

Agile Methodology to Develop Architecture of Information and Knowledge in Organizations (MADAIKE)

Nieto Bernal Wilson and Luna Amaya Carmenza

Abstract—In This article presents a methodology based on the articulation of emerging paradigms for architecture development of information and knowledge (MADAIKE) that in turn promote the integration of various standards and justify the value to the organization. The first one comes from the agile methods and it is inspired on the Scrum model which aim to simplify the complex task of developing a quality software, the second one takes as a reference the processes models whose are oriented toward the development of Architectures of business information as Zachmantm, TOGAftm, SOMFtm, TOGAF, UPDM and others, whose purpose is to articulate in a single model the business architecture and the information architecture, it is framed, of course, in a paradigm of the model driven generation (MDG), and the third approach with project planning (PP) inspired by the PMBOK standard and especially in the projects planning process group, in addition to this, there are important aspects related to the UML 2.5 and the business process modeling that become tools to obtain the products in the MADAIKE methodology. These approaches are integrated eventually leading to the formulation and presentation of an agile methodology called – MADAIKE.

Index Terms—Agile methodology, architecture information enterprise, software process, project planning, validation, verification, management knowledge, systems engineering.

I. INTRODUCTION

The construction of IEA is supported in a Framework, which is constitute by an abstraction of multiple implementations and uses a set of tools like the model of business process, the map for organizational process, process characterization and tools management for application modeling, logical and physical data of the Information Systems that use a specific domain taxonomy, methods, guidelines and the best practices that already incorporate important aspects such methods, taxonomies and best practices, including standards and regulations, however for the purposes of the methodology, the models to get as a result are going to be simplified considerably [1]-[3]. For these reasons this paper presents a methodology called Agile MADAIKE consisting of a flexible multilayer structure, from a baseline of Scrum to obtain the IEA, in Fig. 1. MADAIKE structure, it displays the structure of the methodology we proposed, it is consists of five main layers: business modeling layer, layer information modeling, modeling application layer and the IT infrastructure modeling layer and is supported by a planning project integrated layer.

Manuscript received May 22, 2014; revised December 22, 2014.

Nieto Bernal Wilson and Luna Amaya Carmenza are with Universidad del Norte Colombia, Colombia (e-mail: wnieto@uninorte.edu.co).

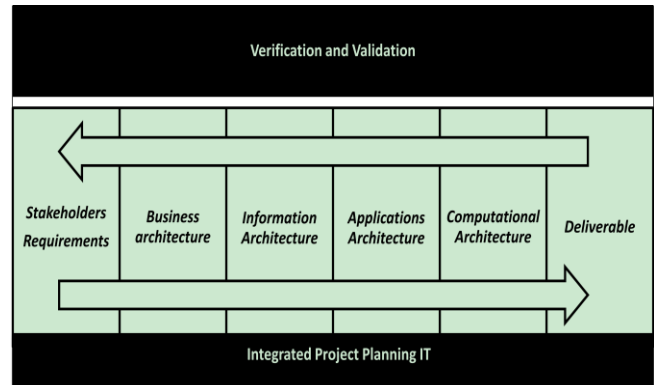


Fig. 1. Methodology structure MADAIKE.

II. RESEARCH METHODOLOGY

The following are the phases of exploratory research that conceptually possible to establish the methodology MADAIKE, three phases were developed and are described as follows:

Phase 1: Analysis of development standards for IEA (state of the art): This phase was aimed to evaluate the different standards for the development of IEA, currently there are several standards used for the development of the IEA on the basis of a new paradigm called MDG (Generation Driver Modeling), in this regard at present there are several technologies such as MDG SysML™, DDS, Zachman Framework™, TOGAF™, SOMF™, and UPDM™ each with different approaches and techniques to address the development of the IEA [4]-[7].

The Systems Modeling Language (SysML) is a general-purpose visual modeling language for systems engineering applications. It supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems. These systems may include hardware, software, information, processes, personnel, and facilities. SysML is a specified as a profile (dialect) of the Unified Modeling Language™ (UML™) [8], [9].

Features:

- Specify, design and analyze complex system models
- Model with all 9 SysML 1.3 diagrams
- Visualize and trace requirements to model elements throughout the entire development lifecycle.
- Custom Search Facility: Perform complex searches, view SysML Allocations and generate reports from the results.
- Support for XMI 2.0, XMI 2.1 and UML 2.x

DDS, Data Distribution Service is a standard widely used for developing the architecture of real-time systems is the

specification for a middleware type publish / subscribe distributed systems, DDS has been created with the purpose of serving the needs of industry standardize data-centric systems.

Key Features:

- Specify Data-Centric Publishers, Subscribers, Topics and QoS Policies.
- Define Data Local Reconstruction mappings for effective DDS data access.
- Target DDS implementations for the Open Splice and RTI platforms.

Framework Zachman is an Enterprise Architecture framework for enterprise architecture, which provides a formal and highly structured way of viewing and defining an enterprise. It consists of a two dimensional classification matrix based on the intersection of six communication questions (What, Where, When, Why, Who and How) with six levels of reification, successively transforming the abstract ideas on the Scope level into concrete instantiations of those ideas at the Operations level.

Key Features:

- Displays the IEA from different perspectives: Executive Management, Process, Architecture, Engineering and Technology.
- Identification, Definition, Representation, Specification, Configuration and Instantiation (Inventory, Process, Networks, Responsibility, Timing cycles and Motivation intentions).

The Open Group Architecture Framework (TOGAF):

It is a framework for developing Enterprise Architecture provides an approach to the design, planning, implementation and governance of enterprise architecture information. This architecture is usually modeled with four levels or dimensions: Business, Technology (IT), Data and Applications.

Key Features:

- Implement all phases of the TOGAF Architecture Development Method (ADM).
- Create visual models of As-Is and To-Be architecture.
- Model all four TOGAF architecture domains: Business, Application, Data and Technology

UPDM, the Unified Profile for DoDAF/MODAF (UPDM) is the product of an Object Management Group (OMG) initiative to develop a modeling standard that supports both the USA Department of Defense Architecture Framework (DoDAF) and the UK Ministry of Defence Architecture Framework (MODAF). The MDG Technology for UPDM provides a model-based framework for planning, designing and implementing the Unified Profile for DoDAF and MODAF (UPDM) architectures. Source: <http://www.sparxsystem.com>

Key Features:

- Create architectural models for complex system-of-systems, which may include hardware, software, data, personnel and organizations.
- Define consistent, accurate architectures with clear separation of concerns to describe services, systems, operations, strategies and capabilities.
- Analyze, specify, design, and verify system models using appropriate levels of abstraction.

- Employ a rigorous, standards based approach to defining and exchanging architecture information using UML, XMI and related standards.

SOMF™ is a model-driven engineering methodology whose discipline-specific modeling language and best practices focus on software design and distinct architecture activities, employed during various stages of the software development life cycle. Moreover, architects, analysts, modelers, developers, and managers employ SOMF to tackle enterprise architecture, application architecture, service-oriented architecture (SOA), and cloud computing organizational challenges.

Key Features:

- To achieve these underpinning milestones, six distinct software development disciplines offer corresponding models whose language notation guide practitioners in designing, architecting, and supporting a service ecosystem: Conceptual Model, Discovery and Analysis Model, Business Integration Model, Logical Design Model, Architecture Model and Cloud Computing Toolbox Model.

Phase 2: Survey and judgment of experts to identify best practices for developing IEA: This phase seeks to know the opinion of experts on the use of methodologies and standards for the development of IEA, for this case work a sample of 25 experts, responsible for addressing projects in organizations raised 10 questions and their answers are:

- Q1: Meet standards, methodologies or processes for developing standardized IEA? (Yes: 35% / No: 50% / Na: 15%).
- Q2: Which of the following standards which has been used in some sort of project? (TOGAF: 10%; Zachman: 12%, UPDM: 5%, SysML: 3%, DoD: 1%, FOBT: 5%, none of the above: 75%), [multiple choice questions].
- Q3: In your company adopt good practices, standards, methodologies and standardized processes for software development in general? (Yes: 45% / No: 40% / Na: 15%)
- Q4: They had difficulties in the projects on the use of standards, methodologies and processes for developing standardized EAI? (Yes: 65% / No: 30% / Na 5%).
- Q5: You have qualified personnel in developing standards for IEA? (Yes: 30% / No: 70% / Na: 0%).
- Q6: You have tools to properly utilize technological resources standards for the development of IEA? (Yes: 50% / No: 40% / Na: 10%).
- Q7: It is easily accessible training of human resources in developing standards for IEA? (Yes: 54% / No: 34% / Na: 12%).
- Q8: The staff of the organizations understands the importance of using standards for the development of IEA? (Yes: 47% / No: 43% / Na: 9.8%).
- Q9: You prefer to outsource the development of IT solutions, which involves using standard methodology or standardized processes for the development of IEA? (Yes: 51% / No: 36% / Na: 13%).
- Q10: You think appropriate to have an agile methodology, simple, efficient and effective for IEA product development, standards-based IEA to develop? (Yes: 61% / No: 18% / Na: 21%).

Phase 3: Analysis of results and considerations on the proposed methodology.

- From Q1: The people responsible for the management and implementation of projects unknown standards, methodologies and processes for developing standardized. IEA regularly use conventional methods for their projects.
- From Q2: The results indicate that very few experiences and projects which use standards, methods or processes for the development of SIA, a high percentage 75% indicate their failure used this type of technology.
- From Q3: On the use of general standards for software development, the results are divided, however 40% do not use these technologies for software development in general.
- From Q4: There are difficulties in the projects for the use of standards, methodologies and processes for developing standardized IEA, most of the time is presented by a lack of qualified staff skills.
- From Q5: One of the great weaknesses of the fact of not having qualified staff that can run standard methodology and standardized processes for the development of IEA.
- From Q6: Consistent with this another difficulty is that organizations do not have the tools, technology resources (HW,SW) to properly utilize the standards for the development of IEA.
- From Q7: In Colombia there are difficulties in accessing qualified human talent formation, standards, methodologies and processes for developing standardized IEA.
- From Q8: There is a percentage is divided in knowledge by the staff of the organizations on the importance of integrating standards and best practices for the development of IEA.
- From Q9: Many of the organizations prefer to outsource the development of the IEA, because of all the above factors, however it is clear that this has led to a technological dependence, which if not managed properly can cause major problems to the organization in the management of their IT resources.
- From Q10: A significant percentage of those surveyed (61%) responsible for the management and implementation of projects of IEA, considered appropriate to develop an agile methodology, simple, efficient and effective for IEA product development.

III. THE DESCRIPTION OF THE AGILE METHODOLOGY FOR DEVELOPING A BUSINESS INFORMATION AND KNOWLEDGE ARCHITECTURE (MADAIKE)

Components MADAIKE described as:

The Business Architecture: It is a layer composed of a set of models that allows visualizing the organization strategic objectives and facilitates decision making related to the integration and development of IT assets [10], [11]. It is based on the paradigm of Enterprise Architect's Model Driven Generation (MDG) Technology, through which we obtain the following models such as: the strategic model, the Balanced Scorecard, Strategy Map, Value Chain, Decision Tree and process model organizational structures (will not be better or better organizational structure based process model organizational structure). The business architecture allows in-depth view of the system context and on this basis to establish the principles that guide the development of the following layers of the business information architecture.

Information Architecture: A layer composed of a set of models to display the organization infrastructure integrated information, partly from conceptual models to reach the physical design of the database, data warehouses and repositories of information, Here are the models [10], [11]: Model of Organizational Information Objects, Logical Data Model (high level Master and Transactional), Logical Data Model (detailed), database design (physical), General Design Model and Data Warehouse Model and design of integrated repositories.

Architecture of Applications: It is a layer composed of a set of models to display the application infrastructure, like the previous layer of conceptual models to reach physical model implementation level, in models describing this layer are: the model of high-level components, the component model and services detailed model of services, applications and software components, the integrated model (services, applications and components sw.) and extended system design or distributed, (production platform).

Infrastructure Architecture: It is a layer composed of a set of models to visualize infrastructure hardware components, devices, connectivity networks that support communication processes, transfer of data, voice and content. Within models and systems that describe these layers are: Model business logistics system, distributed system architecture, Distributed System Design, Architecture technology and network architecture connectivity [9]-[11].

Integrated Planning Project: Is a layer composed of a set of models to visualize the different actions for the integrated planning of projects within them are the following, which are structured in phases of initiation, planning, execution, control and closing so [8], [12].

- Initiation (Define scope, identify stakeholders, Collect Requirements, Create work breakdown),
- Planning (Set of effort and cost estimates, Set Schedule, Set Budget Estimate Investment Risk Manage),
- Execution (Manage the budget, manage the schedule, Manage products, Manage Team),
- Control (Monitor and run, Identify Gaps, take corrective actions) and Close (IEA Project Closure).

Verification: The purpose of Verification (VER) is to ensure that the selected work products meet the specified requirements. Among the methods to carry out the verification process has the following: assessment of software architecture and conformity assessment of implementation, functional decomposition based testing, acceptance testing and continuous integration (e.g. a flexible approach that identifies integration issues in the initial stages).

The verification is oriented to validate each of the products deriving from the application of the methodology on them are considered their syntactic consistency, in this, the correct application of UML notation, BPMn, which describe the models, as well consistency semantics associated with the process model and the representation of a problem, such as relational data model reflects the interactions and organization of the data in the proposed system. As a result of this activity is applied and the following products are obtained: Verification reports, Results of verification, Matrix Cross Reference verification and Corrective Actions for verification.

Validation: The purpose of Validation (VAL) is to demonstrate that a product or product component fulfills its intended use when placed in its intended environment, in this case, when fully deployed for production by organizational level. Among the products tested are taken into account:

- Requirements and product designs and product component.
- Product and product components (e.g. System, hardware units, software and documentation services), user interfaces, user manuals, training materials, process documentation, access protocols, interchange formats reports data.

Examples of validation methods are:

- Discussions with end users, perhaps in the context of a formal review.
- Demonstrations of prototypes.
- Demonstrations functional
- Pilot training materials.
- Testing of products and product components for end users and other relevant stakeholders.
- Analysis of product and product components (e.g. Simulation, modeling, analysis of user).

IV. DESCRIPTION OF LAYERS, PHASES AND ROLES (MADAIKE)

The MADAIKE can be viewed as a multilayer model composed of six layers, in Fig. 2, you can see the multilayer model including four (4) of them framed properly oriented modeling activities, design and development of architecture, business architecture, information architecture, application architecture and Infrastructure Architecture [13], [14]. Plus two supporting layers made up of project planning and verification and validation processes.

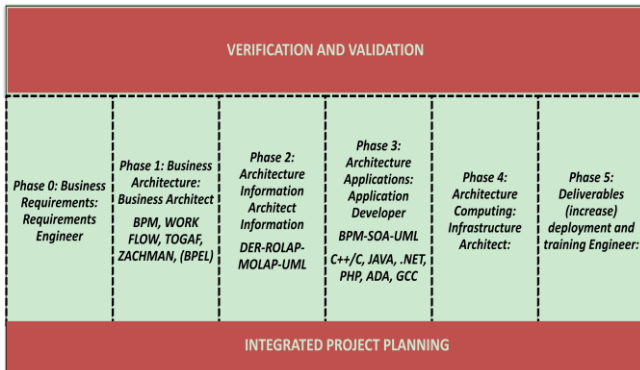


Fig. 2. Multilayer model MADAIKE.

Horizontally, the methodology seeks to represent project phases ranging from the conceptual perspective, logic, implementation, integration and extended. In this way it is structure a methodology composed of layers and phases and giving as a result a matrix model.

In each row and column intercession we see the application of methodological and integration of a standard, represented by the different models, maps, diagrams and representations of the business information architecture.

The methodology can be applied from the perspective of rows or columns, if it is done by rows, the architecture development it's going to initiated by the conceptual phase to the extended phase. On the other hand, if the development is

done by columns, the architecture representation in this case: it goes from the business layer, through the information layer until reach the infrastructure layer.

Important to know, layers of planning and verification are developed that is linear in the case of planning is treated as a single layer, from concept to closing. The same applies to the layer of verification which develop linearly from the verification conceptual architecture to validate the extended architecture.

Phase 0: Business Requirements: Requirements Engineer

Mainly to assume the role of requirements engineer, who is the one responsible to identify, prioritize, structure and grouping of requirements from the interaction with stakeholders and product owners, different techniques can be linked in the activities such as: structured interviews, information analysis, discussion with experts from the business, IT product benchmarking and trend analysis associated with technological product to develop.

Phase 1: Business Architecture: Business Architect

Mainly assumes the role of Business Architect is the one who takes care of modeling, design and business structure seen this as modeling of the structure of processes (BPM), organization, workflow, (WK) weather events (time line), identification of service components (SOA) and business strategic modeling (strategy map).

Phase 2: Architecture Information Architect

Mainly assumes the role of Information Architect is the one who is responsible for identifying and modeling Organizational Information objects (classes, objects, tables), the description of information objects both teachers and transactional (relational data model objects), design database from information objects (physical database), Enterprise Information Integration (EEI), Database, Data Repository, Data Warehouse Design and Repositories Information used in this case comprehensively the DBMS.

Phase 3: Architecture Applications: Application Developer

Mainly it assumes the role of Application Architect is the one who is responsible for developing the architecture applications, the description of high-level software components, the modeling software components and services described the Design Component software and services detailed integration of software components and services detailed and extended system design and component-based distributed software.

Phase 4: Architecture Computing: Infrastructure Architect

Mainly it assumes the role of infrastructure is the architect who is responsible for modeling, design Infrastructure Architecture, the physical layout description of the business, the modeling of the physical distribution business (networking), the design of the physical layout of the business (network), the deployment of physical distribution business (Lan, Man, Wan, Wireless, Pan) and extended physical distribution Design Business-Networking Extend-

Phase 5: Deliverables (increase) deployment and training Engineer

Mainly it assumes the role of deployment and training engineer is the one who is responsible for developing the activities of solution deployment or business information architecture, assumed the task of evangelizing related to the start of production of

the products and services derived from the IEA, in this case, an integrated business requirements, information, applications and infrastructure.

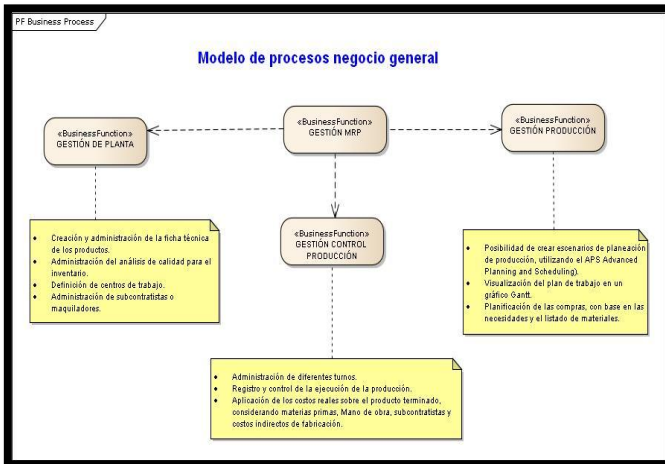


Fig. 3. Business process model general.

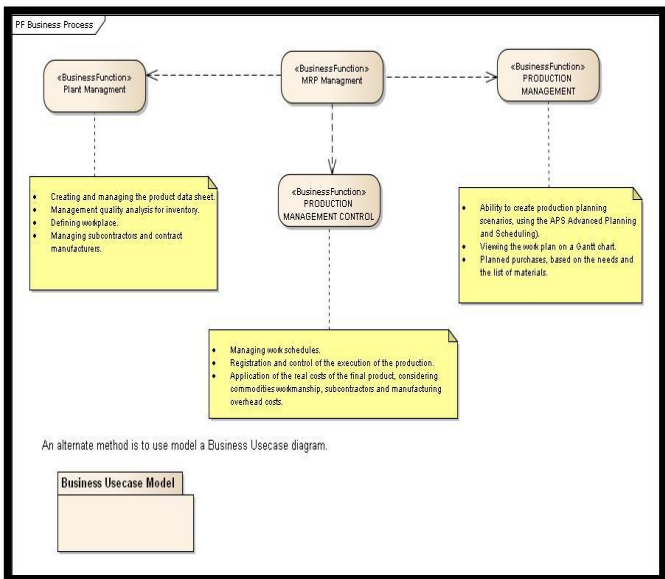


Fig. 4. Plant modeling.

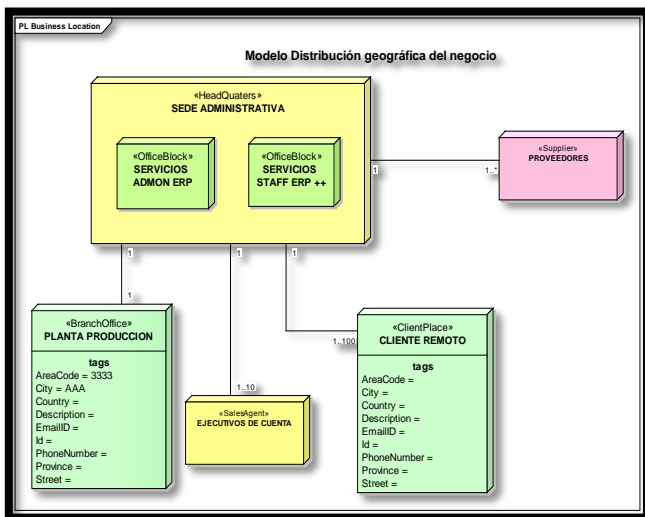


Fig. 5. Geographic distribution business model.

V. DESCRIPTION OF PRODUCTS

The following describes the different products to develop

as a result of the application of MADAIKE well:

Business Architecture, are intended to achieve the following products: Model business processes (Notation BPMn) Organizational Structure Model business (organizational structure), model high-level processes (BPMn Orchestration), Model Geographic Distribution business (model nodes), model work flows (BPMn) Model Integration work flows (BPM) with SOA, Strategy Map (Norton and Kaplan), Value Chain (Michael Porter), BSC (and Kaplan and Norton), Vision, Mission and Business Objectives.

From information architecture, aims to achieve the following products: The object model Organizational Information (UML domain model) Object Model master and transactional information (object relational model), database design from information objects (physical) (database model), the Enterprise Information Integration (EEI), here is the implementation of the data model using the DBMS along with the modeling and design of Data Warehouse and Information Repositories [15], [16].

From Application Architecture, aims to achieve the following products: Software component model (UML components model), model software components and services detailed (internal component model UML), the software component design and detailed services (SOA), the integrated model (services, applications and components sw.), the extended or system design based distributed software components (application framework).

From infrastructure architecture, aims to achieve the following products: The physical distribution model of the business (geographical node model), the model of the physical distribution business (networking) (network diagrams), the networking Design (network design), the design of the physical layout of the business (Lan, Man, Wan, Wireless, Pan) and physical layout design extended Extend Business Networking (backbone networks, transport networks, access networks and networks end user) [17].

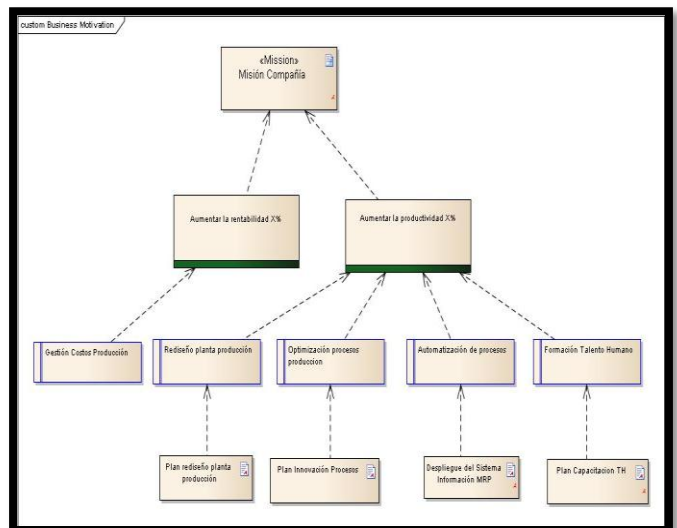


Fig. 6. Motivation custom business.

Integrated Planning, aims to carry out the following activities that allow you to configure the integrated project plan whose structure consists of:

- Initiation: Title of the project, define the scope (EAI project goals), identify stakeholders, Relationship of

Functional and Non-functional requirements, work breakdown description from MADAIKE Methodology

- Planning: Estimates of effort and cost, Schedule Set, Set Budget Estimate and Manage Investment Risk
- Execution: Manage the budget, Manage schedule, manage products, and manage team.
- Control: Monitor and run, Identify Gaps, take corrective actions.
- Close: IEA project.

Some of the representations used in the framework of the methodology shown in Fig. 3-Fig.12.

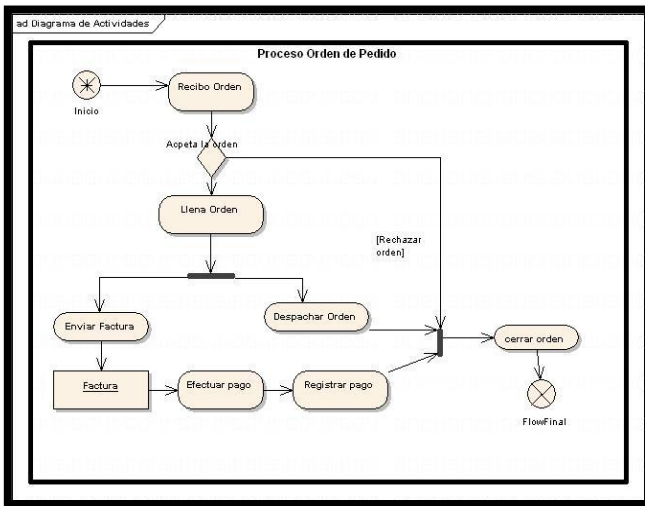


Fig. 7. Activity diagram purchase order process.

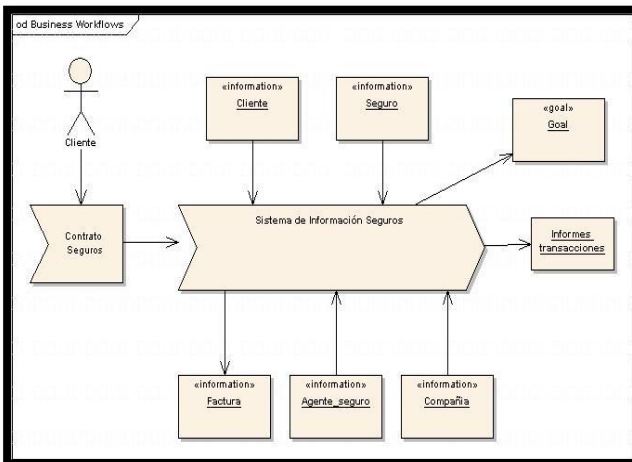


Fig. 8. Business workflow.

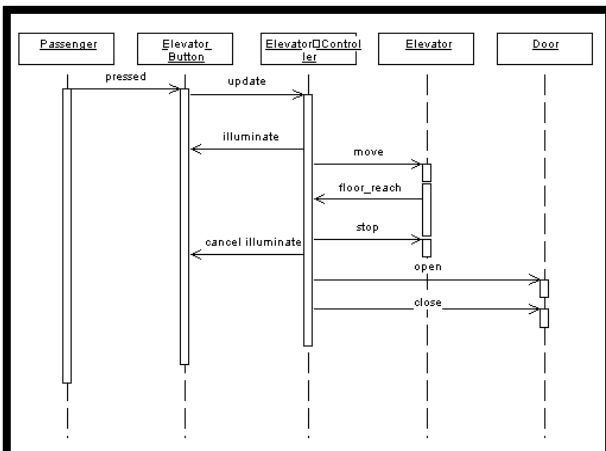


Fig. 9. Sequence diagram.

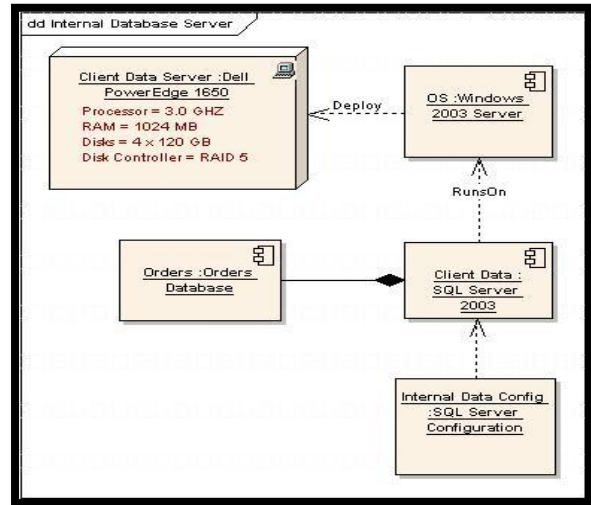


Fig. 10. Internal database server.

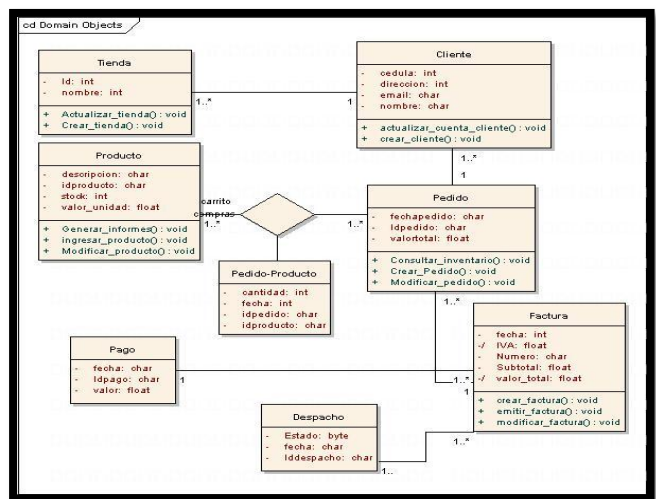


Fig. 11. Domain model.

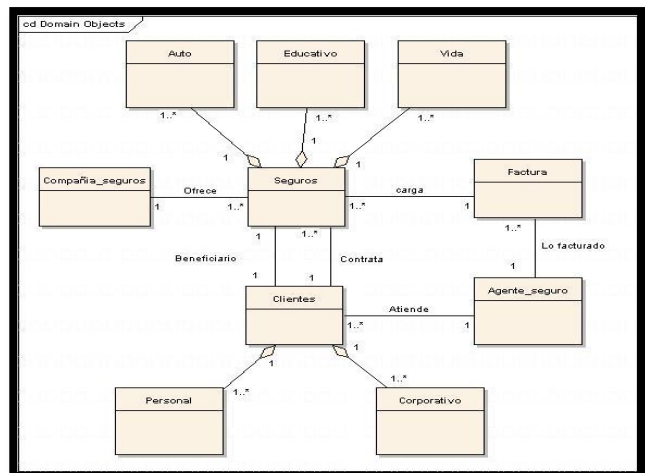


Fig. 12. Domain objects.

VI. CONCLUSION

In this paper we demonstrate that MADAEKI multilayer architecture, you can simplify complex tasks when developing quality software based on agile methods, with reference to processes oriented towards the development of enterprise information architectures which provides a standardized set of tools for project planning.

It is also important to note that the methodology objective

sought to represent project phases ranging from the conceptual perspective, logic, implementation, integration and extended. In this way structure composed of layers methodology and phases resulting in a matrix model.

The methodology MADAEKI can be applied to any project, regardless of size, complexity, additionally can use any tool case that supports UML 2.5 to develop different models and diagrams.

ACKNOWLEDGMENT

Financed Project with resources, “El Patrimonio Autónomo Fondo Nacional de Financiamiento Para La Ciencia, La Tecnología y La Innovación, Francisco José De Caldas”, Proyecto 1215-502-27951 ACESCO-UNIORTE-COLCIENCIAS (Colombia).

REFERENCES

- [1] R. Burlton, *Business Process Management: Profiting From Process*, USA: Sams Publishing, 2001, pp. 45-65.
- [2] R. Chris, *Introduction to Business Architecture*, USA: Cengage Learning, 2010, ch. 2, pp. 11-18.
- [3] C. M. Pereira and P. Sousa, “A method to define an enterprise architecture using the Zachman framework,” in *Proc. the 2004 ACM Symposium on Applied Computing*, 2004, pp. 1366-1371.
- [4] W. Hudson, “Enterprise information architecture: Strategies for the real world,” *Interactions - Bridging the Gap*, vol. 10, no. 6, pp. 53-55, 2003.
- [5] FDI-Intelligence, “FT Business Financial Times,” The FDI report 2012 Global greenfield investment trends, 2012, pp. 2-19.
- [6] P. B. Crosby, *Quality without Tears*, New York: McGraw-Hill, 1984, pp. 95-150.
- [7] F. G. Goethals, M. Snoeck, W. Lemahieu, and J. Vandenbulcke, “Developing the extended enterprise with the FADEE,” in *Proc. the Inf. Syst. Front*, 2006, vol. 8, pp. 67-79.
- [8] G. R. Khoury and S. J. Simoff, *Enterprise Architecture Modeling Using Elastic Metaphors*, in *Proc. the First Asian-Pacific Conference on Conceptual Modeling*, 2004, vol. 31, pp. 65-69.
- [9] *Guide for Unified Profile for DoDAF and MODAF (UPDM)*, Publisher OMG, version 2.1, 2013, ch. 7, pp. 17-25.
- [10] B. C. Nithiya, *MDG Technology for Zachman Framework*, Publisher Sparx Systems Pty Ltd, 2008, pp. 10-43.
- [11] *An Introduction the Open Group Guide for Open Group Standard*, Publisher TOGAF® version 9.1, 2001, ch. 5, pp. 45-56.
- [12] M. Godinez, E. Hechler, K. Koenig, and S. Lockwood, *The Art of Enterprise Information Architecture: A Systems-Based Approach for Unlocking Business Insight*, IBM Press, 2010, ch. 5, pp. 103-144.
- [13] W. Jeanne, W. Peter, and C. David, *Enterprise Architecture as Strategy*, Harvard Business Press, 2006, pp. 25-45.
- [14] PML – The World’S Leading Professional Association for Project Management. [Online]. Available: <http://www.pmi.org>
- [15] SysML Open Source Specification Project. [Online]. Available: <http://www.sysml.org>
- [16] Sparx System. [Online]. Available: <http://www.sparxsystem.com>
- [17] Zachman. [Online]. Available: <http://www.zachman.com>

Wilson Nieto Bernal is a systems engineer and specialist in software engineering, Universidad Industrial de Santander (UIS) Colombia, who is a master/expert in technology management, who received the master of computer degree and PhD degree in computer science from ULPGC-Las Palmas GC, Spain, in 2007, with extensive experience in project management R&D in the area of information technology and applications, knowledge management organizational models for software development.

Carmenza Luna Amaya is an industrial engineer, and a PhD in industrial engineering from the Universidad Polytechnic de Valencia, Spain, with extensive experience in research, development and organizational innovation, currently teaches at the University of the North.