

ANEXO 2

**CARTA DE AUTORIZACIÓN DE LOS AUTORES
(Licencia de uso)**

Bogotá, D.C., Julio 11 de 2013

Señores
Biblioteca Alfonso Borrero Cabal S.J.
Pontificia Universidad Javeriana
Ciudad

Los suscritos:

Daniel Felipe González Obando , con C.C. No 1013606403
_____, con C.C. No _____
_____, con C.C. No _____

En mi (nuestra) calidad de autor (es) exclusivo (s) de la obra titulada:
CreaDevManager – Un Manejador de Proyectos para CreaTools

(por favor señale con una "x" las opciones que apliquen)
Tesis doctoral Trabajo de grado Premio o distinción: Si No

cual: _____
presentado y aprobado en el año 2013 , por medio del presente escrito autorizo (autorizamos) a la Pontificia Universidad Javeriana para que, en desarrollo de la presente licencia de uso parcial, pueda ejercer sobre mi (nuestra) obra las atribuciones que se indican a continuación, teniendo en cuenta que en cualquier caso, la finalidad perseguida será facilitar, difundir y promover el aprendizaje, la enseñanza y la investigación.

En consecuencia, las atribuciones de usos temporales y parciales que por virtud de la presente licencia se autorizan a la Pontificia Universidad Javeriana, a los usuarios de la Biblioteca Alfonso Borrero Cabal S.J., así como a los usuarios de las redes, bases de datos y demás sitios web con los que la Universidad tenga perfeccionado un convenio, son:

AUTORIZO (AUTORIZAMOS)	SI	NO
1. La conservación de los ejemplares necesarios en la sala de tesis y trabajos de grado de la Biblioteca.	X	
2. La consulta física o electrónica según corresponda	X	
3. La reproducción por cualquier formato conocido o por conocer	X	
4. La comunicación pública por cualquier procedimiento o medio físico o electrónico, así como su puesta a disposición en Internet	X	
5. La inclusión en bases de datos y en sitios web sean éstos onerosos o gratuitos, existiendo con ellos previo convenio perfeccionado con la Pontificia Universidad Javeriana para efectos de satisfacer los fines previstos. En este evento, tales sitios y sus usuarios tendrán las mismas facultades que las aquí concedidas con las mismas limitaciones y condiciones	X	
6. La inclusión en la Biblioteca Digital PUJ (Sólo para la totalidad de las Tesis Doctorales y de Maestría y para aquellos trabajos de grado que hayan sido laureados o tengan mención de honor.)	X	

De acuerdo con la naturaleza del uso concedido, la presente licencia parcial se otorga a título gratuito por el máximo tiempo legal colombiano, con el propósito de que en dicho lapso mi (nuestra) obra sea explotada en las condiciones aquí estipuladas y para los fines indicados, respetando siempre la titularidad de los derechos patrimoniales y morales correspondientes, de acuerdo con los usos honrados, de manera proporcional y justificada a la finalidad perseguida, sin ánimo de lucro ni de comercialización.

De manera complementaria, garantizo (garantizamos) en mi (nuestra) calidad de estudiante (s) y por ende autor (es) exclusivo (s), que la Tesis o Trabajo de Grado en cuestión, es producto de mi (nuestra) plena autoría, de mi (nuestro) esfuerzo personal intelectual, como consecuencia de mi (nuestra) creación original particular y, por tanto, soy (somos) el (los) único (s) titular (es) de la misma. Además, aseguro (aseguramos) que no contiene citas, ni transcripciones de otras obras protegidas, por fuera de los límites autorizados por la ley, según los usos honrados, y en proporción a los fines previstos; ni tampoco contempla declaraciones difamatorias contra terceros; respetando el derecho a la imagen, intimidad, buen nombre y demás derechos constitucionales. Adicionalmente, manifiesto (manifestamos) que no se incluyeron expresiones contrarias al orden público ni a las buenas costumbres. En consecuencia, la responsabilidad directa en la elaboración, presentación, investigación y, en general, contenidos de la Tesis o Trabajo de Grado es de mí (nuestro) competencia exclusiva, eximiendo de toda responsabilidad a la Pontificia Universidad Javeriana por tales aspectos.

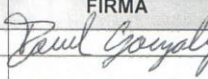
Sin perjuicio de los usos y atribuciones otorgadas en virtud de este documento, continuaré (continuaremos) conservando los correspondientes derechos patrimoniales sin modificación o restricción alguna, puesto que de acuerdo con la legislación colombiana aplicable, el presente es un acuerdo jurídico que en ningún caso conlleva la enajenación de los derechos patrimoniales derivados del régimen del Derecho de Autor.

De conformidad con lo establecido en el artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, "Los derechos morales sobre el trabajo son propiedad de los autores", los cuales son irrenunciables, imprescriptibles, inembargables e inalienables. En consecuencia, la Pontificia Universidad Javeriana está en la obligación de RESPETARLOS Y HACERLOS RESPETAR, para lo cual tomará las medidas correspondientes para garantizar su observancia.

NOTA: Información Confidencial:

Esta Tesis o Trabajo de Grado contiene información privilegiada, estratégica, secreta, confidencial y demás similar, o hace parte de una investigación que se adelanta y cuyos resultados finales no se han publicado. Si No

En caso afirmativo expresamente indicaré (indicaremos), en carta adjunta, tal situación con el fin de que se mantenga la restricción de acceso.

NOMBRE COMPLETO	No. del documento de identidad	FIRMA
Daniel Felipe González Obando	CC 1013606403	

FACULTAD: Ingeniería

PROGRAMA ACADÉMICO: Ingeniería de Sistemas

ANEXO 3
BIBLIOTECA ALFONSO BORRERO CABAL, S.J.
DESCRIPCIÓN DE LA TESIS O DEL TRABAJO DE GRADO
FORMULARIO

TÍTULO COMPLETO DE LA TESIS O TRABAJO DE GRADO			
CreaDevManager – Un Manejador de Proyectos para CreaTools			
SUBTÍTULO, SI LO TIENE			
AUTOR O AUTORES			
Apellidos Completos		Nombres Completos	
González Obando		Daniel Felipe	
DIRECTOR (ES) TESIS O DEL TRABAJO DE GRADO			
Apellidos Completos		Nombres Completos	
Flórez Valencia		Leonardo	
FACULTAD			
Ingeniería			
PROGRAMA ACADÉMICO			
Tipo de programa (seleccione con "x")			
Pregrado	Especialización	Maestría	Doctorado
X			
Nombre del programa académico			
Ingeniería de Sistemas			
Nombres y apellidos del director del programa académico			
Germán Alberto Chavarro Flórez			
TRABAJO PARA OPTAR AL TÍTULO DE:			
Ingeniero de Sistemas			
PREMIO O DISTINCIÓN (En caso de ser LAUREADAS o tener una mención especial):			
CIUDAD		AÑO DE PRESENTACIÓN DE LA TESIS O DEL TRABAJO DE GRADO	
Lyon, Francia		2013	
NÚMERO DE PÁGINAS			6
TIPO DE ILUSTRACIONES (seleccione con "x")			
Dibujos	Pinturas	Tablas, gráficos y diagramas	Planos
		X	Mapas
			Fotografías
			Partituras
SOFTWARE REQUERIDO O ESPECIALIZADO PARA LA LECTURA DEL DOCUMENTO			
Nota: En caso de que el software (programa especializado requerido) no se encuentre licenciado por la Universidad a través de la Biblioteca (previa consulta al estudiante), el texto de la Tesis o Trabajo de Grado quedará solamente en formato PDF.			

PUJ- BG Normas para la entrega de Tesis y Trabajos de grado a la Biblioteca General – Mayo de 2010

MATERIAL ACOMPAÑANTE					
TIPO	DURACIÓN (minutos)	CANTIDAD	FORMATO		
			CD	DVD	Otro ¿Cuál?
Vídeo	2 min c/u	8	X		
Audio					
Multimedia					
Producción electrónica					
Otro Cuál?	Página WEB	1	X		Videos y página WEB en el mismo CD
DESCRIPTORES O PALABRAS CLAVE EN ESPAÑOL E INGLÉS					
Son los términos que definen los temas que identifican el contenido. (En caso de duda para designar estos descriptores, se recomienda consultar con la Sección de Desarrollo de Colecciones de la Biblioteca Alfonso Borrero Cabal S.J en el correo biblioteca@javeriana.edu.co, donde se les orientará).					
ESPAÑOL			INGLÉS		
CreaTools			CreaTools		
Manejo de Proyectos			Project Management		
Proyectos C++			C++ Projects		
Procesamiento de Imágenes Medicas			Medical Image Processing		
BBTK			BBTK		
RESUMEN DEL CONTENIDO EN ESPAÑOL E INGLÉS (Máximo 250 palabras - 1530 caracteres)					
<p>Mantener la organización y la estabilidad de un proyecto cuando se trabaja con el software CreaTools para procesamiento de imágenes médicas es a menudo un problema grande y tedioso. Por esto, este informe presenta una arquitectura y una herramienta para mejorar la fase de desarrollo de cada proyecto hecho en CreaTools, además de simplificar la configuración del proyecto para mantener la estabilidad en el momento de la construcción de cada proyecto. La herramienta propuesta es compatible con la forma de crear proyectos del entorno de desarrollo CreaTools utilizada previa a la realización de la aplicación presentada.</p> <p>To maintain organization and stability of a project when working with the CreaTools software for medical imaging is very often a big and tedious problem. Consequently, this report shows the architecture and presents the tool to improve the development phase of every project made in CreaTools, and simplify the configuration of the project to keep the stability at the time of building each project. The tool proposed is intended to be backwards compatible with the way of creating projects with the CreaTools development environment.</p>					

CREADEVMANAGER: UN MANEJADOR DE PROYECTOS INFORMÁTICOS PARA CREATOOLS

Daniel Felipe González Obando

PONTIFICIA UNIVERSIDAD JAVERIANA

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS

Abril, 2013

Bogotá, 12 de Julio de 2013

Señores

COMITÉ DE TRABAJOS DE GRADO

Facultad de Ingeniería

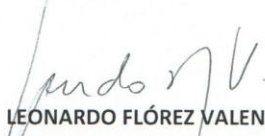
Pontificia Universidad Javeriana

Estimados colegas:

Como tutor del trabajo de grado del señor DANIEL FELIPE GONZÁLEZ OBANDO, titulado CREADEVMANAGER – UN MANEJADOR DE PROYECTOS PARA CREATOOLS, autorizo la entrega del documento final.

La sustentación del trabajo se llevó a cabo el pasado miércoles 10 de abril en la Universidad INSA de Lyon, Francia y fue realizado en presencia de la encargada de intercambios de esta universidad, Madame Béatrice Rumpler quien actuó como asesora del proceso. El concepto final correspondiente al trabajo de grado es *Aprobado*.

Cordialmente,



LEONARDO FLÓREZ VALENCIA

Profesor

Facultad de Ingeniería – Departamento de Sistemas

Pontificia Universidad Javeriana



Computer Science Department
Year 2012/2013

Summary Report

CreaDevManager - A Project Manager for the Black Box Toolkit

Daniel F. Gonzalez Obando,

Enterprise:

CREATIS, UMR CNRS 5220, INSA-Lyon

Bâtiment Blaise Pascal

7, avenue Jean Capelle

69621 Villeurbanne – France

Instructors:

Eduardo E. Davila Serrano, software engineer, info-dev team CREATIS

Maciej Orkisz, head of info-dev team CREATIS

Béatrice Rumpler, Computer Science Department INSA-Lyon

Abstract

To maintain organization and stability of a project when working with the CreaTools software for medical imaging is very often a big and tedious problem. Consequently, this report shows the architecture and presents the tool to improve the development phase of every project made in CreaTools, and simplify the configuration of the project to keep the stability at the time of building each project. The tool proposed is intended to be backwards compatible with the way of creating projects with the CreaTools development environment.

Keywords

Software development project management, software architecture, CreaTools, Black Box, BBTK, Backwards compatibility.

Résumé

Maintenir l'organisation et la stabilité d'un projet lorsqu'on travaille avec le logiciel CreaTools pour l'imagerie médicale est très souvent un gros problème. Pour cette raison ce rapport présente l'architecture et l'outil pour améliorer la phase de développement de chaque projet réalisé en CreaTools, et de simplifier la configuration des projets pour maintenir la stabilité au moment de la compilation de chaque projet. L'outil proposé est destiné à être rétro-compatible avec le mode de création de projets à l'environnement de développement CreaTools.

Mots clefs

Gestion des projets de développement logiciel, architecture logicielle, CreaTools, Boîte Noire, BBTK, retrocompatibilité.

Introduction

CREATIS (Centre de REcherche en Acquisition et Traitement de l'Image pour la Santé) is a laboratory for biomedical imaging research, working in information science and technology along with health sciences. This center is divided in six research teams dedicated to make investigations on particular topic of medical image processing. In addition, an administrative department and an IT department bring support to the different activities developed in the laboratory.

The IT department is divided in two sub-departments, the hardware and equipment section and the software development section (info-dev). The main goal of the info-dev section is "to put in common and maintain the software components developed in the lab, so as to minimize the programming burden of the researchers"¹. One of the most known applications developed by the info-dev team is CreaTools, an open-source suite of tools that integrate different open-source technologies (vtk, itk, gdcM, wxWidgets, KWWidgets, boost, etc.) to create powerful image processing applications. In addition, CreaTools is licensed with CeCILL-B for free software².

One of the features CreaTools provides is a development environment that lets students and researchers create their own applications based on the existing components. Moreover, CreaTools allows developers to manage their projects, develop new modules and even participate to the CreaTools development. To accomplish this CreaTools uses different standalone tools to create the different components of a Crea project and make them available in multiple platforms (Windows, Mac and Linux). Additionally, CreaTools use the CMake technology to configure and compile the projects. Also, when a Crea project is compiled it can also be associated to CreaTools as an additional module, this also achieved by one of the tools provided by CreaTools.

Today, the task of managing Crea projects involve the usage of three standalone tools called creaNewProject, bbCreatePackage and bbCreateBlackBox to create the project's

structure. Besides, it is also necessary to know well the structure of a project as well as the technologies that are used within it. Lastly but not less important, CMake instructions have to be used to configure and compile the project. Clearly, managing a Crea project is not an easy task.

However, an effort to make the tools easier to use for the developers has been made by creating a documentation of the available tools and video tutorials of the usage of them. Nevertheless, this documentation is not easy to follow and the videos can easily become outdated by changes in the software.

This project handles the issue of the complexity of Crea projects by designing and implementing a tool called CreaDevManager (from Crea Development Manager), that unifies all the tools and technologies involved in the development process and provides a clean user interface to manage these projects. Moreover, this tool is intended to be available in Windows, Linux, and Mac operating systems.

This report is split into six sections. The first section explains the main system requirements the developed software has to accomplish according to the user needs. Following, a contextualization of the different medical image projects managers is exposed and their features and drawbacks are explored with the aim of fine-tuning the requirements of the project. Next, a proposed solution is provided beginning with a high level explanation of the system and its integration with the existing software. Then, the actual state of the software and the technical issues of the solution are provided. Finally, the conclusions of the project and the future work review the goals of the project and pose future challenges for CreaDevManager.

Due to the two column style of this report some illustrations could be difficult to read. In spite of that, the relevant diagrams in the project will be referenced to the project's website, where all the full-sized illustrations will be available.

System Requirements

When developing applications, the first step is to define the needs of the users. The UML's [1] use case diagram is not only a good method to explain these needs, but also it allows presenting the functional requirements separated by user type. In Figure 1 the different users and requirements are presented.

¹ Taken from the Creatis website www.creatis.insa-lyon.fr/site/en/info-team (as seen on 13/02/2013)

² The CreaTools website is available in http://www.creatis.insa-lyon.fr/site/en/CreaTools_home (accessed on 13/02/2013)

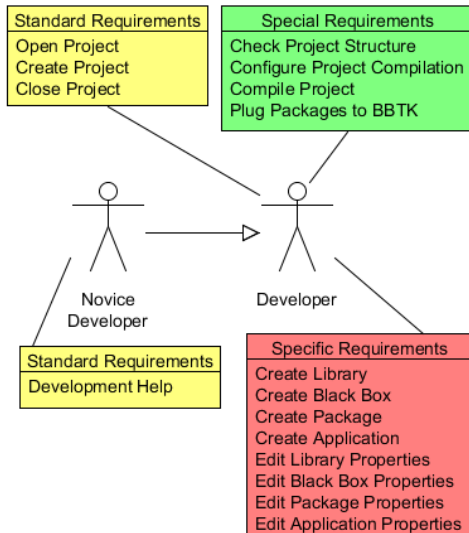


Figure 1 Project's Use Case Diagram

The requirements are divided in three different types. First, standard requirements describe basic requirements present in any project management application. Next, specific requirements represent those requirements that are only present in CreaTools projects. Finally, special requirements imply an interaction with third party tools such as CMake or the different types of compilers, or utilities that connect the project with other tools such as BBTk Graphical Editor³.

In addition, there are two types of users. This application has both novice and experienced developers as target users. For this reason, the development help presented to the user is very important as it could help beginners to gain experience in CreaTools.

Furthermore, it is necessary to establish some restrictions the product must meet in the sense of behavior, system design, and user experience. A standard method to express these restrictions is the quality attributes identification [2]. To propose a solution for the CreaTools project management the following items were selected.

Functionality: The application does the work it was intended to do, "Manage Crea Projects and help the user to develop his projects faster".

Reusability: Components developed in this project can be used in other projects and tools to keep the projects' model consistently.

Usability: Users are able to work intuitively with the application. Also, users should get enough help to get their projects up and running in less than half an hour.

Portability: The application can be run in different operating systems. All the code written must be available to use in Windows, Mac and Linux.

Configurability: The application can be configured by the user in order to work in the most comfortable way.

Context

As CreaTools, there exist some other toolkits that help on the task of medical image processing.

The most known is MeVisLab⁴ which allows the user to create pipelines and modules to build applications in medical image processing and visualization. As this development environment handles projects, it keeps organized all the developed filters and pipelines. The process of application development in this framework is:

1. Connect existing modules to networks.
2. Develop new modules, if necessary.
3. Build user interface (GUI).
4. Build macro modules to recycle complex functionality.
5. Use scripts to control networks, GUIs and macros.
6. Build installer (only with a special ADK license).

However, this tool does not guide the developer through the process of creating modules and packages; it assumes the developer knows how modules are created making the development process complicated for the novice developer. The Figure 2 shows the structure of an image processing pipeline in MeVisLab.

³ BBTkGEditor is a Graphical editor for the bbtck library. Description found in <http://www.creatis.insa-lyon.fr/site/en/CreatoolsBBTKGeditor> (accessed on 20/02/2013)

⁴ MeVisLab is a development environment for medical image processing and visualization. Available in <http://www.mevislab.de/>

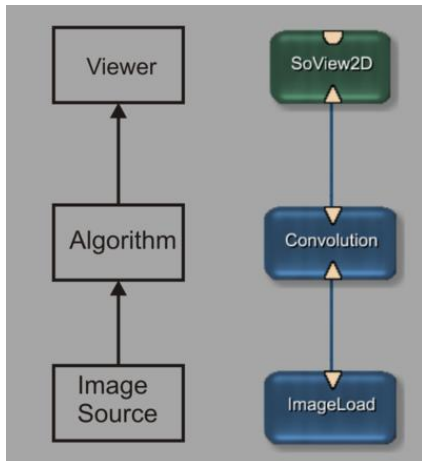


Figure 2 MeVisLab Sample Image Processing Pipeline

Second, GIMIAS⁵ is another framework that works with projects for developing medical imaging processing plugins that integrate with GIMIAS. It provides an API for the plug-ins to interact with them (these plugins can be widgets, front end plugins or processors). Figure 3 shows how this framework divides a project in three parts; modules (where GIMIAS components are used along with plugins to create application prototypes), third parties (external libraries), and plugins (where the customized plugins are located)

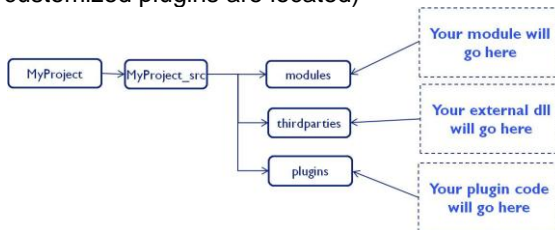


Figure 3 GIMIAS Projects' structure

These tools provide user manuals to guide novice developers, but they don't provide instant help when using each part of their projects. It is worth noting that the project structure of CreaTools differs from that from these tools. For this, CreaTools projects' structure had to be treated specially for this project.

Another known toolkit is LEADTOOLS⁶ Medical Image Suite which provides non open source sdk's and standalone applications for imaging and DICOM manipulation. As this toolkit only provides sdk's and some standalone applications, it doesn't provide a way to manage

⁵ GIMIAS is an environment for solving biomedical image computing problems. Available at <http://www.gimias.org/>

⁶ LEADTOOLS Medical Image Suite. Available at <http://www.leadtools.com/sdk/engine/medical.htm>

projects developed with the provided sdk's; the developer has to rely on Visual Studio to work with his projects.

The third reference tool isn't software for medical image processing; Eclipse is an open source integrated development environment (IDE) for different programming languages⁷. It helps to manage general programming projects. This tool displays a file tree to show the file hierarchy of open projects, as well as it shows dedicated controls for each type of project. This IDE provide code hinting which enhances noticeably the coding speed and guiding novice programmers.

As Crea projects have several components, it isn't easy to understand their organization. The idea of using file trees (Figure 4), as well as wizards, was taken from Eclipse IDE to develop the proposed solution.

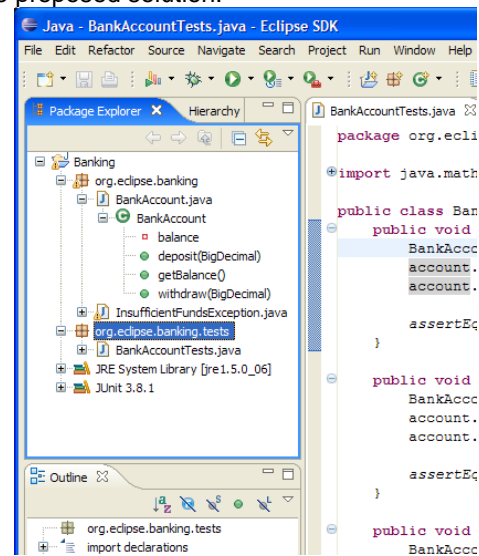


Figure 4 Eclipse IDE. Project file tree.

System Architecture and Design

To show the development of the proposed software solution for the Crea projects management a high level components analysis diagram is presented in Figure 5 showing the components involved in the process of creation, coding and compilation of Crea projects.

The core of CreaTools consists of two main components, Crea and BBTK. Crea handles the structure of Crea projects while BBTK handles the inherent black box architecture of Crea projects. The CreaTools component provides an

⁷ Eclipse is an open source IDE. Available at <http://www.eclipse.org/>

end user access to the tools available in CreaTools. As an example, BBGEditor, a graphical pipeline authoring tool, is one of several tools that come with CreaTools.

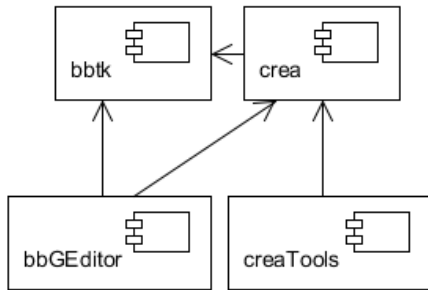


Figure 5 High Level CreaTools Components

In addition to the components, Crea relies on a specific project structure. For this, it was necessary to create an object diagram (Figure 6) that represented the structure of any Crea project.

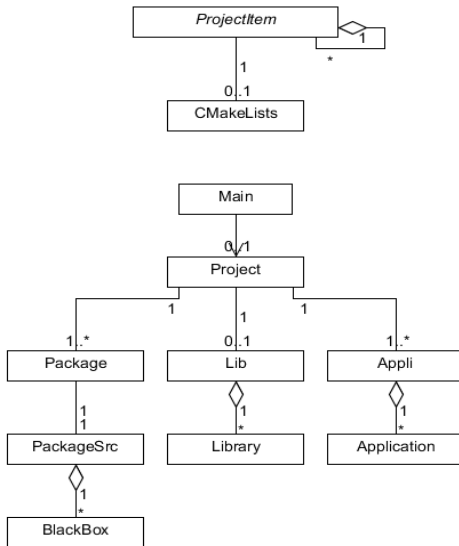


Figure 6 Project Structure Class Diagram

As Crea uses CMake to prepare the projects' compilation almost every item in a project has associated a CMakeLists configuration file. Furthermore, a project is mainly composed of packages (containing black boxes), libraries (located in a folder called lib) and stand alone applications (located in a folder called appli). With this structure a project can be well organized and easily managed. In addition, an activity diagram was created to understand the crea projects' workflow⁸, which explains how Crea projects are created and managed until

they are compiled and executed or used with other tools.

Once the Crea project structure was defined, the application structure was conceived as a PAC (Presentation, Abstraction, Control)[3] where the *abstraction* keeps the logical structure of projects, the *presentation* shows the user the visual controls to manage projects, and the *control* handles the modifications made to the projects. In contrast to MVC architectural pattern, PAC treats each part of the application as an agent; each agent is composed of a triad (presentation, abstraction and control) which can hierarchically contain other agents. This insulates the presentation (view in MVC) and provides the option for each agent to control what the user sees. This pattern was chosen instead of MVC because it could give each part of a project the power to control the presentation, and the control of itself and its children project items.

As explained in the PAC Architecture Components Diagram⁹, each project item has an abstraction, a presentation and a control component associated. Additionally, each project item could have children project items which also have the three PAC components.

Project items are presented in a window showing the project folder structure (a tree view) and the description view of the working project item. Additionally, a panel with the project compilation actions and separate help windows provide support in the process of developing Crea projects. An example of the CreaDevManager Main window presentation is available in Figure 7.

Finally, an Abstraction Classes Responsibilities Diagram¹⁰ is presented to explain what each item in the abstraction component is supposed to do in the final application.

Implementation and Tests

⁸ Available at: <http://www.creatis-insa-lyon.fr/~gonzalez/documentationAnalysis.html#creaWorkflow>

⁹ Available at: <http://www.creatis-insa-lyon.fr/~gonzalez/documentationDesign.html#PACDiagram>
¹⁰ Available at: <http://www.creatis-insa-lyon.fr/~gonzalez/documentationDesign.html#MResponsibilities>

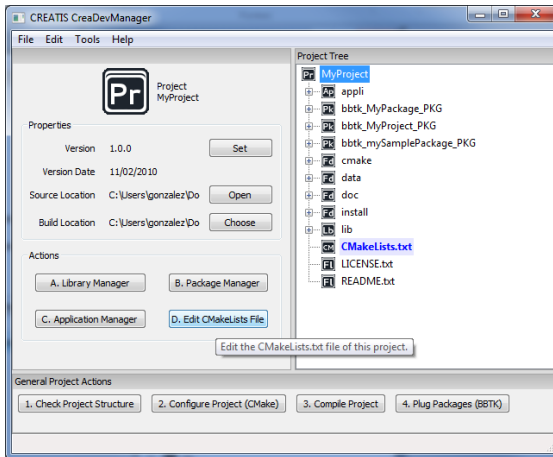


Figure 7 Crea Development Manager. Open Project

An interesting fact of the implementation of CreaDevManager is that it's a Crea project manager developed as a Crea Project. In fact, CreaDevManager was developed as part of the Crea project.

CreaDevManager was implemented using C++ and object-oriented programming style. For managing the project, eclipse was used along with CMake (to generate and compile the project). Moreover, as many other tools at CREATIS, Git was used as the project's source code management system.

Currently, CreaDevManager runs with CreaTools 2.1.0. It uses both wxWidgets 2.8 and CMake 2.8.8 stable versions. It implemented all the requirements shown in the use case diagram, as well as the requirements stated in the software requirement specification document¹¹. Due to time, the product could only be implemented and tested in Linux and Windows; a version for Mac could not be fully implemented. Also, because of insufficient time, some presentation errors are present in the latest version of the product and are considered for future work. An example of these errors is the help dialog handled as a simple string as it should be formatted and shown perhaps as an HTML dialog (Figure 8).

¹¹ A full requirements list is available at: <https://www.creatis-insa-lyon.fr/~gonzalez/documentationRequirements.html>

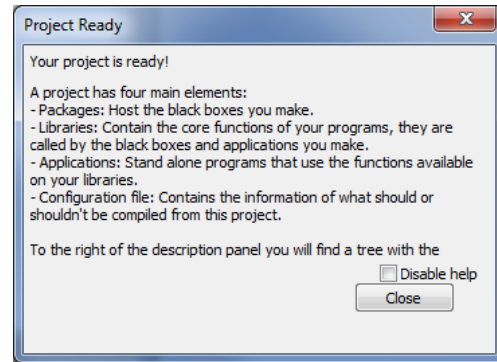


Figure 8 Unformatted Help Dialogs

Even though all requirements were implemented, as Crea projects are managed using different technologies, CreaDevManager behaves according to each system. In spite of that, the tools used are not the same or behave differently from operative system to operative system.

Equally important, tests were developed according to requirements and how each action in the project management process is performed [4]. Tests were run both for Linux and Windows and the project was debugged until all tests passed. These test allowed seeing the importance of developing an accurate architecture and a design according to the real needs of the product.

Currently CreaDevManager is being included in the latest version of CreaTools and is expected to be released in the next version along with a user's manual document.

Conclusions

It is important to provide a way to organize the code developed for medical image processing. CreaTools provides the structure of a project where code is ordered according to its nature and purpose.

In addition, having a project manager for CreaTools increases the productivity of developers as well as it allows to understand the structure of Crea projects and simplifies the development process of CreaTools itself.

Finally, as projects become more and more complex in a toolkit, it is important to provide a way to simplify the management of these projects in order to enhance its productivity.

Future Work

Despite all requirements collected in the analysis phase were implemented, there are still more requirements not featured in the current version of the application. For example, the support of bbs scripts, bbg pipelines, as well as some other needs identified after the product development began. A detailed list of “nice to have” features is presented in the project’s website¹².

In addition, there are some presentation issues that should be taken in account to improve the usability of the application. An example of this was already presented in the previous section where the help dialogs were not formatted and look odd as the window size varies. Some other detected visualization issues were caused by the multi-system nature of the application; for example, in Windows some texts are not correctly wrapped in multiline text boxes while in Linux this is already implemented (Figure 9).

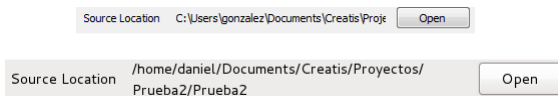


Figure 9 Word wrapping difference between Windows (up) and Linux (down)

Finally, it is worth noting that even though there are changes to be made, the extensibility of the application allows implementing these changes without much effort and providing functionality to users.

References

- [1] Object Management Group. (2011, Aug.) The Unified Modeling Language. [Online]. <http://www.uml.org/>
- [2] Microsoft Patterns & Practices Team, *Microsoft® Application Architecture Guide, 2nd Edition*. Microsoft Press, 2009.
- [3] J. Coutaz, "PAC-ing the Architecture of Your User Interface," *4th Eurographics Workshop on Design, Specification and Verification of Interactive Systems*, pp. 15-32, 1997.
- [4] J. Hartmann, C. Imoberdorf, and M. Meisinger, "UML-Based Integration Testing," *Proceedings of the*, Aug. 2000. [Online]. http://www.cs.cmu.edu/~luluo/Courses/17939Report.pdf

¹² “Nice to have” features are available in <https://www.creatis.insa-lyon.fr/~gonzalez/niceToHave.html>