D3.1

ICT PLATFORM ARCHITECTURE

Approval Status

	Name and Surname	Role in the project	Partner
Author(s)	Issac GARCIA Iacopo Carli Andreas Pusch Federico Morosi	WP3 Leader Researcher Researcher Researcher	Viseo POLIMI GINP POLIMI
Reviewed by	Giandomenico Caruso	WP2 Leader	POLIMI
Approved by	Gaetano Cascini	Project Coordinator	POLIMI

History of Changes

Version	Date	Description of Changes	By
01	09.03.2017	Initial draft of the document	I. Garcia
02	21.03.2017	Add physical structure	A. Pusch
03	22.03.2017	Add process view	I. Garcia
03	27.03.2017	Add logical view of SAR module	I. Carli
04	27.03.2017	Update tracker information	F. Morosi
05	28.03.2017	Review of the document	G. Caruso
06	31.03.2017	Final review and approval	G. Cascini

Document Details

Dissemination Level	Public
Due Date	31.03.2017
Issue Date	31.03.2017
Contract Number	H2020-ICT/2015-688417
Electronic file location	www.SPARK-project.net/wp-deliverables
File name	D3.1_WP3_ICT Platform Architecture

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.

H2020_ICT_2015_SPARK_688417

TABLE OF CONTENTS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.

H2020_ICT_2015_SPARK_688417

1. EXECUTIVE SUMMARY

This deliverable describes the SPARK platform architecture. It presents the results obtained in the task 3.1 "ICT Platform architecture", during this task some decisions were made in terms of interoperability and the needed process to execute correctly a co-design session with the SPARK platform. The development of the independent modules started in WP2. Clearly, the functionalities and features of those modules guided the overall architecture, as they allow identifying possible problems in terms of interoperability that were and are addressed during the WP3. All reflections and discussions during the development and the realization of tasks in the scope of WP2 contribute to the completion of this document.

This document presents the SPARK platform architecture from several points of view: first a physical perspective is presented to show the used devices and connected computers; then a logical perspective describing what are the main operations each module and submodule is capable of performing. Following is the process view, which presents the flow of information when the platform is executing a specific operation. Then the deployment view presents the way the platform should be installed. Eventually, the document discloses the requirements in terms of preparation of a co-design session and finally the conclusions.

Even if this document presents a technical view of the SPARK platform, it has been generalized enough to be platform-independent, that is, without restriction of a specific technology, language, operating system or even the selected technique during the analysis performed in any of WP2 tasks.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.

H2020_ICT_2015_SPARK_688417

2. INTRODUCTION

The SPARK platform is composed of several modules; each module has its own lifecycle in terms of execution and development; however, for the correct execution of a co-design session, these modules should interact and share data and information. Also, a co-design session within the context of the SPARK platform is performed thanks to the usage of external software and devices. The complex nature of the platform, its requirements and its heterogeneity requires starting from the big picture of all components interacting together, i.e. the platform architecture.

The architecture of the SPARK platform is presented in several points of view or perspectives. Each perspective allows understanding well the modules, their specific operations, their behavior and their requirements:

- (Chapter 3) The physical point of view of the architecture describes the physical elements that are part of the platform or that are required for the correct execution. This for example, allows understanding where each of the modules will be executed, but also their requirements in terms of external devices.
- (Chapter 4) The logical point of view describes the internal components of each module. This allows identifying the tasks each module and submodule can perform.
- (Chapter 5) The process point of view allows identifying the transversal interaction of the modules when they interact to perform a specific operation.
- (Chapter 6) Finally, the deployment point of view describes how the platform is installed.

It is important to highlight that the specification, as well as the chosen architecture, is and should always be as generic and abstract as possible, that means that each module could be exchanged by another by using a newer technology stack, if needed in the future. This assumption should remain valid also for the chosen technique.

3. PHYSICAL ARCHITECTURE

The SPARK platform is composed of several physical components, these components vary from devices in the room where the session is running, to servers that could be executed in the cloud, in the same room or in an external datacenter. This section describes these different physical components that will participate in the execution of co-design sessions.

Computer for SAR module: The SAR module should run on a computer that allows flexibility in the use of the platform with different settings. The number of projectors that are connected to the computer varies from two (exceptionally one projector can be sufficient), to several in a

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.

H2020_ICT_2015_SPARK_688417

modular array. For this reason, the computer setup should have enough graphic output ports to connect multiple projectors with high resolution.

Personal Computer: The session leader needs to use a personal computer. This computer does not run any implemented software, the only restriction is that it should have a web browser, and it should be able to connect to the web application provided by the IS module.

Optical Tracker: This system allows tracking the position of the physical mockup relative to a previous-defined reference system in order to be used inside the SAR module for the real-time projection. It uses the infrared technology for the necessity to have invisible markers and to be easily adaptable to both types texture-based or external components.

Projectors: The computer running the SAR module will have several connected projectors. These projectors will be virtually identified by the SAR module. Then, the SAR module decides which images will be displayed at each projector device.

Interaction device: In order to perform interactions in the SAR environment two main types of devices have been identified: a tablet and an interactive table projector. The application running on these devices, is not manipulating the 3D object. Instead, it is an extension of the SAR module to let the user easily perform the functions provided by the application by exploiting a multi-touch gestures approach. The computer, where the SAR application is running, works as a server and the interactive devices as the client. Information about the fingers' position and the functions' activation are sent from the clients to the server in real time. These devices, based on Android OS (for the ongoing implementation), should be running in the same local network as the computer that is running the SAR environment.

Application server: Is a web application server that runs the IS module application. It could be run in the cloud, in a datacenter, or even locally. The only restriction is that the Computer running the SAR module can reach it by HTTP protocol. Also, if there are remote users, they need to reach this server in terms of networking.

Physical structure: Is the structure that needs to be placed in the session's room. This structure allows fixing some device and/or components, such as the trackers and projectors. The structure configuration can vary based on the capacity of the room or the kind of object to be used for example. Figure 1 shows a possible layout of a two-projector configuration arranged on the ceiling of a meeting room for the SPARK meeting room.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

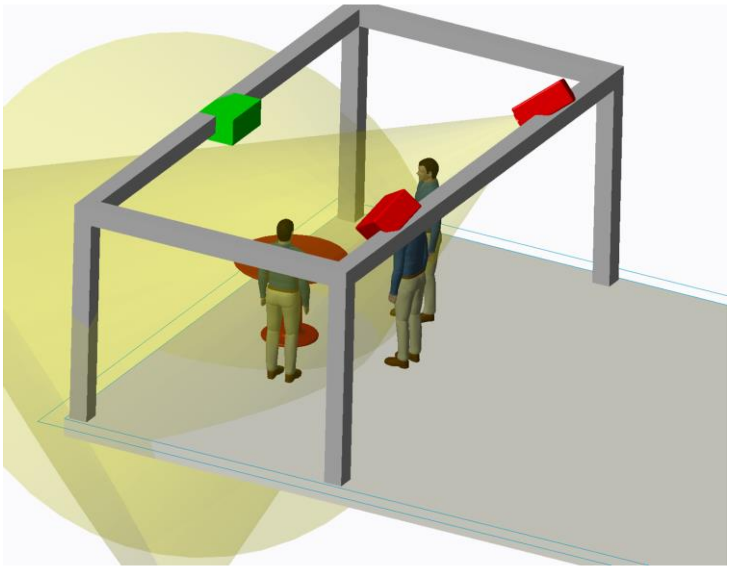

Figure 1Example of a physical structure

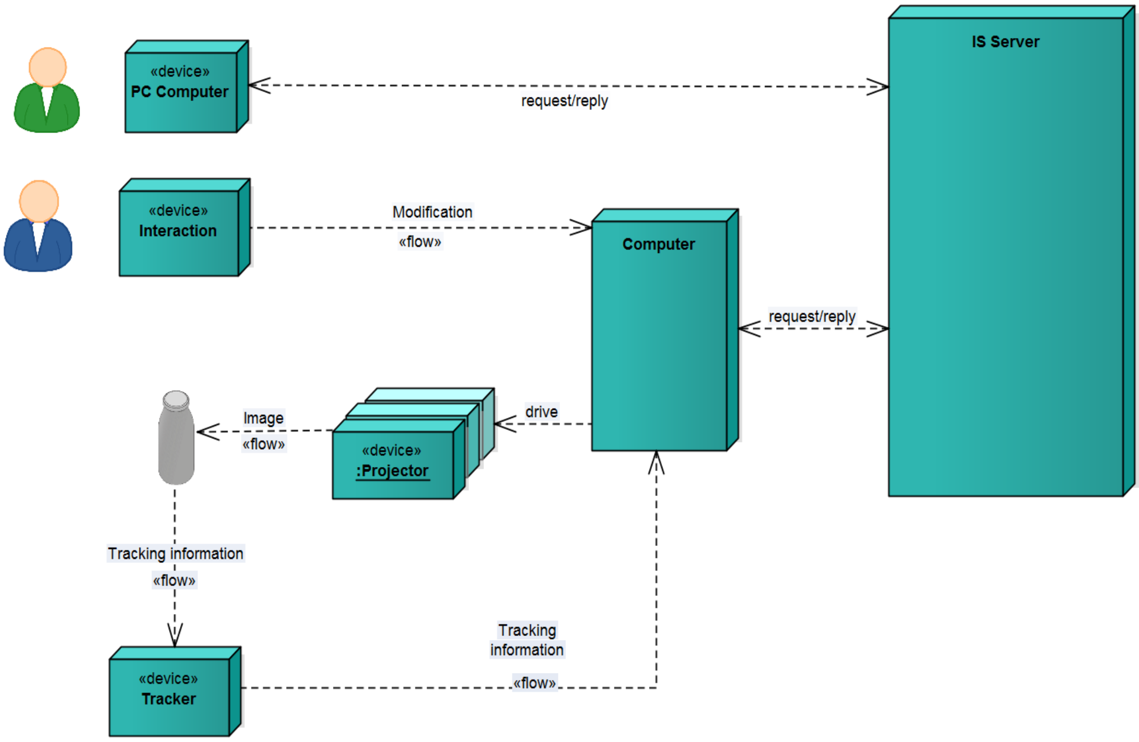

Figure 2 Physical architecture

Figure 2 shows the architecture of all the structured components that interact during a co-design session. The physical structure, where the projectors and/or the tracking devices are placed is not presented in this schema since it is only for supporting the devices. However, it is important to describe this structure, as it should be considered when a room is prepared to host co-design sessions.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

4. LOGICAL ARCHITECTURE

As described in previous deliverables (D2.1 & D2.2), the SPARK platform is mainly composed of two major modules, these modules are the IS module and the SAR module. Each module has a separate lifecycle in terms of execution, but also in terms of development. Therefore, in this section the view of each module is seen separately.

The logical architecture of the platform describes the logical components that are part of each module. This view helps understand the business logic of the components and the separation of concerns of each of them. Each component has a well know defined set of known tasks that can be performed to contribute to the correct execution of the platform.

The IS module and the SAR module have separate business logic and lifecycles, however, both modules handle a set of information related to the virtual prototype. To maintain the compatibility of the SAR module with the IS module, it was defined during the WP2 a shared data model for the "Virtual Prototype". This model, reported in Figure 3, shows the information that is exploited by the SAR module and by the IS module. The common understanding of this model allows mainly to:

- handle any 3D model;
- share the state of the virtual prototype between both systems;
- keep the history of changes of the virtual prototype for further reporting;
- tag or take snapshots of a given state for further analysis.

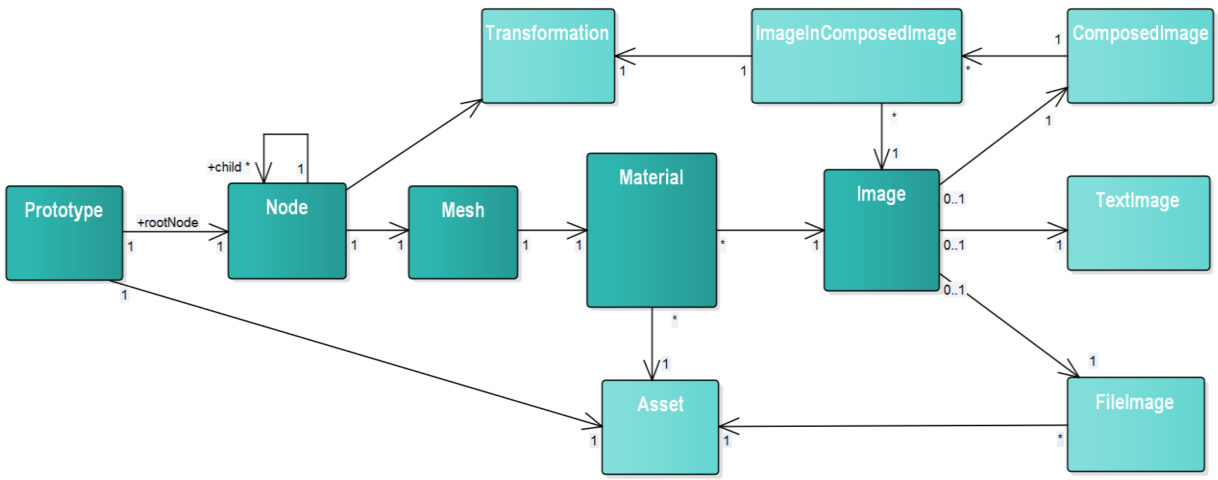

Figure 3 Virtual Prototype Model

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

The importance of the definition of this Virtual Prototype is that all the information produced is platform independent, that means that the same virtual prototype information can be interpreted by the Unity application (the SAR module) and the web browser (Front end of the IS module) using WebGL. This also means that in the future it will be possible to embed more advanced implementations using other 3D technologies.

The information handled in this virtual prototype is also normalized, so as to avoid conflicts with different coordinates system (WebGL, OpenGL, Direct3D, etc).
Figure 4 shows the normalization performed on the coordinate system to be interpreted by the SAR module and the IS module for positioning images, labels or texts onto the virtual prototype texture for positioning images labels onto the virtual prototype texture for positioning images, labels or texts onto the virtual prototype texture.

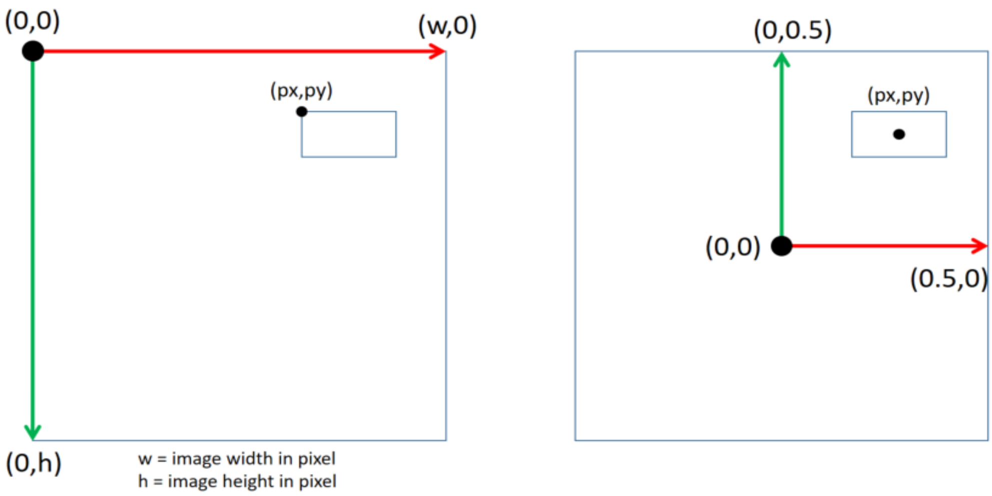

Figure 4 On the left pixel coordinate system, on the right normalized coordinate system

Once defined the common data model, and the normalization of data, we can now describe the logical representation of each module separately by using the same data structure.

4.1. SAR MODULE LOGICAL ARCHITECTURE

The SAR module is a standalone application that is only running when the co-design session is taking place. Also, it runs in the same room, since it drives the projectors and tracking devices. It is worth noticing that this module has a strong dependency in the calibration application before starting its execution (and actually, not during it).

We can say that the SAR module is divided in three major submodules, the calibration submodule, the business logic submodule, and the interaction submodule. Even if those submodules run separately, they are strongly coupled with each other, as each of them is responsible for important aspects of the SAR module implementation.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

The calibration procedure is necessary for the initial setup of the SAR platform and it is studied to be used in an automatic way with a camera-projector stereo system. In order to have this system fully calibrated, it is required to know both the intrinsic parameters (focal length and principal point) of the projector and the camera and the extrinsic parameters (rotation matrix and translation vector) which define the position and rotation of the projector relative to the camera. This is made thanks to the use of the structured light technique, based on the projection of a gray codes, for rows and columns, onto a checkerboard.
To simplify the calibration process it is assumed a pinhole model for both the camera and the projector are used. This approach is largely used in applications like machine vision, robotics and navigation systems, because it is guaranteed to have a correct balance between accuracy and computing rapidity. The pinhole model, the simplest one, is based on the absence of lens and with a single small opening in which the linear light rays pass through. Its parameters include the optical center and the focal length of the camera (intrinsic) and its location in the 3D scene (extrinsic). With this technique, it is also possible to identify the distortion of the images due to the lens as radial and tangential coefficients.
The advantages of this type of calibration are related to the accurate results obtained and the simplicity of use, even if it is required to follow a specific procedure that makes it stricter with respect to others methodologies.

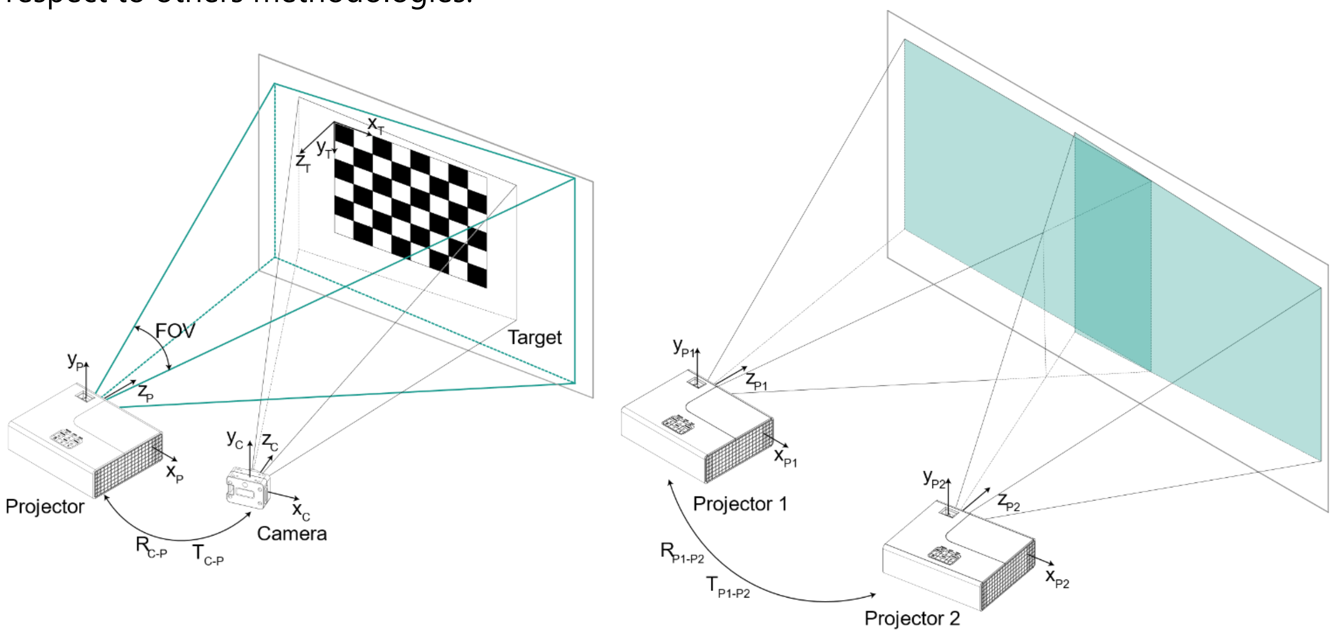

Figure 5 Calibration of a single camera-projector system: initial phase (left) multi-projector calibration (right)

In the SAR application, those fundamental parameters are necessary to correctly build the 3D setup of the platform. This is the 3D pose of each element, cameras and projectors, computed in relation with the same reference system. Other parameters are the Field Of View (FOV) and the aspect ratio of the projector (Figure 5). These are necessary to use multiple projectors (the overlap of the projected images and their correct alignment) and to define the mockup's position-orientation with respect to the projector itself (the tracking algorithm provide this information only in reference to the camera viewpoint).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

The calibration submodule provides two features: the calibration of a single camera-projector system and the calibration of multiple projectors. The first one (Figure 5, left) can be done at the initial phase of the setup once the two components are fixed on the same support; it needs to capture different sets of images with a different orientation of the target chessboard on which it is also projected a specific gray code. The second one instead (Figure 5, right) is necessary to be performed each time the positions of the projectors are changed inside the platform. In this case, it is simultaneously captured an image of the target with the cameras previously calibrated. This expedient allows reducing the time necessary for the calibration once it is adjusted the setup of the platform.

The Business logic of the SAR module is composed by the following submodules:

- The virtual scene (or simply, scene) contains information about the lighting on the virtual prototype, the position and properties of the cameras/projectors. Here there is a place holder component that consists of an empty object that represents the position of the prototype inside the virtual scene, once it is imported and loaded from the IS module.
- The prototype manager handles the state of the virtual prototype. As the application starts, it loads the 3D model into the virtual scene in a predetermined position. Then it sets the virtual prototype attributes (materials, colors…) as defined in the IS web interface. After any modification on the prototype, it sends back to the IS the list of the modifications occurred.
- The IS module manager operates the connection with the IS server when the application is launched. It uses the login and password credentials in order to start the application runtime and it sets the Image Manager and the Prototype Manager active. Once the application is quit, it disconnects from the IS server.
- The visualization manager, contains a texture modifier, a color picker and an Image manager.
    - The texture modifier component manages the functions (mesh selection, graphic elements preview, add, selection, change of position, rotation, scale, layer order) that the user is able to activate through the GUI and perform in the SAR environment during the session. These functions are performed and enabled during the design session by the user through the interaction input system.
    - The color picker component allows the color selection during the session, it sends color information to the background color of the Render Texture camera which is rendering it onto the related material assigned to the mesh component selected of the prototype.
    - The image manager handles the graphic elements list related to the current design session. It downloads them locally and loads into the virtual scene.
- The tracking manager, which receives and interprets the spatial position of the physical prototype.
- The interaction manager receives instructions from the interaction device to manipulate the mixing prototype.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

The interaction submodule is a separate application; it can even be executed on a separate device from the one running the SAR main module. This module will be developed in the scope of the SPARK project, to run on android devices.

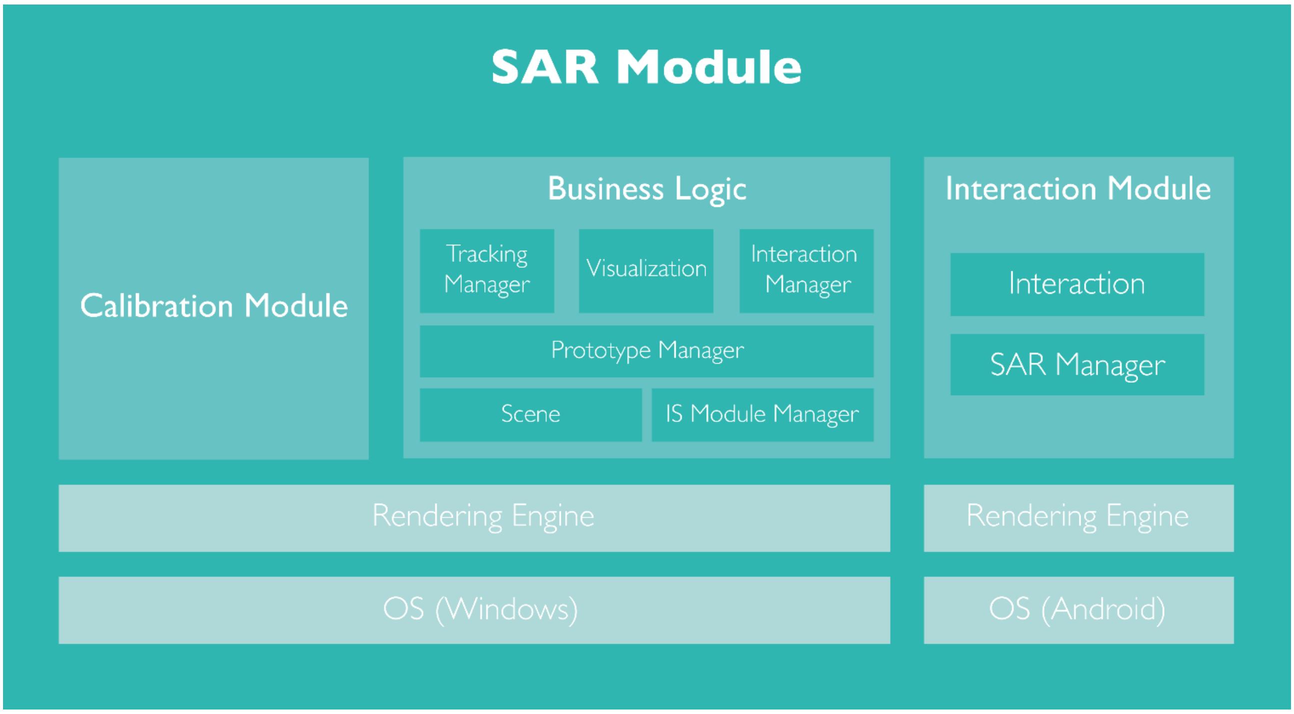

Figure 6 SAR dependencies Logical View

Figure 6 shows the logical architecture of the SAR environment needed to run properly the SAR module. The Interaction Module is an extension of the SAR module; both modules will communicate internally during a co-design session to interpret instructions for prototype manipulation.
The Calibration module on the other hand, is a standalone application that only runs once to calibrate the co-design session's room.

4.1. IS MODULE LOGICAL ARCHITECTURE

The IS module consists of two parts that make use of two different technologies, one for the backend, and another for the frontend. However, it is the back-end that provides the front-end resources that are interpreted by the browser and the JavaScript engine in it.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

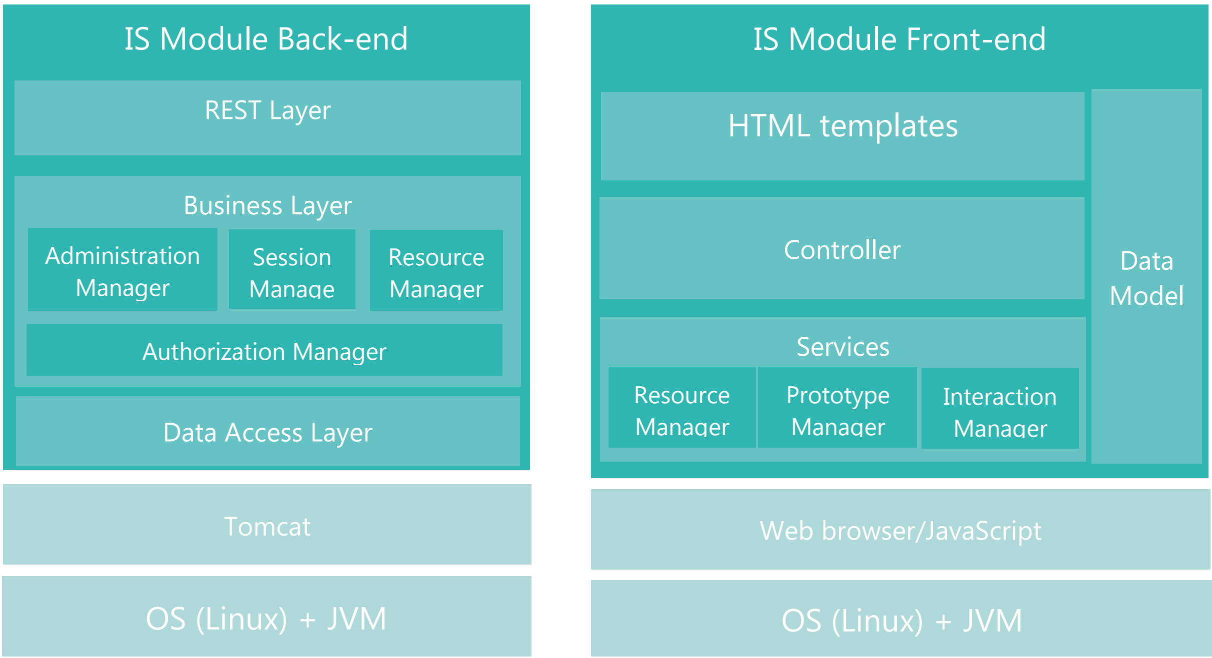

Figure 7 Structure of the IS module (Front-end and Back-end)

Figure 7 shows the structure of the IS module. The backend follows a classic n-tiers architecture, as described hereafter:

- The REST layer provides the resources used to execute the frontend. Also, is the entry point to communicate the frontend with the backend at runtime, but most importantly it provides all the methods exploited by the SAR module. This module is highly coupled with the security mechanism. So, for each received request, before analyzing its content, this layer inspects authentication to grant access or not.
- The business layer ensures the consistency of the information. This layer is basically composed of:
    - Administration manager, which allows performing all the CRUD operations of the structuring entities (users, clients and products).
    - Session manager, to handle all the operations related to co-design session, such as access to and modification of virtual prototypes and access to history of modifications.
    - Resource manager, to manage documents, notes, comments and everything related to external information that helps the preparation of a co-design session.
    - Authorization manager, to ensure the logical access of the information. The requirements defined in the Deliverable 1.3 impose a fine-grained authorization mechanism for each co-design session.
- The data access layer is highly coupled with the database containing all the information. Figure 8 shows the data model of the database it handles. This model is an updated version of the model shown in the Deliverable 2.3. This model is extended in terms of assets, documents, whiteboard and, most important, in terms of virtual prototype and modifications possibly made during the execution of the co-design session.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

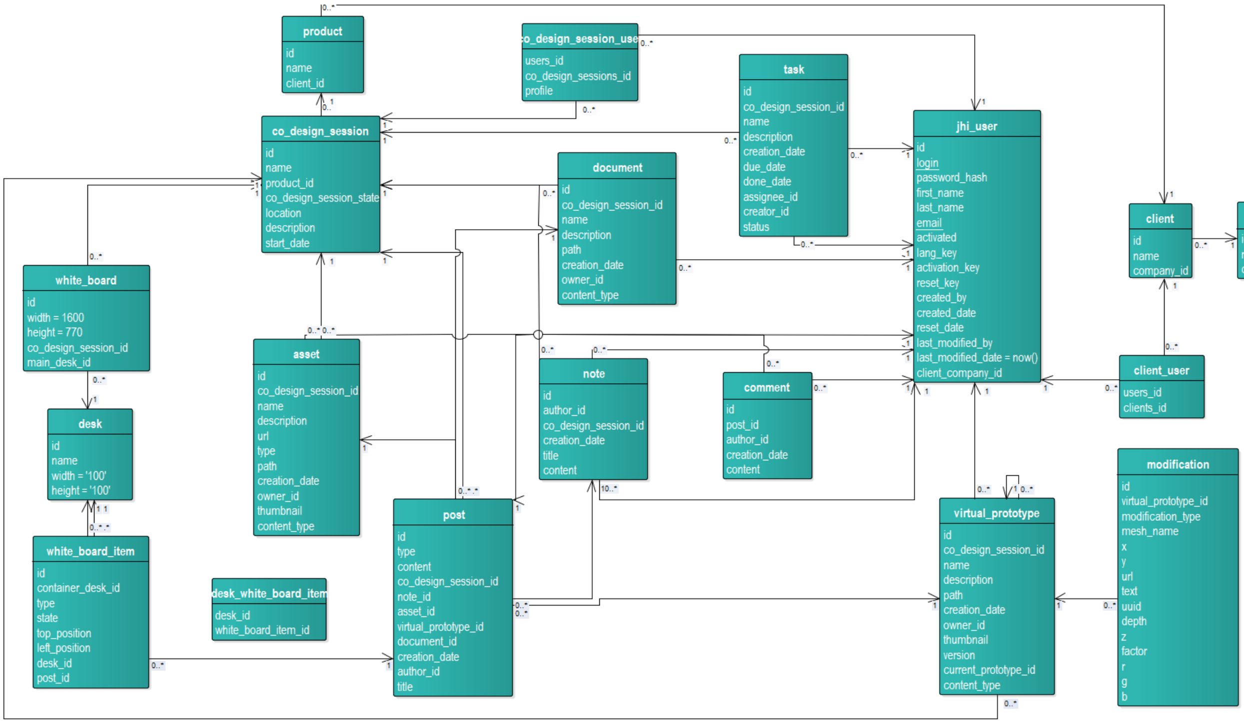

Figure 8 Database model

The frontend is far more complex than the backend. The responsibility of the backend is to maintain the coherence of the information, but there are no complex rules to consider, only access restriction and coherence based on the co-design session state. On the other hand, the frontend performs operations at the same level as the SAR module in terms of 3D manipulation and texture employment, but not in terms of tracking, calibration and multi-projection. This complexity is caused by the need to handle the Virtual Prototype Model, when preparing a co-design session and to replay it when the session is finished.

The frontend follows a Model-View-Controller architectural pattern. It allows having a complex application within the web browser with a good separation of concerns. The frontend is composed mainly of:

- The services, to manage complex operations and to serve as a link between the frontend and the backend. It consists of the followings:
  - A resource manager, to handle the upload/download of resources between the frontend and the backend,
  - A prototype manager, to manage the operations performed to the virtual prototype
  - An interaction manager, to interpret modifications made by external applications, for example the SAR module. It allows visualizing these modifications in "real-time".
- The controllers, to maintain the link between the services, the data and the view.
- HTML templates, to render the data into the web browser

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

- The data model is manipulated by the services, the controllers, and interpreted by the templates to render the contained information. This data model consists on structuring information such as client, product and session. However, most importantly, it consists of the virtual prototype defined in Figure 3.

5. PROCESS VIEW

A live co-design session is performed thanks to the collaboration of diverse applications and/or modules. To understand the collaboration of these modules, this section presents the architecture in a process point of view. The process view consists of the explanation of the interaction between each module and explains the operations for each module to accomplish a more global objective.

Deliverable 2.4 present some process view of the platform; however, they mainly show the internal process of the IS module, or a very abstract point of view of the overall platform.
In this section, the process view presents more details of the interaction and the actions that are performed by each module or component interacting during a co-design session.

The different process that are presented here are:

- Initialization of a co-design session;
- Manipulating the mixed prototype state using the SAR module (i.e., modify the rendered content);
- Manipulating the mixing prototype state using the IS module (i.e., modify the rendered content);
- Manipulating the mixing prototype (i.e. change its spatial position).

The co-design session is initialized by the session leader: it connects to the platform and starts the session. The running session contains one virtual prototype and will be exploited by the SAR module and the IS module. Therefore, the session leader is responsible also of the selection of the virtual prototype to be used during the session.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

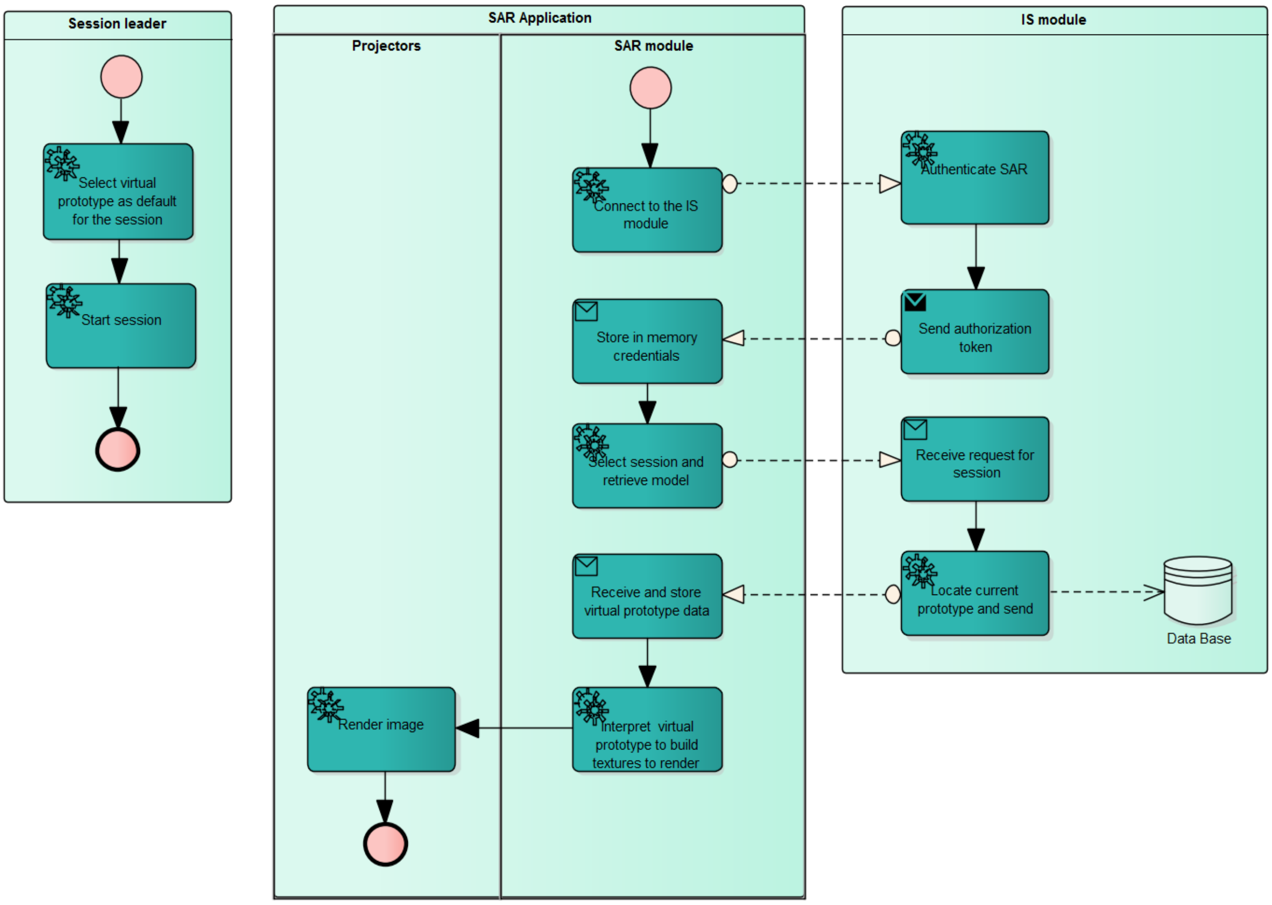

Figure 9 Initialization of a co-design session

Figure 9 shows the overall interaction to initialize a co-design session. Only when the session is started in the IS module, the SAR module will be able to connect to it and receive the virtual prototype that was previously prepared. The SAR module connects to the IS module. When the SAR module receives the virtual prototype and all the resources needed to interpret it, it builds the needed images and sends them to each of the connected projectors.
The SAR module is presented within a SAR application. The SAR application is the set of elements needed to run the spatial augmented reality. That means it is formed by the SAR module, the projectors, the tracking devices component or device used to interact with the mixed prototype.

When the session is started, and the projection is done onto the physical prototype, the participants of the session can start the manipulation of the object: the manipulation, as described in previous deliverables, goes from moving and touching the prototype, to modifying the rendered images.
Figure 10 shows the overall process that occurs when a participant modifies the rendered content of the prototype.

- When a participant performs a modification using gestures or the interacting device, the instruction is sent to the SAR module.
- The SAR interprets the performed operation and applies it to build new images.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

- The images are sent to the connected projectors, and to the IS module.
- The projectors displays the images onto the physical prototype.
- The IS module, stores the modification and notifies connected users using the web browser.
- The application running in the web browser also applies the operation and renders the new result into the browser.

This information flow allows having several connected users, even remote, and all of them will always see on their computer an updated version of the virtual prototype.

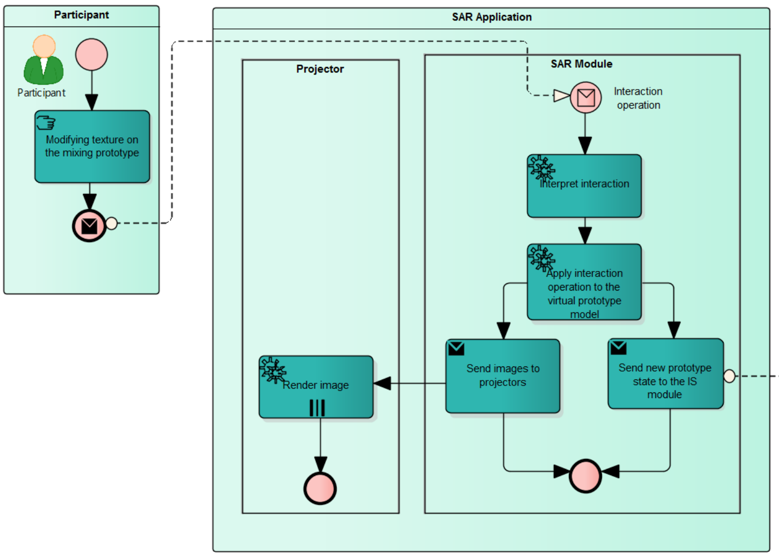

Figure 10 Modify the Virtual Prototype state using the SAR module

The process flow performed when the modification is made using the IS module is a little different (Figure 11). The process is as follow:

- The user, using the web browser, modifies the texture applied to the virtual prototype;
- The modification is sent to the backend server.
- The backend server saves the new virtual prototype state and sends a notification to the SAR module and to all connected users using a web browser.
- When the SAR module receives the notification, it analyses the new virtual prototype version and sends the new images to the projectors to display them onto the physical prototype.
- Other connected web browsers receiving the new version will also render the new virtual prototype version on the screen.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

In both cases, when the manipulation is made using the SAR module, or using the web browser (connected to the IS module), the available operations are the same. The participant can:

- Scale, rotate or translate an image.
- Add or delete a new image.
- Change the background color.
- Change layer order.

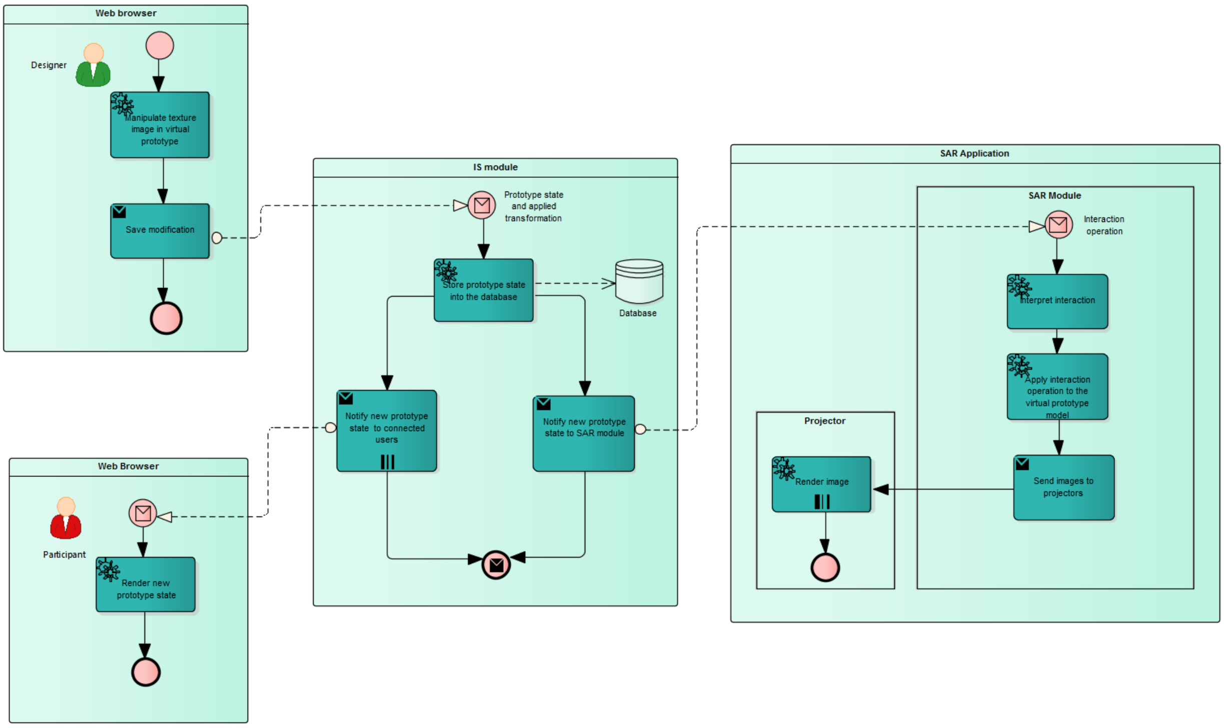

Figure 11 Modify the Virtual Prototype state using the IS module

When the participant moves the physical mockup, only the SAR module participates. When the object is moved in the space:

- The tracking device detects the new position and sends the new values to the SAR module.
- The SAR interprets the new position based on its internal reference system.
- New images are built for each of the connected projectors.
- Each projector displays the new generated images.

This operation does not imply any state changes in the virtual prototype, so the IS module remains untracked and it will not be further analyzed in the reporting section.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

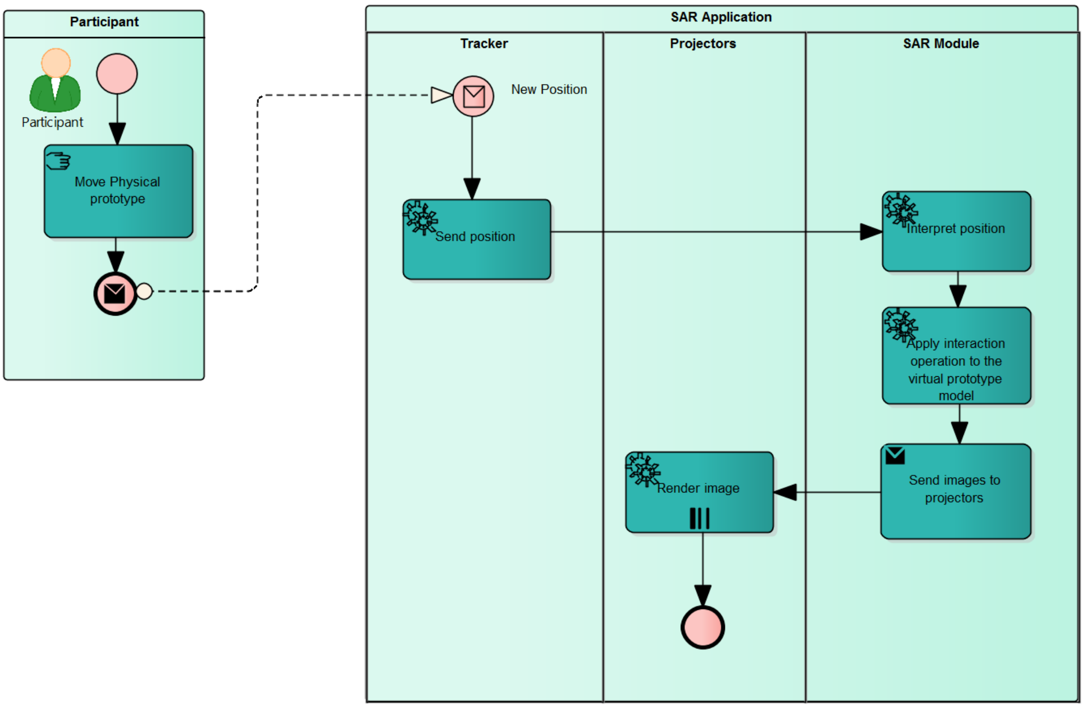

Figure 12 Move Prototype

In comparison to other processes, the participant does not need to use any piece of software when manipulating the physical mockup. The tracking devices automatically detects the changes of the spatial position and performs the needed transformation and operations to display the new images according to the new position and/or perspective.

6. DEPLOYMENT VIEW

The deployment view of the platform describes the software that must be installed to run a co-design session. In addition, this view allows identifying the execution environment of each of the modules and submodules. The presented representation is only one possibility of several available. Figure 13 presents the SPARK platform running in three separate execution environments. Each execution environment represents a device or computer where at least one module or submodule is installed. This configuration can be applied for example, when the interaction application is running on a tablet device, the SAR module is running on a computer, and the execution environment of the IS module is the cloud.
Following the description of each piece of software that is presented in this deployment view:

- The tracking software is the piece of software that comes with the tracking devices or technology, usually it is a middleware that detects the position of the tracked object and sends the information to subscribed software, which receives this information. In our case, the connected software is the SAR module.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

- The SAR module is the developed piece of software that handles all the augmented reality tasks.
- The interaction module captures manipulations of the virtual prototype and sends them to the SAR module for the interpretation. This developed piece of software is logically a submodule of the SAR module, but as it can be deployed on a separate device it is presented separately.
- The Web proxy is technically the web server that provides static web content. In addition, it provides advanced features such as the use of SSL certification (Secure Socket Layer).
- The MoM, as described in the deliverable 2.4, allows an advanced synchronization between the IS module and the SAR module.
- The Search engine is an installed software that adds high performance features to the IS module.
- The database server contains the data model presented in Figure 8.
- The IS module is the backend application that provides resources to the IS module, and to connected web browsers.

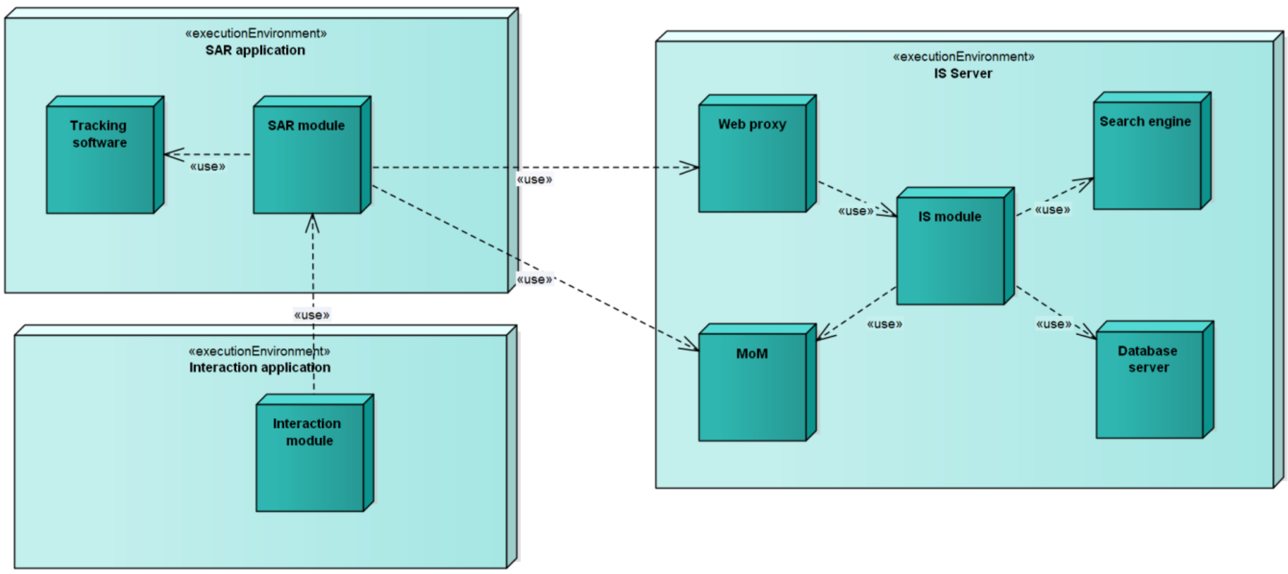

Figure 13 Deployment view of the overall platform

7. CO-DESIGN SESSION EXECUTION

A co-design session is previously prepared before being execution, this by using third-party software, and the components that are part of the SPARK platform. This set of operations needs to be described as are fundamental of the correct execution of a co-design session. Basically, it is required to prepare the digital content to use during the session, and to calibrate the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417

platform to correctly track the physical prototype and display the content on it. These can be summarized with the following steps:

Using third party software to create assets or files that will be used during the session:

- Creation of the 3D mesh of the prototype; this by using 3D modeling software such as 3D Max, Rhino, Blender, among others;
- Generation of the UV map for each component of the 3D model; this also can be done by using the previous 3D modeling software;
- Preparation of the graphic elements in separated layers; this is done by using graphic editors such as Photoshop, Gimp, etc.;

Using SPARK modules (IS, Calibration)

- Upload of the assets (3D models and graphic elements) on the IS;
- Definition of the projectors' and cameras' parameters (position and rotation with respect to the same reference system, FOV and aspect ratio) inside the session room;
- Generation of a file with the platform's setup data.

Following that, the session can start by using the SAR and IS module:

- Initializing the session in the IS module;
- Uploading the setup data (provided by the execution of the calibration) to generate inside the rendering engine the virtual replication of the real platform;
- Executing the SAR to connect to the IS system to download all the assets to have them available in the SAR environment.

8. CONCLUSION

The SPARK platform is a complex ICT platform that combines several physical and logical components such as tablet devices, PC computers, tracking devices, projectors, servers, but also some middleware servers, etc.

This document shows different perspectives of the overall platform solution proposed by the SPARK project. The objective of each perspective or point of view is to better understand how the platform will be built, how the modules will interact, and what are the specific task each module is responsible for. The different perspectives also explain possible configurations of the installation of the platform. The latter is flexible enough to be adapted based on the chosen business plan. To have for example parts of the platform in the cloud, and rent its usages.

The solution proposed in this document is platform independent, that means that each presented module and interaction can be developed by using any available technology. This with some restrictions, for example the technology to use for the SAR module should allow 3D scene manipulation, and the technology to use for the user interface coupled to the SAR module should allow some gesture recognitions.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No.688417.
H2020_ICT_2015_SPARK_688417