Sattar Pourkand

PLA-32820 2019 Mobiiliohjelmointi

Oppimispäiväkirja

2019

SISÄLLYSLUETTELO

[1. Harjoitus 1 – Tutustuminen mobiiliympäristöihin 1](#_Toc20055345)

[2. Harjoitus 2 Git versiohallinta 2](#_Toc20055346)

[3. HARJOITUS 3 - ANDROID STUDIO JA HELLOWORLD 13](#_Toc20055347)

[3.1 Android Studion asentaminen 13](#_Toc20055348)

[3.2 Hello World sovellus 18](#_Toc20055349)

[3.3 Git -versiohallinnan lisäämistä hello world-projektiin 24](#_Toc20055350)

[4. Harjoitus 4 ja 5 30](#_Toc20055351)

[5. Harjoitus 6, 7 ja 8 - Tietokanta ja käyttöliittymä 31](#_Toc20055352)

[6. Harjoitus 9 ja 10 – Firebase 32](#_Toc20055353)

[7. Harjoitus 11 – Sensorit 33](#_Toc20055354)

[8. Harjoitukset 12-14 34](#_Toc20055355)

[9. Harjoitus 15 35](#_Toc20055356)

[Lähteet 36](#_Toc20055357)

# Harjoitus 1 – Tutustuminen mobiiliympäristöihin

Valitsen oma puhelin. Sen perustiedot löytyvät [tästä](https://www.oneplus.com/fi/support/spec/oneplus-5t). Minun puhelimen Android versio tällä hetkellä on 9. Puhelimen RAM on 6 GB ja ROM 64 GB. Puhelimen CPU on Snapdragon 835.

Laitteen (Androidin) mahdolliset ohjelmointikielet ovat: Java, Kotlin, C/C++, C#, BASIC, Corona/LUA ja PhoneGap (HTML, CSS, JavaScript).

<https://www.androidauthority.com/develop-android-apps-languages-learn-391008/>

Ohjelmointiin tarvittavat työkalut ovat mm.: Android Studio, ADB (Android Debug Bridge), AVD (Android Virtual Device) Manager, Eclipse, Fabric, FlowUp, GameMaker: Studio, Genymotion, Gradle, IntelliJ IDEA, Instabug, LeakCanary, NimbleDroid, RAD Studio, Stetho, Source Tree, Unity 3D, Unreal Engine, Visual Studio with Xamarin, Vysor.

<https://www.altexsoft.com/blog/engineering/top-20-tools-for-android-development/>

Laitteesta ominaisuudet: Bluetooth, GPS, NFC, USB, Sormenjälki, kiihtyvyysanturi, gyro, proximity, kompassi, Kamera, Kaiutin ja mikrofoni.

<https://www.gsmarena.com/oneplus_5t-8912.php>

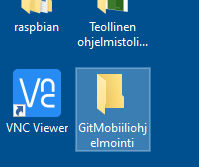
# Harjoitus 2 Git versiohallinta

Edellisten kurssien aikana olen jo asentanut tarvittavat työkalut GIT versiohallintaan.

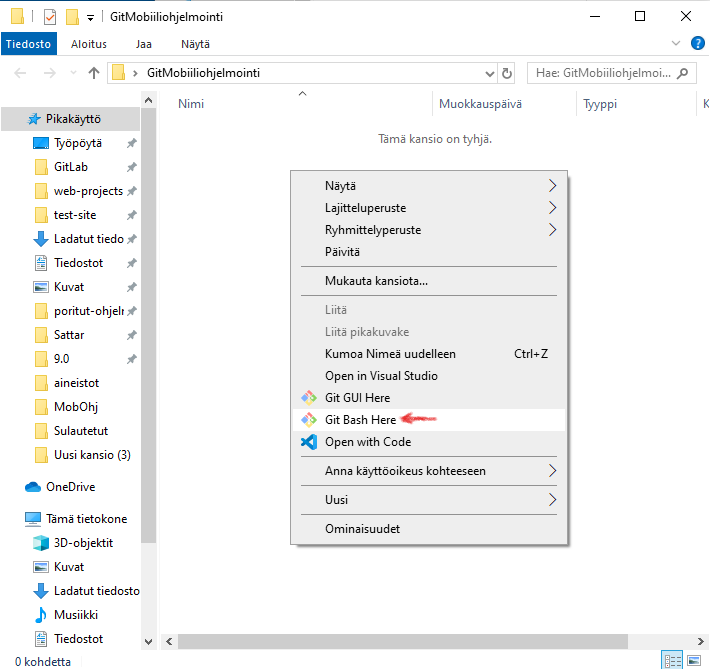


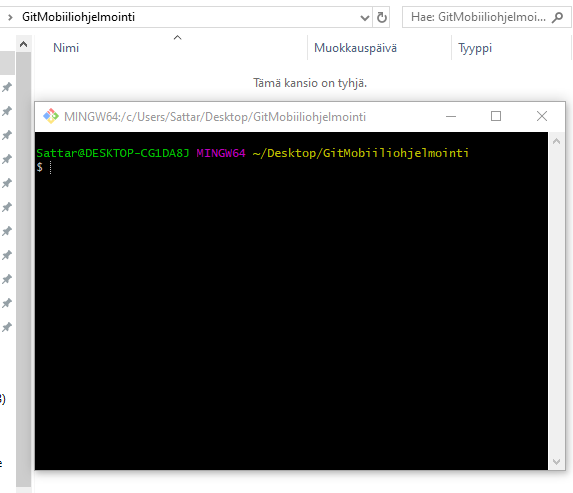
Kloonaan Git Bash terminaalin avulla:

Teen työpöydällä hakemiston nimellä ”GitMobiiliohjelmointi”.

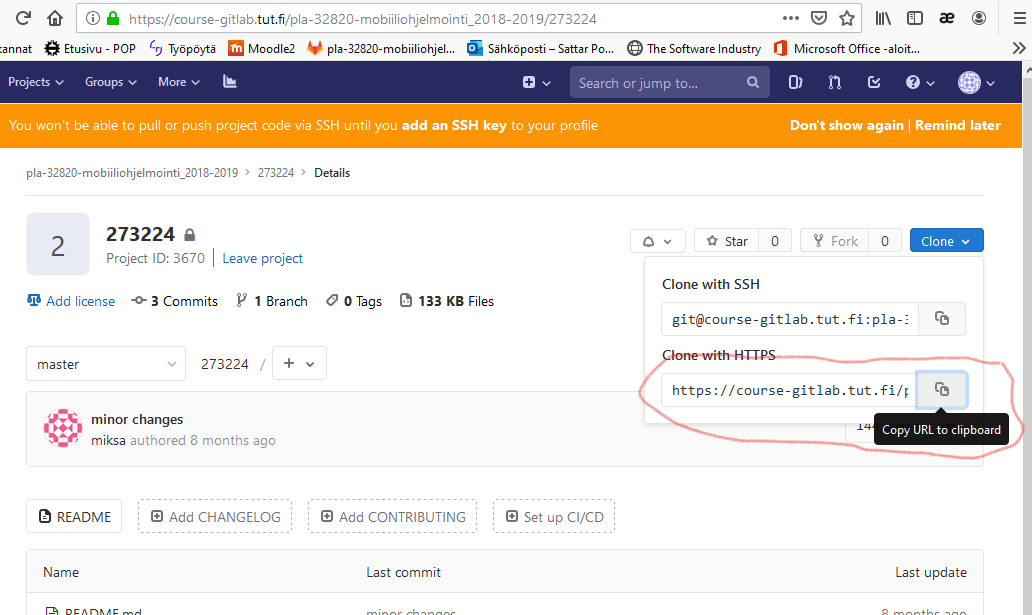


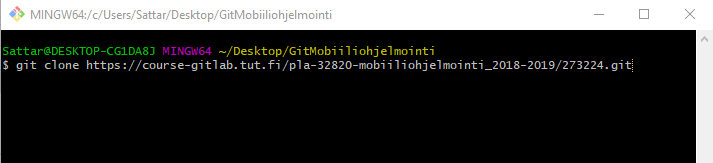
Avaan ”GitMobiiliohjelmointi” hakemiston ja avaan Git Bash terminaalin siellä.





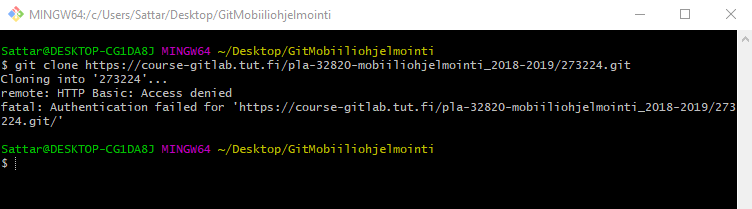
Kopioin GitLab projektin sivusta kloonauksen HTTPS-osoite.





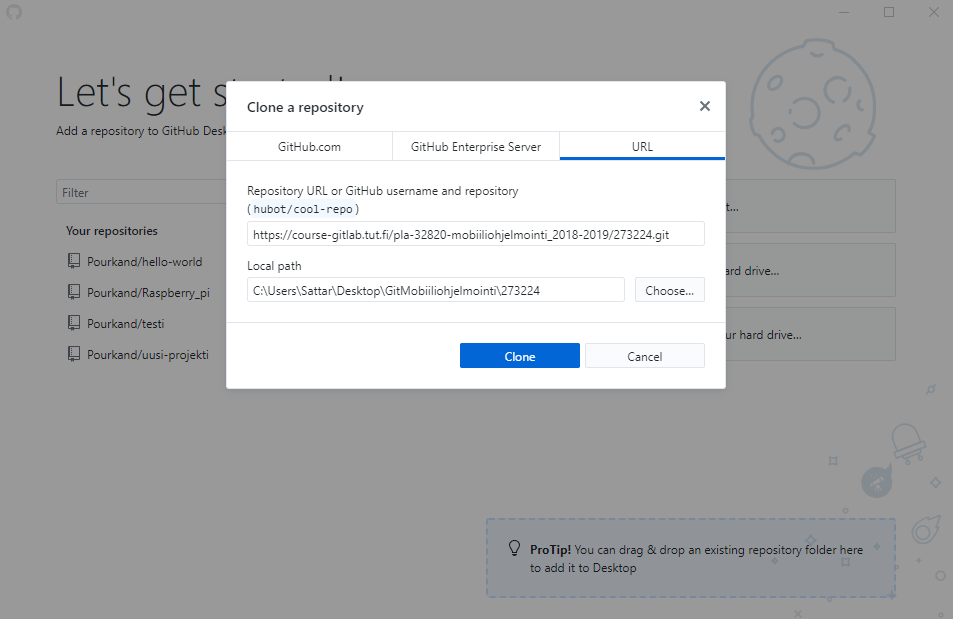
Jostain syystä tämä ei onnistu. Pari viikkoa sitten kloonasin tietokantojen projekti GitLabista ja se onnistui hyvin.

Yritin selvittää mistä ongelma johtuu ja miten voi ratkaista, mutta en saanut selvitetty. Ehkä johtuu Windows 10 päivityksistä.

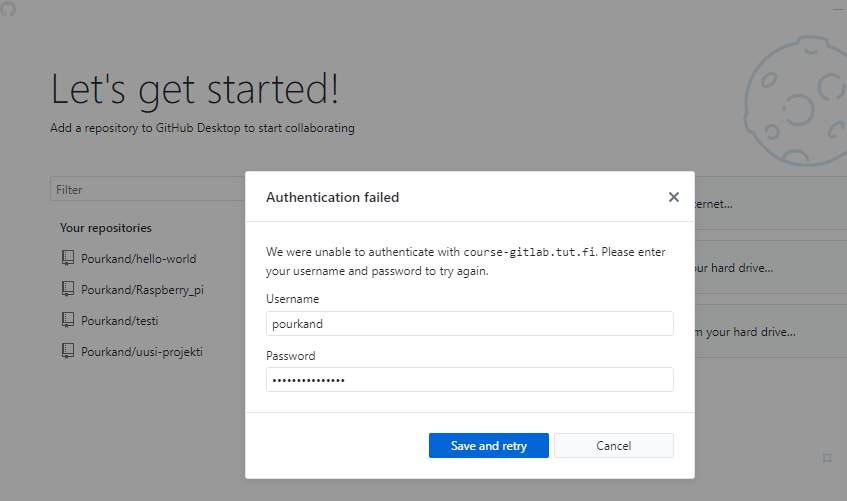


Yritän seuraavaksi kloonata ”GitHub Desktop” ohjelman avulla.

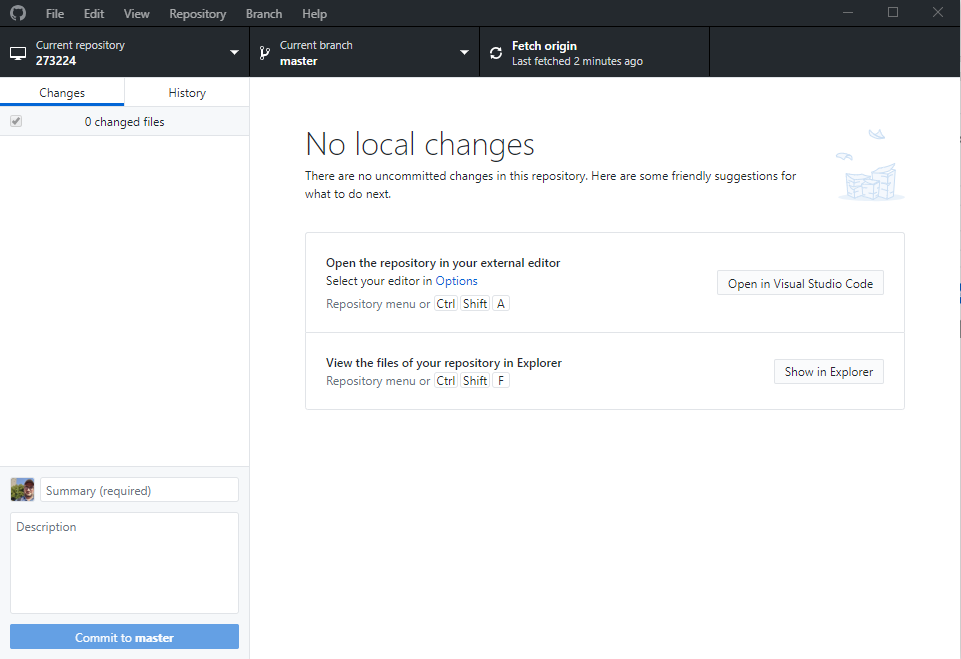
Annetaan ohjelmalle Reopn osoite ja paikallinen hakemisto, johon halutaan kloonata.



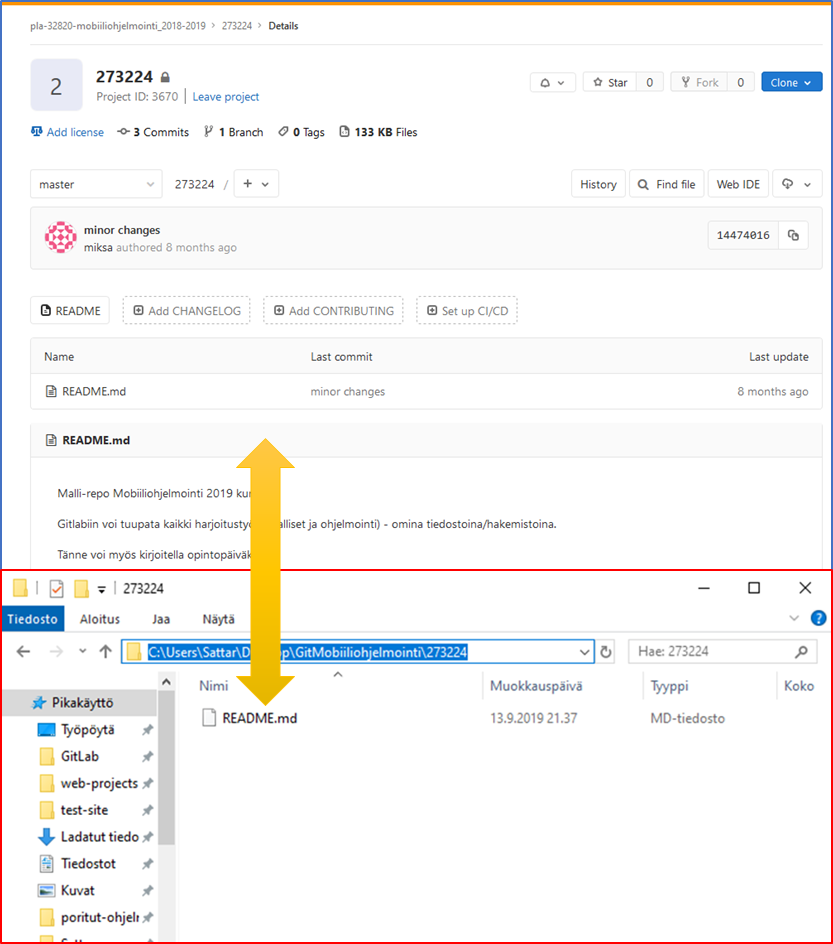
Ohjelma pyytää tilan käyttäjätunnus ja salasana. Pyydetyt tiedot annetaan.



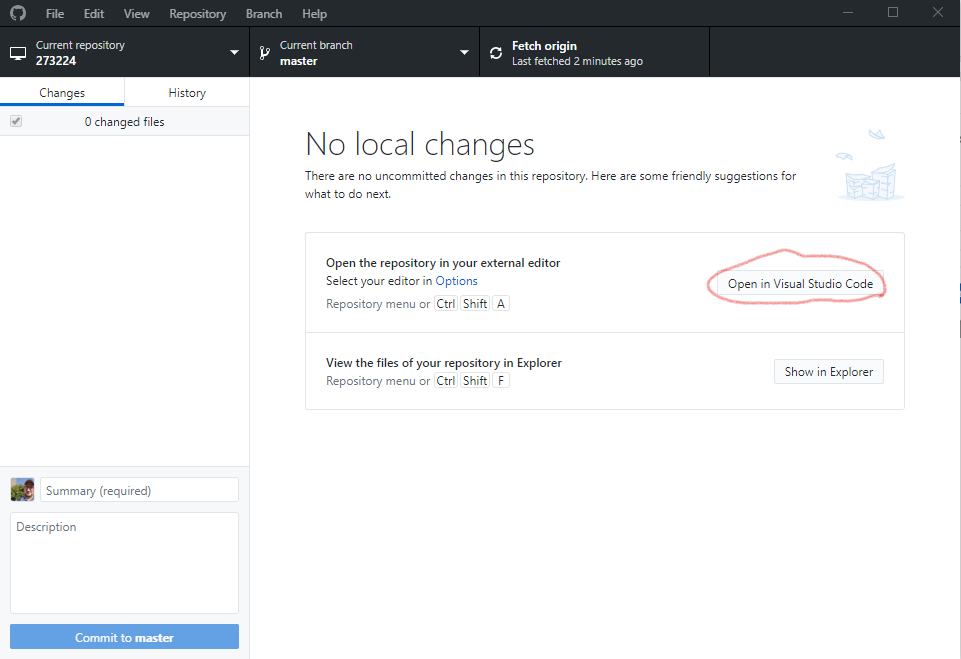
Kloonaus täällä kertaa onnistui.



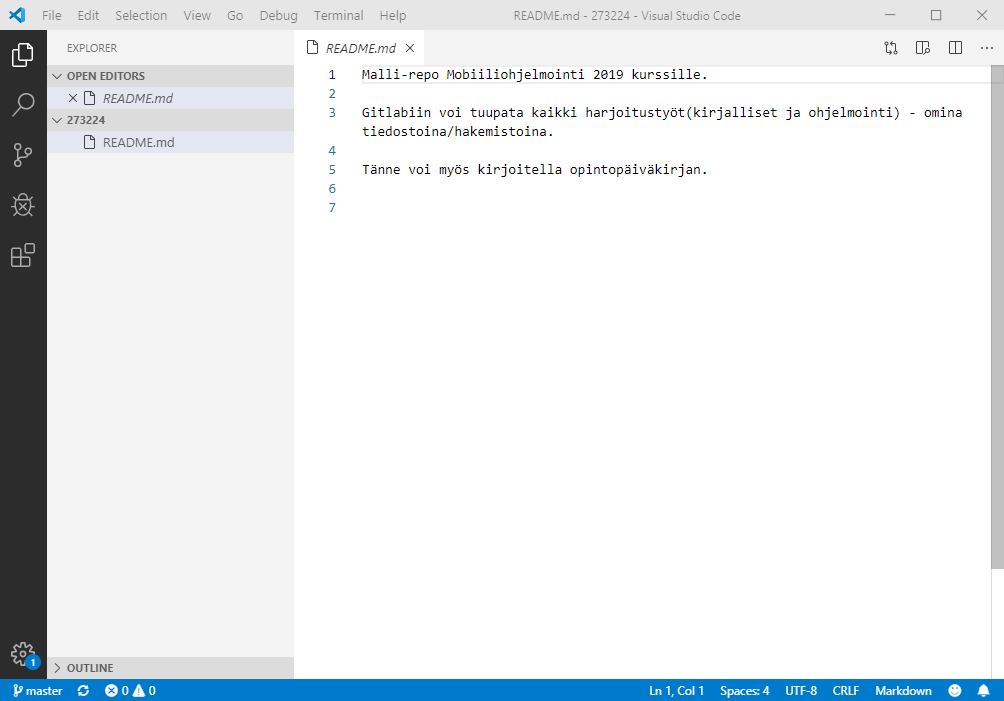
Klikkaamalla Show in Explorer saadaan näkyville kloonatun repon sisältö (nyt vain README.md).



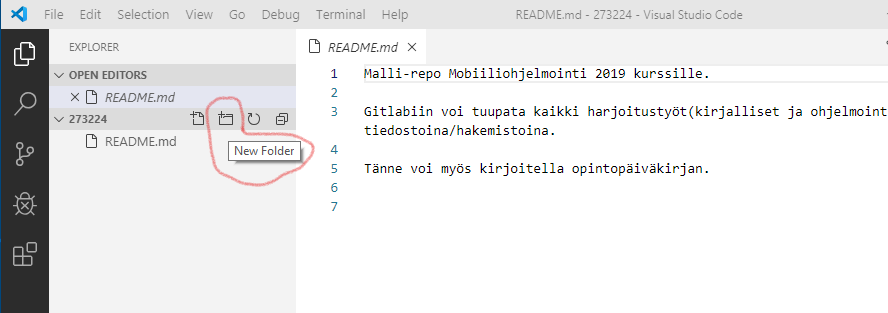
Editorina voidaan käyttää esim. Visual Studio Code.

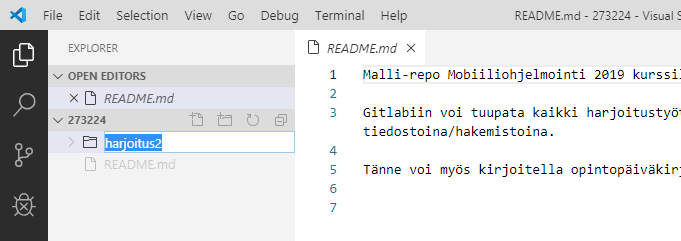


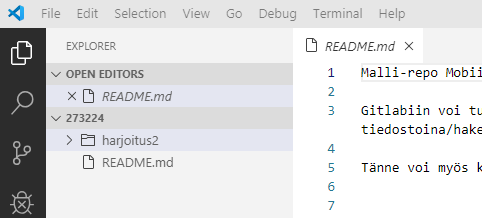
Avattu VSC:ssä.



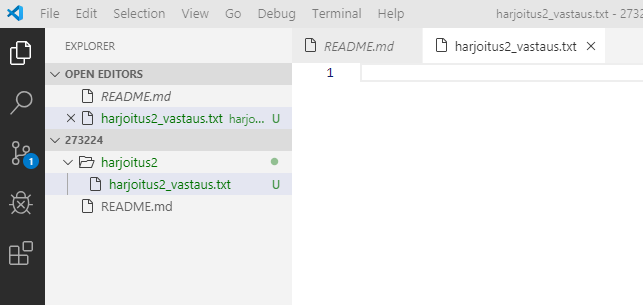
Lisäsin harjoitus2 -hakemisto





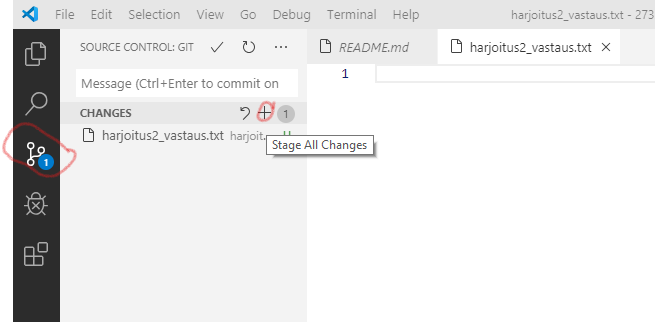


Lisäsin harjoitus2\_vastaus -dokumentti edelliseen hakemistoon

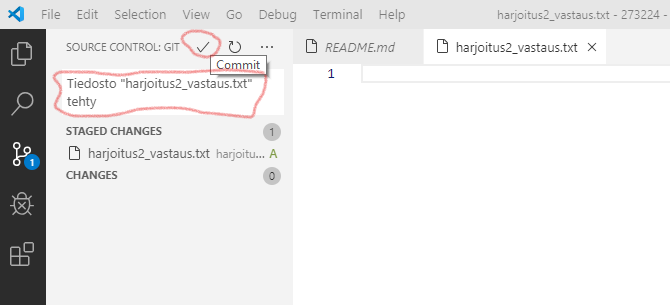


lisäsin harjoitus2\_vastaus -versiohallintaan VSC:n työkalun avulla.

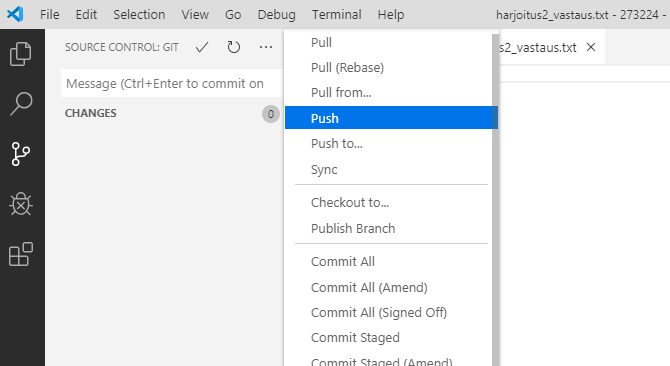
Tämän voidaan tehdä VSC:n terminaalin kautta.



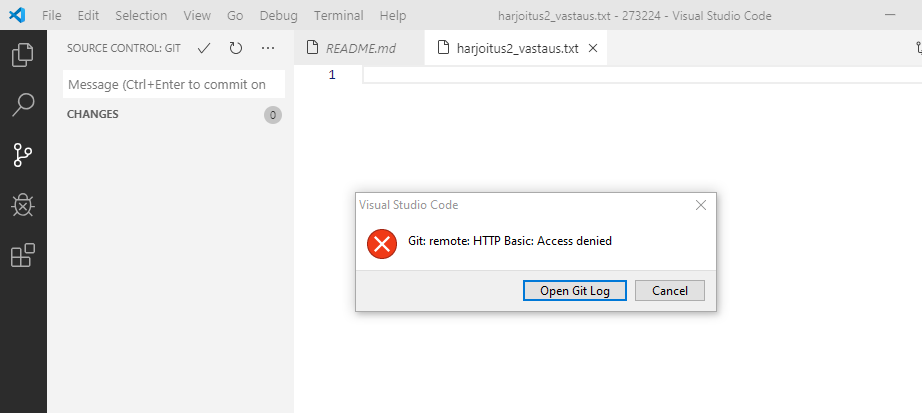
Sitten selitys commit-kenttään viesti mitä tehtiin ja commit.



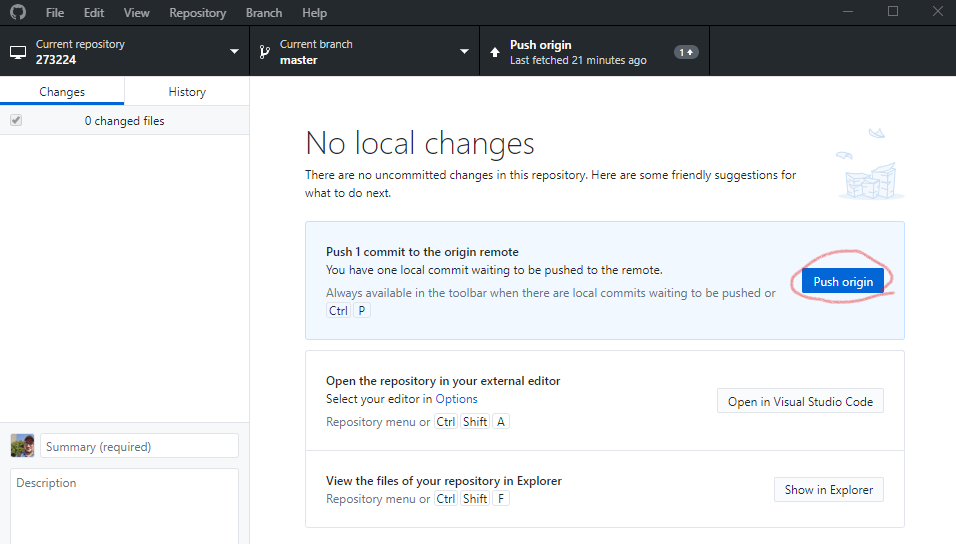
Sitten Push, jotta paikallinen git repo menee palvelimelle.



Push VSC:n kautta ei onnistu.



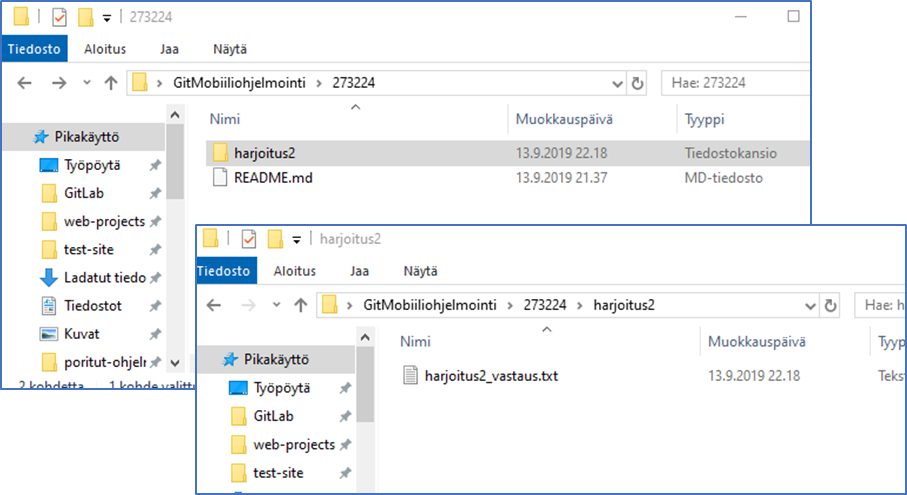
Push onnistuu kuitenkin GitHub Desktop ohjelman kautta.



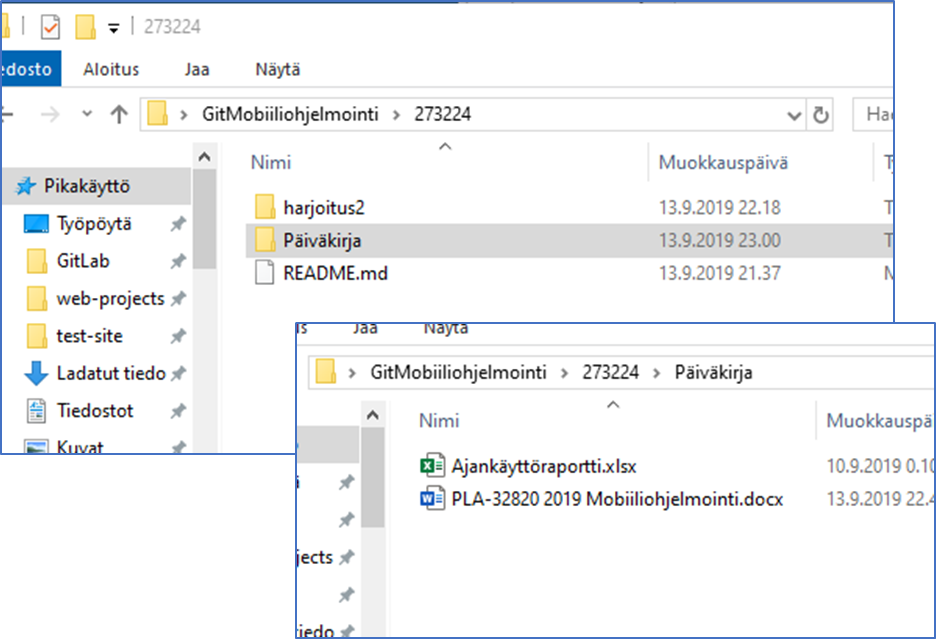
Kun päivitetään projektin sivu GitLabissa, niin saadaan näkyville meidän uusi kansi, uusi tiedosto ja commit.



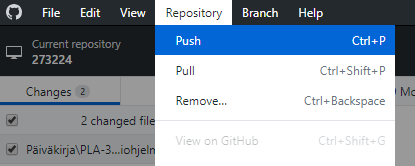
Vastaavat muutokset repon paikallisessa hakemistossa.



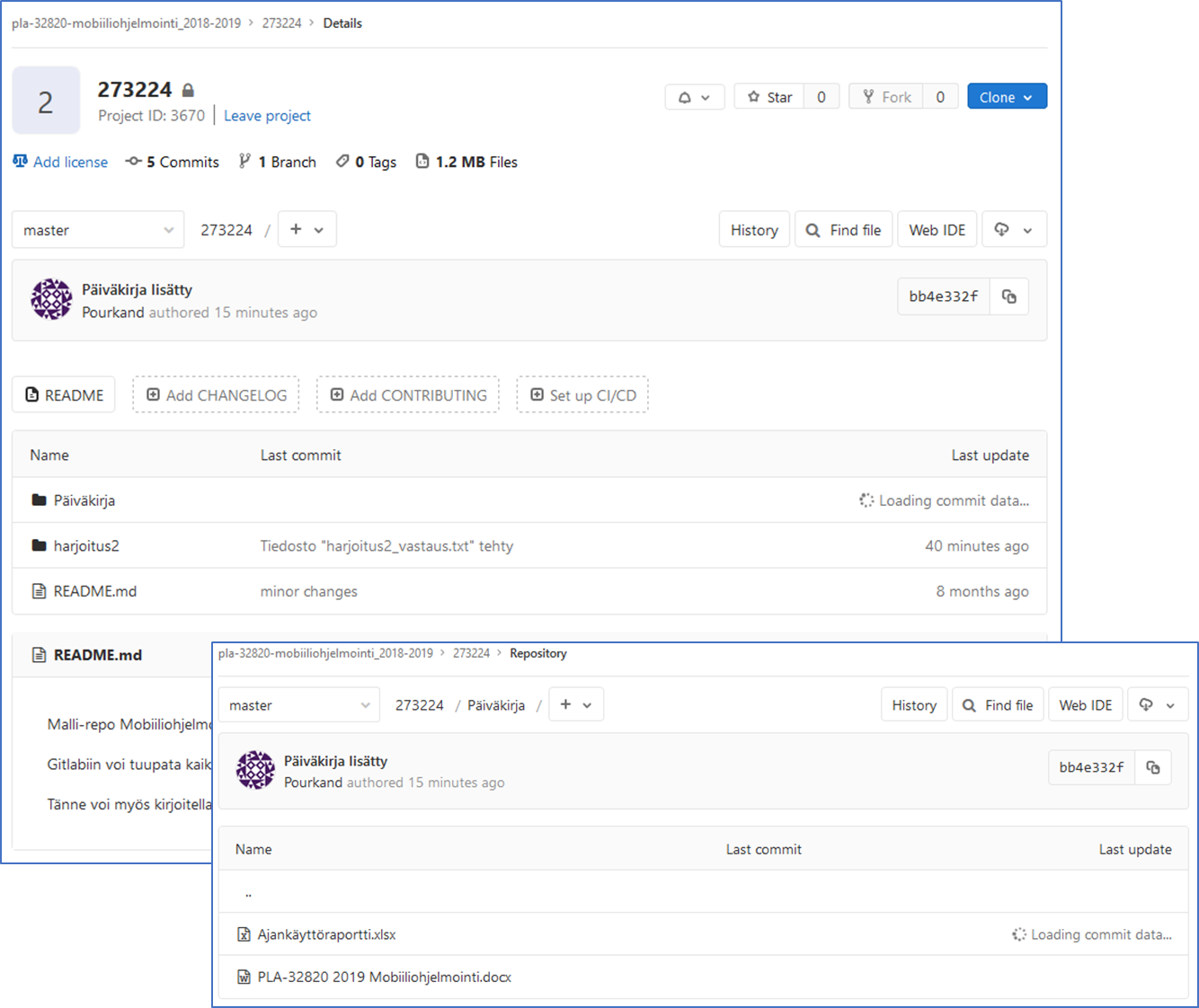
Lisäsin hakemisto Päiväkirja ja kopioin sen sisälle tiedostot PLA-32820 2019 Mobiiliohjelmointi.docx ja Ajankäyttöraportti.xlsx.



Lisäsin commit teksti ”Päiväkirja lisätty” ja tein push GitHub Desktop ohjelmassa.



Sain muutokset näkyville repon sivuun.



Lisään jatkossa jokaiselle harjoitukselle oman hakemiston.

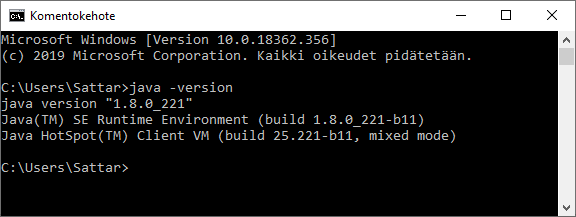
# HARJOITUS 3 - ANDROID STUDIO JA HELLOWORLD

## Android Studion asentaminen

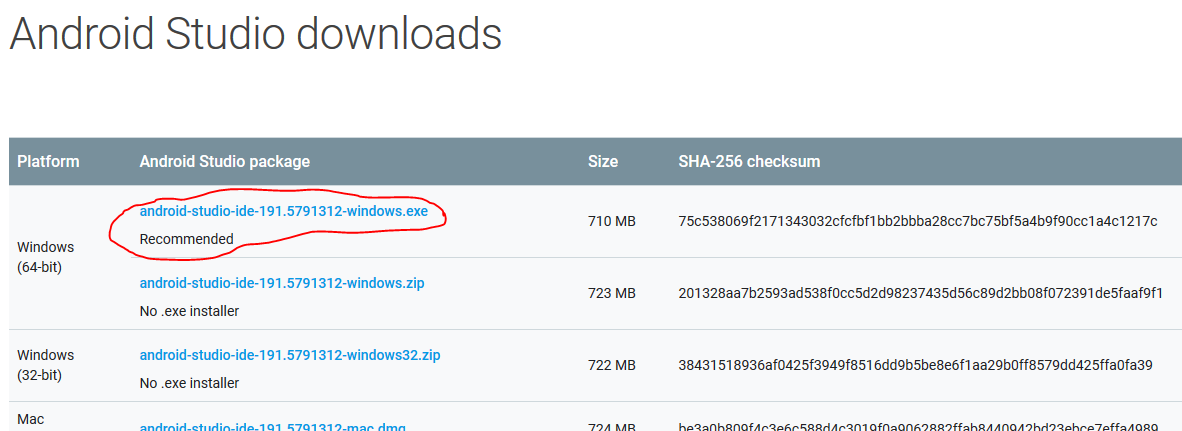
Asenna Android työkalut Johonkin tietokoneeseen.

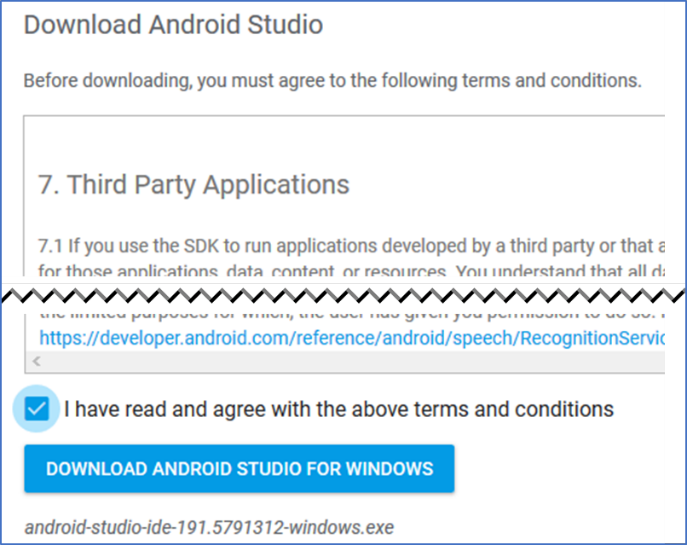
Tarkistan onko Java Developer Kit koneessa.

Versio 8.0\_221 on asennettu.



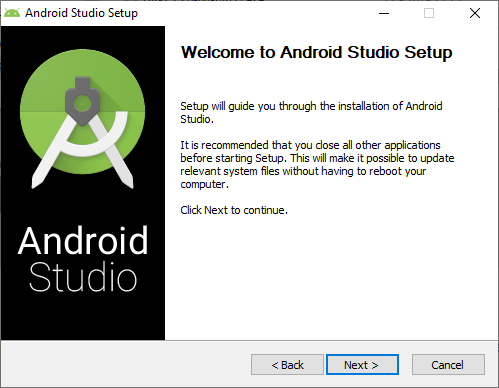
Android studion lataus.

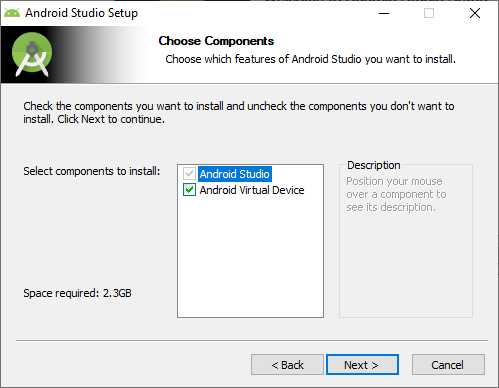


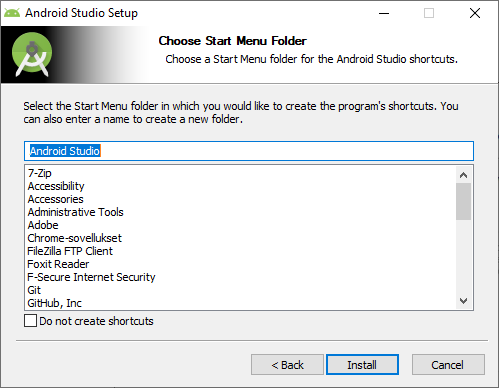


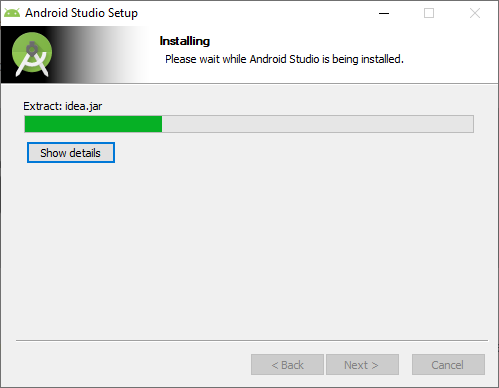
Android studion asennus.

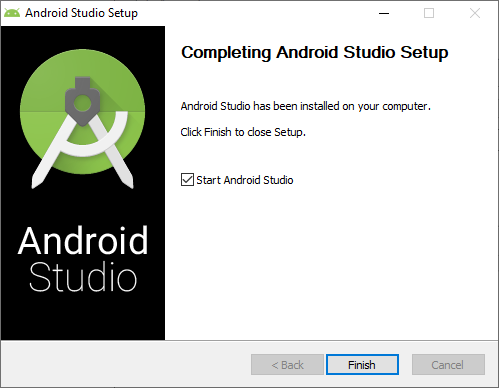


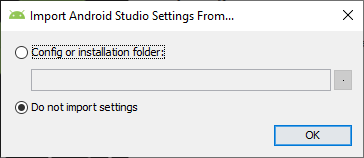


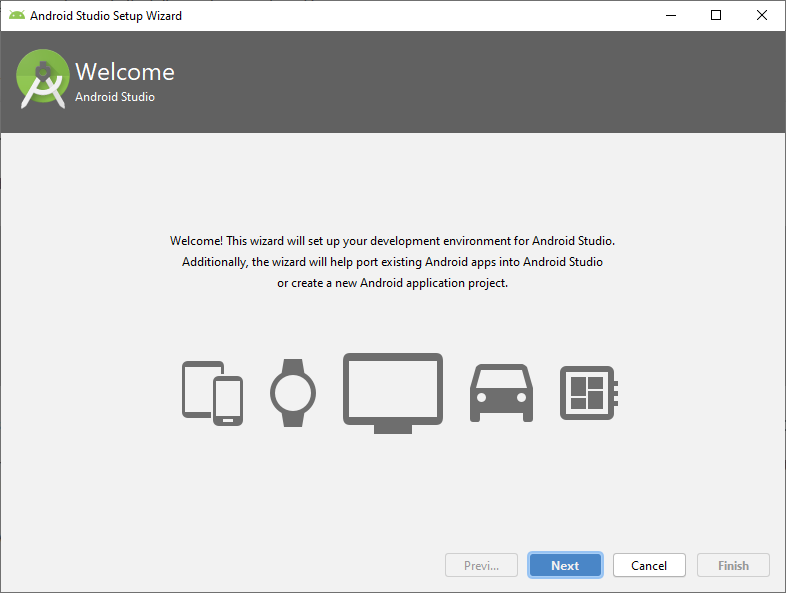




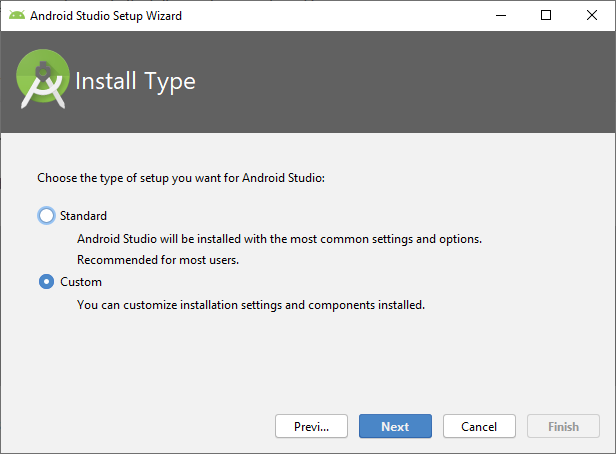


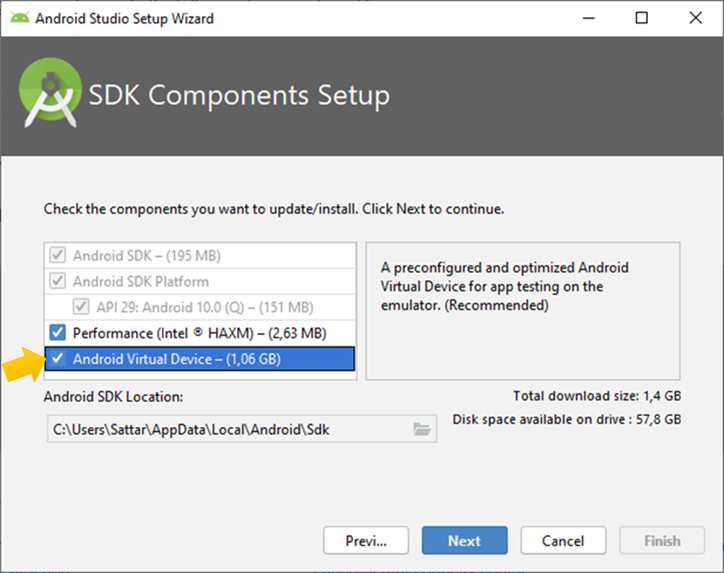


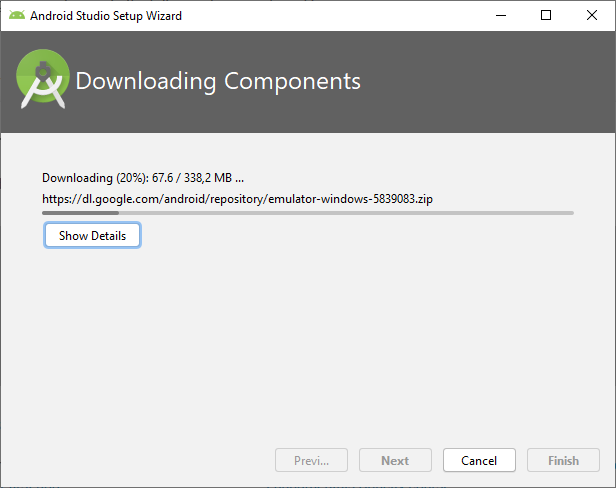


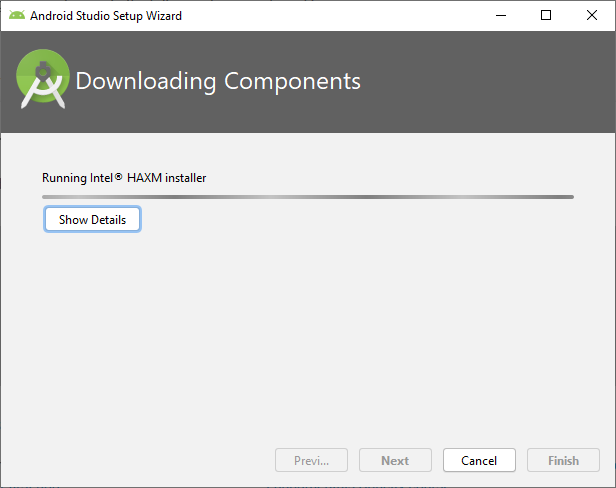


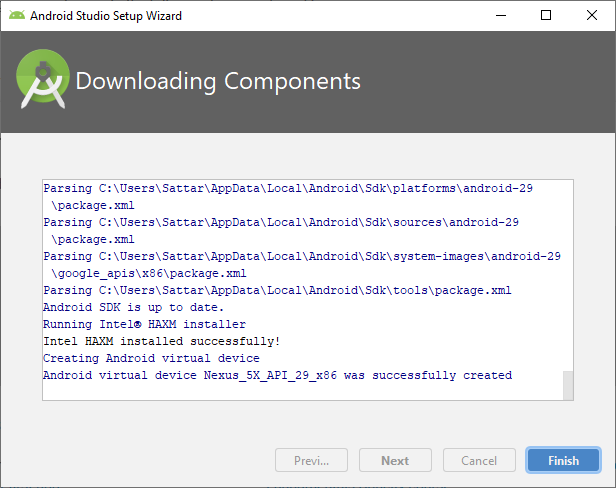
Valitsin Android Virtual Device mukaan.







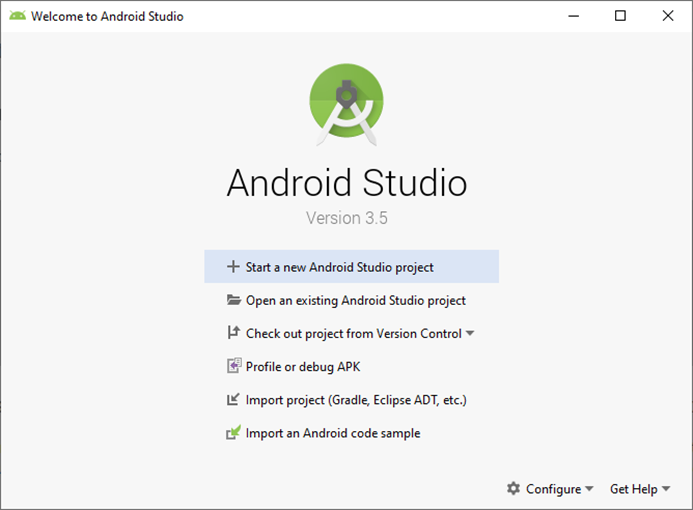




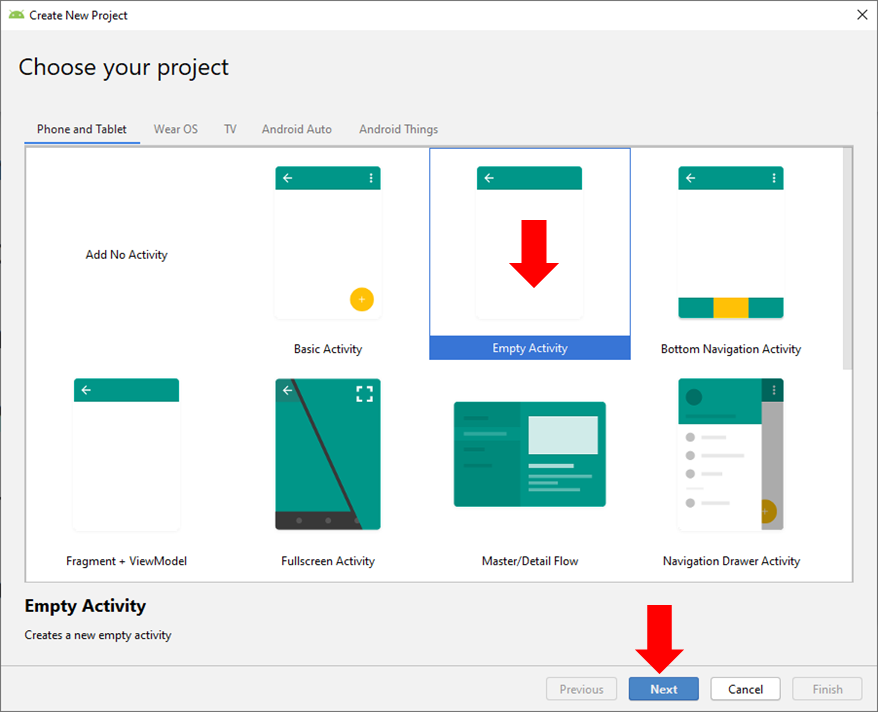
## Hello World sovellus

Testaan ympäristön toimiminen.

Aloitan uuden projektin.



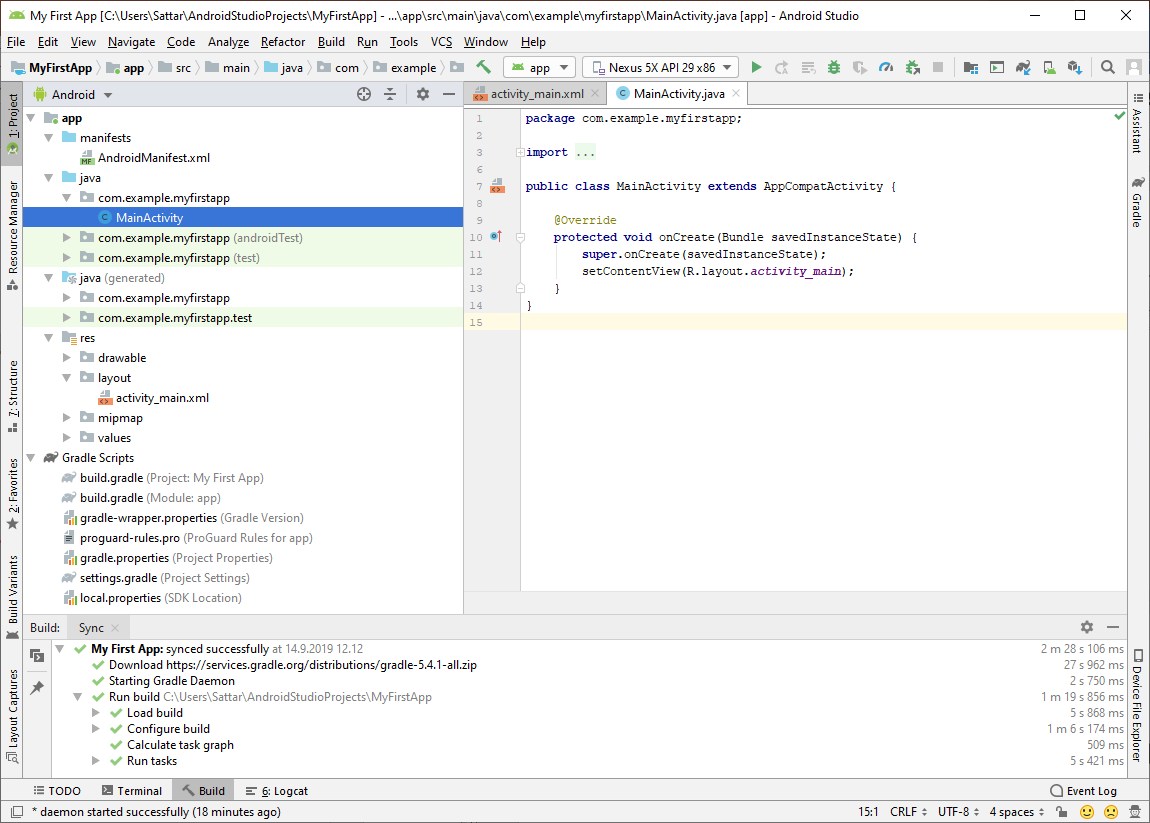
Tyhjä aktiviteetti.



Projekti konfiguroidaan. Projektin nimi ”My First App”, ohjelmointikieli Java jne.



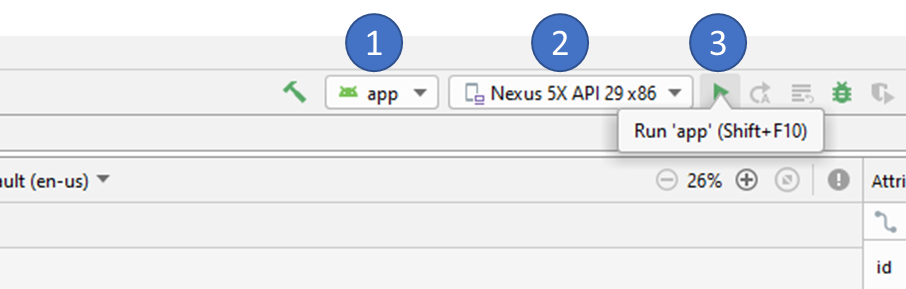
Projekti muodosui.



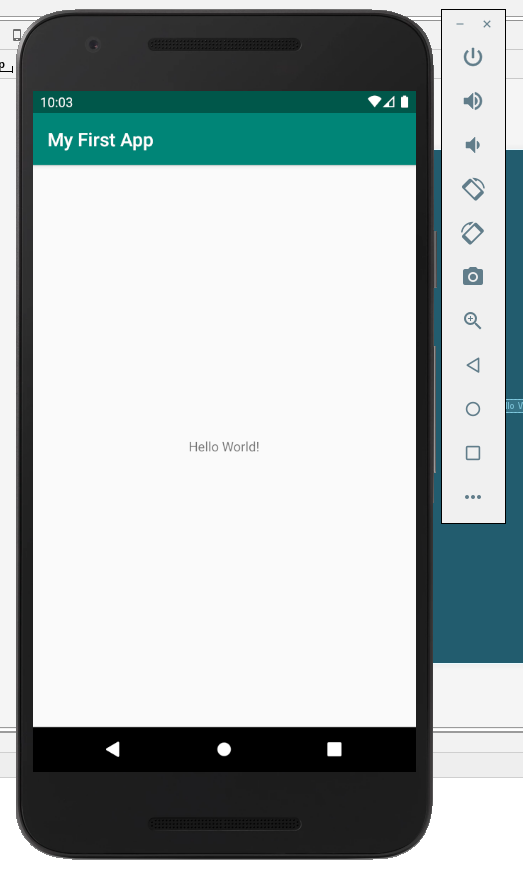
Projektilla on ”MainActivity”!



Ajetaan ohjelman ensin virtuaalilaitteessa.

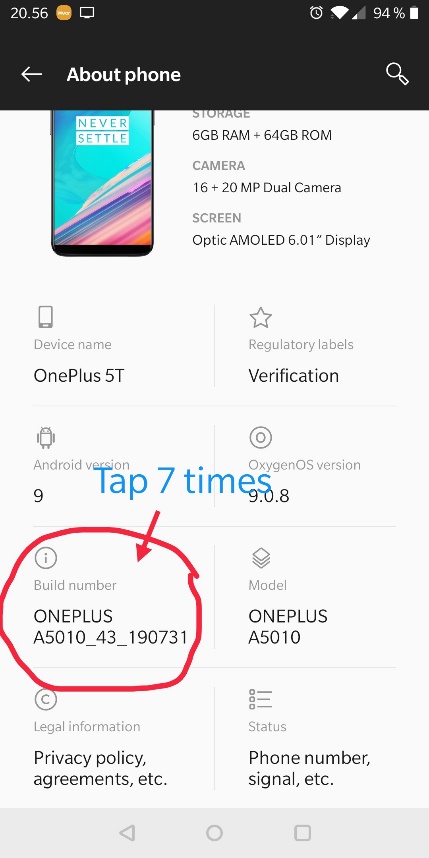


Toimii!

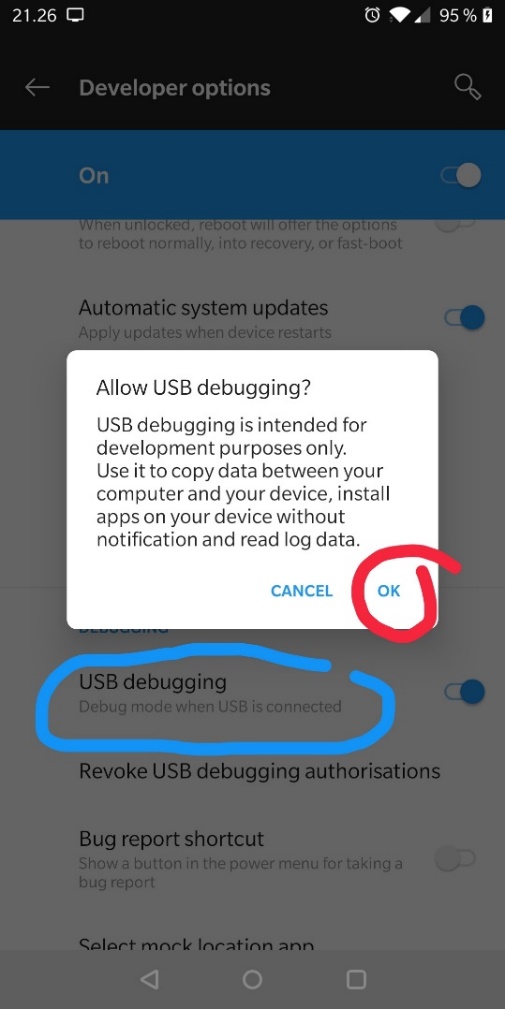
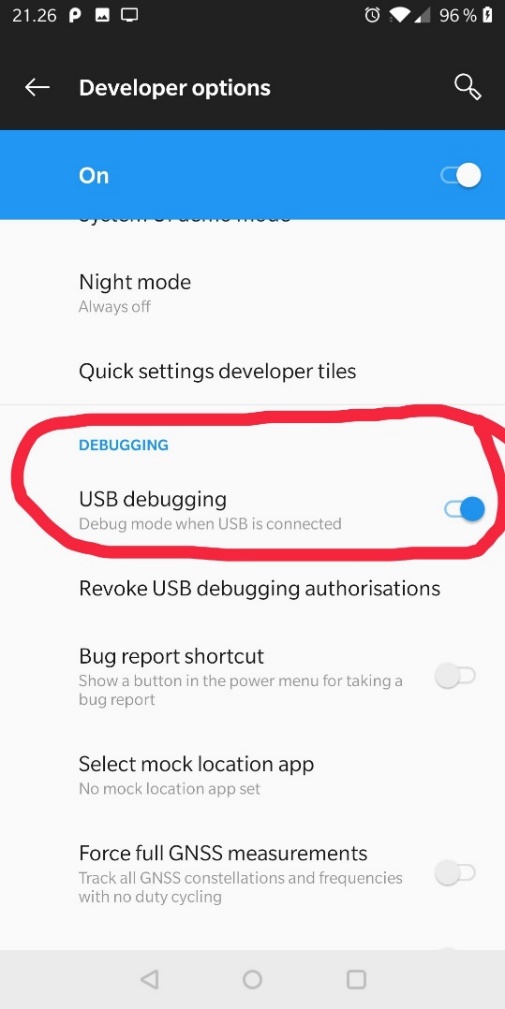


Sitten yritän ajaa omassa puhelimessa.

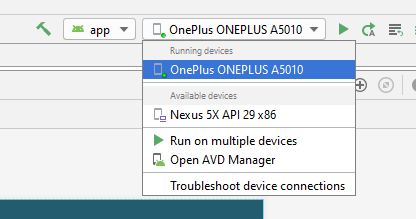
Täytyy napauttaa 7 kertaa peräkkäin kohdetta ”Build number”. Näin kehittäjätila aktivoituu puhelimessa.



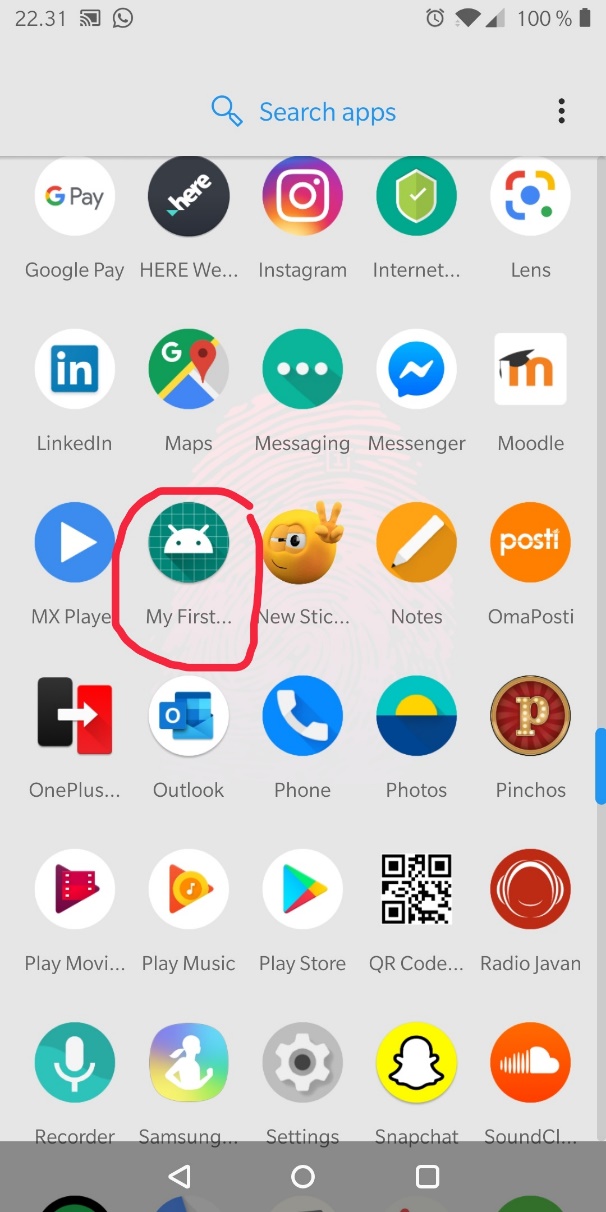
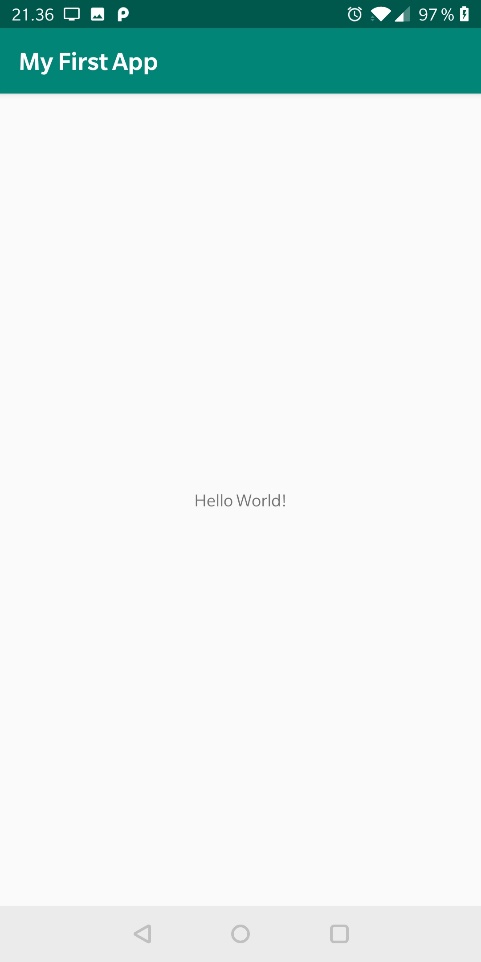
Puhelimen kytkentäajurit asennetaan tietokoneeseen. Yhdistetään puhelin USB-kaapelin kautta tietokoneeseen. Aktivoidaan USB debuging puhelimessa.

Nyt puhelin ilmestyy Android Studion laitevalikossa. Valitaan puhelin ja ajetaan ohjelman.

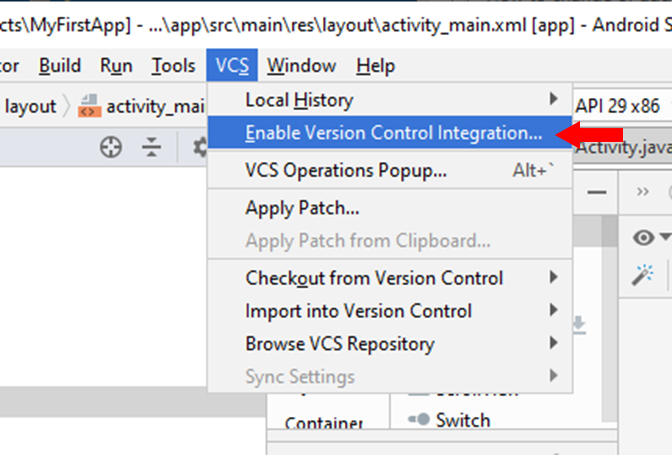


Ohjelma asentuu puhelimeen. Nyt voidaan ajaa Hello World puhelimessa!

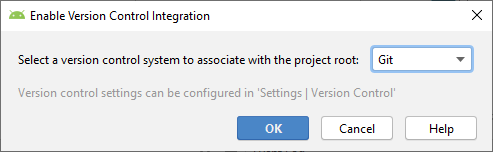
 

## Git -versiohallinnan lisäämistä hello world-projektiin

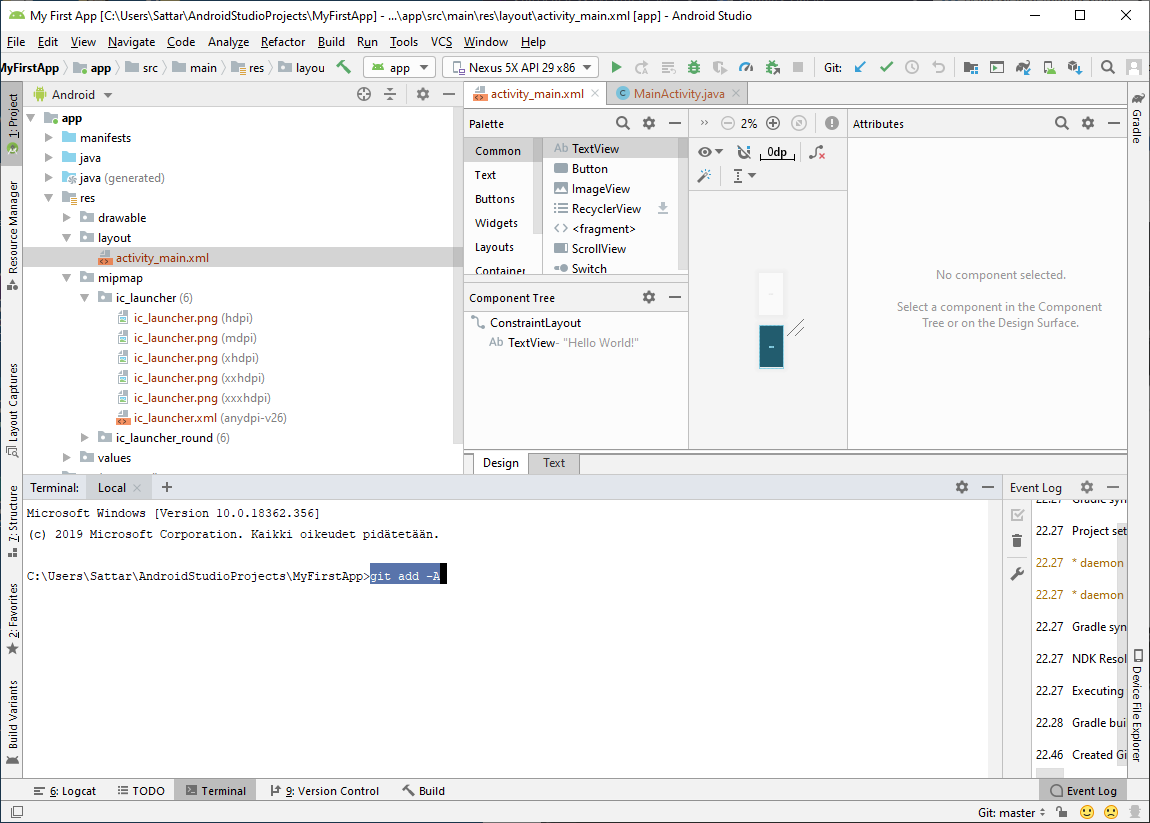
Otetaan versionhallinta Android Studiossa käyttöön.



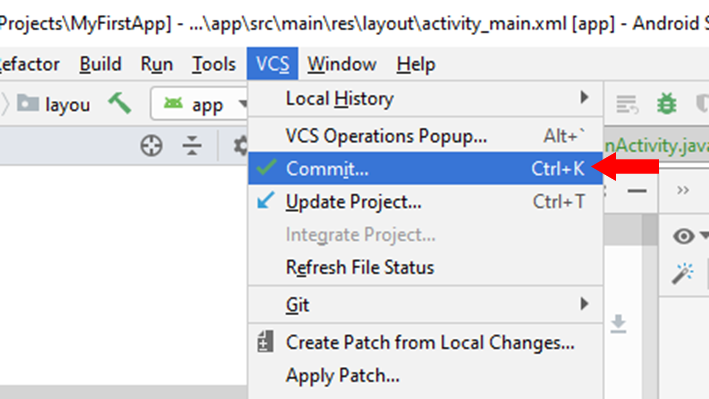
Versionhallintajärjestelmä 🡲 Git

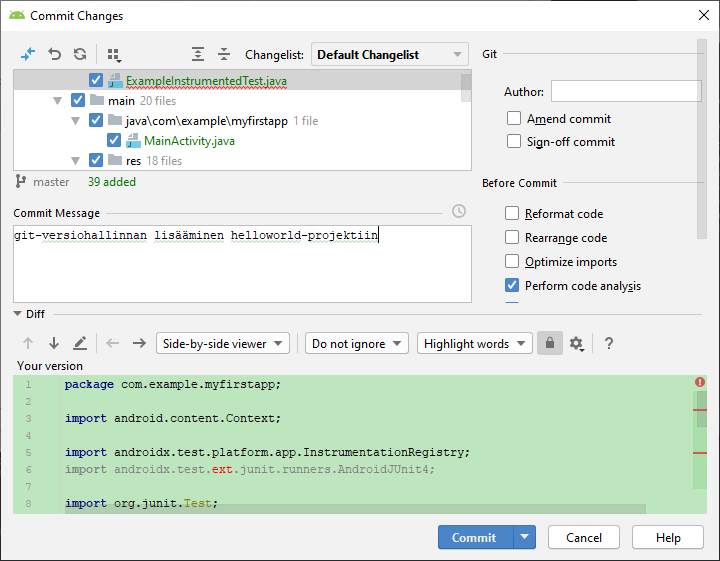


Suoritetaan terminaalissa (tai valikon kautta) ”git add -A”

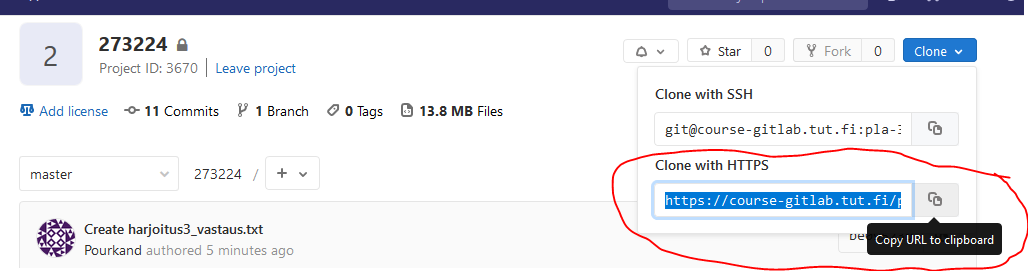


Sitten Commit

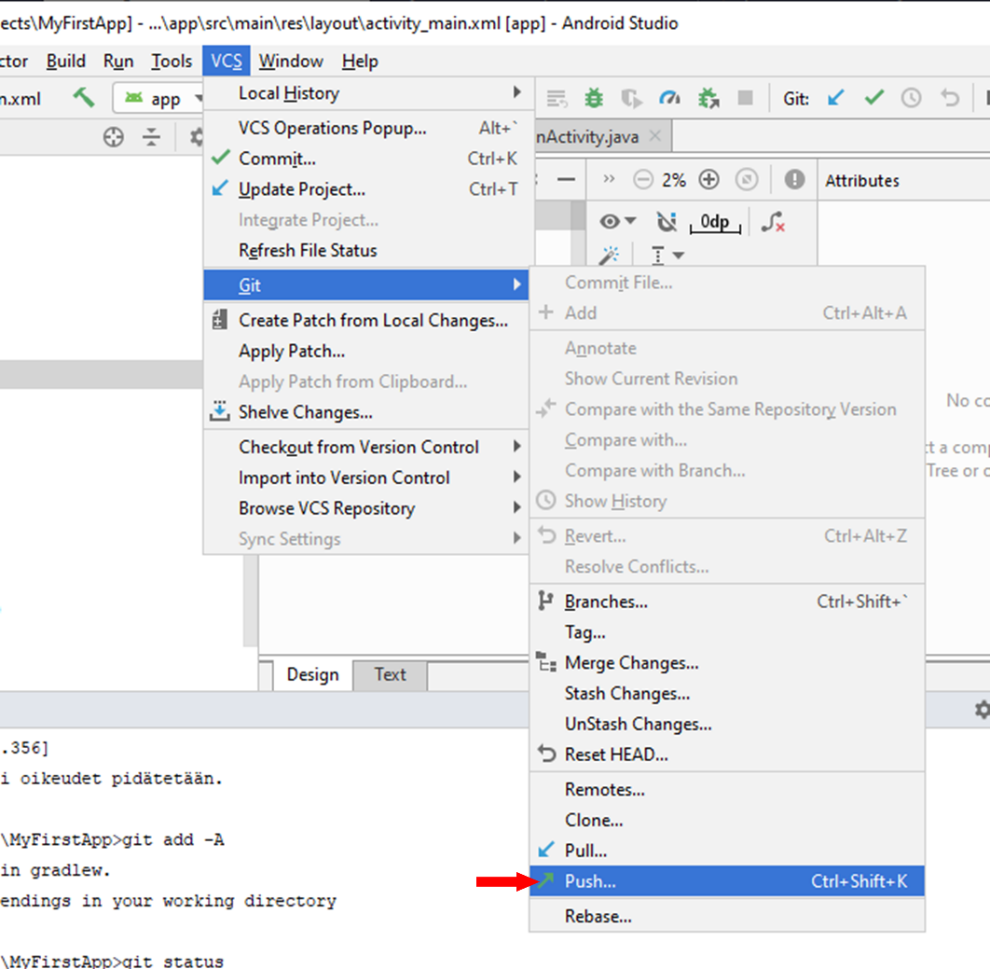




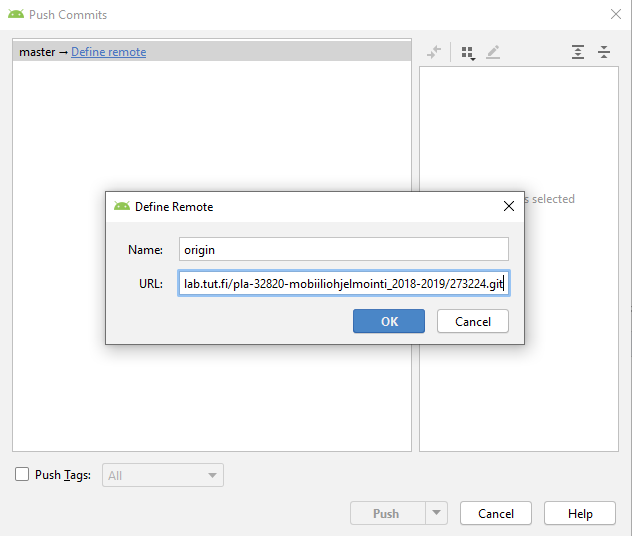
Kopioidaan projektin kloonauksen HTTPS-osoite gitlabista.



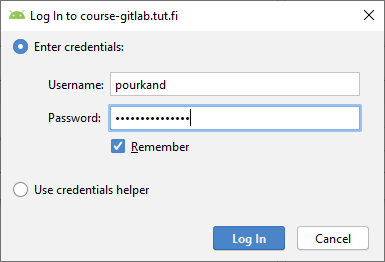
Nyt suoritetaan Push.



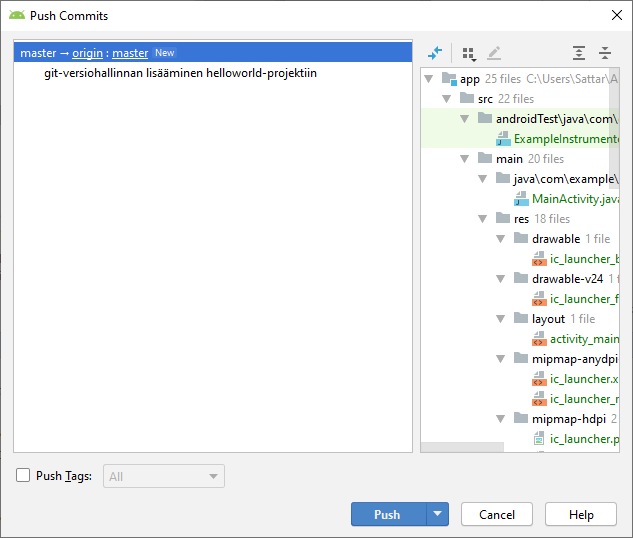
Liitetään kopiotu osoite remoten URL-kenttään.



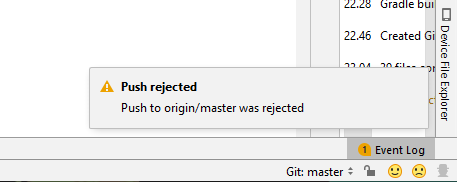
Annetaan kohteen käyttäjätunnus ja salasana.



Ja Push.



Jitlab hylkää yhteyttä. Tämä tapahtui alussa myös Git Bashin käytön yhteydessä.



Yritin terminaanin kautta.

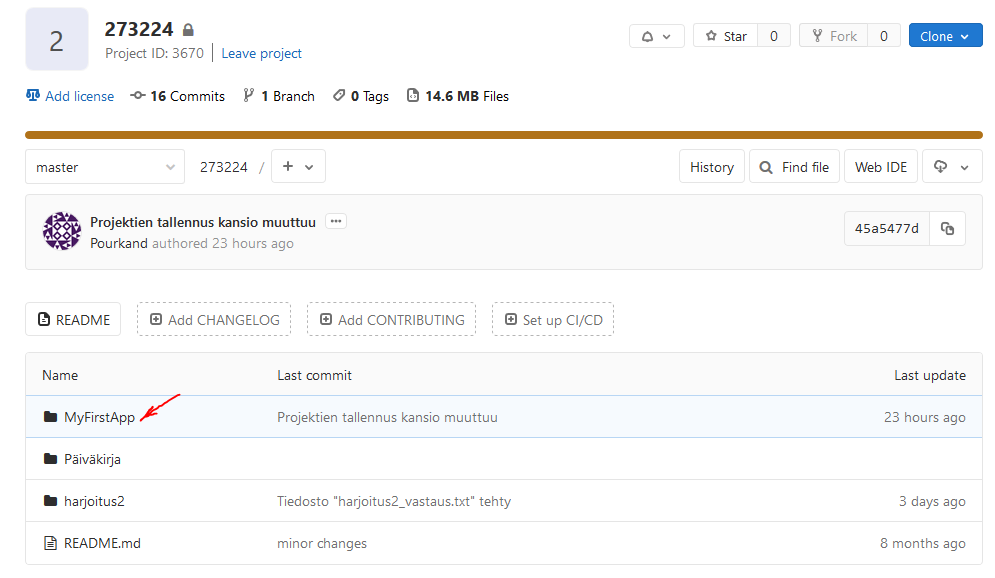
Ei onnistu. Access denied!

Yritin monta kerta uudelleen, mutta en onnistunut.

…

Resetoin Android Studio ja tein HelloWorld projekti uudelleen, mutta tallensin vähän eri tavalla.

Tällä kerta valitsin projektin tallennuspaikka manuaalisesta. Määräsin tallennuspaikaksi sama kuin ”GitHub Desktop” ohjelmaan kuuluvan kloonauskansio, eli ”C:\Users\Sattar\Desktop\GitMobiiliohjelmointi\273224”. Heti Push komento Gitlabiin onnistui!



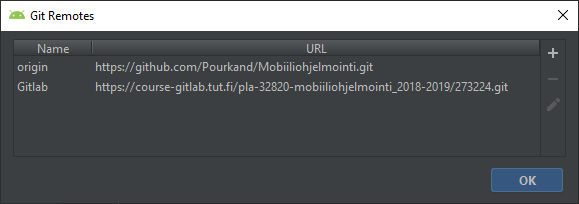
Sitten yritin lisätä projektin Gihubiin.

Käytin sama ohje, joka ei toiminut Gitlabin kanssa. Ohje kuitenkin toiminut hienosti Githubin kanssa. Projekti lisätty Githubiin!

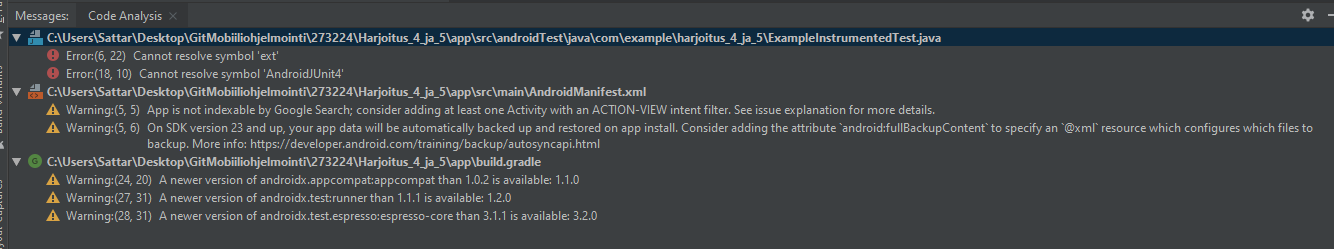


Nyt Android Studion remote listassa on sekä gitlab että github. Push kommento kohdistuu kuitenkin vain ”origin” osoitteeseen. Yritin saada molemmat päivittymään nettiohjeiden avulla terminaalissa, mutta en onnistunut.

Vaihdan origin joka kertaa päivitettävään osoitteeseen. Ei ole vaikea!



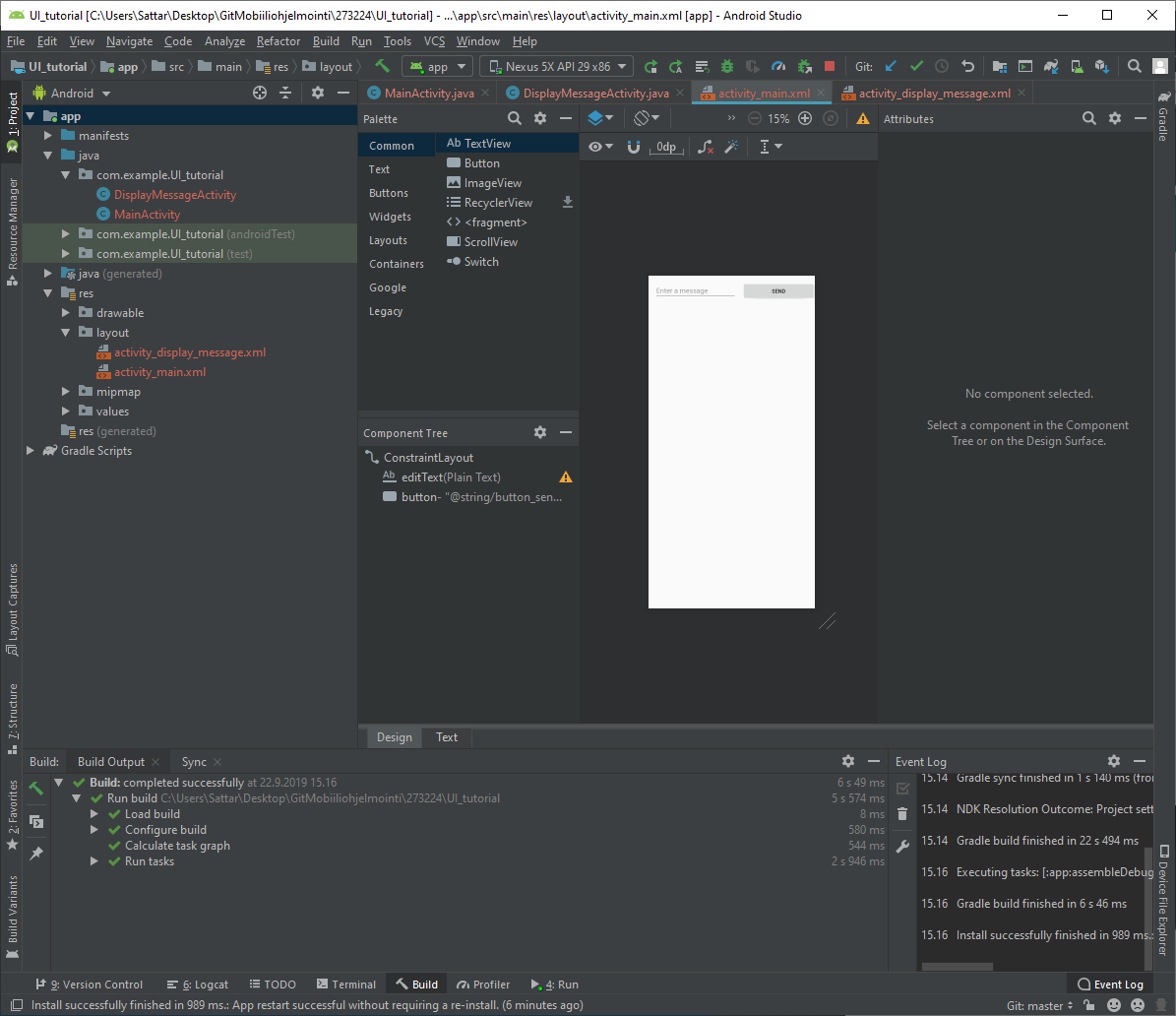
Add komento aiheuttaa varoituksia.

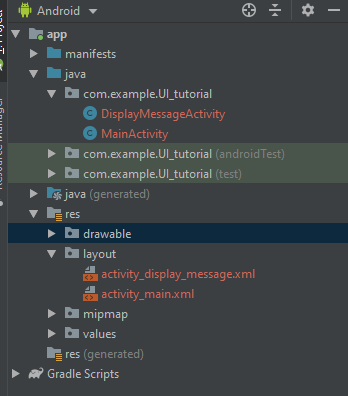


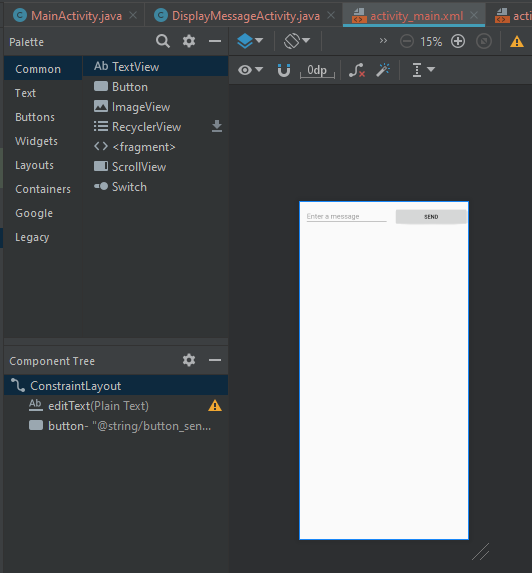
# Harjoitus 4 ja 5

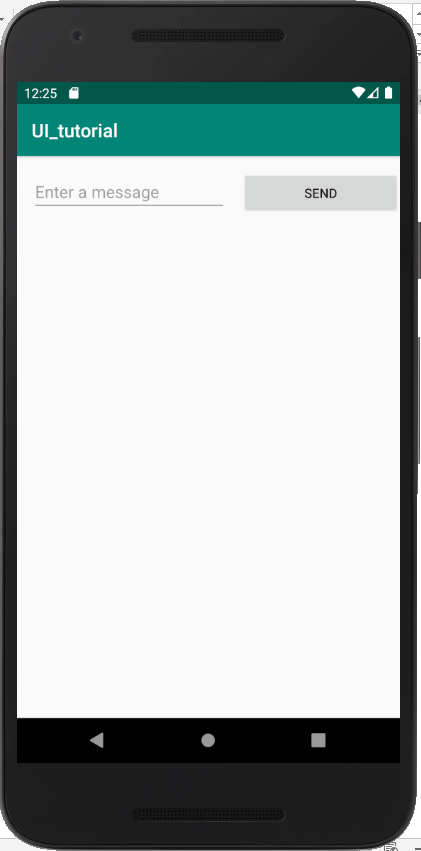
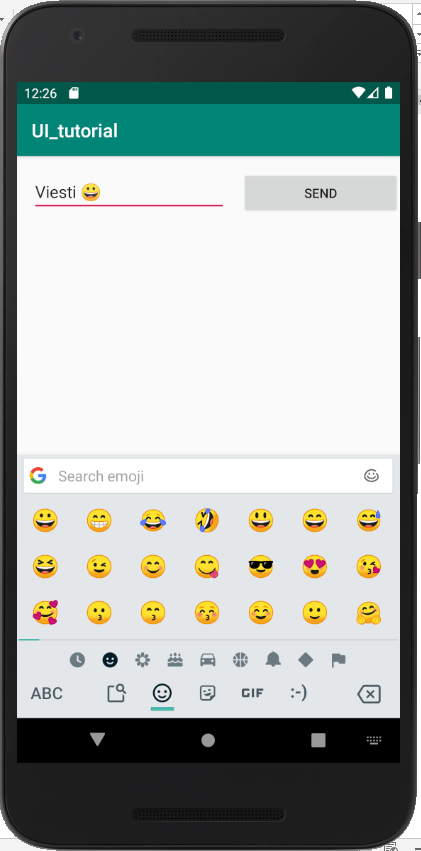
## Tutoriaaliharjoitus

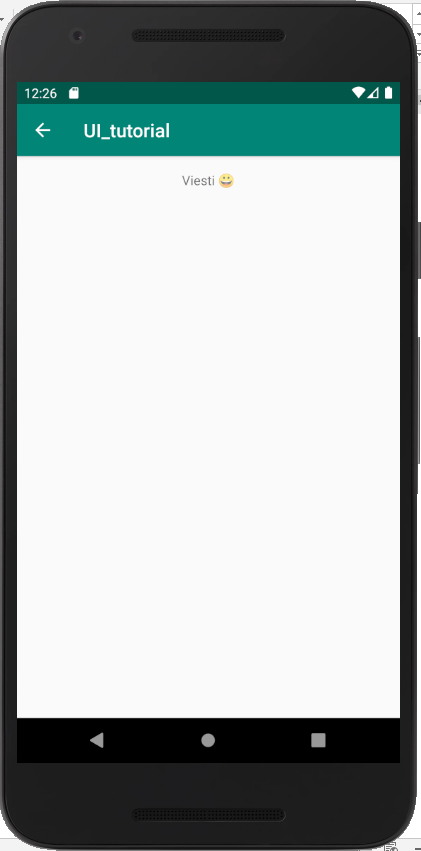
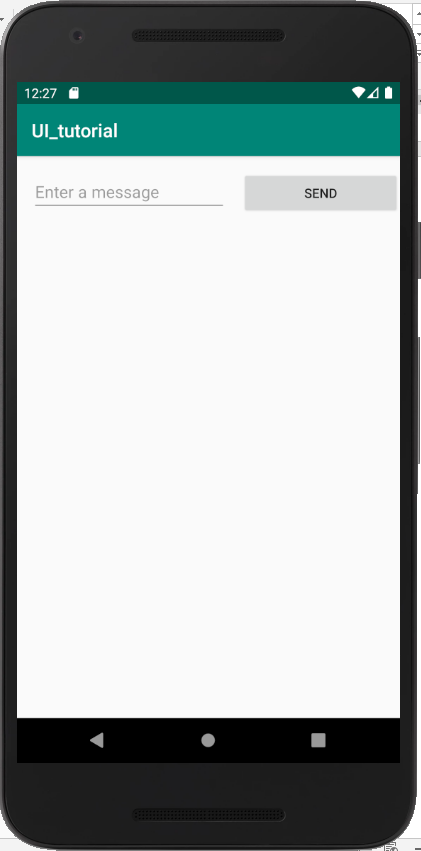
Tein tutoriaaliohjeen mukaan ”http://developer.android.com/training/basics/firstapp/building-ui.html” uuden projektin nimellä ”UI\_tutorial”.







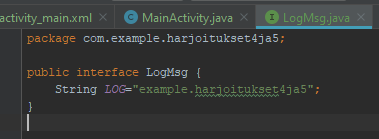
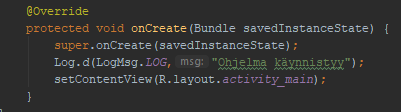
 

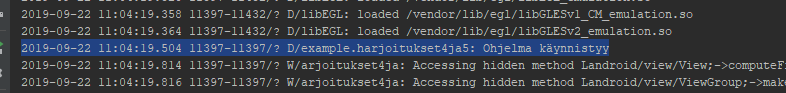
 

Toimii!

## Harjoitus 4

Loin uuden projektin nimellä ”harjoitukset4ja5” ja lisäsin heti siihen Interface ”LogMsg”. Myös otin LogMsg luokka käyttöön sovelluksessa. Nyt tieto näkyy LogCat -ikkunassa.



# Harjoitus 6, 7 ja 8 - Tietokanta ja käyttöliittymä

# Harjoitus 9 ja 10 – Firebase

# Harjoitus 11 – Sensorit

# Harjoitukset 12-14

# Harjoitus 15

# Lähteet

1. Tampereen yliopiston tekniikan alan opinnäytteiden kirjoitusohje. Tampereen yliopisto, Tampere, 2018. Saatavissa: POP > Opiskelu > Diplomityö > Diplomityöohje
2. C. Heinz, B. Moses, J. Hoffmann, Listings – Typeset source code listings using LaTeX, Comprehensive TeX Archive Network (CTAN), 2006. Saatavissa: http://www.ctan.org/pkg/listings
3. T. Oetiker, H. Partl, I. Hyna, E. Schlegl, The Not So Short Introduction to LATEX2ε – Or LATEX2ε in 157 minutes, Version 5.01, 2011, 171 p. Saatavissa: <http://www.ctan.org/tex-archive/info/lshort/english/>