

Tampere University Unit of Computing Sciences  
COMP.SE.610 Software Engineering Project 1  
COMP.SE.620 Software Engineering Project 2

ProjectM  
**MobileMESH**

Test Report

Roope Mantere  
Neea Tienhaara  
Jahid Sagor  
Katja Heiskanen  
Elsa Huovila  
Vertti Nuotio  
Diyaz Yakubov

## Version history

Ver- sion	Date	Author	Description
0.1	19.4.2023	Neea	First test log
0.2	20.4.2023	Katja	Information about tests
0.3	21.4.2023	Katja	Second test log
0.4	26.4.2023	Elsa	Accessibility testing
0.5	3.5.2023	Roope & Vertti	System testing
0.6.	5.5.2023	Diyaz & Katja	Acceptance testing

## Contents

1	Introduction .....	4
	1.1 Purpose and scope of document .....	4
	1.2 Product and environment .....	4
	1.3 Project constraints related to testing.....	4
2	Testing process.....	4
	2.1 General approach.....	4
	2.2 Testing roles.....	5
	2.3 Test schedule.....	5
	2.4 Test documentation.....	5
3	Testing Tools .....	6
4	Test cases and results .....	6
	4.1 Test results.....	6
	4.2 System testing.....	6
	4.3 Usability testing .....	10
	4.4 Accessibility testing .....	11
	4.5 Acceptance testing .....	11
	APPENDIX A [...Z] .....	12

# 1 Introduction

## 1.1 Purpose and scope of document

This test report is an overview of the testing process and results for MobileMESH app. The purpose of the test report is to make clear what have been tested, when, and by who. Testing included system and acceptance tests.

In addition to testing the functional requirements of the app via test cases in system testing, we tested the usability of the app. There were 5 basic functional requirements for the app, and testing the process of those revealed many bugs to be fixed.

Testing was started in the second to last week of development, in the 5<sup>th</sup> sprint, and lasted until completion of the project.

## 1.2 Product and environment

MobileMESH is an android application that facilitates self-organizing MESH network among scouts. The project is stored in a public GitHub repository.

## 1.3 Project constraints related to testing

The biggest constraints to testing the project were time and the MESH connection that the app requires. Testing the app fully was difficult due to the group being slightly behind schedule and the time limitations set by the course, as well as managing making the required documentation.

The mesh network requires that the app is always tested in a group, where two or more devices can be connected and are in proximity. This completely ruled out remote testing and required recruiting participants for physical tests, for which there weren't that many candidates. We by ourselves did not find the time to get actual end-users to test the app, who would regularly be in environments that have no existing network. Otherwise, since the app is mostly a general chat-based app, we decided that participants that we could find would be enough to evaluate the usability.

# 2 Testing process

The coding style of the project follows general Git practice of working on one issue in its own branch, testing and reviewing code in a pull request and merging it to the main branch.

## 2.1 General approach

The testing strategy mostly relies on system testing by downloading the app to devices and then connecting them to each other through the WLAN MESH. A usability test and assessment were also conducted at the end of the testing period.

Customer acceptance tests were done at the end of the course, on 5.5.

## 2.2 Testing roles

Project manager first created the testing plan, which included test cases and requirements.

System testing was conducted by several group members taking turns in test sessions. Testers include Katja Heiskanen, Neea Tienhaara, Elsa Huovila, Vertti Nuotio and Roope Mantere.

Usability testing and evaluation was done by the UI/UX team including Elsa Huovila, Neea Tienhaara and Katja Heiskanen.

Name	Tasks
Vertti Nuotio	System testing
Diyaz Yakubov	Test plan, test cases
Neea Tienhaara	System- and usability testing, documentation
Elsa Huovila	System-, usability- and accessibility testing
Katja Heiskanen	System- and usability testing, documentation
Roope Mantere	System testing
Customer	Acceptance testing

## 2.3 Test schedule

Testing the entirety of the app was started in the last week of development, which was a bit late due to problems with the functionality of the MESH network. Small scale testing of the separately implemented app components and features of course occurred before this.

Testing begun on 19.4. and continued as planned to the end of the testing sprint, 30.4. A lot of bugs in the logic of the app were discovered and fixed during this time, and improvements were tested in the following test rounds. Usability testing was started with separate accessibility tests and then done alongside system testing.

One testing session generally took around 2+ hours, including minor bug fixes. Larger bug fixes took time of developers working independently outside the testing sessions.

## 2.4 Test documentation

Results of system testing sessions were generally documented to this document using tables. Tables describe the scenario and give a short review whether the test succeeded and if not, what changes needed to be implemented.

### 3 Testing Tools

System testing used the Android Studio environment to run tests and debug processes at the same time. Tests were run on several phones using different versions of the Android system.

## 4 Test cases and results

### 4.1 Test results

System testing was the most extensive for bugfixes, and many improvements were made based on them. The logic of the network joining became clearer due to testing, but despite that the results still indicate the process of forming the network is still a bit unstable and can be slow in some cases. Connecting 2 devices is mostly functional, but there were issues within the network when connecting more than 2 devices. At the end of system testing, the app passes test cases 1, 2 and 4 most of the time.

The abandoned test cases were sharing location data in the app (3) and ensuring that it would work in an environment where there is no GSM connection available (5). The former did not get implemented due to many bugs revealed in the basic connectivity, which took a lot of time to try to fix. The latter was not tested due to the difficulty of getting into an environment without existing network and the fact that testing it would have quickly become very time consuming.

In usability testing we found out that the process of creating/joining the network proved somewhat difficult for the user, especially with the existing bugs of the app. There is a lack of feedback on the network status, that could be improved. On the other hand, the chat view and sending a message was very clear and intuitive for the user.

Accessibility testing revealed notable issues in contrast and scaling of the app components, and the most critical issues were fixed. The onboarding screens had the most issues, with a total of 30 detected across four screens. In some areas, the repeated pattern of issues was noticeable, which at least partially explains the high amount in some categories.

### 4.2 System testing

We tested the application in a few different rounds depending on the state of the development. System testing was done by team members to test and ensure the functionality of the app before moving on to usability tests.

We found 8 clear bugs in total, 4 of which are still open. Most of the bugs encountered are critical to the intended use of the app. It was difficult to track the root cause of some of the issues, so there might still be bugs that we didn't discover during testing.

The first table below is the general conclusion of the system tests, including bugs discovered and their severity. The following are the tables that we originally logged results of the tests in.

Severity ranking:

1 – Critical, prevents the app's intended use

2 – Moderately hinders the app's intended use

### 3 – Usability issue, doesn't affect functionality

Test case	Bugs	Severity and status
1. Verify that the app can establish a MESH network between two devices using WLAN or Bluetooth technology	<b>(1) Device doesn't send network request to all devices</b> (2) After forming a connection, the other device is not visible (3) Messages are stuck in a "Waiting" mode	<b>(1) 1, open</b> (2) 3, fixed (3) 1, fixed
2. Verify that the app can authenticate new nodes joining the network and maintain the security and reliability of the network	<b>(4) Messages not received by all devices</b> (5) The new device couldn't find the original device (6) Issue with connecting a new device to 2-device network <b>(7) The process of connecting to a network is confusing to user</b>	<b>(4) 1, open</b> (5) 1, fixed (6) 1, fixed <b>(7) 3, open</b>
4. Verify that the app can handle multiple devices joining and leaving the network without affecting the network's overall functionality	<b>(8) In some cases, crashes on the original or new device after creating the network</b>	<b>(8) 2, open</b>

### Round 1 (19.4.)

In the first official round of system testing, we encountered many bugs. We got a lot of information about the things we still need to fix and add to the application, so despite many tests failing, the testing round was very important.

Test case	Test results	Issues	Testers
1. Verify that the app can establish a MESH network between two devices using WLAN or Bluetooth technology	<b>(1)</b> Fail <b>(2)</b> Fail <b>(3)</b> Pass <b>(4)</b> Fail <b>(5)</b> Pass	<b>(1)</b> Crashed for (probably) unrelated reasons <b>(2)</b> No connection request received, automatically disconnected after timeout <b>(4)</b> After forming a connection, the other device was not visible, and messages were stuck in a "Waiting" mode	Katja Heiskanen Neea Tienhaara Elsa Huovila Vertti Nuotio

2. Verify that the app can authenticate new nodes joining the network and maintain the security and reliability of the network	(1) Fail (2) Fail	(1) Crashed on device 1 after creating the network (2) Connected and sent messages from new node but new node doesn't receive messages	Katja Heiskanen Neea Tienhaara Elsa Huovila Vertti Nuotio
--	----------------------	---	--

Notes from the testing session:

- It would be a good idea to add some sort of indication that the device is connected to other device(s) to both network view and the chat view.
- Sending notifications from the messages should be implemented.
- It is necessary to add error handling and inform the user that there was an error.
- Device list should be updated.
- Make the leave network button does not do anything currently – either remove it or make it do something.
- Delete the sharing location part if it will not be implemented in time.

## Round 2 (21.4.)

In the second round, more bug fixing was done and due to that we did not get to work on more than two test cases. The logic of the network joining became clearer, but the results still indicate that the process of forming the network is still a bit unstable. Connecting 2 devices is starting to mostly work, but there are challenges especially when connecting more than 2 devices.

Test case	Test results	Issues	Testers
1. Verify that the app can establish a MESH network between two devices using WLAN or Bluetooth technology	(1) Fail (2) Pass (3) Fail (4) Pass	(1) Crashed on new testing device (2) Some errors were detected in the process of joining (3) Stuck in the waiting loop again	Vertti Nuotio Roope Mantere Katja Heiskanen Neea Tienhaara
2. Verify that the app can authenticate new nodes joining the network and maintain the security and reliability of the network	(1) Fail (2) Fail (3) Fail (4) Fail	(1) Issue with connecting a new device to 2-device network (2) One device crashed (3) The new device couldn't find the original device and then crashed (4) Unable to connect	Vertti Nuotio Roope Mantere Katja Heiskanen Neea Tienhaara

Notes from the testing session:

- The app takes a long time in sending information through the MESH, could be improved



- List of devices that are connected to the network is needed in indicating their status in network view and network info view

### Round 3 and usability test (28.4.)

This round included both usability testing and recording the demo. More test cases were passed than in previous rounds, which is an improvement.

Test case	Test results	Issues	Testers
1. Verify that the app can establish a network between two devices using WLAN or Bluetooth technology	(1) Pass (2) Pass (3) Fail	(3) Test user's device doesn't send network request to all devices. Button choices in joining the network are confusing to user.	Elsa Huovila Katja Heiskanen Neea Tienhaara Vertti Nuotio
2. Verify that the app can authenticate new nodes joining the network and maintain the security and reliability of the network	(1) Pass (2) Fail	(2) Messages were not received by all devices, so sending messages is not functional in some directions (4 device network).	Elsa Huovila Katja Heiskanen Neea Tienhaara Vertti Nuotio
4. Verify that the app can handle multiple devices joining and leaving the network without affecting the network's overall functionality	(1) Pass (2) Fail (3) Fail	(2) Timed out trying to add 3 <sup>rd</sup> device (3) Timed out and crashed on 2 devices	Elsa Huovila Katja Heiskanen Neea Tienhaara Vertti Nuotio

### Round 4 (4.5.)

Test case	Test results	Issues	Testers
1. Verify that the app can establish a MESH network between two devices using WLAN or Bluetooth technology	(1) Pass (2) Fail	(2) Timed out trying to create network	Roope Mantere Vertti Nuotio

2. Verify that the app can authenticate new nodes joining the network and maintain the security and reliability of the network	(1) Fail (2) Fail (3) Pass	(1) Timed out trying to create network (2) Nothing happened after accepting new node	Roope Mantere Vertti Nuotio
3. Verify that the app can share information and reports, including geolocation information, between devices on the network	(1) Pass (2) Pass	(2) Messages change sides and order seemingly on random	Roope Mantere Vertti Nuotio
4. Verify that the app can handle multiple devices joining and leaving the network without affecting the network's overall functionality	(1) Fail (2) Pass	(1) Nothing happened after adding new node (2) Crashed on the 3 <sup>rd</sup> device for unrelated reasons	Roope Mantere Vertti Nuotio
5. Verify that the app can work in a context where there is no GSM connection available	Test scrapped due to no suitable test location being found	There is no spot near Tampere without GSM connection	Roope Mantere Vertti Nuotio

### 4.3 Usability testing

In the usability tests we focused on making sure that the main tasks in the application, such as joining a network and sending a message, are easy and pleasant to perform. We wanted to find possible issues in the logic of performing those tasks and get comments on the overall outlook of the application.

The usability testing was essentially done as system testing including test participant, whose performance we evaluated. The goal was to see how someone without any prior knowledge of the app would complete the task of creating and joining a network and send messages to the chat. This was done by executing a simple usability test described below with its results and having the user fill a form with questions related to

the testing. In addition, we used Google Accessibility Scanner to make sure our application is accessible. Due to time limitations, usability testing is not very thorough and only one user participated on the tests.

Usability test results are included in system test round 3 table. In summary, the process of creating/joining the network proved somewhat difficult for the user without instructions, and especially because we encountered the bug where all participating devices did not receive the network join request. The user was also confused on which buttons they needed to press when creating a network. On the other hand, the chat view and sending a message was very clear and intuitive for the user, as well as viewing network details and leaving the network.

The test user also filled in a usability questionnaire form, where they left a generally positive review. The form and its answers are included in appendix [U]. The test user commented the following on using the app, noting that there is some lack of feedback on system status: "Sovelluksen käyttö oli helppoa ja intuitiivista, mutta feedback voisi olla näkyvämpää. En esim. ensin huomannut, että olen yhdistettynä networkiin, se selvisi vasta kun avasin chat-välilehden. Siis joku visuaalinen merkki/varmistut että nyt voi chatata valitusta voisi olla hyvä! Ylipäättään positiivinen kokemus!"

## **4.4 Accessibility testing**

The test was done using Google's Accessibility Scanner and revealed 61 issues across all screens tested. The issues were categorized into text scaling, text contrast, image contrast, item label, item descriptions, and touch target. The onboarding screens had the most issues, with a total of 30 issues detected across four screens. In some areas, the repeated pattern of issues was noticeable, which at least partially explains the high amount of issues in some categories. The full accessibility test report provides more in-depth descriptions and recommendations for each issue to make the app more accessible. Improvements were made to the issues that were easiest to fix within this time frame. A total of 34 of the encountered issues were fixed, and these have been marked in the report. The report is included as Appendix [A].

## **4.5 Acceptance testing**

The acceptance test was performed in Tampere 5.5.2023 to verify that the product meets the acceptance criteria set by the customer. The testing team set up a test network with multiple devices and tested the network connectivity, node discovery, and routing.

The team successfully verified that devices can connect to each other and communicate over the mesh network using WLAN data transfer. The network was set up with five devices and tested with different message transfer scenarios.

During the test, the team discovered some issues that were already identified during the system testing. However, the team was able to quickly resolve the issue and continue with the testing.

The team also tested the node discovery and routing functionality of the network. They sent messages from one device to another, and the messages were routed through the network correctly but sometimes some devices did not receive the messages. They also closed the WLAN from a device that was between two devices and verified that

the message was routed in a new route, or the sender was informed that there was no route between nodes.

Overall, the testing team concluded that the product met the acceptance criteria defined by the customer and passed the acceptance test.

## **APPENDIX A [...Z]**

[A] Accessibility test report: [Accessibility test report.docx](#)

[U] [Usability survey form](#)

[U] [Usability survey \(results view\)](#)