Crossroads
The ACM Student Magazine

Introduction

by ***Jerry Guo***

...Typical look on the average student when asked about software engineering ....

The biologists have to contend with the creationism versus evolution debate. Computer scientists have their own controversy in defining software engineering. Is it an engineering discipline, a scientific pursuit, or an artistic craft? Back in 1996 when we published our first issue on software engineering, this slippery concept was debated and it hasn't been resolved since.

But one point is certain. Software engineers have gained significant legitimacy and clout in the last ten years. In the late 1990s, software development was responsible for a quarter of the growth in the United States' GDP and one-sixth of the increase in productivity, ultimately leading to $1 trillion of growth over the last decade.

Given the economic force of this field, it is no surprise that one out of every 200 American workers is a software engineer. To put that in perspective, the number of software engineers is about 60% of the total workforce for all types of traditional engineers. The dominance of software engineers within the engineering workforce means that even if they may not be completely accepted by traditionalists, software engineers are quickly changing the face of the field.

Yet software engineering is also not a completely feel-happy success story. Besides the tech bubble burst of a few years ago, many outsiders do not know the obstacles that software engineers deal with. The average commercial software developer writes about 12,000 lines of code per year. But about half of planned projects are cancelled because of fickle users. Another quarter are never completed by the software engineers because of lack of time or resources. Only the remaining quarter of projects are delivered successfully.

Because of the harsh realities of software engineering, this issue’s articles focus on good development practices through discussions of essential skills and case studies. In our first article, “Why are APIs Difficult to Learn and Use?” Christopher Scaffidi presents strategies taken from traditional engineering to make using an API less of a hassle.

Next, Girish Suryanarayana et al. reveal in “Architecting Trust-Enabled Peer-to-Peer File-Sharing Applications” that using a software architecture-based approach in incorporating trust into P2P applications is feasible and beneficial. In a case study, “RAS: A System for Supporting Research in Online Auctions,” Jarrod Trevathan and Wayne Read describe designing online auction software and present a client-server software model for conducting online English auctions.

Finally, in “TigerEvents: An Online Event Calendar System for Students by Students,” Chris Jordan, Oliver Baltzer, and Sean Smith give a road map on how to start an open source project on campus for the benefit of other students by presenting the path their ACM student chapter took in developing TigerEvents.

Our upcoming issues include Bioinformatics, Computer Gaming, and Graphics. As always, please email us (**crossroads@acm.org**) to submit articles, volunteer for the editorial board, give us feedback, or just chat about technology.

Biography

Jerry Guo (**ji.guo@yale.edu**) is an undergraduate student at Yale University. His main research interests are in bioinformatics and artificial intelligence. He is currently a staff reporter for the Yale Daily News and a rower on heavyweight crew. In his meager spare time, he enjoys sleeping, gaming on his XPS2, and mountain biking. He has served as Editor-in-Chief of Crossroads since August 2005 and been on the editorial board since 2003.