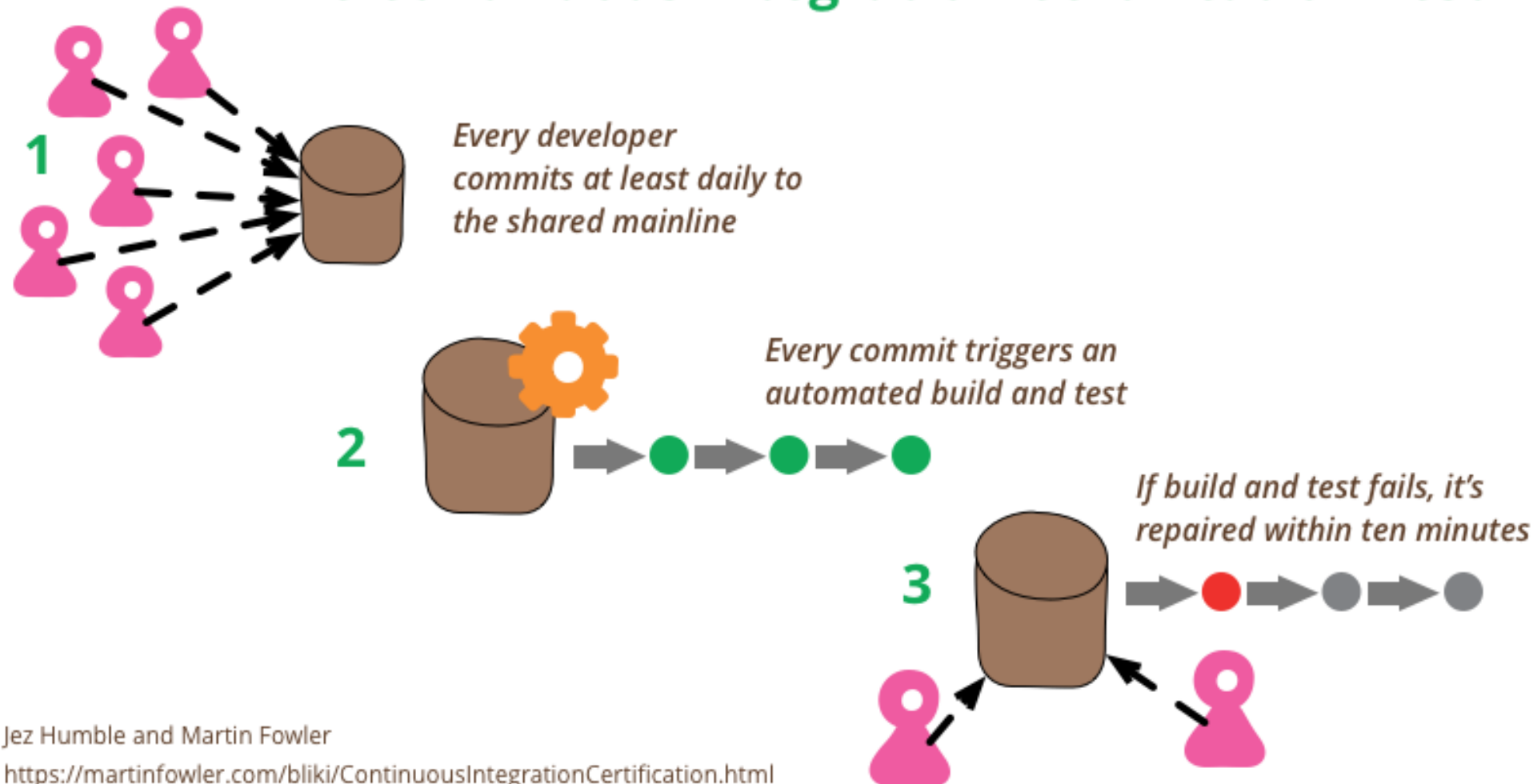


# Integração contínua e delivery com CircleCI

**"CI é uma estratégia de desenvolvimento de software que aumenta a velocidade do desenvolvimento, garantindo a qualidade do código."**

Martin Fowler

## The Continuous Integration Certification Test

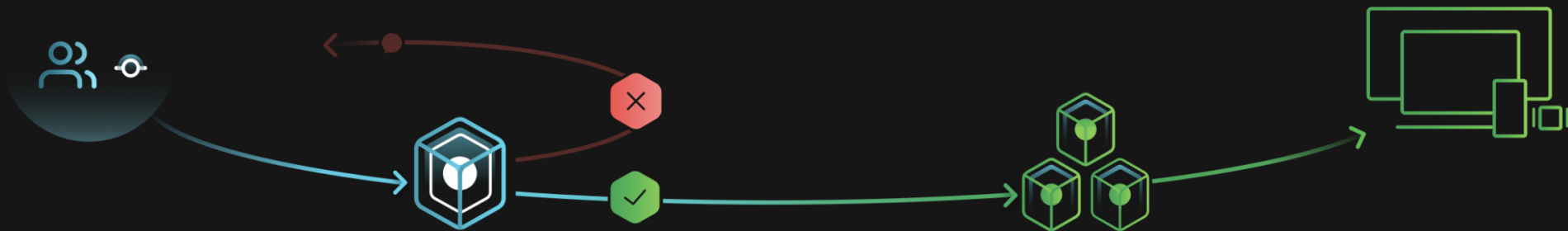


Jez Humble and Martin Fowler

<https://martinfowler.com/bliki/ContinuousIntegrationCertification.html>

## O que é o CircleCI

É uma plataforma de integração contínua e delivery que ajuda equipes de todos os tamanhos a desenvolver e lançar rapidamente software de qualidade em escala.



### VCS Integration

CircleCI integrates with GitHub, GitHub Enterprise, and Bitbucket. Every time you commit code, CircleCI creates a build.

### Automated Testing

CircleCI automatically tests your build in a clean container or virtual machine.

### Notifications

Your team is notified if a build fails so issues can be fixed quickly.

### Automated Deployment

Passing builds are deployed to various environments so your product goes to market faster.

## Usando CircleCI no mundo real

1. Criar conta no CircleCI
2. Criar ou integrar projeto do GitHub, Bitbucket
3. Criar na raiz do projeto **`.circleci/config.yml`**

```
version: 2
jobs:
  build:
    docker:
      # specify the version you desire here
      - image: circleci/node:7.10

      # Specify service dependencies here if necessary
      # CircleCI maintains a library of pre-built images
      # documented at https://circleci.com/docs/2.0/circleci-images/
      # - image: circleci/mongo:3.4.4

    working_directory: ~/repo

    steps:
      - checkout

      # Download and cache dependencies
      - restore_cache:
          keys:
            - v1-dependencies-{{ checksum "package.json" }}
            # fallback to using the latest cache if no exact match is found
            - v1-dependencies-

      - run: yarn install

      - save_cache:
          paths:
            - node_modules
          key: v1-dependencies-{{ checksum "package.json" }}

      # run tests!
      - run: yarn test
```

**Cada commit irá gerar um build automaticamente**



<div>! FAILED</div>	master #271 Create file alagoanes.json with dialect alagoano. (#67)	<div>🔗 default</div> <div>● test</div>	<div>📅 12 days ago</div> <div>🕒 00:08</div> <div>🔑 34ed8ae</div>	2.0
<div>✓ SUCCESS</div>	master #270 Create file alagoanes.json with dialect alagoano. (#67)	<div>🔗 default</div> <div>● lint</div>	<div>📅 12 days ago</div> <div>🕒 00:12</div> <div>🔑 34ed8ae</div>	2.0
<div>✓ SUCCESS</div>	master #269 Create file alagoanes.json with dialect alagoano. (#67)	<div>🔗 default</div> <div>● checkout</div>	<div>📅 12 days ago</div> <div>🕒 00:10</div> <div>🔑 34ed8ae</div>	2.0
<div>! FAILED</div>	master #268 Adicionado novos dialetos (#66)	<div>🔗 default</div> <div>● test</div>	<div>📅 12 days ago</div> <div>🕒 00:08</div> <div>🔑 053d851</div>	2.0
<div>✓ SUCCESS</div>	master #267 Adicionado novos dialetos (#66)	<div>🔗 default</div> <div>● lint</div>	<div>📅 12 days ago</div> <div>🕒 00:04</div> <div>🔑 053d851</div>	2.0
<div>✓ SUCCESS</div>	master #266 Adicionado novos dialetos (#66)	<div>🔗 default</div> <div>● checkout</div>	<div>📅 12 days ago</div> <div>🕒 00:18</div> <div>🔑 053d851</div>	2.0
<div>! FAILED</div>	master #265 add baianês dialects (#68)	<div>🔗 default</div> <div>● test</div>	<div>📅 12 days ago</div> <div>🕒 00:08</div> <div>🔑 ab581f9</div>	2.0
<div>✓ SUCCESS</div>	master #264 add baianês dialects (#68)	<div>🔗 default</div> <div>● lint</div>	<div>📅 12 days ago</div> <div>🕒 00:04</div> <div>🔑 ab581f9</div>	2.0
<div>✓ SUCCESS</div>	master #263 add baianês dialects (#68)	<div>🔗 default</div> <div>● checkout</div>	<div>📅 12 days ago</div> <div>🕒 00:10</div> <div>🔑 ab581f9</div>	2.0

**Vamos praticar!**