FH JOANNEUM (University of Applied Sciences)

**Usage possibilities of WebRTC in a cross-platform developed hybrid app**

**Bachelor Thesis**

**submitted in conformity with the requirements  
for the degree of  
Bachelor of Science in Engineering (BSc)**

Bachelor’s degree program **Internettechnik**FH JOANNEUM (University of Applied Sciences), Kapfenberg

**supervisor:** Dipl. Ing. Johannes Feiner

**submitted by:** Michael Stifter  
**personal identifier:** 1310418054

06 / 2016

**Obligatory signed declaration:**

I hereby declare that the present Bachelor’s thesis was composed by myself and that the work contained herein is my own. I also confirm that I have only used the specified resources. All formulations and concepts taken verbatim or in substance from printed or unprinted material or from the Internet have been cited according to the rules of good scientific practice and indicated by footnotes or other exact references to the original source.

The present thesis has not been submitted to another university for the award of an academic degree in this form. This thesis has been submitted in printed and electronic form. I hereby confirm that the content of the digital version is the same as in the printed version.

I understand that the provision of incorrect information may have legal consequences.

Michael Stifter Graz, 13.06.2016

**Table of contents**

[Abstract 4](#_Toc451611011)

[Kurzfassung 5](#_Toc451611012)

[1 Introduction 6](#_Toc451611013)

[List of tables 7](#_Toc451611014)

[List of figures 8](#_Toc451611015)

[List of listings 9](#_Toc451611016)

[List of abbreviations 10](#_Toc451611017)

[Bibliography 11](#_Toc451611018)

Abstract

Kurzfassung

# Introduction

Over the last years, Web Real Time Communication (WebRTC) has seen a significant rise in popularity, especially in browser-based web applications. Its biggest disadvantage to date is the fact that not all web browsers support WebRTC, although the number of supporting browsers has been continuously rising for a few years now.

This poses a problem for developers who want to use WebRTC in applications today. While nowadays there are few alternatives to web applications in terms of desktop devices, the situation is different for mobile devices. Native apps have become massively popular and deliver substantial advantages when it comes to user experience. This stems from the fact that it is possible to integrate and access many components of the user’s device, such as the list of contacts, the calendar and various sensors into an application with ease. While it is possible to develop a native app that uses WebRTC, it also increases the development effort considerably, since it is necessary to implement the same functionality on multiple platforms, such as Android, iOS and Windows Phone.

A solution to this problem could be the use of a suitable cross-platform development framework that facilitates the use of WebRTC. For a cross-platform developed mobile app, it is not necessary to develop the same application once for each platform it should support, but rather only once. The framework then generates a native app from the shared code base. However, since WebRTC is a technology that can be considered relatively new and is still under development, it is not guaranteed that cross-platform development frameworks fully support the latest version of WebRTC.

This thesis takes a deeper look into popular cross-platform mobile development frameworks and examines them on their ability to support current versions of WebRTC. To analyze this examination, a set of criteria is defined in order to find the most suitable frameworks to develop a cross-platform app that uses WebRTC.

The thesis is structured as follows: The first part describes various ways of implementing a mobile app and highlights the advantages and disadvantages of each method in detail. The second part discusses the history and functionality of WebRTC, together with its benefits and shortcomings. In a third step, the possibilities of using WebRTC on mobile devices are addressed. Following that, the essential insights regarding the implementation of a reference app are pointed out. Chapter 5 discusses the evaluation process and its results. The final section concludes the thesis by summarizing the essential findings and suggesting possibilities to expand the underlying work.

# Cross-platform mobile development

Introduction, motivation

PAPER-1

PAPER-5 [In this context, the challenge for web developers is to de-

velop di\_erent versions of their applications that are cus-

tomized to suit the speci\_c characteristics of the di\_erent

platforms, yet provide a consistent set of features and ser-

vices across all versions.]

* API

-🡪 Model Driven Development? (PAPER-6)

## Differences to native app development

PAPER-2

Difference native/compiled versus web/interpreted code

Sensor/device access

User experience (push notifications, access to phonebook, contacts)

Ability to use app offline (HTML5 application cache – PAPER-3)

## Motivation

Pro / con

PAPER-7

## Approaches

PAPER-1

PAPER-9

### Web apps

### Hybrid apps

### Interpreted apps

### Generated apps

## Cross-platform development frameworks

PAPER-4

Important criteria for choosing a framework

### Apache Cordova (PhoneGap)

### Xamarin

### Titanium

### Ionic

# WebRTC

# Prototype development

# Evaluation

## Setup

## Method

## Results

# Outlook

# Conclusion

List of tables

List of figures

List of listings

List of abbreviations

Bibliography