

# **Exposé – *Evaluating Kotlin Multiplatform for cross-platform development of Android and Web Applications (BA)***

Robin Holzwarth, robin.holzwarth@stud.h-da.de

Darmstadt, May 31, 2023 - 14:57

## **1 Introduction and Motivation**

The significance of mobile applications and websites as end-user platforms has increased significantly over the past decade. Today, nearly every industry uses these platforms to interact with customers, provide services, and conduct business operations. As this trend continues, there is a greater demand than ever for cost-effective and efficient software development that targets multiple platforms.

In contrast, traditional approaches to software development frequently require keeping separate codebases for various platforms. This fragmented approach not only increases development and maintenance costs, but can also lead to platform inconsistencies and a variety of other potential issues.

In response to these challenges, the IT industry has witnessed a rise in the development and adoption of cross-platform frameworks. These frameworks offer the appealing possibility of "write once, run anywhere," allowing developers to maintain a single codebase that can be deployed across multiple platforms. This strategy has the potential to streamline the development process, reduce costs, and assure a uniform user experience across all platforms.

However, the market for cross-platform frameworks is extremely competitive, with numerous options available, each with its own set of advantages and disadvantages. This variety represents a double-edged sword for organizations. On the one hand, it provides a great deal of flexibility in selecting the framework that best meets their particular requirements. On the other hand, it poses a huge challenge: how to select the most appropriate technology or framework for their particular use case. Given the significance of this decision and its complexity, there is an urgent need for a more structured and exhaustive method of evaluating cross-platform frameworks.

Yet, defining a list of requirements is only part of the solution. Equally important is the evaluation of how well existing frameworks meet these requirements. Since Kotlin Multiplatform (KMP) has emerged as a viable option for cross-platform development as of 2023, it is among the candidates that firms can consider when deciding on the cross-platform framework that best suits their needs. KMP enables developers to share as much non-UI code as they desire while writing UI code in the platform's native language. This strategy differentiates KMP from other cross-platform frameworks, presenting both new opportunities and challenges. However, there is currently little to no literature evaluating KMP, which hinders companies in making an informed decision about its viability. Therefore, the second part of this thesis will address this gap by evaluating Kotlin Multiplatform for cross-platform development of Android and Web applications, against the developed list of criteria. This assessment will contribute valuable

insights to the industry and aid companies in deciding whether KMP is a fitting choice for them.

## **2 Problem Statement and Goal**

The problem at the heart of this thesis is the absence of an exhaustive and up-to-date set of requirements for modern cross-platform frameworks. In addition, there are insufficient empirical data concerning the performance of KMP in relation to these requirements. The study focuses primarily on two aspects. Its primary objective is to define a list of requirements for modern cross-platform frameworks. This will be accomplished through the use of literature reviews, expert interviews and a questionnaire. Second, it will evaluate the performance of KMP for cross-platform Android and Web development against the developed set of requirements.

### **2.1 Research Questions**

- In the context of cross-platform development, what are the critical requirements for a modern cross-platform framework, as identified through a systematic review of existing literature, an analysis of expert opinion and a questionnaire?
- How does KMP for Android and Web development perform when evaluated against the established requirements for modern cross-platform frameworks?

### **2.2 What's the goal of this work?**

The goals of this bachelors-thesis are:

- To develop a comprehensive requirements catalog for modern cross-platform frameworks, which can serve as a benchmark for evaluating various technologies in the field.
- Using a real-world application to empirically evaluate KMP's performance against the established catalog and to provide insight into the potential and efficacy of KMP for cross-platform development of Android and Web applications.

## **3 Related work**

The related literature for this thesis comes from the two main areas of focus in this work. The development of a requirements catalog for modern cross-platform frameworks and the evaluation of Kotlin-Multiplatform against this catalog.

### **3.1 Related literature for the development of a requirements catalog for modern cross-platform frameworks**

- Articles and papers that propose systems, methods or metrics to measure the performance and efficacy of modern cross-platform frameworks.
- Literature that defines different categories for the evaluation of modern cross-platform frameworks.
- Comparative studies of modern cross-platform frameworks.

Paper & Year	Description
[RM19]	Presents 33 criteria grouped in infrastructure, development, app, and user perspective and demonstrates the applicability of these criteria on 5 cross-platform frameworks.
[BH+20]	Investigates the performance overhead imposed by cross-platform mobile development frameworks in Android apps compared to the native development approach
[Naw+21]	Analyzes the two most popular approaches, native and cross-platform, and compares identical applications developed using native and Flutter, Xamarin and React Native.
[BH+19]	Reports on the industry’s perspectives and opinions on cross-platform mobile development, with an emphasis on the popularity, adoption, and arising issues related to the use of technical development frameworks and tools.

**Table 1:** Summary of related literature for the development of a requirements catalog for modern cross-platform frameworks.

### 3.2 Related literature for evaluating Kotlin-Multiplatform against the requirements catalog

- Literature that specifically focuses on KMP and it’s strengths and weaknesses.
- Comparative studies where KMP was compared against other cross-platform frameworks.

There is little to no literature on the second part. This is mainly because of the novelty of KMP and it’s uniqueness. KMP differentiates itself from other cross-platform frameworks by enabling developers to ”share what they want”. KMP allows developers to write native code and share as much or as little non-UI related code as they see fit. This complicates the direct comparison with other frameworks since the extent of shared code can vary between different KMP projects based on the developers decisions.

## 4 Approach, Outcome and Evaluation

### 4.1 Approach

- Systematic Literature Review
- Collect data on existing requirements catalogs and evaluation metrics for cross-platform frameworks
- Interviews and Questionnaire to validate, expand and improve the requirements catalog
- Test KMP for Android and Web against the catalog by utilizing the time tracking application built during the internship and developing small sample applications.
  - Certain categories of the catalog like performance, battery usage, etc. can be evaluated for KMP by drawing information from existing studies.
- Compare KMP and it’s approach to other popular cross-platform frameworks like Flutter and React Native

## **4.2 Outcome**

The outcome of the thesis will be:

- A well-developed requirements catalog for modern cross-platform frameworks.
- An evaluation of KMP for Android and Web development against the catalog.
- A comparison of KMP with other cross-platform frameworks.

## **4.3 Evaluation**

Maybe a Likert Scale based evaluation of how well KMP does in the different categories of the catalog.

## 5 Timetable

KW22	- Literature review and topic exploration. Collect and review material for the requirements catalog.
KW23	- Continue literature review. Organize findings into preliminary categories for the catalog. Search for experts to conduct the interviews with.
KW24	- Prepare interviews and questionnaire.
KW25	- Conduct interviews. Send out the questionnaire. Analyze data from the interviews.
KW26	- Review the time tracking application. Does it have the necessary features to test it against the catalog? Add features / write small sample applications for the categories of the catalog, that are not present in the app.
KW27	- Continue testing KMP against the catalog. Analyze the response to the questionnaire. Finalize the catalogue.
KW28	- Complete the testing and analyze the results.
KW29	- Evaluate the performance of KMP against the requirements catalog.
KW30	- Comparison of KMP with other popular cross-platform frameworks like Flutter and React Native
KW31	- Writing, writing, writing...
KW32	- Writing, writing, writing...
KW33	- Get feedback on the thesis from reference persons.
KW34	- Incorporate feedback and final touches. Printing.
KW35	- Submission

## References

- [BH+19] Biørn-Hansen **and others**: “An Empirical Study of Cross-Platform Mobile Development in Industry”. **in** *Wirel. Commun. Mob. Comput.*: (2019). URL: <https://www.semanticscholar.org/paper/An-Empirical-Study-of-Cross-Platform-Mobile-in-Bi%C3%B8rn-Hansen-Gr%C3%B8nli/ef327921173c8a5f2d04bdf4821ce8111ceb1466>.
- [BH+20] A. Biørn-Hansen **and others**: “An empirical investigation of performance overhead in cross-platform mobile development frameworks”. **in** *Empirical Software Engineering*: 25.4 (2020), **pages** 2997–3040. ISSN: 1382-3256. DOI: 10.1007/s10664-020-09827-6. URL: <https://link.springer.com/article/10.1007/s10664-020-09827-6#Sec5>.
- [Naw+21] P. Nawrocki **and others**: “A Comparison of Native and Cross-Platform Frameworks for Mobile Applications”. **in** *Computer*: 54.3 (2021), **pages** 18–27. ISSN: 0018-9162. DOI: 10.1109/MC.2020.2983893.
- [RM19] C. Rieger **and** T. A. Majchrzak: “Towards the definitive evaluation framework for cross-platform app development approaches”. **in** *Journal of Systems and Software*: 153 (2019), **pages** 175–199. ISSN: 01641212. DOI: 10.1016/j.jss.2019.04.001. URL: <https://www.sciencedirect.com/science/article/pii/S0164121219300743>.