PEDAGOGICAL PATTERN LANGUAGE FOR IN TIME STUDENT CONFIDENCE IN STUDIED MATERIAL

Syed Waqar ul Qounain Jaffry
Punjab University College of Information Technology, Lahore, Pakistan
swjaffry@pucit.edu.pk

Abstract

Over the past few years, importing design patterns from software engineering to the computer science education (CSE) is followed by defining patterns and pattern languages suitable for pedagogical needs of Computer Science courses. The main goal of patterns incorporation in CSE was to enhance quality of content delivery and comprehension. However it is observed that soon after learning a concept most of the students are not certain about the degree of command over the studied material. Lack of questioning power makes it more devastating. The existing pattern-based materials seem to be insufficient for settling these problems. Hence there is a requirement of pedagogical pattern language that have twofold power of resolving these issues. This paper presents a pedagogical pattern language based on three newly discovered pedagogical patterns to address in time students confidence in their level of comprehension about learned skills.

1. Introduction

Patterns are predefined potential solutions to recurrent problems in different contexts. [1]. This simple but useful notion of patterns has now been applied to many fields that range from software engineering to education methodologies. During teaching we encounter enormous such problems that reoccur again and again e.g. motivation of students, selection and sequencing of teaching materials, evaluation of students, and the like [2]. In education, existing pattern-based material primarily address content formation and its delivery. One of many recurring issues that reside on students end is their in time confidence in the degree of comprehension of studied material. Present collection of pedagogical patterns and pedagogical pattern languages seems insufficient to address this problem. It is apparent that this assurance cannot achieved until or unless students through an examination. In order to give them in time confidence this examination should be well created and presented soon after the delivery of every concept. In this paper we proposed a pedagogical pattern language that has been tested in CS2 courses and remained very successful to addresses this issue.

Rest of the paper is divided into 4 sections, first three presents three newly discovered pedagogical patterns i.e. Inquiry, Self-Evaluator, and At the End; and last presents proposed pedagogical pattern language based on above three patterns that offers a solution to the mentioned problem.

2. Pattern Name: Inquiry (v 1.0)

2.1. Intent

After the successful delivery of the concept students must be able to answer at least five question based on what, why, how, when, and where about it. This pattern teaches students that how they inquiry about their level of comprehension of the concept they learned.

2.2. Problem/Issue

Mostly, students are unable to examine their complete comprehension of the learned concept, as they are not aware that answer of which questions at least, they must know about the concept, after its delivery.

2.3. Motivation

To boost up student confidence about the level of comprehension of the concept.

2.4. Context

Widely applicable; almost after the delivery of every concept.

2.5. Example

This pattern occurs in all disciplines. For example in CS1 course when we teach concept of loop structure, then after lecturing students must be able to answer

what is loop structure?
a program flow control structures that repeats a set of statements up till a base condition is true.

why we require it?
to repeat a set of statements till a certain condition gets falsified.

how it can be implemented?
in a any programming language it structures goes like.

keyword condition-start-maker base condition condition-end-maker

code-start-maker
set of statements
code-end-maker

e.g. in C++

```
while (a  > 10) {
cout << a++;
}
```

when it will be used i.e. types of problem?
loop control structure will be used in set of problems where a specific task is repeated again and again i.e. display first ten multiple of 2.

where do we write it (code) for successful use?
code of loop control structure will be used as per required in modular breakdown of problem.

2.6. Structure

Answer of questions formed using five words must be in following sequence

what
why
how
when
where

“where” may proceed “when” in some scenarios.

2.7. Variations

Set of fundamentals queries can be extended with respect to context of the concept e.g. type is another fundamental query in may case, types of looping etc.

2.8. Applicability

After the execution of each concept.

2.9. Consequences

Students can comprehend concepts more effectively, and gets information about their level of comprehension of the concept.

2.10. Known Uses

We have used this pattern during lectures of CS1 and CS2 courses taught to graduate and under graduate level at Punjab University College of Information Technology with a remarkable success.

2.11. Related Patterns

In Self-Evaluator students can use this pattern to generate initial set of questions.

2.12. Notes

None.

3. Pattern Name: Self-Evaluator (v 1.0)

3.1. Intent

After the delivery of the concept, students are asked to write questions about it that they can and cannot answer, and then they write the answer of those questions that they can solve. At the end teacher completes this process by answering unanswered questions. Use of this pattern teaches students how to analysis and inquiry a concept, enhances their questioning power, and gives them confidence about the concept they learned.

3.2. Problem/Issue

After the learning of a concept mostly students are not confident about the level of comprehension of material they covered. Lack of questioning power makes it more devastating. So there is a requirement of a pattern that have twofold power of resolving these issues.

3.3. Motivation

To enhance analysis and exploration skill of the student and to examine level of comprehension of the concept.

3.4. Context

Students of any subject at all levels from novice to experienced can use this pattern.

3.5. Example

There may be a number of examples from every subject e.g. during CS2 course after teaching linked list students are asked to write a set of questions that they have in their mind about the concept, they may use inquiry pattern to build up an initial set of questions and then can add up more to it i.e.

what is linked list?
why we require it?
how do we construct it?

and then they can add up few more as per context of the concept.

why do we require destructor for this class?
how do we traverse in reverse order in the list?
can we use list as stack, if it is, when it gets full?

3.6. Structure

Students must write those questions first that they feel they can answer. After hearing and summarizing, teacher should tell them to write those questions that they feel they cannot answer. Finally he answers unanswered questions.

3.7. Variations

This pattern can be used with student design sprint [4] to combine the philosophy of both and achieve better results.

As a variation of At the End pattern teacher can also put a set of questions of different levels at the end of lecture.

In K12 or lower classes teacher may put a set of wrong questions to let students correct it. Teacher may ask the students to make a set of wrong questions as well.

3.8. Applicability

After the delivery of the concept.

3.9. Consequences

Students will get information about their level of comprehension of the concept. Inquiry skill of the students will increase. Teacher will get statistics about the different aspects of concept that students can or cannot answer.

3.10. Known Uses

This pattern has been used during CS2 course taught at Punjab University College of Information Technology, and it remained very successful for both to increasing student comprehension and to boost up student ability of analysis and inquiry.

3.11. Related Patterns

In Test Tube [4] teacher tells about the exploration of answers, here he/she asks for exploration of questions.

3.12. Notes

Teacher may ask students to make questions of different types i.e. short questions, fill in the blanks or multiple-choice questions.

4. Pattern Name: At the End (v 1.0)

4.1. Intent

After the delivery of the concept, teacher poses questions of different difficulty levels e.g. beginner, intermediate and advance, students write the answer of those questions that they think they can answer. At the end teacher completes this process by answering unanswered questions. Use of this pattern not only gives students surety about their level of comprehension and standing in the concept that they studied but also increases their comprehension.

4.2. Problem/Issue

After the learning of a concept mostly students are not confident about the level of comprehension and their standing in the material they studied.

4.3. Motivation

To examine level of comprehension of the concept, and standing in the material they covered.

4.4. Context

Teachers can use this pattern for the students of any subject at all levels from novice to experienced.

4.5. Example

There may be a number of examples from every subject, in CS2 course for example after teaching the concept of linked list teacher can ask following questions i.e.

Beginner: when do we need link list?
Intermediate: how do we implement it?
Advance: how do we traverse in the reverse order in a linear link list?

4.6. Structure

Teacher should prepare a sufficient and necessary set of questions regarding the concepts before delivering lecture. Teacher should provide answer to that entire set of question, which students cannot answer.

4.7. Variations

This pattern can be used successfully after the execution of Self-Evaluator where students formulate their questions first. Teacher may put a wrong set of questions to let students correct it. Teacher may ask the students to make a wrong set of questions.

4.8. Applicability

After the delivery of concept in all subjects.

4.9. Consequences

Students will get information about their level of comprehension of the concept. Teacher will get statistics about the different aspect of concept that students can or cannot answer.

4.10. Known Uses

We have used this pattern during CS2 course taught at Punjab University College of Information Technology with very optimistic results for both to judge students comprehension statistics for different topics and to provide students with a measure of contents comprehension.

4.11. Related Patterns

In Self-Evaluator teachers ask for exploration of questions, here they put a set of questions by themselves.

4.12. Notes

None.

5. Pedagogical Language for in-Time Student Confidence in Studied Material

In the start of pedagogical pattern movement pedagogical pattern collection was just a collection of single examples, contributed by many people, on a common theme. Today the thrust of the endeavours has changed, and conception of pattern language is central to pattern and pattern design methodologies. A pattern language is a set of fundamental patterns that work together to generate complex behaviour and complex artifacts, while each pattern within the language is itself simple [3].

Pattern languages emphasis now on smaller, more tightly constrained areas either singly authored [4], or constructed by a small group of people [2]. This approach, more tightly coupled to specific contexts, is more coherent and therefore successful [5].

Pedagogical pattern language for in time student confidence is a collection of three pedagogical patterns Inquiry, Self-Evaluator and At the End represented in Figure 1. Inquiry pattern is designed so that students can construct a minimal set of fundamental questions at their own that they must be able to answer for every concept after study. This pattern plays a role of starter. Self-Evaluator pattern helps students to enrich their question bank with supplement questions that they think, they can and cannot answer related to learned concept. Finally At the End pedagogical pattern provides students with an opportunity to collect the set of questions that their teacher expect them to answer. Different levels of questions in last pattern lets students measure both, their degree and level of the comprehension of concept.

Before the commencement of lecture, teacher prepares a set of question in three different difficulty levels about the concepts to be taught. After the start of lecture and successful delivery of every concept teacher will initiate the Inquiry Pattern that generates a set of fundamental question that should be answered by the students. In second phase Self-Evaluator Pattern is followed that provides an additional set of questions that students think they can answer. This step gives student a confidence in their comprehension. Then they write those questions that they think they cannot answer. These questions are answered by the teacher that helps in the concept review and increase student compression.

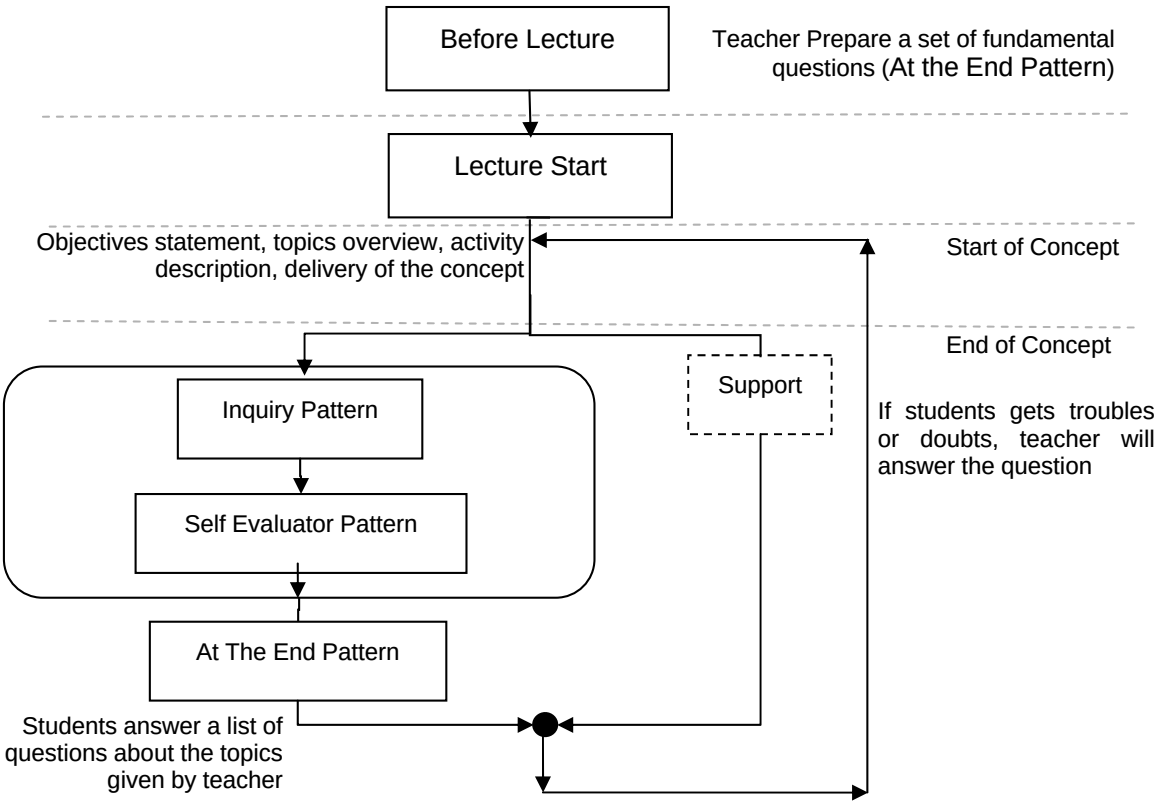

Figure 1. A pedagogical pattern language for in time student confidence in studied material

Finally teacher introduces At the End Patterns that results those set of question that has not yet discussed and are necessary for the completion of the process. He divides these questions into three different difficulty levels so that students become confident in both degree and level of comprehension of the concept.

In Parallel to this activity if students feel any problem in the comprehension of the process, creation or answering of questions, teacher should be available for the support.

6. Conclusion

Patterns have emerged as a simple but most effective belief that has influenced many areas. In the field of teaching; pedagogical patterns has been introduced few years ago and proved very effective methodology to address pedagogical problems and their communication among teaching community. In this article we have introduced a brief conception of pedagogical patterns and presented a set of three newly discovered pedagogical patterns Inquiry, Self-Evaluator, and At the End that are fundamental to solve lack of student confidence in the compression of studied material. Final these three patterns are used as a collection to form Pedagogical Language for Student Confidence that effectively solves in time Students Confidence problem in studied material. It is hoped that this proposed language will be further matured to solve wider aspects of students confidence that not only deals with the level and degree of comprehension of studied material but also its other variants that includes students confidence in the, subject, degree, personality, and environment where he/she lives.

7. Acknowledgements

Author is thankful to all those who have contributed to the Pedagogical Patterns project, especially Joseph Bergin, for his guidance, and appreciation to improve this concept.

8. References

[1] Alexander, C, Ishikawa, S. & Silverstein, M. A Pattern Language. Oxford University Press, Oxford, UK, 1977.
[2] http://www.pedagogicalpatterns.org/
[3] Joseph Bergin, ”A Pattern Language for Initial Course Design”, ACM SIGCSE Bulletin, Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education.

[4] Fricke, A et al, “SEMINARS A Pedagogical Pattern Language about teaching seminars effectively”, EuroPLoP 2000.
[5] Fincher, S et al, “Pedagogical Patterns: Their Place in the Genre”, Proceedings of the 7th Annual Conference on Innovation and Technology in Computer Science Education, p. 199 – 202, 2002.
[6] Bergin, J., “Fourteen Pedagogical patterns for teaching computer science”, 2002. Available: http://csis.pace.edu/bergin/PedPat1.3.html