

Developer Industry Readiness

John Carmack - Dylan Lomax - Lillian Coar - Brandon Mingledorff

Abstract—With the growing importance of formal education for developers to find employment, an obvious question is whether or not that formal education actually prepares the developer for the workforce. We explore that question by asking if the programming language developers are taught in their programs are actually in demand and are present in their work lives.

I. OBJECTIVE

It is our goal to answer the question of whether or not a developer's first language is indicative of their success in the software engineering workplace. We wish to ascertain whether or not the programmer's choice of beginning language, or the language that their curriculum is taught in has any impact whatsoever in their career prospects, or in their ability to adapt and assimilate to a working environment typical to the modern software engineer, which seems incredibly saturated with positions requesting experience with web technologies and JavaScript frameworks.

II. MOTIVATION

Our motivation for pursuing these questions is three-fold. First, as students of computer science who are late in our formal education, the question of our preparedness for the workplace is an immediate one. We have long been sold the idea that the language of our formal training is inconsequential, and we would like to collect the data and analyze it for ourselves to ascertain the validity of such claims. Secondly, a happy consequence of this analysis is that our team will become more conversant in these in-demand technologies and philosophies as we learn how to collect, analyze and scrub large sets of data. Finally, perhaps our findings may illuminate issues within the formal training of computer scientists and software engineers and steps might be taken by the purveyors of such curricula to mitigate gaps in student preparedness.

III. DATA

The collection of data to answer these questions will come from a multitude of sources.

A. Organizational data

We will query organizational web pages such as the CS departments of universities for data relating to the languages that their departments instruct in in addition to the employment rate of their graduates.

B. Government data

We will also query the Bureau of Labor Statistics for employment information.

C. Social Media Polling

Social media platforms such as reddit can prove to be very valuable sources of information when it comes to anonymous polling. Version control applications like bitbucket and github will also be useful in obtaining data related to the prevalence of certain programming languages.

IV. TEAM MEMBER RESPONSIBILITIES

Member responsibilities are relatively flexible to date, with the following guidelines in place for general task assignment. These are the relatively flexible assignments thus far:

A. John Carmack

- 1) Data cleaning.
- 2) Visualization
- 3) Documentation

B. Dylan Lomax

- 1) Data Acquisition.
- 2) Data Cleaning
- 3) Social Media Surveying

C. Lillian Coar

- 1) Data Cleaning
- 2) Floating Responsibility

D. Brandon Mingledorff

- 1) Data Acquisition.
- 2) Data Cleaning
- 3) Social Media Surveying

V. TIMELINE

- 2 weeks (October 8th) Finalization of data sources and tools to be used for gathering and visualization, setting up access where necessary.
- 2 weeks (October 22nd) Gathering of data using sources and tools chosen
- 1 week (October 29th) Code testing and bug fixing, quality assurance of data to make sure it is ready for visualization
- 2 weeks (November 12th) Visualization of data using tools chosen
- 1 week (November 19th) Creation practice of final presentation

VI. EXPECTED OUTCOMES

It is reasonable to assume that the formal training computer science students receive is not sufficient to become competitive as an applicant to industry positions, especially in a space dominated by web development positions. The hypothesis is that a student's beginning language is less consequential to success in the workplace employment than the ability to translate that knowledge to other technologies and be able to learn and utilize them rapidly.