



Build an Android App from Your Website

A complete worked example, from first sign-in
to a live listing on Google Play

No coding required · About 90 minutes of work
appforge.rumonz.com · Rumonz AI Studio
Support (WhatsApp): +880 1719 390400

About this guide

This guide follows one shop through the whole process. Every step shows exactly which button to press and what to type.

The example is an online coffee roaster. If you sell something else — clothes, electronics, furniture, services — the steps are identical. Substitute your own details wherever you see the example values.

The example business

Business	Northline Coffee — a small-batch coffee roaster
Website	https://northlinecoffee.com
Built with	WordPress and WooCommerce
What the site has	Product pages, a cart and checkout, customer accounts, downloadable brewing guides as PDFs, a blog, and a contact form
Why they want an app	Repeat orders. Their customers buy every few weeks, and a home screen icon plus push notifications brings people back far more reliably than email
Bundle ID they chose	com.northlinecoffee.app

This is an illustrative example

Northline Coffee is a fictional business created for this guide. Any resemblance to a real company is coincidental. The website address does not exist — use your own throughout.

The ten stages

Stage	What happens	Time
1 Check the website	Confirm the site is ready to become an app	15 min
2 Create an account	Sign up for AppForge	5 min
3 Create the project	Point AppForge at the website	5 min
4 Let AI configure it	Auto-fill name, colours and permissions	5 min
5 Set the identity	App name, Bundle ID, version, icon	10 min
6 Choose features	The eight that matter for a shop	20 min
7 Connect push	Set up Firebase notifications	15 min
8 Build	Compile the Android app	10 min
9 Store material	Screenshots, listing text, privacy	20 min
10 Submit	Upload to Google Play	30 min

What you need

Item	Where to get it	Cost
AppForge account	appforge.rumonz.com/register.php	Free for 14 days
Google Play developer account	play.google.com/console/signup	\$25, one time
Firebase project	console.firebase.google.com	Free
App icon	1024 × 1024 PNG, or let the AI create one	Free
A build machine	See stage 8 — shared hosting cannot compile	Varies

About the build machine

AppForge handles everything up to compiling: configuration, settings export, store material. Producing the actual app file needs Java and the Android SDK, which shared hosting does not provide. Stage 8 covers the options. Work through stages 1 to 7 first — none of it is wasted.

PART 1

Check the website

Fifteen minutes that prevent a rejection

An app built from a website inherits every problem that website has. Both stores reject apps that are simply a desktop site in a frame. Check these five things before going any further.

1.1 Open the site on a real phone

- 1 Open **https://northlinecoffee.com** in your phone browser — not a desktop browser made narrow.
- 2 Read a product page without pinching or zooming. If you cannot, the site is not mobile-friendly and must be fixed first.
- 3 Add something to the cart using only your thumb. Anything too small to tap reliably will be worse inside the app.
- 4 Scroll sideways. If the page moves horizontally, something is too wide — a common rejection reason.
- 5 Complete a test checkout. If payment fails on mobile web, it will fail in the app.

WooCommerce and Shopify users

Both are mobile-friendly out of the box, but themes and plugins can break that. Test the live site rather than assuming.

1.2 Confirm HTTPS

The address must start with **https://** and show a padlock. Android blocks insecure connections by default, so without it the app shows blank pages.

1.3 Find your privacy policy

Google Play will not let you publish without a publicly reachable privacy policy. Look in your site footer and write the address down:

`https://northlinecoffee.com/privacy-policy`

If the page does not exist, create one. AppForge generates a draft in stage 9, but it has to be published on your own website.

1.4 Decide which permissions you need

You will be asked which phone permissions the app requires. Tick only what the site genuinely uses. Here is how the example shop worked it out:

Does the site...	Permission needed	Northline Coffee	Your site
Let customers upload photos?	Camera, Photo library	No	_____

Does the site...	Permission needed	Northline Coffee	Your site
Show nearby shops or delivery estimates?	Location	No	_____
Record audio or video?	Microphone	No	_____
Have customer accounts?	Cookies kept on (default)	Yes	_____
Offer PDFs or documents?	Media downloader	Yes — brew guides	_____
Need to reach customers later?	Notifications	Yes	_____

So Northline Coffee needs only **Notifications**. Everything else stays off.

Fewer permissions is better

Every permission you request reduces the number of people who accept it, and gives store reviewers another reason to ask questions. If in doubt leave it off — you can add it in a later version.

1.5 List your external links

Links to sites you do not own should open in the phone browser, not inside your app. The example shop listed these:

- Payment provider — the checkout redirects to it
- Instagram and Facebook — links in the footer
- A shipping courier tracking page

Note yours now. You will configure them in stage 6.

PART 2

Create your AppForge account

Five minutes

- 1 Open **appforge.rumonz.com/register.php**
- 2 Enter your name and an email address you can open right now.
- 3 Choose **Company** and enter your business name.
- 4 Enter your mobile number with country code, for example +8801712345678.
- 5 Create a password: 12 or more characters with a capital, a small letter, a number and a symbol.
- 6 Tick the terms box and click **Create account**.
- 7 Open your inbox, find the AppForge message, and click the confirmation link. Check spam if it has not arrived in a few minutes.
- 8 Click **Sign in to AppForge** and sign in.

SCREENSHOT: Registration form
appforge.rumonz.com/register.php

You land on the Dashboard. A green bar at the top shows **14 days left of your Pro trial** — full access, no card needed.

If the confirmation email never arrives

Go to appforge.rumonz.com/resend-verification.php, enter your address and click **Send new link**. Still nothing? Use the green WhatsApp button on any page.

PART 3

Create the app project

Five minutes

- 1 In the left sidebar, under **Workspace**, click **Convert**.
- 2 For **Source type**, choose **Remote URL**. This keeps the app showing your live site, so adding a product to the website adds it to the app immediately.
- 3 For **App name**, type **Northline**. Keep it short — long names are shortened with dots under the icon.
- 4 For **Website URL**, type **https://northlinecoffee.com** including the https.
- 5 For **Stack**, choose **WordPress**. It only helps the AI make better guesses.
- 6 For **Category**, choose **Shopping**.
- 7 Click **Create App Project**.

SCREENSHOT: Convert page, form completed

Sidebar → Workspace → Convert

AppForge creates the project and opens **App config**.

Remote URL or HTML upload?

Remote URL suits almost everyone, and any shop in particular. Your stock, prices and products change constantly, and this way the app reflects them with no resubmission. **HTML upload** only makes sense if the app must work with no internet at all — which rules out a working checkout.

PART 4

Let AI configure it

Five minutes

Rather than filling in every field by hand, let AppForge read the site and propose settings.

- 1 In the sidebar, click **AI setup**.
- 2 Enter **https://northlinecoffee.com**
- 3 Click **Analyse & auto-configure** and wait a few seconds.
- 4 Read each suggestion. These are proposals — you decide what to keep.
- 5 Click **Apply all settings**, or change what you disagree with first.

SCREENSHOT: AI setup results

Sidebar → AI setup, after the analysis finishes

What it found for the example shop

Suggestion	What it proposed	What the owner did
App name	Northline Coffee	Shortened to Northline — the full name was cut off under the icon
Category	Shopping	Kept
Brand colour	#6B4423 — taken from the site header	Kept
Icon source	Found the site favicon at 64 × 64	Replaced it — far too small. Uploaded a proper 1024 × 1024 logo instead
Permissions	Notifications only	Kept — matches the list from stage 1.4
Mobile-ready	Responsive, viewport tag present	Good to continue

Then run the website audit

- 1 Click **Website audit** in the sidebar.
- 2 Enter the same address and run the check.
- 3 Work through anything it flags before you build.

The example shop had two findings: three product images over 2 MB, which made pages slow on mobile data, and a missing privacy policy link in the footer. Both were fixed on the website before continuing.

Fix the site, not the app

If the audit reports slow pages or a layout that scrolls sideways, fix it on the website. The app shows whatever the website shows — there is no way to paper over it from this side.

PART 5

Set the app identity

Ten minutes — one field is permanent

Open **App config** from the sidebar, or click **Config** next to the app in **My Apps**.

5.1 Identity fields

Field	Example value	How to choose yours
Display name	Northline	What appears under the icon. Aim for 12 characters or fewer.
Bundle ID	com.northlinecoffee.app	Permanent. Your domain reversed. Lower case, no spaces, no hyphens.
Version	1.0.0	What customers see. Increase for each release.
Build number	1	A whole number. Must increase on every upload to Play.
Category	Shopping	Used in your store listing.
Splash colour	#6B4423	Your brand colour. The preview updates as you type.
Show app name	Off	Their logo already contains the name.

The Bundle ID can never be changed

Once an app is published as com.northlinecoffee.app, that identifier belongs to it forever. Changing it later creates a completely separate app — your existing customers keep the old one and would have to find and install the new listing from scratch. Get it right the first time.

5.2 Platforms

Tick **Android**. Leave iOS unticked for now — you can add it later without redoing any of this work.

5.3 Permissions

Tick only what you decided in stage 1.4. For the example shop that is **Notifications** and nothing else.

If you do tick Camera or Location, write a specific reason in the field beneath. Vague wording is a known rejection cause:

Weak — likely questioned	Strong — explains the benefit
"We need camera access."	"Scan the code on your coffee bag to reorder it instantly."
"To improve your experience."	"Show cafés near you that stock our beans."

5.4 Upload the icon

- 1 Click **Asset uploads** in the sidebar.

- 2 Click **Upload** and choose a **1024 × 1024 PNG** of your logo.
- 3 Keep important detail away from the edges — Android may round the corners.
- 4 Use a solid background rather than transparency for the main icon.

Click **Save** at the top right of App config before moving on.

SCREENSHOT: App config Identity section with live phone preview
Sidebar → Native features → App config

PART 6

Choose the features

Twenty minutes — eight that matter for a shop

AppForge has 49 native features. A shop needs about eight. Configure these in order and leave the rest alone — every feature you enable is one more thing a reviewer can question.

6.1 URL handling

What it does. Keeps your pages inside the app and sends everyone else's to the browser.

- 1 Set the default to **Open in app**.
- 2 Add **northlinecoffee.com** to the list that stays inside the app.
- 3 Add the external links from stage 1.5 — your payment provider, Instagram, Facebook, courier tracking — and set each to **Open in system browser**.

Why it matters. This is a security setting as much as a usability one. Pages you do not control should never run inside your app. It also stops customers getting stuck on a payment page with no way back.

6.2 Cookie & cache

What it does. Keeps customers signed in and keeps the cart filled.

- 1 Turn **persist cookies** on.
- 2 Leave the caching strategy at its default.

Why it matters. Turn this off and customers are signed out, with an empty cart, every single time they close the app. For a shop this single setting is the difference between a usable app and an abandoned one.

6.3 Splash screen

What it does. The branded screen shown while your shop loads.

- 1 Turn it **on**.
- 2 Set the background to your brand colour.
- 3 Choose your logo image.
- 4 Turn on **Wait for page load**.

Why it matters. Waiting for the page looks far more polished than a fixed timer that expires before the shop has finished loading.

6.4 Pull-to-refresh

What it does. Drag down to reload.

- 1 Turn it **on**.
- 2 Leave the spinner style and threshold at their defaults.

Why it matters. Customers use this to check whether an out-of-stock item has returned. Without it the app feels broken.

6.5 Swipe gestures

What it does. Swipe from the screen edge to go back.

- 1 Turn **swipe right to go back** on.
- 2 Turn **swipe left to go forward** on.

Why it matters. Android users navigate this way constantly — browsing from a product back to a category and on to the next. Leaving it off is the fastest way to make an app feel wrong.

6.6 Offline & reconnect

What it does. What customers see with no internet.

- 1 Choose **Offline screen** as the fallback mode.
- 2 Set the title to "You are offline".
- 3 Set the message to "Check your connection and try again. Your cart is saved."
- 4 Turn **auto-reconnect** on.

Why it matters. Without this the app shows a raw browser error, which looks like your shop has crashed. Mentioning the saved cart stops people worrying they have lost an order.

6.7 Media downloader

What it does. Lets customers save PDFs and documents.

- 1 Turn it **on**.
- 2 Add the file types: pdf, doc, docx.

Why it matters. The example shop offers brewing guides as PDFs. Without this, tapping one does nothing at all and looks like a bug. Also needed if you email PDF invoices that customers open in the app.

6.8 Push notifications

What it does. Messages delivered even when the app is closed.

- 1 Turn it **on**.
- 2 Choose **Firestore** as the provider.
- 3 Leave the key blank for now — stage 7 covers it.
- 4 On the permission prompt tab, choose **ask after a delay** rather than on first launch.

Why it matters. For a shop this is the single most valuable feature. Order updates, restock alerts and new product announcements all bring people back. Asking permission the instant the app opens gets refused far more often than asking once someone has browsed a little and seen the value.

Save each page separately

Every feature page has its own **Save** button at the top right. Moving to another page without saving loses the changes on that page.

PART 7

Connect push notifications

Fifteen minutes

Push notifications need a free Firebase project. This is the only stage that takes you outside AppForge.

7.1 Create the Firebase project

- 1 Go to **console.firebase.google.com** and sign in with a Google account.
- 2 Click **Create a project**.
- 3 Name it after your business and click Continue.
- 4 Google Analytics is optional — you can turn it off.
- 5 Wait for the project to be created, then click Continue.

7.2 Add the Android app

- 1 On the project overview, click the **Android** icon.
- 2 For **Android package name**, enter **com.northlinecoffee.app** — this must match the Bundle ID from stage 5 exactly, character for character.
- 3 Nickname is optional. Leave the SHA-1 field blank for now.
- 4 Click **Register app**.
- 5 Download **google-services.json** and keep it somewhere safe.
- 6 Skip the remaining SDK steps — AppForge handles those.

7.3 Get the server key

- 1 In Firebase, click the gear icon → **Project settings**.
- 2 Open the **Cloud Messaging** tab.
- 3 Copy the **Server key**.

7.4 Put it into AppForge

- 1 Back in AppForge, open **App config**.
- 2 Scroll to **Push basics**.
- 3 Paste the key into **FCM key**.
- 4 Set a default notification title and body for testing.
- 5 Click **Save**.

- 6 Open **Push notifications** in the sidebar and upload **google-services.json** if asked.

The package name must match exactly

If Firebase has `com.northlinecoffee.app` and AppForge has `com.northline.app`, notifications will silently never arrive — no error, nothing in the logs. Check both, character by character.

7.5 What to actually send

Useful notifications for a shop, in rough order of value:

Notification	When	Why it works
Order confirmed	Immediately after checkout	Expected, reassuring, always opened
Order shipped	When you dispatch	The most opened message a shop sends
Back in stock	When a watched item returns	High intent — they already wanted it
New product	Sparingly, for genuine launches	Works until it becomes noise
Cart reminder	A day after abandonment	Effective, but only once

Do not overdo it

Daily promotional notifications are the fastest way to get your app deleted. Order updates are always welcome; marketing messages are tolerated at about one a week.

PART 8

Build the app

Ten minutes, once a runner is connected

8.1 Check the settings file first

- 1 Click **Runtime export** in the sidebar.
- 2 Confirm the website address, app name and Bundle ID are all correct.
- 3 Confirm the eight features you enabled appear as enabled.

This is the file the app actually reads at runtime. Two minutes here saves a wasted build.

8.2 Start the build

- 1 Open **My Apps**.
- 2 Find your app and click **Build**.
- 3 Choose **Android**.
- 4 The build appears in **Builds** as **Queued**, then **Building**.
- 5 When it finishes, click **Logs** to see the result and download the file.

8.3 If it says "Runner missing"

This means no build machine is connected. Shared hosting cannot compile Android apps — there is no Java and no Android SDK. Your options:

Option	What it involves	Cost
Small VPS	A basic Linux server with Java and the Android SDK installed. Most control.	From about \$5/month
GitHub Actions	Free build minutes. AppForge exports the project and Actions compiles it.	Free tier available
Codemagic	Purpose-built for mobile apps. Handles Android and iOS.	Free tier available
Ask support	We can connect a shared build runner for you.	Contact us

Nothing is wasted

Everything through stage 7 is saved. When a build machine is connected, your app compiles from the configuration you already completed. Message support on WhatsApp and we will help you get a runner working.

8.4 If the build fails

- 1 Open the build and click **Logs**.
- 2 Read the last few lines — the reason is usually stated plainly there.
- 3 Click **Copy** to copy the whole log.
- 4 Open **AI build debugger**, paste it, and click **Analyse** for a plain-English explanation.
- 5 If it is still unclear, send the log through the WhatsApp support button.

PART 9

Prepare the store material

Twenty minutes

9.1 Write the listing

- 1 Click **AI store assistant** in the sidebar.
- 2 Choose your app and click generate.
- 3 It produces a title, short description, full description and keywords.
- 4 Edit anything that does not sound like your business.

Field	Limit	Advice
App title	30 characters	Your brand, plus one descriptive word. "Northline Coffee" fits; "Northline Coffee Roasters Online Shop" does not.
Short description	80 characters	The first thing people read. Lead with the benefit: "Fresh coffee, delivered. Reorder in two taps."
Full description	4000 characters	The first two lines matter most — the rest is hidden behind "more".

9.2 Screenshots

Google Play requires at least two phone screenshots, 1080 × 1920 or larger.

- 1 Open **Store submission** from the sidebar.
- 2 Use the screenshot generator, or capture your own from a phone.
- 3 For a shop, show: the home page, a product page, the cart, and the account area.
- 4 Upload them in the Screenshots section.

Show the product, not the interface

Screenshots that show attractive products convert better than screenshots of menus and buttons. Lead with what you sell.

9.3 Privacy declarations

Google requires a Data Safety declaration. Getting it wrong is a common rejection, and shops collect more data than most — names, addresses, payment details.

- 1 Click **AI privacy pack** in the sidebar.
- 2 Generate the pack. It produces a Play Data Safety file and a draft privacy policy.

- 3 Read the declaration and check it matches what your shop actually collects.
- 4 Publish the privacy policy on your website if you do not already have one.
- 5 Note the address — Play asks for it.

You are responsible for accuracy

The generated declaration is a strong starting point, not legal advice. If your checkout collects something the declaration does not mention, correct it before submitting.

9.4 Final risk check

- 1 Click **AI rejection risk scanner**.
- 2 Run the scan on your app.
- 3 Fix every high-risk item before submitting.

The example shop had one finding: the payment provider link was set to open inside the app rather than the browser. Fixed in URL handling in about a minute.

Ten minutes now, or ten days later

Fixing an issue here takes minutes. A Play rejection costs days of waiting and a full resubmission.

PART 10

Submit to Google Play

Thirty minutes, then a wait

10.1 Set up signing

Every Android app must be signed. The keystore file proves that future updates come from you and nobody else.

- 1 Open **Android release** in the sidebar.
- 2 Create a keystore, or upload one you already have.
- 3 Set the output format to **AAB** — Google Play requires it for new apps.
- 4 **Download the keystore file and back it up in two separate places.**

Losing the keystore is permanent

If you lose this file you can never update your app again. Not with support, not with Google. You would have to publish a brand new listing and lose every existing customer, review and rating. Back it up now, in two places.

10.2 Create the Play listing

- 1 Go to **play.google.com/console** and sign in.
- 2 Click **Create app**.
- 3 Enter the app name and default language, then choose **App** and **Free**.
- 4 Accept the declarations and click Create.

"Free" refers to the download

Choosing Free means the app costs nothing to install. You can still sell physical products through it — that is not an in-app purchase. Only digital goods consumed inside the app require Google's billing system.

10.3 Fill in the listing

- 1 Open **Main store listing** and paste the text from stage 9.1.
- 2 Upload your icon (512 × 512), feature graphic (1024 × 500) and screenshots.
- 3 Open **App content** and complete the privacy policy address, Data Safety form and content rating questionnaire.
- 4 Open **Store settings** and set the category and contact details.

10.4 Upload and release

- 1 Open **Production** → **Create new release**.
- 2 Upload the **.aab** file from your build.
- 3 Enter release notes — the AI store assistant generated these.
- 4 Click **Review release**, then **Start rollout to Production**.

Google review usually takes a few hours to a few days for a first submission. You are emailed either way.

Test internally first

Instead of going straight to Production, use **Internal testing** and add your own email address. The app reaches your phone within minutes and you can place a real test order before any customer sees it.

10.5 After approval

- Place a real order through the app to confirm checkout works end to end.
- Send a test push notification from AppForge.
- Check **Analytics** in AppForge after a few days of real use.
- Use **OTA updates** for urgent fixes without waiting for store review.
- When you add products or change prices on the website, the app updates automatically — no resubmission. Only configuration changes require a new build.

Checklist

Print this page and tick as you go.

Before you start

☐	Website opens and reads correctly on a phone
☐	Test checkout completed successfully on mobile
☐	Site uses https with a padlock
☐	Privacy policy page exists and is public
☐	Decided which permissions the site actually needs
☐	Listed external links that should open in the browser

In AppForge

☐	Account created and email confirmed
☐	App project created with the correct website address
☐	AI setup run and suggestions reviewed
☐	Website audit run and issues fixed
☐	Bundle ID set correctly — checked twice
☐	Version and build number set
☐	Android platform ticked
☐	Only necessary permissions ticked, each with a clear reason
☐	Icon uploaded at 1024 × 1024
☐	URL handling configured, external links sent to browser
☐	Cookies set to persist
☐	Splash screen configured
☐	Pull-to-refresh on
☐	Swipe gestures on
☐	Offline screen configured
☐	Media downloader configured
☐	Push notifications enabled

Firebase

- | | |
|---|---|
| ■ | Project created |
| ■ | Android app registered with the matching package name |
| ■ | google-services.json downloaded |
| ■ | Server key pasted into AppForge |

Build and store

- | | |
|---|---|
| ■ | Runtime export checked |
| ■ | Build completed successfully |
| ■ | Store listing text written and edited |
| ■ | Screenshots uploaded |
| ■ | Privacy pack generated and verified against your checkout |
| ■ | Risk scanner run, all high-risk items fixed |
| ■ | Keystore created and backed up in two places |
| ■ | AAB uploaded to Play Console |
| ■ | Data Safety form completed |
| ■ | Content rating completed |
| ■ | Released to internal testing first |
| ■ | Real test order placed through the app |
| ■ | Released to production |

Stuck at any point?

Use the green Support button on any AppForge page, or message **+880 1719 390400** on WhatsApp. Tell us which stage you are on and what you see on screen — it gets you an answer faster.

AppForge Android Tutorial · Rumonz AI Studio · appforge.rumonz.com · Northline Coffee is a fictional example