FH JOANNEUM
University of Applied Sciences

Direct Peer to Peer Communication on iOS Devices

When mediator based technology is not available

Bachelor's Thesis

submitted in conformity with the requirements for the degree of

Bachelor of Science in Engineering (BSc)

Bachelor's degree programme **Software Design and Cloud Computing**

FH JOANNEUM (University of Applied Sciences), Kapfenberg

Supervisor: DI Johannes Feiner

Submitted by: Matthias Bartholomäus

February 2025

TODO: Specify the title, subtitle, author, submission date, study, language, your name, and supervisor/advisor in the main *thesis.typ* file. Then compile with *typst compile thesis.typ*. Finally, remove all TODOs (todo marcos) within your typst ource code.

Preview printed 2025-03-26.

Abstract

Modern mobile devices can make use of a wide variety of communication technologies. Besides having different applicabilities and protocols, standards like Bluetooth, WiFi or 5G need to be wireless to seamlessly transfer data. In resent years global economic demand has impacted research and development to vastly improve data transmission and hardware on smartphones. As of 2024 this led to over 4 billion smartphones users worldwide (Department, 2024). Unfortunately most of the communication methods used on smartphones all rely on mediators. Be it a router in a local network or a cell tower in a cellular network, without these nodes a connection between two peers can not be established, no matter how close neighboring devices may be. However in scenarios where the required infrastructure is not available or data needs to be handled sensitive, communication between two mobile devices can be established via Bluetooth or peer-to-peer WiFi since these do not required pre-existing infrastructure and purely rely on local radio broadcast, whereas the latter is recommended to use. Due to the latest advancements in these technologies and smartphone hardware, it is unclear how good direct Peer to Peer (P2P) networks work under certain conditions. This thesis tries to find and measure metrics that indicate the quality of direct P2P connections on Apple's mobile devices. This can help to evaluate the feasibility of projects that are planning to build on these technologies. In particular the AWDL will be tested for its robustness considering metrics of quality in different scenarios and under different conditions measured from the application layer.

Keywords: FHJ, SWD, iOS, peer-to-peer, ad-hoc, smartphone, Apple

Kurzfassung

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedis, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus.

Contents

List of Figures

List of Tables

List of Listings

1 | Introduction

Nowadays communicating via smartphones or tablets invariably relies on infrastructure. In most homes in developed countries one can find a local network advertised by WiFi technology and handled by a router. Being outside and not connected to a local network, mobile devices communicate with cellular towers that give access to a big underlying network managed by an ISP. Wherever we go, our mobile devices are connected, but relying on infrastructure that is not necessarily available all the time. Different reasons can lead to restricted or broken infrastructure which, in absence of ad-hoc technologies would break all communication. In countries with governmental protests cellular networks might be restricted or monitored. Another reason for disfunctioning infrastructure networks could be overwhelming usage, where a high number of people try to use the same cellular tower or environmental disasters which can take down power supply or network nodes itself. In all these cases users would highly profit from ad-hoc networks, that do not merely rely on any external dependencies.

In recent years support for local peer-to-peer technologies, that allow devices to directly communicate via local radio link broadcasts has improved in modern smartphones. This could improve connectivity in the aforementioned cases or enable scenarios in which numerous end devices connect together, to form a wide spread mesh network. Nowadays many different technologies exist to gain access to wide area ad-hoc connections like LTE-Direct or LoraWan. However, it is important to note that these technologies are not yet supported in smartphones and there is no guarantee that they will ever be. Bluetooth, first introduced in 1998 has been used a lot in mobile computing the last decades. Since data transfer speed overall is continually increasing and Bluetooth was initially developed for lower data rates a new approach started to enter the market. These days WiFi is predominantly used to facilitate local peer-to-peer connectivity in smartphones, whereby unfortunately every mobile operating system has created its own version which makes communication between smartphones from different vendors complex. Furthermore, the extent to which these technologies can provide stable data transfer and the quality with which they can do so, remains unclear and poorly documented. This information is essential for a further evaluation of the feasibility of applications based on direct peer-to-peer connections. Therefore this thesis aims to address the existing uncertainty by measuring metrics that can assist this evaluation of analogous applications.

> *Most smartphone communication leverages one hop radio links to cell towers or WiFi access points. In recent years, however, the major smartphone operating systems have included increasingly stable and useful support for local peer-to-peer communication that allows a device to talk directly to a nearby device (using local radio broadcast) while avoiding cellular and WiFi infrastructure. The ability to create these local links, combined with the ubiquity of smartphones, enables scenarios in which large groups of nearby smartphone users run applications that create peer-to-peer meshes supporting infrastructure-free networking. There are many possible motivations for these smartphone peer-to-peer networks. For example, they can support communication in settings where network infrastructure is censored (e.g., government protests),*

> *overwhelmed (e.g., a large festival or march), or unavailable (e.g., after a disaster or at a remote event). In addition, in developing countries, cellular data minutes are often bought in blocks and carefully conserved—increasing interest in networking operations that do not require cellular infrastructure.*
>
> *(Newport, 2017)*

As of 2024 4 billion people worldwide are using smartphones (Department, 2024), 27% of them iOS. iOS is the operating system used on iPhones and developed by Apple. This operating system also utilizes its own implementation of an ad-hoc WiFi protocol which leads to an incompatibility with devices not from the Apple ecosystem. However it is strongly used by Apples Continuity which bundles applications like AirDrop or AirPlay, Apples universal clipboard, handoff or password sharing for known contacts that want to join a WiFi network, just to name a few.

Apple Wireless Direct Link (AWDL) is Apple's implementation for direct one hop connections and is already used by Apple in Continuity. Until previously this has not been Apple's only option. Until iOS 11 the Multipeer Connectivity Framework also used Bluetooth on the physical layer, which was then removed and replaced by AWDL. Bluetooth can still be accessed by application developers, via the Bluetooth Core Framework, which is more versatile and compatible also for non Apple devices. Although Bluetooth has improved range and speed in recent years, also for mobile end devices, AWDL is the recommended technology to establish direct links between iOS devices. Therefore the Networking Framework, included in the iOS SDK is used to establish and test the quality of connections between iOS devices, which exposes a transport layer interface to use and control protocols like UDP, TCP or QUIC.

This thesis analyses and measures the capabilities of the latest iOS devices to communicate via direct P2P networks. Since most of current technology standards rely heavily on external infrastructure with powerful receiver antennas, direct P2P communication is not yet wide spread third party applications or optimized to function over long distances. However it is unclear, where the boundaries of the current implementations lie and which type of applications could be implemented with direct P2P. The results aim to help assess feasibility of different applications that build upon AWDL. Examples vary from secure communication applications to city wide mesh networks or improved reachability in dangerous situations where no cellular network is in reach.

1.1 Research Definition

This research should help assess feasibility of applications that want to utilise AWDL. It will compare different transport layer protocols in different scenarios. This scenarios will vary in distance, surroundings and transmission data sizes. The Networking Framework is used to quantify the connection through metrics like RTT, jitter, package loss and more and is subsequently referred to as connection quality.

Research Questions

Which aspects influence the AWDL connection quality and how far can it reach?

Hypothesis

H_1

AWDL connection quality on Apple devices depends on the surroundings and functions worse in crowded surroundings.

H_2

AWDL connection quality on Apple devices depends on the transport layer protocol.

Method

A prototype application will be developed for the iOS platform that will serve as a tool to measure the connection quality. The metrics will be precisely defined in the test protocol in Section 5. In addition, different testing surroundings, indicating various characteristics of real life scenarios will be tested to cover most areas of application. Measurement of connection quality will be purely based on values captured by the prototype app itself. The characteristics of the environment will be described based on human perceive and measured with suitable methods.

1.2 Summary

Smartphones mostly rely on infrasructure based networks. As this is a strong dependency that could vastly limit connectivity and advancements in direct connection software and hardware have emerged in recent years it is unclear which connection qualities these methods can produce. Therefore a prototype application developed for the iOS platform serves as a utility for measuring these metrics, with heavy use of the low level Networking Framework API.

1.3 Notice on terminology

peer-to-peer, manet, wifi direct, awdl, ad-hoc,

iOS machines… iOS can be used on macOS or iPadOS and so on…

2 | Related Work

The following is a non-comprehensive discussion of previous research in peer-to-peer technologies in the mobile context. After covering historic considerations of MANET research and D2D communication in cellular networks via standards like LTE-Direct or 5G New radio (NR) Sidelink (SL), a deeper introduction to Apples ecosystem and AWDL is given, where a lot is based on the OWL project from the TU Darmstadt which reverse engineered the AWDL protocol and found several security concerns. Open it up again a bit for not only ios and then at the end going again deeper in the ios section with TU darmstadt

2.1 History of MANET

Already back in 2001 the Proem project (Kortuem, Schneider, Preuitt, *et al.*, 2002) examined different aspects of peer-to-peer applications for mobile ad hoc networks (MANET) to enable proximity-based collaboration. In particular, Proem was an approach to provide high-level support for mobile peer-to-peer application developers and was tested by students of the University of Oregon, whilst creating an MP3 file-sharing system. They already noticed the trend for an ever-larger becoming applicability of personal mobile devices for data sharing but listed resources of mobile devices among other possible limitations. This facet has vastly changed since then and several new ideas like ShAir (Dubois, Bando, Watanabe, *et al.*, 2013), a middleware infrasructure for peer-to-peer sharing between mobile devices or mFerio (Balan, Ramasubbu, Prakobphol, *et al.*, 2009), a peer-to-peer mobile payment system have emerged.

Balan, Ramasubbu, Prakobphol, *et al.* working on mFerio already noticed the problem that mobile devices rely too heavily on static infrastructure. Back then cell phones have already become popular tools that combined calendars, address books, messaging or cameras. The increasing need to use them as a payment vehicle has become ever larger and the authors questioned the state of the art implementations back then. In particular mobile payment solution required constantly stable connections via either SMS or GSM/CDMA based technologies which were connected to a backend payment server. They noticed that these implementation, which were to heavily relying on external systems could not replace cash based systems and aimed to develop a decentraliced approach based on NFC. The goal of their larger term project aimed to create a digital wallet for cellphones which would allow users to store everything on the device which was in their physical wallets, like credit cards, identification or tickets. The applicabilities of this project strongly remind of the Apple Wallet, which was introduced in iOS 6 in 2012 and also leverages NFC.

Some years later in 2013 ShAir was developed as a structured software engineering project written in Java that used WiFi technology on Android devices to share data between them. While Wifi-direct and Bluetooth were also accessible to the developers, they decided to use a combination of WiFi AP mode and WiFi Client mode in a random fashion to create dynamic networks and discover nearby peers because devices would not allow the former without active user interaction. They tested the app by sharing pictures among 12 devices from several vendors using no fixed existing infrastructure. This project also strongly reminds on apple propreitary software, namely AirDrop which has been released by Apple in 2011.

Since then support for direct peer-to-peer connections has matured on various mobile operating systems, including iOS and its Multipeer Connectivity Framework which allows nodes to advertise itself, discover nearby advertisers and attempt to connect to detected nearby advertiser. The concept of that model of motivated Newport to develop a formal definition and comparison of gossip algorithms. He describes and analyses differences in algorithm parameters and how they influence data spreading in a MANET where the goal is that messages spread to entire network (2017). The author claims that these algorithms can help to establish peer-to-peer meshes that support infrastructure-free networking and communication and takes the discontinued FireChat application as an example which offered group chats using smartphone peer-to-peer services such as Bluetooth, WiFi and the Multipeer Connectivity Framework. This application has been adopted in multiple governmental protest or festivals that were located out of reach of cell towers.

2.2 D2D in cellular networks

Although a lot of research exists on D2D communication in cellular networks, most of it is done in a military use case (Gamboa, Ben Mosbah, Garey, *et al.*, 2023), like unmanned aerial vehicles (UVA) or public safety networks (Gamboa, Henderson, Garey, *et al.*, 2024), like vehicle to everything (V2X). Most of this research builds upon 5G New Radio (NR) Sidelink (SL) which has implemented protocol support for vehicle to everything and Proximity Services (ProSe) for public safety networks which allows user equipment to directly talk to each other without the interferance of a base station (gNB). Although approaches existed to also introduce D2D communication to the commercial markets back in 2014 by Qualcomm and Condoluci, Militano, Orsino, *et al.* back in 2015 proved that LTE-Direct, which preceded 5G SL, has some energy and scaling benefits over WiFi-Direct, no support for this technology is given on mobile smartphones Eskimo answer. Like also pointet out by the authors of this critical review of mobile device-to-device communication (Desauw, Luxey-Bitri, Raes, *et al.*, 2023).

2.3 Apple Ecosystem and TU Darmstadt

As already mentioned, the OWL project by Secure Mobile Networking Lab (SEEMOO) by TU darmstadt contributed several papers to research on Apple's wireless ecosystem (Stute, Kreitschmann and Hollick, 2018a). Their goal was to assess security and privacy as well asw enable cross-platform compatibility with other vendors. They started to investigate AWDL in an approach to reverse engineere the 802.11 Wifi based protocol and developed an open source application that implements Apples AWDL. The authors used binary and runtime analyses to reconstruct the daemons and frameworks involved in communicating via AWDL and found that each AWDL node announces a sequence of Availability Windows indicationg that it is ready to communicate with other AWDL nodes. In the process they als found several security issues which allowed an attacker to access the macOS WiFi dirver interface which could lead to DoS attacks (Stute, Kreitschmann and Hollick, 2018b).

In 2019 and 2021 after finding the first vulnerabilities when reverse engineering the AWDL protocol they dedicated a seperate paper on researching security and privacy issues in this protocol. The study uncovers multiple vulnerabilities related to both design flaws and implementation bugs. One of the major findings is the possibility of a man-in-the-middle (MitM) attack, which allows an attacker to stealthily modify files transferred via AirDrop. Additionally, the study identifies denial-of-service (DoS) vulnerabilities that can disrupt communication or force the sudden crash of all nearby devices. The research also reveals privacy weaknesses that allow attackers to track users over extended periods, effectively bypassing MAC address randomization. The vulnerabilities were discovered through a combination of reverse engineering, code analysis, and patent examination. To demonstrate the feasibility of these attacks, the researchers developed proof-of-concept implementations using inexpensive hardware, including a 20 dollar micro:bit device. Following responsible disclosure, Apple addressed one of the DoS attack vulnerabilities. However, the researchers highlight that several of the identified security and privacy risks require fundamental redesigns of some of Apple's services to be fully mitigated. The study ultimately highlights critical security flaws in AWDL, demonstrating the potential impact on over a billion Apple devices and emphasizing the need for stronger security measures and protocol improvements Onebillionopeninterfaces Disrupting continuity.

In 2020 Ian Beer a british computer securtiy expert and white hat hacker, inspired by the work of TU Darmstadt found another severe security issue in AWDL which could remotely trigger an unauthenticated kernel memory corruption that lead to all iOS devices in radio proximity to reboot. He also describes how this issue can lead to a vulnerability that lets an attacker run any code on nearby iOS devices and steal all user information. atIanBeer_2020

Focusing on the Apple ecosystem the Open Wireless Link (OWL) project form TU Darmstadt has made some important contributions to this research field. While reverse engineering the AWDL protocol and investigating Apples wireless ecosystem they found several security concerns and developed useful applications they published as open source software on GitHub.

In 2021 the OWL project found that during the AirDrop authentication handshake with nearby devices, vulnerable hash values of the user's own phone number and email adresses are exchanged. The authors describe two theoretical attacks to exploit these vulnerabilities and propose an application named PrivateDrop to

The study reveals a privacy vulnerability in Apple's AirDrop file-sharing service, which leaks phone numbers and email addresses by exchanging hash values of a user's contact information during the authentication handshake with nearby devices. These hashes are vulnerable to brute-force attacks, allowing attackers to retrieve the original contact details. In a paper presented at USENIX Security'21, researchers describe two attack methods to exploit these vulnerabilities and introduce "PrivateDrop", a privacy-focused alternative to AirDrop that uses private set intersection to enhance security. To demonstrate the severity of the issue, the researchers developed a proof-of-concept attack, showing that an attacker within Wi-Fi range and physical proximity can efficiently extract sensitive user information. The study highlights serious privacy and security implications, including the risk of spear phishing attacks and the ability to map contact identifiers to specific locations using multiple strategically placed "collector" devices. For their proof-of-concept, the researchers created a custom rainbow table that allows them to reverse SHA-256 hashes of phone numbers within milliseconds. They explore the trade-off between success rate and storage requirements for such attacks. After responsibly disclosing their findings to Apple, the team published their proof-of-concept tool, "Air-Collect," on GitHub, demonstrating the feasibility of large-scale contact data collection. The findings underscore critical privacy risks in AirDrop, emphasizing the need for stronger security measures to prevent unauthorized access to user information.

DEMO: AirCollect: Efficiently Recovering Hashed Phone Numbers Leaked via Apple AirDrop. 2021

Disrupting Continuity of Apple's Wireless Ecosystem Security: New Tracking, DoS, and MitM Attacks on iOS and macOS Through Bluetooth Low Energy, AWDL, and Wi-Fi. 2021

PrivateDrop: Practical Privacy-Preserving Authentication for Apple AirDrop. 2021

Welcome to the Open Wireless Link (OWL) project. We are researchers from the Secure Mobile Networking Lab at TU Darmstadt looking into Apple's wireless ecosystem. Our goal is to assess security and privacy as well as enable cross-platform compatibility for next-generation wireless applications. We started by investigating the Apple Wireless Direct Link (AWDL) protocol and will go beyond. You can read our publications and use our open source code projects. If you have questions or would like to collaborate, feel free to contact us.

TODO: LTE Direct mystery and 5G NR sidelink mystery, why do mobile phones not have access to these technologies? at least the app developers do not have access to that most of this technology is used in vehicles and autonomous driving, C-V2X, seems like these direct communication technologies are not available on modern smartphones and mostly used in public safety or automotive applications, look for sources and write a paragraph about this and its use cases

TODO: write a paragraph about newer approaches about wifi direct or bluetooth on smartphones and mobile devices

Open wireless link TU darmstadt

2.4 Summary

3 | Background

3.1 Infrastructure Networks

3.2 Ad-hoc Networks

This should better be part of introduction Describe that all of mobile connections use mediators and why this happend like it and why it is beneficial (bigger antennas that can handle weaker signals and send stronger signals) describe what has changes in the last years on apples platform and why non-mediator communication got so important for apples ecosystem, apple watch pairing, clone app to mac (when opened on iphone, eg. calendar), cmd+c and cmd+v via across iphone and mac what is lorawan and how could it be used one day in smartphones and mobile computing describe why peertopeer wlan or bluetooth is not yet suitable for long distance communications, also mention the longest wlan connection, 273km with bigger antennas

describe what was used on ios device what frameworks exist. how bluetooth was used and how wifi is now used for ptp connections

3.3 Satellite phones

3.4 Starlink Direct to Cell

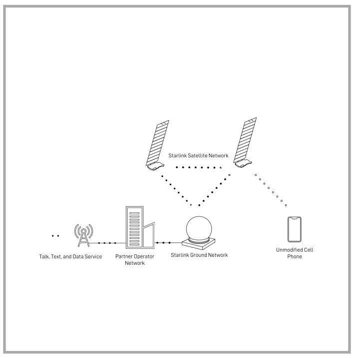

Figure 1: Abstract structure of Starlink’s network.

3.5 OSI Transport layer

3.5.1 TCP

3.5.1.1 Nagle's algorithm

3.5.2 UDP

3.5.3 QUIC

3.6 Ad-hoc technologies

While and solve the issue of dead spots, they also rely on infrastructure.

3.6.1 WiFi

3.6.2 Bluetooth

3.6.3 LoraWan

3.6.4 LTE-Direct

3.6.4.1 gNB

3.6.5 NFC

3.7 Ad-hoc on iPhones

3.7.1 Bonjour

3.7.2 Multipeer Connectivity

3.7.3 Networking Framework

3.7.4 AWDL

biggest kernel module in iOS

3.7.5 Websockets

In the background section you might give explanations which are necessary to read the remainder of the thesis. For example define and/or explain the terms used. Optionally, you might provide a glossary (index of terms used with/without explanations).

Hints for equations in Typst:

Mathematical formulas are (embedded in $) in Typst. For example:

The notation used for **calculating** of *code performance* might typically look like shown in Equation 1, i.e. the first one for **slow** in Eq. I and the other one for **very slow** in Eq. II.

$$O(n) = n^2 \tag{I}$$

$$O(n) = 2^n \tag{II}$$

Equation 1: Equations calculate the performance.

In the text we refer multiple times to ϕ. We define it to be calcultated as shown here:

$$\begin{aligned} d &= 24 - 10 - 7 - \sqrt{3} \\ d &= 14 - 7 - \sqrt{3} \\ d &= 7 - \sqrt{3} \end{aligned} \tag{III}$$

$$\phi := \frac{d}{3} \tag{IV}$$

Equation 2: A custom definition of ϕ allows to shorten upcoming equations.

The Equation 2 explains (for the single steps see Eq. III and Eq. IV) how the overall ϕ is calculated to be used in the upcoming formulas of this thesis.

4 | Concept

The following describes an overall concept on how the evaluation of measuring quality metrics of peer to peer connection between iOS devices is done. It will cover an abstract description of the application design and the used software packages, as well as how testing will work.

4.1 Overall Idea

The overall idea to evaluate the quality of direct connection between mobile iOS devices is to build a prototype application that measures data transfer on the OSI application layer. The application is written in `Swift` using the integrated development environment (IDE) `XCode`, which is the suggested way to build iOS application by Apple. The built artifact is distributed via TestFlight, an online service for installing and testing apps for Apple devices and can be downloaded via a link or directly installed via cable from the developer machine. The application is then simultaniously run on two iOS machines that can then connect via AWDL and test the connection. Measurements taken during the data transfer will then be displayed and documented for later evaluation.

4.2 Layers

Although the application code can not be grouped into the following categories, for understand the testing structure and how the data is collected the application is best presented from an abstract perspective described in layers. For this model the application can be divided into an `User Interface`, `Logic`, `Measuring` and `Networking` layer.

4.2.1 User Interface

The `User Interface` layer is developed using `SwiftUI` and serves the presentation of different interaction points, the tester can interact with to control the underlying workings.

4.2.2 Logic

The next underlying `Logic` layer serves as a gateway between the user input and the underlying `Measuring` and `Networking` logic. The composition of the first two layers follow the MVVM frontend software pattern, whereas the `Logic` layer corresponds to the view model (VM) part.

4.2.3 Measuring

The measuring layer is highly intervened with the networking layer and therefore the most abstract of these listed. It sits upon the receiving and sending functions of the networking layer and is built to minimize computational delays.

4.2.4 Networking

The networking layer is built to abstract the communication with the wifi module via the networking framework which is included in the iOS SDK. The measuremnt logic is tightly integrated but the sole responsibility of the networking layer is to parse raw data to communicate with the Networkign API and read header lengths to distinguish between different package types. These different package types will be more precisely covered in implementation.

4.3 Testing Concept

4.4 Visualization

describe the overall concept of what you have planned

what is the purpose of the app, how are you planning to measure the metrics and implement these measurements in the app

maybe describe which metrics you want to test, but this goes a bit too far into implementation

maybe describe in general your approach to a test protocol and that you want to test it in different scenarios and why you think that these different scenarios are important and how you plan to visiualize the results,

Describe an overall concept of a solution, which could possibly solve a given problem. Design a novel solution and visualise the architecture and relevant (data) flows. Compare and relate your approach to possible alternatives and argue why and in which way(s) the suggested solution(s) will be better.

Hints for formatting in Typst:

1. You can use built-in styles:
   1. with underscore (_) to *emphasise* text
   2. forward dash (`) for `monospaced` text
   3. asterisk (*) for **strong** (bold) text

You can create and use your own (custom) formatting macros:

1. check out the custom style (see in file `lib.typ`):
   1. `#textit` for *italic* text
   2. `#textbf` for **bold face** text

5 | Implementation

use mach_absolute_time to measure the jitter?

```
let task0 = Task { @MainActor in
    for await nearbyPeers in service.nearbyPeersToInvite.values {
        state.nearbyPeersToInvite = nearbyPeers
    }
}

tasks.insert(task0)
```

Listing 1: Saving the reference to the detached task, so it can be canceled when the viewmodel is dereferenced.

https://developer.apple.com/documentation/swift/calling-functions-with-pointer-parameters how to pass pointers to functions

Describe what is relevant and special about your working prototype. State how single features help to solve problem(s) at hand. You might implement only the most relevant features. Features you select from your prioritised feature list assembled in Chapter 4. Focus novel, difficult, or innovative aspects of your prototype. Add visuals such as architectures, diagrams, flows, tables, screenshots to illustrate your work. Select interesting code snippets, e.g. of somewhat complicated algorithms, to present them as source code listings.

For example, you might explain your overall system, then the details of the backend and frontend development in subsections as shown here:

5.1 Overall System

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si.

Hints for images in Typst

Use vector graphic formats such as Scalable Vector Graphics (SVG) for drawings and png for screenshots. Using jpeg is only ok, if you need to show photographic images, such as a picture of a sunset.

For example, the following shows how an SVG image is references using the `image` Typst macro. The image is furthermore embedded in a `figure` macro. The `flex-caption` allows to include a full sentence as caption below the image and a short caption for the list of figures. Also note the use of a label `<fig:companylogo>` which is later referenced with `@fig:companylogo`:

Figure 2: The logo of the FH JOANNEUM, the University of Applied Sciences.

The application uses the logo of the company, see Figure 2, in the navigation bar to provide *home* functionality.

5.2 Testing Protocol

What do I want to achieve? Test one time with bluetooth off and one time with bluetooth on since iPhones use the same antenna for both, which has already caused slow wifi connections when peer to peer was enabled… Bluetooth doesnt turn off completely I want to test different scenarios, so I know which transport protocol performs best under certain circumstances and environments. I want to test different payload sizes, package count from perspective of !application level!. The metrics should be captured depending on distance and other environmental factors like obstacles.

capture: distance obstacles outside temperature

different scenarious: underground forrest city field

do these different scenarios with: 100/1000/10_000/100_000 packages 128/4096/16_384 byte per package 1m/10m/30m/maxm

do every case 5 times, max meters are not done in every scenario, just how far it can go…

also include seperately a comparison between tcp options, noDelay and noPush

done -> inPr -> notStarted code -> test -> write thesis

5.3 Backend

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si.

Hints for code listings in Typst:

The way to include source code in your document is discussed and shown in https://typst.app/docs/reference/text/raw/. For this template we provide a custom macro/function *fhjcode* to support listings with code pulled in from external files and with line numbering. For example:

For example: We implemented a minimal *script* in Python to manage a secure `Messages` in object oriented ways. See Listing 2 and Listing 3 for a minimal `SecureMessage` class.

```python
class Message:
   def __init__(self,txt):
       self.m=txt
   def __str__(self):
       return f"{self.m}"
```

Listing 2: Defining a base class in Python. Here, the base class is named *Message*.

```
class SecureMessage(Message):
    pass
```

Listing 3: For inheritance we might define a specialised class based on another class. Here, the specialised class *SecureMessage* is based on the class *Message*.

For example: As shown in Listing 3 the secure version of the class is just a stub where further improvements and extensions have to be applied.

5.4 Frontend

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si.

Hints for abbreviations and glossary entries *gls(key)* in Typst:

Abbreviations should be written in full length on the first occurrence. Later the short version can be used. Therefore, define glossary entries with a *key* and a *short* as well as a *long* description first (see file *glossary.typ*). In the text you might use `#gls(<key>)` (and `#glspl(<key>)` for plural) usage of an abbreviation. For example:

The system is using Copy on Write (COW) for optimisation. The implementation of COW can be changed by … Note the usage of the special configured Garbage Collection (GC). We compared many GCs to find out ..

6 | Evaluation

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri.

Describe (proof) how your implementation really solved the stated problem. I.e. accept or reject your hypotheses. Provide a range of input data sets. Run experiments and gather the output (of tools) to meter your prototype. For the analysis, collect the measurement-data, process (e.g. filter) data and interpret the data. Include an interpretation of the work. What do the results mean to you? State current limitations of your solution. Give (personal) interpretation where suitable. Your own opinion is relevant, but must be marked clearly as such.

6.1 Setup Experiment

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si.

For example: During the setup the GC was configured for the parallel version using the value `+UseParallelGC` for the command line argument `-XX` (`java -XX:+UseParallelGC`).

Hints on dynamically reading in external data for tables in Typst:

Using the custom macro `fhjtable` it is possible to include data dynamically for table generation. The data has to be specified in Comma-separated Values (CSV) as shown below:

Name	Profession	Experience (in years)
Max	Student	3
Mia	UX-Designer	7
Helga	Programmer	9

Table 1: Professional experience of the test users with databases.

Find in Table 1 the years a user has worked with different relational or nosql databases in a professional context.

6.2 Measurement

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si.

Hints on using tables in Typst:

Somewhere in the normal text of the thesis the interpretation of data and information shown in a table must be discussed. Explain to the readers which numbers are important. Possibly, highlight unexpected or special data points.

	Min	**Max**	$\varnothing$	$\boldsymbol{\sigma}$
Network roundtrip time	34.6s	42.5s	38.1s	2.3s
Time for single request	2.4s	13.5s	**7.1s**	4.3s

Table 2: The numbers in the table above show the minimum, maximum, average $\varnothing$, and standard deviation σ of the 273 measured network times in seconds.

For example: … Table 2 shows some calculated results on the roundtrip and request times measured in the experiment. The average, the minimum, the maximum and the standard deviations hint to a dramatic increase (> 13%) in performance in comparison to the old solution of 2003.

6.3 Interpretation of the Data

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si.

For example: The customisation of the GC seem to have following positive and negative consequences....

Hints on dynamic calculation in Typst:

We might calculate, e.g. `#calc.max(...)`, within our document, such as max of three and seven times two is: 14.

Hints on using logic in Typst:

For example, we might use **for loop** to arrange a few images in a grid box, as shown below.

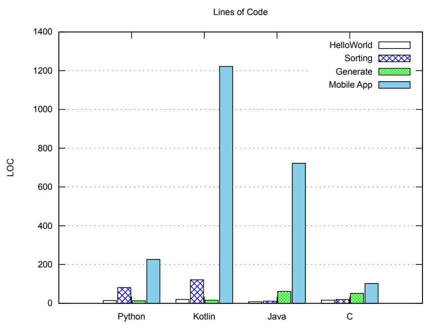

Figure 3: Compared source code by metric 1.

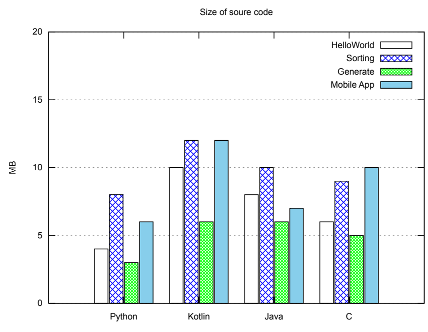

Figure 4: Compared source code by metric 2.

Hints on Charts:

Note: the charts (**vector**! images) shown have been created from raw data using the tool **gnuplot** on the command line. With `gnuplot` you can create charts by use of a textual command language. This is great for automation and it is also great for managing the source code in `git`.

7 | Conclusion and Outlook

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit.

Sum up the results achieved and give an outlook by suggesting further research by explaining how others could built on your results.

Glossary

COW **– Copy on Write** 15

CSV **– Comma-separated Values** 16

GC **– Garbage Collection** 15, 16, 18

P2P **– Peer to Peer** i, 2

SVG **– Scalable Vector Graphics** 13

Bibliography

Balan, R. K., Ramasubbu, N., Prakobphol, K., Christin, N., et al. (2009) mFerio: the design and evaluation of a peer-to-peer mobile payment system. In: *Proceedings of the 7th international conference on Mobile systems, applications, and services.* [Online]. June 2009 Kraków Poland, ACM. pp. 291–304. DOI:10.1145/1555816.1555846.

Condoluci, M., Militano, L., Orsino, A., Alonso-Zarate, J., et al. (2015) LTE-direct vs. WiFi-direct for machine-type communications over LTE-A systems. In: *2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC).* [Online]. August 2015 Hong Kong, China, IEEE. pp. 2298–2302. DOI:10.1109/PIMRC.2015.7343681.

Department, S. R. (2024) *Number of smartphone users worldwide from 2014 to 2029 (in millions).* [Online]. December 2024. https://www.statista.com/forecasts/1143723/smartphone-users-in-the-world.

Desauw, L., Luxey-Bitri, A., Raes, R., Rouvoy, R., et al. (2023) *A critical review of mobile device-to-device communication.* [Online]. https://inria.hal.science/hal-04198528.

Dubois, D. J., Bando, Y., Watanabe, K. & Holtzman, H. (2013) ShAir: extensible middleware for mobile peer-to-peer resource sharing. In: *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering.* [Online]. August 2013 Saint Petersburg Russia, ACM. pp. 687–690. DOI:10.1145/2491411.2494573.

Gamboa, S., Henderson, T. R., Garey, W., Liu, C., et al. (2024) Towards System Level Simulations of Public Safety Applications over 5G NR Sidelink. In: *2024 IEEE World Forum on Public Safety Technology (WFPST).* [Online]. May 2024 Herndon, VA, USA, IEEE. pp. 1–6. DOI:10.1109/WFPST58552.2024.00043.

Gamboa, S., Ben Mosbah, A., Garey, W., Liu, C., et al. (2023) System-Level Evaluation of 5G NR UE-Based Relays. In: *MILCOM 2023 - 2023 IEEE Military Communications Conference (MILCOM).* [Online]. October 2023 Boston, MA, USA, IEEE. pp. 807–814. DOI:10.1109/MILCOM58377.2023.10356291.

Kortuem, G., Schneider, J., Preuitt, D., Thompson, T., et al. (2002) When peer-to-peer comes face-to-face: collaborative peer-to-peer computing in mobile ad-hoc networks. In: *Proceedings First International Conference on Peer-to-Peer Computing.* [Online]. 2002 Linkoping, Sweden, IEEE Comput. Soc. pp. 75–91. DOI:10.1109/P2P.2001.990429.

Newport, C. (2017) Gossip in a Smartphone Peer-to-Peer Network. In: *Proceedings of the ACM Symposium on Principles of Distributed Computing.* [Online]. July 2017 Washington DC USA, ACM. pp. 43–52. DOI:10.1145/3087801.3087813.

Stute, M., Kreitschmann, D. & Hollick, M. (2018b) One Billion Apples' Secret Sauce: Recipe for the Apple Wireless Direct Link Ad hoc Protocol. In: *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. [Online]. October 2018 New Delhi India, ACM. pp. 529–543. DOI:10.1145/3241539.3241566.

Stute, M., Kreitschmann, D. & Hollick, M. (2018a) *The Open Wireless Link Project*. [Online]. 2018. https://owlink.org/.