

---

# Assessing Documents by Comprehension Effort

Master's Thesis submitted to the  
Faculty of Informatics of the *Università della Svizzera Italiana*  
in partial fulfillment of the requirements for the degree of  
Master of Science in Informatics

presented by  
Talal El Afchal

under the supervision of  
Prof. Michele Lanza  
co-supervised by  
Prof. Gabriele Bavota

September 2017



---

I certify that except where due acknowledgement has been given, the work presented in this thesis is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; and the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

---

Talal El Afchal  
Lugano, 1 September 2017



*To my beloved*



Your living is determined not so much by  
what life brings to you as by the attitude  
you bring to life; not so much by what  
happens to you as by the way your mind  
looks at what happens

Gubran Khalil Gubran





# Abstract

Recommender systems for software developers have become increasingly popular in recent years. These systems combine several methodologies to provide suggestions that meet the developer's needs. The recommender systems collect data from online resources as blogs, forums, Q&A websites, and suggest documents or piece of code that are most likely helpful to the developers. However, these systems are not taking into consideration an important aspect as the comprehension effort which may vary depending on the document familiarity and readability. Usually developers are more interested in documents which they are familiar with. By calculating the comprehension effort, the recommender system can re-rank the documents and suggest the most comprehensive and appropriate ones to the developer. In this work, we present our approach to calculating the effort, by creating a language model able to capture a document familiarity, that we combine with the document readability.



# Acknowledgements



# Contents

<b>Contents</b>	<b>xi</b>
<b>List of Figures</b>	<b>xiii</b>
<b>List of Tables</b>	<b>xv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Context . . . . .	1
1.2 Objective and Results . . . . .	3
1.3 Structure of the Thesis . . . . .	4
<b>2 State of the Art</b>	<b>5</b>
2.1 Semantics Code Search . . . . .	5
2.2 Code Search Engines . . . . .	5
2.3 Libra . . . . .	5
<b>3 Approach</b>	<b>7</b>
3.1 Overview . . . . .	7
3.2 Language Model . . . . .	7
3.3 Stormed Island Parser . . . . .	7
3.4 Training the Language Model . . . . .	7
3.5 Accounting for Readability . . . . .	7
<b>4 Study Design</b>	<b>9</b>
4.1 Research Questions . . . . .	9
4.2 Data Collection and Analysis . . . . .	9
4.3 Replication Package . . . . .	9
<b>5 Results</b>	<b>11</b>
<b>6 Threats to Validity</b>	<b>13</b>
<b>7 Conclusion</b>	<b>15</b>
<b>8 Short title</b>	<b>17</b>
8.1 The first section . . . . .	17
8.2 The second, math section . . . . .	18
8.3 third . . . . .	18

<b>A Some retarded material</b>	<b>21</b>
A.1 It's over... ..	21
<b>Glossary</b>	<b>23</b>
<b>Bibliography</b>	<b>25</b>

# Figures





# Tables



# Chapter 1

## Introduction

### 1.1 Context

Software systems complexity is increasing, and new technologies are introduced constantly. The software developers often have to work with new technologies which they are not familiar with, and as increasingly more comes out, the amount of information that they need to know will increase. Android, for example, was introduced 10 years ago in 2007 and nowadays there are more than three million applications available on the Google store.

When android started to become popular, developers had to learn this technology and to stay updated with each new version release. They had to figure out how activities work in Android, and how to use several APIs to implement different tasks assigned to them. Where do they start from? Is there some tutorial on the web where they can learn how to use a specific API or can they find a piece of code that can be reused in their application?

If we search on Google for an Android tutorial we will find 22 million documents <sup>1</sup>, even if we search for a specific field, as an Android Bluetooth tutorial <sup>2</sup>, we get 6 million documents which still a huge number.

StackOverflow is one of the most popular Q&A websites for developers, where a million of questions are tagged as Android <sup>3</sup>.

Github host more than 500 thousand Android repositories <sup>4</sup>, with millions of lines of code. Those numbers are gigantic, and it is obvious that finding the most suitable document is not obvious, since given the big number of available documents it is not feasible for the developer to check all of them and choose the right one.

Searching for documentation and tutorials is a crucial step in learning a new technology. The developers can find a bunch of online resources as blogs, forums, Q&A websites, but the real challenge is to find the most suitable one for their needs.

---

<sup>1</sup><https://www.google.ch/search?site=&source=hp&q=android+tutorial&oq=android+tutorial>

<sup>2</sup><https://www.google.ch/search?client=safari&rls=en&biw=1440&bih=839&q=android+bluetooth+example>

<sup>3</sup><https://stackoverflow.com/questions/tagged/android>

<sup>4</sup><https://github.com/search?utf8=&q=android&type=>

[Singer et al., 1997] reported in 1997 that the most frequent developer activity was code search, and [Sadowski et al., 2015] did a case study on how developers at Google search for code. They figured out that programmers are generally seeking answers to questions about how to use an API, what code does, why something is failing, or where the code is located. The interesting point in this study was the fact that most searches focus on code that is familiar or somewhat familiar to the developers.

Therefore, we believe that recommender systems for software developers have to take into consideration the familiarity of a document when they suggest it to the developer. But first, we have to mention the proposed definition by the organizers of the ACM International Conference on Recommender Systems:<sup>5</sup>

*“Recommendation systems are software applications that aim to support users in their decision-making while interacting with large information spaces. They recommend items of interest to users based on preferences they have expressed, either explicitly or implicitly. The ever-expanding volume and increasing complexity of information [...] has therefore made such systems essential tools for users in a variety of information seeking [...] activities. Recommendation systems help overcome the information overload problem by exposing users to the most interesting items, and by offering novelty, surprise, and relevance.”*

Now that we defined what is a recommender system, we can mention the recommender systems for software engineering (RSSE) definition given by Robillard et al. [2010]:

**An RSSE is a software application that provides information items estimated to be valuable for a software engineering task in a given context.**

The RSSE definition highlights the importance of the **context** and the **valuable information**, therefore RSSEs are an important support for programmers to find the information they should know, and the RSSEs have to consider and evaluate alternative decisions. We believe that effort to comprehend a document must be a part of the document evaluation.

---

<sup>5</sup><https://recsys.acm.org/recsys09>

Understanding a document is a cognitive process, and it depends on the human brain intelligence, but we all agree that if we are familiar with a subject, we will comprehend it with less effort. A computer engineer comprehends a document that explains how to implement a sorting algorithm, with much less effort compared to a document that explains a constitutional law, and the reason is not that the algorithm is not complex, but because a computer engineer is more familiar with sorting algorithms.

In this study, we go further than this, and we try to evaluate more interesting situations, as for example:

Given two documents that have the same subject, how can we decide which one is easier to comprehend?

In order to answer this question we need to introduce two concepts:

- **familiarity**: how much are we familiar with the document content?
- **readability**: how difficult is it to read the document?

The comprehension effort can be derived from the document familiarity and readability.

For example, if we have two documents where, in the first one we have a sorting algorithm implemented in Java, and in the second one a sorting algorithm implemented in Fortran, and the developer is more familiar with Java. And we want to use a tool that gives us a score that indicates which document requires less effort to be comprehended, where a higher score indicates a big effort.

We expect that the first document must have a lower score since logically it requires less effort to be comprehended by developers who are more familiar with Java.

What if both documents have a sorting algorithm implemented in the same programming language? which one will have a lower score?

In this case, the document readability will have a big impact on the comprehension effort.

Certainly, the developer will prefer to read the document with the best readability.

## 1.2 Objective and Results

Our main goal in this thesis is to assess documents by their comprehension effort, which can be used by RSSE to improve their suggestions.

In order to calculate the comprehension effort, we need to find a way to calculate the familiarity, and we need to evaluate our approach to understand if it effectively works.

As a first step, we select a big set of documents, which represent the hypothetical developer knowledge, where each document contains code and natural language. Then we create a **Language Model** that we train with these documents. In this way, we were able to simulate a programmer who is familiar with these documents. Once we trained the language model, we were able to evaluate the familiarity of a given set of documents.

In this experiment, the language model approach satisfied our expectation in capturing the document familiarity.

On the top of this experiment, we did a case study, where we give a set of tutorials to a programmer, and after reading the tutorials, the programmer was asked to evaluate a set of documents, some of them are related to the tutorials and some are not. The evaluation is based on the comprehension effort, by giving a score between 0 and 5 ...

## 1.3 Structure of the Thesis

This thesis consists of seven chapters:

1. **Introduction**
2. **State of the Art** describes the existing related work as code search engines and recommender systems.
3. **Approach** describes our approach to calculate the comprehension effort.
4. **Study design** in this chapter we discuss the research question, the data extraction process, and the analysis method, and the replication package
5. **Result** describes and discuss our experiment and case study results and their implication.
6. **Threat to Validity** describes our assumptions, and the possible threats that could affect the results validity.
7. **Conclusion** this chapter is a summarization of our work, where we present some ideas for a possible future work.

## Chapter 2

# State of the Art

### 2.1 Semantics Code Search

### 2.2 Code Search Engines

### 2.3 Libra





## Chapter 3

# Approach

3.1 Overview

3.2 Language Model

3.3 Stormed Island Parser

3.4 Training the Language Model

3.5 Accounting for Readability



## Chapter 4

# Study Design

### 4.1 Research Questions

### 4.2 Data Collection and Analysis

### 4.3 Replication Package



## Chapter 5

## Results



## Chapter 6

# Threats to Validity





Chapter 7

Conclusion



## Chapter 8

# A chapter title which will run over two lines — it's for testing purpose

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

### 8.1 The first section

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst.

Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

## 8.2 The second, math section

**Theorem 1 (Residue Theorem).** Let  $f$  be analytic in the region  $G$  except for the isolated singularities  $a_1, a_2, \dots, a_m$ . If  $\gamma$  is a closed rectifiable curve in  $G$  which does not pass through any of the points  $a_k$  and if  $\gamma \approx 0$  in  $G$  then

$$\frac{1}{2\pi i} \int_{\gamma} f = \sum_{k=1}^m n(\gamma; a_k) \text{Res}(f; a_k).$$

**Theorem 2 (Maximum Modulus).** Let  $G$  be a bounded open set in  $\mathbb{C}$  and suppose that  $f$  is a continuous function on  $G^-$  which is analytic in  $G$ . Then

$$\max\{|f(z)| : z \in G^-\} = \max\{|f(z)| : z \in \partial G\}.$$

## 8.3 A very very long section, titled “The third section”, with a rather short text alternative (third)

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Some Test

```

1  import IntSpec, ItemSpec;
2
3  sort cart;
4
5  constructors
6  create() → cart;
7  insert(cart, item) → cart;
8  observers
9  amount(cart) → int;
10 transformers
11 delete(cart, item) → cart;
12
13 axioms
14 forall c: cart, i, j: item
15
16 amount(create()) = 0;
17 amount(insert(c,i)) = amount(c) + price(i);
18 delete(create(),i) = create();

```

```
19 delete(insert(c,i),j) =  
20 if (i == j) c  
21 else insert(delete(c,j),i);  
22 end
```

As you can easily see from the above listing ? define something weird based on the BPEL specification [?].

## Appendix A

### Some retarded material

#### A.1 It's over. . .

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.



## Glossary



# Bibliography

M. Robillard, R. Walker, and T. Zimmermann. Recommendation systems for software engineering. *IEEE Software*, 27(4):80–86, July 2010. ISSN 0740-7459. doi: 10.1109/MS.2009.161.

Caitlin Sadowski, Kathryn T. Stolee, and Sebastian Elbaum. How developers search for code: A case study. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, ESEC/FSE 2015, pages 191–201, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3675-8. doi: 10.1145/2786805.2786855. URL <http://doi.acm.org/10.1145/2786805.2786855>.

Janice Singer, Timothy Lethbridge, Norman Vinson, and Nicolas Anquetil. An examination of software engineering work practices. In *Proceedings of the 1997 Conference of the Centre for Advanced Studies on Collaborative Research*, CASCON '97, pages 21–. IBM Press, 1997. URL <http://dl.acm.org/citation.cfm?id=782010.782031>.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultricies. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.