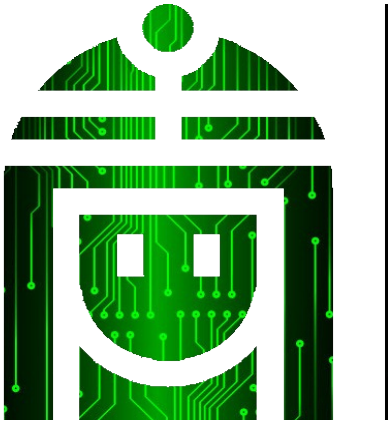
CLEOPATRA

CoLlaborative ExploratiOn of cyber-PhysicAl culTuRal lAndscapes



cleopatra-project.cloud

Cleopatra Project Deliverable: D1.2

Architecture design and technological stack

Authors: Salvatore Venticinque, Massimo Ficco, Rocco Aversa, Angelo Ambrisi

Università degli Studi della Campania *Luigi Vanvitelli*

Dipartimento di Ingegneria

Dipartimento di Lettere e Beni Culturali

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

About CLEOPATRA

The Cleopatra Project aims at increasing the knowledge of the archaeological and historical-artistic sites and to develop new communication techniques for Cultural Heritage. The objective is to promote and rediscover the sense of history and cultural identity by the valorisation of lesser-known areas and sites of the Campania region, but no less interesting.

Two experimental scenarios will be designed proposing "Diffused Museums", through which the territory is known, and developing archeo-trekking or slow-tourism (i.e., cycling tourism), based on the protection and preservation of places requiring responsible, personalized and non-massive use.

The projects aims at achieving the following objectives:

- the creation of user friendly services both in outdoor (archaeological sites, squares, etc.) and indoor spaces (historic buildings, museums that store material documentation), in order to structure thematic itineraries through history, culture and art that involve the territory in its entirety for the construction of an integrated network for tourism promotion which is currently lacking;
- the organization of a system to guide the tourist, combining, from time to time, archaeological and historical-artistic, naturalistic, faunistic and geological elements based on their interests through an informative support; in the case of scenario 2, its peculiarity is emphasized by the naturalistic and geo-environmental background, in which the archaeological sites are located: currently they can be visited only with the aid of expert touristic guides;
- the realization a close interaction between user and avatar; the avatar will guide the tourist to places of difficult access through vocal and visual advices, choosing the most suitable routes based on a series of criteria, such as the available time, the ability of users to move in difficult contexts, clothing and the possibility to take scenic routes; at the same time the user, in his interaction with other users, can recommend new paths and report new elements, thus providing starting points for future research;
- to overcome problems, thanks to the help of the avatar, such as limited interaction with text documents, especially in open spaces, the lack of user, and an insufficient or wrong location of POIs.

For more information

Prof. Salvatore Venticinque, *salvatore.venticinque@unicampania.it*

Prof. Giuseppina Renda, *giuseppina.renda@unicampania.it*

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

Executive Summary

This report presents the software architecture and the technological stack of the Cleopatra platform is presented.

We start from the conceptual model introducing the analysis of requirements by the UML formalism. Than we sketch the software architecture by UML component diagrams and UML sequence diagrams.

A overview of the technologies identified and experimented are briefly introduced, providing a snapshot of the preliminary prototype integration..

Finally we report some complementary activities that focused on the implementation of a back-end tool that will support the building of case studies which will deployed by the designed platform.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

Table of Contents

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

Table of Figures

List of Tables

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

List of Abbreviations

Table 1: List of abbreviations

Abbreviation	Explanation
NLP	Natural Language Processing
MAS	Multi Agents System
KB	Knowledge Base
FEAPI	Front End API
API	Application Programming Interface
XMPP	E**x**tensible **M**essaging and **P**resence **P**rotocol

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

1 Requirement Analysis

1.1 Conceptual Model

Figure 1 shows the conceptual model of the Cleopatra framework. Virtual Agents coordinate a collaborative user-experiences by either one-to-one or one-to-many interactions with human users. Like a human guide the virtual agents can propose and illustrate arguments or can answer questions asked by any visitors. Moreover, they can be at the same time on multiple distributed physical and virtual sites, and can start private conversations with single users, if necessary, as a personal guide.

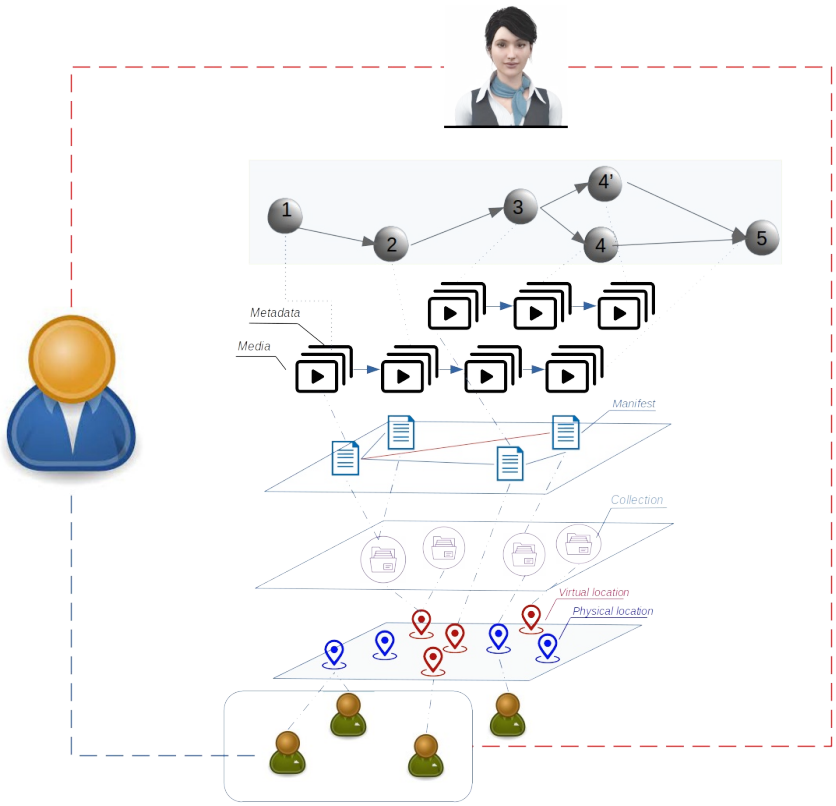


Figure 1: Conceptual model of the Cleopatra Framework

At the lower level, we can see blue-physical points of interest, which are geographical distributed, and red-virtual locations, which can have e geographical attribute or not (a virtual access to a physical museum or multi-media information about a physical location). At the middle level, we have the meta-representation of locations, by collections of manifests. A manifest is a representation of either a concrete artefact or an abstract cultural content. It can be the virtual representation of a building, a monument, a masterpiece, a natural landscape, the room of a museum or any other self-consistent information, trough a sequence of semantically related multimedia objects with their meta-data. The Cleopatra agent will be able to build a cultural itinerary proactively, or answering the users' questions, complementing an interactive storytelling with the presentation of semantically related multi-media.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

1.2 Use Cases

1.3 High Level Use Case and Main Actors

The Use Case diagram shown in Figure 2 provides a high level representation of the cyber-physical collaboration among software agents and human users, enjoying a diffused museum.

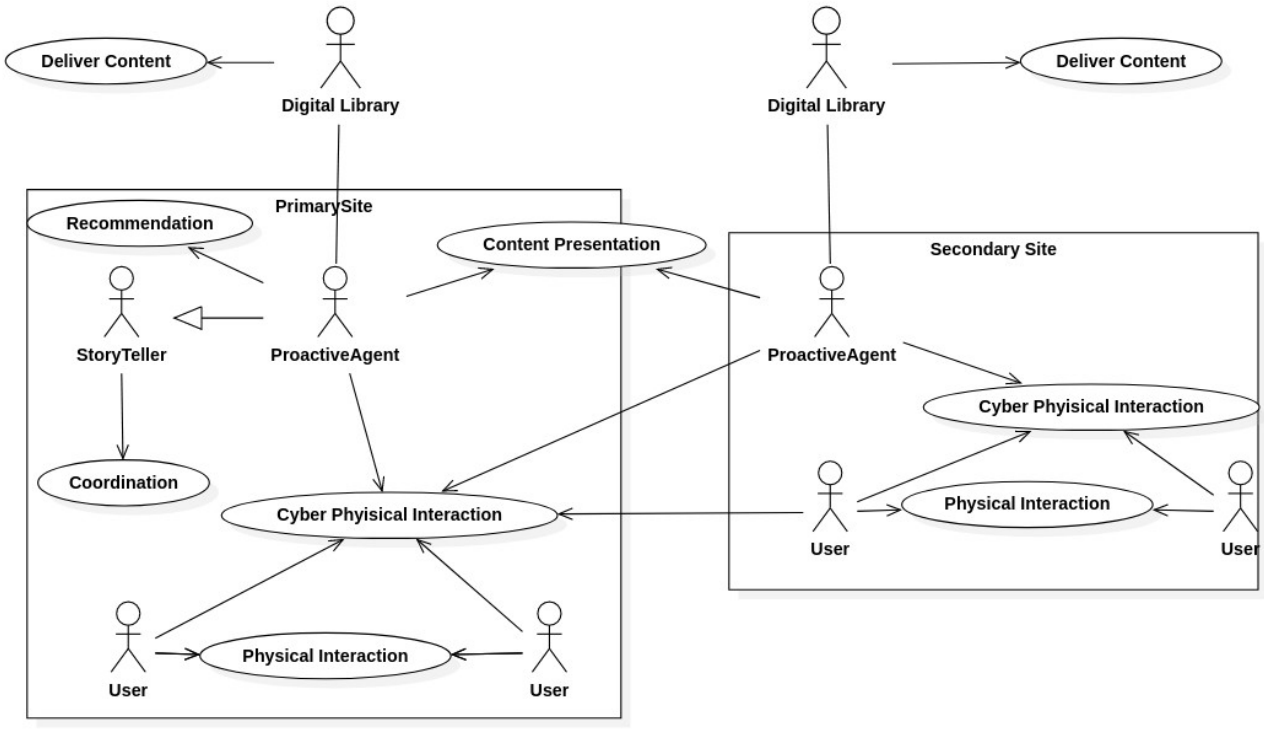


Figure 2: Cleopatra Cyber-physical System

Each user belongs to a primary site (the one she is visiting live or she has joined virtually). In such an environment, she can communicate with other human users or with Proactive Agents located in that site. One Proactive Agent per site is in charge of leading the interactive storytelling and of coordinating the group of users, whose are visiting that site. It plays its role sending messages or presenting digital contents, but also suggesting the interaction with other agents and users located in secondary sites. Even tough, the social interaction between remote users is free, the interaction among local users is monitored and controlled by the local Proactive Agent to maximize the utility of the social experience, leveraging an emergent social behaviour. On the other hand, any secondary Proactive Agent can answer to primary users, or can be stimulated by the primary Proactive Agent to participate to the conversation, but cannot start autonomously any action in order to not affect the local coordination. We assume that each Proactive Agent has its own knowledge about the digital library, describing its own site and eventually is aware about semantic link to cultural content belonging to other sites. In this way, it can recommend virtual or physical itinerary across other sites or can invite, during its storytelling, other Proactive Agents to present related cultural contents.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

1.4 Multi Agents Use Case

The proposed solution distributes the Cleopatra functionalities among different agents, which implements a Multi Agents System (MAS).

NLP Agents usually react to user request providing answers and eventually executing actions. The existing technologies support advance functionalities for natural language understanding, but do not integrate mechanisms to model proactive behaviours. Moreover they support only diadic conversation.

For this reason, in Figure 3 a proactive agents is in charge to implement a bridge that transforms diadic conversation with NLP agents into multi-users conversations with the users. Moreover, the proactive agents pursuing their own goals can decide to send recommendations to the users, to adapt the story telling to a the group of users or to a single users.

We also identified a class of front-end agents, which have a diadic interaction with the proactive agent for:

- notifying events originated from client application or events detected by sensors
- receive control message to start actuators or function in client applications

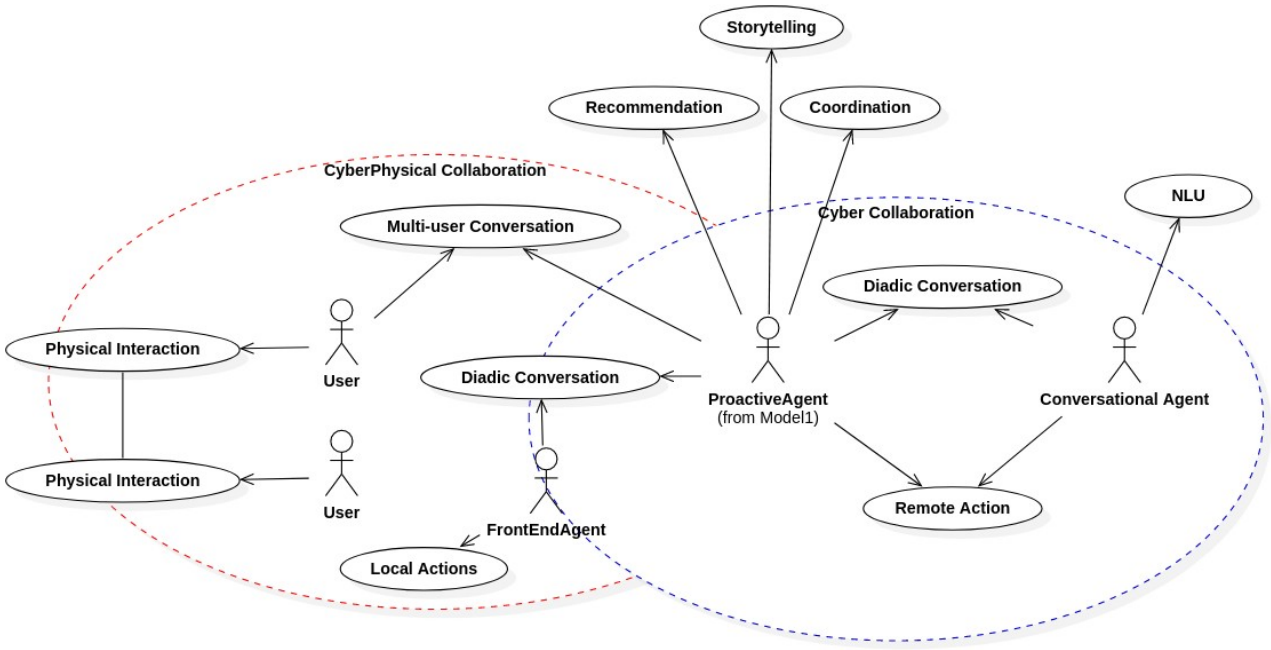


Figure 3: Multi-Agents Use Case

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

2 Architecture Design

2.1 Component diagram

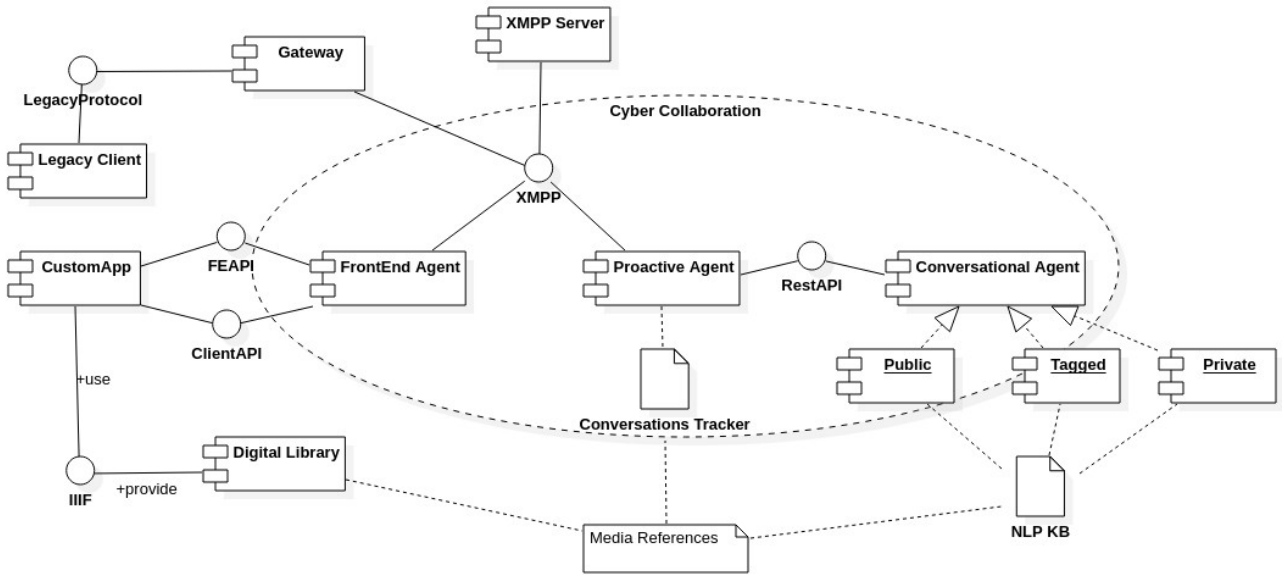


Figure 4: Cleopatra architecture

The component diagram of the Cleopatra platform is shown in Figure 4. The main functionalities implemented have been designed to support multi-user communication, proactivity and standard protocols. Multi-users communication aims at advancing state of art technologies for development of NLP conversation agents, which usually support only diadic interactions. It means that an agent can run multiple conversations at same time, but with only one user each. Moreover, usually conversational agents send a message only for responding to one user's request. On one side, the Proactive Agent has been designed to implement the autonomous and proactive behavior of the cyber side of the Cleopatra platform. On the other side, it implements a gateway between a multi-user conversation that is built upon the XMPP protocol and the diadic interaction with a conversational agent, which uses the NLP technology to provide cyber intelligence to the system [1]. The XMPP protocol is the main choice that allows to exploit all the provided functionalities, but other legacy communication channels will be supported to grant the access by the most widespread messenger applications such as Telegram and Facebook. The Proactive Agent will be able to receive messages from users and forward them to three conversational agents in charge of handling different class of message, according to its recipient. One-To-All messages, whose recipient is the multi-user chat, are public messages without a '@tag'.

They are addressed to everyone in the multi-user conversation. The Proactive Agent can decide whether to process that messages, forwarding them to the conversational agents or to ignore them based on well defined criteria. One-To-Cleopatra messages are sent to the multi-user chat with the '@cleopatra' tag. They are still public, but must be forwarded by the Proactive Agent to a different conversational agent, that will always respond with tagged message visible by all users. One-To-One messages, whose recipient is the Cleopatra agent itself, have been sent privately by the user. The Proactive Agent will forward the message to the last conversational agent and will return back a private response. A distributed shared memory is exploited by the Proactive Agent and the Conversational agents to build a collective knowledge about the multi-user session.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

Multi-media delivery and presentation is another task of the Proactive Agent that supports the delivery of cultural information carrying on aninteractive storytelling. The storyline will include the presentation of digital contents according to the planned plot and semantically related to the current conversation.

The utilization of the IIIF (International Image Interoperability Framework) for the standard delivery and presentation of digital objects allow us to maximize both the cultural and the technological impact of the research activity. IIIF defines several application programming interfaces for describing and delivering images over the web, as well as "presentation based metadata" [11]. It supports the definition of structured sequences of canvas, which can include mainly images, but also annotations and other kind of media contents. The digital library, which ia an outcome of the Cleopatra project, will be made available to external applications by such an interoperable interface. On the other hand, the agents will be able to access and delivery any other compliant IIIF digital library, which has available in the network.

At the stat of art, explicit references link each intent, which is recognized by the Natural Language Understanding unit of the conversational agents, to objects of the digital library. Multi-users coordination is handled by the Proactive Agent pursuing the fulfillment of well defined objectives, which aim at leveraging emergent behaviours by the community of human users.

As an example let us consider what it is shown in Fig. 4. On left side, the yellow nodes represent user's intents. The user's intention is detected by the NLU capability of the NLP agent on each received message. In a set of pre-defined conversations flows (stories), the NLP agent knows which one is going on with each user.

2.2 Sequence Diagrams

2.2.1 User request

In Figure 5 we see the interaction protocol related the the message exchange between users and a software agents trough a multi-users conversation. In particular, the user request is forwarded to tall participant who have joined the chat, as well as the response returned by the agent.

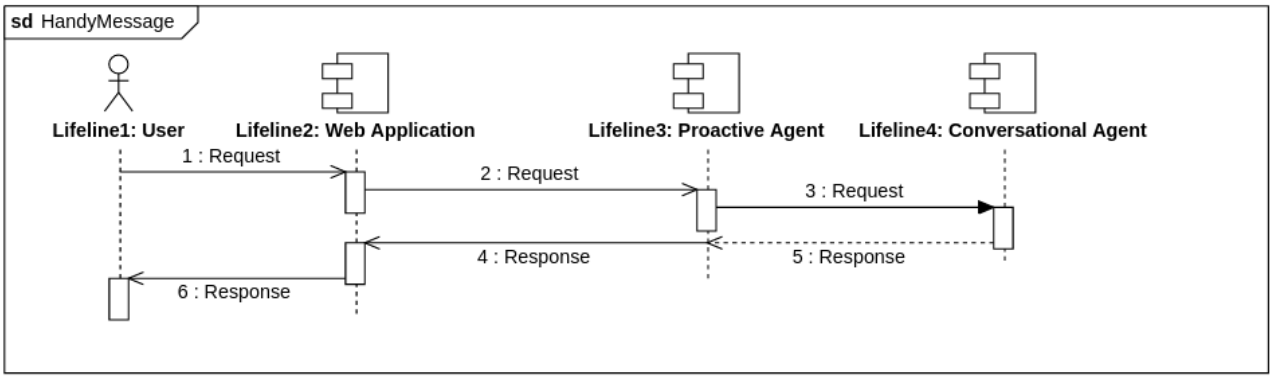


Figure 5: User Request

2.2.2 Asynchronous Event

Asynchronous events originate from the client application or from distributed sensors. They notify to the proactive agents a change in of the user's context. They extend the context awareness of the MAS that is

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

complemented by the result of NLP. The update or historical information are stored in a persistent knowledge base (KB).

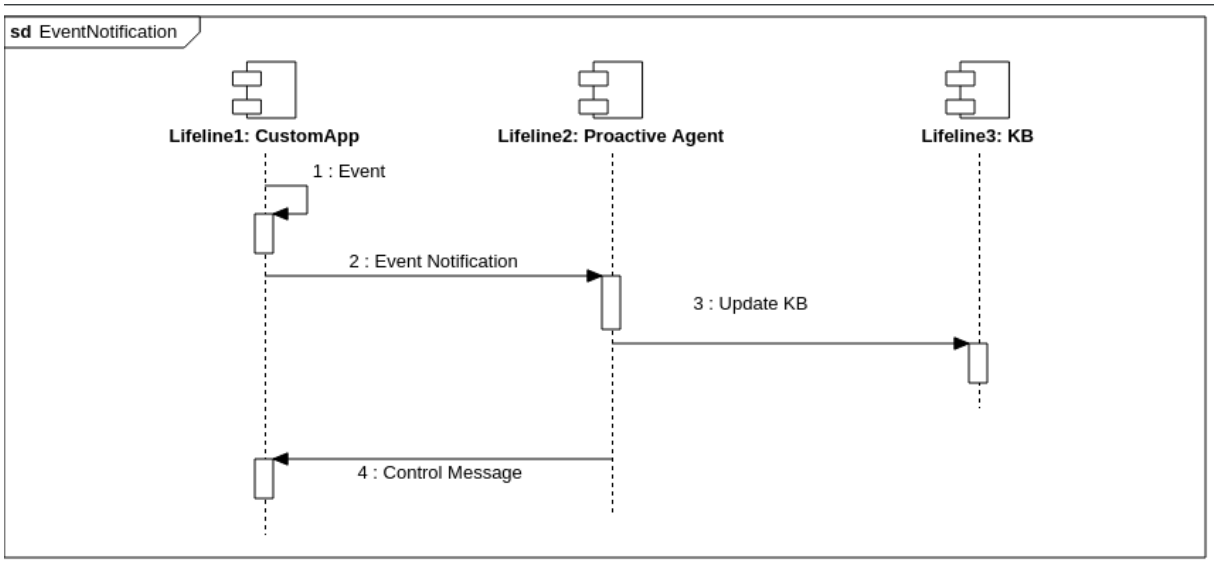


Figure 6: Event notification

2.2.3 Proactive Actions

In Figure 7, the sequence diagram shows how a software agents execute proactive behaviours. A proactive behaviour is executed by a software agent to achieve a goal that can be defined as a preference about the user's context. Any event, which is explicitly originated by the user, which has been independently detected by the application (e.g. by a sensor), or which is internally originated by the software agent (e.g. a timeout), may trigger a proactive behaviour. The behaviour may update the internal state of the agent (2), or my trigger a control message to the client application (e.g. updating what is displayed to the users) or may invoke the conversational agents to retrieve the most convenient message to be sent to the user to achieve the active goal.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

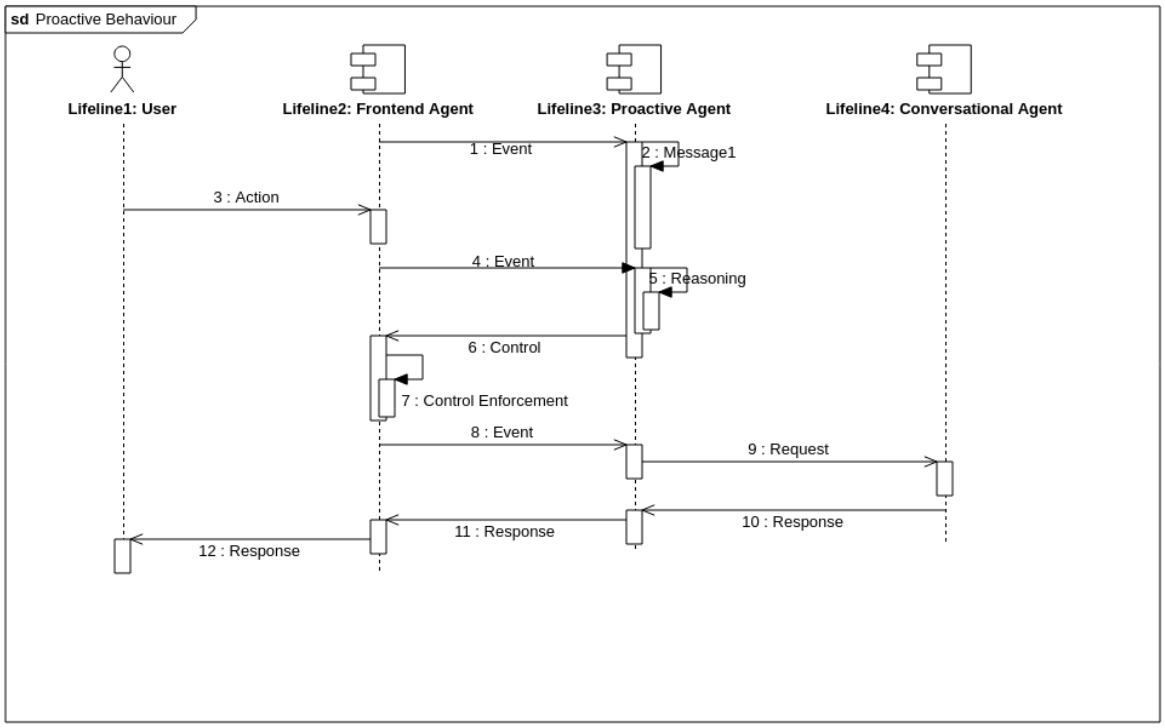


Figure 7: Proactive Behaviour

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

3 Technology scouting

A number of technologies have been identified and experimented to build a software prototype that implements the presented architecture of the Cleopatra platform.

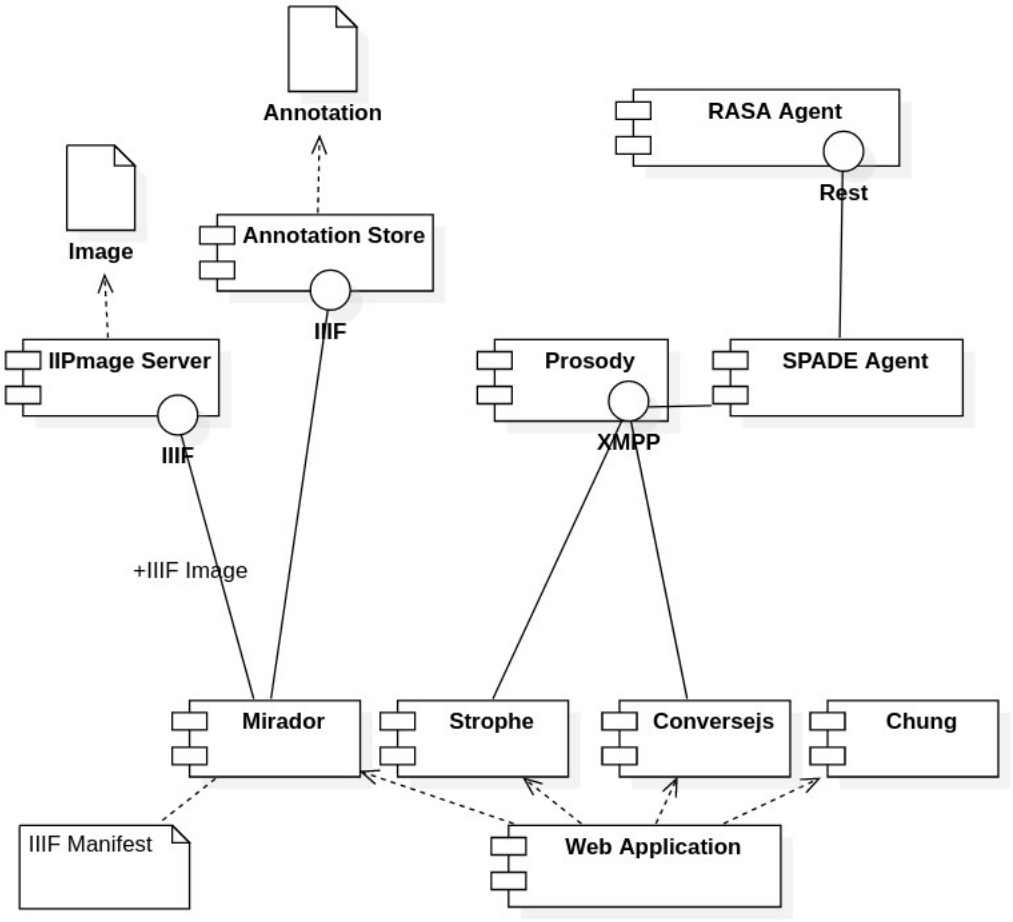


Figure 8: Software technologies.

In Figure 8 the UML Component Diagram of the implemented prototype illustrates the baseline open source technologies integrated both at client and server side. The conversational agents have been built by the RASA open source platform[1], but also Google dialogflow[2] has been evaluated as alternative solution.

The Spade technology has been used to implement the proactive agent[3]. Such a library supports natively the communication via the XMPP protocol. Prosody[4] is the XMPP server technology, which has been chosen to provide a peer to peer overlay for the cyber-physical interaction. In addition to to the legacy channels (such as Facebook, Slack and Telegram), we support and advance multi-modal interaction with the embodied conversational agent integrated into the Mirador IIIF viewer[5].

[1]https://rasa.com
[2]https://dialogflow.cloud.google.com/
[3]https://github.com/javipalanca/spade
[4]https://prosody.im/
[5]https://projectmirador.org/

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

Such solution exploits the IIIF protocol for the interoperable exchange of images trough the web and allows for the integration of digital libraries which have been exposed by this standard. The IIPImage[6] server is used to deliver high resolution images of our self-hosted digital libraries.

3.1 Prosody

Prosody is a cross-platform XMPP server written in Lua. Its features include low resource usage, ease of use, and extensibility. Prosody is used by our architecture in order to let the user to communicate with the Cleopatra framework in an XMPP chat room in which the users will be connected and exchange message with each other and with the platform. Moreover, the Prosody XMPP server will be used also as communication bridge between the Cleopatra Agent and a "Mirador Manager" application that has the responsibility of control and management of the Mirador IIIF Viewer.

3.2 Spade Agents

Smart Python Agent Development Environment (SPADE) is a multi-agent systems platform written in Python and based on instant messaging (XMPP). It allows to develop agents that can chat both with other agents and humans. The Cleopatra agent will be developed using this technology, developing behaviors that will be activated in correspondence with certain events and upon receipt of defined messages. With this technology it is also possible to develop intelligent agents, that is agents that have internal aspects of artificial intelligence. This feature will be particularly important and used during the development of agent Cleopatra. The Cleopatra agents will also have flexible behaviour: reactive, proactive, social.

3.3 RASA Platform

Rasa is an open source machine learning framework to automate text-and voice-based conversations in order to build contextual assistants and voice assistants. Rasa helps you build contextual assistants capable of having layered conversations with lots of back-and-forth. In order for a human to have a meaningful exchange with a contextual assistant, the assistant needs to be able to use context to build on things that were previously discussed – Rasa is also an high scalable framework. Rasa also has components that are equipped with NLP (Natural Language Processing) and NLU (Natural Language Understanding) algorithms which therefore allow not only the understanding of the user's intention to find a certain type of information or to perform certain actions on the system talking to the agent, but also learning from the responses given by users in certain conversations. This platform will be used to implement conversational agents which, coordinated by the orchestrator agent Cleopatra, will be able to converse with the user and help him in achieving certain conversation goals.

3.4 DialogFlow

DialogFlow is another platform, similar to dialogFlow, for building voice and text conversation applications based on machine learning.

To understand even better what it is for, just read the definition of Google Cloud:

"Dialogflow is a natural language understanding platform that makes it easy to design and integrate a conversational user interface into your mobile app, web application, device, bot, interactive voice response system, etc."

This technology was developed to analyze texts and voices and to return answers in both written and vocal form. Basically, the platform is an advanced solution to create real automated conversations, text or verbal chats, which allow communication between human and machine.

[6]https://iipimage.sourceforge.io/

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

The software supports 14 languages and can be integrated both with the main chat platforms such as Google Cloud Assistant, Facebook Messenger, Slack, Skype, Telegram, and within its own applications, via the service API. The operation of DialogFlow is based on the creation of bots that learn on the basis of models that evolve automatically and that can be used for both chat-bots and voice-bots. Thanks to machine learning, predefined entities (times, dates, keywords) are recognized and the intent of the user speaking or writing is understood. In this way, the software generates consistent responses that satisfy the human interlocutor. Within the Cleopatra project, DialogFlow will be used to implement conversational agents within the most famous social networks in order to provide the end user an interaction with the platform through well-known interfaces that are already in the user's experience.

3.5 IIPMage Server

The IIPImage server is a feature-rich high performance image server engineered to be stable, fast and lightweight. IIPImage is an advanced high-performance feature-rich image server system for web-based streamed viewing and zooming of ultra high-resolution images. It is designed to be fast and bandwidth-efficient with low processor and memory requirements. The system can comfortably handle gigapixel size images as well as advanced image features such as 8, 16 and 32 bits per channel, CIELAB colorimetric images and scientific imagery such as multispectral images, image sequences and 3D surface topologies.

3.6 Mirador Viewer

Mirador is a configurable, extensible, and easy-to-integrate image viewer, which enables image annotation and comparison of images from repositories dispersed around the world. Mirador has been optimized to display resources from repositories that support the International Image Interoperability Framework (IIIF) APIs. It provides a tiling windowed environment for comparing multiple image-based resources, synchronized structural and visual navigation of content using OpenSeadragon, Open Annotation compliant annotation creation and viewing on deep-zoomable canvases, metadata display, book reading, bookmarking and more. Mirador viewer will be used by the Cleopatra platform to view the generated IIIF libraries and provide the user with a visual dimension of the conversation they are having with the Cleopatra agent. Mirador will also be extended with ad hoc components in order to increase its interactivity and the degree of user immersion.

3.7 Strophe

Strophe.js is an XMPP library for JavaScript. Its primary purpose is to enable web-based, real-time XMPP applications that run in any browser. It is the base technology for the Converse.js client embedded in our web application, but it is also used as a control channel that allows for the message exchange between the front-end agent and the proactive agent.

3.8 Converse.js

Converse is a web based XMPP/Jabber chat client. You can either use it as a webchat app, or you can integrate it into your own website. It's 100% client-side JavaScript, HTML and CSS and the only backend required is a modern XMPP server. Converse.js is therefore used as the means of communication between the user and the Cleopatra agent when interfacing must take place via textual channel on a virtual museum. In fact, on the web page of a demostrator, the converse.js interface will be displayed above the mirador interface, allowing the user to converse with the agent while viewing the artwork or site of interest.

3.9 A preliminary prototype

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

The development of the presentation layer and the integration of the functionalities which will be defined in each case study will be extensively described in the deliverable D3.1 (“Presentation Mechanisms”).

However, the preliminary result of integration of the conversational interface with the Mirador viewer is shown in Figure 9. An embodied agent shows to the user his position on the map and provides information view about the main archaeological find in the area. The embodied agent is implemented by an open source javascript library[7] that allows for a real time rendering in a web browser and a real time text to speech conversion. The communication is occurs in a chatroom by the conversejs client[8], while control messages, which allows to the agent for controlling the mirador viewer (e.g: change image, zoom, select annotation) are exchanged by an hidden parallel XMPP session, handled by the strophe.js library[9].

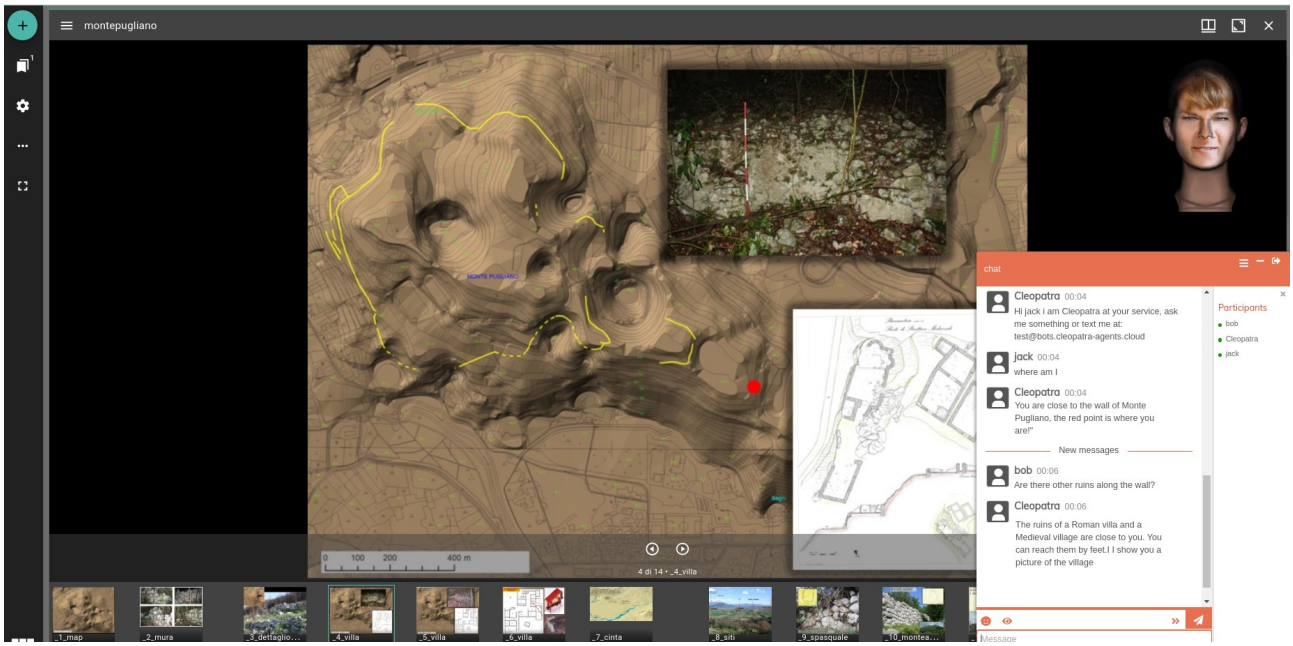


Figure 9: Web interface of a preliminary prototype implementation

[7]https://sourceforge.net/projects/chatbotchung/
[8]https://conversejs.org/
[9]https://strophe.im/strophejs/

The research leading to these results has received funding by the University of Campania “Luigi Vanvitelli” from VALERE 2019 program.

4 A back-end support for development of Cleopatra locations

Cleopatra virtual museums, which complement o replace physical locations, enabling the digital multi-user interaction needs to be designed and will be built by cultural experts, who needs to collect and connect multi-media contents in a way which can be presented to the users and handled by software agents.

As part of the Cleopatra project, a platform for the support and management of multimedia contents within a virtual museum has been designed.

The platform will be based on laravel that is a free, open-source PHP web framework, intended for the development of web applications following the model – view – controller (MVC) architectural pattern and based on Symfony. The management software is able, through an advanced ORM (Object-Relational Mapping) system, to communicate with a database that has been structured to facilitate its translation to the IIIF 3.0 standard supported by the most famous software for the use of multimedia content in virtual platforms for museums (such as Mirador). Eloquent ORM (object-relational mapping) is an advanced PHP implementation of the active record pattern, providing at the same time internal methods for enforcing constraints on the relationships between database objects. Following the active record pattern, Eloquent ORM presents database tables as classes, with their object instances tied to single table rows.

Figure 10 shows the ER diagram on which the management software is based. Proceeding to the description with a Bottom-Up approach, it can be seen how the relationships and the components refer to the hierarchical structure of the IIIF standard. At the lowest point of the hierarchy is the work of art, which has a name, a description, a reference to the author and to the manifest it is associated with. Proceeding upwards we find the manifest which are nothing more than a set of artworks, each manifest refers to a collection which in turn

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

refers to an artshow (like museums or archeological sites).

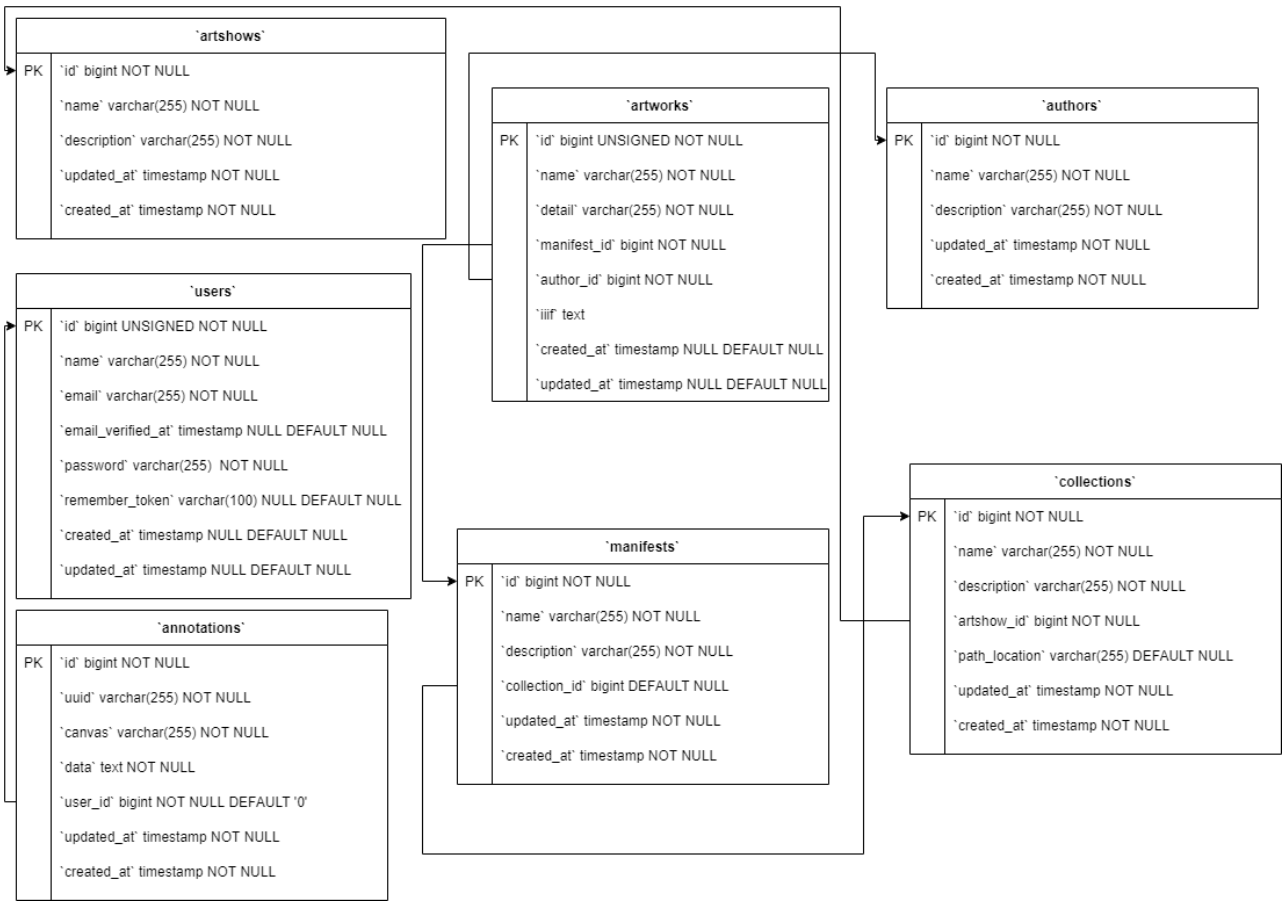


Figure 10: ER diagram of Cleopatra back-end software.

Figure 11 presents the component diagram of the web application. User requests are handled by routing software which dynamically redirects to the corresponding View. Once the view has been reached, it is populated by a Controller that manages the data to be displayed and the possible actions on them. CRUD operations are possible thanks to the Query Builder component that allows interaction with the database shown above. The Routing component also allows access after logging in to the Server IIIF Generator interface which in turn saves the IIIF library generated within the FileServer so that it can be reached by a Mirador instance on the WebServer.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

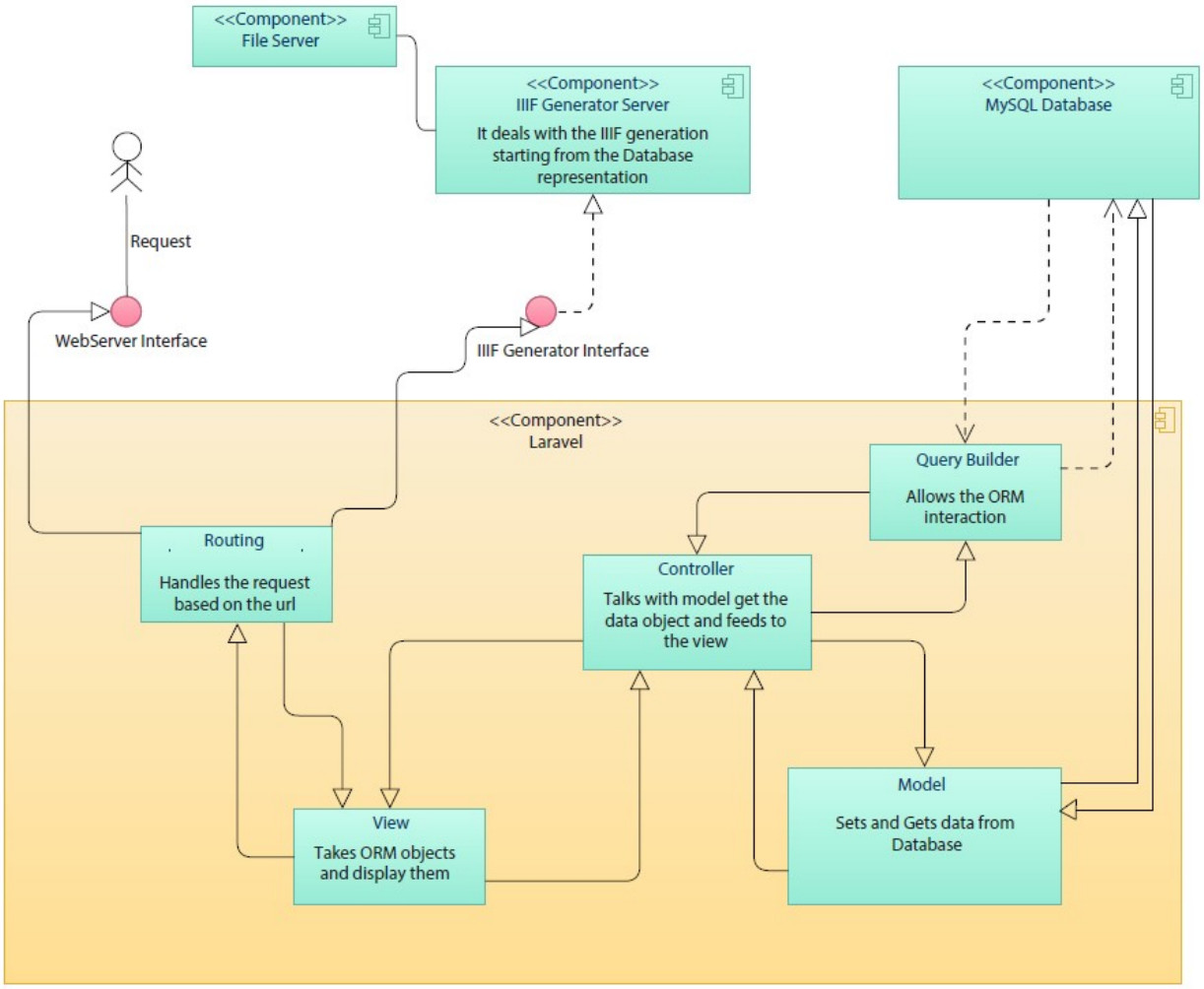


Figure 11: Software architecture of the back-end web application

Figure 12 also provides a more detailed class diagram which focuses on the methods that controllers make available for class management. In addition to the classic actions that an ORM software allows, note how it is also possible to invoke an aggregation method that returns the objects associated to an higher component in the IIIF hierarchy.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

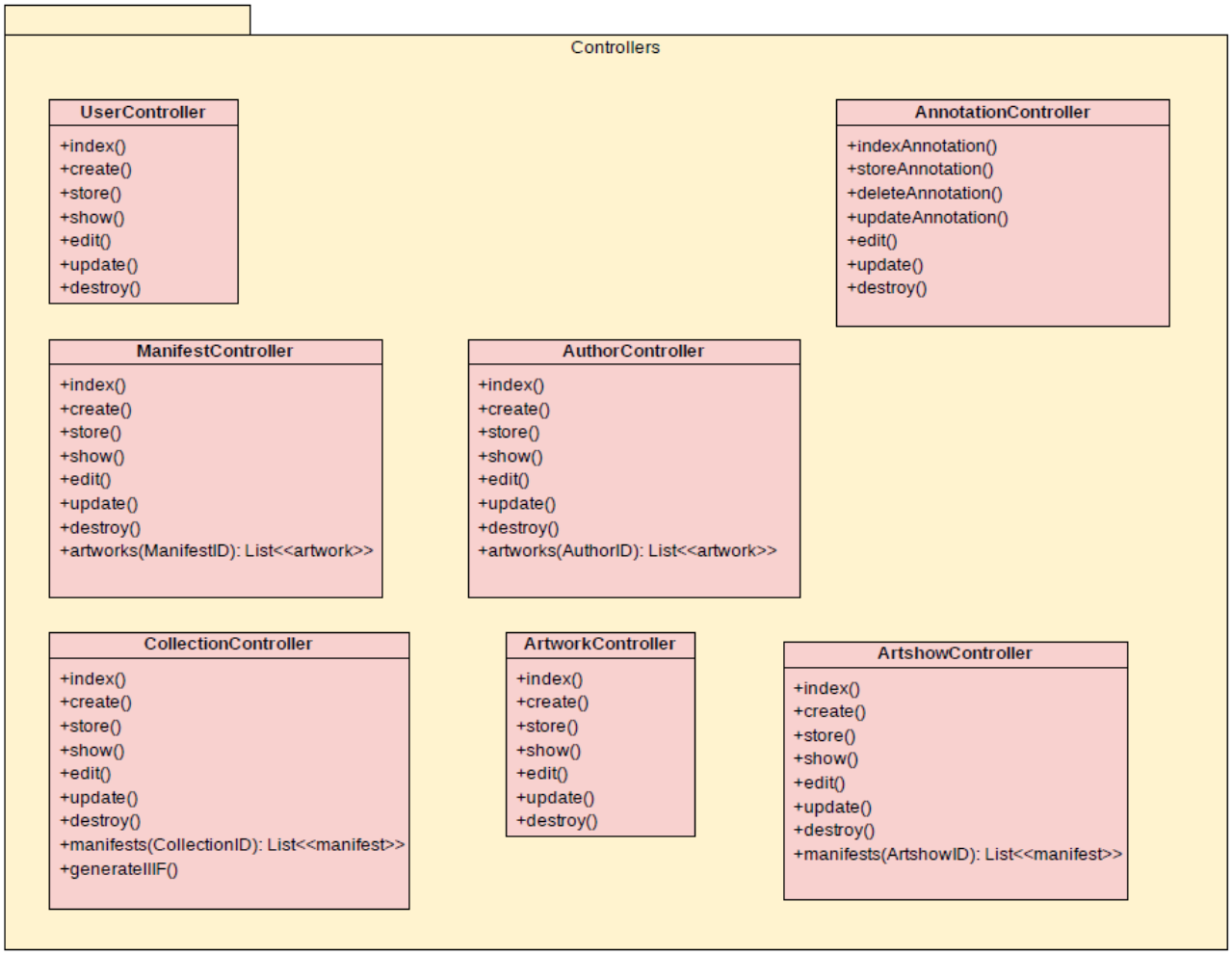


Figure 12: Class diagram of the back-end web application.

Finally, the IIIF Generator server will expose a REST API that is able to receive the name of the collection whose IIIF is to be produced, to generate the corresponding IIIF and return the parent path that contains the generated files.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.

5 Conclusions

To support a seamless interaction among users and software agents, who have to facilitate the social fruition of cultural contents, the main requirement is the utilization of a common communication channel that allows for the conversation and the delivery of media contents.

The effectiveness of such interaction model is related to the relevance of delivered contents and information, but also to the utilization of existing social network (and eventually legacy applications) and of interoperable protocols.

For this reason, the architecture design presented in this deliverable has been conceived to allow, by loosely coupled integration of off-the-shelf and open source technologies, the experimentation of any advanced functionalities trough an original platform, but also the delivery of fundamental services trough well established channels like facebook or telegram.

This deliverable defines the building blocks which will allow to satisfy the requirements defined in “D1.1 Collaborative Models”, “D5.1: Case Studies Definition”, but it also provides the backbone for the implementation of the “D2.3 Reasoning and learning techniques and technologies”, “D3.2 P2P platform”, and “D3.1 presentation mechanism” trough the identification of the key technologies that will be used in the second phase of the project.

The research leading to these results has received funding by the University of Campania “Luigi Vanvitelli” from VALERE 2019 program.

6 References

[1] Ambrisi, A.; Aversa, R.; Ficco, M.; Cacace, D.; Venticinque, S.. (2021) *Intelligent Cloud Agents in Multi-participant Conversations for Cyber-Physical Exploitation of Cultural Heritage.* doi:10.1007/978-3-030-75078-7_11. pp.97-106. in lecture notes in networks and systems - isbn:978-3-030-75077-0.... in lecture notes in networks and systems - issn:2367-3370 vol. 227

[2] Venticinque S., Aversa R., Ficco M., Ambrisi A., Branco D., Renda G. and Mataluna S. (2021) *Intelligent agents for diffused cyber-physical museums*. In: David Camacho, Domenico Rosaci, Giuseppe M.L. Sarné, Mario Versaci (eds) Intelligent Distributed Computing XIV. IDC 2021. Studies in Computational Intelligence,. Springer, Cham.

The research leading to these results has received funding by the University of Campania "Luigi Vanvitelli" from VALERE 2019 program.