Introduction to Software Engineering

Immigration Course

Final Project Report

PhD Program in Computer Science: XXXVIII cycle

Authors:

Name Surname

Name Surname

Supervisors:

Martina De Sanctis

Ludovico Iovino

Prof. Patrizio Pelliccione

December 15, 2022

Contents

Chapter 1

Project Specification: RASTA

1.1 Project description

Context and challenges. The project RASTA (*Augmented Reality and Automated Story-Telling for the valorisation of Cultural Heritage and Tourist Itineraries*) is an ongoing national Italian project involving different universities and companies, including GSSI. Its goal concerns valorising Cultural Heritage and Tourist Itineraries by exploiting new technologies and providing new methodologies supporting the valorisation of the pilots; RASTA considers specific *project's pilots* located in different places in Italy. In particular, GSSI is responsible for the enhancement of **Tourist Itineraries in the Abruzzo National Park**.

The Abruzzo, Lazio, and Molise National Park[1] is famous in Italy and worldwide as a model for nature conservation and environmental protection. Visits to the park are controlled and selected, and there are many achievements to continue to make this park ever more modern and efficient. The Park is a complex reality, which imposes equally complex and virtuous management practices and governance tools, both from an administrative, political and socio-economic point of view, and from an environmental point of view. The park has an extensive ***network of trails*** that covers the entire territory, and a rich and varied historical and natural heritage, ranging from the Neolithic to the Italic and Roman periods, from the Middle Ages to the Renaissance. On its territory there are archaeological sites, castles, fortified villages, churches abbeys, hermitages, and mills from every era. The enhancement of ***points of interest (POIs)*** is a crucial issue for park authorities and often the technologies used tend not to enrich the information with external sources and/or make the use attractive for tourism, to the detriment of the territory and development. The tools that are typically used for this purpose are simple paper display boards with text descriptions. The problems related to this data representation are manifold:

- The usability of the data is limited since the embedded textual representations do not allow interaction and are of little use;
- They require the installation of third-party applications;
- They do not take user profiling into account when planning routes;
- POIs are often not uniformly categorised;
- The use and consultation of routes provide no feedback or usage data.

The ***geolocalisation***, ***classification***, ***and enhancement of POIs*** is a central theme for parks, since tourist attractiveness is a key element. It follows that the usability of data is an important feature in this domain. The use of appropriate technologies will enable a greater and more diversified tourist presence both in the main site and in the other linked sites, thanks to the knowledge of other territorial realities and their cultural value.

Project purpose. The project aims to define a framework for the valorisation of the Park's territory and the presentation of routes for visiting and consulting data with semantic correlation

[1] http://www.parcoabruzzo.it/

and interoperability among sources. The composition of tourist routes must be prepared considering user profiling given that POIs, routes with different completion difficulties, geolocation and interests are only some of the variables to be considered in this context. To this aim, the framework will rely on a ***recommendation system*** that will be employed in the park and that will consider variables and data from ***sensors*** installed in park areas, user profiles, and other data sources.

The ***analysis of tourist flows*** is one of the topics in which park authorities are interested and which are a consequence of the use of tools such as technological instruments for the composition of customised tourist routes. In situations where data is not provided by the use of available tools, it is in any case important to be able to obtain information from tourist transit. Therefore, the project proposes the identification of tourist flows by means of three tools:

1. tracking through the use of route planning tools,
2. automatic identification through phone detection, and
3. user identification through NFC, RFID or QRcode.

The main features / components of the framework can be categorised as follows:

1. The framework should include a Content Management System (CMS) plus a Semantic Metadata Repository (SMR) for the management of points of interest, travel planner, reporting and sensor dashboard to support other activities.
2. A Travel Planner devoted to the planning of trails and paths in the Park. An existing open travel planner can be selected among those available (e.g., Open Trip Planner) and *customized* for the needed purposes, such that it will support the provision of tourist itineraries in the park, by also exploiting data coming from the installed sensors, information about POIs, weather conditions, users preferences, etc.
3. Smartphone detection sensors can be placed at POIs, "cellular and hands-free mobile phone detection" to analyse flows of people and presence to be analysed also in the future.
4. Targeted management and flow analysis: by means of sensors placed in POIs it is possible to correlate meteorological data with user profiling in order to provide useful information.
5. Integration of an automatic POIs categorisation system.
6. A front-end system for the park, for the promotion of the entities managed by the system, exploiting the heterogeneous data sources. We envisage the involvement of a dedicated staff who should integrate content in order to make the portal minimally countable and therefore automatically enriched with information from external sources, e.g., by mentioning a path frequented by the Marsican brown bear, a link is automatically generated to Wikipedia that will give interesting details on this species, without having to employ staff to detail all the contents.

The project assignment consists in eliciting the requirements of the proposed framework and designing its possible software architecture. The details on the assignment follow in the next sections.

1.2 Project Objectives

The assigned project has several objectives:

- Allow you to discover, get familiar and practice with real ongoing research projects;
- Allow you practising teamwork, through brainstorming, discussions, problem analysis, solution identification, report writing;
- Allow you to practice and deeply learn what was presented in the lectures;
- Allow you to practice with the different tools that were introduced in class (plus other tools suggested through this document), to better understand how they differently support the requirements elicitation and software architecture definition.

1.3 Technologies

In this section we report the list of technologies we have seen in class that can be used for the project. As discussed, other equivalent technologies exist, and some of them are suggested through this document. However, *at least* 2 of the listed technologies must be used for the project and all the decisions about the performed selection should be discussed.

- **Context and Use Case Diagrams**
  - draw.io - Available at https://app.diagrams.net/
  - Creately - available at https://creately.com/
- **Goal Models**
  - iStar 2.0
    - Documentation at https://arxiv.org/abs/1605.07767
    - piStar Tool at https://www.cin.ufpe.br/~jhcp/pistar/
    - i^* Wiki at http://istarwiki.org/tiki-view_articles.php
    - iStar 2.0 Language Guide at https://sites.google.com/site/istarlanguage/home
  - Creative Leaf - Graphical Web-based goal modeling tool at http://creativeleaf.portal.chalmers.se/
- **Requirements Elicitation**
  - FRET: Formal Requirements Elicitation Tool
    - FRET tool at https://software.nasa.gov/software/ARC-18066-1
    - Repository at https://github.com/NASA-SW-VnV/fret
    - Installation instruction at https://github.com/NASA-SW-VnV/fret/blob/master/fret-electron/docs/_media/installingFRET/installationInstructions.md
    - User Guide at https://github.com/NASA-SW-VnV/fret/blob/v2.0/fret-electron/docs/_media/user-interface/writingReqs.md
- **Sofware Architecture**
  - draw.io - Available at https://app.diagrams.net/
  - Creately - available at https://creately.com/
  - Genmymodel - available at https://www.genmymodel.com/

1.4 Workflow, deadlines and submission process

Workflow. Each group, made by two members, will be assigned a project equal to the one of another group. Thus, the project workflow envisages two phases:

- **First phase**: during this phase, each group will work *autonomously and independently* from the other groups, by following the assignment given in this document and writing the project report. A specific deadline is established for the first phase of the project (see next paragraph). It will follow an *intermediate project meeting* with us and it will be a prerequisite for the second phase. During the intermediate project meeting, each group will give a presentation to summarise its project outcome.
- **Second phase**: during this phase, groups with the same project will meet (after the intermediate project meeting), discuss the outcomes of the two projects and work on their transformation and merging in a unique final outcome, for which a unique final report will be delivered. The final report will focus on the description of the final solution proposed for the assigned project. For the final report of the second phase, the rationale behind all the undertaken decisions must be discussed in Section 3.3, by referring, where necessary, to the alternative solutions proposed in the two original documents and explaining the rationale for the final solution.

Deadlines. Deadlines for the two phases of the project:

- **First phase**: the report for the first project's phase (one report per group) is due on the 08/01/2023
- The **intermediate project meeting**: will be held on 12/01/2023
- **Second phase**: the report for the second project's phase (one report each 2 groups) is due on the 31/01/2023

Submission process. The project should be submitted via email to: ludovico.iovino@gssi.it, martina.desanctis@gssi.it and patrizio.pelliccione@gssi.it.

Chapter 2

Teams Organization and Reporting

The entire project, from requirements specification to software architecture design, will be realized separately by two teams, during the first phase, and by the two teams in collaboration during the second phase.

We expect, for both project's phases, that the team members will strongly collaborate, organize the overall work, and equally split responsibilities among them.

For the *second phase*, the 2 teams should strongly collaborate each other, also according to a communication and collaboration plan identified as initial activity of the second project's phase. Section 2.2 should describe the plan and briefly report the meetings the two teams have for accomplishing the project by using the provided template table.

2.1 Team organization

The team has to carefully read and understand the project's specification to have an overall idea about the proposed framework. Then, it has to identify the framework functionalities by applying the elicitation of the requirements. Stakeholders (e.g., actors) of the system must be identified. Both *functional* and *non functional* requirements must be defined, starting from the *basic* requirements and further including the *creative* requirements, as seen in class. Data-driven requirements must also be considered. The team has to provide the data structure they envisage to be used in the framework, for complex data type. Actors *goals* must also be elicited.

Lastly, the team has to design one possible architecture of the system. To this aim, it has to consider the possibility of using one of the architectural styles we have seen in class, or a combination of them, or none of them. Each design decision must be motivated and discussed.

For all the assigned tasks, the technologies listed in section 1.3 should be considered. This described process is the same for both the first and second project's phases, with the only differences that the first phase has the project specification as input and two separate reports as output (one per group), while the second phase has the two outcomes of the first phase as input and a unique solution and report as outcome.

Team Responsibilities:

1. *System's requirements* analysis and definition (a Software Requirements Specification (SRS) template can be used[1]). A use case diagram can also be used to list the functionalities, if needed. Context diagram(s) can be used to show the inputs and outputs to and from the system's entity. However, despite the used language for eliciting requirements and goals (including the graphical ones), requirements should always be reported also as complete sentences and numbered. You can also have sub-requirements.

2. *Data structure* for the complex used data shared among actors. For instance, if you have a data of type *Request* this will be probably made by multiple attribute, such as requestId, requestType, date, etc.

3. System's software architecture *design* and *description*. The team can decide which diagram(s) / model(s) to use (e.g., UML component diagram, more abstract components and connectors

[1] See this link for example or this template, in addition to the approaches we have seen during the course.

as we have seen in class, etc.) and which tool among those listed in section 1.3. Any design decision must be discussed in the software architecture description.

4. System's software architecture dynamics (behaviour), by means of one of the diagrams for dynamic description we have seen in class (e.g., Labeled Transition Systems, Automata, UML State Charts, Sequence Diagrams, Activity Diagrams, etc.) **[Optional]**

2.2 Project register

In this section you are supposed to describe how you plan to work during the second project's phase. Thus, please report only inter-teams meetings. This includes also how the different teams are supposed to synchronize and coordinate. This includes a report of the various meetings, which should be documented by using the template of the table in the following.

Meeting date and duration	*Date*: dd/mm/yy *Duration*:
Participants	
Purpose of the meeting	
Problems/Impediments	
Risks, including their impacts and mitigation strategies	*Risk*: *Impact*:
Main decisions taken	

Chapter 3

Final Report

For both the first and second phases, report here only the final version of your project, although you will go through multiple iterations. We expect that the entire project is synchronized. For the next two sections the template is up to you, according to your results and the way you think is better to present them.

3.1 Requirements Specification

3.2 Software Architecture

3.3 Rationale leading to the final solution

This section is only for the final report, not for the intermediate one, where we expect the rationale to be discussed through Sections 3.1 and 3.2.

.

Chapter 4

Conclusion

This section should be written collaboratively and should summarize the project overall design, findings and future work. Please highlight what you learned from this project and, as a retrospective analysis, explain what you would do differently if re-starting your project from scratch.

Bibliography