



A Study of Software Architects' Cognitive Approaches: Kolb's Learning Styles Inventory in Action

Mauricio Hidalgo^{1,2} , Hernán Astudillo³ , and Laura M. Castro⁴ 

¹ FDI - Universidad Finis Terrae, Santiago, Chile
mhidalgob@uft.edu

² DII - Universidad Técnica Federico Santa María, Valparaíso, Chile
mauricio.hidalgob@usm.cl

³ ITiSB - Universidad Andrés Bello, Viña del Mar, Chile
hernan@acm.org

⁴ Universidade da Coruña, A Coruña, Spain
lcastro@udc.es

Abstract. The multidisciplinary nature of software architects demands a diverse set of skills, ranging from technical expertise to interpersonal abilities. Within this domain, software architects are responsible for designing systems that adhere to quality standards, meet functional requirements, and align with organizational goals. However, educating or training software architects presents a challenge due to the complexity of their roles and responsibilities. To address this challenge, this paper proposes an approach to understanding the learning styles of software architects using Kolb's Learning Style Inventory (KLSI). It aims to provide a characterization of their teaching and learning preferences, thus facilitating the design and execution of educational strategies tailored to their specific needs. In conducting our research, we utilized LinkedIn as a platform to distribute the KLSI Test, ultimately gathering a sample comprising 18 Senior and Mid-Senior Software Architects. Through trend analysis of their responses, we consistently observed a discernible pattern. This led us to identify the Deciding Learning Style as the primary approach among the sample of software architects regarding their learning preferences. This finding offers initial insights into the predominant learning style within this profession, providing valuable guidance for educational practitioners and institutions aiming to optimize their training programs for software architects.

Keywords: Software Architects · Software Engineering Education · Learning Styles · Kolb's Learning Styles Inventory

1 Introduction

Software architects face challenges such as increasing system integration, emphasis on reuse, agility, and testability, and quality elements influenced by architectural choices [1]. Additionally, the essential elements of knowledge and key