SAMPLE REPORT · SHOWAH CO.

Spring Boot Upgrade Readiness Report

Application: Spring PetClinic, commit `a5cbb85` (16 Oct 2022) · From Spring Boot 2.7.3 · To Spring Boot 4.1 · Prepared 9 Nov 2026 by Jeff Showah

This sample uses the Spring team's public, Apache-2.0 demo app, not a customer's code. Your report follows the same sections, from your build file and dependency tree.

1. **Ready to start, nothing blocks the first step.** The app can move to the last 2.7 release and Java 17 this week without touching application code.
2. **Seven changes will stop the build on the way to 4.1.** The largest is 50 `javax` imports across the entities, validators and two XML-mapped classes. All seven have a known, mechanical fix.
3. **The risk is in six behaviour changes that compile cleanly.** Hibernate moves from 5.6 to 7.4, Jackson from 2 to 3, Tomcat from 9 to 11. Section 5 lists what to test.

Estimated effort for this app: 2 to 4 working days for one developer who knows the code. It is an estimate, explained in section 6.

1. Where the app stands

ITEM	TODAY	WHY IT MATTERS
Spring Boot	2.7.3	Free support for the 2.7 line ended 30 Jun 2023. The last release was 2.7.18 (23 Nov 2023).
Java level	1.8 (Maven), 11 (Gradle)	The two build files disagree. Spring Boot 3.0 and later need Java 17.
Spring Framework	5.3.22	4.1 manages 7.0.8.
Hibernate	5.6.10.Final	4.1 manages 7.4.1.Final: two major versions.
Jackson	2.13.3	4.1 prefers Jackson 3 (3.1.4), with new package names.
Tomcat	9.0.65	4.1 manages 11.0.22.
Thymeleaf	3.0.15	4.1 manages 3.1.5.
JUnit / Mockito	5.8.2 / 4.5.1	4.1 manages JUnit 6.0.3 and Mockito 5.23.0.
Database drivers	H2 2.1.214, MySQL 8.0.30, PostgreSQL 42.3.6	4.1 manages H2 2.4.240, MySQL 9.7.0, PostgreSQL 42.7.11.
Code size	23 main files, 11 test files	Small. JUnit 5 only, so no JUnit 4 migration.
Security	No Spring Security	Removes the Spring Security 5.8, 6 and 7 migrations from the path.

Sources: the app's `pom.xml`, `build.gradle` and sources at the commit above; the published `spring-boot-dependencies` POMs for 2.7.3 and 4.1.0 on Maven Central; endoflife.date/spring-boot (updated 23 Oct 2026).

2. Support dates

LINE	RELEASED	FREE SUPPORT ENDS	JAVA
2.7 (you are here)	31 May 2022	30 Jun 2023 (ended)	8 to 21
3.5 (last 3.x line)	31 May 2025	30 Jun 2026 (ended)	17 to 25
4.0	30 Nov 2025	31 Dec 2026	17 to 25
4.1 (target)	30 Jun 2026	31 Jul 2027	17 to 26

Stopping at 3.5 does not get this app back into free support. 4.1 is the only line with more than two months of it left. Paid support exists for 2.7 (to 30 Jun 2029) and 3.5 (to 30 Jun 2032) if the upgrade has to wait.

Source: endoflife.date/spring-boot, read 9 Nov 2026.

3. The upgrade path

1. **2.7.3 to 2.7.18, on Java 17.** The Spring Boot 3.0 guide says to move to the latest 2.7 release first. Set `java.version` to 17 in `pom.xml` and `sourceCompatibility` to 17 in `build.gradle` so the two builds agree. 2.7 runs on Java 17, so this step changes no application code. Ship it on its own.
2. **2.7.18 to 3.5.x.** This is the Jakarta step: blockers B1 to B5 below. Add `spring-boot-properties-migrator` while you work and remove it after; it reports renamed properties at startup. Run the full test suite and the checks in section 5 before going on.
3. **3.5.x to 4.0.x.** The Spring Boot 4.0 guide says to start from the latest 3.5 release with no calls to deprecated methods. Blockers B6 and B7 land here. The guide's suggested route is to add the classic starters first, fix imports, then swap to the new focused starters.
4. **4.0.x to 4.1.x.** The 4.1 release notes list nothing that touches this app's code. The Spring team's own upgrade of this app to 4.1 changed build files and test setup only.

4. Blockers, ranked

Each of these stops the build or the tests. Ranked by how much of the code it touches.

	WHAT BREAKS	WHERE IN THIS APP	FIX
B1 · 3.0	`javax` packages become `jakarta` (Jakarta EE 10).	50 imports: `javax.persistence` 38, `javax.validation` 9, `javax.xml.bind` 3. The Spring team's own migration commit touched 15 files.	Rename the imports. The official guide lists OpenRewrite and IntelliJ tooling that does it. Leave `javax.cache` alone: JCache keeps its name.

B2 · 3.0	Ehcache needs the `jakarta` classifier.	`org.ehcache:ehcache` in both build files, declared without one.	Add `<classifier>jakarta</classifier>` (Gradle: `:jakarta`). The Spring team made this exact change.
B3 · 3.0	MySQL driver coordinates changed.	`mysql:mysql-connector-java` in both build files.	Change to `com.mysql:mysql-connector-j`.
B4 · 3.0	Git Commit ID plugin coordinates changed.	`pl.project13.maven:git-commit-id-plugin` in `pom.xml`.	Change to `io.github.git-commit-id:git-commit-id-maven-plugin`.
B5 · 3.0	The XML binding annotations need an explicit API dependency.	`vet/Vet.java` and `vet/Vets.java` use `@XmlElement` and `@XmlRootElement`.	Add `jakarta.xml.bind:jakarta.xml.bind-api` (version managed by Spring Boot). The Spring team added it during their migration.
B6 · 4.0	`@MockBean` is removed.	4 fields in 4 controller test classes (8 mentions counting imports).	Replace with `@MockitoBean` from Spring Framework (package `…test.context.bean.override.mockito`). It is deprecated from 3.4, so do this while still on 3.5.
B7 · 4.0	Spring Boot is split into modules; classes moved packages and tests need their own starters.	Five classes: the JCache customizer, `@WebMvcTest` (5 tests), `@DataJpaTest`, the test-database annotation and `RestTemplateBuilder`.	New packages, e.g. `org.springframework.boot.webmvc.test.autoconfigure` for `@WebMvcTest`. Add the matching test starters (`-webmvc-test`, `-data-jpa-test`) and `spring-boot-starter-restclient`. Rename `spring-boot-starter-web` to `-webmvc`.

Build plugins pinned by hand

These versions are set in `pom.xml`, so Spring Boot will not move them for you.

- **JaCoCo 0.8.7** has only experimental Java 17 support. Official support starts at 0.8.8, Java 21 at 0.8.11, Java 25 at 0.8.14. Raise it in step 1.
- **Checkstyle 8.45.1** with plugin 3.1.2, **spring-javaformat 0.0.31**, **libsass plugin 0.2.26**: all from 2021 or 2022. The Spring team's current build uses Checkstyle 12.3.1 with plugin 3.6.0, spring-javaformat 0.0.47 and libsass 0.3.4. I did not test the old versions on Java 17; expect to raise them.
- **Snapshot and milestone repositories** are declared for both dependencies and plugins. A release build does not need them. The Spring team removed them.

5. Behaviour changes to test

These compile and may pass your tests. Check each one by hand or add a test before you ship.

	CHANGE	WHAT TO CHECK IN THIS APP
T1 · 3.0	A trailing slash no longer matches. `GET /owners/` returned the page on 2.7 and returns 404 on 3.0 and later.	17 mapped routes. Check bookmarks, proxy rules, health checks and any client that adds a slash.
T2 · 3.0 to 4.1	Hibernate 5.6 to 6.1 (Boot 3.0), 6.2 (3.1), then 7.4 (4.1).	The Hibernate 6.0 guide changes default mappings for `UUID`, `Duration` and `Instant` and the default sequence naming. This app uses none of those types and uses `IDENTITY` ids, so the known traps miss it. Still run the suite against MySQL and PostgreSQL, not only H2. Two `@OrderBy` clauses and three `EAGER` collections are worth a look in the SQL log.
T3 · 4.0	Jackson 3 becomes the default JSON library, with different defaults.	`GET /vets` returns JSON. Compare the response body before and after, byte for byte. `spring.jackson.use-jackson2-defaults=true` restores the old defaults if a client depends on them.
T4 · 3.0, 3.5, 4.0	Actuator: only `health` is exposed over JMX (3.0); `heapdump` defaults to no access (3.5); liveness and readiness groups are on by default (4.0).	This app sets `management.endpoints.web.exposure.include=*`. Confirm what is reachable after the upgrade, and decide whether `*` is still what you want.
T5 · 3.0, 3.4	Log timestamps switch to ISO-8601 (3.0). Graceful shutdown is on by default (3.4).	Anything that parses the log files. Deploy scripts that expect the process to stop at once.
T6 · 3.2	Spring no longer reads parameter names from bytecode.	3 handler parameters have no explicit name (one `@PathVariable`, two `@RequestParam`). They work only if both builds compile with `-parameters`. Confirm, or name them.

Also check every page renders after the Thymeleaf 3.0 to 3.1 move (12 templates), and that DevTools live reload is off by default from 4.0 if your team relies on it.

6. Effort estimate

2 to 4 working days for one developer who knows this code. This is my estimate, not a quote and not a measured result.

STEP	ESTIMATE	BASIS
1. Latest 2.7 on Java 17	2 to 4 hours	Two version lines and the plugin bumps.
2. To 3.5	1 to 2 days	50 imports are mechanical. The time goes to the section 5 checks on three databases.
3. To 4.0	0.5 to 1.5 days	Five classes move package, four mocks change, test starters are added, plus the JSON comparison.
4. To 4.1	1 to 2 hours	Version bump and a full test run.

A larger app scales mainly with three things this report counts for you: the number of `javax` imports, the libraries Spring Boot does not manage, and how much of the behaviour in section 5 your tests already cover.

7. What was not checked

- **Nothing was built or run.** This is a read of the build files, the dependency versions and the source. No compile, no tests, no startup.
- **The database servers.** `docker-compose.yml` uses MySQL 5.7 and PostgreSQL 14.1. I did not check them against the 9.7.0 and 42.7.11 drivers or Hibernate 7.4. Verify before step 3.
- **The Hibernate 6.1, 6.2 and 7.x migration guides** were not read line by line, only 6.0. The Thymeleaf 3.1 and Jackson 3 change lists were not read; T3 and the template check cover them by testing.
- **The webjars** (Bootstrap 5.1.3, Font Awesome 4.7.0) and the SCSS build were not reviewed.
- **Security advisories** for the current dependency versions were not looked up. That is a different report.

Sources

Spring Boot 3.0 Migration Guide and Spring Boot 4.0 Migration Guide (github.com/spring-projects/spring-boot/wiki) · Spring Boot 3.1, 3.2, 3.3, 3.4, 3.5 and 4.1 Release Notes (same wiki) · Hibernate ORM 6.0 Migration Guide (docs.hibernate.org) · JaCoCo change history (jacoco.org) · spring-boot-dependencies 2.7.3 and 4.1.0 POMs (repo1.maven.org) · endoflife.date/spring-boot · the upgrade commits in github.com/spring-projects/spring-petclinic (`1315cf6`, `1e56f6f`, `3aa79e3`, `37d5315`, `9a5d50c`). All read 9 Nov 2026.

Advisory only, with no warranty. This report is one engineer's reading of your build against the published upgrade guides. It does not replace compiling, testing and reviewing the upgrade yourself. Spring, Spring Boot and Spring PetClinic are the work of their authors; Showah Co. is not affiliated with them. Showah Co. · showah.co