Juho Kettunen

Maintainability in cloud-native architecture

University of Jyväskylä

Faculty of Information Technology

Author: Juho Kettunen

Contact information: `juho.kettunen@student.jyu.fi`

Supervisor: Oleksiy Khriyenko

Title: Maintainability in cloud-native architecture

Työn nimi: Ylläpidettävyys pilvinatiiveissa arkkitehtuureissa

Project: Research plan

Study line: Software and Telecommunication Technology

Page count: 11+0

Abstract: Goal of the thesis is to find out how maintainability is addressed during the architectural design phase of cloud-native software development lifecycle. To this end, I will conduct a survey among cloud architects of company Nordcloud, where I work. First I will ascertain the perceived importance of maintainability. Then I will categorize the suggested approaches for addressing maintainability and relate this to the respondents' years of experience in the target domain. Finally, I will compare the survey results to approaches suggested in the literature.

Keywords: maintenance, maintainability, public cloud, cloud-native, cloud architect, architecture, software architecture, Master's Theses

Suomenkielinen tiivistelmä: Tutkielman tavoitteena on selvittää kuinka ylläpidettävyys huomioidaan pilvinatiivien sovellusten arkkitehtuurisuunnitteluvaiheessa. Tämän saavuttamiseksi suoritan kyselyn Nordcloud-yrityksen pilviarkkitehtien keskuudessa. Ensin määritän kuinka tärkeänä ylläpidettävyyttä pidetään. Sitten kategorisoin ehdotukset ylläpidettävyyden huomioimiseen, ja suhteutan ne vastaajien kokemuksen määrään kohdealueelta. Lopuksi vertaan kyselyn tuloksia kirjallisuudessa ehdotettuihin lähestymistapoihin.

Avainsanat: ylläpito, ylläpidettävyys julkinen pilvi, pilvinatiivi, pilviarkkitehti, arkkitehtuuri, ohjelmistoarkkitehtuuri, pro gradu -tutkielmat

Contents

1 Introduction

Goal of this thesis is to find out how maintainability is addressed during the architectural design phase of cloud-native software development lifecycle. I chose this topic for three reasons:

1. Choices made during the architectural design phase cascade into development and maintenance phases. Mistakes and oversights are slower and more expensive to correct later.

2. Maintenance phase of the software lifecycle is prevalent. It takes up the majority of total lifetime and costs.

3. Cloud-native approach helps reduce time-to-market and move costs from capital expenditure to operational expenditure. Utilizing a public cloud platform allows you to leverage a highly scalable, reliable and secure infrastructure and a wide variety of easily integratable services.

2 Literature review

I will use these sources for literature:

- "JYKDOK international articles" (2023)
- "Google Scholar" (2023)
- Snowballing from bibliographies of promising sources, e.g. Kaihlavirta (2022)
- + other databases or seach engines that are discovered during research

Example search terms:

- "cloud native" AND maintainability
- "cloud native" AND architecture
- architecture AND maintainability
- "cloud native" AND architecture AND maintainability

3 Research questions

I aim to answer three research questions:

- How much importance do cloud architects place on maintainability during the design phase?
- How to address maintainability concerns when architecting cloud-native applications?
- Do the recommendations from literature match the views of architects working in the field?

4 Method

I will conduct a survey that offers real-life insights into the research questions. It is targeted at cloud architects of company Nordcloud, where I work.

First I will ascertain the perceived importance of maintainability. Then I will categorize the suggested approaches for addressing maintainability and contrast this to the respondents' years of experience in the target domain. Finally, I will compare the survey results to approaches suggested in the literature.

5 Gathering the data

5.1 What

Survey will manifest as a Google Form. It will include a data collection disclaimer and an agreement checkbox to proceed.

Survey might include these questions:

1. How would you prioritize these software quality metrics when designing cloud-native architecture?
2. How can one address maintainability through platform- and technology choices?
3. How can one address maintainability through architecture?
4. Years of experience with cloud-native architecture?
5. Years of experience with software architecture in general?
6. Years of experience in IT?

Question 1 will have a selection of software quality metrics that the architect should set in a priority order.

Questions 2 and 3 are free-text answers. I will interpret the answers to find common categories.

Questions 4 through 6 are single-selection questions, with these ranges:

- 0-2
- 2-5
- 5-10
- 10+

There will be an optional feedback question in the end of the survey.

5.2 Who

Target audience is cloud architects working at Nordcloud, an IBM company.

5.3 How

I will post links to the survey on Nordcloud's internal Slack channels to reach most of our cloud architects. These communities will likely be targeted:

- #tech-infra
- #tech-dev
- #aws
- #azure
- #google-cloud

Some of these communities have overlapping audience, but the total reach is about 1600 people. I assume our company to employ a few dozen cloud architects. My goal is to get 10-20 answers in order to conduct any statistical analysis on the data.

5.4 When

Survey will be sent out during Q2 of 2023. I will repost the survey in Slack up to 4 times once a week until I decide I have enough answers to proceed.

5.5 Storage of data and privacy concerns

I will use my JYU Google account for all Google-related functions.

Survey will be anonymous. The form requires logging in with a Google account, but email addresses are not collected. Answers are stored in Google Forms until the thesis is complete, after which I will delete the survey.

Analysis is stored in Google Sheets until the end of my study rights and subsequent removal of the Google account.

Possible Python scripts used for analysis will be stored in my personal Github repository perpetually. This is OK because it won't contain any personal data.

6 Analysis

6.1 What and how

- Interpret and list categories from answers to questions 2 and 3.
- Calculate the most popular categories and point out possible outliers.
- Compare categories to solutions proposed in literature.
- Calculate statistical correlation between years of experience and:

  - perceived importance of maintainability
  - number of categories proposed in 2 and 3
  - popularity of categories proposed in 2 and 3
- (Create nice graphs to visualize the above)

6.2 Tech used

- Google Forms
- Google Sheets
- Python scripting for data analysis and graphs, if needed

Bibliography

"Google Scholar". 2023. Visited on February 26, 2023. http://scholar.google.com.

"JYKDOK international articles". 2023. Visited on February 26, 2023. https://jyu.finna.fi/primo.

Kaihlavirta, Johanna. 2022. "Time tracking in software maintenance service". Master's thesis, University of Jyväskylä. https://jyx.jyu.fi/bitstream/handle/123456789/84170/URN%3aNBN%3afi%3ajyu-202212015437.pdf?sequence=1&isAllowed=y.