A Numerical Methods Solver for Large-Scale Linear Systems using Android Studio featuring Step-by-Step Visualization

Writer's Profile		
Writer of this template	University name	Negros Oriental State University
	Full name	REFUGIO, Craig N. TAYCO, Ryan O. TECSON, Rose Mae S.
	Relationship	Director of the Office of the University of International Relations Campus Director of NORSU Main Campuses I and II Technical Writer
	Official title	Director of the Office of the University of International Relations Campus Director of NORSU Main Campuses I and II Technical Writer
	Email address	craig.refugio@norsu.edu.ph
	Phone number	(+63)(035) 995 440 8586
Program Profile		
Program	Program name	A Numerical Methods Solver for Large-Scale Linear Systems using Android Studio featuring Step-by-Step Visualization
	Category	C3
Champion (Program's physical leader)	Full name	YASI, Noel Marjon E.
	Official title	University President
	Email address	norsupres@norsu.edu.ph
	Phone number	(+63)(035) 522-5050 local 1000

School, College, or Headquarters to Which the Case Belongs (Please review the list below and write the corresponding number in the box.)	
From the list below, please enter **only one department number** between 1 and 22.	12

Cross-Cutting & Institutional
1. UH – University Headquarters
2. AI – Artificial Intelligence & Data Science
3. IN – Interdisciplinary & Emerging Fields

Professional & Career-Oriented Schools
4. BS – Business / Management
5. LW – Law
6. MD – Medicine
7. DS – Dentistry
8. NS – Nursing & Health Sciences
9. PH – Pharmacy
10. PB – Public Health

Science, Technology & Applied Fields
11. EN – Engineering
12. CS – Computer Science & Information Technology
13. AR – Architecture & Urban Planning
14. AG – Agriculture & Life Sciences
15. EV – Environmental & Sustainability Studies

Humanities, Arts & Social Sciences
16. HU – Humanities
17. SS – Social Sciences
18. ED – Education
19. AD – Art & Design
20. MU – Music
21. PF – Performing Arts (Theatre/Film/Dance)
22. DV – Divinity / Theology / Religion

WURI's Serial Number	(Leave this box blank.)

A Numerical Methods Solver for Large-Scale Linear Systems using Android Studio featuring Step-by-Step Visualization

WURI's Serial Number		(Leave this box blank.)
Summary of Program		
Program Name		A Numerical Methods Solver for Large-Scale Linear Systems using Android Studio featuring Step-by-Step Visualization
Category		C3
Abstract of Program		Solving large systems of linear equations is a fundamental engineering task, but current tools lack transparency and educational effectiveness for students studying advanced numerical methods. This project aimed to design, implement, and assess a "Step-by-Step Numerical Methods Calculator", a tailored Android application that helps students solve up to ten linear equations using five different methods (Cramer's Rule, Gaussian/Gauss-Jordan Elimination, Jacobi/Gauss-Seidel), while providing step-by-step visualizations of the solution process. The app was evaluated through a task-based approach involving 23 engineering students who solved an assigned 4x4 system and completed a post-task questionnaire. The results demonstrated high functional reliability (100% task completion), and students rated the application highly for its user-friendly interface (M=4.61/5.00) and clarity of the step-by-step process (M=4.09/5.00). Cronbach's Alpha of 0.704 confirmed the reliability of the questionnaire. While users commended the app's speed and usefulness, qualitative feedback highlighted the need for improvements in the information architecture to further reduce cognitive load. This research confirms that the app serves as an innovative educational tool that bridges the gap between conceptual knowledge and computational skills, offering a future-ready solution for engineering education by integrating technology into the learning process. The app enhances learning experiences and prepares students for future challenges in engineering and other technical fields.
Details of Program		
Planning		
Objectives	Long-term Goals	Our long-term goal is to revolutionize the way engineering students understand and solve complex mathematical problems. By continuously improving and refining the **Step-by-Step Numerical Methods Calculator**, we envision it becoming an indispensable tool in engineering education worldwide. We aspire for this application to bridge the gap between conceptual knowledge and computational skills, preparing students for future challenges in the ever-evolving field of engineering. In the coming years, we hope to expand the app's capabilities, integrating it with more advanced methods and expanding its accessibility to a global audience of students and professionals.

	Short-term Targets	In the upcoming year, we aim to solidify the app's reliability and functionality by further testing and refining its features. We plan to expand its user base through targeted outreach, gathering data from a diverse set of students and professionals to ensure the app's versatility and educational impact. Our focus will be on perfecting the user experience, enhancing step-by-step visualizations, and integrating additional user feedback to ensure its accessibility and usefulness for future learners. We also aim to release the app on popular platforms like Google Play to reach a wider audience.
	Rationale	The reason for initiating this project stems from the clear gap in existing tools that support students learning complex numerical methods. Engineering education, particularly in numerical analysis, requires transparent, intuitive, and interactive tools to make difficult concepts more accessible. Traditional methods lack the step-by-step breakdowns and visual aids that could enhance student comprehension. By developing this app, we hope to empower students to not only solve large-scale linear systems with ease but also to deeply understand the underlying processes involved. This initiative represents our commitment to shaping future engineers who can approach complex problems with confidence and clarity.
Subject (Leader)	Initiator(s)	REFUGIO, Craig N. TAYCO, Ryan O. RAGAY, Jenny D.
	Champion(s)	YASI, Noel Marjon E.
	Major team member(s)	CANTIL, Jeric Jan BARBOSA, Alfred MAANO, Remejie BESARIO, Harold Jan MELON, Aeron James
Environment	Nature/Society	This application, developed for students tackling complex linear algebra problems, is designed with a clear understanding of the educational landscape. It supports the evolving nature of engineering education by addressing the need for clearer, more accessible tools that enhance conceptual learning. The study's focus on making engineering education more transparent through technology meets society's growing demand for practical, applied learning tools that improve problem-solving skills.
	Industry/Market	The tool meets the industry's needs by aligning with current technological trends, such as mobile applications and AI-driven tools. It facilitates the development of future engineers who are equipped with the technical and problem-solving skills needed by industries that rely on advanced computation. The app directly supports the industry's emphasis on innovation, practical application, and interdisciplinary integration, ensuring students are well-prepared for the market demands.
	Citizen/Government	Government initiatives focusing on digital literacy and STEM education align perfectly with the study's objectives. The app supports national educational frameworks by fostering technical skills and problem-solving capabilities, helping create a generation of engineers who can contribute to national development and infrastructure.
Resources	Human resources	The study relies on a dedicated team of students and advisors, who bring diverse skills in engineering, software development, and educational methodologies. The technical proficiency required to develop the app was guided by faculty advisors, ensuring that the application was both

		scientifically rigorous and educationally valuable. The team's collaboration with professional stakeholders ensures a high level of technical support and pedagogical expertise.
	Financial resources	This project utilized university-funded resources in terms of time, access to software tools like Android Studio, and faculty mentorship. The app is designed to be scalable and cost-effective, allowing for future expansion at minimal costs, especially with the availability of free and open-source platforms for app development.
	Technological resources	The primary technological resource for this study was Android Studio, which facilitated the development of the mobile application. The app leverages modern programming languages like Java and Kotlin to integrate essential mathematical methods and algorithms. The use of mobile technology allows for a greater reach and accessibility for students and professionals, making it a future-proof educational tool that can be expanded with additional features like AI and cloud computing.
Mechanism	Strategy (Weight/Sequence)	The strategy focuses on the development of a highly accessible mobile application that addresses a specific educational gap in solving large-scale linear systems. The project emphasizes the application of various numerical methods such as Cramer's Rule, Gaussian Elimination, and Jacobi Iteration. The primary goal in the short term is refining the tool based on user feedback, with plans to expand its functionalities in future versions, ensuring it meets the growing demand for flexible and interactive learning tools in STEM education.
	Organization	The organizational structure of the university supports this innovation by integrating research and development into the curriculum. The collaborative effort between students and faculty, along with support from the College of Engineering, ensures that the project aligns with the institution's vision of enhancing education through technological advancements.
	Culture	The university culture encourages innovation and technological integration into educational methods. This study is a perfect reflection of that cultural alignment, as it promotes critical thinking, practical learning, and problem-solving skills that are essential for the success of engineering students. The supportive academic environment enables the project to thrive and be shared with a wider academic community.
Doing		
Launch date		The app was launched in **June 2025**, marking a significant milestone in bridging the gap between conceptual learning and computational skill acquisition in engineering education.
Responsible organization		The **College of Engineering and Architecture** at **Negros Oriental State University (NORSU)**, led by a dedicated group of engineering students and supported by the expertise of faculty members, is responsible for the execution of this groundbreaking program. The project was developed as part of the **COE 346 Methods of Research** course under the guidance of **Dr. Craig N. Refugio.**
Program content and process		This program developed a mobile application designed to help students and professionals solve large-scale linear systems with step-by-step solutions using methods like **Cramer's Rule**, **Gaussian Elimination**, and **Jacobi Iteration**, among others. The application is tailored for Android devices and guides users through each solution process, visualizing every step for clarity and understanding. The implementation process followed a structured methodology, with careful design, prototyping, and iterative testing. The

	app was created with a focus on ease of use, educational value, and user engagement, allowing students to solve equations and understand the underlying concepts intuitively.
Key highlights of the content/process	1. **Step-by-Step Visualization**: Every solution step is clearly visualized, making complex mathematical processes more accessible. 2. **Mobile Integration**: By utilizing Android Studio, the application was designed to run smoothly on mobile devices, increasing accessibility for students anytime, anywhere. 3. **Interactive Learning**: The app offers a hands-on learning approach, where users can explore different solution methods, understand their applications, and see real-time results.
Differences from traditional approaches	Traditional methods of teaching numerical methods rely heavily on lectures, static examples, and textbook solutions, often without interactivity or visualization. This app, however, offers **interactive, visual, and real-time feedback**, creating a dynamic learning environment where students actively engage in problem-solving rather than merely following pre-determined steps. Additionally, the mobile platform ensures that learning is accessible beyond the classroom, fostering a more flexible and personalized learning experience.
Progress as of today	As of today, the app has successfully undergone **pilot testing** with engineering students, and the feedback has been overwhelmingly positive. The application has achieved its goal of providing a functional and user-friendly tool for solving large-scale linear systems. **100% task completion** was reported in the evaluation, and the app has been rated highly for its clarity and ease of use (M=4.61/5.00).
Problems in implementation	One challenge faced during the implementation was ensuring **user-friendly interaction** with the mobile interface, particularly in presenting complex solutions clearly and concisely. Some users expressed the need for a more intuitive information architecture to minimize cognitive load, particularly when interpreting long, multi-step solutions.
Approaches to solve the problems	To address the aforementioned issue, we refined the **user interface (UI)** and **user experience (UX)** based on continuous feedback. We also focused on enhancing the **visualization** and **step-by-step navigation** to reduce cognitive load. As a result, the final version of the app is more streamlined, with clear guidance for each step, and it provides better explanations for the user.
Completion date, if completed	The program's **initial launch** and pilot testing phase were successfully completed in **June 2025**. We are now in the process of incorporating user feedback for the **next version**. The full rollout and expansion of features, including integration with more methods and platforms, are anticipated for **early 2026**.
Seeing	
Impacts on students	The **Step-by-Step Numerical Methods Calculator** has had a profound impact on students by enhancing their ability to understand and solve complex linear systems with clarity and confidence. Students have reported high satisfaction with the tool, especially in how it makes difficult mathematical concepts more accessible. The app has fostered a deeper understanding of numerical methods through its interactive visualizations,

	allowing students to engage actively in their learning and gain a thorough understanding of the problem-solving process. With **100% task completion** and impressive ratings for user simplicity, students find the app not just a tool, but a valuable learning companion.
Impacts on professors	Professors have embraced the application as a transformative educational tool that elevates the learning experience. It has reduced the reliance on traditional teaching methods, allowing instructors to focus on guiding students through problem-solving rather than simply delivering lectures. Professors have noted significant improvements in student engagement and understanding, as the app's interactive nature sparks interest and curiosity. The simplicity and clarity of the step-by-step explanations have allowed professors to focus more on conceptual discussions, knowing that students have mastered the technical application.
Impacts on university administration	The university administration has recognized the potential of this project to position **Negros Oriental State University (NORSU)** as a leader in integrating technology with education. The app aligns perfectly with the university's strategic goals of enhancing STEM education and improving student engagement. The positive student and faculty feedback demonstrates the effectiveness of the app, reinforcing the university's commitment to fostering innovation in teaching and learning. The project exemplifies the kind of forward-thinking approach that is integral to the university's educational philosophy.
Responses from industry/market	The response from the industry and local market has been encouraging. Engineering firms and educational technology providers have recognized the potential of this application to not only enhance learning but also contribute to developing a workforce equipped with practical computational skills. Industries that rely on engineers proficient in numerical methods have expressed interest in using this tool as a part of their internal training programs, as it provides practical, hands-on experience with solving real-world problems.
Responses from citizen/government	Local government agencies and educational policymakers have also shown support for this project. They see the app as an important step towards enhancing **digital literacy** and **STEM education** within the region. By supporting educational initiatives like this, they are aligning with national goals to improve access to high-quality educational resources and equip students with the skills necessary for the future workforce. This initiative also plays a role in promoting **digital inclusion** and ensuring that all students, regardless of location, have access to world-class learning tools.
Measurable output (revenues)	While the program is still in its early stages, its revenue potential is promising. The app, once released on platforms like Google Play, is expected to generate income through **app purchases**, **in-app features**, and **educational partnerships** with universities and industries. Further monetization could include offering additional advanced features or enterprise solutions for large educational institutions and corporations, enhancing the scalability and profitability of the project.
Measurable input (expenses)	The primary expenses associated with the development of this application include software tools, **Android Studio** licenses, and the research and development time of the students and faculty involved. Additionally, marketing costs for the app's release and distribution, as well as any support infrastructure (e.g., server maintenance), contribute to the financial inputs. The costs were kept low through university resources and the use of free software development tools, ensuring a cost-effective project launch.

Cost-benefit analysis for effectiveness	The cost-benefit analysis shows a strong return on investment. The initial financial input has been minimal, relying on free tools and student labor. The benefits, however, are substantial: enhancing student learning, improving faculty engagement, and promoting the university's role as a leader in educational technology. The positive student feedback and high completion rates indicate that the project is highly effective in its objectives. Moreover, the app's ability to scale and serve a wider audience globally through mobile platforms further solidifies its potential for long-term success, both educationally and financially.
Future Planning	
Where does the project go from here?	The journey of developing the **Step-by-Step Numerical Methods Calculator** has only just begun. As we look to the future, there are several exciting paths ahead for the project. First, we aim to **expand the app's functionality** by integrating more advanced numerical methods, ensuring that it continues to meet the growing demands of students and professionals in the field of engineering and beyond. We will also explore **cross-platform compatibility**, enabling users on both **iOS** and **Android** platforms to benefit from this innovative educational tool. Our future vision includes the **integration of AI-driven enhancements**, such as real-time feedback based on user input and suggestions for improving problem-solving approaches. This will make the learning experience even more personalized and adaptive to individual learning styles. Furthermore, we are committed to **scaling the app's reach** to a global audience by partnering with universities and educational organizations worldwide, making it an essential part of modern engineering education. As we continue to gather feedback and iterate based on real-world use, we foresee the app evolving into a comprehensive educational platform for numerical methods that can be used by both novice and experienced learners alike.

Addendum

Exhibits, pictures, diagrams, etc.

Appendix A: User Testing Instruments

This appendix contains the documents provided to each participant during the user testing sessions.

A.1 - Post-Task Questionnaire

This is the form given to each student after they solved the sample problem.

App Performance and Usability Questionnaire

Instructions: Please answer the following questions based on your experience using the Numerical Methods Calculator to solve the sample problem.

Part 1: Scaled Questions
Please circle the number that best represents your experience, where:

1 = Strongly Disagree

2 = Disagree

3 = Neutral

4 = Agree

5 = Strongly Agree

1. How easy was it to enter the linear equations into the app?
   1--- 2 --- 3 --- 4---5
2. How easy was it to find and select the solving method?
   1--- 2 --- 3 --- 4---5
3. How would you rate the speed of the app in calculating the solution?
   1--- 2 --- 3 --- 4---5
4. How clear and understandable were the step-by-step results shown by the app?
   1--- 2 --- 3 --- 4---5
5. Overall, how would you rate your experience using this app?
   1--- 2 --- 3 --- 4---5

Part 2: Open-Ended Questions
(Please write a brief answer)

1. What did you like most about the app?
   ____
2. What was the most difficult or confusing part of using the app? Is there anything you would change to make it better?
   ____

A.2 - Sample Problem Handout

This is the problem given to the student testers.

System of Equations:

$$x_1 + x_2 + x_3 + x_4 = 5$$

$$x_1 + 2x_2 - x_3 + x_4 = 9$$

$$2x_1 + 3x_2 + 2x_3 - x_4 = 3$$

$$3x_1 - x_2 - 2x_3 + 2x_4 = 9$$

(Known Solution for Verification: $x_1=1$, $x_2=2$, $x_3=-1$, $x_4=3$)

Appendix B: Raw Survey Data from User Testing

Table B.1 presents the raw, anonymized data collected from the post-task questionnaire. The questionnaire was administered to 23 engineering students after they used the Numerical Methods Calculator to solve a sample 4x4 system of linear equations.

LEGEND:

This legend is for Table B.1.

COLUMNS	DESCRIPTION
ID	Participant ID
Q1	Ease of Input (1-5)
Q2	Ease of Method Selection (1-5)
Q3	Calculation Speed (1-5)
Q4	Clarity of Steps (1-5)
Q5	Overall Experience (1-5)
S1	Positive Feedback (Liked Most)
S2	Areas for Improvement / Difficulties

Table B.1.

Raw Survey Data from User Testing

ID	Q1	Q2	Q3	Q4	Q5	S1	S2
1	4	4	3	4	4	The Functions	The confusing part is UI.
2	5	5	5	5	5	I like how fast it solves and how easy to use it.	It is understandable and not confusing.
3	5	5	5	4	4	Direct result.	"Maybe about the solution or what you call that solving in columns, somehow dazed."
4	5	5	5	4	5	calculator	instructions
5	5	5	3	4	5	I could use different methods.	I think the step by step has so many zeros my eyes can't keep up.
6	4	5	3	3	4	Different methods available.	I think on how the methods work.
7	5	4	4	3	5	Easy Access	Need to scroll a lot.
8	4	4	4	5	4	Easy to understand	"Too many datas compiled, hard to read."
9	5	5	4	3	4	It's easy to use.	Some solutions are confusing.
10	5	5	5	5	5	Fast and reliable	None.
11	5	5	4	3	5	How easy it is to use.	I don't [have] any problems using the app.
12	5	5	5	5	5	I could solve large scale of linear equations.	"I love the app, I can't complain."

ID	Q1	Q2	Q3	Q4	Q5	S1	S2
13	4	4	4	5	5	Quick and easy	Function that datas are compiled and can be exported to like microsoft or something.
14	5	5	4	4	4	Using the app is easy.	No problem at all.
15	5	5	4	5	5	This is really useful for solving linear equations.	Nothing.
16	5	5	4	4	5	Fast in getting the answer.	Option to change the colors.
17	5	5	4	4	4	Different methods to use.	You can change the UI.
18	5	5	5	5	5	The fact that it is very easy to use it.	I suggest putting dark mode on the app.
19	4	4	3	4	4	Easy to Use	Make it available on IOS.
20	3	4	4	5	4	User Friendly and can give the answer immediately.	Would love if it can be used in IOS gadgets.
21	5	5	4	4	5	Easy to use.	I think add more methods.
22	4	4	3	3	4	can run on low-end device	Some solutions are confusing to look
23	4	5	4	3	5	Easy to use	Can be used to IOS if possible

Appendix C: Application Screenshots

This appendix gives a pictorial guide to the user interface of the application. The screenshots provided document the most important screens and are meant to explain the app's central functions and general design structure. By showing the general procedure a user goes through, this section gives a tangible, step-by-step guide to the user process from beginning to end.

Figure C.1: Method Selection Screen

Figure C.1 shows the app's main navigation tool, the side menu. This can be accessed at any time in the app, and the user can easily switch between the various numerical methods supported for the solution of systems of linear equations. The menu contains all the algorithms supported, such as the Gauss-Seidel Method, Jacobi Iteration, Gaussian Elimination, Cramer's Rule, and Gauss-Jordan's Rule. The highlighted item, "Cramer's Rule Calculator," sends concise visual feedback to the user indicating the currently active method. The design makes sure that all of the app's main functionalities are easily discoverable and accessible from a single, central hub.

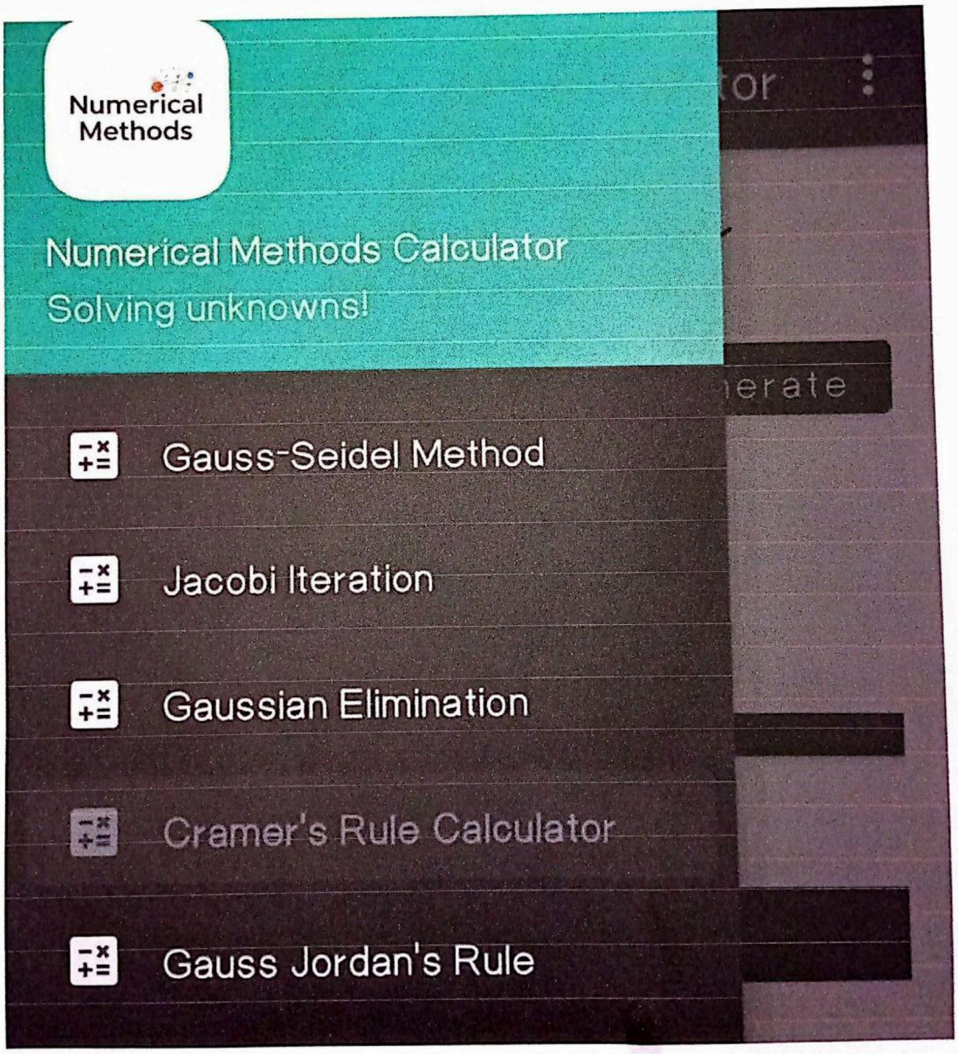


Figure C.1: Method Selection Screen

Figure C.2: Matrix Input Screen

Figure C.2 shows the main data entry form for the "Cramer's Rule Calculator," a look-and-feel that repeats throughout all numerical tools in the application. The process starts by the user defining the "Number of Variables" (1 to 10) for the system of equations to be solved. Upon clicking "Generate," the form automatically generates the corresponding input fields for the Coefficient Matrix (A) and the Constants Vector (B)

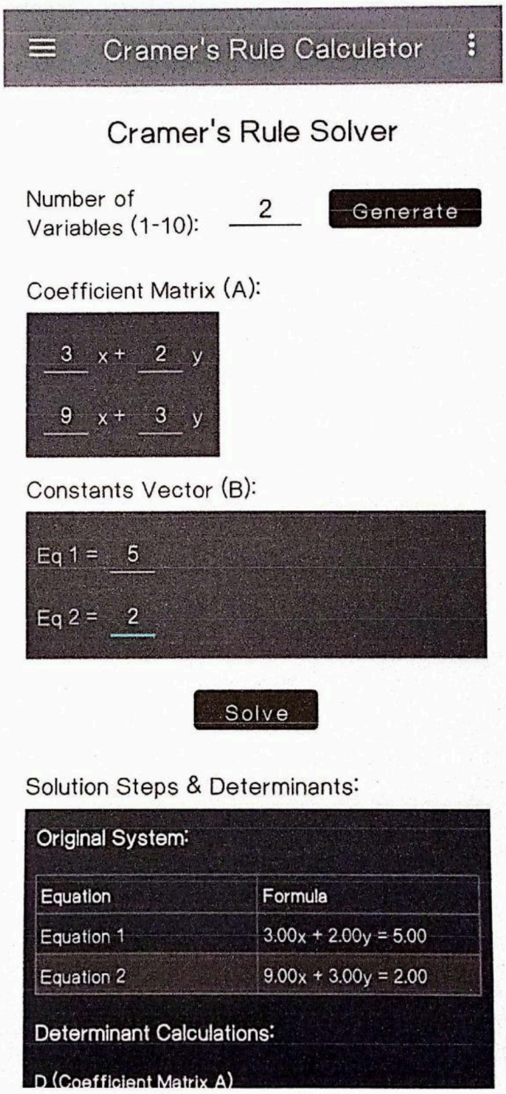
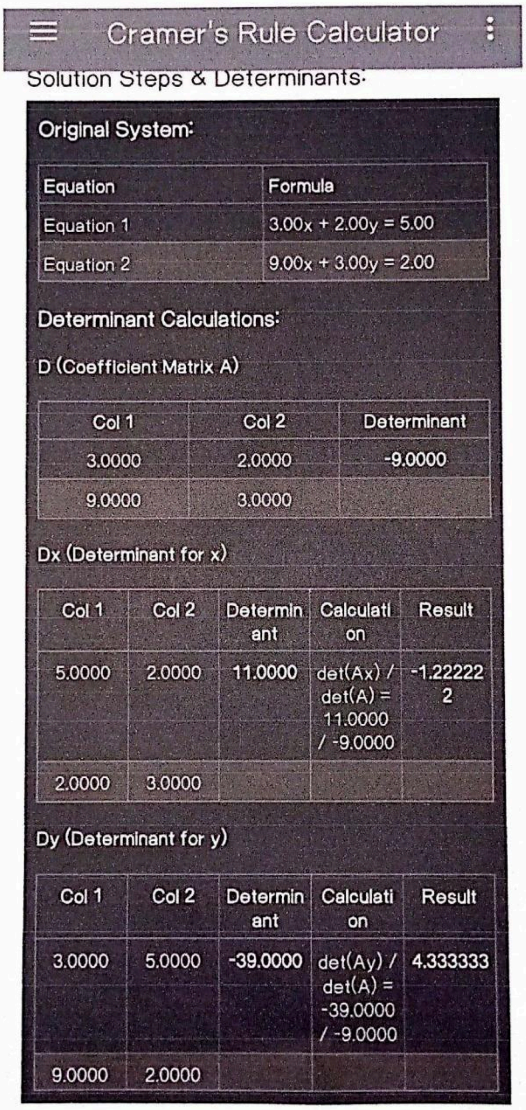


Figure C.2: Matrix Input Screen

Figure C.3: Step-by-Step Solution Display

Figure C.3 demonstrates the application's core educational feature: the detailed solution breakdown. After the user presses the "Solve" button, the interface populates with a comprehensive, step-by-step process for the selected numerical method. The output begins with a clear presentation of the "Original System" in a tabular format, allowing the user to verify that their input was entered correctly.



Equation	Formula
Equation 1	3.00x + 2.00y = 5.00
Equation 2	9.00x + 3.00y = 2.00

Determinant Calculations:

D (Coefficient Matrix A)

Col 1	Col 2	Determinant
3.0000	2.0000	-9.0000
9.0000	3.0000	

Dx (Determinant for x)

Col 1	Col 2	Determinant	Calculation	Result
5.0000	2.0000	11.0000	det(Ax) / det(A) = 11.0000 / -9.0000	-1.222222
2.0000	3.0000			

Dy (Determinant for y)

Col 1	Col 2	Determinant	Calculation	Result
3.0000	5.0000	-39.0000	det(Ay) / det(A) = -39.0000 / -9.0000	4.333333
9.0000	2.0000			

Figure C.3: Step-by-Step Solution Display

Figure C.4: Final Answer Display

Figure C.4 illustrates the culmination of the computational process: the "Final Solution" summary table. This component is strategically placed at the end of the results display to provide a definitive and easily digestible answer set. The design of this table prioritizes clarity and reinforces the educational objectives of the application.

Final Solution:

Variable	Calculation	Value
x	det(Ax) / det(A) = 11.0000 / -9.0000	-1.2222222
y	det(Ay) / det(A) = -39.0000 / -9.0000	4.3333333

Figure C.4: Final Answer Display

Appendix D: Key Source Code Snippets

This appendix contains excerpts from the Java source code to illustrate the implementation of a core algorithm. Including all source code is impractical; therefore, a key function is presented.

D.1 - Gaussian Elimination Algorithm in Java

The following code, written in Java, shows how the core logic of gaussian elimination is implemented:

```java
private void solveGaussianElimination() {

    int numVariables = A.length;
    double[][] augmentedMatrix = createAugmentedMatrix(A, B);

    // --- Part 1: Forward Elimination ---
    for (int p = 0; p < numVariables; p++) {
        int maxRow = p;
        for (int i = p + 1; i < numVariables; i++) {
            if (Math.abs(augmentedMatrix[i][p]) > Math.abs(augmentedMatrix[maxRow][p])) {
                maxRow = i;
            }
        }

        if (maxRow != p) {
            swapRows(augmentedMatrix, p, maxRow);
        }

        double pivotElement = augmentedMatrix[p][p];
        for (int j = p; j <= numVariables; j++) {
            augmentedMatrix[p][j] /= pivotElement;
        }

        for (int i = p + 1; i < numVariables; i++) {
            double factor = augmentedMatrix[i][p];
            for (int j = p; j <= numVariables; j++) {
                augmentedMatrix[i][j] -= factor * augmentedMatrix[p][j];
            }
        }
    }

    // --- Part 2: Back Substitution ---
    double[] solutions = new double[numVariables];
    for (int i = numVariables - 1; i >= 0; i--) {
        double sum = 0;
        for (int j = i + 1; j < numVariables; j++) {
            sum += augmentedMatrix[i][j] * solutions[j];
        }
        solutions[i] = (augmentedMatrix[i][numVariables] - sum) / augmentedMatrix[i][i];
    }
}
```

D.2 - Cramer's Rule Algorithm in Java

The following code, written in Java, shows how the core logic of Cramer's Rule is implemented:

```
private void solveCramersRule() {

    // Part 1: Calculate the determinant of the main coefficient matrix (A).
    double detA = determinant(A, numVariables);

    // Part 2: Check for a unique solution.
    if (Math.abs(detA) < DETERMINANT_ZERO_THRESHOLD) {
        // System may have no unique solution.
        return;
    }

    // Part 3: Calculate the determinant for each variable's matrix (Ak) and solve.
    double[] solutions = new double[numVariables];
    for (int k = 0; k < numVariables; k++) {
        double[][] Ak = new double[numVariables][numVariables];
        for (int i = 0; i < numVariables; i++) {
            for (int j = 0; j < numVariables; j++) {
                Ak[i][j] = (j == k) ? B[i] : A[i][j];
            }
        }

        double detAk = determinant(Ak, numVariables);

        solutions[k] = detAk / detA;
    }
}
```

46

CamScanner

D.3 - Gauss Jordan Algorithm in Java

The following code, written in Java, shows how the core logic of Gauss Jordan algorithm is implemented:

```java
private void solveGaussJordan() {

    // Part 1: Forward and Backward Elimination.
    for (int p = 0; p < numVariables; p++) {
        int maxRow = p;
        for (int i = p + 1; i < numVariables; i++) {
            if (Math.abs(augmentedMatrix[i][p]) > Math.abs(augmentedMatrix[maxRow][p])) {
                maxRow = i;
            }
        }

        if (maxRow != p) {
            swapRows(augmentedMatrix, p, maxRow);
        }

        double pivotElement = augmentedMatrix[p][p];
        if (Math.abs(pivotElement) < PIVOT_ZERO_THRESHOLD) {
            return;
        }

        for (int j = p; j <= numVariables; j++) {
            augmentedMatrix[p][j] /= pivotElement;
        }

        for (int i = 0; i < numVariables; i++) {
            if (i != p) {
                double factor = augmentedMatrix[i][p];
                for (int j = p; j <= numVariables; j++) {
                    augmentedMatrix[i][j] -= factor * augmentedMatrix[p][j];
                }
            }
        }
    }

    // Part 2: Extract Solution.
    double[] solutions = new double[numVariables];
    for (int i = 0; i < numVariables; i++) {
        solutions[i] = augmentedMatrix[i][numVariables];
    }
}
```

D.4 - Gauss Seidel Algorithm in Java

The following code, written in Java, shows how the core logic of Gauss-Seidel is implemented:

```
private void solveGaussSeidel() {

    // Part 1: Initialization.
    if (!isDiagonallyDominant(A)) {
        // Warn the user that convergence is not guaranteed.
    }

    double[] X_current = new double[numVariables];
    Arrays.fill(X_current, 0.0);

    int k = 0;
    boolean converged = false;

    // Part 2: Iterative Solution.
    while (k < MAX_ITERATIONS && !converged) {
        double[] X_previous = Arrays.copyOf(X_current,
numVariables);
        for (int i = 0; i < numVariables; i++) {
            double sigma = 0;
            for (int j = 0; j < numVariables; j++) {
                if (i != j) {
                    sigma += A[i][j] * X_current[j];
                }
            }
            if (A[i][i] == 0) {
                return;
            }
            X_current[i] = (B[i] - sigma) / A[i][i];
        }

        k++;

        // Part 3: Convergence Check.
        converged = true;
        for (int i = 0; i < numVariables; i++) {
            double error = Math.abs(X_current[i] - X_previous[i]);
            if (error > CONVERGENCE_THRESHOLD) {
                converged = false;
                break;
            }
        }
    }
}
```

D.5 - Jacobi Iteration Algorithm in Java

The following code, written in Java, shows how the core logic of Jacobi Iteration is implemented:

```
private void solveJacobi() {

    // Part 1: Initialization.
    if (!isDiagonallyDominant(A)) {
        // Warn the user that convergence is not guaranteed.
    }

    double[] X_current = new double[numVariables];
    Arrays.fill(X_current, 0.0);

    int k = 0;
    boolean converged = false;

    // Part 2: Iterative Solution.
    while (k < MAX_ITERATIONS && !converged) {
        double[] X_next = new double[numVariables];

        for (int i = 0; i < numVariables; i++) {
            double sigma = 0;
            for (int j = 0; j < numVariables; j++) {
                if (i != j) {
                    sigma += A[i][j] * X_current[j];
                }
            }
            if (A[i][i] == 0) {
                return;
            }
            X_next[i] = (B[i] - sigma) / A[i][i];
        }

        k++;

        // Part 3: Convergence Check.
        converged = true;
        for (int i = 0; i < numVariables; i++) {
            double error = Math.abs(X_next[i] - X_current[i]);
            if (error > CONVERGENCE_THRESHOLD) {
                converged = false;
                break;
            }
        }

        // Part 4: Update for Next Iteration.
        X_current = Arrays.copyOf(X_next, numVariables);
    }
}
```

Appendix E: Manual Calculations

E.1 - Cramer's Rule:

This example uses the following equations:

Equation 1: $4x + 7y = 4$

Equation 2: $9x - 3y = 2$

Step-by-step Solution:

1. Form the Coefficient Matrix and Calculate its Determinant (D):

   The coefficient matrix, denoted as A, is formed by the coefficients of the variables:

   $$\mathbf{A} = \begin{pmatrix} 4 & 7 \\ 9 & -3 \end{pmatrix}$$

   The determinant D is calculated as:

   $$D = (4 \times -3) - (7 \times 9) = -12 - 63 = -75$$

2. Form Matrix $\mathbf{A}_x$ and Calculate its Determinant (D_x):

   Matrix $\mathbf{A}_x$ is formed by replacing the first column (x-coefficients) of $\mathbf{A}$ with the constant terms from the right-hand side of the equations ($[4, 2]^T$).

   $$\mathbf{A}_x = \begin{pmatrix} 4 & 7 \\ 2 & -3 \end{pmatrix}$$

   The determinant D_x is calculated as:

   $$D_x = (4 \times -3) - (7 \times 2) = -12 - 14 = -26$$

3. Form Matrix $\mathbf{A}_y$ and Calculate its Determinant (D_y):

   Matrix $\mathbf{A}_y$ is formed by replacing the second column (y-coefficients) of $\mathbf{A}$ with the constant terms from the right-hand side of the equations ($[4, 2]^T$).

   $$\mathbf{A}_y = \begin{pmatrix} 4 & 4 \\ 9 & 2 \end{pmatrix}$$

The determinant D_y is calculated as:

$$D_y = (4 \times 2) - (4 \times 9) = 8 - 36 = -28$$

4. Calculate the Values of x and y:

Using Cramer's Rule formulas, $x = D_x/D$ and $y = D_y/D$::

$$x = \frac{-26}{-75} \approx 0.346667$$

$$y = \frac{-28}{-75} \approx 0.373333$$

Solution:

The solution to the system of equations is approximately $x = 0.346667$ and $y = 0.373333$.

E.2 - Jacobi Iteration

This example uses the following equations:

Equation 1: $36x - 2y - 3z = 2$

Equation 2: $3x - 45y - 8z = -2$

Equation 3: $3x - 4y - 67z = -2$

Step-by-step Solution:

1. Rearrange the Equations:

First, each equation is rearranged to solve for one variable in terms of the others. This assumes diagonal dominance for convergence, though the method can sometimes converge even without it.

From Equation 1 (for x):

$$36x = 2y + 3z + 2$$

$$x = (2y + 3z + 2)/36$$

From Equation 2 (for y):

$$-45y = -3x + 8z - 2$$

$$y = (3x - 8z + 2)/45$$

From Equation 3 (for z):

$$-67z = -3x + 4y - 2$$

$z = (3x - 4y + 2)/67$

2. Initialize Values and Iterate:

Start with an initial guess, typically $x_0 = 0, y_0 = 0, z_0 = 0$. Each subsequent approximation ($x_{k+1}, y_{k+1}, z_{k+1}$) is calculated using the values from the previous iteration (x_k, y_k, z_k). The process continues until the change in values between iterations (error) falls below a predefined tolerance.

Iteration 0 (Initial Guess):

$x_0 = 0, y_0 = 0, z_0 = 0$

Iteration 1:

Using x_0, y_0, z_0:

$x_1 = (2(0) + 3(0) + 2)/36 = 2/36 \approx 0.055556$

$y_1 = (3(0) - 8(0) + 2)/45 = 2/45 \approx 0.044444$

$z_1 = (3(0) - 4(0) + 2)/67 = 2/67 \approx 0.029851$

Iteration 2:

Using x_1, y_1, z_1:

$x_2 = (2(0.044444){+}3(0.029851){+}2)/36 = (0.088888{+}0.089553{+}2)/36 \approx 2.178441/36 \approx 0.060512$

$y_2 = (3(0.055556){-}8(0.029851){+}2)/45 = (0.166668{-}0.238808{+}2)/45 \approx 1.92786/45 \approx 0.042841$

$z_2 = (3(0.055556){-}4(0.044444){+}2)/67 = (0.166668{-}0.177776{+}2)/67 \approx 1.988892/67 \approx 0.029685$

Error Calculation (E):

After each iteration, the absolute error for each variable is computed as $|Value_{k+1} - Value_k|$. The process continues until these errors (or a chosen norm of the error vector) fall below a defined tolerance (e.g., 10^{-10} or 10^{-15}).

3. Final Solution:

As shown in the table, the iterations converge to the following approximate values:

$x \approx 0.060451$

$y \approx 0.043144$

$z \approx 0.029982$

E.3 - Gauss-Seidel

This example uses the following equations:

Equation 1: $128x + 16y + 46z = 99$

Equation 2: $21x + 121y + 58z = 66$

Equation 3: $32x - 53y + 132z = 42$

Step-by-step Solution:

1. Rearrange the Equations:

Similar to Jacobi, each equation is rearranged to solve for one variable. The order of equations and the assignment of variables to equations should ensure diagonal dominance for optimal convergence.

From Equation 1 (for x):

$128x = -16y - 46z + 99$

$x = (-16y - 46z + 99)/128$

From Equation 2 (for y):

$121y = -21x - 58z + 66$

$y = (-21x - 58z + 66)/121$

From Equation 3 (for z):

$132z = -32x + 53y + 42$

$z = (-32x + 53y + 42)/132$

2. Initialize Values and Iterate (Using Updated Values Immediately):

Begin with an initial guess (commonly $x_0 = 0, y_0 = 0, z_0 = 0$). The key difference from Jacobi is that Gauss-Seidel uses the most recently computed values for the variables in the current iteration as soon as they become available.

Iteration 0 (Initial Guess):

$x_0 = 0, y_0 = 0, z_0 = 0$

Iteration 1:

- Calculate x_1:

Using $y_0 = 0, z_0 = 0$:

$x_1 = (-16(0) - 46(0) + 99)/128 = 99/128 \approx 0.773438$

- Calculate y_1:

Using the newly calculated $x_1 = 0.773438$ and $z_0 = 0$:

$y_1 = (-21(0.773438) - 58(0) + 66)/121 = (-16.242198 + 66)/121 = 49.757802/121 \approx 0.411222$

- Calculate z_1:

Using the newly calculated $x_1 = 0.773438$ and $y_1 = 0.411222$:

$z_1 = (-32(0.773438) + 53(0.411222) + 42)/132 = (-24.749984 + 21.794766 + 42)/132 = 39.044782/132 \approx 0.295794$

3. Error Calculation and Convergence:

The absolute error for each variable $(|Value_{k+1} - Value_k|)$ is calculated after each full iteration (where $x_{k+1}, y_{k+1}, z_{k+1}$ have all been computed). The iteration stops when these errors fall below a specified tolerance.

4. Final Solution:

As the iterations converge, the solution approaches:

$x \approx 0.633777$

$y \approx 0.299037$

$z \approx 0.284607$

E.4 - Gaussian Elimination

This example uses the following equations:

Equation 1: $4x + 9y + 2z = 8$

Equation 2: $7x + 3y - 7z = 10$

Equation 3: $9x - 13y + 15z = 10$

Step-by-step Solution:,

1. Form the Augmented Matrix:

The system of equations is first represented as an augmented matrix $[A|B]$:

$$\begin{pmatrix} 4 & 9 & 2 & | & 8 \\ 7 & 3 & -7 & | & 10 \\ 9 & -13 & 15 & | & 10 \end{pmatrix}$$

2. Forward Elimination (Transform to Row Echelon Form):

The goal of this phase is to convert the augmented matrix into an upper triangular form by performing elementary row operations.

Step 2.1: Make the leading coefficient of Row 1 equal to 1.

Divide Row 1 by 4 ($R_1 \leftarrow R_1/4$):

$$\begin{pmatrix} 1 & 2.25 & 0.5 & | & 2 \\ 7 & 3 & -7 & | & 10 \\ 9 & -13 & 15 & | & 10 \end{pmatrix}$$

Step 2.2: Eliminate x-coefficients in Row 2 and Row 3.

Subtract 7 times Row 1 from Row 2 ($R_2 \leftarrow R_2 - 7R_1$):

Subtract 9 times Row 1 from Row 3 ($R_3 \leftarrow R_3 - 9R_1$):

$$\begin{pmatrix} 1 & 2.25 & 0.5 & | & 2 \\ 0 & -12.75 & -10.5 & | & -4 \\ 0 & -33.25 & 10.5 & | & -8 \end{pmatrix}$$

Step 2.3: Make the leading coefficient of Row 2 equal to 1.

Divide Row 2 by -12.75 ($R_2 \leftarrow R_2/-12.75$):

$$\begin{pmatrix} 1 & 2.25 & 0.5 & | & 2 \\ 0 & 1 & 0.823529 & | & 0.313725 \\ 0 & -33.25 & 10.5 & | & -8 \end{pmatrix}$$

Step 2.4: Eliminate y-coefficient in Row 3.

Subtract -33.25 times Row 2 from Row 3 ($R_3 \leftarrow R_3 - (-33.25)R_2$):

$$\begin{pmatrix} 1 & 2.25 & 0.5 & | & 2 \\ 0 & 1 & 0.823529 & | & 0.313725 \\ 0 & 0 & 37.88235 & | & 2.431373 \end{pmatrix}$$

Step 2.5: Make the leading coefficient of Row 3 equal to 1.

Divide Row 3 by 37.88235 ($R_3 \leftarrow R_3/37.88235$):

$$\begin{pmatrix} 1 & 2.25 & 0.5 & | & 2 \\ 0 & 1 & 0.823529 & | & 0.313725 \\ 0 & 0 & 1 & | & 0.064182 \end{pmatrix}$$

The matrix is now in row echelon form (upper triangular).

3. Back Substitution:

Convert the row echelon form back into a system of equations and solve from the bottom up.

- From the third row:

$z = 0.064182$

- From the second row:

$y + 0.823529z = 0.313725$

Substitute z:

$y + 0.823529(0.064182) = 0.313725$

$y + 0.052848 \approx 0.313725$

$y = 0.313725 - 0.052848 \approx 0.260877$ (rounded to 0.26087 in your table)

- From the first row:

$x + 2.25y + 0.5z = 2$

Substitute y and z:

$x + 2.25(0.26087) + 0.5(0.064182) = 2$

$x + 0.5869575 + 0.032091 = 2$

$x + 0.6190485 = 2$

$x = 2 - 0.6190485 \approx 1.3809515$ (rounded to 1.380952 in your table)

Solution:

The solution to the system of equations is approximately:

$x = 1.380952$

$y = 0.26087$

$z = 0.064182$

E.5 - Gauss-Jordan

This example uses the following equations:

Equation 1: $4x + 8y - 4z = 29$

Equation 2: $7x - 9y + 23z = 20$

Equation 3: $24x - 2y + 12z = 32$

Step-by-step Solution:

1. Form the Augmented Matrix:

   The system of equations is first represented as an augmented matrix $[A|B]$:

$$\begin{pmatrix} 4 & 8 & -4 & | & 29 \\ 7 & -9 & 23 & | & 20 \\ 24 & -2 & 12 & | & 32 \end{pmatrix}$$

2. Transformation to Reduced Row Echelon Form (RREF):

   The core of Gauss-Jordan is to transform the augmented matrix into an identity matrix on the left side (diagonal '1's, all other elements '0's), so the solution can be read directly from the right-hand side. This is achieved by creating '1's on the diagonal and then '0's in the rest of each column.

   - Step 2.1: Process Column 1 (for x).
     - Make the (1,1) element 1: Divide Row 1 by 4 ($R_1 \leftarrow R_1/4$):

$$\begin{pmatrix} 1 & 2 & -1 & | & 7.25 \\ 7 & -9 & 23 & | & 20 \\ 24 & -2 & 12 & | & 32 \end{pmatrix}$$

     - Eliminate elements below (1,1):

$R_2 \leftarrow R_2 - 7R_1$

$R_3 \leftarrow R_3 - 24R_1$

$$\begin{pmatrix} 1 & 2 & -1 & | & 7.25 \\ 0 & -23 & 30 & | & -30.75 \\ 0 & -50 & 36 & | & -142 \end{pmatrix}$$

- Step 2.2: Process Column 2 (for y).
  - Make the (2,2) element 1: Divide Row 2 by -23 ($R_2 \leftarrow R_2/-23$):

$$\begin{pmatrix} 1 & 2 & -1 & | & 7.25 \\ 0 & 1 & -1.30435 & | & 1.336957 \\ 0 & -50 & 36 & | & -142 \end{pmatrix}$$

  - Eliminate elements above and below (2,2):

$R_1 \leftarrow R_1 - 2R_2$

$R_3 \leftarrow R_3 - (-50)R_2$ (i.e., $R_3 + 50R_2$)

$$\begin{pmatrix} 1 & 0 & 1.608696 & | & 4.576087 \\ 0 & 1 & -1.30435 & | & 1.336957 \\ 0 & 0 & -29.2174 & | & -75.1522 \end{pmatrix}$$

- Step 2.3: Process Column 3 (for z).
  - Make the (3,3) element 1: Divide Row 3 by -29.2174 ($R_3 \leftarrow R_3/-29.2174$):

$$\begin{pmatrix} 1 & 0 & 1.608696 & | & 4.576087 \\ 0 & 1 & -1.30435 & | & 1.336957 \\ 0 & 0 & 1 & | & 2.572173 \end{pmatrix}$$

  - Eliminate elements above (3,3):

$R_1 \leftarrow R_1 - 1.608696R_3$

$R_2 \leftarrow R_2 - (-1.30435)R_3$ (i.e., $R_2 + 1.30435R_3$)

$$\begin{pmatrix} 1 & 0 & 0 & | & 0.438244 \\ 0 & 1 & 0 & | & 4.691964 \\ 0 & 0 & 1 & | & 2.572173 \end{pmatrix}$$

The matrix is now in reduced row echelon form.

3. Direct Solution:

With the matrix in RREF, the solution for x, y, and z can be read directly from the rightmost column:

$x = 0.438244$

$y = 4.691964$

$z = 2.572173$

Solution:

The solution to the system of equations is approximately:

$x = 0.438244$

$y = 4.691964$

$z = 2.572173$

Appendix F: Cronbach's Alpha Test

The calculation for the Cronbach's Alpha on the questionnaire can be found here. This will prove that the survey's selected questions are valid.

LEGEND:

This legend is for Table F.1.

COLUMNS	DESCRIPTION
ID	Participant ID
Q1	Ease of Input (1-5)
Q2	Ease of Method Selection (1-5)
Q3	Calculation Speed (1-5)
Q4	Clarity of Steps (1-5)
Q5	Overall Experience (1-5)
Sums	Sum value of all answers, indicated by the formula: $\sum x_i$

Table F.1.

Cronbach's Alpha Test

ID	Q1	Q2	Q3	Q4	Q5	Sums
1	4	4	3	4	4	19
2	5	5	5	5	5	25
3	5	5	5	4	4	23
4	5	5	5	4	5	24
5	5	5	3	4	5	22
6	4	5	3	3	4	19
7	5	4	4	3	5	21
8	4	4	4	5	4	21
9	5	5	4	3	4	21
10	5	5	5	5	5	25
11	5	5	4	3	5	22
12	5	5	5	5	5	25
13	4	4	4	5	5	22
14	5	5	4	4	4	22
15	5	5	4	5	5	24
16	5	5	4	4	5	23
17	5	5	4	4	4	22
18	5	5	5	5	5	25
19	4	4	3	4	4	19
20	3	4	4	5	4	20
21	5	5	4	4	5	23
22	4	4	3	3	4	18
23	4	5	4	3	5	21
						σ^2_{total} =4.454545
Variances	σ^2_1 =0.339921	σ^2_2 =0.221344	σ^2_3 =0.498024	σ^2_4 =0.628458	σ^2_5 =0.256917	$\sum \sigma^2_i$ =1.944664

The formula for Cronbach's Alpha is:

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\sum_{i=1}^{k} \sigma_i^2}{\sigma_{total}^2}\right)$$

There are 5 items, so k = 5.

Now, plug these values into the formula:

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\sum \sigma_i^2}{\sigma_{total}^2}\right)$$

$$\alpha = \frac{5}{5-1}\left(1 - \frac{1.944664}{4.454545}\right)$$

$$\alpha = \frac{5}{4}(1 - 0.4365538)$$

$$\alpha = 1.25 \times 0.5634462$$

$$\alpha \approx \mathbf{0.704308}$$

With a Cronbach's Alpha value of 0.704308, this shows that the questionnaires are valid and reliable.

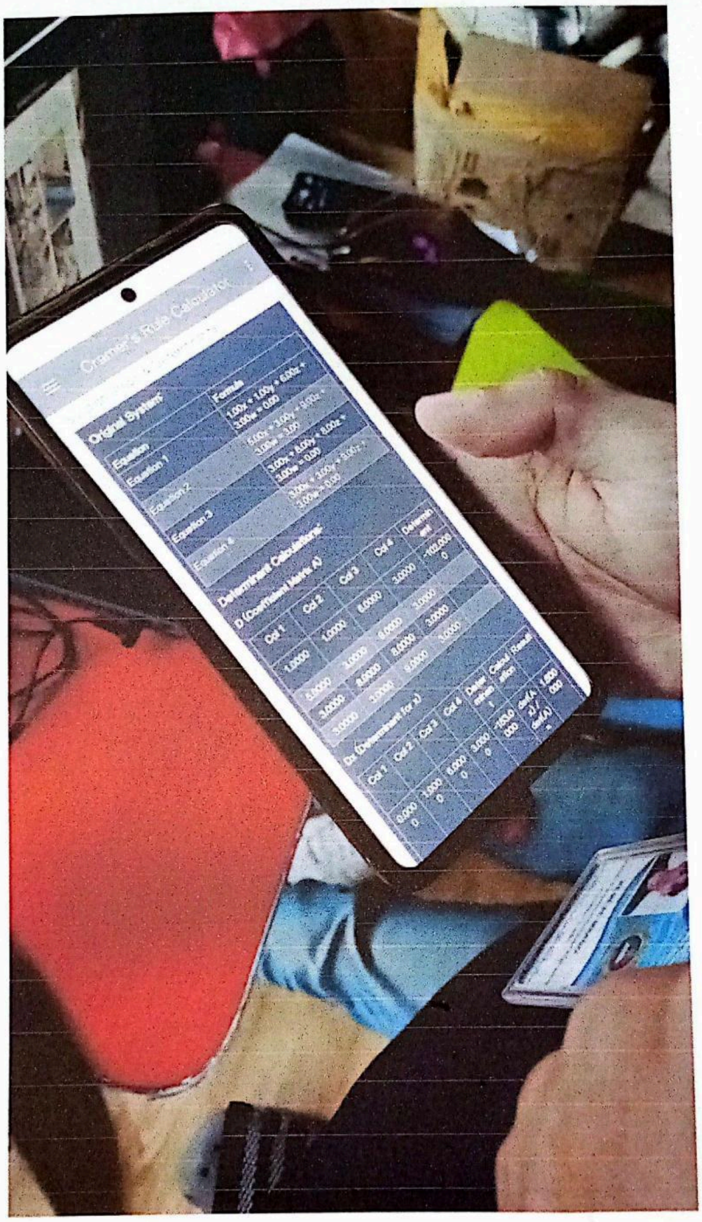


Figure 1. This figure provides a real-world context for the application's use, showing a user actively engaging with the "Step-by-Step Numerical Methods Calculator" on a standard mobile device. The screen displays the detailed output for solving a complex 4x4 system of linear equations using Cramer's Rule.

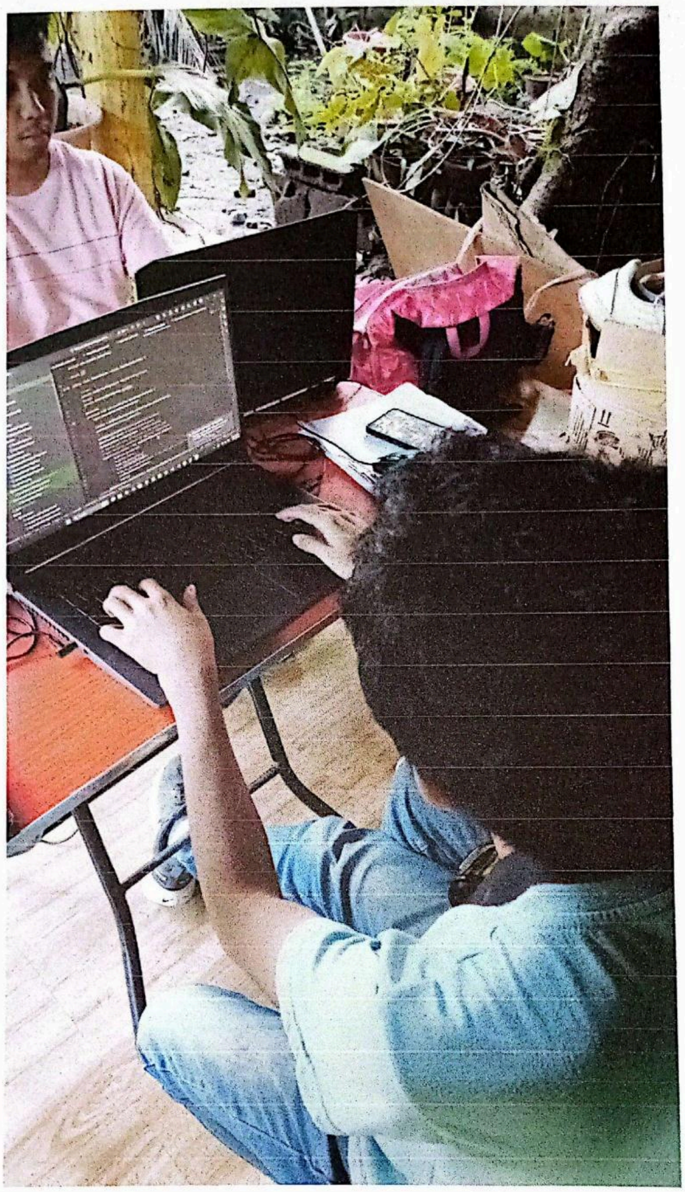


Figure 2. This snapshot takes place during the active development phase of the project. The researchers are in the process of coding the app, and the source code is displayed within an Integrated Development Environment (IDE) on the laptop screen. The image records the critical implementation phase where the theoretical ideas of numerical methods and the user interface designs were converted into the operational mobile software. The collaborative environment emphasizes the iterative nature of coding, testing, and iteration that was necessary in developing a strong and user-friendly learning tool.

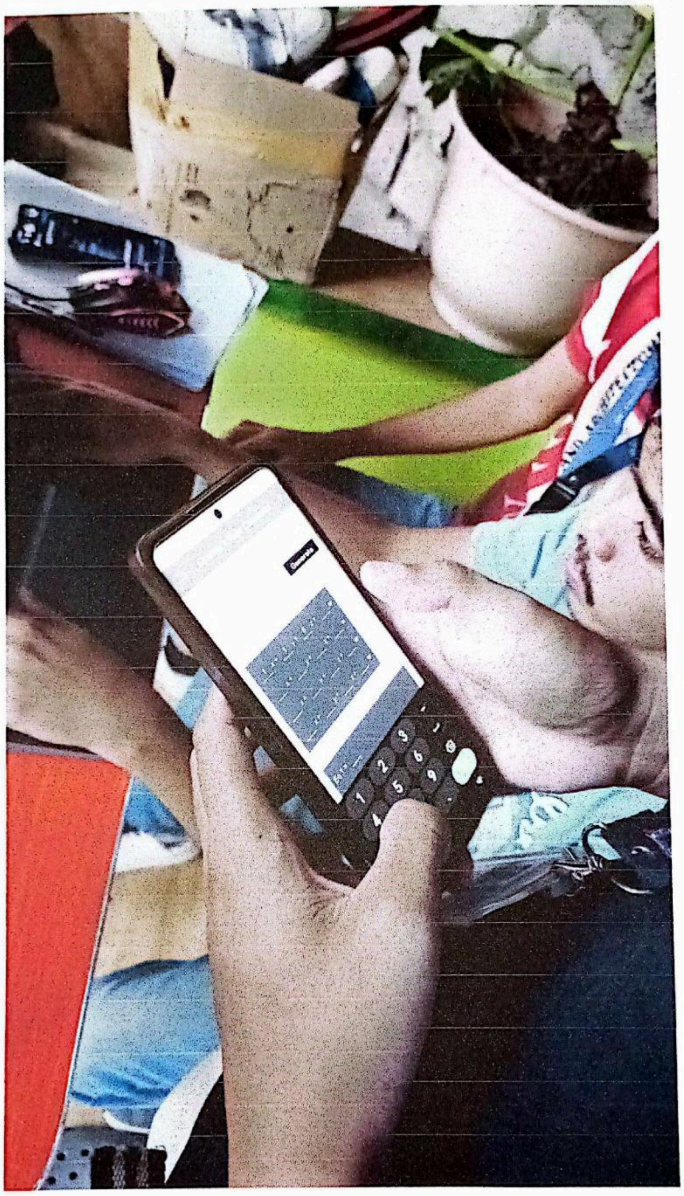


Figure 3. This figure captures a user in the process of entering data into the "Step-by-Step Numerical Methods Calculator." The screen shows the matrix input interface, where the user has already defined the size of the system and is now populating the coefficient matrix and constants for the equations.

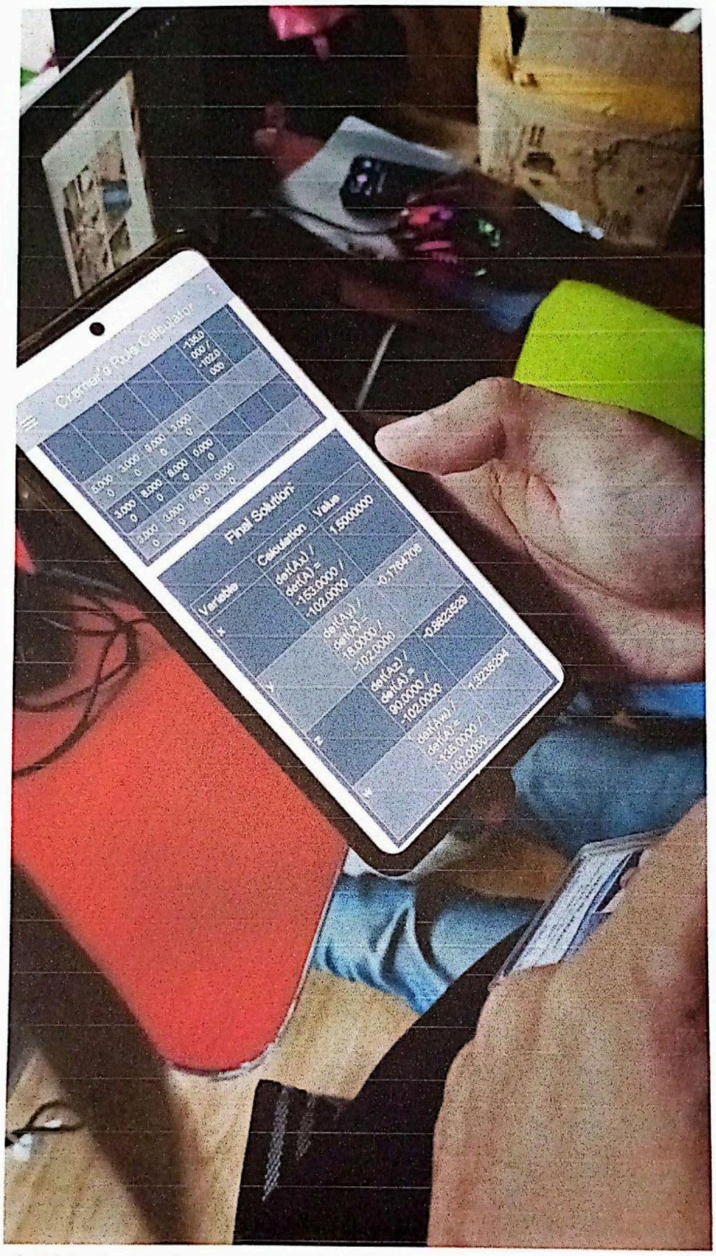


Figure 4. This figure demonstrates the final output of the "Step-by-Step Numerical Methods Calculator," as viewed by a user on a smartphone. The screen displays the "Final Solution" for a 4x4 system of equations solved using Cramer's Rule. This summary is presented in a highly organized table, located at the very end of the scrollable results page.

Figure 5. This image presents another perspective of the development process, emphasizing the teamwork involved in the project. The photograph depicts the research team sitting together in a common room, all working together to develop the application. The researchers are all actively working on their laptops, pointing at screens and debating the work, which shows the coding, debugging, and testing process iteratively. This collaborative setting was necessary for trouble-shooting and making the final application robust enough and user-friendly. The scene captures the hands-on endeavor and collaboration behind the entire implementation process.

Reports, mimeos, monographs, books, etc.

Chapter III
METHODOLOGY

This chapter discusses the methodology used in this mini research project. It begins by describing the research design, the development tool used in developing the app, data collection method, and analytical techniques.

3.1 Research Design

This mini research project employed a mixed-methods evaluation design situated within a development research framework. This methodological approach was carefully chosen as it is suited for research that involves both the creation of the app and the investigation of its effectiveness. The design was selected because it is a practical way to solve a specific real-world problem, in this case, the need for a more accessible and user-friendly educational tool for learning numerical methods. The entire mini research project process was organized into two phases: a Development Phase and Evaluation Phase.

The first phase, Development, followed established rules from both software development and educational design to make sure the app was both well-built and an effective learning tool. The main goal was to turn math algorithms and design plans into a real working app that engineering students could test. This phase wasn't just about coding, it was a creative phase where the researchers' knowledge about numerical methods shaped the design of the application. It began with a comprehensive analysis of the 5 mathematical algorithms to ensure computational correctness. Followed by a user-centered design process for the User Interface (UI) and User Experience (UX), following the material design principles to make the application easy to figure out without instructions and easier to learn. By the end of this phase, the researchers had built one solid, working version of the application that was ready for the user to test.

Following the Development Phase, the Evaluation Phase comes next, which is composed of formal proof-of-concept and usability study. The primary objective of this phase was to move from creation to assessment. This stage was important for gathering data on the application's performance in a controlled setting with its intended end-users. This evaluation was designed to measure three specific criteria:

1. **Functional Efficacy**: Determining whether the application operates smoothly and produces accurate results.
2. **User Experience**: Assessing the efficiency and satisfaction with which users interact with the application, measured through quantitative ratings.
3. **Educative Utility**: Investigating the perceived value of the step-by-step solution feature as a learning aid, capturing user perspectives on its clarity and educational benefit.

3.2 Development Tool

The following is the development tool used in the mini research project:

1. Android Studio

Android Studio is a powerful Integrated Development Environment (IDE) for developing an Android app. This is Google's official IDE. Additionally, it offers lots of tools for coding, debugging, testing, and performance analysis. It supports Java, Kotlin, and C++, and includes an emulator, code editor, and layout editor tailored for Android development (Sandu,2025).

3.3 Application Development Process

The development of the app followed a modified methodology, emphasizing iterative development and testing. The process was organized into the following distinct phases:

1. **Planning and Analysis:** This initial phase involved the foundational research for the project. The mathematical frameworks and specific algorithmic logic for the five numerical methods were thoroughly researched and documented.
2. **Design and Prototyping:** With a clear understanding of the requirements, the focus shifted to the user-facing design. The user interface (UI) and user experience (UX) were designed to be intuitive and mobile-friendly, adhering to Google's Material Design principles. Layouts were created in XML with a design consideration being the dynamic generation of input fields to accommodate systems of varying sizes.
3. **Implementation:** In this phase, the designs and algorithms were translated into a functional application. Each numerical method was implemented as a separate, self-contained module in Java, allowing for clear separation of logic. A significant focus was placed on building the step-by-step visualization feature, which programmatically generates visual outputs to serve as a primary learning aid.
4. **Internal Testing and Refinement:** Before user evaluation, the application underwent an internal testing process to ensure its stability, accuracy, and performance. This process included:
   - **Unit Testing:** Each algorithm was tested against predefined problems from numerical methods textbooks to verify the correctness of the solutions.
   - **Performance and Scalability Testing:** The application was systematically tested to determine its practical performance limits. This involved creating and solving systems of varying sizes, from 1 to 10 unknowns. Through this testing, it was determined that while the algorithms remained accurate, systems larger than 10x10 introduced noticeable computational lag on standard mobile hardware. Based on these findings, the application's input was intentionally

3.4 Data Collection Method

To evaluate the app's performance and effectiveness, a mixed-methods approach was employed, combining quantitative metrics from a questionnaire with qualitative data from open-ended questions and direct observation. The data was collected through a structured, task-based user testing session.

3.4.1 Instrument of Data Collection

The following instruments were prepared to ensure a consistent and systematic data collection process:

1. **Standardized Task Scenario:** A handout detailing a predefined 4x4 system of linear equations (see Appendix A.2). Using a standardized linear equation ensured that all participants performed the identical core task, making their performance and feedback comparable.
2. **Online Post-Task Questionnaire:** A digital survey created using an online form tool, in this case the researcher uses Google Form. This instrument was designed to capture user insights immediately following the task . It consisted of two parts, Part 1 is the Quantitative Metrics, it contained five questions rated on a 5-point Likert scale to measure ease of inputting linear equations, ease of method selection, speed of the app in calculating the solution, clarity of the solution, and overall experience. Part 2 is the Qualitative Feedback, it consists of two open-ended questions prompting participants to describe the app's best features and its most confusing aspects and their suggestions to make the app better.

3.4.2 Data Collection Procedure

Each testing session was conducted fully remote and followed these sequential steps:

1. **Recruitment and Distribution:** Participants were contacted online. They were sent a package containing the application APK file, a document with clear instructions, standardized linear equations to solve, and a link to the Online Post-Task Questionnaire.
2. **Task Execution:** Participants installed the application on their own Android devices and performed the assigned task at their own convenience.
3. **Data Submission:** After completing the task, participants followed the link provided and submitted their feedback through the online questionnaire.

3.5 Evaluation Criteria

The success of the application was evaluated against a set of predefined criteria. These criteria were measured using the quantitative and qualitative data collected from the remote online questionnaire.

1. **Functionality and Performance:** This assessed the reliability and speed of the application. It was measured by:
   - **Task Completion Rate:** Gathered from the 100% questionnaire submission rate, which indicated that all participants were able to install and use the app to generate solutions.
   - **Calculation Speed**: Measured by the mean score of the Likert-scale question, "How would you rate the speed of the app in calculating the solution?"

2. **Usability and User Experience (UX):** This assessed the intuitive, direct to the point, and satisfying the application was to use. It was evaluated quantitatively through the mean scores of three specific Likert-scale items:
   - **Ease of data input** ("How easy was it to enter the linear equations into the app?")
   - **Ease of selecting method** ("How easy was it to find and select the solving method?")
   - **Overall satisfaction** ("Overall, how would you rate your experience using this app?")
3. **Clarity and Educational Value:** This assessed the effectiveness of the application's primary educational feature, the step-by-step solution. The goal was to determine if this feature successfully enhanced user understanding numerical methods. It was measured by:
   - **Clarity of Presentation of the Solutions:** The quantitative mean score for the question, "How clear and understandable were the step-by-step results shown by the app?"
   - **Qualitative User Feedback:** An analysis of the open-ended responses, specifically looking for comments related to the clarity, usefulness, confusion of the solution steps, and suggestions.

CHAPTER IV
DESIGN AND IMPLEMENTATION

This chapter includes the figures and explanation of how the "Numerical Methods Solver" application was created. It covers the foundation of the application and the software tools used in its creation. In the development of this application there are two key objectives: first, to build a reliable tool that correctly solves systems of linear equations, and second, to design a user-friendly interface that clearly shows the solution steps. This chapter outlines the technical work that turned the project's concept into a functional application.

4.1 System Architecture

The architecture follows a modular MVC (Model-View-Controller) design pattern, which is:

1. **Model**: Contains all the mathematical algorithms for matrix operations and for each numerical method.
2. **View**: Comprises user interface layouts using XML to support dynamic input, solution display, and step-by-step walkthrough.
3. **Controller**: Manages user inputs, controls flow between modules, and triggers appropriate algorithms based on user selection

In Figure 4.1 it is shown the flow of how the app works. It follows a simple and step-by-step process. When the app is opened, the user will be looking at the interface which is the 'View.' The View tells the 'Controller' what to do. The Controller then tells the 'Model' to work with the data. After that, the Controller gets the answer from the Model and shows it on the user's interface. The app was built this way to keep everything organized.

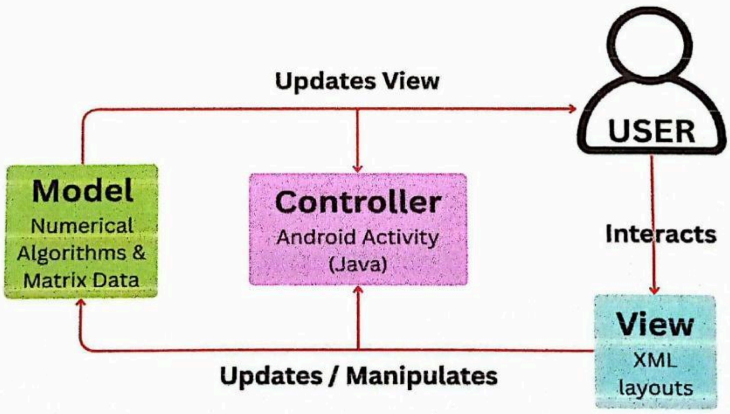


Figure 4.1. Diagram of the MVC Structure. (Source: The authors).

4.3 Development Tools and Technologies

The application was built for the Android platform using a carefully selected suite of modern tools to ensure both reliability and broad device compatibility. At the heart of this toolkit was Android Studio (Electric Eel), chosen for its status as a stable and feature-rich official release from Google (Android Developers, 2023). A complete list of these key development components is detailed in Table 4.1.

Table 4.1.

Development Tools and Specifications.

Category	Specification	Purpose
IDE	Android Studio (Electric Eel or latest version)	The official program for creating Android apps.
Programming Languages	Java, XML	Java was used for application logic. XML was used for designing the user interface.
Target SDK	Android API 24 and above	Ensures the app runs on a wide variety of modern Android devices.
Design Principles	Material Design	A Google design standard used to create a clean and intuitive look and feel.
Supported Devices	Android smartphones and tablets	The app's layout adjusts to fit different screen sizes.

The selection of this technology stack provided a solid foundation for the project. Utilizing these standard and well-supported tools allowed the development to focus on the application's unique features, such as the implementation of the numerical algorithms and the design of the user interface, which are detailed in the following sections.

4.4 User Interface (UI) and User Experience (UX) Design

The design of the user interface targets simplicity and educational clarity. The goal was to create an interface so simple that users could focus their cognitive energy on the mathematical problem, not on learning how to use the app. The user journey was designed to be a logical and linear progression from problem setup to solution analysis.

The first screen the user encounters is for method selection. As shown in Figure 4.2, it presents a clean, list-based menu of the five numerical methods. This direct approach allows the user to immediately select their desired solving technique.

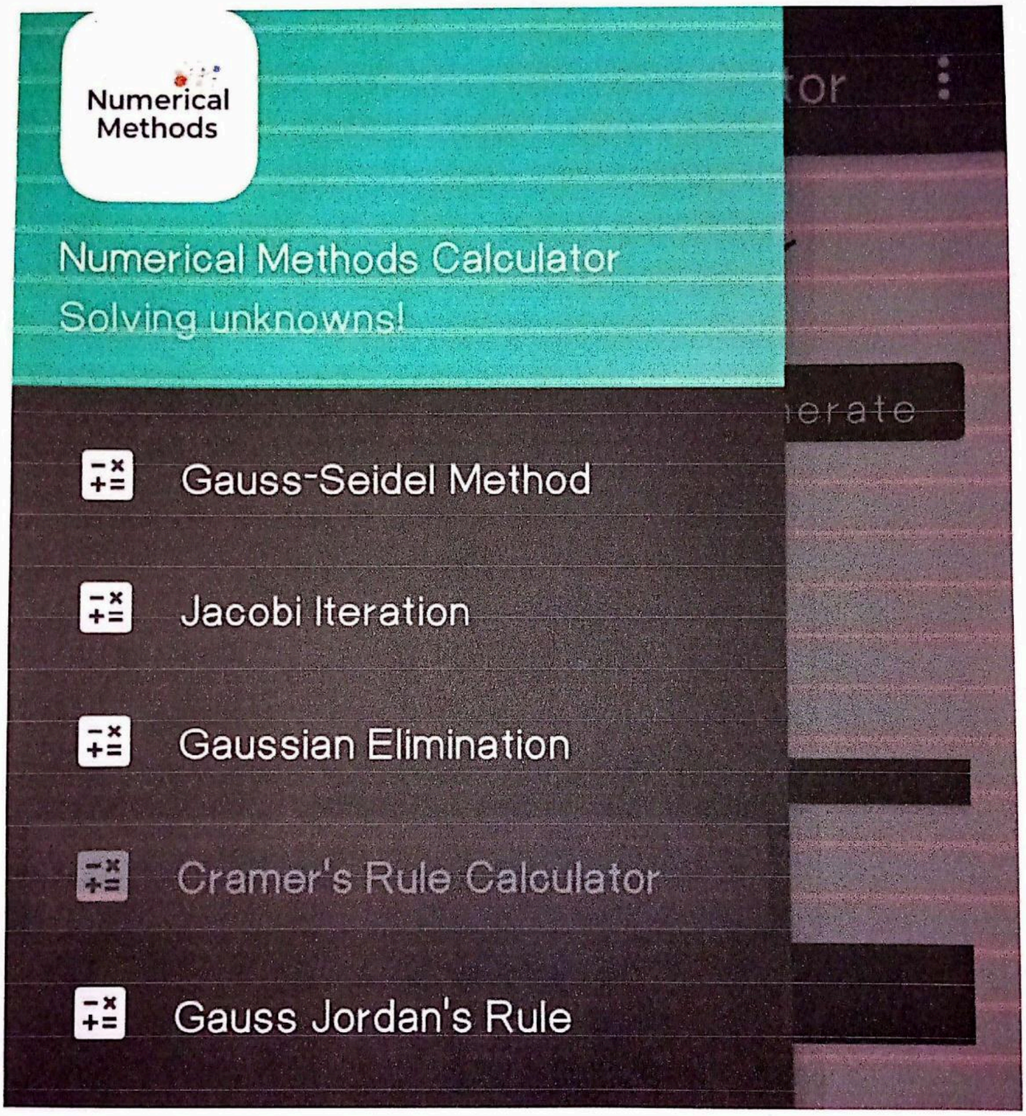


Figure 4.2. The main menu screen for method selection. (Source: The authors).

Once a method is chosen, the user is taken to the data input screen. A key usability feature is the dynamic generation of the input grid. After the user specifies the number of variables, the interface automatically renders a correctly sized matrix, as seen in the example in Figure 4.3. This helps prevent errors during problem setup.

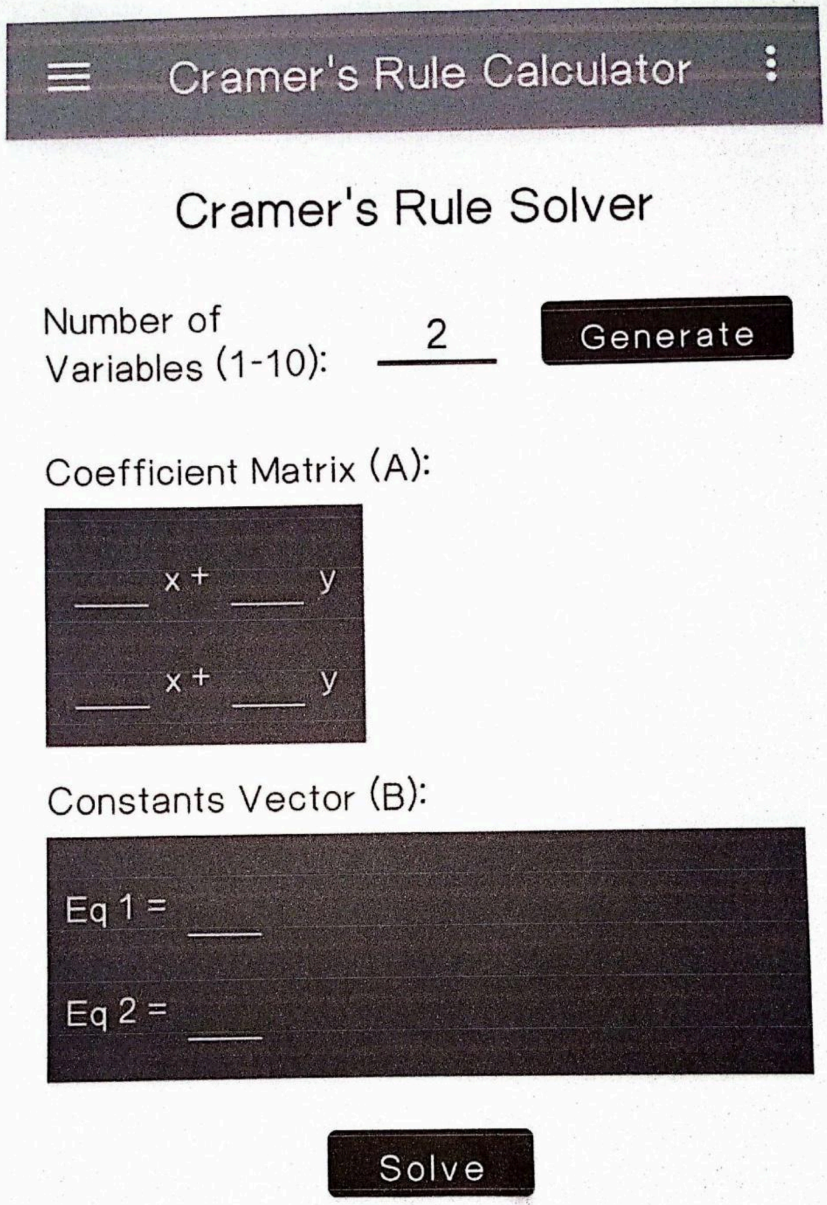


Figure 4.3. The data input screen with a dynamically generated grid. (Source: The authors).

The heart of the app is the solution screen as shown in Figure 4.4. We designed it to be a simple, scrollable table that breaks down each major step of the calculation, so you can easily follow along.

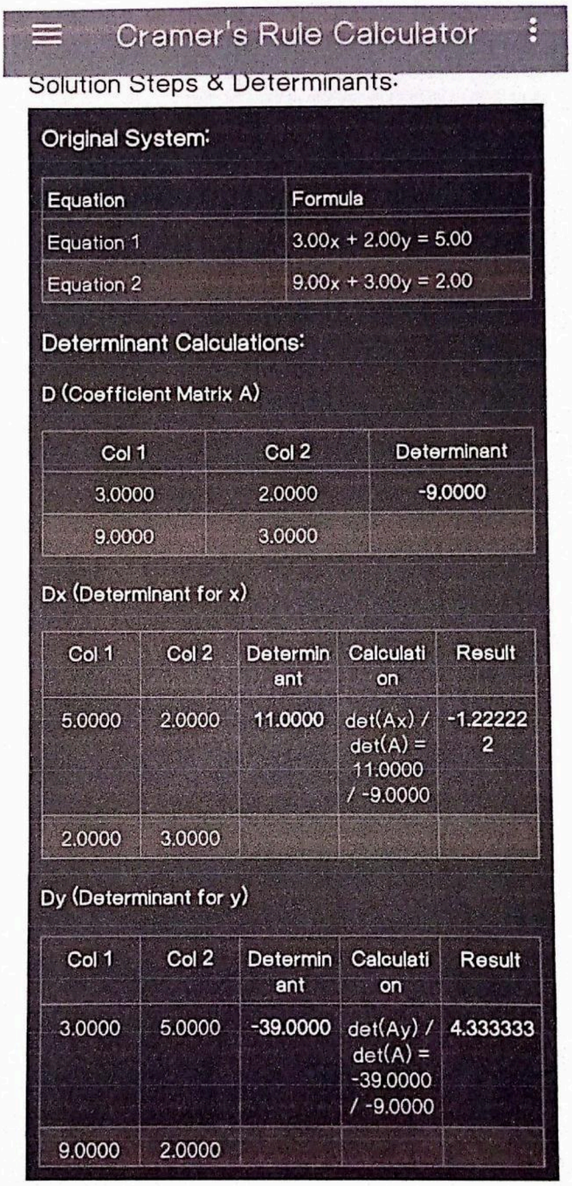


Figure 4.4. The results screen shows a detailed, step-by-step solution. (Source: The authors).

Finally, to provide a clear conclusion, the application presents a summary of the final answer. This screen, shown in Figure 4.5, isolates the calculated values for each of the unknown variables, allowing the user to see the results of the computation at a glance without any extra information.

Final Solution:

Variable	Calculation	Value
x	det(Ax) / det(A) = 11.0000 / -9.0000	-1.2222222
y	det(Ay) / det(A) = -39.0000 / -9.0000	4.3333333

Figure 4.5. The final answer display screen, showing the values for each unknown. (Source: The authors).

4.5 Error Handling and Input

To ensure accurate and user-friendly experience, the application has an error-handling system. This system was designed to maintain accuracy and avoid user confusion by anticipating both user input mistakes and potential computational issues. The error handling is divided into two primary categories, each with its own user feedback mechanism.

The first layer of error handling addresses user input directly. To ensure that the calculation has no failures, all input fields won't accept non-numeric or empty values. This preventive measure is communicated to the user through a notifier as illustrated in Figure 4.6, which prompts for a correction.

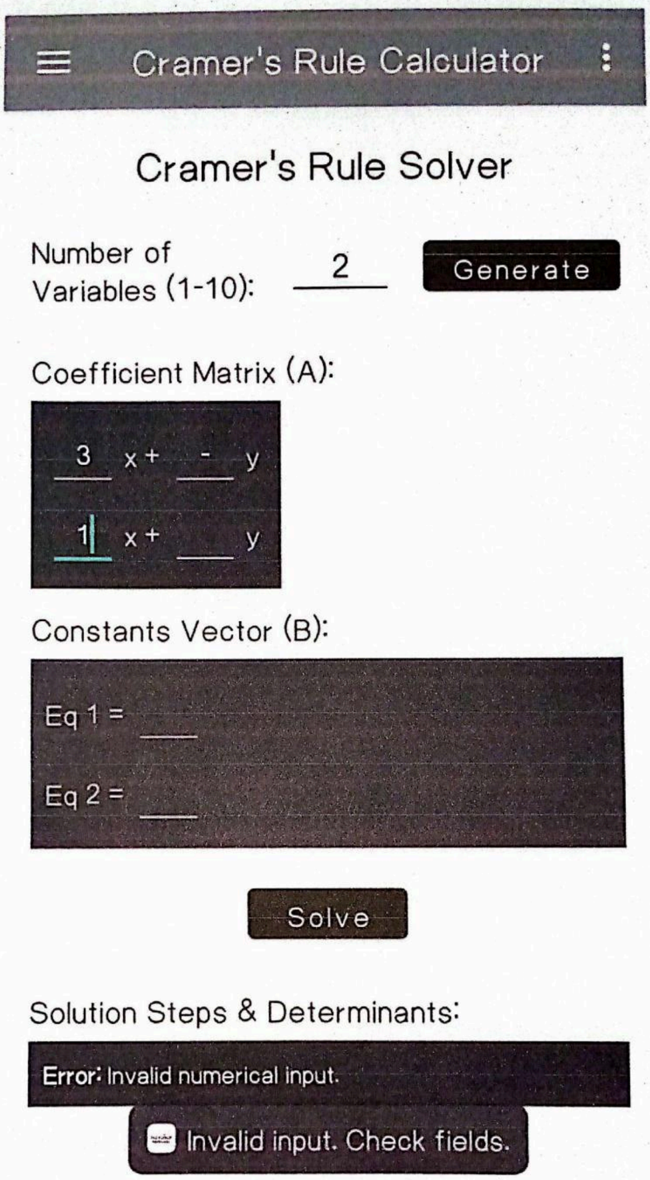


Figure 4.6. An example of an input validation error message for an empty input field or non-numeric input. (Source: The authors).

The app is also designed to manage difficult calculations on its own. It works by repeating the calculation until the result is stable.

Though in such cases where the solution is infinite, the app could get stuck in an endless loop or freezes. To prevent this from happening, the app has a built- in function where it will stop after a certain number of attempts. Thus, it will clearly notify you that the process failed as indicated in Figure 4.7.

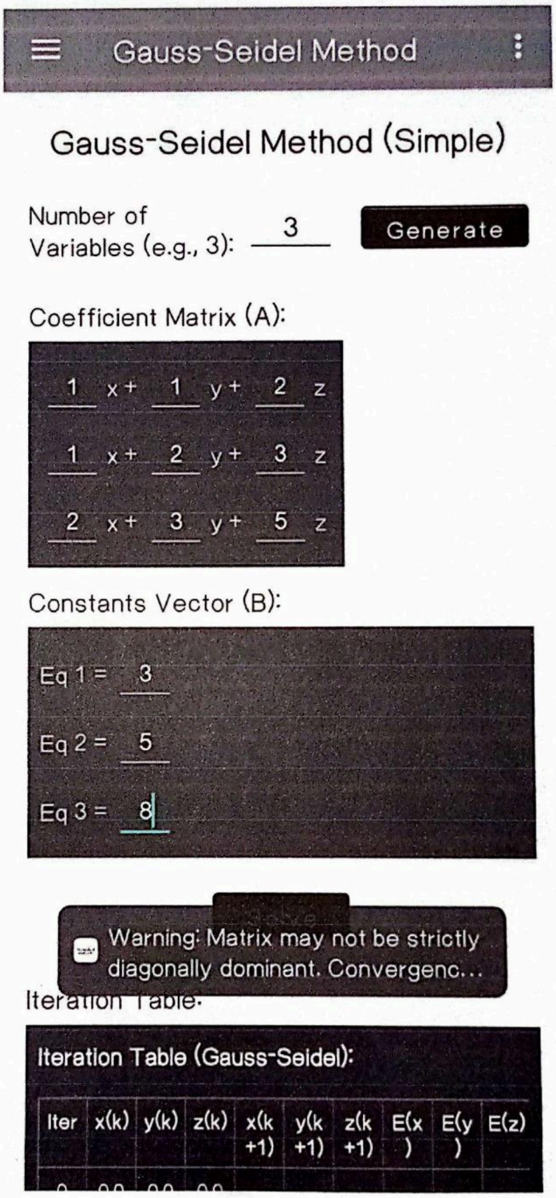
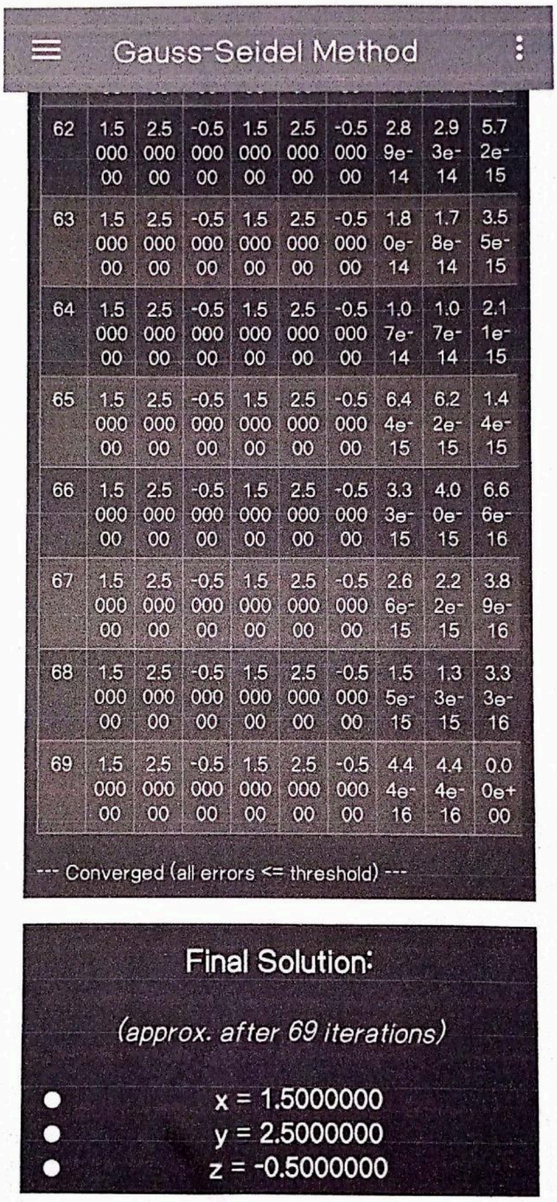


Figure 4.7. A notification message indicating that the iterative method failed to converge. (Source: The authors).

Following this notification, the application then displays the last set of calculated values. This screen, shown in Figure 4.8, presents the user with the best available approximate answer at the point where the process was halted, allowing them to still have a usable result from the computation.



62	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	2.89e-14	2.93e-14	5.72e-15
63	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	1.80e-14	1.78e-14	3.55e-15
64	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	1.07e-14	1.07e-14	2.11e-15
65	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	6.44e-15	6.22e-15	1.44e-15
66	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	3.33e-15	4.00e-15	6.66e-16
67	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	2.66e-15	2.22e-15	3.89e-16
68	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	1.55e-15	1.33e-15	3.33e-16
69	1.500000	2.500000	-0.500000	1.500000	2.500000	-0.500000	4.44e-16	4.44e-16	0.00e+00

Figure 4.8. The display of the final approximate answer after a convergence failure. (Source: The authors).

Chapter V
RESULTS AND DISCUSSION

This chapter presents a detailed analysis of the data collected during the user evaluation phase of the Numerical Methods Calculator. The findings are systematically organized into quantitative results derived from scaled-questionnaire responses and a qualitative thematic analysis of the open-ended feedback provided by the participants.

5.1. Results

The evaluation involved a sample of 23 engineering students from NORSU. Each participant was tasked with solving a predefined 4x4 system of linear equations to assess the application's functional efficacy and usability in a controlled environment.

The results of the task-based assessment were uniformly successful. All 23 participants (a 100% completion rate) were able to independently navigate the application, input the required coefficients, execute a solving method, and arrive at the correct solution. This outcome provides strong evidence of the application's high functional reliability and validates its baseline usability for the intended target audience.

Participants rated key aspects of the application on a 5-point Likert scale, where 1 means the user strongly disagrees, and 5 means the user strongly agrees with the question (ResearchGate, 2016). This is observed in Table 5.1.

Table 5.1

Likert Scale Interpretation Table

Standard Interpretation	*Value*	*Range*
Strongly Disagree	*1*	*1.00 - 1.80*
Disagree	*2*	*1.81-2.60*
Neither/Nor Agree	*3*	*2.61-3.40*
Agree	*4*	*3.41-4.20*
Strongly Agree	*5*	*4.21-5.00*

The mean scores for each evaluation criterion are presented in Table 5.2, providing a quantitative measure of user perception, with the interpretation based on Table 5.1.

Table 5.2

Mean Scores from User Usability Questionnaire (n=23)

The mean score is calculated by the formula: sum of all numbers / amount of numbers

Evaluation Criterion	Mean Score (M)	Standard Interpretation
Ease of entering equations	4.61	*Strongly Agree*
Ease of selecting a method	4.70	*Strongly Agree*
Speed of calculation	4.04	*Agree*
Clarity of step-by-step results	4.09	*Agree*
Overall user experience	4.57	*Strongly Agree*

The quantitative data indicate a highly favorable user reception. The application's highest-rated attributes were **Ease of entering equations** (M=4.61) and **Overall user experience** (M=4.57). These metrics suggest that the primary user interface for problem input is exceptionally intuitive and that the application provides a highly satisfactory user journey.

Furthermore, the core educational feature, **Clarity of step-by-step results**, received a strong mean score of 4.09, affirming its pedagogical effectiveness. The criterion for **Speed of calculation** (M=4.04), while still positive, received the most variance in responses. This suggests that while the application's performance is generally considered efficient, there is a perceptible processing time for some users, likely associated with the rendering of detailed step-by-step solutions.

A **Cronbach's Alpha** test was conducted. It checks if multiple questions produce similar, consistent scores from respondents (George & Mallery, 2019). Table 5.3 is used as a basis for computing Cronbach's Alpha in Appendix F.

Table 5.3

Cronbach's Alpha Basis Table

Coefficient of Cronbach's Alpha	Reliability Level
More than 0.90	Excellent
0.80 - 0.89	Good
0.70 - 0.79	Acceptable

Coefficient of Cronbach's Alpha	Reliability Level
0.60 - 0.69	Questionable
0.50 - 0.59	Poor
Less than 0.59	Unacceptable

The test yielded a value of **0.704**, making it acceptable. The calculation can be seen in Appendix F. This shows that the survey questions are a good reliable fit to the mini research project.

A thematic analysis of the open-ended responses was conducted to provide deeper insight into the user experience and to identify specific strengths and areas for future development.

Three primary themes emerged from the positive feedback:

1. **Usability and Accessibility:** The most prevalent theme was the application's simplicity and intuitive design. Participants frequently reported that the app was "Easy to use" and "User Friendly," validating the user-centered design approach.
2. **Computational Efficiency:** Users expressed significant appreciation for the speed at which solutions were generated compared to manual alternatives. Comments such as "Fast and reliable" and "Direct result" were indicative of this theme.
3. **Functional Breadth:** The application's core value proposition was affirmed by users who praised the availability of "Different methods to use" and the capacity to "solve large scales of linear equations," a function not available on standard scientific calculators.

Constructive feedback provided a clear framework for future iterations, centering on three areas:

1. **User Interface (UI) and Information Presentation:** The most common critique concerned the presentation of results. Participants reported that the output created a high cognitive load, with comments such as "Too much data compiled, hard to read" and "Need to scroll a lot." This feedback suggests that while the information is present, its architecture could be optimized to improve readability and reduce scrolling fatigue.
2. **Feature Expansion and Integration:** Users demonstrated high engagement by suggesting advanced features. The desire to "Make it available on IOS" points to a need for cross-platform support. The suggestion to "export to something like Microsoft or something" indicates a user's desire to integrate the application into their broader academic workflow.
3. **Pedagogical Clarity:** A minority of users noted that "some solutions are confusing," suggesting that while the computational steps are displayed, the underlying mathematical reasoning is not always explicit. This highlights an opportunity to enhance the app's pedagogical value through further instructional scaffolding.

The researchers tested the limits of the app's ability on handling large scale linear systems. As seen in Table 5.4, the time required to solve a linear system increases as the number of unknowns increases, until the app crashes from a memory overflow at 11 unknowns.

Table 5.4

Unknown Count Performance Analysis

Number of unknowns	Time required to solve the equations (s)
1	0.2
2	0.2
3	0.2
4	0.3
5	1.0
6	1.3
7	1.8
8	3.1
9	3.6
10	5.2
11	N/A (The app stops responding)

5.2. Discussion

1. *Can a mobile app be created in Android Studio that correctly solves linear systems with up to 10 variables, using all five of the required numerical methods?*

   The initial research question inquired whether or not it was possible to develop a mobile application in Android Studio that would properly solve systems of up to 10 variables through five numerical techniques. The research presents an affirmative "yes." The presence of the operational application, which was created in Android Studio as described in Chapter 4, constitutes the first evidence. The fact that the application is 100% complete on task execution during user testing confirms its functional reliability. In addition, the internal performance analysis (Table 5.4) categorically indicates the app's ability to solve systems up to 10 unknowns before performance constraints are met, thereby confirming it meets the given scale requirements.

2. *When the app is introduced to engineering students, will they find it simple and straightforward? Is it easy to enter the numbers, pick a method, and get a solution?*

The second question examined if engineering students would find the app simple and easy to use. The quantitative as well as the qualitative data strongly validate this. With extremely high mean values for Ease of entering equations (M=4.61) and Ease of selecting a method (M=4.70), it is evident that the essential user journey is easy for the target group. The "Strongly Agree" score on overall experience (M=4.57) and frequency of qualitative responses such as "Easy to use" and "User Friendly" confirm that the app is successful in not introducing a steep learning curve, thus enabling students to concentrate on the problem and not the tool.

3. *Does the "step-by-step" feature really make it clearer for students to understand how the answer was found?*

The third research question investigated the step-by-step feature's clarity and educational value. Here, the results offer a telling insight: there is a clear dichotomy between the application's computational backend and presentation frontend. Whereas the backend, governing mathematical correctness and efficiency, was greatly admired, the frontend—tasked with presenting the elaborate solutions—was pinpointed as the main area in need of dramatic improvement. The value of the feature is substantiated by a positive mean score for Clarity of step-by-step results (M=4.09). A major flaw, though, emerges from user comments like "Too much data compiled, hard to read" and "Need to scroll a lot." The present linear presentation of all steps can create high cognitive load, impairing partly the objective of making complicated processes easier. Thus, while the idea of the step-by-step function is proven, its implementation today is not maximally transparent and needs to be improved so that it can serve to enhance student understanding.

4. *Based on what the students say, what are the best parts of the app, and what needs to be improved in the future?*

The last question asked the users to name the app's most significant strengths and areas for future enhancement based on their experience. The thematic analysis further sheds light on this. The app's most striking strengths are its Usability and Accessibility (easiness), Computational Efficiency (speediness), and Functional Breadth (variety of approaches for large systems). The requests for an iOS version and data export features are most telling; they indicate that users recognize the application not as a novelty but as a credible utility of useful, long-term value to their scholarly pursuits. The most important need for improvement is the User Interface and Information Presentation, namely refining the results presentation to minimize cognitive load, as covered above.

Chapter VI
CONCLUSION AND RECOMMENDATIONS

This final chapter synthesizes the research findings, presents the study's conclusions, and articulates data-driven recommendations for future research and development.

6.1. Conclusion

The conclusions of this research point out the important findings and results derived through the research process. The conclusions are drawn based on data analysis and interpretation as well as the accomplishment of research objectives. The main conclusions of the research are:

1. Effective Development and Achievement of Research Objectives

   The study effectively chronicles the design, development, and empirical validation of the "Step-by-Step Numerical Methods Calculator." Achieving the core objective to create an Android application that not just solves huge linear systems but also explains the process in a clear step-by-step manner was achieved, rendering the project a highly successful and effective proof-of-concept.

2. Verification of the Core Hypothesis

   The overall hypothesis of the research—that providing simple, step-by-step solutions would align computational capacity with conceptual understanding—was emphatically verified. The study demonstrated that having a step-by-step aspect enables a tool to calculate and teach at the same time, moving beyond mere answer-production to create authentic understanding.

3. High Level of Functional Dependability and User Acceptability

   The evaluation of the app with its target end-users revealed excellent functional reliability, evidenced by a 100% success rate in tasks delegated. Additionally, the app was discovered to be high in usability and gained an extremely positive take-up among users, meaning that it is highly acceptable for its target use.

4. Effectiveness in Addressing Long-Standing Educational Gaps

   The system offers a tangible solution to fundamental education problems by directly tackling the "black box" problem, wherein students are given answers through software but do not know the process. It also breaches the limitations of standard calculators, which are incapable of coping with the level of sophistication needed in engineering studies.

5. Immediate and Far-Reaching Local Impact

   At the local level, the project has an almost instant and wide-ranging effect on its own institution, Negros Oriental State University (NORSU). Designed as a direct reaction to students' hardships, it gives students an avenue to verify their work and learn from mistakes in the moment, as it gives teachers a modern tool to enrich their curriculum.

6. Scalability and National Relevance of the Solution

Outside of the immediate context, the study provides an exciting and replicable model for addressing STEM education problems on a national scale. The app demonstrates how a targeted, tech-driven project can democratize quality learning resources, offering a proven model for developing innovation to serve the needs of the national student body.

6.2. Recommendations

Based on the comprehensive analysis of user feedback, the following recommendations are proposed to guide future iterations of the application and related research:

1. **Optimize the Information Architecture of the Results Display:** The highest priority for future development should be the refinement of the user interface for displaying solutions. It is recommended to implement a more structured layout, utilizing techniques such as collapsible sections, improved typographic hierarchy, and color-coding to reduce cognitive load and enhance the legibility of the step-by-step process.
2. **Enhance UI/UX and Customization:** To address specific user requests and improve overall usability, the implementation of a "dark mode" is recommended. Further customization options, such as theme selection, would also contribute to a more positive and personalized user experience.
3. **Expand Core Functionality and Workflow Integration:** To increase the application's utility, future versions should include an export function (e.g., to TXT, PDF, or clipboard). This would facilitate the integration of the application's output into students' academic workflows, such as lab reports and assignments. The methodological scope could also be broadened by incorporating additional numerical methods.
4. **Pursue Cross-Platform Porting:** Significant user interest in an iOS version was recorded. To maximize the application's accessibility and impact, a feasibility study and subsequent development of an iOS-compatible version are strongly recommended.
5. **Strengthen Pedagogical Scaffolding:** To address feedback regarding the clarity of certain steps, it is recommended to integrate optional, context-sensitive instructional elements. This could take the form of informational icons or tooltips that provide brief, theoretical explanations for key operations within the solution process, thereby reinforcing the educational value of the application.

Others which may help explain the program (including website links)	https://drive.google.com/file/d/1hLTOwd2HPr5lw6qQy7oXc884eK4VaRW7/view?usp=sharing https://norsu.edu.ph/wuri https://norsu.edu.ph/files/wuri/2025/A%20Numerical%20Methods%20Solver%20for%20Large-scale%20Linear%20Systems%20using%20Android%20Studio%20featuring%20Step-by-Step%20Visualization.pdf

Guideline for Application to WURI 2026

1. Prepare Program Cases

- Before the deadline, universities should prepare their innovative program cases using the provided template. One innovative case must be written in one MS/Word template. Any external files are not allowed.
- You may create new cases or revise and update those submitted for **WURI 2025**.

2. Upload Program Cases

- All program files must be written in **MS Word (.docx)** format (**no PDFs**).
- Combine all program cases into a **single ZIP file** and upload it through the **WURI 2026 Google Form**.

3. Revise or Update Cases

- If the same university uploads multiple ZIP files, **only the latest uploaded file** will be used for evaluation.

4. Submission Deadline

- The deadline for submission is **December 5, 2025**.
- After the deadline, **Google Form will no longer be accessible.**