Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)
Vol. 9, No. 4, December 2023, pp. 1233-1251
ISSN: 2338-3070, DOI: 10.26555/jiteki.v9i4.27758

Software Design of Image-based Autocorrection Essays using Deep Learning on the Website and Mobile Application

Novalanza Grecea Pasaribu, Menara Alif, Gelar Budiman, Fityanul Akhyar
Electical Engineering, Telkom University, Bandung, Indonesia

ARTICLE INFO

Article history:

Received December 04, 2023
Revised January 10, 2024
Published February 17, 2024

Keywords:

Automated Essay Scoring;
Deep Learning in Education;
Essay Grading Software

ABSTRACT

Traditional essay assessment methods are often time-consuming and prone to subjectivity. This study proposes a novel Automated Essay Scoring (AES) system, "Essay Mathematic Auto Correction (Emath Toco)," featuring web and mobile app interfaces. Emath Toco leverages visual stimuli and deep learning algorithms like 1D CNN, NasNet Mobile, and GoogleNet to offer objective and efficient essay evaluation. Extensive testing on a 40/60 training/testing data split yielded accurate data classification, validating successful implementation on Flutter-built Android applications and a Firebase-powered web interface. User experience surveys revealed positive feedback on Emath Toco's ease of use, visually appealing interfaces, and effective data collection, confirming its user-friendliness. Emath Toco's innovative use of visual stimuli and deep learning algorithms significantly reduces subjectivity and improves the accuracy of essay evaluation. Emath toco is promising technology with the potential to revolutionize essay assessment and educational methodologies. The research contributes to the field of automated essay scoring in two key ways. First, by integrating visual stimuli as a novel approach, Emath Toco expands the range of factors considered in scoring, potentially leading to more comprehensive and efficient. Second, the successful implementation of the system on both web and mobile platforms demonstrates its flexibility and accessibility, offering educators a versatile tool regardless of technological limitations.

Corresponding Author:

Novalanza Greecea Pasaribu, Electrical Engineering, Telkom University, Bandung, Indonesia
Email: novalanza15@gmail.com

1. INTRODUCTION

Assessment is an important mechanism for evaluating student performance in educational settings. In Indonesia, essay assessments have an important role in measuring progress, fostering critical thinking skills, and evaluating depth of knowledge. However, manual essay grading, especially with a large number of questions, takes a long time and is prone to bias. To overcome these challenges, this research proposes an automated essay assessment system (AES) that uses machine learning technology [1]–[3]. This system is designed to provide more efficient and objective assessments for educators. In the context of this research, we have explored relevant studies on automated assessment methodologies, specifically referring to [4]–[11] as supplementary references. The primary references will be discussed in more detail in the following paragraphs.

The existing literature exhibits a diverse range of applications for machine learning techniques, particularly focusing on Convolutional Neural Networks (CNNs) [12]–[14] in various domains. In one instance [15], CNNs are employed with Machine Learning methods, achieving a commendable accuracy of 93.33% in visual recognition tasks, showcasing their effectiveness in handling image data. The study further extends its application to Android development, utilizing MobileNet for efficient vision applications on mobile devices, ultimately improving speed and model size while maintaining high accuracy.

Expanding on the versatility of CNNs, another study [16] introduces a driver drowsiness detection system that attains an impressive average accuracy of 83.33%. Operating on facial landmarks from mobile device

images, the compact CNN model effectively identifies drowsy driving behavior. The associated Android application, employing Dlib and Java Native Interface (JNI), provides real-time evaluations and timely alerts, demonstrating the efficacy of CNNs in addressing safety concerns.

In the context of education, a proposal for automated exam paper evaluation is presented [17]. Leveraging OCR and key term templates, the system scans handwritten papers, extracts essential information, and generates admin reports. The study underscores the advantages of automated evaluation in the context of transitioning from traditional exams to online assessments, offering insights into performance metrics and the evolving landscape of assessment methodologies.

Shifting focus to character recognition [18], an innovative method for recognizing online handwritten Chinese characters is introduced. The 1-dimensional CNN structure outperforms traditional 2-dimensional CNNs and recurrent neural networks, emphasizing its benefits in terms of recognition accuracy, model storage, and computational speed. The trajectory preprocessing employed eliminates the need for data augmentation and feature image extraction, opening up avenues for compact CNN architectures in online character recognition.

Exploring the healthcare domain [19], a research paper emphasizes the application of machine learning in predicting strokes. The study delves into the creation of an automated algorithm, with Random Forest identified as the top-performing model. The integration of the model with web technology further results in a user-friendly web application dedicated to stroke prediction, highlighting the importance of explainable AI and model interpretability.

Lastly [20], a case study on the Full-Teaching educational web platform evaluates the FlakyLoc technique's efficacy in identifying the root causes of flakiness in End-to-End tests. The technique successfully locates the triggers of occasional failures, providing valuable insights for fixing unstable test cases.

However, these existing approaches often possess limitations. Some focus solely on specific aspects like expression or structure, neglecting the holistic analysis of an essay. Others lack contextual understanding or rely on limited features, hindering comprehensive evaluation. Bridging these gaps, our research proposes a novel AES system, Emath Toco, equipped to provide a more nuanced and objective assessment of mathematical essays. Unlike previous approaches, Emath Toco employs CNNs to extensively analyze essay content, encompassing not only structure and language but also mathematical accuracy and reasoning. This integration of advanced deep learning with a user-friendly web and mobile interface presents a significant advancement in the field of AES. Moreover, Emath Toco's focus on mathematical essays addresses a specific and demanding domain, offering educators a valuable tool for efficient and accurate assessment. By delving into the strengths and limitations of existing research, we have identified the need for a comprehensive AES system capable of handling the complexities of mathematical essays. Emath Toco, with its innovative deep learning approach and user-friendly interface, aims to fill this gap and revolutionize essay assessment in the field of mathematics.

This research makes two main contributions. First, the proposed AES system uses CNNs to deeply analyze essay content, including structure, language, and emotion. Second, this system is implemented on web and mobile platforms, so it can be used by educators and students from various backgrounds. In conclusion, this research has developed an AES system that has the potential to increase the efficiency and objectivity of essay assessment. This system can be used by educators to save time and provide more accurate and fair assessments for students.

2. METHODS

The methods used in deep learning are 1D Convolutional Neural Network (CNN), NasNet Mobile, and GoogleNet models, demonstrating effective data classification and accurate predictive evaluations on 40% training and 60% testing data. However, this journal will only discuss the implementation of deep learning in mobile application and website. The concept the system shows in Fig. 1.

The concept of this system is that the application will be used by students to upload their answers to the lecturer for assessment. Student answers will be entered into the cloud database, namely Firebase, which will be displayed on the website in table form. Some student answers are assessed manually to be used as training data that will be processed at the training stage. This system is trained using the CNN (Convolutional Neural Network) algorithm such as 1D Convolutional Neural Network (CNN), NasNet Mobile, and GoogleNet models. This training stage is carried out by classifying images of student answers that have not been corrected with several answers that have been manually corrected by the lecturer to produce a predicted value output. This training stage is carried out by classifying images of student answers that have not been corrected with several answers that have been manually corrected by the lecturer to produce predicted value output. The

comparison between answers that have been manually corrected and those that have not been corrected is 40% and 60%. This comparison was obtained after conducting experiments with various numbers of comparisons, and it was found that 40% and 60% were the most effective and accurate. The value of the prediction will be stored in Firebase and will be displayed by the application so that students can find out the value with a rating range from 1-100, besides that it will also appear on the website. This process requires an internet connection at every stage, whether on the website, application or deep learning.

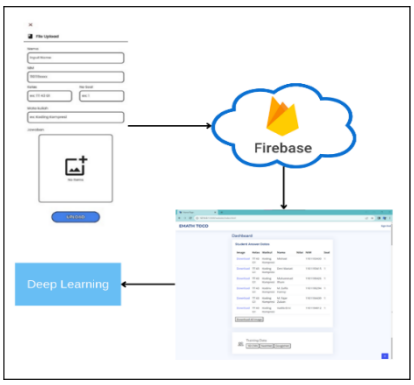

Fig. 1. The concept the system

The 1D CNN (Convolutional Neural Network) [21] method is commonly employed for processing sequential signals such as text or time series data. Utilizing 1D convolution layers, this model proves effective in feature extraction from sequential data by employing 1D filters. NASNet (Neural Architecture Search Network) [22], [23] represents an approach that achieves an optimal network architecture through an automated search process. By employing search algorithms like genetic algorithms or Bayesian search, NASNet automatically discovers network structures that are most suitable for specific tasks. Meanwhile, GoogleNet (Inception) [24] stands out for its use of the Inception module, allowing the incorporation of various filter sizes within a single convolution layer. In this manner, GoogleNet can capture features at different scales and levels of complexity, enhancing the performance of image classification. This groundbreaking approach by GoogLeNet marks the pinnacle of the deep learning research surge [25]. Deep learning block diagram shown inn Fig. 2.

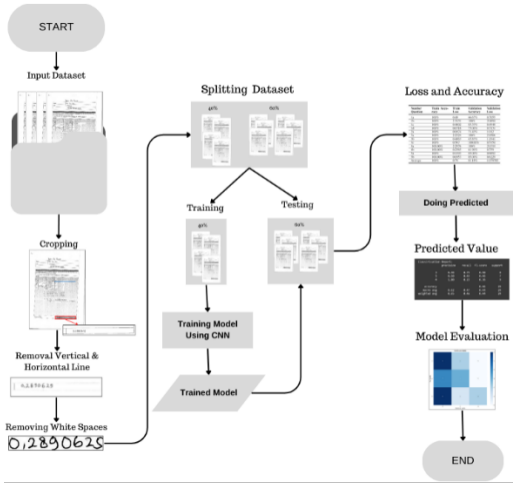

Fig. 2. Deep Learning Block Diagram

The process stages carried out to build a model that will be used to classify data based on scores start from data acquisition in the form of images from student answer sheets in the Coding and Compression course. After the required data is collected, pre-processing of the data is carried out in the form of cutting the image in the desired area (cropping), deleting vertical and horizontal lines, until you get the desired image. The next stage is to carry out data training with the CNN architecture.

2.1. Basic Theory

In its implementation, the website uses JavaScript and Hyper-Text Markup Language (HTML) in its creation and Cascading Style Sheets (CSS) to create visuals for the website in a web browser. Then for the application using the dart language using the flutter sdk. To store data that will be used by websites and applications, the Firebase service is used.

2.1.1. Flutter

Flutter is a notable open-source framework and software development kit (SDK) developed by Google, designed for creating efficient app interfaces that can be deployed on both Android and iOS platforms using a single codebase. The hot reload feature in Flutter eliminates the need to compile or rebuild the entire code with each modification, providing instant visual feedback. Dart, the programming language used in Flutter, brings a sense of familiarity for individuals already acquainted with programming languages like Java or JavaScript. [26].

2.1.2. Dart

Developing Flutter applications involves using the Dart programming language, which may present a learning curve. However, if you are familiar with languages such as JavaScript, Java, or other C-like languages, you'll notice similarities in Dart's syntax. Moreover, those experienced in object-oriented design from languages like Ruby or Python will find Dart comfortable to work with.

2.1.3. Android

Android originated as a Linux-based operating system that was initially created by Android Inc., receiving substantial backing from Google Finance. Google later acquired Android Inc. in 2005. The official launch of Android took place on November 5, 2007, coinciding with the establishment of the Open Handset Alliance. This alliance, focused on advancing mobile device standards, marked a significant development in the Android ecosystem. Subsequently, Google released the Android code under the Apache license, which is a software license and open standard specifically designed for mobile devices. In terms of official Android system distributors, there are two categories: the first enjoys comprehensive support from Google, inclusive of Google Mail services. On the other hand, the second category, known as Open Handset Distribution (OHD), operates independently of Google support and is considered entirely free.

2.1.4. Firebase

Firebase, a service by Google, improves the development of application through its Realtime Database, guaranteeing secure access to the database directly from the client side. This two-way data exchange is particularly efficient for applications in energy distribution. The use of Firebase as a secure data storage server, coupled with Cloud Functions for instantaneous notifications, enhances both data integrity and the overall security of the application [26]–[28].

2.1.5. Web Server

The primary tool for presenting web pages (homepage) is the web browser. This page is fundamentally constructed utilizing Hyper-Text Markup Language (HTML) and Cascading Style Sheets (CSS). Illustrative instances of web browsers include Internet Explorer (for Windows), Safari (for Mac), and Firefox (for Linux Ubuntu). Web browsers currently are not responsive to the user's level of interest when reading a webpage. The webpage is a one-way system where the reader is the passive consumer of the content provided by the web developer [29].

2.1.6. Hyper-Text Markup Language (HTML)

HTML, also known as Hyper-Text Markup Language, serves as the primary coding language for website creation. Through HTML, developers can guarantee the seamless integration of text, images, and multimedia with other elements when the website is accessed through a browser.

2.1.7. Javascript

JavaScript operates as a web programming language on the Client/Browser side, primarily utilized for manipulating HTML elements and automating the addition of styles. It simplifies the enhancement of HTML document interactivity. Being a functional language, JavaScript supports higher-order functions [30].

2.1.8. Cascading Style Sheets (CSS)

CSS, which stands for Cascading Style Sheet, is valuable for enhancing the visual appeal of HTML or specifying the presentation of HTML elements. This involves tasks like defining positioning, modifying text or background colors, and other related aspects [31].

2.2. Data Structure

In the context of mobile applications and websites, the utilization of flowcharts within their data structure is strategically employed to enhance user understanding and interaction. Flowcharts serve as visual representations, offering a succinct and intuitive depiction of the process flow. This choice is motivated by the benefits it extends to users, as flowcharts provide a clear and structured overview of the system's functionality. By visually mapping out the sequential steps and decision points, users can easily comprehend the navigation and functionality of the mobile application and websites. This not only facilitates a user-friendly experience but also aids users in navigating complex processes, thereby contributing to improved usability and overall user satisfaction.

2.2.1. Mobile Application

This application is used for students to provide answers to their essays, so that the data can be sent to Firebase and accessed by the website. Several previous studies were used as references [32]–[43] in creating the Emath Toco (Essay Mathematic Auto Correction) application. Fig. 1 shows a mobile application focused on collecting student answers, with additional features including account registration, login, data editing, and deletion. The workflow begins with users launching the "Essay Mathematic Auto Correction (Emath Toco)" app. From the Landing Page, they can proceed to the Login page for registration or login. Registration requires providing email and creating a password, leading to the home page. Alternatively, users can log in with their registered email and password, redirecting them to the home page. From there, users can access the Class Detail page and then proceed to the Upload Task Amount page, where they can upload or view previously uploaded files.

Fig. 4 depicts the flowchart for the mobile app's login and registration features. It starts from the "Pre-defined Login" point in Fig. 3, guiding users to the Login page. Users on the Login page can either register or log in. Registering involves entering an email and password, redirecting users to the home page. Logging in requires entering the registered email and password, then also leading to the home page.

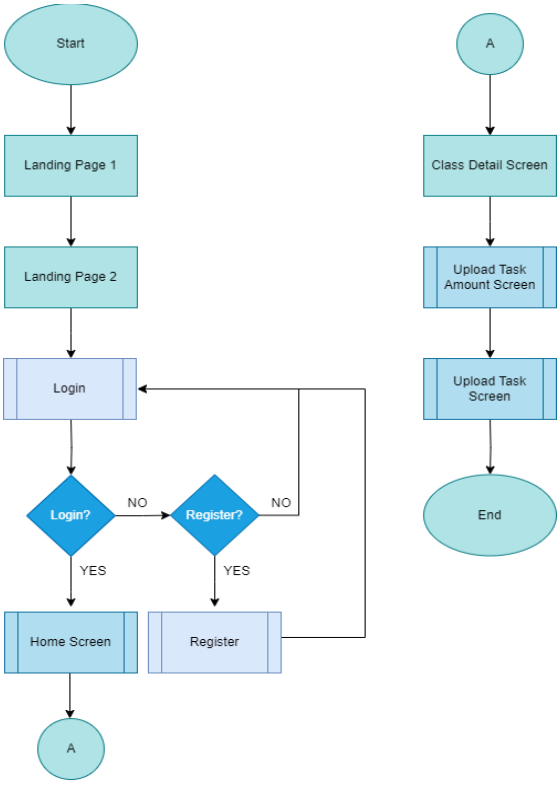

Fig. 3. Mobile Application Block Diagram

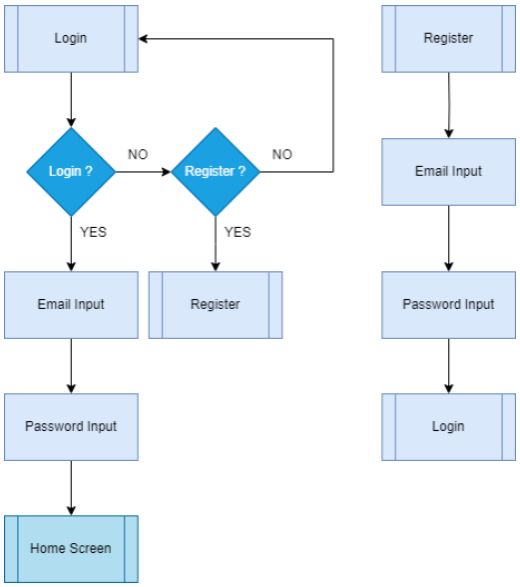

Fig. 4. Login and Register Flowchart

Fig. 5 portrays the flowchart of the mobile application's file uploading feature. "Pre-defined Upload Task" serves as the continuation point from Fig. 3, guiding users to the Upload Task page. The following steps outline the workflow of the file uploading feature: Users will be directed to the Upload Task page, where they are required to provide their Name, Student ID (NIM), Class, Question Number, and Course information. Additionally, users are prompted to select an image from their smartphone's gallery for uploading into Firebase.

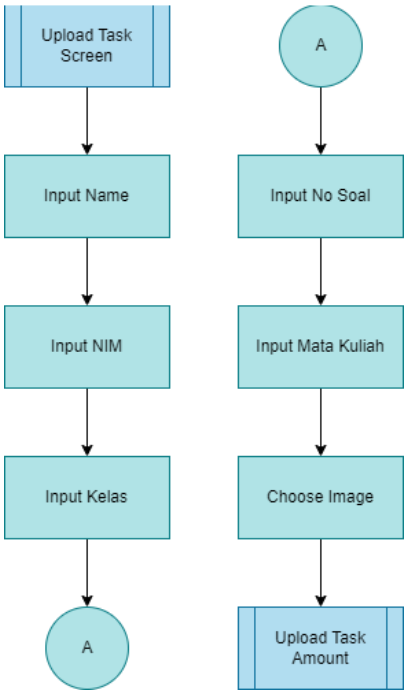

Fig. 5. File Uploader Flowchart

The following steps outline the workflow of the feature for viewing, modifying, and deleting files: When on the Upload Task Amount page, users have the option to either upload files or view previously uploaded files within the application. If the user chooses to view files, they will be directed to the file page, where they can access the uploaded data and photos. Furthermore, users have the ability to modify or delete this data by tapping the "edit" and "delete" icons located at the upper right corner of the page. Clicking the "edit" icon

allows users to make modifications to the uploaded data, while clicking the "delete" icon results in the automatic deletion of the uploaded data. Edit and delete flowchart shown in Fig. 6.

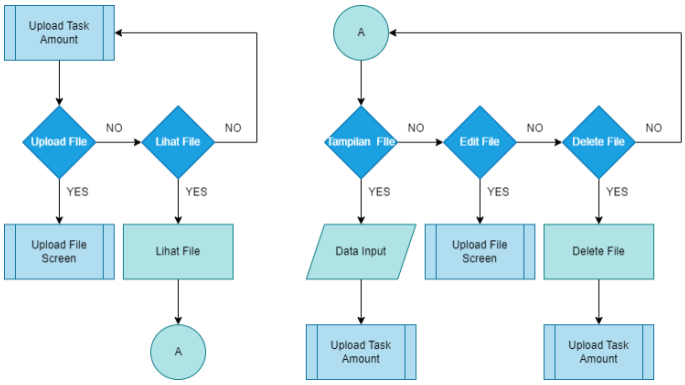

Fig. 6. Edit and Delete Flowchart

The Emath Toco application has user interferences for each feature such as landing page, login, register, home page, drawer, profile, class detail, Upload Task Amount and Upload File. Fig. 7 represents the initial screen of this application, which is the first thing users will see. This screen serves the purpose of providing information about the application, aiming to create a positive impression on the users.

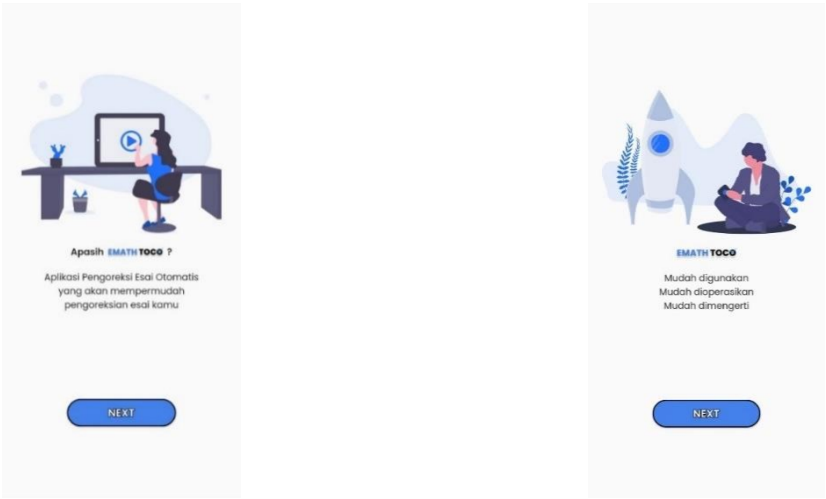

Fig. 7. Landing Page

Fig. 8 portrays the login page that users can access after pressing the "NEXT" button on the next_landing page. On this page, users can log into the application by entering the email and password they registered on the registration page. Fig. 9 depicts the registration page, accessible to users after clicking the "register" text button on the login page. On this page, users are required to register their email and password in order to gain access to the application.

Fig. 10 depicts the home page interface that users can access upon successfully logging in via the login page or completing registration through the register page. Fig. 11 displays the Profile page, accessible to users upon pressing the icon located in the upper-right corner of the home page.

Fig. 12 displays the Drawer page, which users can access by tapping the drawer button located at the upper left corner of the home page. Meanwhile, Fig. 13 displays the Class Detail page, which users can access by tapping the icon button located in the upper right corner of the home page.

Fig. 14 showcases the Upload Task Amount page, accessible to users upon pressing the "Upload Assignment" button. Meanwhile, Fig. 15 depicts the Upload File page, accessible to users by clicking the uploaded file button on the Upload Task Amount page.

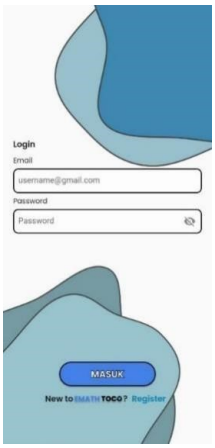

Fig. 8. Login

Fig. 9. Register

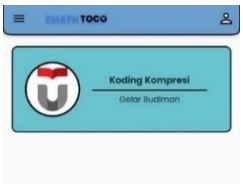

Fig. 10. Home

Fig. 11. Profile

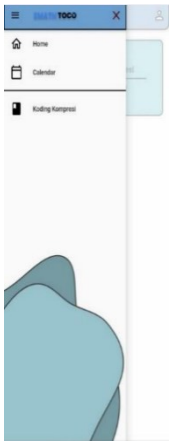

Fig. 12. Home

Fig. 13. Profile

Fig. 14. Upload Task Amount

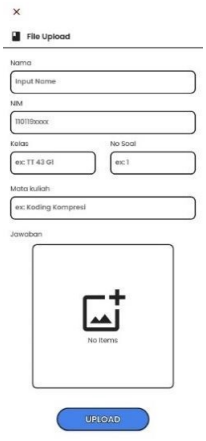

Fig. 15. Upload File

2.2.2. Website

This website is used to monitor data entering Firebase both before and after training and is also a bridge for conducting model training so that user can get training results. Several previous studies were used as references [20], [44]–[50] in creating the Emath Toco website (Essay Mathematic Auto Correction). The Emath Toco website has a usage flowchart shown in Fig. 16 and features such as Sign In/Sign Up and Training Data. To use this website, users begin by accessing the Emath Toco. Upon arrival, they are presented with options to either Sign Up or Sign In. If they are new users, they should first complete the Sign-Up process. For those who have already registered, they can directly Sign In. Once successfully logged in, users are directed to the main page, where they can view a table containing student data entries submitted through the application. Additionally, users have the capability to download the entire image of the student's answers. Should users desire to make use of the automatic correction feature, they can select the Training Data option. Upon completing the data training, the value column within the main page table is populated. If users decide not to select any feature, they will remain on the main page. When their session is complete, users can select the Sign Out feature, which will then redirect them to the Sign In and Sign Up pages for future access.

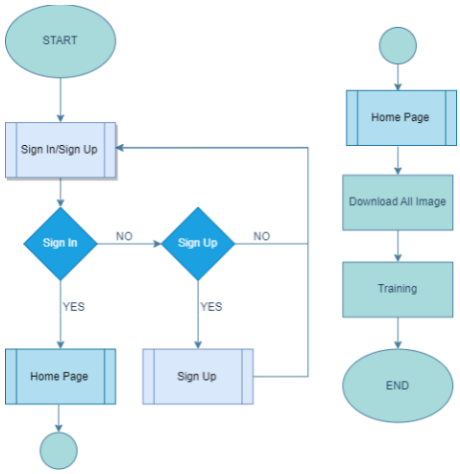

Fig. 16. Website Block Diagram

Fig. 17 is a flow diagram of the Sign In and Sign Up features. The Sign In feature functions to register an account for new users. Where new users will be directed to register their desired name, telephone number, email and password. Meanwhile, the Sign In feature functions to enter the website by entering the registered email and password.

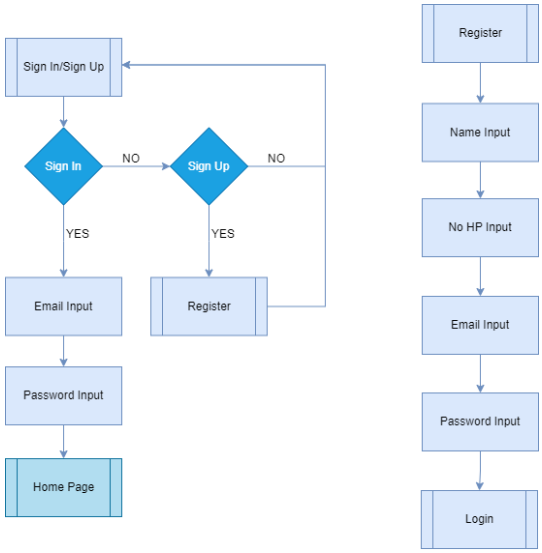

Fig. 17. Flowchart Sign in and Sign up feature

Fig. 18 illustrates the website flow diagram for the autocorrect feature, focusing on the crucial Training Data workflow. This pivotal feature plays a significant role in refining the autocorrection results by leveraging manually corrected student responses and uncorrected data. Users initiate the process by selecting a preferred method, such as 1D CNN, NasNet, or GoogleNet. Upon selection, they are seamlessly guided to the Training Data page, where the training phase unfolds. This phase involves the extraction of data from Firebase, encompassing both manually corrected and uncorrected responses, which is then meticulously transformed into CSV format. The subsequent training process generates output values tailored to each student. These valuable output values are systematically stored in Firebase, ensuring accessibility within the application for individual students. Moreover, they are presented in a tabular format on the main page of the website, underscoring the essential role of the Training Data feature in refining and enriching the autocorrection outcomes.

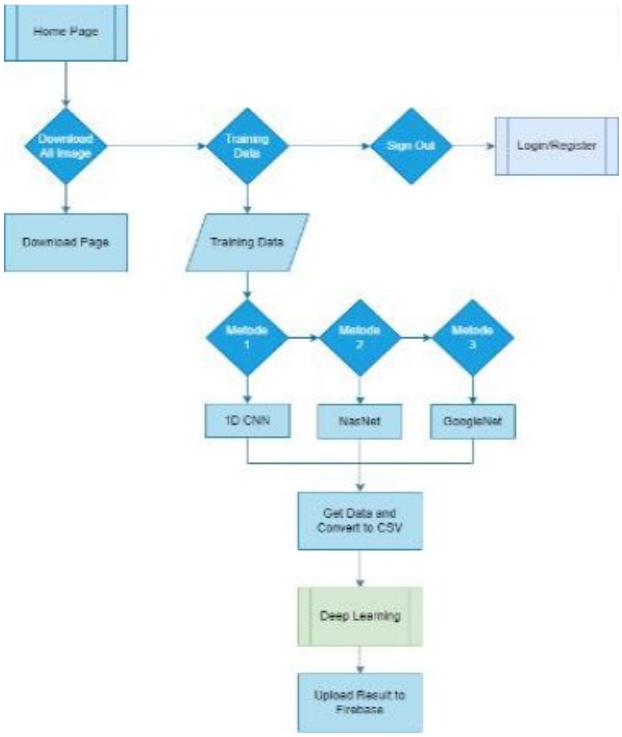

Fig. 18. Flowchart Training data feature

Emath Toco website has User Interferences on the Sign In/Sign Up page and Home Page. Fig. 19 and Fig. 20 is the login page that the user will see the first time when they open the website. The Sign Up page functions to create an account, while the Sign In page functions to enter the main page by entering the registered email and password.

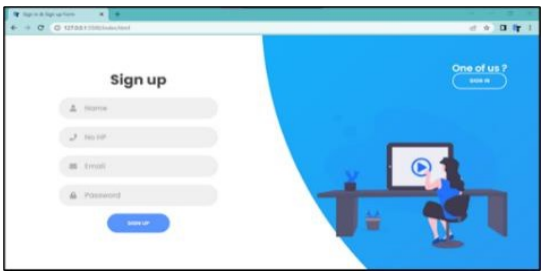

Fig. 19. Sign up

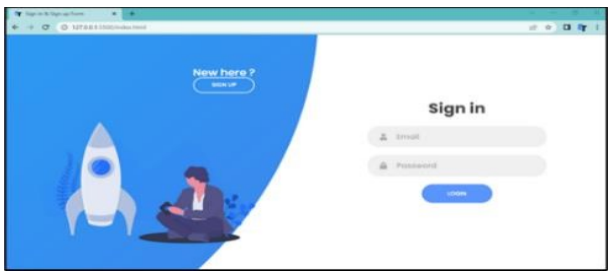

Fig. 20. Sign in

Fig. 21 is the display of the main page on this website which displays student answer data in tabular form and also presents training features. On this page, users can see all the data entered by students into the application, and can conduct data training with the training feature.

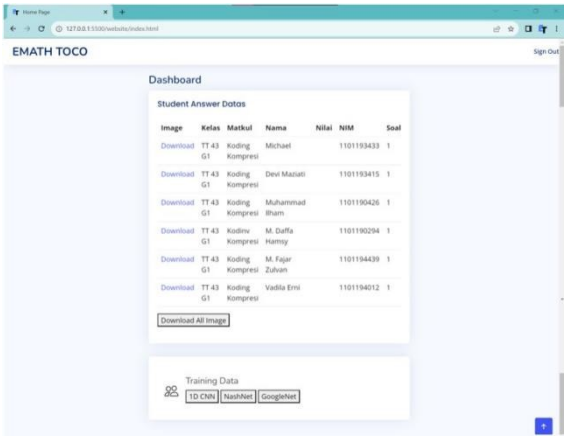

Fig. 21. Home page

3. RESULTS AND DISCUSSION

The results of the journal are in the form of simulations and testing of various aspects of mobile application and websites.

3.1. Mobile Application Implementation

To begin using the Essay Mathematic Auto Correction (Emath Toco) application, the user must first launch it. Upon entering the landing page, they will find the option to proceed to the login page by selecting "NEXT." At the Login page, the user faces a choice between either registering for a new account or logging in with existing credentials.

If the user wants to register, they will be prompted to provide their email and password for the registration process. Subsequently, they will be redirected to the home page. On the other hand, if the user chooses to log in, they must enter the email and password used during registration, after which they will be redirected to the home page.

Once on the home page, the user gains access to the class detail page and can proceed to the upload task Amount page. On the upload task Amount page, the user is granted the choice to either upload new files or view files that they have previously uploaded.

3.2. Mobile Application Results

After the user uploads all the data entered in the application, that data will be processed by machine learning and produce results in the form of values, which will then be uploaded back into Firebase. Subsequently, the application will automatically read and display those values, as shown in the Fig. 22.

Fig. 22. Mobile Application Results

Software testing can be carried out from the functional specification aspect without involving testing the design and program code. The goal is to ensure that the functions, inputs, and outputs of the software conform to the required specifications. One of the methods used is Blackbox Testing, which is an easy method to use because it only requires lower and upper limits of the expected data. An estimate of the amount of test data can be calculated based on the number of entry data fields to be tested, the entry rules that must be met, as well as the upper and lower limit cases that are met. With this method, we can find out whether the software functionality can still receive unexpected data input, which can cause the stored data to become less valid.

1. Blackbox Testing

After the subsystem has been implemented, a functional testing is conducted to verify whether the implemented program operates in accordance with the planned specifications. The method employed for this functional testing is Blackbox Testing. The subsequent Table 1 presents the outcomes of the functional testing performed on the tested application. Based on the results of black box testing in Table 1, it shows good performance. The application runs smoothly and responsively when users use it to access the features provided.

Table 1. Mobile Application Blackbox Results

Interface	Test Scenario	Expected Results	Test Results
Login Page	E-mail and Password is not filled then pressed the enter button.	Can not log in and gives the message “Email is Still Empty” and “Email and Password are Wrong”	Succeed
	Fill in the appropriate email and password during registration.	The system receives login access and will display the home page	Succeed
Registration Page	E-mail and Password is not filled then press the enter button	Can not log in and gives the message “Email is still empty” and “Well, enter the correct email please”	Succeed
Profile Page	Return to Login Page	Display the Login and Logout pages of existing accounts	Succeed
Upload Task Amount Page	Add data	Display the Edit Delete page	Succeed
	Look back at files that have been uploaded	Display the File Uploads page	Succeed

2. Survey Results

In this phase, User Experience (UX) testing is conducted on the Android application to uncover user experiences and impressions while using the designed application. The employed testing method involves distributing a comprehensive survey containing various questions related to the developed application. The survey's outcomes are obtained from a total of 30 respondents who provided their feedback on the application.

We analyzed the survey result using a rating scale where a score of 1 describes "strongly disagree," a score of 2 is "disagree," a score of 3 is "neutral," a score of 4 is "agree," and a score of 5 is "strongly agree." Therefore, the ratings provided by the respondents can help evaluate the extent to which users experience satisfaction and effectiveness with the designed application. This data serve as a crucial basis for making necessary improvements and enhancements to the application to ensure a better user experience in the future.

Specifically, Fig. 23a assesses whether the application is easy to use, Fig. 23b evaluates whether the application can effectively collect data and photos from user responses, Fig. 23c gauges the smoothness of the data and photo input process, and Fig. 23d examines whether the application's interface is visually appealing. This data serves as a crucial foundation for implementing necessary improvements and enhancements to the application to ensure a better user experience in the future.

Question 1: Is the application easy to use?

The pie chart shows that 69.8% of respondents strongly agree or agree that the application is easy to use. This is a positive finding, suggesting that most users were able to navigate the app and complete tasks without difficulty. However, 24.8% of respondents were neutral or disagreed, indicating that there is room for improvement in terms of user-friendliness. It would be helpful to analyze the open-ended feedback from these respondents to identify specific areas where the app can be made easier to use.

Question 2: Can the application help collect data and photos of answers that have been submitted?

The results for this question are less positive. Only 37.8% of respondents strongly agree or agree that the app can effectively collect data and photos. A significant 48.4% were neutral or disagreed. This suggests that there may be issues with the app's data collection functionality, which could be a major barrier to its usability. Again, it is important to look at the open-ended feedback to understand the specific problems users encountered with data and photo collection.

Question 3: Does the data and photo input process go well?

The pie chart for this question is similar to the previous one, with 37.8% of respondents agreeing or strongly agreeing that the data and photo input process goes well. A larger group of 46.7% were neutral, and 15.5% disagreed or strongly disagreed. This suggests that while some users found the input process to be smooth, others experienced difficulties. Again, analyzing the open-ended feedback will provide valuable insights into the specific issues that need to be addressed.

Question 4: Is the application display attractive?

This question received the most positive response, with 59.1% of respondents agreeing or strongly agreeing that the app's display is attractive. However, 30.9% were neutral or disagreed, indicating that there is still room for improvement in terms of visual design. It would be helpful to understand what users found appealing about the app's display, as well as what aspects they found less appealing. This feedback can be used to inform future design decisions.

Conclusion:

The survey results provide a mixed picture of user experience with the math essay correction application. While some users found the app to be easy to use and visually appealing, others encountered difficulties with data collection, photo input, and overall smoothness. It is important to carefully analyze the open-ended feedback from respondents to identify specific areas for improvement. By addressing these pain points, the developers can create a more user-friendly and effective application for correcting math essays.

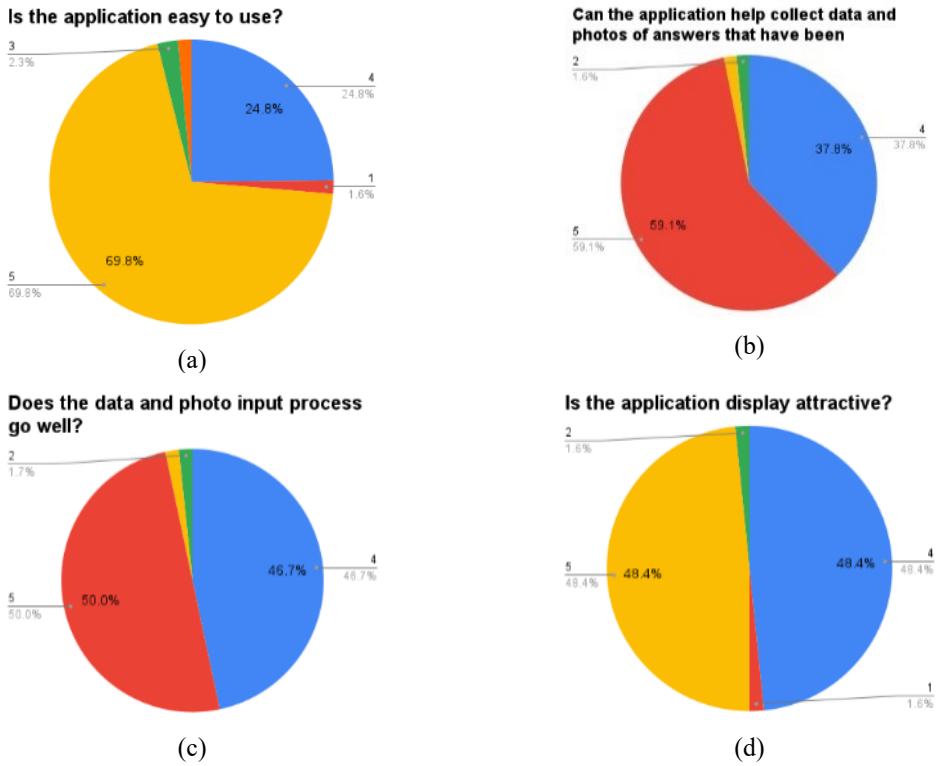

Fig. 23. Survey Question, (a) Question 1, (b) Question 2, (c) Question 3, (d) Question 4

3.3. Website Implementation

To start the Essay Mathematic Auto Correction (Emath Toco) website, the user must first launch it. The first page that appears is Sign Up and Sign In. If users choose to register, they will be asked to provide their name, cellphone number, email, and password for the registration process. Next, they will be redirected to the homepage. On the other hand, if users choose to log in, they must enter their email and password used during

registration, after which they will be redirected to the home page. Once on the home page, the user can see the answer data table which is equipped with a download all image button. In addition, there is a data training feature to make automatic corrections.

3.4. Website Results

After the user has trained with machine learning from the image of the answers, the output will be in the form of a value for each answer. The value will be uploaded into Firebase. Then the website will automatically read and display these values in the answer data table, as shown in the Fig. 24.

Fig. 24. Website Results

The website will be tested using Blackbox testing. The website that has been implemented will be tested using the Blackbox method by checking every button on the website. Testing on a website, if seen only from the functional aspect, can be done using Blackbox Testing, to test the function, input and output of the website is as expected. The testing method using the Blackbox Testing method is widely applied in software testing that is not tested in detail. The following table is the result of functional testing of the application being tested. Website blackbox results shown in Table 2.

Table 2. Website Blackbox Results

Interface	Test Scenario	Expected Results	Test Results
URL	Website access	Display the Sign in/ Sign up page	Succeed
Sign in Page	Email is filled and Password is not filled then press the enter button.	Can not log in and gives the message “Firebase: Error (auth/missing-password)”	Succeed
	Email is not filled and Password is filled then press the enter button	Cannot log in and gives message	Succeed
	Fill in an inappropriate email and password during registration	Cannot login and gives message “Firebase: Error (auth/user-not-found).”	Succeed
Sign up Page	Fill in the appropriate email and pass during registration	The system receives login access and will display the Home page	Succeed
	Name, no cellphone, fill in Password then press the enter button	The system receives login access and gives message “user has been successfully created”, after that clock okay and move to the sign in page	Succeed
	Return to Login Page	Display the Login and Logout pages of existing accounts	Succeed
Home Page	Displays the Training Data page	Display the selected method page	Succeed
	Display the Sign Out option	Display the Sign in/ Sign up page	Succeed

Based on the Blackbox test results from the implemented website, it shows good performance. The website runs smoothly and is responsive when used by users to access the features provided. From existing research, it will be compared with our system. The following table describes the contribution comparison with previous research. which compares research that only helps correct with deep learning/handwriting classification without any applications or websites, then accompanied by testing and surveys of users.

From Table 3 we can conclude that our system meets all the criteria by providing UI/UX for websites and applications, has ML functions and provides surveys and blackbox results. In [9], [11], [14] the authors purposed the implementation in web-based only. In [10], [9], [12] the the authors purposed the implementation in application-based only. In [14] the authors purposed the implementation without have ML function.

This research has all of these parameters, which provides ML functions and implements them on websites and applications along with the UI and also doesn't forget to evaluate the user experience in using them. However, this research has limitations in the number of sentences that can be corrected, as well as in the neatness of writing essay answers.

Table 3. Comparison with Previous Research

Related Work	UI/UX Web	UI/UX Apps	ML Function	User Survey	Blackbox Results	Description
[15]			✓			MobileNets for mobile and embedded vision applications
[16]		✓	✓			Driver Drowsiness Detection
[17]	✓	✓	✓			Digitized Exam Paper Evaluation
[18]		✓	✓			Recognizing online handwritten Chinese character
[19]	✓		✓			Automated Stroke Prediction
[20]	✓					Flakiness Lokalization for Reliable Test Suites
Our	✓	✓	✓	✓	✓	Autoscoring on Essay Mathematics

4. CONCLUSION

In summary, this paper introduces an innovative solution aimed at expediting and improving the essay assessment process through the integration of Automated Essay Scoring (AES) in both mobile application and web interfaces. The AES system, embedded within the "Essay Mathematic Auto Correction (Emath Toco)" Android application and a dedicated web platform, harnesses advanced deep learning technologies such as 1D Convolutional Neural Network (CNN), NasNet Mobile, and GoogleNet models to address subjectivity and biases inherent in traditional essay evaluation methods.

The efficiency of the system is further enhanced by utilizing Flutter, Dart programming language, Android, and Firebase. The study demonstrates successful data classification with a split of 40% training and 60% testing data, affirming the system's strength in automated essay grading. Positive outcomes from simulations, including blackbox testing, and user experience surveys underscore the system's effectiveness. Importantly, the successful implementation of the AES system yields accurate and timely essay assessments, seamlessly displayed on both the mobile application and website interfaces. This advancement in automated essay scoring not only saves educators valuable time but also reduces grading biases, providing a more objective and streamlined approach to essay assessment.

To enhance the robustness of future research endeavors, it is imperative to acknowledge the limitations of the current system and chart potential directions for improvements. The absence of a comprehensive discussion on system limitations and future directions limits the scope of this conclusion. Identifying constraints is crucial for understanding the boundaries of the implemented system and presents opportunities for future enhancements.

In terms of results, when compared to previous studies, as described in [16], the use of Convolutional Neural Networks (CNNs) and Machine Learning (ML) methods achieved a high accuracy of 93.33% in visual recognition tasks. This study also demonstrated a significant improvement in speed and model size, but it primarily focused on image recognition using MobileNet and TensorFlow in an Android application. Meanwhile, in [17], the proposed driver drowsiness detection system, using a CNN-based machine learning method, achieved an average accuracy of 83.33%. The developed model, not exceeding 75KB, effectively identifies drowsy driving behavior, providing real-time evaluations and timely alerts. In this context, the research contributes to the application of AES in assessing driving behavior. Then the results of this research obtained accuracy 100% Validation Accuracy 45.83% for 1D CNN, accuracy 100% validation Accuracy 50% for NasNet, and Accuracy 100% Validation Accuracy 100%% for GoogeNet. So it can be concluded that the

most accurate deep learning is using the GoogleNet model. These results were obtained with an image dataset consisting of mathematical sentences and expressions

User feedback from a survey involving 30 participants highlights the user-friendly nature of the application and website, emphasizing the positive impact on the overall essay assessment experience. in the application survey it was found that 69.8% of users strongly agreed that the application was easy to use, 59.1% of the application was very helpful in collecting answer data, 50% of the data was processed very well, and for the attractive application display 48.4% of users strongly agreed and 48.4% agreed. Meanwhile, for website testing, backbox testing is used and all the features on the Emath Toco website function well. In conclusion, the AES system is an important contribution to the field of Essay Assessment using Machine learning. By offering educators a comprehensive and effective tool, this implementation significantly improves the assessment process, marking major progress in the modernization and optimization of essay evaluation in the educational domain.

Acknowledgments

This paper is in part supported by Research and Community Service Directorate of Telkom University and the National Research and Innovation Agency (BRIN) under program Research and Innovation for Advanced Indonesia (RIIM) No. 30114916519, 2023-2024.

REFERENCES

[1] M. Uto and M. Okano, "Learning Automated Essay Scoring Models Using Item-Response-Theory-Based Scores to Decrease Effects of Rater Biases," *IEEE Trans. Learn. Technol.*, vol. 14, no. 6, pp. 763–776, 2021, https://doi.org/10.1109/TLT.2022.3145352.

[2] J. Xue, X. Tang, and L. Zheng, "A Hierarchical BERT-Based Transfer Learning Approach for Multi-Dimensional Essay Scoring," *IEEE Access*, vol. 9, pp. 125403–125415, 2021, https://doi.org/10.1109/ACCESS.2021.3110683.

[3] M. Uto, I. Aomi, E. Tsutsumi, and M. Ueno, "Integration of Prediction Scores from Various Automated Essay Scoring Models Using Item Response Theory," *IEEE Trans. Learn. Technol.*, vol. 16, no. 6, pp. 983–1000, 2023, https://doi.org/10.1109/TLT.2023.3253215.

[4] E. Shaikh, I. Mohiuddin, A. Manzoor, G. Latif, and N. Mohammad, "Automated grading for handwritten answer sheets using convolutional neural networks," *2nd Int. Conf. New Trends Comput. Sci. ICTCS - Proc.*, no. October, pp. 1–6, 2019, https://doi.org/10.1109/ICTCS.2019.8923092

[5] P. Goel and A. Ganatra, "Handwritten Gujarati Numerals Classification Based on Deep Convolution Neural Networks Using Transfer Learning Scenarios," *IEEE Access*, vol. 11, pp. 20202–20215, 2023, https://doi.org/10.1109/ACCESS.2023.3249787.

[6] C. Vinotheni and S. L. Pandian, "End-To-End Deep-Learning-Based Tamil Handwritten Document Recognition and Classification Model," *IEEE Access*, vol. 11, pp. 43195–43204, 2023, https://doi.org/10.1109/ACCESS.2023.3270895.

[7] Q. D. Nguyen, N. M. Phan, P. Kromer, and D. A. Le, "An Efficient Unsupervised Approach for OCR Error Correction of Vietnamese OCR Text," *IEEE Access*, vol. 11, pp. 58406–58421, 2023, https://doi.org/10.1109/ACCESS.2023.3283340.

[8] Y. Wang, Y. Yang, W. Ding, and S. Li, "A Residual-Attention Offline Handwritten Chinese Text Recognition Based on Fully Convolutional Neural Networks," *IEEE Access*, vol. 9, pp. 132301–132310, 2021, https://doi.org/10.1109/ACCESS.2021.3115606.

[9] S. Gonwirat and O. Surinta, "DeblurGAN-CNN: Effective Image Denoising and Recognition for Noisy Handwritten Characters," *IEEE Access*, vol. 10, pp. 90133–90148, 2022, https://doi.org/10.1109/ACCESS.2022.3201560.

[10] D. Zhelezniakov, V. Zaytsev, and O. Radyvonenko, "Online handwritten mathematical expression recognition and applications: A survey," *IEEE Access*, vol. 9, pp. 38352–38373, 2021, https://doi.org/10.1109/ACCESS.2021.3063413.

[11] A. Anand, S. De, A. Jaiswal, S. Agrawal, and B. C. Narendra, "Automation of Non-Classroom Courses using Machine Learning Techniques," *Proc. 6th Int. Conf. Educ. Technol. ICET*, pp. 90–96, 2020, https://doi.org/10.1109/ICET51153.2020.9276619.

[12] X. Liu, H. Wang, Y. Meng, and M. Fu, "Classification of Hyperspectral Image by CNN Based on Shadow Area Enhancement through Dynamic Stochastic Resonance," *IEEE Access*, vol. 7, pp. 134862–134870, 2019, https://doi.org/10.1109/ACCESS.2019.2941872.

[13] J. Wang, Y. Zheng, M. Wang, Q. Shen, and J. Huang, "Object-Scale Adaptive Convolutional Neural Networks for High-Spatial Resolution Remote Sensing Image Classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, pp. 283–299, 2021, https://doi.org/10.1109/JSTARS.2020.3041859.

[14] X. Luo, X. Qin, Z. Wu, F. Yang, M. Wang, and J. Shang, "Sediment Classification of Small-Size Seabed Acoustic Images Using Convolutional Neural Networks," *IEEE Access*, vol. 7, pp. 98331–98339, 2019, https://doi.org/10.1109/ACCESS.2019.2927366.

[15] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv*

preprint arXiv:1704.04861, 2017, [Online]. Available: http://arxiv.org/abs/1704.04861.
[16] R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen, and K. Barkaoui, "Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application," *IEEE Int. Conf. Informatics, IoT, Enabling Technol. ICIoT*, pp. 237–242, 2020, https://doi.org/10.1109/ICIoT48696.2020.9089484.
[17] R. Rajesh and R. Kanimozhi, "Digitized exam paper evaluation," *IEEE Int. Conf. Syst. Comput. Autom. Networking, ICSCAN*, pp. 1–5, 2019, https://doi.org/10.1109/ICSCAN.2019.8878791.
[18] J. Gan, W. Wang, and K. Lu, "A new perspective: Recognizing online handwritten Chinese characters via 1-dimensional CNN," *Inf. Sci. (Ny).*, vol. 478, pp. 375–390, 2019, https://doi.org/10.1016/j.ins.2018.11.035.
[19] K. Mridha, S. Ghimire, J. Shin, A. Aran, M. M. Uddin, and M. F. Mridha, "Automated Stroke Prediction Using Machine Learning: An Explainable and Exploratory Study With a Web Application for Early Intervention," *IEEE Access*, vol. 11, pp. 52288–52308, 2023, https://doi.org/10.1109/ACCESS.2023.3278273.
[20] J. Morán, C. Augusto, A. Bertolino, C. de la Riva, and J. Tuya, "FlakyLoc: Flakiness localization for reliable test suites in web applications," *J. Web Eng.*, vol. 19, no. 2, pp. 267–296, 2020, https://doi.org/10.13052/jwe1540-9589.1927.
[21] U. Shukla, G. J. Saxena, M. Kumar, A. S. Bafila, A. Pundir, and S. Singh, "An improved decision support system for identification of abnormal EEG signals using a 1D convolutional neural network and Savitzky-Golay filtering," *IEEE Access*, vol. 9, pp. 163492–163503, 2021, https://doi.org/10.1109/ACCESS.2021.3133326.
[22] A. M. Qadri, A. Raza, F. Eid, and L. Abualigah, "A novel transfer learning-based model for diagnosing malaria from parasitized and uninfected red blood cell images," *Decis. Anal. J.*, vol. 9, p. 100352, 2023, https://doi.org/10.1016/j.dajour.2023.100352.
[23] A. Hosseini *et al.*, "A mobile application based on efficient lightweight CNN model for classification of B-ALL cancer from non-cancerous cells: A design and implementation study," *Informatics Med. Unlocked*, vol. 39, p. 101244, 2023, https://doi.org/10.1016/j.imu.2023.101244.
[24] K. T. Ahmed, S. Jaffar, M. G. Hussain, S. Fareed, A. Mehmood, and G. S. Choi, "Maximum Response Deep Learning Using Markov, Retinal Primitive Patch Binding with GoogLeNet VGG-19 for Large Image Retrieval," *IEEE Access*, vol. 9, pp. 41934–41957, 2021, https://doi.org/10.1109/ACCESS.2021.3063545.
[25] F. Yuesheng *et al.*, "Circular Fruit and Vegetable Classification Based on Optimized GoogLeNet," *IEEE Access*, vol. 9, pp. 113599–113611, 2021, https://doi.org/10.1109/ACCESS.2021.3105112.
[26] Sunardi, A. Yudhana, and Furizal, "Tsukamoto Fuzzy Inference System on Internet of Things-Based for Room Temperature and Humidity Control," *IEEE Access*, vol. 11, 2022, pp. 6209–6227, 2023, https://doi.org/10.1109/ACCESS.2023.3236183.
[27] M. T. Ahammed *et al.*, "Real-Time Non-Intrusive Electrical Load Classification over IoT Using Machine Learning," *IEEE Access*, vol. 9, pp. 115053–115067, 2021, https://doi.org/10.1109/ACCESS.2021.3104263.
[28] S. R. Pandey, D. Hicks, A. Goyal, D. Gaurav, and S. M. Tiwari, "Mobile notification system for blood pressure and heartbeat anomaly detection," *J. Web Eng.*, vol. 19, no. 5–6, pp. 747–773, 2020, https://doi.org/10.13052/jwe1540-9589.19568.
[29] A. Kumar, J. Bose, and D. Bansal, "A web browser responsive to the user interest level," *12th IEEE Int. Conf. Electron. Energy, Environ. Commun. Comput. Control (E3-C3), INDICON*, pp. 1–6, 2016, https://doi.org/10.1109/INDICON.2015.7443485.
[30] S. Tilkov and S. Vinoski, "Node.js: Using JavaScript to build high-performance network programs," *IEEE Internet Comput.*, vol. 14, no. 6, pp. 80–83, 2010, https://doi.org/10.1109/MIC.2010.145.
[31] S. Song, J. Hur, S. Kim, P. Rogers, and B. Lee, "R2Z2: Detecting Rendering Regressions in Web Browsers through Differential Fuzz Testing," *Proc. - Int. Conf. Softw. Eng.*, pp. 1818–1829, 2022, https://doi.org/10.1145/3510003.3510044.
[32] Q. M. Ilyas, M. Ahmad, N. Zaman, M. A. Alshamari, and I. Ahmed, "Localized Text-Free User Interfaces," *IEEE Access*, vol. 10, pp. 2357–2371, 2022, https://doi.org/10.1109/ACCESS.2021.3139525.
[33] K. Moulaei, K. Bahaadinbeigy, E. Mashoof, and F. Dinari, "Design and development of a mobile-based self-care application for patients with depression and anxiety disorders," *BMC Med. Inform. Decis. Mak.*, vol. 23, no. 1, pp. 1–18, 2023, https://doi.org/10.1186/s12911-023-02308-y.
[34] N. Li *et al.*, "Standardization Workflow Technology of Software Testing Processes and its Application to SRGM on RSA Timing Attack Tasks," *IEEE Access*, vol. 10, pp. 82540–82559, 2022, https://doi.org/10.1109/ACCESS.2022.3196934.
[35] S. Gao, Y. Wang, and H. Liu, "UiAnalyzer: Evaluating whether the UI of apps is at the risk of violating the design conventions in terms of function layout," *Expert Syst. Appl.*, vol. 239, p. 122408, 2024, https://doi.org/10.1016/j.eswa.2023.122408.
[36] D. D. Henry *et al.*, "tolerances Ac c te d us cr ip t Ac c ep te d us cr t," *Cad. Saude Publica*, vol. 12, no. 1, pp. 1– 30, 2020, https://doi.org/10.1016/j.ijpara.2016.09.005.
[37] L. Shahmoradi, A. Borhani, M. Langarizadeh, G. Pourmand, Z. A. fard, and S. Rezayi, "Predicting the survival of kidney transplantation: design and evaluation of a smartphone-based application," *BMC Nephrol.*, vol. 23, no. 1, pp. 1–7, 2022, https://doi.org/10.1186/s12882-022-02841-4.
[38] Z. Zare, E. Hajizadeh, M. Mahmoodi, R. Nazari, L. Shahmoradi, and S. Rezayi, "Smartphone-based application to control and prevent overweight and obesity in children: design and evaluation," *BMC Med. Inform. Decis. Mak.*, vol. 23, no. 1, pp. 1–14, 2023, https://doi.org/10.1186/s12911-023-02304-2.

[39] Y. Feng, C. Shi, C. Zhang, C. Yin, and L. Zhou, "Effect of the smartphone application on caesarean section in women with overweight and obesity: a randomized controlled trial in China," *BMC Pregnancy Childbirth*, vol. 23, no. 1, pp. 1–8, 2023, https://doi.org/10.1186/s12884-023-06004-7.
[40] M. Doumen *et al.*, "Effectiveness and feasibility of a mobile health self-management intervention in rheumatoid arthritis: study protocol for a pragmatic multicentre randomised controlled trial (AEGORA)," *Trials*, vol. 24, no. 1, pp. 1–12, 2023, https://doi.org/10.1186/s13063-023-07733-y.
[41] J. Schobel, T. Probst, M. Reichert, M. Schickler, and R. Pryss, "Enabling Sophisticated Lifecycle Support for Mobile Healthcare Data Collection Applications," *IEEE Access*, vol. 7, pp. 61204–61217, 2019, https://doi.org/10.1109/ACCESS.2019.2916142.
[42] F. Monteiro-Guerra, O. Rivera-Romero, L. Fernandez-Luque, and B. Caulfield, "Personalization in Real-Time Physical Activity Coaching Using Mobile Applications: A Scoping Review," *IEEE J. Biomed. Heal. Informatics*, vol. 24, no. 6, pp. 1738–1751, 2020, https://doi.org/10.1109/JBHI.2019.2947243.
[43] J. Berrocal *et al.*, "Early Evaluation of Mobile Applications' Resource Consumption and Operating Costs," *IEEE Access*, vol. 8, pp. 146648–146665, 2020, https://doi.org/10.1109/ACCESS.2020.3015082.
[44] L. Ben Ammar, "An Automated Model-Based Approach for Developing Mobile User Interfaces," *IEEE Access*, vol. 9, pp. 51573–51581, 2021, https://doi.org/10.1109/ACCESS.2021.3066007.
[45] Z. Yu *et al.*, "RiceMapEngine: A Google Earth Engine-Based Web Application for Fast Paddy Rice Mapping," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 16, pp. 7264–7275, 2023, https://doi.org/10.1109/JSTARS.2023.3290677.
[46] I. Hafidz, A. Priyadi, M. Pujiantara, D. O. Anggriawan, and M. H. Purnomo, "Development of IoT-Based Portable Power Quality Monitoring on Microgrids by Enhancing Protection Features," *IEEE Access*, vol. 11, 2023, pp. 49481–49492, 2023, https://doi.org/10.1109/ACCESS.2023.3276241.
[47] A. Aqle, K. Khowaja, and D. Al-Thani, "Preliminary Evaluation of Interactive Search Engine Interface for Visually Impaired Users," *IEEE Access*, vol. 8, pp. 45061–45070, 2020, https://doi.org/10.1109/ACCESS.2020.2977593.
[48] M. N. Islam, H. Bouwman, and A. K. M. N. Islam, "Evaluating web and mobile user interfaces with semiotics: An empirical study," *IEEE Access*, vol. 8, pp. 84396–84414, 2020, https://doi.org/10.1109/ACCESS.2020.2991840.
[49] S. Yamaguchi *et al.*, "Web Services for Collaboration Analysis with IoT Badges," *IEEE Access*, vol. 10, no. September, pp. 121318–121328, 2022, https://doi.org/10.1109/ACCESS.2022.3222562.
[50] N. Yousaf, F. Azam, W. H. Butt, M. W. Anwar, and M. Rashid, "Automated Model-Based Test Case Generation for Web User Interfaces (WUI) From Interaction Flow Modeling Language (IFML) Models," *IEEE Access*, vol. 7, pp. 67331–67354, 2019, https://doi.org/10.1109/ACCESS.2019.2917674.

BIOGRAPHY OF AUTHORS

Novalanza Grecea Pasaribu, born in Blora, Central Java, Indonesia, is an enthusiastic individual with a high dedication to the field of signal processing. In 2023, he successfully earned his Bachelor's degree in Telecommunication Engineering from Telkom University. Currently, Novalanza is pursuing his Master's degree in Electrical Engineering at Telkom University, focusing on signal processing. For any professional or academic inquiries, Novalanza can be contacted via email at novalanza15@gmail.com.

Menara Alif, was born in Bekasi, West Java, Indonesia, in 2001. He received a bachelor's degree in telecommunications engineering from Telkom University in 2023. Currently, Menara Alif is pursuing his Master's degree in Electrical Engineering at Telkom University. Email: menaraalif@gmail.com.

Gelar Budiman, was born in Bandung, West Java, Indonesia, in 1978. He received B.E. and M.E. degrees in Electrical Engineering from Sekolah Tinggi Teknologi Telkom (STTTelkom) in 2005. He completed a doctoral degree from the School of Electrical and Informatics Technologi, Institut Teknologi Bandung, in 2021. He started to be a lecturer at Telecommunication Engineering Institut Teknologi Telkom (ITTelkom) in 2008 until the present. He has been an assistant Professor since 2018. The particular competency of his academic career includes signal processing, information hiding, artificial intelligence, computer vision, wireless communications, quantum signal processing, and security. He is a member of IEEE and has published nine papers in reputed international journals, seven papers in Scopus international proceedings, nine intellectual property rights, and one book. He can be contacted at email: gelarbudiman@telkomuniversity.ac.id.

Fityanul Akhyar, received the B.S. degree from the Institut Teknologi Telkom, Bandung, Indonesia, in 2010, the M.Eng. degree from Syiah Kuala University, Aceh, Indonesia, in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering (Group B), Yuan Ze University, Taoyuan, Taiwan, with a fully-funded scholarship. He joined the Polytechnique of Aceh in 2017, where he was a Lecturer, the Secretary of informatics study program and the Chair of the Multimedia Laboratory. He has authored some high-prestigious journal and conference papers. He is also a Lecturer with Telkom University, which is one of the best private universities in Indonesia. His research interests include multimedia communication, computer vision, and an automatic optical inspection system. He can be contacted at email: fityanul@telkomuniversity.ac.id.