Thesis Introduction Draft

Travis Dean
Department of Computer Science
Cal Poly SLO
tdean@calpoly.edu

1 Introduction

Software has become a large part of everyday life. This technology is a part of products from household games and computers to controlling medical devices. As software takes more control over parts of our everyday life, more care needs to be made to ensure the safety and quality of the software. This increases the overall health, wellbeing, and safety of the public the software affects.

In 1993, after deaths were incurred from issues such as faulty practices and lack of a software test plan on the Therac-25 [13], Texas requested a licensing process for licensing professional software engineers [12]. A professional engineering license represents that the bearer has demonstrated at least minimal competency for practicing his or her respective profession [1]. The Association for Computing Machinery (ACM) and the Institute of Electrical and Electronic Engineers (IEEE) started a joint effort in 1993 to pursue licensing software engineers [10]. There are four standard requirements for licensing an engineering discipline as a profession [6]:

1. A code of ethics
2. A body of knowledge
3. ABET Accreditation
4. A Principles and Practices of Engineering (PE) Exam

Since 1993, IEEE and ACM have made progress in these areas. Some areas have been advanced by only IEEE since ACM withdrew from the pursuit in 1998 [2]. The first three requirements have been met, with only the last requirement outstanding. The progress and status of each are outlined below. This paper analyzes these requirements with a focus on their impact to the final requirement, the PE exam.

1.1 Current State of Affairs

Software Engineering Code of Ethics

During their joint effort, ACM and IEEE developed and established the Software Engineering Code of Ethics in 1998 (SECOE) [10]. This code of ethics lays out the foundation of how software engineers shall act in an ethical manner in their profession. The code is broken up into 8 basic principles, and each principle is subdivided into more specific codes. The code also explicitly states that some aspects of the code are subjective, leaving the ethical decision ultimately up to the practicing software engineer.

The preamble to the SECOE emphasizes the responsibility and commitment the software engineer has towards the health, safety, and welfare of the public [11]. The code outlines how software engineers should act and perform in their profession. Since the SECOE does not cover knowledge areas, I do not analyze the code in this paper. However, it is still necessary to note the SECOE’s importance in the profession and its existence as mandatory for licensing to be existent for software engineers.

Software Engineering Body of Knowledge

ACM and IEEE, in their joint effort to increase the professionalism of software engineering, organized a task force to develop a unified body of knowledge for software engineers [7]. ACM formed a committee which in 1998 determined that ACM should withdraw from these efforts and stand against any movement which would further licensure for software engineers. ACM maintains that they will continue to further software engineering as a profession [2].

IEEE successfully created a software engineering body of knowledge (SWEBOK) in 2004 which is available for purchase in PDF or book format, or for free in HTML format on their website [7]. The SWEBOK establishes criteria and norms for professional practice in software engineering upon which "industrial decisions, professional certification, and educational curricula can be based" [2]. The key focus that this paper looks at is what SWEBOK establishes towards professional certification, or more specifically, the principles and practices of engineering exam.

The SWEBOK is broken up into ten knowledge areas for software engineers [7]. These ten knowledge areas are analyzed in detail later in this paper. The information laid out in these ten knowledge areas will also contribute to defining the material to be covered on the PE exam.

ABET Accreditation

ACM and IEEE pursued licensing from ABET for software engineers to strengthen the profession of the discipline. ABET accreditation is also one of the necessary steps towards software engineering licensure [4]. Software engineering became accredited in the early 2000's. 15 of 23 software engineering bachelor programs were ABET accredited...
in 2003[9]. Today, 2010, ABET reports that there are 19 software engineering programs that have obtained ABET accreditation [4].

ABET outlines general requirements for any engineering program to meet to be eligible for accreditation [7]. ABET has more specific guidelines for each particular engineering profession for specific curriculum, student, and faculty requirements before accreditation is received. ABET requires that at least one student has graduated from that institution with a degree in the specified area of study before an evaluation can take place for accreditation [7]. The curriculum specifications outlined by ABET can be used in the analysis and design of a software engineering PE exam specifications.

Software Engineering Principles and Practices of Engineering Exam

The last major milestone for licensing software engineers is the development of a principles and practices of engineering (PE) exam for that discipline. In Samuel Li’s paper on the ethics of licensing software engineers for safety critical applications, all 8 professional software engineers Sam interviewed agreed that software engineers should be licensed for safety-critical applications [12]. In an IEEE-CS survey of software engineers in industry conducted in September 2008, 62.9% of respondents agreed that software engineers should be licensed if they practice in areas affecting public health, safety, and welfare. In addition, 61.5% supported development of software engineering licensure through NCEES Model Law [5].

A minimum of 10 states need to request for NCEES to develop a PE exam before the development of a new exam can start [91]. In 2009, this number was reached, and IEEE-USA and NCEES will be working together to develop such an examination [61]. In IEEE-USA’s call for PE exam writers, Steven Barrett states that their committee consists of approximately 20 registered professional engineers from industry and academia who “represent the different technical specialties within electrical and computer engineering” [5]. The key issue to note here is that the writers are electrical and computer engineers, not software engineers. Yet, this committee is responsible for developing the software engineering PE exam. The question this raises is how will the exam be validated against the software engineering profession when no software engineers are working on the exam?

References


