1 Cheat Sheet (TMP)

Dieses CheatSheet wird später wieder entfernt.
Hier ist Text.

Jetzt ist Text nach so nem Doppelslashteil.
Neuer Absatz.
Neue Zeile. Hehe, wo bin ich?

1.1 Untertitel

1.1.1 Das ist die tiefste Titelebene

Ich bin Text.

Figure 1: My caption

Davor ein Bild.
Mehr dazu in Abbildung 1.

1.2 Quellen

Und das wäre ein zweiter Absatz [1].
Wie einer auf 20min sagte:[2]

> Immer mehr europäische Länder verhängen im Kampf gegen das Virus eine Ausgangssperre.

Beachten sie die Fussnote [1]

1.3 Aufzählung

- Erstens
- Zweitens

1. Erstens
2. Zweitens

Erstens

Zweitens

[1] Ich bin die Fussnote

1.4 Tabelle

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

Figure 2: My table

Das war also Tablle 1.4.

1.5 Code

Fogelnd Code in C#

```
public void SendInformation(string msg)
{
    SendMessage(”INFO”, msg);
}
```

Listing 1: My Caption

Da wär jetzt so code, siehe Listing 1

HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

FHO Fachhochschule Ostschweiz

Google

Dafny Language Server
Bachelor Thesis

Department of Computer Science
University of Applied Science Rapperswil

Spring Term 2020

Authors:	Marcel Hess, Thomas Kistler
Advisors:	Thomas Corbat (lecturer), Fabian Hauser (project assistant)
Expert:	Guido Zgraggen (Google)
Counter reader:	Prof. Dr. Olaf Zimmermann

2 Abstract

Table of Contents

3 Introduction

2do: iwo den Satz ”Zielgruppe die HSR Studenten” einbauen. ”Messbarkeit von Erfolg.”

3.1 Problem Domain

3.2 Relevance

- ...
- ...

3.3 Outlook

Das ist ein Absatz [1].

Und das wäre ein zweiter Absatz. [1]

4 Analysis

4.1 Continuous Integration (CI)

Continuous integration is a verry important part for code quality improvement and colaboration. Unfortunately, the CI process in our student research project extended to almost the entire semester [3].

According to our project plan, we wanted to work on open points regarding the CI initially and have completed the theme accordingly for the remaining duration of the bachelor thesis.

4.1.1 Initial Situation

We achieved in our client CI that code was analyzed with SonarQube and the build failed if it contained TypeScript errors [3]. We did not achieve it within reasonable time headless integration test [3].

On the server side we reached the build process as well as the dafny tests and our own unit tests [3]. Automated integration tests and code analysis by SonarQube remained outstanding [3].

4.1.2 Aimed Solution

According to our research, a major problem was that the scanner for sonarqube does not support any other languages besides C# [4]. This means that in addition to C# in a project, TypeScript (for the client) cannot be analyzed simultaneously. Furthermore there are also single Java files in Dafny project. This also led to conflicts in the Sonar analysis in our student research project [3].

As a simple solution we decided to separate the client (VSCode plugin) and server (Dafny Language Server) into two separate git repositories. This not only simplifies the CI process but also ensures a generally better and clearer separation.

As a result, the client could still be easily analyzed with the previous Sonar scanner. For the Language Server in C# a special Sonar scannerfor MSBuild had to be used, which publishes the analysis in a separate SonarCloud project [2]. Beside the code from our Lanugae server the whole Dafny project code is now analyzed by sonar. This can be very helpful for code reviews.

The only downside is that the code coverage is not analyzed. For .NET OpenCover is a very common tool for code coverage analysis. Unfortunately, it only works on windows and not on our linux CI server [5]. Other tools that works with mono Support .NET Core but not Framework. During our research we came across monocov [6]. This tool would support mono for .NET Framework. Unfortunately this project was archived and has not been supported for almost 10 years [6].
Since we would not gain much added value with sonar code coverage, we decided not to pursue this approach any further. The coverage information is provided by the locally installed IDEs anyway.

For an easier testability of the CI, we now also used local docker. This allows us to test CI customizations efficiently. See the developer documentation for more details [2].

The headless integration tests were a bit more tricky. In consultation with our supervisor, we have removed these tests from the client project and replaced them with own specially written integration tests on the server side.

4.1.3 2do - Kapitelaufteilung komisch

Ich hab hier jetzt in der Analyse auch schon die Lösung vorabgegriffen. Sollen wir das splitten? Bricht das nicht den Lesefluss? Evt besprechen.

5 Design

6 Implementation

7 Result

8 Conclusion

9 Project Management

Glossary

IDE Eine integrierte Entwicklungsumgebung für die Herstellung von Software. 10

References

[1] Rafael Krucker and Markus Schaden. *Visual Studio Code Integration for the Dafny Language and Program Verifier.* `https://eprints.hsr.ch/603/`. HSR Hochschule für Technik Rapperswil, 2017.

[2] Marcel Hess and Thomas Kistler. *Developer Documentation.* HSR Hochschule für Technik Rapperswil, 2020.

[3] Marcel Hess and Thomas Kistler. *Dafny Language Server Redesign.* HSR Hochschule für Technik Rapperswil, 2019/20.

[4] *SonarCloud for C# Framework Project.* URL: `https://community.sonarsource.com/t/sonarcloud-for-c-framework-project/17132`. (Accessed: 23.03.2020).

[5] *OpenCover.* URL: `https://github.com/OpenCover/opencover`. (Accessed: 23.03.2020).

[6] *monocov.* URL: `https://github.com/mono/monocov`. (Accessed: 23.03.2020).

Anhang (Entwickler Doku)