Advanced robotic process automation for enterprise efficiency

Jahongir Yoqubjonov¹, Ruslan Gibadullin¹*, and Marat Nuriev¹

¹Kazan National Research Technical University named after A. N. Tupolev – KAI, Kazan, Russia

Abstract. Utilizing ELegant MAnagement Robotic Process Automation (ELMA RPA) for automating technological processes within an enterprise offers several advantages. Companies can enhance operational efficiency, save time, reduce errors, and streamline their operations by utilizing this advanced automation solution. ELMA RPA is an especially effective tool for tasks involving repetitive processes, like document management and administrative work. The incorporation of ELMA RPA allows businesses to take advantage of software robots, which are programmed to perform tasks with precision and consistency. Through the use of readout modes for interface elements, process modeling, and carefully designed algorithms, ELMA RPA allows for seamless interaction between various third-party programs. This not only enhances productivity but also allows companies to remain competitive in the ever-evolving business landscape. In summary, ELMA RPA is an essential tool for businesses looking to optimize their operations and capitalize on the benefits of process automation. This article describes principles and peculiarities of business process automation development using software robots using ELMA RPA product. The following are considered: readout modes of interface elements (methods that allow recording and accurately reproducing interaction between different third-party programs using their graphical interface), process modeling, and algorithm of the robot's actions. An example of the implementation of business process robotization in a leading consulting and auditing company is considered.

1 Introduction

The latest technology, known as Robotic Process Automation (RPA), is intended to automate business processes by programming specialized software or "robots" to carry out tasks using algorithms on a dedicated platform. RPA robots mimic human actions within a system, utilizing user interfaces to gather, process, and manage data across various applications. This technology is particularly well-suited for monotonous tasks that do not demand creative solutions or expert evaluation from employees.

By implementing RPA, companies can create robot testers to assess the performance of their business processes. This enables them to examine the life cycle of a document and verify the accuracy of its transfer to the responsible personnel. As a result, RPA helps to streamline
workflows, reduce the risk of human error, and free up employees to focus on more complex, value-added tasks.

In essence, RPA technology offers a powerful solution for businesses looking to optimize their technological processes, enhance efficiency, and maintain a competitive edge in the ever-evolving market.

2 Problem statement

Developing a robot using the Russian ELMA RPA product to automate the role of a document workflow business process tester offers a valuable solution for identifying faults within the document management system. To successfully achieve this objective, several tasks must be addressed:

- Technology and Tools Selection: It is crucial to identify the most suitable technology and tools for developing the RPA robot, ensuring seamless integration with the existing infrastructure and optimal performance.
- Creation of Test Accounts: Establishing test accounts specifically for evaluating business processes is necessary to prevent interference with ongoing operations and maintain data integrity.
- Development of Testing Scenario Algorithm: Careful planning and design of the testing scenario algorithm is essential to ensure that the RPA robot can effectively identify and analyze potential faults within the document workflow system.
- Implementation of the Robot in the System: Once the robot has been developed, it must be integrated into the existing system. This step involves deploying the robot, ensuring compatibility with other components, and monitoring its performance to make any necessary adjustments.

By addressing these tasks, a robot built on the ELMA RPA platform can effectively automate the document workflow business process testing, leading to improved efficiency, error reduction, and overall enhanced system performance.

3 The solution to the problem posed

In 1999 a remote team of Novosoft was formed based on the Udmurt State University Mathematics Department. Until 2003 the team was engaged in custom software development. Since 2003, part of the team has been working as an independent EleWise company, continuing to develop custom software and develop software products. Among these products is the ELMA electronic document management system.

ELMA is a Russian IT company, a developer of business process management systems, software products within the framework of BPMS functioning, and software products based on artificial intelligence technology. In 2019, ELMA took 4th place in the ranking of BPMS developers by the CNews Analytics portal [1, 2].

Since 2008, more than 2000 companies in Russia and the CIS have implemented and successfully use the company's software products. For several years ELMA confidently holds a leading position in the market of BPM systems in Russia and the CIS according to the business portal TAdviser.

ELMA RPA system is a system of robotization of business processes using software robots (Robotic Process Automation). The system was released in May 2020 and is available in Community Edition and Enterprise edition. The solution allows you to robotize the filling of forms, reporting, checking the documentation for correctness, etc. [3, 4].
ELMA RPA business process automation technology was added to the unified register of Russian programs on December 30, 2020. It is located in the category "Enterprise Resource Management" and subcategory "Robotic Process Automation" [5, 6].

ELMA RPA robots use computer vision technology. This makes it possible to teach the robot to navigate in any interface that a human can navigate.

Teaching the software robot is very simple: the user shows the system his usual work by performing the usual steps step by step. The robot remembers every step and shows the user what it has learned.

The ELMA RPA approach allows you to work in any environment.

You don't need to install additional plug-ins or learn different techniques in each application. The system works the same way everywhere [7, 8].

The ELMA RPA robot learning process makes it very easy to create universal programs. To do this, you can interact with the interface elements in the recording process using the context menu.

The actions recorded in the training mode form the RPA model of the process. Each action is one card on the visual process model. You can change their order, and adjust parameters and behavior features. Such a model is easy to read and maintain.

ELMA RPA is a powerful corporate product. You can build bots on its basis, which will do all the routine work in the organization. Orchestrating a farm of bots, scheduled tasks, and seamless work with BPMS all allow you to robotize processes and fight routines on any corporate scale.

ELMA RPA processes are flexible solutions. They can be modified at any time - the new version will be quickly scalable to the entire organization. At any moment it is possible to roll back to the previous version using the process change archive.

RPA technology allows you to automate chains of tasks, but the full automation of business processes robots cannot. ELMA RPA robots, on the other hand, have seamless integration with BPM systems, allowing them to solve larger tasks.

ELMA RPA is designed for industry specialists and analysts. The C-system has a full set of activities that allow the robot to perform a variety of work:
- C# arbitrary code execution.
- Easy access to process context.
- Connecting external libraries.

ELMA RPA doesn't need any ongoing payments, and all editions of the software are provided with a perpetual license. Additionally, ELMA RPA Lite is available for small and medium-sized businesses to fully automate their business processes using robot software. [9, 10].

Digital transformation strategies have long relied on business process management technology, but with the advent of new innovations, companies are discovering fresh avenues to become proficient in the digital realm. Among the latest automation tools that are gaining traction in workplaces worldwide is Robotic Process Automation, which is proving to be a valuable asset in enhancing efficiency and productivity [11, 12].

Robotic Process Automation is a type of software technology that enables the use of specialized software to replace human labor with robots. These robots can interact with various applications at the interface level, thereby allowing employees to concentrate on more critical tasks. Using RPA technology, robots can complete mundane tasks like inputting data and initiating actions in multiple systems, relieving employees of these responsibilities. This technology can improve the precision, dependability, and promptness of operations. RPA provides organizations with the ability to program software to perform repetitive and automated tasks at the user interface level, without requiring any changes to the company's IT infrastructure. This removes the need for expensive integration software or custom APIs, while also facilitating the integration of existing applications and systems. [2].
RPA systems have the same effect on business as classical automation: some routine and repetitive tasks are moved from the shoulders of employees to the responsibility of computers [13, 14].

An RPA robot mimics human actions but performs tasks at a quicker pace because:
- The robot doesn't experience fatigue, quit or go on vacation, and is always prepared for tasks, regardless of the time of day.
- A robot typically executes tasks faster than a human.
- Its task results are predictable.
- The robot is not prone to errors resulting from "human error".
- The robot can be easily duplicated to distribute a high volume of tasks.

Examples of tasks that can be solved with the help of RPA systems:
- Transferring data from one program to another.
- Filling out electronic forms, and cards.
- Processing data in Excel tables, and databases.
- Preparation of reports.
- Checking information for accuracy.

ELMA RPA system is a system of robotization of business processes through software robots (Robotic Process Automation). The system was released in May 2020 and is available in Community Edition and Enterprise edition. The solution allows you to robotize the filling of forms, reporting, checking documentation for correctness, etc. [15, 16].

ELMA RPA business process automation technology was added to the unified register of Russian programs on December 30, 2020, by Order No. 799 of the Ministry of Digital Affairs of Russia dated December 30, 2020. It is located in the category "Enterprise Resource Management" and subcategory "Robotic Process Automation" [17, 18].

Before simulating the robot process, the following ELMA system user accounts were created for testing:
- A customer account with a requester1 login and a 9-character password.
- Task distributor account with login rp1 and 9-character password.
- Task performer account with login prep1 and 9-character password.
- Task checker account with login rev1 and 9-character password.

To start the browser, we used a function from the palette called "Start Program" and wrote the path to the browser in the "Command to Start Process" parameter (see Figure 1). After opening the browser, the robot clicks on the address bar and enters a link to the ELMA system there [19, 20].

The ELMA login window opens, where the robot fills in the login and password fields with the customer account and clicks the "Login" button using the "Click" action. In the ELMA Welcome window, the robot clicks the button with the text "Go to SDC service page", in the next service window it clicks the button "New request" to open the catalog with the names of document management services. In the service catalog, the robot clicks to open a drop-down list to select a request category. The window is updated to display the desired requests from the selected category. The robot selects the service by clicking, then the robot waits for 25000 milliseconds (25 seconds) for the request to be generated during this time (see Figure 2).
In the request window, the robot fills in all the mandatory fields about the customer, and the client, before sending the request to the robot, you need to fill in the field with the date and time of the expected deadline (Expected Deadline) of the request. To determine the date and time the "Script" function was used. The DateTime structure is used in C# to work with dates and times. To get the current time and date, the DateTime.Now property was used, which takes the current date and time of the computer. The AddDays() method was used to add dates (see Listing 1) [21, 22].

```csharp
context.dateparam = DateTime.Now.AddDays(14).ToString("dd.MM.yyyy");
context.timeparam = DateTime.Now.ToString("HH:mm");
```

**Listing 1.** Initializing the date and time of the Expected Deadline field.

The "Enter text" action was used to enter the date and time in the request. To send the request for distribution through the "Click" action, click on the "Submit a request" button. Then the request gets the status "In progress", and the robot is given a waiting time of 60000ms (1 minute), so that during this time the request is sent to the task distributors and gets the status "Sent", to check the status the robot is given the action "Read OCR text", if the status has changed from "In progress" to "Sent", the robot in the log file records the current date and text with the words "The request got Sent status", otherwise it records "The
request has not got Sent status". The conditions are set to the robot by the "Gateway" execution (see Figure 3) [23, 24].

![Gateway](image)

**Fig. 3.** Using the "Gateway".

Then the formed name of the request is determined and written to the variable nazvanie_zayavki using the "Write to clipboard" action (see Figure 4).

![Writing the name of the request into the nazvanie_zayavki variable](image)

**Fig. 4.** Writing the name of the request into the nazvanie_zayavki variable.

To log out of a customer account, the robot has been set to "Click" in the upper right corner to reveal a list of account information and an exit button. The robot then clicks the "Logout" button.

The robot logs in with the task distributor account in the application. In the ELMA welcome screen, the robot clicks on the button with the text "Go to SDC service page", on the next service window it clicks on "AC Manager Page". Then selects "Task Management", where the field "Select Request" writes the name of the service from the variable nazvanie_zayavki to search for the request from the database, then the data about our request appears (see Figure 5) [25, 26].

![Information about the request is in the "Task Management" window](image)

**Fig. 5.** Information about the request is in the "Task Management" window.
Tasks for the executor and the checker are not distributed, for this robot selects the task
"Prepare", i.e. the executor by clicking in the "Choice" section and performs the "Enter text"
action in the "Select Action" field, writes the text "Destinate Selected Task" and then by
clicking selects from the drop-down list. Then the panels "Add Selected Tasks Executors"
and "Employee Load Attributes" are displayed for entering data [27, 28].

In the field "Select Employee to Add (Competency)" the robot, through the action "Enter
text", writes the username of the task executor prep1, from the drop-down list the data about
the executor appears, then the robot given the action of pressing "Enter" with the action "Key
combination". After that, in the "Select Employee to Add" field, the task performer we
selected is automatically written.

Then the robot fills in the date and time in the fields "Date Start for Employee Load" and
"Date End for Employee Load" (listing 2).

Then the robot clicks on the "Destinate Task" button and waits for 7000 milliseconds (7
seconds) for the page to refresh. The robot does the same, but in the field "Select Employee
to Add (Competency)" the robot specifies the username of the performer's account, i.e. rev1.

context.DStartELparam =
    DateTime.Now.AddDays(7).ToString("dd.MM.yyyy");
context.DEndELparam =
    DateTime.Now.AddDays(7).ToString("dd.MM.yyyy");
context.TStartELparam = "14:30";
context.TEndELparam = "17:30";

Listing 2. Initializing "Date Start" and "Date End" for employee load.

Now the robot needs to specify the person responsible for the request, to do this in the
window using the "Click" action, the robot clicks the "Refresh" button, after the robot clicks
on this button in the drop-down list will remain only the performer and the checker of the
request. The robot selects the checker as the one responsible for the request using the "Enter
text" action, writes the checker login (rev1), and presses the Enter button using the "Key
combination" action. Next, the robot enters the text "Request for Test" in the "SDC
Comment" field and presses the "Finish Destination" button. After pressing this button, the
robot waits for 9000 milliseconds (9 seconds) for a new window to be formed. The new
window should display a green text bar at the top that says "The fields are correct!" and
"Script sent to destination workflow instance", if it does, the robot should write in the log file
"All fields filled in correctly with preparer and reviewer" and "Script sent to destination"
otherwise "Fields filled in incorrectly with preparer and reviewer" and "Script not sent to
destination". Next, the robot logs out of the task distributor account and logs in from the
performer account [29, 30].

Within the account's user interface, the automaton is engineered to initiate the 'My tasks'
module by selecting the "My task" button. This action prompts the emergence of an input
field, characterized by the question "What should I look for?". The automaton is designed to
input the specific task name into this field, which refers to the data stored in the
"nazvanie_zayavki" variable. Following this data entry, it triggers a database query by
clicking on the "Search" button. The search operation effectively filters the list, leaving only
the task that was allocated via the task distribution algorithm. The robot, upon choosing this
specific task, opens a dialogue window titled "Confirm Independence" pertaining to the
selected task. As the final stage of the operation, the automaton is required to select the
"Confirm No Restrictions" option. This step affirms that there are no existing restrictions
associated with the task at hand [31, 32].

In the task window, the robot enters the number "2" in the "Total time spent" field, after
which the "Hours to Charge" field is also automatically filled with the number "2". In the
"Total time spent – comments" field the robot enters the text "Test" and clicks the "Add
Worktime" button waiting 4000 milliseconds (4 seconds) (see Fig. 6). To complete the task, the robot clicks the "Done" button with the "Click" action, a confirmation window appears with the text "Are you sure? (Please approve that you've filled the results and timesheets.)" clicks the "Yes" button.

The robot waits 9000 milliseconds (9 seconds) to form a new page window, and the text at the top of the window should be on a green background with the text "Task completed". After checking, the robot writes "Task preparer done" on the log file, otherwise, it writes "Task preparer not done". The robot logs out of the preparer account and logs in with a user who has the authority to check the task. It does the same thing as it did from the user account, only at the end it writes "Task reviewer completed" or "Task reviewer not completed" in the log file. After that, the robot waits for 60000 milliseconds (1 minute). Since the reviewer is also set as the person responsible for the request, he has to send the task back to the customer that the request is done.

A window opens where the robot writes "Test" in the "Result Comments" field, then selects a score in the score field with the "Click" action and clicks the "Send to the Requestor" button. The robot then waits for 10000 milliseconds (10 seconds) for a new window to open, where it checks the text on the green background that says "Task completed" after finding such text in the log file, it writes "Send to the Requester completed" or "Send to the Requester not completed".

To complete this, the robot logs out of the checker account and logs in to the ELMA system from the customer account. Then it clicks on "My tasks" using the "Click" action. The robot enters the name of the request, i.e. the data from the nazvanie_zayavki variable, in the input field with the text "What to look for?" and then clicks on the "Search" button to search for the request from the database. It opens the task and selects "Approve and Close Request" in the field named "Approve" with a click, then puts a score and clicks the "Save and Close" button. The robot waits for 5000 milliseconds (5 seconds) for a new window to form. In the new window in the field "Request" with a link to the application clicks to go, in the application robot checks the status of the application, if the status of the application says "Closed", then in the log file writes "The application has acquired the status Closed" in other cases, "The application has not acquired the status Closed" (see Figure 7).

![Fig. 6. Example of a timesheet filled out by the robot.](image)

![Fig. 7. Example of a closed application created by the robot.](image)
4 Conclusion

Within the framework of the work the set goal was achieved: a robot using ELMA RPA business process automation technology was developed to test document workflow applications in ELMA BPM business process management system [33].

The great potential of RPA and the necessity of implementation of robotization of business processes in companies were revealed, and an example of successful use was given.

Thus, we can conclude that RPA is a new and very promising technology, which allows us to radically change the very approach to the execution of repetitive tasks associated with manual data entry and processing and create software robots.

References

15. Z. M. Gizatullin, R. M. Gizatullin, M. G. Nuriev, 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), 120-123 (2020)
16. G. Marin, D. Mendeleev, B. Osipov, and A. Akhmetshin, E3S Web of Conferences 178, 01033 (2020)
27. K. Kulagin, M. Salikhov, R. Burnashev, 2023 International Russian Smart Industry Conference (SmartIndustryCon), 690-694 (2023)
29. R. Burnashev, A. Enikeeva, I. F. Amer, A. Akhmedova, M. Bolsunovskaya, A. Enikeev, Lecture Notes in Networks and Systems 544 (2023)