

[Open in app ↗](#)

34



## AI-Driven

Member-only story

# Migrating a Spring Boot 2.x project using Claude Code

Claude Code: a new approach for AI-assisted coding



Saeed Zarinfam

[Follow](#)

9 min read · Mar 30, 2025

64

2



...

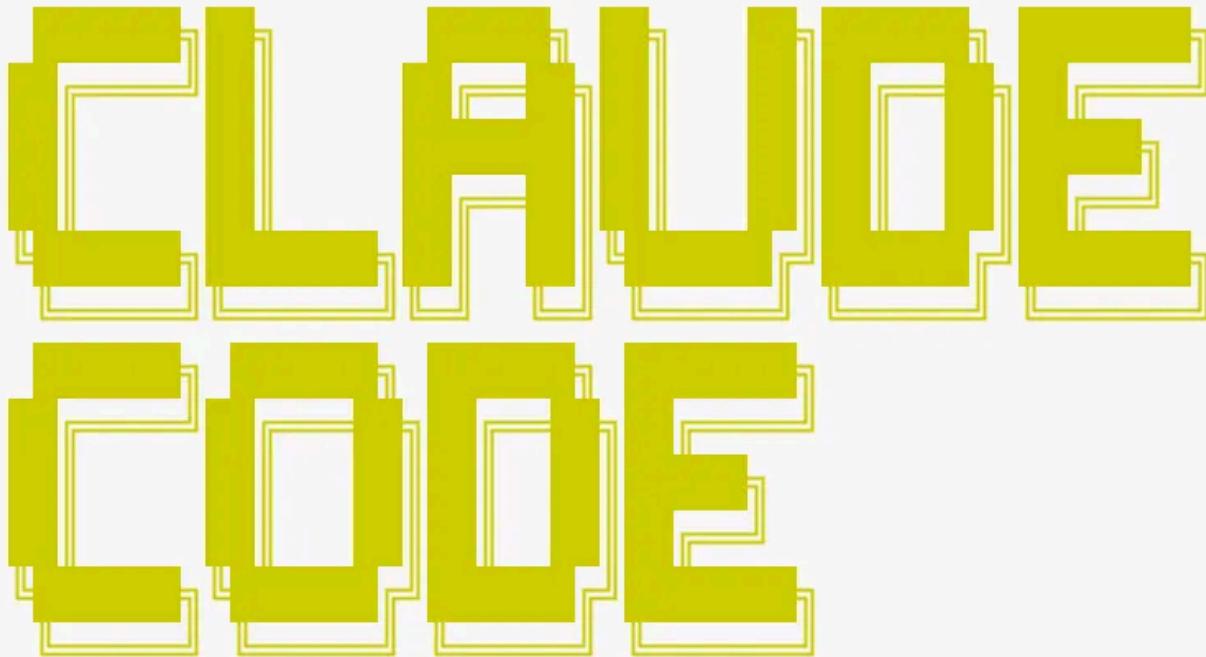
Almost all of us know Anthropics models (Claude Sonnet) as the best LLM for coding assistance and code generation. On the other hand, many developers have started using AI-integrated IDEs such as Cursor, Windsurf, or VS Code with GitHub Copilot and chose Claude Sonnet models as the underlying model in these agentic tools for reasoning and code generation.

Now, Anthropic has released a new AI-power tool called Claude Code, which uses Anthropic models but a different way to help developers with coding. In this article, I will review and examine this new agentic tool from Anthropic. First, I will compare its approach with other agentic tools and then see how Dev Container makes its installation easy, finally, I will talk

about my experience with **migrating** an old Spring Boot 2.x to the latest version.

Not a Medium member? [Read this article for free.](#)

\* Welcome to **Claude Code** research preview!



Claude Code banner in terminal

- [Claude Code: A terminal-based AI-assisted coding](#)
- [How does Claude Code work?](#)
- [Clean and straightforward installation using Dev Containers](#)
- [Migrating a Spring Boot 2.x project using Claude Code](#)
- [1 First prompt: Migrate the service from Spring Boot 2.3.1 and Java 8 to Spring Boot 3.4.4 and Java 17](#)
- [2 Second prompt: creating a docker image using Jib](#)
- [Test the result of refactoring](#)

- 3 Third prompt: Run the skaffold command from the parent folder
- 4 Forth prompt: Start a new round of refactoring
- Final Thoughts

## Claude Code: A terminal-based AI-assisted coding

I have written several articles about AI-assisted coding before and tried to examine them as a Java developer to implement practical examples:

### **Implementing a Spring Boot service using Windsurf and Claude**

Can AI agents take over programmers' jobs?

itnext.io

### **Part 5: Implementing a Web UI using Vaadin and GitHub Copilot Agent Mode**

Why LLMs are not suitable for lesser-known programming languages and frameworks

itnext.io

**Cursor**, **Windsurf**, or **VS Code with GitHub Copilot** define a new standard in AI-assisted coding by introducing AI-integrated IDEs that offer more traditional code editing experiences enhanced by AI inside an IDE.

**Claude Code**, introduced by Anthropic, takes a different approach.

Instead of operating inside the IDE, this agentic coding tool operates primarily through the terminal.

In the beginning, it seems weird to have **AI-assisted coding in the terminal**, but it has several advantages over other approaches. For example, we have a **simple setup** process without the need to install additional tools and IDE plugins.

Since Claude Code lives in the terminal and does not need a GUI, we can install it everywhere that is accessible from the terminal.

As you might know, Anthropic introduced the **MCP protocol**, Which enables LLMs to access external tools and data sources.

Configuring and using MCP servers in Claude Code is fairly simple and handy.

Another interesting feature of Claude Code for me is creating **git commits and PRs** and resolving **merge conflicts**.

 Bear in mind that Claude Code is in beta at the time this article is written and introduced as a **research preview** to gather developer feedback.

## How does Claude Code work?

Claude Code can easily be installed as an NPM package on **macOS 10.15+**, **Ubuntu 20.04+/Debian 10+**, or Windows via WSL and directly connect to Anthropic's API without intermediate servers.

According to its official website, Claude Code's key features are:

- *Editing files and fixing bugs across your codebase*
- *Answering questions about your code's architecture and logic*
- *Executing and fixing tests, linting, and other commands*
- *Searching through git history, resolving merge conflicts, and creating commits and PRs*

## Clean and straightforward installation using Dev Containers

Another approach to installing Claude Code that I liked very much was using Dev Containers configs provided by Anthropic. Dev Containers is one of the dominant standards for DEaC (Development Environments as Code), which I wrote about in my previous articles:

### From IaC to DEaC (Development Environments as Code)

Compare 4 dominant standards for Development Environments as Code

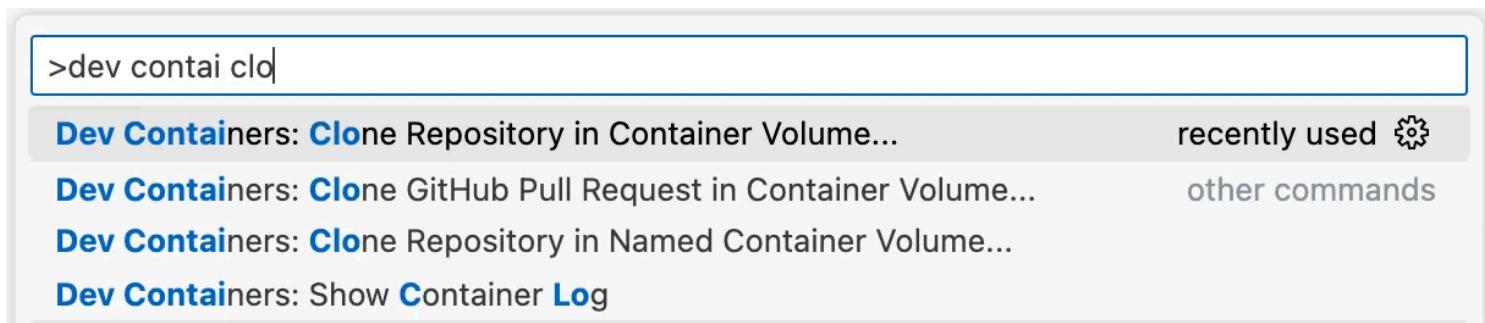
itnext.io

Using the Dev Container approach enables us to have a consistent, reproducible, and secure environment inside a container without installing extra software and config on our development machine.

Both VS Code and JetBrains IDEs support Dev Container standard. Through this article, I will use the [VS Code Dev Container extension](#) to install Claude Code using Dev Containers.

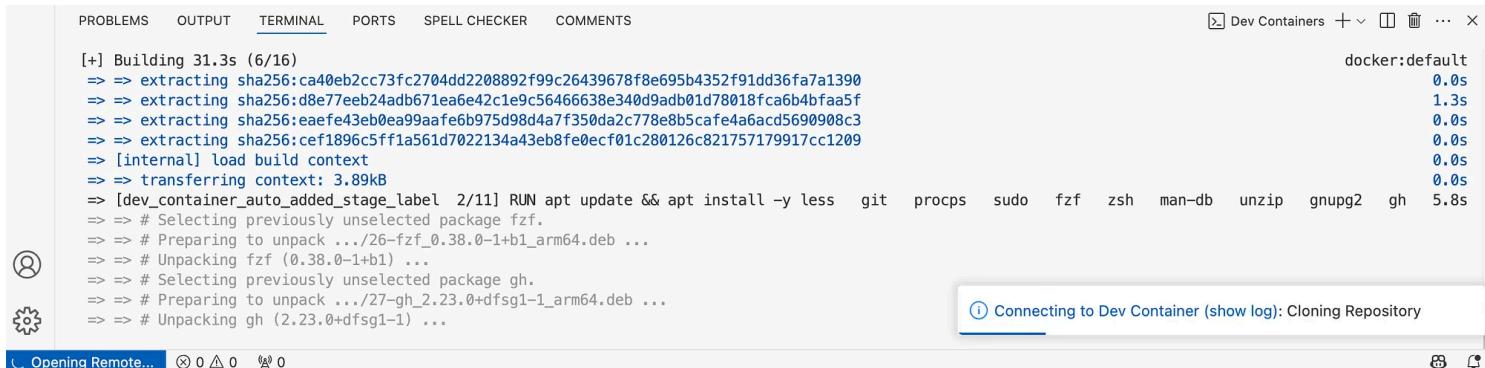
Anthropics provides a [reference implementation](#) repository that contains the Dev Containers configuration in the `.devcontainer/devcontainer.json` folder.

The easiest way to create a Dev Container from [this repository](#) is to first make sure that you have Docker running on your system, then open VS Code and choose “**Dev Containers: Clone Repository in Container Volume...**” from the VS Code Command Palette.



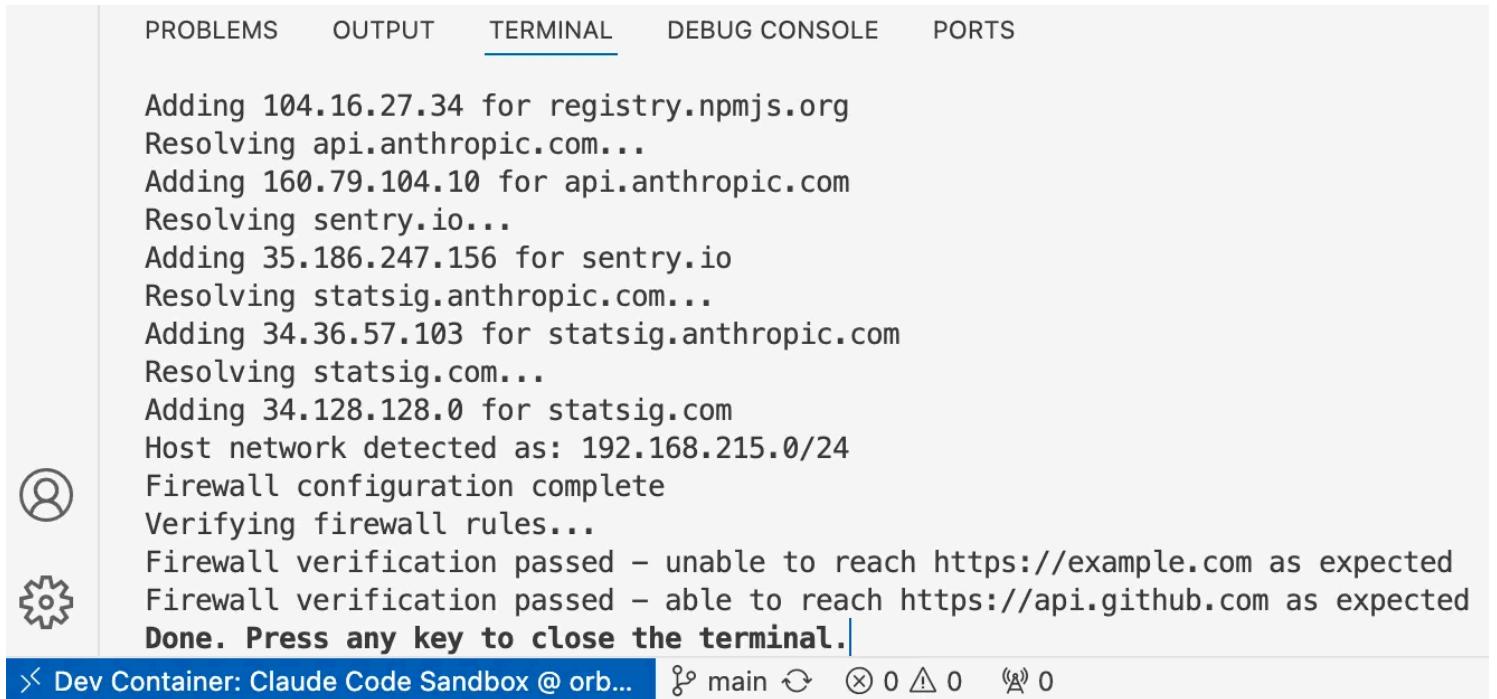
Dev Containers: Clone Repository in Container Volume...

Then you need to wait until the VS Code downloads the required docker image and install configurations:



## Opening a dev container by VS Code

After that, VS Code automatically connects to the created docker container remotely:



```
Adding 104.16.27.34 for registry.npmjs.org
Resolving api.anthropic.com...
Adding 160.79.104.10 for api.anthropic.com
Resolving sentry.io...
Adding 35.186.247.156 for sentry.io
Resolving statsig.anthropic.com...
Adding 34.36.57.103 for statsig.anthropic.com
Resolving statsig.com...
Adding 34.128.128.0 for statsig.com
Host network detected as: 192.168.215.0/24
Firewall configuration complete
Verifying firewall rules...
Firewall verification passed – unable to reach https://example.com as expected
Firewall verification passed – able to reach https://api.github.com as expected
Done. Press any key to close the terminal.
```

× Dev Container: Claude Code Sandbox @ orb... ⚡ main ⏪ 0 ✘ 0 ⚡ 0

Using Claude Code in a Dev Container in VS Code

Installation is done in a **safe** and **isolated** container, now we can open a terminal in VS Code, type the `claude` command, log into your Anthropic account, and start using Claude Code:

PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

PORTS 1

\* Welcome to **Claude Code** research preview!

/help for help

cwd: /workspace

Tips for getting started:

1. Run /init to create a CLAUDE.md file with instructions for Claude.
2. Run /terminal-setup to set up terminal integration
3. Use Claude to help with file analysis, editing, bash commands and git
4. Be as specific as you would with another engineer for the best results

> Try "create a util logging.py that..."

? for shortcuts

Claude Code welcome message

## Migrating a Spring Boot 2.x project using Claude Code

To examine Claude Code, I want to refactor the Transaction History Service from the Bank of Anthos microservices, which was implemented by Spring Boot 2.3.1 and Java 8 to Spring Boot 3.4.4 and Java 17.

**Bank of Anthos**

Test User ▾

OVERVIEW

**Checking Account****i Account Number:** 1011226111

CURRENT BALANCE

\$7,722.22

Deposit Funds Send Payment **Transaction History**

DATE	TYPE	ACCOUNT	LABEL	AMOUNT
MAR 28	• Credit	1055757655	Bob	-\$35.92
MAR 26	• Credit	1033623433	Alice	-\$58.45

Bank of Anthos microservices**1 First prompt: Migrate the service from Spring Boot 2.3.1 and Java 8 to Spring Boot 3.4.4 and Java 17**

After forking the parent repository into my GitHub account, I cloned the repository in the Claude Code dev container, then went to the `transactionhistory` folder, and started with this simple prompt:

 *Migrate this project from Spring Boot 2.3.1 and Java 8 to Spring Boot 3.4.4 and Java 17*

\* Welcome to **Claude Code** research preview!

/help for help

cwd: /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory

Tips for getting started:

1. Run /init to create a CLAUDE.md file with instructions for Claude.
2. Run /terminal-setup to set up terminal integration
3. Use Claude to help with file analysis, editing, bash commands and git
4. Be as specific as you would with another engineer for the best results

> Migrate this project from Spring Boot 2.3.1 and Java 8 to Spring Boot 3.4.4 and Java 17

First prompt: Migrate the service from Spring Boot 2.3.1 and Java 8 to Spring Boot 3.4.4 and Java 17

After a couple of seconds of thinking for the first time about scanning the codebase, the Claude Code recommended the first change in the `pom.xml` file:

- Based on my analysis, I'll now update the `pom.xml` file to migrate to Spring Boot 3.4.4 and Java 17. Here are the changes I need to make:

- `Update(file_path: pom.xml)...`

#### Edit file

```
pom.xml
28   <parent>
29     <groupId>org.springframework.boot</groupId>
30     <artifactId>spring-boot-starter-parent</artifactId>
31     <version>2.3.1.RELEASE</version>
31     <version>3.4.4</version>
32   </parent>
33
34   <properties>
35     <java.version>17</java.version>
35     <java.version>17</java.version>
36     <bootstrap.version>4.2.1</bootstrap.version>
37     <checkstyle.config.location>./checkstyle.xml</checkstyle.config.location>
38     <spring-cloud.version>Hoxton.SR5</spring-cloud.version>
38     <spring-cloud.version>2023.0.0</spring-cloud.version>
39   </properties>
40
41   <dependencyManagement>
```

Do you want to make this edit to `pom.xml`?

- > 1. Yes  
 2. Yes, and don't ask again this session (shift+tab)  
 3. No, and tell Claude what to do differently (esc)

The first suggested edit by Claude Code

After asking for a couple of changes in `pom.xml`, I set the auto-accept edits on mode, and the Claude Code continued to apply changes to the codebase and did not ask for approval again.

When the changes are finished, it asked for running the `./mvnw clean compile`.

- Now let's run Maven to check if the migration is successful:

- Bash(cd /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory && ./mvnw clean compile)...

#### Bash command

```
./mvnw clean compile
Cleans and compiles Maven project using wrapper script
```

Do you want to proceed?

- > 1. Yes
- 2. Yes, and don't ask again for `mvnw` commands in `/workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory`
- 3. No, and tell Claude what to do differently (esc)

Claude Code asks to run a command in the terminal

After getting an error when running the `./mvnw clean compile` command, Claude Code applied some changes to the `pom.xml` file until they finally could successfully build the service.

```
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ transactionhistory ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 7 source files to /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory/target/classes
[INFO] /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory/src/main/java/anthos/samples/bankofanthos/transactionhistory/TransactionCache.java: /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory/src/main/java/anthos/samples/bankofanthos/transactionhistory/TransactionCache.java uses unchecked or unsafe operations.
[INFO] /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory/src/main/java/anthos/samples/bankofanthos/transactionhistory/TransactionCache.java: Recompile with -Xlint:unchecked for details.
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 1.165 s
[INFO] Finished at: 2025-03-29T13:26:07Z
[INFO]
```

- The compilation worked! Now let's run the tests to make sure everything is working correctly:

Successful maven build by Claude Code

At this stage, Claude Code asked to run tests to make sure everything works correctly:

- The compilation worked! Now let's run the tests to make sure everything is working correctly:
- **Bash**(cd /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory && ./mvnw test)...

**Bash command**

```
./mvnw test  
Runs Maven tests using the project's wrapper script
```

Do you want to proceed?

- › 1. Yes
- 2. Yes, and don't ask again for ./mvnw test commands in /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory
- 3. No, and tell Claude what to do differently (esc)

Run tests by Claude Code

When all tests passed, it asked to run mvn package to make sure that the service can package properly:

- Great! All the tests are passing. Let's make sure we can package the application:

```
Bash(cd /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory && ./mvnw package)...
```

**Bash command**

```
./mvnw package  
Builds Maven project and creates deployable package
```

Do you want to proceed?

- › 1. Yes
- 2. Yes, and don't ask again for ./mvnw package commands in /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory
- 3. No, and tell Claude what to do differently (esc)

Run maven package command by Claude

In the end, it gave me a summary of the steps and changes:

- Great! We've successfully migrated the project from Spring Boot 2.3.1 with Java 8 to Spring Boot 3.4.4 with Java 17. Here's a summary of the changes we made:
  1. Updated Spring Boot version from 2.3.1 to 3.4.4
  2. Updated Java version from 1.8 to 17
  3. Updated Spring Cloud version to match Spring Boot 3.4.4
  4. Updated dependencies:
    - Upgraded from spring-cloud-gcp to com.google.cloud with version 4.8.4
    - Updated logging libraries (log4j-api and log4j-core) to 2.23.1
    - Updated Guava to 33.0.0-jre
    - Updated Jackson to 2.16.2
    - Updated Lettuce to 6.3.2.RELEASE
    - Updated JWT library to 4.4.0
    - Updated Mockito to 5.11.0
    - Added Jakarta APIs for servlet and XML binding
  5. Fixed code for Spring Boot 3:
    - Updated javax imports to jakarta (javax.persistence → jakarta.persistence)
    - Updated the return type of step() method from String to Duration
    - Fixed ResponseEntity diamond operator usage
    - Added null checks for environment variables to prevent NPEs
    - Updated test initialization from initMocks to openMocks
  6. Updated build plugins to compatible versions:
    - Maven clean plugin 3.2.0
    - Maven compiler plugin 3.8.1
    - Maven jar plugin 3.2.2
    - Maven surefire plugin 3.2.5
    - Maven checkstyle plugin 3.3.1
    - Jacoco plugin 0.8.11

The application now successfully compiles, passes all tests, and can be packaged, indicating a successful migration.

Summary of the changes by Claude Code

## 2 Second prompt: creating a docker image using Jib

Since the skaffold configs will try to build a docker image using the Jib maven plugin, my second prompt was:

 Make sure that we can create a docker image using the Jib maven plugin

After making some changes in the `pom.xml`, Claude Code asked to run the `./mvnw compile jib:buildTar` and then `./mvnw compile jib:dockerBuild` commands:

- Great! JIB successfully built the container image and saved it as a tar file. Let's also try the dockerBuild goal which builds directly to the local Docker daemon:
- **Bash(cd /workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory && ./mvnw compile jib:dockerBuild)...**

**Bash command**

```
./mvnw compile jib:dockerBuild  
Builds Docker image for Java project using Maven wrapper
```

Do you want to proceed?

- › 1. Yes  
2. Yes, and don't ask again for `./mvnw` commands in  
`/workspaces/bank-of-anthos-demo/transactionhistory/transactionhistory`  
3. No, and tell Claude what to do differently (esc)

Claude Code tries to create a docker image using Jib

And finally, Claude Code asked to check the Docker image was successfully created:

- Let's check if the Docker image was successfully created:
- **Bash(docker images | grep transactionhistory)...**

**Bash command**

```
docker images | grep transactionhistory  
Lists Docker images containing 'transactionhistory' keyword
```

Do you want to proceed?

- › 1. Yes  
2. No, and tell Claude what to do differently (esc)

Claude Code checks the docker image

Eventually, Claude Code asked me to run the docker image, but I chose no because the project ran using a skaffold, and there were other dependencies there.

## Test the result of refactoring

When I tried to run all microservices using the `skaffold dev --port-forward` command, I got this error:

```
[ERROR] eclipse-temurin:17-jre-alpine is a manifest list,
but the list does not contain an image for architecture=arm64, os=linux.
If your intention was to specify a platform for your image, see https://github.c
[INFO]
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time:  4.610 s
[INFO] Finished at: 2025-03-29T13:41:40Z
[INFO] -----
[ERROR] Failed to execute goal com.google.cloud.tools:jib-maven-plugin:3.4.0:doc
```

I asked about this error from Claude Code, and it fixed the problem:

- That error occurs because you're trying to build for ARM64 architecture (which your system is using), but the base image `eclipse-temurin:17-jre-alpine` doesn't have an ARM64 variant available.

Let's modify the Jib configuration to use a base image that supports ARM64 architecture. We'll use the official Eclipse Temurin image from Docker Hub that supports ARM64:

```
● Update(file_path: pom.xml)...
└ Updated pom.xml with 1 addition and 1 removal
  185      <version>3.4.0</version>
  186      <configuration>
  187          <from>
  188              <image>eclipse-temurin:17-jre-alpine</image>
  189              <image>eclipse-temurin:17-jre</image>
  190          </from>
  191          <to>
  192              <image>gcr.io/bank-of-anthos/transactionhistory:v1</image>
```

Claude Code fixes the Skaffold run problem

I tried to run all microservices again using the `skaffold dev --port-forward` command, and this time, I got an error during the transaction history service

## deployment in Kubernetes:

```
deployment/transactionhistory failed. Error: container transactionhistory termin
```

### 3 Third prompt: Run the skaffold command from the parent folder

So I asked this prompt from Claude code:

 Go to the parent folder and run this command to run all microservices: `skaffold dev --port-forward`

For security reasons, Claude Code can not navigate to the parent folder. So, I exited the current session to start a new one from the parent folder.

```
> exit
```

►► auto-accept edits on (shift+tab to toggle)

```
Total cost: $2.93
Total duration (API): 6m 47.1s
Total duration (wall): 47m 12.2s
Total code changes: 116 lines added, 57 lines removed
```

Claude Code total statistic for a session

As you can see in the image above, one good feature of Claude Code is that it shows the total cost of the current session.

### 4 Forth prompt: Start a new round of refactoring

It seems the **Transaction History Service** refactoring was successful, and Claude Code could run it independently, but since this service is part of a cloud-native application called **Bank of Anthos**, it uses Skaffold for development and running all services, and there were some problems when we want to run all microservices using the `skaffold dev - port-forward` command.

I used this prompt to find the problem:

 Run this command to check everything has been refactored successfully:

```
skaffold dev -m setup -m ledger-db -m transactionhistory --port-forward
```

I allowed Claude Code to run commands continuously without asking for permission, and the result took a long time, but it was amazing. Every time Claude Code made changes in the Transaction History Service, it ran the `mvn clean compile`, `mvn test`, and `mvn package` to make sure that the service worked independently and then ran the scaffold command to check the problem.

After several attempts, Claude Code fixed the problem by disabling all actuator endpoints, tracing, and Google Cloud Platform integration in the Transaction History Service config file.

- vscode → [/workspaces/bank-of-anthos-demo/transactionhistory \(main\)](#) \$ kubectl get pod

NAME	READY	STATUS	RESTARTS	AGE
accounts-db-0	1/1	Running	0	5m8s
balance-reader-648585d5fd-xc62x	1/1	Running	0	2m22s
contacts-5d86d79644-lf4jq	1/1	Running	0	5m6s
frontend-79744bb68-jxm6s	1/1	Running	0	4m54s
ledger-db-0	1/1	Running	0	4m42s
ledgerwriter-78f8c9445d-fsqnw	1/1	Running	0	4m41s
loadgenerator-6fb9d97658-t544g	1/1	Running	0	63s
transactionhistory-745789c4b9-64h92	1/1	Running	0	3m25s
userservice-59c7c87d7d-6kzz2	1/1	Running	0	3m38s

All pods for Bank of Anthos are up and running

I personally did not like the last changes (disabling) by Claude's code, but eventually, that disabling fixed the problem!

You can find the final changes generated by Claude Code in [my fork](#) of the Bank of Anthos GitHub repository.

## Final Thoughts

This was my third try to use AI-assisted coding tools to implement or refactor a service. In previous times, I used AI-integrated Development Environments (IDE) ([Windsurf](#) and [VS Code with GitHub Copilot](#)), but this time, it was a totally new experience with AI-assisted coding integrated with the terminal ([Claude Code](#)).

I like the AI IDEs more, and I also checked this task (Migrating a Spring Boot 2.x project) with VS Code

with GitHub Copilot Agent mode and got a similar result even with Claude 3.5 Sonnet.

However, inside the IDE, I can easily compare changes in each step and even revert one step entirely.

Of course, we must keep in mind that the Claude Code is in beta version, and we will see more features and enhancements from it in the near future.

⭐ This list contains my articles related to **AI-integrated IDEs, Agentic coding tool, AI-assisted coding, and more:**

The screenshot shows a Medium list titled "AI Coding" by Saeed Zarinfam. It features 14 stories. A prominent thumbnail for a post about "Varp 2.0" is displayed, showing a landscape with a large white arrow pointing right. Other thumbnails include "Agents" (with an 'A' icon), "Terminal" (with a terminal icon), and "D...". To the right of the list, there is a sidebar with various icons and links, including "emini CLI", "Google Cloud", "Gr...", "A...", "exe", "RPA", and "Medium".

🙏 *Thanks for reading. You can connect with me on:*

[Medium](#) |  [Twitter](#)

[Claude Code](#)

[Spring Boot](#)

[Claude](#)

[Ai Code Generator](#)

[Skaffold](#)



## Published in AI-Driven

[Follow](#)

44 followers · Last published Jul 20, 2025

Exploring the evolving world of AI-powered development, from AI coding assistants and integrated IDE tools to implementing AI agents and agentic systems.



## Written by Saeed Zarinfam

[Follow](#)

2.8K followers · 1K following

👉 Whether it's Software Development, AI, K8s, Containers, Cloud, Java, Go, Spring, IDEs, or anything in between!

## Responses (2)



Infoservices LLC

What are your thoughts?



Hossein Mobasher

Jul 30

...

Great, as usual! I also tried migrating code using Claude on IntelliJ! It works perfectly fine, and IntelliJ does the file comparison.



1



1 reply

[Reply](#)



Lucas Fernandes



he

Mar 31

...