Chapter 1

Risk analysis

1.1 Risks

This document evaluates and weights all possible risks and defines actions to minimize them as good as possible.

1.1.1 Unclear requirements

Requirements are somehow vague at the beginning of each project and if they are not well defined as soon as possible, they stay vague trough out the whole project and this can lead to a disaster.

1.1.2 New technologies

The new technologies which are present in this project are:

- Gradle
- Travis CI
- Python

Each of them brings his own risk.

1.1.3 Interface Boilerpipe

The Boilerplate algorithm needs to be integrated into the text extraction framework. Every interface of an external component is a possible risk factor.

1.1.4 Interface Justext

The Justext algorithm needs to be integrated into the text extraction framework. Every interface of an external component is a possible risk factor.

1.1.5 Implementation RSS algorithm

The development and implementation of a new algorithm is predestined to generate risks.

1.2 Risk analysis

Risk	Impact	Probability of occurrence	Risk factor
Unclear requirements	2	4	8
New technologies	3	3	9
Interface Boilerpipe	5	1	5
Interface Justext	5	5	20
Implementation RSS algorithm	1	5	5

1.3 Consequences

1.3.1 Unclear requirements

As I am working with the client each and every day, it is very easy prevent missunderstanding with asking the client at once. Even though misunderstandings can occur between student and expert. To prevent this, it is necessary to have a document to define the requirements as soon and as exact as possible. This well be done in the form of the system requirement specification in the first mile stone. Possible ambiguities can be clarified at the first mile stone meeting.

1.3.2 New technologies

It is important to do prototyping with new technologies in the first phase of the project to eliminate these risks as soon as possible.

Gralde and Travic CI are needed in the first mile stone to set up the programming environment. So if there is any problem it will occur in a very early stage of the project and a possible solution can be found.

The risks about python are related to the chapter 1.3.4.

1.3.3 Interface Boilerpipe

This risk is rated much lower than the Justext interface becaus it's implementation is in Java and it provides a Java API. Never then less a prototype should be done as soon as possible to prevent any nasty surprises with the interface.

1.3.4 Interface Justext

This point is classified as the biggest risk of all. This is because the implementation is in Python and it is not clarified yet how it will be integrated into the text extraction framework. An analysis of possible solution with prototypes needs to be done as soon as possible.

1.3.5 Implementation RSS algorithm

This risk has a very high probability of occurrence because it is very likely that a development and an implementation of a new algorithm is going to cause problems. There is no real solution to that risk. But because of this requirement is nice to have, the impact on the outcome of the project is very low. Further more Patrik Lengacher, the tutor of this project, is very experienced in this subject area and will be able to help out if any problems occur.