDoc and external information systems.
- **PSA's ICT infrastructure** – the ICT infrastructure of the Public Services Agency hosting the State Register of Population and State Register of Legal Entities, which will provide relevant data related to the document recipients. Additionally, this infrastructure hosts service providing information systems which will store in MDoc the documents issued to individuals/businesses or will receive the documents issued by other service providing information systems necessary for the business processes of the service providing information systems owned by PSA.
- **Service Provider's ICT infrastructure** – the ICT infrastructure of a service provider (deployed in the common governmental platform MCloud or in the data center owned by the service provider) hosting the service providing information systems which will store in MDoc documents issued to individuals/businesses or will received documents issued by other service providing information systems necessary for the business processes of the service provider.

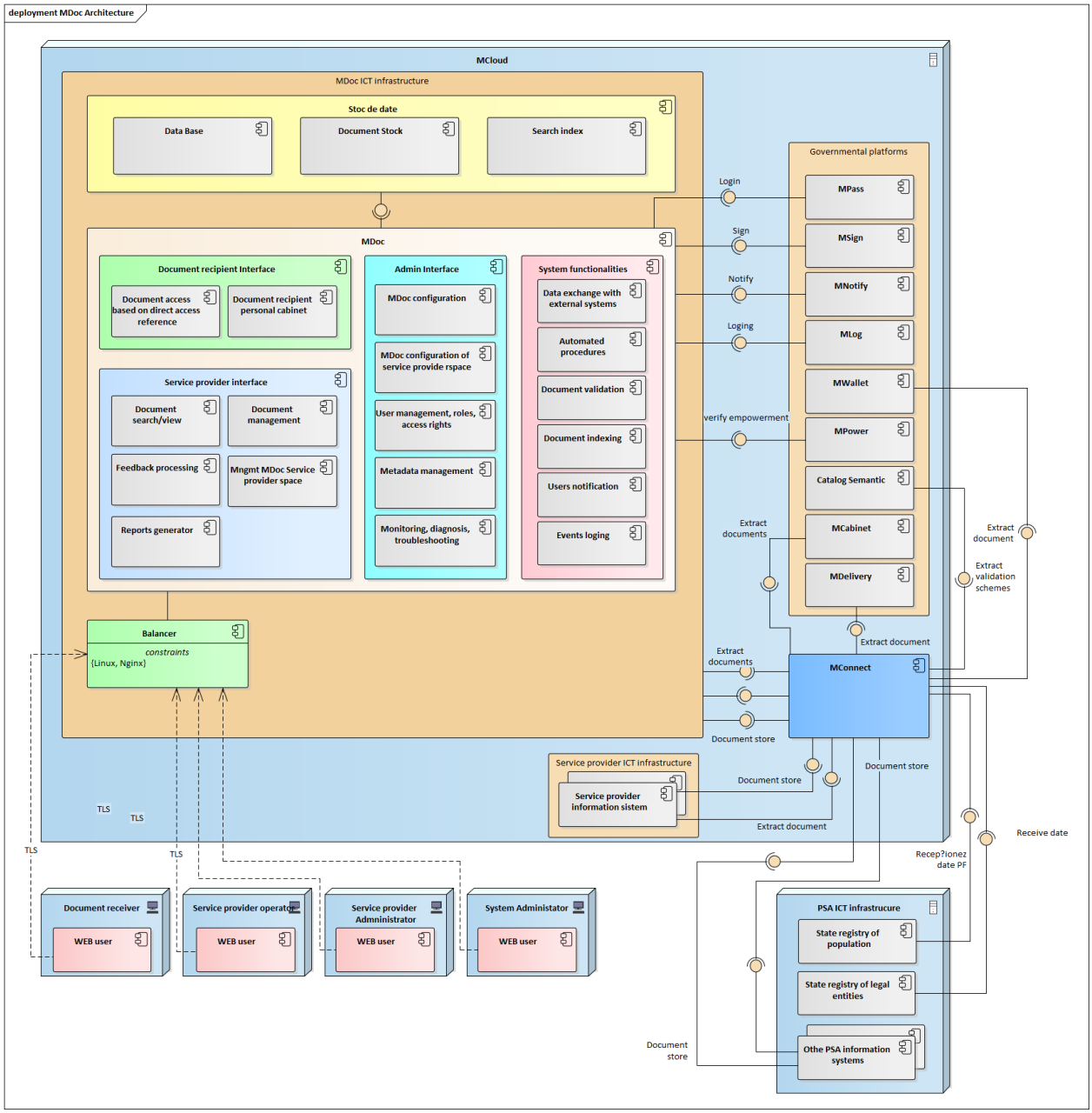

Figure 4. Indicative MDoc Architecture

As may be seen in Figure 4, MDoc has 6 key components:

- **Balancer** – a solution for balancing user requests to avoid MDoc server overloading and ensure the desired performance parameters of the information system.
- **Document Recipient Interface** – a functional component provided by MDoc to Document Recipients to access, view and download the documents provided by the service providing information systems.
- **Service Provider Interface** – a functional component of MDoc intended for the authorized users of the service providing public authorities (which do not have information systems to provide services). This functional component supports the upload of documents related to the services provided to individuals and businesses in MDoc by the service providing public authorities.

- **Administration Interface** – a functional component of MDoc intended for the authorized users with administrative roles, which supports the configuration, administration and monitoring of MDoc operation.
- **System Functionalities** – a functional component of MDoc which provides automatic routines to ensure efficient MDoc operation.
- **Data Store** – a component of MDoc supporting the storage of MDoc related data, including the database, the documents related to the delivered services and the search index.

To implement the required functionalities, MDoc will consume specific platform services and APIs provided by governmental and external information systems as follows:

- **Authenticate** – a service provided by the governmental platform service MPass to implement user authentication procedures.
- **Authorize** – a service provided by the governmental platform service MPower to check the credentials of the users authorized to take specific actions in MDoc.
- **Sign -** a service provided by the governmental platform service MSign to implement the mechanism for application/checking the electronic signature of files generated and stored in MDoc;
- **Notify -** a service provided by the governmental platform service MNotify to implement functionalities to notify authorized users about occurrence of specific business events.
- **Log -** a service provided by the governmental platform service MLog to log specific business events.
- **Receive Individual Data** provided by th**e** State Register of Population to support the receipt of identification data of the individuals receiving documents.
- **Receive Business Data** provided by the State Register of Legal Entities to support receipt of relevant data of the legal entities receiving the documents.

MDoc will expose the following services to the external information systems (through the interoperability platform MConnect):

- **Store document -** intended for the integration with the information systems of the service providing public authorities to support receipt and storage of documents related to the service (document files and associated metadata).
- **Extract document** – intended for the integration with the information systems receiving the documents stored in MDoc (document files and associated metadata).

Annex 2. Technical Requirements

This section defines the non-functional requirements that must be taken into account during the MDoc design, development, implementation and operation stages.

A2.1. Documentation Requirements

No.	Type of requirement	Description
1.	**Documents for end-users**	The Consultant must prepare and deliver the following documentation for MDoc end-users: 1. Interactive guide adjusted to user roles (Document Recipient, Third-Party Recipient, Service Provider Clerk, Service Provider's Space Administrator, System Administrator) 2. Downloadable user manuals in PDF format for all MDoc roles. All end-user documentation will be provided in Romanian and Russian.
2.	**Technical documentation**	The Consultant must prepare and deliver the following technical documentation: 1. Documentation for the MDoc application architecture, data architecture and technological architecture (including the description of models in UML language, which will include a sufficient level of detail of the implemented architectures). 2. Test strategy (including test scenarios). 3. Compilable and documented source code for applications, components and unit tests developed within the project . All technical documentation will be in Romanian.

A2.2. Training Requirements

No.	Type of requirement	Description
1.	**Training sessions**	The Consultant will deliver training at Client's premises and online training for the following roles: 1. System Administrator. 2. Service Provider Clerk. 3. Service Provider's Space Administrator. 4. MDoc Trainer (trainers to deliver further training).
2.	**Training materials**	The Consultant must prepare and deliver all the documents required for proper training of users.
3.	**Training language**	All training content/materials will be in Romanian.

A2.3. Property Rights

No.	Type of requirement	Description
1.	**Perpetual software license**	The Consultant must grant to the Client the unlimited/total right to run and further develop MDoc with all included software components with no constraints on time, location and functionality.
2.	**Redistribution rights**	The Consultant must grant to the Client the right to re-distribute the solution. While the Client does not intend to redistribute the solution on a large scale, it might happen that the software solution will have to be transferred to another state agency, for example due to a possible reorganization. The Client might also to re-locate the entire e-Government platform elsewhere.
3.	**Full data rights**	The Client keeps full rights on the data created by means of the solution.
4.	**Open data format**	MDoc shall keep data in an open format or provide mechanisms to extract data from the system in an open format thus enabling the transfer/migration of data to another information system.

A2.4. Architecture Requirements

No.	Type of requirement	Description
1.	**Open standards**	MDoc architecture must be based on relevant open standards and shall not use proprietary standards. MDoc architecture must be conceptualized using an integrated vision, based on the good practices of the ICT industry.
2.	**Service-oriented architecture**	MDoc must be based on a Service Oriented Architecture. It must be a modular architecture based on reusable components and abstract interfaces to support the implementation of a multilevel architecture with a clear delimitation of architectural levels. The system components must be relatively independent and interact with each other through dedicated interfaces.
3.	**Hosting environment**	MDoc must not include any hardware component and, once ready, will be deployed on the governmental cloud environment (MCloud).
4.	**Running environment**	MDoc must run on Docker container engine and shall not depend on specific host OS instance. Building container images shall be automated (refer to the following link for details: https://docs.docker.com/develop). When running in a container-based environment, the application must be elastic, including when application container instances are added /removed (above minimum required instances for HA); changing the system parameters

No.	Type of requirement	Description
		and configurations shall not affect the work in progress, such as active sessions, queries, etc.
5.	**Multiple sites**	MDoc architecture must ensure high availability, including when new versions are deployed, and the possibility to run simultaneously on multiple sites.
6.	**WEB browser compatibility**	MDoc must be compatible with the latest two major versions (to be considered at the time of system acceptance) of the following web browsers: Chrome, Safari, FireFox and Edge.
7.	**Detailed data model**	MDoc detailed data model must be described fully in a an electronic data scheme (e.g. using DDL language for relational databases). The Consultant shall agree with the Client in advance on the detailed format of the data model structure.

A2.5. Integration Requirements

No.	Type of requirement	Description
1.	**Integration with governmental platform services**	MDoc must integrate with the following governmental platform services: • *MPass* – for user authentication; • *MSign* - for the digital signing of files stored and shared through MDoc; • *MNotify* - for user notification about MDoc generated events; • *MLog* - for logging actions/business events and extraction of statistics on the use of services exposed by MDoc. • *MPower* - for authorizing actions in MDoc. • *MConnect* - for data exchange with external information systems. • *Semantic Catalogue* - to receive relevant metadata and validation schemes for the files stored in MDoc. • *MCabinet* - to allow recipients to access and download files through their space in MCabinet. • *MDelivery* - to send files through the channels provided by the governmental service MDelivery. • *MWallet* - to store and share data related to documents with the help of a QR code. • *Open Data Portal* - to submit usage statistics.
2.	**Integration with external systems**	MDoc must allow the integration with the following external information systems:

No.	Type of requirement	Description
		• *State Register of Population* to receive the identification data of individuals receiving the documents stored in MDoc. • *State Register of Legal Entities* to receive registration data of businesses receiving documents stored in MDoc. • *eArhivă* to provide documents stored in MDoc and associated metadata required for electronic archiving of documents. • *Service Providing IS* to receive documents and associated metadata that need to be stored in MDoc and send documents stored in MDoc that are necessary for the business processes implemented in these information systems.

A2.6. Performance Requirements

No.	Type of requirement	Description
1.	**Asynchronous processing**	MDoc must use asynchronous processing whenever possible to perform input-output.
2.	**Concurrent users**	The system standard load and performance shall be guaranteed for 5000 concurrent human users and external information applications.
3.	**Response time**	The response time for MDoc functions must be less than 3 (three) seconds. The Consultant shall list the exceptions, if any, and discuss/agree them with the Client during the analysis and design stages.
4.	**Performance indicators**	MDoc must meter and expose key performance indicators in the System Administrator interface. The Consultant shall provide a list of proposed indicators and discuss/agree them with the Client. The Consultant shall perform internal MDoc performance tests in accordance with the performance test plan (stress and load testing) agreed with the Client. Upon final acceptance, the Client will independently perform the MDoc performance testing and will send the test results to the Consultant for information or to fix the detected performance issues.

A2.7. User Interface Requirements

No.	Type of requirement	Description
1.	**Multilanguage interface**	MDoc must support a multilanguage user interface. This will include formats specific to data types (such as date, time, time spans, currencies, etc.).

No.	Type of requirement	Description
		The Document Recipient's interface will be delivered in Romanian, Russian and English languages. The Service Provider's interface will be delivered in at least Romanian and Russian languages. The MDoc administration interface will be delivered in at least Romanian language. The default language for the end-user must be Romanian.
2.	**User interface accessibility**	User interface must be compliant with at least Level AA of Web Content Accessibility Guidelines 2.1. https://www.w3.org/TR/WCAG21/
3.	**Responsive/Adaptive design**	User interface must automatically adapt to various display resolutions of MDoc users' devices. Minimal display width must be 480px. MDoc user interface shall be implemented using progressive web application (PWA) technologies and shall be functional on mobile and touch devices.
4.	**Contextual help**	MDoc user interface elements shall include Tips and Hints for the components intended for authorized users.
5.	**User support**	All pages must include user support contacts. MDoc must additionally integrate with other information system to allow sending technical support requests.
6.	**Bookmarks**	All major MDoc pages must be bookmarkable and the User shall be able to access bookmarked pages later. The bookmarkable pages will be defined at the analysis stage.
7.	**Friendly URLs**	MDoc must use friendly URLs for accessing its components.

A2.8. Maintenance Requirements

No.	Type of requirement	Description
1.	**System logs**	MDoc must log its various actions and events in a structured manner. Logging shall be configurable and based on extensible logging framework (such as log4net, nlog, etc.). The logging framework shall as a minimum support JSON format and the following targets: console, rolling files, UDP and HTTP POST.
2.	**Logging levels and event log records**	MDoc must differentiate the events and actions it logs by at least the following levels: Critical, Error, Warning, Info, Debug. Critical and Error level events shall be logged only for unrecoverable errors that require human intervention. Event log records will include at least: • the type of the event; • timestamp when the event took place; • event level;

No.	Type of requirement	Description
		• the system component that generated the event; • user/user agent, IP that triggered the event; • the identifier of the affected information object; • textual details of the event.
3.	**Graceful shutdown**	MDoc must implement graceful shutdown, i.e. shutting down an application container instance must not impact any work in progress, such as active sessions, queries, event logs, etc.
4.	**Source code**	The Consultant must supply the source code for MDoc components that are not available as COTS from third parties. The source code must use package managers for dependencies to 3rd party libraries. All prerequisite software must be part of container image definition and based on the public container repository.
5.	**Deployment**	The Consultant must supply the deployment procedure and related supporting tools. The deployment procedure must meet all the prerequisites before proceeding to system installation. The deployment shall be automated and include database initialization.
6.	**System upgrades**	System upgrades must be automated, including database upgrade/downgrade scripts or code. To enable rolling upgrades in the production environment, the recommended practice is to perform minor changes in database.

A2.9. Security Requirements

No.	Type of requirement	Description
1.	**Secure architecture**	MDoc must have a secure design and meet the requirements specified in GD 201 of 28.03.2017 (https://www.legis.md/cautare/getResults?doc_id=98644&lang=ro). The Consultant must supply the documentation describing the design and supporting evidence that such a design is secure. The Consultant will coordinate with the Client the format of the documentation, supporting evidence and the list of requirements that must be met.
2.	**Least privilege principle**	MDoc components must be based on the least privilege principle and run under such a limited privilege account under the OS rights model. The documentation must specify the required privilege level for each of the MDoc components and the considerations that force use of that level of access.
3.	**Secrets and addresses**	Secrets (passwords, private keys and certificates, connection strings, etc.) and addresses of external services must be clearly

No.	Type of requirement	Description
		indicated in the configuration documentation and easily modifiable using automated scripts.
4.	**Secure communication channels**	All MDoc communication with external systems or users shall take place through encrypted communication channels.
5.	**User authentication**	MDoc shall use MPass as user authentication mechanism.
6.	**Minimized storage of personal information**	MDoc shall minimize the amount of personally identifiable information stored and comply with the personal data processing requirements specified in GD 1123 of 14.12.2010 (https://www.legis.md/cautare/getResults?doc_id=16012&lang=ro). The Consultant shall coordinate with the Client the list of personal data protection requirements that must be met.
7.	**Security checks against vulnerabilities**	MDoc shall include security checks for all its components against at least OWASP Top 10 vulnerabilities. Please refer to https://owasp.org/Top10/
8.	**Health-check API**	MDoc must expose readiness and health-check API via HTTP GET requests. The health-check shall check the health of as many system components as possible. In case of a health check error, a human-readable error message will be sent.
9.	**Role management**	MDoc will have a user role management functionality to allow adding, removing or configuring roles.
10.	**Session timeout**	MDoc must include a session expiration mechanism requesting users to log in again after a period of lack of actions. The session timeout must be configurable and by default will be set to 15 minutes.
11.	**Input validation**	All input data must be validated both by client and server.
12.	**User content**	User content can be captured in text format only. MDoc must forbid entry of special characters used to format and mark special Web content. It must be possible to enter/view all UNICODE characters in MDoc components.
13.	**Unauthorized access attempts**	When MDoc registers unauthorized access attempts it must: • log such attempts with at least ERROR level; • send users with a warning message that access is not authorized and that abuse will be investigated.
14.	**Data integrity**	The Consultant must ensure data integrity by providing an appropriate solution for preventing unauthorized internal activities (for ex. deletion of email message directly from database).

A2.10. Support and Warranty Requirements

No.	Type of requirement	Description
1.	**Support**	During the warranty period the Consultant must provide necessary technical assistance to the Client (12 months after MDoc rollout)
2.	**Warranty**	During the warranty period the Consultant must: • fix all defects reported by the Client; • address all incidents reported by the Client in accordance with the agreed SLAs; Note: The response and resolution time shall not exceed 24 hours for non-critical errors and 4 hours for critical errors. Incidents must be addressed in 4 business days for non-critical errors and 2 business days for critical errors since the moment of escalation. Hourly progress reports will be provided for critical errors.

A2.11. Quality Requirements

No.	Type of requirement	Description
1.	**Key activities**	The Consultant must organize the acceptance testing of the information system. The testing must be carried out after the completion of each iteration (if applicable) and upon MDoc final acceptance. To this end, the Consultant must carry out at least the following activities: • define the test strategy and procedure; • prepare detailed test plans, including test scenarios; • receive requests on errors and fix them; • prepare the plan with the final test results, including the status of all detected errors. Test unit coverage for MDoc capabilities will be at least 90%.
2.	**Deliverables**	The Consultant must deliver the acceptance test plan to Client for coordination and acceptance. The Consultant must deliver to Client for coordination and acceptance the test scenarios for all types of functional tests (unit testing, integration testing, system testing, acceptance testing) and non-functional tests (security testing, performance testing, usability testing, compatibility testing.). The Consultant must deliver to Client for coordination and acceptance the report on MDoc test results.
3.	**Acceptance criteria**	The Consultant must perform jointly with the Client all the tests planned according to the Test Plan. The final test results will require acceptance by the Client.

No.	Type of requirement	Description
		The test results will be accepted when zero critical nonconformities and less than 3 major nonconformities are detected. Acceptance of test results will be dated with the day when all non-conformities detected upon delivery are addressed. MDoc acceptance protocol must be signed by both the Consultant and the Client.

A2.12. Acceptance of Project Deliverables

No.	Type of requirement	Description
1.	**Submission of deliverables**	The Consultant must submit the deliverables to the Client's authorized persons at least 2 days before their expected acceptance.
2.	**Examination of deliverables**	The Client will review the received deliverable, will prepare the objections indicating the reason of the objection, or will sign the deliverable (if deliverable is accepted with no objections). If the deliverable is rejected or returned with non-conformity objections, the Client must identify the issues or non-conformities which require fixing by the Consultant.
3.	**Examination and resolution of nonconformities**	At no cost to the Client, the Consultant shall examine and address all the indicated non-conformities and re-submit the deliverable no later than five (5) business days from the receipt of objections. The Client shall either accept or reject the resubmitted deliverables within five business days. Deliverables shall be deemed accepted when signed by the Client. If the Client neither accepts nor rejects the deliverables within the specified timeframe, the Consultant shall escalate the lack of response according to the escalation provisions defined in the Contract.

Anexa 3. Legislation regulating the business processes and procedures to be automated with MDoc

The analysis of the Moldovan legislation and international practice has revealed a number of laws and regulations, standards and good practices which must be taken into account when designing, developing and implementing MDoc.

More specifically, the laws and regulations include:

1. Law no. 982 of 11.05.2000 on access to information, Monitorul Oficial no. 88-90 of 28.07.2000;
2. Law no. 1069 of 22.06.2000 on informatics, Monitorul Oficial no. 73-74 of 05.07.2001.
3. Law no. 467 of 21.11.2003 on computerization and state information resources, Monitorul Oficial no. 6-12 of 01.01.2004;
4. Law no. 71 of 22.03.2007 on registers, Monitorul Oficial no. 70-73 of 25.05.2007.
5. Law no. 241 of 15.11.2007 on electronic communications, Monitorul Oficial no. 51-54 of 14-03-2008.
6. Law no. 133 of 08.07.2011 on personal data protection, Monitorul Oficial no. 170-175 of 14.10.2011.
7. Law no. 91 of 29.05.2014 on electronic signature and electronic document, Monitorul Oficial no. 174-177 of 04.07.2014.
8. Law no. 142 of 19.07.2018 on data exchange and interoperability, Monitorul Oficial no. 295-308 of 10.08.2018.
9. Government Decision no. 1123 of 14.12.2010 on the approval of Requirements for ensuring personal data security during their processing in personal data information systems, Monitorul Oficial no. 254-256 of 24-12-2010.
10. Government Decision no. 546 of 20.07.2011 on the approval of the Regulation on provision of services of the telecommunications system of the public administration authorities and amendment of certain Government decisions, Monitorul Oficial no. 118-121 of 22.07.2011.
11. Government Decision no. 7104 of 20.09.2011 on the approval of the Strategic Program for Technologic Modernization of Governance (e-Transformation), Monitorul Oficial no. 156-159 of 23.09.2011
12. Government Decision no. 656 of 05.09.2012 on the approval of the Interoperability Framework Program, Monitorul Oficial no. 186-189 of 07.09.2012.
13. Government Decision no. 1090 of 31.12.2013 on the electronic government authentication and access control service (MPass), Monitorul Oficial no. 4-8 of 10.01.2014.
14. Government Decision no. 128 of 20.02.2014 on the joint government technological platform (MCloud), Monitorul Oficial no. 47-48 din 25.02.2014.
15. Government Decision no. 405 of 02.06.2014 on the electronic government integrated digital signature service (MSign), Monitorul Oficial no. 147-151 of 06.06.2014.
16. Government Decision no. 700 of 25.08.2014 on open government data, Monitorul Oficial no. 256-260 of 29.08.2014.
17. Government Decision no. 708 of 28.08.2014 on the government electronic logging service (MLog), Monitorul Oficial no. 261-267 of 05.09.2014.
18. Government Decision no. 201 of 28.03.2017 on the approval of minimum mandatory cybersecurity requirements, Monitorul Oficial no. 109-118 of 07.04.2017.

19. Government Decision no. 414 of 08.05.2018 on measures to consolidate data centers in the public sector and rationalize administration of state information systems, Monitorul Oficial no. 157-166 of 18.05.2018.
20. Government Decision no. 211 of 03.04.2019 on the interoperability platform (MConnect), Monitorul Oficial no. 132-138 of 12.04.2019
21. Government Decision no. 375 of 10.06.2020 on the approval of the Concept of the automated information system „Register of representation powers based on electronic signature" (MPower) and Regulation on keeping the Register of representation powers based on electronic signature, Monitorul Oficial no. 153-158 of 26.06.2020.
22. Government Decision no. 376 of 10.06.2020 on the approval of the Concept of the government electronic notification service (MNotify) and Regulation on the operation and use of the government electronic notification service (MNotify), Monitorul Oficial no. 149-151 of 19.06.2020.
23. Government Decision no. 822 of 11.11.2020 on the approval of actions required after the inventory of existing state information systems and resources and amendments to certain Government decisions, Monitorul Oficial no. 304-312 of 20.11.2020.
24. Government Decision no. 152 of 25.08.2021 on the approval of the Concept of the government delivery service (MDelivery), Monitorul Oficial no. 212-218 of 10.09.2021.

While designing, developing, implementing and using MDoc the following ICT industry standards and good practices should be kept in mind:

1. Technical Regulation RT 38370656-002:2006 „Software Life Cycle Processes", approved through the order of the Ministry of Information Technology and Communications no. 78/2006.
2. The Standard of the Republic of Moldova SMV ISO CEI 15288: 2009, System and Software Engineering. System Life Cycle Processes.
3. SM ISO/CEI 12207 System and Software Engineering. Software Life Cycle Processes.
4. SM ISO/CEI 14764:2015 – Software Engineering. Software Life Cycle Processes. Maintenance.
5. SM ISO/CEI 27002 Information Technology. Security Techniques. Code of Good Practice for Information Security Management.
6. SM ISO/CEI 15408-1 Information Technology. Security Techniques. Information Technology Security Evaluation Criteria. Part 1: Introduction and General Model.
7. SM ISO/CEI 15408-2 Information Technology. Security Techniques. Information Technology Security Evaluation Criteria. Part 2: Functional Security Requirements.
8. SM ISO/CEI 15408-3 Information Technology. Security Techniques. Information Technology Security Evaluation Criteria. Part 3: Security Requirements.
9. Michael O. Leavitt, Ben Shneiderman, Research-Based Web Design & Usability Guidelines, https://www.usability.gov/sites/default/files/documents/guidelines_book.pdf
10. World Wide Web Consortium (W3C) Recommendations on the quality of webpage content, possibilities to properly view information, using largely-used Internet browsers and compatibility with various information platforms (http://www.w3c.org).
11. WAI (Web Accessibility Initiative) Recommendations to ensure accessibility to the website resources for the people with disabilities (http://www.w3c.org/WAI).
12. WCAG (Web Content Accessibility Guidelines) Recommendations http://www.w3.org/TR/WCAG21/