2026 ASEE ANNUAL CONFERENCE June 21–24, 2026 | Charlotte Convention Center, NC

Host Sponsor UNC Charlotte

Paper ID #53062

Exploring FreeCAD for Teaching Engineering Graphics (GET 102) in Developing Nations

Joshua Oluwafisayo Olunlade, University of Cape Town

Joshua Olunlade is an MPhil student in Engineering Education at the University of Cape Town. He is interested in equity in engineering education, open-source tools for technical education, and sustainable education models in developing contexts. A recurring thread in his work is expanding equitable access to educational opportunity and how practical tools like FreeCAD can help bridge that gap.

Dr. Moses Olayemi, The University of Oklahoma

Dr. Mo (Moses Olayemi) is an Assistant Professor of Engineering Pathways at the University of Oklahoma. He is the Founding President of the African Engineering Education Fellows in the Diaspora, a non-governmental organization that leverages the experiences of African scholars in engineering education to inform and support engineering education policy, practice, and pedagogies in Africa. His research revolves around culturally relevant engineering education, STEM education in resource-limited contexts, instrument development, study abroad outcomes and processes, mixed methods research, course design and alignment of content, assessment, and pedagogy, comparative analyses of engineering education systems, and the food-energy-water nexus.

Exploring FreeCAD for Teaching Engineering Graphics (GET 102) in Developing Nations

INTRODUCTION

Computer-Aided Design (CAD) is no longer just a drafting aid, but rather the lingua franca of the modern engineering profession. Parametric modeling and digital simulation expertise is now increasingly a requirement to enter the world of the global engineering economy. Therefore, engineering programs all around the globe have included CAD training as a prime requirement, to shift learners out of merely playing with geometrical forms, and into what DaMaren et al. [1] calls strategic flexibility, the ability to approach complex design problems using digital tools. Therefore, even though the pedagogical need of CAD is already established, the methods of its delivery remain contested, especially in the Global South.

Proprietary, industry-standard software packages including SolidWorks, AutoCAD, and Fusion 360 are some popular software used for teaching CAD. These tools are strong, but they are based on licensing schemes which are more suited for corporations and not the economic reality of teaching and learning institutions in developing countries. To institutions in Nigeria, the recurring cost of commercial licenses is in direct competition with the basic infrastructure requirements such as maintenance of hardware and power generation. According to Lotfi et al. [2], such economic obstacle introduces the so-called gatekeeping effect, in which only institutions with funds can access high-quality engineering tools, which further widens the digital divide. More so, this proprietary reliance also leads to a post-graduation divide: graduates who are trained on a single costly software, find themselves out of that local economy unable to afford the tools that they have spent 5 years learning to use. This issue is thrust into a new perspective with the recent introduction of the Core Curriculum and Minimum Academic Standards (CCMAS) in Nigeria.

As Abdullahi et al. [3] have mentioned, the CCMAS is a paradigm shift to the employable skills and learner-centered pedagogy. The curriculum stipulates that students learn the Introduction to Engineering Graphics and Solid Modelling (GET 102), a course that intends to expose students to parametric design and the concept of design thinking. Nevertheless, the mismatch between these lofty curricular ambitions and economic realities in the region poses a very serious question: How can Nigerian universities provide world-class engineering education at a time when the average cost of industry standard tools is many times greater than the yearly income of an entry-level engineer?

A good solution to this dilemma is Free and Open-Source Software (FOSS). FreeCAD and similar tools provide a free and legal choice, which allows students the liberty to install, experiment, and practice without the uncertainty of license expiration or even the legal dangers of software piracy. However, there are still doubts that FOSS tools can meet the demanding technical needs of a formal engineering major.

This paper responds to that concern by assessing the feasibility of FreeCAD as a main teaching tool towards GET 102. In contrast to past research where researchers focused majorly on the overall adoption of FOSS, this study focuses on mapping the particular features and capabilities of FreeCAD against the learning outcomes stipulated by the Nigerian CCMAS. We explore the technical feasibility of the software, compare the economic aspects of such adoption to the proprietary alternatives, and provide the empirical evidence on the student perceptions of such a pilot implementation. Through the identification of a low-cost, high-accessibility pathway to CAD proficiency, this research provides a practical roadmap for sustainable engineering training in resource-constrained settings.

LITERATURE REVIEW

Free and Open-Source Software in Engineering Education

The introduction of Free and Open Source Software (FOSS) in the engineering curricula is no longer a fad but a strategic requirement driven by the need to provide equal access. Lotfi et al. [2] noted that commercial platforms offered by vendors such as MathWorks or National Instruments offer a rich educational experience, but their purchase cost is a major prohibitive factor among institutions that do not have strong financial support. FOSS alternatives have therefore become a reasonable choice, as they provide flexibility and lower cost, which is consistent with the democratization of science [4]. FOSS lets institutions prioritize funding on necessary hardware and infrastructure instead of having to pay for recurring license fees

In addition to financial sustainability, FOSS promotes collaborative learning. According to Du et al. [5], an open platform reduces duplication of effort in developing teaching material, which enables educational professionals to be more innovative in their pedagogy, and not spend time on software development. The open-source revolution has been applied to hardware too; Beattie et al. [4] observed that the use of open-source software and hardware (e.g. 3D printing and single-board computers) enable labs in resource-constrained settings to obtain scientific equipment at a small fraction of the cost of proprietary products. The availability of these are important in science diplomacy, which helps to bridge the divide between well-funded laboratories in developed countries and those in developing regions.

Within the particular area of Computer-Aided Design (CAD), the open-source ecosystem is providing very strong tools that compete with commercial standards. Machado et al. [6] emphasized that FOSS CAD tools are a fundamental part of Open-Source Hardware (OSH) development because these tools enable models to be studied, edited, and distributed without vendor lock-in. Machado et al. [6] carried out a comparative examination of OpenSCAD and FreeCAD and showed that, whereas script-based modelling software such as OpenSCAD are prevalent, FreeCAD provides a greater range of engineering instructional attributes due to their capacity to address standard parametric formats (such as STEP) and integrate a graphical user interface with Python scripting [6]. Moreover, Hu and Taylor [7] have noted that FreeCAD is a very flexible and extensible system, implying in the process that it is a good platform to create Intelligent Tutoring Systems (ITS). This would enable educators to come up with their own modules to teach the intricate concepts of solid modelling.

CAD in Engineering Learning and Choice of Tools.

Computer-Aided Design (CAD) is currently accepted globally as an essential skill in the contemporary engineering profession, but its teaching and learning methodology is still inconsistent. DaMaren et al. [1] noted that teaching CAD comprehensively is often limited by a lack of resources, as instruction ranges from basic software interaction to advanced product data management. To take care of this they came up with a Visual CAD Learning Framework whereby knowledge is classified into three areas: declarative (knowing what the tool can do), procedural (knowing how to execute commands), and strategic (knowing how to approach design problems). Good training in engineering design should be more than mere procedural teaching. This would aid in developing strategic versatility in the students.

This difference between the processes of using the tool and designing with the tool is critical. Brink et al. [8] conducted a phenomenographic study of educators who use technology in their classes and discovered that frequently, the priorities of teachers shift toward covering the software rather than engaging in actual design thinking. They contended that teaching can be stuck with the procedural part of the software such that students are unable to acquire the problem solving skills that they require when tackling complex engineering problems. This pedagogical issue is fueled by some traditional approaches to teaching CAD, that frequently involves step-by-step textbook tasks, which only reinforce one approach to building a model [7].

According to research conducted by Hu and Taylor, strategic knowledge that is necessary in professional practice can be developed more effectively by intelligent systems that promote multiple solutions to problems [7]. These pedagogical outcomes depend on the choice of a CAD tool greatly. Although industry may require students to be familiar with particular proprietary software (e.g. SolidWorks, AutoCAD), the basic understanding of parametric modeling is a transferable skill. Machado et al. [6] found no compromises in the ability to export to standard formats for educational and open hardware purposes [6]. While FreeCAD has a steeper learning curve than simpler tools, it provides stronger grounding in parametric design principles through its combined graphical and scripting interface. Moreover, Lotfi et al. [2] suggested that exposure to open-source platforms deepens learning by allowing students to examine real-world system development details that are hidden within commercial “black box” software.

Technology Adoption Problems in Developing countries.

The introduction of engineering programs in developing countries has a special complex of structural and systemic issues. The recent shift in the Benchmark Minimum Academic Standards (BMAS) to the Core Curriculum and Minimum Academic Standards (CCMAS) in Nigeria is a serious attempt to streamline the education at universities to meet the realities of the 21st century [3]. The CCMAS, which was introduced in late 2022, is focused on learner-centered pedagogies and acquisition of employable skills [9]. Nevertheless, there is still a huge disparity between policy intent and classroom reality. Ewunonu et al. [9] also placed a lack of proper instructional material, large classes, and time constraints to conduct lessons, as some of the main barriers to the implementation of the CCMAS. The harsh resource constraints prevalent in the Nigerian universities add to these problems.

Beattie et al. [4] have noted that sophisticated equipment and software are usually costly and available only in affluent institutions due to the high costs associated with scientific research and education. The cumulative financial cost of proprietary licenses of CAD software is usually prohibitive in systems where access to consistent electricity and equipment is already limited. FOSS provides direct mitigation for these challenges. Open-source tools are cost-effective because they eliminate the high costs of software licenses as explained by Lotfi et al. [2] and Beattie et al. [4]. Further, the utilization of FOSS directly helps to meet the CCMAS objective of facilitating the principle of active participation in learning [3] by offering students the tools that they can legally install on their personal devices. This approach ensures that learning is not restricted to limited computer lab time, which is particularly important given the time constraints identified by Ewunonu et al. [9].

Research Gap

While existing literature underscores the financial and collaborative advantages of FOSS in engineering education [2], [4] and demonstrates FreeCAD's capacity for teaching parametric design [6], [7], a critical gap remains regarding its targeted application in developing nations. Previous research has primarily focused on broad FOSS adoption, lacking a systematic mapping of FreeCAD’s capabilities against specific national curriculum mandates, such as Nigeria's CCMAS for Engineering Graphics (GET 102). Furthermore, empirical data regarding student perceptions of these tools within African university classrooms is notably absent. This study addresses these gaps by evaluating both the technical adequacy and the pedagogical viability of FreeCAD as a sustainable alternative to proprietary software in resource-constrained environments.

METHODOLOGY

Research Design

This study employs a mixed-methods research design to evaluate FreeCAD's suitability for teaching the GET 102 course at Nigerian universities. A mixed-methods approach is methodologically justified here because it allows for a comprehensive triangulation of objective technical parameters with subjective user experiences. The design integrates three complementary elements: a systematic curriculum-to-tool mapping, a comparative cost-benefit analysis, and a student perception survey. This combination effectively balances the quantitative metrics of software alignment and economic feasibility with qualitative insights into learner friction and pedagogical value, yielding a holistic understanding of technology adoption in resource-constrained contexts.

Curriculum Mapping Procedure

The curriculum mapping aspect was used to systematically assess the compatibility between the learning outcomes of the CCMAS GET 102 in Nigeria and the technical abilities of FreeCAD. We started with the extraction of seven learning outcomes and fifteen technical topics which are outlined in the CCMAS Engineering and Technology curriculum document [10]. Such themes were orthogonal projections (first and third angle), isometric projections, sectioning, sketching, solid modeling, technical drawing production, assembly modeling, sheet metal design, surface modeling, rendering and simulation among others.

With every topic of the curriculum, we performed hands-on exploration with the stable release version, version 1.0, available during the time of the study. This was done by trying to complete normal GET 102 tasks and assignment with the help of relevant FreeCAD workbenches, such as Sketcher, Part Design, TechDraw, Assembly, Sheet Metal, Surface, FEM (Finite Element Method) and Path. The workbenches and characteristics of the work that could be executed in each workbench, as well as the potential and limitations of each were also noted.

All the topics have been rated in terms of three categories of alignments. “Fully Supported” meant that FreeCAD possessed all the functionality needed to satisfy or even exceed the requirements of the learning outcomes to enable students to gain the necessary competencies without much restriction. “Partially Supported” was understood to mean that FreeCAD is functionally capable, with constraints in finish, completeness or ease of use compared to commercial offerings; students can build core capabilities at the risk of difficulty. “Not Supported” meant that FreeCAD does not yet have the necessary functionality to meet the learning outcome, and students cannot develop the necessary competencies with the help of the FreeCAD only.

The mapping activity generated the detailed assessment table (Table 1) that recorded all the extracted topics with their CCMAS learning outcomes, their corresponding FreeCAD workbench, their main characteristics, their rating of alignment and their implementation notes. Such a systematic process provided full coverage of all the curriculum requirements and at the same time provided the specific areas of strength and limitation.

Cost-Benefit Comparison

The cost-benefit analysis was performed between FreeCAD and three popular proprietary CAD software programs, which included Autodesk Fusion 360, Autodesk Inventor, and Dassault Systemes SolidWorks. These platforms were chosen due to their popularity in engineering education and practice.

In the case of each platform, we have collected data on various dimensions. Both the student license (cost, duration and restrictions) and professional license (yearly subscription) were recorded in terms of licensing costs. We determined five-year total cost of ownership in a post-graduation outlook, which in this case would be the financial burden that graduates would incur to keep using professionally the software that they studied in school. Data on system requirements such as compatibility with operating systems, RAM requirements, GPU specifications and storage needs were obtained using official vendor documentation. All pricing information had been gathered on official vendor websites in January 2025: education.autodesk.com to get Fusion 360 and Inventor, and solidworks.com to get SolidWorks.

To put this in context, we translated cost into the Nigerian Naira to the current exchange rates and compared it with the Nigerian engineering salaries at the entry level. The review resulted in a detailed comparison table (Table 2) of cost, technical requirements, accessibility factors, and essential capabilities of each platform. The information allows the judicious evaluation of trade-offs among software capability, accessibility, and affordability in the long term.

Student Perception Survey

To get user perspectives on FreeCAD pedagogical performance, we surveyed students who have undergone GET 102 with the help of FreeCAD as the main instructional CAD tool in the 2024/2025 Second Semester. The course had a population of more than 100 students in various engineering departments such as Mechanical, Civil, Electrical & Electronics, Chemical and Computer Engineering.

Our questionnaire was a 12-item questionnaire that was conducted through Google Forms. It consisted of four sections, these included: background information (current level, department, previous CAD experience), four Likert items comparing ease of learning, adequacy to course objectives, interface usability, and feature adequacy (ranged on a scale of 1-5), a series of multiple-choice items clarifying the challenges faced by the participants, comparative evaluation of students with prior AutoCAD/SolidWorks experience, and three open-ended questions asking the participants to answer what they liked the most in FreeCAD, what they thought could be improved, and whether they would recommend FreeCAD.

The survey was sent through email and WhatsApp, and all students who were enrolled were allowed to participate voluntarily and anonymously. The survey was conducted over a period of seven days, and a reminder was sent after three days. The survey was filled by twenty-three students, which represented a 23% response rate. This is a modest sample but is sufficient for exploratory descriptive analysis and thematic analysis.

The descriptive statistics consisted of mean, standard deviation, and frequency distribution, which we employed to analyze quantitative data (Likert responses). Thematic analysis was used in the analysis of the open-ended responses: all responses were read, similar comments were grouped and coded, recurrent themes were identified, and representative quotes demonstrating the important views were extracted. This mixed-methods analysis provided quantitative and qualitative evaluations and would aid in the experiences of students.

Limitations

There are a few limitations to be noted. To begin with, the research is based on the implementation of FreeCAD in GET 102, which occurred at one institution. The results may not necessarily transfer to the context of other infrastructure, preparation levels of students, or methods of instruction. Second, the response rate of the student survey was 23 percent (n=23). Although sufficient for exploratory analysis, the sample size is too small to provide statistical generalizability, and voluntary participation can lead to self-selection bias toward students with strong opinions. Third, the research focuses on capturing the perceptions right after taking the courses. Long-term retention of skills, transfer to advanced courses, and applicability to professional contexts cannot be assessed within this timeframe. Fourth, the research will not be able to separate the effects of instructional quality and pedagogical approach on student perceptions, which may be influenced by software capabilities as well. Lastly, not all the participating students had previous experience with commercial CAD platforms, which restricted the validity of comparative assessment. Nevertheless, the study makes significant contributions to initial evidence about the pedagogical success of FreeCAD in a resource-constrained environment and creates a methodology that may be used in the context of other evaluations of educational technology based on open-source programs.

RESULTS AND DISCUSSION

Curriculum Mapping Results

The entire curriculum-to-tool mapping as shown in Table 1 is a systematic assessment of how FreeCAD meets every GET 102 learning outcome in the standards of the CCMAS in Nigeria. Among the 15 topics studied, 8 (53%), got full support rating, 6 (40%), partial support rating and, 1 (7%) was not supported by FreeCAD. This distribution demonstrates that there is a substantial and yet imperfect correlation between the FreeCAD functionality and the GET 102 prerequisites.

The eight fully supported topics are basic engineering graphic skills. Orthographic projections (first and third angle) are fully supported by the TechDraw workbench of FreeCAD that automatically creates projection views of 3D models with customizable projection standards and ISO-conformity. Isometric projections also have the advantage of direct TechDraw support of axonometric views at the correct scale. Sectioning, such as simple, offset, and aligned section views, works well, although the hatching patterns can sometimes need a hand redefinition. The Part Design and Part workbenches provide full support of three-dimensional solid modeling (arguably the core competency of GET 102), with pad, pocket, revolve, loft, and sweep functionality all being parametric, with industry-standard feature-based modeling. FreeCAD is also very strong in producing technical drawing: the TechDraw workbench allows multi-view sheet layouts with automatic updating when the 3D models are modified, which meets the CCMAS requirement of having automated viewports. Lastly, the additive and subtractive manufacturing can be prepared with excellent support in STL/OBJ export of 3D printing and the Path workbench of simple CNC toolpath generation.

The six partially supported topics display the weakness of FreeCAD in the high-end functionalities as well as its sufficiency in the initial level of teaching. By being a pedagogical tool and not a functionality of any CAD software, design thinking cannot be supported out-of-the-box by any given CAD platform; the parametric design philosophy of FreeCAD and constraint-based sketching do invite the use of iterative design processes, but faculty has to explicitly incorporate design thinking pedagogy. The capability to do sheet metal modeling is present as a specific workbench with bend, unfold and junction operations, although the tooling is less developed than the commercial offerings and could pose a challenge to students who need complex sheet metal prototyping.

Surface modeling capabilities, although available via the Surface workbench and the more advanced loft/sweep functions, do not match professional NURBS modelers; simple surface work can be done, but high-end surface modelling is not available at this point in FreeCAD. The greatest gap is probably assembly modeling: there are already several different assembly solutions (the built-in Assembly Workbench, Assembly 3, Assembly 4, A2plus), but there is no common, standardized assembly method that would be easy to use, and none of them have the feel of commercial assembly interfaces yet. Rendering features relies on external engines (POV-Ray, LuxRender) and need further configuration and the absence of the integration of commercial rendering features. Lastly, FEM workbench simulation and analysis offer fundamental structural and thermal analysis to be used in introducing the concepts of finite elements but is not as sophisticated or as rich as more specific FEA systems.

Freehand drawing is at this point not supported by FreeCAD. FreeCAD is good at constraint-based sketching, but not freehand. This means that additional paperwork or tablet-based sketching practice are required to complete that part of the course.

This pattern of complete support on the lower tier skills and partial support on the upper-level features has an excellent pedagogical fit in the introductory position of GET 102. The key competencies that first-year engineering students must build on are mostly in the area of visualization, orthographic projection, parametric modeling and technical documentation, where FreeCAD is entirely competent. Such advanced options as advanced assembly modeling, photorealistic representation, and extensive simulation are above the learning goals of first year courses. The curriculum mapping recommends hence that FreeCAD can effectively be used as the primary CAD platform of GET 102 with the realization that engineering courses at higher levels could offer exposure to commercial tools that have more advanced features in specialized areas.

Table 1: Simplified Curriculum Mapping Table

CCMAS Topic	FreeCAD Workbench	Support Level	Key Notes
Orthographic Projections	TechDraw	● Full	Auto-generation, ISO standards
Isometric Projections	TechDraw	● Full	Native axonometric support
Sectioning	TechDraw	● Full	All section types supported
3D Solid Modeling	Part Design, Part	● Full	Feature-based parametric modeling
Technical Drawings	TechDraw	● Full	Multi-view sheets, auto-update
Viewports	TechDraw	● Full	Automated viewport generation
Manufacturing Prep	Path, Mesh	● Full	STL export, basic CAM
Sketching (2D/3D)	Sketcher	● Full	Constraint-based sketching
Design Thinking	General Platform	◐ Partial	Supports iterative process
Sheet Metal	Sheet Metal Workbench	◐ Partial	Basic operations functional
Surface Modeling	Surface, Part Design	◐ Partial	Limited advanced surfacing
Assembly	Assembly 3/4, A2plus	◐ Partial	Multiple solutions, not standardized
Rendering	Render Workbench	◐ Partial	Requires external engines
Simulation/FEA	FEM	◐ Partial	Basic structural/thermal analysis
Freehand Sketching	N/A	O Not Supported	Requires supplementary methods

Legend: ● Full Support | ◐ Partial Support | O Not Supported

Cost and Accessibility Analysis

Table 2 shows that the cost difference between FreeCAD and proprietary solutions is drastic, and it has very deep repercussions on the education of engineers in situations where resource scarcity exists. Although the four discussed platforms have free or affordable student licensing, the transition at post-graduation shows widely varied avenues for sustainable continued use of the software.

FreeCAD is and will always be free. This means students, instructors and institutions do not have to pay for purchase, subscription, maintenance or licensing fees. This is in contrast to proprietary platforms where the usage of the service by professionals is associated with significant amounts of further investment: Fusion 360 is priced at $680 per year ($3,400 over five years), Autodesk Inventor costs $2,585 per year ($12,925 over five years), and SolidWorks costs $2,820 - 4,716 per year ($14,100 – $23,580 over five years). These costs do not just imply a professional costs to Nigerian engineering graduates, but an overwhelming obstacles to economic engagement. Nigeria Starting salaries of entry-level engineers are in the range of N150,000-300,000 [11] per month (with current exchange rates, approximately 110 – 220 USD). SolidWorks professional license therefore constitutes 128 - 214 months of gross income that has to be invested entirely in software licensing for a single year.

These calculations do not take into account other costs, and this means such costs are practically insurmountable to new graduates who may wish to have their own consulting firms, begin engineering businesses or keep practicing CAD skills without corporate jobs. This barrier is a cost factor that maintains structural inequality in engineering practice. Individuals who are fortunate enough to find jobs in large companies with established software budgets are able to still use and perfect the CAD skills that they have learned while in school. Employees in small-to-medium business, startups, state organizations with constrained budgets, or in rural environments must make a harsh decision: it is either they give up ethics and use pirated software, therefore sacrificing professional ethical standards and potentially incurring legal liability, or they continue depending on licenses provided by employers.

FreeCAD, however, may help to reduce the barriers. Students after graduation can practice what they studied in academia as soon as possible and sustainably. In addition to direct costs, accessibility issues make the use of FOSS in developing situations even more advantageous. System requirements show a major difference: FreeCAD can be run with 4GB RAM (8GB suggested) and basic OpenGL 3.0 graphics, whereas Inventor and SolidWorks require a minimum of 16GB RAM (32GB recommended) with certified workstation GPUs. With relatively low hardware requirements, FreeCAD allows more people to participate in contexts where the students and institutions tend to be using older or lower-specification hardware. Table 2 summarizes the comparison between FreeCAD, Fusion 360, Autodesk Inventor and SolidWorks.

Platforms also vary in terms of installation and deployment issues. The size of the download (around 1GB) of FreeCAD allows it to be easily shared even with internet connection at lower speeds and since it utilizes an opensource license, they can easily be legally shared via USB flash drives, whereas the size needed for the download and installation of Inventor (40GB) places significant constraints in sites with high Internet access costs or slow internet access. This lack of licensing servers, activation, or periodic verifications, make FreeCAD easy to use in institutional computer laboratories with no administrative overhead or technical hassle that characterize commercial software administration in resource-constrained environments.

Pedagogical implications of education are not limited to personal access but also to institutional sustainability. A Nigerian engineering university department of with 500 students taking various courses could, using FreeCAD, implement the full range of CAD facilities at no software cost, and thus the budgetary allocation that would have been spent on software licensing could be diverted to other pressing needs such as laboratory equipment, faculty development, maintenance of infrastructure, and research support. FreeCAD therefore does not only facilitate affordability, but also autonomy, by allowing institutions to make pedagogical choices on the basis of educational merit instead of financial limitations.

Table 2: Summary of Comparison between FreeCAD, Fusion 360, Autodesk Inventor and SolidWorks.

Criterion	**FreeCAD**	**Fusion 360**	**Inventor**	**SolidWorks**
Student License	Free, unlimited	Free, 1 year renewable	Free, 1 year renewable	$60/year (individual)
Professional License	**Free**	$680/year	$ 2,585/year	$2,820 - 4,716/year
5-Year Total Cost	**$0**	$3400	$12,925	$14,100 – $23,580
Platform	Win/Mac/Linux	Win/Mac (cloud)	Windows only	Windows only
Min RAM	4 GB	8 GB	16 GB	16 GB
Internet Required	No	Yes (constant)	Activation only	Activation only
Storage	1GB+ of free disk space	8.5GB+ *of* free disk space	40 GB+ *of* free disk space	25 GB+ *of* free disk space

Criterion	FreeCAD	Fusion 360	Inventor	SolidWorks
Assembly Modeling	Functional (multiple solutions)	Good	Excellent	Excellent
Learning Curve	Steep	Moderate	Moderate	Moderate
Industry Adoption	Growing (FOSS)	Moderate	High	Very High (global)

Student Perceptions

To gather initial user perspectives on FreeCAD's pedagogical performance, an exploratory survey was administered to students enrolled in GET 102 during the 2024/2025 Second Semester. Given the single-institution context and the 23% response rate (23 out of 100+ students), these findings are framed as exploratory insights rather than definitive conclusions.

Respondents represented diverse engineering disciplines: **26.1%** from Mechanical Engineering, **26.1%** from Civil Engineering, **13%** from Electrical & Electronics Engineering, **30.4%** from Computer Engineering, and **4.3%** from Agricultural & Biosystems Engineering, ensuring cross-departmental representation. Regarding prior CAD experience, **47.8%** reported GET 102 as their first exposure to CAD software, **39.1%** had previous experience with AutoCAD, **13%** had used SolidWorks, and none of the participants surveyed reported any experience with other CAD platforms, indicating that most participants were genuine novices to computer-aided design.

Quantitative Findings

Table 3 presents descriptive statistics for the four Likert-scale items assessing student perceptions of FreeCAD's pedagogical effectiveness.

Students rated FreeCAD's ease of learning at M=**3.43** (SD=**1.16**), suggesting **moderately easy** initial learning experience but with significant split in student experiences. This rating reflects the tension between FreeCAD's powerful capabilities, and its steeper learning curve compared to more polished commercial interfaces. The adequacy for meeting GET 102 learning outcomes received M=4.22 (SD=0.951), indicating that students generally perceived FreeCAD as highly adequate for the course requirements. User interface usability was rated M=3.96 (SD=0.928), revealing satisfactory perceptions of FreeCAD's interface design and workflow. Feature sufficiency scored M=**3.96** (SD=0.876), suggesting students found FreeCAD's capabilities highly sufficient for completing assignments and projects.

Analysis of specific challenges encountered (Q8, multiple-selection) revealed the following patterns: **26%** of respondents identified "difficult to understand the interface," **4%** reported "frequent software crashes or bugs," 9**%** cited "lack of learning resources/tutorials," 4**%** noted "difficulty with specific features," 9**%** experienced "poor performance on my computer," 9**%** found "confusing workbench organization," 9**%** identified "limited rendering/visualization capabilities," and **26%** reported "I did not encounter significant challenges." The most commonly identified challenge was difficulty in understanding the software interface, highlighting the need for enhanced instructional support or software documentation.

Table 3: Student Perceptions of FreeCAD Across Key Measures of Usability and Learning Outcomes in GET 102

Perception Measure	**Mean**	**SD**	**Interpretation**
Q4: Ease of Learning (1=Very Difficult, 5=Very Easy)	3.43	**1.16**	Moderately Easy (High Variance)
Q5: Adequacy for GET 102 Outcomes (1=Did not meet, 5=Fully met)	**4.22**	**0.951**	High Adequacy / Strongly Aligned
Q6: User Interface Usability (1=Very Poor, 5=Excellent)	**3.96**	**0.928**	Good / Satisfactory Usability
Q7: Feature Sufficiency (1=Insufficient, 5=Sufficient)	**4.30**	**0.876**	Highly Sufficient

As shown in Table 4, among respondents with previous AutoCAD or SolidWorks experience (Q9, optional), **13 out of 15 (87%)** rated FreeCAD comparably or favorably for learning basic engineering graphics concepts, while **only 2 (13%)** rated commercial platforms as superior. This comparative assessment suggests that while FreeCAD may lack some polish of commercial alternatives, it remains pedagogically viable for foundational competency development.

Table 4: Comparative perceptions of FreeCAD versus proprietary tools (e.g., SolidWorks, Inventor) among students with prior CAD experience (n=19).

Comparison to Commercial Software (Q9)	**Counts**	**% of Total**	**Cumulative %**
FreeCAD is much better	6	31.6%	31.6%
FreeCAD is slightly better	3	15.8%	47.4%
They are about the same	4	21.1%	68.5%
Commercial Software is slightly better	1	5.3%	73.8%
Commercial Software is much better	1	5.3%	79.1%

Comparison to Commercial Software (Q9)	Counts	% of Total	Cumulative %
Not applicable - No prior CAD experience	*4*	*21.1%*	*100.0%*
Total	**19**	**100%**	

Qualitative Student Feedback

Student responses to open-ended questions provided deeper insight into the quantitative results, revealing a clear distinction between the *pedagogical value* of FreeCAD and the *technical friction* of using it. A thematic analysis of the responses regarding what students liked most (Q10), suggested improvements (Q11), and their likelihood to recommend the software (Q12) yielded three primary findings.

- **Accessibility and Foundational Learning (Strengths):** When asked what they liked most about FreeCAD (Q10), respondents frequently cited accessibility and simplicity. The fact that the software is "completely free and open source" was highlighted as a critical enabler, allowing students to "practice outside of lab hours without worrying about licenses." Beyond cost, a significant number of students identified the software's role in developing foundational engineering cognition. Respondents noted that the tool "helped [them] to think deeply especially the design aspect" and provided a "fundamental yet vast understanding of engineering drawings." One student explicitly described the learning curve as a benefit, noting that "now that we're using AutoCAD, it's quite easier having used a much harder software," suggesting that FreeCAD's rigor served as effective scaffolding for complex proprietary tools.
- **Interface and Stability Challenges (Weaknesses)**: Suggestions for improvement (Q11) were heavily concentrated on User Interface (UI) navigability and software stability. Multiple students requested modern UI features, specifically a "search bar" to locate tools rather than navigating through icon menus, which they found "unintuitive and fragmented across different workbenches." Stability was another recurring concern, with reports that the software "crashes, and systems with lower specs would have problems." Pedagogically, students requested "more videos with shorter duration" and "more practical examples" to bridge the gap between abstract tool functions and real-world application.
- **Recommendation for Future Cohorts**: Despite the reported technical challenges, the endorsement for FreeCAD was overwhelming. As summarized in Table 4, when asked whether they would recommend FreeCAD to future students (Q12), 91.3% (n=21) offered unqualified recommendations. The rationale for these recommendations largely centered on FreeCAD's role as a "steppingstone." Students described it as "an excellent foundational tool" that makes transitioning to complex software easier. Only 4.3% (n=1)

discouraged its use, while another 4.3% (n=1) recommended it with the specific caveat that it should be viewed strictly as a learning aid rather than an industrial standard.

Discussion and Implications

The combination of curriculum mapping, cost analysis and student perception data triangulation gave a holistic insight into the question of the viability of FreeCAD in the context of Nigerian engineering education. The results indicate that FreeCAD does not represent a feature-by-feature copy of proprietary standards, but rather a unique set of benefits, which might be more congruent with the particular objective of introductory engineering education of developing countries.

The curriculum mapping (Table 1) provides a very important differentiation between professional capability and instructional sufficiency. Although FreeCAD only supports 53% of the total potential capabilities of a modern CAD suite, it supports 100% of the core competencies of GET 102 (Orthographic Projection, Isometric Projection, Sectioning, and Basic Solid Modeling). The identified gaps, namely, high-end surface modeling, simulation, and standardized assembly, are high-level features, which are normally revealed later in the curriculum. Thus, even though FOSS tools might not be university-ready when it comes to complex final-year design projects, they are technically over-qualified when it comes to first-year engineering graphics. FreeCAD's application to GET 102 exemplifies the Fit for Purpose principle: students are not burdened with the complexity of commercial interfaces containing thousands of features they are not yet ready to use.

The deepest consequence of this work is the economic one. According to Table 2, , the difference between zero cost and commercial licensing is not merely a budgetary consideration; it is a matter of professional survival. Through the use of FreeCAD, institutions do not only save money; they make access democratic. The qualitative data emphasizes the importance of the fact that the students appreciated the possibility to use the software during non-laboratory hours, not being worried about licenses. This also has a direct effect of extending the classroom so that learners who have less powerful hardware (4GB RAM laptops) are still able to use the software. This is a far cry from the proprietary model whereby learning is usually limited to the working hours of a university computer lab or limited student licences.

The relatively lower ease-of-learning rating (M=3.43) may initially appear to be a weakness, but further examination suggests it may actually reflect a pedagogical strength. The qualitative feedback suggests that the rigor of the learning process in FreeCAD is due to the fact that students need to work through a strict constraint-based workflow, which makes them think more thoroughly about geometric relationships, and not use the built-in "wizards" of the commercial software. The data about the fact that 87 percent of students, who had prior experience with proprietary CAD, rated FreeCAD as equal or superior to commercial standards, imply that once the initial barrier is cleared, FreeCAD's underlying logic is compelling. According to one of the students, the transition to AutoCAD was easier after working with FreeCAD. This suggests that FreeCAD is a good example of a pedagogical gym. It is a more challenging exercise which develops more robust underlying mental frameworks of parametric design, and which can be readily adapted to more refined commercial applications.

For Nigerian universities implementing the CCMAS, these findings suggest a reconsideration of software procurement strategy.

- FOSS-First Sequencing: The curriculum should adopt a FOSS-first approach, with students using FreeCAD in Years 1 and 2 to develop foundational skills, and proprietary licenses reserved for advanced electives in Years 4 and 5.
- Instructional Support: Students' primary complaints about unintuitive interfaces and insufficient tutorials can be addressed not by changing the software but by developing targeted Open Educational Resources (OER) aligned with the CCMAS syllabus.
- Hardware Independence: FreeCAD's modest system requirements allow institutions to reduce capital spending on high-end workstations, making use of existing lab equipment or students' personal computers.

CONCLUSION

This paper assessed the feasibility of FreeCAD as an alternative to proprietary software for delivering Engineering Graphics and Solid Modelling (GET 102) course in Nigeria. Through triangulation of curriculum mapping, cost analysis and student response we have been able to show that Open-Source Software is not just a second-best option in resource-constrained settings, but a strategic resource in sustainable engineering education.

Technically, FreeCAD meets all core CCMAS competencies for the introductory graphics course, despite fully supporting only 53% of commercial suite features. Economically, transitioning to FreeCAD eliminates software costs for institutions, students, and instructors. Pedagogically, while students noted a steeper learning curve, 87% of those with prior CAD experience rated FreeCAD as comparable or superior for learning foundational engineering graphics concepts, suggesting the software effectively develops critical skills for engineering practice. It is important to note that while FreeCAD fully satisfies GET 102's foundational requirements, it is not intended to replace commercial CAD platforms for advanced or industry-level applications; it serves as a robust and accessible pedagogical steppingstone.

We recommend that institutions in developing countries adopt a FOSS-first approach in Years 1 and 2. Using tools like FreeCAD as the primary instructional platform can help decouple educational quality from funding fluctuations, ensuring that every student, regardless of personal economic status or institutional budget, graduates with legal, permanent access to functional design tools.

Future work should focus on developing locally adapted Open Educational Resources (OER) to support instructor transitions, ensuring that the next generation of Nigerian engineers is grounded in the principles of accessibility, sustainability, and technical independence.

REFERENCES

[1] E. DaMaren, R. Consell, A. Gryguć, C. Koenig, A. Olechowski, and J. Rosales, "An Educator's Framework for Modern Computer-Aided Design Instruction," in *Proceedings of the Canadian Engineering Education Association (CEEA)*, Jul. 2025. doi: https://doi.org/10.24908/pceea.2025.19633.

[2] N. Lotfi, K. Mbanisi, D. Auslander, C. Berry, L. Rodriguez, and M. Molki, "Promoting Open-source Hardware and Software Platforms in Mechatronics and Robotics Engineering Education," in *2020 ASEE Virtual Annual Conference Content Access Proceedings*, Virtual On line: ASEE Conferences, Jun. 2020, p. 35107. doi: 10.18260/1-2--35107.

[3] G. Abdullahi, E.-A. Youboere, and E. Tamunodni, "Core Curriculum and Minimum Academic Standards (CCMAS) and Foundation of Teacher Education in Nigeria: Matters Arising," *DJERD*, vol. 22, no. 1, pp. 255–269, Apr. 2025, doi: 10.61448/djerd22124.

[4] R. Beattie, S. Hippenmeyer, and F. M. Pauler, "SCOPES: Sparking Curiosity Through Open-Source Platforms in Education and Science," *Front. Educ.*, vol. 5, p. 48, May 2020, doi: 10.3389/feduc.2020.00048.

[5] D. Du, T. J. Baird, S. Bonella, and G. Pizzi, "OSSCAR, an open platform for collaborative development of computational tools for education in science," 2022, doi: 10.48550/ARXIV.2203.12019.

[6] F. Machado, N. Malpica, and S. Borromeo, "Parametric CAD modeling for open source scientific hardware: Comparing OpenSCAD and FreeCAD Python scripts," *PLOS ONE*, vol. 14, no. 12, p. e0225795, Dec. 2019, doi: 10.1371/journal.pone.0225795.

[7] Y. Hu and M. Taylor, "WORK IN PROGRESS: A Computer-Aided Design Intelligent Tutoring System Teaching Strategic Flexibility," in *2016 ASEE Annual Conference & Exposition Proceedings*, New Orleans, Louisiana: ASEE Conferences, Jun. 2016, p. 27208. doi: 10.18260/p.27208.

[8] H. Brink, N. Kilbrink, and N. Gericke, "Teach to use CAD or through using CAD: An interview study with technology teachers," *Int J Technol Des Educ*, vol. 33, no. 3, pp. 957–979, Jul. 2023, doi: 10.1007/s10798-022-09770-1.

[9] T. Ewunonu, C. Etus, A. Otuonye, I. Ajare, I. Ezeh, and O. Nwokonkwo, "Teachers Perceived Difficulties in Implementing Core Curriculum Minimum Academic Standards on Cyber Security Education," *STPP*, vol. 9, no. 1, pp. 63–70, Jun. 2025, doi: 10.11648/j.stpp.20250901.16.

[10] National Universities Commission (NUC), *Core Curriculum and Minimum Academic Standards (CCMAS) for Nigerian Universities: Engineering and Technology*, Abuja, Nigeria., 2023. [Online]. Available: https://nuc-ccmas.ng/ccmas-book-series-engineering-and-technology/

[11] "Engineer Salary in Nigeria: How Much Do They Really Earn?" Accessed: Jan. 24, 2026. [Online]. Available: https://kashgain.net/blog/engineer-salary-in-nigeria-how-much-do-they-really-earn/