

Risks of Concurrent Execution in E-Commerce Processes

Janis Bicevskis, Anastasija Nikiforova, Girts Karnitis
[[0000-0001-5298-9859, 0000-0002-0532-3488,
0000-0002-7563-6383]
Faculty of Computing University of Latvia
{Janis.Bicevskis, Anastasija.Nikiforova, Girts.Karnitis}@lu.lv

Ivo Oditis, Zane Bicevska
[0000-0003-2354-3780,
0000-0002-5252-7336]
DIVI Grupa Ltd
{Ivo.Oditis, Zane.Bicevska}@di.lv

Abstract—The development of ICT facilitates replacing the traditional buying and selling processes with e-commerce solutions. If several customers are served concurrently, e.g. at the same time, the processes can interfere each other causing risks for both the buyer and the seller. The paper offers a method to identify purchase/sale risks in simultaneous multi-customer service processes. First, an exact model of buying-selling processes is created and the conditions for the correct process execution are formulated. Then an analysis of all the possible scenarios, including the concurrently executed buying-selling scenarios, is performed using a symbolic execution of process descriptions. The obtained result allows both the buyer and the seller to identify the risks of an e-commerce solution.

Index Terms—concurrent processes, risk analysis, e-commerce.

I. INTRODUCTION

THE development of ICT has created preconditions for radical changes in buying and selling processes worldwide. The traditional buying/ selling process, when sellers and buyers meet each other in person and communicate directly, is replaced by a remote communication, the so-called e-commerce.

Traditionally the buyer chooses a suitable product, checks its quality, receives a product, and pays for it directly to the seller (in cash or using safe bank services) in a shop, and no additional tools are needed. The advantages of e-commerce lie in the global spread allowing entrepreneurs to develop remote marketing and sales on an unlimited geographical scale. E-commerce solutions are perceived as the future of commerce, as more and more customers want to buy products without leaving their places [1]. By switching to remote seller-buyer communication, the information exchange may become more complicated and, at the same time, riskier for both the seller and the buyer. The seller is not willing to risk by sending the product to the buyer before being sure that the purchase is paid for; the buyer does not want to risk paying for a product is not seen yet and whose quality he has not been able to be sure of. Additionally, the

processes are complicated because of many different product delivery channels and product payment options as well as by many simultaneous customers.

The main task of this research is to analyze the risks of remote buying and selling processes when e-commerce solutions are in use and simultaneous service of several customers take place.

This paper contains an analysis of several e-commerce cases that are common in different industries: theater ticketing, online stores, hotel reservation systems. The algorithm, applied for the analysis of process correctness, has been developed through a theoretical research for correctness of concurrent process execution [2]. The proposed algorithm identifies the possibility of incorrect execution of remote purchase-sale processes, thus revealing the risks for both buyers and sellers.

This paper is structured as follows: the background (Section 2), risk analysis of selected e-commerce processes (Section 3), analysis of the proposed solution (Section 4), conclusions and the future work (Section 5).

II. THEORETICAL BACKGROUND

II.A. State of the Art

A literature review reveals different internet shopping-related risk classifications established over the last decades. A total number of risks considered to have a significant impact on users' intention for online shopping vary from one study to another and ranges from two to eight [3].

Forsythe et al. [4] identified six types of perceived risks that may have a negative impact on the experience of buyers: financial, product performance, social, psychological, physical, and time loss. Respondents found financial risk to be the most important and significant. Considering the age of this study and the development of e-commerce over the past years, most of the identified risks have already been processed and resolved. However, some of them remain valid, for instance, financial risks.

Formerly, financial risks were primarily associated with potential losses of money due to fraudulent misuse of credit card information. Nowadays, the paradigm on financial risks has changed [5]-[6]. The online credit card usage-related risks are thoroughly discussed in security-related studies,

The research leading to these results has received funding from the research project "Competence Centre of Information and Communication Technologies" of EU Structural funds, contract No. 1.2.1.1/18/A/003 signed between IT Competence Centre and Central Finance and Contracting Agency, Research No. 1.6 "Concurrence analysis in business process models".

and practical solutions are invented in online shopping platforms, including the implementation of 128-bit RSA encryption, digital certificates, firewalls etc. [6]. Another financial side-related risk is less covered: the trust between the customer and the service or/ and shopping service provider [5]-[8]. Trust and reputation are considered as the concepts dominating in e-commerce most [9], now. Bezes [10] proposes a classification where the probability of bank or personal data being stolen is understood as a “transaction risk”. The probability of losing money when buying from an online store is defined as a “financial risk”. There are studies rejecting the significance of financial risks, e.g. [11]. Based on a survey that has been carried out between 245 country residents, the study identifies the “convenience risk” and the “non-delivery risk” to be very significant, as well as the “reliability of shipper” and the “settling disputes”. However, it should be mentioned that, other classifications consider both abovementioned risks as financial risks.

Another risk, namely “performance risk” is associated with the potential failure of a product or website to meet expected performance requirements, i.e., the uncertainty regarding the after-sales service [8], [12]. According to [12], “risk perceptions” and the “online shopping intention” have a significant impact on online shopping.

[3] and [7] revealed that risks such as privacy, source, performance, payment and delivery risks are predominant dimensions in Internet shopping. The authors of [7] have carried out an in-depth analysis of risk-relievers. Although only one rather limited example of shopping has been analysed with a sample size of 471 respondents, the authors’ research suggests that 18 risk-relievers make sense for shopper. The main risk-relievers are (1) payment security, (2) money-back guarantee, as well as possibility (3) of exchanging the item, (4) of viewing the item, (5) of seeing item in a store, (6) price and (7) website reputation. [10] claims that guarantees provided by online sellers and insurance against any kind of adverse situations were assessed as the most important factor (88.7% of respondents) to drive online shopping.

In this study we provide a method that allows the identification of risks arising from the concurrent execution of processes without the e-commerce risks mentioned above. This topic is especially relevant due to COVID-19 pandemic as online shopping has become a daily phenomenon for most of the population, i.e. online stores have been launched in countries and cities where they did not exist before.

II.B. Analysis Basics

This study considers an algorithm for detection of incorrect concurrent execution of business processes that use a transaction mechanism. First, let’s clarify the basic concepts, which detailed description is provided in [13].

The process will be defined as a set of actions described in a modeling language. Two levels of process descriptions are possible: (1) the model is informal without well-defined semantics of operations; (2) the process is described with program code in a programming language where the seman-

tics of the actions are unambiguously defined. In the real world, many processes run concurrently, i.e., multiple instances of processes are executed simultaneously with different inputs, and shared information resources can be used. If several concurrently executed processes perform operations on the same data, then the data may be changed by another process during the breakpoint of one process, where a breakpoint is a process activity at which a process can be stopped and later restored from the state it was in before the break [13]. This can lead to incorrect system operation, which cannot happen if the processes are executed serially. Algorithms of business process analysis to determine the possibility of incorrect result of concurrent execution are the result of the theoretical research [2].

The method, proposed in [2], let us identify an incorrect result of concurrent execution of several processes. In the case of databases, we consider the concurrently executable processes $P_j, P_k \dots P_m$ and define the correctness of their execution according to the DBMS ACID correctness [13]: the result is correct if one process is executed without simultaneous execution of other processes and all transactions within the process are executed serially. If several processes are executed serially, the result is also correct, though there may be several different but correct results, depending on the sequence of process execution. The exact criterion to achieve a correct result for any process and any input data is to execute processes serially.

II.C. Risk Analysis Algorithm

There are six main steps of the universal algorithm.

Step I: create a description of the business process. In order to analyze a business process, a model of a business process should be created. If the business process is described informally, the model can be designed as a graphical diagram where the vertices of the graph represent the activities of the business process and the arcs the sequence of activities. The model’s author must be able to assess the feasibility of scenarios and the outcome of scenarios. If the model consists of a program code, the execution of statements and the sequence of activities are strictly defined. In this case, the code analysis can be performed automatically by a tool. The program code is executed symbolically: the tool compiles the conditions for the execution of the scenario and calculates the result of the execution of the scenario.

Step II: define business process transactions. Business process activities or a set of activities are defined as transactions in cases when their execution is delegated to another system (for example, to a DBMS) or their execution requires timeframe during which access to common resources may not be blocked for other processes. For example, the process of selling theater tickets can be divided into three transactions: (1) read from the database the seats sold, (2) let the client choose a free seat in the hall, (3) let the client pay for the selected tickets. The ticket selection process should not be blocked for other remote customers, as the selecting may take some time.

Step III: define the incorrect business process execution. This step identifies situations that are not acceptable from a business perspective. If the process execution scenario leads to a situation that does not meet the business requirements, then the definition of the business process needs to be revised to avoid incorrect execution results.

Step IV: construct a feasible scenario tree. The model's author selects different process execution scenarios and evaluates their feasibility; the author makes sure that there is input data that will make the selected scenario executable. The result of the analysis is represented in a tree, where each branch of the tree represents one feasible scenario and the tree contains all possible different scenarios. Depending on the business process this can be a difficult goal to achieve. If the model is defined by a program code, the "white box" analysis is used by symbolic execution of the program code, which enables to compile the conditions for the execution of a pre-defined scenario. When solving the conditions, the solution obtained is a test case that should be executed to cover pre-defined paths.

Step V: calculate scenario execution results. The model's author evaluates the expected result of the scenario execution from the business point of view using a symbolic execution.

Step VI: identify scenarios that lead to incorrect business process execution. According to [13], two sets of process execution scenarios are analyzed – a set of concurrent execution scenarios (C) and a set of serial execution scenarios (S). If at least one scenario S_j from S can be found for the scenario C_i from C such that the set of conditions and results of fulfillment of C_i coincides with the set of conditions and results of fulfillment of S_j , then the concurrent execution C_i is correct, otherwise it is incorrect.

III. A PROPOSED SOLUTION

In this section, the algorithm that identifies risks in e-commerce processes will be applied to several e-commerce solutions: business processes for theater ticketing, online stores and hotel reservation systems. All these e-commerce cases consist of three steps: ordering a product/service, payment for the goods, and delivery of the goods. The steps vary depending on the industry and the implementation. We will identify risks for both the seller and the buyer using different purchase-sale scenarios. To simplify the analysis, the processes will first be considered for ticket sales, and then they will be modified for other sectors.

III.A. Internet Shop for Theatre Ticketing

In the past, theater tickets were sold at ticket offices, and customers were served in presence. The currently available ticketing systems offer e-commerce functionality: connect to the system remotely, select a ticket, pay for it and receive a copy of the ticket. The purpose of the following sections is to identify the potential risks posed by the concurrent service of several customers.

A.1. Defining of Incorrect Process Execution

The correctness of the ticketing system's performance will be assessed by the status assigned to the seats in a hall. The status *seatStatus* will be determined by the values of two parameters: *availability of a seat* – {*available, reserved, sold*} and *status of payment* = {*paid, not paid*}. The ticketing system works correctly if the attribute *seatStatus* for any seat in the hall has either *<available, not paid>* or *<sold, paid>* as values. Any other result shall be considered as incorrect. If there are seats with the status "sold" and "not paid" at the same time, then the ticket system works unacceptably.

Process execution scenarios will be analyzed below to determine if there are possible scenarios that could lead to incorrect results.

A.2. The First Phase of the Business Process: Select a Seat

The ticketing system consists of three sequential phases: *Select a seat, Pay for a ticket* and *Send a ticket*. The first phase *Select a seat* consists of three activities: *readSeats, selectSeats, reserveSeats*. The activity *readSeats* reads information from the database about the customer's chosen performance and, if the event is not sold out, shows it to the customer. The activity *selectSeats* allows the customer to mark the chosen seats in the halls plan. The activity *reserveSeats* changes the information on occupied seats in the database by assigning the value "reserved" to the seat.

The *selectSeats* operation can take a longer time and therefore, during its execution, the common resource may not be locked; the information about the seats should be available to other customers. All three operations - *readSeats, selectSeats, reserveSeats* - will be executed as separate transactions. Thus, the ticketing system can execute many transactions from different customers' business processes "simultaneously", ensuring the execution of successful transactions for each individual process.

Unfortunately, the simultaneous service of several customers can lead to incorrect execution of the process. Let us construct a concurrent execution scenario of two processes P_1 and P_2 :

$$P_1(\text{readSeats, YES}) \Rightarrow P_2(\text{readSeats, YES}) \Rightarrow \\ P_1(\text{reserveSeats}) \Rightarrow P_2(\text{reserveSeats})$$

This scenario is feasible but there is a risk of selling the same seat to two customers if customers from both processes P_1 and P_2 choose the same seat. This is unacceptable, and the simplified seat selection process is risky. The situation changes drastically when seat reservation is used: a control mechanism checks whether the seat is already booked by another process. A correct process model, in which the data on free seats in the hall are re-read before reserving a seat and in case the seat selected by P_1 is already reserved in another process P_2 , the seat selection step is repeated. The business process is changed by adding additional controls before the actual reserving in *reserveSeats*.

A.3. The Second Phase: Pay for a Ticket

Banking systems offer many ways to pay for the tickets bought. In all cases, the step *Pay for a ticket* must be per-

formed as a separate transaction, because the service of other customers may not be interrupted until the end of the ticket payment process.

Step *Paying for a ticket* poses risks to both the seller and the customer: the seller reserves tickets for a certain period, preventing them from buying other customers, and the customer, in turn, pays for the ticket, believing that he will receive the ticket on time.

It is even more difficult for the ticket system to get a secure ticket payment because it is done by an external (bank's) payment system. Different banks have different payment solutions, which makes it difficult for the ticket system to unify payment processes.

The phase *Pay for a ticket* contains three activities: *readAccount*, *checkValue* and *writeAccount*. The activity *readAccount* reads the customer's account balance from the bank's database, the activity *checkValue* checks whether the customer enough means to pay the ticket price.

If payment can be made, the activity *writeAccount* deducts the amount payable for tickets from the account balance and stores the new account balance in the bank system's database. Activities *readAccount* and *writeAccount* are executed as independent transactions. And it leads to risks that the payments may be executed incorrectly if run concurrently [2]. However, as concurrent payment execution from a common resource (from one bank account) for several customers is unlikely, there are grounds to assume that the *Pay for the ticket* transaction is executed as one indivisible transaction and it cannot affect the service of other customers.

A.4. The Third Phase: Send a Ticket

It is possible two situations: (1) the payment was not completed successfully - the ticketing system sends a message to the customer about the refusal to purchase tickets, the corresponding seats in the halls plan are marked as available by the activity *changeStatus*, and the seller may sell the ticket to another customer; (2) the payment was not confirmed timely - this situation may occur if the message has not been received from the payment system timely. Different solutions are possible: resend the invoice to the customer or cancel the purchase, mark the corresponding seats as available for resale (*changeStatus*). This solution runs the risk that the ticket is actually paid for, but the ticket is resold to another customer due to a delay in reporting.

Sending a ticket to the customer without receiving a feedback and relying on the stable operation of the Internet are debatable. This defect can be remedied by providing a confirmation of receipt of tickets sent by the customer.

III.B. Online Store

The operation of the online store is determined by four steps of the process - marketing, selection of goods, payment for goods and delivery of goods. We will not consider marketing issues in this paper, the other three steps are similar to the operation of ticketing systems. However, there are certain peculiarities of the industry in the processes of

online stores, which are often related to the efficiency of delivery processes.

The activity *selectItem* differs from the choice of tickets significantly because the customer wants to choose the product personally, look at it and evaluate it in detail. The online store can only display similar samples from catalogues remotely. The customer will make the final assessment of the product only after receiving the product, when the product has already been paid for. If the payment for the goods is not prompt, the online stores can sell the goods to another customer who has paid for the goods faster. Such situations occur regularly in practice.

Even more risky is the customer's cooperation with the online store in cases when the online store orders goods only after placing a customer's order or payment. In other words, online stores without warehouses with stocks of goods are quite risky in terms of delivery. If the goods are not reserved upon receipt of the customer's order, the delivery of the goods to another customer who has made the payment earlier is not excluded, thus extending the delivery time.

The activity *payItem* does not differ significantly from the payment of tickets, however, additional risks are expected to be made if the amounts to be paid are significant and some purchases may lack money. The promptness of the payment has a significant impact on the process, as the delivery terms of the goods depend on it. Payment via Internet banking at the time of ordering (*Banklink*) is not only the safest, but also the most modern for shopping in online stores.

The risks of the activity *deliveryItem* are similar to those of ticketing processes – (1) the customer receives a product the quality of which has not been checked, (2) product delivery terms are determined by the customer's product payment efficiency and online store processes, (3) the online store can sell the product ordered by the customer to another customer who has paid for the product faster.

III.C. Hotel Reservation

The hotel reservation and payment processes are characterized by the fact that a hotel room is reserved for a specific customer for a specific timeframe. It is not possible to assign the same room to two customers, and it reduces the risks of the process. However, there is a risk that the room reserved for the customer will not be released in time, for instance, because the previous customer has extended his stay in a hotel. Such situations can be resolved only by a hotel staff.

An insignificant risk exists if the customer is trying to find the most advantageous hotel among others for a longer time. In the moment of booking, the special offer may no longer be valid because another customer has already booked it.

Payment for hotel services is usually made during the check-out process. The customer acknowledges the services received and pays for them. Unfortunately, the credit card may have not enough coverage to pay for the services received.

Summarizing the described process, it contains several risks caused by the concurrent process execution. These risks can be mitigated by identifying them in the information system and involving hotel staff.

IV. ANALYSIS OF THE PROPOSED SOLUTION

Summing up our experience, analysis of concurrently executable e-commerce processes is required in at least three cases:

1. for online store customers to identify the risks of purchasing: does the quality of the selected goods meet the customer's requirements? Will the goods be delivered on time? Is there a risk of non-delivering for pre-paid goods?
2. for online store owners to make sure the customer's solvency and reliability as well as to develop and improve the business processes;
3. for online store developers before the final implementation stage to detect vulnerabilities, defects, and errors in the business processes. This can save significant programming resources that could be wasted for implementing an incorrect business process (also in line with [13]).

V. CONCLUSION

The development of ICT has created preconditions for performing traditional buying and selling processes remotely using e-commerce solutions. The paper offers an algorithm for the analysis of e-commerce processes to identify the risk caused by their concurrent execution. The main steps of the algorithm:

- an accurate model of the e-commerce process is developed at such a level of accuracy that it is possible to determine the feasibility of any process scenarios and calculate the result of the scenario execution,
- process execution correctness conditions are formulated, and it allows each scenario to determine the compliance of its execution result with the process correctness conditions,
- the analysis of all different possible purchase-sale scenarios is performed using the symbolic execution of the process description, whereby concurrent execution of several customer processes is allowed. The obtained result of scenario execution allows identifying the process risks, which in turn allows both buyer and seller to choose an e-commerce solution acceptable to them.

The further development of the research can be devoted to additional e-commerce processes. The implementation risks should be analyzed, the processes improved, and as a result the use of modern e-commerce solutions would be expanded.

REFERENCES

1. Ecommerce Guide, "Ecommerce Statistics", online: <https://ecommerceguide.com/ecommerce-statistics>, last accessed: 21.05.2021
2. J. Bicevskis, G. Karnitis, "Testing of Execution of Concurrent Processes," In *International Baltic Conference on Databases and Information Systems* (pp. 265-279). Springer, Cham, 2020, https://doi.org/10.1007/978-3-030-57672-1_20.
3. E. O'Callaghan and J. Murray, "The International Review of Retail, Distribution and Consumer Research," *The International Review of Retail, Distribution and Consumer Research*, vol. 27, no. 5, p. 435-436, 2017 doi:10.1080/09593969.2017.139195
4. S. Forsythe and B. Shi. "Consumer patronage and risk perceptions in Internet shopping," *Journal of Business research*, 56(11), 867-875, 2003.
5. R. M. Al-dweeri, Z. M. Obeidat, M. A. Al-dwiry, M. T. Alshurideh, A. M. Alhorani, "The impact of e-service quality and e-loyalty on online shopping: moderating effect of e-satisfaction and e-trust," *International Journal of Marketing Studies*, 9(2), 92-103, 2017, <http://doi.org/10.5539/ijms.v9n2p92>.
6. R. Thakur, M. Srivastava, "A study on the impact of consumer risk perception and innovativeness on online shopping in India," *International Journal of Retail & Distribution Management*, 2015, <https://doi.org/10.1108/IJRDM-06-2013-0128>.
7. P. Rita, T. Oliveira, A. Farisa, "The impact of e-service quality and customer satisfaction on customer behavior in online shopping," *Heliyon*, 5(10), e02690, 2019, <https://doi.org/10.1016/j.heliyon.2019.e02690> Get rights and content.
8. N. Pappas, "Marketing strategies, perceived risks, and consumer trust in online buying behavior," *Journal of retailing and consumer services*, 29, 92-103, 2016, <https://doi.org/10.1016/j.jretconser.2015.11.007>.
9. C. Sellami, M. Baron, M. Bechchi, A. Hadjali, S. Jean, D. Chabot, "Towards a Unified Framework for Computational Trust and Reputation Models for e-Commerce Applications," In *Research Challenges in Information Science. RCIS 2021*, vol 415. Springer, Cham. https://doi.org/10.1007/978-3-030-75018-3_44, 2021
10. C. Bezes, "Comparing online and in-store risks in multichannel shopping," *International Journal of Retail & Distribution Management*, 2016, <https://doi.org/10.1108/IJRDM-02-2015-0019>.
11. K. Wai, O. Dastane, Z. Johari, N. B. Ismail, "Perceived risk factors affecting consumers' online shopping behavior," *The Journal of Asian Finance, Economics and Business*, 6(4), 246-260, 2019, <https://dx.doi.org/10.2139/ssrn.3498766>.
12. U. Otika, E. Olise, O. B. Oby, "Risk Perceptions and Online Shopping Intention among Internet Users In Nigeria," *Global Journal of Management And Business Research*, 2019.
13. A. Nikiforova, J. Bicevskis, G. Karnitis, "Towards a Concurrence Analysis in Business Processes," In *2020 Seventh International Conference on Social Networks Analysis, Management and Security (SNAMS)* (pp. 1-6). IEEE, 2020, DOI: 10.1109/SNAMS52053.2020.9336566