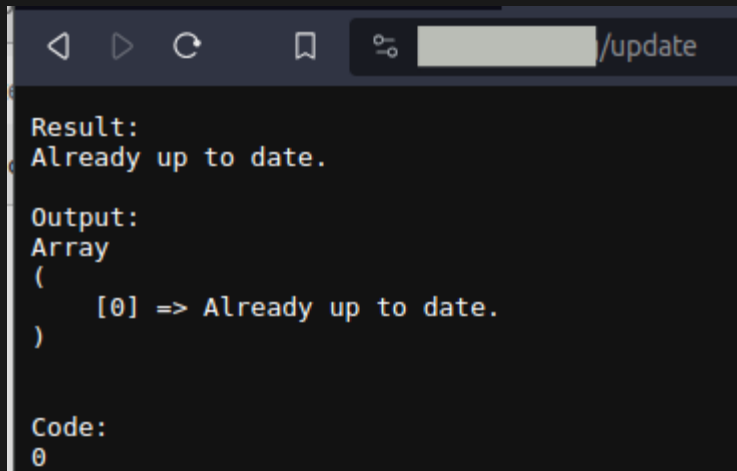


CI/CD

```
return function () {  
    header("Content-Type: text/plain");  
    $output = "";  
    $code = "";  
    $res = exec("cd .. && git pull", $output, $code);  
  
    echo "Result:\n";  
    print_r($res);  
    echo "\n\n";  
    echo "Output:\n";  
    print_r($output);  
    echo "\n\n";  
    echo "Code:\n";  
    print_r($code);  
}
```



```
Result:  
Already up to date.  
  
Output:  
Array  
(  
    [0] => Already up to date.  
)  
  
Code:  
0
```

# TARGETS

- build
- test
- metrics
- backup
- deployment

# BUILD

- npm
  - install
  - build
- composer
  - install

# TEST

- php unit tests
  - mocking, data providers, asserts,...
- js test
  - karma
  - jest

# METRICS

- test coverage (total, diff)
- bug finder (phpstan)
- requirements (clean code, max file size, ...)

# BACKUP

- database
- files

# DEPLOYMENT

- compress, upload, replace
- configuration (token replacement)
- db migrations
- cache warmup



# composer.json

```
{
  "scripts": {
    "phpunit": "vendor/bin/phpunit --display-warnings --disp",
    "coverage": "vendor/bin/phpunit --coverage-html build/re",
    "csfixer": "vendor/bin/php-cs-fixer fix --allow-risky=yes",
    "phpstan": "vendor/bin/phpstan analyse src tests --memor",
    "rector": "vendor/bin/rector process --no-ansi",
  }
}
```

<https://github.com/tempestphp/tempest-framework/blob/main/composer.json>

**PHPSTAN**













































# CSFIXER, RECTOR

- PHP Coding Standards Fixer (PHP CS Fixer)
- upgrades and refactors the PHP code
- ide integration available
- use as git hooks

# GITLAB PIPELINE

# TERMS

- Stage
- Job
- Environments
- Variables (global, Environments, process)
- Artifact

Key ↑	Value	Environments	Actions
API_URL  <span>Expanded</span>	***** 	production 	 
API_URL  <span>Protected</span> <span>Expanded</span>	***** 	dev 	 
APP_PATH  <span>Protected</span> <span>Expanded</span>	***** 	production 	 
APP_PATH  <span>Protected</span> <span>Expanded</span>	***** 	dev 	 
SSH_HOST  <span>Expanded</span>	***** 	All (default) 	 
SSH_PORT  <span>Protected</span> <span>Expanded</span>	***** 	All (default) 	 
SSH_PRIVATE_KEY_BASE64  <span>Protected</span> <span>Masked</span> <span>Hidden</span> <span>Expanded</span>		All (default) 	 
SSH_USER  <span>Protected</span> <span>Expanded</span>	***** 	All (default) 	 
TEMP_DIR  <span>Protected</span> <span>Expanded</span>	***** 	All (default) 	 

All 10 Finished Branches Tags

[View analytics](#)

[Clear runner caches](#)

[New pipeline](#)



Filter pipelines





Show Pipeline ID ▾

Status	Pipeline	Created by	Stages	
<span>✓ Passed</span> 🕒 00:03:21 📅 4 days ago	<a href="#">[CHANGE] paginator</a> #1542992303  main  f1441767 <span>latest</span>		<span>✓</span> <span>✓</span>	▾
<span>✓ Passed</span> 🕒 00:03:31 📅 5 days ago	<a href="#">[ADD] add webcam to screen recording</a> #1538763703  main  5122854d		<span>✓</span> <span>✓</span>	▾
<span>✓ Passed</span> 🕒 00:03:33 📅 6 days ago	<a href="#">[CHANGE] (cta) input size</a> #1538268101  main  8b1faf2b		<span>✓</span> <span>✓</span>	▾
<span>✓ Passed</span> 🕒 00:03:37 📅 1 week ago	<a href="#">Update .gitlab-ci.yml file</a> #1536313262  main  c7a016d9		<span>✓</span> <span>✓</span>	▾
<span>✓ Passed</span> 🕒 00:03:22 📅 2 weeks ago	<a href="#">Update .gitlab-ci.yml file</a> #1523617341  main  10c5f3da		<span>✓</span> <span>✓</span>	▾  ▾
<span>⚠ Warning</span> 🕒 00:02:43 📅 2 weeks ago	<a href="#">Update .gitlab-ci.yml file</a> #1523599993  main  a861c064		<span>✓</span> <span>⚠</span>	▾   ▾

## [CHANGE] paginator

 **Passed** Falk M created pipeline for commit `f1441767`  4 days ago, finished 4 days ago

For `main`



**latest**  3 jobs  3.97  3 minutes 21 seconds, queued for 1 seconds


**Pipeline**

Jobs **3**



Tests **0**

### build

 build-job-node 

 build-job-php 















### deploy

 deploy-job-dev 



# Artifacts

Total artifacts size 0 B

<input type="checkbox"/>	Artifacts	Job	Size	Created	
<input type="checkbox"/>	▼ 3 files	 deploy-job-dev GO #1542992303 -o f1441767 📁 main	36.99 MiB	4 days ago	  
<input type="checkbox"/>	dev.zip <span>archive</span>			36.97 MiB	 
<input type="checkbox"/>	metadata.gz <span>metadata</span>			202 B	 
<input type="checkbox"/>	job.log <span>trace</span>			19.57 KiB	 
<input type="checkbox"/>	> 1 file	 build-job-node GO #1542992303 -o f1441767 📁 main	5.47 KiB	4 days ago	  

```
1 stages:
2   - build
3   - deploy
4
5 workflow:
6   rules:
7     - if: '$CI_PIPELINE_SOURCE == "push" && $CI_COMMIT_REF_NAME == "main"'
8       when: always # Run pipeline for push event on main branch
9     - when: never # Prevent pipeline for all other cases
10
11 .frontend_node_modules-cache: &frontend_node_modules-cache
12   key:
13     files:
14       - 'frontend/package-lock.json'
15       - 'frontend/package.json'
```

```
1 stages:
2   - build
3   - deploy
4
5 workflow:
6   rules:
7     - if: '$CI_PIPELINE_SOURCE == "push" && $CI_COMMIT_REF_NAME == "main"'
8       when: always # Run pipeline for push event on main branch
9     - when: never # Prevent pipeline for all other cases
10
11 .frontend_node_modules-cache: &frontend_node_modules-cache
12   key:
13     files:
14       - 'frontend/package-lock.json'
15       - 'frontend/package.json'
```

```
7     when: always # Run pipeline for push event on main branch
8     - when: never # Prevent pipeline for all other cases
9
10
11 .frontend_node_modules-cache: &frontend_node_modules-cache
12   key:
13     files:
14       - 'frontend/package-lock.json'
15       - 'frontend/package.json'
16   paths:
17     - 'frontend/node_modules'
18   policy: pull
19
20 .api_composer-cache: &api_composer-cache
21   key:
22     files:
```

```
16 paths:
17   - 'frontend/node_modules'
18 policy: pull
19
20 .api_composer-cache: &api_composer-cache
21 key:
22   files:
23     - 'api/composer.json'
24     - 'api/composer.lock'
25 paths:
26   - 'api/vendor'
27 policy: pull
28
29 .frontend_build-cache: &frontend_build-cache
30 key: build-$CI_PIPELINE_ID
31 paths:
```

```
24     - 'api/composer.lock'
25   paths:
26     - 'api/vendor'
27   policy: pull
28
29 .frontend_build-cache: &frontend_build-cache
30   key: build-$CI_PIPELINE_ID
31   paths:
32     - frontend/dist
33   policy: pull
34
35 build-job-php:
36   stage: build
37   image: php:latest
38   cache:
```

```
30 key: build-${CI_PIPELINE_ID}
31 paths:
32   - frontend/dist
33 policy: pull
34
35 build-job-php:
36   stage: build
37   image: php:latest
38   cache:
39     - <<: *api_composer-cache
40     policy: pull-push
41   before_script:
42     - apt-get update
43     - apt-get install zip unzip
44     - cd api
```

```
37 image: php:latest
38 cache:
39   - <<: *api_composer-cache
40   policy: pull-push
41 before_script:
42   - apt-get update
43   - apt-get install zip unzip
44   - cd api
45   - php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
46   - php composer-setup.php
47   - php -r "unlink('composer-setup.php');"
48 script:
49   - echo "Compiling the code..."
50   - php composer.phar install
51   - echo "Compile complete."
```



```
42     apt-get update
43     - apt-get install zip unzip
44     - cd api
45     - php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
46     - php composer-setup.php
47     - php -r "unlink('composer-setup.php');"
48     script:
49     - echo "Compiling the code..."
50     - php composer.phar install
51     - echo "Compile complete."
52
53     build-job-node:
54     stage: build
55     image: node:latest
56     cache:
57     - *front-end-node-modules-cache
```

```
49     echo "Compiling the code..."
50     - php composer.phar install
51     - echo "Compile complete."
52
53 build-job-node:
54     stage: build
55     image: node:latest
56     cache:
57         - <<: *frontend_node_modules-cache
58           policy: pull-push
59         - <<: *frontend_build-cache
60           policy: push
61     script:
62         - echo "Compiling the code..."
63         - cd frontend
64         - npm install
```

```
52
53 build-job-node:
54   stage: build
55   image: node:latest
56   cache:
57     - <<: *frontend_node_modules-cache
58       policy: pull-push
59     - <<: *frontend_build-cache
60       policy: push
61   script:
62     - echo "Compiling the code..."
63     - cd frontend
64     - npm install
65     - npm run build
66     - echo "Compile complete."
```

```
1 deploy-job-dev:
2   environment: dev
3   extends: .deploy-job
4   artifacts:
5     name: "dev"
6     paths:
7       - frontend.zip
8       - api.zip
9     when: on_success
10
11 deploy-job-live:
12   environment: production
13   extends: .deploy-job
14   when: manual
15   artifacts:
```

```
8     api.zip
9     when: on_success
10
11 deploy-job-live:
12   environment: production
13   extends: .deploy-job
14   when: manual
15   artifacts:
16     name: "prod"
17     paths:
18       - frontend.zip
19       - api.zip
20   when: on_success
21
22 .deploy-job:
23   stage: deploy
```

```
20   when: on_success
21
22 .deploy-job:
23   stage: deploy
24   image: ubuntu:latest
25   dependencies:
26     - build-job-php
27     - build-job-node
28   cache:
29     - <<: *api_composer-cache
30     - <<: *frontend_node_modules-cache
31     - <<: *frontend_build-cache
32   rules:
33     - if: '$CI_COMMIT_REF_NAME == "main"'
34   before_script:
35     - apt-get update
```

```
33     - if: '$CI_COMMIT_REF_NAME == "main"'
34   before_script:
35     - apt-get update
36     - apt-get install zip unzip
37     - 'which ssh-agent || ( apt-get update -y && apt-get install openssh-client -y
    )'
38     - mkdir -p ~/.ssh
39     - eval $(ssh-agent -s)
40     - '[[ -f /.dockerenv ]] && echo -e "Host *\n\tStrictHostKeyChecking no\n\n" >
    ~/.ssh/config'
41     - (echo "$SSH_HOST") > ~/.ssh/known_hosts
42     - chmod 644 ~/.ssh/known_hosts
43     - ssh-add <(echo "$SSH_PRIVATE_KEY_BASE64" | base64 -d)
44   script:
```

```
50     eval "$(ssh-agent -s)"
40     - '[[ -f /.dockerenv ]] && echo -e "Host *\n\tStrictHostKeyChecking no\n\n" >
~/ssh/config'
41     - (echo "$SSH_HOST") > ~/.ssh/known_hosts
42     - chmod 644 ~/.ssh/known_hosts
43     - ssh-add <(echo "$SSH_PRIVATE_KEY_BASE64" | base64 -d)
44     script:
45     - echo "Replace token"
46     - find ./frontend/dist/deployment -iname "main.*.js" | xargs sed -i -e "s~\
[API_URL\]~$API_URL~g"
47     - echo "compress artefacts"
48     - zip -qr frontend.zip frontend/dist/deployment
49     - zip -qr api.zip api
50     - echo "upload artefacts"
51     - current_date_time=$(date +"%Y%m%d_%H%M%S")
```



```
42 - chmod 644 ~/.ssh/known_hosts
43 - ssh-add <(echo "$SSH_PRIVATE_KEY_BASE64" | base64 -d)
44 script:
45 - echo "Replace token"
46 - find ./frontend/dist/deployment -iname "main.*.js" | xargs sed -i -e "s~\
[API_URL\]~$API_URL~g"
47 - echo "compress artefacts"
48 - zip -qr frontend.zip frontend/dist/deployment
49 - zip -qr api.zip api
50 - echo "upload artefacts"
51 - current_date_time=$(date +"%Y%m%d_%H%M%S")
52 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "mkdir $TEMP_DIR/$current_date_time"
53 - scp -P$SSH_PORT -r api.zip "$SSH_USER@$SSH_HOST:$TEMP_DIR/$current_date_time"
54 - scp -P$SSH_PORT -r frontend.zip
"$SSH_USER@$SSH_HOST:$TEMP_DIR/$current_date_time"
```

```
46     find ../frontend/dist/deployment -iname main.js | xargs sed -i 's/\([API_URL\]~\)$API_URL~g"
47     - echo "compress artefacts"
48     - zip -qr frontend.zip frontend/dist/deployment
49     - zip -qr api.zip api
50     - echo "upload artefacts"
51     - current_date_time=$(date +"%Y%m%d_%H%M%S")
52     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "mkdir $TEMP_DIR/$current_date_time"
53     - scp -P$SSH_PORT -r api.zip "$SSH_USER@$SSH_HOST:$TEMP_DIR/$current_date_time"
54     - scp -P$SSH_PORT -r frontend.zip
55     "$SSH_USER@$SSH_HOST:$TEMP_DIR/$current_date_time"
56     - echo "unzip artefacts"
57     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "mkdir $TEMP_DIR/$current_date_time/src"
58     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "unzip -q
59     $TEMP_DIR/$current_date_time/api.zip -d $TEMP_DIR/$current_date_time/src"
```

```
52 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "mkdir $TEMP_DIR/$current_date_time"
53 - scp -P$SSH_PORT -r api.zip "$SSH_USER@$SSH_HOST:$TEMP_DIR/$current_date_time"
54 - scp -P$SSH_PORT -r frontend.zip
"$SSH_USER@$SSH_HOST:$TEMP_DIR/$current_date_time"
55 - echo "unzip artefacts"
56 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "mkdir $TEMP_DIR/$current_date_time/src"
57 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "unzip -q
$TEMP_DIR/$current_date_time/api.zip -d $TEMP_DIR/$current_date_time/src"
58 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "unzip -q
$TEMP_DIR/$current_date_time/frontend.zip -d $TEMP_DIR/$current_date_time/src"
59 - echo "override application"
60 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/frontend/dist/deployment/. $APP_PATH/frontend"
61 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/api/src/. $APP_PATH/api/src"
```

```
$TEMP_DIR/$current_date_time/frontend.zip -d $TEMP_DIR/$current_date_time/src"
59   - echo "override application"
60   - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/frontend/dist/deployment/. $APP_PATH/frontend"
61   - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/api/src/. $APP_PATH/api/src"
62   - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/api/migrations/. $APP_PATH/api/migrations"
63   - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/api/templates/. $APP_PATH/api/templates"
64   - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/api/vendor/. $APP_PATH/api/vendor"
65   - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
$TEMP_DIR/$current_date_time/src/api/config/. $APP_PATH/api/config"
66   - echo "remove unzip path"
```

```
63     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
      $TEMP_DIR/$current_date_time/src/api/templates/. $APP_PATH/api/templates"
64     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
      $TEMP_DIR/$current_date_time/src/api/vendor/. $APP_PATH/api/vendor"
65     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
      $TEMP_DIR/$current_date_time/src/api/config/. $APP_PATH/api/config"
66     - echo "remove unzip path"
67     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "rm -rf
      $TEMP_DIR/$current_date_time/src"
68     - echo "execute db migration ans text migration"
69     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "/usr/bin/php81
      $APP_PATH/api/bin/console doctrine:migrations:migrate"
70     - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "/usr/bin/php81
      $APP_PATH/api/bin/console app:migrate-text"
71     - echo "Application successfully deployed."
```

```
63 - ssh -p$SSH_PORT $SSH_USER@$SSH_HOST "cp -ru
    $TEMP_DIR/$current_date_time/src/api/templates/. $APP_PATH/api/templates"
64 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
    $TEMP_DIR/$current_date_time/src/api/vendor/. $APP_PATH/api/vendor"
65 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "cp -ru
    $TEMP_DIR/$current_date_time/src/api/config/. $APP_PATH/api/config"
66 - echo "remove unzip path"
67 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "rm -rf
    $TEMP_DIR/$current_date_time/src"
68 - echo "execute db migration ans text migration"
69 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "/usr/bin/php81
    $APP_PATH/api/bin/console doctrine:migrations:migrate"
70 - ssh -p$SSH_PORT "$SSH_USER@$SSH_HOST" "/usr/bin/php81
    $APP_PATH/api/bin/console app:migrate-text"
71 - echo "Application successfully deployed."
```

# THANKS

[print view](#)