International Journal of Education and Development using Information and Communication Technology (IJEDICT), 2019, Vol. 15, Issue 1, pp. 35-54

Instructional design enabled Agile Method using ADDIE Model and Feature Driven Development method

Christian Misobi Budoya, Mussa M. Kissake and Joel S. Mtebe
University of Dar es Salaam, Tanzania

ABSTRACT

Software developers nowadays are adopting agile methods to overcome challenges faced by traditional methods in developing software. However, agile methods lack instructional design to take care of learning needs required in designing software for supporting teaching and learning. This study aimed at designing an integrated model using Feature Driven Development Process (FDDP) and Analysis, Design, Development, Implementation, and Evaluation (ADDIE) model to accommodate learning needs at design stage. Participatory action research method and four design strategies including decompositional, compositional, template based, and incremental and evolutionary strategies were used in designing the integrated ADDIE-FDDP Model. A case study of designing Interactive Multimedia Content for Life Skills Education (IMCLSE) was used to test the model, especially its analysis and design phases. A questionnaire was adapted to evaluate the IMCLSE design involving 10 software developers and 65 teachers who teach life skills subjects in 10 primary schools. The results show that software developers and teachers agreed that the integrated FDDP-ADDIE Model was effective in guiding the design process of software that support teaching and learning. Therefore, the integrated FDDP-ADDIE Model can be adopted and used for developing effective and quality learning software.

Keywords: *ADDIE Instructional Design Model, Feature Driven Development Process, Interactive Multimedia Content, Instructional design enabled agile method*

INTRODUCTION

In the process of developing software product, the design stage is very important as it is a stage where a software product is defined and realized (Hong, 2005; Bosch, 2014). It is a bridge between the requirement analysis and implementation stages that serves as an implementation guideline (Lowgren, 1995; Hong, 2005; Sommerville and Addison, 2006). It is also a first stage in software system creation in which quality requirements can begin to be addressed (Hong, 2005; Sommerville and Addison, 2006). Errors made at this stage can be costly (Giddings, 1984; Yang, Hu and Jia, 2008), even impossible to be rectified (Hong, 2005). Design methodologists tend to characterize design as a type of problem solving or decision making (Sommerville and Addison, 2006) in the face of uncertainty (Mathiassen and Stage, 1990), with high penalties for error (Hapter and Tremblay, 2001; Keith, 2002). In order to avoid errors while solving the problem and achieving goals within certain constraints (Aftab *et al.*, 2018), selection and proper application of appropriate and well-established design and development methods are required for systematic guidance (Hong, 2005; Aftab et al, 2018).

Agile methods such as Extreme Programming, Crystal methods, Lean Development, Scrum, and Adaptive Software Development (ASD) are preferred over traditional methods as they are more flexible, iterative, cost effective and delivery time is considered (Qasaimeh, Mehrfard and Hamou-lhadj, 2008; Moniruzzaman and Hossain, 2013). They reduce the software process overheads (like documentation) and concentrate more on code rather than the design (Sunner, 2016) while releasing a working software early and continuously improving it with customers. However, these

methods lack instructional design to address educational needs such as curriculum setting at design stage (Huang, 2005). They cannot support establishing learning objectives, identifying learning instructions, learning outcomes, and creation of subject content (FAO, 2011; Moradmand, Datta and Oakley, 2014). In order to use preferable agile methods in designing interactive multimedia content for learning, it must be combined with a compatible instructional design model so that the learning educational needs are considered during the designing process (Moniruzzaman and Hossain, 2013).

This study aimed at integrating an instructional design model with an agile development method and tested it in designing Interactive Multimedia Content (IMC) for enhancing the provision of Life Skills Education (LSE) at the primary school level in Tanzania. The Analysis, Design, Development, Implementation, and Evaluation (ADDIE) instructional design and Feature Driven Development Process (FDDP) were chosen as the instructional design and agile method respectively, to guide the design process in consideration of educational needs. The integrated model was very effective after testing it in designing the IMCLSE.

Agile Software Development Methods

Agile software development is a group of software development methods which have the following features: lightweight, iterative, cost effective, customer involvement, rapid delivery of quality software product, short design phases, incremental (evolutionary) approaches, and capable of incorporating rapid changes in requirements (Moniruzzaman and Hossain, 2013). The agile methodologies originated in the "Manifesto for Agile Software Development" (www.agilemanifesto.org) which describes the four comparative values underlying the agile position: individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan. Examples of Agile software include the Scrum, Extreme programming (XP), Feature Driven Development (FDD), Adaptive Software Development (ASD), Crystal Methods, and Dynamic System Development (Moniruzzaman and Hossain, 2013). Comparison of agile methods valuing one method over the other in terms of key points, special features and identified weakness are shown in Table 1.

Table 1: *General Features and Comparison of Agile Methods (Moniruzzaman & Hossain, 2013)*

Method	Key Points	Special features	Identified weakness
ASD	Adaptive culture, collaboration, mission-driven component based iterative development	Organizations are seen as adaptive systems. Creating an emergent order out of a web of interconnected individuals	ASD is more about concepts and culture than the software practice
DSDM	Application of controls to RAD, use of time boxing and empowered DSDM teams.	First truly agile software development method, use of prototyping, several user roles: "ambassador", "visionary" and "advisor"	While the method is available, only consortium members have access to white papers dealing with the actual use of the method
XP	Customer driven development, small teams, daily builds	Refactoring - the ongoing redesign of the system to improve its performance and responsiveness too change	While individual practices are suitable for many situations, overall view & management practices are given less attention
SCRUM	Independent, small, self-organizing development teams, 30-day release cycles	Enforce a paradigm shift from the "defined and repeatable" to the "new product development view of Scrum"	While Scrum details in specific how to manage the 30-day release cycle, the integration and acceptance tests are not detailed
FDD	Five-step process, object-oriented component (i.e. feature) based development.	Method simplicity, design and implement the system by features, object modelling	FDD focuses only on design and implementation. Needs other supporting approaches.

As shown in Table 1, among all agile methods, the most object-oriented method is FDD. It is simple to use, its design and implementation are based on object oriented components, and is good for complex and large projects, architectural designing and modelling (Moniruzzaman and Hossain, 2013). Its weaknesses such as lack of clear requirement gathering and lack of instructional design (Awad, 2005; Huang, 2005; Sandhna and Brügge, 2007; Seabra and Almeida, 2015) can be addressed by integrating with ADDIE instructional design. The details of FDD are represented diagrammatically in Figure 1 below.

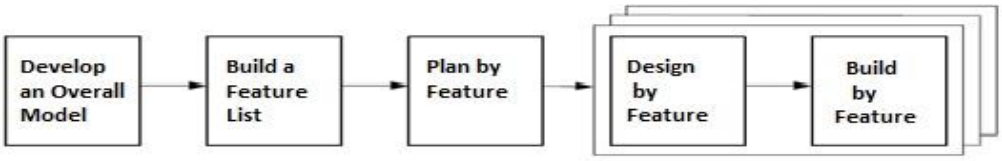

Figure 1: *The simple steps of FDDP (Qasaimehet al, 2008)*

ADDIE Instructional Design Model

According to Aldoobie (2015), the ADDIE model is one of the most common models used in the instructional design field. It helps instructional designers and teachers to create an efficient, and effective teaching design by applying the processes of the ADDIE model on any instructional product (Aldoobie, 2015). ADDIE is an acronym for (1) analyse (2) design (3) develop (4) implement and (5) evaluate (Moradmand, Datta and Oakley, 2014). In this study, the dynamic and flexible ADDIE instructional design models are preferred. Although many instructional design models exist, they all contain five generic phases (Taylor, 2004; Moradmand, Datta and Oakley, 2014; Ahmadigol, 2015). These phases are shown in Figure 2 below.

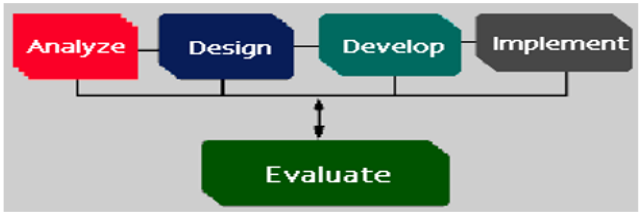

Figure 2: *ADDIE Instructional Design Phases (Taylor, 2004)*

The phases of the ADDIE instructional design model shown in Figure 2 are described below.

- **Analysis phase**: needs analysis, target audience analysis, topic and task analysis (FAO, 2011; Moradmand, Datta and Oakley, 2014).
- **Design phase**: establishing specific objectives; specifying learning activities, identifying instructional or pedagogical strategies, designing learning activities, and creating subject contents or materials (FAO, 2011; Moradmand, Datta and Oakley, 2014).
- **Development phase**: creating and building all content and components based on the design phase, constructing teaching and learning program structure, making the program available on selected media of delivery (FAO, 2011; Moradmand, Datta and Oakley, 2014).

- **Implementation phase**: implementing instructional materials to the real world environment, providing support to users, and using evaluation instruments to investigate the instructional material and programs' values (FAO, 2011; Moradmand, Datta and Oakley, 2014).
- **Evaluation phase**: Evaluating the effectiveness of the instructional materials, tools, and activities; investigating the achievement of learning objectives, impact of teaching and learning process, and identifying changes and modification for future delivery (FAO, 2011; Moradmand, Datta and Oakley, 2014).

ADDIE instructional design models have been used for designing and developing various learning and teaching software (Moradmand, Datta and Oakley, 2014; Ahmadigol, 2015). However, the ADDIE model lacks agile properties which are found in FDDP (Bichelmeyer and Ph, 2005; Jasmy *et al.*, 2014). Therefore, this study has proposed an integrated model between ADDIE instructional design and FDDP to overcome their weaknesses, and utilize their strengths in guiding the development process of interactive multimedia content for learning, which requires incorporation of learning needs at the design stage.

LITERATURE REVIEW

Some studies (Tirumala and Ali, 2016; Nawaz, Aftab and Anwer, 2017; Aftab *et al.*, 2018) have focused on modifying FDDP by adding new features such as the security feature, reuse feature, requirement analysis, and process simplification for small projects (Pang and Blair, 2004; Nawaz, Aftab and Anwer, 2017). However, the lack of instructional design in FDDP for handling learning needs has never been addressed. Generic ADDIE instructional design can handle learning needs (Moradmand, Datta and Oakley, 2014; Ahmadigol, 2015) but it also lacks important agile features (Rawsthorne, 2005; Bahl, 2012).

Abidin and Tho (2018) did a study on designing and developing a hands-on practical physics activity for studying resonance using the ADDIE model. The study aimed at solving lack of interactive learning and teaching methods for difficult resonance experiments in physics, at Sultan Idris Education University in Malaysia. Experiments were used to collect data in which two essential free mobile apps were used: TrueTone (a frequency sound generator) and Advanced Spectrum Analyzer (a frequency meter) to find the fundamental frequency. The experiments produced results with small error between 5 % and 14 % and the flexibility of doing experiments at low cost increased. Zhi et al. (2008) in addressing the lack of adequate learning material and training courses in robotics for high school students in Taiwan, designed and developed multimedia instructional material using the ARCS and the ADDIE design models. After evaluation using triangulation method, the product was found to be effective and the high school students were satisfied.

Jasmy et al. (2014) conducted a study on designing and developing interactive software for teaching and learning physics at a high school in Indonesia using ADDIE and the Life Cycle Model. The Life Cycle Model was used to overcome the weakness of ADDIE by making its process cyclical and repetitive. An experimental group with 38 students learnt using software and a control group with 37 students learnt using a traditional method. The group that engaged in the learning process using software found it more effective and were more motivated than those students that engaged in learning using a traditional method. Ahmadigol (2015) conducted a survey to evaluate the impact of ADDIE instructional design and multimedia on learning key skills of Futsal in Iran. Three groups of 12 experimental and control groups were used. It was found that students trained by multi-media had high mean scores (performance) compared to the students trained using a traditional method. The students trained using the ADDIE model had high mean scores compared to the students learning key skills of Futsal by a traditional method.

Nawaz et al. (2017) conducted a study to simplify the FDD process by introducing requirement elicitation and customized the phases and roles of FDD to make it work effectively for a small project. The Simplified Featured Driven Development model (SFDD) was improved by adding story cards for new requirement elicitation techniques and a testing phase within the iteration to improve the software quality, but it was not evaluated. Likewise, Aftab et al. (2018) conducted a study to compare the Feature Driven Development Process (FDDP) and the Simplified Feature Driven Development (SFDD). The SFDD was proposed to overcome limitations faced by the FDD, such as lack of requirement elicitation, less ability to respond to the changing requirements, no focus on small projects and lack of testing phase. After testing both FDD and SFDD in developing a web based application, it was found that SFDD was more effective than classical FDD in terms of quality, efficiency and effectiveness.

Other studies that have modified some elements of FDDP by integrating with other agile methods include those in (Pang and Blair, 2004; Firdaus *et al.*, 2013; Singh, 2015; Tirumala and Ali, 2016). However, the majority of these studies remained silent about addressing the lack of instructional design faced by FDDP in handling learning needs during designing learning software. They also did not work on weaknesses of the generic ADDIE which lacks agile features. Therefore, this study has proposed to integrate ADDIE instructional design model with FDDP in order to overcome these weaknesses.

METHODOLOGY

Designing Proposed Integrated ADDIE – FDDP Model

Participatory action research method and a mixed approach of well-known design strategies for building up models including decompositional, compositional, template based, and incremental and revolutionary strategies, were adopted in designing the proposed integrated model (Greenwood, Whyte and Harkavy, 1993; Cross, 2000; Hong, 2005). The process of designing the integrated model started by converting the ADDIE model into a template using a template based method which regarded the ADDIE phases as template components. It was followed by decomposition of FDDP processes into simplified sub processes. The simplified sub processes were merged into related components of the template using the composition method and a first version of the model was produced. The participatory action research method was then used as rapid assessment and review of the designed model through discussion involving combined professionals.

Ten software developers and lecturers from University of Dar es Salaam and ten primary school teachers participated in the discussion. Among the software developers, 4 were females and 6 were males, while for teachers, 3 were males and 6 were females. During the discussions it was determined that the design errors include *inconsistencies, inefficiencies, ambiguities, and inflexibilities* and action was taken to reflect the assessment results. An incremental and revolutionary strategy was used in managing the action of incorporating assessment results to improve the next version model. The process was repeated until the free design error and satisfied integrated ADDIE-FDDP model was produced as shown in Figure 3.

The proposed integrated model shown in Figure 3 is described in the three phases below.

- **Analysis phase***:* Containing FDDP sub processes included, *develop overall model*, *build a features list, and plan by feature.* Included are instructional and process activities such as requirement elicitation, needs analysis, target audience analysis, overall model, use case analysis, and coding planning (Pang and Blair, 2004; Arkün, 2008; FAO, 2011).

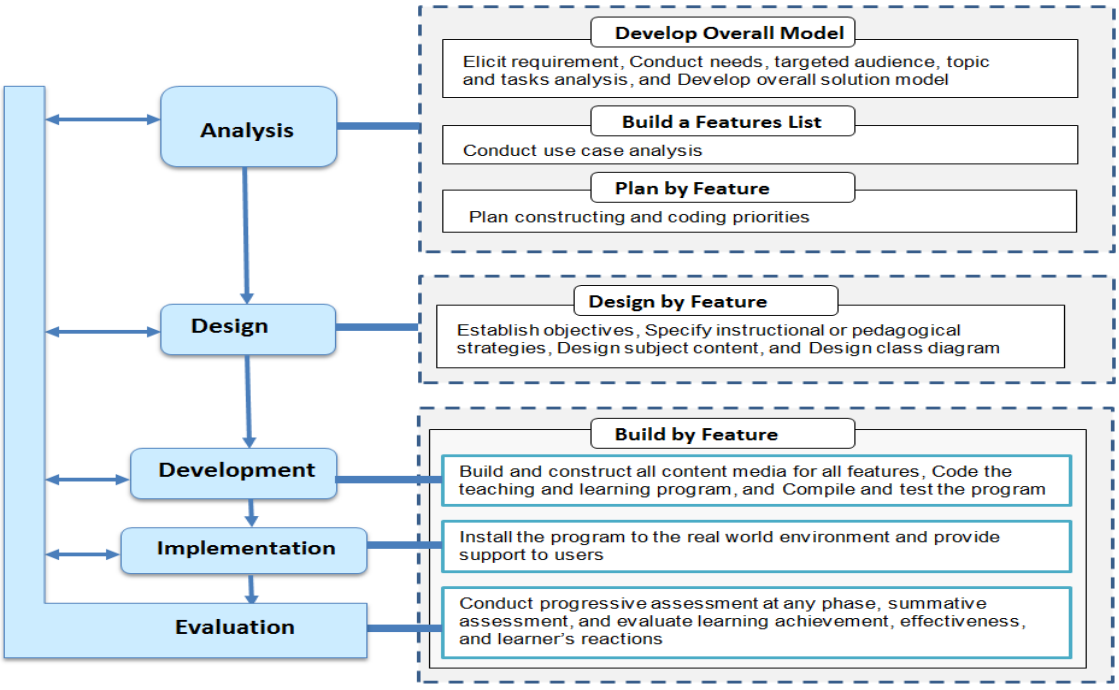

Figure 3: *Proposed Integrated ADDIE Instructional Design Model and FDDP*

- **Design phase***:* Containing FDDP sub process such as design by feature and instructional and process activities such as establish learning objectives, specify instructional strategies, design subject content, and design class diagrams (Sandhna and Brügge, 2007; Rychl, 2008).

- **Development, Implementation and Evaluation phases***:* Containing FDDP sub process, build by feature. Also includes instructional and process activities such as build content features, code all features, integrate and test all features. Other activities for implementation and evaluation are installation, distribution, managing learner activities, conducting formative evaluation, summative evaluation which includes reactions of the learners, learning achievements, and effectiveness of the product (Pang and Blair, 2004; FAO, 2011).

Testing the Integrated ADDIE-FDDP Model

A case study approach was used to test the effectiveness of the integrated ADDIE-FDDP model. The Analysis and Design phases of the integrated model were applied in designing IMCLSE for primary school pupils in Tanzania. In the Analysis phase, interviews and questionnaires were used to collect requirements. The collected requirements were problems faced during the provision of life skills education, targeted audience requirements, learning requirements, topic or content, and learning tasks and outcomes. Twenty interviews were conducted in ten primary schools. Questionnaires were distributed to 65 teachers who teach life skills subjects, and to 407 standard five pupils who passed the Standard Four National Examination in 2017.

The ten primary schools were selected randomly from Dar es Salaam region where many life skills cases were reported (TGSHS, 2017). Needs analysis, targeted audience analysis, topic and task analysis were conducted using content analysis and descriptive statistics analysis where the collected data were translated from Swahili to English, edited, coded, and counted. Tables with

support of Microsoft Excel Program 2010 were used to summarize the results from which graphs were drawn. After the Analysis phase, the Design phase followed in which the content design, functional design, structure design, and class diagram were realized. Finally, the design for IMCLSE was produced. The distributions of the teachers and pupils that participated in the study are shown in Figure 4 and Figure 5 respectively.

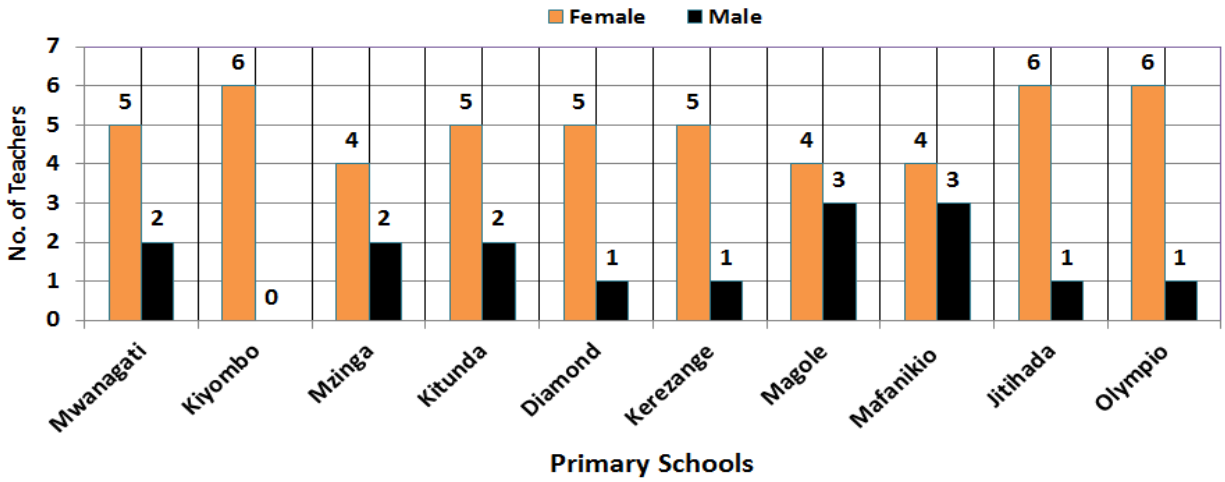

Figure 4: *Distributions of participating teachers per school*

As shown in Figure 4, at least 6 teachers from every primary school participated in the study. The majority of participants were females ranging from 4 to 6 per school.

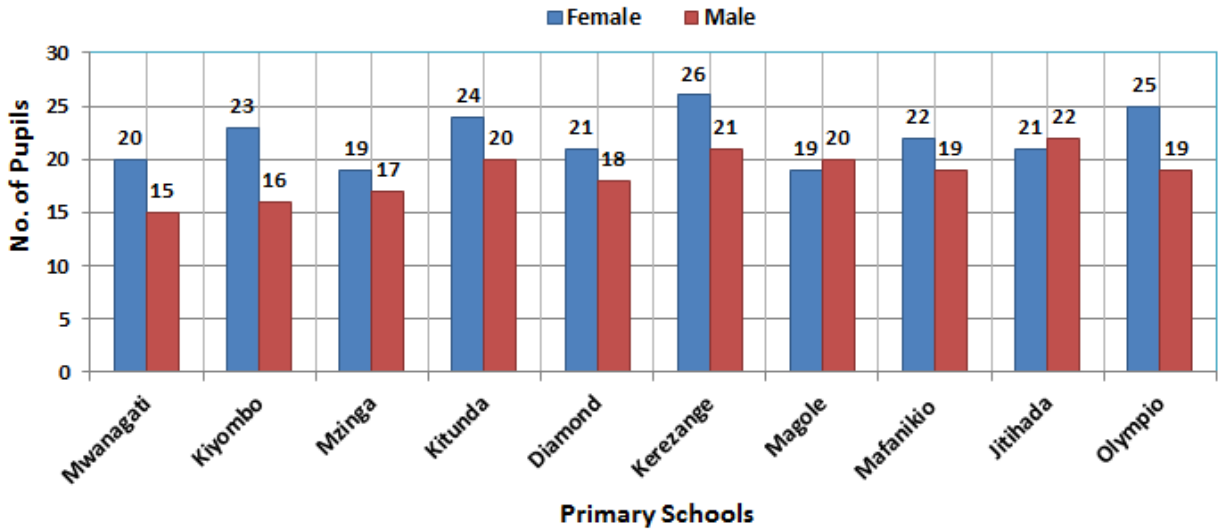

Figure 5: *Distribution of pupils participated in the study*

As shown in Figure 5, at least 35 pupils from every school participated in the study and the majority of them were females (220 females out of a total of 407 pupils).

Evaluating the Effectiveness of the Integrated ADDIE-FDDP Model

A Likert scale based questionnaire was adopted in evaluating the IMCLSE design to see if the integrated ADDIE-FDDP model was effective in guiding the design process. The questionnaire approach was chosen to give the reviewers better defined responsibilities and to make them play a more active role (Parnas and Weiss, 1987; Bertram, 2006). The same ten software developers and lecturers, and ten primary school teachers who participated in designing the integrated ADDIE-FDDP model also participated in evaluating the IMCLSE design. Twenty documentations of IMCLSE design and 20 questionnaires were distributed to the 20 reviewers. The reviewers first studied the documentations before responding to the questionnaire items. The evaluation looked at the quality attributes of software design including the software design objectives and properties for good software design. All 20 questionnaires were returned and analysed quantitatively using statistical analysis.

FINDINGS

Effectiveness of a mixed design strategy in producing integrated ADDIE-FDDP Model

It was found that the participatory action research method and mixed design strategies, including decompositional, compositional, template based, and incremental and evolutionary were very useful in producing the Integrated ADDIE-FDDP model. The template based method helped to set out the ADDIE phases as basic components of foundation for the integrated model. The decompositional method played its role in decomposing FDDP processes into simplified sub processes and the compositional method merged the simplified sub processes into related template components to produce a model. Participatory action research method played a critical role in assessing and reviewing every version of the model, detecting design errors and taking actions to reflect the assessment results. The incremental and evolutionary strategy was used to reproduce improved versions of the integrated model. An iterative mixed or integrated design strategy for integrating two different models was finally realized as shown in Figure 6.

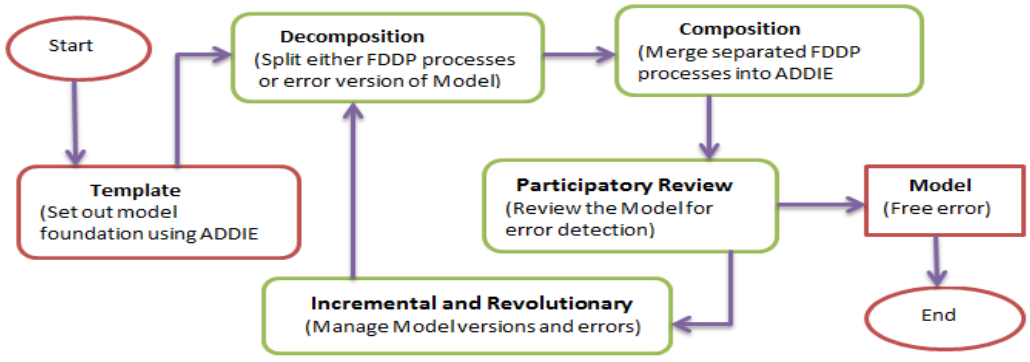

Figure 6: *An iterative mixed or integrated design strategy for integrating two different models*

Lack of teaching materials, professional teachers, and poor teaching methods

The *needs analysis* from the integrated ADDIE-FDDP model was conducted looking at the availability of the basic needs for teaching life skills including teaching materials, professional teachers, and teaching and learning methods. It was found that teaching materials including textbooks, practical and demonstration equipment, and interactive content were not enough,

especially in public schools where some of the subjects did not have even a single textbook for pupils and in some schools one textbook was shared by 5 to 20 pupils. Interactive content was also not available in all schools.

Lack of professional teachers in primary schools was another finding, especially in public schools in which the deficits of teachers ranged from 3 to 12 teachers per school. The inadequate number of professional teachers was further compounded by the increase in enrolment of pupils after the free education declaration in 2015 which led to increases of 2 to 4 streams per class. The size of each stream ranged from 70 to 150 pupils, while the starndard is 45. It is difficult for a teacher to teach the lagre number of pupils above the standard, assess them, and supervise them closely.

It was also found that most of the teachers used traditional methods such as the lecturing style in which a teacher teaches pupils by writing on the blackboard while speaking and explaining. The authority or lecturing style of learning is dominated by the teacher and pupils are less involved. Demonstrating or coaching, facilitating or activity, and hybrid or blended based teaching styles are student-centered and the learning methods which utilize active learning strategies include role play, storytelling, practical and demonstration based learning, debating, and group discussion were very rarely used. Figure 7 shows that the lecture is the most applied teaching style used.

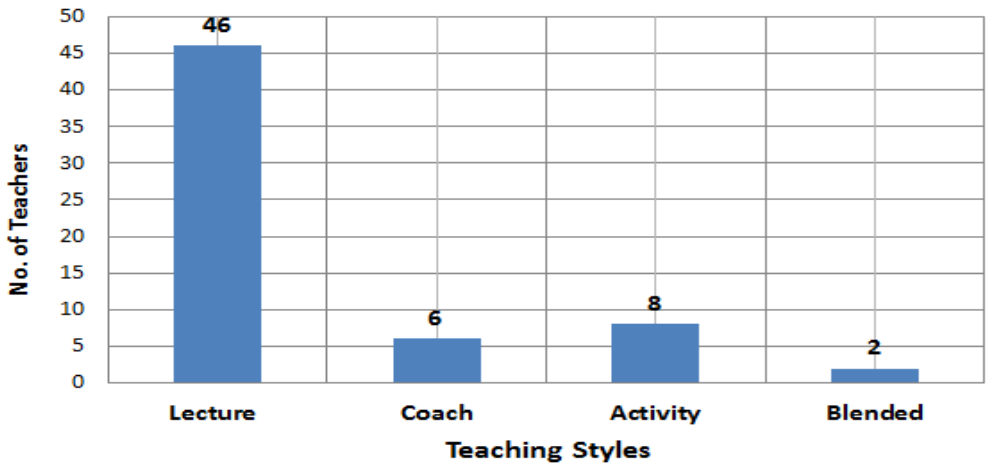

Figure 7: *Teaching styles applied by teachers*

Pupils' access to ICT computing devices and their experiences in using interactive content

It was found that most pupils were playing games at home using their parents' smartphones and computers. Other pupils used to go to business centers where they paid money and played various computer based games as shown in the Figure 8. This is an important finding that informs the users' requirement since the IMC will run on ICT computing devices such as desktop computers, laptops, smartphones, and tablets. This finding was revealed after conducting *targeted audience analysis* from the integrated ADDIE-FDDP looking at pupils' access to ICT computing devices and their experiences using interactive content

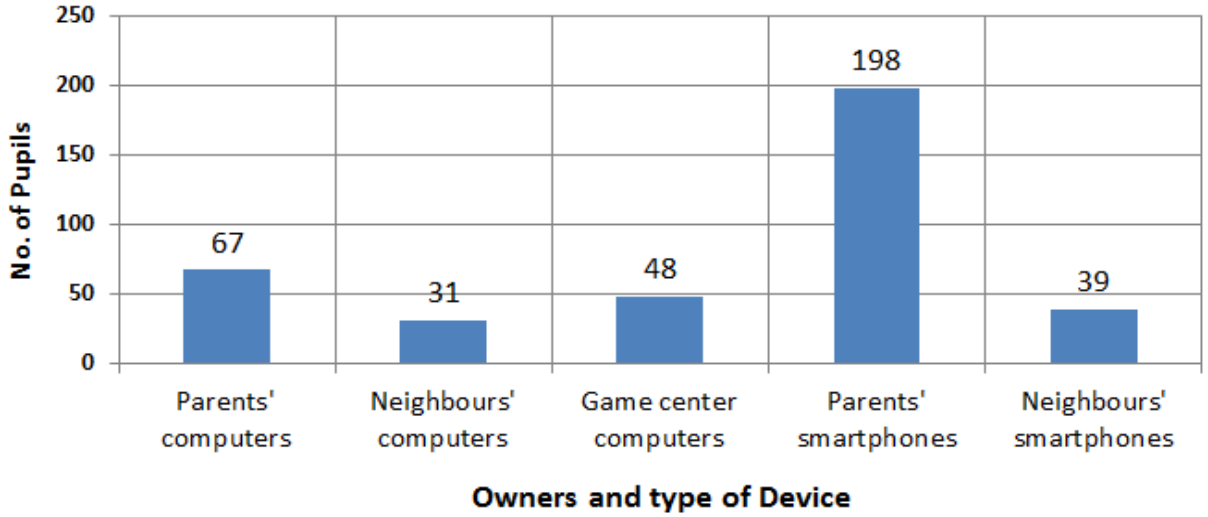

Figure 8: *Pupils with access and experience in using ICT devices and interactive content*

Availability of ICT Infrastructure at primary schools

The *system requirement analysis* from the integrated ADDIE-FDDP model was conducted looking at availability of ICT computing devices and Internet connection which are needed to support operationalization of the IMC. It was found that most of the teachers were using 3G and 4G mobile based Internet connections and two schools had wireless link Internet connection. It was also found that there were a limited number of ICT computing devices in the schools except Mzinga Private School which had a computer laboratory with 12 desktop computers and two public schools which each had 144 tablets available for use.

Availability of challenging topics in teaching the pupils

The *topic analysis* from the integrated ADDIE-FDDP model was conducted to identify the most challenging topic in teaching life skills.

It was found that the most challenging life skills topics which demands active learning that cannot be provided by conventional and traditional teaching and learning methods, according to the curriculum of 2005 (MoEVT, 2005), are *Health Care, Communication, Relationship and Cooperation, Road Safety, Problems and Risk Behaviors, Critical Thoughts, Good Decisions and Plans, and Ethics and Humanity.*

When 65 primary school teachers were asked to select the most challenging topics, the results were as shown in Figure 9 below. The most challenging topic was "*Road Safety*" followed by "Problems and Risk Behaviours".

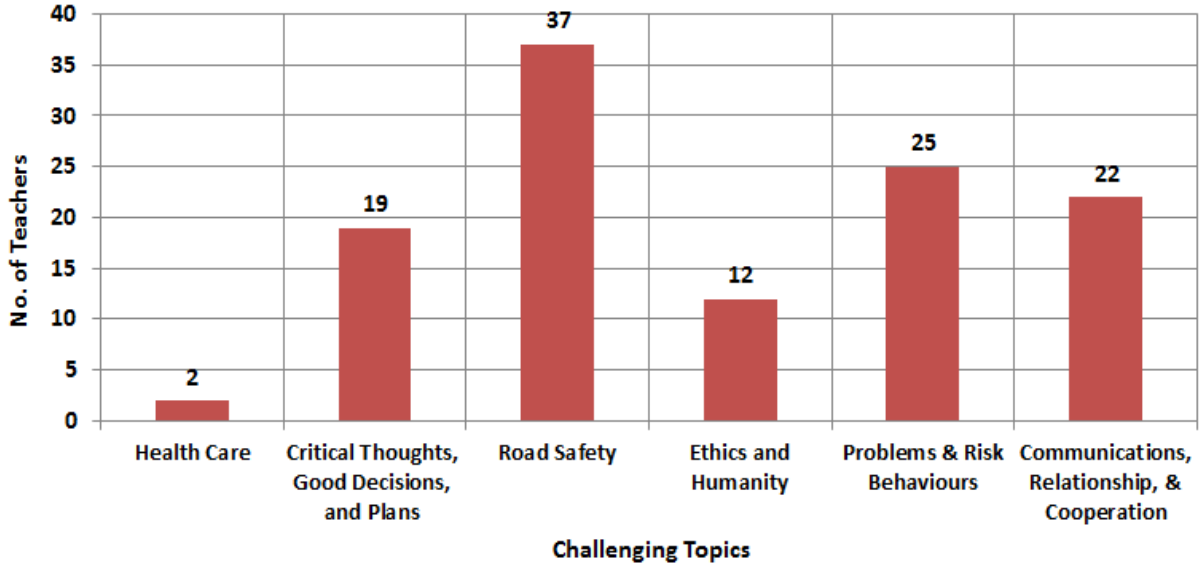

Figure 9: *Number of responses per topic*

Realization of an object solution model

The *Overall Solution Model* from the integrated ADDIE-FDDP model was developed using the identified basic learning tasks. Every task was mapped into a feature. The features were *Introduction* for introducing the subject matter briefly, *Tutorial* for lecturing or narrating new concepts, *and Demonstration* to demonstrate learned concepts. The *Practical was* for providing pupils practical activities concerning learned concepts, while *Assessment* was for assessing and providing feedback. All features were combined to produce an object solution model as shown in the Figure 10

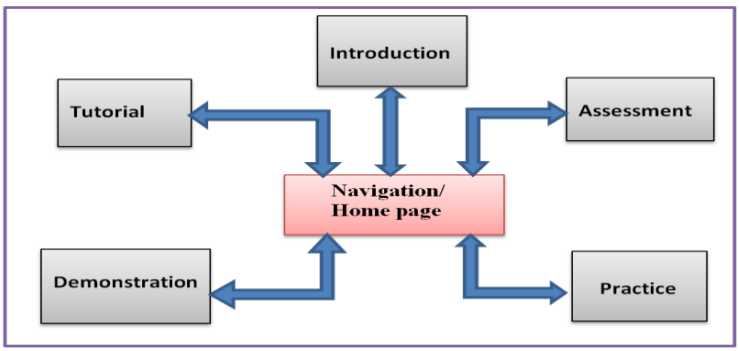

Figure 10: *Featured Object Solution Model*

Activity list per feature and development priorities

The last two steps in the analysis phase of the integrated ADDIE-FDDP model are *Build a Features List* which deals with identification of activities in every feature and the *Plan by Feature* which deals

with priorities of developing the feature. After realizing the featured object solution model which is made up of features, the list of activities per feature, targeted users, and priorities were determined as shown in Table 2 below.

Table 2: *Features, Activities Lists, Targeted User, and Priorities*

Features	Priority	List of activities or actions	Users
Introduction	1	Set introduction environment, Select topic, Play the topic, Navigate to home page, Record all actions, duration	Pupils
Tutorial	2	Set tutorial environment, Select topic, Play the topic, Navigate to home page, Record all actions, duration	Pupils
Demonstration	3	Set demonstration environment, Select topic, Play the topic, Navigate to home page, Record all actions, duration	Pupils
Practice	4	Set practice environment, Select topic, Play the topic, Navigate to home page, Record all actions, duration, Record scores, Display progress performance	Pupils
Assessment	5	Set assessment environment, select player, Display continuous assessment, Display performance, Display area to improve, Display general report, Navigate to home page or exit	Pupils, parents, teachers
Navigation/ Home	6	Select language, select play mode, Register a learner, Select assess feature, Select playable features, Record all actions, Exit/close the application	Pupils, parents, teachers

Establishing subject learning objectives or goals

The *Design by Feature* step from the integrated ADDIE-FDDP model was used to establish subject learning objectives. The content for the selected topic "Road Safety" was divided into subtopics including *types of roads, road crossing signs, and traffic rules and conditions used to guide ways of crossing the road.* Each subtopic or combination of subtopics was used to create subject learning goals which should be achieved by the learners after being trained by IMC on how to cross roads safely. Therefore, on completion of the topic, the pupils should be able to:

a) Identify types of roads available in Tanzania
b) Identify various signs and options that indicate the places to cross the road
c) Find a right place to cross the road depending on the type of road
d) Check the safety condition before crossing the road
e) Make the right decision to cross the road at the right time, place and at safe condition
f) Cross the road in a safe way

Identifying learning instructional or pedagogical strategies

The *Design by Feature* from the integrated ADDIE-FDDP model was used to identify learning activities or sequence of instructions as ordered steps with logical relationships which enable pupils to acquire specified skills, knowledge, attitudes, and meet the learning objectives. The learning activities were based on constructivist theory which views instructional design as learning by doing and the learning activities or instructions were ordered according to the Gagne's Nine Steps of Instructional Events (Faryadi, 2009; Aytekin & Isman, 2011) which are summarized in Figure 11.

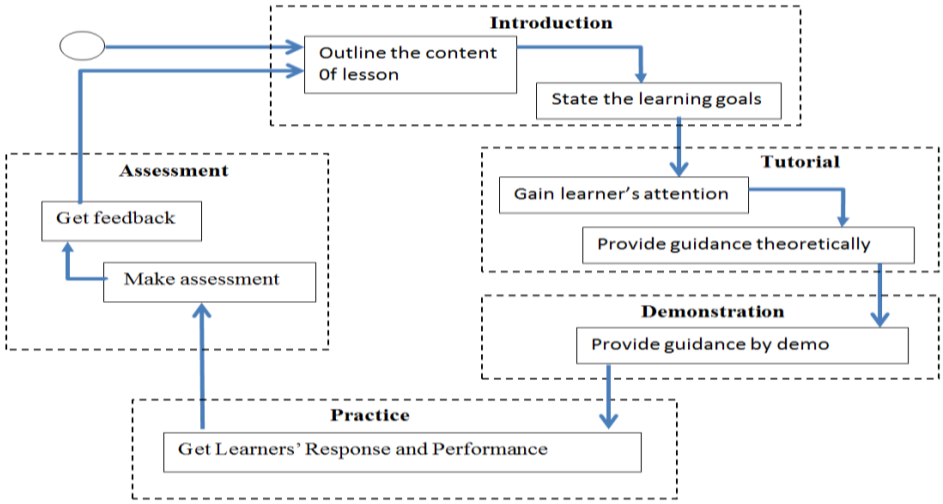

Figure 11: *Instructional or Pedagogical Strategies*

Identifying subject content per instruction

The *Design by Feature* from the integrated ADDIE-FDDP model was used to identify subject content based on the instructions from Figure 10 in which every feature has at least one instruction. The subject content was mapped into its relevant specific instruction as shown in Table 3. The types of media in every feature are also shown in the table.

Table 3: *Features, Media type, Instructions, and IMC Content*

Features and media	**Instructions**	**IMC Content**
Introduction Media: Text, Audio, Video,	Outlines of a lesson/ subtopics to be taught	Identify type of signs that indicate the place to cross the road, Finding the right place, Check condition, Make right decision, Cross the road
	State learning goals to be achieved	To be able to find the right place, check road condition, Make the right decision, cross the road safely
Tutorial *Text, Image, Animation, Audio, Video*	State the title of subject matter	Crossing the road safely
	Define keywords from title	What is crossing the road safely?
	Show actions for learners' attention	Show accident of a pupil who was hit by car
	Narrate details of concepts and principles applied	Type of roads, type of areas to cross the road, observe traffic rules
	Show relevant application examples	Show relevant application examples
Demonstration *Animation or simulation*	State the concepts to be demonstrated	Find the right place, Check road condition, Make the right decision, Cross the road safely
	Demonstrate the concepts	Find the right place, Check road condition, Make the right decision, Cross the road safely
	Narrate the demonstration	Animation with narration
Practice Media: *Game*	State the concept/challenge to be practiced	Guide a given character to cross a road safely
	Let the learner practice the concept or solve the challenge	Guide the character in finding the right place, checking road condition, making the right decision, crossing the road safely
	Record all actions taken by learner	All actions and where applied are recorded

	Display the score obtained by learner as continuous assessment while still practicing to motivate the learner	The score is displayed when a learner clicks the right button
Assessment Media: *Text, Image*	Capture learner and challenging concepts to be assessed	Learner: A Topic: Crossing the road safely
	Assess the correct and wrong actions, area of actions, score of actions	Find the right place=5/15 Check road condition= 10/35 Make the right decision=20/35 Cross the road safely=7/15
	Store all actions, area of actions, and scores attained by a learner	All records are saved in a file
	Show learners performance, action details. area of actions, scores, strength and weakness of the learner, suggest area to improve	Final results are shown in tabular form
Help *Text, Image*	Put all step by step descriptions about how to access and use a feature	Descriptions for Introduction, Tutorial, Demonstration, Practice, Assessment, and Navigation
Navigation *Text, Image*	Put all links including automatic links and action driven links	At main page, there is a menu navigation which contain all links

Designing integrated classes for IMC

In the last part of design, the *Design by Feature* from the integrated ADDIE-FDDP model was used to design the class diagram for IMC. Using features, activities' lists, and targeted users from Table 2, the classes for controlling users' interaction were determined and every feature with its own class is shown in Figure 12. The *feature* class is a super class which contains all common fields and methods found in subclasses such as *Introduction, Tutorial, Demonstration, Practice, and Help*. The home class is the main class which controls the first form with interactive functions such as navigate to any feature, select play mode, and exit. In any feature, a pupil can watch animation except in Practice feature which is for playing a game and the Assessment feature which is for performance reporting only.

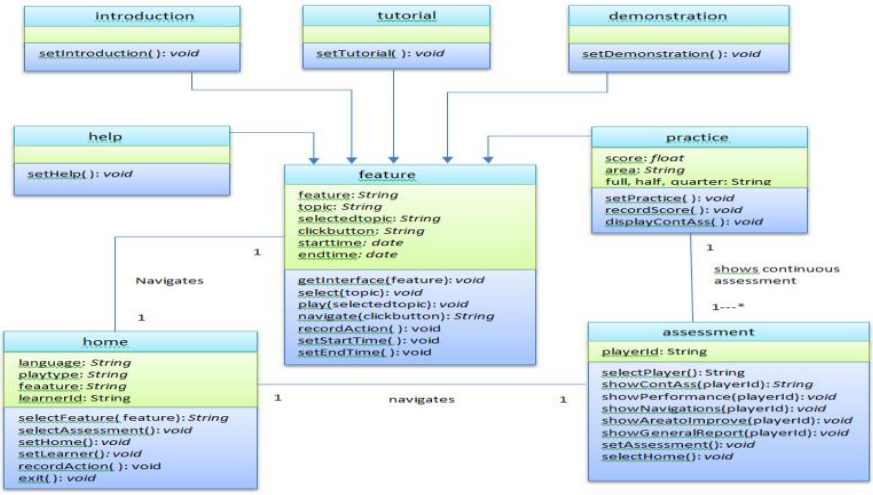

Figure 12: *Integrated class diagram for IMC*

Evaluation of the IMCLSE design

The *evaluation phase* from the integrated ADDIE-FDDP model especially the progressive assessment was used to evaluate the IMCLSE design. According to Hong (2005), there are two different aspects of quality of software design including the objectives of software design which are *modularity, portability, malleability, and conceptual integrity*, and requirements for good software design which are *well structured, simple, efficient, adequate, flexible, practical, implementable, and standardized*. All 20 questionnaires from ten software developers and ten primary school teachers were successfully collected and analysed quantitatively. The results in Table 4 and Table 5 show that the quality of the IMCLSE design was 100% satisfied. It implies that the integrated ADDIE-FDDP model was effective in guiding the design process of the IMCLSE

Table 4: *Objectives of Software Design Evaluation*

S/N	Objectives	Responses				
		Strongly Agreed	**Agreed**	**Neither**	**Disagree**	**Strongly Disagreed**
1	Modularity	19	1			
2	Portability	20				
3	Malleability	20				
4	Conceptual Integrity	20				

Table 5: *Requirements for Good Software Design Evaluation*

S/N	Requirement	Responses				
		Strongly Agreed	**Agreed**	**Neither**	**Disagree**	**Strongly Disagree**
1	Well structured	20				
2	Simple	20				
3	Efficient	20				
4	Adequate	20				
5	Flexible	20				
6	Practical	20				
7	Implementable	19	1			
8	Standardized	20				

DISCUSSION

The aim of this study was to add instructional design features to agile methods especially FDDP in order to take care of learning needs during the development process of any learning software. Agile methods have a number of preferable features such as customer involvement, cost effective, flexible and iterative (Moniruzzaman and Hossain, 2013). FDDP still lacks instructional design to deal with learning needs in the process of developing learning software (Huang, 2005). On the other hand, the ADDIE model lacks agile features to move parallel with the new era of demands such as rapid changing of requirements, iterative, cost effective, and timely delivery of software (Jasmy *et al.*, 2014). In this study, a mixed approach was used to integrate the ADDIE instructional design model and FDDP to combine instructional design and agile features. The integrated ADDIE-

FDDP model was evaluated using a case study of designing IMCLSE and a questionnaire to measure satisfaction of its quality of software design.

The study found that the participatory action research method and mixed design strategies, including decompositional, compositional, template based, and incremental and evolutionary were very useful in producing the Integrated ADDIE-FDDP model. In testing the model, it was found that the requirement elicitation from the analysis phase of the model played a vital role in gathering relevant data. The data collection instruments were prepared based on four categories of requirements including; challenges faced in provision of life skills education, targeted audience requirements, learning content and tasks, and the proposed solution domain. Analysis phase is one of area in which the FDDP lacked depth and some studies tried to improve it (Nawaz, Aftab and Anwer, 2017) but the Integrated ADDIE-FDDP model has addressed it.

It was also found that the tasks analysis played an important role in identifying learning tasks which were mapped into features such as Introduction, Tutorial, Demonstration, Practice, and Assessment that were used in creating an object oriented solution domain at analysis stage. Features are the key factors for FDDP towards producing a high quality of software product, since they are further designed into units or feature-sets which can be assigned to different teams with priorities. One unit can be developed in two weeks and shared with customers for improvement. It is where users are involved, timely delivery of software is achieved, it assures quality of software, and users' requirements are easily met. The generic ADDIE model lacks these advantages since it is too linear and customers are involved in evaluation at the end of the project (Jasmy *et al.*, 2014). Therefore, the integrated ADDIE-FDDP model has addressed the weakness of ADDIE by adding agile features.

Another important finding was that the instructional design from the integrated ADDIE-FDDP model was successful in identifying learning objectives and outcomes, creating learning instructions to achieve learning objectives, and creating subject content. The instructional design from design phase guided the process of incorporating the learning needs at design stage. The integrated ADDIE-FDDP model has addressed the FDDP weakness of lacking instructional design (Huang, 2005). It has also addressed some of the factors affecting asynchronous e-learning quality in developing countries, including course design and content support (Hadullo, Oboko and Omwenga, 2018).

Another finding is the iteration between phases was very important in producing quality software design since it helped to review and improve the next version of design. After conducting progressive assessment over the first design, design errors were identified; analysis and design phases were repeated until quality software design was produced. The original FDDP iterative property was limited between Design by Feature and Build By Feature (Rychl, 2008; Firdaus *et al.*, 2013; Nawaz, Aftab and Anwer, 2017). The integration has improved the iteration among the phases.

Code reuse is another finding which helped to reduce or control duplication of functions which occupy large quantities of computer memory space, and reduces the performance of the program. The code reuse was applied during the design of IMCLSE where common functions and states were placed in the superclass called feature, and the sub classes such as introduction, tutorial, and demonstration inherited them. Singh (2015) tried to improve the FDDP component reuse by integrating with the reuse concept but the learning tasks from the ADDIE model which were mapped into features or objects were the factors used to improve the code reuse properties of the FDDP.

The progressive assessment from evaluation phase was found to be very useful in determining whether the quality of software design for IMC was good or poor. The determination of the quality of software design before its implementation helps to avoid wastage of resources in case the design is poor, which at the end could produce poor quality of software. It also assures delivery of good quality of software product to customers. According to Aftab *et al.* (2018) and Nawaz, Aftab and

Anwer (2017), the FDDP lacked this type of assessment or testing phase and through integration with the ADDIE model its limitation has been addressed.

CONCLUSION

The main purpose of this study was to add instructional design to FDDP by combining with the ADDIE model so that it can be used in designing and developing learning software. The FDDP was selected due to its unique properties compared to other agile methods, such as the object oriented method and the ability to handle complex and large projects. The ADDIE model was selected due to fact that it is generic and standard for almost all instructional design models and it has been frequently applied in designing learning software even though it lacks agility. The integrated ADDIE-FDDP model was applied, especially its analysis and design phases, in designing IMCLSE at the primary school level in Tanzania.

After applying the integrated ADDIE-FDDP model, it was found that the model handled the learning needs very well. Its performance was effective in setting out the learning objectives and outcomes, creating learning instructions to achieve learning objectives, and creating subject content which could not be done by FDDP alone. There were many other important improvements including improvement of requirement gathering, unlimited iteration between phases, code reuse, and progressive assessment. These improvements imply that the integrated ADDIE-FDDP model is more agile than the individual FDDP and ADDIE models, with the ability to handle the designing process of learning software through its instructional design characteristics. It concurs with the results of evaluation of quality of software design which shows strong satisfaction. According to Hong (2005), quality software design implies quality of software product, therefore, the integrated ADDIE-FDDP will improve production and quality of learning software.

The beneficiaries of the integrated ADDIE-FDDP model are course designers, researchers, and software developers who are dealing with designing and developing learning software and e-learning content. The model has agile characteristics and instructional design features which will help software developers and course designers to improve quality of course and content design, and the production of quality learning software. Researchers can learn and apply in similar studies; the way in which the participatory action research method was combined with four design strategies to produce the mixed approach for designing the model. Three types of evaluations of the model were used, including the review of versions of integrated models, use of the case study method, and evaluation of quality of the software design using a Likert scale-based questionnaire. These three types of evaluation were the major strengths of the study which can be helpful to other researchers if they adapt them for related studies.

Therefore, this study is recommending that course designers and software developers adopt the integrated ADDIE-FDDP model since it has many agile and instructional design features to improve quality of designing and developing learning software. There is no doubt that it will also improve production of quality learning software since three types of evaluation have proved its effectiveness. In this study, only analysis and design phases of the integrated ADDIE-FDDP model were tested in designing IMCLSE. Further study is recommended to test the remaining phases in producing ideal software and conducting summative assessment to achieve learning goals, which is the main target for any learning software.

REFERENCES

Abidin, N. A. S. Z. and Tho, S. W. (2018) 'The Development of an Innovative Resonance Experiment Using Smartphones with Free Mobile Software Applications for Tertiary Education', *International Journal of Education and Development using Information and Communication Technology*, vol. 14, no. 1, pp. 164–176.

Aftab, S. *et al.* (2018) 'Comparative Analysis of FDD and SFDD', (January).

Ahmadigol, J. (2015) 'A Survey of the Effectiveness of Instructional Design ADDIE and Multimedia on Learning Key Skills of Futsal', (January).

Aldoobie, N. (2015) 'ADDIE Model', 5(6), pp. 68–72.

Arkün, S. (2008) 'A Study on the development process of a multimedia learning environment according to the ADDIE model and students ' opinions of the multimedia learning environment', vol. 17, no. 17, pp. 1–19.

Awad, M. A. (2005) 'Software Development Methodologies'.

Aytekin and Isman (2011) 'Instructional Design in Education : New Model', vol. 10, no. 1, pp. 136–142.

Bahl, R. (2012) 'Adaptive-ADDIE: SCRUM framework for Instructional System Design'.

Bertram, D. (2006) 'Likert Scales: CPSC 681—Topic Report', *Poincare*, pp. 1–11. doi: 10.1002/9780470479216.corpsy0508.

Bichelmeyer, B. A. and Ph, D. (2005) '"The ADDIE Model" – A Metaphor for the Lack of Clarity in the field of IDT Barbara A. Bichelmeyer, Ph.D.,' pp. 1–7.

Bosch, J. (2014) 'Software Architecture Design : Evaluation and Transformation by Jan Bosch and Peter Molin', (May). doi: 10.1109/ECBS.1999.755855.

Cross, N. (2000) *Engineering design methods*, *John Wilez / Sons, Ltd*. doi: 10.1016/0142-694X(90)90015-5.

FAO (2011) *E-learning methodologies*.

Faryadi, Q. (2009) 'Constructivism and the Construction of Knowledge', 1(2).

Firdaus, A. *et al.* (2013) 'Developing Secure Websites Using Feature Driven Development (FDD): A Case Study', 1(4). doi: 10.7763/JOCET.2013.V1.73.

Giddtngs, R. V (1984) 'Accommodating Uncertainty', (Dd).

Greenwood, D., Whyte, W. F. and Harkavy, I. (1993) 'Participatory Action Research as a Process and as a Goal.' doi: 10.1074/jbc.272.2.843.

Hadullo, K., Oboko, R. and Omwenga, E. (2018) 'Factors affecting asynchronous e-learning quality in developing countries . A qualitative pre-study of JKUAT University Kennedy Hadullo Technical University of Mombasa , Kenya Robert Oboko and Elijah Omwenga University of Nairobi , Kenya', vol. 14, no. 1, pp. 152–163.

Hapter, C. and Tremblay, G. (2001) 'S OFTWARE D ESIGN', (May).

Hong, Z. (2005) *Software Design Methodology*.

Huang, C. (2005) 'Designing high-quality interactive multimedia learning modules', vol. 29, pp. 223–233. doi: 10.1016/j.compmedimag.2004.09.017.

Jasmy, M. *et al.* (2014) 'Development and Evaluation of the Effectiveness of Computer-Assisted Physics Instruction', vol. 7, no. 13, pp. 14–22. doi: 10.5539/ies.v7n13p14.

Keith, E. R. (2002) 'Agile Software Development Processes A Different Approach to Software Design Agile Software Development Processes A Different Approach to Software Design'.

Lowgren, J. (1995) 'Applying Design Methodology to Software Development', pp. 87–95. doi: 10.11606/T.11.2012.tde-25102012-083603.

Mathiassen, L. and Stage, J. (1990) 'Complexity and uncertainty in software design', pp. 482–489. doi: 10.1109/CMPEUR.1990.113661.

Moniruzzaman, A. B. M. and Hossain, S. A. (2013) 'Comparative Study on Agile Software Development Methodologies'.

Moradmand, N., Datta, A. and Oakley, G. (2014) 'The Design and Implementation of an Educational Multimedia Mathematics Software : Using ADDIE to Guide Instructional System Design', vol. 4, no. 1, pp. 37–49.

Nawaz, Z., Aftab, S. and Anwer, F. (2017) 'Simplified FDD Process Model', (September), pp. 53–59. doi: 10.5815/ijmecs.2017.09.06.

Pang, J. and Blair, L. (2004) 'Refining Feature Driven Development - A methodology for early aspectsPang, Jianxiong'.

Parnas, D. L. and Weiss, D. M. (1987) 'Active design reviews: Principles and practices', *The Journal of Systems and Software*, vol. 7, no. 4, pp. 259–265. doi: 10.1016/0164-1212(87)90025-2.

Qasaimeh, M., Mehrfard, H. and Hamou-lhadj, A. (2008) 'Comparing Agile Software Processes Based on the Software Development Project Requirements', pp. 49–54. doi: 10.1109/CIMCA.2008.54.

Rawsthorne, P. (2005) *Agile Instructional Design*.

Rychl, M. (2008) 'A Tool for Supporting Feature-Driven Development'.

Sandhna, G. and Brügge, B. (2007) 'Major Seminar On Feature Driven Development Agile Techniques for Project Management Software Engineering By Sadhna Goyal Guide : Jennifer Schiller Chair of Applied Software Engineering'.

Seabra, C. and Almeida, A. M. (2015) 'Project Management on Multimedia Projects : Preliminary Results on Communication , Interaction and Team Work Dynamics Project management on Multimedia Projects : preliminary results on communication , interaction and team work dynamics', *Procedia - Procedia Computer Science*. Elsevier Masson SAS, 64(October), pp. 816–823. doi: 10.1016/j.procs.2015.08.633.

Singh, H. (2015) 'FDRD : Feature Driven Reuse Development Process Model', (May 2014). doi: 10.1109/ICACCCT.2014.7019376.

Sommerville, I. and Addison, W. (2006) *Software Engineering, 9th Edition*. doi: 10.1111/j.1365-2362.2005.01463.x.

Sunner, D. (2016) 'Agile : Adapting to need of the hour: Understanding Agile methodology and Agile techniques', in *2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pp. 130–135. doi: 10.1109/ICATCCT.2016.7911978.

Taylor, L. (2004) 'Educational Theories and Instructional Design Models. Their Place in Simulation' .

TGSHS (2017) 'Tanzania Mainland Global School based S tudent Health Survey', (May).

Tirumala, S. S. and Ali, S. (2016) 'A Hybrid Agile model using SCRUM and Feature Driven Development', vol. 156, no. 5, pp. 1–5.

Yang, B., Hu, H. and Jia, L. (2008) 'A study of uncertainty in software cost and its impact on optimal software release time', *IEEE Transactions on Software Engineering*, vol. 34, no. 6, pp. 813–825. doi: 10.1109/TSE.2008.47.

Zhi, E. *et al.* (2008) 'Developing Multimedia Instructional Material for Robotics Education 2 Literature Review', vol. 7, no. 11, pp. 1102–1111.