Stan: A dynamic web-based application for financial statement analysis

Adam P. Filippidis
Information Management Department, School of Management and Economics,
Kavala Institute of Technology, Greece,
adfilip@im.teikav.edu.gr

Kosmas Kosmidis
Information Management Department, School of Management and Economics,
Kavala Institute of Technology, Greece,
kosmidis@teikav.edu.gr

Vassilios Mardiris
Information Management Department, School of Management and Economics,
Kavala Institute of Technology, Greece,
mardiris@teikav.edu.gr

Abstract
In a globalized ever-changing world, financial analysis systems should be flexible and extensible in terms of accounting standards, financial expertise and languages. In this paper a new web-based tool for financial statement analysis is proposed which exhibits all the preceding attributes. The proposed tool named STAN (Statement Analysis) is a fully automated, multilingual, flexible and extendable platform, and it can be efficiently used even by non-financial experts. The core system can perform a thorough financial statement analysis by employing Trend Indicators, Vertical analysis and financial ratios simultaneously. Data from different accounting standards can be processed and the resulting financial ratios are presented as plain text based on decision tree analysis. The rules and the text of the decision tree are fully customizable from the user interface. The application is developed using exclusively free and open source software tools. The reliability of the results of the application is examined in a case study analysis of a well-known enterprise which reveals its advantages – disadvantages. Topics for future research are also proposed.

Keywords: Financial statement analysis, STAN, web-based application, Decision tree analysis.

JEL classifications: G32, G33, M49, C88

Introduction

We live in a globalized ever changing world which is dominated by multiple complexity factors. This environment makes the economic system even more chaotic. In such conditions financial experts and managers who have the need of financial information must adapt to the new economic situations. Analysis of financial statements using financial ratios (or accounting ratios) is the classical approach for many years. They are used by managers within a firm, by current and potential shareholders and by company's creditors. Furthermore, financial analysts use financial ratios to compare the strengths and weaknesses in various companies (Groppelli and Nikbakht, 2000).

In addition to ratio analysis, there are other classical approaches like trend indicators and vertical analysis. With the use of trend indicators researchers attempt to spot patterns to financial statements historically. Trend indicators (or trend analysis) are used to predict decisions concerning future events. Vertical analysis is a method according to which the balance sheet is transformed as a proportion of the total assets. The main advantage of this method is that businesses of different size can be compared. Financial ratios, vertical analysis and trend analysis are the classic methods used by researchers for many years. By combining these three methods an analyst can perform a thorough financial analysis including both time-series and cross-sectional analysis (Foster, 1986).

Even nowadays where researchers adopt more sophisticated approaches from the computer science field, financial ratios are widely used. Examples of such sophisticated methods in the field of data mining are the method of clustering, association rules and decision making trees (Yanhong et al., 2006). In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data but not decisions; rather the resulting classification tree can be an input for decision making (Jedrzejowicz et al., 2011).

An attempt to create a web-based tool to computerize financial analysis and diagnose is performed by Pavaloaia and Cuza (2009). The use of decision tree analysis in a normative approach has been proposed by Kosmidis et al. (2010) in a web-based analysis tool development, not just to classify companies but also to extract comprehensible conclusions to managers and other decision makers with no financial expertise.

In this paper a new approach for a web-based dynamic decision-tree financial analysis tool development, named STAN, is presented. STAN, acronym stands for STatement Analysis, was developed by using a mix of agile and object oriented computer programming techniques. The main advantage of the proposed tool in contrast to the work of kosmidis et al. (2010) is that it provides to the analyst the capability to dynamically setup his own decision tree by setting, the decision rules, the resulting text for every node and the depth of the tree. One of the main advantages of STAN is its modular and flexible design. STAN employs the same evolutionary data management techniques, as Content Management Systems (CMS) to create a similar structure for financial statements data (Rockley et al., 2003). Additionally at the proposed approach the core system is not limited by local accounting standards but it can easily support unlimited accounting standards. The core system can perform a thorough financial statement analysis by employing trend indicators, vertical analysis and financial ratio analysis simultaneously. STAN tool architecture is based on several plugins, making it very flexible and extensible to future improvements. Finally STAN is multilingual in order to facilitate financial analysis report extraction for different country companies.

Methodology

The analysis of financial statements according to the position of the analyst and the stages performed is distinguished as follows:

1 In the internal analysis carried out by persons of the company that have direct access to its books and accounts and

2 In the external analysis conducted by persons outside the business, where the analyst has access to financial information available exclusively from published financial statements.

Obviously, the latter analysis is much more difficult than the former due to the scarcity of publicly available financial information. The main objective of this paper is to construct a web based tool for non financial experts in order to perform external financial analysis. However, one of the main obstacles an external analyst has to deal with is the volatility of the predictive power and information content of financial ratios among different market, periods of time and economies (Altman et al., 1977; Altman, 1984; Johnsen and Melicher, 1994).

The decision making process of financial analysts varies depending on the models they use and their financial expertise. In that sense, the decision of a financial expert can be based on different criteria-financial ratios than the decision of anyone else or even if these criteria are the same, their gravity-weights are different. Consequently, STAN is designed in a way to fulfill the preceding requirement. The decision making process of STAN is normative (Whittington, 1980) and it is based on a decision tree where each node is a financial ratio. The rule-benchmark for each node-financial ratio can be either proposed by STAN or be inserted by the user. If the calculated ratio of a particular company exceeds the benchmark, STAN reports a complimentary comment about the performance of the company and vice versa.

The final report of the company is based on a number of decision trees. STAN proposes five decision trees, one for each ratio category (liquidity, activity, profitability, viability and investment) but the number of decision trees can be altered by the user. Moreover, STAN proposes the financial ratios that should be used in the nodes of the decision tree but that can be altered by the user as well as their order in it. In addition, STAN proposes rules-benchmarks for each financial ratio but due to their non stationarity, the user has access to update them. In our implementation the number of decision trees, the rule used for each node, the depth of the decision trees and the acceptance/rejection text are fully dynamic.

The software tool proposed at the present paper acquires company data from published financial statements and calculates the most significant financial ratios (Table 1). However the tool is not limited to the calculation of the preceding ratios, but also provides an automated assessment report – financial analysis. The generated report could certainly be useful for non-financial experts but it may also be helpful subsidiary tool for financial experts. The report is a text document including a number of statements which are resulting from a decision tree analysis based on the above calculated ratios.

In order for the analysis tool to automatically produce the text report, a new algorithm is designed and implemented. The proposed algorithm includes six steps for every decision tree that the user decides to adopt for his analysis. A decision tree is a canonical structure of nodes and branches. Every node corresponds to a decision rule. According to the node's rule (benchmark), the decision can be true or false, which is leading the decision flow to different nodes in the next layer. According to this scheme, the number of nodes in a layer is given by the equation $2n-1$ where n is the number of the

layer. The total number of nodes is 2n-1. Every node is connected with a branch to its parent node in the previous layer and with two branches to its child nodes in the next layer. The node of the first layer has only child branches and the nodes of the last layer have only parent branches. Layers are numbered from 1 to n and nodes are numbered from 1 to 2n-1. Every node is corresponding to a financial ratio, to a benchmark rule and to a text that the node produces when it is selected. The algorithm's six steps are described as follows.

Step 1. Set Layer to value 1 and Node to value 1.
Step 2. Check if financial ratio of the company corresponding to the current Node matches the rule the analyst has set.
Step 3. Set Layer to value Layer+1.
Step 4. If the check result is true, then set Node to value 2*Node. If the check result is false, then set Node to value 2*Node+1.
Step 5. Produce the text corresponding to the current Node.
Step 6. If current Layer is the last finish, else go to Step 2.

One of the main goals of the proposed web based tool is the evaluation of a company based on results of specific financial ratios, such as those resulting from the published financial statements. Table 1 presents a selected set of standard financial ratios and its formulas which are used by the proposed software tool.

Table 1: Financial ratios Formulas

Ratio	Formula
Current Ratio	Current Assets / Current Liabilities
Quick Ratio	(Current Assets – Inventory)/ Current Liabilities
Defensive Interval Ratio	(Current Assets–inventory)/ Daily Operating Expenses
Receivables Turnover Ratio	Receivables / Annual Sales X 360
Inventory Turnover Ratio	Inventory / Cost of Goods Sold X 360
Trade Creditors Turnover Ratio	Trade Creditors/Cost of Goods Sold–depreciation X360
Equity Turnover Ratio	Annual Sales / Equity
Fixed Assets Turnover Ratio	Annual Sales / Fixes Assets
Total Assets Turnover Ratio	Annual sales / Total Assets
Sales Growth Ratio	(Annual Sales(t)–Annual Sales(t-1))/Annual Sales(t-1)
Gross Margin Ratio	Gross Profit / Annual Sales
Net Margin Ratio	Earnings before Taxes / Annual Sales
Return on Assets Ratio (ROA)	Earnings before Taxes / Total Assets
Return on Equity (ROE)	Earnings before Taxes / Equity
Return on Capital Employed (ROCE)	Earnings before Taxes +Interest paid/capital employed
Financial Leverage Ratio	ROE / ROCE
Times Interest Earned	Earnings before Taxes / Interest Paid
Solvency Ratio	Total Liabilities / Total Assets
Debt Equity Ratio	Total Liabilities / Equity
Long Term Debt Equity Ratio	Non-Current Liabilities / Equity
Equity Debt Ratio	Equity / Total Liabilities
Fixed Assets Total Assets Ratio	Fixed Assets / Total Assets
Long Term Capital Fixed Assets Ratio	Equity + non-Current Liabilities / Fixed Assets
Earnings per Share	Net Income – Dividends on Preferred Stock / Average Outstanding Shares
Dividend per Share	Dividends / Number of Shares
Dividend Yield	Dividends to equity
Payout Ratio	Dividends / Earnings Before Taxes
Dividend Equity Ratio	Dividends / Equity
Price Earnings Ratio	Market value / Earnings
Market to Book Value	Market value / Book Value X 100

Software Implementation

The software implementation of the proposed tool is based on an information system using extensible architecture. The block diagram of the system's architecture is presented in figure 1.
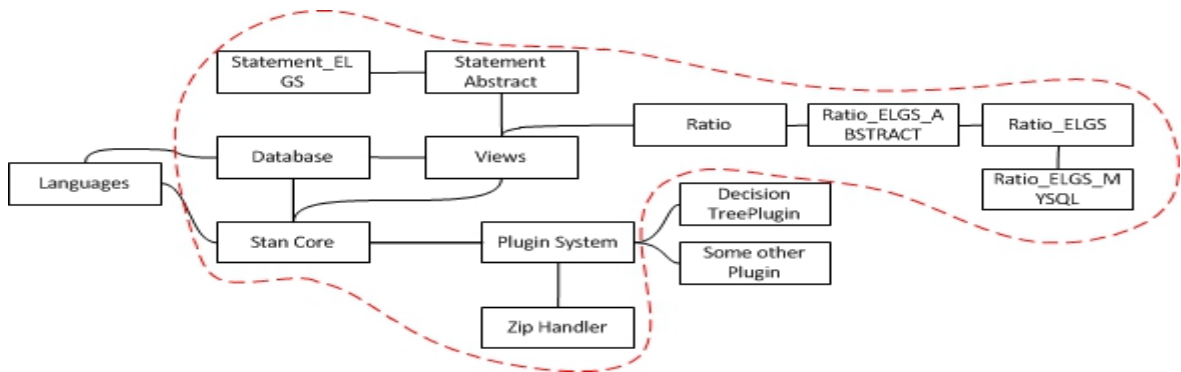
Figure 1: The block diagram of the system's architecture

The system's data is applied on a relational database. The database structure is organized as follows:
- Each company, described by (name, city, address, national identification, description) has statements described by (date).
- Each statement has entries described by (amount.
- Furthermore, there are accounting standards described by (standard name, description).
- Each standard has categories described by (parent_id, name, position.
- And each category has entries in a financial statement described by (amount.
- Decision tree table described by (name, description) is consisted of tree elements described by (branch_id, ratio_id, rule, position, rule_true, rule_false). The brach_id is the number of the branch in the current tree layer (position). Rule_true and rule_false are the text that will be produced if the rule for the specified ratio is achieved by the company.
- The Ratio_name table described by (name) contains the names of financial ratios.
- The Ratio_settings table described by (option) is used to map categories of a standard and is needed for financial ratio fractions to work properly for different standards.

A block diagram of the database structure is presented in figure 2:
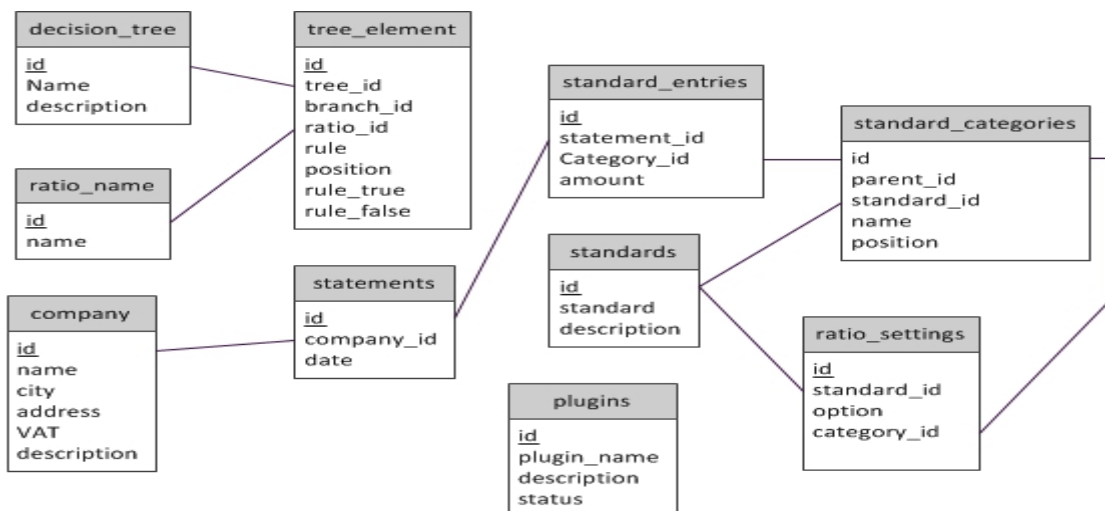
Figure 2: The object relational diagram of the database

The folder structure of the tool's implementation (figure 3) consists the following:

- **Plugins folder:** in the plugins folder the plugins loaded in the system are exist. Each plugin has a folder with its name and inside a miniature of the whole system exist. Each plugin that is active on the system is loaded automatically in the headers, and the menu sidebar. Furthermore languages are automatically loaded based on file structure logic inside the plugin.
- **Languages folder:** the languages folder contains the language files that are existing in the system. All languages share a common template.
- **JavaScript folder:** This folder contains the JavaScript files that are used from the system.
- **CSS folder:** The CSS folder maintains all the Cascading Style Sheets, used for better data representation.
- **Includes folder:** Contains the functions, views and classes of the system
- **Functions folder:** contains functions not yet been incorporated into classes
- **Classes folder: con**tains all the classes that are used from the core system.
- **Views folder: con**tains all the views of the system.
- **Header views folder:** this folder contains custom headers are needed by views.
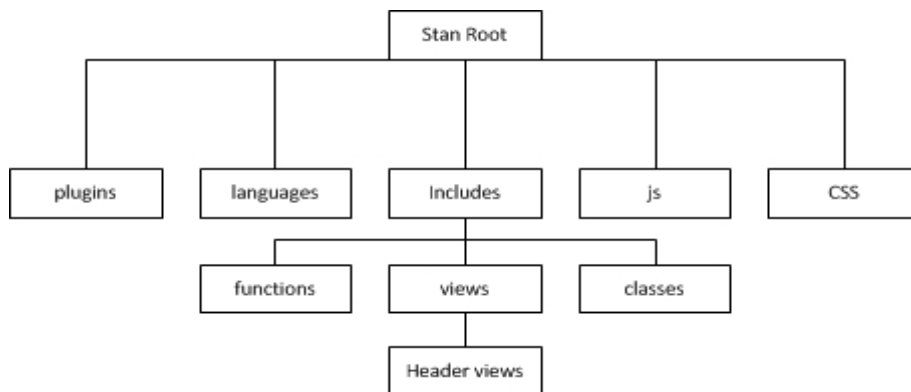
Figure 3: The implementation folder structure block diagram of the proposed tool

The web application is developed using exclusively free software tools and software open source code. HTML language (HyperText Markup Language) (Powell, 2003) is used to design user interface, PHP programming language (PHP: Hypertext Preprocessor) was used as the main programming tool for the implementation of dynamic application functionality (Hughes and Zmievski, 2001) and MySQL is used as the required relational database management system (Welling and Thomson, 2005). For better visualization of the user interface the Cascading Style Sheets (CSS) was used (Bos, 2011).

For the system to be easy for researchers to add or improve functionality object-oriented design and auto-plugin-insert capability has been adopted. For example to support a different accounting

standard or financial ratios, the researcher has just to inherit a new statement and ratio class for the specific standard and use the methods that exist in the system. Furthermore plugins are automatically installed in the system by applying a zip archive form to the system, which contains a proper folder structure and an installation file in the root of the zip with proper instructions. Plugins have two states, active and inactive. Plugins with inactive status continue to exist in the system but their use is disabled.

Results and discussion

The main operations a user can perform in the system are the following:

- **Manage a company**: Here the user can add a new company or edit an existed one.
- **Manage Balance sheet**: The user can add a new balance sheet for a specific company, a specific accounting standard for the company for the specified date. Furthermore, the user can show the balance sheets that exist for an existing company.
- **Accounting standards**: The user can fully manage different accounting standards which imply different names and categories of the financial statements structure.
- **Statement analysis**: Here the user can perform financial ratio, trend indicators and vertical analysis. Furthermore, ratio settings are managed for the specific standard. The output of financial ratio analysis for a case study company is shown in Table 2.

Table 2: Output of financial ratios for a case study company

Ratio	2011-12-31	2010-12-31	2009-12-31	2008-12-31	2007-12-31
Current Ratio	154.57%	107.61%	51.36%	30.5%	117.41%
Quick Ratio	49.37%	52.73%	25.69%	27.67%	117.41%
Defensive Interval Ratio	226	414	2187	6433	279965
Receivables Turnover Ratio	76	100	295	0	0
Inventory Turnover Ratio	467	345	3239	0	0
Trade Creditors Turnover Ratio	191	430	4562	0	-257455
Equity Turnover Ratio	57.42%	55.51%	4.47%	0%	0%
Fixed Assets Turnover Ratio	58.29%	36.38%	2.31%	0%	0%
Total Assets Turnover Ratio	30.47%	22.33%	1.73%	0%	0%
Sales Growth Ratio	43.58%	1314.04%	0%	0%	0%
Gross Margin Ratio	24.56%	20.85%	33.52%	0%	0%
Net Margin Ratio	0.15%	0.29%	9.12%	0%	0%
Return on Assets Ratio	0.05%	0.07%	0.16%	-0.95%	-0.1%
Return on Equity	0.09%	0.16%	0.41%	-3.53%	-0.16%
Return on Capital Employed	2.62%	0.16%	0.2%	-0.95%	-0.1%
Financial Leverage Ratio	3.44%	100%	205%	371.58%	160%
Times Interest Earned	0	1	3	0	0
Solvency Ratio	46.94%	59.77%	61.22%	73.1%	35.25%
Debt Equity Ratio	88.47%	148.57%	157.88%	271.72%	54.43%
Long Term Debt Equity Ratio	35.11%	71.92%	53.76%	72.51%	0%
Equity Debt Ratio	113.03%	67.31%	63.34%	36.8%	183.73%
Fixed Assets Total Assets Ratio	52.27%	61.38%	75.01%	81.58%	55.55%
Long Term Capital Fixed Assets Ratio	137.16%	112.68%	79.49%	56.89%	116.58%
Earnings per Share	0.07€	0.09€	0.2€	-1.02€	-0.05€
Divided per Share	0%	0%	0%	0%	0%
Divided Yield	0%	0%	0%	0%	0%
Payout Ratio	0%	0%	0%	0%	0%
Divided Equity Ratio	0%	0%	0%	0%	0%
Price Earnings Ratio	0%	0%	0%	0%	0%
Book Value	78.15€	56.3€	49.47€	28.8€	29.64€
Market to Book Value	0€	0€	0€	0€	0€

- **Plugins:** In the plugins section the user can add a new plugin in the system, or view the existing plugins.
- **Decision Trees: T**his is the first plugin of the system. Here the user can do three actions. Create and manage a decision tree, create and manage the different layer of a tree and extract text results for the last year financial ratios of a company based on one or more decision trees. An illustration of how managing tree elements look like is embedded in figure 4.
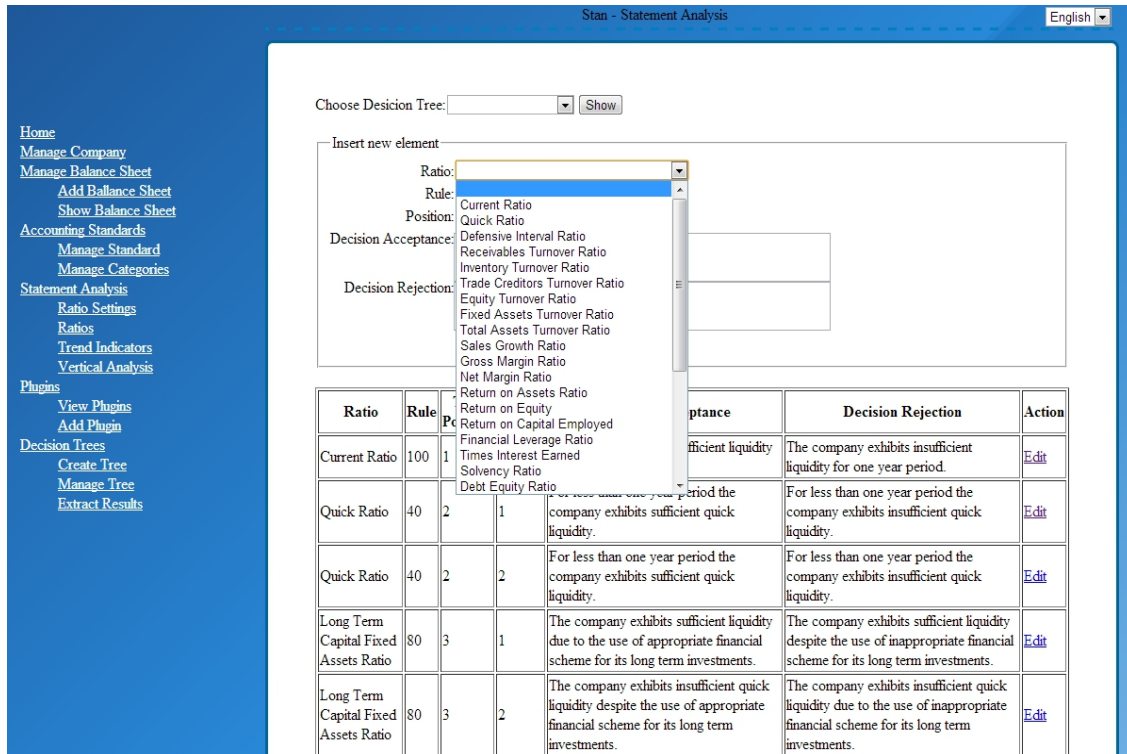
Figure 4: Illustration of tree elements management

Below in table 2 we can see an illustration of a decision tree with 4 layers along with the financial report concerning the liquidity of a case study company. The decision tree is consisted of Current Ratio in the first layer, Quick Ratio in the second, Long Term Capital to Fixed Assets Ratio in the third and Inventory Turnover Ratio in the fourth layer.

Table 2: Illustration of the financial report of a case study company

Company Ratio	Our Benchmark	Text Produced
154.57%	150	The company exhibits sufficient liquidity for one year period.
49.37%	100	For less than one year period the company exhibits insufficient quick liquidity.
137.16%	125	The company exhibits insufficient quick liquidity despite the use of appropriate financial scheme for its long term investments.
467	100	The company maintains high inventory level with inefficient working capital management and obvious quick liquidity problems.

Conclusions

In this paper a new approach for a web-based dynamic decision-tree financial analysis tool development. The proposed tool provides to the analyst the capability to dynamically setup his own decision tree by setting, the decision rules, the resulting text for every node and the

depth of the tree. The proposed tool is not limited by local accounting standards but it can easily support unlimited accounting standards. The core system can perform a thorough financial statement analysis by employing trend indicators, vertical analysis and financial ratio analysis simultaneously. The architecture of the tool is based on several plugins, making it very flexible and extensible to future improvements. Finally STAN is multilingual in order to facilitate financial analysis report extraction for different country companies.

The proposed tool is helping managers to extract information from financial statements without prior knowledge of financial statement analysis or any other financial expertise. It is easier for financial analysts to extract conclusions using this tool, because the work of the analysis is minimized to data entry of the financial statements, and configuration of the accounting standards used and the decision tree benchmarks.

The system can work even as a core framework for further research in which more methodologies could be implemented as plugins like z-score to predict a company's bankruptcy (Altman, 1968) or combined techniques to detect falsified financial statements.

References

Altman, E. (1968), "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *Journal of Finance*, **23**, 589-609

Altman, E. Haldeman, R. & Narayanan, P. (1977), "Zeta Analysis: A new model to identify bankruptcy risk of corporations," *Journal of Banking and Finance*, **1**, 29-54

Altman, E. (1984), "The success of business failure prediction models: An International Survey," *Journal of Banking and Finance*, **8**, 171-198

Bos, B. (2011), Descriptions of all CSS specifications, World Wide Web Consortium

Foster, G. (1986), Financial statement Analysis (2nd ed.), Prentice Hall International Editions

Groppelli, A.A. and E. Nikbakht, 2006, Finance (5th ed.), Barron's Educational Series

Groppelli, A.A. & Nikbakht, E. (2000), Finance (4th ed.), Barron's Educational Series

Hughes, S. & Zmievski, A. (2001), PHP Developer's Cookbook (2nd ed.), SAMS

Johnsen, T. & Melicher, R. (1994), "Predicting corporate bankruptcy and financial distress: Information value added by multinomial logit models," *Journal of Economics and Business*, **46**, 269-286

Kosmidis, K. Michailidou, A.E. Ntaliani, A. & Chatzis, V. (2010), "Web-based application for financial statement analysis: A normative approach," *Prime International Journal*, 3(1), 114-127

Jedrzejowicz, P. Nguyen, N.T. & Hoang, K. (2011), "Computational Collective Intelligence: Technologies and Applications", Proceedings of the Third International Conference, ICCCI 2011, Gdynia, Poland

Powell, T. (2003), HTML & XHTML: The Complete Reference (4th ed.), McGraw Hill Osborne

Rockley, A. Kostur, P. & Manning, S. (2003), Managing Enterprise Content: A Unified Content Strategy, New Riders

Pavaloaia, V.D. & Cuza, A.I. (2009), "Web Based Application for SMEs Economic and Financial Diagnose," *Communications of the IBIMA*, **9**(4), 24-30

Welling, L. & Thomson, L. (2005), PHP and MySQL Web Development (3rd Ed.), SAMS

Whittington, G. (1980), "Some basic properties of accounting ratios," *Journal of Business Finance and Accounting*, **7**(2), 219-232

Yanhong, L. Peng, L. & Zheng, Q. (2006), "An Analysis Model of Financial Statements Based on Data Mining," 3rd International IEEE Conference on Intelligent Systems, London, 847-850, doi: 10.1109/IS.2006.348531