

Dezvoltare Android pe platforma Windows, fara Android Studio

- Dezvoltare Android pe platforma Windows, fara Android Studio
- Instalarea JAVA DEVELOPMENT KIT
 - Descarcarea Java SE
 - Instalarea Java SE
 - Setarea PATH-ului
 - Verificarea existentei comenzilor
- Instalarea Android Command Line Tools
 - Descarcati Android Command-Line Tools
 - Instalarea Android Command-Line Tools
 - Adaugati directorul "bin" al cmdline-tools la PATH
 - Adaugati variabile la environment: ANDROID_HOME si ANDROID_SDK_ROOT
 - Instalarea restului instrumentelor
 - Care sunt restul instrumentelor
 - Versiuni Android
 - Instalarea propriu-zisa
- Instalare gradle
 - Adaugati calea in PATH
- Verificati disponibilitatea comenzilor
 - adb
 - gradle
 - avdmanager
 - emulator
 - Verificati toate componentele folosind sdkmanager
- Emulator
 - Creati un avd
 - Porniti emulatorul
 - Lista de dispozitive existente
 - Instalati aplicatia pe dispozitiv
 - Verificati existenta aplicatiei (powershell)
- Instalari pariale
 - Platform tools
- Acces telefon hardware
 - Pregatirea dispozitivului
 - Activarea optiunilor de dezvoltare
 - Cateva comenzi adb
 - adb shell pm list packages
 - adb logcat
 - screenshot
 - screen recording
 - adb shell getprop
 - adb shell settings put

- Alte comenzi
- Conectarea unui dispozitiv USB la o masina virtuala
- Alte instrumente utile
 - git
 - Windows terminal
 - Descarcarea directa din repository
 - Alte variante
 - Scrcpy
 - Instalare
 - Utilizare

Instalrea JAVA DEVELOPMENT KIT

JAVA de obicei este distribuită in două feluri: JRE (Java Runtime Edition) si JDK (Java Development Kit). Pentru orice dezvoltare avem nevoie de JDK. JDK poate fi descarat fie de la OpenJDK, fie direct de la Oracle de pe pagina, ca parte din distributia numita "Java Standard Edition" - Java SE. Vom lucra cu Java SE 21 (cea mai recenta versiune este 23).

Descarcarea Java SE

<https://www.oracle.com/java/technologies/downloads/#jdk21-windows>

Java 23, Java 21, and earlier versions available now

JDK 23 is the latest release of the Java SE Platform.

JDK 21 is the latest *Long-Term Support (LTS)* release of the Java SE Platform.

Earlier JDK versions are available below.

[Learn about Java SE Subscription](#)

[JDK 23](#) [JDK 21](#) [GraalVM for JDK 23](#) [GraalVM for JDK 21](#)

JDK Development Kit 23.0.1 downloads

JDK 23 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions \(NFTC\)](#).

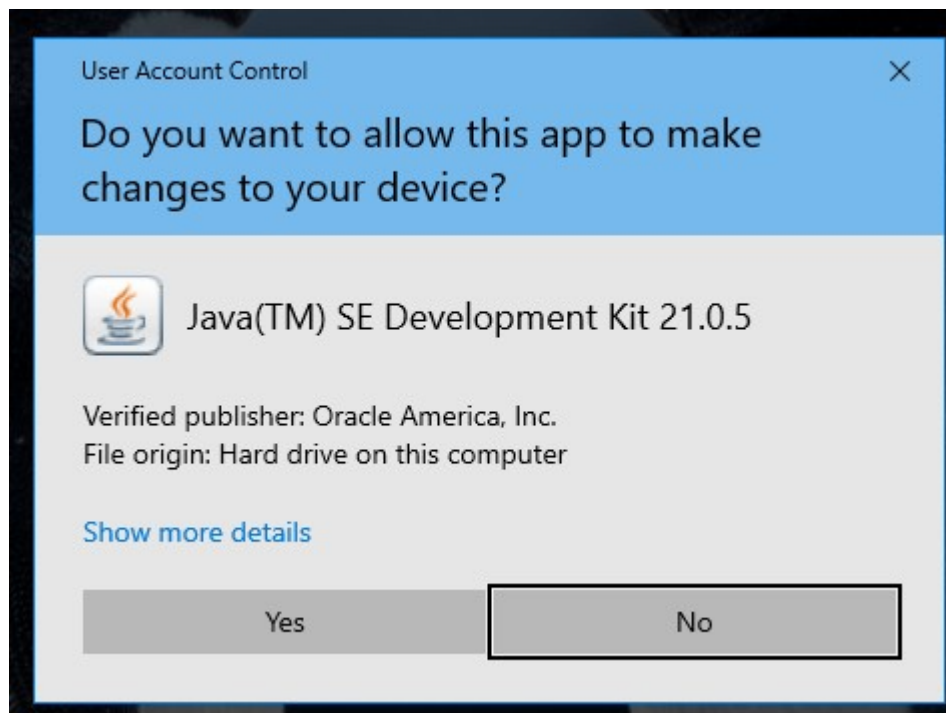
JDK 23 will receive updates under these terms, until March 2025, when it will be superseded by JDK 24.

[Linux](#) [macOS](#) [Windows](#)

Product/file description	File size	Download
x64 Compressed Archive	228.70 MB	https://download.oracle.com/java/23/latest/jdk-23_windows-x64_bin.zip (sha256)
x64 Installer	205.21 MB	https://download.oracle.com/java/23/latest/jdk-23_windows-x64_bin.exe (sha256)
x64 MSI Installer	203.96 MB	https://download.oracle.com/java/23/latest/jdk-23_windows-x64_bin.msi (sha256)

Descarcati X64 Installer. Va trebui sa obtineti un fisier numit jdk-21_windows-x64_bin.exe. Rulati acest fisier. Aveti nevoie de drepturi de administrator pe masina respectiva.

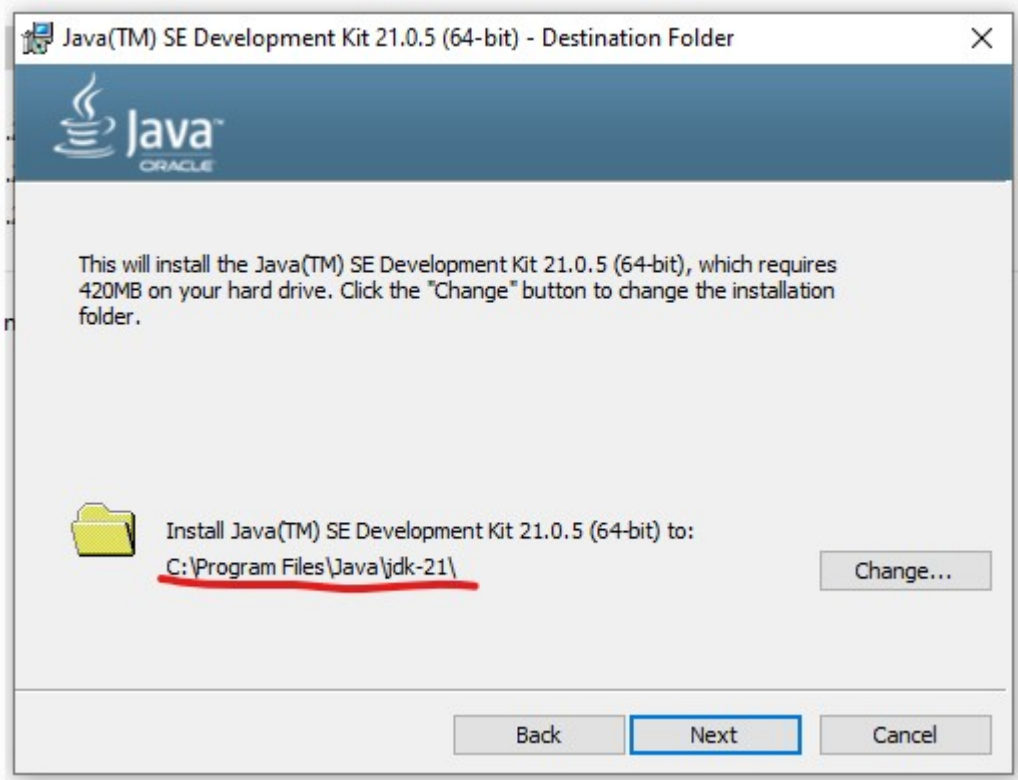
Instalarea Java SE



Urmariti pasii installerului



Retineti calea in care va fi instalata distributia si care va aparea in al doilea ecran al installerului

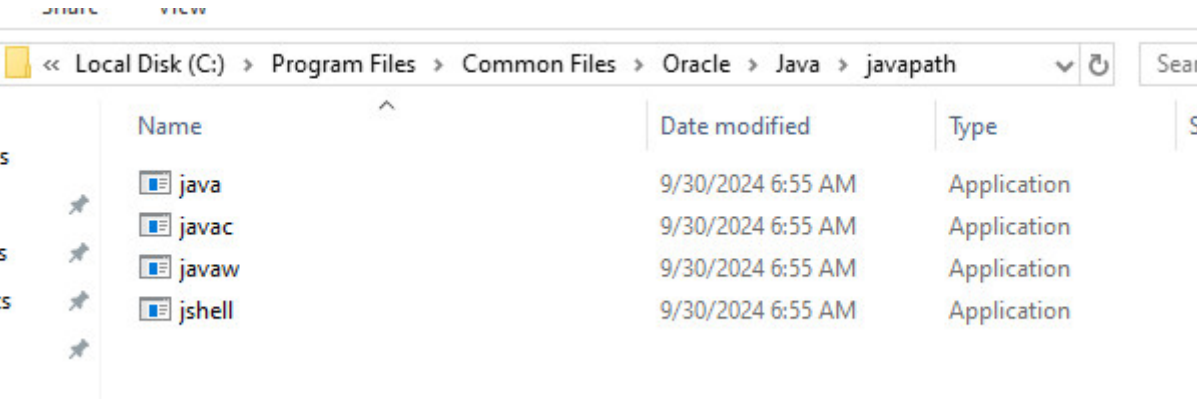


Dupa finalizarea instalarii, comanda `$env:Path - split ';'` in powershell va afisa toate componentele path-ului disponibil.

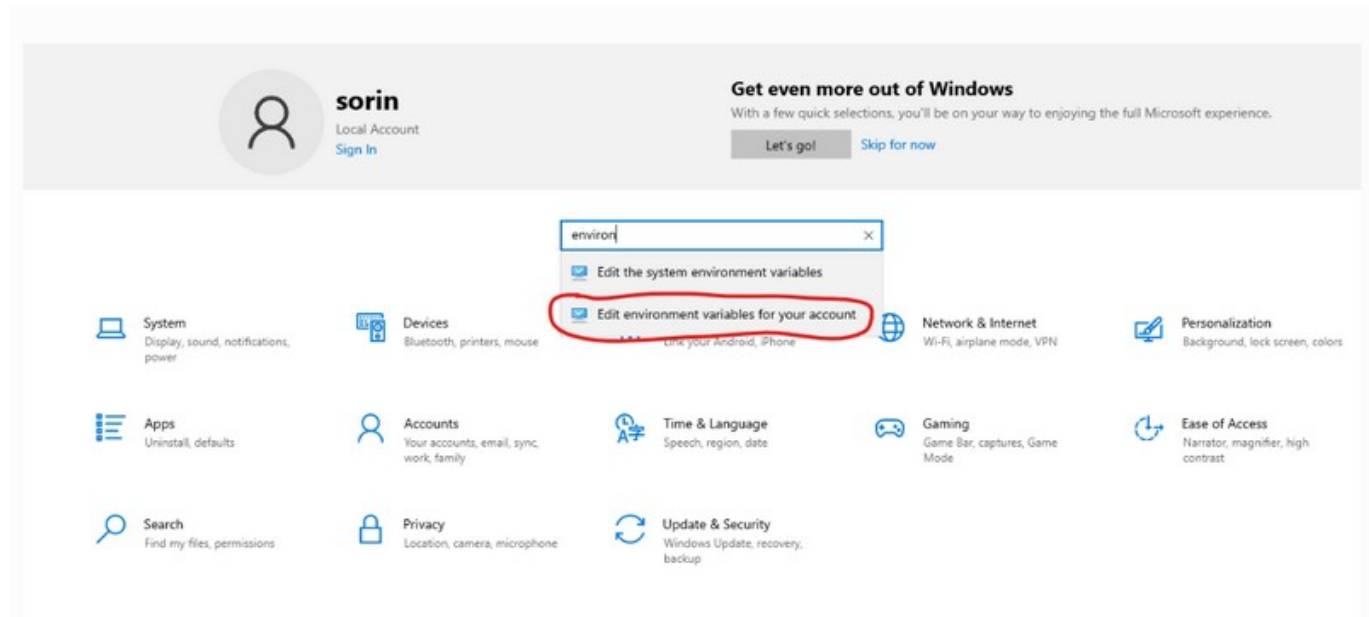
```
PS C:\Users\sorin> $env:Path -split ';'
C:\Program Files\Common Files\Oracle\Java\javapath
C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
C:\Users\sorin\AppData\Local\Microsoft\WindowsApps
```

Setarea PATH-ului

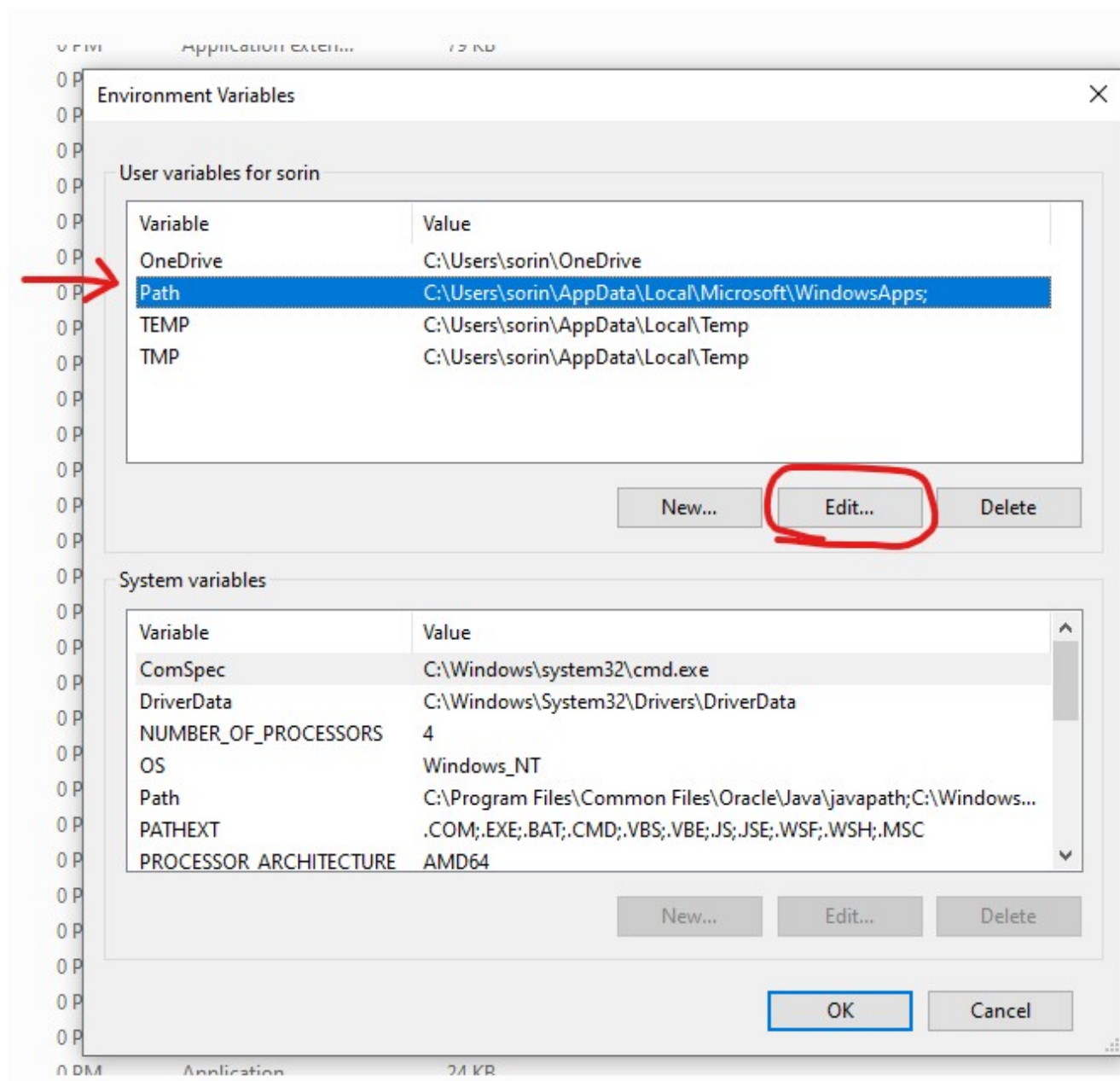
Se observa aparitia caii `C:\Program Files\Commoon Files\Oracle\Java\javapath` care asigura disponibilitatea in powershell a comenzilor java (pornirea masinii virtuale) si javac (compilarea surselor). Acestea insa sunt pozitionate intr-un director separat, care contine foarte putine fisiere:



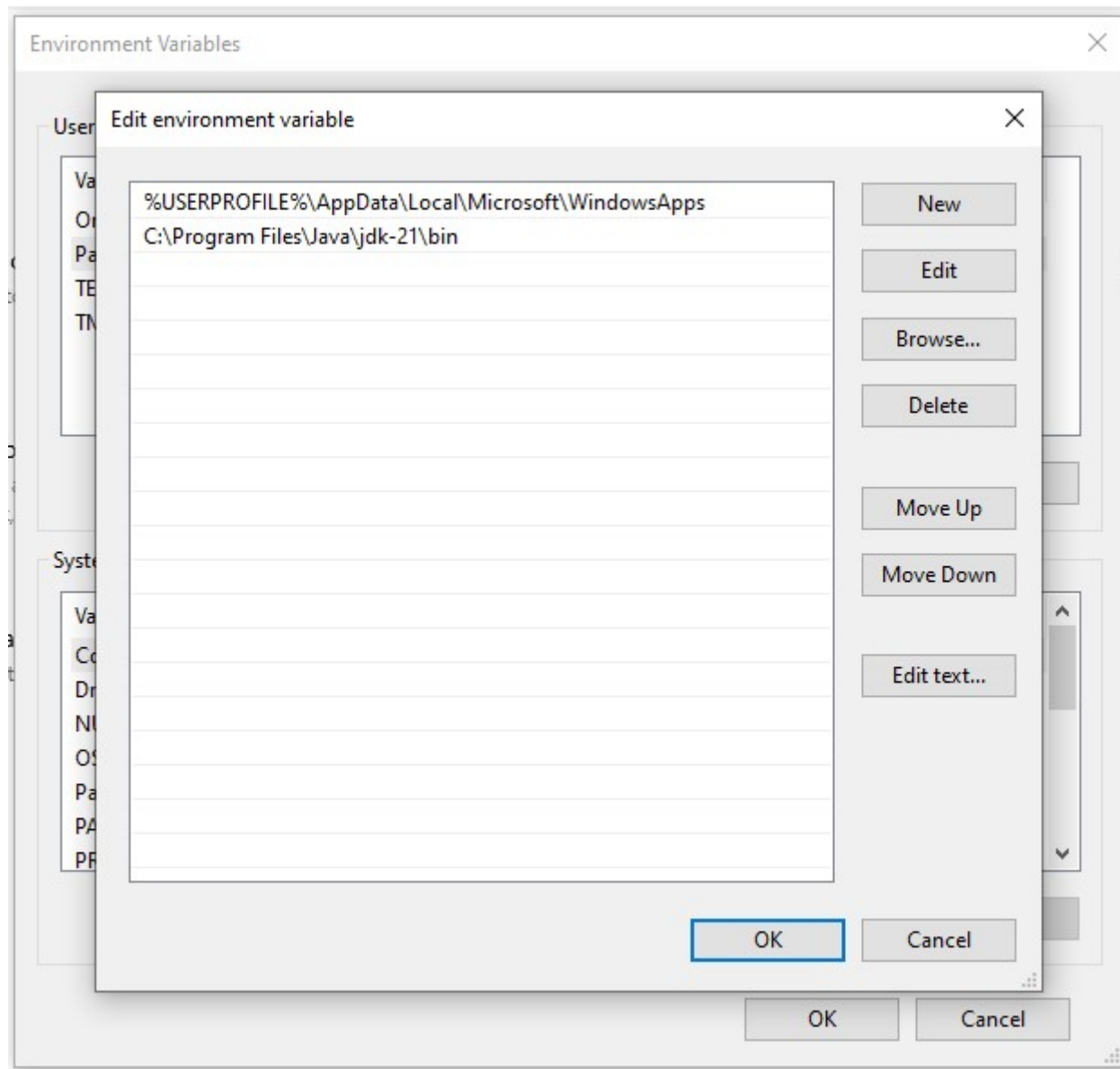
Acestea sunt suficiente pentru a rula si compila programe java dar in realitate distributia contine mult mai multe instrumente care pot fi utile (instrumente de inspectie a arhivelor jar, instrumente de semnare a acestora, instrumente de debug, etc.). Este in general o idee buna sa adaugam si directorul "bin" din distributia jdk (plasata in locul retinut anterior) la path-ul userului curent. Pentru aceasta, apelati



Alegeti variabila PATH si apasati butonul Edit



Apoi apasati butonul "New" si scrieti calea catre jdk, inclusiv directorul "bin".



Verificarea existentei comenzilor

Dupa salvarea dialogului respectiv, puteti verifica in powershell (deschizand o noua sesiune) disponibilitatea comenzilor java, javac, jar, jpackage, etc.

```
PS C:\Users\MYUSER> java
Usage: java [options] <mainclass> [args...]
        (to execute a class)
or java [options] -jar <jarfile> [args...]
        (to execute a jar file)
or java [options] -m <module>[/<mainclass>] [args...]
   java [options] --module <module>[/<mainclass>] [args...]
        (to execute the main class in a module)
or java [options] <sourcefile> [args]
        (to execute a single source-file program)
```

Arguments following the main class, source file, -jar <jarfile>, -m or --module <module>/<mainclass> are passed as the arguments to main class.

....

```
PS C:\Users\MYUSER> javac
Usage: javac <options> <source files>
where possible options include:
@<filename>           Read options and filenames from file
-Akey[=value]          Options to pass to annotation processors
--add-modules <module>(,<module>)*
    Root modules to resolve in addition to the initial modules,
    or all modules on the module path if <module> is ALL-MODULE-PATH.
--boot-class-path <path>, -bootclasspath <path>
    Override location of bootstrap class files
--class-path <path>, -classpath <path>, -cp <path>
    Specify where to find user class files and annotation processors
-d <directory>         Specify where to place generated class files
-deprecation
    Output source locations where deprecated APIs are used
--enable-preview
    Enable preview language features.
    To be used in conjunction with either -source or --release.

...
```

```
PS C:\Users\MYUSER> jar
Usage: jar [OPTION...] [ [--release VERSION] [-C dir] files] ...
Try `jar --help' for more information.
```

```
jpackage
Usage: jpackage <options>
Use jpackage --help (or -h) for a list of possible options
```

Instalarea Android Command Line Tools

Descarcati Android Command-Line Tools

<https://developer.android.com/studio#command-tools>

Navigați la adresa de mai sus și cautați în josul paginii.

Command line tools only

Platform	SDK tools package	Size	SHA-256 checksum
Windows	commandlinetools-win-11076708_latest.zip	153.6 MB	4d6931209eebb1bfb7c7e8b240a6a3cb3ab24479ea294f3539429574b1eec862
Mac	commandlinetools-mac-11076708_latest.zip	153.6 MB	7bc5c72ba0275c80a8f19684fb92793b83a6b5c94d4d179fc5988930282d7e64
Linux	commandlinetools-linux-11076708_latest.zip	153.6 MB	2d2d50857e4eb553af5a6dc3ad507a17adf43d115264b1afc116f95c92e5e258

Command-line tools are included in Android Studio. If you do not need Android Studio, you can download the basic Android command-line tools above. You can use the included [sdkmanager](#) to download other SDK packages.

Descarcati versiunea pentru windows. Vi se va solicita acceptarea unei licente.

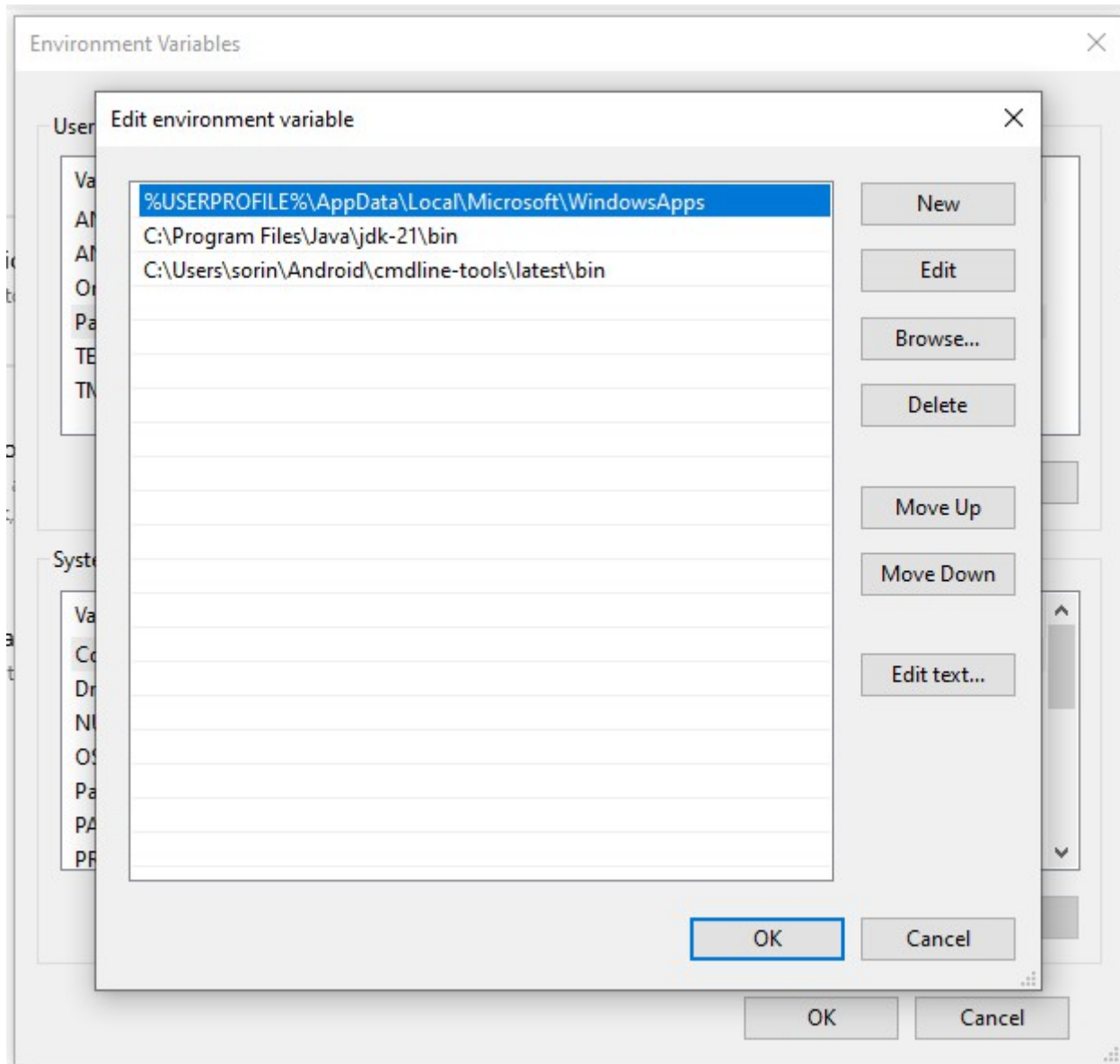
Instalarea Android Command-Line Tools

Extrageți fișierele din arhiva respectivă într-un director care va deveni spațiul principal în care veți poziționa toate instrumentele de dezvoltare. De exemplu, C:\Users\MYUSER\Android

Va rezulta un director C:\Users\MYUSER\Android\cmdline-tools care conține, printre altele, un subdirector "bin"

Mutați conținutul acestui director într-un subdirector "latest", astfel încât instrumentele să fie plasate în C:\Users\MYUSER\Android\cmdline-tools\latest

Adăugați directorul "bin" al cmdline-tools la PATH



Dupa adaugarea acestuia, comanda "sdkmanager.bat" trebuie sa devina disponibila intr-o sesiune noua de PowerShell.

```
PS C:\Users\MYUSER> sdkmanager.bat
Usage:
  sdkmanager [--uninstall] [<common args>] [--package_file=<file>] [<packages>...]
  sdkmanager --update [<common args>]
  sdkmanager --list [<common args>]
  sdkmanager --list_installed [<common args>]
  sdkmanager --licenses [<common args>]
  sdkmanager --version

...
```

Adaugati variabile la environment: ANDROID_HOME si ANDROID_SDK_ROOT

ANDROID_SDK_ROOT este varianta mai recenta a variabilei de mediu necesara in timp ce ANDROID_HOME este varianta mai veche.

Instalarea restului instrumentelor

Care sunt restul instrumentelor

Pentru dezvoltarea aplicatiilor android sunt necesare mai multe tipuri de componente, biblioteci, platforme, etc. Acestea, impreuna cu versiunile lor, pot fi vazute cu ajutorul comenzii `sdkmanager.bat --list`. Aceasta lista va afisa urmatoarele componente:

add-ons;addon-google_apis-google

Conține API-urile Google necesare pentru simularea serviciilor Google Play pe dispozitivele Android. Este folosit pentru a dezvolta aplicații care se bazează pe serviciile Google, cum ar fi autentificarea, notificările push și altele.

build-tools

Include un set de instrumente necesare pentru a construi aplicații Android, cum ar fi aapt (Android Asset Packaging Tool), dx (dexter) și zipalign. Este esențial pentru procesul de construire a aplicațiilor.

cmake

CMake este un sistem de automatizare a construcțiilor, folosit pentru a compila aplicațiile C/C++ pe platformele Android. Este necesar pentru construirea aplicațiilor native în Android.

cmdline-tools

Acesta este pachetul care conține instrumentele de linie de comandă pentru SDK-ul Android. Incluzând sdkmanager și avdmanager, permite gestionarea SDK-ului și crearea de dispozitive virtuale Android (AVD).

emulator

Pachetul Emulator conține emulatorul Android, care permite rularea aplicațiilor Android într-un mediu virtualizat. Acesta simulează un dispozitiv Android pe computerul tău, oferind o platformă pentru testare.

extras

conține pachete suplimentare care nu sunt esențiale pentru construirea aplicațiilor, dar pot oferi funcționalități adiționale, cum ar fi instrumentele de dezvoltare sau pachetele SDK pentru diverse API-uri și platforme.

ndk

NDK (Native Development Kit) este un set de instrumente care permite dezvoltarea aplicațiilor Android folosind limbaje de programare native, cum ar fi C și C++. NDK este folosit pentru aplicațiile care necesită performanță ridicată.

platforms

Acesta este pachetul care conține platformele Android necesare pentru a construi aplicații pentru diverse versiuni ale sistemului de operare Android. Fiecare platformă este asociată cu un anumit API Level.

skiaarser

Instrument care analizează fișierele .skia utilizate de Android pentru gestionarea graficii. Este folosit în mod special pentru optimizarea performanței grafice și a renderizării în aplicațiile Android.

sources

sursele sistemului de operare Android. Aceste surse sunt utile pentru dezvoltatori care vor să studieze sau să modifice codul sursă al Android pentru a înțelege mai bine funcționarea internă a platformei.

system-images

Imagini de sistem folosite pentru a crea dispozitive virtuale Android (AVD). Ele includ imagini de sistem pentru diverse versiuni ale Android și arhitecturi de procesor (armeabi-v7a, x86, etc.), permițând testarea aplicațiilor pe diferite configurații.

Pentru dezvoltarea aplicațiilor Android este nevoie de câteva dintre aceste elemente:

- build-tools
- emulator
- platform-tools
- imagini de sistem
- platforme

Evident, dacă dorim să construim o aplicație care să aibă cod nativ (c/c++) este nevoie și de alte componente, cum ar fi cmake și ndk.

Versiuni Android

Android are mai multe versiuni, începând de la Android 1 și terminând cu Android 15. Fiecare versiune are una sau mai multe API-uri corespunzătoare (subversiuni). Versiunile inițiale au de obicei mai multe niveluri API în timp ce versiunile mai recente au una sau 2. Din acest motiv, deși versiunea principală de Android cea mai recentă este 15, ea corespunde cu API Level 35. Din punct de vedere al dezvoltării, aplicațiile noastre trebuie să fie compilate pentru un anumit nivel API. Acestea de obicei nu vor rula și pe nivele inferioare, dacă dorim ca aplicația noastră să ruleze pe mai multe dispozitive este o idee bună să "tintim" un nivel de API inferior.

Un nivel API (API level) reprezintă o versiune specifică a sistemului de operare Android, care definește setul de funcționalități și caracteristici disponibile pentru dezvoltatori. Fiecare versiune a Android are un nivel API corespunzător. Împreună cu acest nivel API va trebui să avem în timpul dezvoltării pachetul SDK corespunzător acestui nivel.

O listă a acestor versiuni și niveluri API se găsește la următoarea adresă:

<https://apilevels.com/>

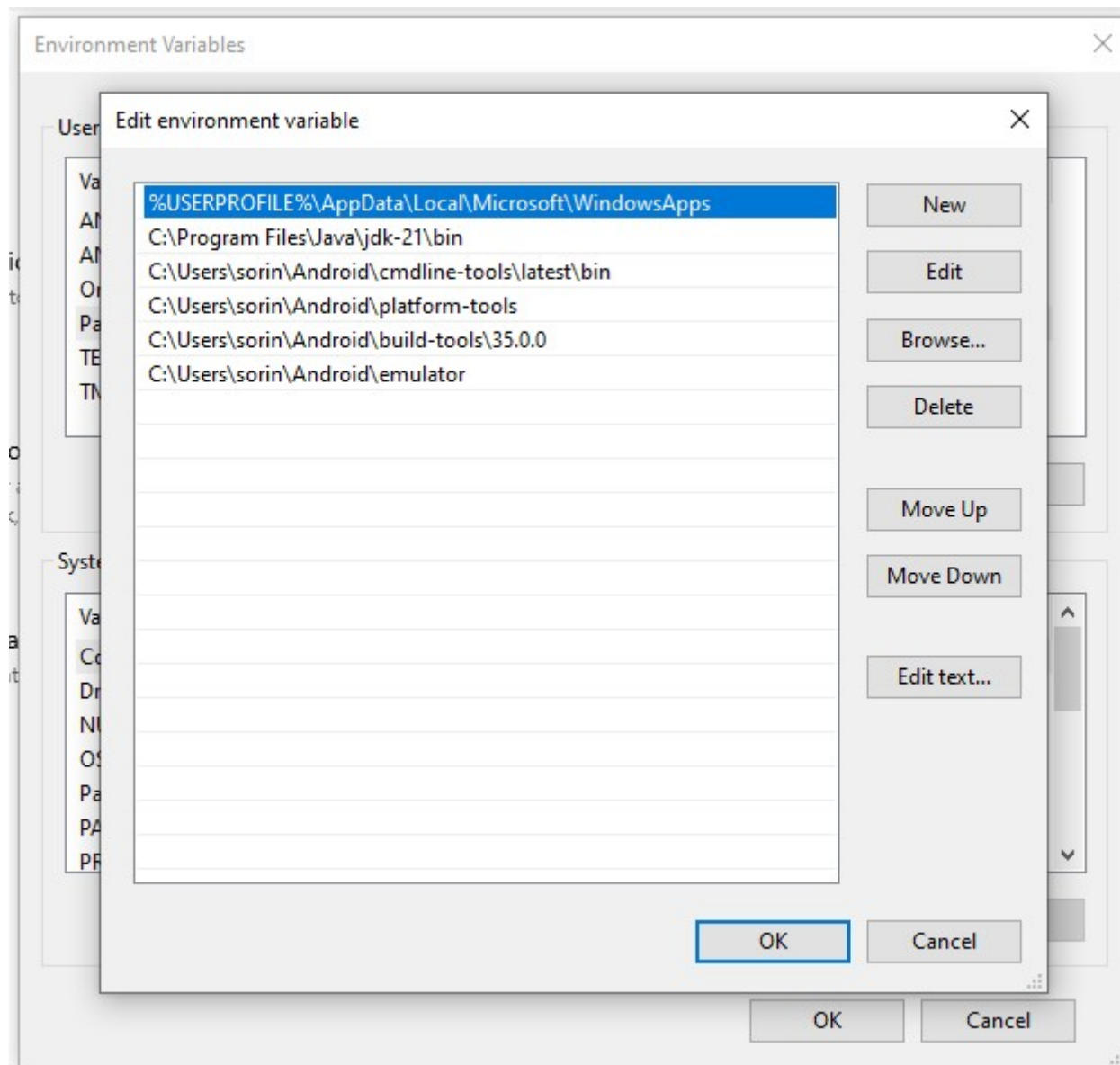
Instalarea propriu-zisă

Instalarea propriu-zisă se face cu ajutorul utilitarului sdkmanager caruia i se dă o comandă de tipul următor:

```
sdkmanager.bat "platform-tools" "build-tools;35.0.0" "platforms;android-26"  
"emulator" "system-images;android-26;google_apis;x86" "cmdline-tools;latest"
```

Evident, alegeti versiunile dorite. Build-Tools poate fi cat mai recent posibil dar platforma si imaginile de sistem trebuie sa corespunda cu versiunea minima de Android pe care doriti ca aplicatia dvs. sa poata rula.

Adaugati urmatoarele componente la PATH



Instalare gradle

Gradle este un sistem de automatizare a construirii aplicatiilor. Este frecvent utilizat în dezvoltarea aplicațiilor Android, dar poate fi folosit și pentru alte tipuri de proiecte (Java, C++, proiecte web).

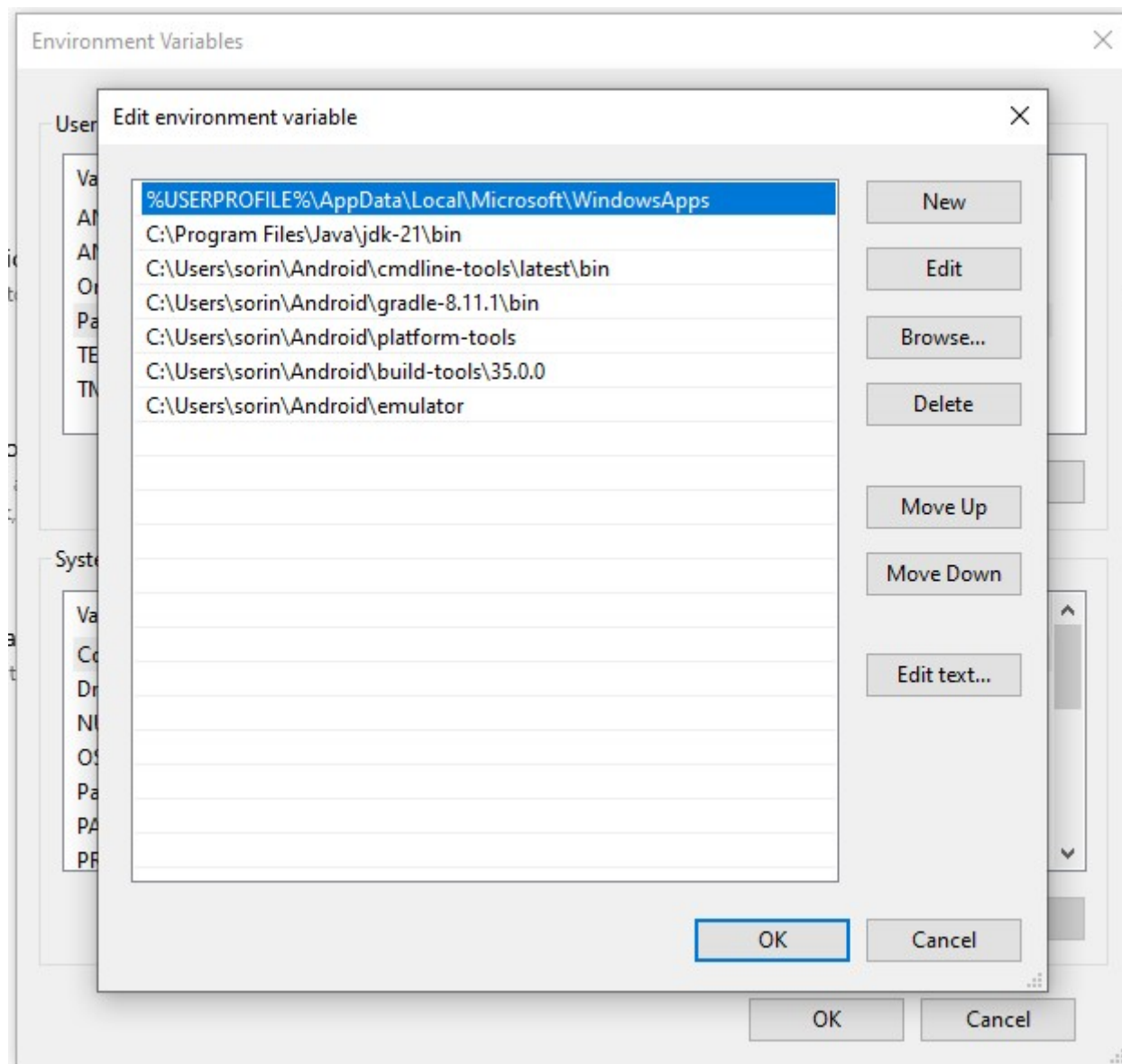
Gradle gestionează cu ușurință bibliotecile externe (de exemplu, prin integrarea cu Maven Central sau alte repository-uri) si permite automatizarea întregului flux de lucru, de la compilare la testare și livrare.

Descarcati gradle de la <https://gradle.org/releases/>

Dezarhivati gradle intr-un director oarecare

"C:\Users\MYUSER\Android\gradle-8.11.1"

Adaugati calea in PATH



Verificati disponibilitatea comenzilor

Dupa setarea tuturor componentelor PATH-ului inchideti si redeschideti instanta PowerShell si verificati disponibilitatea urmatoarelor comenzi.

adb

```
PS C:\Users\MYUSER> adb
Android Debug Bridge version 1.0.41
Version 35.0.2-12147458
Installed as C:\Users\MYUSER\Android\platform-tools\adb.exe
Running on Windows 10.0.19045
...
```

gradle

```
PS C:\Users\MYUSER> gradle

> Task :help

Welcome to Gradle 8.11.1.

Directory 'C:\Users\MYUSER' does not contain a Gradle build.

To create a new build in this directory, run gradle init

For more detail on the 'init' task, see
https://docs.gradle.org/8.11.1/userguide/build\_init\_plugin.html

For more detail on creating a Gradle build, see
https://docs.gradle.org/8.11.1/userguide/tutorial\_using\_tasks.html

To see a list of command-line options, run gradle --help

For more detail on using Gradle, see
https://docs.gradle.org/8.11.1/userguide/command\_line\_interface.html

For troubleshooting, visit https://help.gradle.org
```

avdmanager

```
PS C:\Users\MYUSER> avdmanager

Usage:
  avdmanager [global options] [action] [action options]
Global options:
-s --silent      : Silent mode, shows errors only.
-v --verbose     : Verbose mode, shows errors, warnings and all messages.
--clear-cache   : Clear the SDK Manager repository manifest cache.
-h --help       : Help on a specific command.

Valid actions are composed of a verb and an optional direct object:
- list          : Lists existing targets or virtual devices.
- list avd      : Lists existing Android Virtual Devices.
- list target   : Lists existing targets.
- list device   : Lists existing devices.
- create avd    : Creates a new Android Virtual Device.
- move avd      : Moves or renames an Android Virtual Device.
- delete avd    : Deletes an Android Virtual Device.
```

emulator

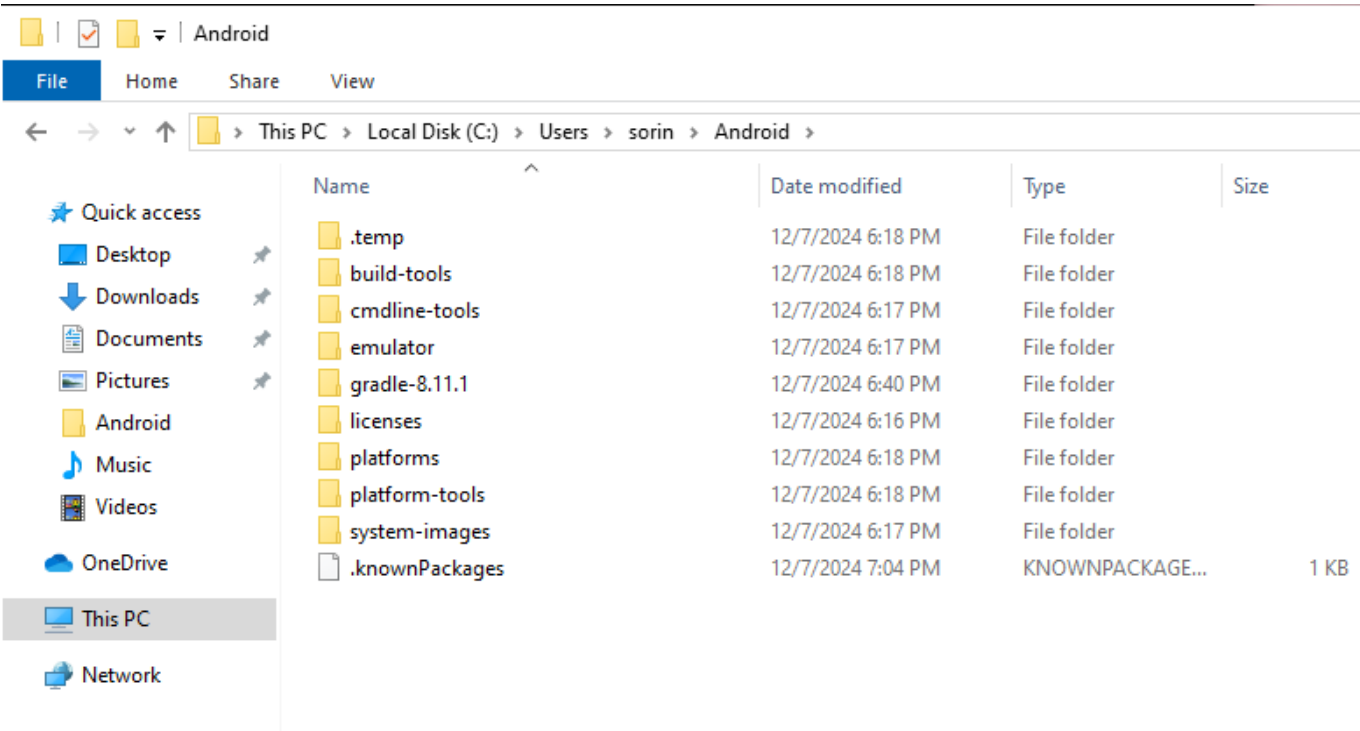
```
PS C:\Users\MYUSER> emulator
INFO      | Android emulator version 35.2.10.0 (build_id 12414864) (CL:N/A)
INFO      | Graphics backend: gfxstream
ERROR     | No AVD specified. Use '@foo' or '-avd foo' to launch a virtual device
          | named 'foo'
```

Verificati toate componentele folosind sdkmanager

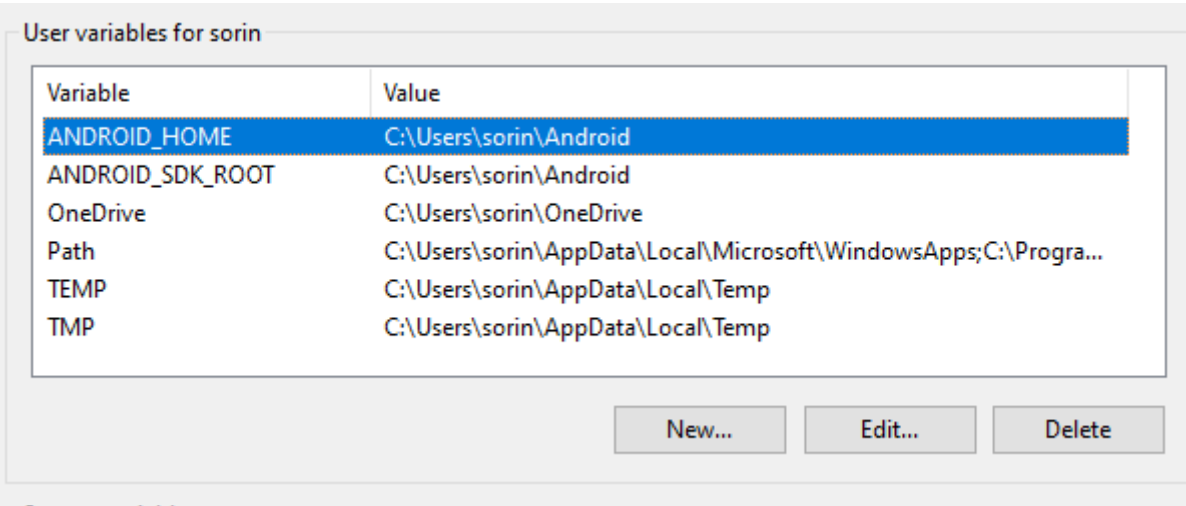
```
sdkmanager.bat --list_installed
```

```
Installed packages:
  Path                                | Version | Description
  | Location
  | -----
  | -----
  build-tools;34.0.0                 | 34.0.0  | Android SDK Build-Tools
34                                  | build-tools\34.0.0
  emulator                           | 35.2.10 | Android Emulator
  | emulator
  platform-tools                     | 35.0.2  | Android SDK Platform-
Tools                               | platform-tools
  platforms;android-34               | 3        | Android SDK Platform 34
  | platforms\android-34
  system-images;android-34;google_apis;x86_64 | 14      | Google APIs Intel x86_64
Atom System Image | system-images\android-34\google_apis\x86_64
```

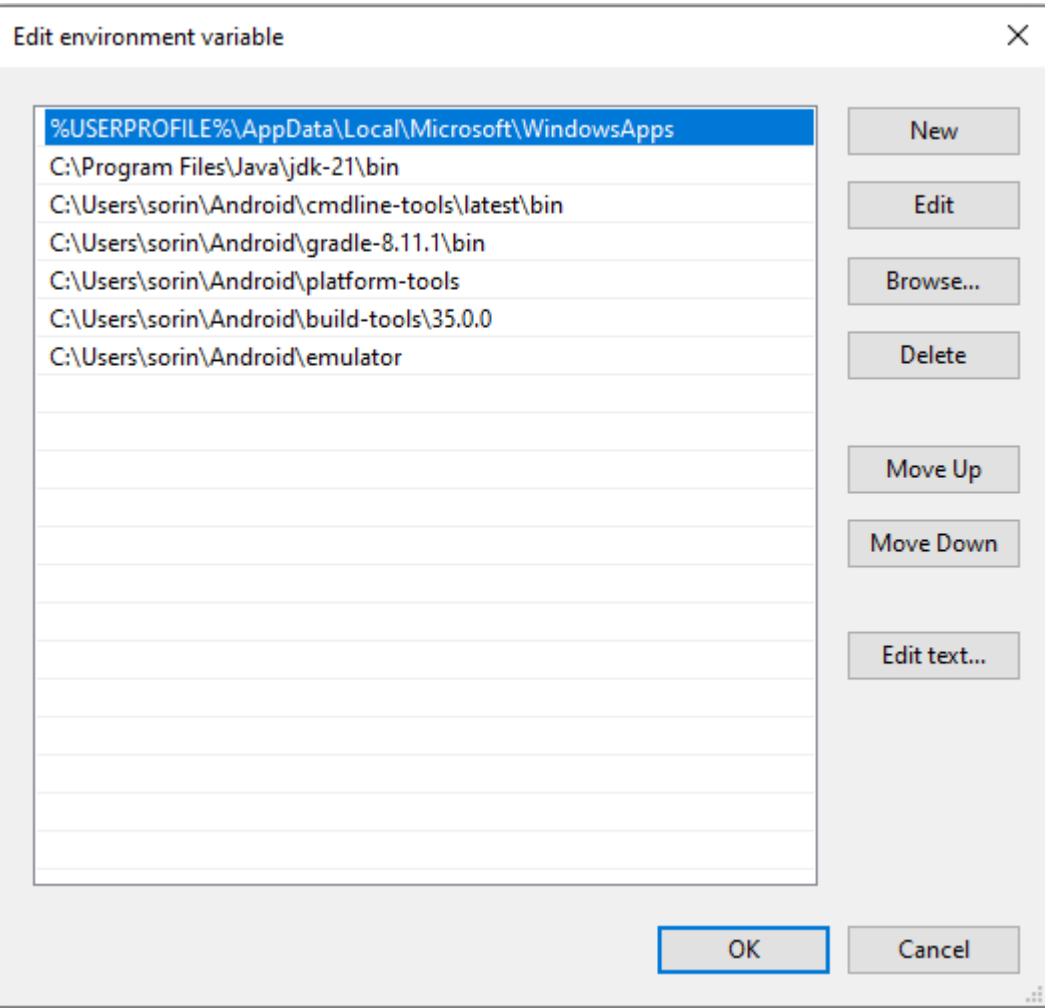
Directorul in care sunt instalate instrumentele de dezvoltare Android ar trebui sa arate astfel:



Variabilele de mediu trebuie sa arate astfel:

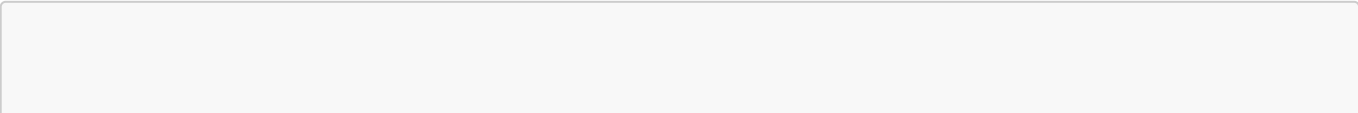


Continutul variabilei PATH este astfel:



Emulator

Creati un avd



```
avdmanager.bat create avd -n 'winkril' -k "system-images;android-26;google_apis;x86" -d pixel_7
```

Porniti emulatorul

```
emulator.exe -avd winkril
```

Lista de dispozitive existente

```
adb.exe devices
```

```
List of devices attached  
emulator-5554    device
```

Instalati aplicatia pe dispozitiv

```
adb -s emulator-5554 install .\app\build\outputs\apk\debug\app-debug.apk
```

Verificati existenta aplicatiei (powershell)

```
adb shell pm list packages | grep {application_name}
```

Instalari parțiale

În anumite situații puteți dori să instalați doar anumite componente din întregul lanț de instrumente prezentat mai sus. Un posibil exemplu este acela în care, fără să doriți să compilați și să instalați propriile aplicații pe un dispozitiv extern (în special unul real) ați dori să aveți acces la funcționalitățile acestuia utilizând adb (Android Debug Bridge). adb este parte din pachetul platform-tools.

Platform tools

<https://developer.android.com/tools/releases/platform-tools>

Downloads

If you're an Android developer, you should get the latest SDK Platform-Tools from Android Studio's [SDK Manager](#) or from the `sdkmanager` command-line tool. This ensures the tools are saved to the right place with the rest of your Android SDK tools and easily updated.

But if you want just these command-line tools, use the following links:

- [Download SDK Platform-Tools for Windows](#)
- [Download SDK Platform-Tools for Mac](#)
- [Download SDK Platform-Tools for Linux](#)

Although these links do not change, they always point to the most recent version of the tools.

Veti descarca o arhiva zip al carei nume este platform-tools-latest-windows.zip. Aceasta se dezarchiveaza intr-un director oarecare (in continuare se recomanda crearea unui ANDROID_HOME, un director doar pentru aceste instrumente) dar nu este obligatoriu. De asemenea, se recomanda plasarea directorului "platform_tools" dezarhivat in PATH-ul sistemului de operare (general sau user) dar nici aceasta nu este necesara, instrumentele se pot rula direct din directorul unde au fost dezarhivate.

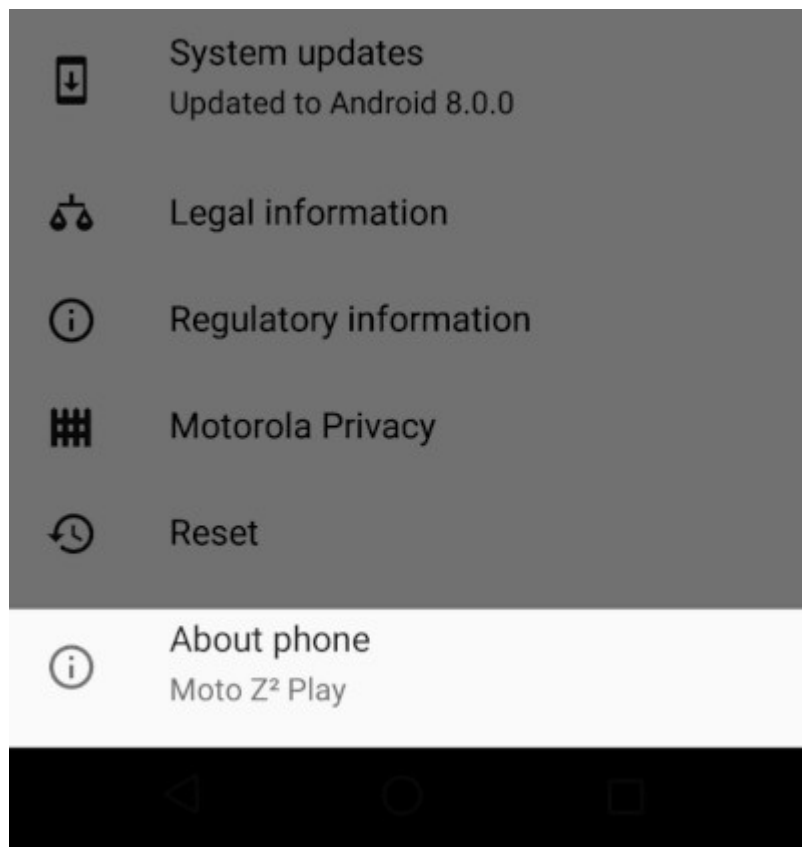
Acces telefon hardware

Pregatirea dispozitivului

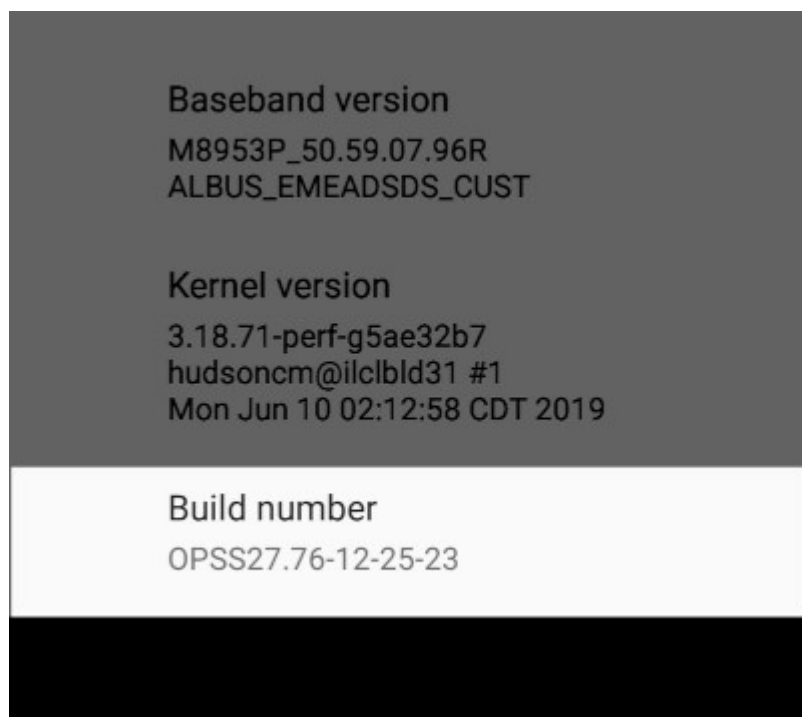
Pentru a putea conecta telefonul la computer in scopul operatiilor de dezvoltare sunt necesare cateva operatii de pregatire pe terminalul Android.

Activarea optiunilor de dezvoltare

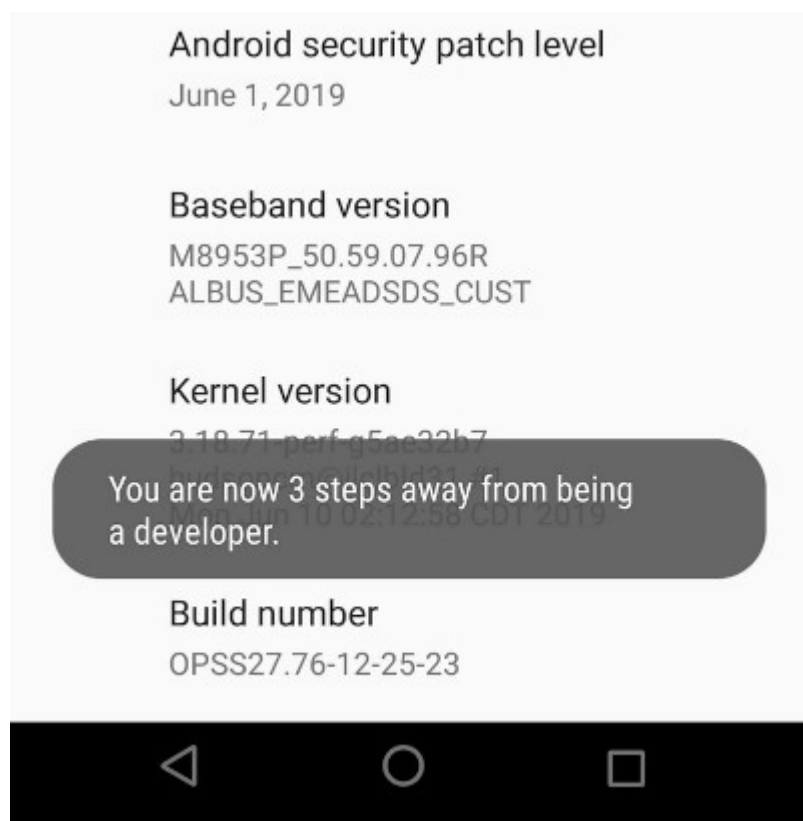
Cautati in grupul de setari pe telefon o optiune numita "About phone"



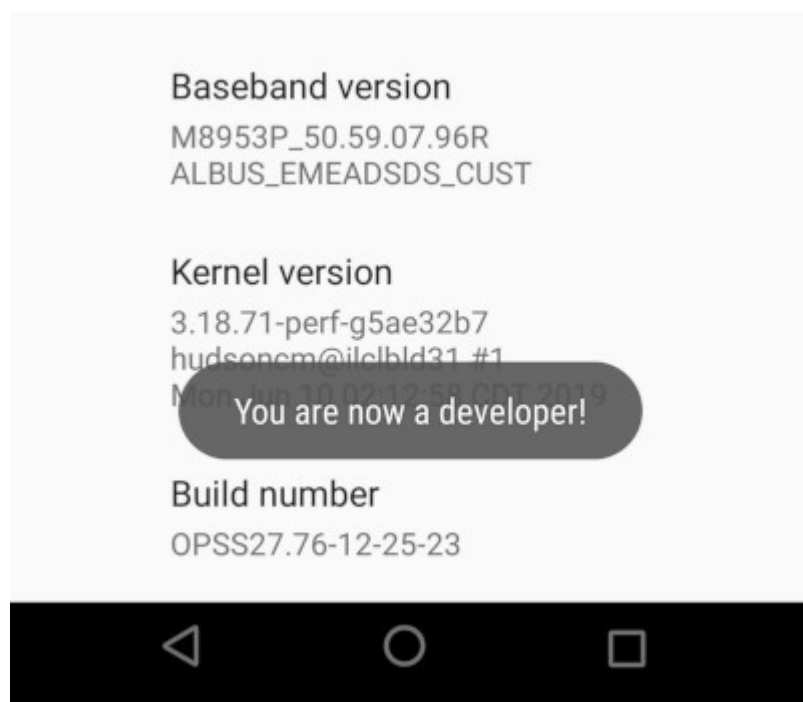
In interiorul acesteia, identificati intrarea numita "Build number"



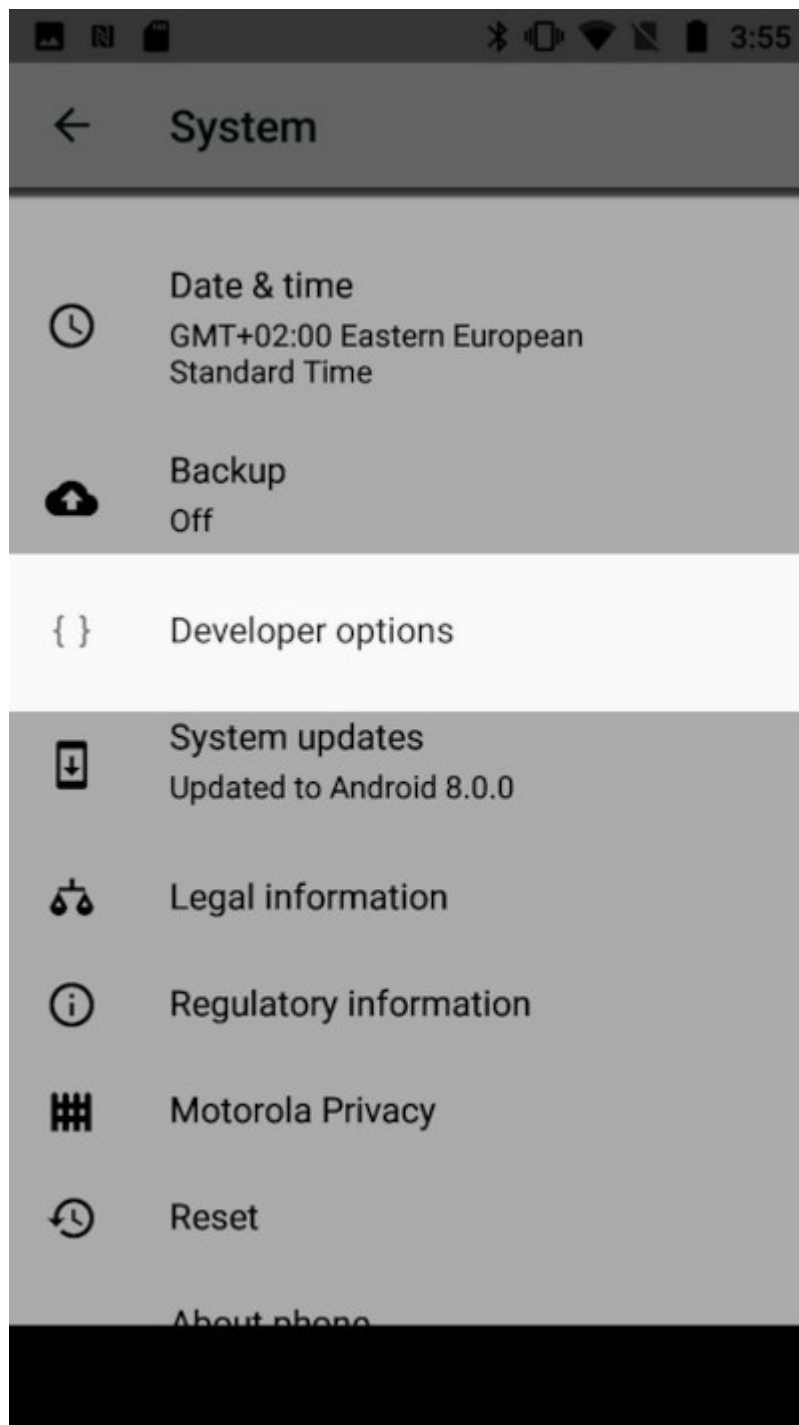
apasati de 7 ori pe optiunea "Build number". Dupa cateva apasari, va apare o atentionare care va va spune ca mai aveti un anumit numar de apasari pana cand deveniti developer:



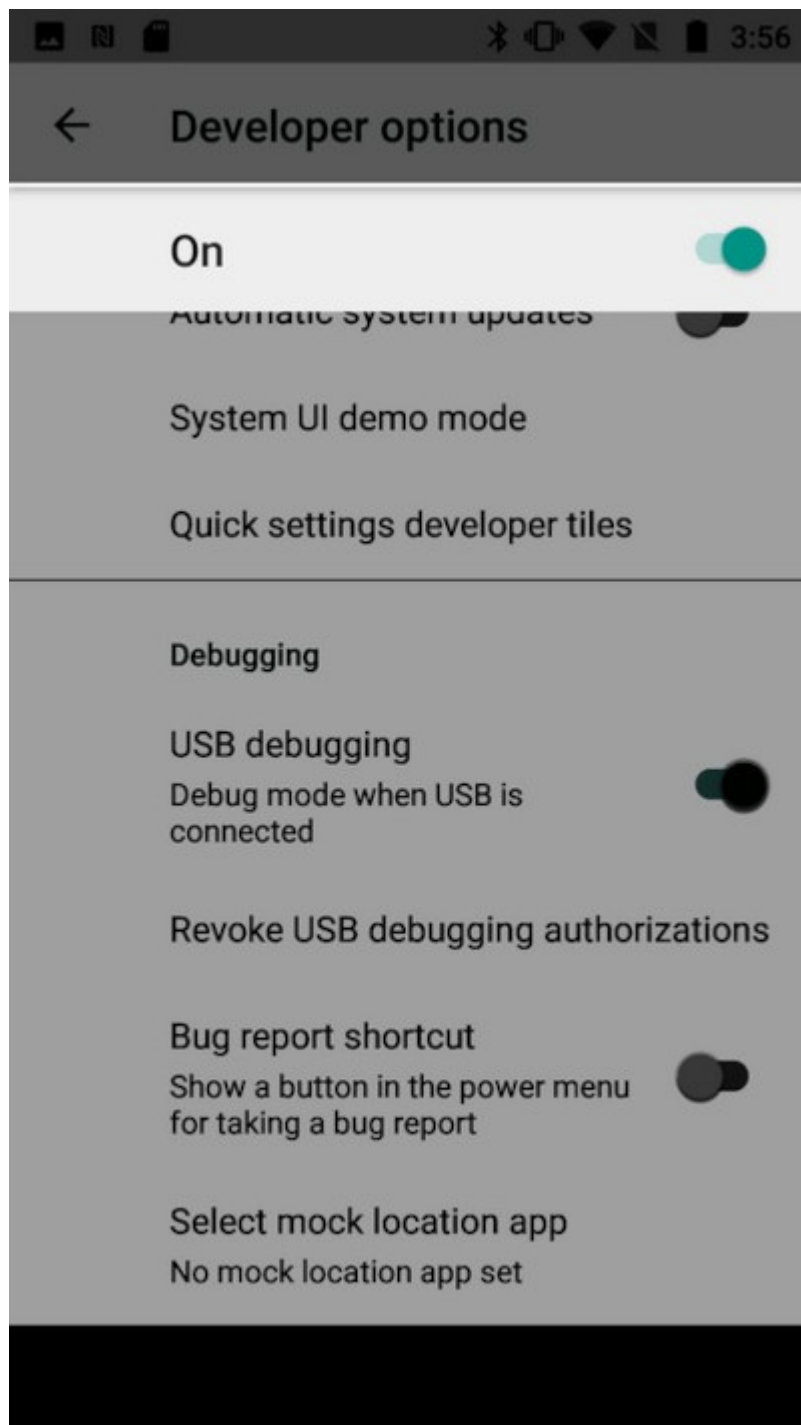
Dupa finalizarea celor 7 apasari, apare o alta atentionare care va spune ca ati devenit developer - optiunile pentru dezvoltatori au fost activate



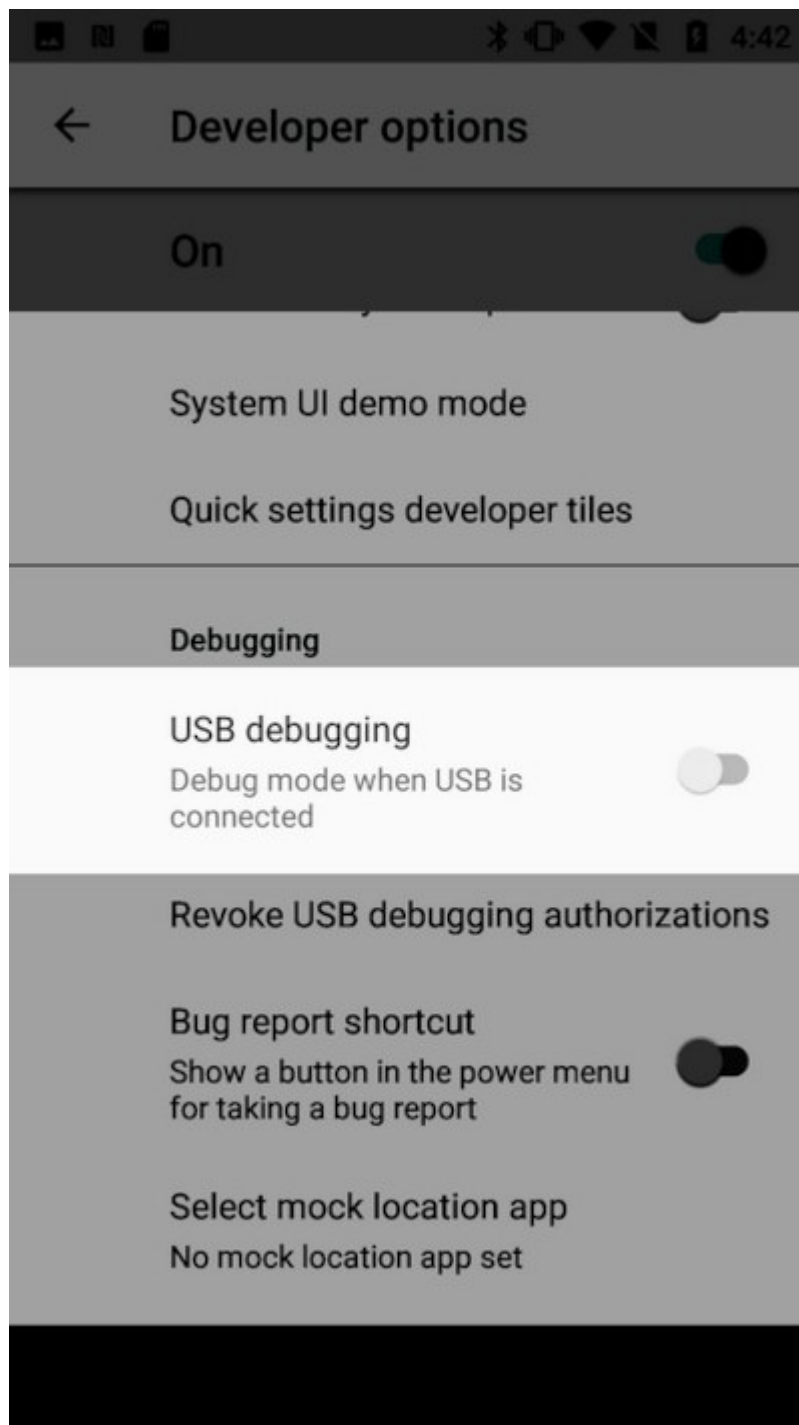
Optiunile pentru dezvoltare sunt disponibile sub forma unui meniu suplimentar in zona de setari a telefonului:



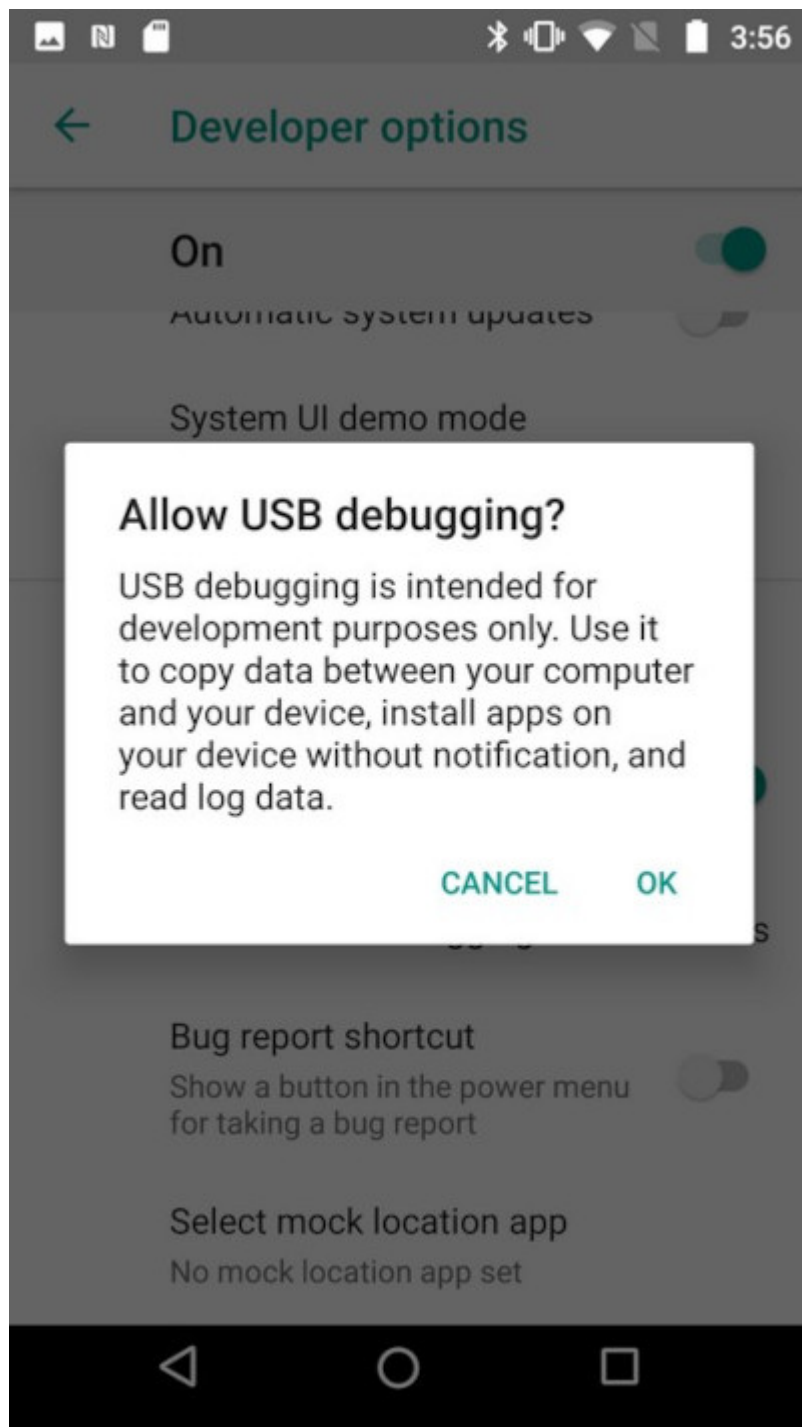
Daca intrati in sectiunea la care duce acest meniu (si va trebui sa o faceti) veti observa in primul rand un slider care permite oprirea acestor optiuni:



si apoi o serie de optiuni care trebuie activate pentru a putea accesa telefonul de pe computer in modul de dezvoltare, cea mai importanta dintre ele fiind USB debugging:

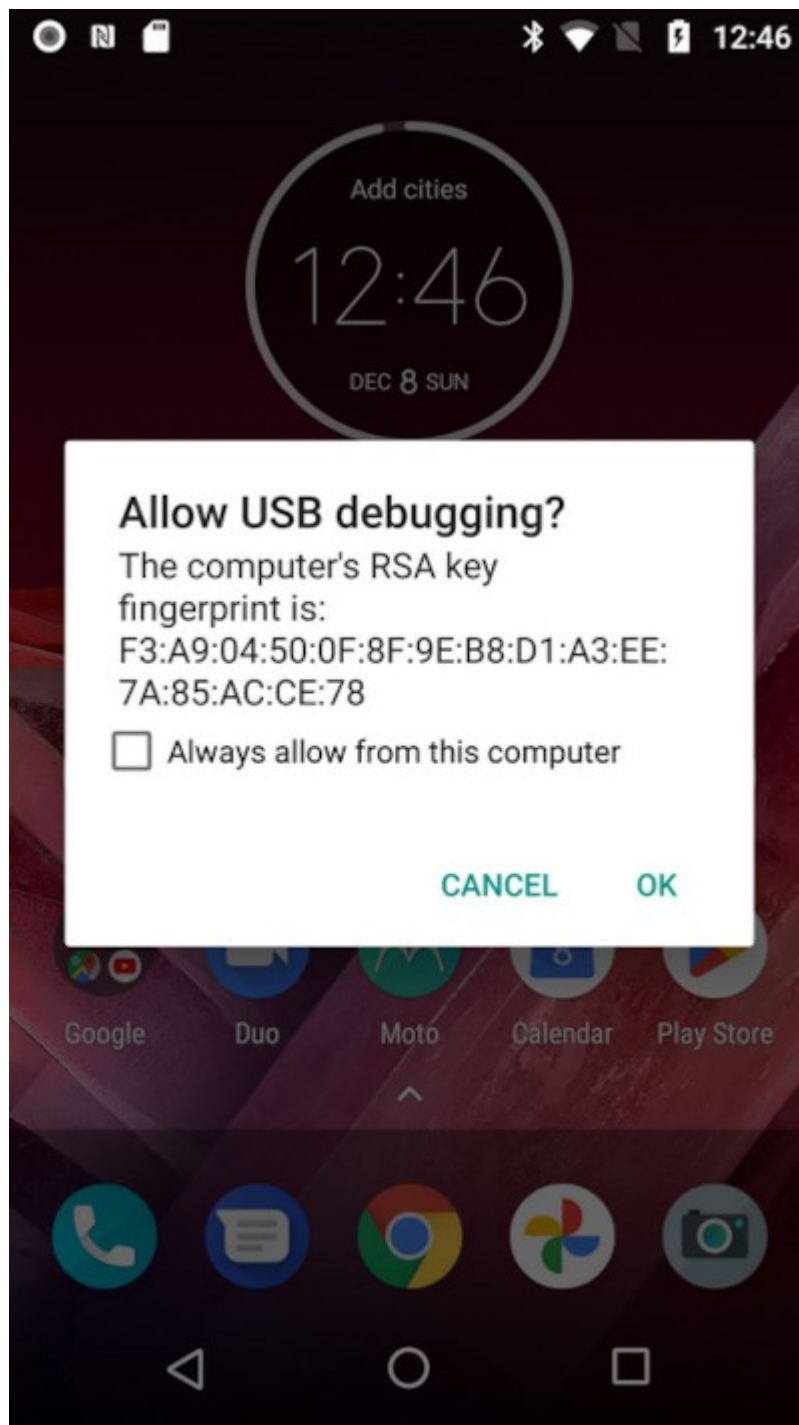


La activarea acestei optiuni va aparea un dialog care va solicita o confirmare. Evident, aceasta trebuie acceptata.

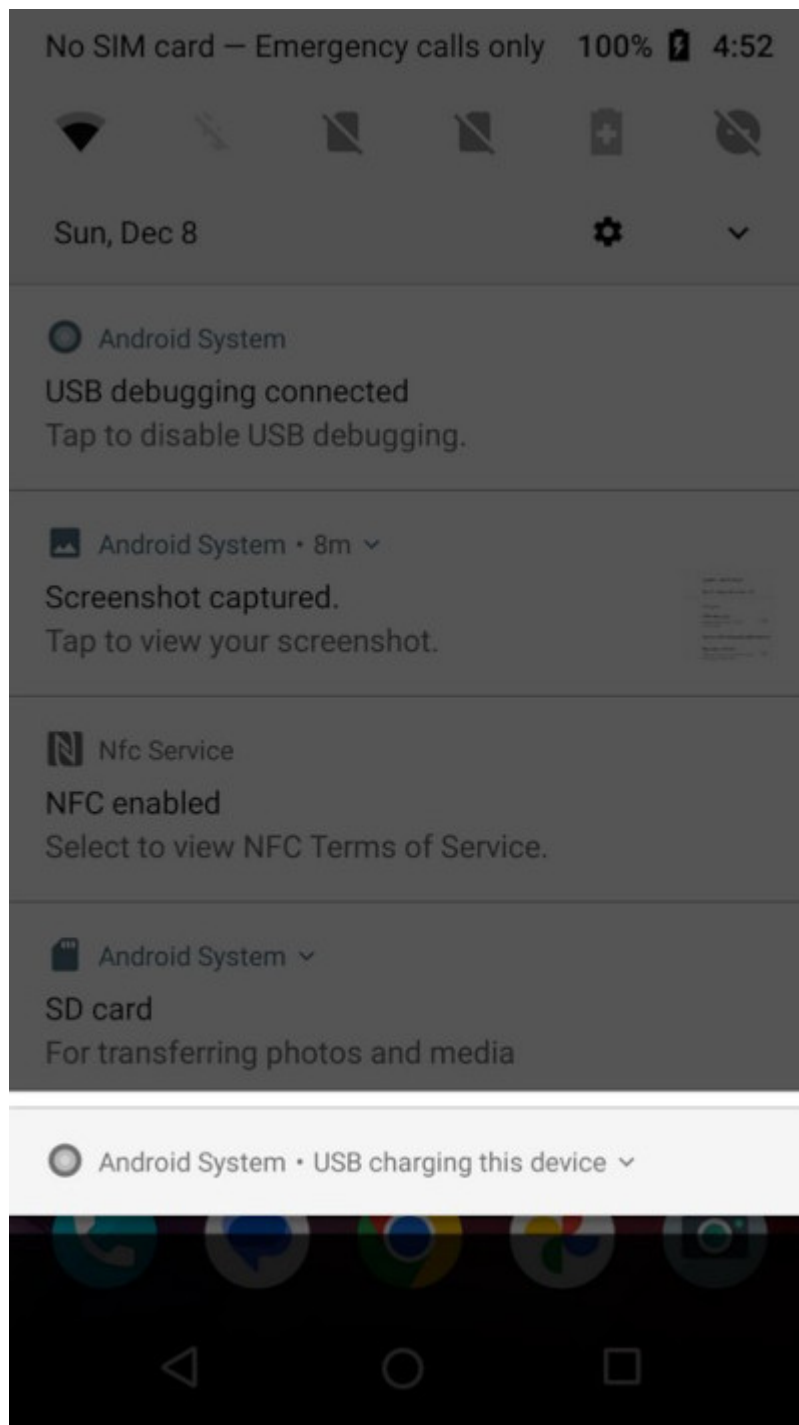


Alte optiuni utile in acelasi ecran (dar nu obligatorii) sunt OEM Unlockin, Stay Awake, Allow Mock Locations. O parte dintre ele sunt insotite de dialoguri de confirmare, altele nu.

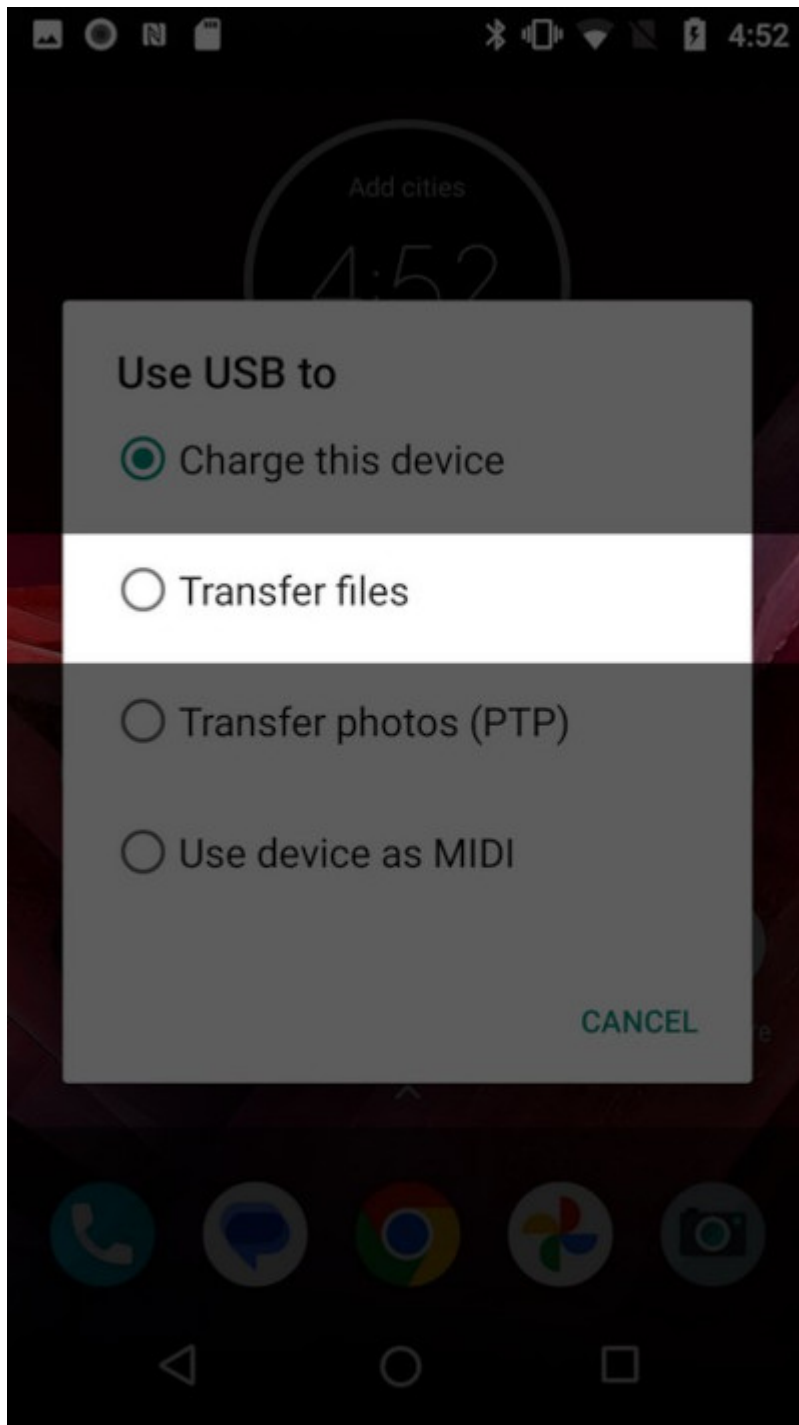
Dupa activarea acestor optiuni, conectati telefonul la PC folosind un cablu USB. Va aparea un nou dialog de confirmare, de aceasta data direct legat de pc-ul conectat:



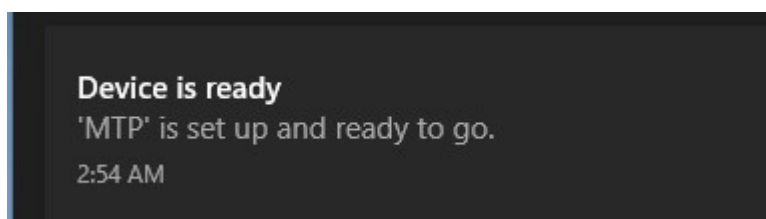
Confirmati aceasta conexiune, apoi, verificati tipul de conexiune USB care s-a stabilit cu computerul. Aceasta este de obicei disponibila dupa conectare in sectiunea de notificari:



De cele mai multe ori, la initierea unei conexiuni noi, aceasta este pusa pe "Charging", va trebui modificata in "Transfer files"



Daca conexiunea este corecta, veti observa o notificare venita de la sistemul de operare:



In cazul in care doriti sa verificati mai departe, puteti folosi comanda `Get-PnpDevice -Class USB` (ideal cu o rulare inainte de conexiunea respectiva) pentru a verifica prezenta dispozitivului USB in subsistem. De cele mai multe ori acesta apare ca UDB Composite Device.

```
PS C:\Users\MYUSER> Get-PnpDevice -Class USB
```

Status	Class	FriendlyName
InstanceId		
-----	-----	-----

OK	USB	USB Composite Device
USB\VID_22B8...		
OK	USB	USB Root Hub (USB 3.0)
USB\ROOT_HUB...		
OK	USB	Intel(R) USB 3.0 eXtensible Host Controller - 1.0
(Microsoft)		PCI\VEN_8086...

Daca toptul a mers bine si aveti instalat pachetul "platform tools", comanda adb va va afisa dispozitivul prezent:

```
C:\Users\MYUSER\Android\platform-tools> adb devices
List of devices attached
ZY224F8Q2G      device
```

Ocazional, se poate intampla ca dispozitivul sa fie afisat ca fiind neautorizat, caz in care va trebui sa verificati dispozitivul unde va aparea un nou dialog de confirmare.

```
C:\Users\MYUSER\Android\platform-tools> adb devices
* daemon not running; starting now at tcp:5037
* daemon started successfully
List of devices attached
ZY224F8Q2G      unauthorized
```

Cateva comenzi adb

adb shell pm list packages

```
C:\Users\MYUSER\Android\platform-tools> adb shell pm list packages
package:com.android.cts.priv.ctsshim
package:com.aura.oobe.ml
package:com.google.android.youtube
package:com.google.android.ext.services
package:com.motorola.motocare
package:com.android.providers.telephony
package:com.android.sdm.plugins.connmo
package:com.google.android.googlequicksearchbox
package:com.android.providers.calendar
package:com.android.providers.media
package:com.google.android.apps.docs.editors.docs
package:com.qti.service.colorsense
```

```
...
```

adb logcat

```
C:\Users\MYUSER\Android\platform-tools> adb logcat
----- beginning of system
12-07 23:47:59.505 638 843 I SFPerfTracer: triggers: (rate: 0:6) (8441 sw
vsyncs) (0 skipped) (588:6468 vsyncs) (590:15013)
12-07 23:47:59.596 2777 4285 I chatty : uid=1000(system) Binder:2777_B expire
22 lines
12-07 23:47:59.856 2777 3475 I chatty : uid=1000(system) Binder:2777_3 expire
13 lines
12-07 23:48:01.124 2777 2777 I chatty : uid=1000 system_server expire 11 lines
12-07 23:48:01.655 2777 3608 I chatty : uid=1000(system) VoldConnector expire
11 lines
12-07 23:48:01.819 2777 13625 I chatty : uid=1000(system) Binder:2777_19 expire
13 lines
12-07 23:48:06.348 2777 2822 I chatty : uid=1000(system) android.fg expire 12
lines
12-07 23:48:06.354 2777 3873 I chatty : uid=1000(system) Binder:2777_8 expire
11 lines
12-07 23:48:06.395 2777 3690 I chatty : uid=1000(system) ConnectivitySer expire
15 lines
```

screenshot

```
adb shell screencap -p /sdcard/screen.jpg adb pull /sdcard/screen.jpg
```

screen recording

Inregistrarea ecranului se opreste automat dupa 3 minute sau poate fi oprita cu Ctrl-C

```
adb shell screenrecord /sdcard/screen.mp4
adb pull /sdcard/screen.mp4
```

adb shell getprop

Android mentine o baza de date de proprietati (un fel de registry) care poate fi vizualizat:

```
(base) PS C:\Users\MYUSER\Android\platform-tools> adb shell getprop
[DEVICE_PROVISIONED]: [1]
[af.fast_track_multiplier]: [1]
[audio.deep_buffer.media]: [true]
[audio.dolby.ds2.enabled]: [true]
[audio.offload.buffer.size.kb]: [64]
```

```
[audio.offload.disable]: [false]
...
```

adb shell settings put

Valorile din aceasta baza de date pot fi modificate

```
adb shell settings put system screen_off_timeout 60000
```

Alte comenzi

adb shell ps adb shell input tap 500 500

```
adb shell dumpsys battery
Current Battery Service state:
  AC powered: false
  USB powered: true
  Wireless powered: false
  MOD powered: false
  Max charging current: 500000
  Max charging voltage: 5000000
  Charge counter: 3059000
  status: 5
  health: 2
  present: true
  level: 100
  scale: 100
  voltage: 4378
  temperature: 260
  technology: Li-ion
  mod level: -1
  mod status: 1
  mod flag: 0
  mod type: 0
  mod power source: 0
```

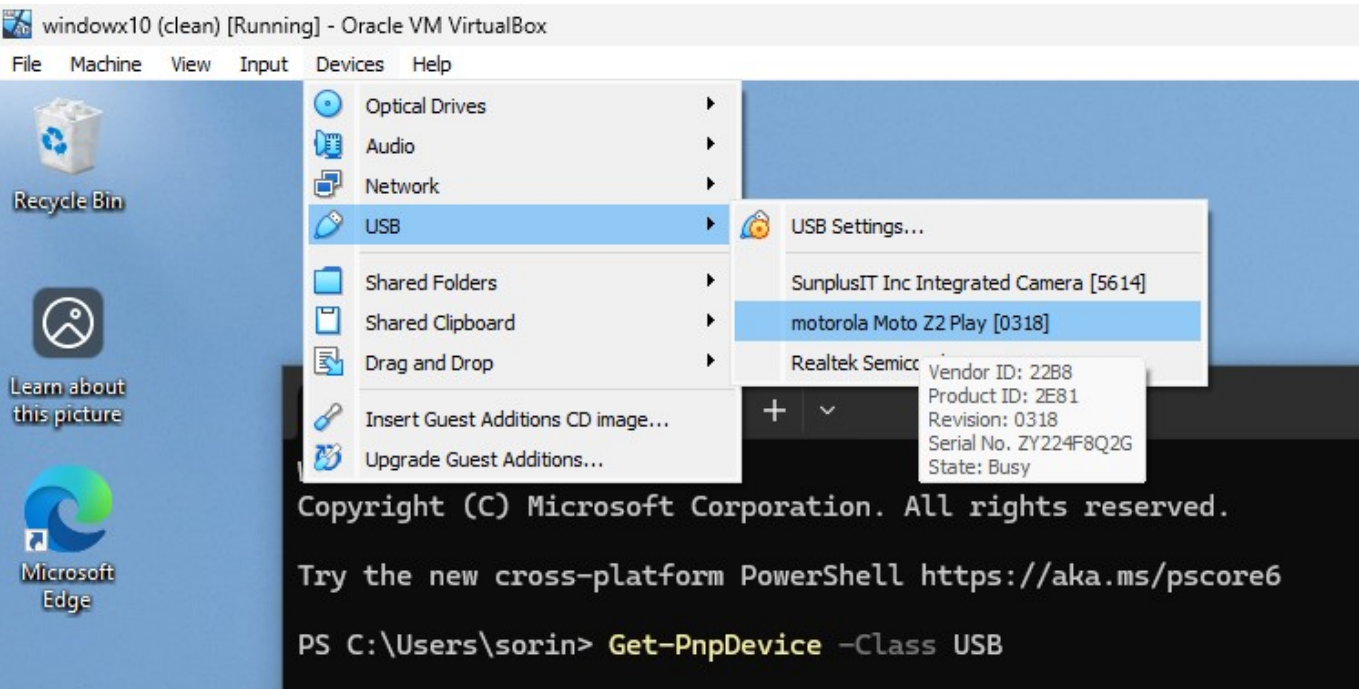
```
adb shell ifconfig wlan0
wlan0      Link encap:UNSPEC      Driver wcnss_wlan
           inet addr:192.168.27.110  Bcast:192.168.27.255  Mask:255.255.255.0
           inet6 addr: fe80::8ab4:a6ff:fec9:b03d/64 Scope: Link
           UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
           RX packets:947981 errors:0 dropped:0 overruns:0 frame:0
           TX packets:192772 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:1000
           RX bytes:1224398272 TX bytes:17229241
```

Lista proceselor care ruleaza in linux

```
(base) PS C:\Users\MYUSER\Android\platform-tools> adb shell ps
USER          PID  PPID   VSZ   RSS  WCHAN          ADDR S  NAME
root           1     0  18232   1976 SyS_epoll_wait    0 S  init
root           2     0     0     0 kthreadd          0 S  [kthreadd]
root           3     2     0     0 smpboot_thread_fn 0 S  [ksoftirqd/0]
root           5     2     0     0 worker_thread    0 S  [kworker/0:0H]
root           7     2     0     0 rcu_gp_kthread    0 S  [rcu_preempt]
root           8     2     0     0 rcu_gp_kthread    0 S  [rcu_sched]
root           9     2     0     0 rcu_gp_kthread    0 S  [rcu_bh]
root          10     2     0     0 rcu_nocb_kthread  0 S  [rcuop/0]
root          11     2     0     0 rcu_nocb_kthread  0 S  [rcuos/0]
```

Conectarea unui dispozitiv USB la o masina virtuala

In cazul in care instrumentele de dezvoltare sunt instalate intr-o masina virtuala (ex. VirtualBox), aceasta nu va putea rula emulatorul de dispozitiv asa ca dezvoltarea va trebui sa fie facuta direct pe un dispozitiv real. Pentru ca acesta sa devina vizibil in sistemul de operare guest trebuie atasat cu ajutorul instrumentelor tipice hipervizorului. De asemenea, in majoritatea situatiilor este necesar ca instrumentele client ale sistemului de masini virtuale (ex: Virtual box guest additions).



Dupa atasarea dispozitivului cu ajutorul meniului, adb trebuie sa permita deschiderea unei conexiuni la

Alte instrumente utile

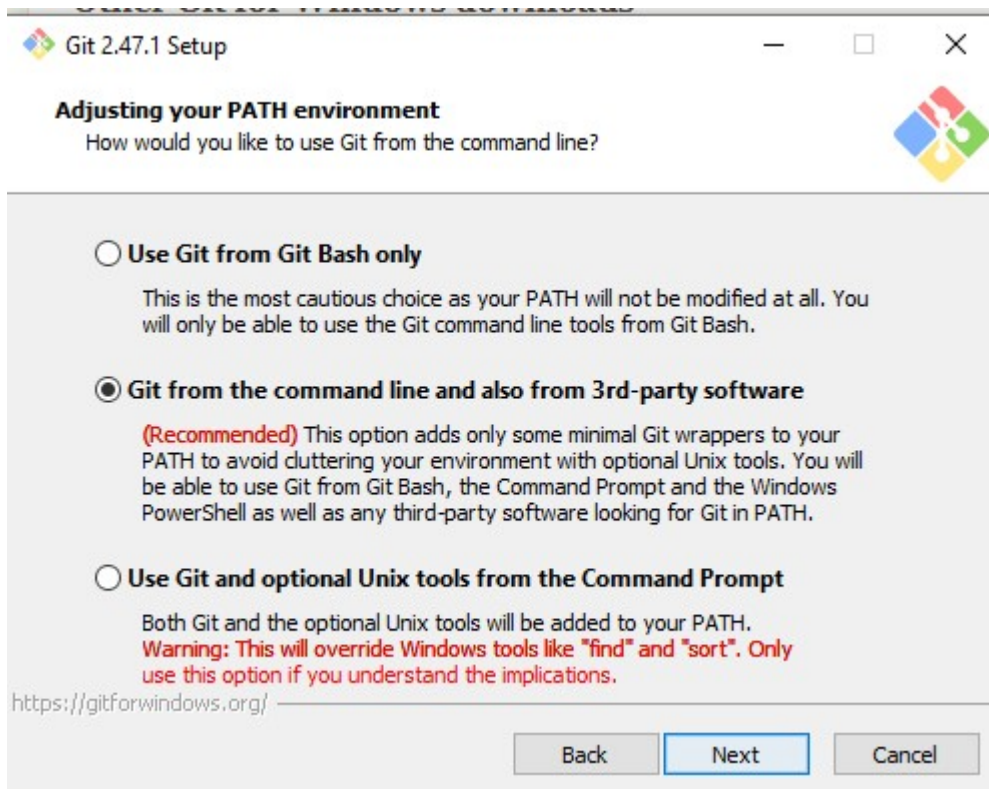
git

<https://git-scm.com/downloads/win>



The screenshot shows the Git website's 'Download for Windows' page. The header features the Git logo and the tagline '--fast-version-control'. A search bar is located in the top right corner. The left sidebar contains navigation links: 'About', 'Documentation', 'Downloads' (highlighted in red), 'GUI Clients', 'Logos', and 'Community'. The main content area is titled 'Download for Windows' and includes a link to download the latest (2.47.1) 64-bit version of Git for Windows, noting it was released 13 days ago on 2024-11-25. Below this, there are links for 'Other Git for Windows downloads', including 'Standalone Installer', '32-bit Git for Windows Setup', '64-bit Git for Windows Setup', 'Portable ("thumbdrive edition")', '32-bit Git for Windows Portable', and '64-bit Git for Windows Portable'. A sidebar on the left mentions the 'Pro Git book' by Scott Chacon and Ben Straub, available online for free, with dead tree versions on Amazon.com.

În timpul instalării Git există o etapă importantă - selecția celor care vor fi adăugate la PATH. Varianta recomandată (și de obicei selectată automat) este cea care va face comanda "git" să fie disponibilă din orice punct al sistemului. Această distribuție de Git este însoțită și de alte instrumente tipice pentru Unix/Linux care pot fi utile (ex: grep, find, etc.). Acestea pot fi utile dacă lucrați mult în consolă în Windows și dacă alegeți opțiunea a 3-a vor deveni disponibile dar trebuie să țineți cont de faptul că vor invalida câteva comenzi ale sistemului Windows (nu critice).

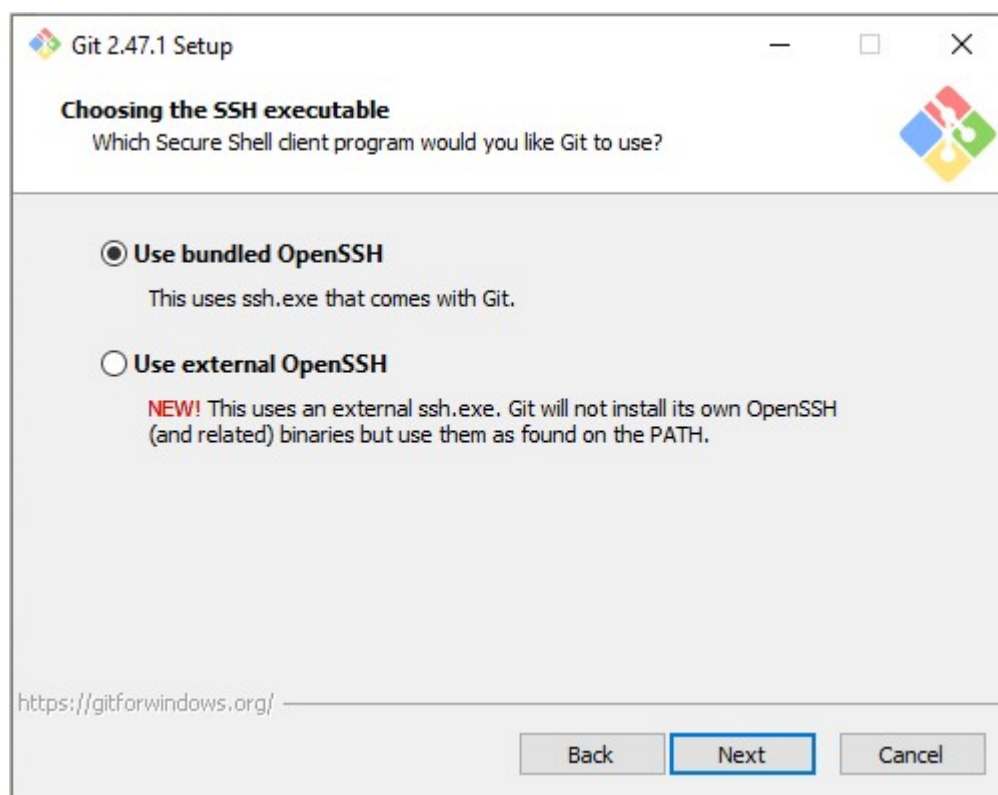


Git folosește openssh pentru a comunica cu repository-urile de la distanță atunci când se solicită comunicarea prin protocolul SSH. Distribuția de GIT pentru Windows este însoțită de o versiune de OpenSSH. Windows 10 sau 11 au și ele o versiune de OpenSSH. Putem verifica prezența acestuia cu ajutorul comenzii `Get-Command ssh`

```
PS C:\Users\MYUSER> Get-Command ssh.exe
```

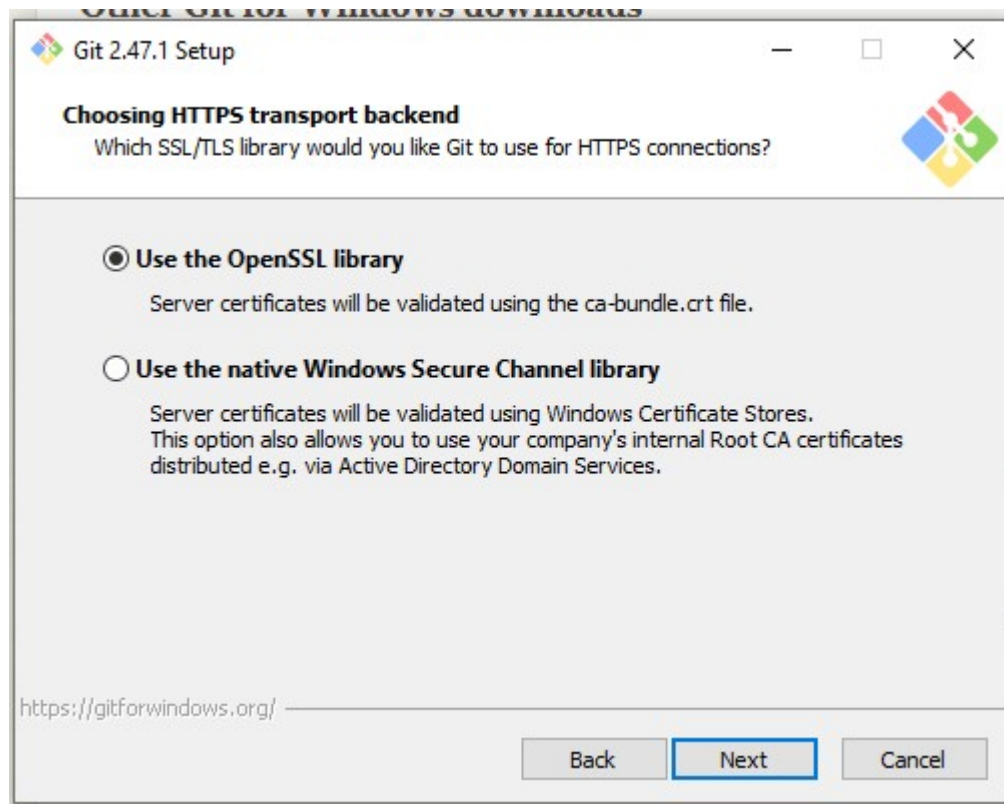
CommandType	Name	Version
Source		
-----	----	-----
--		
Application	ssh.exe	9.5.2.1
C:\Windows\System32\OpenSSH\ssh.exe		

Noile versiuni ale distributiei pentru windows ofera posibilitatea de a folosi ssh-ul disponibil deja in sistemul de operare.

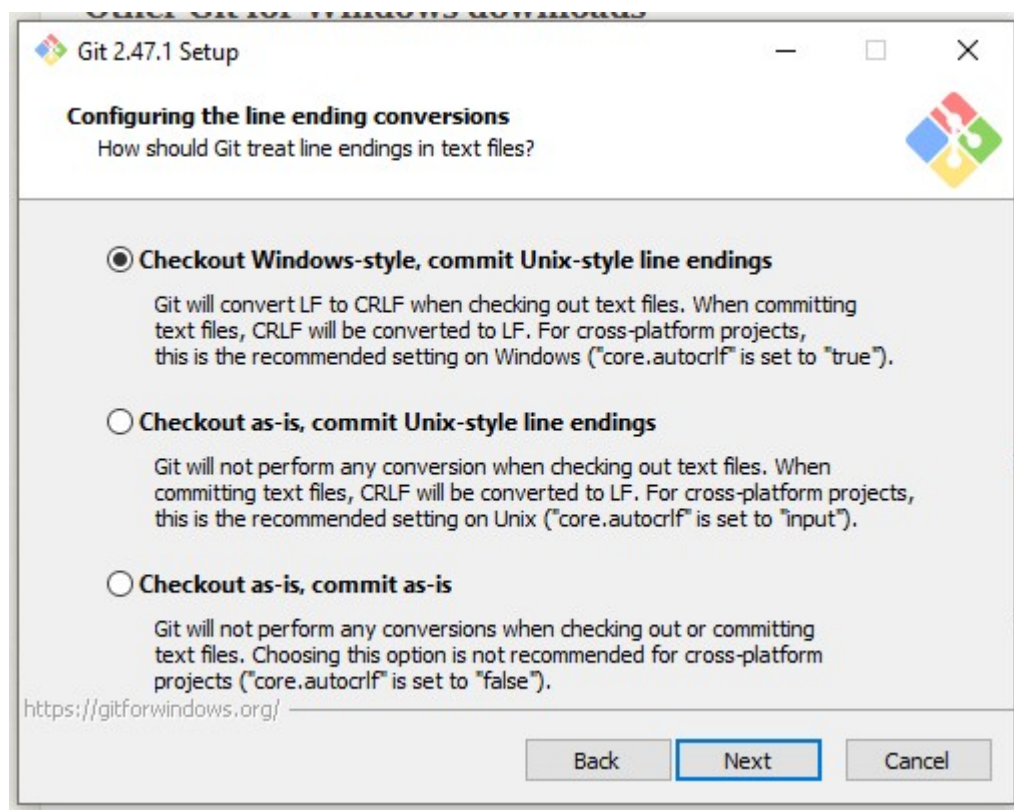


In cazul in care accesul la repository-uri este facut prin protocolul HTTPS, este nevoie de identificarea validitatii certificatelor serverelor externe. Mecanismul de identificare porneste de la o lista redusa de certificate ROOT public cunoscute. Unul dintre acestea se gaseste la baza lantului de certificate. Aceste certificate ROOT se pot prelua dintr-un fisier distribuit cu git-ul sau direct din Windows. Varianta a doua este utilizata in cazul in care exista certificate ROOT locale, nepublice (emise de institutii si de obicei distribuite automat pe computerele de serviciu).

Alegeti varianta a 2-a daca trebuie sa accesati un repository aflat intr-un spatiu privat de stocare al carui certificat este bazat pe un ROOT non-public. In caz contrar, oricare dintre variante va functiona.

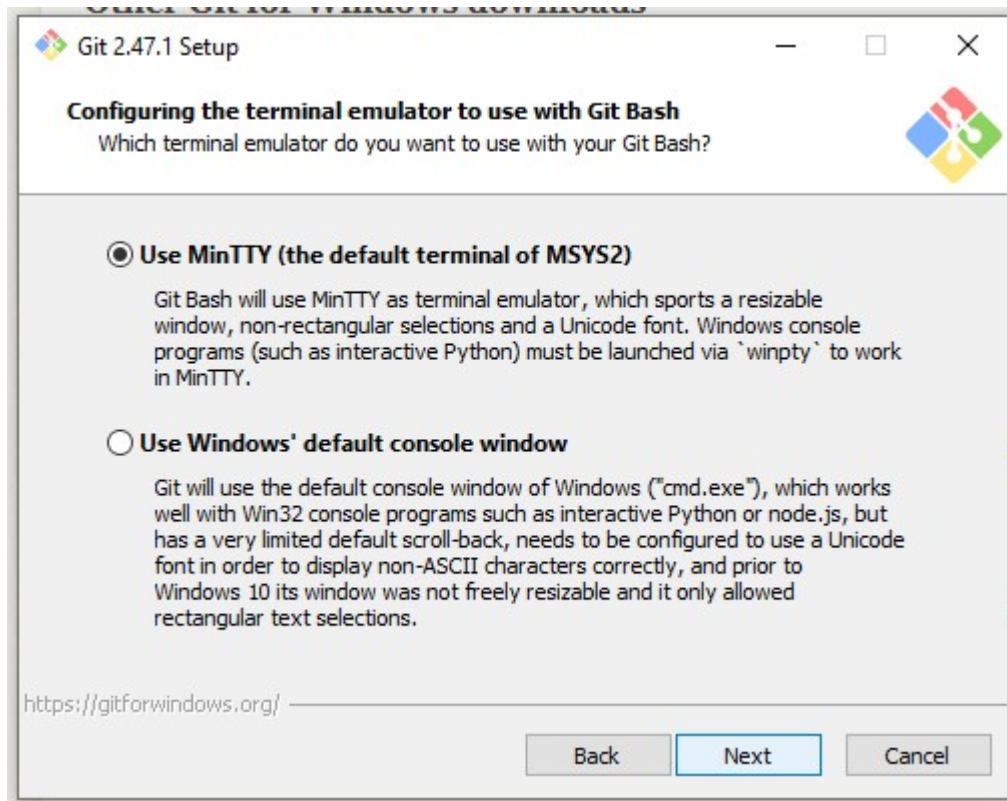


Urmatorul ecran se refera la modul in care sunt tratate finalurile de rand in fisierele de tip text. Toate fisierele care sunt versionate in git sunt fisiere text. Intr-un fisier text, finalul unui rand este marcat diferit in Windows si Unix. Pentru dezvoltarea pe Windows se recomanda selectia primei optiuni.



La instalarea pachetului git, comanda git.exe va fi disponibila in orice terminal existent in windoes (Windows Terminal, cmd, etc.). Dar, acest pachet poate instala si un lansator numit "git bash" care ofera un mediu de tip unix (o consola care ruleaza bash si are la dispozitie comenzile Unix cunoscute, ls, cat, etc.). Se recomanda

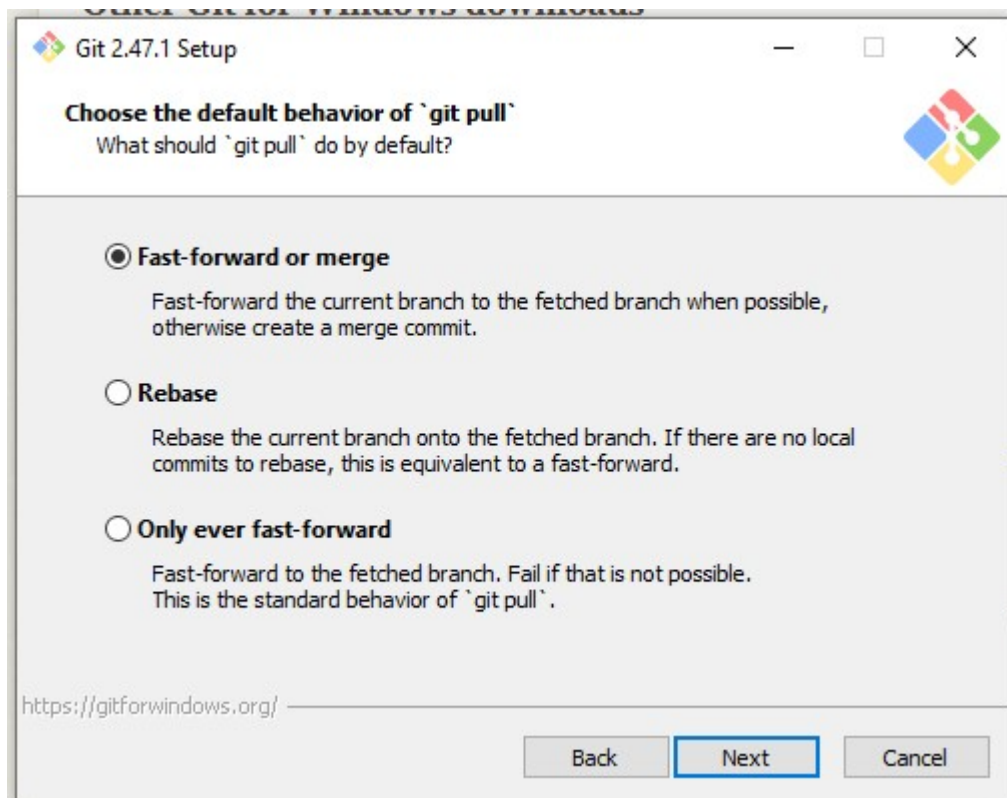
alegerea primei optiuni. Daca alegeti cea de-a doua optiune nu va exista "git bash" si veti putea apela git doar din consola normala a windows-ului.



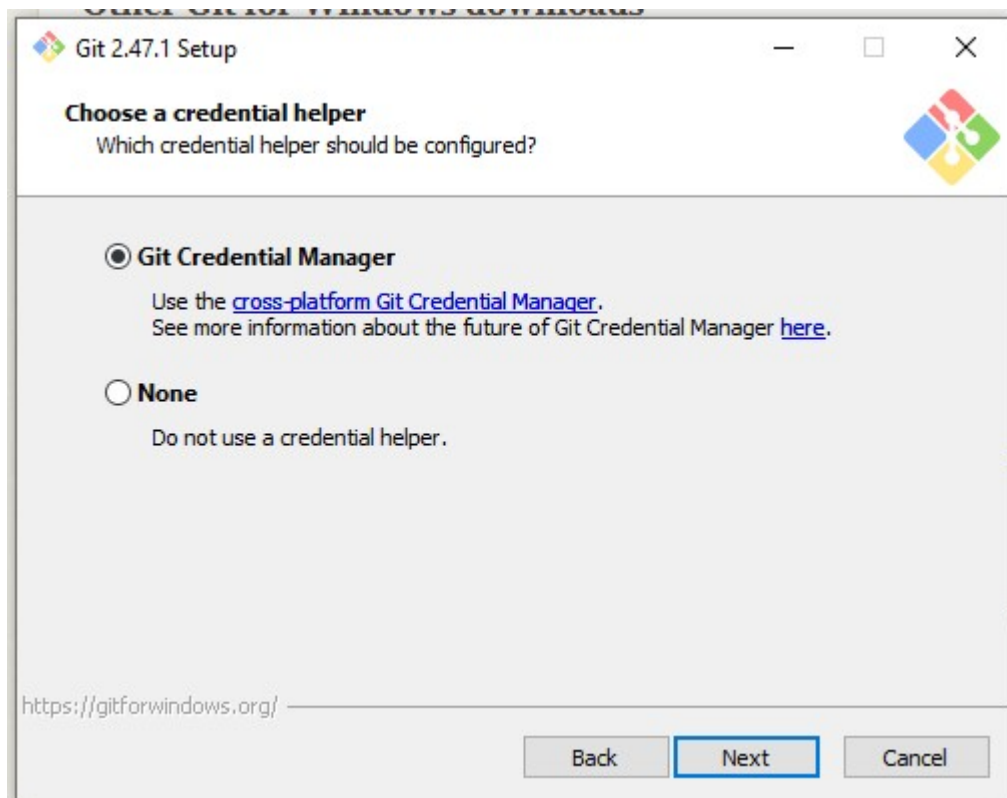
Urmatorul ecran permite selectarea comportamentului comun al comenzii "git pull". Comanda "git pull" aduce ultimele modificări de pe un repository de la distanță (de obicei, de pe un server sau GitHub) și le integrează în ramura curentă din repository-ul local. Practic, git pull combină două operațiuni: git fetch si git merge.

- Fast-forward or merge: Dacă ramura locala este identica cu cea de pe server, Git va face un fast-forward, adică va actualiza ramura locală direct la ultima versiune de pe server. Daca exista modificari, Git va actualiza ramura locală prin unirea modificarilor locale cu cele de pe server si va crea un commit de fuziune. Acest proces de fuziune poate esua, daca exista in acelasi timp modificari locale si pe server in acelasi fisier. In acest caz conflictele trebuie rezolvate manual.
- Rebase: Git va aduce modificările de pe server și le va aplica peste modificările tale locale, rescriind istoricul. Astfel, istoricul va fi mai curat și fără commituri de fuziune.
- Only ever fast-forward: Git va actualiza ramura locală doar dacă există posibilitatea unui fast-forward. Dacă există modificări locale care nu au fost trimise pe server, Git va refuza să facă un pull iar conflictul va trebui rezolvat manual (prin commit, merge sau rebase). Aceasta este de fapt comportamentul standard al comenzii "git pull"

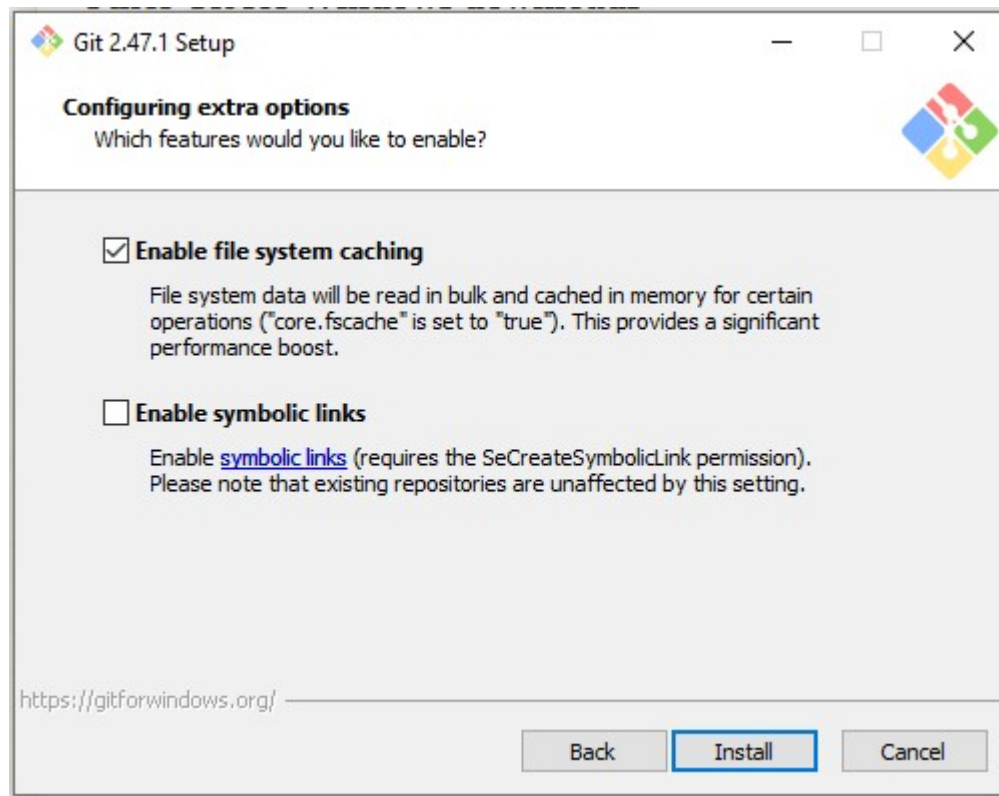
Se recomanda alegerea primei optiuni (selectata in mod predefinit).



Interactiunea dintre git si repository-urile aflate la distanta este autentificata. Git ofera un "credential manager" - un spatiu local, ca o baza de date, in care se pot retine credentialele diferitelor servere. Daca se alege cea de-a doua optiune (None) atunci credentialele (nume de utilizator, parola, token, etc.), vor trebui specificate la orice operatie care presupune interactiunea cu repository-ul remote (pull, fetch, etc.)



Urmatorul ecran poate fi lasat asa cum este. Anumite proiecte pot avea symlinkuri, in special daca sunt create pe sisteme de operare de tip Unix. Daca lucrati cu un altfel de proiect, e posibil ca optiunea "Enable symbolic links" sa fie utila dar trebuie retinut ca Windows solicita permisiuni speciale pentru crearea acestora pe care s-ar putea sa nu le aveti.



Windows terminal

Windows terminal este o aplicatie de tip consola semnificativ mai buna decat Command Prompt (consola predefinita din Windows, programul cunoscut ca cmd.exe).

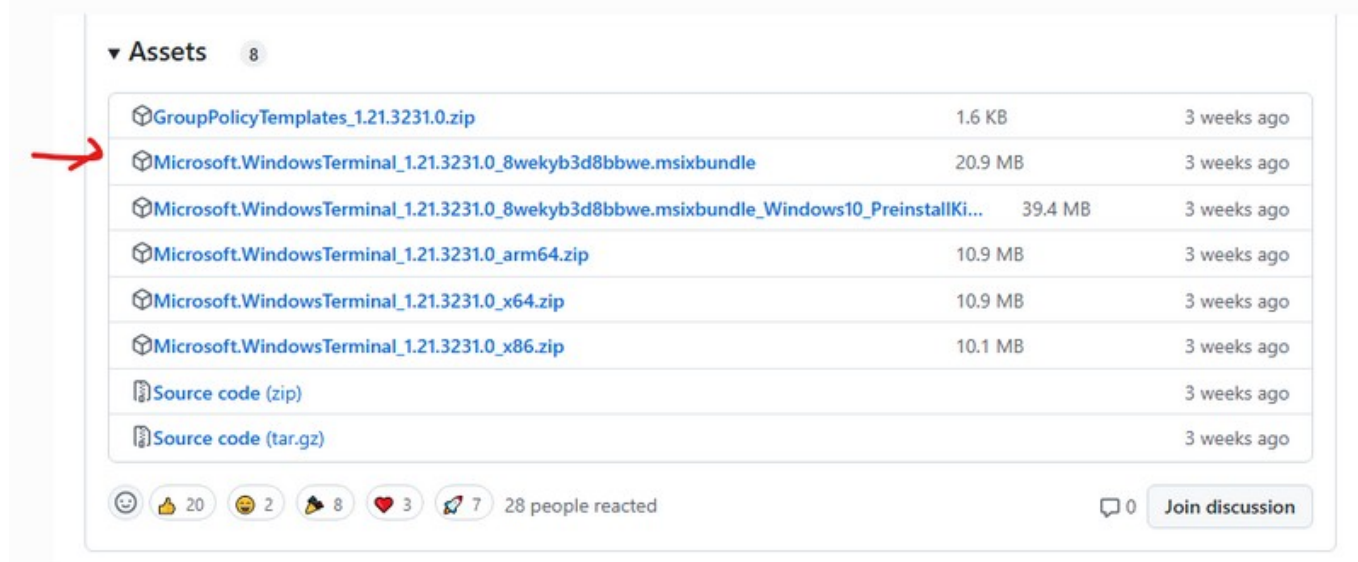
Windows Terminal este instalat odata cu sistemul de operare pe anumite versiuni de Windows 10 (22 H2) si pe toate variantele de Windows 11. In cazul in care nu exista, poate fi instalat direct din Microsoft Store dar, daca aceasta optiune nu este disponibila, puteti folosi urmatoarele variante:

Descarcarea directa din repository

Vizitati

<https://github.com/microsoft/terminal/releases>

Release-urile stabile au linkuri de descarcare in categoria "Assets". Pentru a obtine un pachet instalabil, alegeti pachetul al carui nume se termina cu "mxbundle"



Acest fisier poate fi lansat direct in executie dupa descarcare.

Alte variante

Alte variante, inclusiv folosind managerul de pachete chocolatey puteti gasi aici:

<https://woshub.com/install-windows-terminal/>

Scrcpy

Instalare

Scrcpy este un utilitar open-source care permite afisarea continua a ecranului unui dispozitiv mobil pe ecranul unui PC, cu ajutorul unei conexiuni USB debugging. Aceasta conexiune este oricum deschisa pentru instalarea aplicatiilor si procesul de dezvoltare. Cu ajutorul acestui instrument, presupunand ca exista un dispozitiv mobil hardware, se pot dezvolta aplicatii Android in conditii similare cu prezenta unui emulator intr-o masina virtuala.

Descarcati scrcpy de pe site. <https://scrcpy.org/>

Veti obtine o arhiva zip. Dezarhivati arhiva intr-un director oarecare. Puteti porni scrcpy direct din directorul dezarhivat sau puteti adauga si acest director la PATH sau puteti plasa un symlink pe desktop.

Utilizare

Odata stabilita conexiunea USB, programul poate fi lansat direct cu ajutorul comenzii scrcpy simplu sau insotita de optiuni

`scrcpy --help` afiseaza toate optiunile disponibile

`scrcpy --max-size 1024` reduce dimensiunea ferestrei afisate

`scrcpy --bit-rate 2M` reduce latimea de banda video

`scrcpy --record file.mp4` porneste inregistrarea ecranului

`scrcpy --stay-awake` mentine ecranul telefonului pornit

scrcpy --crop 1080:1920:0:0 afiseaza doar o portiune din ecranul telefonului

