Funded by the European Union NextGenerationEU

RECOVERY AND RESILIENCE PLAN

GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC

UNIVERSITY OF ŽILINA Faculty of Management Science and Informatics

BRAIN:IT

Detailed design of the solution

- Part 1: Detailed design of the solution for **Functional** and **Non-functional requirements**.
- Part 2: Detailed design of the solution for **Technical requirements**.
- Part 3: Detailed design of procedures and products for the Testing and Production Deployment phases.

Project name

Document version:	
Project manager - contracting authority:	
Project manager - supplier:	
Chair of the Project Steering Committee:	

Contents

1 INTRODUCTION 4
1.1 IDENTIFICATION AND PURPOSE 4
1.1.1 Change log 4
1.1.2 Reference to documents 4
1.1.3 Abbreviations 4
1.1.4 Glossary of terms 5
1.1.5 List of figures 5
1.1.6 List of tables 5
2 EXECUTIVE SUMMARY 5
2.1 Outside the project scope 5
2.1.1.1.1.1.1.1.1 Part 1: Functional specification and detailed design of the solution 5
3 SOLUTION INTENT 6
3.1 OBJECTIVES AND BENEFITS OF THE PROPOSED SOLUTION 6
3.2 DESCRIPTION OF THE PROPOSED SOLUTION 6
3.3 MEASURABLE AND PERFORMANCE INDICATORS (KPI) 6
3.4 ACCEPTANCE CRITERIA 6
3.5 CATALOGUE OF REQUIREMENTS 6
4 DESCRIPTION OF THE ANALYSIS PROCEDURE 6
4.1 ARCHITECTURE DEVELOPMENT PROCEDURE 6
4.2 APPLICATION DESIGN LINKED TO THE ARCHITECTURE 7
5 PROCESSES SUPPORTED BY THE PROPOSED SOLUTION 7
6 BUSINESS ARCHITECTURE OF THE SOLUTION 7
7 DESCRIPTION OF DATA ENTITIES 7
8 SOLUTION DESIGN OF END SERVICES 7
8.1 Module 7
9 USE CASES 8
9.1 SCREEN VISUAL DESIGN SUPPORTED BY THE E2E PROCESS 8
9.2 USER ROLES – ACTORS 8
9.3 GENERAL USE CASES 8
9.4 SEQUENCE DIAGRAMS 8
9.4.1.1.1.1.1.1.1 Part 2: Technical specification and detailed design of the solution 8
10 APPLICATION ARCHITECTURE 8
11 PHYSICAL ARCHITECTURE 8
11.1 CI/CD PROCESS 9
11.2 CONTAINERIZATION AND ORCHESTRATION 9
11.3 DESCRIPTION OF THE TEST ENVIRONMENT 9
11.4 DESCRIPTION OF THE PRODUCTION ENVIRONMENT 9
12 SOFTWARE LICENCES 9
13 BACKUP AND RECOVERY OF THE SOLUTION 9
13.1 DESCRIPTION OF THE BACKUP SOLUTION 9

13.2 DESCRIPTION OF THE RECOVERY SOLUTION 10
14 ENSURING AVAILABILITY 10
14.1 DESCRIPTION OF THE SOLUTION FOR THE AVAILABILITY OF THE IS 10
15 SECURITY, AUTHORITY MANAGEMENT 10
15.1.1 Secure development principles 10
15.2 DEFINITION OF USER ROLES 12
15.3 SECURING COMMUNICATION AND INTEGRITY 12
15.3.1 Ensuring integrity 12
15.3.2 Securing communication 12
15.4 ACCESS CONTROL (AUDIT TRAILS – MONITORING OF LOGINS TO THE SYSTEM) 13
15.4.1 Access control 13
15.4.2 System logging and collection of audit records 13
15.5 USER MANAGEMENT METHODOLOGY 13
16 DATA CONVERSIONS, DATA MIGRATION, DATA MODEL AND SOLUTION DESIGN 13
16.1 DATA MIGRATION PROCEDURE 13
16.2 DATA MIGRATION VERIFICATION 13
16.2.1.1.1.1.1.1 Part 3: Detailed design of procedures and products for the Testing and Production Deployment phases 14
17 PROJECT PHASES AND TIMELINE – CONSOLIDATED OVERVIEW 14
17.1 Analysis and Design phase 14
17.2 Implementation and Testing phase 14
17.3 Deployment and Post-Implementation Support phase – preparation and deployment into operation 14
17.4 Project closure 14
18 DEPENDENCIES AND SOLUTION DESIGN 14
18.1 DEPENDENCIES (PROJECT/INTER-PROJECT) 14
18.2 OTHER (ADD AS NEEDED) 14
19 DOCUMENTATION 15
20 OPEN ISSUES 15
21 ANNEXES 15
21.1.1.1.1.1.1.1 Annex 1 - Catalogue of requirements 15

1 INTRODUCTION

1.1 IDENTIFICATION AND PURPOSE

1.1.1 Change log

Version	Date	Author	Reason for change	Chapter, item
1.0	24.6.2025	BrainIT	Initial version of the document	

1.1.2 Reference to documents

Identification	Document title
Annex 1	Requirements catalogue
Annex 2	List of figures

This document has been prepared within QA quality management in accordance with the rules set out in Decree 85/2020 Coll. and according to the recommended template of MIRRI SR. The contractual structure/template of the DNR was designed by the customer in accordance with Decree 85/2020 Coll. and according to the customer's needs.

1.1.3 Abbreviations

Abbreviation	Explanation
IS	Information system
DNR	Detailed design of the solution
API	Application programming interface, application platform interface

1.1.4 Glossary of terms

Term	Explanation
Application interface	A programming interface of an information system through which an authorized user is enabled to use an electronic service
Role	A standard application role that will be registered in the system
Code list	A set of data in the form of individual code list items, which are described by at least the pair of data elements "item code" and "item name"

1.1.5 List of figures

The list of figures is provided in Annex 2 of this document.

1.1.6 List of tables

2 EXECUTIVE SUMMARY

Add a description of the Executive summary chapter.

2.1 Outside the project scope

The following activities have been identified as outside the project scope:

- Implementation coverage of requirements approved as change requests.
- Implementation coverage of requirements marked as CANCELLED by decision of the Project Steering Committee.
- Implementation coverage of any requirements outside the approved list of requirements (defined by the relevant DNR (detailed design of the solution) documents from the Analysis and Design stage.
- Creation of the Contracting Authority's internal guidelines.
- Training beyond the scope defined in this document.
- Legislative and organizational preparation for putting the system into operation.
- SLA and operation of the solution.

2.1.1.1.1.1.1.1 Part 1: Functional specification and detailed design of the solution

The aim of this part of the document is to describe the system from the architectural perspective at the level of business, applications and technologies. The individual parts show the key entities and their relationships or dependencies.
In describing the architecture, ArchiMate 3.1 diagrams were used, created according to the principles set out in the document "Architektonické rámce verejnej správy, Enterprise architektúra VS SR 2014-2020".
Important links:
Main page for ArchiMate: https://pubs.opengroup.org/architecture/archimate3-doc/.

3 SOLUTION INTENT

Add a description for the Solution intent chapter.

3.1 OBJECTIVES AND BENEFITS OF THE PROPOSED SOLUTION

Add a description for the Objectives and benefits of the proposed solution chapter.

3.2 DESCRIPTION OF THE PROPOSED SOLUTION

Add a description for the Description of the proposed solution chapter.

3.3 MEASURABLE AND PERFORMANCE INDICATORS (KPI)

The measurable and performance indicators (KPI) are part of the table in chapter 3.1 Objectives and benefits of the proposed solution of this document.

3.4 ACCEPTANCE CRITERIA

Add a description for the Acceptance criteria chapter.

3.5 CATALOGUE OF REQUIREMENTS

This chapter forms part of the DNR document as Annex 1 to the DNR document.

4 DESCRIPTION OF THE ANALYSIS PROCEDURE

In developing the architecture, the TOGAF framework and the Archimate language are used within this project. The use of TOGAF framework components has been adapted for the purposes of this project. Primarily, the part of the Architecture Development Method concerning the development of architecture in all four architectural domains was adopted from the framework. The Architecture Development Method is a process with recommended steps and activities that have been set up taking into account the objectives and nature of this project.

4.1 ARCHITECTURE DEVELOPMENT PROCEDURE

The project is being developed by two development teams managed using the Scrum methodology with a two-week cycle for delivering planned increments. From the very beginning, regular weekly joint meetings of architects and other experts from both teams have been set up. Every week there are also meetings of the product owners, at which the basic exchange of information and synchronization of work on the project take place.

The above-mentioned meetings of architects and individual IT experts are of fundamental importance for architecture development. Each meeting is preceded by the preparation and summarization of current questions, their rapid assessment and prioritization, with the aim of including the topics with the highest priority and potential impact in the current meeting.

4.2 APPLICATION DESIGN LINKED TO THE ARCHITECTURE

The Enterprise Architect tool is used for the design of the application components being developed. The application interface design is created using the Figma tool.

5 PROCESSES SUPPORTED BY THE PROPOSED SOLUTION

Add a description for the Processes supported by the proposed solution chapter.

6 BUSINESS ARCHITECTURE OF THE SOLUTION

The chapter describes the business architecture of the solution.

7 DESCRIPTION OF DATA ENTITIES

The chapter describes the design of new entities so that the required functionality is covered.

Each entity contains:

- Definition of the entity
- List of basic attributes
- Attribute name
- Data type
- Description – either an explanation of what the attribute expresses, if it is not obvious from its name, or a description of the validation.

8 SOLUTION DESIGN OF END SERVICES

8.1 Module

9 USE CASES

One of the objectives of the solution design is to gather the customer's requirements for the Client zone. Use cases, which describe the relationships between the system and the system user, are used to capture the functional requirements.

The system user, also called the actor, initiates an interaction with the system in order to achieve a certain purpose. Each use case represents a series of activities and events from the perspective of the actor, who has defined permissions to perform the given activity.

9.1 SCREEN VISUAL DESIGN SUPPORTED BY THE E2E PROCESS

The chapter describes a structured design of the screen content based on the identified process and the data contained in this process.

The chapter is described in the chapter Solution design of end services.

9.2 USER ROLES – ACTORS

The chapter presents part of the design of user roles and permissions, which are subsequently assigned to the use cases in the following chapter. The following paragraphs provide a brief description of the conventions used to depict the model of user role interconnection.

9.3 GENERAL USE CASES

The following text sets out the individual user scenarios for...

9.4 SEQUENCE DIAGRAMS

The chapter describes the sequence diagrams.

9.4.1.1.1.1.1.1 Part 2: Technical specification and detailed design of the solution

10 APPLICATION ARCHITECTURE

Chapter describing the application architecture.

11 PHYSICAL ARCHITECTURE

The chapter describes the physical architecture and defines the requirements for the test and production environments.

The proposed operation is planned on two identical environments differing only in the amount and type of resources:
- **PROD** environment – production environment with real data and operation of the public portal.
- **TEST** environment – testing, training, pre-production environment. The entire TEST environment is accessible via VPN for privileged users.

11.1 CI/CD PROCESS

Add a description for the CI/CD process chapter.

11.2 CONTAINERIZATION AND ORCHESTRATION

Add a description for the Containerization and orchestration chapter.

11.3 DESCRIPTION OF THE TEST ENVIRONMENT

Add a description for the Description of the test environment chapter.

11.4 DESCRIPTION OF THE PRODUCTION ENVIRONMENT

Add a description for the Description of the production environment chapter.

12 SOFTWARE LICENCES

Component	Description	Licence type	Licence terms	Note

13 BACKUP AND RECOVERY OF THE SOLUTION

13.1 DESCRIPTION OF THE BACKUP SOLUTION

Add a description for the Description of the backup solution chapter.

13.2 DESCRIPTION OF THE RECOVERY SOLUTION

Add a description for the Description of the recovery solution chapter.

14 ENSURING AVAILABILITY

Operating parameters of the individual modules:

Description	Parameter	Note
Operating hours		
Service interventions		
Availability of the production environment		

14.1 DESCRIPTION OF THE SOLUTION FOR THE AVAILABILITY OF THE IS

Add a description for the Description of the solution for the availability of the IS chapter.

15 SECURITY, AUTHORITY MANAGEMENT

15.1.1 Secure development principles

The requirements laid down in this document must be met for all components that are used during development and all artefacts that are considered products of development.

By products of development we mean in particular:
- High-level Design (HLD), Low-level Design (LLD), Entity Relationship Diagram (ERD), process diagrams and the like
- software source code
- executable programs that are part of the delivery
- configurations
- database schemas
- test data sets

It is advisable to divide the security requirements into three parts:
- Requirements for the development environment
- Requirements for the developed product itself – the application
- Requirements for the production environment in which the product is operated.

The aim of these measures is to reduce the likelihood and impact of threats associated with software development and delivery, in particular:
- introduction of vulnerabilities as a result of poor programming techniques,
- introduction of vulnerabilities as a result of using insecure and unsupported frameworks,
- introduction of undesirable functionality into the code, whether through negligence or intentionally,
- delay in delivery due to an outage of development environments,
- breach of personal data protection principles due to its improper handling,
- introduction of vulnerabilities due to incorrect configuration of environments,
- criminal liability as a result of copyright infringement,
- leakage of sensitive data, loss of sensitive data,
- inability to secure evidence in the event of a security incident, both in development and in operation, and the like.

During the development of the project, it is necessary to prevent and avoid the introduction of the most common implementation errors, vulnerabilities and risks by adhering to a secure development life cycle, best practices and an appropriately chosen security standard and/or methodology. During the implementation of the proposed solution, the software is verified during development against well-known vulnerabilities and security recommendations, which form part of secure development practice to prevent the occurrence of the most common security risks for web applications.

By components used during development we mean in particular:
- workstations used in development
- virtualization environment
- servers used in development (development, test)
- network environment used in development, including home networks
- printers and other active elements
- physical environment in which development takes place, including home offices
- supporting services used in development, such as communication tools (MS Teams, Jira, email...)
- development tools (editors, compilers)

- development frameworks
- middleware (web servers, proxy servers, application servers)
- source code and configuration repositories (github)
- backup applications/services/systems and backed-up data
- authentication mechanisms (AD, AAD, LDAP, Radius, Kerberos, SAML, cloud access security broker and the like)

Securing the development environment
Development follows security best practice principles, including secure administration, configuration, patching and hardening of workstations and servers, anti-malware protection, segregation of physical and logical environments, account policies, credential management and secure login procedures, e.g. in accordance with ISO/IEC 27001.

All devices used in development will be protected by a firewall, both at the network perimeter (the interface between networks) and locally by means of a host firewall that permits only necessary connections. Only devices that are needed for development may be connected to the networks in which development takes place.
Endpoints on which development takes place must not be accessible to or used by other persons. In the case of computers, automatic screen locking protected by a password will be ensured and required.
The computer will not automatically connect to Wi-Fi networks other than the protected networks used in development. The Wi-Fi SSID to which a developer's computer is connected must have access controlled by WPA2 and a complex password, ideally authentication via WPA2 Enterprise. Where a physical LAN is used, access protection must be used.
All devices used in development, including the OS of network and security devices (routers, switches, firewalls, Wi-Fi access points), must have current security patches installed, and the firmware version must be at least N-1.

All network communication to and from the internet must pass through a device that filters dangerous and malicious content. All communication from computers used for development that are located outside protected internal networks (for example, when working from home) must pass through the corporate VPN gateway, which inspects and filters dangerous content.
Only legal, supported and up-to-date software will be used in development. The disks of all portable devices (laptops) must be encrypted. It is also necessary to encrypt the disks of non-portable workstations and servers that are not located in sufficiently protected data centre and server room premises.

All configurations must be protected against unauthorized changes. All configuration changes will be automatically detected and reported.

Only personal accounts may be used to log in to systems. Personal accounts used for ordinary work must be separate from personal accounts for administration. The use of generic and shared accounts (root, administrator) is prohibited except for disaster recovery. All accounts must be protected by SSH keys or, in exceptional cases where necessary, by a sufficiently strong password or passphrase with a high level of entropy.
On all systems used for development, only the minimum necessary set of software required for the specific activity may be installed.

All test data, sensitive data, passwords and the like are stored in defined locations.
i

15.2 DEFINITION OF USER ROLES

The rights themselves, whether for access to information, permissions for working with data, or for access to individual modules and functionalities, will be governed by permissions defined in the central user management and identity management, the so-called IAM (Identity and Access Management), using oAuth with 2FA, on the basis of which the individual user roles will be assigned.

15.3 SECURING COMMUNICATION AND INTEGRITY

15.3.1 Ensuring integrity

Validation of all user inputs and actions will be performed on the server side and, for suitable operations, also on the client side. All user inputs must be checked on the server side by means of whitelists (where possible and appropriate) or regular expressions in the context in which they are used.
Both the web server process and the backend database process must be configured to run under a unique user account with a limited set of privileges.
The web server will be configured so that files with web content are accessible to the process belonging to the web server service for reading only, not for writing (unless the service and process themselves require it).
The web content and all logs it generates will be located on a separate hard disk or on a different logical partition than the OS and the web server.
To meet the security requirements associated with the development of the TatraLeasing – client zone project, a suitable tool will be used that provides continuous code quality control, validation of the secure life cycle and inspection that detects bugs, duplicated code and security holes.

15.3.2 Securing communication

All communication channels will be known and described. The individual system components will be separated according to layers that perform a specific role.
The basic layers are:
- Access layer
- Application layer
- Database layer

The individual layers will be located in separate, logically segregated network segments, with communication between them permitted on the basis of firewall rules and, in addition, continuously monitored and filtered.
The administration and monitoring interfaces of network elements and servers are not accessible directly from the external network, the internet. The administration and monitoring interfaces are accessible only from pre-specified networks (e.g. the administrator network).
The web server will be accessible exclusively via the HTTPS protocol. In line with best practices, the web portal is not to be accessed via HTTP.
The identity of the web portal will be ensured by a valid, trusted certificate issued for the domain on which the portal in question is available. The web portal must not use untrusted or expired SSL/TLS certificates.
Data that are sensitive in terms of integrity or confidentiality must be transmitted only via an encrypted SSL/TLS connection. Sensitive data (especially login credentials) must be transmitted exclusively via an encrypted channel. The web portal will not store sensitive information in unencrypted form on either the client side or the server side.
The web server will support TLS 1.2, TLS 1.3 and newer. It will not support the old versions SSLv2, SSLv3, TLS 1.0 and TLS 1.1.
The web server is not to support ciphers with a key shorter than 112 bits and a block shorter than 64 bits. The web server must not support (they must be disabled) the HTTP methods OPTIONS, TRACK and TRACE.
The web application will, to the greatest possible extent, take into account and apply recommendations for the elimination of well-known vulnerabilities according to best practices.

15.4 ACCESS CONTROL (AUDIT TRAILS – MONITORING OF LOGINS TO THE SYSTEM)

The basic logical topology of the system solution will be divided into three parts in the AWS environment, namely:

1. Transfer – communication via F5 secured by a firewall but not towards the internet
2. Internal – application layer
3. Restricted – database and storage as a whole

15.4.1 Access control

IAM is used for access control to the application; it assigns user roles on the basis of permissions defined in the central user management based on oAuth and 2FA.
Access control between the individual zones is ensured by firewalls that use defined security groups to define the rights of individual roles. The zones divided into 3 parts represent virtual areas in the AWS environment that use Fargate ECS network ACLs for access control and communication between them.

15.4.2 System logging and collection of audit records

Logging and monitoring of the system will be ensured by AWS Fargate technology, which monitors and provides an overview of the resources used for individual accounts. It is recommended to set suitable metrics for monitoring resource utilization and capacity for each monitored device/system.
Monitoring and logging of the system will include alerts that should be provided to the system administrator in cases such as: storage capacity approaching full, abnormally high system load, an unusual security event detected.
Logs and audit records will contain specific information about the date, time and time zone used, they will be transmitted over a secure channel, and the granularity of logging will correspond to the requirements of the given service. Log retention will be set so as to prevent the logging space from filling up and not to affect the stability of resources, for example by using AWS EFS, whereby this service will also be used as a backup.

15.5 USER MANAGEMENT METHODOLOGY

The rights themselves, whether for access to information, permissions for working with data, or for access to individual modules and functionalities, will be governed by permissions defined in the central user management and identity management, the so-called IAM (Identity and Access Management), using oAuth with 2FA, on the basis of which the individual user roles will be assigned. IAM will also be used to define the manner in which individual users may operate the system (CRUD operations).

16 DATA CONVERSIONS, DATA MIGRATION, DATA MODEL AND SOLUTION DESIGN

No data migration will be performed within the solution.

16.1 DATA MIGRATION PROCEDURE

-

16.2 DATA MIGRATION VERIFICATION

-

16.2.1.1.1.1.1.1 Part 3: Detailed design of procedures and products for the Testing and Production Deployment phases

17 PROJECT PHASES AND TIMELINE – CONSOLIDATED OVERVIEW

PROJECT TIMELINE

Project activity	From	To
Analysis and design		
Implementation and testing		
Deployment		

TIMELINE OF THE WORK – part M4 and OUTPUTS

17.1 Analysis and Design phase

17.2 Implementation and Testing phase

17.3 Deployment and Post-Implementation Support phase – preparation and deployment into operation

17.4 Project closure

18 DEPENDENCIES AND SOLUTION DESIGN

18.1 DEPENDENCIES (PROJECT/INTER-PROJECT)

Add a description for the Dependencies chapter.

18.2 OTHER (ADD AS NEEDED)

19 DOCUMENTATION

The following documents will be delivered within the project and the work

Document	Created within the project
Technical documentation	Yes
Operational documentation	Yes
User documentation	Yes

20 OPEN ISSUES

No open issues arose from the following solution.

21 ANNEXES

Identification	Document title
Annex 1	Requirements catalogue
Annex 2	List of figures

21.1.1.1.1.1.1.1 Annex 1 - Catalogue of requirements

This chapter contains an overview of functional, non-functional and integration requirements.