

# Bitbucket Pipelines Tham Khảo Nhanh

CI/CD pipeline, caching, artifact, deployment

## Cơ Bản Pipeline

### Cách Hoạt Động

<b>bitbucket-pipelines.yml</b>	File cấu hình ở thư mục gốc repo
<b>Docker containers</b>	Mỗi step chạy trong container riêng
<b>Triggers</b>	Push, PR, tag, lịch trình, hoặc thủ công
<b>Build minutes</b>	Hạn ngạch phụ thuộc vào gói dịch vụ

### Bật Pipelines

```
# Repository Settings → Pipelines → Enable
# Thêm bitbucket-pipelines.yml vào thư mục gốc
# Push đầu tiên sẽ kích hoạt pipeline
```

## bitbucket-pipelines.yml

### Cấu Hình Tối Giản

```
image: node:20
pipelines:
  default:
    - step:
      script:
        - npm install
        - npm test
```

### Pipeline Theo Branch

```
pipelines:
  branches:
    main:
      - step:
          script:
            - npm run build
            - npm run deploy
```

### Pipeline Theo Tag & Pull Request

```
pipelines:
  tags:
    'v*':
      - step:
          script:
            - npm run release
  pull-requests:
    '**':
      - step:
          script:
            - npm test
```

## Steps

### Tùy Chọn Step

<b>name</b>	Tên hiển thị của step
<b>image</b>	Ghi đề Docker image toàn cục
<b>script</b>	Danh sách lệnh shell cần chạy
<b>size</b>	1x (4GB) hoặc 2x (8GB) bộ nhớ
<b>max-time</b>	Timeout tính bằng phút (mặc định 120)
<b>trigger</b>	manual cho step chỉ chạy thủ công

### Step Song Song

```
- parallel:
  - step:
      name: "Lint"
      script:
        - npm run lint
  - step:
      name: "Test"
      script:
        - npm test
```

## Step Thủ Công

```
- step:
  name: "Deploy to Production"
  trigger: manual
  script:
    - ./deploy.sh prod
```

## Biến

### Các Loại Biến

<b>Repository variables</b>	Settings → Pipelines → Variables
<b>Deployment variables</b>	Giới hạn trong môi trường deployment
<b>Secured variables</b>	Mã hóa, ẩn trong log
<b>Pipeline variables</b>	Định nghĩa trực tiếp trong YAML

### Sử Dụng Biến

```
pipelines:
  default:
    - step:
        script:
          - echo $MY_VAR
          - docker login -u $DOCKER_USER -p $DOCKER_PASS
```

### Biến Built-in

<b>\$BITBUCKET_COMMIT</b>	SHA commit đầy đủ
<b>\$BITBUCKET_BRANCH</b>	Tên branch
<b>\$BITBUCKET_TAG</b>	Tên tag (pipeline theo tag)
<b>\$BITBUCKET_BUILD_NUMBER</b>	Số build tăng dần
<b>\$BITBUCKET_REPO_SLUG</b>	Slug của repository

## Caching

### Cache Có Sẵn

```
- step:
  caches:
    - node # ~/.npm
    - pip # ~/.cache/pip
    - docker # Docker layer cache
  script:
    - npm install
    - npm test
```

### Cache Tùy Chỉnh

```
definitions:
  caches:
    gradle: ~/.gradle/caches
    mylibs: vendor/libs
pipelines:
  default:
    - step:
        caches:
          - gradle
        script:
          - ./gradlew build
```

### Hành Vi Cache

<b>Thời hạn</b>	Cache hết hạn sau 7 ngày
<b>Phạm vi</b>	Chia sẻ trên tất cả pipeline trong repo
<b>Xóa</b>	Pipelines → Caches → Delete

## Artifacts

### Truyền File Giữa Các Step

```
- step:
  name: "Build"
  script:
    - npm run build
  artifacts:
    - dist/**
- step:
  name: "Deploy"
  script:
    - ls dist/ # artifacts có sẵn
    - ./deploy.sh
```

### Tùy Chọn Artifact

<b>artifacts</b>	Glob pattern cho file cần truyền
<b>Download</b>	Tự động có sẵn ở các step tiếp theo
<b>Max size</b>	1 GB mỗi step
<b>Retention</b>	Có sẵn trong 14 ngày sau khi build

## Deployments

### Môi Trường Deployment

```
- step:
  name: "Deploy Staging"
  deployment: staging
  script:
    - ./deploy.sh staging
- step:
  name: "Deploy Production"
  deployment: production
  trigger: manual
  script:
    - ./deploy.sh prod
```

### Loại Môi Trường

<b>test</b>	Môi trường kiểm thử
<b>staging</b>	Môi trường tiền production
<b>production</b>	Môi trường live, theo dõi trên dashboard

## Các Pattern Thường Gặp

### Docker Build & Push

```
- step:
  services:
    - docker
  script:
    - docker build -t myapp:$BITBUCKET_COMMIT .
    - docker login -u $DOCKER_USER -p $DOCKER_PASS
    - docker push myapp:$BITBUCKET_COMMIT
```

### Service Containers

```
definitions:
  services:
    postgres:
      image: postgres:16
      variables:
        POSTGRES_DB: testdb
        POSTGRES_PASSWORD: secret
pipelines:
  default:
    - step:
        services:
          - postgres
        script:
          - npm test
```

# Bitbucket Pipelines Tham Khảo Nhanh

---

## Step Điều Kiện với Pipe

```
- step:  
  name: "Deploy to S3"  
  script:  
    - pipe: atlassian/aws-s3-deploy:1.1.0  
      variables:  
        AWS_ACCESS_KEY_ID: $AWS_KEY  
        AWS_SECRET_ACCESS_KEY: $AWS_SECRET  
        S3_BUCKET: my-bucket  
        LOCAL_PATH: dist/
```