Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 11, No. 2, June 2023, pp. 349~365
ISSN: 2089-3272, DOI: 10.52549/ijeei.v11i2.4645

Development and Evaluation of a High-Performance Electrochemical Potentiostat-Based Desktop Application for Rapid SARS-CoV-2 Testing

Faisal Ahmed Assaig[1], Teddy Surya Gunawan[2], Anis Nurashikin Nordin[3], Rosminazuin Ab. Rahim[4], Zainiharyati Mohd Zain[5], Rozainanee Mohd Zain[6], Fatchul Arifin[7]

[1,2]Electrical and Computer Engineering Department International Islamic University Malaysia, Kuala Lumpur, Malaysia
[2]School of Electrical Engineering, Telkom University, Bandung 40257, Indonesia
[3,4]MEMS-VLSI Research Unit, ECE Department, International Islamic University Malaysia, Kuala Lumpur, Malaysia
[5]Electrochemical Material and Sensor (EMaS) Research Group, Faculty of Applied Sciences, University Teknologi MARA, Shah Alam 40450, Selangor, Malaysia
[6]Institute for Medical Research (IMR), National Institute of Health (NIH), 40170 Shah Alam, Selangor, Malaysia
[7]Department of Electronic and Informatics Engineering, Universitas Negeri Yogyakarta, Yogyakarta, Indonesia

Article Info

Article history:

Received Mar 30, 2023
Revised Apr 29, 2023
Accepted May 13, 2023

Keywords:

SARS-CoV-2
Electrochemical potentiostat
Web technologies
Desktop application
Virus detection
Rapid testing
Cross-platform apps
User-friendly interface

ABSTRACT

The COVID-19 pandemic has necessitated the development of rapid and trustworthy diagnostic tools. Reverse transcription-polymerase chain reaction (RT-PCR) is the gold standard for detecting SARS-CoV-2 but has cost and time constraints. The sensitivity, specificity, and low cost of electrochemical biosensors make them an attractive alternative for virus detection. This study aims to develop and evaluate a high-performance desktop application for an electrochemical potentiostat-based SARS-CoV-2 test device, with a user-friendly interface that automatically interprets results, to expedite the testing process and improve accessibility, particularly in resource-limited settings. The application was built with the Electron framework and the HTML, CSS, and JavaScript programming languages. Our findings indicate that the developed electrochemical potentiostat-based desktop application demonstrates high accuracy compared to commercial software, achieving rapid detection within 30 seconds. The graphical user interface was found to be straightforward and user-friendly, requiring minimal training for efficient system operation. Our electrochemical potentiostat-based desktop application represents a valuable tool for rapid SARS-CoV-2 testing, particularly in settings with limited resources. This research contributes to developing rapid and reliable diagnostic tools for SARS-CoV-2 and potentially other pandemic-causing viruses, addressing the pressing need for improved public health surveillance and response strategies.

Corresponding Author:

Teddy Surya Gunawan
Electrical and Computer Engineering Department,
International Islamic University Malaysia
Kuala Lumpur, Malaysia.
Email: tsgunawan@iium.edu.my

1. INTRODUCTION

The global pandemic of COVID-19 caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) has resulted in widespread morbidity and mortality. More than 761 million confirmed cases and nearly 6.9 million deaths have been attributed to COVID-19 as of March 2023 [1]. The virus can cause severe respiratory and cardiovascular damage, and identifying and isolating infected individuals as soon as possible is crucial for preventing its spread [2]. Numerous virus detection methods, including molecular, antigen, and serological, have been developed to facilitate rapid and efficient screening [3]. However, an urgent unmet demand remains for a sensitive, fast, and reliable diagnostic tool to address this global problem.

Due to its high sensitivity and specificity, reverse transcription-polymerase chain reaction (RT-PCR) was the primary diagnostic tool for identifying infected individuals in the early stages of the pandemic [4]. However, RT-PCR-based tests are time-consuming and necessitate extensive sample preparation, expensive lab equipment and facilities, and skilled personnel, which are only routinely available in first-world countries [3]. Furthermore, the extensive training required to operate these systems adds to the burden on the medical staff, who must invest significant time and resources to master their usage. Consequently, these drawbacks impose additional costs and delays on medical institutions conducting tests, as they must allocate more resources for training and wait longer for results. This highlights a research gap in developing a user-friendly desktop application that offers a faster startup and reduced training time. Such an application would enable medical staff to operate the system without extensive prior knowledge or experience. By enhancing automation and user-friendliness in the software application, users can operate the system more efficiently and obtain accurate results more rapidly.

Alternative virus detection technologies, such as point-of-care testing (POCT) procedures, have been created to address the limitations of RT-PCR. It enables the detection of viruses outside of a laboratory setting and can assist in overcoming some of the limitations of RT-PCR [3]. Nucleic acid detection methods like CRISPR (clustered regularly interspaced short palindromic repeat) [5] and LAMP (loop-mediated isothermal amplification) [6] can detect SARS-CoV-2.

Despite the advent of alternative virus detection techniques, there is an urgent need for a reliable, cost-effective, and quick diagnostic tool for SARS-CoV-2. In response, we created and evaluated a high-performance desktop application for an electrochemical potentiostat-based SARS-CoV-2 test apparatus. Our primary objective is to develop a user-friendly interface that allows automatic interpretation of outcomes. Our study bridges the gap between RT-PCR's limitations and the need for a fast and reliable SARS-CoV-2 diagnosis technique.

Medical software development, particularly for diagnostic applications, has become an essential part of modern healthcare systems. The increasing dependence on digital technologies to support medical diagnosis and treatment necessitates strict adherence to international standards to ensure such software's safety, reliability, and efficacy. One widely recognized standard in this domain is the International Electrotechnical Commission (IEC) 62304:2006+A1:2020, "Medical device software - Software life cycle processes," which provides a framework for the development and maintenance of medical device software [7]. This standard encompasses the entire life cycle of medical device software, including planning, development, testing, validation, and post-market surveillance.

Several other standards and guidelines have been developed to complement IEC 62304, addressing various aspects of medical software development. ISO 14971:2019, "Medical devices - Application of risk management to medical devices," is one such standard that focuses on the systematic identification, assessment, and mitigation of risks associated with medical devices [8]. In the context of software development, the guidelines provided by ISO 13485:2016, "Medical devices - Quality management systems - Requirements for regulatory purposes," are also relevant, as they establish the quality management system requirements for organizations involved in the design, production, and distribution of medical devices [9]. Furthermore, the US Food and Drug Administration (FDA) has published guidance on software validation for medical devices, emphasizing the importance of rigorous testing and validation procedures to ensure that software performs as intended [10].

The rapid advancements in digital technologies have led to the development of medical software for diagnostic applications, necessitating strict adherence to international standards to ensure safety, reliability, and efficacy. Despite comprehensive standards, such as IEC 62304, ISO 14971, and ISO 13485, challenges persist in consistently integrating these guidelines into developing and maintaining medical device software. Consequently, the problem is developing a high-performance electrochemical potentiostat-based desktop application for rapid SARS-CoV-2 testing while ensuring compliance with relevant international standards and focusing on safety, risk management, and quality assurance throughout the software life cycle.

2. ELECTROCHEMICAL BIOSENSORS

Electrochemistry is the study of the relationship between electricity and chemical reactions. Electrochemistry is utilized in biosensors to detect biomolecules by transforming their chemical signals into quantifiable electrical impulses. Due to their great sensitivity, selectivity, and low cost, electrochemical biosensors are frequently applied in biomedical and environmental settings [11].

Electrochemical biosensors use an electrochemical transducer to transform the biochemical interaction between an analyte and a bioreceptor into a quantifiable electrical signal. The bioreceptor may be an enzyme, antibody, nucleic acid, or any other biomolecule that interacts specifically with the target analyte. The electrical signal produced by the transducer is proportional to the analyte concentration, enabling quantitative detection [11]. The development of electrochemical biosensors involves the design of a suitable

bioreceptor, selecting the transducer, and optimizing the detection scheme. Recent advances in nanomaterials, microfabrication, and signal processing have significantly improved the performance and versatility of electrochemical biosensors [12]. Several electrochemical methods can be utilized for biosensing, including voltammetry, impedance spectroscopy, and chronopotentiometry.

2.1. Voltammetry

Voltammetry is an electrochemical method for measuring the current response of an analyte as a function of an applied voltage. A voltage is given to the working electrode in voltammetry, which causes an electrochemical reaction between the analyte and the electrode surface. The ensuing current response is measured and used to calculate the analyte concentration. Biosensors often employ cyclic voltammetry (CV), differential pulse voltammetry (DPV), and square wave voltammetry (SWV) [13]. Utilizing carbon nanotubes, graphene, and other nanomaterials as electrodes and integrating microfluidics and electrochemical sensors for point-of-care applications are recent breakthroughs in voltammetry [14].

2.2. Impedance Spectroscopy

Impedance spectroscopy is an electrochemical technique that measures the resistance of an analyte to the flow of alternating current (AC) at different frequencies. The impedance of the analyte is affected by various factors, including the analyte's concentration, the bioreceptor's nature, and the electrode surface's characteristics. By analyzing the impedance spectrum, it is possible to determine the concentration of the analyte and other parameters related to the binding of the bioreceptor to the analyte [15]. Impedance spectroscopy has been applied to detect various analytes, including proteins, nucleic acids, and bacteria. It has shown great promise for the rapid and sensitive detection of biomolecules [16].

2.3. Chronopotentiometry

Chronopotentiometry is an electrochemical technique that analyses the potential response of an electrode to a constant current during a predetermined time interval. This approach examines electrochemical processes involving charge transfer or mass transport kinetics by calculating the electroactive species' rate constants and diffusion coefficients [17]. Chronopotentiometry is commonly used in electrochemical biosensors to detect biomolecules, such as glucose, cholesterol, and uric acid [18]. A typical chronopotentiometry measurement applies a constant current to the working electrode, and the potential is measured as a function of time. The resulting current-potential curve can be used to extract information about the kinetics of the electrochemical reaction [19].

Voltammetry examines the current response of an analyte as a function of the applied potential. In contrast, impedance spectroscopy evaluates the analyte's resistance to the flow of alternating current at various frequencies. Chronopotentiometry analyses the potential response of an electrode to a constant current supplied over a defined period. Voltammetry is more sensitive than impedance spectroscopy and chronopotentiometry regarding detection sensitivity due to its capacity to detect minor changes in current at low analyte concentrations. However, impedance spectroscopy and chronopotentiometry can be used to study electrochemical processes involving charge transfer or mass transport kinetics, which can be beneficial for designing and optimizing biosensors.

Our work chose voltammetry as the electrochemical approach for detecting SARS-CoV-2 due to its excellent sensitivity, selectivity, and capacity to deliver quick, real-time biomolecule measurements. In addition, recent advances in nanomaterials and microfabrication have substantially enhanced the performance and versatility of voltammetric biosensors, making them a promising method for point-of-care diagnostics [11, 20].

3. DESKTOP APPS FRAMEWORK BASED ON WEB TECHNOLOGIES

Developing desktop apps for electrochemical biosensors necessitates a user-friendly interface that can analyze data effectively and produce real-time results. Due to its support for .NET libraries and easy interaction with hardware and sensors, such as PalmSens, WinForms, and WPF have traditionally been the most popular frameworks for developing desktop applications.[21]. Various frameworks may be used to build desktop applications. JavaFX is a well-known framework that enables developers to create dynamic and rich user interfaces. Electron is a web-based framework that allows developers to create cross-platform desktop applications using HTML, CSS, and JavaScript [22]. However, as web technologies have emerged, a new framework known as Electron has gained favor among developers. Electron is an open-source framework that enables the creation of cross-platform desktop applications utilizing web technologies such as HTML, CSS, and JavaScript. Electron has been used to create programs for multiple platforms, including Windows, macOS, and Linux, and offers a native-like experience for consumers.

Electron is superior to WinForms and WPF because it can construct platform-independent apps, meaning that the same code may be used across multiple operating systems. It reduces the need to build different code for each platform, decreasing development time and expenses. In addition, Electron enables simple integration with web-based technologies, which is advantageous for creating web-based electrochemical biosensors.

We chose the Electron framework to create our desktop application in the context of our study because of its cross-platform interoperability and flexibility in integrating web technologies. It enabled us to design a user-friendly interface that anybody can access and utilize regardless of their operating system. In addition, using web technologies permits incorporating other web-based technologies that may be advantageous for electrochemical biosensing, such as cloud-based data storage and machine learning techniques. In conclusion, whereas WinForms and WPF have historically been the most popular frameworks for desktop application development, the arrival of the Electron framework offers developers a new, platform-independent method that is well-suited for the construction of electrochemical biosensors.

Web-based desktop application development has various advantages over traditional desktop program development. Initially, a cross-platform desktop program reduces the developer's development time. It allows the developer to construct cross-platform desktop apps for macOS, Windows, and Linux in less time than with native software development tools [23]. Secondly, customizable and easy-to-learn UI design because web technology desktop applications are based on familiar technologies, such as HTML, CSS, and JavaScript, to build desktop applications. Moreover, it has a large community that contributes to and supports it. Two primary desktop web app frameworks are being maintained: Electron and NW.js [24].

Electron is an open-source and multi-cross-platform framework for developing native apps using web technologies like JavaScript, HTML, and CSS on node.js for the backend. Electron operates by rendering web pages using the chromium engine [25]. Also, the framework utilizes a modern Chromium engine that supports cutting-edge technology, such as React, Angular, and Vue.js. Desktop apps designed and developed with Electron would offer more excellent maintainability by employing a single source code, cutting the development cost. It helps to reduce the needed time for development. However, The traditional method of developing desktop applications employs multiple development teams for different operating systems, resulting in many versions of source code, each of which is linked to the specific operating system [26].

The main objective of this research is to develop a user-friendly desk application that allows the operator to generate results quickly. The desktop application is designed to provide users with a high-quality user interface. Furthermore, the operator does not require lengthy training to utilize all the desktop application's features. A short training period is sufficient for the operator to use the application's full potential. Even though there is an increasing demand for web applications, desktop applications still exceed web applications in several areas. Web apps depend and rely entirely on Web Browsers as the client interface [27] and internet connectivity.

4. DEVELOPMENT OF THE NACOTS DESKTOP APPLICATION

This paper stems from a multidisciplinary research project titled "Nanosystem for Covid-19 DNA/Antibodies On The Spot Test (NACOTS) (CV02211032)", which brought together experts from various fields, including electrochemistry, sensor design, and hardware and software development. The collaboration of these diverse specialists enabled the creation of a comprehensive and innovative solution for COVID-19 testing. This paper is one of several that have been or will be published, such as [28-31], showcasing this groundbreaking project's different facets and outcomes. The collective efforts and expertise of the research team have resulted in a series of valuable contributions to the scientific community, addressing the urgent need for efficient and accurate diagnostic tools in the fight against COVID-19.

This research focused on developing a desktop application to compete with current desktop applications for lab tests. After studying a few of these desktop applications, it was found that these lab software tests lack essential features, such as a user-friendly graphical user interface. Hence, the desktop application is developed to have a user-friendly, modern design based on user acceptance and feedback. The main objective of the application is to produce the final results (positive or negative) without the need for a long training period to interpret the numerical results. However, data visualization is another feature that the application has. This feature allows the operator to double-check and confirm the generated results from the app.

4.1. Software Design

The server side was designed to accept and store data, and the device is based on a potentiostat chip, as depicted in Figure 1. The device, desktop application, and Server side are system components. The desktop application connects to and controls the device and receives and processes sample data. The application

displays processed data and indicates whether or not the patient is positive. These are the modules for desktop applications:

- **Device Controller**: this module communicates to the device. After the connection is established, the module sends the instructions to the device and receives the device's data.
- **Data preprocessing**: the received data requires processing to convert into numerical data that can be processed by threshold algorithm.
- **Threshold Algorithm**: this module contains the algorithm which decides the positive and negative sample based on the numerical data.
- **API Call**: this module sends a request to the server containing numerical data in JSON format for further study.

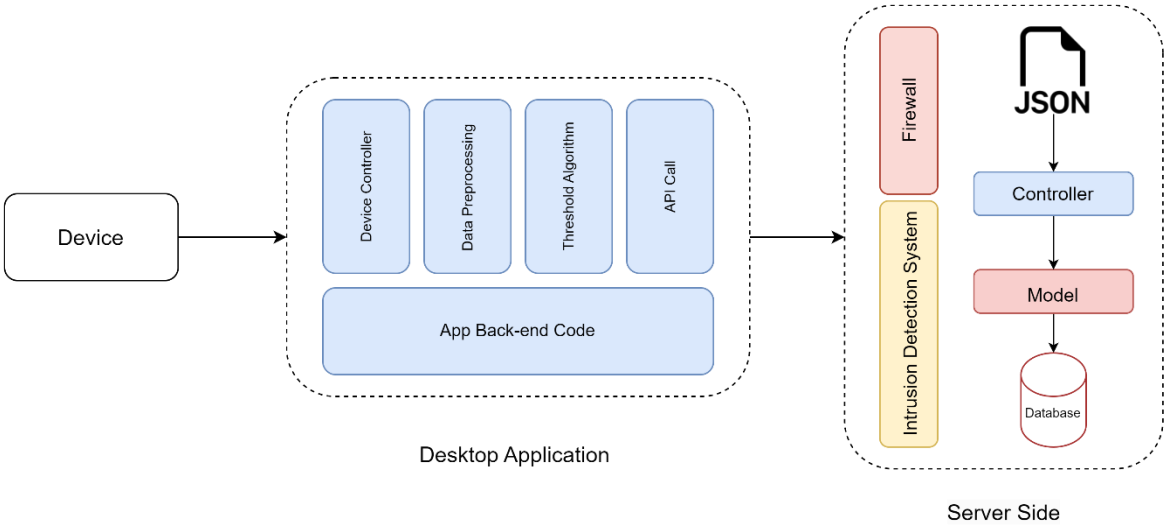

Figure 1. System Architecture

4.2. Hardware Design

Differential pulse voltammetry (DPV) is the electrochemical technique employed for this device. A portable potentiostat microcontroller is used to accomplish this approach. In electrochemical research, a potentiostat is an essential tool that enables researchers to apply a potential to a system to detect and assess the accompanying current or vice versa. Therefore, the gadget is based on the Emstat Pico potentiostat module. It is a module that facilitates using numerous electrochemical techniques, including DPV, CV, and EIS. As shown in Figure 2, the module can be linked via serial communication utilizing the computer's USB-serial interface. Emstat Pico is a fully integrated module with a tiny footprint and low power consumption, enabling the system's portability. Moreover, its precision is remarkable. MethodSCRIPT scripting language is employed for module communication. MethodSCRIPT enables programmers to quickly program the potentiostat module using a legible script through a serial (TTL) connection. All supplied electrochemical methods can be executed using the scripting language. Additionally, Figure 3 shows the NACOTS device in action.

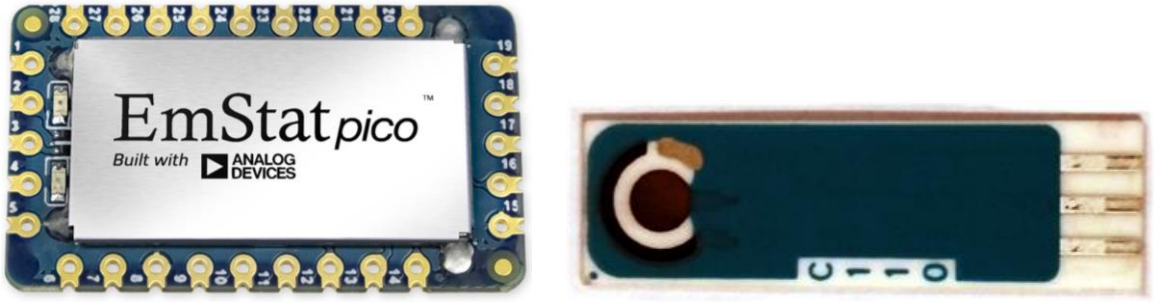

Figure 2. EmStat Pico [29] and the developed nanogold deposited biosensor

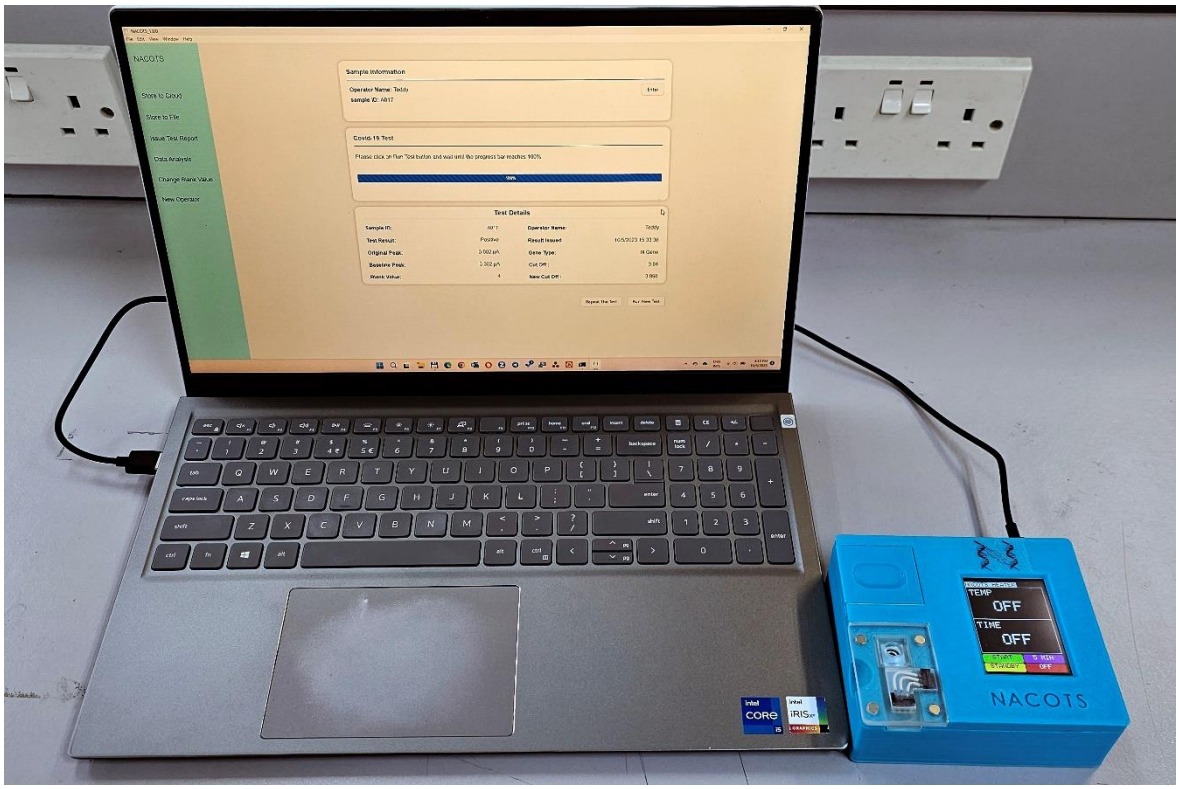

Figure 3. NACOTS hardware and software in action

4.3. Data Processing

Figure 4 depicts the data processing sequence commencing when the user initiates the test by clicking the "Run Test" button within the desktop application. First, the application identifies and connects to the appropriate port the potentiostat device uses. Subsequently, the application sends instructions to the potentiostat chip, initiating the sample analysis in real-time.

During the second stage, the application receives raw data from the device as it processes the sample. This raw data is then passed through a preprocessing module that converts it into numerical data in a structured format. In the third stage, the algorithm processes the numerical data further, transforming it into readable data with standard international (SI) base units. This conversion ensures that the data is easily interpretable and consistent across different tests and devices.

In the final stage, the algorithm evaluates the processed data and generates the ultimate user results, classifying the sample as negative or positive for SARS-CoV-2. This streamlined data processing workflow enables the rapid and accurate determination of test results, providing users with a convenient and efficient tool for COVID-19 testing.

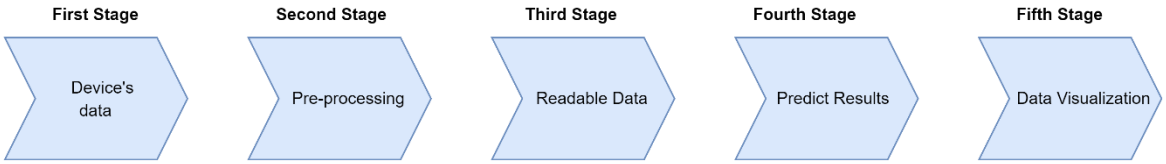

Figure 4. Data processing stages

4.4. Result Generation Algorithm for SARS-Cov-2 Detection

The system operates on an electrochemical Differential Pulse Voltammetry (DPV) signal. A six-step algorithm is employed to distinguish between positive and negative samples. First, the Savitzky–Golay filter is applied to the DPV input numerical data (x) to smooth the plot and minimize noise, as illustrated in Figure 4. Second, the peak value of the current ($p(x)$) is identified. In the third step, the baseline is determined, representing the current level measured when no electrochemical reaction is occurring. The baseline is established by drawing a line from the minimum point (A) before the value $p(x)$ to the minimum point (B) after the value $p(x)$.

In the fourth step, a line is drawn from the baseline to the value $p(x)$, and the length of this line represents the new peak value ($Peak_{measured}$). The Delta value is calculated by determining the difference between the peak blank and the new peak values ($Peak_{measure}$), mathematically represented as $\Delta = Peak_{blank} - Peak_{measured}$. Lastly, the Δ value is compared to the predetermined threshold value, as demonstrated in Pseudocode Algorithm 1. This comparison enables the algorithm to classify the sample as either positive or negative for SARS-CoV-2, providing a reliable and efficient method for detecting the virus in test samples.

Pseudocode Algorithm 1

$s = savitzkyGolay(current)$
$P = p(s)$
$range_A, range_B = findRange(P)$
$A = \min(range_A)$
$B = \min(range_B)$
$line_{base} = findBaseLine(A, B, voltage)$
$Peak_{measured} = distance(line_{base}, P)$
$\Delta = Peak_{blank} - Peak_{measured}$
$If\ \Delta\ Threshold$
    $Covid\ 19\ sample\ is\ positive$
$else$
    $Covid\ 19\ sample\ is\ negative$
$endif$

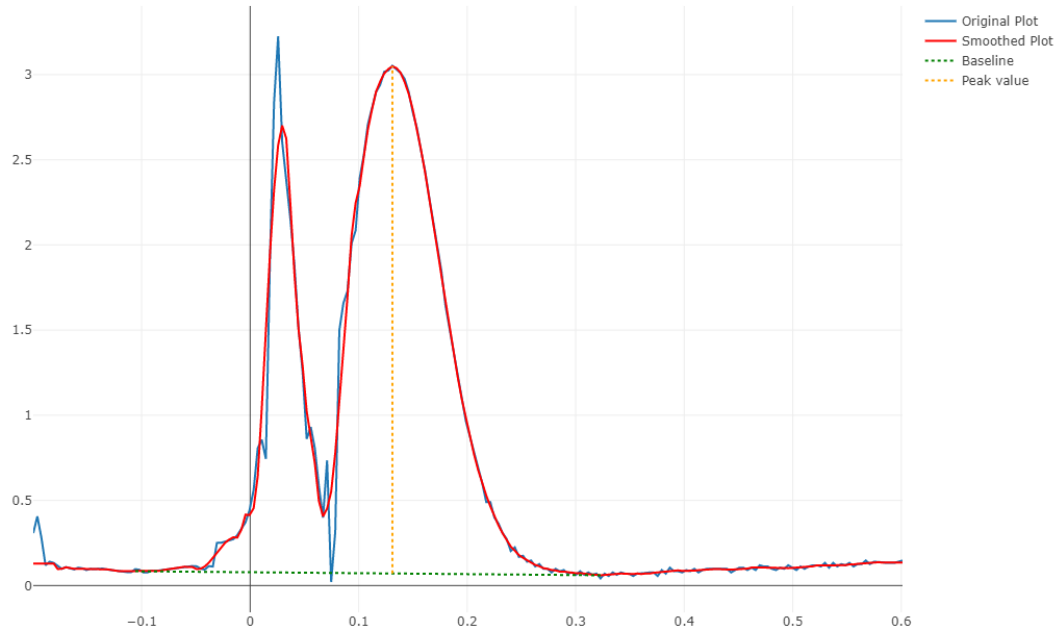

Figure 5. Data Visualization

4.5. Server-side Architecture

The server-side implementation employs a robust architectural pattern, which divides the system into three primary components: Model, View, and Controller (MVC). This architecture facilitates the efficient handling of API calls and streamlines the data flow between components, as shown in Figure 6. The Controller is responsible for processing client requests containing the user information and test results in JSON format. After processing the request, the Controller returns a response to the desktop application, indicating whether the data has been successfully stored in the database or not. The Model serves as the interface between the

database and the business logic. It receives data from the Controller, processes it according to the defined logic, and stores the resulting information in the database. This separation of concerns within the MVC architecture enables easier server-side system maintenance, scalability, and adaptability. This efficient server-side architecture ensures seamless integration between the desktop application and the database, providing users with a reliable and responsive platform for SARS-CoV-2 testing and data management.

- **Backend**: it is developed using node.js language with express.js framework. Ubuntu server is the selected operating system for the server side.
- **REST API**: The architectural style for API is based on representational state transfer (REST). The information is transferred with JSON formats via HTTP protocol. The HTTP header contains metadata and authorization.

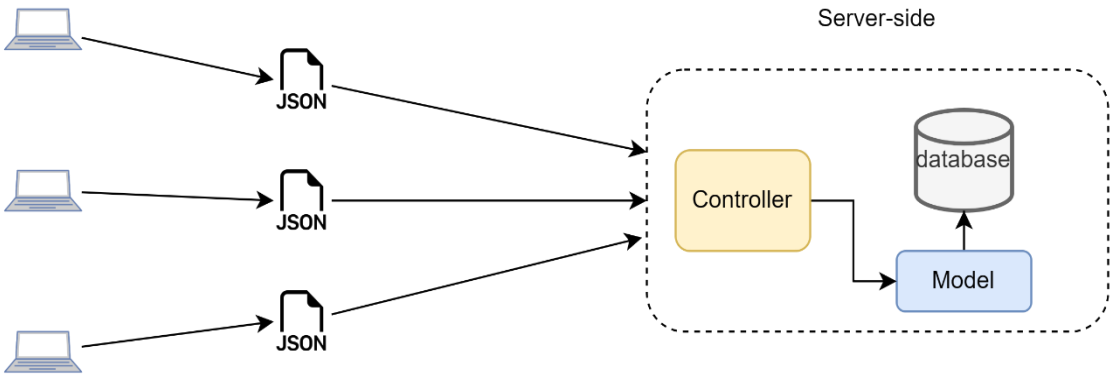

Figure 6. Server-side Architecture

4.6. Security in E-healthcare System

In software development, system security is frequently considered of secondary importance. Despite this, the development of an e-healthcare system depends on and relies on security. The E-healthcare system contains patients' medical records and private information. This information must therefore be protected with the utmost care [32]. To prevent the loss or disclosure of the patient's medical information, which could result in a misdiagnosis or inadequate treatment. Cyber dangers within the E-healthcare system also warrant attention. Medical results of a patient are processed, stored, and analyzed on the server side of an e-healthcare system.

Due to the rapid expansion of technology, the development of E-healthcare system security must address various threats and inadequacies [33]. This intrusion detection security layer protects the server side from frequent cyberattacks on E-health care systems, such as DDOS and DOS attacks, which render the server side inoperable during an attack [34, 35]. In an E-healthcare system that must be accessible 24 hours a day, seven days a week, this is unacceptable. Therefore, an intrusion detection system detects cyberattacks before blocking or redirecting them. However, most existing intrusion detection systems are trained on obsolete data sets, resulting in a high false positive rate. In addition, the importance of preserving the security of E-health highlights the need for a modern and effective intrusion detection system with a low false positive rate and prediction latency.

5. USER INTERFACE DESIGN

5.1. Login Interface

The initial step in utilizing the application involves connecting the computer to the internet. To access the system, the operator must provide a username and password for authentication. These credentials are transmitted to the server for validation, ensuring the system's high level of security. Upon verification of the submitted credentials, the server responds to the application, either granting or denying access to the user based on the legitimacy of the provided information. Implementing this security layer is crucial for industry-grade lab testing applications, as it safeguards sensitive data and ensures that only authorized personnel can access the system. Figure 7 displays the application's login page, highlighting the user-friendly interface designed to provide operators with a seamless and secure experience during SARS-CoV-2 testing and data management.

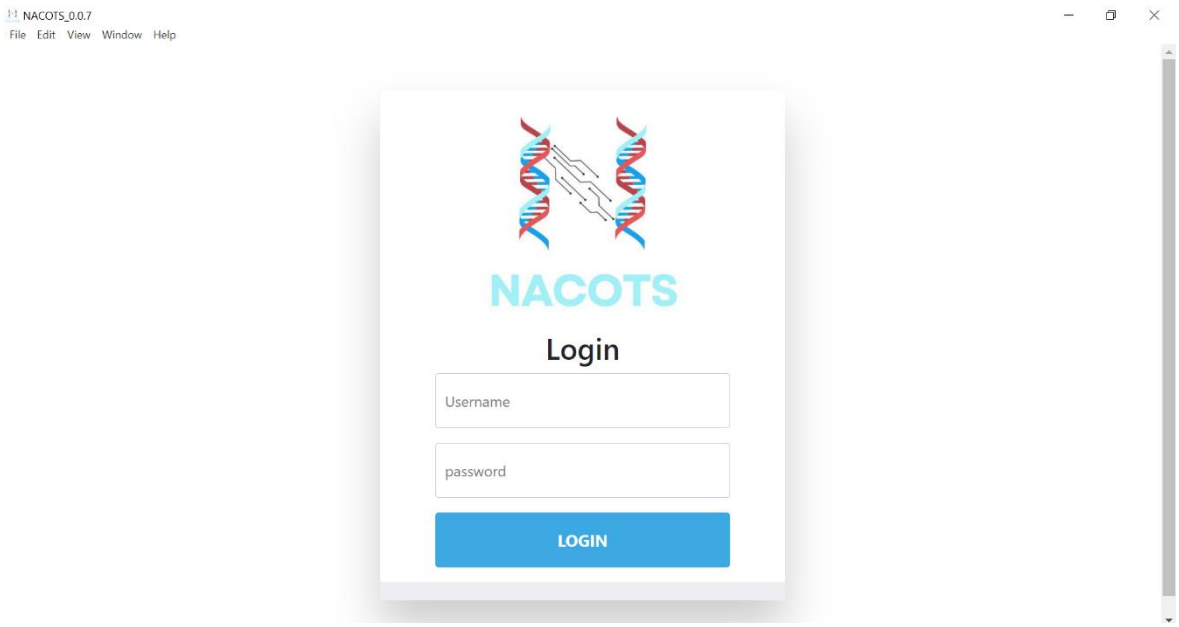

Figure 7. Login page.

5.2. Testing Interface

The application is meticulously designed to be user-friendly and requires minimal operator training. To initiate testing, the operator connects the device to the laptop and clicks the Run Test button. An additional feature of the application is its auto-connection capability. The application establishes a connection to the device with a single click and commences the testing process. Upon clicking the Run Test button, a progress bar appears and gradually fills until it reaches 100%, indicating the completion of the test, as depicted in Figure 8. This intuitive interface streamlines the testing process, allowing operators to efficiently conduct SARS-CoV-2 tests with ease and precision.

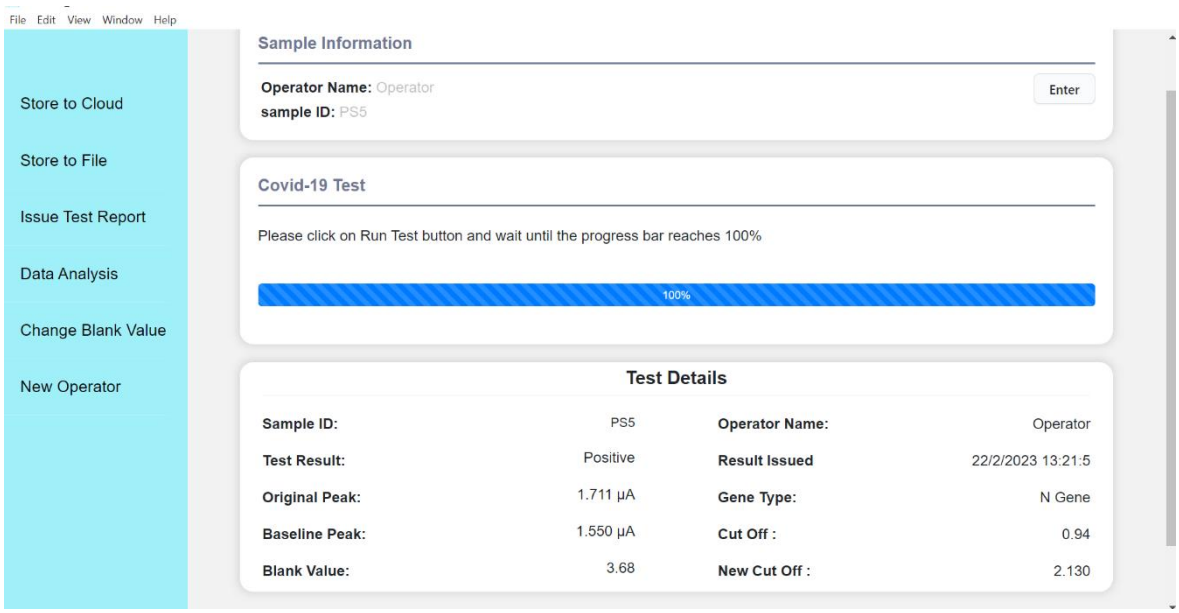

Figure 8. Test result page

The test Details Section shows the test information and final results. The test details help the operator to confirm the generated results. Additionally, the operator can repeat the test by clicking Repeat Test Button. As shown in Figure 8, the side menu bar holds multiple options:

- **Store to Cloud button**: it sends the results to the server side.
- **Store to File button**: this option saves all data into file.csv and displays data using an Excel sheet.

- **Issue Report Button**: this option generates a PDF file containing the result that can be given to the patient right after the test. This option aims to accelerate the testing time and reduce the steps of issuing the report.
- **Data analysis Button**: clicking on this button opens the data analysis page that allows the operator to have further data analysis of the test and confirm the patient results, as shown in Figure 8.

5.3. Data Analysis Interface

The Data Analysis Interface is a crucial component of the electrochemical potentiostat-based desktop application, providing operators with the tools necessary to scrutinize and validate the results of SARS-CoV-2 tests. This user-friendly interface presents a comprehensive overview of the test data, enabling operators to make informed decisions based on the analysis.

As depicted in Figure 8, the sample test returns a positive result, with a peak value of less than 3 µA, corroborating the application's prediction that the COVID-19 sample is positive. The interface displays a graphical representation of the current-voltage relationship, allowing for easy identification of peak values and trends in the data. This visual representation aids in the quick interpretation of results, enhancing the overall efficiency of the testing process.

In addition to the graphical display, the Data Analysis Interface features a Raw Data section, which enumerates the current values alongside their corresponding voltage for each data point. With 214 data points for each sample test, this comprehensive dataset provides an in-depth insight into the test results, enabling operators to explore the nuances of the data and draw meaningful conclusions. The Raw Data section also allows operators to export the data for further analysis or integration with external tools, facilitating seamless data management and sharing.

The Data Analysis Interface is integral to the electrochemical potentiostat-based desktop application for rapid SARS-CoV-2 testing. By presenting comprehensive test data in a user-friendly manner and offering advanced analytical tools, the interface empowers operators to thoroughly investigate and confirm test results, ensuring the accuracy and reliability of the diagnostic process.

6. RESULTS AND DISCUSSION

6.1. Software Benchmarking

In this section, we delve into the benchmarking process comparing the performance of our developed desktop application with the commercial application PSTrace by PalmSens. PSTrace is a user-friendly software designed specifically for PalmSens instruments, offering support for various electrochemical techniques. The primary objective of this comparison is to validate the performance, accuracy, and reliability of the developed desktop application for rapid SARS-CoV-2 testing.

As illustrated in Figure 4, the data received from the device necessitates preprocessing to be interpretable and analyzable. An experiment was conducted using four COVID-19 samples to validate the preprocessing module of the developed software. Table 1 provides a detailed experiment description, including sample information and testing parameters.

Table. 1. Test Details

Negative samples	2
Positive samples	2
Number of rounds	2

The four samples underwent testing in two separate rounds. The samples were assessed using the developed desktop application in the first round. In the second round, they were evaluated with the PSTrace application. Following each round, the results were exported in both .xlsx and .csv formats, enabling efficient data comparison and analysis. Figure 9 visually represents the compared data, highlighting the similarities and discrepancies between the two software applications. The benchmarking revealed that the developed desktop application successfully processed and analyzed the data, producing results consistent with those obtained using the PSTrace software. Moreover, the comparison demonstrated that the preprocessing module of the developed application effectively transformed raw data into a readable format, ensuring accurate interpretation and analysis.

As demonstrated in Figure 9, the first round of testing using the developed desktop application yielded results that closely aligned with those obtained from PSTrace, as evidenced by the orange plot. This outcome validates the performance of the preprocessing module employed by the desktop application, indicating that it functions as intended. For instance, the NS2 sample revealed that both the developed desktop application and PSTrace generated the same peak value, further substantiating the accuracy of the application.

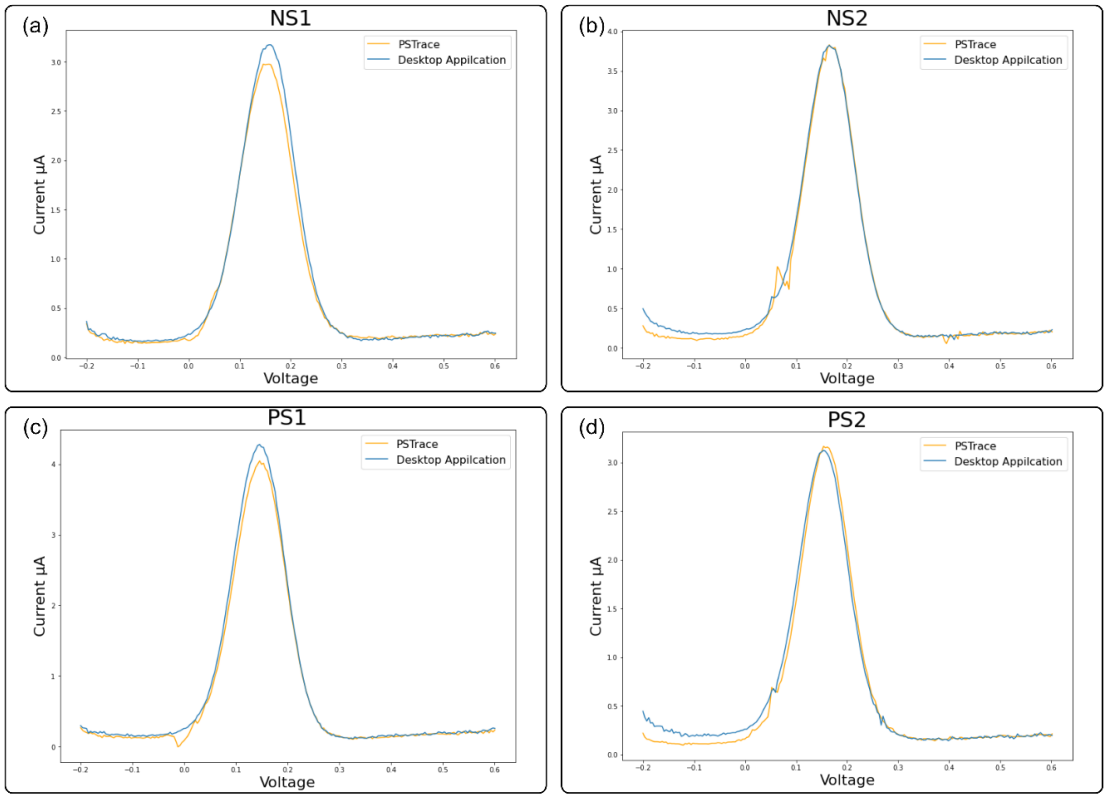

Figure 9. Sample plots: (a) Negative sample NS1, (b) Negative sample NS2,
(c) Positive sample PS1, (d) Positive sample PS2.

The PS6 sample was also tested to strengthen the comparison. Figure 10 (a) displays the peak value for the PS6 sample as determined by the developed desktop application, which measured 2.625 µA. In contrast, Figure 10 (b) presents the PSTrace plot for the same PS6 sample, with a peak value of 2.580 µA. Despite the slight difference in peak values, the results are closely related, further validating the efficacy of the developed desktop application in processing and analyzing test data.

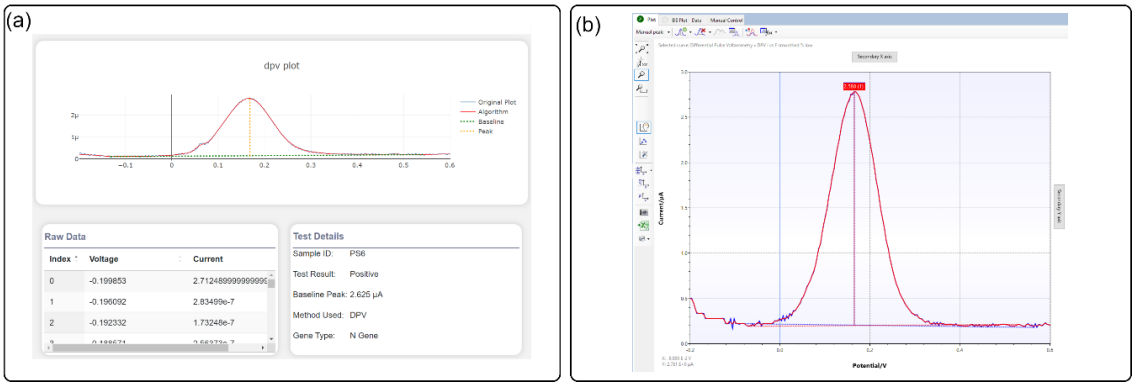

Figure 10. Comparison of NACOTS software and PSTrace for PS6 sample:
(a) NACOTS desktop application, (b) PSTace application

6.2. Test Procedure and User Guide Development

A clear and concise test procedure and a comprehensive user guide have been established to ensure the smooth operation of the developed desktop application for rapid SARS-CoV-2 testing. The test procedure consists of five straightforward steps designed to streamline the testing process and minimize the risk of user error.

1. **Sample Preparation**: The sample is heated for 35 minutes at temperatures of around 65°C, ensuring optimal conditions for subsequent analysis. Note that we were using the LAMP method for SARS-CoV-2 detection.
2. **Electrode Calibration**: To calibrate the threshold value for the specific electrode used in the test, the operator runs a blank test using a ferrocyanide solution. This procedure establishes the peak blank value, a reference point for subsequent tests. Due to the inherently low consistency of the lab-produced nano gold electrodes, measuring the peak blank each time the electrode is changed is necessary. It ensures accurate results and compensates for potential variations in electrode performance.
3. **Sample Application**: The COVID-19 sample and ferrocyanide solution are placed on the electrode and allowed to interact for 5 minutes, facilitating the electrochemical reaction necessary for analysis.
4. **Peak Blank Value Input**: The operator enters the previously determined peak blank value into the designated field within the application, as illustrated in Figure 10. This value serves as a baseline for comparison during the analysis of the COVID-19 sample.
5. **Test Execution and Analysis**: The user initiates the test by clicking the "Run Test" button. The application then analyzes the COVID-19 sample for 30 seconds, presenting the data and automatically predicting the results.

Figure 11 shows the initialization of the Peak Bank value. To further assist users in conducting tests, a pop-up note containing the instructions for the test procedure is displayed at the beginning of each new session, as shown in Figure 12. This feature provides a helpful reminder for first-time users and is a convenient reference for experienced operators. In summary, the test procedure and user guide development for the electrochemical potentiostat-based desktop application aim to simplify the SARS-CoV-2 testing process, ensuring a user-friendly experience while maintaining the accuracy and reliability of the test results.

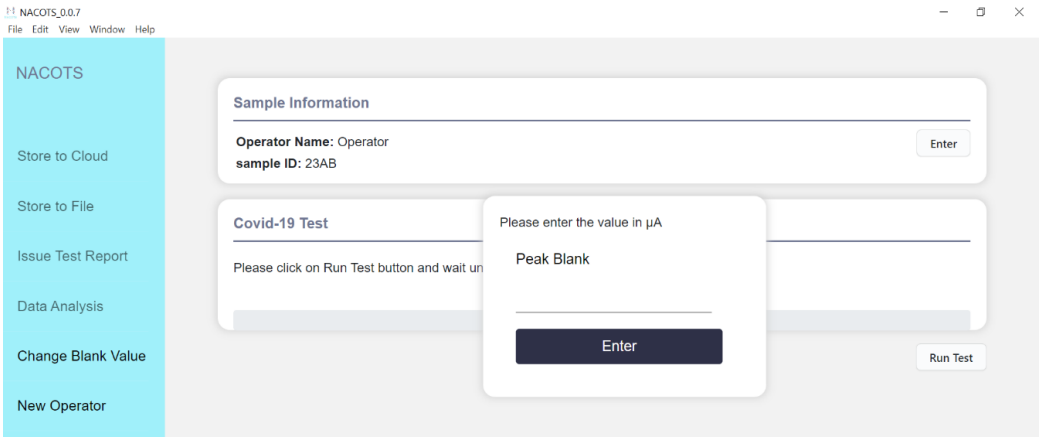

Figure 11. Enter the initial Peak Blank value

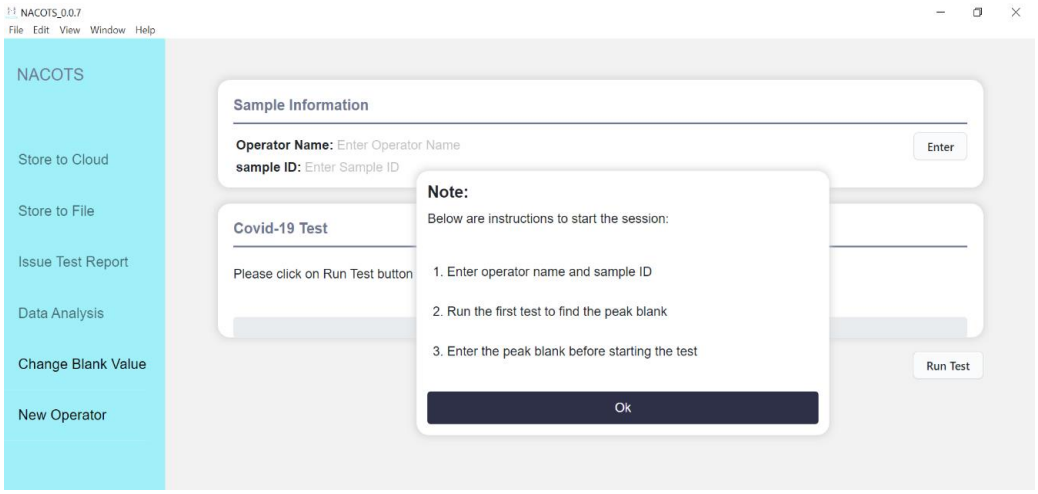

Figure 12. Instructions note to operator

6.3. Experiment on Various Samples and Analysis

A series of experiments were carried out to rigorously evaluate the performance of the desktop application, focusing on aspects such as user interface, application features, data visualization, and the generation of automatic results. In total, seven samples were examined during these experiments, and the details of each test can be found in Table 2. The comprehensive assessment of the application's performance ensured its reliability and usability, highlighting its potential as a practical and efficient tool for rapid SARS-CoV-2 testing.

Table. 2. Experiment Details

Negative samples	3
Positive samples	5
Blank value	4.174, 4.6, 5.215
Threshold value	0.94

The desktop application performed impeccably, accurately evaluating sample 3AP during testing, as illustrated in Figure 13. In Figure 13(a), the test details box displayed an extensive overview of the test results, including essential information such as the baseline peak, cut-off value, and final outcomes, enabling a thorough examination of the test's performance. Moreover, the application's predictions were precise and error-free, as demonstrated in Figure 13(a).

Figure 13(b) showcased the data analysis page, offering a more in-depth understanding of the sample. The application accurately plotted the data, visually depicting the sample's attributes. Furthermore, the application effectively employed the classification algorithm, the Result Generation Algorithm for SARS-CoV-2 Detection, allowing the application to deliver valuable insights into the sample's properties. The application's proficient performance ensured comprehensive data analysis and dependable results for the user.

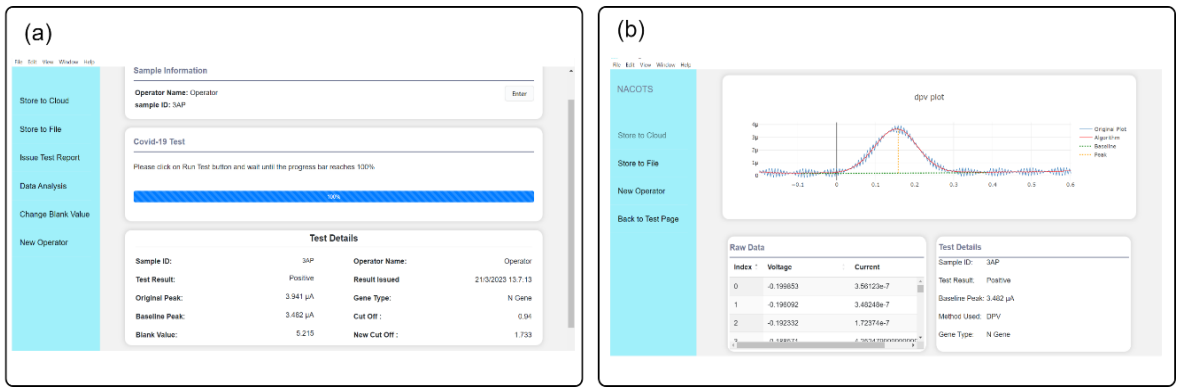

Figure 13. Sample 3AP evaluation in more details

Out of the seven samples, one false positive and one false negative were observed. This outcome is attributed to the peak values of negative samples falling within the range of approximately 2.9 µA to 3.7 µA, the same range as the peak values of positive samples. The similarity in peak values between negative and positive samples led to misclassifying some negative samples as positive. This issue could result from inconsistencies in the development of the nanogold electrode, causing the peak blank values to vary. Another potential cause could be using a static threshold, which might not adequately account for sample characteristics or electrode performance variations. Note that, the actual results was obtained using RT-PCR.

Machine learning techniques could be employed to address these challenges and enhance the accuracy of the desktop application. By leveraging machine learning algorithms, the application could dynamically adjust the threshold based on the input data, accounting for variations in electrode performance or other factors affecting the test results. This approach would enable the software to adapt to different testing scenarios and provide more accurate predictions than the static threshold method.

In conclusion, this experiment highlighted the importance of refining the desktop application to account for variations in peak values between negative and positive samples. Implementing machine learning techniques to adjust the threshold dynamically could offer improved performance and increased accuracy in analyzing SARS-CoV-2 test results, ultimately leading to a more reliable diagnostic tool.

Table. 3. Comparison between the application and actual results

Sample Name	App Results	Actual Results	Blank value	Peak Value
1AP	Positive	Positive	5.215 µA	3.731µA
2AP	Positive	Positive	5.215 µA	3.366 µA
NS2	Negative	Negative	4.600 µA	3.820 µA
3AP	Positive	Positive	5.215 µA	3.482 µA
4AN	Positive	Negative	5.215 µA	3.586 µA
2BP	Positive	Positive	5.215 µA	3.162 µA
3BP	Positive	Positive	5.215 µA	4.104 µA
6AN	Positive	Negative	4.174 µA	3.034 µA

6.4. Application Performance Evaluation

The developed software was installed and tested on a laptop with relatively low specifications to assess the performance of the desktop application, including 8 GB RAM and an Intel(R) Core(TM) i5-8250U CPU at 1.60GHz. Despite the modest hardware, the desktop application demonstrated efficient performance, characterized by rapid boot times and minimal RAM usage, which amounted to approximately 5% of the available memory. No signs of lagging or slow performance were observed throughout the testing process during sample analysis or data processing.

The codebase for the desktop application has been designed to be compatible with multiple platforms, allowing for seamless operation across various devices and operating systems. The application has been successfully tested on devices running Windows 10 and Windows 11, indicating its adaptability and versatility. Future testing could expand to other operating systems, such as macOS and Linux, further broadening the range of devices on which the application can operate.

Additionally, performance optimization could be explored to enhance the application's responsiveness and efficiency further, ensuring it can function effectively on various hardware configurations. It could involve code refactoring, reducing computational complexity, and employing caching techniques, among other strategies. By optimizing the application's performance, its usability and accessibility could be improved, enabling users with diverse hardware specifications to benefit from the rapid SARS-CoV-2 testing capabilities it provides. The performance evaluation of the desktop application for rapid SARS-CoV-2 testing indicates its ability to operate efficiently on devices with modest hardware configurations. The successful testing on multiple devices and operating systems highlights its adaptability and versatility. At the same time, opportunities for further optimization could expand its usability and accessibility to a broader range of users.

6.5. User Acceptance Test (UAT)

User Acceptance Testing (UAT) is an important stage in software development when the acceptance and happiness of users with the generated software system are evaluated. UAT is the final phase of testing before deployment to end users or production. The primary purpose of UAT is to determine if the built software system satisfies the anticipated goals and is accepted by the intended users. In addition, it assures that the software system is well-developed and production-ready.

In this project, UAT was conducted by requesting user participation through the desktop application. Participants were given a questionnaire after using the application to evaluate different areas, including the user interface, user experience, data visualization, and automated patient result creation function. The poll utilized a scoring system, with a score of one reflecting the lowest possible score and a score of five being the highest. Users scored the desktop application's functionality and features across 12 questions into four categories, as shown in Tables 4 to 7.

Table 4. User Interface Design Section Scores

ID	Statement	Average Score
UAT01	How appealing is the layout and design of the user interface	3.3
UAT02	How consistent is the visual design throughout the app	4
UAT03	How well do the color scheme and typography match the app's purpose	4
	Overall Score	**3.8**

Table 5. User Experience Section Scores

ID	Statement	Average Score
UAT04	How easy is it to navigate through the user interface	3.3
UAT05	How user-friendly is the user interface	3.3
UAT06	How well does the app design function and perform its intended tasks	4.7
	Overall Score	**3.8**

Table 6. Data Visualization and Automated Patient Results Generation Section Scores

ID	Statement	Average Score
UAT07	How effective is the data visualization	4
UAT08	How effective and useful is the automated patient results generation	4
UAT09	Do data visualization features provide all the necessary information to decide whether the patient is positive or negative	4.7
	Overall Score	**4.2**

Table 7. Overall User Experience of the Desktop Application Section Scores

ID	Statement	Average Score
UAT10	How simple is it to browse the user interface	3.3
UAT11	Please rate the functionality of the desktop application. Does the application function as it should be	4.3
UAT12	What is your Overall experience with desktop application	3.7
	Overall Score	**3.8**

UAT09 received a high score, indicating that the desktop application successfully delivers all the necessary information for the operator to determine whether a sample is positive or negative. Furthermore, the "Data Visualization and Automated Patient Results Generation" section scored an impressive 4.2, the highest among all test sections. This high rating reflects the application's ability to automatically provide the final results (positive or negative) to the operator, along with a comprehensive set of data visualization and analysis features that enable the operator to confirm the automatically generated results and gain a better understanding of the tested sample data. However, UAT01 and UAT04 revealed that the desktop application could benefit from further development and improvement in the user interface, explicitly concerning design and visual appeal. The application can provide an even more seamless and user-friendly experience by addressing these areas, ultimately enhancing its overall performance and usability.

6.6. Discussion

Our recently developed NACOTS desktop application has been successfully tested through a rigorous User Acceptance Testing (UAT) process, demonstrating its effectiveness and performance in real-world scenarios. The application's ability to meet users' needs and deliver accurate, reliable results is a testament to its potential as a valuable tool in SARS-CoV-2 detection.

Furthermore, using Node.js in developing the NACOTS desktop application has allowed for compatibility with multiple operating systems, including Windows, MacOS, and Linux. This versatility enables a broader range of users to access and benefit from the application, making it a more adaptable and inclusive solution for detecting the virus.

Our study has identified two primary limitations: the inconsistencies in nanogold electrode development and the use of a static threshold for detection. First, the fabrication process of the nanogold electrode is prone to variations, which can result in differences in its properties, such as the size, shape, and density of gold nanoparticles. These inconsistencies can negatively impact its performance and, subsequently the biosensor's response. The peak blank values, a critical parameter that indicates the background signal from the electrode surface, can be affected by factors such as the quality or concentration of the gold precursor solution, variations in deposition time or temperature, or differences in surface functionalization of the electrode.

Additionally, our research relies on a static threshold, which may not account for unique sample characteristics or inherent variations in electrode performance adequately. This limitation could potentially reduce the accuracy of detection and analysis, as the static threshold does not adapt to individual sample properties or electrode performance fluctuations.

Future research should focus on the following strategies To address the limitations of our study. First, the nanogold electrode fabrication process could be optimized. Efforts should be directed toward minimizing inconsistencies in the electrode development process. Researchers could investigate the effects of varying gold precursor solutions, deposition times, temperatures, and surface functionalization techniques to determine the optimal conditions for producing reliable nanogold electrodes. Secondly, adaptive thresholding techniques using machine learning could be explored. Instead of using a static threshold, researchers should consider implementing adaptive thresholding techniques based on machine learning algorithms. By incorporating various statistical features, machine learning models can better accommodate unique sample characteristics and account for variations in electrode performance. These adaptive techniques could help improve the accuracy and reliability of the electrochemical potentiostat-based desktop application.

By addressing these limitations, future research can contribute to developing a more precise and dependable system for detecting SARS-CoV-2, ultimately enhancing the effectiveness of diagnostic tools in managing and controlling the spread of the virus.

7. CONCLUSIONS AND FUTURE WORKS

In this research, we developed and evaluated a desktop application for quick SARS-CoV-2 testing, utilizing a high-performance electrochemical potentiostat. The application exhibited exceptional performance in the user interface, program features, data presentation, and automated patient outcome generation. Furthermore, it operated efficiently on devices with modest hardware configurations. User Acceptance Testing (UAT) identified areas for improvement, mainly in user interface design and aesthetic appeal. Future work should address these areas, incorporating machine learning techniques to adjust the threshold for increased accuracy dynamically, optimizing the application for a broader range of devices and operating systems, and incorporating advanced data analysis and visualization tools. Expanding the application's capabilities to handle other diagnostic tests would increase its flexibility for various medical purposes, providing valuable potential in public health, clinical settings, and research. In conclusion, the electrochemical potentiostat-based desktop application for rapid SARS-CoV-2 testing represents a significant advancement in creating a user-friendly, accurate, and efficient diagnostic tool for COVID-19. By addressing the identified areas for improvement and exploring additional enhancements, this application has the potential to become an invaluable asset in the ongoing battle against the COVID-19 pandemic and future public health challenges.

ACKNOWLEDGMENT

This work was accomplished through the collaborative efforts of the Electrochemical Material and Sensor (EMaS) Research Group at the Faculty of Applied Sciences, University Teknologi MARA, the VLSI-MEMS Research Group at IIUM, and the Institute for Medical Research (IMR). We gratefully acknowledge the funding and support the Ministry of Science, Technology, and Innovation (MOSTI) provided under grant number CV02211032 (SPG22-026-0026). Ethical clearance for the SARS-CoV-2 experiments was obtained from the IMR, ensuring the responsible conduct of research. We would also like to note that the NACOTS logo, color scheme, and software screenshots presented in this paper may be subject to copyright. Any use or reproduction of these materials should be done following the appropriate copyright and intellectual property regulations.

REFERENCES

[1] WHO. "World Health Organization, Coronavirus (COVID-19) Dashboard." https://covid19.who.int/ [accessed 27 March, 2023].

[2] C. Huang *et al.*, "Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China," *The lancet*, vol. 395, no. 10223, pp. 497-506, 2020.

[3] S. A. Abid *et al.*, "Biosensors as a future diagnostic approach for COVID-19," *Life sciences*, vol. 273, p. 119117, 2021.

[4] M. Wang *et al.*, "Clinical diagnosis of 8274 samples with 2019-novel coronavirus in Wuhan," *MedRxiv*, p. 2020.02. 12.20022327, 2020.

[5] J. P. Broughton *et al.*, "CRISPR–Cas12-based detection of SARS-CoV-2," *Nature biotechnology*, vol. 38, no. 7, pp. 870-874, 2020.

[6] G. S. Park *et al.*, "Development of Reverse Transcription Loop-Mediated Isothermal Amplification Assays Targeting Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2)," *J Mol Diagn*, vol. 22, no. 6, pp. 729-735, Jun 2020, doi: 10.1016/j.jmoldx.2020.03.006.

[7] IEC, "IEC 62304:2006+A1:2020. Medical device software - Software life cycle processes," International Electrotechnical Commission, 2020.

[8] ISO, "ISO 14971:2019. Medical devices - Application of risk management to medical devices," International Organization for Standardization, 2019.

[9] ISO, "ISO 13485:2016. Medical devices - Quality management systems - Requirements for regulatory purposes," International Organization for Standardization, 2016.

[10] USFDA, "General Principles of Software Validation; Final Guidance for Industry and FDA Staff," US Food and Drug Administration, 2002.
[11] A. Singh *et al.*, "Recent advances in electrochemical biosensors: Applications, challenges, and future scope," *Biosensors*, vol. 11, no. 9, p. 336, 2021.
[12] I.-H. Cho, D. H. Kim, and S. Park, "Electrochemical biosensors: Perspective on functional nanomaterials for on-site analysis," *Biomaterials research*, vol. 24, no. 1, pp. 1-12, 2020.
[13] J. Wang, "Electrochemical biosensors: towards point-of-care cancer diagnostics," *Biosensors and Bioelectronics*, vol. 21, no. 10, pp. 1887-1892, 2006.
[14] A. Saleh Ahammad, J.-J. Lee, and M. A. Rahman, "Electrochemical sensors based on carbon nanotubes," *sensors*, vol. 9, no. 4, pp. 2289-2319, 2009.
[15] E. Yuhana Ariffin, L. Y. Heng, L. L. Tan, N. H. Abd Karim, and S. A. Hasbullah, "A highly sensitive impedimetric DNA biosensor based on hollow silica microspheres for label-free determination of E. coli," *Sensors*, vol. 20, no. 5, p. 1279, 2020.
[16] C. Kataoka-Hamai and Y. Miyahara, "Label-free detection of DNA by field-effect devices," *IEEE Sensors Journal*, vol. 11, no. 12, pp. 3153-3160, 2011.
[17] S. Wang, J. Zhang, O. Gharbi, V. Vivier, M. Gao, and M. E. Orazem, "Electrochemical impedance spectroscopy," *Nature Reviews Methods Primers*, vol. 1, no. 1, p. 41, 2021.
[18] M. Sumitha and T. Xavier, "Recent advances in electrochemical biosensors–A brief review," *Hybrid Advances*, p. 100023, 2023.
[19] A. J. Bard, L. R. Faulkner, and H. S. White, *Electrochemical methods: fundamentals and applications*. John Wiley & Sons, 2022.
[20] R. J. Narayan, *Medical Biosensors for Point of Care (POC) Applications*. Woodhead Publishing, 2016.
[21] PalmSens. "PalmSens SDKs for .NET." https://www.palmsens.com/palmsens-sdks-for-net/ [accessed 27 March, 2023].
[22] G. L. Scoccia and M. Autili, "Web Frameworks for Desktop Apps: an Exploratory Study," presented at the Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), 2020.
[23] G. L. Scoccia, P. Migliarini, and M. Autili, "Challenges in Developing Desktop Web Apps: a Study of Stack Overflow and GitHub," presented at the 2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR), 2021.
[24] P. B. Jensen and C. Zhao, "Cross-Platform Desktop Applications USING ELECTRON AND NW.JS " 2017.
[25] M. Jasim, *Building Cross-Platform Desktop Applications with Electron*. 2017.
[26] K. Kredpattanakul and Y. Limpiyakorn, "Transforming JavaScript-Based Web Application to Cross-Platform Desktop with Electron," *Information Science and Applications 2018*, pp. 571-579, 2019, doi: 10.1007/978-981-13-1056-0_56.
[27] A. Kumar B and S. Anitha, "Development of Cross-Platform Desktop Apps using Electron Framework," 2018.
[28] M. K. F. M. Atan, R. A. Rahim, A. N. Nordin, T. S. Gunawan, and Z. M. Zain, "Investigation of Fluid Flow System Performance for Biosensor Application," in *2022 IEEE 8th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, 2022: IEEE, pp. 159-162.
[29] M. A. Abdul Ghani *et al.*, "Portable Electrochemical Biosensors Based on Microcontrollers for Detection of Viruses: A Review," *Biosensors*, vol. 12, no. 8, p. 666, 2022.
[30] M. A. A. Ghani, A. N. Nordin, R. A. Rahim, T. S. Gunawan, and Z. M. Zain, "Evaluation of Portable Potentiostats for Electrochemical Measurements: Voltammetry and Impedance Spectroscopy," in *2022 IEEE 8th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, 2022: IEEE, pp. 132-137.
[31] A. C. M. Nor, Z. M. Zain, and M. S. A. Noorden, "Application and Modification of RT-LAMP for Rapid Detection of SARS-CoV-2 Viral Genome," *Malaysian Journal of Medicine and Health Sciences*, vol. 19, no. 2, pp. 286-292, 2023.
[32] M. Akshay Kumaar, D. Samiayya, P. Vincent, K. Srinivasan, C. Y. Chang, and H. Ganesh, "A Hybrid Framework for Intrusion Detection in Healthcare Systems Using Deep Learning," *Front Public Health*, vol. 9, p. 824898, 2021, doi: 10.3389/fpubh.2021.824898.
[33] T. S. Gunawan, M. K. Lim, M. Kartiwi, N. A. Malik, and N. Ismail, "Penetration testing using Kali linux: SQL injection, XSS, wordpres, and WPA2 attacks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 2, pp. 729-737, 2018.
[34] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "Machine learning and deep learning approaches for cybersecuriy: A review," *IEEE Access*, 2022.
[35] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837-99849, 2022.