Your Code Runs in Docker, Why Not Your IDE?

Thomas Bley, June 2021 @ ipc

About me

- Senior PHP Developer
- Linux, PHP, MySQL since 2001
- studied at TU München
- working for Bringmeister in Berlin

Why use Docker?

- easy to manage different isolated environments
- quick and easy to setup development environments
- easy to share with other developers
- easy to integrate in continuous integrations
- easy to test new things
- write once, run anywhere
- big ecosystem with Docker Hub

➔ we love to run our code in Docker

Why use an IDE?

IDE = Integrated Development Environment

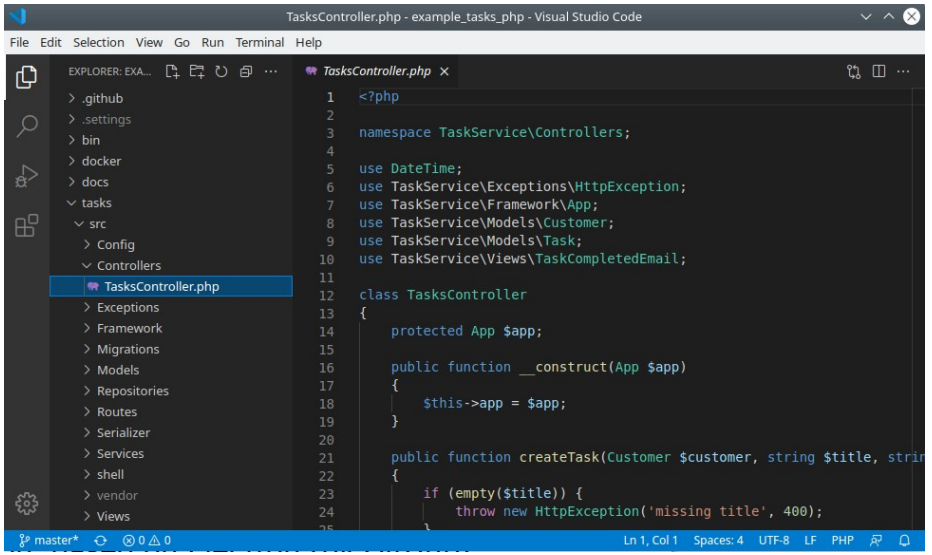
- flexible text editor
- code completion, formatting
- automatic checks for errors, debugging
- integrated version control
- many other features with extensions

➔ more productivity, more quality

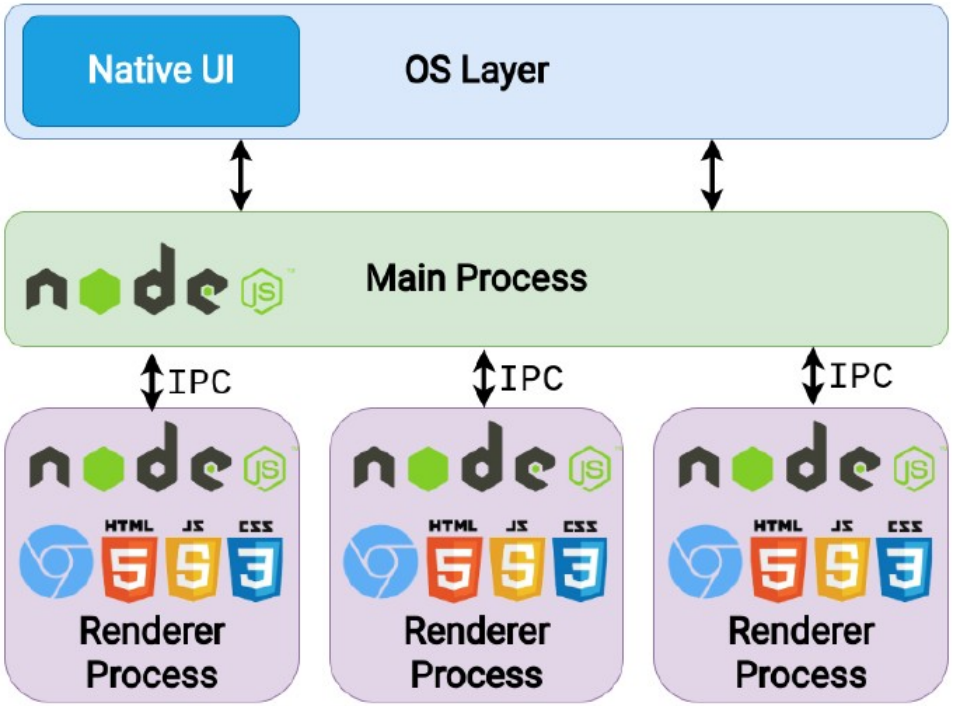
but: installed as desktop application, configured manually

What is Visual Studio Code?

- free and open source IDE from Microsoft
- lightweight, very fast
- supports many programming languages
- big community
- many extensions
- desktop application, based on Electron (by GitHub)
- marketplace is not open source, proprietary license



What is Electron?

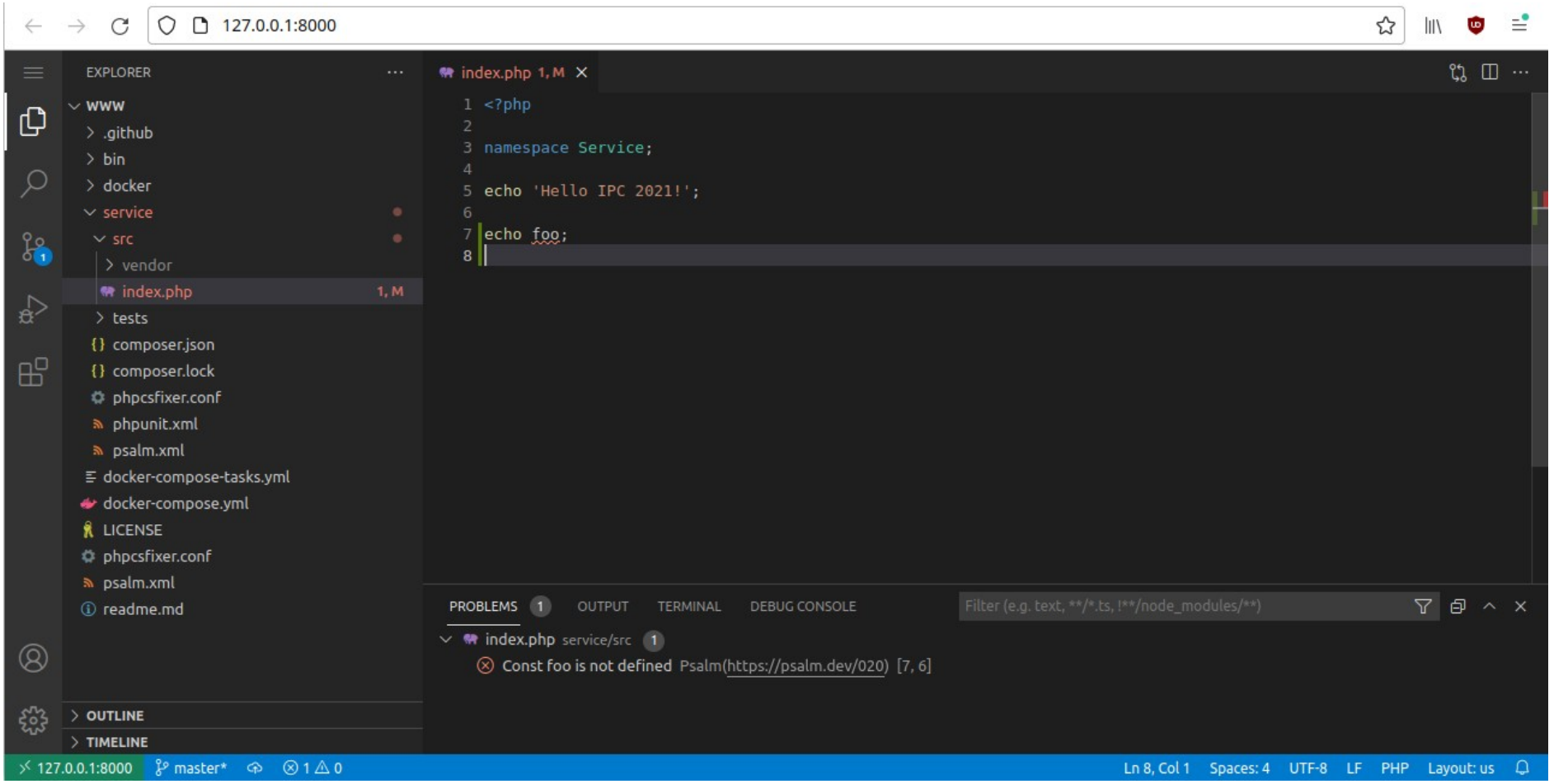


Framework for cross platform desktop applications from GitHub

- combines the Chromium rendering engine (Frontend)
- with node.js runtime (Backend)

source:
livecodestream.dev/post/how-to-build-desktop-applications-using-electron-the-right-way/

Can we run VS Code in the browser?

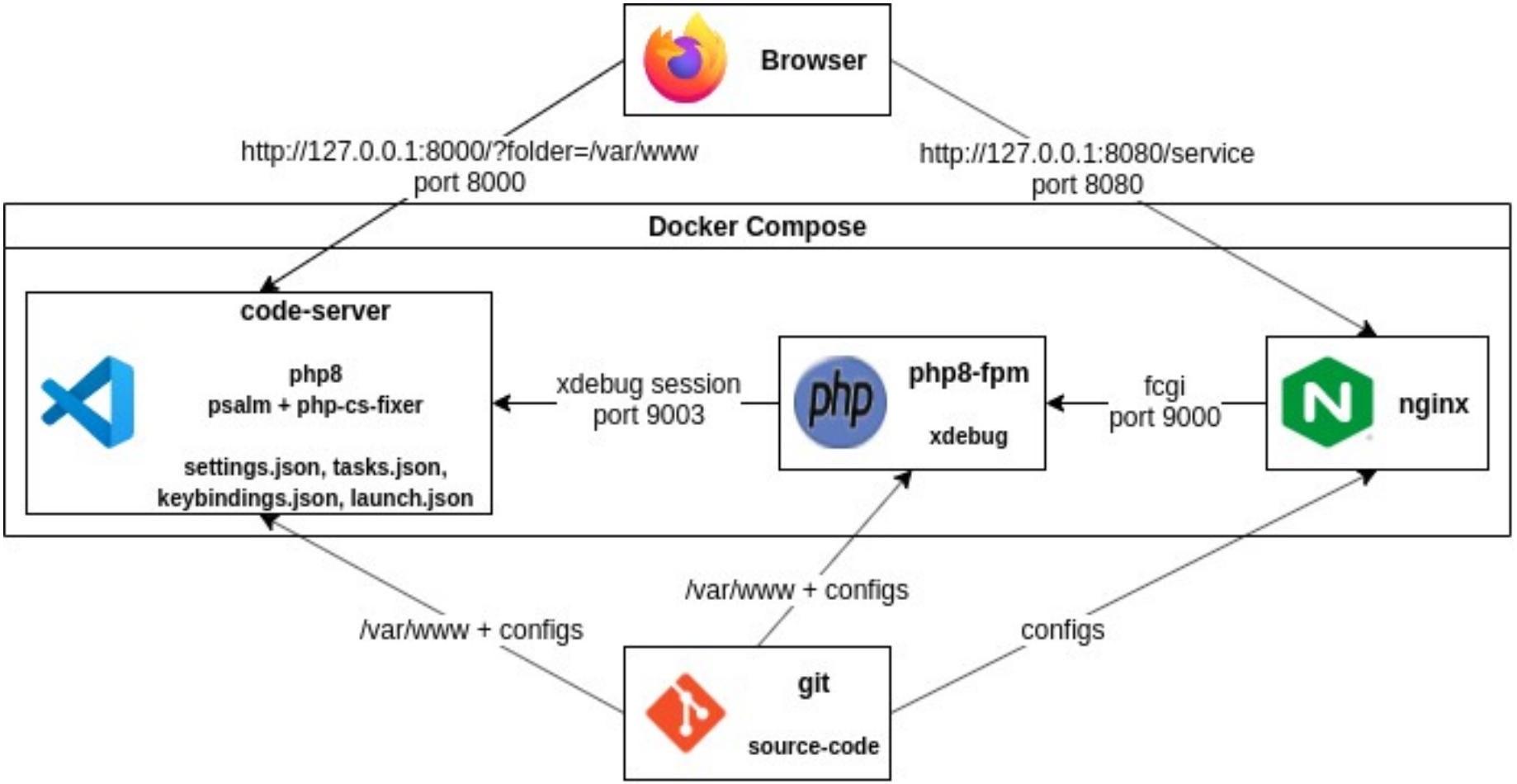


Code-Server

- “VS Code in the browser”
- fork of VS Code, open source
- runs backend part on server, frontend part in browser
- supports VSX extensions
- can be installed as Debian package (94M)
- not allowed to use VS Code Marketplace
- uses Open VSX Registry as marketplace (open-vsx.org)

sources:
github.com/cdr/code-server
github.com/cdr/code-server/blob/main/docs/FAQ.md

Setup


Browser
http://127.0.0.1:8000/?folder=/var/www
port 8000
http://127.0.0.1:8080/service
port 8080
Docker Compose
code-server
php8
psalm + php-cs-fixer
settings.json, tasks.json,
keybindings.json, launch.json
xdebug session
port 9003
php8-fpm
xdebug
fcgi
port 9000
nginx
/var/www + configs
/var/www + configs
configs
git
source-code

Dockerfile

```
# Setup a Debian container with PHP 8.0, Composer and code-server

FROM debian:buster-slim
RUN apt-get -y update \
    && DEBIAN_FRONTEND=noninteractive apt-get -y --no-install-recommends install apt-transport-https curl \
       ca-certificates \
    && echo "deb https://packages.sury.org/php/ buster main" >/etc/apt/sources.list.d/ondrej-debian-php.list \
    && curl -s https://packages.sury.org/php/apt.gpg >/etc/apt/trusted.gpg.d/php.gpg \
    && apt-get -y update \
    && DEBIAN_FRONTEND=noninteractive apt-get -y --no-install-recommends install git php8.0-cli \
        php8.0-mbstring php8.0-curl php8.0-xml php8.0-zip \
    && curl -#L -o /tmp/code-server.deb \
       https://github.com/cdr/code-server/releases/download/v3.10.0/code-server_3.10.0_amd64.deb \
    && dpkg -i /tmp/code-server.deb \
    && curl -#L -o /usr/bin/composer https://getcomposer.org/download/2.0.13/composer.phar \
    && chmod +x /usr/bin/composer

ARG UID
RUN useradd -m docker -u ${UID} -d /config -s /bin/bash
USER docker
```

Dockerfile #2

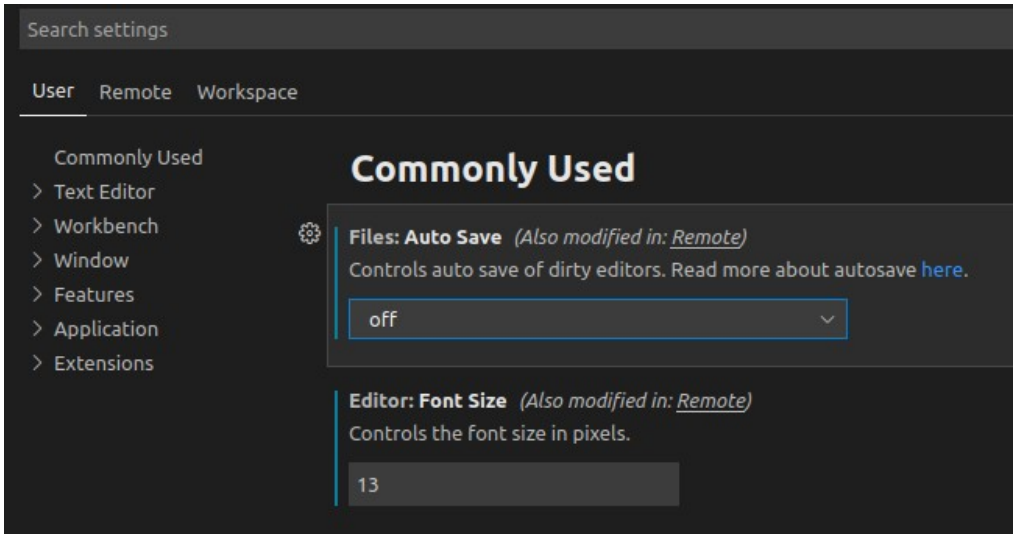
```
# Add composer libraries, install VS Code extensions, start code-server

ENV SERVICE_URL=https://open-vsx.org/vscode/gallery
ENV ITEM_URL=https://open-vsx.org/vscode/item

RUN composer global require --no-suggest --no-progress --no-cache vimeo/psalm friendsofphp/php-cs-fixer

ADD getpsalm.psalm-vscode-plugin-1.2.2.vsix /tmp/
RUN code-server --install-extension felixfbecker.php-debug \
                --install-extension fterrag.vscode-php-cs-fixer \
                --install-extension /tmp/getpsalm.psalm-vscode-plugin-1.2.2.vsix \
                --install-extension alphabotsec.vscode-eclipse-keybindings \
    && chmod -R a+w /config

WORKDIR /var/www
EXPOSE 8000
CMD [ "code-server", "--auth", "none", "--disable-telemetry", "--disable-update-check", "--bind-addr", "0.0.0.0:8000" ]
```

docker-compose.yml

```
code-server:
  build:
    context: ./docker/code-server
    args: [UID]
  cap_drop: [all]
  userns_mode: "host"
  volumes:
    - ./:/var/www
    - ./bin:/usr/local/sbin
    - ./docker/code-server/settings.json:/config/.local/share/code-server/User/settings.json
    - ./docker/code-server/tasks.json:/config/.local/share/code-server/User/tasks.json
    - ./docker/code-server/keybindings.json:/config/.local/share/code-server/User/keybindings.json
    - ./docker/code-server/launch.json:/var/www/.vscode/launch.json  #xdebug
  ports:
    - "127.0.0.1:8000:8000"
```

Pre-Configure VS Code settings

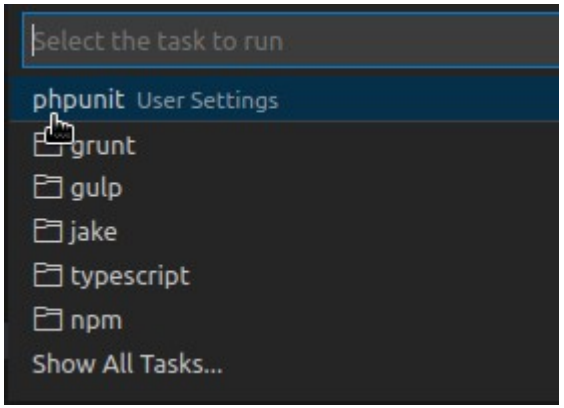
```
# settings.json

{
   "files.autoSave": "afterDelay",
   "editor.minimap.enabled": false,
   "editor.renderIndentGuides": false,
   "editor.fontSize": 13,
   "workbench.colorTheme": "Default Dark+",
   ....
}
```

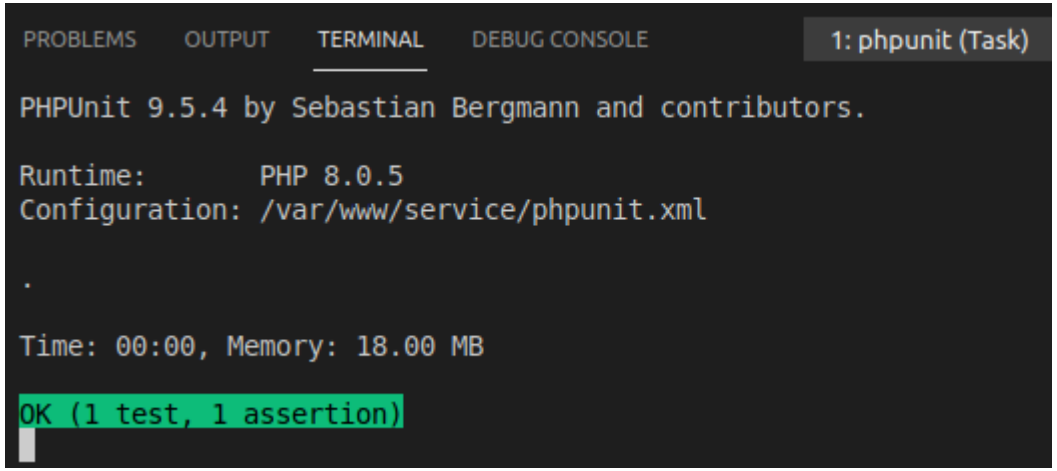



```
git diff docker/code-server/settings.json
+++ b/docker/code-server/settings.json
@@ -35,5 +35,6 @@
+    "files.autoSave": "off"
}
```

Can we run tasks?

```
# tasks.json (Ctrl + Shift + t)

{
   // See https://code.visualstudio.com/docs/editor/tasks#vscode
   "version": "2.0.0",
   "tasks": [{
      "label": "phpunit",
      "type": "shell",
      "problemMatcher": [],
      "command": "phpunit.phar",
      "options": {"cwd": "/var/www/service"},
      "presentation": {
         "showReuseMessage": false,
         "echo": false,
         "clear": true
      }
   ]}
}
```

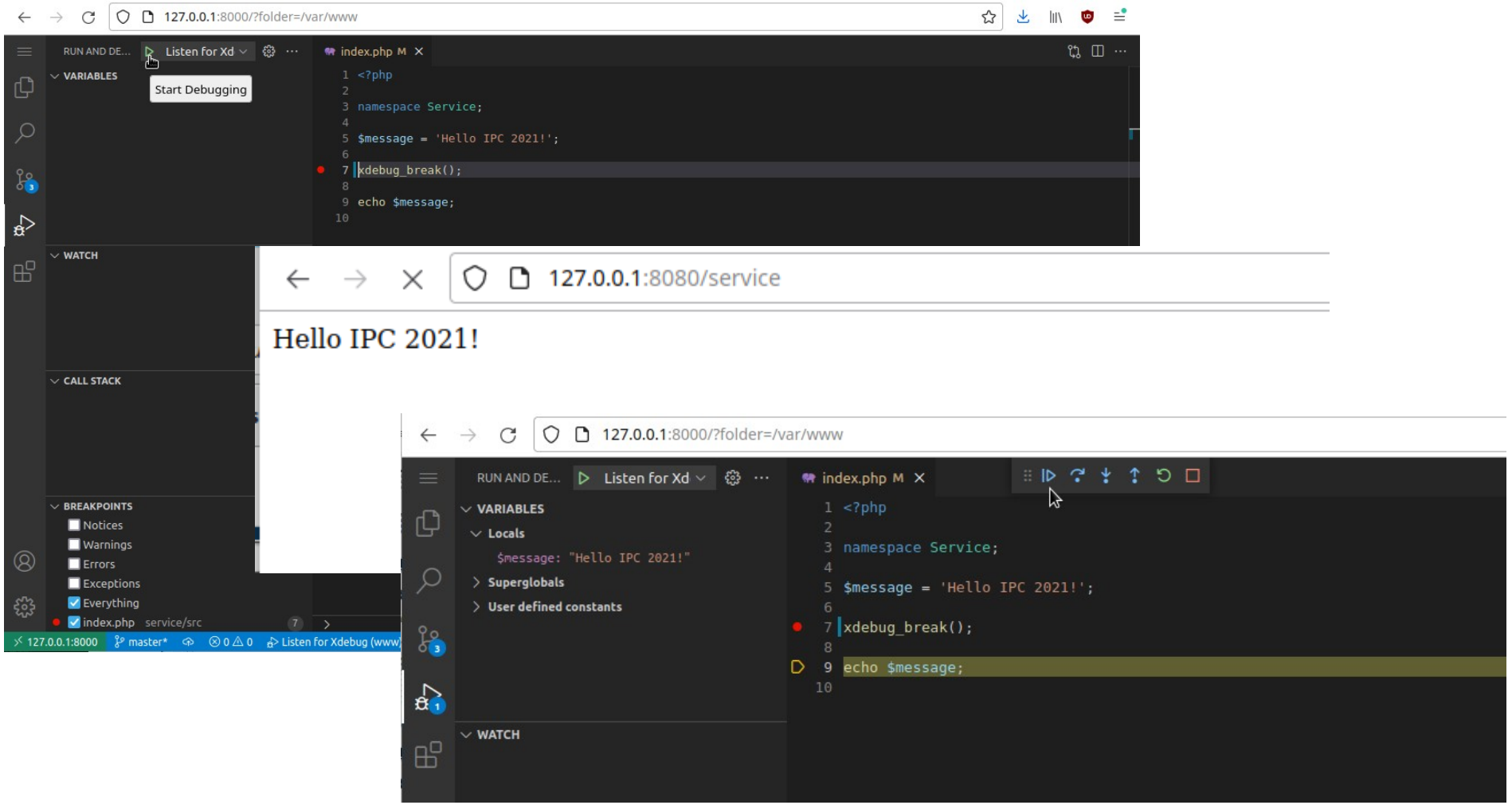




How about keyboard shortcuts?

```
# keybindings.json

[
   {"key": "ctrl+shift+t", "command": "workbench.action.tasks.runTask"},

   // switch file tabs, normally: ctrl+tab
   {"key": "ctrl+[Backquote]", "command":
            "workbench.action.quickOpenLeastRecentlyUsedEditorInGroup"},
   {"key": "ctrl+[Backquote]", "command":
            "workbench.action.quickOpenNavigateNextInEditorPicker",
            "when": "inEditorsPicker && inQuickOpen"},

   // close active file tab, normally: ctrl+w
   {"key": "ctrl+e", "command": "workbench.action.closeActiveEditor"}
]

# Some keyboard shortcuts are exclusively reserved to the browser (e.g. ctrl+tab).

# With Chrome you can use “Menu -> Install code-server” to run code-server
   as a Chrome App to bypass reserved keyboard shortcuts.
```

Debugging (xdebug)



Auto-Completion (psalm language server)

index.php 2, M

```php
<?php

namespace Service;

echo myfoo
```

\Service\myfoo… function Service\myfoobar(int $a…

```php

/** a foobar function */
function myfoobar(int $a, string $b, bool $c): string {
    return sprintf('%s-%b-%c', $a, $b, (int) $c);
}
```

index.php 1, M

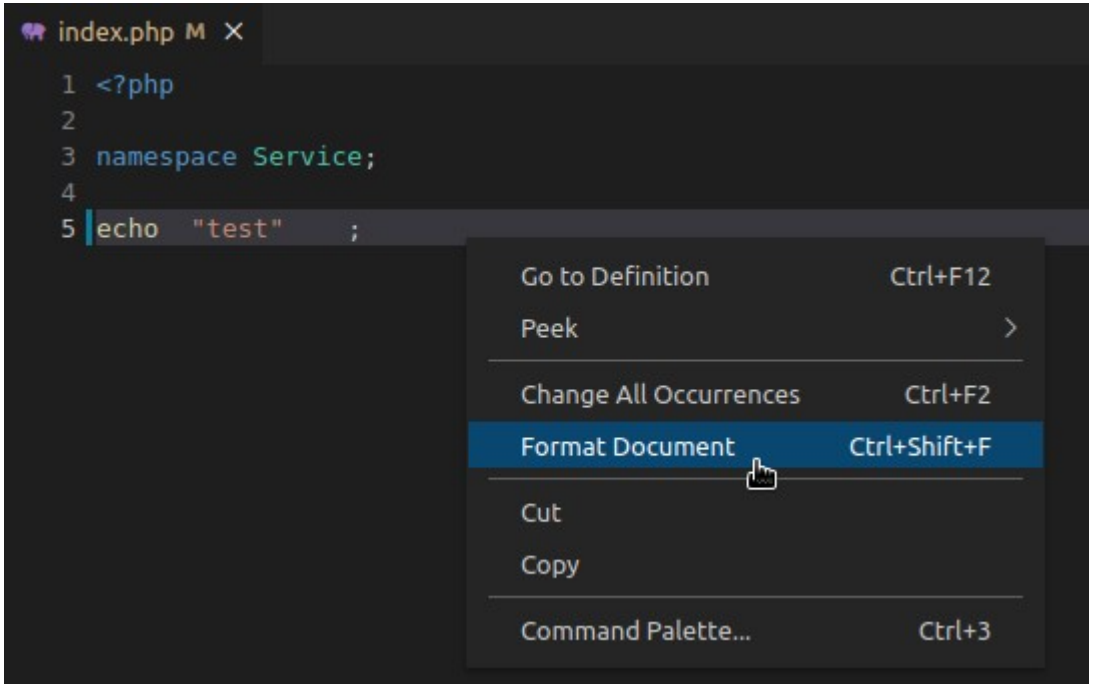
```php
<?php

namespace Service;

```

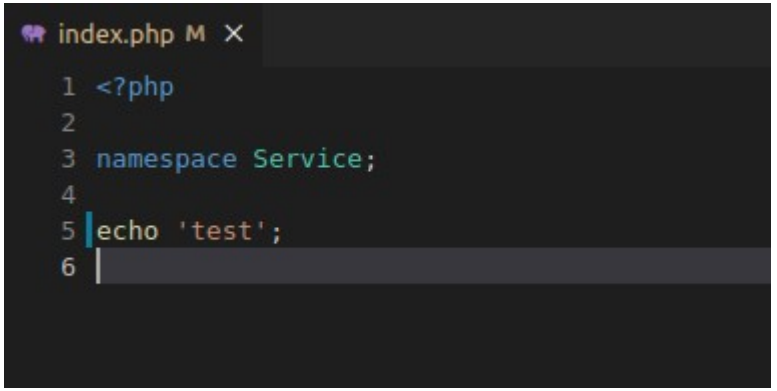
Service\foobar(int $a, string $b, bool $c)

a foobar function

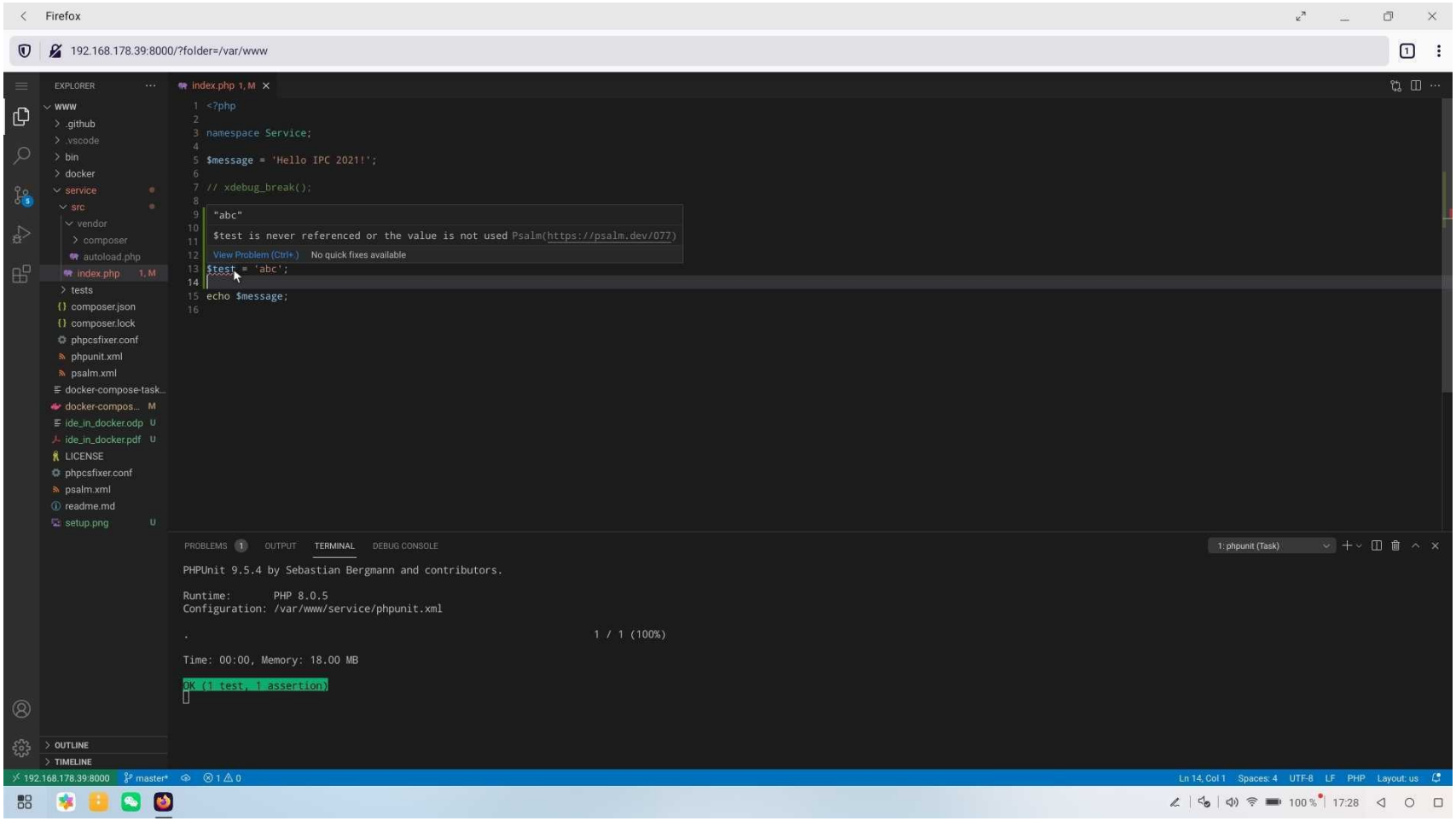
```php
echo foobar();

/** a foobar function */
function foobar(int $a, string $b, bool $c): string {
    return sprintf('%s-%b-%c', $a, $b, (int) $c);
}
```

Auto-Formatting (php-cs-fixer)





DEMO



Firefox 88 on Huawei P40 Desktop mode

Thanks for listening!

Questions?

slides and sources:
github.com/thomasbley/ide_in_docker