ISSN 1512-1232

Development of Robust and Cost-Effective Predictive Models for Improving Students' Performance in Programming Courses

Temitayo Fagbola[1], Oluyinka Akintayo[2], Olatayo Olaniyan [3], Ayodele Oloyede[4], Funmilola Egbetola[5]

[1,3]Federal University, Oye-Ekiti, Nigeria, [3,4]Caleb University, Lagos, [5]Ladoke Akintola University of Technology
[1]temitayo.fagbola@fuoye.edu.ng, [2]oluyinka.akintayo@gmail.com, [3]olatayo.olaniyan@fuoye.edu.ng, [4]deledeee@yahoo.com, [5]funmilolaegbetola@yahoo.com

Abstract
In this paper, a robust and cost-effective mobile-oriented system for predicting students' performance in tertiary education programmes in Federal University, Oye-Ekiti, Nigeria was developed. The factors influencing the performance of students in programming related courses were investigated. Statistical approaches such as frequencies, mean, standard deviation, correlation and multiple regression were used for descriptive analyses and model development. Thorough analysis of the obtained dataset showed that major factors affecting the performance of students in programming courses are erratic power supply, bad university facilities, student health and students' attendance. The developed predictive models will assist University stakeholders, managers and students in cost-effective and robust decision making that could facilitate improved student performance in programming courses in any prototype developing economy.

Keywords*: student performance modeling, programming courses, developing economy*

1. Introduction

Student performance indicates the learning outcome of a student relative to the examination(s) taken with regards to a predefined finite set of subjects, registered as the case may be, after a teaching/learning process has taken place (Irfan &Shabana, 2012). Student performance has been defined in different ways by different set of individuals, institutions and organizations over the years but all centered on evaluation based on acceptable standards, the capabilities of students relative to examinations, tests, quizzes or their assertiveness and participation in class activities. Student performance has also been found to be influenced by several human and non-human factors (Ogbogu, 2014; Prince et al., 2013) and have been a topic which has consistently been researched in recent times (Akinola&Nosiru, 2014).

The success rate of any educational institute or organization may depend on the prior evaluation of student's performance. This prior evaluation can be used in many different ways to direct the structuring of learning processes to optimize effectiveness on student performance (Justin & Dmitry, 2015).Teachers and students alike have for so long been unable to determine the effect that certain factors have on academic performances but rather anticipate good performances in the long run. This way, it becomes impossible for student to quickly re-adjust and retune performance demeaning factors surrounding them or probably their responses to such surrounding factors. Different methods have been used for student's performance evaluation and more than ever before, information generated by evaluation can be helpful to help students and tutors take timely, meaningful and effective decisions.

Traditionally, results of students in various assessments successfully completed often make up the performance data which its analysis has been a tool for prior evaluation of academic quality

ISSN 1512-1232
and performance of students in educational institutions (Colin, 2006). Colin (2006) emphasized that tutors should become proficient in methods to improve on existing knowledge and make appropriate scaffolding available through the revelation of what the students already know and what should be learnt. This information, if obtained at a defined level of accuracy and timeliness would improve student's performance through the value of its feedback (Shaymaa et al., 2015).

Programming is an expression or application of creative skills and imaginations, which requires the individuals' ability to interpret challenges into solutions (Kofi et al., 2013). Computer Science and Information Technology students are often required to offer several programming courses as contained in their curriculum. One of the main reasons that may be attributed to the decline in number of undergraduates who offer computer science is the perception that computing, especially programming is not easy to accomplish (Mustafa, 2013). Students in their early years of studies are required to study programming. This requirement often includes the knowledge of programming tools and languages, problem-solving skills, and effective strategies for program design and implementation (Kofi et al., 2013).

However, computer programming is an inseparable part of computer science and its related programmes in education. It is an absolutely necessary and extremely important skill that must be mastered by anyone intending to study computer science (Kofi et al., 2013).As a matter of fact, programming has become one of the most dreaded courses in which many students fail, probably because it demands a high level of abstraction and its languages have very complex syntax and semantic structures (Gomes et al., 2007). Hence, it can be argued that the same set of students who failed programming courses performed better in other courses offered alongside programming courses (Akinola&Nosiru, 2014). Normally in teaching computer programming, students are first introduced to algorithms, the concept of programming, basic data structures and are taught on how to effectively analyze problems, apply specific techniques to illustrate the problem solution and validate the solution. Computer programming courses are a part of many universities' curriculums and among the most important subjects for computer science students as well as information technology students. Computer programming is often regarded to as one of the fundamental part of Computer Science curriculum but it is often quite problematic (Gomes et al., 2007, p. 118-124). The failure rate in programming courses at the University level suggests that learning to program is a difficult task (Akinola&Nosiru, 2014). This performance is strongly influenced by several social, psychological, economic, environmental and personal factors which vary across individuals, institutions and countries (Irfan&Shabana, 2012).

The educational sector in developing countries is however been faced by a series of multi-factored challenges that contributes to the rapid fall in the performance of students located within such developing economy. Ogbogu (2014) noted that challenges such as poorly equipped departmental and central libraries, overcrowded lecture rooms, method of collating and accessing semester results, interruption of electricity supply, poor access to internet facilities, incessant strike and closure of school and poor accommodation facilities which are pertinent to developing countries affect student performance.

In this paper, factors influencing the performance of students in programming courses being offered in Federal University OyeEkiti (FUOYE)were investigated. These factors were then subjected to a series of analysis in a bid to extract the factors that were extremely significant to the performance of students in programming courses. The identified significant factors were then employed in the development of models which were later validated using some samples of the response from the respondents.

2. Literature review

Students' performance assessment has become a pressing issue that requires fair attention from all regardless of differences in interest and intentions (Amiroh&Farinda, 2016; Irfan &Shabana, 2012). Students' performance in recent times was noted not only to have been the concern of educators and academics alone, since corporations also have become concerned. This is

ISSN 1512-1232
because the supply chain of graduates for the labor market recognizes them(corporations), as the end user. Chermahini (2013) noted that students are different based on their ability in learning, how they respond to instructional practices, their motivational differences from one individual to another and that the more students understand the differences in their abilities, the better the chances they have to meet their different learning needs in order to achieve good scores in examinations.

Student performance is usually affected by the students' learning environment (Masura et al., 2012). Unfortunately, poor performances have ravaged the academic institutions because of institutions' failure to provide an accommodating environment that is conducive to the students' educational and learning needs (Ogbogu, 2014).

2.1 Related Works

Hijazi and Naqvi (2006) considered five (5) exogenous variables as predictors to the academic performance of students (Y) which is an endogenous variable. These factors includes; Attendance (ATT), Study hours (SH), Family income (FI), Mother Age (MA) and Mother's education (ME). The developed model is described as follows:

$$Y = -0.25313 + 1.026912ATT - 0.00209SH - 5.8 * 10^{-7}FI - 0.00453MA + 0.012193ME$$

The evaluated R-square value for the model was 0.24, which suggests that the five (5) factors considered explains 24% influence on the performance of a student while the remaining 76% influencing factors were unaccounted for by the presented model. Furthermore the model shows the study hours (SH) of the student as negative contributor to their performance although the authors believed a positive association would have been much more appropriate. Upon carrying out a F-Statistic test to determine the overall strength of the model, a highly significant value of 20.083 was obtained. This implies that the model is valid and very significant in the prediction of student performance. Irfan and Shabana (2012) explored four (4) predictors and referred to them as important in the determination of the academic performance of students. These factors include learning facilities, communication, proper guidance and family stress.

As noted by the authors, these factors had the following correlation and 2-tailed significance value when correlated with student performance. Communication had a value of 0.132 and 0.002, learning facilities had a value of 0.137 and 0.040, proper guidance had a value of 0.200 and 0.013 while family stress had a value of -0.020 and 0.809. Furthermore, the regression model developed as deduced from the presented table of coefficients, is described as follows:

$$SP = 2.514 + 0.204Cm + 0.160Lf + 0.177Pg - 0.132Fs$$

Where SP (student performance) is the dependent variable and Cm (Communication), Lf (Learning facilities), Pg (Proper guidance) and Fs (Family stress) are the predictors. This model hence shows that communication accounts for about 20%, learning facilities accounts for about 16%, proper guidance accounts for about 17 % positive variation in student performance while family stress accounts for about 13% negative variation in student performance.

Justin and Dmitry (2015) constructed a model using five (5) factors as the predictor of a student performance (SP) in a study conducted in Tanzania. These factors include Lack of interest (LACKINT), Triviality and lack of practice (TRILACK), Lack of drive and enthusiasm (LACKDRIV), Perception and attitude (PERCATT) and Lack of qualified teachers (LACKQUAL).

$$SP = 20.18 - 1.31LACKINT + 2.13TRILACK + 0.37LACKDRIV + 0.97PERCATT - 1.07LACKQUAL$$

Siti, Razifah and Nurhafizah (2015) examined the influence of family characteristics, self-efficacy and university features in the academic performance of a student. The duo noted, university features, and family characteristics were very significant to the study but self-efficacy was regarded as insignificant owing to its P-value of 0.891.

ISSN 1512-1232

As deduced from the table of coefficients, the multiple linear regression model presented by the author is as follows:

$$SP = 1.162 + 0.308\text{UF} - 0.013\text{SE} + 0.319\text{FC}$$

Where SP is the student performance which is the all dependent on the variables UF (University Features), SE (Self-Efficacy) and FC (Family Characteristics).

3. Metodology

3.1. Research Questions

i. Does practicing with personal computer help students perform better in programming courses?
ii. Do students who attend introductory programming classes perform better than those who don't?
iii. Do students who attempt their assignment by themselves perform better in programming courses?
iv. Are students who have a strong background in physics liable to perform better in programming courses?
v. Are students who have a strong background in mathematics liable to perform better in programming courses?
vi. Do older students perform better at programming than the younger ones?
vii. Do male students perform better in programming courses than their female counterparts?
viii. Do students who offer programming courses as a domicile department requirement perform better than students who offered programming for rudimentary knowledge purposes?

3.2. Research Hypothesis

The following hypothesis was developed for the purpose of this study;

i. Practicing with a personal computer is significantly related to student performance in programming courses.
ii. There exist a significant relationship between attending introductory classes and the academic performance of students in programming courses.
iii. Attempting programming assignment personally is significantly related to the performance of students in programming courses.
iv. A good background in physics is significantly related to the performance of students in programming courses.
v. A good background in mathematics is significantly related to student academic perfoTrmance in programming courses.
vi. There is a significant relationship between the age of a student and performance in programming courses.
vii. here exists a significant relationship between gender of a student and performance in programming courses.
viii. Domicile department requirement is significantly related to the performance of students in programming courses.

3.3. Research Study Area

This study was conducted at Federal University Oye-Ekiti, located at Aare-Afao Road, Oye-Ekiti Local Government, Ekiti state, Nigeria. With a coordinate representation of $7.7796°\ N$and$5.3155°\ E$. An observation through the university community of students who had offered programming courses at one time or the other during their academic pursuit was carried out. This was in a bid to isolate the factors that had significant influence on the performance of

ISSN 1512-1232
students in programming courses within that locality. This was done by visiting the lecture theatres to observe the peculiarities ascribed to students and lecturers at large.

3.4. Data Gathering, Representation and Coding

The primary data used was gathered using a structured student questionnaire. The questionnaire was made available both in soft (e-questionnaire) and hard form (printed). The e-questionnaire which was designed specifically for the collection of data for this research contained exactly the same question and metrics as its hardcopy equivalent and was used to obtain the responses of respondents who were not present at the institution due to their internship program. Each questionnaire contains a total of 80 variables, all in seven sections.

The first and seventh section of the questionnaire had six (6) and five (5) variables respectively and were used for hypothesis testing, while the second section had sixteen (16) variables, the third section had nineteen (19) variables, the fifth section had thirteen (13) variables, the sixth section had twelve (12) variables and the fourth section had ten (10) variables. All the variables represented on the questionnaire were aimed at investigating factors that were intrinsic to the students, lecturers, university environment and family among others. The variables presented in sections two (2) to six (6) are statements in a 5-points Likert scale ranging from 1 representing “strongly disagree” to 5 for “strongly agree”. The respondents (students) were required to respond to the questionnaire based on a programming course that has being offered in the university. Simple random sampling procedure was used to select undergraduates that participated in the study.

The seven (7) sections of the questionnaire were coded as follows:

Sections (2-6) of the questionnaire were coded as presented in Table 1. The factor coding was determined by the number of variables investigating a particular factor. Factors being investigated by three variables were coded as presented in Table 2 while factors that are investigated using four variables were coded as presented in Table 3. The respondent’s age from the Section 1 of the questionnaire was collected using series of age range and was coded as presented in Table 4.

1’s or 0’s were used in the representation of variables that are either true or false, yes or no, male or female and also in the representation of departments. Such that students whose department offer programming courses by default are represented by one while others are represented by zero.

Table 1: Likert Scale for Variables Represented on the Questionnaire

Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
5	4	3	2	1

Table 2: Likert Scale for Factors with three (3) Variables

Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
13 – 15	10 - 12	7 – 9	4 – 6	1 – 3

Table 3: Likert Scale for Factors with four (4) Variables

Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
17 – 20	13 – 16	9 – 12	5 – 8	1 – 4

Table 4: Likert Scale for Respondent’s Age Representation

Below 16	16 – 19	20 – 25	26 – 30	Above 30
1	2	3	4	5

ISSN 1512-1232

Table 5: Likert Scale for Respondent's Grade (Performance)

A	B	C	D	E	F
1	2	3	4	5	6

3.5. Factor Extraction

Twenty-One (21) factors were being investigated by the questionnaire with a total of 81 variables. Each factor was coded based on the cumulative of thevariables designated to investigate it. These various factors and their respective coding is described in sections 3.5.1 – 3.5.21 where $x_1, x_2 \dots x_{70}$ are the variables represented in the questionnaire as presented in Appendix A.

3.5.1. Student Study Habit (SSH)

This is the amount of the student's effective study in programming courses offered relative to the frequency of revising and practice and hours spent on revising the lecture notes. It was investigated by three variables x_1 , x_2 , x_3 and was coded as presented in Table 2.

3.5.2. Student Fear and Perception (SF)

This is the students' fearful perception of programming courses where a positive perception implies a reduction in fear factor of the student. This was investigated by the variables x_4, x_5, x_6 and coded as presented in Table 2.

3.5.3. Student Attendance (SATD)

This is the level of effort, seriousness and devotion of students towards learning to program. Investigated by the variables x_7, x_8, x_9 and coded as presented in Table 2.

3.5.4. Student Attitude (SAT)

This is the level of responsiveness of a student relative to their interest, behavior and seriousness to programming courses, and characterized by student's participation in class activities, assignment, willingness to learn, and motivation from friends, colleagues and lecturer(s). This was represented bythe variables x_{10}, x_{11}, x_{12}, x_{13} and coded as presented in Table 3.

3.5.5. Tutorials and Extra Classes (ST)

These are the extra effort put in place by students in other to have a clear understanding of the subject matter(s) discussed programming classes. This includes extra-classes attended, assistance from friends and use of online forums and materials. This factors was investigated by the variables x_{14}, x_{15}, x_{16} and coded as presented in Table 2.

3.5.6. Lecturer Attitude (LAT)

This is defined as the lecturers' assertiveness, interest to explicitly expatiate on the subject matter, ability to motivate the student and relate with the student in a means to improve their interest in the course. This was investigated by variables x_{17}, x_{18}, x_{19}, x_{20} and was coded as presented in Table 3.

3.5.7. Teaching Style (LTS)

This is defined as the pattern of teaching of the lecturer in charge (probably dishes out voluminous handouts or excessive assignments). Whether he carries the class along and helps the student conceptualize the concept of that particular programming course. This was investigated by variables x_{21}, x_{22}, x_{23}, x_{24} respectively and was coded as presented in Table 3.

3.5.8. Communication Skills (LCS)

This is the ability of the lecturer to deliver the course content in a less ambiguous manner and to the understanding of the students. This entails the clarity and explicitness of the lecturer.

ISSN 1512-1232

This was investigated by variables x_{25}, x_{26}, x_{27}, x_{28} respectively and was coded as presented in Table 3.

3.5.9. Lecturer Availability (LA)

This is the presence and accessibility of the lecturers' when they are needed by the student(s). This factor was coded as presented in Table 2 and was investigated by the variables x_{29}, x_{30}, x_{31} respectively.

3.5.10. Lecturer Dedication (LD)

This is the devotion of the lectures to the programming courses they tutor. This includes the assertiveness of the lecturers to their duty and extra effort put in place to ensure an excellent student performance. This factor was coded as presented in Table 3 and was investigated by the variables x_{32}, x_{33}, x_{34}, x_{35} respectively.

3.5.11. Health (OH)

This is the influence of medical condition on students' performance in programming courses. This factor was coded as presented in Table 2 and was investigated by the variables x_{36}, x_{37}, x_{38} respectively.

3.5.12. Electricity (OE)

This is defined as the erraticism of power supply as it affects the students' practice using computers and also other laboratory works. This factor was coded as presented in Table 2 and was investigated by the variables x_{39}, x_{40}, x_{41} respectively.

3.5.13. Background knowledge (OB)

This is the academic strength of the student in other courses that are elementarily related to computer programming (mathematics and physics). This factor was coded as presented in Table 3 and was investigated by the variables x_{42}, x_{43}, x_{44}, x_{45} respectively.

3.5.14. Facilities (UF)

This is the availability of appropriate programming learning facilities (computer laboratory) within the university environment. This factor was coded by as presented in Table 3 and was investigated by the variablesx_{46}, x_{47}, x_{48}, x_{49} respectively.

3.5.15. Class population (UCP)

This is the student to tutor population ratio during the programming course class. This factor was coded as presented in Table 2 and was investigated by the variables x_{50}, x_{51}, x_{52} respectively.

3.5.16. Lecture time (ULT)

This is the conduciveness of the lecture schedule. This factor was coded as presented in Table 2 and was investigated by the variables x_{53}, x_{54}, x_{55} respectively.

3.5.17. Teaching aids (UTA)

This is the availability of teaching aids (audio visuals) for the demonstration of the concept of programming courses. This factor was coded as presented in Table 2 and was investigated by the variables x_{56}, x_{57}, x_{58} respectively.

3.5.18. Family income (FI)

This is the robustness of the family income of the student. As it influence the ability of the student to afford textbook materials, print handout or even own a personal computer for effective study. This factor was coded as presented in Table 2 and was investigated by the variables x_{59}, x_{60}, x_{61}.

3.5.19. Family stress (FS)

This is the degree of disturbance from home. An unsettled home creates a paranoid atmosphere which seemly affects student performance. This factor was coded as presented in Table 2 and was investigated by the variables x_{62}, x_{63}, x_{64} respectively.

3.5.20. Parent education (FPE)

This is the degree of education of the students' parent. A poor motivation from home might destabilize the student cognitive sense, hence influencing the students' performance in programming. This factor was coded as presented in Table 2 and was investigated by the variables x_{65}, x_{66}, x_{67} respectively.

3.5.21. Proper guidance (FPG)

This is the student's family guidance and support level for programming courses. A student from a family of computer scientist is prone to having huge support and guidance from home. This factor was coded as presented in Table 2 and was investigated by the variables x_{68}, x_{69}, x_{70} respectively.

Reliability test was used to establish the identity of correlation coefficient of the variables and factors that were tested in this study. Cronbach's alpha was used to estimate the average correlation of both the variable dataset and the factor dataset to determine if they are standard or not. The reliability of the presented questionnaire as presented in Table 6 is acceptable at a Cronbach's alpha value of 0.731 for the variables and a Cronbach's alpha value of 0.530 for the factors. The instrument (questionnaire) employed for this study is hence acceptable since Siti, Razifah and Nurhafizah (2015) affirmed that a Cronbach's alpha value of 0.9 – 1.0 is excellent, 0.8 – 0.89 is good, 0.7 – 0.79 is acceptable, 0.6 – 0.69 is questionable while 0.5 – 0.59 is poor and value less than 0.5 is unacceptable.

Table 6: Reliability Statistics of the questionnaire's Variables

Cronbach's Alpha	N of Variables
.731	71

Table 7: Reliability Statistics of the Extracted Factors

Cronbach's Alpha	N of Variables
.530	22

3.6 Data Analysis

Statistical Package for Social Scientists (SPSS) version 16.0 was used to analyze the gathered data. Furthermore, two datasets were employed in the determination of the significant predictors to the student performance in programming courses. The first data set contains 70 variables which are the representation of all the variables in sections two (2) to six (6) of the questionnaire while the other data set includes all the extracted factors which was coded as presented in Table 2 and Table 3 and discussed in section 2.5 of this study. These data sets as

ISSN 1512-1232
obtained and factored were used in the analysis approach involving a Statistical (SPSS) and Excel Spreadsheet.

3.6.1. Correlation Analysis

In agreement with Varalakshmi et al. (2005), the Coefficient of correlation was used for measuring the magnitude of the linear relationship between student's performance and the predictors (factors) as suggested by Karl Pearson, a biometrician and statistician. The formula employed includes:

TABLE I. $\mathbf{r} = \frac{\sum xy}{n\sigma_x \cdot \sigma_y}$ where $\boldsymbol{\sigma}_x$ and $\boldsymbol{\sigma}_y$ are the Standard Deviation of x and y respectively

TABLE II. $\mathbf{r} = \frac{\sum xy}{\sqrt{\sum x^2 \cdot \sum y^2}}$, $\quad \mathbf{X = x - \bar{x}}$, $\mathbf{Y = y - \bar{y}}$.

When the deviations are taken from the actual mean, any of these methods can be applied. The correlation algorithm implemented in SPSS 16.0 was used to calculate the correlation between student performance, the presented variables and the extracted factors. Invariably, all variables present in the questionnaire and the extracted factors were correlated with student performance (grade) to determine the degree of correlation between them. The correlation between these variables and student performance (grade) was regarded as significant at a Sig. (2-tailed) value greater than or equal to 0.25. A data set of correlates was then generated from each of the two datasets on which the correlation analysis was performed.

3.6.2. Regression Analysis

Regression was used to measure the average relationship between student performance (grade) and the predictors. Functional relationship between student performance (SP) and a set of variables $\mathbf{x_1, x_2, \dots, x_n}$ can hence be expressed as:

$\mathbf{SP = f(x_1, x_2, \dots, x_n)}$where$\mathbf{x_1, x_2, \dots, x_n}$ are the several variables that are being considered.

3.7. Student Performance Model Development

Several multi-linear regression models could be developed with the aim of examining the effects of predictors that were intrinsic to the students who offered programming courses. Various models such as those representing the influences of the lecturers, university environment, family and all other associable factors on student performance were concisely structured into three (3) categories. This categorization includes the Student Controllable Performance Model (SCPM), Student Uncontrollable Performance Model (SUPM) and Hybrid Student Performance Model (HSPM) perspectives.

3.7.1. Student Controllable Performance Model (SCPM)

The controllable performance model was designed to predict student performance relative to factors that can be directly controlled or adjusted by the students themselves. The proposed model hence considers performance with respect to the study habit, perception and the rate of fear, attendance, attitude and extra classes (tutorials) attended by the students with the exclusion of all the factors that are insignificant.

3.7.2. Student Uncontrollable Performance Model (SUPM)

The Uncontrollable perspective was modelled to predict student performance with regards to factors that cannot be controlled (influenced) directly by the student. This includes factors that are intrinsic to the lecturers, University and Environment, Family and Other factors which might have an effect on student performance. All insignificant factors were duly excluded from the model being presented.

3.7.3. Hybrid Student Performance Model (HSPM)

The hybrid model relates the performance of students in programming courses to both the factors that can be controlled by students and those that cannot be controlled by them (students).

4. Results and Discussion

The result of analysis as obtained through the application of the Statistical Package for Social Studies (SPSS) and Excel Spreadsheet was discussed in this section as applied to the variable dataset and the factor dataset.

4.1. Results for Variable Dataset

The results pertinent to the variable dataset were discussed in this section. These results were as obtained through the analysis carried out with SPSS and Excel.

4.1.1. Demographic Analysis

The demographic analysis performed on the variable dataset describes the percentage frequency of the responses of the respondents. These frequencies are described in Table 8

Table 8: Demographics for the variables

	A	B	C	D	E	F
X80 (Grade)	31.2	28.8	35.6	2.4	2.0	0.0

4.1.2. Student Performance Model

Correlation and Regression analysis were enacted on the variable dataset which was coded directly from the questionnaire. A detail of this analysis is as follows:

4.1.2.1. Correlation Analysis

The degree of dependability between all the individual variables and respondents' performance was evaluated. Eleven (11) of the seventy (70) variables were found to be in correlation to the performance of students at a significant level of $\geq 0.25 \pm$.

Table 9: Correlated factors to the student performance

Correlates	Correlation Coefficients	
X4	-0.260	Programming sounded very scary
X5	-0.355	I was always nervous during programming classes
X6	-0.345	I was always nervous during programming examinations
X9	0.403	I was very serious with programming classes
X10	0.345	I believed I could understand the programming course
X11	0.290	I had interest in programming beyond class level
X14	-0.335	Group discussions helped me to understand programming
X19	0.342	Programming languages lecturers were never

ISSN 1512-1232

		partial in their dealings with students
X48	-0.266	Lack of computer programming facilities disrupted clear understanding of programming lessons
X54	-0.290	Programming courses were scheduled to non-conducive times
X55	-0.257	We had programming classes at unfavorable times

The variables that correlate with the grade of the respondents are presented in Table 9. The negative but relevant correlation value of the variable X4 implies that students to whom programming sounded scary had a lower performance. Variable X5 had a relevant but negative correlation value of -0.355 which implies that the higher the fear of students in programming classes the lower their performance and a lower fear factor during the programming performance increases the performance of students in programming courses. Variable X6 which defines a student's fear factor during an examination had a relevant but negative relationship with student performance having had a value of -0.345. This implies that the higher the fear expressed by students for programming during an examination, the lower their performance.

However, the variable X9 which defines the seriousness of a student with programming classes had a positive correlation value of 0.403 indicating that a decrease in its values implies a decrease in the performance of a student and an increase would mean an increase in performance of a student. The variable X10 which defines the attitude of students to understanding programming had a relevant and positive correlation value of 0.345 suggesting that its increase would yield an increase in student performance. Hence, the attitude of students has a relevant correlation to the performance of a student and its decrease would mean a decrease in student performance. The variable X11 which also defines the attitude of students with respect to their interest to program beyond class level, had a relevant correlation value of 0.290 in the positive direction. Implying that an interest to program beyond class level constitutes an increase in student performance and also suggests that students who intend to make a future out of programming tends to perform better in programming examinations. Variable X14 which defines group discussion however had a negative but relevant correlation with the performance of students in programming courses.

This suggests that the more students discuss about programming, the lower their performance. This might be as a result of increase in tension (fear) which as earlier discussed, negatively affects the academic performance of the students in programming courses. Variable X19 which connotes the attitude of programming lecturers and defined by the non-partiality of programming lectures, had a relevant correlation in a positive direction with a value of 0.342. By implication, there exists an uplift in the attitude of programming lecturers with regards to their non-partiality,which in turn, transcends to an increase in the performance of students.

Variable X48 defines the lack of computer programming facilities as a disruption to understanding programming lessons and had relevant correlation of 0.266 in the negative direction. This implies that the higher the lack of computer programming facilities the lesser the performance of students while the more the computer programming facilities provided for the use of student during programming lessons, the higher the academic performance of students in programming courses. The variables X54 and X55 however define the scheduling of programming courses to non-conducive and unfavorable times respectively. With both having a negative correlation value of -0.290 and -0.257 respectively. This suggests that scheduling programming lectures to non-conducive and favorable times, causes a decline in the academic performance of students.

4.1.2.2. Regression Analysis

Using the variable dataset, a hybrid model was developed for student performance in programming courses. This model considered only the variables which were significant to the

ISSN 1512-1232
performance of students programming courses. Fifty-three (53) out of the entire seventy (70) variables were retrieved after the exclusion of all the insignificant factors, from which a model of significant variables was developed.

Table 10: Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.961[a]	.923	.906	.295

a. Predictors: (Constant), X70, X46, X35, X65, X52, X6, X11, X62, X8, X39, X2, X45, X25, X7, X61, X13, X29, X15, X23, X12, X47, X42, X17, X36, X9, X34, X67, X49, X58, X54, X24, X33, X69, X26, X41, X19, X59, X37, X31, X40, X53, X5, X51, X48, X63, X43, X38, X14, X30, X21, X16, X57, X55

Table 11: ANOVA

Model	Sum of Squares	Df	Mean Square	F	Sig.
Regression	251.139	53	4.738	54.389	.000[a]
Residual	20.996	241	.087		
Total	272.136	294			

a. Predictors: (Constant), X70, X46, X35, X65, X52, X6, X11, X62, X8, X39, X2, X45, X25, X7, X61, X13, X29, X15, X23, X12, X47, X42, X17, X36, X9, X34, X67, X49, X58, X54, X24, X33, X69, X26, X41, X19, X59, X37, X31, X40, X53, X5, X51, X48, X63, X43, X38, X14, X30, X21, X16, X57, X55

b. Dependent Variable: X80

The developed model of strictly significant variables as evidenced by Table 10, had a R-Square value of 0.923. This indicates that the fifty-three (53) variables considered by this model causes 92.3% variation in the performance of students in programming courses.
As presented in Table 11, the F-Statistics value of 54.389 shows that the model is strong and is adequately fit since it has a P-Value of 0.00 which is less than the alpha level of 0.05.

The developed hybrid model is presented thus:

$$SP = 9.046 + 0.358X_2 - 0.276X_5 + 0.287X_6 - 0.187X_7 + 0.323X_8 + 0.664X_9 - 0.126X_{11} - 0.263X_{12} + 0.139X_{13} - 0.394X_{14} - 0.398X_{15} + 0.469X_{16} - 0.262X_{17} + 0.207X_{19} + 0.133X_{21} - 0.310X_{23} - 0.304X_{24} - 0.273X_{25} - 0.154X_{26} - 0.208X_{29} + 0.178X_{30} - 0.148X_{31} - 0.412X_{32} + 0.111X_{34} + 0.298X_{35} - 0.166X_{36} + 0.276X_{37} + 0.149X_{38} - 0.265X_{39} + 0.173X_{40} - 0.249X_{41} - 0.286X_{42} + 0.280X_{43} + 0.455X_{45} - 0.261X_{46} - 0.359X_{47} + 0.110X_{48} - 0.168X_{49} + 0.140X_{51} + 0.227X_{52} + 0.387X_{53} - 0.222X_{54} - 0.237X_{55} - 0.395X_{57} + 0.427X_{58} + 0.305X_{59} - 0.165X_{61} - 0.417X_{62} + 0.223X_{63} - 0.494X_{65} + 0.278X_{67} - 0.127X_{69} - 0.335X_{70} \quad (1)$$

ISSN 1512-1232

Instances of the model validation result is as presented thus (the bold are the predicted and actual grade, respectively);

4, 3, 3, 4, 5, 3, 4, 1, 3, 2, 1, 4, 1, 2, 4, 4, 5, 4, 3, 3, 1, 4, 5, 4, 5, 4, 5, 5, 4, 3, 3, 5, 4, 4, 5, 4, 5, 3, 3, 2, 4, 4, 5, 4, 5, 1, 5, 4, 2, 1, 2, 4, 5, **4.945, 5**

5, 2, 4, 5, 5, 4, 3, 2, 1, 4, 3, 5, 1, 2, 4, 4, 5, 4, 3, 4, 4, 3, 5, 4, 5, 4, 5, 4, 3, 2, 4, 5, 4, 4, 5, 4, 4, 3, 5, 4, 5, 4, 5, 5, 4, 1, 4, 5, 4, 4, 5, 5, 4, **4.92399999999999, 5**

4, 2, 3, 4, 4, 5, 3, 2, 1, 4, 3, 2, 1, 2, 4, 3, 2, 5, 4, 3, 2, 2, 4, 3, 3, 5, 4, 5, 5, 3, 3, 4, 4, 2, 5, 4, 5, 1, 3, 4, 5, 4, 4, 4, 4, 3, 4, 4, 5, 4, 1, 1, 2, **4.989, 5**

5, 2, 1, 3, 5, 4, 5, 1, 5, 5, 5, 5, 1, 3, 3, 4, 4, 4, 1, 5, 4, 3, 4, 4, 3, 2, 5, 3, 3, 5, 5, 3, 3, 1, 5, 5, 5, 4, 5, 5, 5, 3, 3, 1, 5, 1, 5, 4, 1, 3, 4, 4, 4, **5.927, 6**

4.2. Results for Factor Dataset

4.2.1. Demographic Analysis

The demographic analysis performed on the factor dataset describes the percentage frequency of the responses of the respondents. This frequencies are described in Table 13

Table 12 &13: Demographics for Factors

Factors	Frequencies (%)				
	Strongly Disagree	Disagree	Undecided	Agree	Strongly Agree
SSH	0.7	6.5	33.9	46.8	12.2
SF	11.5	18.7	19.3	36.6	13.9
SATD	0	2.4	19	58.9	19.7
SAT	0	5.1	26.4	63.8	4.7
ST	4.1	8.8	14.9	29.8	42.4
LAT	0	0.7	49.1	34.9	15.3
LTS	0	2.0	37.3	48.5	12.2
LCS	0	2.7	31.2	57.3	8.8
LA	6.1	8.1	24.8	47.8	13.2
LD	0.7	8.1	13.2	54.9	23.1
OH	12.5	33.3	32.8	7.8	13.6
OE	0	4.7	26.1	48.2	21
OB	0	2.7	21	44.4	31.9
UF	0	7.5	14.5	53.6	24.4
UCP	6.8	18.3	24.4	29.5	21
ULT	4.7	16.3	18	32.2	28.8
UTA	12.5	19	26.8	22	19.7
FI	0.7	2.7	28.8	40.3	27.5
FS	23.7	25.8	26.8	13.5	10.2
FPE	3.4	13.2	11.5	47.2	24.7
FPG	2.0	7.2	33.5	48.5	8.8

	A	B	C	D	E	F
GRADE	31.2	28.8	35.6	2.4	2	0

ISSN 1512-1232

4.2.2. Descriptive Analysis

Mean and standard deviation are the descriptive analysis used in this study to analyze the factors being investigated.

Table 14: Descriptive Statistics for Extracted Factors

Descriptive Statistics					
	N	Minimum	Maximum	Mean	Std. Deviation
GRADE	295	2.00	6.00	4.8475	.96210
SSH	295	3.00	15.00	11.1966	2.19197
SF	295	3.00	15.00	8.8746	3.32091
SATD	295	6.00	15.00	10.9898	2.00592
SAT	295	4.00	20.00	14.3831	3.06268
ST	295	3.00	15.00	11.3085	3.42705
LAT	295	4.00	19.00	12.0814	3.35945
LTS	295	8.00	20.00	13.4169	2.70155
LCS	295	6.00	18.00	12.9831	3.07995
LA	295	3.00	15.00	9.6339	2.91633
LD	295	4.00	20.00	14.2610	3.43793
OH	295	3.00	15.00	7.5085	3.29178
OE	295	5.00	15.00	10.4847	2.26304
OB	295	5.00	20.00	15.0847	3.05554
UF	295	4.00	20.00	14.7424	3.21900
UCP	295	3.00	15.00	9.5051	3.45513
ULT	295	3.00	15.00	10.2746	3.52771
UTA	295	3.00	15.00	8.4780	3.45013
FI	295	3.00	15.00	10.6508	2.32777
FS	295	3.00	15.00	7.1831	3.43557
FPE	295	3.00	15.00	10.3119	3.08127
FPG	295	6.00	15.00	11.3322	2.27635
Valid N (listwise)	295				

4.3. Student Performance Model

Details of the Correlation and Regression analysis as enacted on the dataset of factors extracted from the variable set is as follows:

4.3.1. Correlation Analysis

Correlation analysis carried out on the extracted factors showed that only six (6) out of the twenty-one (21) factors been investigated were found significant to student performance in programming courses. Factors such as SSH, SATD, LCS, LA, LD, OH, OB, OE, FS, FPE, FPG, UCP, UTA, UF and FI were found to be non-significant to the study while factors such as SF, SAT, ST, LAT, LTS and ULT were found significant. These bolded statistically relevant factors all have a correlation value which is greater than or equal to 0.25 which is the statistically acceptable benchmark of correlation relevance for a variable.

Table 15: Coefficient Table of Factors

	GRADE	*SSH*	*SF*	*SATD*	*SAT*	*ST*	*LAT*	*LTS*	*LCS*

ISSN 1512-1232

GRADE	1.000								
SSH	0.145	1.000							
SF	**-0.384**	-0.146	1.000						
SATD	0.177	-0.119	-0.128	1.000					
SAT	**0.327**	0.089	-0.381	0.196	1.000				
ST	**-0.289**	-0.013	0.317	0.068	-0.005	1.000			
LAT	**0.271**	0.311	-0.393	-0.220	0.159	-0.331	1.000		
LTS	**-0.309**	0.197	0.344	-0.248	-0.456	0.380	-0.122	1.000	
ULT	**-0.273**	-0.088	0.438	0.029	-0.263	0.374	-0.480	0.378	-0.450
UTA	0.051	0.099	0.070	0.070	-0.028	0.061	0.253	0.219	0.398

Student Fear and Perception (SF) had a correlation value of -0.384 which implies that a positive perception of students who offered programming courses gave their performances a facelift. Hence the more control a student has over the fear factor that emanates predominately from programming courses the better performance of such a student.

Tutorials and Extra Classes (ST) had a significant correlation coefficient of 0.289 in the negative direction which denoted that students who had attended a group discussion in search of better understanding might end up being rattled and confused. Invariably, this suggests that the higher the tutorial or group discussion of a student, the higher the risk of having a decrease in the performance of student in programming courses as connoted by the correlation coefficient.

The lecturer’s Teaching Style (LTS) also correlates negatively with the performance of students as it has a relevant correlation value of -0.309. This conveys that the lecturer’s teaching technique of a programming course doesn’t necessary have to excellent to achieve a better student academic performance in programming courses.

Lecture time (ULT) had a correlation coefficient of 0.273 in the negative direction. This insinuates that a favorable or conducive lecture time is significant to the academic performance of students who offer programming courses and that the more favorable or conducive the lecture time is the better the performance of student in programming courses.

Student attitude (SAT) had a correlation value of 0.327 in the positive direction, suggesting that the attitude of students is directly proportional to their academic performance in programming courses. Hence the more positive the attitude of a student is to learning programming the better the performance of such a student while a decline in the attitude of a student will directly lead to a decline in academic performance.

Lecturer’s attitude (LAT) had a relevant correlation value of 0.271 indicating that there exists a strong relationship between the attitude of programming course lecturers and the performance of their students. This suggests that a more positive attitude from programming courses lecturers would cause students offering their courses perform better.

It is however important to note that correlation measures the magnitude of linear relationship between the student’s performance and the predictors as stated in Section 3.6.1 of this study.

Hence, the weak correlation of a variable does not depict its insignificance.

4.3.2. Regression Analysis

Multiple regression was used to examine the association between the factors affecting the academic performance of students in programming courses. This implies an analysis of the relationship between the criterion (dependent variables) and the predictors (independent variables).

ISSN 1512-1232

The factor datasets was subjected to this analysis in a bid to evaluate the magnitude of the relationship that exist between student academic performance and each of the extracted factors.

Several approaches were gainfully employed in an effort to obtain a dutiful and accurate quantification of the relationship that exists between the dependent and independent variables. These approaches were bent on defining the magnitude of influence of factors based on a defined scope. Hence, three model scopes were defined as thus; Hybrid Model, Controllable Model and the Uncontrollable Model.

4.3.2.1. Hybrid Student Performance Model (HSPM)

This model applies the influence of all the possible factors (Controllable and Uncontrollable) on student performance without giving preference to any perspective whatsoever thereby providing a means of estimating student performance with regards to all applicable and significant factors. Regression analysis of the significant factors was then evaluated after the exclusion of all factors with a p-value

Table 16: HSPM Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.634a	.402	.381	.75674

a. Predictors: (Constant), FPG, FPE, SF, OE, UF, SATD, OH, SAT, FI, FS

The newly developed model of ten (10) significant factors had a R-Square value of 0.381 explaining about 38.1% of the students' performance. To determine the overall strength of the model presented in Table 16, F-Statistics test was carried out.

Table 17: HSPM ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	109.502	10	10.950	19.122	.000[a]
	Residual	162.634	284	.573		
	Total	272.136	294			

a. Predictors: (Constant), FPG, FPE, SF, OE, UF, SATD, OH, SAT, FI, FS

b. Dependent Variable: Grade

As presented in Table 17, a valid F-Statistics test value of 19.122 was obtained. This F-Statistics test value describes the model as very strong.

Table 18: HSPM Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	T	Sig
		B	Std. Error	Beta		
1	(Constant)	5.088	.513		9.917	.000
	SF	-.077	.015	-.264	-5.071	.000
	SATD	.085	.025	.178	3.417	.001
	SAT	.059	.017	.189	3.423	.001
	OH	.049	.017	.167	2.838	.005
	OE	.077	.020	.181	3.863	.000
	UF	-.060	.015	-.201	-3.957	.000
	FI	.060	.023	.146	2.613	.009

ISSN 1512-1232

FS	-.068	.017	-.242	-4.006	.000
FPE	-.057	.015	-.182	-3.811	.000
FPG	-.107	.023	-.253	-4.711	.000
a. Dependent Variable: Grade					

$$SP = 5.088 - 0.077SF + 0.085SATD + 0.059SAT + 0.049OH + 0.077OE - 0.060UF + 0.060FI - 0.068FS - 0.057FPE - 0.107FPG \quad (2)$$

As described in Table 18 based on the Beta coefficients of the result of regression analysis, Student Fear and Perspective (SF) causes 7.77% variation of student performance in programming courses in the negative direction, Student Attendance (SATD) causes 8.5% variation in the academic performance of students, 5.9% variation in academic performance in programming courses is attributed to the Student Attitude (SAT) while 4.9% variation is caused by the Health (OH) factor of students offering programming courses and a variation of 7.7% is caused by the Electricity factor. University Factors (UF) causes 6.0% variation in student performance in the negative direction, family income also causes 6.0% variation in student performance but in the positive direction while a negative direction variation of 6.8%, 5.7% and 10.7% were obtained for Family Stress (FS), Parent Educational Level (FPE) and Parental Guidance (FPG) respectively.

The developed model was then validated using a series of randomly selected respondent data. A few of the model validation instances is as presented thus;

SF	SATD	SAT	OH	OE	UF	FI	FS	FPE	FPG	Predicted	Original
10	10	10	6	8	14	12	8	12	12	4.036	4
9	13	13	7	13	9	13	8	14	8	5.653	6
11	12	11	14	10	17	10	9	5	11	4.872	5
8	13	10	14	11	15	10	14	8	7	5.243	5
12	11	14	14	14	16	10	13	14	12	4.363	4
3	13	16	8	7	14	10	6	11	12	5.278	5
11	8	11	7	11	13	7	3	12	11	4.335	4
3	11	19	8	13	13	14	5	11	12	6.115	6
12	15	16	3	7	20	15	3	14	15	4.162	4
9	11	13	14	8	15	10	9	11	6	5.218	5

4.3.2.2. Student Controllable Performance Model (SCPM)

This model as against the hybrid model presents a student perspective of student performance in programming courses by considering only the factors that are intrinsic to and can be controlled by students. Hence, the performance of students was determined and predicted based on factors which are peculiar to the students and the students alone.

Table 19: SCPM Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.503a	.253	.240	.83853
a. Predictors: (Constant), ST, SAT, SSH, SATD, SF				

The five (5) factors intrinsic to students alone are ST, SAT, SSH, SATD and SF. As evidenced by Table 19, the developed model had a R Square value of 0.253 which implies that the model explains 25.3% of the students' performance.

ISSN 1512-1232

Table 20: SCPM ANOVA

Model	Sum of Squares	Df	Mean Square	F	Sig.
1 Regression	68.929	5	13.786	19.606	.000[a]
Residual	203.207	289	.703		
Total	272.136	294			

a. Predictors: (Constant), ST, SAT, SSH, SATD, SF

b. Dependent Variable: Grade

The ANOVA analysis presented in Table 20 shows that the model is very strong since the F-Statistics test value is 19.606. Also the model was regarded as fit since it as a p-value of less than 0.05.

Table 21: SCPM Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	T	Sig
		B	Std. Error	Beta		
1	(Constant)	3.837	.512		7.500	.000
	SSH	.049	.023	.111	2.134	.034
	SF	-.056	.017	-.194	-3.268	.001
	SATD	.067	.025	.140	2.645	.009
	SAT	.068	.018	.215	3.829	.000
	ST	-.066	.015	-.235	-4.315	.000
a. Dependent Variable: Grade						

All the factors considered in this model were found to be significant to determining the academic performance of student programming courses. The model is then presented as thus;

$$SP = 3.837 + 0.049SSH - 0.056SF + 0.067SATD + 0.068SAT - 0.066ST \quad (3)$$

As evidenced in Table 21, on the basis of Beta coefficients the result of regression analysis for study hours (SSH) in the model causes positive 4.9% variation in student academic performance in programming courses.

Student fear and perception causes a negative 5.6% variation in student performance in programming courses while student attendance causes a positive 6.7% variation and student attitude (SAT) causes a positive 6.8% variation in student performance. Finally, tutorial (ST) causes a negative 6.6% variation in the performance of students in programming courses.

Instances of the model validation result is as presented thus:

SSH	SF	SATD	SAT	ST	Predicted	Original
11	10	10	10	12	4.374	4
12	13	11	15	11	4.728	5
13	5	13	13	10	5.289	5
11	11	12	11	7	4.85	5
10	8	13	10	9	4.836	5
10	3	13	16	12	5.326	5
14.	11	8	11	14	4.267	4
14	3	11	19	3	6.186	6
10	9	11	13	8	4.916	5

ISSN 1512-1232

10	10	6	14	13	4.263	4

4.3.2.3. Student Uncontrollable Performance Model (SUPM)

Uncontrollable performance model considers all perspectives that are not within the control of students. This includes factors that are intrinsic to the lecturers, university, health, family and other factors. Hence presenting a model from which the performance of the students in programming courses can predicted considering only factors that cannot be directly influenced by them (students).

Table 22: SUPM Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.513[a]	.264	.251	.83270
a. Predictors: (Constant), FPG, OE, UF, LTS, FS				

As described in Table 22, the model presented had a R Square value of 0.264, implying that 26.4% of student performance in programming courses can be explained by this model through the five (5) factors considered.

Table 23: SUPM ANOVA

Model		Sum of Squares	Df	Mean Square	F	Sig.
1	Regression	71.748	5	14.350	20.695	.000[a]
	Residual	200.387	289	.693		
	Total	272.136	294			

a. Predictors: (Constant), FPG, OE, UF, LTS, FS

b. Dependent Variable: Grade

As evidenced in Table 23, the derived model had a significant P-Value of 0.000 which is less than 0.05 hence the model can be concluded to be adequately fit. Furthermore, the F-Statistics test which indicates the strength of the model had a value 20.695, indicating that the model is very strong.

Table 24: SUPM Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	T	Sig
		B	Std. Error	Beta		
1	(Constant)	7.800	.500		15.615	.000
	LTS	-.127	.020	-.357	-6.438	.000
	OE	.102	.022	.239	4.700	.000
	UF	-.066	.016	-.221	-4.146	.000
	FS	-.048	.016	-.170	-2.956	.003
	FPG	-.088	.023	-.208	-3.753	.000
	a. Dependent Variable: Grade					

ISSN 1512-1232

The presented model is as follows:

$$SP = 7.800 - 0.127LTS + 0.102OE - 0.066UF - 0.048FS - 0.088FPG \qquad (4)$$

As evidenced by Table 24 on the basis of Beta coefficients, the result of regression analysis for Lecturers' Teaching Style (LTS) causes 12.7% variation in student performance in the negative direction. Electricity factor (OE) causes 10.2% variation in the performance of students in programming courses while University Facilities (UF), Family Stress (FS), Parental Guidance (FPG) causes 6.6%, 4.8% and 8.8% variation in student performance in the negative direction respectively.

Instances of the model validation result is presented thus:

LTS	OE	UF	FS	FPG	Predicted	Original
18	8	14	8	12	3.966	4
12	13	20	10	11	4.834	3
17	12	15	11	8	4.643	5
13	13	9	8	8	5.793	6
13	13	9	7	12	5.489	5
13	11	15	14	7	4.993	5
17	12	17	13	10	4.239	4
16	14	16	13	12	4.46	4
15	14	14	9	15	4.647	5
12	7	14	6	12	4.722	5

4.4. Hypothesis Testing

After evaluation for the validity or significance level of the factors on which the proposed hypotheses of this study are based, the following are the propositions made.

The P-value for the gadgets variable is 0.000 which is less than the benchmark value of 0.05. This implies that the hypothesis has a 100% probability of occurrence. Therefore, we fail to reject the hypothesis since there is a statistically proven significance between gadget and student performance. Hence, practicing with a personal computer is significant to the determination of student performance in programming courses. Attending an introductory programming class was evaluated as significant at a P-value of 0.016. As a result, there is 98.4% probability that the hypothesis will occur and 1.6% probability that it will not occur and as such was accepted. This indicates that attending introductory classes is significant in the determination of students' academic performance in programming courses. ASSIGN was found to be significant at P-Value of 0.000, which denotes that the probability that it is 100% significant. Therefore, attempting to solve assignments personally is grossly significant to the academic performance of students who offer programming courses. The P-Value of 0.067 of the PHY variable indicates that there is no significant relationship between a strong background in physics and students performance in programming. Hence, thehypothesis is rejected. MTH variable had a P-Value of 0.776 which indicates that there is no significance between a strong background in mathematics and academic performance in programming courses. However, AGE variable was significant at a P-Value of 0.00, indicating the existence of a significant relationship between the age of students and their performance in programming courses, so, the hypothesis was accepted. GENDER variable had a P-Value of 0.002 which indicates a significant relationship between the gender and performance of a student in programming courses and as such the hypothesis was accepted. The DEPT variable had a significant P-Value of 0.000. This implies that the domicile department requirement is

ISSN 1512-1232
significantly related to the performance of students in programming courses, sequel to which the hypothesis was accepted.

4.5. Comparative Evaluation of the Student Performance Models

The presented models conforms with the already existing body of knowledge in that the positive significance of factors such as Student Attendance (SATD), Student Attitude (SAT), and the negative significance of factors such as Family Stress(FS) were in tune with the model presented by Hijazi and Naqvi (2006) and that presented by Irfan and Shabana (2012). However probably due to the prevalence of geographical influences or difference in variable coding, factors such as Student Study Habit (SSH), University Facilities (UF), Family Proper Guidance (FPG) and Family Income (FI) had a varying significance.The models also took factors such as Lecturers' Teaching Style (LTS), Health (OH), Electricity (OE), Parental Education (FPE), Student Fear and Perception (SF), and Tutorials and Extra Classes (ST) which have not been duly considered by authors into consideration.

5. Conclusion

This study was conducted to explore the factors affecting the academic performance of undergraduates in programming courses and develop a predictive model with which the performance of students can be improved. The research was conducted on a sample of students who offered PASCAL, QBASIC or Java between 2011 and 2016 within the Federal University Oye-Ekiti, Ekiti State, Nigeria. The statistical (SPSS) approach was gainfully employed to the analysis of the retrieved data from 295 respondents. Using the appropriate statistical techniques and tools, findings showed that the attitude of students, the fearful perception of programming by students, tutorials, lecturers' attitude, lecturers' teaching style and the lecture hour cumulatively have a strong correlation with the performance (grade) of the students in programming courses while factors such as erratic power supply, university facilities, student health, students attendance in lectures and a few other factors were significant to the performance of students in programming courses. The multi-factor predictive models developed in this paper offer some cost saving benefits, improved and effective decision making enhancement features. However, future works could develop generalized predictive models to evaluate students' performance in each of the three (3) systemic levels of education in Nigeria or other similar educational systems in any developing economy. Within the tertiary education system, generalized models to measure the performance of students in all state, federal and private institutions in Nigeria could also be developed.

Appendix

Questionnaire

SECTION ONE

x_{77} What department are you: …………………………………………………

Gender: ○ Male ○ Female

x_{76} Level of study: ………..

x_{79} Programming language being evaluated (Tick one):

○ Q-Basic ○ Pascal ○ Java

x_{80} What was your grade in the course selected in (4) above: ....…………...

x_{78} How old were you then: ○ Below 16 ○16-19 ○20-25 ○26-30 ○above 30

SECTION TWO

Please tick the option that best describe your opinion about these expressions.

ISSN 1512-1232

Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
5	4	3	2	1

S/N	Expressions	5	4	3	2	1
x_1	I had enough time to study programming					
x_2	Studying before attending a class aided my assimilation during programming classes.					
x_3	Studying programming was never a wasted effort					
x_4	Programming sounded very scary					
x_5	I was always nervous during programming classes					
x_6	I was always nervous during programming examinations					
x_7	I attended programming classes regularly					
x_8	Blending in after missing a class was very easy					
x_9	I was very serious with programming classes					
x_{10}	I believed I could understand the programming course					
x_{11}	I had interest in programming beyond class level					
x_{12}	Programming was not confusing and did not cause headache					
x_{13}	Programming is relevant to my pursuit					
x_{14}	Group discussions helped me to understand programming					
x_{15}	Attending programming tutorials was very helpful					
x_{16}	Programming courses tutorials helped me so much					

SECTION THREE

S/N	Expressions	5	4	3	2	1
x_{17}	Motivation of programming lecturers encouraged my commitment towards learning programming					
x_{18}	Programming language lecturers helped me develop interest in programming					
x_{19}	Programming languages lecturers were never partial in their dealings with students					
x_{20}	Programming lecturers were friendly during lectures					
x_{21}	Programming language lecturers enforced discipline during their lectures					
x_{22}	Programming languages lecturers were too serious during lectures					
x_{23}	Teaching methods and styles of programming lecturers inhibited lecture clarity					
x_{24}	Programming language lecturers wasted time on matters with less relevance in class					
x_{25}	Programming language lecturers were always clear, precise and communicates understandably					
x_{26}	Programming language lecturers made use of enough relevant instructional materials					
x_{27}	Programming language lecturers delivered course contents well and to my understanding					
x_{28}	Programming language lecturers were very clear and explicit					
x_{29}	Programming language lecturers didn’t miss classes					
x_{30}	Programming language lecturers attended to me whenever I had difficulties with their course(s)					

ISSN 1512-1232

x_{31}	Programming lecturers were always available					
x_{32}	Programming course lecturers allowed students to ask questions and take time to explain					
x_{33}	Programming course lecturers came to class fully prepared					
x_{34}	Programming languages lecturers spent extra time to explain things during class					
x_{35}	Programming language lecturers usually came early to class					

SECTION FOUR

S/N	**Expressions**	5	4	3	2	1
x_{36}	I fell sick quite often					
x_{37}	Prolong usage of computer caused me headache					
x_{38}	I took a few compulsory medications frequently					
x_{39}	It was difficult to charge my computer even within the campus					
x_{40}	Erratic power supply reduced the effectiveness of my practice					
x_{41}	Consistent power supply helped me in programming courses					
x_{42}	I had a good background in physics					
x_{43}	I had a good background in mathematics					
x_{44}	I had a good background in English					
x_{45}	Strong background in Physics and Mathematics helped me in programming					

SECTION FIVE

S/N	**Expressions**	5	4	3	2	1
x_{46}	Absence of accessible ICT facilities inhibited my programming performance					
x_{47}	The environment where we had programming lectures was not conducive					
x_{48}	Lack of computer programming facilities disrupted clear understanding of programming lessons					
x_{49}	The school library was not equipped with materials relevant to programming					
x_{50}	Large class population disrupted my concentration during programming lectures					
x_{51}	Population of students offering programming courses debarred my commitment to learning					
x_{52}	Effectiveness of the programming lecturers' teaching was reduced by huge programming class population.					
x_{53}	Programming lectures were scheduled after an equally tiring lecture					
x_{54}	Programming courses were scheduled to non-conducive times					
x_{55}	We had programming classes at unfavorable times					
x_{56}	Programming lecture theatres were equipped with audio-visuals and learning aids					
x_{57}	Programming courses were analyzed clearly to sight					
x_{58}	I had a visual understanding of what the programming lecturer was implying					

SECTION SIX

S/N	Expressions	5	4	3	2	1

x_{59}	Expensive cost of living did not affect my performance in programming classes					
x_{60}	My family could afford to buy enough programming textbooks					
x_{61}	My family sponsored my academic pursuit					
x_{62}	Quarrel between family members is normal					
x_{63}	I had to travel to settle quarrels within my family					
x_{64}	Quarrel between my family members escalates a times					
x_{65}	My father is familiar with computers					
x_{66}	My mother is familiar with computers					
x_{67}	My parents are well educated					
x_{68}	My parent would want me to offer programming courses					
x_{69}	I received educational advices from family members often					
x_{70}	My family believed that a proper study will help me in programming courses					

SECTION SEVEN

Please tick the option that best describe your opinion about these questions.

S/N	Questions	Yes	No
x_{71}	Did you attend the introductory classes?		
x_{72}	Did you practice programming with your own personal computer?		
x_{73}	Did you attempt your programming assignment by yourself?		
x_{74}	Do you like mathematics?		
x_{75}	Do you like physics?		

REFERENCES

[1] Gomes, Anabela& Mendes (2007). Learning to program - difficulties and solutions. In International Conference on Engineering Education (118 – 124). Coimbra, Portugal.

[2] Justin & Dmitry (2015). Modeling factors influencing mathematics learning and performance in Tanzanian secondary schools. Asian journal of mathematics and applications, 2015, 1 – 11.

[3] Irfan, I. &Shabana, N.K. (2012). Factors affecting students' academic performance. Global Journal of Management and Business Research, 12, 9, 189 – 200.

[4] Akinola, O.S. &Nosiru, K.A. (2014). Factors influencing students' performance in computer programming: A Fuzzy set operations approach. International Journal of Advances in Engineering and Technology, 7, 4, 1141 – 1149.

[5] Mustafa, B. (2013). Attitude, Gender and Achievement in Computer Programming. Middle-East Journal of Scientific Research, 14, 2, 248 – 255.

[6] Ogbogu, C.O. (2014). Institutional Factors Affecting the Academic Performance of Public Administration Students in a Nigerian University. Canadian Center of Science and Education, 3, 2, 171 – 177.

[7] Masura, R., Shahrani, S., Rodziah, L., Faezah, M.Y., Faridatul, A.Z. &Rohizah A.R. (2012). Major problems in basic programming that influence student performance.Procedia - Social and Behavioral Sciences, 59, 287 – 296.

[8] Kofi, A.S., John, K.A. & Prince, Y.O. (2013). Causes of Failure of Students in Computer Programming Courses: The Teacher – Learner Perspective. International Journal of Computer Applications, 77, 12, 27 – 32.

ISSN 1512-1232
[9] Prince, Y.O., Kofi, A.S., John, K.A. and Charles, A. (2013). Performance of Students in Computer Programming: Background, Field of Study and Learning Approach Paradigm. International Journal of Computer Applications, 77, 12, 17-21.
[10] Colin (2006), Modeling Student Performance in a Tertiary Preparatory Course (Doctoral Dissertation)
[11] Shaymaa, Tsunenori, Kazumasa& Sachio (2015). A Predictive Model to Evaluate Student Performance. Journal of Information Processing, 23, 2, 192-201.
[12] Hijazi& Naqvi (2006). Factors Affecting Students' Performance – A Case of Private Colleges. Bangladesh e-journal of Sociology, 3, 1, 1 – 10.
[13] Siti, Kazifah, &Nurhafizah (2015). Statistical Analysis on the Determinants of Students' Academic Achievement: A Study in UITM JOHOR. In Proceeding of the 3rd Global Summit on Education (271 – 282) Kuala Lumpur, Malaysia.
[14] Chermahini (2013). Learning Styles and Academic Performance of Students in English as a Second - Language Class in Iran. Bulgarian Journal of Science and Education Policy, 7, 2, 322 – 333.
[15] Varalakshmi, V., Suseela, N., Sundaram, G.G., Ezhilarasi, S., &Indrani, B. (2005). Correlation. Statistics Higher Secondary School – First Year (191 – 193; 218 – 229);
[16] Amiroh&Farinda (2016). Factors Affecting the Academic Performance of Executive Diploma Students: The case of University of Malaya Center for Continuing Education.

Article received: 2017-11-24