

React Native



INSTALASI REACT NATIVE

React Native

react native adalah sebuah *framework* dari Javascript yang banyak dikembangkan oleh perusahaan besar dalam bidang teknologi seperti Facebook. Jadi, dengan *react native* kita tidak perlu membuat aplikasi *hybrid*. Selain itu, *react native* mampu mengkompilasi aplikasi ke dalam sebuah *native code* pada Android maupun iOS.



Keunggulan *react native*

Beberapa alasan mengapa kita harus menggunakan *react native* adalah kemudahan yang diberikan dan sudah memiliki reputasi yang terbaik. Maka sebagai bahan pertimbangan mengapa kita harus menggunakannya adalah sebagai berikut:

- *React native* digunakan oleh beberapa perusahaan besar di dunia. Sebut saja mulai dari perusahaan sekelas Tesla, Walmart, hingga Instagram. Selain itu, Facebook juga menggunakannya begitu pula dengan Airbnb, Baidu, dan Gyroscope.
- *React native* memberikan kemudahan bagi para pemula, khususnya bagi kita yang belum begitu mahir soal *coding*. Kita bisa melihat tutorial yang ada dan tutorial ini akan dibawakan dengan gaya yang mudah dipahami baik itu oleh *developer* dan beberapa praktisi lain di bidangnya.
- Dalam proses pembuatan, aplikasi *react native* memiliki keunggulan, yakni lebih cepat dan lebih efisien di dalam pengeluaran. Di sini akan ada banyak *template* dengan cara *copy paste* dari kode tersebut dan kamu langsung bisa melihat hasilnya.
- Keunggulan *react native* adalah juga sangat cocok digunakan untuk *startup*. Hal ini karena dibangun dengan [Javascript](#) sehingga sangat mudah ketika diprogram. Selain itu, dalam proses pembuatannya juga bisa *cross platform*.

Instalasi React Native

1. Untuk tahap penginstalannya bisa kita lihat di laman <https://reactnative.dev/docs/environment-setup>.
2. Pertama sebelum kita lakukan tahap penginstalan react native, kita direkomendasikan oleh laman web untuk menginstal Node dan Python2 melalui Chocolatey, pengelola paket populer untuk Windows. React Native juga membutuhkan Java SE Development Kit (JDK) serta Python2. Keduanya dapat diinstal menggunakan Chocolatey. Caranya sudah tertera di laman Chocolatey sendiri. Kita akan menyalin code yang tersedia pada laman tersebut kemudian jalankan pada windows powershell. Jika tidak ada error pada instalasi kita bisa ketikkan “choco” atau “choco -?” untuk melihat apakah sudah terinstall dan juga versinya akan tertera.

Chocolatey Install:

Individual **Organization**

1. First, ensure that you are using an [administrative shell](#) - you can also install as a non-admin, check out [Non-Administrative Installation](#).
2. Install with powershell.exe

NOTE: Please inspect <https://chocolatey.org/install.ps1> prior to running any of these scripts to ensure safety. We already know it's safe, but you should verify the security and contents of *any* script from the Internet you are not familiar with. All of these scripts download a remote PowerShell script and execute it on your machine. We take security very seriously. [Learn more about our security protocols](#).

With PowerShell, you must ensure [Get-ExecutionPolicy](#) is not Restricted. We suggest using [Bypass](#) to bypass the policy to get things installed or [AllSigned](#) for quite a bit more security.

- o Run [Get-ExecutionPolicy](#). If it returns [Restricted](#), then run [Set-ExecutionPolicy AllSigned](#) or [Set-ExecutionPolicy Bypass -Scope Process](#).

Now run the following command:

```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]
```

3. Paste the copied text into your shell and press Enter.
4. Wait a few seconds for the command to complete.
5. If you don't see any errors, you are ready to use Chocolatey! Type [choco](#) or [choco -?](#) now, or see [Getting Started](#) for usage instructions.

Chocolatey.org uses cookies to enhance the user experience of the site. [I accept](#)

```
Select Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS C:\WINDOWS\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex
((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
Getting latest version of the Chocolatey package for download.
Getting Chocolatey from https://chocolatey.org/api/v2/package/chocolatey/0.10.15.
Downloading 7-Zip commandline tool prior to extraction.
Extracting C:\Users\WMAHYU-1\AppData\Local\Temp\chocolatey\chocInstall\chocolatey.zip to C:\Users\WMAHYU-1\AppData\Local\Temp\chocolatey\chocInstall...
Installing chocolatey on this machine
Creating ChocolateyInstall as an environment variable (targeting 'Machine')
  Setting ChocolateyInstall to 'C:\ProgramData\chocolatey'
WARNING: It's very likely you will need to close and reopen your shell
before you can use choco.
Restricting write permissions to Administrators
We are setting up the Chocolatey package repository.
The packages themselves go to 'C:\ProgramData\chocolatey\lib'
(i.e. C:\ProgramData\chocolatey\lib\yourPackageName).
A shim file for the command line goes to 'C:\ProgramData\chocolatey\bin'
and points to an executable in 'C:\ProgramData\chocolatey\lib\yourPackageName'.

Creating Chocolatey folders if they do not already exist.

WARNING: You can safely ignore errors related to missing log files when
upgrading from a version of Chocolatey less than 0.9.9.
'Batch file could not be found' is also safe to ignore.
'The system cannot find the file specified' - also safe.
chocolatey.nupkg file not installed in lib.
Attempting to locate it from bootstrapper.
PATH environment variable does not have C:\ProgramData\chocolatey\bin in it. Adding...
WARNING: Not setting tab completion: Profile file does not exist at 'C:\Users\WMAHYU-1\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1'.
Chocolatey (choco.exe) is now ready.
You can call choco from anywhere, command line or powershell by typing choco.
Run choco /? for a list of functions.
You may need to shut down and restart powershell and/or consoles
first prior to using choco.
Ensuring chocolatey commands are on the path
Ensuring chocolatey.nupkg is in the lib folder
PS C:\WINDOWS\system32> choco
Chocolatey v0.10.15
Please run 'choco -?' or 'choco <command> -?' for help menu.
PS C:\WINDOWS\system32>
```

3. Setelah itu, lanjut untuk penginstalan Android Studio. Saat berada di wizard penginstalan Android Studio, pastikan kotak di samping semua item berikut dicentang:

- Android SDK
- Android SDK Platform
- Android Virtual Device
- Jika Anda belum menggunakan Hyper-V: Performance (Intel ® HAXM)(Lihat di sini untuk AMD atau Hyper-V)

Kemudian, klik "Berikutnya" untuk menginstal semua komponen ini. Jika kotak centang berwarna abu-abu, kita akan memiliki kesempatan untuk menginstal komponen ini nanti. Setelah penyiapan selesai, kita akan disajikan dengan layar Selamat Datang, lanjutkan ke langkah berikutnya.

4. Langkah berikutnya adalah install Android SDK. Android Studio menginstal Android SDK terbaru secara default. Membangun aplikasi React Native dengan kode asli, bagaimanapun, membutuhkan Android 10 (Q)SDK secara khusus. Android SDK tambahan dapat diinstal melalui SDK Manager di Android Studio.

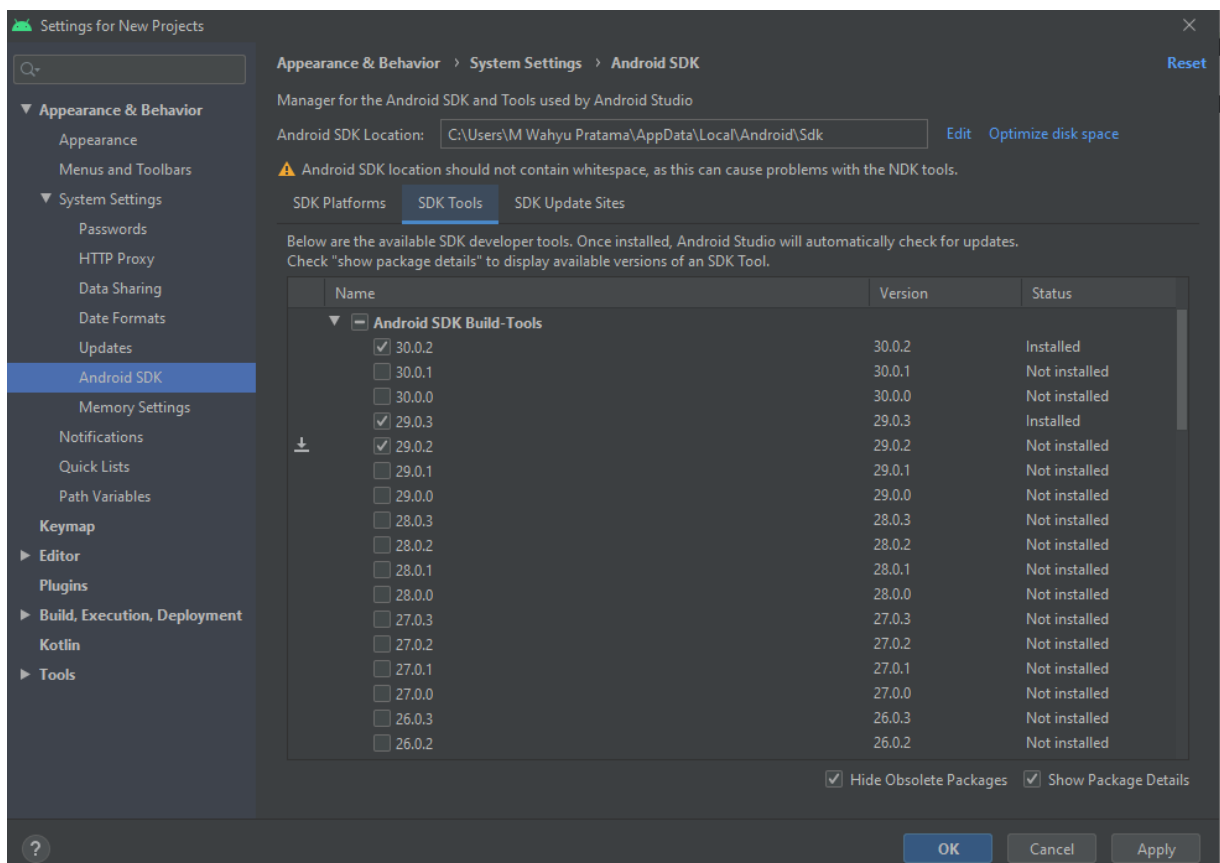
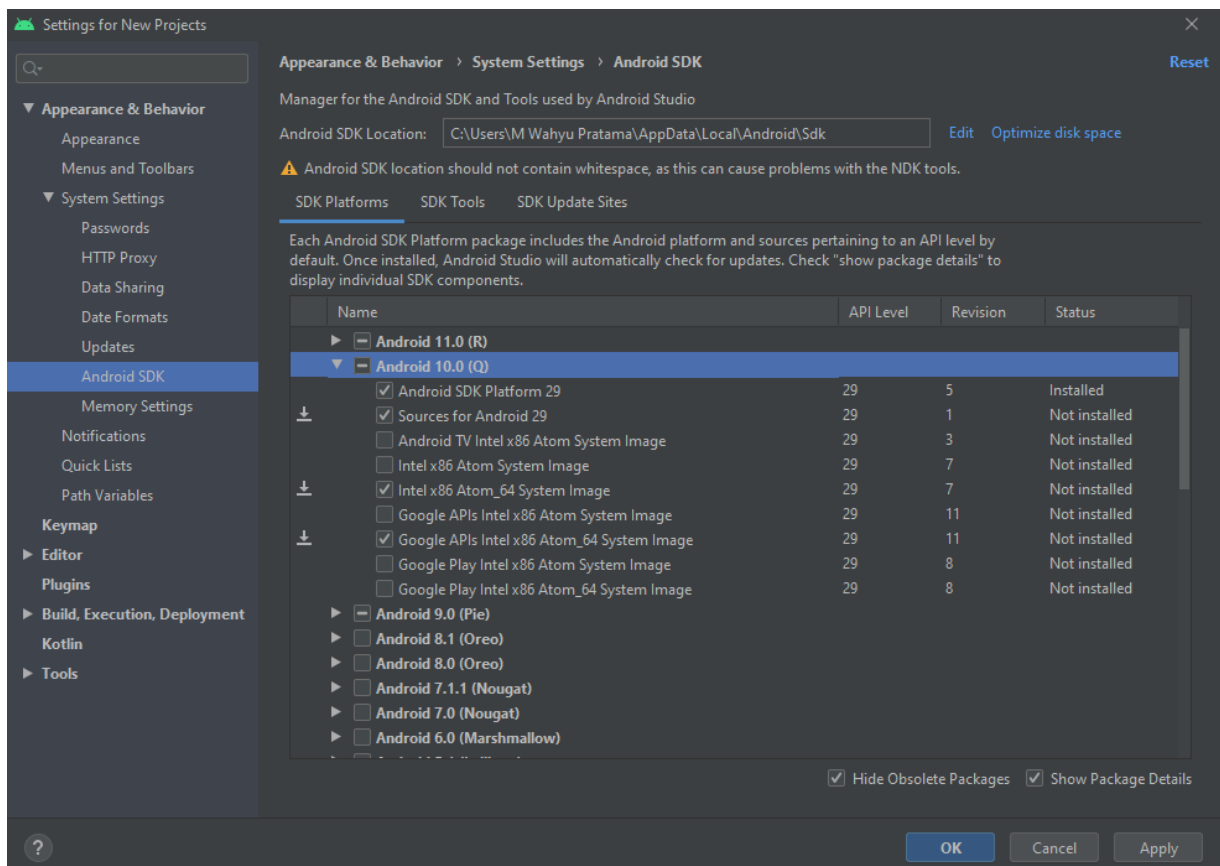
Untuk melakukannya, buka Android Studio, klik tombol "Configure" dan pilih "SDK Manager".

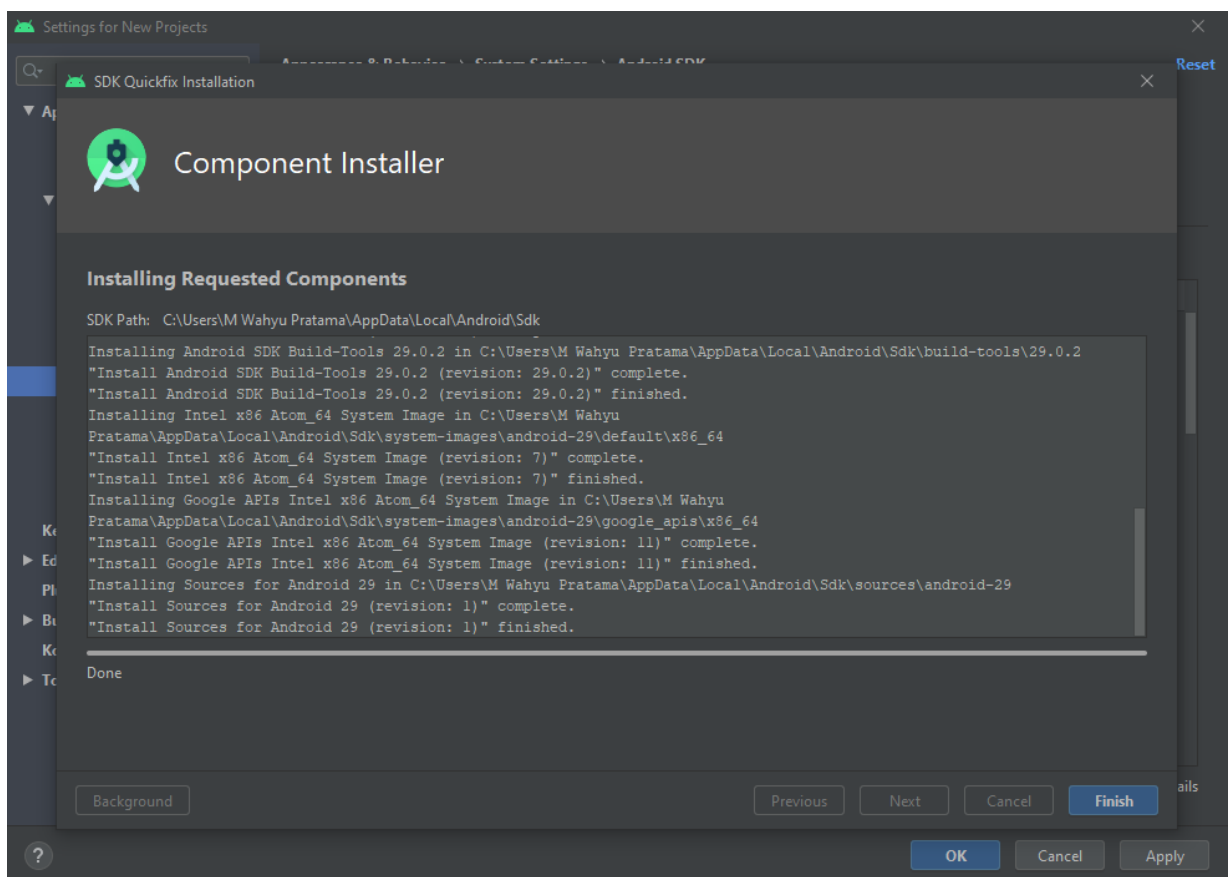
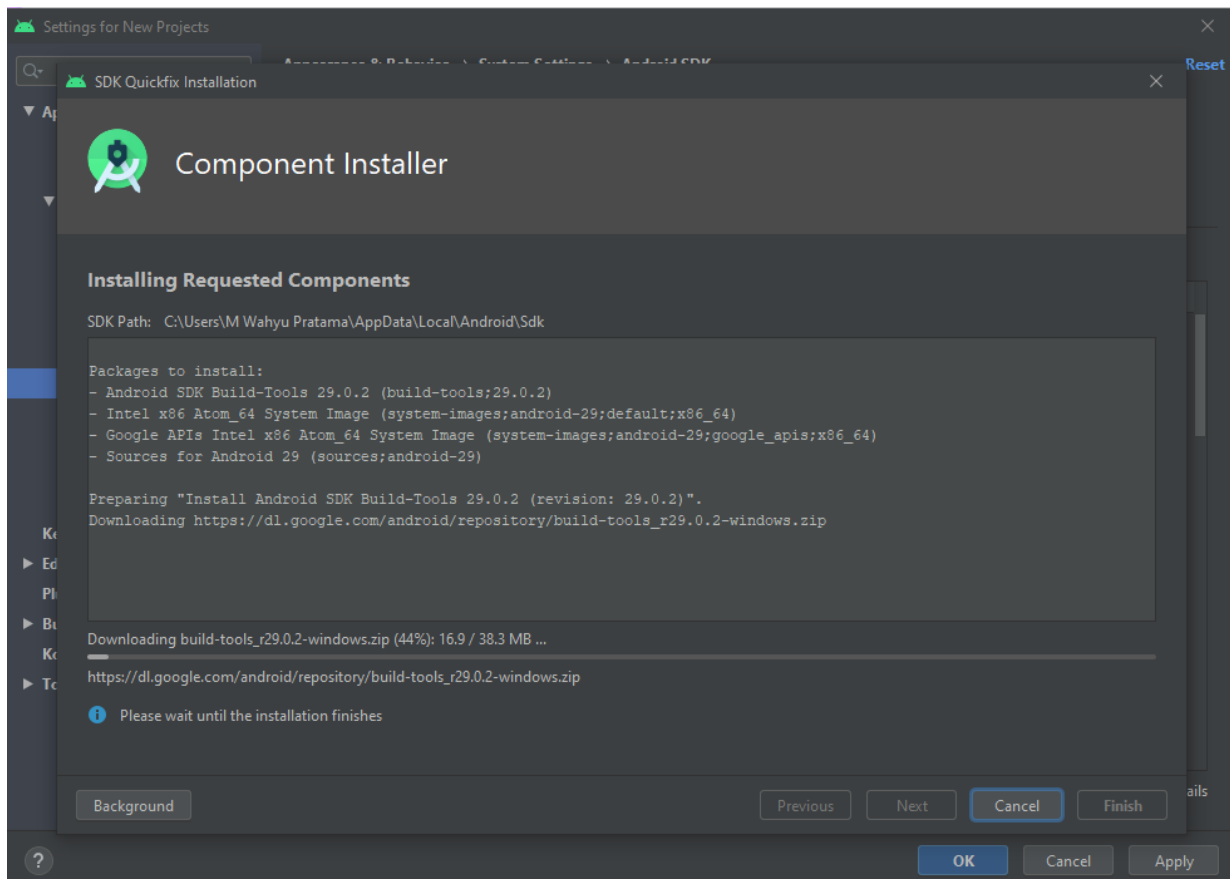
Pilih tab "Platform SDK" dari dalam SDK Manager, lalu centang kotak di samping "Tampilkan Detail Paket" di pojok kanan bawah. Cari dan luaskan Android 10 (Q)entri, lalu pastikan item berikut dicentang:

- Android SDK Platform 29
- Intel x86 Atom_64 System Image atau Google APIs Intel x86 Atom System Image

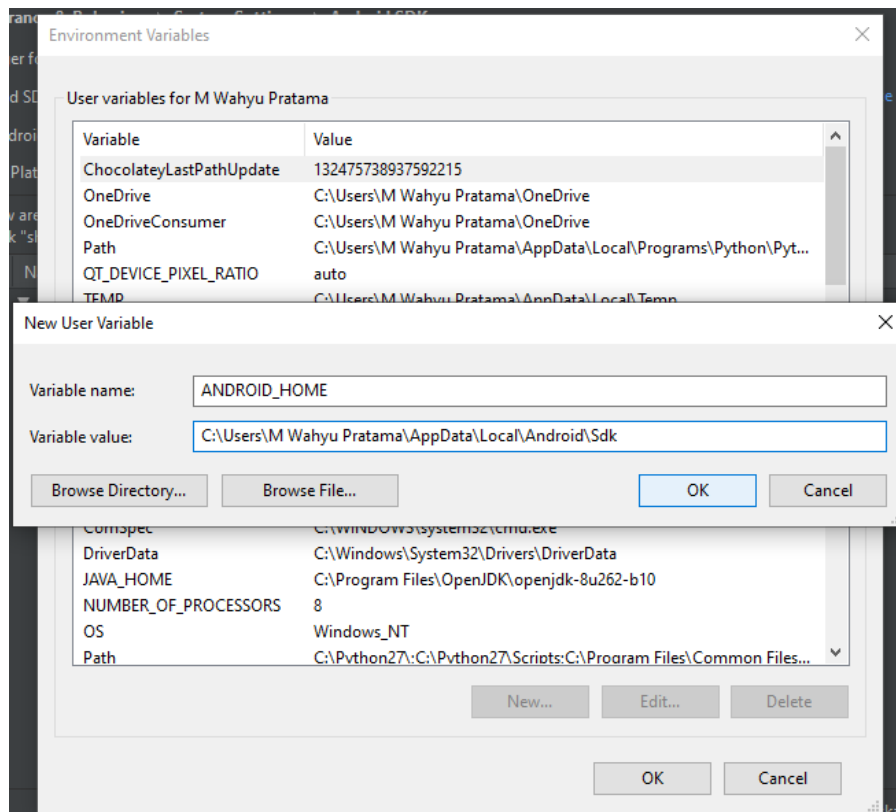
Selanjutnya, pilih tab "Alat SDK" dan centang kotak di samping "Tampilkan Detail Paket" di sini juga. Cari dan luaskan entri "Android SDK Build-Tools", lalu pastikan 29.0.2sudah dipilih.

Terakhir, klik "Terapkan" untuk mengunduh dan memasang Android SDK dan alat pembuat terkait.





5. Setelah itu, Langkah Selanjutnya adalah Konfigurasi variabel lingkungan “ANDROID_HOME”. Alat React Native memerlukan beberapa variabel lingkungan untuk disiapkan untuk membangun aplikasi dengan kode asli.
- Buka Panel Kontrol Windows.
 - Klik Akun Pengguna, lalu klik Akun Pengguna lagi
 - Klik Ubah variabel lingkungan saya
 - Klik New ... untuk membuat ANDROID_HOME variabel pengguna baru yang mengarah ke jalur ke SDK Android kita.



Setelah itu, Buka jendela Command Prompt baru untuk memastikan variabel lingkungan baru dimuat sebelum melanjutkan ke langkah berikutnya.

- Buka PowerShell.
- Salin dan tempel Get-ChildItem -Path Env: \ ke PowerShell
- Verifikasi ANDROID_HOME telah ditambahkan.

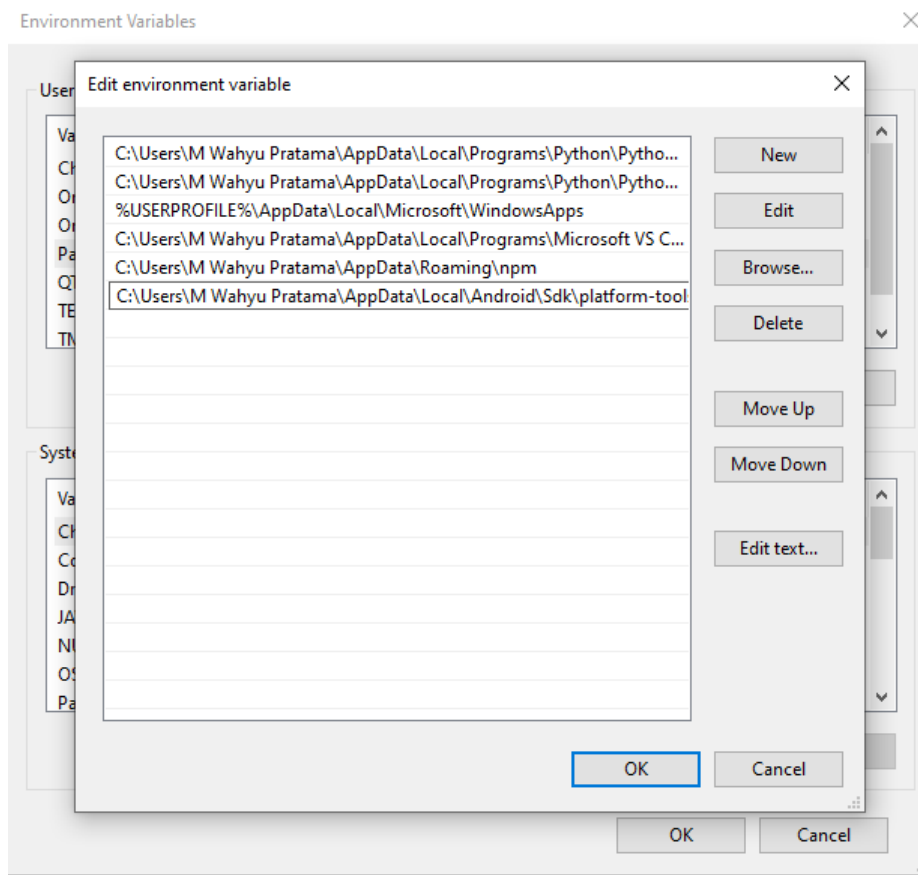
```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> Get-ChildItem -Path "C:\Users\M Wahyu Pratama\AppData\Local\Android\Sdk"

Directory: C:\Users\M Wahyu Pratama\AppData\Local\Android\Sdk

Mode                LastWriteTime         Length Name
----                -
d-----         10/19/2020    5:18 PM             .downloadIntermediates
d-----         10/19/2020    5:20 PM             .temp
d-----         10/19/2020    5:20 PM          build-tools
d-----         10/2/2020    10:05 AM          emulator
d-----         10/8/2020    3:25 PM           extras
d-----         10/9/2020    3:03 PM           fonts
d-----         10/19/2020    5:22 PM          licenses
d-----          9/23/2020   11:21 AM          patcher
d-----          9/23/2020   11:21 AM    platform-tools
d-----         10/17/2020   12:23 PM          platforms
d-----          9/23/2020   11:49 AM           skins
d-----         10/19/2020    5:20 PM          sources
d-----         10/19/2020    4:49 PM    system-images
d-----          9/23/2020   11:23 AM           tools
-a-----         10/19/2020    5:22 PM           16 .knownPackages

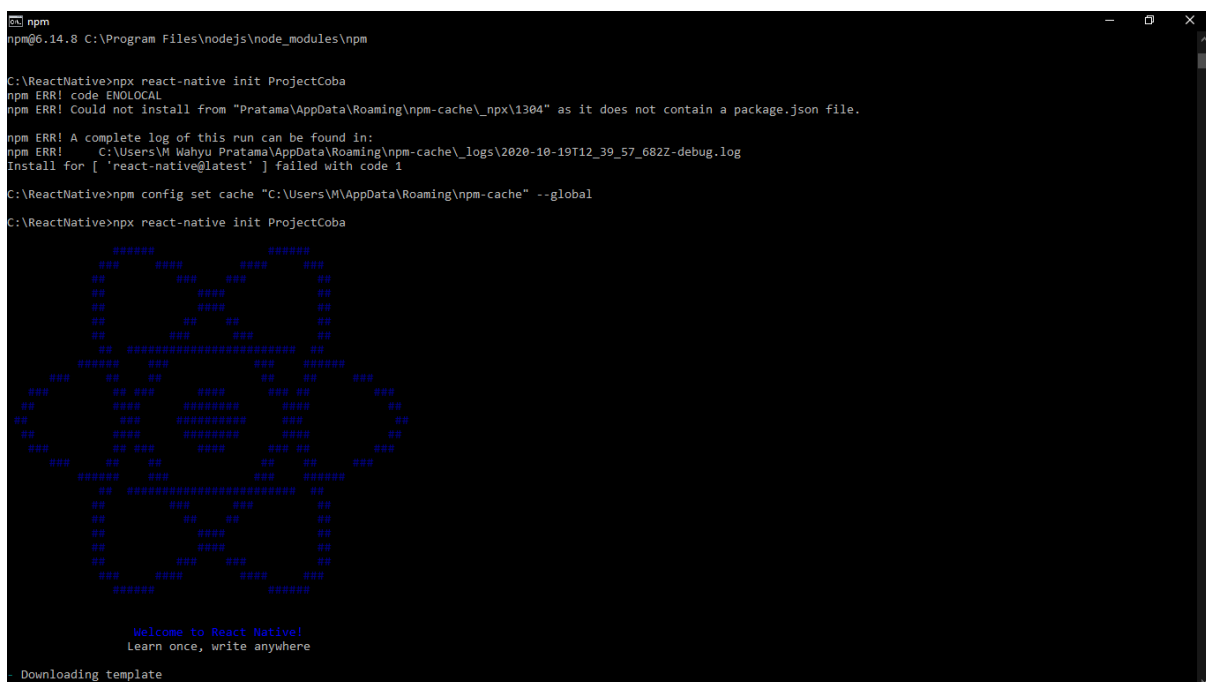
PS C:\WINDOWS\system32>
```

6. Langkah terakhir adalah tambahkan alat platform ke Path dengan cara :
- Buka Panel Kontrol Windows.
 - Klik Akun Pengguna, lalu klik Akun Pengguna lagi
 - Klik Ubah variabel lingkungan saya
 - Pilih variabel Path .
 - Klik Edit.
 - Klik Baru dan tambahkan jalur ke alat platform ke daftar.



Penggunaan React Native Dengan Menggunakan Emulator Android Studio.

1. Setelah semua tahap instalasi selesai, kita akan membuat sebuah project baru react native dengan menggunakan emulator android. Namun, sebelum itu kita harus membuat folder baru terlebih dahulu di partisi atau tempat file yang di inginkan. Kemudian kita buka prompt perintah dengan hak akses administrator lalu pindah ke partisi atau folder tempat kita buat file untuk project baru tersebut kemudian ketik code ini “ `npx react-native init namafolder`”. Jika terjadi error pada saat pembuatannya, silahkan lihat solusi permasalahannya di link <https://stackoverflow.com/questions/48143945/what-is-the-err-code-enolocal-npm-err-could-not-install-from-error> dan sesuaikan dengan error yang ada.



```
npm
npm@6.14.8 C:\Program Files\nodejs\node_modules\npm

C:\ReactNative>npx react-native init ProjectCoba
npm ERR! code ENOLocal
npm ERR! Could not install from "Pratama\AppData\Roaming\npm-cache\npx\1304" as it does not contain a package.json file.

npm ERR! A complete log of this run can be found in:
npm ERR! C:\Users\Wahyu\Pratama\AppData\Roaming\npm-cache\logs\2020-10-19T12_39_57_682Z-debug.log
Install for [ 'react-native@latest' ] failed with code 1

C:\ReactNative>npm config set cache "C:\Users\Wahyu\Pratama\AppData\Roaming\npm-cache" --global

C:\ReactNative>npx react-native init ProjectCoba

  Welcome to React Native!
  Learn once, write anywhere

Downloading template
```

2. Setelah tahap tersebut selesai, mari kita buka teks editor Microsoft Visual Studio Code (Jika belum ada bisa di install terlebih dahulu dengan mendownload file installernya di <https://code.visualstudio.com/download>) kemudian buka folder project kita tadi dan juga install ekstensi untuk support react native tersebut seperti React-Native, NodeJs, dan beautify.

- Setelah di install dan buka folder projectnya, cari pada bagian “npm scripts” lalu “android” kemudian akan keluar codenya. Setelah itu tekan tombol start untuk menjalankan react native dalam emulator android dan tunggu hingga prosesnya selesai hingga terlihat dashboard utama react native di emulator android studio.

The screenshot displays the Visual Studio Code interface with the following components:

- Menu Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Explorer Panel:** Shows the file structure with folders for OPEN EDITORS, PROJECTCOPA, OUTLINE, and NPM SCRIPTS. The `package.json` file is selected under NPM SCRIPTS, and the `android` script is highlighted.
- Editor Panel:** Displays the content of `package.json` with the following JSON structure:

```
{
  "name": "ProjectCoba",
  "version": "0.0.1",
  "private": true,
  "scripts": {
    "android": "react-native run-android",
    "ios": "react-native run-ios",
    "start": "react-native start",
    "test": "jest",
    "lint": "eslint ."
  },
  "dependencies": {
    "react": "16.13.1",
    "react-native": "0.63.3"
  },
  "devDependencies": {
    "@babel/core": "^7.8.4",
    "@babel/runtime": "^7.8.4",
    "@react-native-community/eslint-config": "^1.1.0",
    "babel-jest": "^25.1.0",
  }
}
```
- Terminal Panel:** Shows the output of the `npm install` command, indicating that the React Native Packager is running. The terminal text includes:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\ReactNative\ProjectCoba>
```

[illegible]

Dan hasil akhirnya seperti ini.

```
npm
g strip tool for ABI 'X86_64'. Packaging it as is.
Unable to strip library 'C:\ReactNative\ProjectCoba\android\app\build\intermediates\merged_native_libs\debug\out\lib\x86_64\libhermes-inspector.so' due to missing strip
tool for ABI 'X86_64'. Packaging it as is.
Unable to strip library 'C:\ReactNative\ProjectCoba\android\app\build\intermediates\merged_native_libs\
for ABI 'X86_64'. Packaging it as is.
Unable to strip library 'C:\ReactNative\ProjectCoba\android\app\build\intermediates\merged_native_libs\
ool for ABI 'X86_64'. Packaging it as is.
Unable to strip library 'C:\ReactNative\ProjectCoba\android\app\build\intermediates\merged_native_libs\
tool for ABI 'X86_64'. Packaging it as is.
Unable to strip library 'C:\ReactNative\ProjectCoba\android\app\build\intermediates\merged_native_libs\
ool for ABI 'X86_64'. Packaging it as is.
Unable to strip library 'C:\ReactNative\ProjectCoba\android\app\build\intermediates\merged_native_libs\
I 'X86_64'. Packaging it as is.

> Task :app:installDebug
08:32:12 V/ddms: execute: running am get-conf
08:32:12 V/ddms: execute: 'am get-conf' on 'emulator-5554' : EOF hit. Read: -1
08:32:12 V/ddms: execute: returning
Installing APK 'app-debug.apk' on 'Pixel 3a_API 30_X86(AVD) - 11' for app-debug
08:32:12 D/app-debug.apk: Uploading app-debug.apk onto device 'emulator-5554'
08:32:12 D/Device: Uploading file onto device 'emulator-5554'
08:32:12 D/ddms: Reading file permission of C:\ReactNative\ProjectCoba\android\app\build\outputs\apk\deb
08:32:12 V/ddms: execute: running pm install -r -t "/data/local/tmp/app-debug.apk"
08:32:15 V/ddms: execute: 'pm install -r -t "/data/local/tmp/app-debug.apk"' on 'emulator-5554' : EOF hi
08:32:15 V/ddms: execute: returning
08:32:15 V/ddms: execute: running rm "/data/local/tmp/app-debug.apk"
08:32:15 V/ddms: execute: 'rm "/data/local/tmp/app-debug.apk"' on 'emulator-5554' : EOF hit. Read: -1
08:32:15 V/ddms: execute: returning
Installed on 1 device.

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use "--warning-mode all" to show the individual deprecation warnings.
See https://docs.gradle.org/6.2/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 4m 51s
27 actionable tasks: 27 executed
Info Connecting to the development server...
8081
Info Starting the app...
Starting: Intent { cmp=com.projectcoba.MainActivity }
```