Journal of Theoretical and Applied Information Technology
30th April 2019. Vol.97. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

LTL FORMULA PATTERNS FOR ENHANCING THE PERFORMANCE ANALYSIS OF E-BUSINESS STRUCTURE

AYMAN MOHAMED MOSTAFA, JABER ALROUDHAN, MOHAMED. M. HASSAN

Faculty of Computers and Information, Zagazig University, Egypt
E-mail: am_mostafa@zu.edu.eg

ABSTRACT

One of the major significant trends in Internet technology is the use of E-business applications for conducting organizational resources. The structure of E-business applications can enhance the analysis performance based on customer preferences. Model checking analysis and linear temporal logic (LTL) provide formula patterns for obtaining better query of user requirements. These patterns are suitable for infinite traces of customer preferences. In this paper, an enhanced framework of E-business application based on B2C and G2C is presented. A web server log is generated and optimized to record infinite traces of event actions and applying them to a set of finite E-business processes. An enhanced set of mathematical LTL formula patterns are applied to the finite traces on the web server log for enhancing the performance analysis and predicting users' behavior. To this end, we added new features of linear temporal logic (LTL) for classifying and enhancing the analysis processes and achieving better prediction of real-life event logs. A pattern analysis process is applied to evaluate the presented LTL formula patterns for enhancing the prediction of user behavior in web server event logs.

Keywords: *E-Business, Linear Temporal Logic, Web Server Log, And User Behavior*

1. INTRODUCTION

E-business is a set of tools, processes and applications that help organizations and companies in using Internet and communication technology [1]. There are many types of E-business structures based on the user or partner participation. The major partners of E-business are government, consumer, and business with exchanging each partner with other partners. The engineering of E-business involves applying technologies and models for improving E-business web services. As presented in [2], the web service quality is a major concern in protecting and maintaining large amount of information. In addition, there is a lack of control and governance over E-business web services that lead to incomplete and inconsistent data and information. Despite different E-commerce and E-business frameworks and models have been presented, there is a lack of implementing and applying these frameworks [3]. In addition, existing frameworks are utilized only in private or closed environments that lead to a limited number of consumers [2]. Securing customer information is another major concern in controlling and managing E-business web servicers. As presented in [4] a digital signature model is applied on E-business application for obtaining integrity, confidentiality, and authenticity of user actions.

The providers of E-business services provide a quality and efficiency to web services by investing a large amount of financial resources in E-business technology like web service design, graphical user interface, security and privacy, and web server log performance. In this research, a framework is needed for measuring the E-business performance [5]. As presented in [6], a comparative study for measuring students' attitude to an E-business application is applied. This research aims to dividing the students into business and engineering groups to measure the degree of their interaction with the E-business application under the study. The contributions of this paper are as follows:

- We proposed building a B2C and G2C subsystems for E-business processes.
- An E-business framework for protecting user transmissions for both B2C and G2C subsystems is proposed.
- A structured web server log for recording the viewing and transaction operations in user sessions is developed.

- We proposed analyzing users' behavior by enhanced set of mathematical formula patterns based on linear temporal logic (LTL) and Declare patterns.
- A pattern analysis process is proposed for evaluating and predicting user behavior.

2. RELATED WORK

For improving the design of E-commerce and E-business web services, a web usage mining is presented [7]. In this research, a pre-processing of data, mining of data and data analysis are applied to an E-commerce web site called "OrOliveSur". In this web site, a K-means clustering technique and an Apriori algorithm of association rule mining technique are applied to the collected data set to measure the significance, unusualness, sensitivity, and fuzzy confidence.

A method for discovering different clusters of E-commerce interest patterns are presented in [8]. In this research, an approach for measuring the frequency of page visits and the time spent in each product category is applied. Based on this approach, the user's behavior of the web service structure can be analyzed and measured.

As presented in [9], different mining operations are explained for building user sessions from web server log. The web mining is classified into three categories: web content mining, web structure mining, and web usage mining. Web content mining is based on using the content of web application for mining processes. Web structure mining identifies the structure of the hyperlinks for mining processes while web usage mining is based on mining the records of the web server log for predicting user behavior.

The authors of [10] presented a framework for selecting sample size of E-business application data. In this research, a two-fold sample size is selected for measuring the degree of trust to all E-business applications. One of the recent researches for analyzing users' behavior in E-business and E-commerce web sites are presented in [11]. The authors of this research developed a web server log for storing user tracing records. A data preprocessing is applied to a ready-made E-commerce web site for identifying users' behavior patterns and considering different actions performed in user sessions.

The increasing availability of data recorded into the event logs of E-business applications makes the analysis, mining, and prediction of user behavior a complex task [12]. Each event recorded into the event log is called an activity and is related to a specific action or process in the E-business application. Additional information is stored into the event log such as timestamp of the process, the user who initiating the process, and browsing history. This information is easy to be recorded but hard to be processed and analyzed.

Declare patterns consist of a set of constraints called templates. Each template has a specific semantic through LTL formulas that should obtain a front-end graphical user interface and must be understandable by users. Each LTL mathematical formula can specify certain constraint based on the ordering of activities [13]. Declare patterns provide an obvious front-end translation to the formal back-end LTL formula patterns.

The effectiveness and interoperability should also be maintained in recent E-business applications. One of the recent researches for reviewing E-business interoperability frameworks are presented in [14]. The authors of that research define the interoperability as the ability of information and business processes to be exchanged. Different interoperability levels are explained in that research in regard to different E-business frameworks such as application and software framework, ATHENA framework, enterprise framework, and grid wise context setting framework. The paper concludes that an E-business interoperability framework should be simple, understandable, should be enhanced easily, and must ensure consistency and avoid redundancy.

3. FRAMEWORK METHODOLOGY

The E-business framework is divided into two main processes. Firstly: the G2C and B2C user transactions. Secondly: the authentication mechanism for securing both inner user operations and electronic payment transmissions.

As presented in Figure 1, the overall E-business framework is based on government-to-consumer (G2C) and business-to-consumer (B2C). The G2C subsystem presents a set of governmental services embedded into the overall system. The presented governmental services are the payment of electricity, water, Internet, car violations, school fees, and flight reservation. In

G2C, each customer is considered as an administrator of the subsystem.

The B2C subsystem presents an online shopping application for managing sales and ordering of a set of products. In B2C subsystem, all user processes are recorded into the web server log that will be analyzed and managed using Modern Checking Analysis (MCA) technique for analyzing users' behavior.

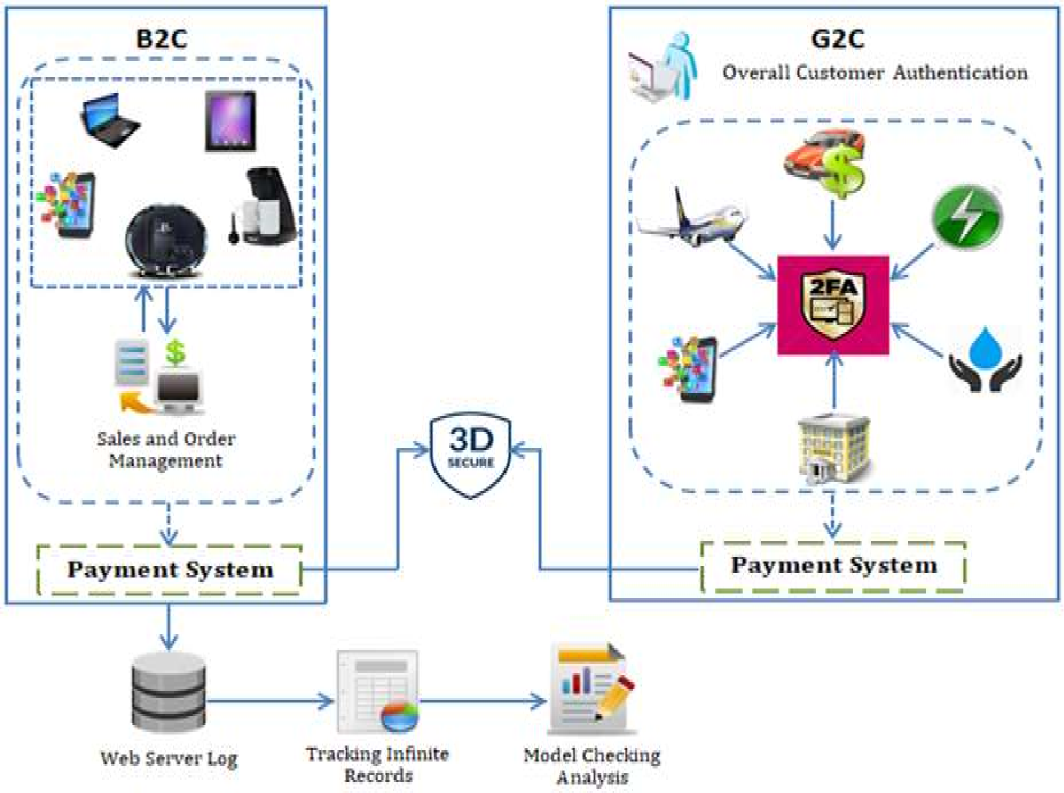

Figure 1: E-Business Framework

4. USER BEHAVIOR MECHANISM

The structure of the B2C E-business application is based on ordering and transmitting customer requests based on their preferences. In the B2C subsystem, a web server log is created for recording all browsing and requests created by customers in order to enhance the quality of web service information and predict users' behavior.

As presented in Figure 2, the web service structure consists of different secondary sections related to the main section. The main sections of the B2C structure starts with the homepage with level 0. Level 1 consists of a set of categories from C_1 to C_n such as Mobiles, Electronics, etc. Each category contains a set of related products from P_1 to P_n as shown in level 2. In level 3, the products are composed of a set of interrelated items from I_1 to I_n. In level 4, the items are composed of a set of interrelated sub-items from S_1 to S_n.

As shown in Figure 2, the shaded circle refers to the leaf items that have no sub-items. The items that have no sub-items are stored in the web server log with a NULL value. To identify a specific product or item, the B2C structure depends on the formula $L_{i,j}$ where i refers to the structure level and j refers to the product or item number in the web server log.

For each category, the process of tracking users' behavior consists of five sequential event actions. These event actions are viewing (V), revision (R), payment (P), execution (E), and item return (IR). A counting process is executed for determining the customer preferences for tracing users' behavior.

The viewing process (V) is based on browsing different products, items, or sub-items for the same category or different categories according to customer preferences. The customer can view an infinite number of categories with the ability of database administrator to add a new sub-level of the sub-item according to the nature of the product.

The revision process (R) is based on revising the customer's products, the number of items purchased, products specifications, price for each category, and the total payment required. In this step, the customer has the ability to rollback the process for adding, modifying, or cancelling the products purchases. The payment process (P) is the second step after revision process (P) for performing the payment action, determining the security participates: acquirer side, issuer side, and interoperability side for the 3D secure mechanism. In this step, the customer has the ability to rollback the process before execution.

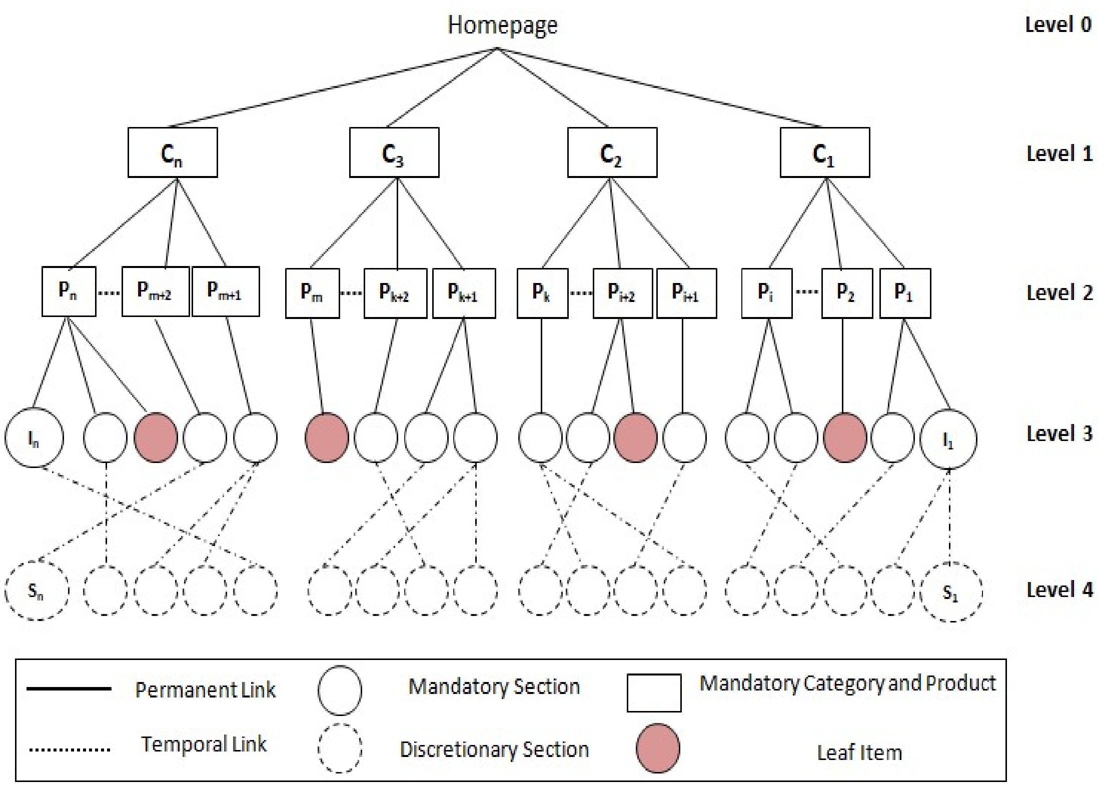

Figure. 2: Main B2C Structure

The execution process (E) is the last step after payment process (P) for executing and performing the transmission process. Once the process is executed, data will be committed in the web service database with no ability to rollback the process. The item return process (IR) is one of the recent features in the presented E-business architecture for handling and manipulating the product in case it is returned back.

Based on the previous event actions, if $\mathbb{A}$ is a set of all distinct action operations into which a single atomic operation $a \in \mathbb{A}$, then it is possible to define the following:

Definition 1 (Event Action Transition). A finite state for executing event actions is based on the triple $\mathbb{T}_{\mathbb{A}} = (S, S_0, \rightarrow)$, where S is the total number of states in the web server log and $S_0 \in S$ while $\rightarrow$ is the transition relation between different states stored in the web server log and is defined as $\subseteq S \; x \; 2^A \; x \; S$.

It determines the overall number of actions 2^A in the web server log with the total number of state transitions S before and after each action stored in the web server log.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The transition action $\mathbb{T}_{\mathbb{A}}$ runs with an infinite sequence based on the mathematical formula (1):

$$\mathbb{T}_{\mathbb{A}} = S_0 \overset{X_0}{\rightarrow} S_1 \overset{X_1}{\rightarrow} S_2 \overset{X_2}{\rightarrow} S_3 \overset{X_3}{\rightarrow} S_4 \overset{X_4}{\rightarrow} \ldots \overset{X_{n-1}}{\longrightarrow} S_n \quad (1)$$

Where $(S_i, X_i, S_{i+1}) \in \rightarrow for\ i \geq 0$ and the transition relation $\rightarrow$ are executed over infinite number of trace actions 2^A.

Definition 2 (Level Action Structure). In the web server log, two main procedures are executed to recall all user actions corresponded to the E-business structure, namely, Main (Secondary) sections and event actions. Let $N \in \mathbb{N}$ be the number of all levels of the E-business web structure. Let $\mathbb{V}_{\mathbb{A}}$ be the total set of event actions.

$\forall S \in N$, where $S = \{ S_i \mid i \in \ldots\ldots N \}$ be the main or secondary web structure levels. $\forall t \in \mathbb{V}_{\mathbb{A}}$, where $t = \{ V, R, P, E, IR, AP, RP, MP \}$ be the view, revision, payment, execute, item return, add post, remove post, and modify post respectively. Each level S_i must be $\leq N$ and each visited page must have the proposition formula S_i $(\mathbb{V}_{\mathbb{A}\,t})$.

5. FORMULA PATTERNS' DISCOVERY

Linear temporal logic (LTL) is a model checking technique that uses a set of formula patterns for making easier query of customer preferences [15]. It is considered one of the major declarative approaches for storing all user actions over an extended period of time for unlimited event streams [16]. Customer browsing and preferences are recorded into the web server log for predicting user's behavior and enhancing the E-business application structure.

In this paper, an enhancement of declarative approaches is presented by applying a set of formula patterns for predicting users' behavior and enhancing E-business structure. The main LTL pattern operators presented are: next operator '**X**' with the symbol '**O**', always operator '**G**' with the symbol □, and eventually operator '**F**' with the symbol ◇.

5.1 Mathematical LTL Formula Patterns

As presented in **Definition 2**, Let $N \in \mathbb{N}$ be the number of all levels of the E-business web structure into which each view level $VL_i \in S$, where S is the main or secondary web level.

Definition 3 (Home Page Browsing). When the customer browses and views only the homepage of the E-business structure, the LTL formula pattern will be $LTL = VL_0$, where VL_0 , is the view level 0 for the B2C web structure.

Definition 4 (Ordered Levels Browsing). When the customer browses and views subsequent levels of the E-business structure, the LTL formula pattern will be as follows:

$$\boldsymbol{F}(\boldsymbol{VL_1} \,\&\, \boldsymbol{X\,F}(\boldsymbol{VL_2} \,\&\, \boldsymbol{X\,F}(\boldsymbol{VL_3} \,\&\ldots)\ldots \boldsymbol{X\,F}(\boldsymbol{VL_n})))(2)$$

, into which the view level 1 (VL_1) will be followed by VL_2 using the operator **X** and the browsing process using the eventually operator ***F*** means that the order is not frequent.

The formula is converted to a set of linear temporal logic (LTL) operators as follows:

$$\Diamond(VL_1 \,\&\, O \Diamond(VL_2 \,\&\, O \ \Diamond VL_3 \,\&\, O \ \Diamond \& \ldots)\ldots O\ (VL_n))) \quad (3)$$

Definition 5 (Non-Ordered Levels Browsing): When the customer browses random levels of the E-business structure without a specific order, the LTL formula pattern will be as follows:

$$\mathbf{F(VL_1)} \,\&\, \mathbf{F(VL_2)} \,\&\, \mathbf{F(VL_3)} \,\&\ldots\ldots\&\, \mathbf{F(VL_n)} \quad (4)$$

The formula is converted to a set of linear temporal logic (LTL) operators as follows:

$$\Diamond(VL_1) \,\&\, \Diamond(VL_2) \,\&\, \Diamond(VL_3) \,\&\ldots\&\, \Diamond(VL_n) \quad (5)$$

Definition 6 (Ordered with Finite Action): If the customer performs an ordered process starting from viewing E-business structure levels and proceeding to revising, payment, and execute processes, then the LTL formula pattern will be as follows:

$$\boldsymbol{F}\,(\boldsymbol{VL_1} \,\&\, \boldsymbol{X\,F}\,(\boldsymbol{VL_2} \,\&\boldsymbol{X\,F}\,(\boldsymbol{VL_3} \,\&\ldots\ldots \boldsymbol{X\,F}\,(\boldsymbol{VL_n} \,\&\, \boldsymbol{X\,F}\,(\boldsymbol{R} \,\&\, \boldsymbol{X\,F}\,(\boldsymbol{P} \,\&\, \boldsymbol{X\,F}\,(\boldsymbol{E}\,))))))) \quad (6)$$

The formula is converted to a set of linear temporal logic (LTL) operators as follows:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$\Diamond(VL_1 \,\&\, O\Diamond(VL_2 \,\&\, O\Diamond(VL_3 \,\&\ldots O\Diamond(VL_n \,\&\, O\Diamond(R \,\&\, O\Diamond(P \,\&\, O\Diamond(E))))))) \quad (7)$$

Definition 7 (Non-Ordered with Finite Action): In case of viewing non-ordered levels of E-business structure and finally performing a finite action procedure, the LTL formula pattern will be as follows:

$$\boldsymbol{F}\,(\boldsymbol{VL_1}) \,\&\, \boldsymbol{F}\,(\boldsymbol{VL_2}) \,\&\, .\;.\;.\;.\;. \,\&\, \boldsymbol{F}\,(\boldsymbol{VL_n} \,\&\, \boldsymbol{X}\,\boldsymbol{F}\,(\boldsymbol{R} \,\&\, \boldsymbol{X}\,\boldsymbol{F}\,(\boldsymbol{P} \,\&\, \boldsymbol{X}\,\boldsymbol{F}\,(\boldsymbol{E}\,)\,)\,)\,)\,)\,) \quad (8)$$

In this formula pattern, the next operator $\boldsymbol{X}$ will not exist in the view levels due to the non-ordering processes of browsing and viewing categories. Starting from the last viewing level of a specified sub-item, the next and eventually operators $\boldsymbol{X}$ and $\boldsymbol{F}$ respectively will be added to the formula due to the ordering of revising, payment, and executing processes. The formula is converted to a set of linear temporal logic (LTL) operators as follows:

$$\Diamond(VL_1 \,\&\, \Diamond(VL_2 \,\&\, \Diamond(VL_n \,\&\, \ldots O\Diamond(VL_n \,\&\, O\Diamond(R \,\&\, O\Diamond(P \,\&\, O\Diamond(E))))))) \quad (9)$$

5.2 Declare Patterns

Declare patterns are used based on the assumption that at each specific time, one and only one process is executed for each user and this process eventually terminates [17]. For dealing with finite processes of program execution, a set of definitions are presented based on the B2C web structure.

Definition 8 (Existence Patterns): As presented in Table 1, the existence patterns simplify how many times the action is being executed. The view level (VL_i) and revision (R) are executed at least once in the transaction while the payment (P), execution (E), and item return (IR) are executed at most once in the same transaction.

Table 1: Existence Patterns

Formula Pattern	Pattern Description
$\Diamond VLi$	The view level VL_i must be executed at least once in the transaction.
$\Diamond R$	The revision (R) can be executed at least once in the same transaction.
$\neg\Diamond(P \wedge \Diamond P)$	The payment (P) can be executed at most once in the same transaction.
$\neg\Diamond(E \wedge \Diamond E)$	The execution (E) can be executed at most once in the same transaction.
$\neg\Diamond(IR \wedge \Diamond IR)$	The item return (IR) process can be executed at most once if the customer rollbacks the item.

Definition 9 (Choice Patterns): The choice pattern determines different alternative actions that can be executed in the E-business structure. As presented before, the database administrator can add, modify, or delete posts from the B2C E-business application. The LTL formula pattern for these actions can be presented as follows:

$\Diamond AP \vee \Diamond RP \vee \Diamond MP$, where the operation add post or remove post or modify post must be executed.

Definition 10 (Precedence Patterns): The precedence patterns simplify the priority of executing one action before another action. The formula pattern $\neg\, \boldsymbol{E}\; \boldsymbol{W}\, \boldsymbol{P}$ means that the execution of transaction (E) must be executed only if the payment (P) has been executed before. This means that the payment (P) must precede the transaction execution (E).

The formula pattern $\neg\, \boldsymbol{P}\; \boldsymbol{W}\, \boldsymbol{R}$ means that the payment (P) must be executed only if the revision (R) has been executed before. This means that the revision (R) must precede the payment (P). In this case, the item return (IR) cannot depend on the precedence pattern because rolling back an item is not a frequent option in the E-business structure.

Definition 11 (Succession Patterns): The succession patterns simplify that executing some actions can follow the execution of other actions. Two formula patterns are presented as follows:

$\Box\,(R \rightarrow \Diamond P) \wedge (\neg\, P\; W\; R)$, where the payment (P) must be executed after revision (R) and revision (R) must precede payment (P).

$\Box\,(P \rightarrow \Diamond\, E) \wedge (\neg\, E\; W\; P)$, where the transaction execution (E) must be executed after payment (P) and the payment (P) must precede the transaction execution (E).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. BEHAVIORAL PATTERN ANALYSIS

For analyzing the web server log, the nature and structure of the log must be determined. If the web server log corresponds to an E-business structure where the execution sequences of operations are finite and constrained, then it is called procedural logs. If the web server log corresponds to an open system where infinite operations sequences are executed, then it is called spaghetti log.

In spaghetti logs, large number of activities and execution options are recorded; making the analysis and prediction of user's behavior a very complex task [15]. A declarative mechanism is used to deal with these problems. LTL is considered one of the major declarative approaches for storing all user actions over an extended period of time for unlimited event streams [16].

A program execution is considered as an ordered sequence of Boolean formulas that satisfy program states. Different data mining techniques such as classification, clustering, association rule mining, and sequential pattern mining are used to analyze users' behaviors [18]. In open systems such as E-commerce and E-business websites, the discovery of process-oriented model for tracing users' behavior becomes difficult [19]. This is due to the multiplicity of user operations and interactions to the E-business applications structure.

In addition, data mining techniques cannot work in a direct way with a sequence of events during user navigation through E-business or E-commerce websites, but they work on an abstraction of such sequence [11].

This paper presents an enhanced set of formula patterns for improving the analysis performance of large number of user records that are continuously stored in the web server log. As presented in [15], two formula patterns are applied to trace user records based on ordered and non-ordered actions (A_n) where the type of actions is not simplified. In this work, the formula patterns are classified to browsing or viewing processes and action processes.

The proposed formulas presented in **definition 4** and **5** utilize the concept of linear temporal logic (LTL) for browsing ordered and non-ordered levels respectively. In **definition 4**, the customer traces a sequence of category, product, item, and sub-item for each web service. Thus, the view level 1 (VL_1) or (category) must be followed by the view level 2 (VL_2) or (product) using a next operator **'X'** between both operations until the last level is reached (VL_n).

In **definition 5**, the customer traces non-ordered levels for the same or different web services while other levels may not be traced. Thus, the eventually operator **'F'** is used between each view level (VL_{n-1}) and (VL_n).

In this work, we added new features of linear temporal logic (LTL) for classifying and enhancing the analysis processes and achieving better prediction of user behavior. The new features depend on the event stream actions that were presented in [16].

These features are: ordered with finite action and non-ordered with finite action to consider the dynamic nature of user transactions as presented in **definition 6** and **7**. In **definition 6**, the viewing or browsing processes are combined with action processes to perform an ordered finite action based on the two operators next **'X'** and eventually **'F'** until the final action $X\ F\ (\ E\ )$ is executed.

Definition 7 presents the most common user actions in E-business structure. The formula of **definition 7** is based on performing non-ordered and random actions on different levels of the E-business web services and finally a finite procedure is executed.

Thus, the eventually operator **'F'** is used between each view level (VL_{n-1}) and (VL_n) while the finite payment procedure is combined to the viewing processes using two operators next **'X'** and eventually **'F'** based on the formula $X\ F\ (\ R\ \&\ X\ F\ (\ P\ \&\ X\ F\ (\ E\ )\ )\ )$.

The remaining formula patterns present different classifications of user actions that can lead to better analysis performance. The existence patterns presented in **definition 8** simplify how many times each action can be executed. Some actions such as revision◇ R can be executed at least once in the same transaction because each user can revise the products that he intends to purchase many times while other actions such as payment◇(P), execute◇ (E), and finally the item return ◇ (IR) can be executed at most once in the same transaction. In **definition 9**, a choice pattern is used to determine the action type being executed by E-business administrator whether to add, remove, or modify web service posts.

In this paper, the precedence pattern is used to determine the priority of executing each action. As presented in **definition 10**, the revision (R) must be executed before payment (p) and the payment (p) must be executed before execution (E). In **definition 11**, the succession pattern uses the always operator to refer to the necessity of performing payment (P) after revision (R) and the execution (E) after payment (P).

7. FLAG PARAMETER ALGORITHM

Based on the presented formula patterns, an analysis of the web server log is clarified for enhancing the performance of E-business structure. This enhancement is based on classifying each event action with the corresponded main or secondary web structure levels based on the proposition formula S_i $(\mathbb{V}_{\mathbb{A}\,t})$. The pattern analysis is classified into four main processes: flag parameter log, customer duration log, web server customer log, and web server item log.

Each transaction being executed by customers is recorded based on a 'flag' parameter. As presented in Table 2, the view level VL_i is divided into category, product, item, and sub-item such that $VL_i \in S$ where S is the main or secondary web level. The event action $\mathbb{V}_{\mathbb{A}}$ is divided into two flags: viewing flag (VF) and action flag (AF) where each visited service page updates the flag value to 1 based on the proposition formula S_i $(\mathbb{V}_{\mathbb{A}\,t})$.

As presented in Algorithm 1, the transition action from view level $\mathrm{VL_i}$ to the following view level $\mathrm{VL_{i+1}}$ will update the flag parameter log table. For example, if the transition action is "item", the flag item FL_I will be set to 1 and all proceeding flags (FL_P, FL_C) will be updated to 1 also because the customer cannot proceed to the "item" without proceeding to its category (FL_C) and product (FL_P).

The processing of action flags is based on a force transition because the customer cannot execute payment (FL_{PT}) without passing through the review flag (FL_R) and payment flag (FL_{PT}). If the category has no item or sub-item, then their corresponding flag value will be equal to 0.

8. CUSTOMER DURATION LOG

As presented in Algorithm 2, the transition action of web service customers whether on viewing flags or action flags or both will be recorded into a customer duration log. For each transition or transaction being executed, a new record will be inserted into the customer duration log.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Algorithm 1: Flag Parameter

1. Initialize viewing flags FL_C , FL_P , FL_I , FL_S = 0
2. Initialize action flags FL_R , FL_{PT} , FL_E , FL_{RT} = 0
3. Repeat While (Viewing Flag = True)
4. If T_A = Category Then // transition action
5. Set FL_C = 1
6. Else if T_A = Product Then
7. Set FL_P , FL_C = 1
8. Else if T_A = Item Then
9. Set FL_I , FL_P , FL_C = 1
10. Else if T_A = Sub-item Then
11. Set FL_S , FL_I , FL_P , FL_C = 1
12. Else if Item = Null Then
13. Set FL_I = 0
14. Else if Sub-item = Null Then
15. Set FL_S = 0
16. [End of If]
17. [End of While Loop]
18. Repeat While (Action Flag = True)
19. Set Viewing Flag = True
20. Repeat While (T_A != Review) // force transition
21. Wait ()
22. FL_R = 1
23. [End of While Loop]
24. Repeat While (T_A != Payment) // force transition
25. Wait ()
26. FL_{PT} = 1
27. [End of While Loop]
28. Repeat While (T_A != Execute) // force transition
29. Wait ()
30. FL_E = 1
31. [End of While Loop]
32. Repeat While (T_A != Return Item) // force transition
33. Wait ()
34. FL_{RT} = 1
35. [End of While Loop]
36. [End of While Loop]

This record contains a starting timestamp, username, MAC address, and IP address of the customer. Once the transition or transaction is ended, an end timestamp will be recorded and finally the net timestamp will be recorded by subtracting the end timestamp from start timestamp. As presented in [11], a preprocessing phase is executed for cleaning the web server log from automatic requests and deleting erroneous status codes but the resulted web server log after the preprocessing lacks to register the customer MAC address and browsing process is based on only two levels of sections. In addition, the executed operation is registered only based on GET and POST.

In our proposed methodology, a complete classification of E-business structure levels are presented with different methods for dealing with the web server log that provides flexibility and high efficiency in analyzing user behavior.

Algorithm 2: Duration Log

1. Repeat While (Viewing Flag = True & Action Flag = False) or (Viewing Flag = True & Action Flag = True)
2. Calculate Start Timestamp
3. Register Username
4. Register MAC address
5. Register IP address
6. Calculate End Timestamp
7. Calculate Net Timestamp
8. [End of While Loop]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: Flag Parameter Log

User name	MAC Address	IP Address	Net Timestamp	Category	Product	Item	Sub-item	Viewing Flag				Action Flag			
								FL_C	FL_P	FL_I	FL_S	FL_R	FL_{PT}	FL_E	FL_{RT}
Adam	34-81-2B-13-A2-62	192.168.1.9	00:08:05:216	Mobile	Huawei	Mate	10	1	1	1	1	1	1	1	0
Adam	34-81-2B-13-A2-62	192.168.1.9	00:12:17:045	Mobile	Sony	Xperia	XZ3	1	1	1	1	1	0	0	0
Jones	74-C4-3B-52-33-51	192.168.1.3	00:19:44:317	Electronic	DELL	Inspiron	XPS 13	1	1	1	1	1	1	1	1
Scott	74-C8-6B-58-52-60	192.168.1.7	00:02:09:178	Electronic	Lenovo	Tab 7	Null	1	1	1	0	1	1	1	0
Jones	74-C4-3B-52-33-51	192.168.1.3	00:06:37:149	Electronic	DELL	Inspiron	XPS 15	1	1	1	1	1	1	0	0

9. CONCLUSION AND FUTURE WORKS

E-business applications provide customers with an endless browsing of products, categories, and items. These products must be organized in a structured framework for controlling and modifying all customer preferences and recommendations easily. The browsing and viewing processes must be monitored and recorded for tracing customer operations and predicting customer's behavior.

This paper presented a structured framework for B2C and G2C subsystems. The G2C subsystem presents a set of governmental services embedded into the overall system. The presented governmental services must be protected by applying authentication mechanisms for securing customer transmissions based on 3D secure system.

The B2C subsystem presents an online shopping application for managing sales and ordering of a set of products. Analyzing infinite customer browsing and transactions is considered a very complex task. In this paper, an enhanced formula patterns and algorithms based on linear temporal logic and modern checking analysis are applied for tracking and predicting user's behavior. Future directions of this research focus on implementing the E-business framework and applying different E-business processes and transactions. Security features should be added to the implemented framework for maintaining secrecy and confidentiality to user transactions.

REFRENCES

[1] R. Ciarniene and G. Stankeviciute, "Theoretical Framework of E-Business Competitiveness," *In proceeding of the International Scientific Conference of Economics and Management, Procedia - Social and Behavioral Sciences, ELSEVIER*, vol.123, 2015, pp.734-739

[2] K. Chao, "E-services in E-Business Engineering," *ELSEVIER Journal of Electronic Commerce Research and Applications*, vol.16, no.2, 2016, pp.77-81

[3] A. Basu, S. Muylle, "Assessing and Enhancing E-Business Processes," *ELSEVIER Journal of Electronic Commerce Research and Applications*, vol.10, no.4, 2011, pp.437-499

[4] G. Saha, A. Ghosh, N. Saha, "Digital Signature Modeling in E-business," *In Proceeding of IEEE International Conference on E-business Engineering*, 2014, pp.350-354

[5] G. Ping, "Analysis the Application of E-business for the Tourism Enterprises' Performance Evaluation in China," *ELSEVIER Journal of Energy Procedia*, vol.5, 2011, pp.849-854

[6] A. Ahrens and J. Zascerinska, "A Comparative Study of Engineering and Business Students' Attitude to E-Business Application," *In Proceeding of IEEE International Conference on E-business*, 2014, pp.31-38

[7] C.J. Carmona, S. Ramirez, F. Torres, E. Bernel, M.J. Del Jesus, S. Garcia, "Web Usage Mining to Improve the Design of an E-commerce Website: OrOliveSur.com," *ELSEVIER Journal of Expert Systems with Applications*, vol.39, 2012, pp.11243-11249

[8] Q. Su and L. Chen, "A Method for Discovering Clusters of E-commerce Interest Patterns using Click-Stream Data," *ELSEVIER Journal of Electronic Commerce Research and Application*, vol.14, no.1, 2015, pp.1-13

[9] A. Hemapriya, M. Bhuvaneswari, K. Muneeswaran, "Reconstruction of User Sessions from Web Server Log," *IEEE International Conference on Computing and Communication Technologies*, 2015, pp. 37-42

[10] A. Ahrens, J. Zascerinska, "A Framework for Selecting Sample Size in Educational Research on E-Business Application," *In Proceeding of IEEE International Conference on E-business*, 2014, pp.39-46

[11] H. Sergio, P. Alvarez, J. Fabra, J. Ezpeleta, "Analysis of Users' Behavior in Structured E-commerce Websites," *IEEE Transaction Access Journal*, vol.5, 2017, pp.1-17

[12] F. Maggi, R. Bose, W. Van der Aalst, "Efficient Discovery of Understandable Declarative Process Models from Event Logs," *International Conference on Advanced Information Systems Engineering, SPRINGER*, 2012, pp. 270-285

[13] S. Zugal, J. Pinggera, B. Weber, "The Impact of Test cases on the Maintainability of Declarative Process Models," *International Conference on Exploring Modeling Methods for Systems Analysis and Design, SPRINGER*, 2011, pp. 163-177

[14] R. Rezaei, T. Chiew, S. Lee, "A Review on E-business Interoperability Frameworks," *ELSEVIER Journal of Systems and Software*, vol.93, 2014, pp. 199-216

[15] P. Alvarez, J. Fabra, S. Hernandez, J. Ezpeleta, "Alignment of Teacher's Plan and Students' Use of LMS Resources: Analysis of Moodle Logs," *In Proceeding of IEEE International Conference on Information Technology Based Higher Education and Training*, 2016, pp.1-8

[16] A. Burattin and M. Cimitile, "Online Discovery of Declarative Process Models from Event Streams," *IEEE Transaction on Service Computing*, vol.8, no. 6, 2015, pp.1-14

[17] G. De Giacomo, R. De Masellis, M. Montali, "Reasoning on LTL on Finite Traces: Insensitivity to Infiniteness," *In Proceeding of ACM International Conference on Artificial Intelligence*, 2014, pp.1027-1033

[18] Q. Zhang and R. Segall, "Web Mining: A Survey of Current Research, Techniques, and Software," *International Journal of Information Technology and Decision Making*, vol.7, no.4, 2008, pp.683-720

[19] N. Poggi, V. Muthusamy, D. Carrera, R. Khalaf, "Business Process Mining from E-Commerce Web Logs," *In Proceeding of the International Conference on Business Process Management, SPRINGER*, 2013, pp. 65-80