Master Thesis Title

Authors

1 February - 30 June

Date	Svetomir Kurtev
Date	Tommy Aagaard Christensen

AALBORG UNIVERSITY
STUDENT REPORT

Department of Computer Science
Computer Science
Selma Lagerlöfs Vej 300
Telephone 99 40 99 40
Telefax 99 40 97 98
http://cs.aau.dk

Title:

Working Title

Project period:

2 February - 30 June

Project group:

Participants:
Svetomir Kurtev
Tommy Aagaard Christensen

Supervisor:
Bent Thomsen

Abstract:

Placeholder abstract

Pages: 8

Appendices: 0

Copies: 2

Finished: 30 June 2016

Preface

The following report was written by Svetomir Kurtev and Tommy Aagaard Christensen in accordance with the conclusion of the tenth and final semester of the Computer Science Master Program at Aalborg University.

We would like to thank Bent Thomsen for the help and guidance he provided us with throughout the development of the project.

Contents

Chapter 1

Introduction

Computer programming has increasing relevance to today's advancement of technologies. Therefore, existing and established programming languages are constantly improved and new ones are created to meet that demand. The languages which are considered most suitable for introductory programming, are being adopted by educational institutions as part of their computer science curriculum e.g. Java, Python and more recently, Scratch TODO: Maybe a reference(s) is needed here . Similarly, some languages are considered arguably better than others in their intended purpose in the software industry. However, formal evaluation methods for assessing programming languages are very few and limited in their use and most evidence gathered to support such claims are anecdotal in nature.

1.1 Initial Questions

Part I

Problem Analysis

Chapter 2

Related Work

Programming languages have been used for years but still there has not been established a robust and efficient way to asses and evaluate them. However, plenty of research has been done on the topic and this chapter explores some of it.

M. Farooq et al. 2014 TODO: ref needed, wrote a paper introducing an evaluation framework which provides a comparative analysis of widely used first programming languages (FPLs), or namely languages which are used as a first language for teaching introductory programming. The framework was based on technical and environmental features and it was used for testing the suitability of existing imperative and object oriented languages as an appropriate FPL. In order to support their framework, they came up with a customizable scoring function for computing a quantitative suitability score for a given language. Lastly, they evaluated the most widely used FPLs by computing their suitability scores.

Part II

Experiment Setup

Part III

Conclusion

Chapter 3

Conclusion

Part IV

Bibliography