Private DEV sample for extended wa-config plugin

Copyright © MONWOO 2022, service@monwoo.com

moonkiosk.monwoo.com/en/missions/wa-config-monwoo/

wa-config dev generator is available under private missions.

This file is an example of the developpment readme you will get on top of our wa-config starter base if you work with us.

moonkiosk.monwoo.com/en/produit/consulting-for-wordpress-blogs-or-woocommerce

To support us and/or help us on top of our missions, send us subventions with :

www.monwoo.com/don

Build by **Miguel Monwoo** in 2022, [**service@monwoo.com**](service@monwoo.com)

This documentation describe the private source that can be buy on top of regular products. It's **price will be ajusted to the resources needed** to produce an open source code open enough for your usage.

Indeed, our web solutions only focus on the end user result with productions builds only.

For specific use case, we can book workloads on some of our experts to produce and deliver the source code linked to the features targeted by the use case.

Those **workloads starts at 80€/hr**. **Only some budget estimations** could be done on some products that do not exist yet...

moonkiosk.monwoo.com/en/produit/consulting-for-wordpress-blogs-or-woocommerce

Web Agency Certificate **Owner** : miguel.monwoo.com

Web Agency Certificate **Demo** : web-agency.cert.monwoo.com

Web Agency Certificate **Production build target from 2022-01-28 to 2022-05-29** : www.web-agency.app

Wa-config **Production build targets** :   2 / 9

- moonkiosk.monwoo.com
- www.web-agency.app
- axelo.web-agency.app
- wa-blog-basic.web-agency.app
- wa-e-com.web-agency.app

Web Agency Certificate **Review Monitor** : axelo.web-agency.app

Plugin developpement :

```bash
WA_PLUGIN=wa-config
WA_TOOLS_DIR=tools
# Going to the plugin dir if you get this code from a full WebAgency
starter :
cd e-commerce/wp-content/plugins/$WA_PLUGIN
# Or from a bundle of plugins :
cd $WA_PLUGIN

# Link the plugin source folder in some wordpress folder to load the dev
version :
# Setup this plugin in WebAgency WP base starter :
cd ../../WebAgencySources/e-commerce/wp-content/plugins
ln -s ../../../../WA-wp-plugins/wa-config $WA_PLUGIN
cd -

# Download getcomposer if not already in tools folder
(cd "$WA_TOOLS_DIR" && curl -sS https://getcomposer.org/installer | php)
alias composer="php '$WA_TOOLS_DIR/composer.phar'"

# Build css from scss to update assets/styles.css
composer compile-scss

# Build js from TypeScript to update assets/app.js
composer compile-ts

# https://wordpress.org/support/article/installing-wordpress-in-your-
language/
# https://codex.wordpress.org/I18n_for_WordPress_Developers

# Update WP cli tool if not already in tools folder :
(cd "$WA_TOOLS_DIR" && curl -O https://raw.githubusercontent.com/wp-
cli/builds/gh-pages/phar/wp-cli.phar)

alias wp="php -d memory_limit=512M '$WA_TOOLS_DIR/wp-cli.phar'"

# Generating .po files from src folder :
wp i18n make-pot . --exclude='src/App.src.php' ./languages/$WA_PLUGIN.pot
--file-comment="🌓🌓 Copyright Monwoo 2022 🌓🌓, build by Miguel Monwoo.
Apache-2.0 Licence. service@monwoo.com.
"

# We use french as base src, so export the fr translations :
cp ./languages/$WA_PLUGIN.pot ./languages/${WA_PLUGIN}-fr_FR.po

# Build translations binary format for wordpress loads
for file in `find ./languages -name "*.po"` ; do msgfmt -o ${file/.po/.mo}
$file ; done
```

```
# Build the doc
( cd "$WA_TOOLS_DIR" && curl -sSLO 'https://phpdoc.org/phpDocumentor.phar'
)
alias phpdoc="php '$WA_TOOLS_DIR/phpDocumentor.phar'"
phpdoc
# or with composer, just do :
composer compile-doc

# Download codecept if not already in tools folder
curl https://codeception.com/codecept.phar > "$WA_TOOLS_DIR/codecept.phar"

# link codecpt tool
alias codecept="php '$WA_TOOLS_DIR/codecept.phar'"
# setup our wp loader absolut path env variable for CLI end to end tests
export WA_STANDALONE_WP_LOADER_FOR_E2E_CLI="../../WebAgencySources/e-
commerce/wp-load.php"

# bootstrap tests if none :
codecept bootstrap

# running the tests :
codecept run
# running one test only :
codecept run -- 'acceptance' Frontend/E2E_EnsureFooterCreditsCept
# verbose output for test run :
codecept -vvv run -- 'acceptance' E2E_EnsureAdminConfigPanelCest
# https://codeception.com/docs/reference/Commands
# https://codeception.com/docs/07-AdvancedUsage
# Add a test (simple plain PHP format) :
codecept generate:cept acceptance Frontend/E2E_EnsureFooterCredits
# Add a test (customisable test class format) :
codecept g:cest acceptance E2E_EnsureAdminConfigPanel

# 🌓🌗  Short cut to test specific test only 🌓🌗
composer e2e -- 'acceptance' 'Frontend/E2E_EnsureFooterCreditsCept'

composer e2e -- 'acceptance' 'E2E_EnsureAdminConfigPanelCest'

composer e2e -- 'acceptance'
'E2E_EnsureAdminConfigPanelCest::testingWAConfig_panelsAvailability'

# Force test ends if for some reason your last test run did not end :
curl -H 'wa-e2e-test-mode: wa-config-e2e-tests' \
"https://web-agency.local.dev/e-commerce/wp-admin/admin-ajax.php?
action=wa-e2e-test-action&wa-action=force-clean-and-restore-users"

# if you wan you IDE to load class etc, use vendor, but
# be aware that vendor will load instead of .phar for codecept tool
composer require --dev codeception/codeception
composer require --dev codeception/module-phpbrowser \
 codeception/module-asserts codeception/module-rest
# https://codeception.com/docs/02-GettingStarted#Interactive-Pause
```

```
composer require --dev hoa/console # Abandoned package ? replacement ?
composer e2e -- 'acceptance'
'E2E_EnsureAdminConfigPanelCest::testingWAConfig_emailTrackingForComments'
--html --debug
```

```
# use  --no-redirect to avoid codecption loads from ./vendor folder :
composer e2e -- 'acceptance'
'E2E_EnsureAdminConfigPanelCest::testingWAConfig_paramsSaveFromAdmin' --
html --no-redirect
open tests/_output/report.html

# with codcept in vendor, plugin site launch of acceptance may fail
# so remove vendor to test the web launch of acceptances tests :
mv vendor _vendor

# Or you can simply remove the dev dependencies clashing with local
codeception ?
composer remove --dev codeception/codeception codeception/module-
phpbrowser \
 codeception/module-asserts codeception/module-rest hoa/console

# Build the production plugin for clients delivery :
alias yakpro-po="php '$WA_TOOLS_DIR/yakpro-po/yakpro-po.php'"
yakpro-po --config-file ./tools/yakpro-po.cnf.php src/App.src.php -o
App.php
composer archive --format=zip

# 🌒🌒 Short cut to generate Zip production build 🌒🌒
composer dist

# to clean builds and dist files
composer clean

# to also clean vendors files (will need full dev re-install)
composer clean-full

# Save git logs as csv :
git log --all --date=iso --pretty=format:'"%h","%an","%ad","%s"' | g\rep
'\[wa-config\]' > miguel-monwoo-git-logs.csv

# List your git fetch and pull urls :
git remote -v
```

Wa-config Static JS head deployer :

```
# Pre-render a web-url as a static website (some links may have to be
updated...)
HOMEBREW_NO_AUTO_UPDATE=1 brew install wget

# Prepare the output directory :
mkdir build

# Example 1 : Re-deploy our head example :
zip -r build/static.zip heads/test-custom-frontend

# Example 2 : fetch a static build to heads/web-agency.cert.monwoo.com
from online url :
WA_FRONTEND_HOST="https://web-agency.cert.monwoo.com"
wget --mirror -w 2 --page-requisites --adjust-extension --convert-links
"$WA_FRONTEND_HOST" -P heads

# Example 2 : zip it to deploy it :
zip -r build/static.zip heads/web-agency.cert.monwoo.com

# https://www.npmjs.com/package/n
# Manage globally your local version of node :
npm install -g n

# switch node version for our current command only
# (npm will use global node,
# so know what you do with this alias before using it...) :
# alias node="n run 16"

# Switch node version globally :
sudo n 18

# Install dependencies :
# npm install -D 'node-fetch@^2.6.7' form-data dotenv
# npm install -D cryptr chrome-remote-interface chrome-launcher
# Already installed if you use our package.json, just do :
npm install


# 🌓🌒 Deploy locale folder heads/test-custom-frontend to remote 🌕🌘
( export WA_FRONTEND_HOST="https://web-agency.local.dev"
export WA_HEAD_TARGET="my-first-deploy"
export WA_USER_LOCATION="my-laptop-from-france-in-2022"
export DEBUG=false
export WA_SSL_ALLOW_SELFSIGNED=true
export WA_BACKEND_HOST="$WA_FRONTEND_HOST/e-commerce"
export WA_REST_API_SERVER="$WA_BACKEND_HOST/wp-json/wa-config/v1"
node tools/wa-deploy.cjs)

# Debug tips :
# https://stackoverflow.com/questions/66724333/is-there-a-way-to-log-the-
```

```
node-fetch-request
# NODE_DEBUG=http
# NODE_DEBUG=http,net,stream
# (Use `node --trace-warnings ...` to show where the warning was created)
# https://code.visualstudio.com/docs/nodejs/nodejs-debugging#_attaching-
to-nodejs
# node --inspect program.js # attach the VS Code debugger to an external
Node.js program
# node --inspect-brk program.js # if the program shouldn't start running,
but must wait for the debugger
# Then you need to type cmd-shift-p to access commands to the related
terminal you just
# launch your command from.
# Look for command : 'Debug : Attach to Node Process' => click and follow
;)
( export WA_FRONTEND_HOST="https://web-agency.local.dev"
export WA_HEAD_TARGET="my-first-deploy"
export WA_USER_LOCATION="my-laptop-from-france-in-2022"
export DEBUG=true
export WA_SSL_ALLOW_SELFSIGNED=true
export WA_BACKEND_HOST="$WA_FRONTEND_HOST/e-commerce"
export WA_REST_API_SERVER="$WA_BACKEND_HOST/wp-json/wa-config/v1"
node --inspect-brk tools/wa-deploy.cjs)
```

Biblio

- https://www.php.net/manual/fr/language.oop5.traits.php#language.oop5.traits.conflict
- https://blog.martinhujer.cz/how-to-use-data-providers-in-phpunit
- https://www.sitepoint.com/how-to-translate-a-wordpress-plugin-description/

Thanks

Thanks for buying our products.

For any questions or suggestion, please contact :

- Monwoo : service@monwoo.com
- Web Agency Team : mail@web-agency.app

To support us and help us open more software, send us a subvention with :

www.monwoo.com/don