VOL. 12, NO. 21, NOVEMBER 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

SOFTWARE FOR MONITORING AND CONTROL OF ENZYMATIC EXTRACTS: A TOOL FOR RESEARCH

Fernando González-Fernández, Claudia Lorena Garzón-Castro and Erlide Prieto
Engineering Faculty, CAPSAB and GIPA Research Groups, Universidad de La Sabana, Campus Universitario del Puente del Común, Km 7 Autopista Norte de Bogotá, Chía, Cundinamarca, Colombia
E-Mail: claudia.garzon@unisabana.edu.co

ABSTRACT

A key aspect in bioprocesses is the monitoring and control of the physical variables that affect it. Hence, the tailor-made software development, capable of: reading, storing and graphing information of process variables is of vital importance to users. This article presents the development of SICOAVENZ (Automatic Variable Control System for Enzymatic Reactions), an application that communicates with a data acquisition board, to which sensors and actuators that allow interaction with a bioreactor are connected, in which an enzymatic extract is obtained. The software was developed under object-oriented programming using C # (C Sharp). Where it is shown that implementing low-cost systems using efficient, easy-to-manage and potentially scalable software architectures, is possible. It was also possible to develop a software that allows: 1) control pH and temperature variables, 2) keep a constant record of the behavior of these variables, 3) generate reports in Microsoft Excel, 4) handle information according to the role (Administrator or User) and 5) to observe through a video camera the experiment in progress. This development is characterized because it can: 1) have a wide range of application in different fields of action, 2) ensure reproducibility of the process and 3) give independence to scientific personnel.

Keywords: computer applications, monitoring and control software, software architecture, software design, enzyme extracts.

1. INTRODUCTION

Lipases are specific enzymes used to catalyze the hydrolytic separation of long fatty acids chain present in the acylglycerols [1]. The analysis of the release speed of the fatty acids can be done by titrating the pH decrease over time. The above, taking a constant temperature value according to the experiment and the used substrate [2].

Although there are several known applications developed for the monitoring and control of bioprocesses [3], [4], most of them are expensive. A huge investment that is not economically viable if they are meant to be used in educational institutions [5]. Thus, a low-cost software development used to control bioprocesses, which reaches user requirements, is an interesting field of application [5], [6].

In order to determine functional and non-functional requirements, it is important to understand the biological process and to work interdisciplinary. When having clarity of the requirements, it is possible to: 1) establish the software architecture allowing to fulfill each of these [7], and 2) selecting as indicated [8] the structural elements and their interfaces, the composition of the structures, the behavior of each of the subsystems and the relationships between them [7], [9]. This allows to fulfill the characteristics of social skills, autonomy and proactivity that stands out [10]. It is important to emphasize that using methods analysis based on software architecture, results in benefits such as: performance analysis, component validation, among others [11].

Therefore, the importance of having a software focused on monitoring and controlling in real time, in order to obtain a tool to perform experimental research. In addition to meeting the specific requirements of the users.

Currently at the Universidad de La Sabana, there is no equipment that allows the performance of specific experiments with enzymatic extracts. This led to the development of a laboratory scale system to solve this need. The objective of this article is to show the advantages of the developed software, which is called SICOAVENZ (Automatic Variable Control System for Enzymatic Reactions). This is characterized by: 1) making a configuration according to the characteristics of each experiment; 2) generating reports in Microsoft Excel with information as: name of the investigator, period of time in which the experiment was run and values registration of the variable process; 3) managing the information according to the a given role (Administrator or User); 4) visualizing through a video camera the behavior of the experiment in real time, simulating a laboratory; and 5) connecting to a data acquisition board via either the RS232 or USB interface.

2. OVERVIEW OF THE SYSTEM

In Figure-1 a general scheme of SICOAVENZ is shown. Its objective is to monitor and control the variables process (temperature and pH) in a bioreactor (BR) containing an enzymatic extract. The development of both hardware and software were tailor-made.

2.1 Hardware

The designed hardware was divided into three subsystems: 1) data acquisition board, 2) sensors and actuators and 3) BR.

A. Data acquisition board. It works with TTL levels to capture and send digital data via RS232 port or USB port. It has 8 digital inputs pins, 8 analog inputs pins, 8 digital outputs and 2 pulse width modulation outputs (PWM). Data and transmission capture of information frames towards the computer are performed automatically

ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

every 100 ms. A MC68HC908GP32 Freescale microcontroller semiconductor was used. This one belongs to the high performance microcontrollers family, where it combines its: reduced size, high speed with low power and high immunity to noise. The main features of this low cost device [12] can be found in Table-1.

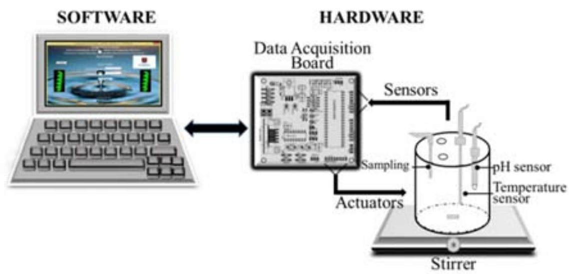

Figure-1. General scheme of SICOAVENZ.

Table-1. Main features of microcontroller.

Characteristics	
Flash memory	32 Kbytes
RAM memory	512 bytes
CPU internal bus frequency	8 MHz
Port D	Serial peripheral interface (SPI)
Port E	Serial communications interface (SCI)
2 timers that are programmable independently as input capture, output compare or PWM signal.	

B. Sensors and actuators. The environment temperature is monitored through a PT100, which is a particular RTD type (Resistance Temperature Detector) and the pH is monitored with a Hanna HI100 electrode. As actuators the following items were used: 1) proportional solenoid valves of direct action, 2) a heater, 3) a dispenser pump and 4) a video camera (this allows the user who enters as an administrator to have real time monitoring of the experiment).

C. Bioreactor. The BR corresponds to a beaker with a 1 L working volume. It has ports for: supplying the base or acid according to the experimental conditions, the PT100 and the electrode. The electromagnetic stirring is provided by a Cold-Parmer StableTemp plate.

2.2 Software

The software was developed with object oriented programming, using C Sharp (C #) programming language, this way it managed to control the variables through the Development Environment (IDE) Integrated tool, as Microsoft® Visual Studio. The reports are generated with the Microsoft® Excel office tool. Additionally, a monitoring and control system was included to be able to monitor the experiment giving simulating being in the laboratory.

For this development, software engineering features established by [13] were taken into account: modifiability, interoperability, availability, security, portability, performance, reliability and low cost were considered. In Figure-2, one of the graphical user interfaces (GUI) of the software is shown.

Figure-2. User welcome GUI.

The minimum computer requirements for SICOAVENZ to properly work are listed in Table-2.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-2. Minimum computer requirements.

Requirements	
Operating system	**Microsoft® Windows XP SP2 o SP3/Windows Vista SP2/Windows 7 SP1/Windows 8**
Processor	Intel Pentium 4 o AMD compatible superior de 2.0 GHz
RAM memory	1.5 GB (4GB recommended)
Hard disk drive	10 GB
Screen resolution	800x600 pixels
Interface	USB 2.0 o RS232 Port
Office tool	Microsoft® Excel
DB motor	SQL Server 2008
Framework	.NET Framework 3.5 SP1
USB to Serial drivers	Trendnet TU-S9 Version 1.xR

3. SOFTWARE DESIGN

According to the functional and non-functional requirements, the software was divided into four modules: process management, user interaction, interoperability and event detection.

3.1 Process management

This module: 1) executes a thread whose process continuously makes readings of the sensors as per a pre-programmed frequency, 2) compares the values of the readings with the configuration parameters stored in the database and 3) makes the necessary adjustments in the system to keep the values within the configured set-points. Since the sampling conditions of the process variables may vary depending on the experiment to be performed, ten timers had to be implemented. These are divided as follows:

A. Three physical timers. Designed only for the researcher to enter the relevant information for the data gathering of the process variables: temperature and pH. Each one of these timers allows setting sampling times in minutes for each number of hours (data set by the investigator), see Figure-3.

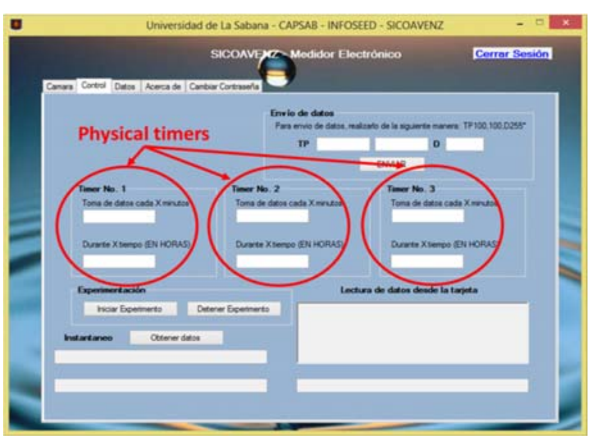

Figure-3. Physical timers.

B. Seven neutral timers. Configured in the source code, they are not visible to the user. Its tell each physical timer when to start and end. Each neutral timer allows to set sampling times in minutes for a certain number of hours for each physical timers (data set by the investigator). As a result, each of the physical timers works with the neutral timers to set both, the time interval in which each process variable must be measured, as well as the sampling time.

It is not necessary to use the three physical timers; it can also happen that only set the first timer is set by the investigator. In this case only two of the neutral timers are used (one counts in minutes for the sampling time and the other counts the minutes to complete the experiment). If two physical timers are required, the software will use five of the neutral timers. Only in the case of activating all the physical timers, the seven neutral timers will be used (see Figure-4).

3.2 User interaction

This module belongs to the section in which the user may: 1) configure, monitor and modify the parameters of experimentation without having to stop the experiment; and/or 2) analyze its data. The above as per the given role you have. The existing roles are: 1) Administrator and 2) User.

A. Administrator role. The user declared as Administrator has full access and software control. You may: 1) perform the addition of the records for any kind of person, 2) stop the video camera system, 3) perform the experiments as many times as desired and 4) generate the behavior reports of the variables involved in the process. This role is shown by the use case diagram shown in Figure-5.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

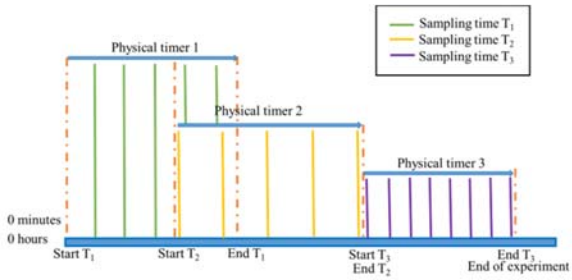

Figure-4. Physical and neutral timers.

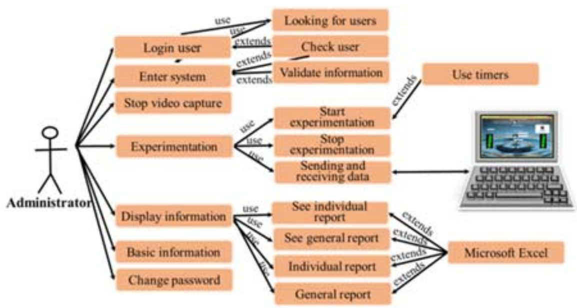

Figure-5. Use case diagram for administrator role.

B. User role. The person with a User role has restrictions, such as not being able to: 1) access new user registrations and 2) stop the surveillance system. Figure-6 shows the corresponding use case diagram.

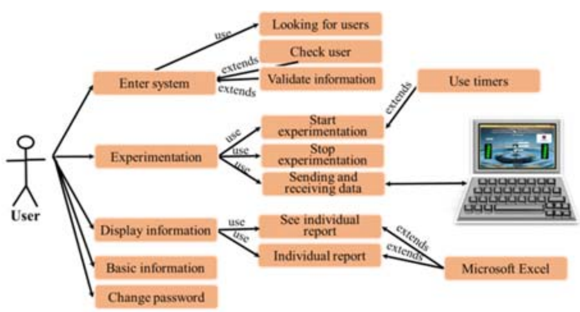

Figure-6. Use case diagram for User Role.

Any of the previously mentioned roles, through the graphical user interface can modify your password and manipulate the information with Microsoft® Excel office.

3.3 Interoperability

Interoperability refers to systems that work within the software to exchange information. These systems are:

Surveillance. This requires the use of Dynamic Library Link (DLL) files for the proper control and operation of the video camera experiment inspection. The source code presented in Figure-7, the manipulation of the AVICAP32.DLL file is shown, and how it was integrated into the development environment. This file is a module that contains functions for the Application Programming Interface (API) of the Microsoft® Windows Operating System. Additionally, it provides the respective control of the video input devices that are used to capture this in AVI format. In addition, it searches the computer for all the installed video devices, stores this information in a memory and then the list is added to a Combo Box, used and integrated with SICOAVENZ.

```
class DeviceManager
    {
        [DllImport("avicap32.dll")]
        protected static extern bool capGetDriverDescriptionA(short wDriverIndex,
            [MarshalAs(UnmanagedType.VBByRefStr)]ref String lpszName,
           int cbName, [MarshalAs(UnmanagedType.VBByRefStr)] ref String lpszVer, int cbVer);
        static ArrayList devices = new ArrayList();

        public static Device[] GetAllDevices() {
            String dName = "".PadRight(100);
            String dVersion = "".PadRight(100);
            for (short i = 0; i < 10; i++) {
                if (capGetDriverDescriptionA(i, ref dName, 100, ref dVersion, 100)) {
                    Device d = new Device(i);
                    d.Name = dName.Trim();
                    d.Version = dVersion.Trim();
                    devices.Add(d);
                }
            }
            return (Device[])devices.ToArray(typeof(Device));
        }

        public static Device GetDevice(int deviceIndex) {
            return (Device)devices[deviceIndex];
        }
    }
```

Figure-7. Source code of surveillance system.

B. Interconnectivity. It works once the TRENDnet TU-S9 (USB to Serial Port Converter) cable is installed and configure correctly, as long as the data acquisition board is switched on.

C. Reports generation. Its operation will be correct, only if the Microsoft® Excel office tool is installed.

3.4 Event detection

Event detection may be: defined or triggered by the user. See Table-3.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-3. Event detection.

Defined by the user	Caused by the user
Let the software run in the background while performing other activities on the same computer.	In cases where the access, registration or control forms are not completely typed.
Welcome message when the user successfully accesses the program.	When the board or communications port number is disconnected or misconfigured.
Not allowing typing numeric characters where only letters should be entered, and vice versa.	If the user tries to use the timers without following the respective order
Avoid running more than one window at the same time, this in order to use properly the physical memory of the computer without affecting the performance of itself.	

4. SOFTWARE ARCHITECTURE

The layer architecture of the software is shown in Figure-8. There, each of the layers present in the software is structured, to fulfill a proposed goal.

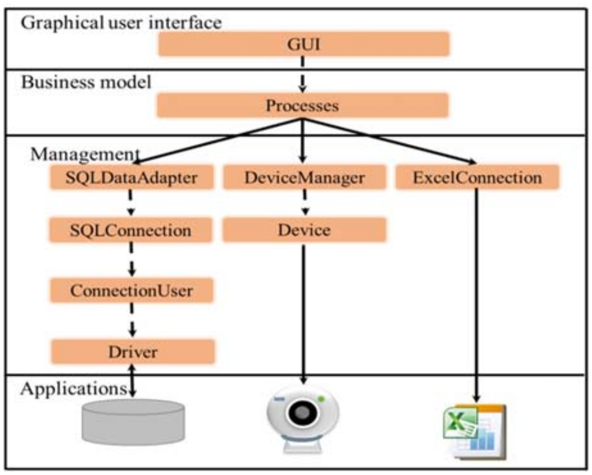

Figure-8. Simplified Layer Diagram.

4.1 GUI (Graphical User Interface)

It is composed by all the graphical interfaces that form the structure that allow interaction between the user and the software. Each graphical interface was designed in order to facilitate the handling the software so it works smoothly.

4.2 Business model

This layer is composed by of all the classes that make up the application. The source-code based construction is found here, using the software model. This layer is characterized by the state and behavior of all objects in the class that share and give life to the application. It can be said that it is the most important layer, since it performs the management and control of all internal and external transactions of the program. For instance, if the researcher completes his experimental work, the data is read through it, stored in memory, uploaded and sent to Microsoft® Excel.

4.3 Management

All the instances and methods used in classes that allow the proper use and operation of each of the devices and applications (database, video camera and Microsoft® Excel office tool), are found in this section. This layer interacts in real time with the Business Model layer.

4.4 Applications

This layer is made of all the programs and/or devices that are in use, such as: the database engine in Microsoft® SQL Server, the Microsoft® Excel office automation tool and the video camera.

The database has a local connectivity level on a server installed to a SQL Server 2008 database engine or a higher version. The above with the help of the libraries present in the .NET Framework 3.5. Here, the signals produced through the software are incorporated for a later analysis and, on the other hand, reference is made to the user registers and the data generated through the data acquisition board connected to the system.

5. RESULTS AND DISCUSSIONS

SICOAVENZ being an application developed to conduct research, the reliability of the reported data must guarantee. For this reason, stability and performance measurements were made. The above, using the performance tool of the task manager and one of the administrative tools using the Windows Operating System as the performance monitor. The latter allows: 1) to view detailed information in real time on the hardware resources (CPU, Central Processing Unit, disk and memory) used by the Operating System and 2) to analyze deadlock processes, see wait chains for threads and locks presented by the applications that are currently running on the Operating System. The analysis was performed on two computers with different characteristics, see Table-4.

VOL. 12, NO. 21, NOVEMBER 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

Table-4. Characteristics of the computers in which the analysis was performed.

Characteristics	Laptop	Desktop Computer
Sistema operativo	Microsoft® Windows 8 Profesional x 64	Microsoft® Windows 7 Profesional x 64
Procesador	Quad-Core AMD A8-3500M 1.50GHz	Dual-Core Intel Core 2 Duo E6300 1.86GHz
Memoria RAM	6 GB	2 GB
Microsoft® Office	2013 Professional Plus 64 bits	2013 Home and Student x86
Librería .NET Framework	4.5 SDK	3.5 SDK

In Figure-9 the behavior of laptop computer performance with and without running software are shown. For example, Figure-9b shows a high processing of information by the CPU, because the data acquisition board sends information for processing. Even when the amount of information to be processed is high, the usage of physical memory is minimal. Additionally, it is guaranteed that the monitor will not reach 100% usage since the software was designed taking that into account. This allows the equipment to be poorly affected in performance quality performance.

In order to verify the correct communication of the data acquisition board with the software, tests of stability and veracity of the information provided by the hardware were carried out. Since the data acquisition board takes a sample every 100 ms, it was discovered that the last two data received and processed by the software were not complete. This data is stored in the attached richTextBox tool in the Visual Studio integrated development environment. To avoid errors, the decision made was to take the data "n-2" of the richTextBox, where n is the quantity of each of the frames acquired and processed by the software. The alternative of taking the value "n-2" to store in the database, does not affect at all the functionality of the system since the required sampling by the experiments is much greater than 100 ms. This guarantees the integrity of the information.

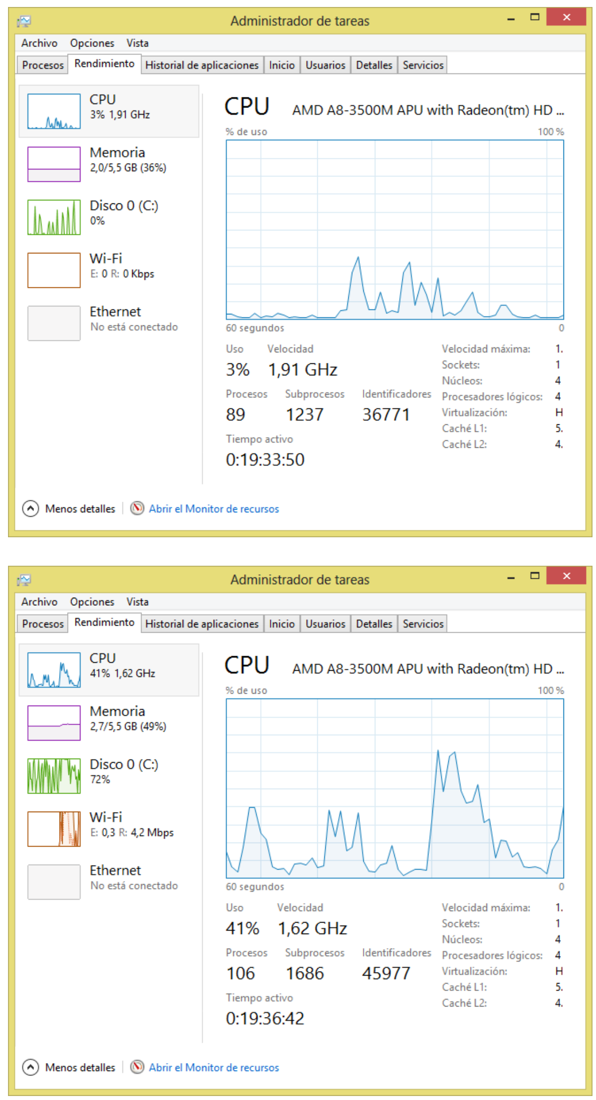

Figure-9. Machine performance: a. no software and b. with execution software.

When testing on different computers with different characteristics in hardware configuration, it could be guaranteed that the software would work properly on slow computers.

Although in the industry there are different software and systems designed to control and monitor bioprocesses, while being general developments, they do not meet the specific requirements of researchers. Additionally, the cost is high if you keep in mind that the application is required to conduct research at the academy.

6. CONCLUSIONS

The development of software for research purposes was achieved, and this can give independence to scientific personnel and increase the reproducibility of routine production processes. The software was designed in a modular and orderly way in order to be able to address new requirements in a quick and effective way.

The interdisciplinary character of this work shows the importance of itself, because it is possible to realize developments that a single discipline would not be able to implement. It was possible to develop software: 1) low cost, 2) easy to use, 3) tailor made, 4) capable of interacting with hardware in charge of monitoring and control variables temperature and pH, 5) provide real and accurate information, and 6) guarantee the investigators the precision and reliability of its use.

In the future, you may increase the number of variables to control. The above, to study what effect it has on each or the relationship of these in an enzymatic extract.

ACKNOWLEDGEMENTS

The authors would like to thank at Departamento Administrativo de Ciencia, Tecnología e Innovación COLCIENCIAS by the financing for the development of “Caracterización y mejoramiento de la acción de la lipase de *P. aeruginosa* para su utilización en la producción de biodiesel y olequímicos” project. Additionally, they would also like to thank at Universidad de La Sabana for the support and provided facilities.

REFERENCES

[1] E. A. Snellman, E. R. Sullivan and R. R. Colwell. 2002. Purification and properties of the extracellular lipase, LipA, of Acinetobacter sp. RAG-1. Eur. J. Biochem. 269(23): 5771-5779.

[2] J. González-Bacerio, J. R. Hernández, and A. del M. Martínez. 2010. Las lipasas: enzimas con potencial para el desarrollo de biocatalizadores inmovilizados por adsorción interfacial. Rev. Colomb. Biotecnol. 12(1): 124-140.

[3] F. Capraro, S. Tosetti, and F. Vita Serman. 2010. Laboratorio Virtual y Remoto para Simular, Monitorizar y Controlar un Sistema de Riego por Goteo en Olivos. Rev. Iberoam. Automática e Informática Ind. RIAI. 7(1): 73-84.

[4] A. Nakrachi, C. Chera and C. Dimon. 2006. Air-stream and Temperature Plant Remote Control. Proc. Multiconference Comput. Eng. Syst. Appl. 1: 20-21.

[5] L. M. Beltrán, C. L. Garzón-Castro, F. Garcés and M. Moreno. 2012. Monitoring and control system used in microalgae crop. IEEE Lat. Am. Trans. 10(4): 1993-1998.

[6] W. R. Jones, M. J. Spence, A. W. Bowman, L. Evers and D. A. Molinari. 2014. A software tool for the spatiotemporal analysis and reporting of groundwater monitoring data. Environ. Model. Softw. 55: 242-249.

[7] H. Muccini, M. Dias and D. J. Richardson. 2006. Software architecture-based regression testing. J. Syst. Softw. 79(10): 1379-1396.

[8] Booch, J. Rumbaugh and I. Jacobson. 1998. The Unified Modeling Language User Guide. Vol. 3.

[9] I. A. W. Group. 2000. IEEE Std 1471-2000, Recommended practice for architectural description of software-intensive systems. p. i-23.

[10] H. Soza Pollman. 2014. Evaluando características del agente software Evaluation of the characteristics of software agent. 22(no. August): 435-444.

[11] A. Bertolino, P. Invernardi, and H. Muccini. 2003. Ormal Methods in Testing Software Architectures. in Formal Methods for Software Architectures, vol. 2804, M. Bernardo and P. Invernardi, Eds. Berlin Heidelberg: Springer-Verlag. pp. 122-147.

[12] Freescale Semiconductor. 2008. MC68HC908GP32 Data Sheet: M68HC08 Microcontrollers. http://cache.freescale.com/files/microcontrollers/doc/data_sheet/MC68HC908GP32.pdf. [Online]. Available: http://cache.freescale.com/files/microcontrollers/doc/data_sheet/MC68HC908GP32.pdf.

[13] F. P. Brooks. 1995. No Silver Bullet - Essence and Accident in Software Engineering. pp. 1-16.