

Sistema de Moeda Estudantil Gerênciа de Configuração

Ana Flávia de Carvalho Santos

Júlia Moreira Nascimento

Lorryne Marayze Silva de Oliveira

Pedro Henrique Pires Rodrigues

Pedro Henrique de Almeida Costa



Objetivo do Trabalho

- Criar um Plano de Gerenciamento de Configuração (PGC)
- Implementar testes automatizados:
 - Unidade
 - Integração
 - Aceitação
- Configurar pipeline CI/CD com:
 - Build
 - Execução de testes
 - Empacotamento
 - Deploy (Docker)
- Aplicar práticas de DevOps e Integração Contínua

Tecnologias Utilizadas

Backend

- Spring Boot
- Maven
- JDK 17 (Eclipse Temurin)

Infra e DevOps

- Git + GitHub
- CircleCI
- Docker
- MySQL

Frontend

- React
- TypeScript

Funcionamento Geral

- O SyncCoin é um sistema de moeda estudantil usado para registrar e distribuir moedas entre alunos.
- O frontend (React + TypeScript) envia requisições para a API.
- O backend (Spring Boot) trata regras de negócio e acessa o banco.
- O banco de dados MySQL armazena usuários, transações e saldos.
- A aplicação é empacotada em containers Docker para facilitar execução e deploy.

Escopo PGC

- Código-fonte do backend (Spring Boot) e frontend (React).
- Documentação do projeto: requisitos, visão, arquitetura, testes.
- Scripts e estrutura do banco de dados.
- Arquivos de configuração e ambiente Docker.
- Artefatos de build, releases e empacotamento.
- Processos de controle de mudança (Issues + Pull Requests).
- Aplica-se a todos os membros do projeto.

Responsabilidades

Gerente de Projeto

- Aprova baselines.
- Autoriza mudanças de escopo.
- Acompanha cronograma.

Gerente de Configuração

- Administra o repositório GitHub.
- Define branches, releases e backups.
- Garante organização e versionamento.

Desenvolvedores

Realizam commits corretos.

Criam Pull Requests.

Resolvem conflitos e impleme

Responsabilidades

Desenvolvedores

- Realizam commits corretos.
- Criam Pull Requests.
- Resolvem conflitos e implementam mudanças.

Identificação da Configuração

- Itens de configuração divididos por módulos:
 - backend, frontend, banco, documentação, scripts.
- Versionamento semântico: MAJOR.MINOR.PATCH (ex: v1.0.0).
- Artefatos: código, imagens Docker, pacotes Maven.
- Uso de tags para marcar baselines (v0.1.0, v0.5.0, v0.8.0, v1.0.0).
- Estrutura padronizada de diretórios.

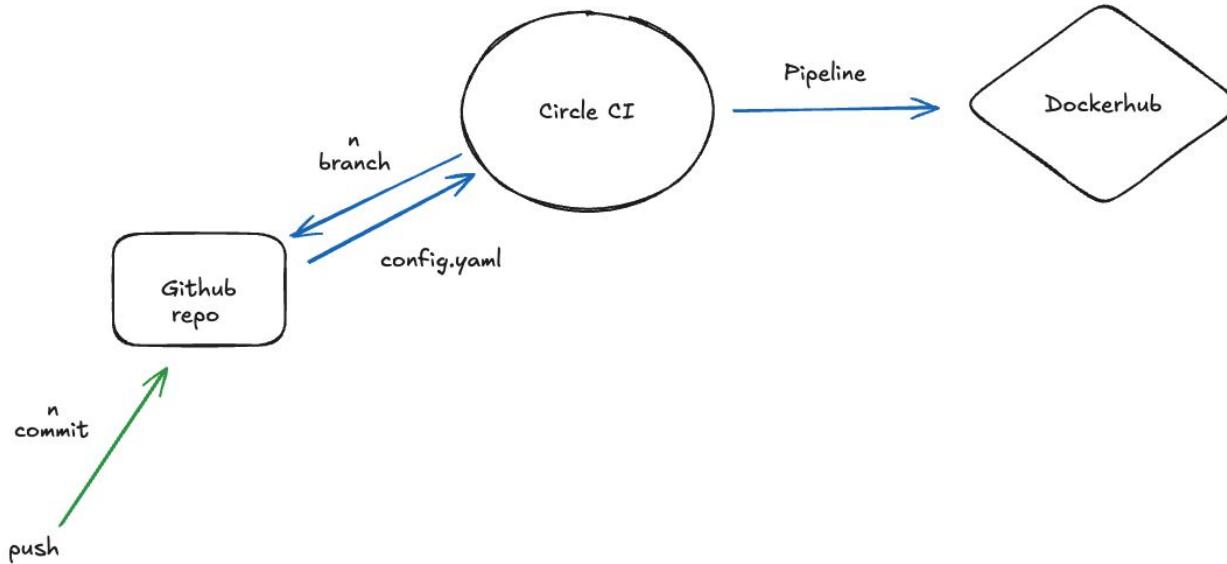
Tarefas do RUP Adaptadas ao Projeto

- Configurar Ambiente CM
 - Definição do repositório, .gitignore e proteção da main
- Estruturar Modelo de Implementação
 - Diretórios organizados: /src, /docs, /db.
- Espaço de Trabalho de Integração
 - Branch main => integração contínua.
- Espaços de Trabalho de Desenvolvimento
 - Branches feature: feature/nova-funcionalidade.

Fluxo de Controle de Mudanças

1. Uma mudança começa como Issue no GitHub.
2. Implementação ocorre em uma branch separada.
3. Pull Request enviado com descrição e evidências.
4. CircleCI executa build + testes automaticamente.
5. Após aprovação, o PR é integrado na main.
6. Baselines são criadas via Tags.

Funcionamento



Github Integration

The screenshot shows a GitHub repository page for 'LPS_LAB3'. The repository is private and has 2 branches and 0 tags. The main branch is selected. There are 8 commits from 'ped-alim' with the message 'removing fixed docker version'. The commit hash is 6c6452f and it was made 'now'. The page includes standard GitHub navigation links like Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings.

TempGestao / LPS_LAB3

Code Issues Pull requests Actions Projects Security Insights Settings

LPS_LAB3 Private

main 2 Branches 0 Tags

Go to file Add file Code

ped-alim removing fixed docker version 6c6452f · now 8 Commits

Github Integration

LPS_LAB3 Private

main ▾ 2 Branches 0 Tags Go to file Add file Code About

ped-alm removing fixed docker version 6c6452f · 1 minute ago 8 Commits

.circleci removing fixed docker version 1 minute ago

.vscode adicionando teste de aceitacao e melhorando pipeline 33 minutes ago

Artefatos copy 1 hour ago

api Fix acceptance test 27 minutes ago

Some checks haven't completed yet

1 pending check

ci/circleci: build Pending Details

my-app

.gitignore

README.md

SETUP.md copy 1 hour ago

No des
provide
Rea
Acti
Cus
0 st
0 w
0 fo
relea
Create a

Circle CI

The screenshot shows the CircleCI web interface for a project named "LPS_LAB3" on the "main" branch. The pipeline is named "ci-cd-pipeline" and is currently "Running". A notification bar at the top indicates a new feature: "Smarter Testing - Get faster feedback on tests with Test Impact Analysis and Dynamic Test Splitting. Join Waitlist". Below this, pipeline details are shown: Duration 1m 54s, Branch main, Commit 6c6452f, and Author & Message "removing fixed docker version". The pipeline itself consists of four sequential steps: "build" (25s), "unit-tests" (1m 4s), "integration-tests" (19s), and "acceptance-tests". All steps have passed successfully, as indicated by green checkmarks.

circleci •

Dashboard Project Branch Workflow

All Pipelines > LPS_LAB3 > main > ci-cd-pipeline

ci-cd-pipeline Running

New: Smarter Testing - Get faster feedback on tests with Test Impact Analysis and Dynamic Test Splitting. [Join Waitlist](#)

Duration: 1m 54s | Branch: main | Commit: 6c6452f | Author & Message: removing fixed docker version

build 25s | unit-tests 1m 4s | integration-tests 19s | acceptance-tests

Insights Rerun

Home Pipelines Projects Deployments Insights Runners Org Plan

Github Integration

The screenshot shows a GitHub repository named "LPS_LAB3" (Private). The repository has 2 branches and 0 tags. The main branch is selected. The repository contains files like .circleci, .vscode, Artefatos, api, my-app, .gitignore, README.md, and SETUP.md. A modal window titled "All checks have passed" displays 6 successful checks: ci/circleci: acceptance-tests, ci/circleci: build, ci/circleci: build-docker-image, ci/circleci: deploy, ci/circleci: integration-tests, and ci/circleci: unit-tests. The GitHub interface includes navigation tabs (Code, Issues, Pull requests, Actions, Projects, Security, Insights, Settings), a search bar, and various repository statistics.

LPS_LAB3 (Private)

main 2 Branches 0 Tags

ped-alim removing fixed docker version ✓ 6c6452f · 18 minutes ago 8 Commits

.circleci removing fixed docker version 18 minutes ago

.vscode

Artefatos

api

my-app

.gitignore

README.md

SETUP.md

README

All checks have passed

6 successful checks

- ✓ ci/circleci: acceptance-tests Details
- ✓ ci/circleci: build Details
- ✓ ci/circleci: build-docker-image Details
- ✓ ci/circleci: deploy Details
- ✓ ci/circleci: integration-tests Details
- ✓ ci/circleci: unit-tests Details

Config:

```
.circleci > ! config.yml
285
286   workflows:
287     version: 2
288     ci-cd-pipeline:
289       jobs:
290         - build:
291           filters:
292             branches:
293               only:
294                 - main
295                 - master
296         - unit-tests:
297           requires:
298             - build
299             filters:
300               branches:
301                 only:
302                   - main
303                   - master
304         - integration-tests:
305           requires:
306             - unit-tests
307             filters:
308               branches:
309                 only:
310                   - main
311                   - master
312
313   - acceptance-tests:
314     requires:
315       - integration-tests
316     filters:
317       branches:
318         only:
319           - main
320           - master
321   - build-docker-image:
322     requires:
323       - acceptance-tests
324     filters:
325       branches:
326         only:
327           - main
328           - master
329   - deploy:
330     requires:
331       - build-docker-image
332     filters:
333       branches:
334         only:
335           - main
336           - master
%l to chat, %K to generate
```

Config:build

```
.circleci > ! config.yml
You, 5 minutes ago | 2 authors (You and one other)
1  version: 2.1
2  jobs:
3    build:
4      docker:
5        - image: maven:3.9-eclipse-temurin-17
6        working_directory: ~/project/api
7        steps:
8          - checkout:
9            path: ~/project
10         - run:
11           name: Verificar versões Java e Maven
12           command:
13             - |
14               java -version
15               mvn -version
16         - restore_cache:
17           keys:
18             - maven-v1-{{ checksum "pom.xml" }}
19             - maven-v1-
20         - run:
21           name: Baixar dependências Maven
22           command: mvn dependency:go-offline -B
23         - save_cache:
24           paths:
25             - ~/.m2
26           key: maven-v1-{{ checksum "pom.xml" }}
27         - run:
28           name: Compilar aplicação
29           command: mvn clean compile -B
30           environment:
31             SPRING_PROFILES_ACTIVE: test
```

Config: unit-tests

```
31  unit-tests:
32    docker:
33      - image: maven:3.9-eclipse-temurin-17
34      working_directory: ~/project/api
35    steps:
36      - checkout:
37        path: ~/project
38      - restore_cache:
39        keys:
40          - maven-v1-{{ checksum "pom.xml" }}
41          - maven-v1-
42      - run:
43        name: Executar testes de unidade
44        command: |
45          # Executa testes de unidade (services, models, dtos, exceptions, security)
46          # Exclui testes de integração e aceitação
47          mvn test \
48            -Dtest="*ServiceTest,*Tests,*DTOTest,*ExceptionTest,*FilterTest,*ProviderTest" \
49            -DfailIfNoTests=false \
50            -B
51        environment:
52          SPRING_PROFILES_ACTIVE: test
53      - run:
54        name: Resumo dos testes de unidade
55        command: |
56          if [ -f target/surefire-reports/TEST-*.xml ]; then
57            echo "=== Resumo dos Testes de Unidade ==="
58            find target/surefire-reports -name "TEST-*.xml" -exec grep -h "testsuite" {} \; | head -
59          fi
60      when: always
61    - store_test_results:
62      path: ~/project/api/target/surefire-reports
63    - store_artifacts:
64      path: ~/project/api/target/surefire-reports
65      destination: unit-test-results
```

Config:integration-tests

```
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99 integration-tests:
  docker:
    - image: maven:3.9-eclipse-temurin-17
  working_directory: ~/project/api
  steps:
    - checkout:
        path: ~/project
    - restore_cache:
        keys:
          - maven-v1-{{ checksum "pom.xml" }}
          - maven-v1-
    - run:
        name: Executar testes de integração
        command:
          # Executa testes de integração (controllers e testes com Spring context)
          mvn test \
            -Dtests="*ControllerTest,ApiApplicationTests" \
            -DfailIfNoTests=false \
            -B
        environment:
          SPRING_PROFILES_ACTIVE: test
    - run:
        name: Resumo dos testes de integração
        command:
          if [ -f target/surefire-reports/TEST-*.xml ]; then
            echo "==== Resumo dos Testes de Integração ==="
            find target/surefire-reports -name "TEST-*.xml" -exec grep -h "testsuite" {} \; | head -10
          fi
        when: always
    - store_test_results:
        path: ~/project/api/target/surefire-reports
    - store_artifacts:
        path: ~/project/api/target/surefire-reports
        destination: integration-test-results
```

Config:acceptance-tests

```
100
101 acceptance-tests:
102   docker:
103     - image: maven:3.9-eclipse-temurin-17
104   working_directory: ~/project/api
105   steps:
106     - checkout:
107       path: ~/project
108     - restore_cache:
109       keys:
110         - maven-v1-{{ checksum "pom.xml" }}
111         - maven-v1-
112     - run:
113       name: Executar testes de aceitação
114       command: |
115         # Executa testes de aceitação (end-to-end)
116         mvn test -Dtest=*AcceptanceTest* -B
117       environment:
118         SPRING_PROFILES_ACTIVE: test
119     - run:
120       name: Resumo dos testes de aceitação
121       command: |
122         if [ -f target/surefire-reports/TEST-*.xml ]; then
123           echo "==== Resumo dos Testes de Aceitação ==="
124           find target/surefire-reports -name "TEST-*.xml" -exec grep -h "testsuite" {} \;
125           echo ""
126           echo "==== Detalhes dos Testes ==="
127           find target/surefire-reports -name "*.txt" -exec cat {} \;
128         fi
129       when: always
130     - store_test_results:
131       path: ~/project/api/target/surefire-reports
132     - store_artifacts:
133       path: ~/project/api/target/surefire-reports
       destination: acceptance-test-results
```

Config:build-docker-image

```
134  build-docker-image:
135    docker:
136      - image: docker:24.0.5
137      working_directory: ~/project/api
138      steps:
139        - checkout:
140          path: ~/project
141        - setup_remote_docker
142        - run:
143          name: Build da imagem Docker
144          command:
145            docker build -t student-currency-system:latest .
146            docker images student-currency-system:latest
147        - run:
148          name: Testar container Docker
149          command:
150            # Inicia o container em background
151            docker run -d --name test-container -p 8080:8080 student-currency-system:latest
152            sleep 30
153            # Verifica se a aplicação está respondendo
154            curl -f http://localhost:8080/actuator/health || echo "Health check failed"
155            # Para e remove o container
156            docker stop test-container
157            docker rm test-container
158        - run:
159          name: Salvar imagem Docker
160          command:
161            docker save student-currency-system:latest | gzip > student-currency-system.tar.gz
162        - store_artifacts:
163          path: ~/project/api/student-currency-system.tar.gz
164          destination: docker-image
```

Config:deploy

```
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
deploy:
  docker:
    | - image: docker:24.0.5
  working_directory: ~/project/api
  steps:
    - checkout:
        | path: ~/project
    - setup_remote_docker
    - run:
        name: Login no Docker Hub
        command: |
          echo "==== Fazendo login no Docker Hub ==="
          echo "$DOCKERHUB_PASSWORD" | docker login -u "$DOCKERHUB_USERNAME" --password-stdin
          echo "Login realizado com sucesso!"
    - run:
        name: Build da imagem Docker
        command: |
          echo "==== Construindo imagem Docker ==="
          docker build -t ${DOCKERHUB_USERNAME}/student-currency-system:${CIRCLE_SHA1} .
          docker build -t ${DOCKERHUB_USERNAME}/student-currency-system:latest .
          echo "Imagens construídas:"
          docker images | grep student-currency-system
    - run:
        name: Push para Docker Hub
        command: |
          echo "==== Fazendo push para Docker Hub ==="
          echo "Push da imagem com tag: ${CIRCLE_SHA1}"
          docker push ${DOCKERHUB_USERNAME}/student-currency-system:${CIRCLE_SHA1}
          echo "Push da imagem latest"
          docker push ${DOCKERHUB_USERNAME}/student-currency-system:latest
          echo "=====
          echo "Deploy realizado com sucesso!"
          echo "Imagen disponível em:"
          echo "  - ${DOCKERHUB_USERNAME}/student-currency-system:${CIRCLE_SHA1}"
          echo "  - ${DOCKERHUB_USERNAME}/student-currency-system:latest"
          echo "====="
    - run:
        name: Logout do Docker Hub
        command: |
          docker logout
        echo "Logout realizado"
        when: always
```

CircleCI: Sucesso

The screenshot shows the CircleCI web interface for the pipeline **LPS_LAB3**. The pipeline **build-and-test** has completed successfully, triggered by a push commit pushed event. The total duration was 6m 53s, starting 20m ago.

Pipeline Details:

- Pipeline:** build-and-test
- Status:** Success
- Workflow:** ci-cd-pipeline
- Checkout source:** main
6c6452f removing fixed docker version
- Trigger event:** Push commit pushed
- Start:** 20m ago
- Duration:** 6m 53s
- Actions:** [View](#), [Re-run](#), [Delete](#), [...](#)

Jobs:

Job	Attempts	Duration
build	22	25s
unit-tests	23	1m 4s
integration-tests	24	58s
acceptance-tests	25	54s
build-docker-image	26	1m 55s
deploy	27	1m 13s

CircleCI: Falha

A screenshot of a CircleCI pipeline interface. The pipeline is named "ci-cd-pipeline" and has a branch "main". A commit "e75738f fixing docker hub" was pushed 30 minutes ago. The pipeline status is "X Failed". The pipeline consists of six jobs:

Job	Status	Duration
build 16	✓ Passed	25s
unit-tests 17	✓ Passed	57s
integration-tests 18	✓ Passed	52s
acceptance-tests 19	✓ Passed	55s
build-docker-image 20	X Failed	0s
deploy 21	Not Started	

CircleCI: Teste quebrado

The screenshot shows a CircleCI pipeline interface. At the top, there are navigation links: Dashboard, Project, Branch, Workflow, and Job. Below these, a breadcrumb trail shows the current path: All Pipelines > LPS_LAB3 > main > ci-cd-pipeline > acceptance-tests (7). A red "Failed" badge is visible next to the "acceptance-tests" link.

A notification bar at the top indicates a new feature: "Smarter Testing - Get faster feedback on tests with Test Impact Analysis and Dynamic Test Splitting. Join Waitlist".

Below the notification, there are several status indicators: Duration / Finished (1m 3s / 50m ago), Queued (0s), Executor / Resource Class (Docker / Large), Branch (main), Commit (0429bdb), and Author & Message ("adicionando teste de aceitacao e melhorando pipeline").

The main area displays the pipeline steps. The "Steps" tab is selected, showing the following steps:

- Spin up environment (9s)
- Preparing environment variables (0s)
- Checkout code (1s)
- Restoring cache (2s)
- Executar testes de aceitação (49s, failed)

The "Executar testes de aceitação" step is highlighted in pink and contains the following command and its output:

```
#!/bin/bash -eo pipefail
# Executa testes de aceitação (end-to-end)
mvn test -Dtest="*AcceptanceTest" -B
```

Output log:

```
785 |     Forwarded URL = null
786 |     Redirected URL = null
787 |     Cookies = []
788 | [ERROR] Tests run: 2, Failures: 1, Errors: 0, Skipped: 0, Time elapsed: 8.112 s <<< FAILURE! -- in com.lps.api.acceptance.StudentRegistrationAcceptanceTest
789 | [ERROR] com.lps.api.acceptance.StudentRegistrationAcceptanceTest shouldProjectToValidStudentData ... Time elapsed: 0.122 s <<< FAILURE!
```

Github: Teste quebrado

The screenshot shows a GitHub repository interface with a sidebar on the left and a main content area on the right.

Sidebar:

- fixing docker hub
- Fix acceptance test
- adicionando teste de aceitacao e me
- basic circle
- Merge pull request #1 from TempGestao/circleci-project-setup

Main Content Area:

A modal window titled "Some checks were not successful" is displayed over the repository list. The modal contains the following information:

1 failing and 3 successful checks

Status	Check Name	Details
✗	ci/circleci: acceptance-tests	Details
✓	ci/circleci: build	Details
✓	ci/circleci: integration-tests	Details
✓	ci/circleci: unit-tests	Details

DockerHub

The screenshot shows the Docker Hub interface. At the top, there is a blue header bar with the 'hub' logo, 'Explore', and 'My Hub' tabs. On the far right of the header are icons for search, notifications, and profile. Below the header is a sidebar on the left with a user profile for 'pedr0almeida' (Docker Personal) and sections for 'Repositories', 'Hardened Images', 'Collaborations', 'Settings' (with 'Default privacy' and 'Notifications' options), 'Billing', 'Usage', 'Pulls', and 'Storage'. The main content area is titled 'Repositories' and shows a list of repositories under the 'pedr0almeida' namespace. A search bar at the top of the list allows searching by repository name or all content. The repository list includes columns for Name, Last Pushed, Contains, Visibility, and Scout. One repository, 'pedr0almeida/student-currency-system', is listed with its last push 17 minutes ago, containing an image, and being public and inactive.

Name	Last Pushed	Contains	Visibility	Scout
pedr0almeida/student-currency-system	17 minutes ago	IMAGE	Public	Inactive

Obrigado !