

## Executive Summary

### D3.3 – CrossCult Integration and Testing Report

This deliverable presents the results of the integration and testing activities conducted in task T3.6, “Integration and alpha testing”. This task is focused on integrating the modules developed in the preceding tasks into the CrossCult Platform, and testing them in controlled environments, prior to deploying the solutions in the venues and pilots of WP5. In T3.6, the technical partners cross-validate their implementation, looking for correct outputs as well as computational efficiency and usability.

The modules to be tested are the following:

- **A Knowledge Base (KB).** It contains all the Knowledge Organisation components and data of the CrossCult Platform.
- **Profiler Service.** It provides different APIs to support the construction and gathering of the user profiles. This service is composed of the following microservices:
  - **A Question-based profiler:** to analyse the user’s answer to a questionnaire to extract relevant profile traits.
  - **A Location-based profiler:** to analyse users’ movements in the venue to infer relevant profile traits.
  - **A Carousel-based profiler:** to use a carousel of images to identify user’s interests.
  - **An Interaction-based profiler:** to analyse the interactions of the user with the App content in order to extract relevant profile traits.
- **A Recommender Service.** It provides different APIs to fetch recommendations for a particular user. This service is composed of the following microservices:
  - **A Person recommender:** to help creating well balanced teams using separate user profiles.
  - **An Item recommender:** to identify Points of Interests that are relevant to a specific user and concepts more or less closely related to given ones.
  - **A Path recommender:** to dynamically build tours respecting both the interest of a user and certain constraints.
  - **An Association recommender:** to identify associations in the Knowledge Base that can be used by experts to create narratives.
- **A Situation Analyser.** It provides different APIs to support the management and fetching of venue profiles.
- **A Social Service.** It supports all “social network” related activities that will be performed within the platform, excluding 3rd parties like Facebook or Twitter.
- **A Crowdsourcing Service.** It provides different APIs to support all crowdsourcing related activities, such as storing user-generated content and its analysis and validation.

To support the implementation of the different services, a series of tools have been made accessible to the development team. Among these tools, the most notable are:

- A. **GitLab:** a private collaborative git-based development platform that mainly acts as a central shared code repository. It is the main code-sharing platform of the project to be used by the CrossCult partners.
- B. **Nexus:** a private software artefact repository allowing publishing and sharing libraries/binaries between CrossCult developers. Nexus supports many different artefacts types like java or Docker artefacts.

To simplify developments, a microservice chassis was made available to the developers in the early months of WP3. A microservice chassis is a framework designed to facilitate the creation of new microservices by allowing handling most of the crosscutting technical concerns like exposing REST APIs, creating logs, handling error, or externalising configuration. The objective is to allow each team to focus on the development of robust logic elements and mutualise the time spent on technical issues.

In order to validate each service, a “contract first” development process has been followed. It is an iterative process that can be summarised by the following workflow:

1. For a specific service, the team in charge of development begins by writing the API description.
2. This API description is subject to review by the other teams to make sure it is consistent with other API services and that it supports the expected use of the service.
3. When the API description is agreed upon, the actual development can commence.

Microservices testing and integration results have been made for the CrossCult Backend and the Frontend separately:

- 1- The CrossCult Backend contains the following modules that underpin its operation:
  - The Knowledge Base.
  - The Authentication module.
  - The Logging module.
  - The Application Programming Interface Gateway module.
  - The Event Management service.
  - The Digital Resource service.
  - The App Generator Interconnection service.
- 2- CrossCult Frontend platform is the access point of the project and to almost all the functionalities of it, through which the different profiles in the project will be serviced. It has to cater for: unified authentication and access to the CC platform, access rights based on user profile, unified look and feel checking, accessibility through web technologies, and several other interoperability considerations. Frontend tools being developed are for Administrators, Museum Curators, Data Scientists, Experience Designers and users.

Initial investigations on the efficiency and scalability of the CrossCult Platform have been performed in order to deliver optimal experience to the pilot users. The CrossCult Platform should be able to deliver efficiently content and services to pilot users, avoiding sluggish behaviour and degraded performance that could degrade the user experience. Efficiency should be maintained under all foreseen amounts of the number of concurrent users and data volume.

Regarding integration, test deployments have been performed successfully and we have developed the necessary know-how to instantiate the platform for each pilot. The deployments have been mostly automated thanks to the standardisation efforts performed so far.

In parallel of the Apps development efforts of WP5, WP3 developers will continue to refine the existing components based on the feedbacks from the Apps developers. New features will be added regularly based on the effective needs of each pilot.

Regarding testing per se, code quality and test coverage will be continuously enhanced for all existing services. To support this, the use of SonarQube will be generalised and internal quality standards will be discussed in the following weeks so that all developments can be assessed on a common scale.

The integration scenarios for the four pilots will be translated to automated suites of tests that will be performed regularly to ensure compliance of the developments. They will be maintained up to date to reflect actual Apps needs.

Finally, different compositions of the available services will be tested. This work can feed the pilot development, opening new possibilities for Apps developers and fostering cross-pilot reuse of components.

[www.crosscult.eu](http://www.crosscult.eu)

© Copyright GVAM Guías Interactivas and other members of the CrossCult consortium 2016 - 2019