**Modernization of Government Services**

**in the Republic of Moldova Project**

**Project ID No. P148537**

# Terms of Reference

**CONSULTING SERVICES FOR THE DESIGN, DEVELOPMENT, CONFIGURATION, DEPLOYMENT OF THE MPOWER INFORMATION SYSTEM**

## Background

The Government of Moldova is determined to fundamentally change the way 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. Therefore, one of the main objectives of the Administration Reform Strategy 2016-2020 is the modernization of public services.

The Government of Moldova, via e-Governance Agency (EGA) as implementing agency is carrying out a World Bank-funded PAR operation, planned for 2017 - 2023- Modernization of Government Services Project – (MGSP) [[1]](#footnote-2). The e-Governance Agency is responsible for development and implementation of Government e-Transformation Agenda, which intends to ensure country’s sustainable development through efficient use of ICT. EGA was established in August 2010 on behalf of State Chancellery to manage e-transformation activities at governmental level.

The Government of Moldova, in accordance with the Public Administration Reform (PAR) Strategy for the years 2016-2020 (especially the component "Modernization of Public Services") is undergoing a major transformation exercise (qualitative and quantitative) of administrative public services, provided by central public administration authorities through: a) removing outdated public services or merging several services in one; b) increasing access to local public services through various channels; c) reducing the number of documents required for public services, and the delivery time; e) ensuring a high level of satisfaction with the quality of government service delivery. There are service providers which provide services exclusively online (such as the e-reporting service at the National Social Insurance House, the National Health Insurance Company, the State Tax Service). To get an online service, some providers require beneficiaries to register their users and representatives and authorize them to carry out different actions in those systems (e.g. to create reports, to sign reports electronically, to send reports, etc.). Traditionally, this is done by submitting a written request from the name of the administrator of the legal entity, authorizing accountants, human resources managers, other individuals with the authorization to carry out certain activities on behalf of the legal person - authorizing party. Also, the service providers register the authorized parties in specialized registers of authorizations. However, these registers are used only at departmental level and do not facilitate users' experience with online transactions with public authorities.

In this context, the Electronic Authorization Registry (further referred to as MPower) is an effective solution to solve these problems by providing a safe, flexible and efficient mechanism for management of authorizations for individuals and legal entities, the possibility to verify the existence and validity of authorizations by interested parties, including public authorities of the Republic of Moldova.

The following entities are interested or will be involved in the development and operation of MPower:

* e-Governance Agency, as owner of MPower, is financing the project and is an active stakeholder during the information system implementation, launch and operation, including validation and acceptance of the delivered information solution.
* IT and Cyber Security Service (RO: Serviciul Tehnologii Informaționale și Securitate Cibernetică - STISC), as owner and technical administrator of MCloud, is responsible to provide all necessary infrastructure from MCloud in order to host MPower.
* Public authorities and institutions of all levels, as providers of public services and owners of information systems will integrate with MPower.

Based on the particularities of the processes to be implemented in MPower, the following roles can be defined:

* MPower Owner and Holder – Public Institution “e-Governance Agency”;
* MPower Registrars:
  + individuals, who will be able to create, revoke, accept or reject assigned authorizations;
  + administrators of legal entities, who will create, revoke, suspend authorizations;
  + public authorities or public service providers migrating their registered paper-based authorizations;
* MPower administrators.

## Objective of the Assignment

The Client is looking for an ICT consulting company to develop the MPower Information System with demonstrated experience in the design and development of similar complexity projects with local presence in Moldova to perform key client-facing activities, and to provide on-going maintenance and technical support.

## Scope of work and Development approach

The work will be broken down into 2 stages as listed below.

Stage 1: Development and Deployment of MPower core functionality;

Stage 2: Development and Deployment of MPower extended functionality;

**Stage 1**

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

The development of MPower 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 MPower development.

**Iterative development**

In contrast to waterfall software development approach, MPower 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 Client together with the Consultant.

**Agile development**

MPower development shall follow agile principles by allowing change and flexibility in implementation. Client will maintain the master list of generic requirements for MPower – *product backlog*, which consists of ordered functional and non-functional 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, the Consultant shall fix them without impacting the time schedule and at no additional costs. 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 SLAs.

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

**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 of MPower. Mainly 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 development team for answering to 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 at the end of each sprint for acceptance. The Client shall accept the working package or notify the Consultant of any defects within the duration of 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 a prompt reaction to them.

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 ensure the appropriate product development for MPower.

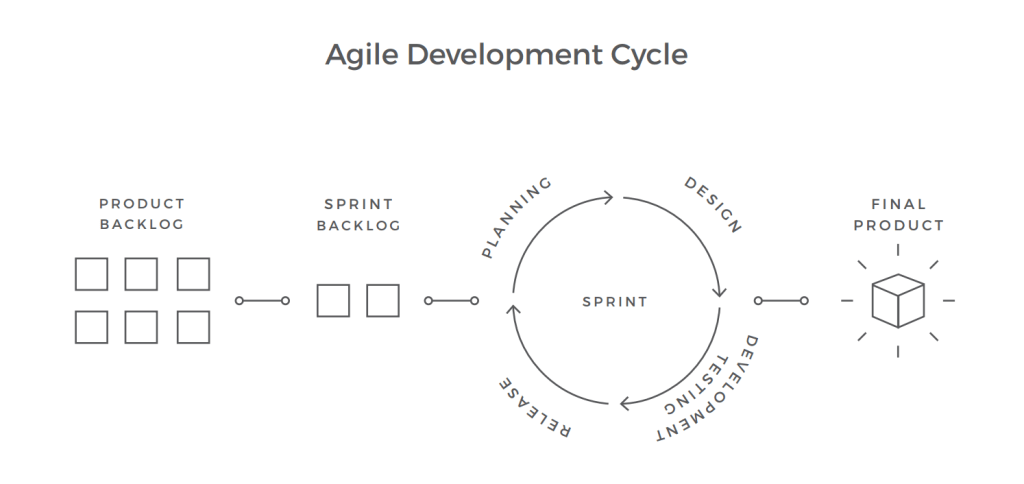


Figure 1. The indicative illustration of the Agile Development Cycle/Process

**Warranty**

The Consultant shall provide three (3) months of warranty for the developed solution. During the warranty period the Consultant shall fix any identified defects.

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

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 SQL Server 2016.
* Container engine is Docker.
* Container orchestrator is Kubernetes.
* Caching 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 these shall be added through amendments to the contract.

**Stage 2**

Stage 2 is conditional and will be implemented if during the exploitation of the service in production certain additional functionality will be identified and requested by end-users of MPower, or adjustments are needed due to eventual changes in the legislation. The implementation of Stage 2 shall be subject to successful implementation of Stage 1, Consultant’s satisfactory performance and budget availability. Prior to the implementation of Stage 2 the Consultant will be requested to develop a detailed proposal and the contract extended for a period of up to 6 months, based on the same fee rates.

The development defined under Stage 2 will follow agile iterative software development principles.

## Expected Deliverables

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

1. **A fully-fledged MPower** **Information System** with all core 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 the Annex 2.
3. **Training sessions and training materials** developed according to the Client’s training requirements defined in the Annex 2.

## Reporting Requirements

The following reports will be provided during the assignment:

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

1. Sprint Backlog, includingbreakdown 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;
2. Training reports, submitted after each training session, including:

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

## Timing

## The tasks defined under Stage 1 are estimated to be performed in 12 months – 9 months for development and 3 months of warranty period.

Subject to satisfactory performance, the contract can be extended for a second stage (Stage 2 defined under the Scope of Work section) based on the same fee rates. Stage 2 is estimated to be performed in up to 6 months.

## 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 management responsibilities and efficient coordination with stakeholders.

A Product Owner will be appointed by the Client, who will coordinate and decide on all issues related to the technical elements of the Contract. The Product Owner will be responsible for the issuance of the administrative notice for the start of the implementation of the contract and other duties having an administrative nature.

The Clientwill 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 will cover the costs for Azure DevOps subscription needed for its experts.

The **Consultant** will ensure that adequate working conditions (work space/office premises for the 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 day-to-day management of the project team and availability of all necessary resources.

The operational base for the project will be in Chisinau.

The communication languages will be Romanian and English.

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

## Qualification Requirements

##### Consultant qualifications requirements

The Consultant shall be a resident of the Republic of Moldova or have a permanent office in Moldova or undertake to set up such office for the duration of the contract and the warranty period. It will 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 Technology (IT) Systems, including software and services.
2. Has demonstrated experience in conducting at least two (2) contracts of similar complexity in the last three (3) years.
3. Has demonstrated experience in software development using agile technology.

***Note:*** For this section the term “similar complexity project” means a software development project of approximately 1200 man/days implemented using agile methodologies.

##### Staff qualifications requirements

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

* Key expert 1. Team Leader/Scrum master
* Key expert 2. Software Developer
* Key expert 3. Software Developer
* Key expert 4. Software Developer
* Key expert 5. Software Tester
* Key expert 6. Trainer

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

* **Key expert 1. Team Leader/Scrum master**:

The Team Leader is responsible for the day-to-day liaison with the Client; s/he must ensure the internal coordination and guidance of all experts of the project and coordination of the project with external counterparts.

The Team Leader must also ensure availability of suitable experts in accordance with the Project Work Plan.

The Team Leader 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 3 years of experience in software using required technology stack
  + Experience with unit testing, continuous integration, DevOps
  + Experience working as a team in software development using agile approach with other key experts is an asset
  + Certification in the required technology stack is an asset,
  + At least 7 years of experience in software development
  + At least 5 years proven tracked experience in team/project management using agile development approach, with at least 2 projects implemented in the last 3 years
  + Experience in system analysis
  + Ability to communicate in Romanian and English
  + At least three 3 successfully delivered projects of similar complexity.
* **Key Expert 2. Software Developer**:
  + University degree in Computer Science or another relevant domain
  + At least 5 years’ experience in software development
  + At least 3 years’ experience in software using required technology stack,
  + At least 2 projects implemented in the last 3 years using agile approach
  + Experience with unit testing, continuous integration, DevOps
  + Experience working as a team in software development using agile approach with other key experts is an asset
  + Certification in the required technology stack is an asset,
  + Proven experience in web UI design and development using responsive frameworks, progressive web apps
  + Experience in system analysis
  + Ability to communicate in Romanian and/or English
* **Key Expert 3. Software Developer**:
  + University degree in Computer Science or another relevant domain
  + At least 5 years’ experience in software development
  + At least 3 years’ experience in software using required technology stack,
  + At least 2 projects implemented in the last 3 years using agile approach
  + Experience with unit testing, continuous integration, DevOps
  + Experience working as a team in software development using agile approach with other key experts is an asset
  + Certification in the required technology stack is an asset,
  + At least 5 years’ experience in database design, development and optimization,
  + At least 3 years’ experience with Entity Framework
  + Experience in system analysis
  + Ability to communicate in Romanian and/or English
* **Key Expert 4. Software Developer**:
  + University degree in Computer Science or another relevant domain
  + At least 5 years’ experience in software development
  + At least 3 years’ experience in software using required technology stack,
  + At least 2 projects implemented in the last 3 years using agile approach
  + Experience with unit testing, continuous integration, DevOps
  + Experience working as a team in software development using agile approach with other key experts is an asset
  + Certification in the required technology stack is an asset,
  + At least 5 years’ experience in systems’ integration, API design and development
  + Experience in system analysis
  + Ability to communicate in Romanian and/or English
* **Key Expert 5. Software Tester**:
  + University degree in Computer Science or another relevant domain
  + At least 3 years’ experience in software testing in projects of similar complexity
  + Experience in software testing analysis and design
  + Experience in performance (load and stress) testing and security testing
  + Experience in automated testing
  + Ability to communicate in Romanian and/or English
  + Experience working as a team in software development using agile approach with other key experts is an asset
  + Certification in the required technology stack is an asset.
* **Key Expert 6. Trainer**:
  + University degree in Computer Science or another relevant domain
  + Experiences in conducting training sessions for end-users and IT specialists in at least 2 similar projects
  + Experience in writing technical and end-user documentation
  + Experience in on-line training development using Moodle e-learning system
  + Ability to communicate in Romanian
  + Knowledge of English would be an asset

***Note:*** For this section the term “similar complexity project” means a software development project of approximately 1200 man/days implemented using agile methodologies.

## Annexes

## Annex 1. MPower Core Functional Requirements

This Annex contains an indicative set of functional requirements reflecting the core functionality of MPower (Stage 1).

**MPower Actors**

The relationships between MPower actors depict cases when an actor can act in different capacities, for example an Identity may represent depending on context:

* an Authorizing Party where he/she assigns authorizations to an Authorized Party,
* an Authorized Party where another Authorizing Party assigned him/her an authorization,
* a Co-signer who co-signs an authorization when needed.



**Public User**

Any anonymous user.

**User**

Any authenticated user.

**Identity**

An abstract actor (individual, organization, group of people, etc.) that can act as an Authorizing Party, Authorized Party, Co-signer, Issuing Authority, Service Provider, or Governing Authority.

**Authorizing Party**

An Identity which can issue, accept, suspend or revoke Authorizations.

**Authorized Party**

An Identity that is authorized by an Authorizing Party.

**Co-signer**

A co-signing party (such as Witness, Guarantor, Board Member, etc.) which is required to sign an Authorization before it is considered effective, depending on Authorization Type configuration.

**Issuing Authority**

An Identity which can register Authorizations of certain types on behalf of an Authorizing Party.

**Service Provider**

Any Identity which consumes information from MPower in order to make decisions based on existing Authorizations.

**Governing Authority**

Any Identity which has the mandate to Suspend/Activate or Revoke an Authorization.

**Authorization Type Owner**

The User that can maintain Authorization Types.

**MPower Administrator**

The User that can perform MPower technical administration tasks.

##### 1.1.1 MPower main information objects and workflows

Authorization is the main information object of the system, representing a record about a right or a set of rights given by an identity (authorizing party) to another identity (authorized party), thus enabling the authorized identity with new powers.

An Authorization is of a certain Authorization Type with specific attributes and constraints and is usually issued for a limited period of time. An Authorization has a lifecycle, transitioning it through different states: drafted/deleted, created, signed, accepted/rejected, suspended/revoked, visualized and validated.

An interdiction is a record about an act of prohibiting or forbidding certain rights or actions for an identity (individual or legal entity). Conceptually, interdictions are the opposites of authorizations, although the functionality for them in MPower is very similar to that for authorizations.

Both authorizations and interdictions are meant to be used by service providers in their decisions while delivering their services.

The following diagram shows the Authorization states and transitions.



An Authorization is created by an Authorizing Party, although depending on Authorization type, it could be created (drafted) by any User on behalf of an Authorizing Party.

Each Authorization must be signed by the Authorizing Party.

Depending on Authorization Type, some Authorizations may need to be co-signed, for example by a Witness.

Depending on Authorization Type, Authorized Party could accept or reject an assigned authorization.

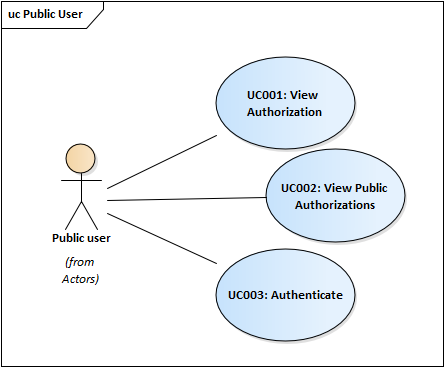
Only Authorizations signed by Authorizing Party, co-signed by Co-signers (if necessary) and accepted by Authorized Party (if necessary) are considered valid authorizations.

An Authorization could be revoked or suspended by Authorizing Party who issued this Authorization or by any Governing Authority.

Authorizations are considered as “Canceled” either when they are rejected or revoked.

##### 1.2 Functional Performance Requirements of the System

###### 1.2.1 Public User Requirements



UC001: View Authorization

Any Public User can view some of the authorization details based on authorization ID.

UC002: View Public Authorizations

Any Public User can view public authorizations, such as list of notaries, translators, bailiffs, etc.

UC003: Authenticate

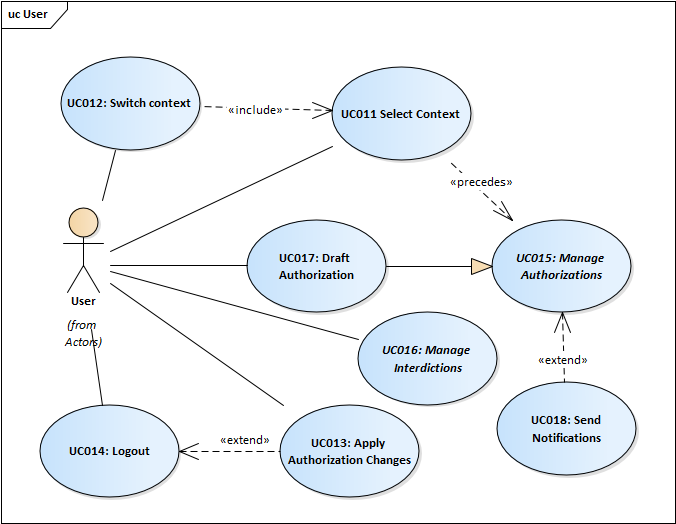
Any Public User shall be able to authenticate in MPower using MPass.

UC004: MStyle Compliance

The MPower will comply with MStyle user interface requirements.

##### 1.2.2 User Requirements

A User represents any form of Identity such as Authorizing Party, Authorized Party etc. This section specifies the requirements relevant to all authenticated Users.



UC011: Select context

The User shall be able to select the context he or she uses MPower.

UC012: Switch context

The User shall be able to switch context during an active session in MPower

UC013: Apply Authorization Changes

The User shall be able to submit new or changed authorizations by signing them electronically.

UC014: Logout

The User shall be able to logout from MPower

UC015: Manage Authorizations

Depending on its role, a User shall be able to perform authorization management actions such as: List and search, View the details of, Create, Revoke, Suspend/Activate, Acknowledge/approve, Request or Co-sign, etc.

UC016: Draft Authorizations

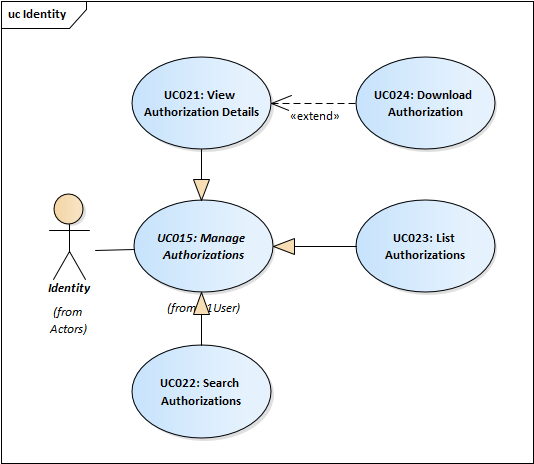
Any User shall be able to draft an authorization

UC018: Send Notifications

When managing authorizations, the User may receive relevant notifications.

###### 1.2.3 Identity Requirements

Identity is an abstract actor which may be any stakeholder involved in Authorization management. This section specifies requirements common to all Identities.



UC021: List Authorizations

A User shall be able to list authorizations.

UC022: Search Authorizations

A User shall be able to search for authorizations.

UC023: View Authorization Details

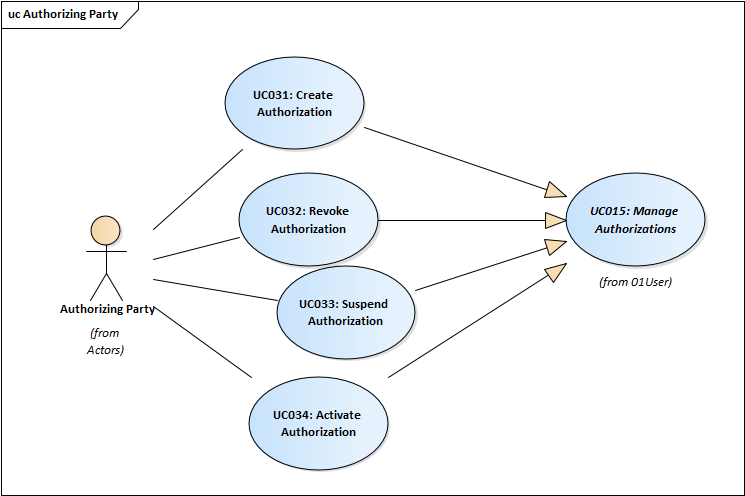
A User shall be able to view authorization details. As Authorizing Party the User shall be able to check the log of activities performed by the Authorized Party based on the authorization.

UC024: Download Authorization

A User shall be able to download an Authorization.

###### 1.2.4 Authorizing Party Requirements

This section describes the requirements regarding User representing an Authorizing Party.

UC031: Create Authorization

Authorizing Party shall be able to create personal or organizational authorizations. MPower shall implement nested authorizations, so assigning a higher level-authorization to an Authorized Party automatically adds other relevant lower level authorizations.

UC032: Revoke Authorization

Active authorizations can be revoked when necessary.

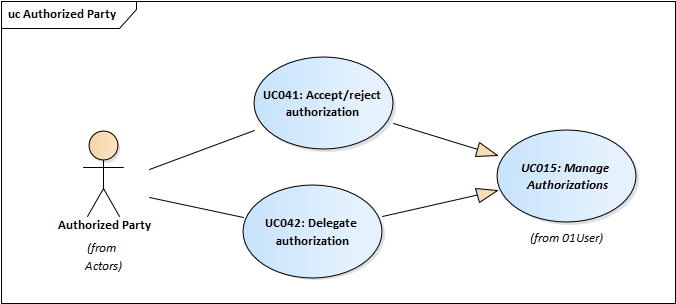
UC033: Suspend Authorization

Active authorizations can be suspended for a time period.

UC034: Activate Authorization

Authorizing Party shall be able to activate suspended authorization by referencing original authorization and Authorized Parties and confirming intention of activation.

###### 1.2.5 Authorized Party Requirements



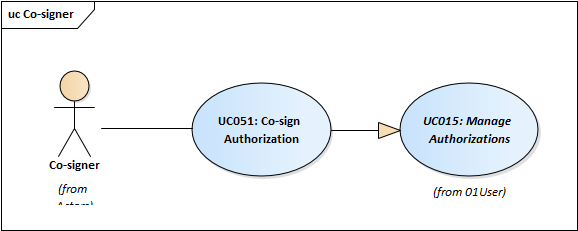
UC041: Accept/reject authorization

Depending on authorization type, some authorizations may require acceptance from the Authorized Party.

UC042: Delegate authorization

Depending on authorization type, some authorizations may be further delegated by Authorized Parties to other eligible Identities.

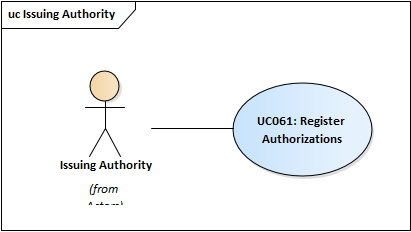
###### 1.2.6 Co-signer Requirements



UC051: Co-sign Authorization

Depending on their type, some authorizations may need signatures of third party co-signers in order to become effective.

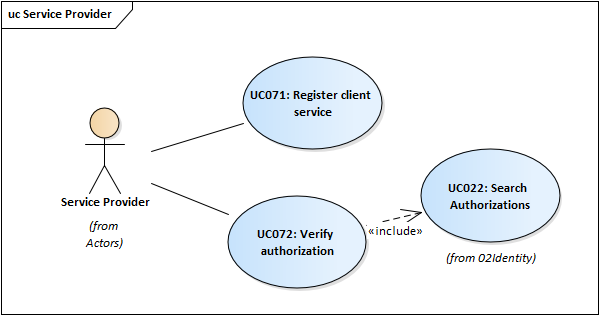
###### 1.2.7 Issuing Authority Requirements



UC061: Register Authorizations

MPower will consider existing authorizations issued by different identities before introducing the system. There are several authorities (e.g. Tax Authority, National House of Social Insurance etc.) which were collecting paper-based authorizations in the past, typically from legal entities.

###### 1.2.8 Service Provider Requirements



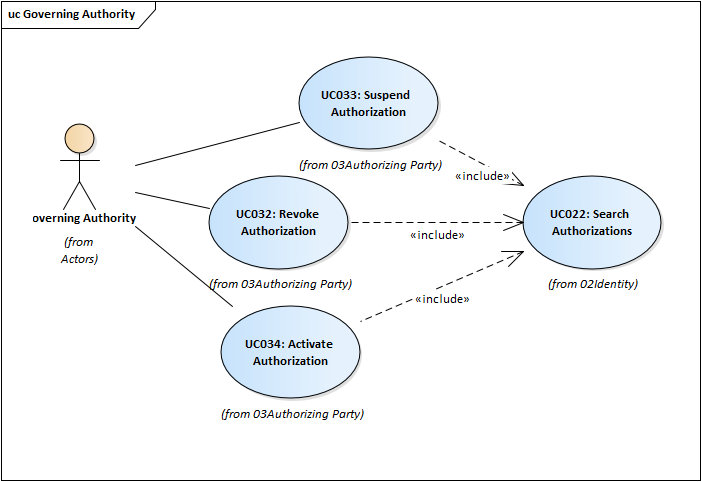
UC071: Register client service

Prior to using MPower for verification of authorizations, Service Providers will have to register their services as clients of MPower.

UC072: Verify authorization

When needed, Service Providers can verify the status of authorizations of configured authorization types.

###### 1.2.9 Governing Authority Requirements



UC081: Manage interdictions

An interdiction is an action of prohibiting or forbidding something for a subject (individual or legal entity).

Interdictions are conceptually the opposites of authorizations, although the functionality for them is very similar to that for authorizations.

###### 1.2.10 Authorization Type Owner



###### 1.2.11 MPower Administrator Requirements



UC101: Manage Authorization types

All MPower authorizations are created based on a registered authorization type.

UC102: Manage MPower parameters

MPower will use some global configuration parameters for its functions.

## Annex 2. MPower Non-Functional Requirements

##### Documentation requirements

|  |  |
| --- | --- |
| **User Documentation** | The Consultant will prepare and deliver the following documentation for end-users:   * + Interactive guidance included in user interface of MPower adjusted to user role (Authorizing party, Authorized party, etc.)   + Downloadable user manuals in PDF format for Authorizing party, Authorized party, etc.   + Downloadable user manual for MPower Administrators in PDF format   All end-user documentation will be provided in Romanian. |
| **How-To video tutorials** | The Consultant will prepare How-To video tutorials for MPower main functions (e.g. Create authorizations, Sign authorizations, Accept/reject authorizations, etc.). |
| **Technical documentation** | The Consultant will prepare and deliver the following technical documentation:   * System architecture documentation (including description of models in UML language, which will include a sufficient level of details of the system architecture) * Test strategy * Compilable and documented source code for applications, components and unit tests developed within the project * System installation and configuration manual (including code compilation, container image build scripts, system installation, hardware and software requirements, platform description and configuration, backup and disaster recovery procedures)   All technical documentation will be provided in English. |
| **API documentation** | The Consultant will prepare and deliver:   * API integration guide * Integration samples in .NET and Java * Human and machine-readable description in a standard description language (e.g. WSDL or Swagger). |

##### Training requirements

|  |  |
| --- | --- |
| **Training sessions** | The Consultant will provide on-line training sessions using developed e-learning modules for the following target groups:   * MPower Administrators from eGA and STISC. * Service Providers and Issuing Authorities representatives. * Governing Authorities representatives. |
| **Training materials** | * + Training documentation – curricula, training courses (manuals, video tutorials, quizzes, etc.) for administrators, services providers and end-users (individuals and businesses) developed in e-learning platform based on Moodle.   All training content/materials will be provided in Romanian. |

##### Rights requirements

|  |  |
| --- | --- |
| **Perpetual software license** | The Consultant grants to the Client the rights to run and use entire solution with all included software components with no constraints on time, location and offered functionality. |
| **Redistribution rights** | The Consultant shall grant to the Client the right to re-distribute the solution.  While the Client does not intend to re-distribute at a massive scale it still envisions the need to transfer the software solution to another state agency due for example to potential reorganization. Also, the Client might get the opportunity to re-deploy the entire e-Government platform elsewhere. |
| **Full data rights** | The Client keeps full rights on data created by the means of this solution. |
| **Open data format** | The solution preserves the data in an open format or includes mechanisms to extract data from the system in an open format thus enabling the capability to transfer/migrate the data into another system. |

##### Architecture requirements

|  |  |
| --- | --- |
| **Open standards** | The solution architecture shall be based on relevant open standards. The solution architecture shall not use proprietary standards. |
| **Service Oriented Architecture** | The solution shall be based on a Service Oriented Architecture. |
| **Hosting environment** | The solution shall not include any hardware components and will be deployed on governmental cloud environment (MCloud). |
| **Running environment** | System shall 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>)  Running in a container-based environment, the application must be elastic, including when adding/removing application container instances (above minimum required instances for HA), changing of configurations and system parameters has no impact on any work in progress, such as any active sessions, requests, etc. |
| **Multiple sites** | The solution architecture shall ensure high availability including during new versions deployment and the possibility to run simultaneously on multiple sites |
| **Browser compatibility requirements** | The system shall be compatible with latest two major versions (to be considered at the time of system acceptance) of following web browsers: Chrome, Safari, FireFox and Edge. |
| **API for integration with governmental platform services and third-party systems** | MPower shall expose API for core functionalities to be consumed by governmental platform services (at least for MPass, MCabinet and MAccess) and by third party systems.  For example, MPass must be able to retrieve the list of active authorizations of certain types for an Authorized Party, including the authorization details.  Third party systems must be able to check for certain authorization types, similar to “Verify authorization” use case.  The full list of logically applicable APIs and their format will be detailed during analysis and design stages. |
| **Detailed data model** | System's detailed data model shall be described fully in a machine-readable data scheme for example using a DDL language for relational databases.  The Consultant shall coordinate the detailed data model schema format with the Purchaser in advance. |

##### System Integration requirements

|  |  |
| --- | --- |
| **Governmental platform services integration** | * MPass shall be used to authenticate users * MSign shall be used to sign and verify electronic signatures * MLog shall be used to journal business critical events. The events that are business critical will be defined at analysis and design stages and must be configurable. * MNotify shall be used for notifications. |
| **Third party systems integration** | MPower will be integrated with state registries (e.g. SRLE, SRP, etc) and third-party information systems (e.g. Service Desk, Tax Authority IS, etc).  The integration with any registry or external information system will be implemented through MConnect. |
| **Open data integration** | MPower shall publish agreed sets of data to Open Data portal located at date.gov.md using its API. |

##### System Performance requirements

|  |  |
| --- | --- |
| **Asynchronous processing** | System shall use asynchronous processing whenever possible to perform any input-output. |
| **Concurrent users** | The system standard load and performance shall be guaranteed for 100 concurrent human users. |
| **Concurrent system requests** | The system shall be designed to respond (via API requests) to at least 100 concurrent external system requests. |
| **Response time** | Response time for system functions shall be under 3 (three) second. The Consultant shall list the exceptions, if any, and discuss/agree them with the Client at analysis and design stages. |
| **Daily transactions** | The system shall be designed to process at least 10000 transactions per day. |
| **Key performance Indicators** | The system shall meter and expose its key performance indicators. The Consultant shall propose the list of indicators and discuss/agree them with the Client. |

##### User Interface requirements

|  |  |
| --- | --- |
| **Multilanguage User Interface** | The system shall support multilanguage user interface. This support includes data type specific formats (such as date, time, time spans, currencies, etc.). The system will be delivered with at least Romanian and English interfaces. The default language shall be the Romanian. |
| **User Interface accessibility** | User interface shall conform at least to Level A of Web Content Accessibility Guidelines 2.0.  https://www.w3.org/TR/WCAG20/ |
| **Responsive/Adaptive design** | The system user interface shall automatically adapt to various display resolutions. Minimal display width is 480px.  The system’s UI shall be implemented using progressive web application (PWA) technologies and shall be functional on mobile devices. |
| **Contextual help** | User Interface elements shall include Tips and Hints for user interface elements. |
| **Client support** | All pages shall include client support contacts. |
| **Bookmarks** | All major MPower pages shall be bookmarkable and the User shall be able to access bookmarked pages later.  The bookmarkable pages will be defined at analyzing stage. |
| **Friendly URLs** | MPower shall use friendly URLs for accessing its pages. |

##### System maintenance requirements

|  |  |
| --- | --- |
| **System logs** | The system shall 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.). Logging framework shall minimally support JSON format and the following targets: console, rolling files, UDP and HTTP POST. |
| **Log levels and event log records** | The system shall differentiate events and actions it logs into at least following levels: Critical, Error, Warning, Info, Debug  Critical and Error level events shall be logged only for non-recoverable error that require human intervention.  Event log records will include at least:   * the type of the event * timestamp when the event took place * event level * system component that produced the event * user/user agent, IP that triggered the event * information object identifier affected * textual details about the produced event |
| **Graceful shutdown** | The system shall implement graceful shutdown, i.e. shutting down an application container instance at any time shall not impact any work in progress, such as any active sessions, requests, event logs, etc. |
| **Source code** | The Consultant shall supply all the source code for system components that are not available as COTS from third parties.  The source code shall use package managers for dependencies to 3rd party libraries. All prerequisite software must be part of container image definition and based on public container repository. |
| **System deployment** | The Consultant shall supply the deployment procedure and supporting tools for this. Deployment procedure shall cover all the prerequisites before proceeding to system installation. The deployment shall be automated and include database structure initialization and seeding. |
| **System upgrades** | System upgrades shall be automated, including database upgrade/downgrade scripts or code. To enable rolling upgrades in production environment, the recommended practice is to perform database breaking changes in incremental changes. |

##### Security requirements

|  |  |
| --- | --- |
| **Secure architecture** | The system shall be secure by design and comply with the relevant requirements specified in GD 201 from 28.03.2017 (<http://lex.justice.md/md/369772/>).  The Consultant shall supply documentation describing this design and supporting evidences that such a design is secure.  Note that the Consultant will coordinate with the Purchaser the format of the documentation, supporting evidence and list of requirements to comply with. |
| **Least privilege principle enforcement** | The system's components shall rely on the least privilege principle and run under such a limited privilege account under the OS rights model.  The documentation shall highlight each of the system's components required privilege level and considerations that force use of that level or access. |
| **Secrets and addresses** | Secrets (passwords, private keys and certificates, connection strings) and addresses of external services shall be clearly delineated in configuration documentation and easily modifiable via automated scripts. |
| **Secure communication channels** | All system's communication with external systems or users takes place over encrypted communication channels. |
| **No Username/Password authentication** | The system shall rely on authentication via MPass. Other forms of user authentication shall not be used. |
| **Minimize personal information storage** | The system shall minimize the amount of personally identifiable information stored. For example, there is no need to store a user's First and Second names since this will be provided after authentication by MPass.  The system shall comply with the relevant requirements related to personal data processing specified in GD 1123 from 14.12.2010 (<http://lex.justice.md/md/337094/>)  Note that the Consultant shall coordinate with the Purchaser the list of requirements to comply with. |
| **Secure against OWASP Top 10 vulnerabilities** | The system shall include security controls for all its components for at least OWASP Top 10 vulnerabilities. Refer <https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project> |
| **Health-check API** | The system shall expose readiness and health-check API via a HTTP GET requests. The health-check shall check the health of as many system components as possible. In case of health check error, a human-readable error message shall be returned. |
| **Users’ roles management** | The users and their roles will be managed in MPass. The system shall retrieve the users’ roles from MPass. |
| **Session expiration** | The system shall include a session expiration mechanism when after a specific period of inactivity, the user is required to authenticate again. The period of inactivity shall be configurable and by default it is 15 mins. |
| **Authorized access to personal content** | Users are granted access to content designated as belonging to them. Content belongs to a user if it has been assigned/addressed to their personal IDNP. |
| **Input validation** | All input data shall be validated on client and server side. |
| **User content** | User content can be captured in text format only. The system shall forbid entry of special characters used for formatting and markup of special Web content.  Otherwise all UNICODE characters shall be possible to enter/view by system's components. |
| **Unauthorized access attempts** | Unauthorized access attempts  When the system registers unauthorized access attempts it shall:   * log such attempts with at least ERROR level * provide users with a warning message that access is not authorized and that abuse will be investigated |
| **Data integrity** | The Consultant will ensure data integrity by providing appropriate solution for prevention of unauthorized internal activities (for ex. deletion of authorizations records directly from database).  Note that chaining the signed declarations in a blockchain could be a solution. |

##### Support and Warranty requirements

|  |  |
| --- | --- |
| **Support** | During the warranty period the Consultant shall provide necessary technical assistance to the Client; |
| **Warranty** | During the warranty period the Consultant shall:   * fix all defects reported by the Client; * solve all incidents reported by the Client according to the agreed SLAs;   Note: The response and resolution time shall not exceed 60 minutes for non-critical errors and 15 minutes in case of critical errors.  The incidents shall be solved within 2 working days for non-critical errors and within 4 working hours for critical errors starting from escalation time. Hourly progress report will be provided for critical errors. |

## Annex 3. Relevant legal codes and regulations that govern the business processes and procedures that will be automated with the System.

1. Law nr.91/2014 on electronic signature and electronic document - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=353612.
2. Law nr.71/2007 on registries - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=325732.
3. Law nr.1069/2000 on informatics - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=312902.
4. Law nr.467/2003 on informatics and state informational resources - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=313189.
5. Law nr.982/2000 on access to information - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=311759.
6. Law nr.133/2011 on personal data protection - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=340495.
7. Law nr.142/2018 on data exchange and interoperability - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=376762.
8. Government Decision nr.710/2011 on approving strategic Programme of technological modernization of government (e-Transformation) - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=340301;
9. Government Decision nr.1140/2017 on approving the Regulation  
   of the activity of the certification service providers in the field  
   application of the electronic signature - http://lex.justice.md/md/373494/.
10. Government Decision nr.1141/2017 on approving the Regulation on modality of application of the electronic signature on electronic documents by functionaries of legal persons governed by public law in the electronic document circulation - http://lex.justice.md/md/373495/.
11. Government Decision nr.1123/2010 on approving requirements regarding security of personal data during its processing by systems designed to work with personal data - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=337094.
12. Government Decision nr.128/2014 on Government single technological platform (MCloud) - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=351760.
13. Government Decision nr.1090/2013 on the governmental electronic service of authentication and access control (MPass) - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=351035.
14. Government Decision nr.405/2014 on the governmental electronic integrated service for digital signature (MSign) - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=353239.
15. Government Decision nr.708/2014 on the governmental electronic journaling service (MLog) - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=354589.
16. Government Decision nr.916/2007 on the concept of a Government Portal - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=324962.
17. Government Decision nr.330/28.05.2012 on development and administration of a single public services portal - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=343406.
18. Government Decision nr.701/2014 approving the Methodology of government open data publication - http://lex.justice.md/index.php?action=view&view=doc&lang=1&id=354534.

1. Modernization of Government Services in the Republic of Moldova (http://projects.worldbank.org/P148537?lang=en) [↑](#footnote-ref-2)