

Lyle Franklin
M.S. in Computer Science

Introduction: Master's in Computer Science at Stanford

I am applying to the Master's in Computer Science program at Stanford University. My area of interest is software engineering.

Experiences: Better software through industry and academic techniques

I have worked in both industry and academia, and these experiences have prepared me for graduate school at Stanford.

1.1 Beginning research at Notre Dame

In Summer '11, I gained my first exposure to research in the REU program at the University of Notre Dame, working with Dr. Aaron Striegel. My partner and I developed a data visualization tool for use in a sociological study, utilizing an iterative design process based on feedback from graduate students and professors. We practiced agile software development guided by research in data visualization and HCI, a mix of industry and academic techniques. We ended the summer by winning the poster competition, affirming the best software is made from both industry and academic practices.

1.2 Game development at BSU

The following Spring, I was selected to participate in a semester-long immersive learning project at Ball State University mentored by Dr. Paul Gestwicki. Twelve interdisciplinary students and I were to create an educational video game for our community partner, the Children's Museum of Indianapolis. For the duration of the project, we were no longer students but employees at a game development studio working 40 hours a week.

I was appointed the Technical Lead, and my main responsibility was communication between the developers and the graphic design team. We utilized agile software development techniques such as daily Scrum meetings, test-driven development, and regular code reviews. My exposure to these techniques prompted me to take a critical look at the process of creating software, beginning my interest in software engineering research.

Since the end of the semester, I have continued to work with Dr. Gestwicki to submit the game to academic conferences such as Meaningful Play. We are currently conducting empirical research to better understand the impact of educational video games. We will interview students playing the game, perform qualitative data analysis, and publish the results. These findings will help game designers develop more effective educational games, yet another example of software development guided by academic research.

1.3 Software engineering research at UIUC

In Summer '12, I was accepted to the UIUC ITI undergraduate research program working with Dr. Danny Dig, whose academic focus is software engineering. My project for the summer was to develop two automated refactorings in Java to support lambda expressions, which will be introduced

in Java 8. I worked with another intern and employees from Oracle to submit the refactorings as a patch to the NetBeans IDE, an open source project used by millions of developers.

We analyzed the code in open source projects in order to evaluate the applicability, value, and effort saved by our tool. This evaluation could only be accomplished through analysis of empirical data. Our findings showed that our refactorings were highly applicable and would provide value to developers. With such positive results, we decided to submit our work as a research paper to the International Conference on Software Engineering, one of the top conferences in computer science.

By the end of the summer, our tool was accepted as a patch to an open source project and our paper was submitted to a top tier software engineering conference. This experience convinced me I had the mettle to succeed at a graduate program like Stanford.

Interests: Further investigations in software engineering

During my time in graduate school, I will focus on expanding my knowledge of software engineering. Specifically, I am interested in developing tools to improve developer productivity, such as automated refactorings, bug detection software, and tools to enforce best practices. To accomplish this, I will take courses in software engineering, programming languages, and compilers during my graduate education. The undergraduate versions of these courses were the most useful of my education, and these skills proved invaluable in my internships.

But, developing these tools requires more than just technical skill. Creating the proper tool requires a design process guided by empirical findings. Research in software engineering is taking a critical look at how we do things and asking, “Is there a better way?” I wrote this statement of purpose in LaTeX and committed my changes to a version control repository. The process of creating just a simple text document can be greatly improved through software engineering.

While at Stanford, I will continue to gain skills from academia and industry and have the best possible preparation for my future career in software engineering.

The Future: Attending Stanford

I am applying to Stanford University because the interests of its faculty closely match my own. During my time in graduate school, I will focus on software engineering and programming languages. Stanford University employs experts in these fields such as Dr. Alex Aiken. In Summer ‘12, I took an online course on compilers through Coursera taught by Dr. Aiken. Not only was Dr. Aiken a skilled lecturer, he also helped to develop the Cool programming language and DeduceIt application which were used in the assignments. I was fascinated not only by the material in the course, but also by the technologies used to teach it. During my summer internship, I used the knowledge of parsing and abstract syntax trees from this course to build an automated refactoring open source project. I look forward to continuing my education with world-class faculty in person at Stanford University.