Journal of Universal Computer Science, vol. 19, no. 12 (2013), 1792-1807
submitted: 31/8/12, accepted: 31/1/13, appeared: 28/6/13 © J.UCS

A Framework to Develop Web Applications Based on RFID Panels

Pedro G. Villanueva, Ricardo Tesoriero, José A. Gallud
(University of Castilla-La Mancha, Campus Universitario de Albacete S/N
02071, Albacete, Spain
{pedro.gonzalez, ricardo.tesoriero, jose.gallud}@uclm.es)

Abdulrahman H. Altalhi
(King Abdulaziz University (KAU), Jeddah, Saudi Arabia
ahaltalhi@kau.edu.sa)

Abstract: The pervasiveness of the Web, and the emergence of new technologies such as the RFID, allow users to interact with the physical environment. This fact has caused the conception of new types of applications, for instance, the Web applications based on RFID panels that allow users to retrieve and store resources from and to the Web through RFID panels. The development of these applications is defined by three main concepts: the Tool that performs domain model functionality, the Command that represents an action to be performed by the tool, and finally, the Resource representing the result of the command on the tool. Thus, this work presents a framework that enables developers to model Web applications based on RFID panels in terms of these concepts. The paper also exposes how to extend the framework to build four applications that belong to absolutely different domains.

Keywords: Framework Development, RFID, Models Development, Web Applications, Human-Computer Interaction (HCI)
Categories: D.2.11, H.1.2, I.2.5

1 Introduction

Nowadays, software solutions based on mobile technology (i.e. smart phones, PDAs, etc.) are becoming very popular in the society. It is due to the spreading of communications networks (Wi-Fi and GPRS, 3G, etc.) that provides users with connectivity almost everywhere. Thus, users are able to access the Web through their mobile devices.

Mobile technology provides to users with the ability to take photographs; write ideas, record out thoughts on voice and video, etc. Besides, it is also capable of sensing the environment through novel technologies, for instance accelerometers, cameras, RFID, IRdA, light sensors, touch panels, microphones, etc.

Up to now, informative panels offer static information about the environment. This information is usually disorganized and it is difficult to understand because of several factors, such as limited physical space or being obsolete. Besides, these panels are passive because they are not able to interact with the readers. As consequence, people tend to lose the interest on the panel information.

Therefore, a new set of applications have exploited the capabilities of mobile devices to improve the information panels by providing them with dynamic information through new interaction techniques. Thus, an interactive dialog is achieved between the user and the environment or among users. Some examples of Web application based on RFID panels are: ecologic information panels, electronic museum guides, m-learning systems and GIS interfaces that will be discussed further.

Thus, the goal of this work is the definition of a framework that implements common functionality to support the development of this kind of applications.

The article is organized as follows: Section 2 exposes related work describing some scenarios that will be implemented by the proposed framework. Then, Section 3 exposes the conceptual view of Web applications based on RFID panels, as well as the system architecture. Further, Section 4 describes the proposed framework and Section 5 shows how to model applications described on Section 2 using the framework. Finally, Section 6 exposes conclusions and future work.

2 Related Work

This section introduces a set of most relevant related work in this field. These systems show how diverse is the domain application of interactive panels.

For instance, they can be used to: encourage people involvement in ecological issues (Interactive ECO Panels), to provide information to museum visitors (Museum guides), to implement novel learning activities (Interactive Learning Panels), to interact easily with GIS (Geo-referenced Panels), and so on.

2.1 Interactive ECO Panels

The goal of this application is to encourage the society into ecologic issues, such as city garbage recollection, collective decisions about green zone locations, illumination and transport system suggestions. Interactive ECO Panels are location-aware applications that provide citizens with the ability to express themselves about daily issues [Tesoriero, 08-a].

This work describes a solution based on the successful integration of social software and location-aware applications offering users a collaborative application that allows them to both, retrieve and share comments, about topics related to environmental issues just by performing natural and intuitive gestures.

Thus, this interaction allows the participation of the society serving as a tool to in-crease the social responsibility for creating a sustainable environment. Besides, Inter-active ECO Panels reprocesses the informative panels that are present in our squares, streets, bus stations, etc., turning them into interactive panels based on RFID.

These panels are based on the Web 2.0 technology that is the foundation of the social software concept. They provide users with collaboration tools through the Web. In fact, panels will take social software from the Internet and will put it into the streets.

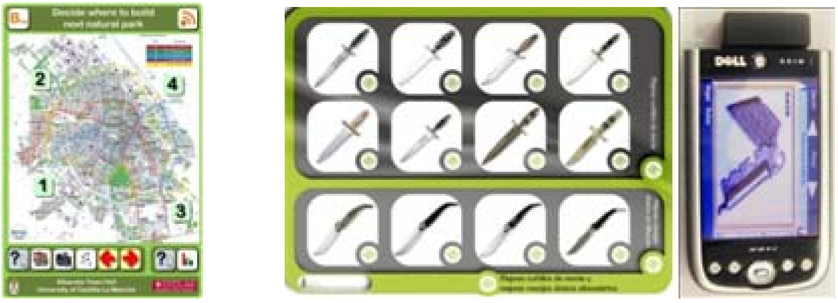
Interactive ECO Panels provide users with multimedia content such as, audio, text, video, etc. Besides, it provides citizens with the ability to express their thoughts using the Vote, the Poll, or the Blog system. It also allows users to get news about important events through RSS feeds. Thus, Interactive ECO Panels enable citizens to both, retrieve and share information about ecologic issues.

A concrete example may be the decision by a town council about where to locate the next green zone among the expansion areas of the city. A possible panel is shown in Figure 1a. This panel offers 4 zones (identified by the numbers 1, 2, 3 and 4), which show candidate locations for the green zone in the city. It also offers a set of commands that allows users to interact with different multimedia resources and social software tools provided by the system.

2.2 Museum Guides

Museum guides allow visitors to retrieve information about objects that are part of a cultural environment (museum, art exhibitions, old buildings, etc.) On the one hand, some guides are designed to work in indoor environments [Butz, 00; Ciavarella, 03; Ciavarella, 04; Paternò, 07; Tesoriero, 08-b]. On the other hand, others, based on GPS technology, are designed to operate in outdoor [Long, 96] or mixed environments [Davies, 98; Yang, 99]. Most of these applications provide information to visitors by the identification of marks or via the location system.

Figure 1b shows a prototype of a knife showcase where users are able to retrieve in-formation from knives, groups of them and even from the showcase itself. Thus, the system provides multimedia information about objects, such as audios, videos, texts and so on. The implementation of the system is described in detail on [Tesoriero, 08-b]. As in the Interactive ECO Panel, the system also allows users to exchange experiences and comments through a Blog or express interests using a Vote or Pool tool. The use of interactive panels in museums is based on marks (see the dashed rectangle on Figure 1b). In order to retrieve information from an object visitors have to approach the device to the mark. Thus, users express the intention to obtain information from art pieces, masterpieces, sculptures, antiques, rooms or even buildings.



(a) (b)

Figure 1: Interactive ECO Panel (left) and museum guide (right)

2.3 Interactive Learning Panels

The application of computing technologies on education starts almost simultaneously to the beginning of computing science itself improving education methods, either formal or informal, and educational mediums, such as online, in-situ or blend-ed. Substantial achievements in this field include: the introduction of new multimedia infrastructure into schools and colleges, such as personal computers connected to the Internet, and the development of new educational software that improve student learning through imaginative software.

Combs in [Coombs, 74] defines informal learning as "the spontaneous, unstructured learning that goes on daily in the home and neighbourhood, behind the school and on the play field, in the workplace, library and museum, and through the various mass media, informal learning is by far the most prevalent from of adult learning".

Thus, mobile technologies become a powerful tool to support contextual life-long learning, by being highly portable, individual, unobtrusive, adaptable to the context of learning, the learner's evolving skills, and knowledge [Sharples, 00].

Some characteristics of this type of applications described in [Chen, 02] include: the urgency of learning need, the initiative of knowledge acquisition, the mobility of learning setting, the interactivity of the learning process, the "Situatedness" of instructional activities and the integration of instructional content.

At first, informal learning was bounded on formal school and non-formal learning in courses [Coombs, 74]. From the point of view, of the organizational informal learning, the processes are not formally organized [Watkins, 90]. A broader approach is that of Livingstone that is geared towards learning, self-directed and self-emphasizes of the process, by the student [Livingstone, 02].

Thus, the work presented in [Bradburne, 01] indicates that there are two types of exhibits, static and dynamic. For instance, [Bradburne, 01] makes a distinction between exhibit (broadcasts facts) and informal learning. Some examples of dynamic exhibitions are [Sparacino, 02; Semper, 02].

On [Tesoriero, 09] a prototype developed in our research laboratory treats the idea of teaching the Geography subject at the school, the library or any other learning environment. It presents a connection between the student and the physical material provided by this subject [Fardoun, 10].

In this case, it is a panel, which goal is the identification of regions in a country and the flags of each region into a political map.

This panel is presented in such a simple and intuitive way students can understand and interact with it without the help of any tutor (see Fig 3a).

On the other hand, the PDA asks for personal information and operation modes. Students are free to choose between two operation modes: the exam or a practice. The difference between these two modes lies on whether results are sent to the professor or not. To select the option, the student has just to approach the PDA to the mode icon on the panel.

Once the operation mode is selected, a question is displayed. The question may be the identification of a region in the map or a flag corresponding to a defined region on the PDA screen. For instance: "Select a state flag and then locate it into the map". At the right bottom of the screen the students observes the number of the question he/she has answered and the total number of them. To select the answer, the student has just to approach the PDA to the flag or region image depicted in the panel. Once a student fulfilled a questionnaire, the results are displayed on the PDA screen.

To carry out these tasks, at the bottom of the panel appears the control panel that shows the following icons: the operation modes, the go to previous question, the go to next question, the accept answer, the cancel answer, the exit application without sending results and the exit application sending result to student and professor (if in exam mode) emails. Besides, this application supports the playback of multimedia content (audio, text, video, etc.).

2.4 Geo-referenced Panels

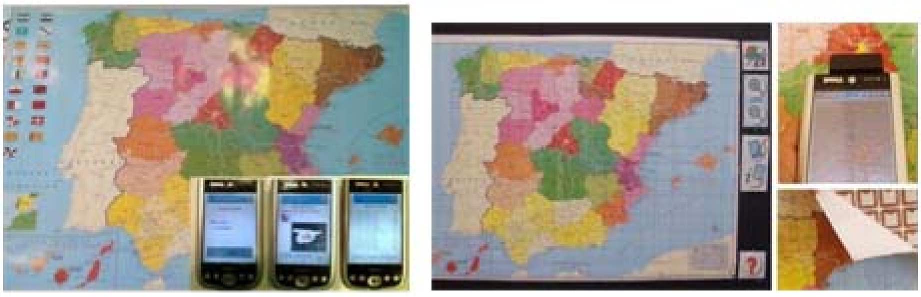
Geo-referenced panels expose a solution to extends and manipulate physical maps, which are present in our daily life. Thought these panels the system is able to interpret a position selected by the user into the map and turn it into a geographical position. Thus, users are able to store and retrieve information related to this location.

These panels can be deployed on town councils, public administrations, and cultural or educational institutions to share information.

In tour scenario, tourists are able to share experiences about places through the map. For instance, users are able to select a region on the map and associate comments to that region. Other users then read these comments just by querying these comments.

Other scenarios include the use of these panels in bust stops, railway and metro stations and so on. For instance, they may provide information about traffic in the public transport network.

An implementation of this system provides the ability to query information from a zone in the map. A typical map is shown on Figure 3b that is located into the entrance of a public building.



(a) (b)

Figure 2: Interactive learning panels (left) and Geo-referenced panels (right)

The map is divided into two main zones: the map itself and the command region. While the map zone allows user to identify a geographic zone in the map (UTM coordinates), the command region allows user to perform action related to the selected zone in the map.

Some actions that are available in the system are the zoom in, the zoom out, the road map view, and serial map view, add comment, retrieve information and display help.

Thus, to add a comment the user approaches the device to a zone in order to select it, then the user approaches the device to the add comment action. Afterwards, he/she writes the comment to be associated to the map zone. In order to retrieve these comments users have to select the zone and then executing the command retrieve information.

Thus, the system associates information to a location in the map extending the functionalities of traditional maps through natural gestures.

A detailed description of the prototype developed in our research laboratory is exposed on [Romero, 09].

3 Web Applications Based on RFID Panels

In this section we study the conceptual foundations of Web applications based on RFID panels. Thus, these applications can be defined in terms of have three basic concepts: tool, command and resource. The relationship between these entities is defined by the Command design pattern [Gamma, 95] where the Tool plays the role of Receiver and the Command plays the role of ConcreteCommand.

- Tool: The tool concept represents a domain model component that defined a set of commands to be executed by system users. Thus, the tool stores the state of a transient operation on the system. Some examples of tools defined by related work are: the blog, the resource player, the vote or pool, the questionnaire and so on.
- Command: The command concept represents an action performed on a tool. For instance, Next, Previous, Zoom, Retrieve information, Post message, Vote, etc.
- Resource: The resource concept is the result of applying a command to a tool. For instance, the URL that points to the resource to be displayed by the client.

As we have stated before, the tool concept represents a domain model component that stores the state of transient operations in the system. Therefore, in the ECO Panel scenario the following concepts are considered tools because: the Blog stores the set of messages posted by citizens, the ResourceView stores a list of resources (photos, audios, texts or videos) to be visualized by the citizens, the Vote keeps track of the amount of votes on each proposal and the RSS stores the email addresses of citizens interested in the subject of the panel.

Thus, the following operations were defined in the ECO Panel scenario according to the tool they affect: Next, Previous and SetRegion are related to the ResourceView, PostBlog and ShowBlog are related to the Blog tool, DoVote and ShowVote are related to the Vote tool, and finally, the Subscribe/Unsubscribe command is related to the RSS tool.

Finally, the result of applying a command to a tool is a resource. In the ECO Panel scenario, such resource may be an audio, a text, a post, an email, video file, etc.

Analogously, the same criterion was adopted to define the tools and commands in the rest of scenarios.

3.1 Concept Mapping

Table 1 shows the mapping between the concepts described on scenarios exposed in Section 2 and the tool, command and resource concepts.

	ECO Panels	**Museum**	**ILP**	**Geo-referenced**
Tools	Blog, ResourceView, Vote, RSS	ResourceView MapView ShowcaseView	Questionnaire OperationMode	MapViewer
Commands	SelRegion, Next, Previous PostBlog, ShowBlog, DoVote, ShowVote, Suscribe / Un-suscribe	SelPiece SelRoom SelShowcase	PrevQuestion NextQuestion AcceptAnswer CancelAnswer ExitWResult ExitWOResult SelRegion SelFlag	SelRegion ZoomIn ZoomOut RoadMode AerialMode PostComment ShowComments ShowHelp
Resources	Audio, Video Text, Image Posts, Email	Audio, Video Text, Image	URL	Map Comments

Table 2: Concept mapping on related work scenarios

3.2 System Architecture

The system communication is supported by a client-server application. This communication is based on a SOA architecture that allows users to store and retrieve information.

Thus, when the RFID tag, identification (ID), is read; by the RFID reader, in the client device, the ID is sent to the server to be processed via a web service call. The server then turns this ID into a command or tool accordingly. Thus, a command is executed on a tool resulting in a resource that is sent back to the client.

In order to carry out this process, we have organized the application in the following packages: client, server, framework, framework extension and commons; as depicted on Figure 3.

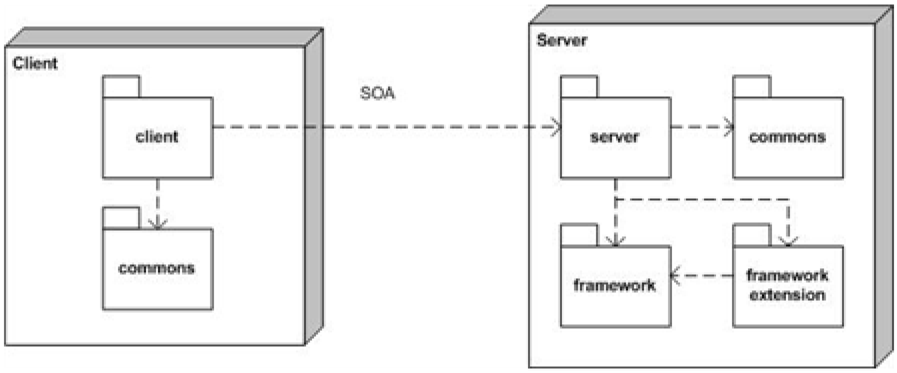


Figure 3: Package diagram

The client packages contains those classes that are part of the client application, the server package contains those classes that are part of the server application, the commons package contains those classes that are part of both the client and the server applications, and finally the framework and framework extension packages that are also part of the server application.

3.2.1 The Client Application

The client application conceptual model is depicted on Figure 4. These classes are part of the client package.

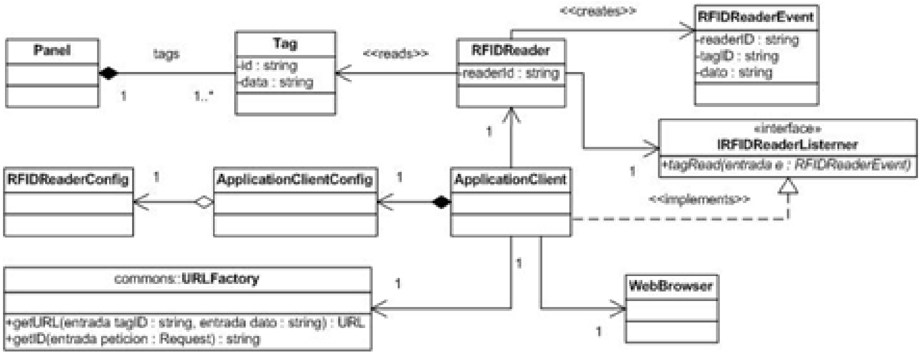


Figure 4: Client application conceptual model

Conceptually a Panel is composed by a set of Tags that are read by a RFIDReader. When the reader reads a tag it creates RFIDReaderEvent that is set to a set of IRFIDReaderListeners (implementing the Observer design pattern [Romero, 09]).

As the ApplicationClient implements the IRFIDReaderListener interface, it receives a RFIDReaderEvent when a tag is read. Thus, The ApplicationClient sends the getURL message to the URLFactory jointly with the tag id and data read by the reader. The getURL method returns a URL that represents an action on the server. The URL is built from the data and id information (http://data?id=id). Then this URL is sent to the server via the WebBrowser showing the resource that is the result of the operation.

Besides, the ApplicationClientConfig that holds the RFID reader configuration represented by the RFIDReaderConfig configures the ApplicationClient.

3.2.2 The Server Application

On the server side, the application is defined by two main parts: the server infrastructure that defines basic services to be used by the framework to carry out its job, and the server framework itself that is described in Section 4.

The server infrastructure conceptual model is depicted on Figure 5. These classes are part of the client package.

When the server application receives a request from the network, the WebHandler creates the Request and Response objects and sends them as parameter to the ClientProxy through the process method. The ClientProxy is the representation of a client at the server side of the application keeping the client state on the server.

Thus, the process method retrieves the tag ID from the Request object through the getID method defined by the URLFactory. Then, it delegates to the framework the responsibility of processing the request. The result of this process is then sent back to the client by the Response object.

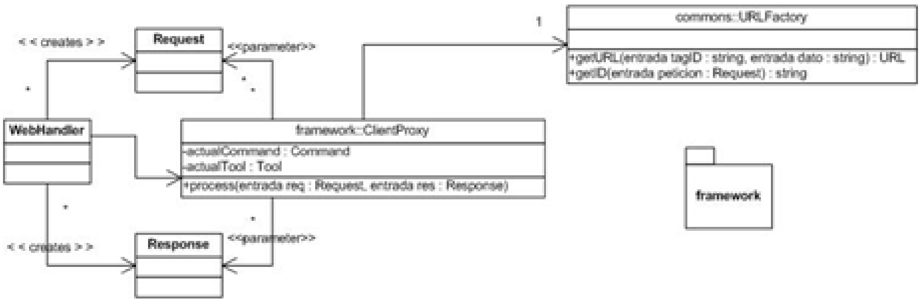


Figure 5: Server infrastructure conceptual model

4 Framework

This section exposes main characteristics of the proposed framework describing the core of this work.

The framework is based on the combination of two design patterns, the Proxy and the Command [Gamma, 95] design patterns. Whereas we have applied the Proxy design pattern to represent the client state on the server side of the application, the Command design pattern was used to model the operations performed by the user through the physical interface.

The development process of Web applications based RFID panels is defined in two simple steps:

1. The application definition
2. The application configuration

While the application definition is based on framework subclasses definition that implement the logic of the application (i.e. tools, commands, etc.), the application configuration defines the mapping between physical information (RFID tag information) and the application tools and commands.

4.1 Application Definition

This framework is based on three main classes to be sub-classified by developers in order to build Web applications based on RFID panels. These classes are the ClientProxy, Command and Tool. The framework is depicted jointly with the extensions implementing the applications exposed on related work (Section 2)

The ClientProxy is the entry point to the framework functionality. The process method receives two parameters, Request and Response. These parameters represent the Web Request and Response objects. This method retrieves the RFID from the Request object through the getID method defined in the URLFactory class. The RFID is then turned into a command, a tool or both of them. Two hush tables that contain tools and commands indexed by RFIDs carry out this process. They are loaded in the loadToolsAndCommands method from an XML file following the schema de-fined in the Figure 7.

Once tool and command (represented by the currentTool and currentCommand in-stance variables) were updated, the command is executed on the tool via the executeOn method. Thus, the desired command represented by a Command subclass performs the method representing the action to be performed on the tool represented by a Tool subclass. The result of this process is a Resource that represents the result of the operation. To illustrate the framework functionality we have depicted sequence diagram of the museum application sample representing how information is processed through the framework on Figure 6.

First step to develop a Web applications based on RFID panels using this framework is the definition of the application root tool. This tool is a subclass of the Tool class that represents the root of all tools in the system. It will define, at least, one method per command to be defined in the application.

Then, the set of commands that will be supported by the tools of the application will be defined.

All these classes will be subclasses of a root command class that subsequently will be subclass of the Command parameterized class. The class parameter will be the root tool class we have defined in the previous step. Subclasses of command root class will override the executeOn method by calling the method on the tool that represents the command.

Once methods and tools were defined, we have to define the application proxy class. This class is a subclass of the ClientProxy parameterized class that takes the root command and root tool classes as parameters. The only method that should be redefined is the loadToolsAndCommands method to load classes from XML mapping files. Thus, the application definition phase of the development process concludes.

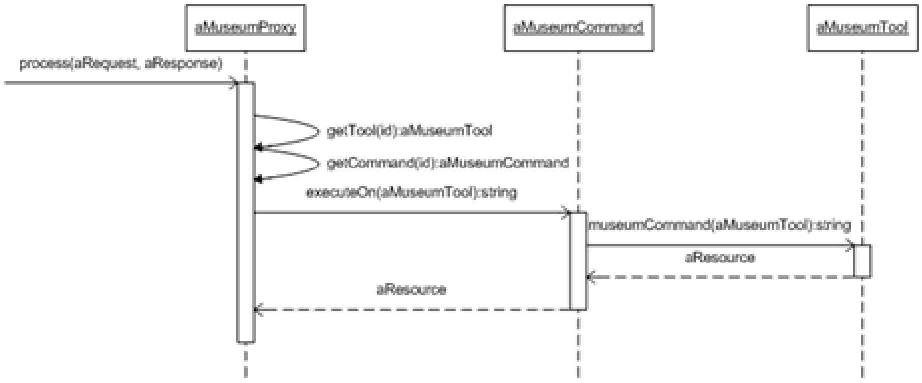
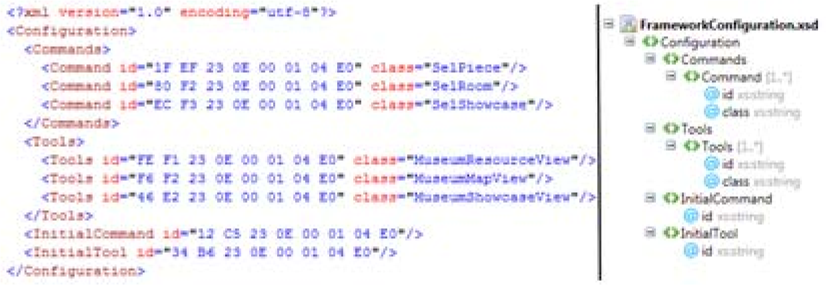

aMuseumProxy
aMuseumCommand
aMuseumTool
process(aRequest, aResponse)
getTool(id):aMuseumTool
getCommand(id):aMuseumCommand
executeOn(aMuseumTool):string
museumCommand(aMuseumTool):string
aResource
aResource

Figure 6: Sequence diagram exposing the process method

4.2 Application Configuration

The application configuration phase is far simpler than the application definition. It consists on assigning RFIDs to a command, tool or eventually both of them. This assignation is characterized in the XML file defined on Figure 7.



```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
  <Commands>
    <Command id="1F EF 23 0E 00 01 04 E0" class="SelPiece"/>
    <Command id="80 F2 23 0E 00 01 04 E0" class="SelRoom"/>
    <Command id="EC F3 23 0E 00 01 04 E0" class="SelShowcase"/>
  </Commands>
  <Tools>
    <Tools id="FE F1 23 0E 00 01 04 E0" class="MuseumResourceView"/>
    <Tools id="F6 F2 23 0E 00 01 04 E0" class="MuseumMapView"/>
    <Tools id="46 E2 23 0E 00 01 04 E0" class="MuseumShowcaseView"/>
  </Tools>
  <InitialCommand id="12 C5 23 0E 00 01 04 E0"/>
  <InitialTool id="34 B4 23 0E 00 01 04 E0"/>
</Configuration>
```

Figure 7: XML schema of framework configuration

Thus, each application will define the relationship between the physical environment it is immersed and the command and tools that are part of the application domain logic.

5 Supporting Web Applications Based on RFID Panels

This section exposes the implementations of the related work exposed in Section 2 using the proposed framework. The implementation is based on a set of extensions of the framework. Each extension is depicted on Figure 8 surrounded by a dashed rectangle. Next subsections will expose these extensions in detail.

5.1 Extension for Interactive ECO Panel

Following the procedure exposed on Section 4, we have defined root tool represented by the ECOTool abstract class extending the Tool class. This class implements the following commands according to Section 2.1.

Then, we have defined the ECOBlog, ECOResourceView, ECOVote and ECORSS classes extending the ECOTool class. These subclasses implement the domain model logic of the application.

Afterwards, before defining each command, we have to define the root command. In this particular case, the ECOCommand class that extends the Command class defines it. The ECOCommand class is parameterized by the ECOTool class, which defines the interface to execute tool actions. One ECOCommand subclass should be defined for each action to be performed by the tool. Thus, the executeOn method should be redefined to invoke the right tool method.

Finally we have to define the extension of the ClientProxy abstract class. For Interactive ECO Panels we defined the ECOProxy class parameterized with the ECOTool and ECOCommand root classes.

The extension classes for the Interactive ECO Panel application are defined in the second dashed rectangle on Figure 8.

5.2 Extension for Museum Guides

In this case, to define the extension for the museum guides, we have defined root tool represented by the MuseumTool abstract class extending the Tool class. This class implements the following commands according to Section 2.2.

Then, we have defined the MuseumResourceView, MuseumMapView and the MuseumShowcaseView classes extending the MuseumTool class. These subclasses implement the domain model logic of the application.

Afterwards, before defining each command, we have to define the root command. In this particular case, the MuseumCommand class that extends the Command class defines it. The MuseumCommand class is parameterized by the MuseumTool class, which that defines the interface to execute tool actions. One MuseumCommand subclass should be defined for each action to be performed by the tool. Thus, the executeOn method should be redefined to invoke the right tool method.

Finally we have to define the extension of the ClientProxy abstract class. For Museum Panels we defined the MuseumProxy class parameterized with the MuseumTool and MuseumCommand root classes.

The extension classes for the Museum Panel application are defined in the first dashed rectangle on Figure 8.

5.3 Extension for Interactive Learning Panels

To define the extension for the interactive learning panels, we have defined root tool represented by the LearningTool abstract class extending the Tool class. This class implements the following commands according to Section 2.3.

Then, we have defined the Questionnaire and the OperationMode classes extending the LearningTool class. These subclasses implement the domain model logic of the application.

Afterwards, before defining each command, we have to define the root command. In this particular case, the LearningCommand class that extends the Command class defines it. The LearningCommand class is parameterized by the LearningTool class, which that defines the interface to execute tool actions. One LearningCommand subclass should be defined for each action to be performed by the tool. Thus, the executeOn method should be redefined to invoke the right tool method.

Finally we have to define the extension of the ClientProxy abstract class. For Interactive Learning Panels we defined the LearningProxy class parameterized with the LearningTool and LearningCommand root classes.

The extension classes for the Interactive Learning Panel application are defined in the fourth dashed rectangle in Figure 8.

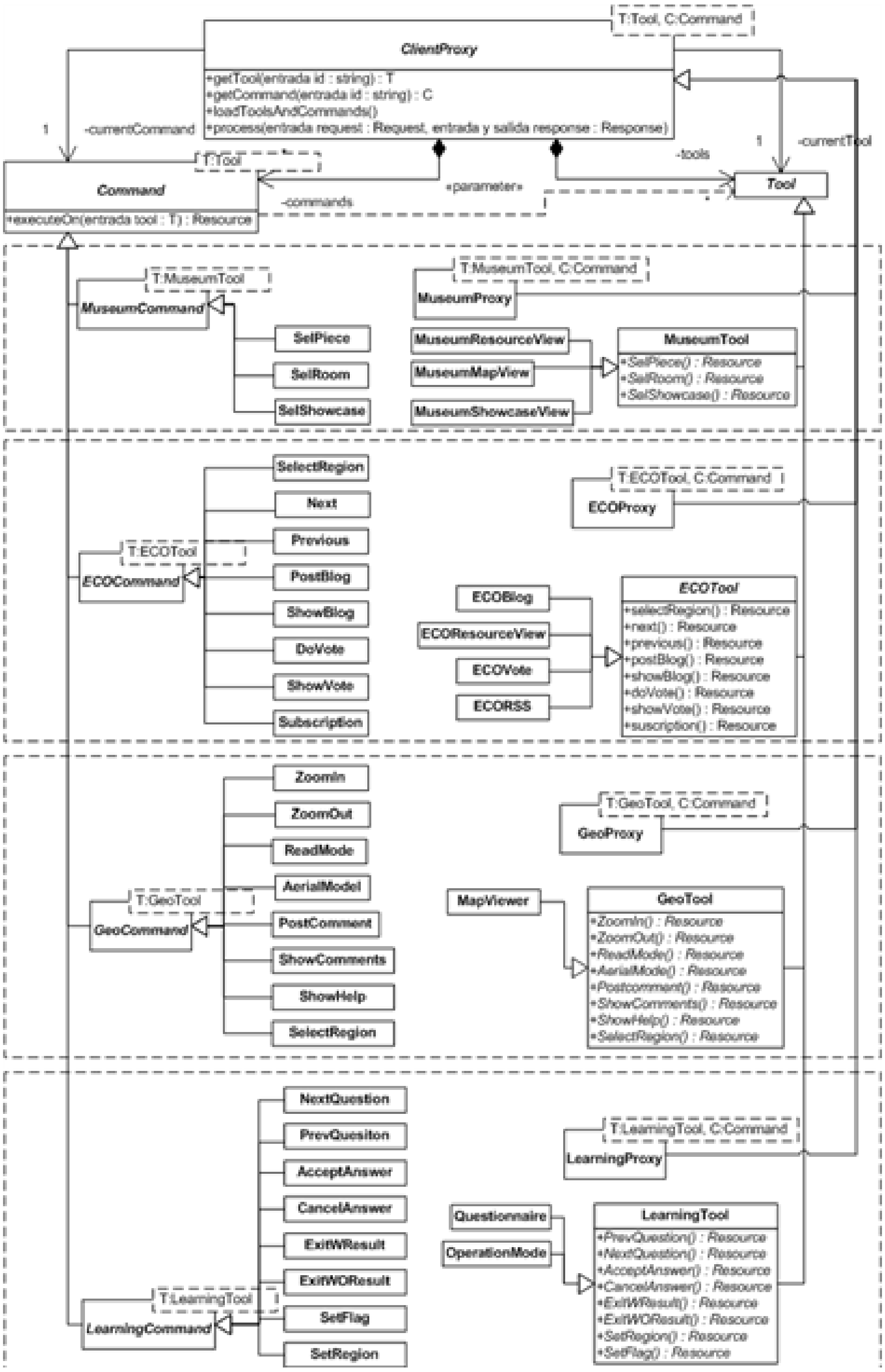


Figure 8: Class diagram of the framework and related work extensions

5.4 Extension for Geo-referenced Documents

Finally, to define the extension for the geo-referenced panels, we have defined the root tool represented by the GeoTool abstract class extending the Tool class. This class implements the following commands according to Section 2.4.

Then, we have defined the MapViewer class extending the GeoTool class. This subclass implements the domain model logic of the application. Afterwards, before defining each command, we have to define the root command. In this particular case, the GeoCommand class that extends the Command class defines it. The GeoCommand class is parameterized by the GeoTool class, which defines the interface to execute tool actions. One GeoCommand subclass should be defined for each action to be performed by the tool. Thus, the executeOn method should be redefined to invoke the right tool method. Finally we have to define the extension of the ClientProxy abstract class. For geo-referenced Panels we defined the GeoProxy class parameterized with the GeoTool and GeoCommand root classes.

The extension classes for the Geo-referenced Panel application are defined in the third dashed rectangle in Figure 8.

6 Conclusions and Future Work

This work presents a framework to develop Web applications based on RFID panels that allows developers to build this type of applications in terms of three simple concepts: Tool, Command and Resource.

It employs a client-server architecture based on a SOA (Service Oriented Architecture) that allows panel user to retrieve and store information through the Internet.

This work also exposes how to model and build four applications that belong to different domains using this framework, showing the flexibility and extensibility of this software artifact.

The most relevant advantages of employing this framework are:

- The simplicity and flexibility of its design, which is based on, design patterns [Gamma, 95].
- The ability to create novel gestural interfaces for mobile devices equipped with RFID technology.
- The provision of a scalable solution to add: new tools (applications), new commands (functionality to tools) and resources to the system.
- The ability to create novel ways of extending the social software concept in the Web to physical environments
- The ability to create applications that provide internationalized information in physical spaces.
- The deployment costs are low (apart from compact flash reader) and it is a highly scalable solution

Finally, we plan to create a content management application that enables developers for customization according to the application domain. We also are extending the framework to cope with other technologies, such us IrDA, Bluetooth, BIDIs, Code bars, etc. It is due to the fact that RFID reader is actually quite expensive.

Acknowledgements

The heading of section ‘Acknowledgement’ must be 10 pt, bold, left justified, with 12pt empty space before and 6pt after. It is absolutely imperative that the references are formatted correctly, i.e. first comes the abbreviation in square brackets, then follows the second name of the author followed by abbreviation of the first name.

References

[Bradburne, 01] Bradburne, J.: A new strategic approach to the museum and its relationship to society. Museum Management and Curatorship 19(1), 75–84 (2001)

[Butz, 00] Butz, A., Baus, J., Kruger, A.: Augmenting buildings with infrared information. In Proceedings of the International Symposium on Augmented Reality ISAR 2000, IEEE Computer Society Press, 93-96 (2000).

[Chen, 02] Chen, Y.S., Kao, T.C., Sheu, J.P., Chiang, C.Y.: A Mobile Scaffolding-Aid-Based Bird-Watching Learning System. In: Milrad, M., Hoppe, H.U. (eds.) IEEE Inter-national Workshop on Wireless and Mobile Technologies in Education, pp. 15–22. IEEE Computer Society, Los Alamitos (2002)

[Ciavarella, 03] Ciavarella, C., Paternò, F.: Design Criteria for Location-aware, Indoor, PDA Applications. In: Chittaro, L. (ed.) Mobile HCI 2003. LNCS, vol. 2795, pp. 131–144. Springer, Heidelberg (2003)

[Ciavarella, 04] Ciavarella, C., Paternò, F.: The design of a handheld, location aware guide for indoor environments. Personal and Ubiquitous Computing, Vol.8 N.2, Springer Verlag, 82-91. (2004)

[Coombs, 74] Coombs, P., Ahmed, H.: Attacking rural poverty. How nonformal education can help, Baltimore (1974)

[Davies, 98] Davies, N.; Mitchell, K.; Cheverest, K. and Blair, G.: Developing a Context Sensitive Tourist Guide. First Workshop on Human Computer Interaction with Mo-bile Devices, GIST Technical Report G98-1 (1998)

[Fardoun, 10] Fardoun, H. M., Pedro G. Villanueva, Juan Enrique Garrido, Gabriel Sebastian Rivera, and Sebastian Romero Lopez. 2010. Instructional m-Learning System De-sign Based on Learners: MPrinceTool, In Proceedings of the 2010 Fifth Interna-tional Multi-conference on Computing in the Global Information Technolo-gy (ICCGI '10). IEEE Computer Society, Washington, DC, USA, 220-225. http://dx.doi.org/10.1109/ICCGI.2010.49

[Gamma, 95] Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Abstraction and Reuse in Object-Oriented Designs, Addison-Wesley Professional, ISBN 0201633612. (1995)

[Livingstone, 02] Livingstone, D.W.: Mapping the Iceberg. NALL Working Paper number 54 (2002)

[Long, 96] Long, S.: Rapid Prototyping of Mobile Context-aware Applications: The Cyberguide Case Study. 2nd ACM International Conference on Mobile Computing and Networking (MobiCom'96), November 10-12, (1996)

[Paternò, 07] Paternò, F, Santoro, C.: Exploiting mobile devices to support museum visits through multi-modal interfaces and multi-device games. WebIST 2007, Barcelona, Spain, March 3-6, (2007) [Romero, 09] Romero, S., Tesoriero, R., Villanueva, P. G., Gallud, J. A., Penichet, V. M. R.: Sistema Interactivo para la Gestión de Documentos Georeferenciados basado en RFID. Interacción 2009, X Congreso Internacional de Interacción Persona-Ordenador. Barcelona. ISBN-13:978-84-692-5005-1, (2009)

[Semper, 02] Semper, R., Spasojevic, M.: Devices and a wireless web-based network to extend the museum experience. In: Museums and the Web 2002 (2002)

[Sharples, 00] Sharples, M.: The design of personal mobile technologies for lifelong learning. Computers & Education 34, 177–193 (2000)

[Sparacino, 02] Sparacino, F.: The museum wearable: real-time sensor-driven understanding of visitors interests for personalized visually-augmented museum experiences. In: Museums and the Web 2002, Archives & Museums Informatics, Pittsburgh (2002)

[Tesoriero, 08-a] Tesoriero, R., Tebar, R., Gallud, J. A., Penichet, V. M. R.; Lozano, M. D.: Inter-active ECO-Panels: Paneles Ecológicos Interactivos basados en RFID. Proceedings IX International Congress of Person-Computer Interaction 2008. ISBN: 978-84-691-3871-7. Pages: 155-165. Editors: M. D. Lozano and J. A. Gallud. Publisher: University of Castilla-La Mancha. Albacete. Spain. (June 2008)

[Tesoriero, 08-b] Tesoriero, R, Gallud, J.A., Lozano, M., Penichet, V.M.R.: Using Active and Passive RFID Technology to Support Indoor Location-Aware Systems. IEEE Transactions on Consumer Electronics, Volume 54, Number 2 ISSN: 0098-3063, pp. 578-583, (May 2008)

[Tesoriero, 09] Tesoriero, R, Fardoun, H.M., Gallud, J.A., Lozano, M., Penichet, V.M.R.: Interac-tive Learning Panels. Proceedings of 13th International Conference on Human-Computer Interaction, Town and Country Resort & Convention Center, San Die-go, CA, USA, Lecture Notes in Computer Science, ISSN: 0302-9743, vol 5613, pp. 236-245. Springer Berlin / Heidelberg, (2009)

[Watkins, 90] Watkins, K., Marsick, V.: Informal and Incidental Learning in the Workplace (1990)

[Yang, 99] Yang, J.; Yang, W. Denecke, M.; and Waibel, A.: Smart sight: a tourist assistant system. 3rd International Symposium on Wearable Computers, San Francisco, California, 18-19 October, 1999, pp. 73-78 (1999).