Using a Flipped Classroom to improve students' understanding of Software Design

Pragasen Mudali, University of Zululand

This project focused on an Introductory Software Engineering module offered at the University of Zululand. An analysis of past student performance in the Introductory Software Engineering module showed that students require additional learning support for the Software Design portion of the module.

The additional learning support for the Software Design topic should take the form of tutorials but the institution has a shortage of tutorial venues and also does not provide officially-timetabled tutorial sessions for the module being considered.

The flipped classroom model allowed the lectures to be offered asynchronously via recordings. The timetabled lecture sessions were re-purposed to offer synchronous Software Design tutorial sessions where the students engaged with relevant case studies and demonstrations.

This project aimed to:

1. improve student understanding and application of Software Design principles
2. adopt the flipped classroom model to offer students additional support for the Software Design topic
3. demonstrate to Departmental and Faculty colleagues that the flipped classroom model is viable given our particular context

The study employed a phased approach adapted from Mdunyelwa, Futcher and van Niekerk (2020)[1]. The four phases employed are depicted in Figure 1. Phase 1 attempted to better understand the students enrolled for the Introductory Software Engineering module. This phase gathered information relating to the learning styles prevalent amongst the student population.

[1] Mdunyelwa, V, Futcher, L, & van Niekerk, J 2020, 'A Framework for Teaching Secure Coding Practices' in Proceedings of the 49th Annual Conference of Southern African Computer Lecturers' Association (SACLA), 6-9 July, pp. 99-114.

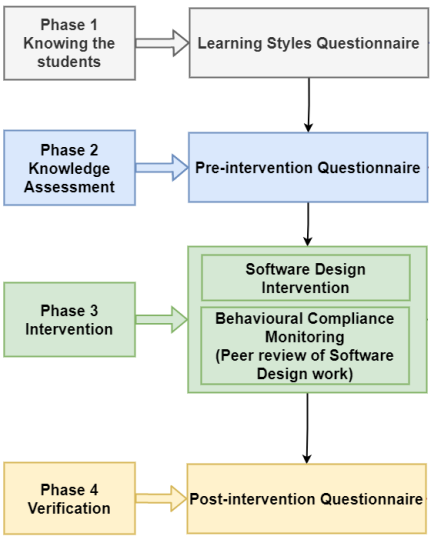

Figure 1 - Phased research approach (adapted from Mdunyelwa, Futcher and van Niekerk (2020)

The Learning Styles Indicator survey based on the Felder and Silverman learning style model[2] was used as the model was reported to be the most widely-used for this purpose.

There are four dimensions of learning styles in the Felder and Silverman model: Activist/Reflector; Sensing/Intuitive; Visual/Verbal; and Sequential/Global. The students surveyed[3] indicated strong preferences for the Reflector, Sensing, Visual and Sequential learning styles, as shown in Table 1. Reflectors think quietly and by themselves. Sensors prefer facts and use well-established methods to problem-solve. Visualists prefer multimedia and Sequentialists gain understanding in small, linear steps.

Phase 2 of the flipped classroom experiment assessed the prior Software Design knowledge gained by students in prior modules. The average performance was 57% in a pre-intervention quiz. This showed that students were exposed to Software Design in prior modules and did retain a useful amount of knowledge.

[2] Felder, RM, & Silverman, LK 1988, 'Learning and Teaching Styles In Engineering Education' in Engineering Education, vol, 78, no, 7, pp. 674-681.
[3] n=59

Table 1 – prevalent Learning Styles amongst students

Activist/Reflector		**Sensing/Intuitive**		**Visual/Verbal**		**Sequential/Global**	
Activist	Reflector	Sensing	Intuitive	Visual	Verbal	Sequential	Global
36%	**64**%	**82**%	18%	**73**%	27%	**64**%	36%

In Phase 3, a tailored intervention was designed to cater for the prevalent learning styles identified in Phase 1. Pre-recorded, dual-language lecture videos catered for the Visual preference. The lecture videos could be watched alone and replayed for Reflective learning before contact sessions. Processes were explained to cater for the Sensing preference and several examples were worked thorough in the videos to cater for the Sequential preference. Tutorial sessions were designed to consider case studies where students were shown the steps required to document a software design. The tutorials provided a structured and sequential learning opportunity to engage with the Unified Modelling Language (UML), a visual design language. Reflective formative assessments were provided and a Software Design assignment was offered where students had to follow a pre-defined sequence of steps to develop and document a design. The assignment was peer graded with a provided rubric to reinforce learning.

Phase 4 of the flipped classroom experiment was used to verify the effectiveness of the intervention. Students were surveyed on their preference for the dual-language lecture videos and a post-intervention Software Design quiz was administered. Despite the isiZulu language offering, only 32% of the respondents relied on both the English and isiZulu lectures. Of those respondents who relied on the lecture videos in both languages, 50% of these students indicated that their understanding of Software Design was better than their understanding of the other topics covered in the module that were only offered in English. The average performance in the post-intervention quiz was 73% and the comparison of the pre- and post-intervention scores, depicted in Figure 2 below shows that the intervention did improve students' understanding of Software Design.

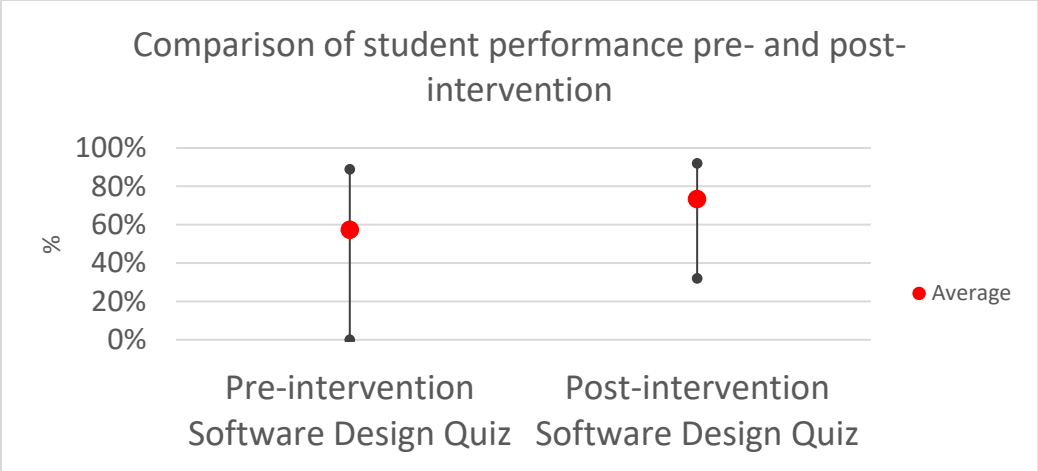

Figure 2 - Comparison of student performance pre- and post-intervention

The changes in the teaching practice of this module combined to successfully improve students' understanding of the Software Design topic. The flipped classroom model incorporating tutorial support will be extended to all of the topics within the Introductory Software Engineering module. The provision of the dual-language lecture videos acknowledged that language can be a learning barrier. Hence, the provision of lecture videos in the students' predominant home language addresses a social justice issue by recognising that student learning does not only have to be facilitated in English. The provision of the isiZulu lecture videos also contributes in a small way to the development of the language as an academic language for the teaching of Computer Science. The benefit attributed to those students who used the isiZulu version to improve their understanding of the corresponding English lecture, will see isiZulu versions of all lecture videos being provided for all topics in the Introductory Software Engineering module.

During the course of this project, various presentations have been made to Department, Faculty and institutional colleagues. It is anticipated that this pilot project will cause departmental and Faculty colleagues to take note of the use of the Flipped Classroom model and adopt it in pilot projects within their own modules. Over time, it is envisaged that departmental and Faculty teaching practices will make increasing use of the Flipped Classroom model to offer the full complement of lectures, tutorials and practical sessions on a weekly basis.