LOGICAL DESIGN OF THE APPLICATION REGARDING FINANCIAL STATEMENTS SUBMISSION

CLAUDIA ISAC, ALIN ISAC, MIRCEA PETRINI *

ABSTRACT: *System outputs are the results of automatic processing, and they represent one of the main means of materialization of the objectives of any computing system. This means providing on-demand or periodic data needed by the real system. Thus, the contents of this paper present the main elements of the logical design of the application that submits eFinance annual financial statements, elements related to designing outputs, to designing inputs, selecting and designing the database, implementing secure transactions with PHP and MySQL.*

KEY WORDS: *e-Finance; secure transactions; elements designing outputs; elements designing inputs; financial statements.*

JEL CLASSIFICATION: *M 48, C81*

1. INTRODUCTION

In terms of the life cycle of an information technology product, the design is a step in the systematic, sequential approach, of software development and according to the phases of these cycles, we know the following models:

- *the waterfall model* in which each stage in the life cycle is evaluated so that it is possible to return to previous stages (an improved version of this model is the V- model, also improved by Hodgson in 1991 as the X model);
- *the incremental build model or the planned improvement of the product* allows a segmentation of the process that leads to the development of the system in a sequence of stages that result in the achievement of functional components;

* *Assoc. Prof., Ph D, University of Petrosani, Romania, isacclaudia@gmail.com*
Lecturer, Ph.D., University of Petrosani, Romania, isacalin@gmail.com
Lecturer, Ph.D. Student, University of Petrosani, Romania, petrini_mircea@yahoo.com

- *the spiral model* is the most realistic approach of application development; it evolves in successive stages and it is based on continuous refinement until the desired system is completed (Figure .1.)
- *the pyramid model* marks the transition from structured techniques used in software engineering applied to a project, to object-oriented structured techniques applied at company level;
- *other models characteristic to certain fields* (the basketball model, the pinball model, the fountain model).

2. e-FINANCE DESIGN METHODOLOGY FOR TRANSMITTING FINANCIAL STATEMENTS

2.1. Designing outputs

A financial statement is a report that must be submitted to the tax authorities, by those established by law and it contains, in principle, a statement of the financial position, of properties or income which result in financial obligations. In order to file a financial statement the taxpayer fills out a form provided freely by the tax and submits it to the registry by mail or electronically. Failure to submit financial statements gives the tax authority the right to determine tax liabilities by estimating the tax base, but not before the taxpayer has been notified that the deadline for filing financial reports has been exceeded.

The main requirements that must be met by system outputs are:

- Output compliance with the legal regulations in financial accounting field;
- providing necessary information requested by companies, on the one hand and institutions that accept statements submitted by them, on the other hand;
- the possibility of obtaining outputs corresponding to previous accounting periods in order to reconstruct some elements;
- providing output flexibility by: selecting facilities, data ordering and grouping, presenting visual data on the screen or in print; the possibility to define situations based on existing data.

The usefulness and viability of the application is determined by the type, the content and the efficiency with which outputs are obtained and result in:

- reports that have different forms and are highly complex;
- graphs of different forms;
- Web site;
- list of values, usually ordered by one or more criteria;
- point values which are obtained instantly (percentages, totals, indices)

In the case of e-Finance, the output data is represented, on the one hand by confirmations received by economic agents and on the other hand by the reports received by the portal administrator.

Economic agents receive a confirmation e-mail each time they submit online statements and after they have been verified and stored on the server.

The administrator has the opportunity to generate the following types of reports:

- the list of companies that have submitted financial statements, fiscal declarations on time;
- the list of companies that haven't submitted financial statements, fiscal declarations on time;

Company name	The situation of submitting financial statements and fiscal declarations
	YES/NO

- the list of companies that have registered profit and its value;

Company name	Companies that registered profit	The value of the profit
	x	

- the list of companies that have registered losses and its value;

Company name	Companies that registered losses	The value of the loss
	x	

- filtering all registered companies by the following criteria: status, registered for VAT, county, city and NACE codes.

Company name	VAT payer	Declaration 300 has/has not been submitted

2.2. Designing inputs

Software inputs are primary data needed to obtain the outputs of the system. Primary, internal or external company information reflects the state and the dynamics of economic phenomena and processes that are necessary for creating, updating the database and obtaining the output statements. The initial information goes through a sequence of stages from its generation and collection until its actual use within the applications, namely:

- recording data from input documents;
- centralization of data and their conversion into a form accepted by the system;
- syntactic and semantic checking of input data;
- erroneous data correction, etc..

In each software component, inputs are subject to outputs on two levels:

- *the logical level*, where every output is the result of applying one or more operators on a set of input data;
- *the technological level*, where the required output characteristics influence the required input characteristics.

In conclusion, the delimitation of input data requires an analysis based on output situations while determining the inputs of the system starting from its output can be possible using the output-input concordance technique (Figure nr.1). The implementation of this process has the disadvantage that not always one can forecast all kinds of output data and all the necessary input information can be determined.

Figure 1. The relation between input data and output reports

Another type of process is the one that determines the entries without knowing, in detail, the information output of the system and it has the disadvantage that some inputs may never be necessary or on the contrary, sufficient to obtain the output. These entries should reflect the real system object in order to be able to obtain any type of information output.

Defining inputs must include all the elements needed for further input documents and it must provide solutions for data acquisition with the computer system. After their determination in terms of content, the source of origin will be established for each input, the frequency, volume, etc. as well as the possibilities and methods of data collection, verification and storage will be agreed upon.

The design of the input aims to develop specifications needed both from the programmer's point of view and from the user's point of view. As a logical design phase, the design of the input includes: *emphasizing input documents, their content, data volume, the frequency of each document, the types of support, data validation criteria that are subject to processing,* etc. (Bătăgan, 2008).

In this case the input data consist of:

- financial statements, ie the balance sheet, the profit and loss account, informatory data;
- fiscal declarations, such as: the Declaration on payments to the state budget, code 14.13.01.99/bs; Declaration on payments to the social insurance budget and special funds, code 14.13.01.40; Income Tax Declaration, code 14.13.01.04, Value added tax return, code 14.13.01.02.

After the first stage of defining the results of automatic processing, and after the second step of dividing the outputs of the system it is required that the technical solution should be determined, ie designing the database.

2.3. Database Design

A properly designed database provides access to accurate, updated information. Since a correct design is essential, it is necessary that duplicate or redundant information be removed as they take up space and increase the likelihood of errors.

In order to create the e-Finance data base, financial statements are collected from companies. as two text files containing accounting information from the balance sheet, the profit and loss account, appendices to the balance sheet the attributes of which are coded and include character and numeric fields. Similarly, for tax returns there are also necessary two text files the structure of which is identical. The input data introduced in the system describes the economic agent, meaning it provides identification information.

Based on this information we developed the information system that contains all the data from the documents mentioned above, common and uncommon, taken only once.

2.4. File Design

The logical design aims at transposing the conceptual model into solutions that can be implemented. Thus, starting from the file, which is a set of data identified as a homogeneous total and from the basic structural unit of the file which represents the logical registration one can define the items as logic units for data processing. In this respect, the logical design involves three subtasks:

- the logical structure of data
- determining the logical characteristics of the data
- estimating the amount of data within the files

This phase aims to outline the solution proposed by the logic design from technical point of view. This is possible if we take into account the characteristics of equipment and the related technical support.

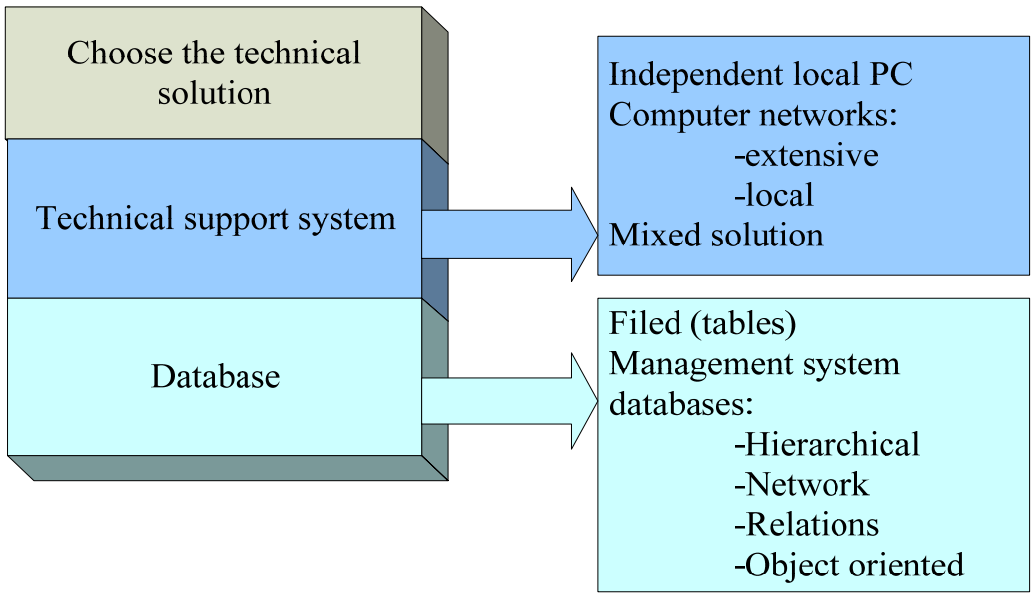

Figure 2. Technical solutions for creating an information processing system

As one can see in the previous figure, there must be a correlation between input data, the actual technical possibilities and information requirements. Due to this fact, the design implies three subtasks: designing models for collecting data, determining physical characteristics of files, determining the necessary technical support for files.

2.5. Interfaces Design

Interface design and implementation is a key element of computer system functionality. After identifying the outputs of the system, a major concern for designers is to create interface models which mean presenting recommendations for structuring and shaping the data entered, following feedback and designing online help systems (Oprea & Mesnita, 2003).

The interface allows users to access the internal components of the system in a relatively simple way; it represents the standard by which one can evaluate the system and the easier the access is, the better the interface can be.

The main components of an interface are:

- *Communication language* that allows interaction with the system in a variety of styles. For example, data communication can be done by switching on the monitor screen to another form, the transition from one field to another and so on.;
- *Presentation language* allows the presentation of data in a variety of formats (reports, tables, graphs, icons, etc.) necessary to transmit information and commands to the computer system.

An interface can be plotted either extended, in which case one uses the interface class symbol and it includes the list of all available operations or synthetic when it is attached to a class or component that supports it (Zaharie & Rosca, 2002).

In addition to this structure, there are the following types of interfaces:

- interfaces using natural language;
- question and answer interfaces;
- video input / output interfaces;
- command-line interfaces;
- interfaces that allow direct manipulation.

From the user's point of view, interfaces can be structured as follows:

- GUI (Graphical User Interface) that supports inputs from devices such as keyboard and mouse sites and it provides a detailed graphic output on the monitor;
- Web-based interfaces or advanced interfaces which accept inputs and provide outputs by generating web sites that are transmitted via the web and viewed by users through browsers.

In the application presented I combined both interfaces: the video input / output interface and web-based interfaces. Users will be able to choose the desired menu from the header or from the left side and its execution will be displayed in the center of the page. The interface used is shown in Figure 3 and it is executed by the launch module page.php.

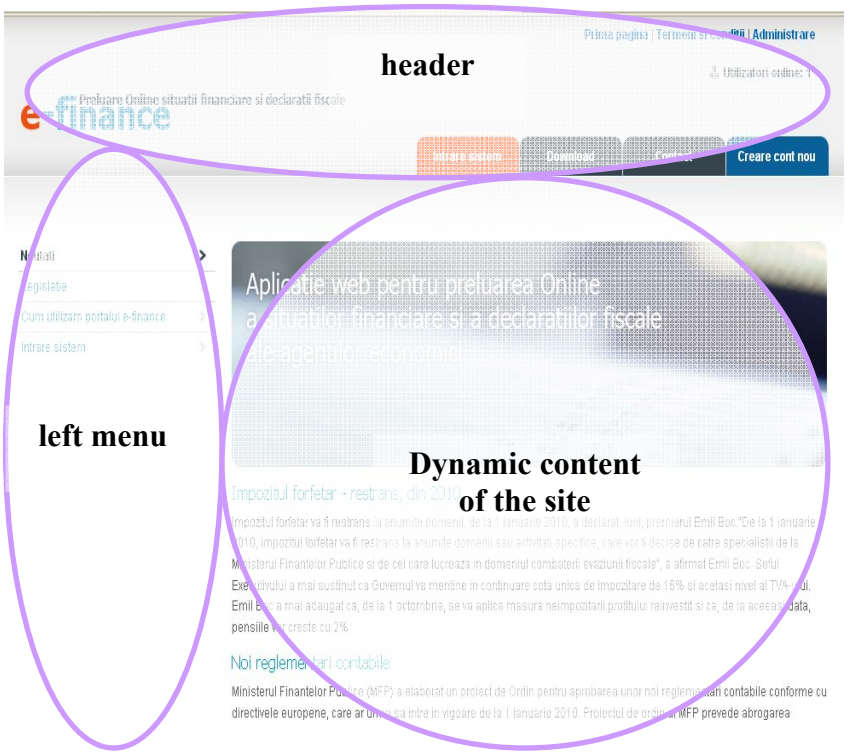

Figure 3. The general interface of the application

3. CONCLUSIONS

In conclusion, interface allows users to access the internal components of the system in a relatively simple way; it represents the standard by which one can evaluate the system and the easier the access is, the better the interface can be. When implementing the e-Finance application we used the SSL technology. The user's access is based on the tax identification code and the password chosen at the registration of the company in the database. It also uses protection against the introduction of source codes in the login fields.

REFERENCES:

[1]. Bătăgan, L. (2008) *Virtual Places,* Economic Publishing House, Bucharest, pp.67-69

[2]. Balog, A.; Ivan, I. (2006) *Researches on the Quality of Public Online Services*, The Informatics and Automatics Review, 16(4), pp.39-48

[3]. Balog, A. (2008) *Methodological Aspects Regarding the Development of the Model Measuring the Quality of Public Online Services,* The Informatics and Automatics Review, 18(2), pp.15-25. http://www.miteapl.ro/index.php

[4]. Belu, M.; Paraschiv, D.; Comănescu, A.M. (2004) *Tranzacţii pe Internet*, Editura Economică, Bucureşti

[5]. Ghilic-Micu B.; Stoica, M. (2006) *Economia digitală şi societatea bazată pe informaţie şi cunoaştere*, în volumul *Societatea cunoaşterii*, coord. Roşca Gh.I., Editura Economică, Bucureşti, pp.119-121

[6]. Isac, A. (2010) *Urmărirea situațiilor financiare şi a obligațiilor fiscale ale agenților economici pe Internet*, Teză de doctorat, Craiova

[7]. Lucey, T. (2005) *Management Information Systems*, Published by Cengage Learning EMEA

[8]. Oprea, D., Meşniță, G. (2003) *Information Processing Systems for Managers*, Polirom Publishing House, Iaşi, pp.208-210

[9]. Radu, I., Ursăcescu, M., Ioniță, F. (2002) *Computer Science for the Company Management,* Tribuna Economică Publishing House, Bucureşti, pp.108-110

[10]. Zaharie D., Roşca I. (2002), *Object-Oriented Design of Computer Systems*, Dual Tech Publishing House, Bucureşti.