Intelligent Verification/Validation for XR Based Systems

Research and Innovation Action

Grant agreement no.: 856716

D6.5 – Market Research Report

iv4XR – WP6 – D6.5

Version 1.5

June 2022

Project Reference	EU H2020-ICT-2018-3 - 856716
Due Date	30/06/2022
Actual Date	8/07/2022
Document Author/s	Jeremy Cooke, Jean-Yves Donnart, Marta Couto, Ian Saunter, Beatriz Marin
Version	1.5
Dissemination level	Public
Status	Final

This project has received funding from the European Union's Horizon 2020 Research and innovation programme under grant agreement No 856716

Document Version Control			
Version	**Date**	**Change Made (and if appropriate reason for change)**	**Initials of Commentator(s) or Author(s)**
1.0	29/12/2021	Initial document structure and contents	IS/JC/MC/BM/JD
1.1	12/05/2022	Incorporation of whitepaper	JC
1.2	14/06/2022	Restructure, Edit and Inclusion of license options, addressable market summaries.	IS
1.3	30/06/2022	Finalize business models, competition, conclusions from Prague F2F meeting	IS
1.4	08/07/2022	Modification of the “Education and Training” paragraph in Section 4.	JD
1.5	08/07/2022	Final arrangements for submission	RP

Document Quality Control			
Version QA	**Date**	**Comments (and if appropriate reason for change)**	**Initials of QA Person**
1.4	08/07/2022	Minor formating changes	RP

Document Authors and Quality Assurance Checks		
Author Initials	**Name of Author**	**Institution**
JC	Jeremy Cooke	GWE
MC	Marta Couto	INESC-ID
IS	Ian Saunter	GWE
JD	Jean-Yves Donnart	THA-AVS
BM	Beatriz Marin	UPV
RP	Rui Prada	INESC-ID

TABLE OF CONTENTS

Executive Summary

This document is an extension and clarification of the work undertaken in task T6.2 Exploitation and Valorisation - promoting and preparing future exploitation and valorisation of project results. Here, we outline the current state-of-the-art and challenges in manual and automated XR testing. We detail the iv4XR approach to improving automated testing using an agent-based test framework. We assess the commercial opportunity for iv4XR framework intelligent agents: functional test agents and emotional test agents to automate testing of XR applications and the evaluation of UX.

In this D6.5 deliverable, we report on the key findings of market research that we have undertaken with key market players to guide our further research of the potential for future commercial exploitation and valorisation of the project outputs. This deliverable's findings will guide further market-led activities in the final phase of the project which will be further discussed in D6.6 (3rd Dissemination & Exploitation Plan - M36).

Section 1 - Introduction

The XR industry relies on iterative development methodologies that require a diverse toolset. Such toolsets must support multiple needs, including design and authoring of the experience, development of the software and integration and testing for quality assurance (QA).

Software testing is particularly critical to assure both a high quality user experience (UX) and to drive design and authoring throughout the iterative development process. Testing will often continue after a project is released, because updates, patches and improvements may be required. This is particularly common in server-based online experiences, such as multi-user games.

Good testing practices involve engaging users frequently. However, this procedure is costly and can be complex to manage [2]. In general, the toolsets available to XR developers currently lack structural testing technologies to support such practice because existing automated QA techniques are unable to handle XRs' high interactivity and realistic environments.

Today, testing of XR systems is mainly done manually. Testers are assigned to tasks for testing within manually created scripts that are crafted to validate component parts of the XR system functionality. Testing therefore requires a lot of human time and effort - thereby making the testing process overly expensive. Moreover, this requires testers to interact with a huge interaction space whilst also looking to assure the user experience (UX).

The lack of appropriate QA tools is an obstacle to the XR industry's growth and profitability. We state this because the extensive user testing required to ensure reliability of sophisticated virtual and augmented XR environments and to enhance a high quality user experience is time

consuming and expensive. The iv4XR toolkit solution/s in development will unlock many new commercial opportunities if we remove this obstacle. So, the clear goal of the project is to develop advanced and powerful testing tools that will 'power up' the toolsets available to the XR industry. The rise of extended reality (XR) systems relies on the benefits they provide in both the digital and the real world. A notable benefit of these systems is that they allow users to interact in simulated environments of different domains - such as healthcare, education, entertainment, automotive and aviation. The development, verification and validation of XR systems is an ever increasing challenge because of the requirements for iterative development, the integration processes, and the current need for human testing. Moreover, the increased requirement for quality assurance provides new opportunities for automation.

iv4XR project presents a toolset to improve the quality assurance of XR systems by using intelligent agents [1]. The iv4XR framework enables automated testing of XR systems and evaluation of UX by using intelligent agents - Functional Test Agents (FTA's) and Socio-Emotional Test Agents (SETA's).

D6.5 presents a brief description of the problem being addressed by the iv4XR project and the solution proposed by iv4XR. Beyond this the project team have engaged with likely customer prospects with a view to establishing commercial exploitation routes and business benefits that may be exploited by prospect companies developing XR systems.

Section 2 - The iv4XR framework

At its core, the iv4XR Framework is an agent-based system. A testing task is formulated, essentially, as a pair of goals and tactics for a test-agent to perform. The goal declaratively specifies what the task should accomplish, and the tactic is a program expressing a heuristic for accomplishing/solving the goal [3].

This is fundamentally different from the traditional way of testing, in which a testing task is formulated as a step-by-step program formulating the sequence of steps that are involved in conducting the test.

When testing a highly interactive system like an XR system, the sequence would consist of many interactions. A test that runs for 30 seconds can easily consist of over 1000 interactions. For example and in contrast, tests on web or applications rarely exceed 50 interactions. XR systems also often behave non-deterministically or even adversariously (computer games and simulators are typical examples).

The figure below shows the top-level architecture of the iv4XR Framework and its typical workflow.

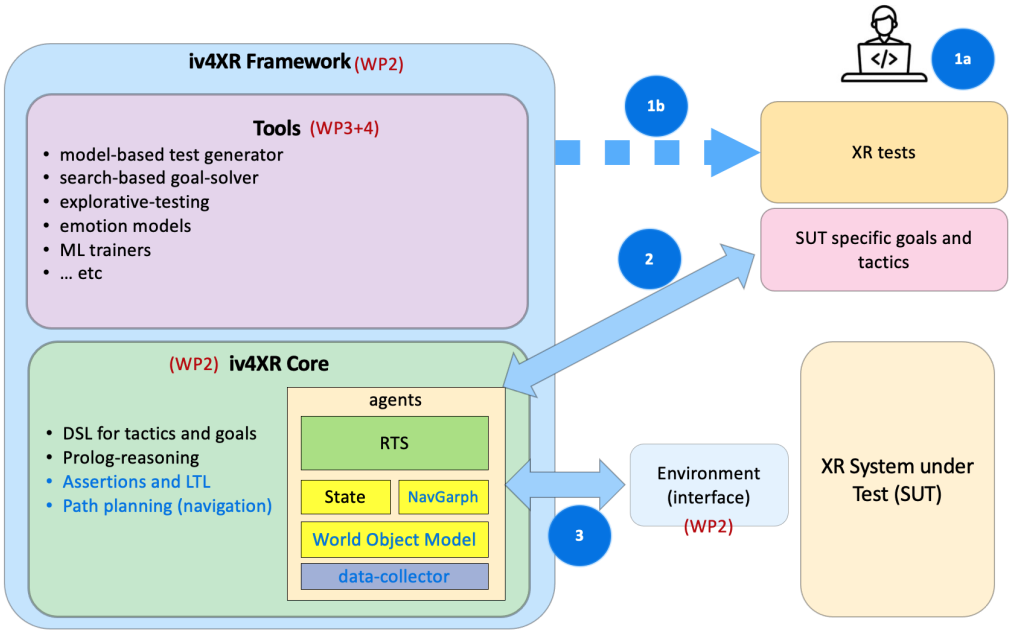

Fig.1: Top-level architecture of iv4XR framework

It is important to note that iv4XR strives to remain neutral towards the XR technology used in the system to be tested (System Under Test or SUT). However, this neutrality means that developers must first implement an interface (called Environment in Figure 1) between the system they wish to test and the iv4XR agents which control or work within the SUT.

Additionally, a library of domain specific abstract/high-level actions need to be defined and implemented. An example of such an action might be to toggle an in-world switch. Depending on the SUT, this action may first require the agent to approach the switch until it is close enough to it, and only then it can toggle the switch. Such knowledge must come from the SUT domain itself, as there is no way the agent could otherwise know this.

Accordingly, a programmer (or several) will be required to write an appropriate interface and any scripts which will need to be run on the SUT. We refer to these programmers here as Test Engineers.

The iv4XR solution has the potential to reduce time and costs, since it automates a percentage of the tests. Companies could reduce the percentage of budget allocated for testing and redirect that money to other endeavors. Our framework also goes beyond testing functionality and aims to automate user experience testing, providing developers with a set of measures that indicate the quality of user experience without having to find a representative sample of end users to test a prototype.

Generally speaking, testing regimes may be described as Functional Testing or Non-Functional Testing. Functional testing is usually carried out first and focuses on functional requirements to validate the behavior of the application and verifies that each function of the SUT operates in

conformance with its specification. Non-Functional Testing, on the other hand, is more associated with checking software performance, usability and, particularly in the case of games, entertainment and fun value. The iv4XR agent-based approach should accommodate both forms of test regime [4] [5] [6] [7], as the Socio-Emotional Test Agents under development in WP4 can be used for certain non-functional tests [8].

In Figure 2 we show how iv4XR test agents fit within the test cycle of a typical development which, here, is based on a massively multiplayer online game (Space Engineers) which is frequently updated with new features and virtual environments.

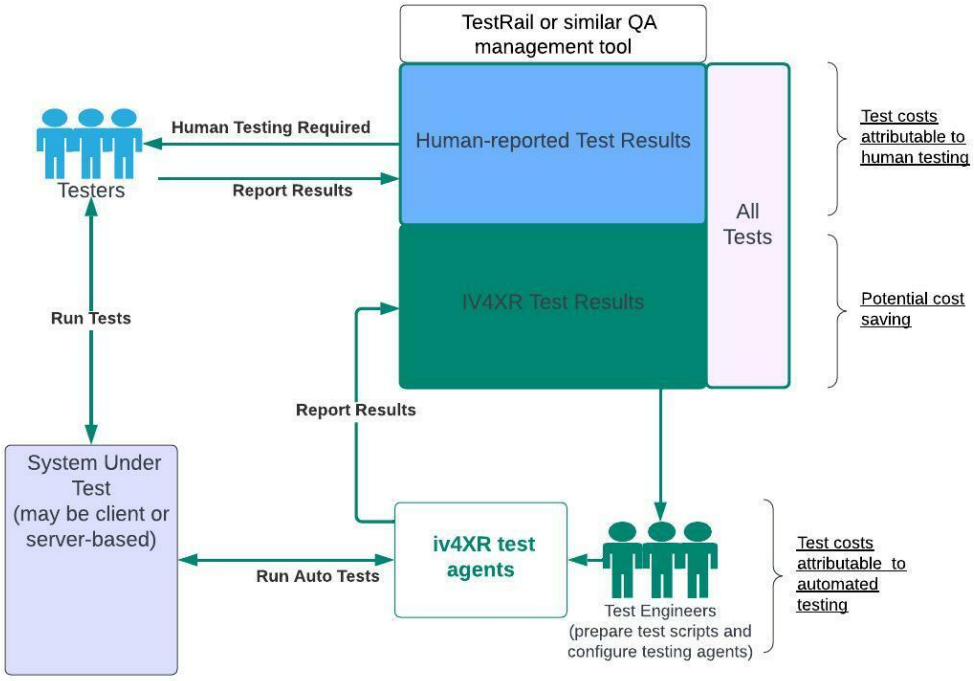

Fig.2: Example application of iv4XR framework to Space Engineers game

As is shown in Fig. 2, a list of all bugs, problems and issues is stored in a QA management tool (essentially a custom database) such as TestRail. Some tasks require human testing (the blue area) and other tests are considered suitable for automated testing (the green area). Test specifications are prepared by the Test Engineers. It is possible for many thousands of such tests to be performed in the time it would take human testers to perform a few hundred. Both manual and automated test results are reported back to the TestRail database for the game developers to address. As revisions to the game code are made, the process is repeated iteratively until the team lead is satisfied that the code is suitable for release.

Note that once the iv4XR test agent specifications have been prepared, there is little to no further requirement on the Test Engineers and so their cost can be considered as a 'once-off' cost. However, each test iteration which requires human testing will consume more resource costs. The same is true, indeed, for subsequent code updates and releases.

The potential for cost-saving is thus likely to be significant and may be substantial over a product's lifetime. The commercial challenge which iv4XR will face is in identifying which industries and, within these, which types of application development may yield the best opportunities to exploit such savings.

From October 2021 - April 2022 we have studied several software enterprise areas where we believe the automated testing approach offered by iv4XR could be most appropriate. In the next section we look at the results from this research.

Section 3 - Market Survey and Results

3.1 Survey Methodology

Any industry that uses XR technologies needs to test these technologies. To define the industries we were going to contact, we brainstormed and created a list of industries we knew were using XR technologies and that we could reach. In this market report, we cannot say we have produced an exhaustive list of industries using XR, but we did try to have an overview of the main sectors currently.

We decided to contact people in the Gaming and Entertainment industries since these sectors have been on the front line of developing XR technologies. Also, we have two partners who are engaged in this industry. We decided to contact software testing companies since these are engaged in the primary industry we aim to help. The iv4XR toolkit aims to automate validation and verification of XR systems; therefore, we need to know the current state of the art of software testing and the need and interest in our framework. Education and Training are also sectors in which XR technologies have been developing. We are all familiar with flight simulators and educational games, for example. Some of our partners even participated in a previous European project that developed training software for police interrogations using virtual reality (RAGE project).

We also targeted the Automotive and Aviation industries, given their highly technical nature and the expanded role XR development is playing in autonomous vehicles, safety and training within these sectors.

Finally, we also wanted to contact healthcare companies, since there are several XR applications in therapy, namely AR applications for patients with Parkinson's or patients recovering from strokes. However, we were unable to reach companies using this type of software.

We created a questionnaire using Google Forms and an informed consent form suitable for distribution via email. The questionnaire consisted of 12 questions (see Appendix 1) to characterize our sample (industry type and company size) and collect information regarding the extended reality systems used and testing procedures, including costs and participants opinions on agent-based testing and its potential usefulness in their testing procedures. We distributed the questionnaire to some 300 companies between December 21 and January 22 via email to direct contacts and asked them to share the questionnaire with appropriate contacts. We allowed 8 weeks for responses to the survey.

Despite several efforts to encourage a higher response rate we received 34 responses, however, alongside anecdotal evidence we believe we have been able to derive some valuable feedback which we summarise in the next section.

3.2 Survey Findings

The companies which responded represented 7 different industries as shown in the figure below

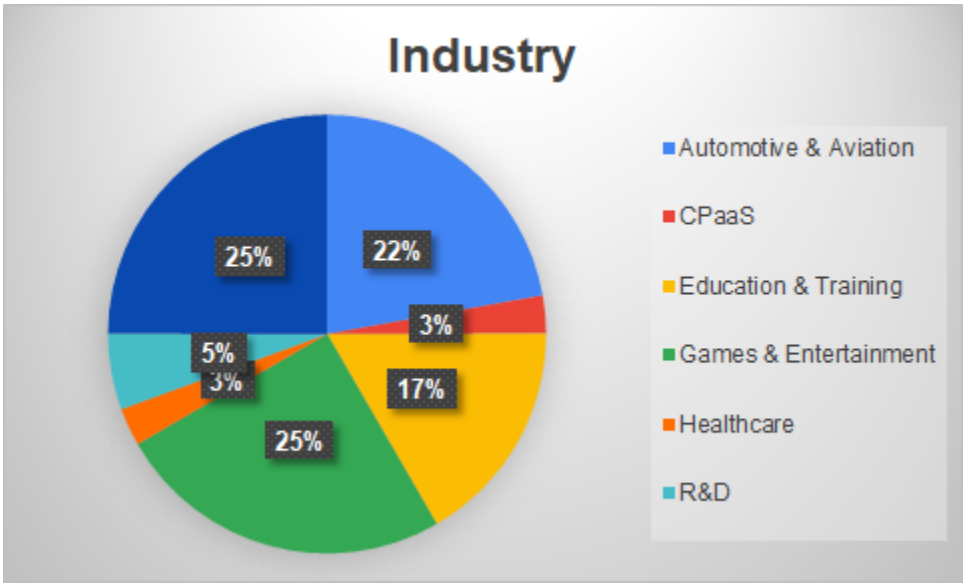

Fig.3: Surveyed Industries using XR technology

As Figure 4 shows, the companies were mostly micro companies (30.6%) or large companies (33.3%), and the majority of the companies developing their own systems reported spending 10 to 20% of their budget on testing (40% of responses).

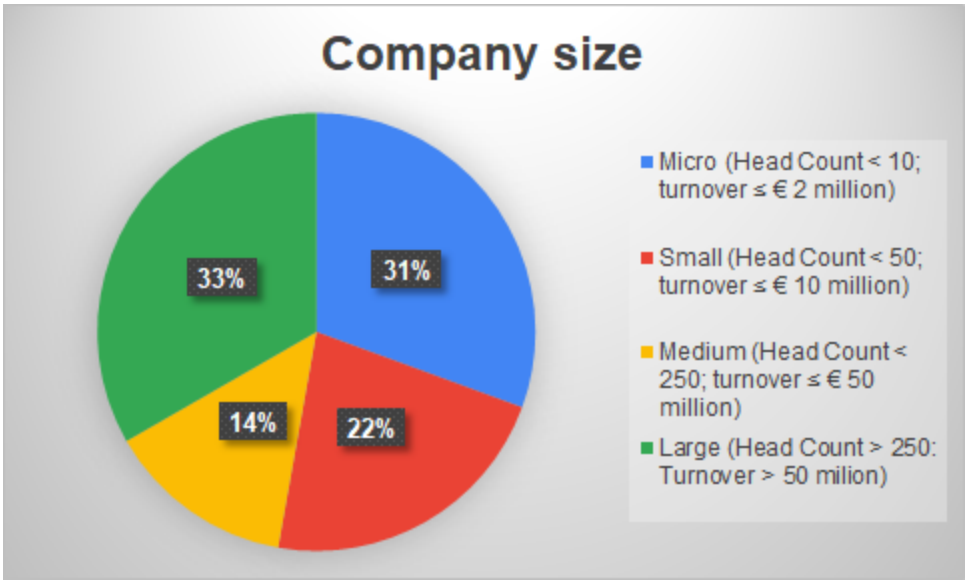

Fig.4: Size of surveyed companies using XR technology

We also asked the companies why they were using or developing XR technologies. The replies were diverse and were naturally dependent on the industry but many companies reported using XR technologies for the audio and visual features. Some were taking advantage of those features to improve teaching, including offering a more immersive experience to trainees. Others are using it to let clients visualise what the final product will look like. Games and entertainment companies are also using XR to improve immersion and offer more immersive experiences.

Comments from the Games and Entertainment sector. 25% of those responding to the iv4XR questionnaire described themselves as being from the Games and Entertainment Industry, making this the largest sector represented in the survey results. This fact alone might indicate that the intuitive notion that games would be well-served by the iv4XR technologies is correct.

Games and Entertainment accounted for the highest number of MR and AR developers in our survey (67% and 56% respectively) and the second highest for Other 3D and VR-based XR environments, whilst Other 2D graphical environments were least useful. This appears to support the anecdotal evidence already discussed that the game industry's development environments are at the leading edge of XR and therefore a potentially valuable commercial revenue source for iv4XR outputs post-project.

Comments from the Automotive and Aviation sectors. Clearly, these are enormous industries, and we anticipated that it would be challenging to get responses that would represent these sectors. These two industries deal with large projects, many of which are kept secret until they are ready to go to market. We did have some feedback from people who were unsure if they could answer the questionnaire or replied that they could not respond because they did not want to give away their strategy for developing and using XR systems.

However, eight companies did respond to our questionnaire, representing five large companies (HeadCount > 250: Turnover > 50 million) and three small companies (HeadCount < 50; turnover ≤ € 10 million). We asked them what kind of XR technologies they are using, and 62.5% of the responses indicated that the companies are currently using VR technology. Only 25% selected AR, and 50% selected mixed reality systems. Although we could see more news related to AR applications in our initial search online, it seems that at least these companies are investing in mixed reality systems more than they are investing in dedicated AR systems.

We also asked about 2D virtual environments (e.g. browser-based data handling/analytics, 2D games/simulations) and 3D Virtual Environments (e.g., simulations, games, CAD systems). 62.5% indicated using 3D environments, and 37.5%, 2D environments. Looking at the data per company, we noted that the two large companies reported using all types of systems, which is not surprising. However, the small companies seemed clustered into two groups. Some used only VR, AR, and/or MR, whereas the others only selected 2D and 3D virtual environments.

62.5% indicated that they use systems developed by a third party. Only one of the companies developing their systems reported spending less than 5% of a project's budget. All the companies have internal test departments responsible for ensuring that the systems function correctly.

We were also interested in knowing whether these companies felt a need for automated testing tools. We asked them to rate the company's current need for automated testing tools, from not required to vitally important. Only 25% reported a high need for automated testing tools. We looked at the data by company and noted that the two largest companies, who also develop their own systems, both reported a higher need for automated testing tools. This result is not surprising given that these tend to be companies with bigger budgets who are also spending more money on testing. For companies who do not develop their own systems, it is perhaps understandable that automated software testing tools do not represent the same value at this time.

We provided a brief explanation of the intelligent automated testing tools iv4XR is developing and asked what the company's interest is in that tool. 62,5% indicated being interested, to very interested.

Comments from the Healthcare sector. Out of all respondents to the questionnaire, only one reported being from the healthcare industry. We engaged in direct conversation here and understood that they use XR, in particular VR, to develop immersive applications to help patients in certain therapeutic scenarios. There is a need for support in automating the testing process both in terms of functional as well as affective dimensions. The environment is very similar to that of LabRecruits hence the iv4XR solution could be applied to their application and similar ones.

General comments. Results indicated that 25% of the companies which answered the questionnaire were from companies that develop XR systems, but do not belong within the previously mentioned sectors.

Regarding the company size, 44% correspond to micro companies (Head Count < 10; turnover ≤ € 2 million), 44% correspond to medium companies (Head Count < 250; turnover ≤ € 50 million) and 11% correspond to large companies (Head Count > 250: Turnover > 50 million).

With respect to the type of XR systems that the company use, these were distributed as follows:

- 78% of respondents select Virtual Reality (VR) systems,
- 55% select Augmented Reality (AR) systems,
- 44% of respondents select Mixed Reality (MR) systems,
- 55% select Other 2D Virtual Environments (e.g. browser-based data handling/analytics or 2D simulations),
- 89% select Other 3D Virtual Environments (e.g. simulations, games, or CAD systems).

When we asked if they develop all the software or they use third party companies in their development process, 44% answered that they develop their XR systems in-house, 22% of respondents say that they develop the XR systems as well as third party companies, and 33% of companies only use XR systems developed by third party companies.

Regarding the question about testing the XR systems, 89% state that they have an internal department for testing, and 11% reported that they use a combination of the internal department with an external department.

After that, we ask about the percentage of the software development cost spent on testing, 55% of subjects didn't answer the question since they do not develop XR systems. From the remaining subjects, 22% answered that they don't know, 22% stated between 10 and 20 percent. This reflects that even though the testing phase is known to be expensive, actually companies do not have a clear number of how much they are spending on that.

We asked participants what XR technologies are used for in their company. As expected, the answers varied considerably. Five responses indicated that companies are using XR technologies to create products, such as serious games. Companies using XR for training and simulation range from companies making use of XR to aid neurodiverse students or increasing immersion for trainees, to immersive flight simulators, EMT scenarios and even relaxing environments. Four responses indicated using XR technology for simulation and visualisations. Two companies reported using XR technology to create a virtual experience of products to demonstrate to their clients and to test User Interface and User Experience variables. Remote maintenance, saving time on inspections, testing physical ergonomics, improving communication, and training are some of the ways the automotive and aviation industries are using XR technology. Aside from these answers, we had others we could not readily group

together such as quality assurance, metaverse experiences, developing autonomous characters and improving workflow.

Finally, regarding the current need for automated testing of XR systems, 44% of companies state that they are not required at present, 11% stated that they might be required, and 44% state that they do require automated solutions for testing their XR systems. The last question was related to the interest of companies for automated testing tools for the future. 44% of companies state that they will require it, 22% state that they could require it, and 33% stated that they will not require the automated testing tools for XR systems.

We conclude from the results of our survey, as well as feedback from presentations made and papers published by the consortium, that there is a sufficiently significant interest in the iv4XR approach and solutions to justify post-project distribution and/or commercialisation of the project's outputs. However, there are numerous options as to how to achieve this, including licensing the iv4XR tools - and how this is done may depend on industry norms in the sectors being targeted. We look at these issues in the next section.

Section 4 - Licensing Models and Target Sector Characteristics

4.1 License Options

The consortium will need to determine the best means through which the iv4XR technologies can be made available to developers. These will very likely depend on how the technologies are packaged and documented at the end of the project as well as to the business sector being addressed. But, essentially, there will be a choice as between a charging model (i.e. a proprietary software license) or whether the software will be made available as open-source.

A proprietary software license would enable the creators of the iv4XR software (or an entity agreed upon by the consortium, such as an iv4XR governance body to be set up to manage post-project activities) to maintain control over who uses the software, how it is used, and the code base itself. The 'proprietary' description relates to control over the Intellectual Property of the software or its legal status and use, whereas a commercial software license is designed for software that is intended to generate revenue - either for its creators or the people who use it. To be clear, if the consortium determines that there is sufficient potential in iv4XR solutions to enable a business to be built around its exploitation, then it would need to draw up commercial licenses to cover terms of use and pricing. These may vary from sector to sector or as a function of which code modules were required. The vast majority of commercial licenses are for proprietary software and these can take many forms at the enterprise level, which are briefly described here:

- **On-premises license**. This is for any software that is installed on the user's own computer or server. Software with this type of licensing is accessed locally through the user's own device. In the era of floppy disks and CD-ROMs, on-premise licensing was

the only deployment method available for software licensing, but now cloud-based licensing is quickly overtaking it.

- **Cloud-based software licensing.** Here, the licensed software is run, tracked, and managed over the internet, rather than on the user's device or servers. Cloud-based software is usually much more convenient for users to install, and allows us to offer updates and new features easily. iv4XR developers can gain valuable insights from real-time data about how their customers use the software, which they can use to improve the product, adjust sales models, or even discover unauthorized use.

  One of the biggest advantages of cloud-based licensing, however, is that it makes a wide variety of licensing models a lot more convenient. With cloud-based licensing, you can change the details of a license agreement without new hardware getting anywhere near your user's device. The customer can change the features, add or subtract authorized users, or renew their license with a few clicks.

- **Perpetual license.** Here, a customer buys the software once and then can keep using it as long as they want. This was the most common model before cloud-based licensing became an option, although it still has some advantages - being long-term with a single, clear fee, convenient and easy to understand. On the other hand, perpetual licenses don't usually cover access to upgrades, patches, and support.

- **Subscription license**. Effectively the opposite of a perpetual license, here the custom is charged a fee for a limited-time use. Usually, customers will pay per month or year. If customers don't renew the license agreement at the end of that period, they lose access to the software. This has become much more convenient, and more popular, since the growth of cloud-based licensing.

  Subscription-based licensing gives users a lot more flexibility and requires a lot less commitment than perpetual licensing. As a developer, the subscription model can provide for an ongoing relationship with customers, which means that a minimal license agreement for a short period at a low price has the potential to turn into an extensive license agreement that could be renewed for years at much higher returns/prices.

- **SaaS License (Software as a Service License)** SaaS customers pay to use the software for a certain amount of time, instead of buying a copy and owning it. This type of subscription model often comes bundled with other services such as customer support, upgrades, and access to helpful content. The software vendor hosts the software in the cloud instead of the customer installing it on their own server or device.

SaaS licenses are becoming the *de facto* standard model for almost all enterprise software. Because the software is hosted in the cloud, it is easy to update license agreements, allowing the software to scale easily. Vendors can also efficiently offer SaaS licensing for the same software at a wide variety of price points and so this model would provide a flexible and quite manageable way to license iv4XR.

Of course, it is also possible to create Hybrid Licenses which combine elements of the above mentioned formats. Furthermore, other license models or variants could be considered, for example for use of iv4XR in *Academic* institutions; on a *Trial* basis; or on an *On-demand* basis.

Open Source Options

At the end of the project's funded timeline, it may not prove possible to package the project's outputs in such a way as to make a commercial release viable. To achieve this, its' Technical Readiness Level should be at least TRL8 or 9, however the minimum project goal at its outset was TRL 6 (the technology is demonstrated in an industrially relevant environment). We expect to exceed this target, but achieving a fully documented and release-ready software suite by the project's close is not guaranteed. For this reason, we shall also consider making code available on an *Open-Source* basis.

Open-source software is shared publicly and anyone who has access to it can take the code, customize it, and build on it. It needs not necessarily be free and there can be limits on how it is used, but the original developer still has much less control once customisation starts to take place.

After discussion within the consortium, we conclude that a BSD 3-Clause License would represent the likeliest open-source distribution terms. BDS licenses are permissive, meaning that they don't require much from users of the licensed software except that they include the full text of the license and any original copyright notices if they copy, modify or distribute that licensed code. This makes a BSD 3 option easy to implement and very flexible for developers and users alike and should allow us to reach the widest possible audience for iv4XR outputs.

4.2 Addressable Market Characteristics

4.2.1 Games and Interactive Software

Over the past 20 years or more the gaming and entertainment industries have proven to be, in many ways, the main engines of XR technology development, as many of these XR technologies started in those sectors as developers strive to make their games closer and closer to immersive filmic experiences. In fact, other industries sometimes use game engines such as Epic Games' Unreal Engine to create XR experiences.

The gaming and entertainment industries do not have the same challenges and potential 'life or death' consequences as most of the other sectors we are focusing on. However, because they

have been using and developing these technologies longer and at a speedier rate, they have encountered challenges that the other industries do not have.

Game developers must also deal with creating increasingly complex environments in which sometimes entire worlds are simulated, and users have demanding expectations for these environments. For example, they expect the possibility of interacting with hundreds or thousands of other users without it affecting their game's performance; they expect the system to save changes they make to the environment or game progress.

Entertainment and games are very fast-paced industries - so games and entertainment are industries where people want constant novelty with a great user experience. A slow, unengaging product may mean that potential users will not buy the product or will stop subscribing, with commensurate and significant impacts on the developers and their publisher's revenues. This constant demand for novelty and a good user experience means that testing the systems is critical, and the faster, the more efficient and the more reliably accurate it is, the better.

So quality assurance of a video game product plays a significant role throughout the development cycle of a game, though comes more significantly into play as the game nears completion. Unlike other software products or productivity applications, video games are fundamentally meant to entertain, and thus the testing of video games can appear more focused on the end-user experience than the accuracy of the software code's performance, which leads to differences in how game software is developed.

Because game development is focused on the presentation and gameplay as seen by the player, there may be little rigor applied in maintaining and testing backend code during the early stages of development since such code may be readily disregarded or discarded if there are subsequent changes to the gameplay or even the game content (graphic and audio assets, in-game text, speech or instructions, UI mechanics etc). Some automated testing may be used to assure the core game engine operates as expected, but most game testing comes via game testers, who enter the testing process only when a playable prototype is available. This may be one level or subset of the game software that can be used to any reasonable extent - often referred to as when it reaches 'first playable' milestone or a 'vertical slice'.

Whilst the use of testers may be lightweight during the early stages of development, the testers' role becomes more predominant as the game nears completion, becoming a full-time role alongside programmers and artists. Ideally, testing should be considered a key part of game design, however man-power may be constrained at this stage because it is likely that the studio is simultaneously completing another project - which can often involve a manpower 'crunch'.

As code matures and the gameplay features solidify, then development typically includes more rigorous test controls including regression testing to make sure new updates to the code base do not change working parts of the game. Games are complex software systems, and changes in one code area may cause unexpected problems in a seemingly unrelated part of the game. Testers are tasked with repeating play through updated versions of games in these later stages

to look for any issues or bugs not otherwise found. This can be a very monotonous task, playing the same game over and over, and a purely manual process can risk games being released with uncaught bugs or glitches. Automated testing of the most mundane of such tasks or frequently recurring game actions, would free up human resources to work on more interesting aspects of the game, including the quality of gameplay itself - or the fun factor.

There are other factors inherent to video games that can make testing difficult. This includes the use of randomized or procedural gameplay systems, non-deterministic outcomes, and novel AI algorithms for non-player characters, navigation, emotion modeling and so forth, all of which require testing for both game balance (e.g. playability or difficulty level and its progression) as well as bug tracking. These issues become increasingly marked in non-linear games and multiplayer games as opposed to, say, puzzle-based games. So the balancing of cost and time to devote to testing as part of the development budget; and assuring that the game still remains fun and entertaining to play as changes are made to it can also be considered quite unique to game development.

Further complicating the game developer's position is the increasingly frequent requirement to support a game release with regular upgrades and application features which may be sold as 'in-app' purchases or add-on packs (new levels, new environments, characters etc), or distributed as free updates to a subscriber community. These updates help to secure the long-term viability of a game franchise and its value as an Intellectual Property (e.g. World of Warcraft, Call of Duty, Total War, Clash of Clans, Angry Birds etc.). New releases also afford the developer an opportunity to address any known issues reported since the previous release of the game and so the whole application must go through the entire test process again.

Given its importance to a successful product release, its increasingly high time and manpower cost and its sheer complexity, it should be evident that any cost saving that automation of the test process may yield should prove of great interest to the games industry.

4.2.2 Education and Training

Training simulations and other education environments pose challenges to designers and developers. For example, simulations for training should be as complete as possible and allow users to train all situations they might encounter. The system needs to react appropriately; if not, the simulation is missing stimuli that will impact the user's reaction if they encounter a similar situation in real life.

Testing a simulation for education or training purposes may be compared to the test of a very realistic video game where players are not constrained by the predefined scenario implemented by a game designer. Because such simulations prepare the students to face difficult situations in real operations, the tests must be designed carefully to avoid that the training process introduces a bias in the student behavior that might have severe consequences in real life.

Most of the simulation's tests in the education and training domain are performed by real operators who replace the students in predefined scenarios which aim to simulate the large panel of situations the students may encounter and the great diversity of behaviors these students may be able to exhibit in these situations.

This testing phase is always very costly in terms of manpower, implying dozens of testers and hundreds of workdays for complex training simulations. Today, automated tests are essentially used to validate the functioning of basic components of the simulation and, more rarely, during the test of operational scenarios where students are supposed to be in the loop.

In such applications, the work of testers is not only to play the role of ideal students that follow all the rules and procedures, but also to use their human intelligence to discover short-cuts in the simulation that will not be possible in real life or specific tactics that will defeat the scenario by unrealistic means.

Coverage of the tests is another important issue of education and training simulations. Even if the scenarios used for the testing are defined to represent the majority of situations the students may encounter, they are by definition limited in terms of their number and of the time needed to be performed with real operators.

All these observations lead the education and training domain to extend progressively the use of automated testing by replacing human testers by AI agents with the same capacities to test operational scenarios but with the additional faculties to run these scenarios much quicker and in a more exhaustive way.

4.2.3 Automotive and Aviation

In the automotive industry, companies like Mercedes and BMW have been investing in Augmented reality to boost navigation by improving accuracy and ease of use in navigation systems. For BMW, the front view camera sends live footage to a curved touchscreen where interactive arrows show the driver exactly where they need to go. And in Germany, they have been testing a system that helps drivers find parking spots in busy car parks.

But AR in the automotive industry goes beyond improving the navigation experience. It can significantly improve security by helping the driver perceive the environment better, for example, by highlighting pedestrians and other vehicles. These systems need a high level of accuracy. Testing procedures should be critical to ensure that the system is working to improve safety without being distracting or providing inaccurate information that could have adverse effects.

Another industry that has shown a growing interest in XR technologies is the aviation industry. For the aviation XR weekend, the International Air Transport Association (IATA) showed some examples of companies using XR technologies. These examples included training of ramp staff (Qatar airways), inflight entertainment (Iberia and Sri Lanka airline), and an AR bag sizer for a

mobile app (LATAM and KLM). Projection of data information and design details to help construction and inspection (Airbus) and Qantas has designed their latest lounge using VR.

In the aviation industry, flight simulators have been used for several years to provide training for pilots. From Aviation Training Devices that train basic skills to full flight simulators that allow pilots to train emergency procedures without actually having to experience a malfunction or accident. The same occurs for healthcare applications where physicians can train surgery skills and how to respond to unexpected but possible problems without risking someone's life.

When people think about the automotive and aviation industries, they often think about car brands or plain manufacturers. Still, these industries have thousands of companies providing services and materials, from components to marketing services. Therefore when we say automotive and aviation, we are talking about an enormous range of companies that use XR with different goals.

4.2.4 Software Design and Testing

Testing is the most used technique for quality assurance in industry. Nevertheless, the last “Fail Watch report” [9] states that there are more bugs in production than we ever imagined, demanding our attention in order to try and avoid “the software apocalypse”. A large part of our target companies are software development companies that develop XR systems for different applications, such as XR systems for meetings, XR systems for building sector, XR systems for marketing, among others - and all of these applications require extensive testing to ensure good software quality standards.

Nowadays, both design and testing phases of XR systems require human effort to be carried out properly, which are error-prone and time-consuming tasks. We analyze this sector in order to gain knowledge about how companies are testing the software, how many resources they spend in that task and the possibility to use the iv4XR framework in their future developments.

4.2.5 Healthcare

XR technologies have various applications in the healthcare domain, including, but not limited to therapeutic games. Development and testing practices are still heavily reliant on manual efforts. In particular testing activities, when present, are limited to “trying” the application as would the end user without little or no support for automation. This could be partly due to the fact that the developers are not always from an IT background, but rather from the healthcare domain and that the applications are developed by customizing existing development platforms or simply reusing existing components from libraries. Hence systematic automated testing of XR applications in the domain is typically missing.

Potentially, the iv4XR solutions, in particular emotional/affective testing, are well suited for further customization with different emotion models, for example. Currently emotion related feedback are gathered from people directly through biometric sensors, which could make testing

quite difficult and expensive. Adopting emotional testing approaches developed in iv4XR could be a viable option both in terms of cost and privacy/ethics.

4.3 XR Market Size & Value

Whilst it is always difficult to accurately predict potential market value, according to a forecast by MarketsandMarkets[1],the XR market is expected to grow by 380 percent from USD 33.0 billion in 2021 to USD 125.2 billion in 2026. Perhaps unsurprisingly, the main industries driving this expected growth were found to be very much in line with our own expectations and as reported herein. These were the education sector, the automotive industry, the healthcare sector, and the entertainment and gaming industry.

Market leaders in the market are companies from the US and Asia, including Microsoft (US), Sony (Japan), Meta (US), HTC (Taiwan), and Google (US). However, "European industry is aware of the high potential of XR and is ready to use it"[2]. A study published in 2021 by Ecorys and the XR Association (XRA) produced the following key findings: (i) Over 93% of XR companies surveyed in Europe predict growth in their sales over the next three years, with two thirds expecting their revenues to grow by more than half. (ii) The total market value of the European XR industry is expected to increase to between €35 billion and €65 billion by 2025, which would represent a gross added value of between €20 billion and €40 billion, and directly creating employment for some 440,000 to 860,000 people.

We believe that this growth and its value will continue to be driven by the sectors we have examined here and that demand for automated testing should, likewise, have the potential to grow alongside these trends.

4.4 Competition

Whilst there are many systems which enable automated software testing, these are very largely limited to web-based environments. Some examples would include Selenium, TestIM and Cucumber alongside many other open-source tools which deal with browser-based systems under test. As such, they differ greatly from the iv4XR approach and solutions which are focused on XR applications running on native operating systems.

However, we have identified two systems which appear to be competitive with iv4XR. These are Game Driver (https://gamedriver.io) and Modl-test (https://modl.ai/our_products/modl-test/). It is notable that both these tools are specifically targeted towards the games industry and that they are both heavily focused on the Unity platform.

[1] MarketsandMarkets
(2020) – https://www.marketsandmarkets.com/Market-Reports/extended-reality-market-147143592.html
[2] Eurescomm message Summer 2022

Game Driver is based in California, USA and has recently received $2 million USD in investment funds (March 2022) . Their software presently requires to be run in the Unity Engine, which limits their audience to some extent and is in a Free Trial stage, so little can be gathered about their longer term commercial prospects, although their product roadmap aims to provide support for “multiple game engines, test frameworks, and execution platforms, including Unity, NUnit, Jenkins, MSTest, BitBar, and Oculus, with more to come as the product evolves”.[3] GameDriver’s publicity and available demonstrations on YouTube make no mention of an Agent-based or AI-driven system and to that extent their product architecture is likely more traditional. It could well be interesting to reach out to Game Driver towards the end of the project to determine if there might be interest in them licensing or embedding iv4XR outputs into their toolkit.

Modl-test is a tool developed and distributed via the Unity Asset Store or on a B2B basis by Modl.ai, a Copenhagen-based company specializing in artificial intelligence, machine learning and game development tools. Modl.ai has so far received $1.7 million in equity funding. The Modl.ai approach is closest to iv4XR’s to anything else that we have seen so far. They are engaging in typically straightforward game environments (e.g. exploration, platform and combat games) which implies the testing platform may still be at an early stage of maturity, but they have clear experience in game development as well as the AI domains they are employing to run their bot-based testing system. Again, it would be interesting to engage in a dialogue with this team in order to explore potential synergies both in terms of technology and the commercial potential of a future partnership or distribution arrangement. It could be advantageous that Modl.ai is a European-based company (which Game Driver is not).

Section 5 - Conclusions

Through the market survey discussed herein, web-based research, anecdotal evidence and internal discussion we can comfortably conclude that there is significant interest in advanced automated XR testing tools to replace or support labour-intensive, manual testing of complex XR applications.

Our findings indicate that the key markets for automated XR testing are as follows: Automotive, Aviation, Education, Training simulations, Gaming and Entertainment. In this document we have outlined the main reasons why the iv4XR solution is most applicable to these markets. We have also identified some potentially addressable markets at the leading edge of industries such as Construction, Healthcare and R&D.

[3] https://www.bloomberg.com/press-releases/2022-03-17/gamedriver-the-gaming-industry-s-first-out-of-the-box-automated-testing-solution-is-now-widely-available-receives-2

Whilst a number of commercial models are under consideration, digital distribution via engaged user portals/app stores such as Unity platform for Game Developers and digital creatives is high on the agenda for commercial review and consideration post project completion.

Some key questions remain, however. These will depend on the TRL level we are able to achieve at the project's end and how much further development, if any, will be required to bring the iv4XR solutions to market readiness. In other words, how much time, how much additional funding and what additional human resources would be needed to either launch a product or to make it available in some other, functional and documented form via a preferred Open Source license (BSD-3),

Consortium partners will conduct further research as to how the project's objectives match industry requirements. We shall also continue to discuss these issues, as well as the prospects for the consortium members themselves (jointly or severally) to manage post-project commercialisation, during the remaining 6 months of the project and intend to provide a summary of final conclusions reached within D6.6 (3rd Dissemination & Exploitation Plan) and the strategy for development sustainability post-project.

Acknowledgments

We would like to thank all those who participated in the market survey and by way of other conversations and contributions of their valuable know-how in the XR-space.

References

[1] Prada, R., Prasetya, I. S. W. B., Kifetew, F., Dignum, F., Vos, T. E., Lander, J., ... & Fernandes, P. M. (2020, October). Agent-based Testing of Extended Reality Systems. In *2020 IEEE 13th International Conference on Software Testing, Validation and Verification (ICST)* (pp. 414-417). IEEE.
[2] Alves, R., Valente, P., & Nunes, N. J. (2014, October). The state of user experience evaluation practice. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational* (pp. 93-102).
[3] Prasetya, I. S. W. B., Dastani, M., Prada, R., Vos, T. E., Dignum, F., & Kifetew, F. (2020, June). Aplib: Tactical agents for testing computer games. In *International Workshop on Engineering Multi-Agent Systems* (pp. 21-41). Springer, Cham.
[4] Vos, T. E., Aho, P., Pastor Ricos, F., Rodriguez-Valdes, O., & Mulders, A. (2021). testar–scriptless testing through graphical user interface. *Software Testing, Verification and Reliability*, *31*(3), e1771.
[5] Ferdous, R., Kifetew, F., Prandi, D., Prasetya, I. S. W. B., Shirzadehhajimahmood, S., & Susi, A. (2021, October). Search-Based Automated Play Testing of Computer Games: A Model-Based Approach. In *International Symposium on Search Based Software Engineering* (pp. 56-71). Springer, Cham.
[6] Shirzadehhajimahmood, S., Prasetya, I. S. W. B., Dignum, F., Dastani, M., & Keller, G. (2021, August). Using an agent-based approach for robust automated testing of computer games. In *Proceedings of the 12th International Workshop on Automating TEST Case Design, Selection, and Evaluation* (pp. 1-8).
[7] Prasetya, I. S. W. B., Shirzadehhajimahmood, S., Ansari, S. G., Fernandes, P., & Prada, R. (2021, April). An Agent-based Architecture for AI-Enhanced Automated Testing for XR Systems, a Short Paper. In *2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)* (pp. 213-217). IEEE.
[8] Fernandes, P. M., Jørgensen, J., & Poldervaart, N. N. (2021, December). Adapting Procedural Content Generation to Player Personas Through Evolution. In *2021 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 01-09). IEEE.
[9] The cost of poor software quality in the us: A 2020 report, 2020. URL: https://www.it-cisq.org/pdf/CPSQ-2020-report.pdf.

Further Reading

To learn more about the project and receive the latest updates, check out the website and follow us on twitter and facebook: https://iv4XR-project.eu

@iv4XR

https://www.facebook.com/iv4XR

For our scientific publications check Zenodo:

zenodo https://zenodo.org/communities/iv4XR-project/?page=1&size=20

Other resources:

Automotive and XR technologies https://europe.autonews.com/

International Air Transport Association (IATA) IATA - Home
Flight simulators simulator AVIATION NEWS International Aviation News, Airshow reports, Aircraft facts, worlds largest Aviation Museum database. Civillian, Military & Space, We cover it All

APPENDIX 1 - QUESTIONNAIRE

Q1) Please indicate your company's size

Micro (Head Count < 10; turnover ≤ € 2 million)
Small (Head Count < 50; turnover ≤ € 10 million)
Medium (Head Count < 250; turnover ≤ € 50 million)
Large (Head Count > 250: Turnover > 50 milion)

Q2) What is your role in the company?

Q3) What extended reality (XR) systems does your company use, if any? Choose all that apply.

Virtual Reality
Augmented Reality
Mixed Reality
Other 3D Virtual Environments (e.g. simulations, games, CAD systems)
Other 2D Virtual Environments (e.g. browser-based data handling/analytics, 2D games/simulations)
Other…

Q4) What are XR technologies used for in your company? Please briefly list the main reasons for using XR technologies.

Q5) Is your company developing these systems independently or do you use systems developed by a third party?

We develop our own systems
We use systems developed by a third party
Other…

Q6) Please indicate your industry

Q7) How or why did your company decided to use third-party systems

Q8) How complex is the testing process? [1-5]

Q9) On average, what percentage of your software development cost is spent on testing?

Less than 5%
Between 5 and 10%
Between 10 and 20%
More than 20%
I don't know

Q10) How does the company test the XR systems to ensure everything is working properly? Select one item

Internal Test Department
External Testing House
Combination

Q11) How would you rate your company's current need for automated testing tools? [0-5]

Q12)How would you rate your company's interest in intelligent automated testing tools in the future? [0-5]