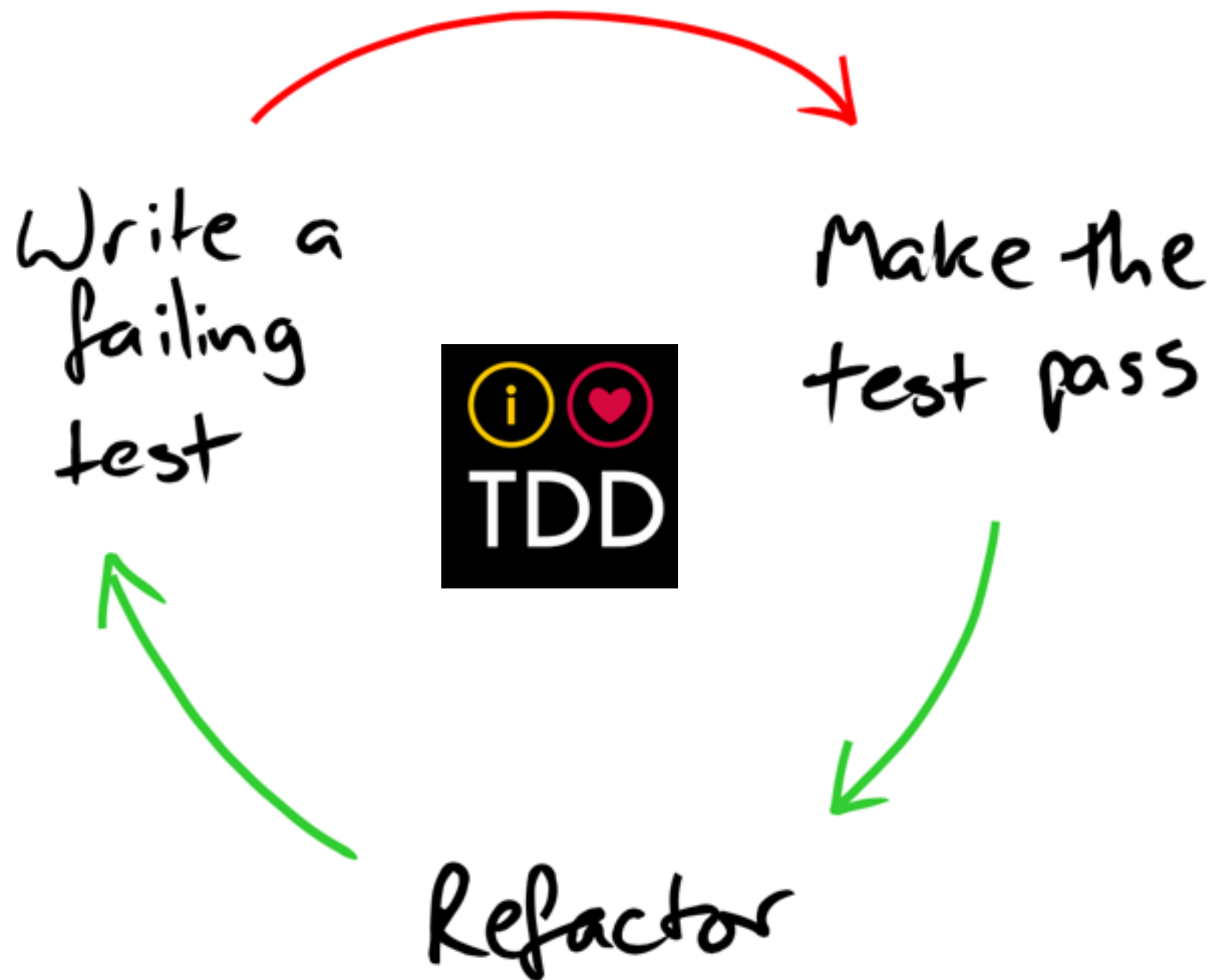


Continuous Integration



Problem

The screenshot shows the Google Play Console interface for an APK. The URL in the browser is <https://play.google.com/apps/publish/>. The page is titled "APK" and includes a "Switch to advanced mode" button. It features three main sections: "PRODUCTION" (Version 11009), "BETA TESTING" (Set up Beta testing for your app), and "ALPHA TESTING" (Set up Alpha testing for your app). Below these is the "PRODUCTION CONFIGURATION" section, which includes a button to "Upload new APK to Production" and a lightbulb icon. The "CURRENT APK" section shows it was published on 16 May 2014 at 03:06:02. A red box highlights the "Supported devices" section, which shows 3572 devices and a "See list" link. The "Excluded devices" section shows 0 devices and a "Manage excluded devices" link. At the bottom, a table lists the current production version.

VERSION	UPLOADED ON	STATUS	ACTIONS
11009 (1.1)	16 May 2014	In Prod	

Problem



What we need ?



Continuous Integration

Software development practice

What people do, not about what tools

Integrate frequently

Fast feedback



Jenkins



Jenkins

An extensible open source continuous integration server

[BLOG](#)[CONNECT](#)[BUG TRACKER](#)[WIKI](#)[CI](#)[TUTORIALS](#)[ARCHIVES](#)[DONATION](#)[ABOUT](#)

Jenkins User Conference 2015

Don't miss out on the latest from the Jenkins Community!

U.S. East - June 18-19

Europe - June 23-24

Israel - July 16

U.S. West - September 2-3

Early Bird Ends May 15th - REGISTER NOW



Meet Jenkins

Find out what Jenkins is and get started.



Use Jenkins

See how to get more out of your Jenkins.

Download Jenkins

[Release](#)[Long-Term Support Release](#)

Java Web Archive (.war)

Latest and greatest (1.614)

[changelog](#) | [past releases](#) | [RC](#)

upgrading from Hudson?

Or native package



Windows

Ubuntu/Debian

Building



Building Android app

Source code

JDK

Build tool

Android SDK

Building Android app

Source code



JDK

Build tool



Android SDK



Step to build

Source code

JDK

Build tool

Android SDK

Source code



GitHub



Install git

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



The image shows the Git website banner. At the top left is the Git logo (an orange diamond with a white branching diagram) followed by the text "git --distributed-is-the-new-centralized". To the right is a search bar with the placeholder text "Search entire site...". Below the logo, there are two paragraphs of text describing Git. To the right of the text is a diagram showing a network of nodes (stacks of papers) connected by colored lines (red, blue, yellow). Below the text, there is a section titled "Learn Git in your browser for free with Try Git." with a small icon of a cat. At the bottom, there are four sections: "About" (with a gear icon), "Documentation" (with a book icon), "Downloads" (with a download arrow icon), and "Community" (with a speech bubble icon). On the right side, there is a computer monitor displaying the latest source release "2.4.1" and a button "Downloads for Mac".

git --distributed-is-the-new-centralized

Search entire site...

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.

 **Learn Git in your browser for free with Try Git.**

About
The advantages of Git compared to other source control systems.

Documentation
Command reference pages, Pro Git book content, videos and other material.

Downloads
GUI clients and binary releases for all major platforms.

Community
Get involved! Bug reporting, mailing list, chat, development and more.

Latest source Release
2.4.1
Release Notes (2015-05-13)
[Downloads for Mac](#)

Baby step with git

\$git init

\$git status

\$git add

\$git commit

Create account

GitHub

[Explore](#) [Features](#) [Enterprise](#) [Blog](#) [Sign up](#) [Sign in](#)

Build software better, together.

Powerful collaboration, code review, and code management for open source and private projects. Need private repositories?
[Upgraded plans start at \\$7/mo.](#)

Use at least one lowercase letter, one numeral, and seven characters.

[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We will send you account related emails occasionally.

Working with github

\$git remote

\$git push

\$git pull

Let's start with

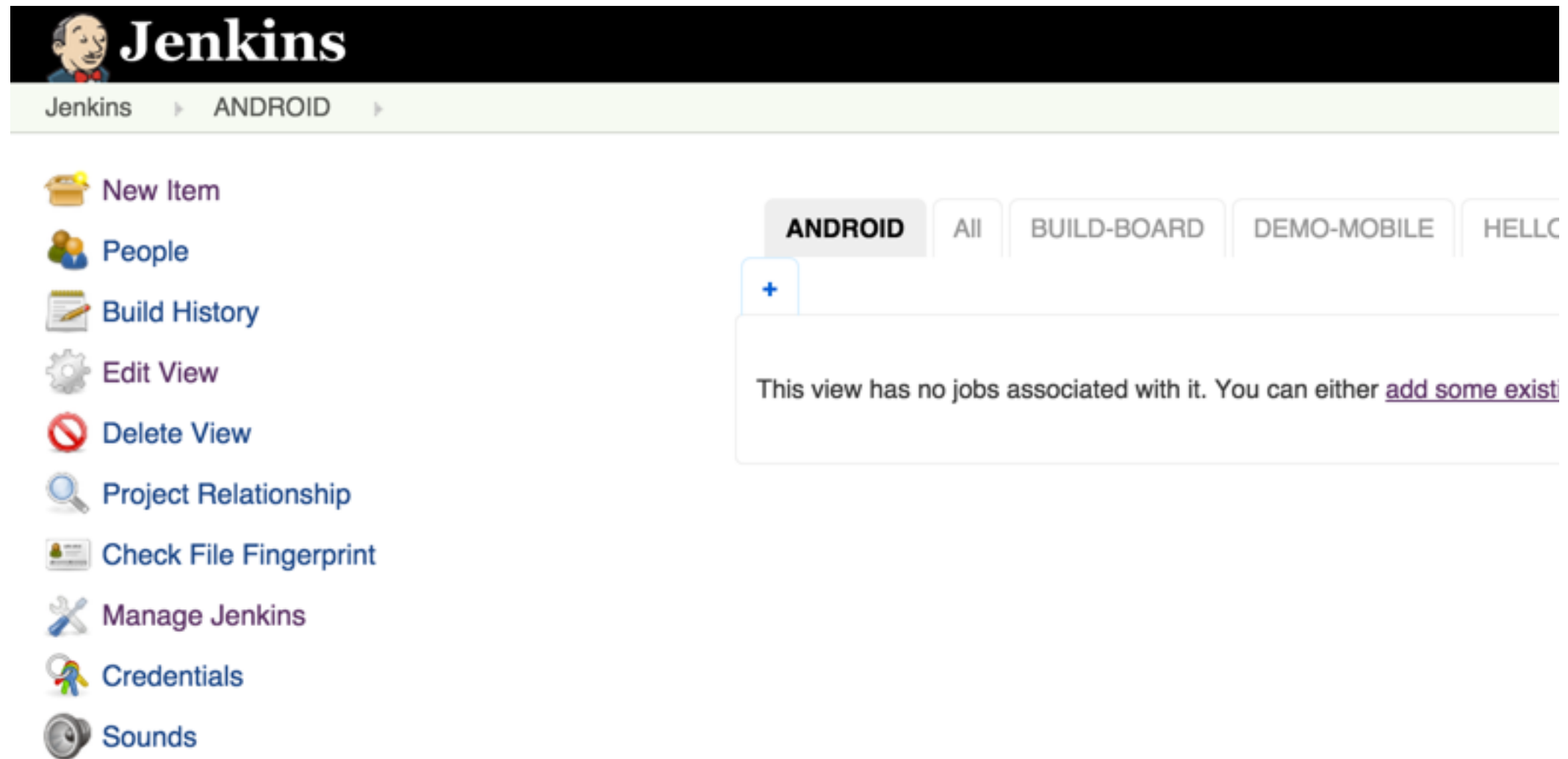


Jenkins


Start jenkins server

```
$java -jar jenkins.war
```


Goto localhost:8080





Create new job


 **Jenkins**


Jenkins > ANDROID >


 **New Item**


 People


 Build History


 Edit View


 Delete View

 Project Relationship

 Check File Fingerprint

 Manage Jenkins

 Credentials

 Sounds

Item name

☒ **Freestyle project**

This is the central feature of Jenkins. Jenkins will build your project for something other than software build.

☐ **Maven project**

Build a maven project. Jenkins takes advantage of your POM file.

☐ **External Job**

This type of job allows you to record the execution of a process you can use Jenkins as a dashboard of your existing automation.

☐ **Multi-configuration project**

Suitable for projects that need a large number of different configurations, etc.

☐ **Copy existing Item**

Copy from

Source control management

Source Code Management

- ☐ None
- ☐ CVS
- ☐ CVS Projectset
- ☒ Git

Repositories

Repository URL

git@github.com:up1/android_ci_demo.git

 **Please enter Git repository.**

Credentials

- none -

 Add

Add build step

Build

Execute shell

Command `./gradlew :App:connectedAndroidTest`

See [the list of available environment variables](#)

Add build step ▼

Config ANDROID_HOME

Windows

```
set ANDROID_HOME= <your path>
```

Mac

```
export ANDROID_HOME= <your path>
```

Add build step

Build

Execute shell

Command

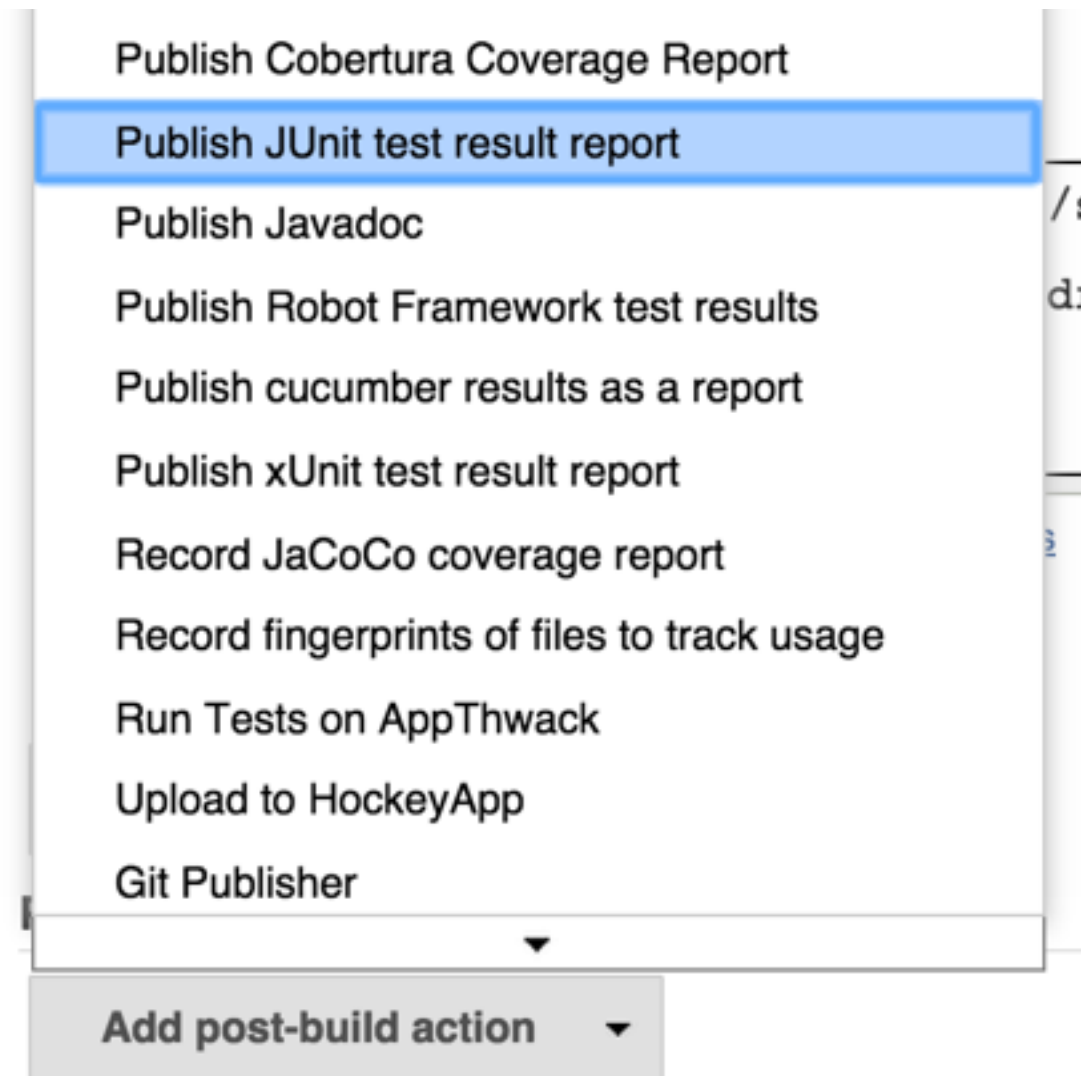
```
export ANDROID_HOME=/Users/somkiat/data/software/android-sdk-macosx  
./gradlew :App:connectedAndroidTest
```

See [the list of available environment variables](#)

Add build step ▼

Post build action

Generate report of each build



Publish JUnit test result report

Post-build Actions

Publish JUnit test result report

Test report XMLs

app/build/outputs/androidTest-results/connected/*.xml

[Fileset 'includes'](#) setting that specifies the generated raw XML report files, such as 'myr reports/*.xml'. Basedir of the fileset is [the workspace root](#).

☐ Retain long standard output/error

Health report amplification factor

1.0

1% failing tests scores as 99% health. 5% failing tests scores as 95%

See report

Project 01-APP-BUILD

 [add description](#)

[Disable Project](#)



[Workspace](#)



[Recent Changes](#)



[Latest Test Result](#) (no failures)

Test Result Trend



[\(just show failures\)](#) [enlarge](#)

Permalinks

- [Last build \(#7\), 43 sec ago](#)
- [Last stable build \(#7\), 43 sec ago](#)
- [Last successful build \(#7\), 43 sec ago](#)
- [Last failed build \(#4\), 13 min ago](#)
- [Last unsuccessful build \(#4\), 13 min ago](#)

Auto build ?

Build trigger => check every minute

Build Triggers

- ☐ Build after other projects are built
- ☐ Build periodically
- ☒ Poll SCM

Schedule

* * * * *

 Do you really mean "every minute" when you say "* * * * *"? Perhaps "* * * *" to poll once per hour

Ignore post-commit hooks ☐

Try it by yourself