# PrismArch

## Deliverable No D6.2

## Report on evaluation plan and usability study

| | |
|---|---|
| **Project Title:** | PrismArch **-** Virtual reality aided design blending cross-disciplinary aspects of architecture in a multi-simulation environment |
| **Contract No:** | 952002 - PrismArch |
| **Instrument:** | Innovation Action |
| **Thematic Priority:** | H2020 ICT-55-2020 |
| **Start of project:** | 1 November 2020 |
| **Duration:** | 24 months |
| **Due date of deliverable:** | 31 December 2021 |
| **Actual submission date:** | 14 January 2022 |
| **Version:** | 0.5 |
| **Main Authors:** | Arun Selvaraj, Dinos Ipiotis, Kostas Kostalampros |

**Project funded by the European Community under the H2020 Programme for Research and Innovation.**

| | |
|---|---|
| Deliverable title | Report on evaluation plan and usability study |
| Deliverable number | D6.2 |
| Deliverable version | Final |
| Contractual date of delivery | 31 December 2021 |
| Actual date of delivery | 14 January 2022 |
| Deliverable filename | Prismarch_D6.2_Report on evaluation plan and usability study_v0.5 |
| Type of deliverable | Report |
| Dissemination level | PU |
| Number of pages | 49 |
| Work Package | WP6 |
| Task(s) | T6.2 |
| Partner responsible | SWECO |
| Author(s) | Dinos Ipiotis, Arun Selvaraj, Kostas Kostalampros |
| Editor | Dinos Ipiotis |
| Reviewer(s) | Martin Brosamle (ETH), Spiros Nikolopoulos (CERTH), Helmut Kinzler (ZHVR) |

| | |
|---|---|
| Abstract | The purpose of this report is to set the basic roadmap for evaluation and testing of the Prismarch platform. The software is being developed and will be released as a minimum viable product to satisfy the development requirements set on previous project deliverables.<br>The evaluation consists of two basic components for data gathering. The evaluation plan focuses on the software outputs |

| | |
|---|---|
| | and deliverables and how these compare against the predefined set of minimum requirements. There is a more technical approach to the evaluation plan and outputs are easier to measure and quantify. The second component is the usability study which aims to measure and quantify the interaction and user experience with the software, in a diversified crowd of users. As an outcome of the evaluation and usability process, a set of measurable quantitative and qualitative outcome measures will be produced, aiming to assess the overall software performance, as well as serve as future input to improve and optimise Prismarch. |
| Keywords | Evaluation, Usability, Testing, Trial, Methodology, Software, AEC, VR |

## Copyright

1. ETHNIKO KENTRO EREVNAS KAI TECHNOLOGIKIS ANAPTYXIS (CERTH)
2. UNIVERSITA TA MALTA (UOM)
3. ZAHA HADID LIMITED (ZAHA HADID)
4. MINDESK SOCIETA A RESPONSABILITA LIMITATA (Mindesk)
5. EIDGENOESSISCHE TECHNISCHE HOCHSCHULE ZUERICH (ETH Zürich)
6. AKT II LIMITED (AKT II Limited)
7. SWECO UK LIMITED (SWECO UK LTD)

## Deliverable history

| Version | Date | Reason | Revised by |
|---|---|---|---|
| 0.1 | 10/12/2021 | Table of Contents | Dimitrios Ververidis (CERTH) |
| 0.2 | 09/01/2022 | Initial review by consortium members | Jeg Dudley (AKTII)<br>Daria Zolotareva (ZHVR) |
| 0.3 | 11/01/2022 | Internal review in the context of quality assurance | Martin Bromsale (ETH)<br>Spiros Nikolopoulos (CERTH)<br>Helmut Kinzler (ZHVR) |
| 0.4 | 13/01/2022 | Internal review complete. Final edits based on reviewers comments | Arun Selvaraj (SWECO)<br>Kostas Kostalampros (SWECO)<br>Dinos Ipiotis (SWECO) |
| 0.5 | 14/01/2022 | Submitted | Dinos Ipiotis (SWECO) |

## List of abbreviations and Acronyms

| Abbreviation | Meaning |
|---|---|
| AB | Advisory Board |
| BIC | Bank Identifier Code |
| CA | Consortium Agreement |
| CFS | Cost Financial Statement |
| DoA | Description of Action |
| DR | Deliverable Responsible |
| EC | European Commission |
| GA | Grant Agreement |
| GNU | GNU is not Unix |
| IBAN | International Bank Account Number |
| IoT | Internet of Things |
| IP | Intellectual Property |
| IPR | Intellectual Property Rights |
| NDA | Non-Disclosure Agreement |
| PC | Project Coordinator |
| PHP | PHP: Hypertext Preprocessor |
| PM | Person-Month |
| PMB | Project Management Board |
| PTM | Project Technical Manager |
| R&I | Research and Innovation |
| SB | Project Supervisory  Board |
| SBM | Supervisory  Board Member |
| ToC | Table of Contents |
| UML | Unified Modeling Language |
| QMR | Quarterly Management Report |
| WP | Work Package |
| WPL | WP Leaders |
| AEC | Architecture, Engineering and Construction |
| AR | Augmented Reality |
| BIM | Building Information Modelling |
| CAD/CAM | Computer-Aided Design & Computer-Aided Manufacturing |
| ICT | Information and communication technology |
| NDA | Non Disclosure Agreements |
| SME | Small and Medium-sized Enterprises |
| UG | User Group |

| | |
|---|---|
| VR | Virtual Reality |

# Executive Summary

The rationale behind D6.2 is to provide an outline of the evaluation plan and usability study of the PrismArch immersive design environment currently under development. The report analyses the existing methods of evaluating a virtual reality platform, ways to capture and collect the information provided by the trial users and how to improve the different tools of PrismArch.

The deliverable covers analysis of different software methodologies and how the software testing modules are formed to deliver a successful product. More specifically, the deliverable defines an evaluation plan, usability study and analyses how to capture the data. Detailed descriptions and examples are given to clarify the intended goals while conducting the evaluation plan and usability study.

Further analysis on the quality and ethics is provided where we make reference to the publication policy to support the arguments. To better facilitate the needs and understanding of what the evaluation plan and usability study shall cover, the usage scenarios described in D6.1 (cf. D6.1) are reflected in this deliverable, and the relevant AEC partners' opinions on the evaluation plan and usability study should cover from their perspective. The deliverable concludes with a summary of the analysis and how the evaluation plan and usability study contributes to the overall development of PrismArch.

# Table of Contents

## List of Figures

## List of Tables

# 1. INTRODUCTION

Whenever a piece of software is developed, tests will be conducted before releasing the software to the public. Several trials will be performed while developing the software, user experience data collection, observations of how the software operates and lots of other factors that determine whether a piece of software works as intended, or not. This deliverable aims to identify the processes of evaluation and usability that need to be undertaken while PrismArch is being developed. This process will focus on the software usage, how easy it is to use the different software tools, and the user's experience within the PrismArch platform. The evaluation plan and usability study shall target audiences from multiple disciplines involved within the construction industry. However, the first trials will focus on architects, structural and MEP engineers. As PrismArch evolves, more disciplines are intended to get involved. The language used to communicate throughout the process shall be the English language. PrismArch aims to develop a prismatic blend between aesthetics, simulation models and meta-information that can be presented in a contextualised and comprehensive manner in Virtual Reality (VR) to allow collaborative manipulation of the design and accurate assessment of new design decisions. To achieve this, an evaluation plan is necessary. PrismArch is not age restrictive. However, we strongly encourage participants to be adults. Although experience within the AEC industry will not be required to use the platform if it becomes a commercial product, encouraging testing from people with relevant experience either in an engineering discipline or computer-based environments is mandatory. The document contains several subjects. The first one explains the evaluation plan and the actions required for effective evaluation of the platform, the tools to be evaluated, ways to capture the necessary data and how this data will be used to extract results. Subtopics within the document separate these to provide clear and detailed information, making it easier to execute the evaluation plan initially and throughout the evolution of PrismArch.

The second one investigates the usability study, user experiences, ways to improve from a user interface and experience perspective, and the users' emotions while using the PrismArch platform. The usability study does not focus on tools. It focuses on the immersive human, the human itself, and the aesthetic ecosystem PrismArch offers. Similarly to the first subject, this contains relevant information about the usability study and how this shall be conducted.

The third subject discusses the evaluation plan’s  quality control and quality assurance factors and the usability study. The relevant factors are described giving a better understanding to the reader of what measures have been taken so that the evaluation plan reaches the desired quality and value.

The fourth subject describes the ethics and ethical factors considered while forming the evaluation plan and usability study, ensures that the trials will not breach any ethical aspects and that the evaluation plan and usability study will be conducted smoothly from a moral perspective.

Subject five aims on describing the usage case scenarios and the evaluation in action. The three disciplines involved will briefly describe the usage case scenario and what each discipline aims to evaluate within PrismArch.

Further testing is discussed in chapter six. This analysis is driven by the computer gaming industry and PrismArch’s intention to investigate and adopt such tests.

Chapter seven concludes the evaluation plan and usability study deliverable by summarising what is aimed and the benefits of conducting the trials.

# 2. TERMINOLOGY

## 2.0 DEFINITION OF ASPECTS FOR SOFTWARE TESTING

### 2.0.1 Definition of the evaluation plan

Developing an evaluation plan includes a series of actions and methodologies. Krenn, H., (2017) recognises the evaluation profession as multifaceted and characterised by its theoretical debates, ethical considerations and proprietary interests. Community toolbox (2021) describes the evaluation plan to inspect and monitor the initiative progress. There might be a need for adjustments, so an evaluation plan is needed to identify and mitigate any processes to keep the profitability of the initiative. Additionally, if the initiative is an utter failure, the evaluation plan will determine, and then the consortium needs to figure out a way to cut the losses. An evaluation plan is a complex process. A methodology of simply getting a handbook and following its steps should not be adopted. The handbook methodology might end up with lots of useless data, which will merely cause waste. However, there are several reasons why an evaluation plan should be developed:

- The plan lays out the several steps of the evaluation process
- It helps on inform content decisions really needed
- Prevents wasted time from unnecessary data gathering
- Identifies the best possible methods and strategies for data capture
- Improvement of PrismArch

The evaluation plan can be expanded to several other forms. Nie et al. (2021) state the evaluation plan as the way to obtain training, validation and testing sets. Although this way of evaluation was developed for machine learning code learning tasks we can identify similarities to evaluation plans used in other industries. For example, Law Insider (n.d.) defines the evaluation plan as the following series of actions:

- Objectives of the scheme to be evaluated
- Evaluation Questions
- Result indicators
- Envisaged methodology to conduct the evaluation
- Data collection requirements
- Proposed timing of the evaluation
- Description of the independent body conducting the evaluation
- Modalities for ensuring the publicity of the evaluation

Furthermore, Brown University (n.d.), defines the evaluation plan as an integral part of a grant proposal that provides information to improve a project during development and implementation. To this point, an evaluation plan is vital to improve the initiative, understand any occurred issues and further develop the requirements of the project outcome efficiently and feasibly.

The evaluation plan will follow similar strategies and methodologies. It will also determine if PrismArch delivers what has been discussed, researched, compiled and executed.

### 2.0.2 Definition of the usability study

Apart from the evaluation plan, the need to focus on the actual users and how they interact with PrismArch is mandatory. Interaction Design Foundation defines the usability study (or usability testing) practice of testing how easy a design is to use with a group of representative users. This may include methods from simple observations of the users as they attempt to complete tasks to complex reports defining the successful usage of the final product. A usability study may contain the following:

- If the testers complete tasks successfully and independently
- How long does it take to complete a task
- Assess the performance and mental state as they try to complete the task aiming to evaluate the effectiveness of the design
- Is PrismArch an enjoyable experience
- Identify changes
- Identify problems and their severity
- Find solutions

In a virtual reality environment, the existing literature defines the usability study to identify users' cognitive performance, users' interaction performance and the system suitability to design review practises (Paes, D., Irizarry, J., 2018). Overall, the usability study focuses on the users' interactions and the ways to improve PrismArch so it becomes an experience capable of solving the construction industry's countless design issues under the virtual reality immersive environment.

### 2.0.3 Role of the participant

The participant's role is to understand the requirements either with the assistance of the documents and training or by improving the skills required to execute the tests after getting the necessary information by trying, discussing and resolving queries with the consortium. The participant needs to understand the test plan. What the process is, what actions will need to be undertaken to perform the test and the overall process. The participant will also need to be trained and properly communicated on how to submit the information after the test is executed. The participant shall also submit any bugs found through appropriate methodologies (Appendix D).

### 2.0.4 Software

As software, we define the tool or interface which an individual uses in a computer, tablet or smartphone to interact and do specific work (Ujagare, A., 2019). PrismArch is considered a revolutionary window-based software that intends to raise the bar by combining virtual reality and window-based applications.

### 2.0.5 Software testing

IBM (2022) defines software testing as the process of evaluating and verifying that a software product does what it is supposed to do. Additionally, the definition of software testing is categorised into these types:

- Acceptance testing: Verifying whether the whole system works as intended.
- Integration testing: Ensuring that software components or functions operate together.

- Unit testing: Validating that each software unit performs as expected. A unit is the smallest testable component of an application.
- Functional testing: Checking functions by emulating business scenarios based on functional requirements. Black-box testing is a common way to verify functions.
- Performance testing: Testing how the software performs under different workloads. Load testing, for example, is used to evaluate performance under real-life load conditions.
- Regression testing: Checking whether new features break or degrade functionality. Sanity testing can verify menus, functions and commands at the surface level when there is no time for a full regression test.
- Stress testing: Testing how much strain the system can take before it fails. Considered to be a type of non-functional testing.
- Usability testing: Validating how well a customer can use a system or web application to complete a task.

Furthermore, the importance of software testing is analysed based on the following factors:

1. Find architectural flaws
2. Eliminate poor design decisions
3. Errors in functionality
4. Issues with Scalability

Additionally, and in a more simple definition, software testing is an activity of evaluating or exercising specific software to find bugs in it (Ujagare, A., 2019). However, software testing is not a simple process. It follows certain principles to be executed successfully. Testing is based on context. Patterns cannot be the same for window-based and web-based software; therefore, testing is different based on the context of the software.

Software testing cannot cover everything. Based on time and costs, the tests need to cover a specific part of the software. However, it cannot cover all options, combinations and functions. For example, if a password contains a combination of eight letters, symbols, uppercase and lowercase letters, it would be impossible to test all combinations.

Testing should begin as early as possible. The reason is that the earliest the bugs are found, the fewer costs will be involved to resolve them.

Defect clustering should be considered. If an error is found in a specific area of the software, then the adjacent functions should also be considered to be tested.

Software testing should evolve. If a test is performed repeatedly until all bugs are resolved, the test will not raise any new errors. New scenarios need to be developed to include different functions and interactions when the software develops and includes the bug fixes.

Although a test is conducted to reveal errors and fix any issues within the software, it does not disclose the existence of an error or bug. The only way to eliminate a bug's probability is to perform software evaluation plans and usability studies constantly. Therefore PrismArch will need to be under trial to ensure all issues are continuously resolved.

### 2.0.6 Error terminology

To prevent misunderstanding, the definition of an error is driven in this deliverable by Ujagare, A., (2019). More precisely:

**Error**: A mistake in code or program

**Defect**: An error found by a tester

**Bug**: Defect accepted by the software developer

**Failure**: The Build is not meeting with requirements

**Fault**: Every unexpected corruption within the software after launching in a market

### 2.0.7 Software Development Lifecycles

Every software developed goes through a procedure that contains certain phases. The relevant phases are presented in Figure 2.0.7.1 and analysed below:

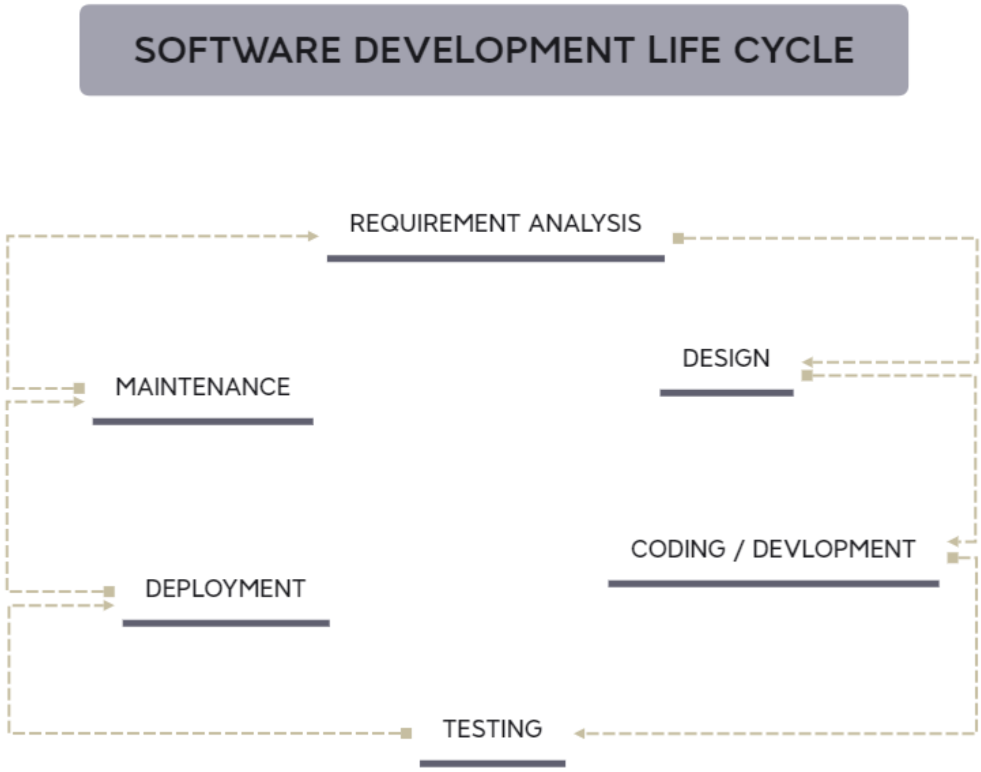


*Figure 2.0.7.1: Software Development Life Cycle*

1. Requirement analysis: Analysis of software requirements, the intention of the software, purposes the software will serve etc. These requirements have been extensively covered in the Work Packages and Deliverables researched and conducted by the consortium.
2. Design: The project architecture, the database used, the user interface, and the data flow are just examples of the design phase of software development. The design phase is covered through relevant documentation, meetings and execution of tasks.

3. Coding: After the design is evaluated and agreed the coding process starts where the software developers build the different modules assigned to them, creating the actual software.
4. Testing: After completing the coding process, testing and evaluation begin to verify that the software meets the requirements and standards set in the design process.
5. Deployment: Once testing is complete and the software is approved to be released to the public, the deployment process occurs. This is the actual delivery of the software to the client.
6. Maintenance: Any issues the clients may return, any enhancement requests or requirements update falls under the maintenance phase of the product.

### 2.0.8 Software development models

Effectiveness in software development is vital to reduce development costs and increase profitability. PrismArch adopts the prototype development model, which is described in the following paragraphs alongside other development models, to help better understand the evaluation process. The development models investigated, as defined by Ujagare, A., (2019), are the following:

- Waterfall method
- Prototype Model
- Incremental Development Model
- Spiral Model
- Agile Model

#### 2.0.8.1 Waterfall method

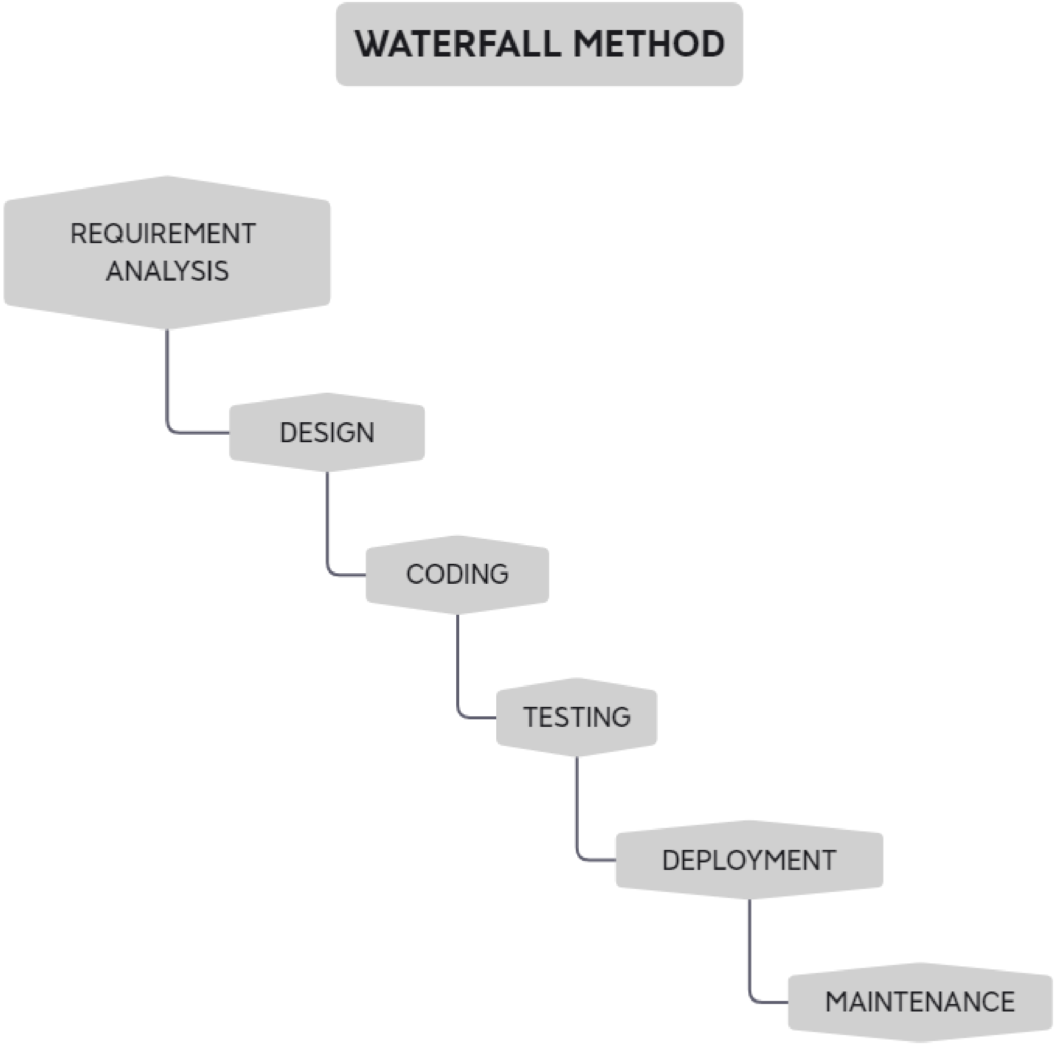


*Figure 2.0.8.1.1: Waterfall Method*

The sequence in a waterfall method is irreversible. Similar to a waterfall, the process cannot go back. Once a subject is complete, the next one starts, and we cannot revert to make adjustments. The requirement analysis primarily applies to software not driven by a client's requirement but on the software's actual product. As an example, engineering software produces engineering design solutions. Consultancies test the software and adopt the one which suits their needs. After an accurate understanding of the requirements, design can be based on three factors:

1. Admin (e.g. Add/remove users, add/remove customers, add/remove permissions)
2. Operations (e.g. Create/ Delete bills, Send/ Receive bills, Add/ delete products)
3. Reports (e.g. Create / Delete reports, Pending payments and chase of debt, Financial reports)

Further details can be added under each factor (e.g. Add user name credentials, contact details etc.). After completing the design, the coding begins in the agreed language (e.g. PHP). The conclusion of coding moves the process to test to identify any errors. Once errors are identified as bugs and are resolved, the deployment process brings the product to the client. Finally, maintenance actions are undertaken after the policies and agreement between the client and the software developer. However, there are a couple of advantages and disadvantages identified in the process. Table 2.0.8.1 demonstrates a summary:

| Advantages | Disadvantages |
|---|---|
| Quite a logical process, easy to understand and digest | No modifications are allowed upon completion of the stage |
| Useful for product-based software with fixed requirements | Requirements need to be precise. Otherwise, the effectiveness is affected |
| Clearly defined stages of development | It might become expensive since tests are performed after completion of coding and bugs identified later rather than sooner. |
| Task arrangement is straightforward and contributes to the achievement of milestones. | Not suitable for complex projects |
| Allows for appropriate documentation | Not suitable for projects where the requirements frequently change |

*Table 2.0.8.1.1: Advantages and disadvantages of the Waterfall Model*

### 2.0.8.2 Prototype Model

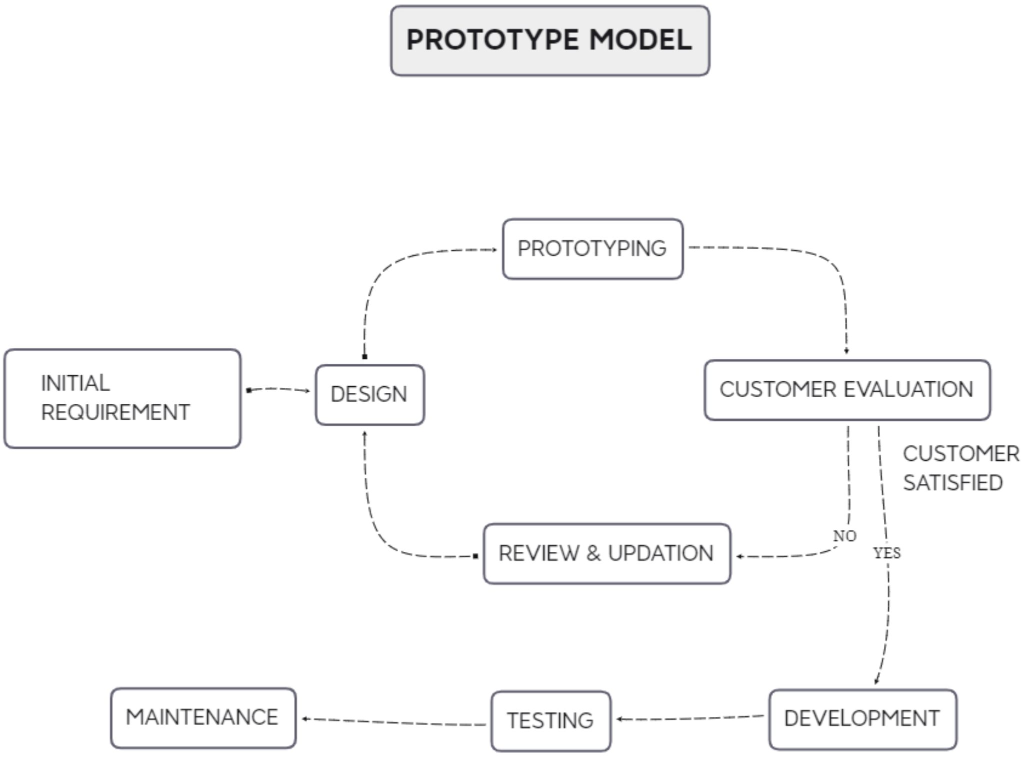


*Figure 2.0.8.2.1: Prototype Model*

The prototype model fits the PrismArch software development methodology. The client creates the initial requirement based on this design, and the developer creates a prototype. This prototype is presented to the customer to understand the product better. After the evaluation and review process, the developer updates the product. This cycle is repeated until the customer is satisfied with the product and after the final approval the product moves for finalisation of the development. The process then proceeds to software testing for bugs identification, resolution, and deployment. As already mentioned, advantages and disadvantages come with every method. Table 2.0.8.2.2 demonstrates these:

| Advantages | Disadvantages |
|---|---|
| A clear understanding of the software to be developed from both the users and the developers | A time-consuming process, especially prototype preparation |
| Early identification of errors | The developer prefers the client not to have too much involvement |
| Quick user feedback gives better solutions | The flow of the development might be disturbed by the client’s engagement and changes suggestions |

| | |
|---|---|
| Easy identification and resolution of missing functionalities | Time and effort falls to the developer regarding costs |
| | The client may lose interest in doing business with the developer if the latter does not satisfy the initial prototype requirements. |
| | The client often perceives the time frame needed to deliver the final product after the prototype is presented. |

*Table 2.0.8.2.1: Advantages and disadvantages of the Prototype Model*

### 2.0.8.3 Incremental Development Model

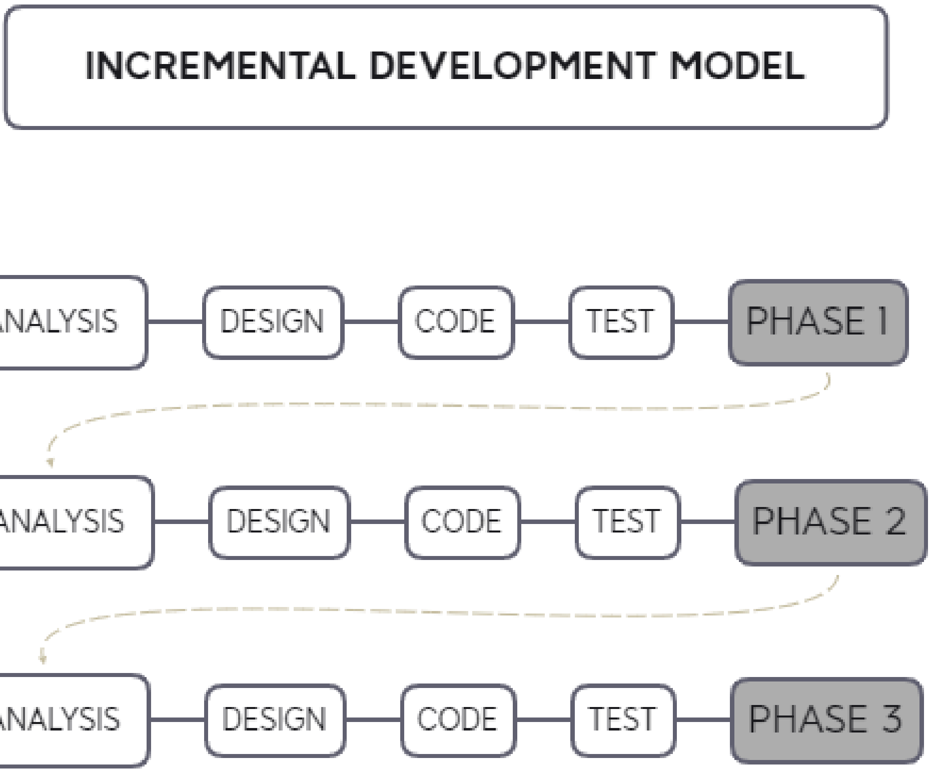


*Figure 2.0.8.3.1: Incremental Development Model*

The incremental development model divides the whole requirement into various builds. More easily managed modules form multiple waterfall cycles, implemented in phases throughout the development process. All the critical phases (analysis, design, code etc) are applied in each module. A working version of the software is produced in each phase resulting in early working conditions of the software. As the modules are completed, an incremental completion of the final product is achieved. However, this is not fully accomplished until all modules are completed. Advantages and disadvantages of this process are demonstrated in Table 2.0.8.3.1:

| Advantages | Disadvantages |
|---|---|
| The early generation of working software during the life cycle | Needs detailed planning and proper design |
| Flexibility and fewer costs related to requirements changes and further scope | The whole system needs to be identified and understood. Otherwise, the incremental breakdown will be unsuccessful. |
| Test and debug become more accessible due to smaller iterations | Higher total cost compared to the waterfall model |
| The customer can evaluate each built | |
| Lower initial delivery costs | |
| Easier management of risks due to identification while the module is processed. | |

*Table 2.0.8.3.1: Advantages and disadvantages of the Incremental Development Model*

#### 2.0.8.4 Spiral Model

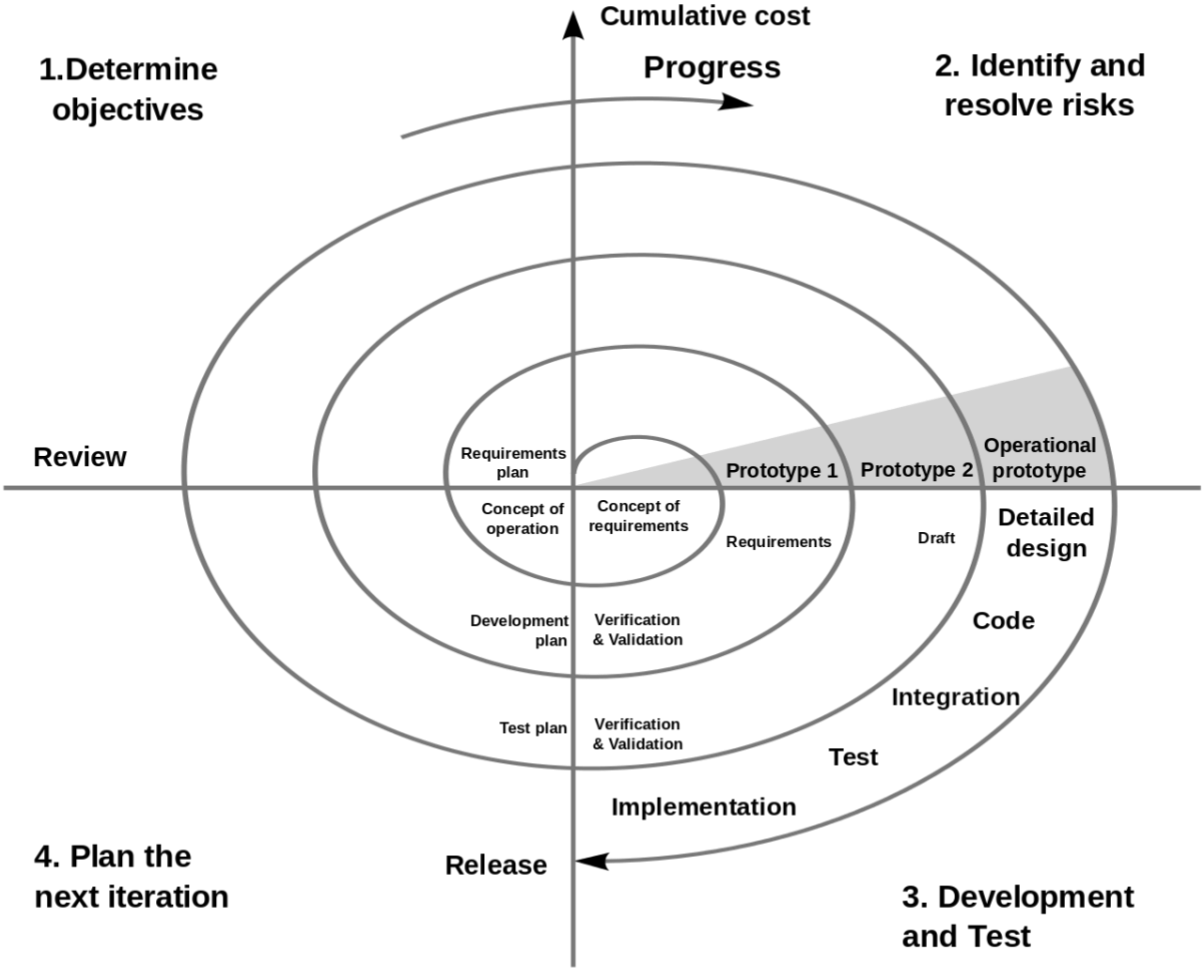


*Figure 2.0.8.4.1: Spiral Model*

The spiral model inherits elements from the waterfall and the prototype model. There are four steps involved and executed repeatedly. The spiral model is implemented when a risk is involved in a project. A risk may be an employee on sick leave or a hardware failure (Ujagare, A., 2019). Additionally, the spiral model can be implemented when there are alternative ways for project implementation or a fixed budget is not agreed upon for the project. The gathering of requirements is executed between the communications held by the customer and the system analyst. The risk analysis is planned as per the requirements, the development and testing are performed to get the customers feedback. As the spiral iteration becomes subsequent, the built is sent again to the customer for feedback. The evaluation is performed, and planning for the next iteration is fed to the development team. The process is falling into a spiral as it evolves. Initially, the requirements are defined, the draft is executed, and the detailed design occurs. In each phase, the product is verified and validated to move from the test plan to the development and operational procedures. The advantages and disadvantages are inevitable. Therefore Table 2.0.8.4 demonstrates these:

| Advantages | Disadvantages |
|---|---|
| Risk avoidance in early stages since there is a high amount of risk analysis | Spiral model is costly |
| Ideal for large and/ or mission-critical projects | The expertise must be high for the risk analysis |
| Even if the project progresses, additional functionality can be added | The risk analysis is the most important factor for the project's success |
| Extensive use of prototyping is allowed | Not suitable for small projects |
| Early investigation of the system by the client | Complicated project management and project implementation |
| Early identification of risks and managed since the project is divided in smaller parts | Probability of indefinite spiral and inability to set exact date of project completion |

*Table 2.0.8.4.1: Advantages and disadvantages of the Spiral Model*

### 2.0.8.5 Agile Model

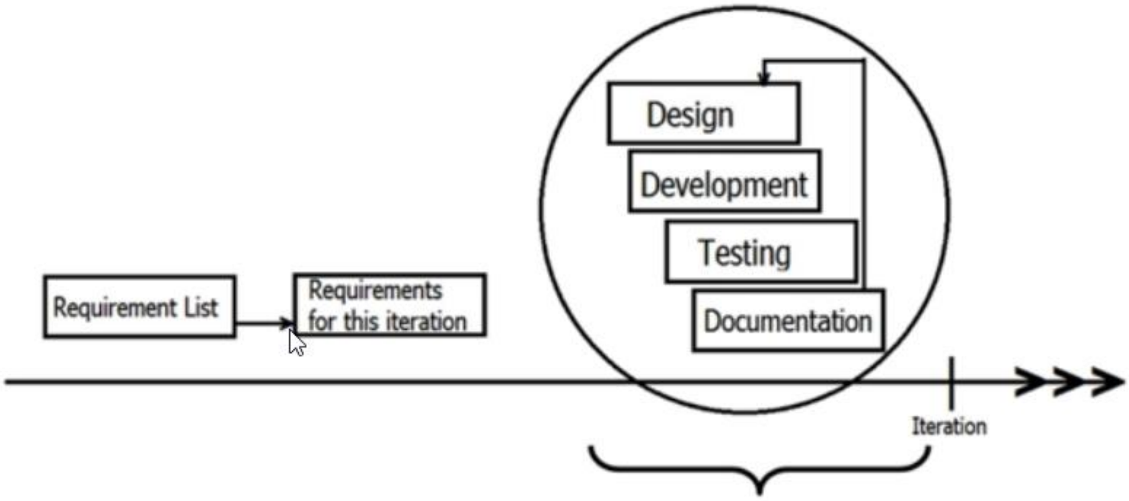


*Figure 2.0.8.5.1: Agile Model*

The agile model breaks tasks into small parts. This results in minimal planning requirements. These short iterations are presented to the client. The client evaluates and either approve or identifies changes minimising the risk. The project is able to adapt to time-critical applications. The small iterations are called sprints. A good example is the one where within a working week four days are captured for software development where the fifth day is used for client review. The client was given a working project to evaluate. Finally, the advantages and disadvantages of the Agile Model are presented in Table 2.0.8.5.1:

| Advantages | Disadvantages |
|---|---|
| The phases are completed quicker | The project can drive away from the initial requirements if the developers do not have clear information and instructions |
| There is an easier adaptation to requirements amendments since client is involved in each iteration | Decisions and expertise require highly skilled staff |
| Late changes are easier to implement | |
| Testing team tests the software easier due to the nature of the information and software development flow | |
| Early investigation of the system by the client | |
| Early identification of risks and managed since the project is divided in smaller parts | |

*Table 2.0.8.5.1: Advantages and disadvantages of the Agile Model*

## 2.1 CONTENTS OF EVALUATION PLAN

### 2.1.0 Testing life cycle

Although the previous chapter analysed different methodologies, there are some common terms and processes used. These are also applied through the software testing process. Regardless of the software methodology implementation, the software testing life cycle is a common process. To successfully test PrismArch the following modules need to be developed and executed:

- Requirement analysis
- Test Planning
- Test Designing
- Test Environment Setup
- Test Execution
- Bug reporting

To better understand the modules, Figure 2.1.0.1 demonstrates the flow of testing software life cycle:

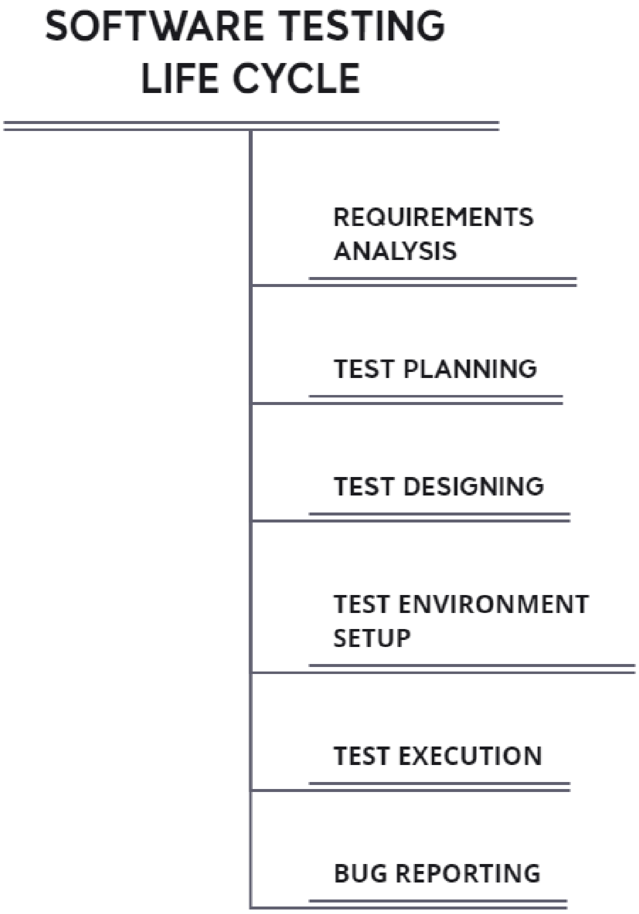


*Figure 2.1.0.1: Software Testing Life Cycle*

**Requirement analysis:** Purpose of the software. What is PrismArch, how it will benefit the users and aid in the successful execution of given tasks.

**Test Planning:** The test planning includes several iterations:

1. How the tests will be conducted
2. What will be the test strategy
3. Definition of the test enjoinment
4. Testing methodologies
5. Hardware/Software requirements
6. Test accomplishment requirements

**Test designing:** The part where the consortium members design the tests to be conducted

**Test Environment Setup:** What will be the test environment. Whether the testing environment be held locally to the participant's terminal or held within a server. The testing environment setup also determines if software installation is required or if it will be held within a virtual desktop infrastructure replicating the end-users environment.

**Test Execution:** This is the part where the tests are conducted to identify whether the software meets the requirements or not.

**Bug reporting:** This is the process of the creation of a report which describes the error and is forwarded to the developer team for addressing.

The software testing cycle is a waterfall model and has been further developed to the V Model. The V model is an extension of the waterfall model, where testing starts parallelly. Figure 2.1.0.2 demonstrates the V Model:

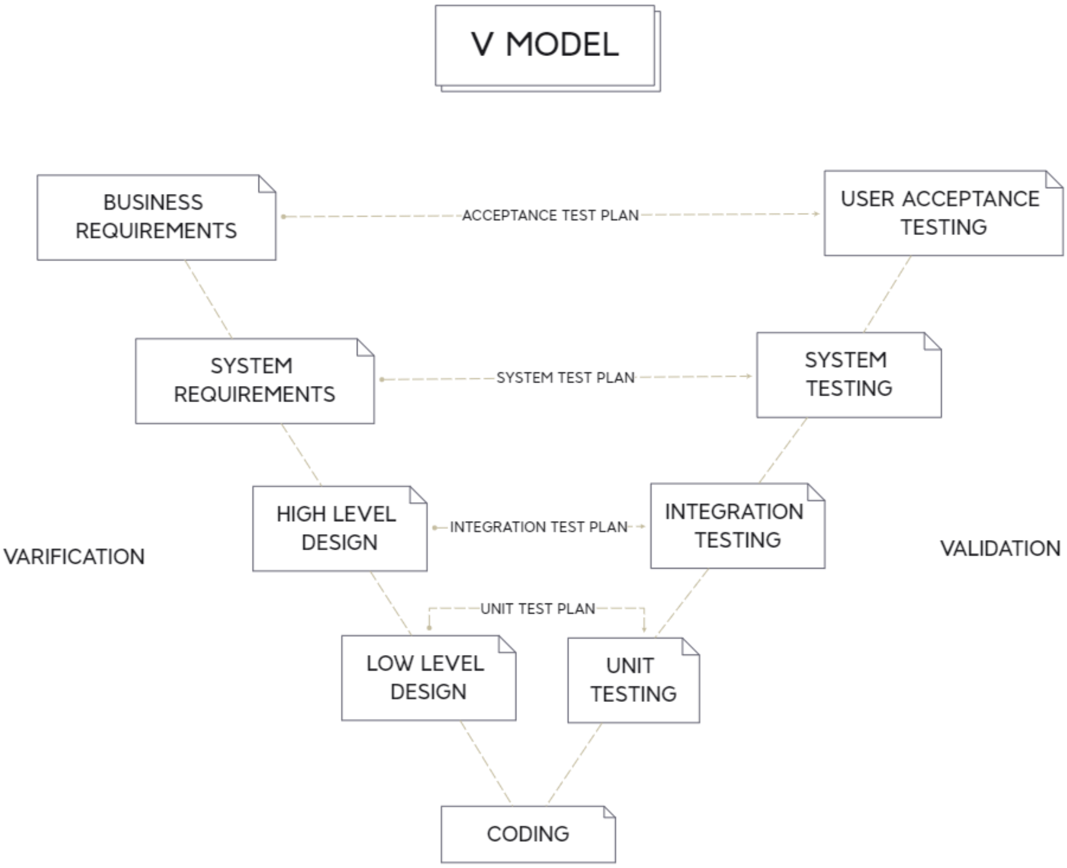


*Figure 2.1.0.2: The V Model*

The left arm of the V model demonstrates the conventional waterfall model of the development. The right arm represents the testing levels (Ujagare, A., 2019). The left arm contains the following modules:

- Business requirements and user acceptance settings: This includes the gathering of requirements from the client and once these are captured the acceptance plan is formed. Once the software is being developed, the testing evaluates the successful implementation of the requirements.
- System requirements and system testing: This involves the system requirements documentation and the testing executed based on these requirements.

- Integration testing at High-Level Design: High-Level design involves the system architecture and the database design. Nevertheless, the High-Level Integration testing needs to be able to add related modules as these are being progressed.
- Unit Testing at Low-Level Design: This covers the small parts of code testing, where small modules are formed instead of the wider software architecture.
- Coding: The interconnection of the modules, the formulation of the final software product is included in the Coding module.

The V Model is ideal for easily understood requirements and small projects. The simplicity of usage of the V Model makes it straightforward and easy to adopt. Compared to the waterfall model, the V model can include tests for the design in earlier stages. Thus said, a significant amount of time can be saved allowing for more space for success. However, the V Model is not flexible. Early prototypes simply do not exist within a V Model and the product is being developed during the implementation phase. Additionally, the testing and requirements documentation needs to be updated in case changes happen while the development moves forward.

Since the V Model is better for the design in earlier stages, PrismArch shall inherit the V Model. The Platform up to this day is still under development and there isn't a fully working prototype available.

### 2.1.1 Objective

The evaluation plan is a set of processes to test PrismArch and the relevant tools created to aid the design solution. The evaluation plan will conduct specific methodologies for testing the actual software tools, their efficiency and whether they serve the actual purpose they were designed for. The overall plan will be based on D1.1(D1.1, 2020) and D6.1 (cf. D6.1)deliverables and the relevant contents. D1.1(D1.1, 2020) describes the software tools within PrismArch and the usage, while D6.1 (cf. D6.1) describes the usage scenarios in which PrismArch will be tested. As defined in D1.1,(D1.1, 2020) the PrismArch tools are the following:

- Multi presence on-boarding system
- Tagging tool
- Query tool
- Dashboard tool
- Admin tool
- Contact / Communication tool
- Toggle Camera Perspective tool
- Toggle View mode tool
- Multi Selection tool
- Speech to text / Typing tool
- Commenting and Mark-Up tool
- List Maker tool
- Whiteboard tool
- Clipping Plane tool
- Spatial Orientation tool
- Design Support and Evaluation tool

The usage scenarios described in D6.1 (cf. D6.1) requires a combination of the usage of the tools described above. However, the tools will be separated based on the individual reports set up for the evaluation plan. The usage case scenarios will define a subset of tools to be tested. As the tools are constantly developed the testing phase shall be adjusted accordingly.

### 2.1.2 Selection of Evaluation Plan methodologies

A series of actions, including polls, interviews, and surveys will facilitate the evaluation plan. Polls are an effective way to listen to people. Although the broader public polls might not be the best way to gather information, business leaders and companies often use them to get people's insight (Dionne Jr, E., Mann, T., 2003). Polls shall be used to identify PrismArch's essential functions' effectiveness quickly. The polls should contain information as boolean operators (yes/no, true/false) to answer the fundamental question of whether the tool works or not.

In the broad sense, interviews are the process whereby individuals exchange information (Trull, S., 1964). Interviews in the instance of the evaluation plan will form a better way of communication between the interviewer and the interviewee, focusing on the usage of PrismArch tools. The interviewees will sign a consent form stating that they are willing to participate voluntarily and agree with the Terms and Conditions applied to the evaluation plan. The themes of the evaluation plan interviews shall cover whether the tools are easy to use, the way the tools are designed aided in the resolution of the usage scenarios, and the interactivity and complexity of performing a task. The themes shall investigate the user interface, deploy different tools, and evaluate if the tools perform as expected or if there are any glitches or defects while the tools are used and whether the overall result is fluent and straightforward.

A survey is a research method used for collecting data from a predefined group of respondents to gain information and insights into various topics of interest. They can have multiple purposes, and researchers can conduct them in many ways depending on the methodology chosen and the study's goal. An online survey is a set of structured questions that the respondent completes over the internet, generally filling out a form. It is a more natural way to reach out to the respondents as it is less time consuming than the traditional way of gathering information through one to one interaction and is less expensive (QuestionPro, 2021). Within PrismArch, online surveys will be conducted as part of the evaluation plan. The surveys will contain arguments on the usage of tools. Scales of satisfaction will be set up to facilitate the surveys, and selectable bullets will determine the user's selection. An example of a survey is demonstrated in Appendix 1.

### 2.1.3 Specifications

To facilitate the evaluation plan based on polls, interviews and surveys, the following platforms and software are proposed:

1. Doodle Poll Maker (https://doodle.com/poll-maker): This is a freeware tool that can be used to create online polls. As mentioned, the poll’s aim is to gather information through boolean variables (Yes/No, True/False). The volunteers will give a high-level overview of PrismArch, the usage and user experience.
2. Interviews: The interviews will be held either physically or online. The online interviews shall be held via Zoom Calls or Microsoft Teams. The interviews shall contain relevant themes over the evaluation of PrismArch with a structured

methodology based on the usage scenarios. A consent form shall be submitted from the interviewer to the interviewee before the interview is conducted. The interviewee will need to approve voluntary testing of the PrismArch platform and consent to the interview recording and data usage for results extraction. The interview shall be recorded and transcribed based on the provided tools by the software.

3. Surveymonkey (https://www.surveymonkey.com/): SurveyMonkey is one of the most popular platforms for online surveys. The surveys shall cover the satisfaction rates from the tools usage and the users' satisfaction. Appendix A demonstrates a Survey template and Appendices E and F focus on specific tools. The values will contain levels of satisfaction or agreement. SurveyMonkey also hosts Wufoo (https://app.wufoo.com/forms) as an online form creation tool. The next tables demonstrate the levels of satisfaction and agreements to be implemented in the surveys:

| | | | | |
|---|---|---|---|---|
| Strongly Disagree | Disagree | Neither Agree Nor Disagree | Agree | Strongly Agree |

*Table 2.1.3.1: Survey agreement levels*

| | | | | |
|---|---|---|---|---|
| Strongly dissatisfied | Dissatisfied | Neither Satisfied Nor Dissatisfied | Satisfied | Strongly Satisfied |

*Table 2.1.3.2: Survey satisfaction levels*

### 2.1.4 Procedure

All consortium members shall have the ability to follow the step-by-step evaluation process, access the information and results and be aware of the next steps. The procedure is broken down into the following subjects:

- Call for entries
- Training
- Access and testing PrismArch
- Evaluation Plan and Usability Study

#### 2.1.4.1 Participants - Call for entries

PrismArch aims initially to cover the needs of Architectural, Structural and MEP disciplines. The consortium members have already discussed the need to add more disciplines. However, this will need further development. Only volunteers from these three disciplines will initially participate in the evaluation and usability studies and more disciplines are intended to be added as the software is being developed. A call for entries will be conducted within the consortium and afterwards through social media. All volunteers shall fill in a consent form (Appendix B) after receiving the information sheet template (Appendix C) and submit it to the consortium. The consent form will contain the authorisation of the consortium to gather, process and extract data from the conducted tests. Initially, there shall be five members from each discipline. The five volunteers' proposal can be expanded if needed, and more members can be added upon review. The duration of 'call for entries' should not exceed a calendar month, and as soon as the positions are filled, the process shall begin.

#### 2.1.4.2 Training

All volunteers shall receive training before the evaluation plan and usability study. The training shall be conducted online via video tutorials or Zoom, Teams or Skype workshops. The appointed consortium members shall demonstrate the usage scenarios and what actions are needed for each scenario. Additionally, the tools and how to perform the actions shall be demonstrated, and all queries from the study participants will be resolved before volunteers access PrismArch. Video tutorials shall not be lengthy and should not exceed two to three minutes. The online training shall be facilitated with a Q&A session, and the video tutorials shall be hosted online privately with comments active so that participants can comment on them. The users shall add their comments and resolve any further queries. The videos shall be private for the evaluation period and, if needed, can be agreed upon among the consortium to release those publicly at a later stage.

#### 2.1.4.3 Access and testing PrismArch

After the volunteers have received training and any queries have been resolved, the consortium shall grant access to PrismArch. The users shall be given instructions as per D5.2 (cf. D5.2) to where to access the executable file and start the evaluation process. Details shall be provided for further authorisation when needed, and the users shall begin the evaluation process based on the D6.1 (cf. D6.1) usage scenarios. The tools to be evaluated will be driven from D5.2 (cf. D5.2) and their availability. The testing process shall include the usage case scenarios as these are described in D6.1 (cf. D6.1). Section 6 of this deliverable contains the usage case scenarios as a preliminary version. As the platform is developed over the next months, the tests and context shall be updated and refined accordingly. Initially, a timeframe for the session duration shall not be given. However, a deadline shall be offered for submitting the information. The sessions can be monitored if needed, or the users may perform the tests independently. This shall be agreed upon during the process. After the users finish the tests, they should start the evaluation process.

#### 2.1.4.4 Evaluation Plan submission

Once the users finish the PrismArch access and testing, they should start the evaluation plan process. The users shall inform electronically that they have completed the tests and need the forms to fill. The consortium shall send the relevant links and information. Also, the communication shall determine the interview to be conducted and will push forward necessary actions. The time frame shall be determined to submit the information. Generally, a week shall be an adequate time frame to fill and submit the data since there should be a preference to have the experience fresh in the users' minds. The participants shall fill in the relevant polls, surveys and conduct the interview, and after they finish the overall process, they shall inform the consortium electronically.

## 2.2 CONTENTS OF USABILITY STUDY

### 2.2.1 Objective

The objective of the usability study is to capture the experience the users get by using PrismArch. A set of tools shall be set up to capture the user's emotions, mental state and satisfaction levels while using PrismArch. PrismArch intends to create positive emotions while using PrismArch and find the experience appealing. The tools shall aid through this achievement. However, the need to capture positive and negative moods, feelings and overall experiences are vital for evaluating the success of PrismArch. The usability study will capture the relevant information based on the usage scenarios. This shall include:

- Confirm that the tasks are understood by the users
- Positive or negative aspects the user encounters while using PrismArch
- If the requirements where clear
- Overall feeling of the user interface
- Where users get "stuck" with the usage of tools
- Ease of use of the modelling tools
- Overall experience while working within a virtual reality environment
- If sphering levels have a positive or negative impact as an innovative approach
- Motion sickness
- Ease of communication

### 2.2.2 Selection of Usability Study forms

The usability study will run parallel to the evaluation plan. Therefore, the same tools will be utilised to capture the necessary information. This shall include

1. Polls
2. Interviews (physical or online)
3. Surveys

The context of the usability study shall include subjects related to Chapter 2.2.1.

### 2.2.3 Specifications

The specifications of the usability study tools shall include the same tools as these are described in Chapter 2.1.3. More specifically:

1. Doodle Poll Maker
2. Interviews
3. SurveyMonkey

The context of the specifications shall be agreed upon and described in D6.3 (cf. D6.3) and after further discussions and contributions with the rest of the consortium members.

### 2.2.4 Procedure

The procedure shall follow the same pattern as described in Chapter 2.1.4. The flow of actions shall remain identical to what has already been identified. To be more precise:

1. Call for entries
2. Training
3. Access and testing PrismArch
4. Usability Study submission

## 2.3 OUTCOME MEASURES

The gathered information from the evaluation plan and usability study shall be processed to understand better the user's experiences and the effective design of the tools. The outcome measures shall be communicated to the consortium members in various ways. Apart from the native information, data analytics tools shall visualise the data, in turn providing us with a better understanding of user behaviour and possible issues. The preferred software to be used is PowerBI©. Data analytics tools aid in data visualisation and will facilitate the digestion of information from all members regarding their data knowledge and experience. The interviews will be transcribed, and the main quotes will be shared with consortium members.

## 2.4 STATISTICS

The statistics which will be gathered will be shared among the consortium members and discussed in future meetings. The interviews shall be transcribed, and main quotes shall be shared to understand better what needs improvement. The statistics and data extraction shall be shared online with the consortium members and monitored and updated as the software develops. The BI information shall be hosted online and given to consortium members. The transcribed interviews and the native audio files will be stored in a shared location as this will be agreed upon by the consortium members.

## 2.5 HANDLING AND RECORD-KEEPING OF GATHERED DATA

The data gathered from users' participation shall be stored initially in the native platforms where the data input has been created. As soon as the participants inform the consortium of submitting their information, the data will be extracted by an appointed consortium member to at least a common file format that third-party applications can manage. Additionally, if any other file formats are needed, then the extraction process shall follow the needs of the evaluation plan and usability study process. The appointed member shall upload the files to PrismArch's Google Drive in a designated location. This will give the ability to the consortium members to access the data and conduct the analytics required. The data stored online in the relevant survey or poll providers shall be investigated to be deleted appropriately when the consortium agrees.

If the data needs revisioning, it is encouraged to use a similar pattern to ISO19650. As described in BS EN ISO 19650-2:2018: Preliminary revisions of information containers should be two integers, prefixed with the letter 'P' – for example, P01. Preliminary revisions of information containers in a 'work in progress' state should also have a two-integer suffix to identify the version of the preliminary revision – for example, P02.05. Using this methodology, it is important to keep track of 'work in progress' versions of information containers shared by a task team with the rest of their delivery team. The initial revision of information containers should be P01.01. Contractual revisions of information containers should be two integers, prefixed with 'C' – for example, C01.

### 2.5.1 Example of information submission and revision process

An individual agrees verbally to participate in the evaluation process of PrismArch. The participant receives the consent form and information sheet. After reading, approving and signing, the participant sends the information back to the consortium. The consortium arranges all necessary actions to start the evaluation process. This includes the information, training and access rights needed to access and evaluate PrismArch based on the usage scenarios. The participant enters PrismArch, tests the tools and exits. Afterwards, communication between the participant and the consortium is held, and the participant starts the evaluation process. The participant is asked to fill in a survey. The participant enters the relevant platform and submits the information. A confirmation email is submitted to the consortium to inform the completion of the evaluation plan. The appointed consortium member then accesses the platform and extracts the survey information. The consortium member gives a suffix of P.01.01 to the file name to distinguish this as a WIP document and uploads it on PrismArch designated Google Drive location.

Option A: If there isn't anything else needed from the participant and the document can be finalised the consortium member renames the suffix to P01.

Option B: If there is an error within the submission, any parts missed or information misconducted, the appointed consortium member informs the participant, and the information needs to be submitted again. The process repeats until the consortium member extracts the data from the platform. Then, the revision which will be added as a suffix will be P01.02. If no further actions are needed, the extracted information can be renamed to P01 and uploaded.

The process can be repeated until the extracted information is finalised. All WIP revisions beforehand shall be uploaded and stored apart from the final document.

The designated Google Drive folder shall contain subfolders with the participants' credentials and shall include the extracted information and interview files (audio and transcript format)

# 3. QUALITY CONTROL AND QUALITY ASSURANCE

The planned and systematic actions that are established to ensure that the trial is performed and the data are generated, documented (recorded), and reported include:

- Training the personnel involved in the evaluation process. Each one should be familiar with all details of the trial protocols.
- Using the distinguished tools to be followed in the process of recording the data
- Nominating a consortium member who will monitor the organization of the collected data. Such person will be nominated in each phase of the evaluation plan and usability study
- Verifying (with at least one more consortium member) the adherence of all recruited subjects to the inclusion-exclusion criteria.
- Weekly backups of data sets
- Repeat of statistical analysis

# 4. ETHICS

Evaluation Plan protocols are conducted in accordance with ISO 9001 and ISO 9241. A policy of strict compliance with the evaluation plan and usability study protocol will be adopted. The researchers will place particular attention to any ethical issues that will arise and will address them in a professional way following very closely established EU regulations and corresponding national laws about user privacy, confidentiality and consent. Briefly, the adopted ethical practises include: a) Protecting the rights of the participants, b) Protecting the safety of the participants, c) Protection privacy of the participants and d) employing non-discriminative policies.

The participants will be informed that this project is an experiment and their participation will be a valuable contribution to scientific research without any particular long term personal benefit. They will acknowledge that they agree with the temporary use of the PrismArch platform which will stop at the end of the trial no matter how useful it might be to them.

# 5. USAGE CASE SCENARIOS

PrismArch consortium has developed usage scenarios that will be the basis of testing PrismArch. Within these scenarios, a number of interactions will be investigated. The main purpose of the usage case scenarios is to demonstrate the encompassing the overall AEC ecosystem that is PrismArch's central innovation, as well as its ability to hold all related information persistently to arrive at a Golden Thread record. To fulfil its purpose, this Golden Thread record must be fully searchable and all data retrievable using the PrismArch 3D graph and UI interface. It also requires that all participating authors are manifested and are able to retain a liable and true copy of their work. The usage case scenarios will test the tools as these have been developed until the day this deliverable is submitted. As PrismArch develops, more evaluation plans will be added. The tools to be tested within the usage case scenarios are the following (cf. D5.2):

- Speckle authorisation system
- Avatars (ReadyPlayer.me integration)
- Rendering modes
- Design tools (i.e. Drawing tool, Tracing Paper tool, Whiteboard tool, Markup tool)
- VR devices (Oculus Quest 2, Oculus Rift S and HP Reverb 2 compatibility)
- Multi-playing EOS technology
- Mozilla DeepSpeech (Voice Recognition)
- AIGenTool
- Query tool

## 5.1 Architect Usage Case Scenario

The architectural Usage Case scenario focuses on platform registration and on-boarding, tool kit usage and asset sharing inside the architectural discipline, including sharing of sphered assets. It involves making internal meeting arrangements and observing a temporary increase in an asset access level (following the Sphereing Level (SL) concept introduced in D6.1 (cf. D6.1, Chapter 1). This Usage Case is designed to demonstrate how end users join a project and create, review and evaluate internal, discipline-specific project information, data, and produced design objects inside the PrismArch Project Sphere (PAPS) using the PrismArch core functionality.

**Starting a new Project**

The Concept Stage preparation of project data and information is demonstrated first, involving an Architectural Project Director (PD_A) logging into PrismArch, creating a New Project with a default quota of 5GB (that can be increased later), and uploading project information from the client (including 2D drawings [dwg, 3dm], local regulations [doc], client checklist [pdf], cultural and architectural references [jpg] and room programs [xls], as well as sketches and site photos for a potential architectural project at a specific global coordinate, Location A [longitude, latitude] taken during a site visit on [DD/MM/YYYY]). **Key Result:** The PrismArchCentralModel (PACM) now contains site photos, sketches, 2D drawings and translated local regulations. PrimArch Tags (PAT) should be automatically assigned to each uploaded project asset, and these should include timestamps with the Project Director's name, project role, content categories (e.g. photography, sketches, site plan, etc) PrismArchProjectSphere ID and SL classification.

**Internal Meeting, Project Kick-Off**

Next, the sphereing tool, meeting functionality, and multi-user interface is demonstrated through an internal kick-off meeting [SL2] organised by the ProjectDirector with new project members to review the loaded content together. In order to do this, PD_A first needs to register their team with the PrismArchProjectSphere (PAPS). The PrismArchAdministrativeSystem approves the request and sends an invitation to a Lead Designer (LD_A), a Senior Designer (SD_A), and a Lead Interior Designer (LID_A), giving the registered project members access to the PAPS. Choosing location and view: PD_A must choose the location and rotation of the meeting. They choose to see the loaded project assets from the axonometric view at a scale of 1:500 (in VR, scale is experienced as the equivalent distance from the PACM). **Key Results:** Each attendee receives a meeting sphere link and the event is automatically added to their calendars in their PUI dashboard. They each visit the arranged location and rotation by accessing the meeting link from their individual PWS [SL1] at the arranged time.

At the end of the meeting, PD_A assigns tasks for each project member. LD_A is responsible for site investigation and developing a site model. SD_A is responsible for assisting LD_A and adds volumetric studies on top of the site model. LID_A is responsible for material and interior mood boards. **Key Results:** A record of the meeting, assigned tasks, and digital assets reviewed persist as part of the searchable Golden Thread. Subsequently, the individual team members are able to create personal SL1 instances of the sphered SL2 data shared by PD_A.

**Individual Task**

LD_A is assigned a site investigation task by PD_A. In their Personal Work Sphere (PSW), SL1, they create an instance of the SL2 rated PrismArch assets shared by PD_A that are now loaded as part of the PrismArchCentralModel (PACM). LD_A intends to mark the terrain on the 2D drawing to understand the landscape and ground features and constraints of the site. The results will be used as the base of the volumetric study and will be shared within D_A.

In the PUI inside their Personal Work Sphere (PWS), LD_A has the Mindesk API and PrismArch default tools installed (multi presence on-boarding system, tagging, query, dashboard, admin, contact, multi selection, commenting and markup, punch list, spatial orientation and design support and evaluation tools). Using a mix of proprietary and PrimArch default tools, LD_A needs to mark up the maximum allowed height, site offset, local regulation, vegetation/ trees and historical and any protected conditions on top of the 2D drawing loaded at the PACM. They do this by placing lines and points inside the environment. **Key Results:** With every new object created in the PWS by LD_A, a new tag is assigned with timestamps with their name, project role, content categories, PAPS ID and SL classification. They can also select multiple points by creating a compound of the site data, and they can name the sphere compound (e.g. 'contour heights').

**<u>Internal Review Meeting 1:1</u>**

To report the work progress to PD_A, LD_A can filter their last 5 hours worth of work using the QUI and create a sphere compound with a unique name (e.g. 'PD_A_SiteModel_Review_v1'). When they received the project assets from PD_A, the initial view location and rotation for the compound sphere was set to 'axonometric perspective' and presented at 1:500 scale. LD_A can decide to keep the sphere locations as-is for their meeting with PD_A. They arrange a new meeting sphere and enter the meeting title, purpose, time, location, scale, meeting sphere size and attendees (2 people; only PD_A is invited for this internal review). **Key Results:** LD_A and PD_A meet in the arranged PAMS at the arranged meeting sphere location and rotation to review the content of 'PD_A_SiteModel_Review_v1' in the context of the PACM. LD_A is the meeting organiser and host. The meeting assets loaded at the PACM include an instance version of assets prepared by LD_A with special visibility permission for PD_A.

**<u>Using the Whiteboard and Tracing Functionality</u>**

During the meeting, LD_A loads a 2D drawing on the horizontal plane of the PACM, then navigates the meeting sphere to 1:200 world coordinate and sets the view angle to 'TOP'. The meeting host controls the model navigation, so PD_A also navigates accordingly together with the meeting sphere. **Key Results:** Both PD_A and LD_A are able to see the same objects from the same 'TOP' view. LD_A can change the opacity of the 2D drawing and use it as a canvas for tracing. Both PD_A and LD_A should have access to their own tool kits in their PUI and be able to make sketches together.

**<u>Task Completion</u>**

The next day, PD_A and LD_A meet at the same 1:500 axonometric location and PD_A approves the design objects relating to the site. **Key Results:** The assets are upgraded to SL2 with the approval of the PrismArchAdministrativeSystem (PAAS); LD_A's task is resolved and disappears from their PUI dashboard. The approved design objects are now accessible for all project members to load inside their PersonalWorkSpheres at the PACM to instance the content.

**<u>Introducing Additional Team Members</u>**

Sometime later (e.g two weeks), PD_A invites a new starter (Architectural Assistant/AA) to the Project and the PAAS grants the AA's access to the PrismArchProject Sphere (PAPS). SeniorDesigner_A (SD_A) is then responsible for briefing this new team member. **Key Results:** The meeting sphere location and rotation at the axonometric perspective established for the kick-off meeting is still valid, and a spatial hyperlink is available in SD_A's PUI dashboard. SD_A visits the location and rotation and organises a meeting to brief the new starter. With the QUI, SD_A filters five latest versions of the massing studies developed since the project started. SD_A then creates a sphere compound of the filtered metadata nodes and includes their own avatar, assigning a tag (e.g. 'Welcome_Meeting'). A timestamp is assigned to each massing option on the creation of the sphere compound. SD_A then sends the meeting invite to all D_A project members.

**<u>Presenting Multiple Options of Design Studies</u>**

During the meeting, the five pre-loaded massing options are presented.

**Key Results:** The pre-loaded design objects appear in a time-collapsed manner following the asset production timeline. SD_A uses the QUI 'timeline' functionality to show and hide different design options - swiping the disk towards left to show version 1, right to show later versions.

**<u>Querying Asset Metadata</u>**

The Lead Interior Designer (LID_A) prepares a selection of chairs for the living room and can access all relevant product information inside PrismArch. **Key Results:** The LID_A uses the QUI to pull up the manufacturer info and other associated metadata for each furniture item, and e-mail this information to the PD_A to review and forward to the client.

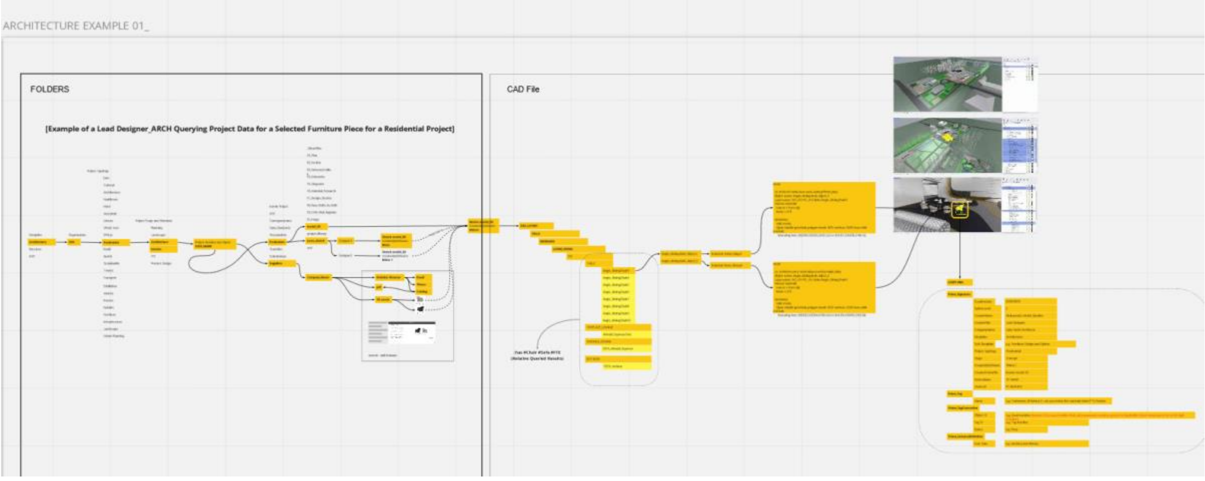

ARCHITECTURE EXAMPLE 01_
FOLDERS
[Example of a Lead Designer_ARCH Querying Project Data for a Selected Furniture Piece for a Residential Project]
CAD File

*Figure 6.3.1: Architectural query flowchart within Prismarch*

## 5.2 Structural Engineer Usage Case Scenario

The structural engineering Usage Case scenario are modelled on typical occurrences in the structural engineering industry: situations where an engineer must work individually on design exercises, as well as situations where they must collaborate and interact with colleagues from within their company as well as designers in other disciplines - to present, query, approve and merge design solutions.

The diagram below outlines the principle operations covered during this Scenario:

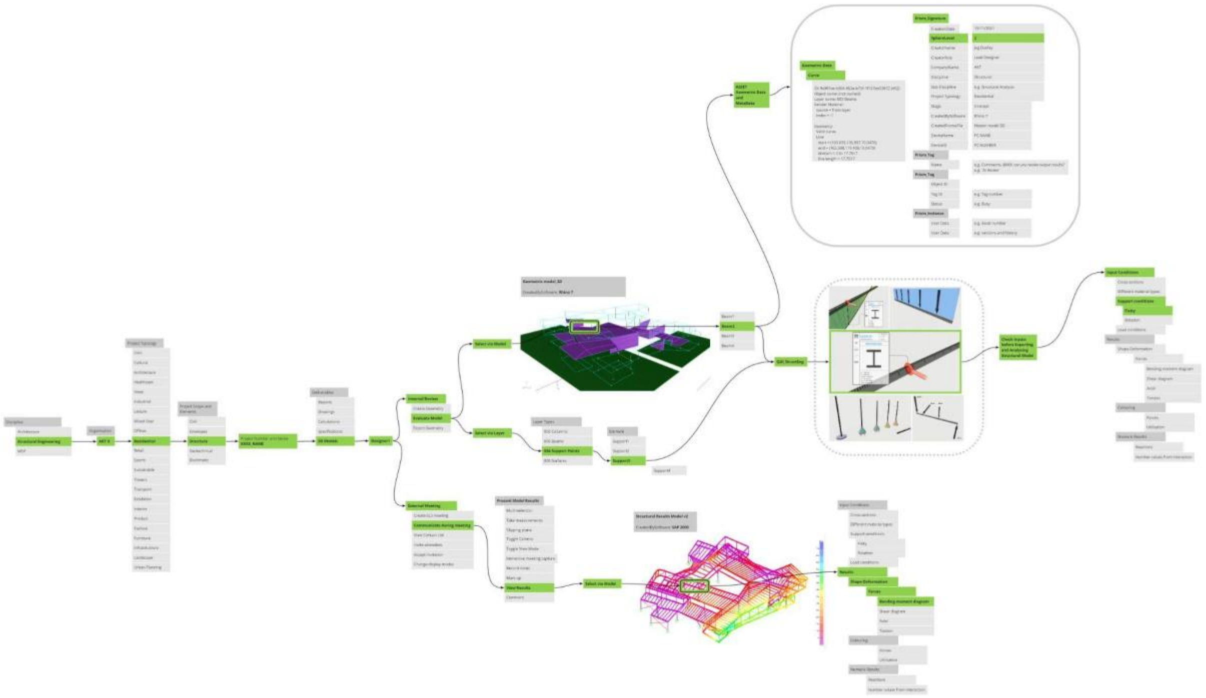


*Figure 6.3.1: Query flowchart for creating, and then evaluating, a structural engineering analysis model Usage Scenario within PrismArch*

### First Phase

The First Phase of this scenario begins with a series of activities that are necessary for all PrismArch users, regardless of their discipline. The participant will create a user account (PA_PASS), and join the PrismArch system. They will be invited into a *Test Project* set up specifically for this User Scenario, and introduced to the interface.

At this point the Scenario for structural engineers diverges from the other disciplines. The participant will be asked to take on the role of SE_S, and asked to set up an internal SL2 meeting between themselves and LD_S to review existing information (PADOs and PATs) provided by the architectural partner on the project. After reviewing the existing information, they will be assigned a basic task by LD_S.

They must make a new SL2 PAA - a copy of the existing architectural model - that is the framework on which they design an abstracted structural engineering global FEA model. This FEA model is exported to a structural engineering package, and the participant applies typical engineering properties, such as cross sections, materials, supports, loads and load cases. Using the structural engineering connector developed in PrismArch, they then import this model into the PASD as a PAA, and assign various tags and a PAST to this new entity. As a last step in this phase, the participant (still acting as SE_S) creates a PAMS to present the new model and achieve sign-off from LD_S.

During the First Phase, the participant will need to utilise the following PrismArch tools:

| **Tool** | **Potential uses within scenario** |
|---|---|
| *PrismArch Modelling Tools **** | Creating a basic geometric 'scaffold' for the FEA model to be built. E.g. using the line, point and mesh creation tools within the interface to create basic 'Stick' or 'Mesh' structural model geometries. |
| *Tagging Tool **** | Assigning groupings, material suggestions, layers, and so on to the newly-created structural model. |
| *Clipping Plane Tool* | Clipping or hiding portions of the model to select internal elements for visualisation and querying. |
| *Toggle Camera/ Toggle View Mode* | Switching between different visualisation models to ensure the structural results are highly legible within the surrounding CAD model. |
| *Commenting & Mark-Up Tool* | Highlighting critical elements within the structural model - e.g. elements with highest forces/ highest utilisation -to PD_A and PD_M. |

**** denotes tasks critical features for this Phase.*

During this meeting various PrismArch tools (whiteboard, clipping plane tool, and so on) are utilised to enhance the evaluation process.

### Second Phase

At this point we move to the Second Phase, with the participant now acting as the PD_S. In this phase they copy the model to SL3, in order to share it with other disciplines in the project. They must produce images and views of the critical design features, and create a new PAMS to present the model to PD_A and PD_MEP.

During the Second Phase, the participant will need to utilise the following PrismArch tools:

| **Tool** | **Potential uses within scenario** |
|---|---|
| *Clipping Plane Tool* | Clipping or hiding portions of the model to select internal elements for visualisation and querying. |
| *Toggle Camera/ Toggle View Mode* | Switching between different visualisation models to ensure the structural results are highlighly legible within the surrounding CAD model. |
| *Multi-Selection Tool **** | Evaluating groups of elements for similar diverse structural behaviour. |
| *Commenting & Mark-Up Tool **** | Highlighting critical elements within the structural model - e.g. elements with highest forces/ highest utilisation -to |

| | |
|---|---|
| | PD_A and PD_M. |
| *Whiteboard Tool **** | Sketching design alternatives with the PD_A and PD_MEP in response to structural analysis results. |
| | **** denotes tasks critical features for this Phase.* |

This presentation and discussion is more formal than the earlier internal design reviews, so would also likely utilise the PVM.

### 5.3 MEP Usage Case Scenario

The MEP usage case scenarios are driven by issues found occasionally in a construction project. The MEP usage case scenarios are using several of the PrismArch tools to identify, communicate and resolve the issues. The first scenario involves amendments in a facade and communications between the Project Director (PD_MEP), the Lead Designer (LD_MEP) and the Senior Engineers (SE_M, SE_E, SE_P, the suffix is as per discipline). The Architect's Project Director (PD_A) arranges a meeting (PAMS) with PD_MEP and LD_MEP. All load their models and log in to the PAMS. After going through a quick review they discuss a change needed to the facade. After the meeting, they log out and continue to work on the changes. LD_MEP works in his/ her PWS and uses the PrismArch tools to query data. (multi-presence on-boarding system, tagging, query, dashboard, admin, contact, multi selection, commenting and markup, punch list, spatial orientation and design support and evaluation tools). The LD_MEP by using proprietary software and feeding this information to PrismArch via the Speckle database system updates the information. After LD_MEP finishes the necessary work, a new PAMS is arranged to demonstrate the changes to PD_MEP. LD_MEP and PD_MEP log in the new PAMS and PD_MEP approves the design. PD_MEP upgrades the SL2 status to SL3.

The forms of the usage case scenarios are following the same pattern. In D6.1 (cf. D6.1) these are described in detail. By using Miro Board the PrismArch consortium members also demonstrated sequences to query information. These shall be included as separate exercises within the evaluation plan.

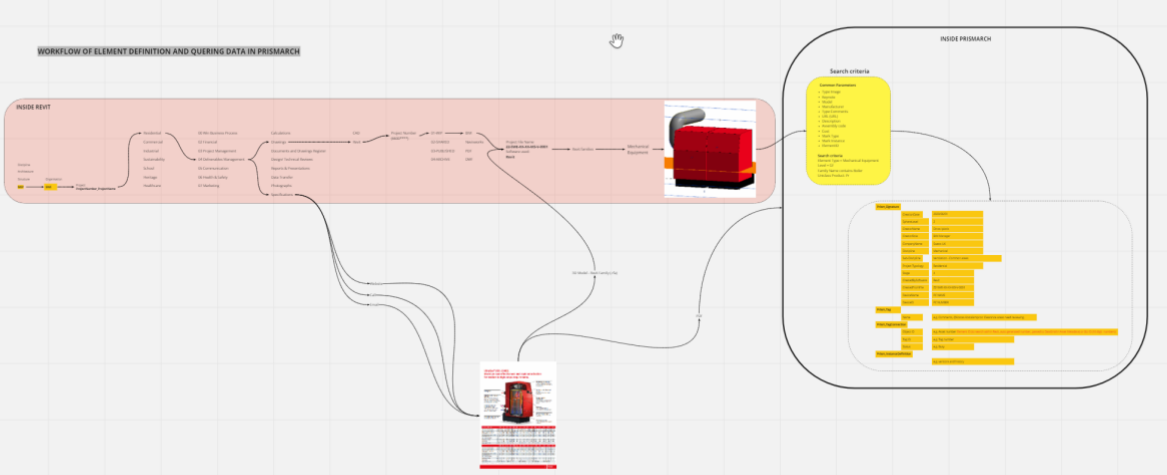


*Figure 6.3.3 Query flowchart within Prismarch*

# 6. FURTHER EVALUATION

The evaluation plan and usability study aims at testing specific tools and relevant experiences within the platform. However, since PrismArch relies on the Unreal Engine an approach similar to the computer gaming industry is inevitable for evaluation and testing purposes. A computer game before is released is tested among the following aspects:

- GUI testing
- Functionality testing
- Security testing
- Console testing
- Online Game testing
- Compliance testing
- Performance testing
- VR/MR testing

The following paragraphs describe a parallel approach which is encouraged to be considered furthermore to what has already been described in this deliverable. To ensure a holistic approach, the evaluation plan and usability study forms of data capture may include parts that cover the following aspects:

## 6.1 GUI testing

The ability to test if the user interface of the game is working as required within the virtual reality environment. In PrismArch this includes the landing space where the project to access is selected, the landing space of the actual project and the user interface while navigating around the structure. Additionally, further GUI should be tested such as the force graph and any additional GUI which may occur.

## 6.2 Functionality testing

The functionality testing is the ability to test the functionalities of PrismArch in general and apart from the tools strategically selected. This is the overall test of the platform and the evaluation that it works flawlessly.

## 6.3 Security testing

PrismArch is a cross-discipline, multi-user experience. This means that all relevant tests need to be conducted to ensure the data security of the users and the data security of the actual platform.

## 6.4 Console testing

PrismArch is aimed to be deployed to personal computers. Tests to ensure successful deployment to various models and setups of laptops or desktop terminals should occur. The tests will need to meet the minimum system requirements. However, it is encouraged to test PrismArch to low and high spec terminals. Additionally, in case in the future PrismArch becomes compatible with gaming consoles, consideration of additional tests shall be required to comply with the relevant hardware

### 6.5 Online performance testing

PrismArch needs to function smoothly and transfer the data flawlessly through the internet. Latency values and bandwidth consumption need to be tested and considered to deliver a successful product. Internet usage needs to be monitored throughout the evaluation process. The identification of minimum internet connection speed is vital for a smooth experience.

### 6.7 Performance testing

Testing and monitoring of hardware components, benchmarking while using PrismArch and if needed server monitoring is encouraged to identify the best equipment to use with the platform. This way PrismArch will be working as intended.

### 6.8 VR/MR testing

Ensure PrismArch is compatible with the latest technologies and devices. Identify any stability issues and resolve them. Make sure that a diverse selection of devices is compatible with the Platform and future expansion considerations.

Furthermore, a series of additional aspects need to be considered to fully capture the users' experience and evaluate PrismArch:

- As with every software, a series of bugs will be identified. After the developers make the necessary adjustments tests will need to be taken to identify that these have been fixed across the platform.
- PrismArch enhances collaboration. Therefore, it is inevitable to test the multiplayer experience, from avatars scale and appearance to latency and presence within the environment.
- As virtual meetings shall occur within PrismArch, the audio needs to be tested. People will speak from different microphones and input devices, therefore the sound quality needs to be evaluated.
- Graphics, textures, frame rates and interactions with materials and lighting. The users need to have a precise representation of the real world within the virtual environment. Lighting needs to be effectively simulated. Evaluation of lighting performance, materials quality and precision of values (reflection, ambience, specularity etc) needs to be tested.
- Balancing. All users need to gain the same experience within PrismArch. Thus said, the data needs to be synced as desired and the users will see what is only allowed to them. Therefore, PrismArch needs to be tested to ensure users see only the relevant information.

## 7. CONCLUSIONS

PrismAch is still under development and partial information is distributed regularly. The evaluation plan over the next months will be reviewed and improved since we do not have a fully working prototype until the date this deliverable is submitted. The evaluation plan and usability study will play a vital role in developing PrismArch to become a successful product. As the industry shifts towards more transparency and collaboration, PrismArch will be a pioneer in the AEC industry. Evaluating the platform and gaining insights from industry practitioners will improve the experience and the tools. The ongoing evaluation process will also gather data that will contribute to the existing research of how virtual reality environments contribute to innovative ways of working and shall be used in relevant research papers. The evaluation of the tools and experience shall determine the areas of further development, improvement and corrections needed. Participants will be able to submit the information in a secure way and express their feelings while using PrismArch. As this is an innovative approach of working collaboratively, the evaluation plan and usability study shall capture the impressions PrismArch leaves to the users. In case the consortium needs to adjust the methodologies, this document shall reflect the changes and adjustments needed in future revisions.

# 8. REFERENCES

Law Insider, (n.d.). Evaluation Plan definition. Retrieved from: https://www.lawinsider.com/dictionary/evaluation-plan

Krenn, H. (2017). The Step-by-Step Guide to EVALUATION. How to Become Savvy Evaluation Consumers *W.K. Kellogg Foundation*

Community toolbox, Chapter 36, Section 5: Developing an evaluation plan. Retrieved from: https://ctb.ku.edu/en/table-of-contents/evaluate/evaluation/evaluation-plan/main

Nie, P., Zhang, J., Li Jessy, J. Mooney, R., and Glicoric, M. (2021). Evaluation Methodologies for Code Learning Tasks. *The University of Texas at Austin*

Brown University, Writing an Evaluation Plan. Retrieved from: https://www.brown.edu/research/conducting-research-brown/preparing-and-submitting-proposal/proposal-development-services/writing-evaluation-plan

Paes, D., Irizarry, J., A Usability Study of an Immersive Virtual Reality Platform for Building Design Review: Considerations on Human Factors and User Interface. *Construction Research Congress 2018*

Dionne Jr, E., Mann, T. (2003). Polling and the public opinion: The good, the bad and the ugly. Retrieved from: https://www.brookings.edu/articles/polling-public-opinion-the-good-the-bad-and-the-ugly/

Trull, S. (1964) Strategies of effective interviewing. Retrieved from: https://hbr.org/1964/01/strategies-of-effective-interviewing

QuestionPro, What is a Survey – Definition, templates, methods, characteristics, and examples. Retrieved from: https://www.questionpro.com/blog/surveys/

BS EN ISO 19650-2: 2018 Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) – Information management using building information modelling

Ujagare, A., (2019). Practical Approach of Software Testing. ISBN-13: 978-9388644198. *Adhyyan Books*

IBM: What is software testing? Retrieved from: ttps://www.ibm.com/topics/software-testing

International Organization for Standardization. (2015) *Quality Management Systems - Requirements* (ISO standard no. 9001:2015)

International Organization for Standardization. (2006) Ergonomics of human-system interactions (ISO standard no. 9241:2006)

(D1.1) Deliverable 1.1 - Report on current limitations of AEC software tools, leading to user and functional requirements of PrismArch, Daria Zolotareva, Risa Tadauchi, Helmut Kinzler, Arun Selvaraj, Oussama Yousfi, Edoardo Tibuzzi and Jeg Dudley, February '21

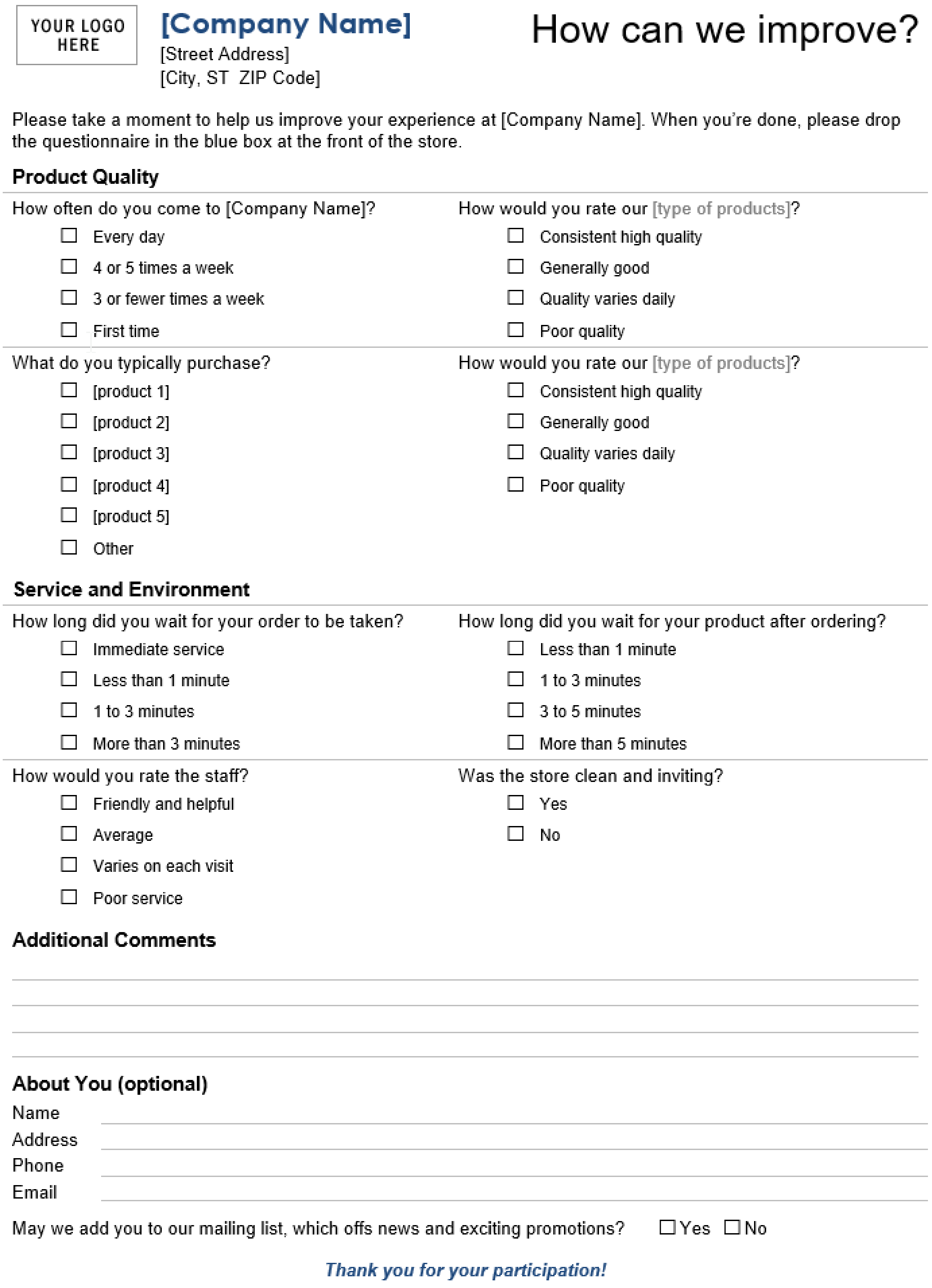
(D5.2) Deliverable 5.2 - First prototype of the VR-aided design platform, (M14)

(D6.1) Deliverable 6.1 - Define the architectural projects and usage scenarios for demonstration and evaluation, Daria Zolotareva, Risa Tadauchi, Helmut Kinzler, Arun Selvaraj, Dinos Ipiotis, Oussama Yousfi, Dinos Ipiotis, Edoardo Tibuzzi, Jeg Dudley, July '21

[D6.3] Deliverable 6.3 - Report on testing and evaluating the PrismArch platform, (M15)

# 9. APPENDICES

## 9.1 Appendix A - Example of survey template



YOUR LOGO HERE

**[Company Name]**
[Street Address]
[City, ST ZIP Code]

How can we improve?

Please take a moment to help us improve your experience at [Company Name]. When you're done, please drop the questionnaire in the blue box at the front of the store.

**Product Quality**

How often do you come to [Company Name]?

□ Every day
□ 4 or 5 times a week
□ 3 or fewer times a week
□ First time

How would you rate our [type of products]?

□ Consistent high quality
□ Generally good
□ Quality varies daily
□ Poor quality

What do you typically purchase?

□ [product 1]
□ [product 2]
□ [product 3]
□ [product 4]
□ [product 5]
□ Other

How would you rate our [type of products]?

□ Consistent high quality
□ Generally good
□ Quality varies daily
□ Poor quality

**Service and Environment**

How long did you wait for your order to be taken?

□ Immediate service
□ Less than 1 minute
□ 1 to 3 minutes
□ More than 3 minutes

How long did you wait for your product after ordering?

□ Less than 1 minute
□ 1 to 3 minutes
□ 3 to 5 minutes
□ More than 5 minutes

How would you rate the staff?

□ Friendly and helpful
□ Average
□ Varies on each visit
□ Poor service

Was the store clean and inviting?

□ Yes
□ No

**Additional Comments**

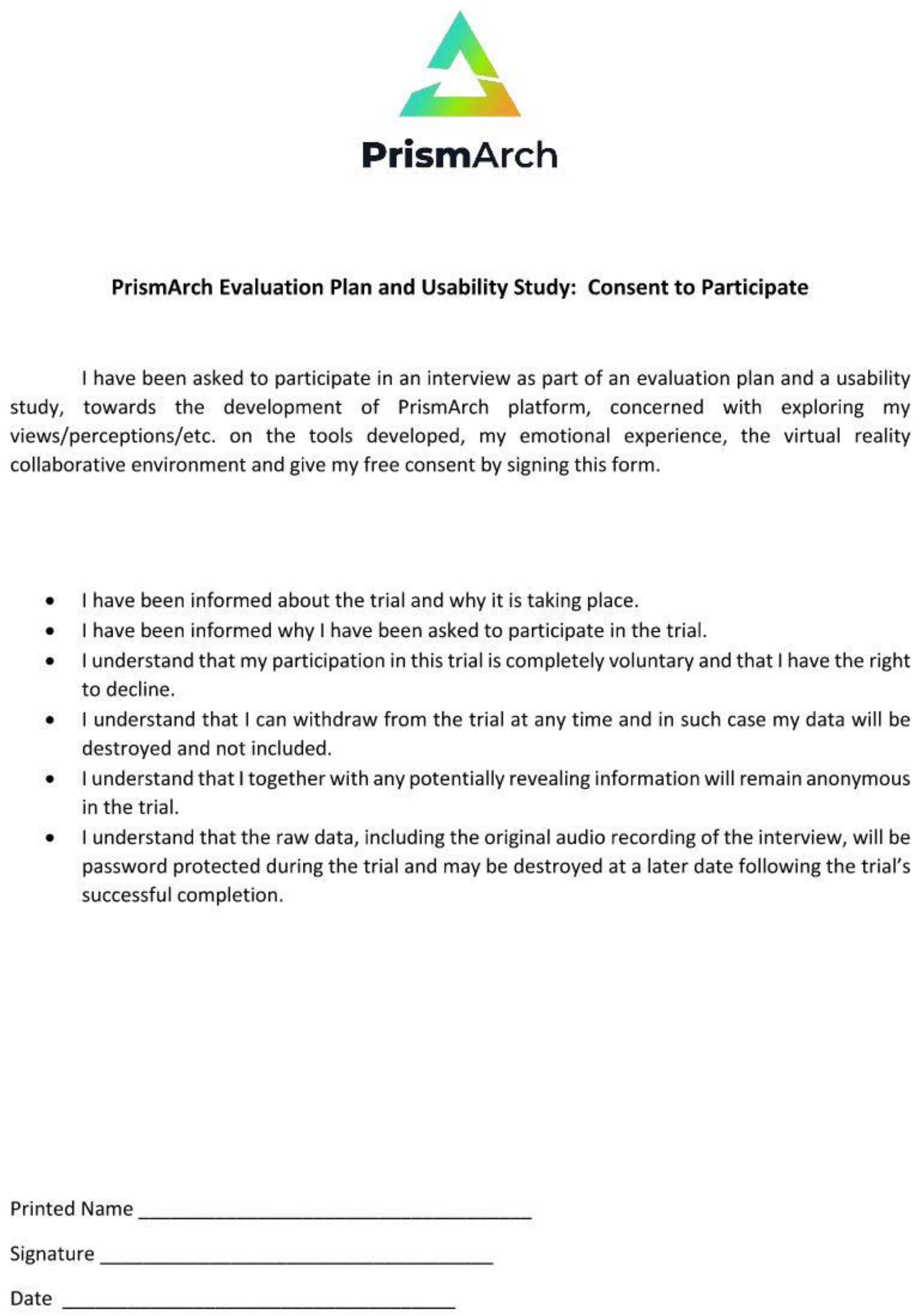
**About You (optional)**

Name
Address
Phone
Email

May we add you to our mailing list, which offs news and exciting promotions? □ Yes □ No

*Thank you for your participation!*

## 9.2 Appendix B - Consent form template



PrismArch

**PrismArch Evaluation Plan and Usability Study: Consent to Participate**

I have been asked to participate in an interview as part of an evaluation plan and a usability study, towards the development of PrismArch platform, concerned with exploring my views/perceptions/etc. on the tools developed, my emotional experience, the virtual reality collaborative environment and give my free consent by signing this form.

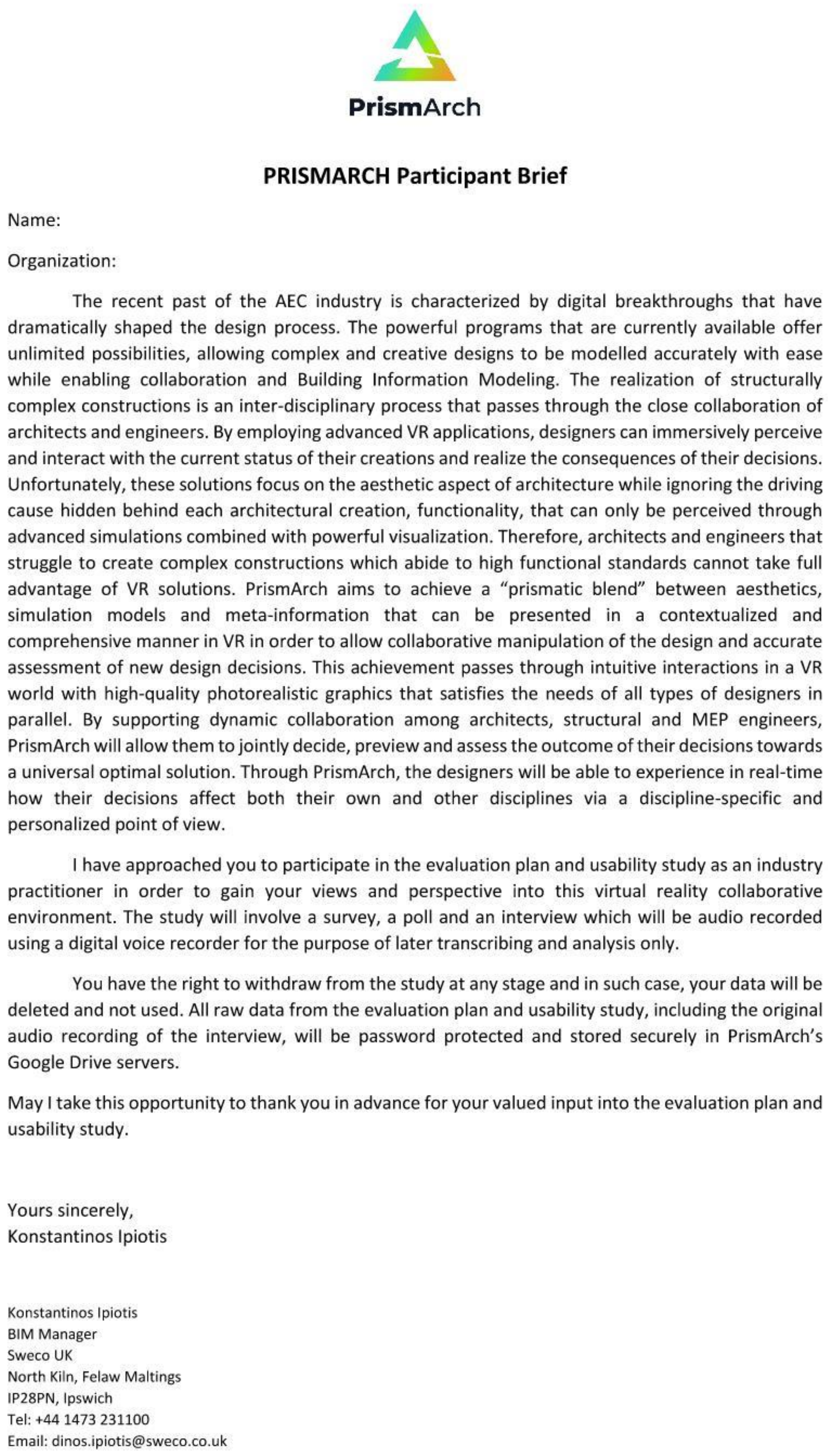
- I have been informed about the trial and why it is taking place.
- I have been informed why I have been asked to participate in the trial.
- I understand that my participation in this trial is completely voluntary and that I have the right to decline.
- I understand that I can withdraw from the trial at any time and in such case my data will be destroyed and not included.
- I understand that I together with any potentially revealing information will remain anonymous in the trial.
- I understand that the raw data, including the original audio recording of the interview, will be password protected during the trial and may be destroyed at a later date following the trial's successful completion.

Printed Name ______________________________________

Signature ______________________________________

Date ______________________________________

## 9.3 Appendix C - Information sheet template



PrismArch

**PRISMARCH Participant Brief**

Name:

Organization:

The recent past of the AEC industry is characterized by digital breakthroughs that have dramatically shaped the design process. The powerful programs that are currently available offer unlimited possibilities, allowing complex and creative designs to be modelled accurately with ease while enabling collaboration and Building Information Modeling. The realization of structurally complex constructions is an inter-disciplinary process that passes through the close collaboration of architects and engineers. By employing advanced VR applications, designers can immersively perceive and interact with the current status of their creations and realize the consequences of their decisions. Unfortunately, these solutions focus on the aesthetic aspect of architecture while ignoring the driving cause hidden behind each architectural creation, functionality, that can only be perceived through advanced simulations combined with powerful visualization. Therefore, architects and engineers that struggle to create complex constructions which abide to high functional standards cannot take full advantage of VR solutions. PrismArch aims to achieve a "prismatic blend" between aesthetics, simulation models and meta-information that can be presented in a contextualized and comprehensive manner in VR in order to allow collaborative manipulation of the design and accurate assessment of new design decisions. This achievement passes through intuitive interactions in a VR world with high-quality photorealistic graphics that satisfies the needs of all types of designers in parallel. By supporting dynamic collaboration among architects, structural and MEP engineers, PrismArch will allow them to jointly decide, preview and assess the outcome of their decisions towards a universal optimal solution. Through PrismArch, the designers will be able to experience in real-time how their decisions affect both their own and other disciplines via a discipline-specific and personalized point of view.

I have approached you to participate in the evaluation plan and usability study as an industry practitioner in order to gain your views and perspective into this virtual reality collaborative environment. The study will involve a survey, a poll and an interview which will be audio recorded using a digital voice recorder for the purpose of later transcribing and analysis only.

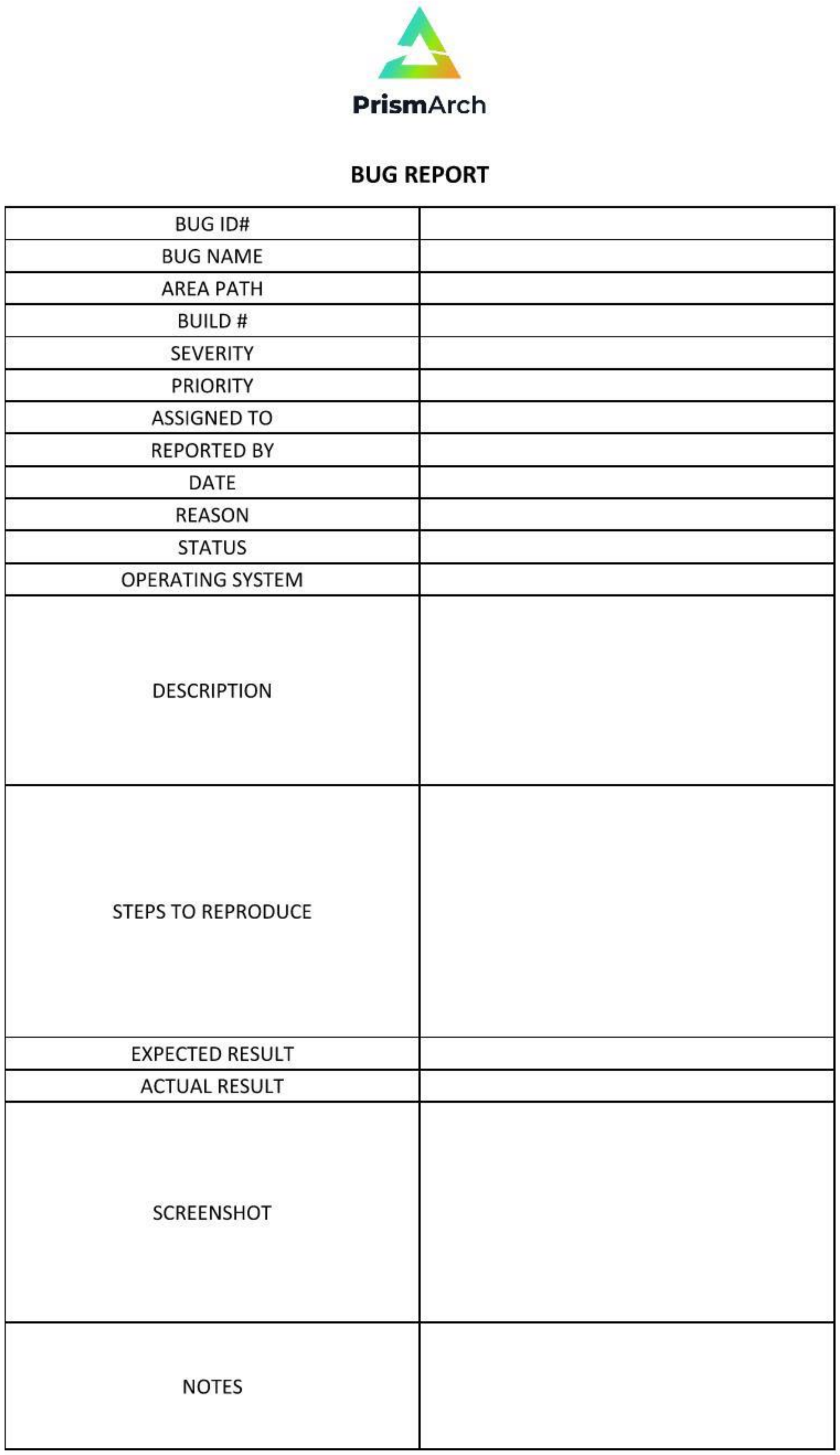
You have the right to withdraw from the study at any stage and in such case, your data will be deleted and not used. All raw data from the evaluation plan and usability study, including the original audio recording of the interview, will be password protected and stored securely in PrismArch's Google Drive servers.

May I take this opportunity to thank you in advance for your valued input into the evaluation plan and usability study.

Yours sincerely,
Konstantinos Ipiotis

Konstantinos Ipiotis
BIM Manager
Sweco UK
North Kiln, Felaw Maltings
IP28PN, Ipswich
Tel: +44 1473 231100
Email: dinos.ipiotis@sweco.co.uk

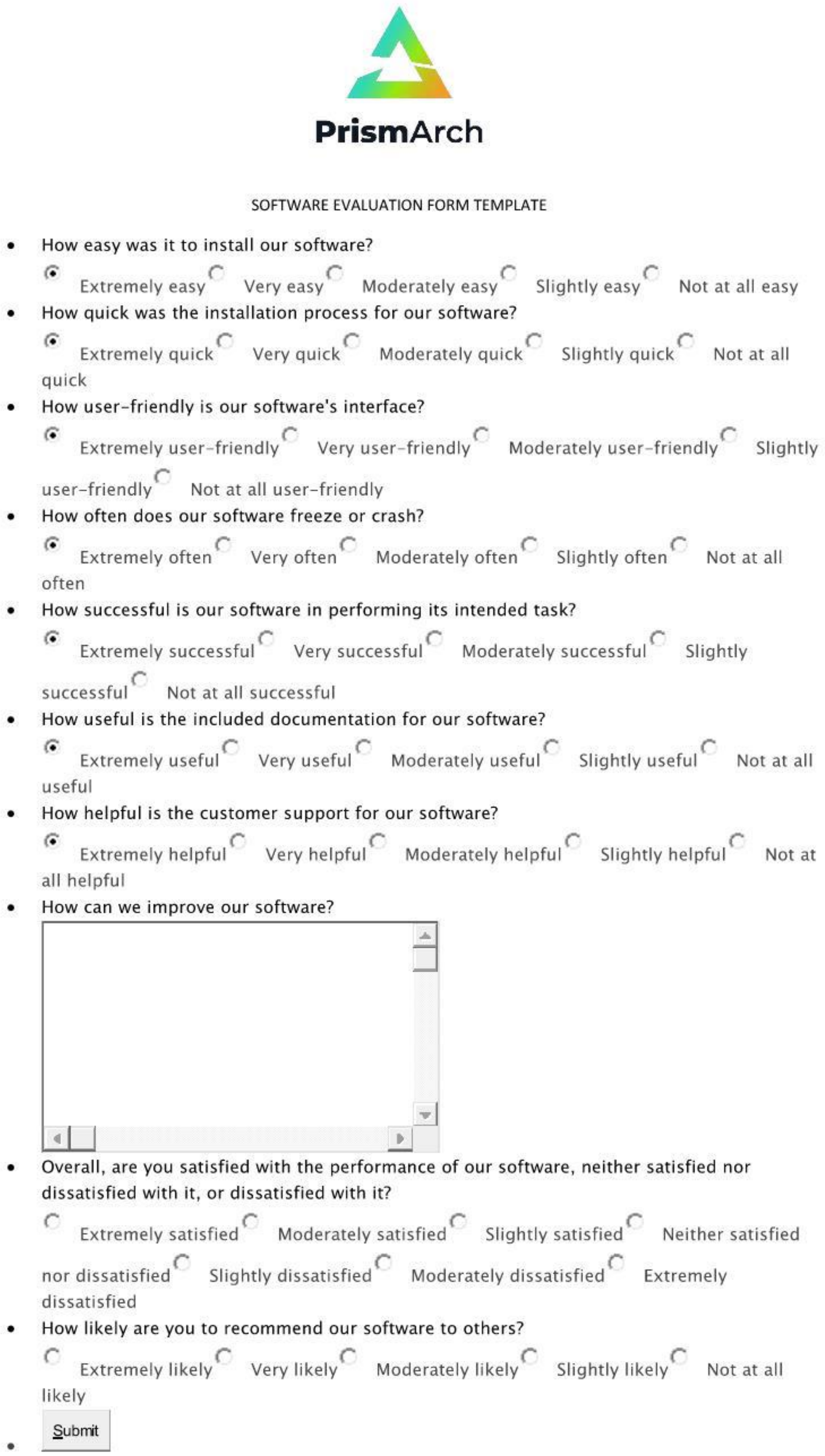
## 9.4 Appendix D - Bug report form



**BUG REPORT**

| | |
|---|---|
| BUG ID# | |
| BUG NAME | |
| AREA PATH | |
| BUILD # | |
| SEVERITY | |
| PRIORITY | |
| ASSIGNED TO | |
| REPORTED BY | |
| DATE | |
| REASON | |
| STATUS | |
| OPERATING SYSTEM | |
| DESCRIPTION | |
| STEPS TO REPRODUCE | |
| EXPECTED RESULT | |
| ACTUAL RESULT | |
| SCREENSHOT | |
| NOTES | |

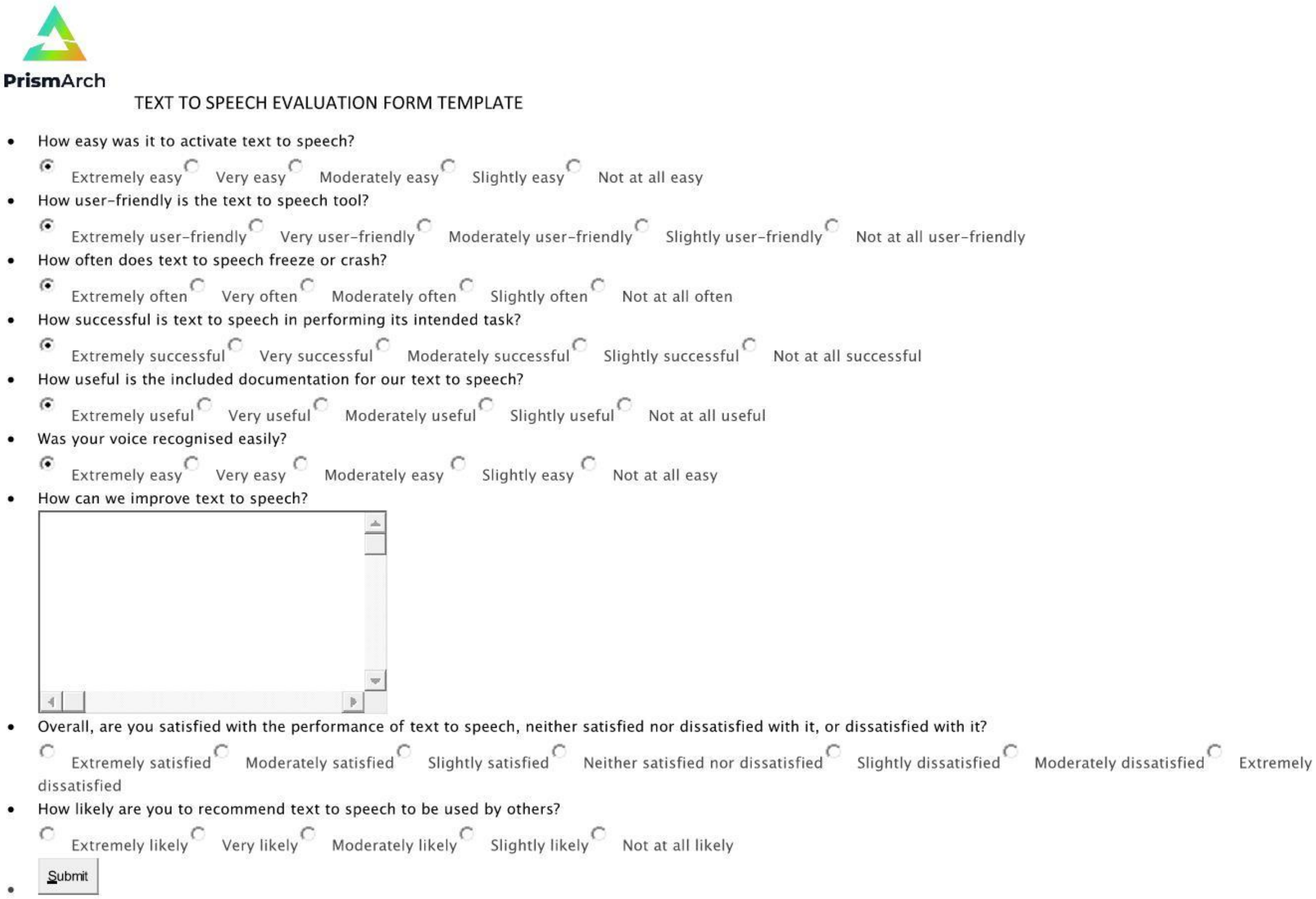
## 9.5 Appendix E - Software Evaluation Form Template



SOFTWARE EVALUATION FORM TEMPLATE

- How easy was it to install our software?
  ● Extremely easy ○ Very easy ○ Moderately easy ○ Slightly easy ○ Not at all easy
- How quick was the installation process for our software?
  ● Extremely quick ○ Very quick ○ Moderately quick ○ Slightly quick ○ Not at all quick
- How user-friendly is our software's interface?
  ● Extremely user-friendly ○ Very user-friendly ○ Moderately user-friendly ○ Slightly user-friendly ○ Not at all user-friendly
- How often does our software freeze or crash?
  ● Extremely often ○ Very often ○ Moderately often ○ Slightly often ○ Not at all often
- How successful is our software in performing its intended task?
  ● Extremely successful ○ Very successful ○ Moderately successful ○ Slightly successful ○ Not at all successful
- How useful is the included documentation for our software?
  ● Extremely useful ○ Very useful ○ Moderately useful ○ Slightly useful ○ Not at all useful
- How helpful is the customer support for our software?
  ● Extremely helpful ○ Very helpful ○ Moderately helpful ○ Slightly helpful ○ Not at all helpful
- How can we improve our software?
- Overall, are you satisfied with the performance of our software, neither satisfied nor dissatisfied with it, or dissatisfied with it?
  ○ Extremely satisfied ○ Moderately satisfied ○ Slightly satisfied ○ Neither satisfied nor dissatisfied ○ Slightly dissatisfied ○ Moderately dissatisfied ○ Extremely dissatisfied
- How likely are you to recommend our software to others?
  ○ Extremely likely ○ Very likely ○ Moderately likely ○ Slightly likely ○ Not at all likely
-

## 9.6 Appendix F - Text to speech evaluation form template



PrismArch

TEXT TO SPEECH EVALUATION FORM TEMPLATE

- How easy was it to activate text to speech?

  ● Extremely easy ○ Very easy ○ Moderately easy ○ Slightly easy ○ Not at all easy
- How user-friendly is the text to speech tool?

  ● Extremely user-friendly ○ Very user-friendly ○ Moderately user-friendly ○ Slightly user-friendly ○ Not at all user-friendly
- How often does text to speech freeze or crash?

  ● Extremely often ○ Very often ○ Moderately often ○ Slightly often ○ Not at all often
- How successful is text to speech in performing its intended task?

  ● Extremely successful ○ Very successful ○ Moderately successful ○ Slightly successful ○ Not at all successful
- How useful is the included documentation for our text to speech?

  ● Extremely useful ○ Very useful ○ Moderately useful ○ Slightly useful ○ Not at all useful
- Was your voice recognised easily?

  ● Extremely easy ○ Very easy ○ Moderately easy ○ Slightly easy ○ Not at all easy
- How can we improve text to speech?
- Overall, are you satisfied with the performance of text to speech, neither satisfied nor dissatisfied with it, or dissatisfied with it?

  ○ Extremely satisfied ○ Moderately satisfied ○ Slightly satisfied ○ Neither satisfied nor dissatisfied ○ Slightly dissatisfied ○ Moderately dissatisfied ○ Extremely dissatisfied
- How likely are you to recommend text to speech to be used by others?

  ○ Extremely likely ○ Very likely ○ Moderately likely ○ Slightly likely ○ Not at all likely

-