Spring 4-9-2014

Teaching Information Systems Development Through An Integrated Framework

Mohammed Salahat
Ajman University of Science & Technology, UAE, m.salahat@ajman.ac.ae

Steve Wade
University of Huddersfield, s.j.wade@hud.ac.uk

Teaching Information Systems Development Through an Integrated Framework

Mohammed Salahat
Ajman University of Science and Technology, UAE
m.salahat@ajman.ac.ae &
University of Huddersfield, UK,u0423855@hud.ac.uk
Steve Wade
Informatics Department, School of Computing and Engineering, University of Huddersfield, UK
s.j.wade@hud.ac.uk

Abstract
This paper presents an evaluation of the framework SDDD through teaching information systems development module. The framework combined techniques from Soft Systems Methodology (SSM), the Unified Modelling Language (UML), and an implementation pattern. The evaluation is done to find the applicability of the framework as an approach for teaching and developing information systems. Feedback from Msc students of the module "Methods and Modeling" and reflections from the lecturers are presented. Feedback received from all participants are used to enhance the framework development. The results are supported by our previous work proposing the integrated framework as an approach to enhance the understanding of the systems modeling and implementation skills and as an approach for ISD teaching.

Keywords: SSM, UML, Multimethodology, Soft Domain-Driven Design, Modelling, Teaching

1.0 Introduction

Teaching Information Systems Development in a proper way may well contribute to the better understanding and mastering of development skills by the students in order to develop the required software system. The failure of software support systems has been well documented over the years, and many of these failures have been attributed to poor business process modelling (Barjis, J., 2008). Other researchers have been attributed software support system failures to IS Education (Huy V. Vo etl, 2006). IS education must be reformed in order to concentrate on "Soft" issues like organizational problem solving (Lyytinen and Robey,1999), political aspects, ethics, individuals' interest, communications, etc. This will help IS professionals to learn from failures and this support the need of Systems Thinking to be incorporated into IS Education to deal with the complexity of the messy situation and to consider the system soft issues. Considering both Hard and Soft system aspects through IS education is

expected to enrich the educators' knowledge which will be reflected in the system design in the future and may contribute to the reducing of software support systems failure. From the modelling and implementation view, the systems failed because the business process model developed did not adequately support the process of designing and implementing the software support system. One of the main reasons for information systems failure is a tendency to concentrate on the technical aspects of design rather than understanding the business needs (Alter, S., 2007). There is a need for a systematic approach for capturing the information required by business processes (Barjis, J., 2008). This suggests a need to bridge the gap between business process modelling, information systems modelling, and implementation. This bridging framework ((Salahat et al, 2008), (Salahat, M., Wade, S., 2009), (Salahat, et al, 2009), (Salahat, M., Wade, S., 2012)) may well enhance the development of proper information systems and support teaching the IS development process.

This paper present further pedagogical evaluation in addition to our previous works(Salahat &Wade,2012), (Wade, et al, 2012) by using the SDDD framework as a teaching framework for the module Methods and Modelling . The students completed the module and were investigated using different tools including a background questionnaire. This paper will focus on the questionnaire analysis and briefing the other tools presented in our previous works. Section 2 presents the related works. Section3 briefs the research methodology used. Section 4 introduces the framework as a multimethodology approach. Section 5 presents evaluation through teaching. Section 6 is a discussion and conclusion.

2.0 Related Work

2.1 Teaching the Module Methods and Modelling

Teaching business information systems modeling using UML will not lead to a complete understanding that help the students or developers to implement a software support system combining all the business experts' requirements (Salahat&Wade,2009),(Salahat&Wade,2012). We argue that using an integrated framework in teaching business domain investigation and modelling can enhance understanding of such problematic situations and may be lead to a substantial software system. Based on this, the module Methods and Modelling in Informatics Department in the University of Huddersfield has been taught to the Msc Advanced Computer Science and MSc Information Systems Management students using the SDDD framework which combines tools from SSM, UML, and implementation pattern(Naked Objects, True Viewer).

This approach is applied in a wide range of situations including requirements analysis for information systems design. Other researchers have explored the relationship between SSM and object oriented analysis and design techniques in general (Bustard, D et al, 1996; Lai, L.S. 2000) but less has been written about the application of these techniques in the context of the UML. We argue that UML models can encourage early design decisions before opportunities for improvement can be agreed and that SSM lacks the detailed information required by designers developing domain models. This leads to the conclusion that there could be some advantage in using the techniques together. We expected from using this integration, in teaching systems modeling, that the students will see the whole systematics picture of the business domain and the modeling will be understandable and will lead to a sufficient business domain model for coding the required software system.

2.2 Domain Driven Modeling (DDM)

The business domain for any organization accommodates the organization business process that must be well defined and modelled for the implementation. Business domain comprises the business process that can be defined as 'the transformation of something from one state to another state through partially coordinated agents, with the purpose of achieving certain goals that are derived from the responsibility of the process owner' (D., Platt,1994). There are many definitions of "business process", and the most of these definitions are based on the idea of a business process as a deterministic system that receives inputs and transforms into outputs following a series of activities. For example (Daveport, T., 1993) defines business processes as "''structured sets of activities designed to produce a specified output for a particular customer or market''. Business processes are similar in different business domains running the same industry of business. To support the business domain, good information systems software is used to support the organization work by handling the internal business process and controlling all aspects affecting the execution of the process. The business process must be supported with good business process modeling (domain modeling) and implementation techniques that can analyze, model, and implement the business process in a professional way to achieve the organizational goals (Warboys et al, 1999). Then we argue that understanding this process by students who are studying IS Development may support their effort to develop a successful software support systems.

2.3 Domain-Driven Design

Domain-Driven Design can be used to model the business process as a business domain model (Evan, Eric, 2004). A Ubiquitous Language (UL) is generated first as a communication tool between different stakeholders and the domain model will be generated and implemented based on UL.

UML diagrams are sufficient tools for requirement modelling to support business process modelling in an object-oriented domain model (Svatopluk Štolfa, Ivo Vondrák, 2008). When it comes to implementing the system we have made use of the DDD implementation pattern (i.e. Naked Objects or True View) to reflect the system interface directly from the domain model. Naked Objects and True View Domain Modeler are used for exploring Business Domains and creating rapid prototypes using Domain Driven Design. It helps you to work with your Domain Experts to understand business entities, relationships and the business' ubiquitous language and to write classes using .NET and the Naked Objects or True View framework. This approach will make it easy for students to follow such an approach to reach to the required software.

2.4 Soft Domain-Driven Design

Soft Domain Driven Design (Salahat et al, 2009), is an approach that seeks to model the system processes as a domain model and develop a software support system based on it. In DDD Ubiquitous Language was used to create the domain model by the developers and domain experts (Evan, Eric, 2004) and to facilitate the communication between different stakeholders. UML, as a part of SDDD, defines a number of diagrams that can be used to model the business process (Al Humaidan, F.,2006) but lacks the ability to explore the soft issues related to the problematic situation which can be handled using Soft System Methodology. SSM ((Checkland, P., Poulter, J., 2006), (Checkland, P., 1999), and Checland, P., Howell, S.E,1998) is an established means of problem solving that focuses on the development of idealized models of relevant systems that can then be compared with real world counterparts. SSM is used in SDDD to model the business domain using rich pictures, root definition, and conceptual model. In our previous work (Salahat et al, 2009), we have adapted the idea of a Ubiquitous Language into a “Soft Language” which incorporate certain artifacts of a SSM analysis into the model. The first step of the SDDD approach is to develop a ‘Soft Language’ as result of the application of Soft System Methodology. This language is an a compliment of the Ubiquitous Language described in Domain-Driven Design (Eric Evan,2004) which consists of different concepts, diagrams, and

documents to facilitate the communications between the developers and domain experts. Some researchers have explored the relationship between SSM and object oriented analysis and design techniques in general (Bustard, D. W et al, 1996) but less has been written about the application of these techniques in the context of the UML. An object-oriented domain model can be extracted from this Soft Language through a transition process from SSM Conceptual Model to UML Use Cases. We argue here that SSM helps the developer to gain a deep understanding of different stakeholders' perspectives which will need to be represented in the Soft Language. In this paper we argue that this transition supported the students understanding of modeling the business domain and implementing the software support system based on that.

As described in our previous work (Salahat et al, 2009), SDDD framework guides the developer into creating a "Soft Language" which consists of the output of the SSM stage to deal with the soft aspects which are not handled explicitly by Domain Driven Design. The SSM Conceptual Primary task Model (CPTM) is used to map human activity to a UML use-case model using a new elaboration technique. Use-cases, as abstractions of business activities, are used to model the business process in a domain model using UML diagrams and based on the philosophy of DDD which employs the idea of "Knowledge Crunching" during the different stages. To the best of our knowledge, this combination has not been applied in an intervention before, and an evaluation in teaching context and the application in business projects will be a contribution to this domain of research and software development.

3.0 Research Methodology

This research, as part of on-going research work, aims to answer the following research question:

1- *How the proposed approach, for modelling and implementation, can support the process of teaching the module "Methods and Modelling" for Msc students in Informatics Department?*

As authors, we are involved in teaching in our universities. This encouraged us to use the approach of Action Research since we are actors and part of any system in the education environment. The action research project aimed at improving educational delivery on one module. A variety of forms of action research have been proposed in the context of higher education (McPherson and Nunes, 2004). In a typical action research project the researcher will occupy two roles: one as the proponent of an educational theory and the other as a user of that

theory. The typical action research project will be based on an iterative lifecycle embracing problem identification, action planning, implementation, evaluation, and reflection. The insights gained from an initial cycle feed into planning of the second cycle for which the action plan is modified and the research process repeated. The detailed methodology presented in our previous work(Wade etcl 2012), and what a available in this paper is to focus on the questionnaire analysis results and to the a void the repletion.

We found from teaching and the literature review that many software systems failed and the reasons of failures reported in related works section.

The methodology followed to evaluate the framework as an approach of teaching the module Methods and Modelling for Msc students in Informatics Department is explained in details in the previous works (Wade etcl 2012) and (Salahat & Wade,2012) and summarized here since it's a continuation work.

- Getting the feedback from students through:
  - Pre-course questionnaire: to establish a background knowledge
  - Short, un-assessed, anonymous in-class surveys: to evaluate student competence and confidence in key techniques as they are introduced.
  - Analysis of common mistakes in coursework: to find the problems and weaknesses of students.
  - Short reflective essays from students: about the perceived benefits or disadvantages of following the framework.
  - Post-Course Questionnaire: to find detailed reflections and the contribution of the framework in a achieving the module aims.
- Reflection on the framework as an approach of teaching that support the module aim achievement.

This paper will focus on the questionnaire analysis and present the results. Other methods presented in our previous work (Wade etcl,2012) will be briefly presented and related to this work in order to have an integrated picture about the evaluation process as a teaching approach. As a development , the evaluation is presented in the previous works ((Salahat & Wade,2009), (Salahat &Wade,2012) with a complete case study. So the complete picture about the methodology can be gained by checking our previous work referenced above.

4.0 The SDDD Framework

The SDDD framework (Salahat et al, 2009) is briefed here in order to relate it with the evaluation process to facilitate the understanding of the reader. SDDD was developed into an action research intervention based on research of multimethodology, which justifies combining methods for the same business intervention (Minger, J., 2000). It is a multi-method framework which intended to guide the developer through an investigation of a problematic situation. The purpose here is to insure that a comprehensive understanding is achieved in order to facilitate the modelling and implementation of the domain-driven business processes as a software support system. As mentioned in the previous work (Salahat et al, 2009), the framework was been developed through a series of "action research" case studies. Accordingly our case studies have involved development projects within our own school. The researchers are part of the school and they are participating in the daily activities related to the case studies. They supervised the students and guided them to the final stage of the projects and teaching courses related to business domain modeling and implementation.

The SDDDF Framework (Figure 1) is focused on modelling and implementation of the domain-driven business process as a software support system. SSM is used as a guiding and learning methodology with techniques including UML and implementation pattern (Naked Object or TrueView) embedded within it. Using (Minger, J., 2000) generic model which discussed in (Salahat et al, 2009), the SDDD framework consists of four phases and each phase consists of a group of activities. SDDD framework is presented in Figure1, Figure 2 represents the conceptualization of the framework, and Figure 3 represents the logical processes embedded in it. For more details about these phases refer to our previous work (Salahat et al,2009).

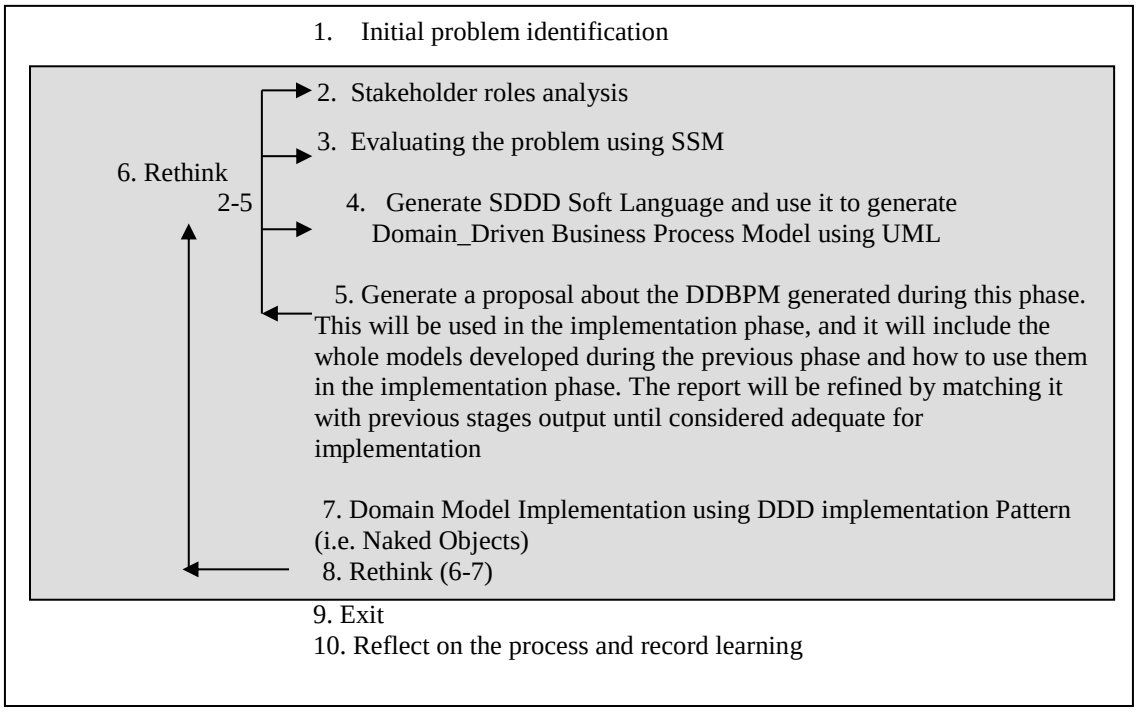

Figure 1: A Systemic Soft Domain-Driven Design (SDDD)

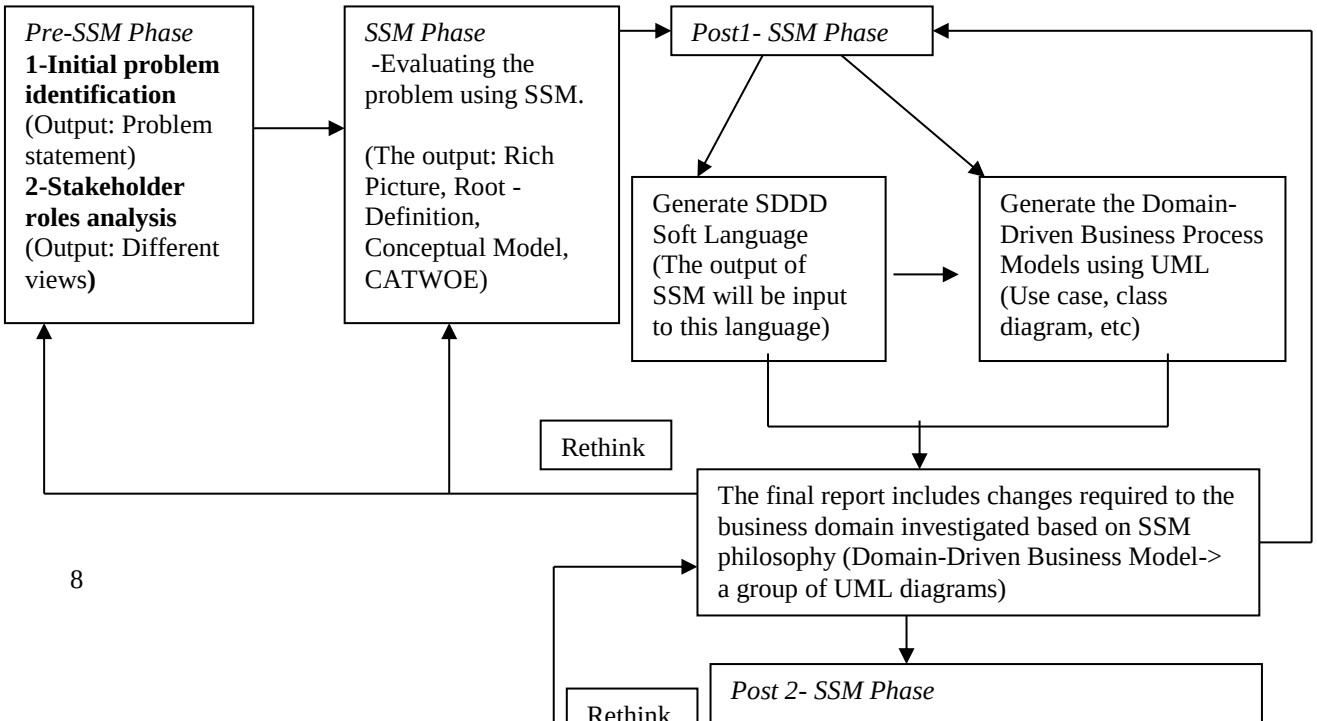

Figure 2. The conceptualization of SSDDDF

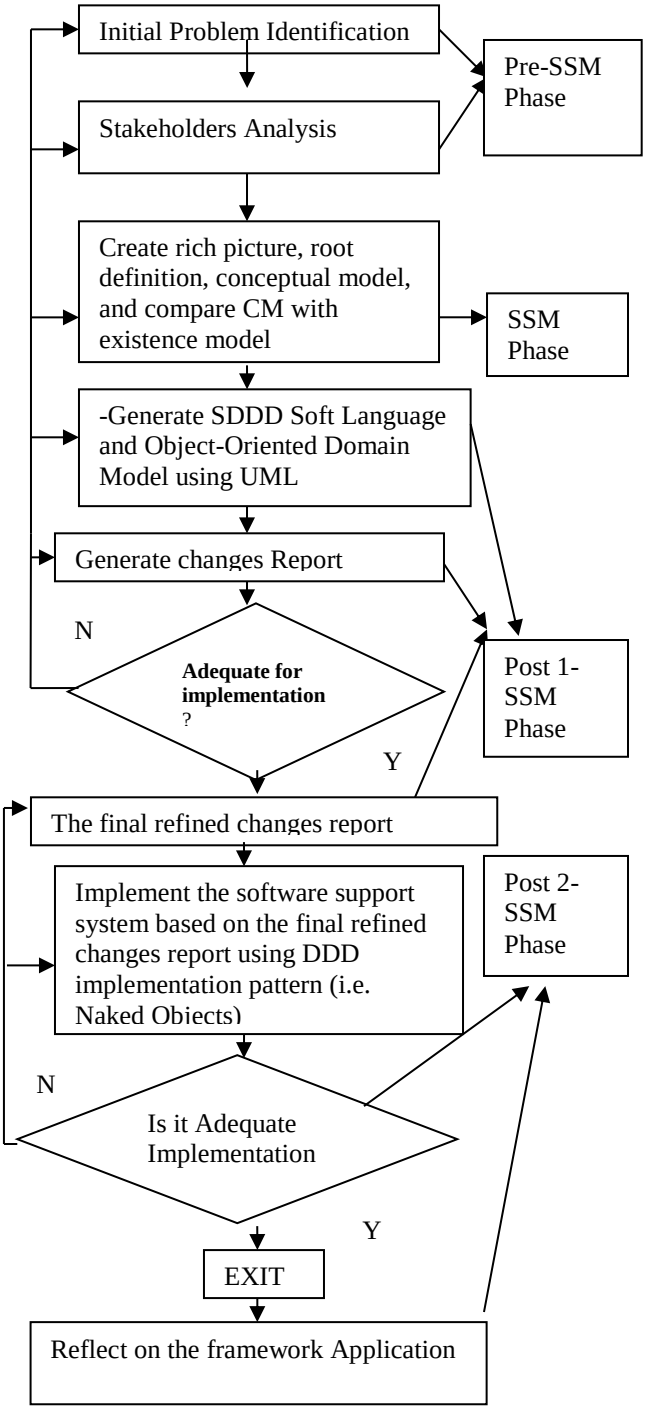

Figure 3. The embedded logic in SSDDDF

5.0 Evaluation of SDDD Through Teaching

5.1 Pre-course Questionnaire:

Thirty eight students joined the Information Systems Design module in 2011. A background questionnaire was distributed to them before the first class to gather information about their prior learning in this area. An analysis of the questionnaire shows that there were broadly two types of student taking the module, and the finding are summarized in Table(1).

MSc Students of the Module Methods and Modelling	
MSc Advanced Computer Science	**MSc Information Systems Management**
- 18 students	- 20 Students
- Strong background in programming	- Don't have strong background in programming
- Some experience of modelling but not with the UML	- Some experience of modelling but not with the UML
- None of them were familiar with the idea of multimethodology	- None of them were familiar with the idea of multimethodology
- None of them had heard of SSM	- Most of them had heard of SSM
• Parallel with this module, advanced software development modules in areas such as "internet application development".	• Parallel with this module, they studying information systems modules in areas such as "competing in a digital economy".

Table(1). Background Questionnaire Finding

These results helped us to know how to deal with the students during this module, and how to investigate them during and at the end of the module about the Framework adapted to teach this module. Next section will present the background questionnaire results.

5.2 Feedback Questionnaire:

5.2.1 Data Collection:

A questionnaire is designed to evaluate the proposed SDDD Framework as an integrated approach for teaching Information Systems Development. The design of the questionnaire is focused on the Framework Components and their contribution to the module aim achievement. The questions included in the feedback question derived from the module components and from the students interactions during the course. Students remarks and notices helped us to design the questionnaire which will be used to evaluate the module aim achievement.

The module aim is : *To Provide students with the knowledge and critical understanding of modern software and IS development methods, and skills to practice what they learned in an integrated project.* In teaching, there are different factors (*variables)* that may affect the achievement of any module aim. In the case of "Methods and Modelling" module for Msc students in the Department of Informatics at the University of Huddersfield, we focused the investigation on one of these variables which is related to the "teaching approach" used and here it is "The integrated Framework". We believed that using the framework SDDD which combined different tools of systems modelling and development would contribute to the achievement of the module aim. This framework is suggested as a teaching approach based on our previous evaluation of it in teaching and Information Systems modelling and development . Since the aim of the module is clear, we assumed that if the components of SDDD framework are understood and practiced well then this may be contribute to the module aim achievement. Variables of the study and hypothesis will be as follows:

- **Variables of the study:**

"The module aim achievement" is ***a dependent variable*** which depends on ***five independent variable*** affecting it; each variable specialized with one of the five ***hypothesis*** of the study.

-Dependent Variable (DV): *{The module aim achievement}.*

-The Independent variables(IVs):

The independent variables derived from the components of the framework and the application of them in teaching and real case studies development. So, the *independent variables* affecting the achievement of the module aim (*dependent variable*) are:

IV1: SSM Tools

IV2: UML Tools

IV3: Linking SSM&UML

IV4: Implementation Pattern

IV5: Integrating all components in SDDD Framework

- **Hypotheses of the study:**

Using the above variables we need to assess the assumption of using Soft Domain-Driven Design Framework for teaching the module "Methods and Modelling" may be contribute to the achievement of the module aim as a dependent variable. To do that, the following hypotheses are

formulated and tested to measure the effect of the above dependent variables on the module aim as a dependent variable:

H1: Understanding and practising SSM tools as part of SDDD Framework contributes to the achievement of the module aim.

H2: Understanding and practising UML tools as part of SDDD Framework contribute to the achievement of the module aim.

H3: Understanding and practicing the process of Linking SSM and UML contribute to the achievement of the module aim.

H4: Understanding and practicing the implementation pattern contribute to the achievement of the module aim.

H5: Understanding and practicing the integration of all the components of the framework contribute to the achievement of the module aim.

At the end of the module, a feedback questionnaire was distributed among students to collect the data about the contribution of each of the framework components to the achievement of the module aim. Likert approach with 5 ranks used for this purpose. 5=Strongly Agree, 4=Agree, 3=Don't Know, 2=Don't Agree, and 1=Strongly Disagree. The data analysed using SPSS- statistical software. Means and Standard Deviations proposed to analyse the descriptive data. collected through 30 valid copies of the questionnaires out of 33 responses. The total number of the students in the module "Methods and Modelling" conducted between September, 2011 and December, 2011 was 38, 33 of them participated in this investigation and 5 absent. The analysis results presented in the following section.

5.2.2 Feedback Questionnaire Data Analysis

To validate the hypothesis, Means and Standard deviations were used for the paragraphs relating to the above five hypothesis. Tables (2,3,4,5,6), presented the descriptive analysis related to these hypothesis prospectively, and Table 7 presented the hypothesis acceptance based on t test.

Rank	No	Item	Mean	Standard Deviation
1	1	I found the tools of SSM were easy to use	4.27	.78
5	2	I can see how SSM tools would help me to understand the logic of business processes	4.03	.72

1	3	I can see how SSM tools would help me to understand customer requirements	4.27	.91
3	4	I can see how SSM tools could facilitate communication between business experts and developers	4.23	.90
8	5	I found it easy to understand and communicate with my team using SSM techniques	3.67	.80
4	6	I can see how an SSM Rich Picture can provide a comprehensive overview of a business system	4.20	1.00
6	7	I can see that SSM Root definition technique depicts the required system objectives	3.83	1.02
10	8	I am confident that I could use SSM Conceptual Models to depict the detailed logic of business processes.	3.47	.90
8	9	I can see how SSM conceptual models represent the business domain processes	3.67	1.03
7	10	I am confident that I could use SSM techniques to identify the user requirements	3.70	.79
		SSM Tools	3.93	.595

Table (2). Means and standard deviations of the paragraphs relating to the first hypothesis in descending order according to Means

Table (2) shows that the means were between (4.27-3.47) , the highest mean was (4.27) for items number (1) and (3) which were " I found the tools of SSM were easy to use" and " I can see how SSM tools would help me to understand customer requirements" while the lowest mean was (3.47) for the item number (8) which was " I am confident that I could use SSM Conceptual Models to depict the detailed logic of business processes.". The arithmetic mean for all the items in SSM tools was (3.93).

Rank	No.	Item	Mean	Standard Deviation
1	1	I found that UML is easy to use for modeling business processes.	4.30	.88
3	2	I can see how Use Case diagram can be used to represent system processes.	3.97	1.10

7	3	I am confident that UML Use Cases are good tools for business process modeling	3.67	1.03
8	4	I found it easy to extract Use Cases from the SSM Conceptual model	3.43	.94
4	5	I found it easy to draw a sequence diagram based on each use case.	3.87	1.07
5	6	I found it easy to draw the Class Diagram based on the sequence diagrams.	3.87	1.04
2	7	I can see that UML Class Diagram represents the domain model of the investigated system.	4.10	.71
6	8	I understand how code can be generated from the domain model(Class diagram).	3.70	.99
		UML Tools	3.86	.618

Table (3). Means and standard deviations of the paragraphs relating to the second hypothesis in descending order according to Means

Table (3) shows that the means were between (4.30-3.43) , the highest mean was (4.30) for item number (1) which is " I found that UML is easy to use for modeling business processes. " while the lowest mean was (3.43) for the item number (4) which is " I found it easy to extract Use Cases from the SSM Conceptual model ". The arithmetic mean for all the items in UML tools was (3.86).

Rank	No.	Item	Mean	Standard Deviation
5	1	I found the transition from Conceptual Models to Use Case Models is an easy process	3.57	.94
1	2	I found that some of the activities in the Conceptual Model did not map directly to use cases.	3.83	.83
3	3	I can see that the resultant use cases represent the key activities of the conceptual model	3.70	.84
6	4	I found that the adapted method for transition is easy to use and practice	3.50	.57

4	5	I'm confident that I can depend on the resultant use cases to draw other diagrams like sequence and class diagrams	3.60	.89
2	6	I found it's useful to use SSM at the beginning to investigate the business domain and to move to UML and implementation	3.83	.91
		Linking	3.67	.517

Table (4). Means and standard deviations of the paragraphs relating to the third hypothesis raked according to Mean

Table (4) shows that the means were between (3.83-3.57) , the highest mean was (3.83) for items number (2) and (6) which are " I found that some of the activities in the Conceptual Model did not map directly to use cases." and " I found it's useful to use SSM at the beginning to investigate the business domain and to move to UML and implementation " while the lowest mean was (3.50) for the item number (4) which is " I found that the adapted method for transition is easy to use and practice ". The arithmetic mean for all the items in linking between SSM and UML tools was (3.67).

Rank	No.	Item	Mean	Standard Deviation
1	1	I found the implementation pattern is an easy to adapt and use for implementation(Name of pattern:----------)	3.63	.89
3	2	I found moving from Domain model (class diagram) to code is easy and not complicated	3.60	.72
4	3	I found the implementation pattern easy to represent the domain model processes in code.	3.60	.62
2	4	The interfaces generated by the implementation pattern are easy to use.	3.63	.67
		Implementation	3.62	.429

Table (5). Means and standard deviations of the paragraphs relating to the fourth hypothesis in descending order according to Means

Table (5) shows that the means were between (3.63-3.60) , the highest mean was (3.63) for items number (1) and (4) which are " I found the implementation pattern is an easy to adapt and use for implementation(Name of pattern:----------)" and " The interfaces generated by the implementation pattern are easy to use. " while the lowest mean was (3.60) for the items number (2) and (3) which are " I found moving from Domain model (class diagram) to code is easy and not complicated " and " I found the implementation pattern easy to represent the domain model processes in code.". The arithmetic mean for all the items in the implementation pattern was (3.62).

Rank	No.	Item	Mean	Standard Deviation
2	1	I found that integrating all the above tools in one development framework helped me to do the required project Easley	3.87	.78
3	2	I'm confident that this framework can be used to develop a complete software support system	3.70	.70
4	3	I'm confident that the whole systems components (soft and hard) can be investigated, modeled, and implemented using this framework.	3.70	.92
1	4	I found that this framework helped me to see an integrated picture of the required system in the project	4.07	.78
		Integrating	3.83	.631

Table (6). Means and standard deviations of the paragraphs relating to the fifth hypothesis in descending order according to Means

Table (6 shows that the means were between (4.07-3.87) , the highest mean was (4.07) for item number (4) which is " I found that this framework helped me to see an integrated picture of the required system in the project " while the lowest mean was (3.70) for the items number (2) and (3) which are " I'm confident that this framework can be used to develop a complete

software support system " and " I'm confident that the whole systems components (soft and hard) can be investigated, modeled, and implemented using this framework.". The arithmetic mean for all the items in integration of all components was (3.83).

5.2.3 Hypothesis Acceptance Testing

To ensure that the hypothesis was statistically accepted, its mean was compared with the suggested arithmetic mean (3) which is the standard for accepting the hypothesis at the level of significance (α= 0.05). [the average of Likert values: (5+4+3+2+1)/5 =3) and if the Mean is greater than this average then the hypothesis will be accepted]. Table (7) presented the five hypothesis testing using t test.

Hypothesis	N	Mean	Std. Deviation	t	df	Sig. (2-tailed)
1-SSM Tools	30	3.93	.595	8.592	29	.000
2- UML Tools	30	3.86	.618	7.645	29	.000
3-Linking SSM&UML	30	3.67	.517	7.123	29	.000
4-Implementation Pattern	30	3.62	.429	7.870	29	.000
5- Integrating all Tools	30	3.83	.631	7.235	29	.000

Table (7). T. Test of the suggested arithmetic mean (3) the standard for accepting the hypothesis. One-Sample Statistics Test Value = 3

The arithmetic mean of the hypothesis was tested with the suggested arithmetic mean (3) the standard for accepting the hypothesis when T value was (8.592) when the level of significance was (.000) which is lower than (0.05) and it is statistically significant. So the Null hypothesis:

1-**"Understanding and practicing SSM tools as part of SDDD Framework contributes to the achievement of the module aim"** is accepted. Figure(4) shows the results of responses of students towards the questionnaire items. Results show that item 1 and 3 got the highest mean, while item 8 got the lowest mean.

2- **"Understanding and practicing UML tools as part of SDDD Framework contribute to the achievement of the module aim"** is accepted. Figure5 shows the results of responding of students towards the questionnaire items. Results show that item 1 and 3 got the highest mean, while item 4 got the lowest mean.

3- **Understanding and practicing the process of Linking SSM and UML contribute to the achievement of the module aim."** is accepted. Figure 6 shows the results of responding of students towards the questionnaire items. Results show that items 2 and 6 got the highest mean, while item 4 got the lowest mean.

4- **Understanding and practicing the implementation pattern contribute to the achievement of the module aim"** is accepted. Figure 7 shows the results of responding of students towards the questionnaire items. Results show that items 1 and 4 got the highest mean, while items 2 and 3 got the lowest means.

5- **Understanding and practicing the integration of all the components of the framework contribute to the achievement of the module aim."** is accepted. Figure 8 shows the results of responding of students towards the questionnaire items. Results show that item 4 got the highest mean, while items 2 and 3 got the lowest means.

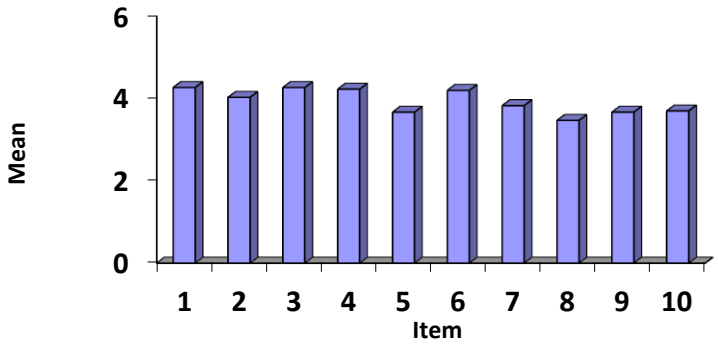

Figure 4. **Means and standard deviations of the paragraphs relating to the first hypothesis in descending order according to Means**

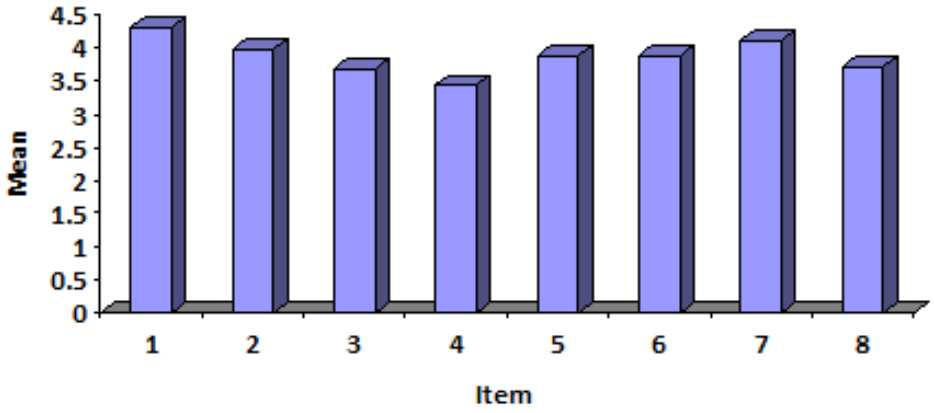

Figure5. **Means and standard deviations of the paragraphs relating to the second hypothesis in descending order according to Means**

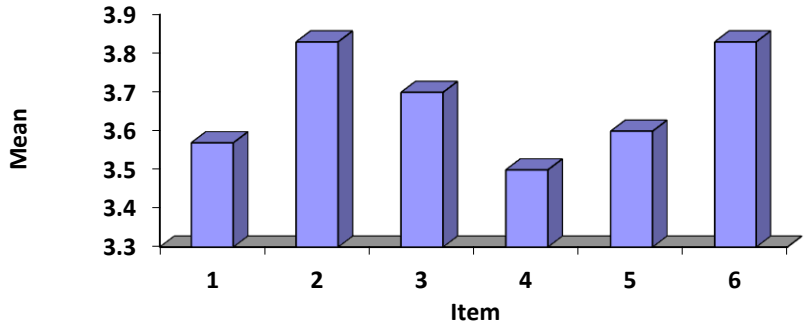

Figure 6. **Means and standard deviations of the paragraphs relating to the third hypothesis.**

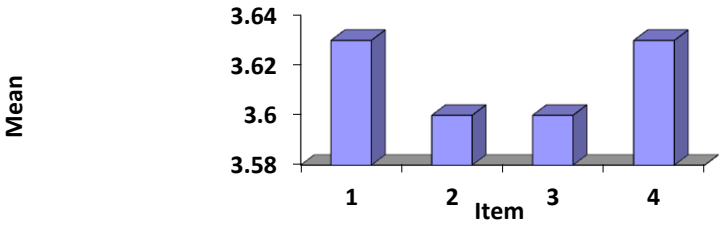

Figure7. **Means and standard deviations of the paragraphs relating to the fourth hypothesis in descending order according to Means**

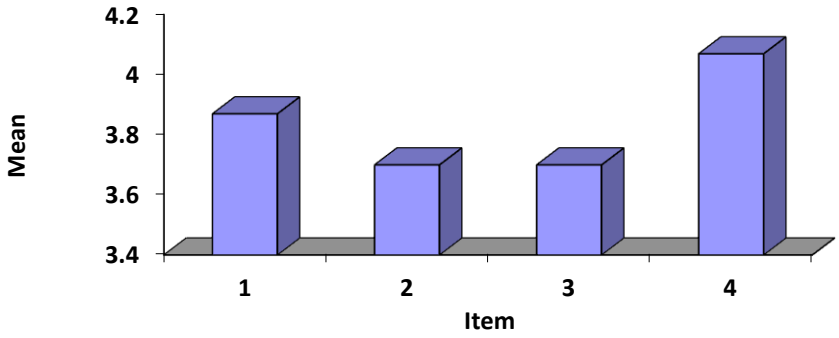

Figure8. **Means and standard deviations of the paragraphs relating to the fifth hypothesis in descending order according to Means**

By accepting the five hypothess, this lead to that the framework SDDD used to teach the Module and Modelling can contribute to the *Module Aim Achievement* as proposed at the beginning of this investigation. These results of this statistical analysis will be related to the other techniques results in the discussion section.

5.2.4 UML Tools Ranking

Relating to this hypothesis 2 UML diagram, separate question no. 9 asked to find which important diagram among the given set: "*Which UML Diagram you believe is the most important one for business domain modeling among other UML diagrams".* To answer this, ranking them from the highest to the lowest Mean is done, and Table (8) shows the results.

Rank	No.	Item	Mean	Standard Deviation
1	1	Use Case Diagram	4.57	.73
2	2	Class Diagram	4.33	.84
4	3	Activity Diagram	3.70	.65
3	4	Sequence Diagram	3.80	.85
5	5	State Chart	1.97	.85
6	6	Collaboration Diagram	1.60	.93

Table (8). Most important diagrams from highest to lowest

Table (8) shows that the most important diagram for business domain modelling among other UML diagrams is the "Use Case Diagram" with a mean of (4.57) and standard deviation (.73) which is statistically significant. The lowest diagram was "Collaborative Diagram" with a mean of (1.60) and standard deviation (.93) which is also statistically significant. These results are presented in Figure 9.

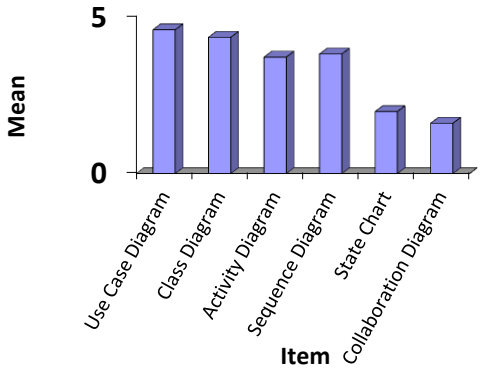

Figure 9. Most important diagrams from highest to lowest

5.3 Reflective essays:

For the final part of the coursework portfolio students were asked to write a reflective essay including a discussion on how the module reinforced (or otherwise) their appreciation of the techniques and processes employed in undertaking a development project. These essays provided generally positive feedback and one of comments is "All of the techniques have proved very useful for me. I know how to design systems properly now". This supported the finding of the feedback questionnaire that using this framework in teaching is contribute to the achievement of the module aim. For more details you can refer to(Wade, et al,2012.

5.4 Analysis of the common mistakes in the class work:

As presented in the previous work (Wade, et al, 2012)., the analysis of the coursework submitted by the students revealed a number of common mistakes. A list of common errors is presented and we are working on developing patterns that will steer future students away from making these types of mistake. This will be subject of the new publication.

5.5 In-class surveys:

Our previous work (Wade, et al 2012) presented how we applied this technique to evaluate student satisfaction on a week-by-week basis. From these it was apparent that our focus on identifying patterns to help students through difficult techniques was helpful. The majority of the students (approximately 60%) claimed no prior experience of developing business models but after completing the module, 86% said they felt confident with the use of Soft Systems techniques. There was 100% agreement that the ongoing feedback provided in this module was very useful. This supports the new finding of the Feedback questionnaire analysis that the module supports the achievement of its aim.

6.0 Discussions & Conclusion

This paper has reviewed our experience of delivering an Information Systems Development module to a postgraduate, largely international, group of students. The framework SDDD used to deliver this module evaluated by using a number of feedback mechanisms (including in-class surveys, feedback questionnaires, analysis of common mistakes in class work and reflective essays) and a sympathetic assessment strategy. This paper focussed more on the feedback questionnaire and the results show that the framework contributes to the module aim achievement. This is considered an important addition to our earlier assessment using the other

evaluation techniques. Different comments from the statistical analysis helped to understand how to deal with this framework as an Information Systems Development teaching and development approach. We have concluded that the approach yielded significant benefits for the one module discussed here but might also have wider applicability in teaching and IS development as illustrated in our previous evaluations.

REFERENCES

Al Humaidan, F.(2006) Evaluation and Development Models for Business Processes, PhD thesis, University of Newcastle, UK.

Al-Humaidan, F.,Rossiter, N.(2004) Business Process Modelling with OBPM combining soft and hard approaches, in Proceeding of 1st Workshop on Computer Supported Activity Coordination (CSAC), 6th International Conference on Enterprise Information Systems, Porto, , pp 253-260.

Alter, S., (2007) The work system method: Connecting people, processes and IT for business results, Work System Press, Larkspur, CA.

Barjis Joseph (2008) The importance of business process modelling in software systems design, Science of Computer Programming Journal, vol 71 ,pp 73–87.

Bustard, D. W., Dobbin, T. J., Carey, B. N.(1996) Integrating Soft Systems and Object-Oriented Analysis, IEEE International Conference on Requirements Engineering, Colorado Springs, Colorado, pp. 52-59.

Checkland, P., Poulter J.(2006) Learning for Action. A short Definitive Account of Soft Systems Methodology and its use for Practitioners, Teachers and Students, John Wiley and Sons Ltd, West Sussex, England.

Checkland, P.(1999) Systems Thinking, Systems Practice", John Wiley and Sons Ltd, West Sussex, England.

Checkland, P., Holwell, S.E.(1998) Information, Systems and Information Systems, Making sense of the field, John Wiley and Sons Ltd, West Sussex, England.

Checkland, P., Jim, S. (1990) Soft Systems Methodology in Action, Toronto: John Wiley and Sons.

Daveport, T., (1993) Process innovation: Reengineering work through information technology, Harvard Business School Press, Boston, Mass.

D. Platt, (1994) Process Modelling and Process Support Environment to Design Management, Department of Civil Engineering, Faculty of Engineering, University of Bristol, UK.

Eric Evan , (2004) Domain-Driven Design –Tackling Complexity in the Heart of Software, Addisson Wesley.

Huy V. Vo etl.(2006). "Integrating Systems Thinkinginto IS Education", Systems Research and Behavioral Science, Vol 23, 107-121.

Lyytinen, K. & Robey, D. (1999). "Learning Failure in Information System Development", Information Systems Journal, vol. 9, 85-101.

McPhersonM. and M.Nunes (2004). Developing Innovation in Online Learning: an Action Research Framework. London: Routledge Falmer. [CrossRef]

Mingers J.(2001) Combining IS Research Methods: Towards a Pluralist Methodology, Information Systems Research, 12, 3, Institute for Operations Research and the Management Sciences (INFORMS), pp. 240-259.

[online] available at: <http://kar.kent.ac.uk/21768/1/validation_of_object-oriented_oliver.pdf> accessed on 24th June 2010.

Pawson R. Mathews R.(2002) Naked Objects, John Wiley and Sons Ltd, West Sussex, England.

Salahat , M., Wade, S., Lu, J.(2008) A systemic Framework for Business Process Modelling and Implementation, In the proceeding of 5th International Conference on Innovations of Information Technology (Innovations'08), UAE University, Al Ain, UAE, in IEEE xplore 978-1-4244-3397-1/08..

Salahat, M., Wade S.(2009) A Systems Thinking Approach to Domain-Driven Design, In the proceeding of UKAIS2009 conference, Oxford University, Oxford, UK.

Salahat , M., Wade, S., Ul-Haq, I.(2009) The Application of A systemic Soft Domain Driven Design Framework, WASET online Journal, Issue57,pp476-486.

Salahat, M., Wade S.(2012) Pedadogical Evaluation of S systemic Soft Domain-Driven Design, In the proceeding of UKAIS2012 conference, Oxford University, Oxford, UK.

Wade, S., Salahat, M., Wilson, D.(2012) A Scaffolded Approach to Teaching Information Systems Design. ITALICS Volume 11 Issue 1 June 2012 - ISSN: 1473-7507

Sewchurran, K. & Petkov D.(2007) A systemic Framework for Business Process Modelling Combining Soft Systems Methodology and UML, Information Resources Mnagement Journal, 20, 3, IGI Publishing, PA,USA, P. 46-62.

Svatopluk Štolfa, Ivo Vondrák,(2006) Mapping from Business Processes to Requirements Specification, Retrieved on 7th Aug, 2008 from 85.255.195.219/conf/esm/esm2006/abstract.pdf.

Warboys, Brian, Kawalek, Peter, Robertson, Ian, and Greenwood, Mark, (1999) Business Information Systems-A process approach, McGraw-Hill, UK.