Practice Based Closure Quality Loop Optimization in Teaching Learning Process: A Case Study in Software Engineering Design

[1]Shefali Sonavane, [2]Mayur Rathi, [3]Sachin Patil
[1,2]Department of IT, [3]Department of CSE
[1,2,3]Walchand College of Engineering
Sangli (MH) 416415
INDIA
[1]shefali.sonavane@walchandsangli.ac.in, [2]mayur.rathi@walchandsangli.ac.in,
[3]patil.sachin@walchandsangli.ac.in

Abstract: - The worldwide expansion of an education field highly demands for skill based learning with professional approach. In response to this greedy need, the paper emphasises practice based learning in engineering education. The iterative method has been followed for extensive role play simulation in order to develop the product for realistic client. The described learning model is implemented over a student class of an undergraduate level course, 'Software Engineering and Design'. The evolutionary process is adopted to optimise the students' performance assessment in terms of critical thinking, technical competency and project management skills. The judgmental abilities are tried to quantify with the help of attainment calculations which define the *Difficulty Index* of the course. This process helped to form the basis and maintain the quality benchmark for course assessment and provided the opportunity to improve the same through the closure loop.

Key-Words: - Practice Based Learning (PBL), Course Outcome (CO), Software Requirement Specification (SRS)

1 Introduction

The basic aim of engineering education is to impart the fundamental knowledge of the stream engineering to understand the problem significance, execution methodology, process models or the system frameworks. Besides knowledge depth, student should have ability to work across the breadth with concerned stake holders. In addition to the stream knowledge, engineers have to deal with the market strategies, business activities, creativity and innovations, ethics and social responsibility through realisation of the advanced engineering deployable solutions. In general, engineering education has been criticized for being theoretical and conceptual. It becomes necessary to have keen attention to adopt Practice Based Learning (PBL) [1]. Pedagogical approach of teaching learning process suggests the effective combination of technical skills along with personal, interpersonal and professional approach in technical education. The study carried out in this paper, proposes client communication, requirement analysis, system design and data operation for product implementation. The course outcomes for the experiment are as per the Bloom's Taxonomy to endorse the *Graduate Attributes* with Knowledge, Skill and Attitude *(K-S-A)* of the theme [2~3].

The experiments carried out in the disciplinary stream are under continuous evaluation process with Course learning Outcome (CO) attainment. The course objectives and outcomes are designed prioritising the needs and the basic requirements of the associated industry in concern with the society. The working of undergraduate teams with interdisciplinary professionals in market executes the on site development of the product. Thus, the conceptual knowledge imparted to the students in classroom teaching has been practiced in handling live assignments [4]. This entire role play becomes interesting and reflects the marginal scope in students' performance improvement.

2 Practice Based Learning (PBL)

As the backbone of PBL, the supervisor acts as a facilitator and motivates students through active engagement in problem solving using integrated teaching approaches like cooperative learning, online and face-to-face tutoring, project exhibits, case studies, simulations etc.

However, typically it measures students' learning ability for:

a. Collaborative and Reflective Learning
b. Scholarship of Knowledge and Critical Thinking
c. Proficiency and Usage of Modern Tools
d. Core Competency and Problem Solving
e. Project and Finance Management
f. Presentation and Documentation Skills

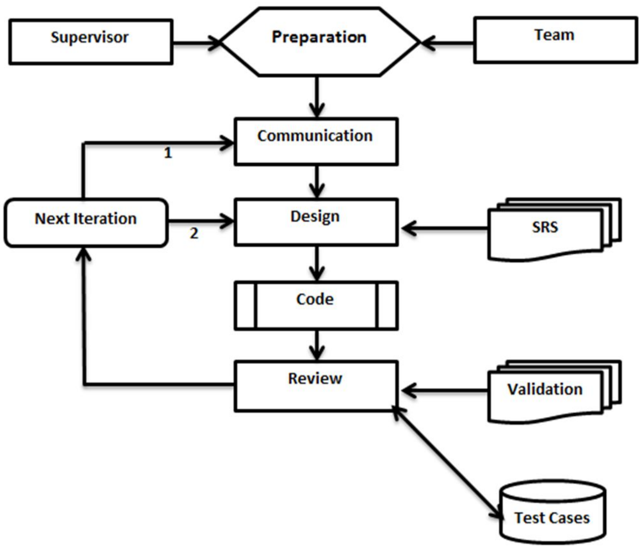

Fig 1: PBL Extensive iterative process for Quality Closure Loop

Based on similar concepts, figure 1 encapsulates the form of a closed loop PBL environment where activities in each stage benefit cross functional communication in software development life cycle [4]. The analogy of the understanding is briefed as follows:

a. Preparation

Supervisor along with team has to prepare a plan for development of an assignment. Risk management and critical path has to be designed for the time estimation of individual increment in the project. Collaborative and reflective learning are the core competencies of this phase.

b. Communication

Student in a group is supposed to interact with a client for requirement specification. The customer has to share and put up a presentation about the working platforms and requirements to fill the SRS document.

c. Design and Code

The product code can be designed based on the SRS documentation. It undergoes *Brain Storming* as a major discussion part towards end solution which helps to inculcate the problem solving ability within the students.

d. Review

Reviewing and necessary debugging are the important tasks against product testing. If product quality *Test Case* fails or not satisfactory, the model needs to step back in a loop for next iterative cycle of the progress.

e. System Testing and Validation

Team has to review the product in several revisions followed by consensus of the client with some acceptance testing. If it passes, the product is allowed to be deployable to client work station otherwise needs to be rebuild by the execution and related subroutines.

3 Case Study: Software Engineering Design

As an experiment, the extensive role play is carried out for the course 'Software Engineering and Design' (SED) at the Department of Information Technology for undergraduate engineering students [3]. The group of students are asked to have on site visits for market survey to fix up the problem relevance and their categorical requirements.

The tutorial is handled for learner's evaluation process against PBL and tried to analysed whether it can be improved further to maintain the Quality Closure Loop as described in figure 1.

a. CO Understanding and Analysis

The Course learning Outcomes (COs) are the statements that describe the students' expected proficiency learning outcome to the level. In view of this, Table I lists the framed COs for the execution of the course.

Table I: SED Course learning Outcomes (CO)

CO	After completion of the course, it is expected that student should be able to:	Bloom's Cognitive	
		Level	Descriptor
CO1	Classify the quality attributes of Software Engineering	2	Understanding
CO2	Apply appropriate framework for software life cycle	3	Applying
CO3	Design and Implement UML based model for software development process	6	Creating

If COs are overlooked, it reflects noticeable parameters for course evaluation. In a deeper sense, the action verb, condition and its standard become very important while judging the performance. It can further be divided into next level domains. e.g. figure 2 gives the third CO defined in Table I and its logical decomposition into three components.

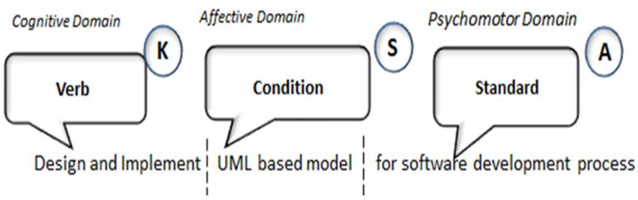

Fig. 2: Components of Course learning Outcome

Verb: It is expected that the verb should count some action succeeding cognitive levels defined in the Bloom's Taxonomy [2]. In this component, the action verb extracts the intellectual thinking and knowledge parameter (K), necessarily required in outcome based evaluation. 'Design and Implement' are the tasks under sixth cognitive level viz. 'Creation'. Students are expected to be able to create a Unified Modelling Language (UML) solution for any product for a client in extensive role play simulation [4].

Condition: It describes the behavioural environment under which learner's performance is to be ensured. Condition extracts the skill (S) achieved in affective domain. In component two; the condition, *'UML based model'*, highlights the domain requirement for development of the product.

Standard: Standard supports the measures for acceptable level of the learner's performance. It is also known as a psychomotor domain reflecting the attitude (A) in PBL. In component three, the term *'for software development process'*, defines the standard of student's engagement for learning activity.

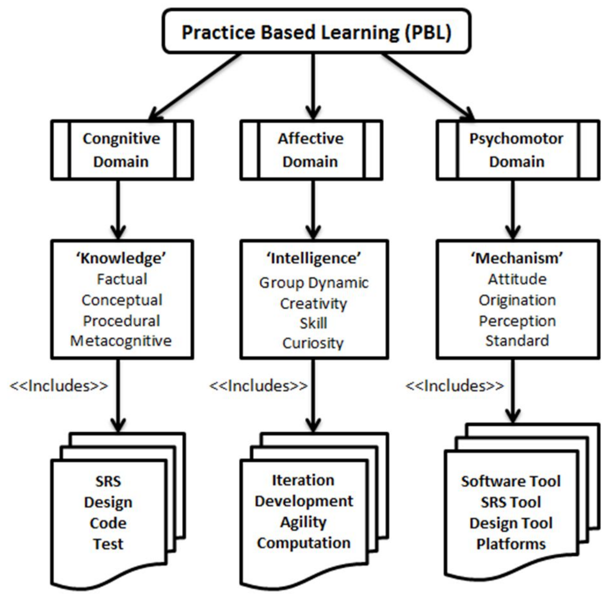

Fig 3: PBL: Operational Domains

The concept of a broad inclusion of PBL system into three major domains: Cognitive, Affective and Psychomotor Domain is given in figure 2. Its generalization in terms of quantitative measures like Knowledge, Intelligence and Mechanism is illustrated further as per figure 3.

If this framework is made applicable for any software development process then the inferred mapping can be made as follows:

i. Cognitive Domain

It defines the higher order, intellectual, rational thinking for any software engineering problem for the solution through affective domain. It has knowledge discovery attributes for various problems and solution space. Knowledge can be specified further as Factual, Conceptual, Procedural and Metacognitive. Factual knowledge signifies the basic principle, key terminology, facts and specific details that student must know while performing the ongoing assignment for a client in software development (e.g SRS document). Conceptual knowledge includes the schema outline definition for development, database and relationship among quality attributes of design, development and testing criteria. Procedural knowledge points towards algorithm, methodology and skills in software paradigm. Metacognitive knowledge implies the evaluation, verification and validation of modules/submodules of the project and related assignments [5].

ii. Affective Domain

Affective domain reflects the locus of control for the cognitive approach. It defines the condition for reasoning and intellectual thinking for any constraint based client requirement in development phase. It gathers collective intelligence of group dynamics, creativity, skill and curiosity.

iii. Psychomotor Domain

Psychomotor Domain suggests the standard approach and mechanism for any predefined outcome. It signifies the attitude, perceptions and origination of concepts, analysis and application. It includes standard SRS tools, design and testing tools to validate the customer requirements through various coding platforms.

b. Attainment and Evaluation

In the given study, every group consists of approximate three students as such class has 25 groups. Each student has a specific role as *Requirement Analyst, Designer* or *Project Manage*r. Group performs the project assignment allocated to them in consensus with the client.

As a part of continuous evaluation pattern, the academic cell of the Institute evaluates the performance through four major assessment parts structured for the course: In Semester Evaluation (ISE-I, II), Mid Semester Evaluation (MSE) and End Semester Evaluation (ESE).

In general, the defined COs are having equal importance and hence the weightage. This indicates that, in total if three COs are targeted; the overall distribution and its impact lies in the range of 30-40%.

Let Table II be referred as the proposed guideline weightage distribution over defined COs given in Table I for the case study under discussion.

Table II: CO Weightage Assigned

Course learning Outcome Weightage			Total
CO1	CO2	CO3	
30%	30%	40%	100%

Table III enlists the series of assignments distributed over 25 groups in a class. Though, the titles differ with the groups, the execution style and hence the assessment parameters are similar which refer to some common predefined rubrics for the evaluation. Using the marks obtained by students, the individual CO attainment is calculated. These calculations help in defining the *Difficulty Index* (DI) of the course.

Table III: Assessment Parameters for Group Assignment

Group No.	Assignment Title	Assessment Parameter
01	Blood Bank Management System	• To conceptualise and to analyse knowledge to obtain innovative solutions • To think critically for continual improvement in system designs considering its impact on rest of the modules • To hypothesize with laid assumptions for future enhancement and persistence • To design and to implement functional test cases • To ensure comprehensive system usability and learnability with proper documentation
02	Online Ticket Reservation System	
04	Online Mobile Recharge System	
05	Canteen Food Ordering and Management System	
06	Library Management System	
≈	≈	
25	Stationary Store and Bill Management System	

$$\text{DI} = \frac{CO_{Cumulative\_sum}}{CO_{Total}} \qquad (1)$$

Where, $CO_{Cumulative\_sum}$ is summation of marks obtained in CO_i; and

CO_{Total} is summation of individual weightage assigned to CO_i; $(1 \le i \le m)$

$$\text{CO}_{\text{Total}} = \sum_{i=1}^{m} CO_{i\_marks} \qquad (2)$$

Course Difficulty Index (CDI) is the ratio of Difficulty Index DI_i to the total number of COs in the related course.

$$\text{CDI} = \frac{\sum DI_i}{m} \qquad (3)$$

Table IV: Attainment of Difficulty Index

Group No.	CO Wise Marks Obtained			DI (%)	CDI (%)
	CO1	CO2	CO3		
1	15	21	21	Using Equation 1 and 2, DI for CO1 $DI = \frac{15+22+18+21+18}{150}$ =62.6 %	Using Equation 3, $CDI = \frac{62.6+68+81}{3}$ =70.5 %
2	22	26	32		
3	18	18	36		
4	21	20	35		
5	18	17	38		
$CO_{Cumulative\ sum}$	94	102	162		

Table IV illustrates the procedure for sample calculation of DI (for CO1: five student groups), $CO_{Cumulative\_sum}$ and CDI for three COs.

The graph is drawn in figure 4 indicating comparative CO attainment. Even if, the range of the marks obtained by student groups is across the wider range, the percentage DI calculation concentrates to a limited span and eventually can figure out with a single value of CDI.

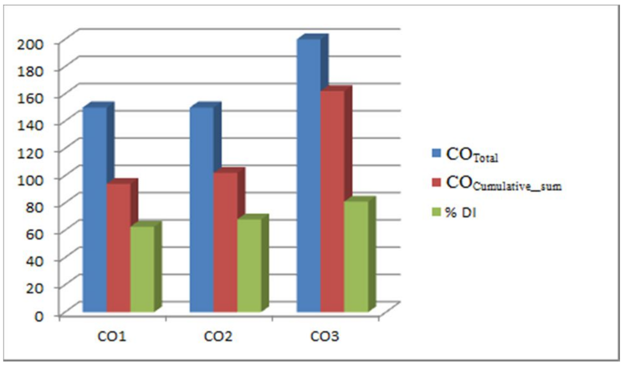

Fig.4: Percentage DI and CO Attainment

Table V implies the target attainment decided by the course supervisor, achieved attainment and realization of gap after evaluation. If achieved attainment is not within the predefined acceptable range; proper action can be planned to fulfill the gap. This process supports the concept of the Closure to Quality Loop through the development of the client requirement as described in Figure 1.

Table V: Closure to Quality Loop

CO	Target Attainment	Attainment Achieved	Attainment Gap	Remark
CO1	65 %	62.6 %	-2.4 %	Satisfactory (Needs to improve)
CO2	70 %	68 %	2 %	Average
CO3	75 %	81 %	6 %	Excellent

4 Discussions

The case study presented helped to demonstrate the components of PBL system. It focuses on the methodical procedure involving novice learners towards gaining expertise through self-learning and shared responsible commitment to the client. Throughout the phases of the iterative learning; the knowledge perceived, design verifications, test validations and corrective actions are to be followed for maintaining Quality Closure Loop. The outcome shows the *Difficulty Index* achieved by students in potentiating the team work through extensive role play simulation. The performance can further be enhanced by setting higher threshold of *Course Difficulty Index* in successive increments.

References:

[1] Patricia Charlton, Katerina Avramides, "Knowledge Construction in Computer Science and Engineering when Learning Through Making", *IEEE Transactions on Learning Technologies,* 2016, Vol. 9, pp. 379-390.

[2] S. Selvakumar, Kanchana Rajaram, "Achieving Excellence in Engineering Education through Improved Teaching-Learning Process", *IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, 2015, pp. 247-252.

[3] Peter Blatchford, Peter Kutnick, Ed Baines, Maurice Galton, "Toward a social pedagogy of classroom group work", *International Journal of Educational Research*, 2003, Vol 39, pp. 153-172

[4] Torstein Elias L. Hjelle, "The Role of Strategies of Practice-Based Learning for Becoming a Member of a Community of Practice", *48th Hawaii International Conference on System Sciences (HICSS)*, 2015, pp. 3691-3700.

[5] Xiao song Li, "Learning from Industry by using an Inquiry Based Learning Approach", *IEEE Frontiers in Education Conference (FIE),* 2013, pp. 943-949.