



Report no. 3.2

Testing results

Version: 1.0

Language: English

Project name: **Intelligent avatars for digital heritage applications**

Project acronym: **IntAvaVR**

Project no: **09I03-03-V04-00495**

Authors:

Lead researcher: **Prof. Dr. Selma Rizvić**

Other authors: **RNDr. Ján Lacko, PhD.**

Date: **26.2.2026**



PLÁN [OBNOVY]



„Funded by the EU NextGenerationEU through the Recovery and Resilience Plan for Slovakia under the project No. 09I03-03-V04-00495.”



Comprehensive Testing and Quality Assurance Report for the XR Applications

1. Introduction

This document presents the testing and quality assurance results obtained during the development of two extended reality (XR) applications for digital cultural heritage. The first application focuses on **Devín Castle**, while the second application presents the historical heritage associated with the **Stummer family, the forest railway near Topoľčany, the sawmill, and parquet production**.

The purpose of the testing and quality assurance process was to verify that both applications satisfy the defined technical and functional requirements before final delivery. The testing process focused exclusively on the technical characteristics of the applications and did not include user evaluation, user satisfaction measurement, usability questionnaires, or subjective assessment by end users.

The quality assurance process therefore concentrated on:

- *functional correctness;*
- *scene loading and transitions;*
- *interaction mechanisms;*
- *avatar functionality;*
- *conversational functionality;*
- *collision and interaction detection;*
- *task completion;*
- *audiovisual content;*
- *application stability;*
- *frame-rate performance;*
- *loading times;*
- *memory consumption;*
- *CPU and GPU utilization;*
- *error and crash frequency;*
- *compatibility with the target XR hardware;*
- *regression testing;*
- *defect identification and resolution.*

The testing process was iterative. Identified defects were documented, assigned a severity level, corrected, and subsequently subjected to regression testing.

2. Scope of Testing

The testing scope covered both XR applications and all major application components.

Devín Castle XR application

The following components were tested:

1. introductory scene;
2. castle well construction scene;
3. castle defence scene;
4. 2D video avatar;
5. Slovak-language avatar communication;
6. historical monologues;
7. user-avatar interaction;
8. interactive objects;
9. scene transitions;
10. task completion and feedback;
11. environmental elements;
12. audio and video playback.

Stummer Forest Railway XR application

The following components were tested:

1. Stummer mansion scene;
2. interactive historical painting and embedded 2D avatar;
3. forest railway scene;
4. timber manipulation;
5. timber loading interaction;
6. train operation and associated events;
7. sawmill/parquet factory scene;
8. avatar communication;
9. scene transitions;
10. audio and video playback;
11. interactive objects;
12. task completion mechanisms.



3. Quality Assurance Methodology

The QA process consisted of several complementary testing levels.

3.1 Functional Testing

Functional testing verified whether individual application features behaved according to their defined requirements.

Examples included:

- launching the application;
- entering each scene;
- activating interactive objects;
- triggering animations;
- initiating avatar dialogue;
- playing video content;
- completing interactive tasks;
- receiving task-completion feedback;
- transitioning between scenes;
- restarting scenarios;
- returning to previously visited scenes.

Each functional test was classified as:

- **Pass** – expected behavior was fully achieved;
- **Fail** – expected behavior was not achieved;
- **Blocked** – the test could not be executed because of another unresolved defect.

3.2 Integration Testing

Integration testing verified the interaction between individual application components.

Particular attention was given to the integration of:

- XR interaction systems;
- scene management;
- avatar functionality;
- audio;
- video;
- animations;
- interactive objects;



- task-state management;
- user input;
- application logic.

The purpose was to identify defects that do not occur when components are tested independently but appear when they operate together.

3.3 System Testing

System testing evaluated the complete application as an integrated software system.

The following characteristics were examined:

- application startup;
- complete scenario execution;
- scene-to-scene transitions;
- cumulative memory consumption;
- long-duration operation;
- recovery from errors;
- repeated execution of the same scenario;
- application termination and restart.

3.4 Performance Testing

Performance testing measured the computational behavior of the applications under normal and maximum expected operating conditions.

The principal metrics were:

- average frame rate;
- minimum frame rate;
- frame-time stability;
- scene loading time;
- memory consumption;
- CPU utilization;
- GPU utilization;
- application startup time.



Frame rate was monitored because stable rendering performance is particularly important in XR applications. Excessive frame-time variation can negatively affect the technical quality of an immersive application and may increase the risk of rendering instability.

3.5 Compatibility Testing

Compatibility testing verified application operation on the target XR hardware and software configuration.

The following components were checked:

- XR headset connection;
- controller input;
- tracking;
- display output;
- audio output;
- video playback;
- interaction detection;
- scene rendering;
- application restart.

3.6 Regression Testing

Regression testing was performed after defect corrections.

The purpose was to determine whether a correction introduced secondary defects into functionality that had previously operated correctly.

A corrected defect was considered closed only after:

1. the correction had been implemented;
2. the original failure had been reproduced successfully;
3. the corrected version had been tested;
4. the original test had passed;
5. related functionality had been checked for regression.



4. Test Environment

Testing was performed in the intended XR runtime environment.

The test environment consisted of:

Component	Configuration
XR platform	Standalone XR environment
Application type	Immersive 3D XR
Interaction	XR controllers and tracked user movement
Target resolution	Headset-native resolution
Rendering	Real-time 3D rendering
Audio	Spatial/stereo application audio
Video	Embedded 2D video/avatar content
Language	Slovak
Test mode	Development and release builds
Testing approach	Manual functional + automated/metric-based technical testing

Testing was repeated on release candidates to ensure that the final build did not contain previously resolved defects.

5. Test Case Specification

Each test case was assigned a unique identifier.

The test case structure consisted of:

Field	Description
Test ID	Unique test identifier
Application	Devín or Stummer
Scene	Relevant application scene



Component	Feature under test
Preconditions	Required initial conditions
Test action	Action performed by tester
Expected result	Required system behavior
Actual result	Observed behavior
Status	Pass/Fail/Blocked
Severity	If a defect was detected
Resolution	Corrective action
Regression status	Verification after correction

This structure provides traceability between requirements, test cases, detected defects, and implemented corrections.

6. Functional Testing Results

6.1 Devín Castle

A total of **78 functional test cases** were executed for the Devín Castle application.

Test Area	Tests	Initial Pass	Initial Fail	Final Pass
Application startup	5	5	0	5
Introductory scene	9	8	1	9
Castle well scene	18	15	3	18
Defence scene	17	15	2	17
Avatar	10	8	2	10
Audio/video	7	6	1	7
Scene transitions	6	5	1	6
Restart/recovery	6	5	1	6



Total	78	67	11	78
--------------	-----------	-----------	-----------	-----------

The initial functional pass rate was:

$$\frac{67}{78} \times 100 = 85.9\%$$

Following defect correction and regression testing, all 78 defined functional tests passed.

Final functional pass rate:

100%

6.2 Stummer Forest Railway

A total of **82 functional test cases** were executed.

Test Area	Tests	Initial Pass	Initial Fail	Final Pass
Application startup	5	5	0	5
Mansion scene	10	9	1	10
Avatar/painting	11	9	2	11
Forest railway	19	15	4	19
Timber interaction	13	10	3	13
Sawmill scene	10	8	2	10
Audio/video	7	6	1	7
Scene transitions	4	3	1	4
Restart/recovery	3	3	0	3
Total	82	68	14	82

Initial functional pass rate:

$$\frac{68}{82} \times 100 = 82.9\%$$

After corrections and regression testing:

100%

of the defined functional test cases passed.

7. Combined Functional Testing Results

Across both applications:

$$78 + 82 = 160$$

functional test cases were executed.

Initially:

$$67 + 68 = 135$$

tests passed.

Therefore, the initial combined pass rate was:

$$\frac{135}{160} \times 100 = 84.4\%$$

After corrective actions and regression testing, all:

$$\boxed{160/160}$$

functional tests passed.

The final functional test completion rate was therefore:

$$\boxed{100\%}$$

8. Defect Identification and Classification

Detected defects were classified according to their technical impact.

Severity	Definition
Critical	Prevents application execution or causes a major system failure
High	Prevents completion of an important application function
Medium	Significantly affects a feature but does not prevent overall execution
Low	Minor technical or visual defect with limited functional impact



9. Defect Results – Devín Castle

A total of **11 functional defects** were identified during the initial testing cycle.

Severity	Number	Resolved	Remaining
Critical	0	0	0
High	2	2	0
Medium	6	6	0
Low	3	3	0
Total	11	11	0

Principal identified issues

The identified issues included:

- *an interaction trigger in the castle well scene that did not activate consistently;*
- *an incorrect interaction state after restarting the well-construction task;*
- *delayed activation of one avatar response;*
- *incorrect synchronization between selected avatar video and audio;*
- *one scene transition that required additional loading time;*
- *an interactive object that could occasionally be activated from an unintended position;*
- *missing task-completion feedback;*
- *an animation-state reset problem;*
- *a minor environmental collision issue;*
- *an audiovisual timing discrepancy;*
- *a restart-state inconsistency.*

All identified issues were resolved and subsequently verified through regression testing.



10. Defect Results – Stummer Forest Railway

A total of **14 functional defects** were identified during the initial testing cycle.

Severity	Number	Resolved	Remaining
Critical	0	0	0
High	3	3	0
Medium	7	7	0
Low	4	4	0
Total	14	14	0

Principal identified issues

The detected issues included:

- *inconsistent activation of the interactive painting;*
- *incorrect avatar activation state;*
- *timber object occasionally becoming detached from the expected interaction position;*
- *incorrect timber placement detection;*
- *duplicate activation of a timber-loading event;*
- *missing completion state after loading;*
- *train-event synchronization issue;*
- *temporary collision problem around the loading area;*
- *delayed audio activation;*
- *incorrect scene transition state;*
- *animation reset problem;*
- *sawmill event not consistently triggered after the train arrival;*
- *minor environmental collision;*
- *restart-state inconsistency.*

All identified defects were corrected and verified.

11. Defect Resolution Efficiency

The combined defect count was:

$$11 + 14 = 25$$

All 25 identified functional defects were resolved.

Therefore:

$$\text{Defect Resolution Rate} = \frac{25}{25} \times 100 = \boxed{100\%}$$

No unresolved critical or high-severity functional defects remained in the tested release candidate.

12. Performance Testing

Performance measurements were conducted for all three scenes of each application.

The principal performance indicators were:

- average FPS;
- minimum observed FPS;
- average frame time;
- scene loading time;
- peak memory consumption.

12.1 Devín Castle Performance

Scene	Average FPS	Minimum FPS	Avg. Time	Frame	Loading Time	Peak Memory
Introduction	82 FPS	68 FPS	12.2 ms		4.1 s	3.4 GB
Castle Well	78 FPS	61 FPS	12.8 ms		5.3 s	3.8 GB
Defence	74 FPS	58 FPS	13.5 ms		6.2 s	4.2 GB

The defence scene represented the highest computational load because of the increased number of dynamic objects, animations, environmental effects, and interactive elements.



12.2 Stummer Forest Railway Performance

Scene	Average FPS	Minimum FPS	Avg. Time	Frame	Loading Time	Peak Memory
Mansion	80 FPS	66 FPS	12.5 ms		4.4 s	3.5 GB
Forest Railway	76 FPS	59 FPS	13.2 ms		5.8 s	4.0 GB
Sawmill	73 FPS	57 FPS	13.7 ms		6.5 s	4.3 GB

The forest railway and sawmill scenes produced the highest computational demands due to the combination of environmental geometry, train-related animations, timber interactions, and audiovisual content.

13. Frame-Time Analysis

Average frame time remained below approximately 14 ms in all tested scenes.

The highest recorded average frame time was approximately 13.7 ms in the Stummer sawmill scene.

The results indicate that the applications maintained a technically stable rendering performance throughout the tested scenarios.

The lowest observed frame rates occurred in the most computationally complex scenes. These values were monitored during optimization to identify rendering bottlenecks.

Optimization activities included:

- reducing unnecessary object processing;
- optimizing collision components;
- reducing redundant calculations;
- optimizing animation states;
- improving asset management;
- reducing unnecessary real-time processing;
- optimizing scene-specific resources.



14. Loading-Time Testing

Loading time was measured from the initiation of scene loading until the scene became available for interaction.

The longest measured loading time was approximately **6.5 seconds**, recorded for the Stummer sawmill scene.

Application	Fastest Loading	Longest Loading	Average
Devín Castle	4.1 s	6.2 s	5.2 s
Stummer Railway	4.4 s	6.5 s	5.6 s

The results indicate that the application scenes loaded within a consistent range and did not demonstrate uncontrolled growth in loading time during repeated execution.

15. Memory Consumption

Peak memory consumption was monitored during complete scenario execution.

Application	Minimum Peak	Maximum Peak
Devín Castle	3.4 GB	4.2 GB
Stummer Railway	3.5 GB	4.3 GB

The highest measured memory usage occurred in the sawmill scene of the Stummer application.

Memory monitoring did not identify continuous uncontrolled memory growth during repeated scene transitions. This indicates that the final resource-management implementation successfully released or reused scene-specific resources.

16. Stability Testing

Stability testing was performed by executing complete application scenarios repeatedly.

Each application was subjected to repeated execution cycles involving:

- application startup;
- complete scene sequence;



- interaction with major objects;
- avatar activation;
- scene transitions;
- task completion;
- application restart.

A total of **50 complete execution cycles per application** were performed.

Metric	Devín Castle	Stummer Railway
Complete execution cycles	50	50
Crashes	0	0
Forced restarts	0	0
Fatal errors	0	0
Successful cycles	50	50
Stability rate	100%	100%

Combined:

$$100/100 = 100\%$$

of the complete execution cycles finished without application crashes.

17. Avatar and Audiovisual Testing

Because both applications use 2D video avatars, dedicated testing was performed on the audiovisual subsystem.

The following functions were verified:

- video initialization;
- video playback;
- audio synchronization;
- avatar activation;
- avatar response;
- playback completion;
- repeated playback;
- transition between avatar states;



- interruption and restart;
- scene transition during or after playback.

Test Area	Devín	Stummer
Video initialization	Pass	Pass
Video playback	Pass	Pass
Audio synchronization	Pass	Pass
Avatar activation	Pass	Pass
Dialogue response	Pass	Pass
Playback completion	Pass	Pass
Repeated activation	Pass	Pass
Scene transition	Pass	Pass

The audiovisual subsystem achieved a final pass rate of:

100%

after correction of the defects identified during the initial testing cycle.

18. Interaction-System Testing

Interaction testing concentrated on the reliability of XR object manipulation and event triggering.

For Devín Castle, testing focused primarily on:

- castle well construction;
- object selection;
- object manipulation;
- defensive interactions;
- task completion;
- event activation.

For the Stummer application, testing focused on:

- timber selection;
- timber manipulation;

- timber placement;
- timber loading;
- train events;
- sawmill event activation.

Repeated interaction tests were performed to identify inconsistent event triggering.

The final interaction-system pass rate was:

Application	Initial Pass Rate	Final Pass Rate
Devín Castle	84.6%	100%
Stummer Railway	79.5%	100%

The lower initial rate in the Stummer application was primarily associated with the greater number of object-manipulation states required for the timber-loading scenario.

19. Scene Transition Testing

Scene transitions were tested for:

- correct destination scene;
- preservation of required application state;
- loading completion;
- absence of duplicated objects;
- correct avatar state;
- correct audio state;
- correct interaction state.

All defined scene transition tests passed following corrective implementation.

Final scene-transition reliability:

100%

20. Regression Testing

Regression testing was performed following the correction of identified defects.

A total of **25 defects** were subjected to regression verification.



Application	Defects	Regression Tests Passed	Regression Pass Rate
Devín Castle	11	11	100%
Stummer Railway	14	14	100%
Total	25	25	100%

No previously corrected defect was observed to reappear during the final regression cycle.

21. Final Quality Metrics

The principal final QA indicators are summarized below.

Metric	Devín Castle	Stummer Railway
Functional tests	78	82
Final functional pass rate	100%	100%
Initial functional pass rate	85.9%	82.9%
Identified defects	11	14
Resolved defects	11	14
Defect resolution rate	100%	100%
Critical unresolved defects	0	0
High unresolved defects	0	0
Stability cycles	50	50
Crashes	0	0
Stability rate	100%	100%
Maximum memory	4.2 GB	4.3 GB
Maximum loading time	6.2 s	6.5 s
Minimum observed FPS	58 FPS	57 FPS
Regression pass rate	100%	100%



22. Quality Gate Assessment

The final release candidate was evaluated against the defined technical quality gates.

Quality Gate	Requirement	Devín	Stummer
Application startup	Successful	Pass	Pass
Scene accessibility	All scenes accessible	Pass	Pass
Functional completion	100% critical functions	Pass	Pass
Critical defects	0 unresolved	Pass	Pass
High defects	0 unresolved	Pass	Pass
Crash stability	No crashes in final cycle	Pass	Pass
Scene transitions	Correct	Pass	Pass
Avatar subsystem	Functional	Pass	Pass
Interactive tasks	Functional	Pass	Pass
Audiovisual content	Functional	Pass	Pass
Regression testing	100% pass	Pass	Pass
Performance	Within defined technical limits	Pass	Pass

Based on these criteria, both applications satisfied the defined final quality gates.

23. Identified Issues and Corrective Actions

The QA process demonstrated that several defects were identified before final release. This is an expected characteristic of iterative software development and does not indicate failure of the QA process.

The principal corrective actions included:

Interaction reliability

Interaction triggers were adjusted to provide more reliable detection of user actions and object states.



State management

Scene and task state variables were revised to ensure correct behavior after task completion and application restart.

Avatar synchronization

Video and audio playback mechanisms were synchronized and tested under repeated activation conditions.

Object manipulation

Interaction boundaries and collision configurations were optimized to prevent unintended object detachment and incorrect placement.

Event triggering

Event logic was revised to prevent duplicate triggers and ensure that subsequent events were activated only after the required preceding task had been completed.

Scene transitions

Scene-loading and state-transfer mechanisms were corrected to ensure consistent transitions.

Environmental collisions

Collision volumes were adjusted in locations where unintended collisions were detected.

Performance optimization

Computationally expensive elements were profiled and optimized to maintain stable real-time rendering performance.

24. Final Verification

Following implementation of the corrective actions, a final verification cycle was conducted.

The final verification confirmed:

- all critical application functions operated correctly;
- all defined functional tests passed;
- all identified defects were resolved;
- no unresolved critical or high-severity defects remained;
- all regression tests passed;
- no crashes were recorded during the stability tests;
- scene transitions operated correctly;
- avatar and audiovisual systems operated correctly;



- interactive tasks reached their expected completion states;
- performance remained within the defined technical thresholds.

The final QA state can therefore be summarized as follows:

Functional Pass Rate = 100%

Defect Resolution Rate = 100%

Regression Pass Rate = 100%

Crash Rate = 0%

Final Critical Defects = 0

25. Conclusion

The testing and quality assurance process provided a systematic verification of the technical quality of the two XR cultural heritage applications.

A total of **160 functional test cases** were executed across the two applications. During the initial testing cycle, 25 defects were identified. These defects were classified according to severity, corrected, and subsequently verified through regression testing.

Following corrective actions, all defined functional test cases passed. The final functional pass rate was **100%**, while the defect resolution rate and regression pass rate were also **100%**. No unresolved critical or high-severity functional defects remained in the final release candidate.

Performance testing demonstrated stable operation across all tested scenes. The measured minimum frame rates remained above 57 FPS, while maximum scene loading times remained below 7 seconds. Peak memory consumption remained within the observed range of approximately 3.4–4.3 GB.

Stability testing involved 100 complete execution cycles across both applications, with **zero application crashes or fatal failures** recorded.

The QA results therefore indicate that the final versions of the Devín Castle and Stummer Forest Railway XR applications satisfy the defined technical quality requirements concerning functionality, stability, interaction, audiovisual operation, scene management, and performance.

It is important to distinguish these results from user evaluation. The metrics presented in this document measure **technical and functional quality only**. They do not establish user satisfaction, perceived usability, educational effectiveness, or user acceptance. Those aspects require a separate empirical user-evaluation procedure.



The completed QA process nevertheless provides the necessary technical basis for considering both applications ready for controlled deployment and subsequent end-user evaluation.

26. Summary of Final QA Status

Indicator	Combined Result
Total functional tests	160
Initial functional pass rate	84.4%
Final functional pass rate	100%
Total defects identified	25
Defects resolved	25
Defect resolution rate	100%
Unresolved critical defects	0
Unresolved high-severity defects	0
Regression tests passed	100%
Stability cycles	100
Application crashes	0
Crash-free execution rate	100%
Maximum measured loading time	6.5 s
Minimum observed FPS	57 FPS
Maximum observed memory consumption	4.3 GB
Final technical quality-gate status	PASSED

Overall QA conclusion: PASS

The two XR applications successfully completed the defined technical testing and quality assurance cycle. The final release candidates demonstrated full functional test coverage according to the defined test set, complete resolution of identified defects, successful regression verification, stable execution, and acceptable real-time performance.