

Fábio Dias Hauck - Testing Mundo Wap

Segui o passo a passo do Readme Mundo Wap

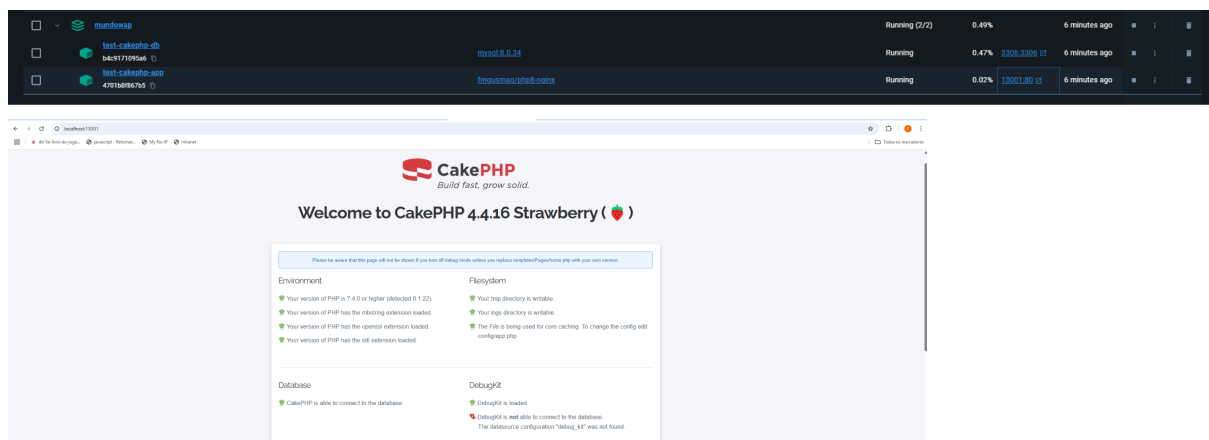
Foi identificado alguns comandos fora de contexto no install e foi comentado para não apagar o container ao instalar as dependências do composer

```
shell > $ install.sh
1  #!/usr/bin/env bash
2
3  #./exec.sh stop
4
5  if [[ "$@" =~ "-rm" ]]; then
6      sudo rm -rf ./app/vendor/
7  fi
8
9  docker compose build --force-rm
10
11  ./exec.sh composer install
12
13  if [[ "$@" =~ "-start" ]]; then
14      ./exec.sh start
15  #else
16  #  ./exec.sh stop
17  fi
18
```

depois de instalar as dependências, rodei o

`./exec.sh start` para rodar os containers.

Com o container rodando,



utilizando `docker exec -it mundowap-test-cakephp-app bash` entrei no container e tentei utilizar o migrate para criar o migrate da tabela Address mas deu problema de permissão, onde no docker-compose vi comentado um user 1000:1000 e então defini o user

como root, e finalmente rodou os migration 3c6664c4e9e4:/var/www/html# bin/cake bake migration CreateAddresses

Creating file /var/www/html/config/Migrations/20250312003735_CreateAddresses.php
Wrote `/var/www/html/config/Migrations/20250312003735_CreateAddresses.php`
3c6664c4e9e4:/var/www/html# bin/cake bake migration CreateVisits

Creating file /var/www/html/config/Migrations/20250312004835_CreateVisits.php
Wrote `/var/www/html/config/Migrations/20250312004835_CreateVisits.php`
3c6664c4e9e4:/var/www/html# bin/cake bake migration CreateWorkdays

Creating file /var/www/html/config/Migrations/20250312004900_CreateWorkdays.php
Wrote `/var/www/html/config/Migrations/20250312004900_CreateWorkdays.php`
3c6664c4e9e4:/var/www/html#

mas em alguns momentos eu tive problema de permissão novamente, ao descer e subir o container, então olhando o reame novamente reparei em um comando de permissão que erroneamente tinha pulado, retirei o user e então rodei novamente o container, seguindo corretamente os passos do readme

com os migrantes já criados na pasta config/Migrations.
Contrui os migrates utilizando o arquivo db_structure como exemplo, e utilizei a documentação <https://book.cakephp.org/migrations/4/pt/index.html> para definir os campos corretamente

utilizando, bin/cake migrations migrate, foi inserida as tabelas no bd.

Iniciando vou criar as controllers também utilizando...

bin/cake bake controller Addresses

bin/cake bake controller Visits

bin/cake bake controller Workdays

utilizando as documentações conferi os padrões utilizados
<https://book.cakephp.org/4/en/development/rest.html> e
<https://book.cakephp.org/4/en/controllers.html>

No segundo dia de desenvolvimento tive problemas na conexão ao banco de dados pelo dbeaver, tive que alterar o plugin de autenticação no banco de dados, para conseguir acessar pelo aplicativo.

```
root@FabioHauck:~/mundo wap/MundoWap# docker exec -it mundowap-test-cakephp-db bash
bash-4.4# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.34 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'admin'@'%' IDENTIFIED WITH 'mysql_native_password' BY 'admin';
Query OK, 0 rows affected (0.02 sec)
```

Como mostra no print anterior, o comando ALTER USER 'admin'@'%' IDENTIFIED WITH 'mysql_native_password' BY 'admin'; resolveu o problema logo depois, criei as rotas para as controllers,

```
$routes->scope('/api', function (RouteBuilder $builder) {
    $builder->setExtensions(['json']);
    $builder->resources('Addresses');
    $builder->resources('Visits');
    $builder->resources('Workdays');

    $builder->get('/csrf-token', ['controller' => 'Addresses', 'action'
=> 'getCsrftoken']);
});
```

e também adicionei a rota pra pegar o csrf que está pedindo, quando fiz o primeiro teste.

```
public function getCsrftoken()
{
    $this->request->allowMethod(['get']);

    $token = $this->request->getAttribute('csrfToken');

    return $this->response
        ->withType('application/json')
        ->withStringBody(json_encode(['csrfToken' => $token]));
}
```

acompanhei as documentações, <https://book.cakephp.org/migrations/4/pt/index.html> para a route, e para csrf <https://book.cakephp.org/4/en/security/csrf.html> e <https://book.cakephp.org/4/en/security/csrf.html#cross-site-request-forgery-csrf-middleware>

Após o teste na route, e “dar certo” no postman, fui para as models, seguindo os padrões dos outros comando, bin/cake bake model ...

```
ec271048a833:/var/www/html# bin/cake bake model Visits
One moment while associations are detected.

Baking table class for Visits...

Creating file /var/www/html/src/Model/Table/VisitsTable.php
Wrote `var/www/html/src/Model/Table/VisitsTable.php`

Baking entity class for Visit...

Creating file /var/www/html/src/Model/Entity/Visit.php
Wrote `var/www/html/src/Model/Entity/Visit.php`

Baking test fixture for Visits...

Creating file /var/www/html/tests/Fixture/VisitsFixture.php
Wrote `var/www/html/tests/Fixture/VisitsFixture.php`
Bake is detecting possible fixtures...

Baking test case for App\Model\Table\VisitsTable ...

Creating file /var/www/html/tests/TestCase/Model/Table/VisitsTableTest.php
Wrote `var/www/html/tests/TestCase/Model/Table/VisitsTableTest.php`
Done
```

```
ec271048a833:/var/www/html# bin/cake bake model Workdays
One moment while associations are detected.

Baking table class for Workdays...

Creating file /var/www/html/src/Model/Table/WorkdaysTable.php
Wrote `var/www/html/src/Model/Table/WorkdaysTable.php`

Baking entity class for Workday...

Creating file /var/www/html/src/Model/Entity/Workday.php
Wrote `var/www/html/src/Model/Entity/Workday.php`

Baking test fixture for Workdays...

Creating file /var/www/html/tests/Fixture/WorkdaysFixture.php
Wrote `var/www/html/tests/Fixture/WorkdaysFixture.php`
Bake is detecting possible fixtures...

Baking test case for App\Model\Table\WorkdaysTable ...

Creating file /var/www/html/tests/TestCase/Model/Table/WorkdaysTableTest.php
Wrote `var/www/html/tests/TestCase/Model/Table/WorkdaysTableTest.php`
Done
```

e o mesmo pro Addresses

A seguir, fui ajustando a controller começando pelo Addresses, tinha dúvidas de padrões de

sintaxe do cakephp que mudam um pouco, e então analisando a documentação

<https://book.cakephp.org/4/pt/controllers/request-response.html>,

<https://book.cakephp.org/4/pt/development/errors.html>,

Ao aplicar algumas regras na controller do endereço, ao consumir a rota via CEP reparei, que no browser funcionava mas na aplicação não, então sabendo que no docker usa por padrão o dns 127.0.0.11 e que ele não é confiável em alguns serviços externos, mudei o dns para o dns do google,

```
app:
  image: fmgusmao/php8-nginx
  container_name: mundowap-test-cakephp-app
  ports:
    - "13001:80"
  dns:
    - 8.8.8.8
    - 8.8.4.4
  volumes:
    - ./app:/var/www/html
  extra_hosts:
    - "host.docker.internal:host-gateway"
  env_file:
    - .env.app
  depends_on:
    - db
  networks:
    - backend
volumes:
```

desci o container e subi o container novamente,

subindo as alterações do endereço e percebendo que o endereço só vai poder ser adicionado quando for registrado uma visita, comecei na tabela Visits que é o “core” do desenvolvimento, então aproveitei o padrão das funções do controller do endereço e adicionei com o mesmo padrão dos cruds feito no addressess deixando sem as regras por enquanto para o “dias úteis” e visitas e em sequência já fui ajustando a tabela de visitas.

Após terminar os cruds dos 3 controllers por enquanto sem pensar muito em regras de negócio, deixando só a regra de endereço pronta, e agora sim vou focar nos Endpoints que deverão ser criados:

- Criação de visitas (visits); add ok
 - regras para serem atendidas
 - associar com o endereço ok
 - função que define a soma de duration e atualiza e atende visits/workdays ok
 - criar um novo registro no workdays se não existir para o data ok
 - Ao criar uma visita em data existente, atualizar os campos do workdays ok
 - limitar até 8 horas por dia ok
- Edição de visitas (visits); edit ok
 - regras para serem atendidas
 - sempre que mudar um endereço deletar e criar um novo ok
 - sempre que mudar os valores de form e produto e a data, atualizar o visits e workday ok
 - limitar até 8 horas ok

- Listagem de dias úteis (workdays); index ok
- Listagem de visitas (visits); ok
 - A requisição deve receber um parâmetro obrigatório cujo valor deve ser uma data que será utilizada para filtrar as visitas de uma determinada data (visits.date); ok

- Fechar dia de trabalho (workdays): ok

Requisição deve receber um parâmetro obrigatório cujo valor deve ser uma data ok

Operação para que todas as visitas pendentes desta data devem ser movidas para o início do próximo dia, caso este não comporte todas as visitas devido ao limite de horas, as últimas visitas desta data devem ser realocadas para o início da próxima data, e assim por diante.

Ao fechar os fluxos, comecei pensando aqui sobre a possibilidade de criar testes automatizados como um bonus, entou criando os testes utilizando o ciclo de teste chamado de AAA que é basicamente configurar os dados, executar os testes e verificar os resultados, pensei em criar um banco de dados novo para realizar os testes e não dar conflito com o banco de dados do app, entao o fiz,

```
db_test:
  container_name: mysql_test
  image: mysql:8.0
  environment:
    MYSQL_ROOT_PASSWORD: root
    MYSQL_DATABASE: test_database
  ports:
    - "3307:3306"
  networks:
    - backend
  tmpfs: /var/lib/mysql
  command: --default-authentication-plugin=mysql_native_password --max-connect-errors=1000 --connect-timeout=60
  restart: always
  depends_on:
    - db
```

Configurei tambem no app.php o banco de teste que criei,

```
/*
 * The test connection is used during the test suite.
 */
'test' => [
    'className' => Connection::class,
    'driver' => Mysql::class,
    'persistent' => false,
    'host' => 'mysql_test',
    'username' => 'root',
    'password' => 'root',
    'database' => 'test_database',
    'encoding' => 'utf8mb4',
    'timezone' => 'UTC',
    'cacheMetadata' => true,
    'quoteIdentifiers' => false,
    'log' => false,
    'init' => ['SET GLOBAL innodb_stats_on_metadata = 0'],
],
```

ajustei o CSRF no Application.php pra não validar nas rotas api, pois não faz sentido CSRF em rotas APIREST e estava atrapalhando os testes.

Pensando em isolar os testes, e torna-los independentes podendo limpar o banco de dados sem preocupação e torna-lo seguro..

São muitos casos de testes para implementar, mas pensei nos principais, e simplificando sem utilizar o Fixture, iniciei pensando em testar os endpoints, começando pelos endpoints de criar uma visita com sucesso, pensando no AAA, configurei os dados da requisição, em seguida, fiz a chamada da route, e verifiquei o statuscode, e o resultado da duração.

logo em seguida fui para o teste onde valido se um workday criado com mais de uma visita vai ter o valor total correto, de acordo com as regras do formulario e do produto, novamente preparei os dados da requisição para as 2 visitas, enviei para a route, e validei os statuscode, a duração de cada visita, e em seguida adicionei uma consulta para trazer os dados Workday pelo dia informado em uma das visitas sabendo que as 2 tem a mesma data, então valido o statuscode e o total da duration.

depois criei um teste para validar se uma visita criada e depois sendo editada, campos editados foram o endereco, a data e vai validar se o id do endereço mudou, ou seja se ele deletou e adicionou um endereço novo, tambem sabendo que a nova data não tem registro, vou validar a duração do work para a nova data que vem no relacionamento da edicao.

depois fiz o teste da rota que buscar uma visita pela data, adicionei uma visita e busquei pela mesma data, pegando a primeira posição avaliei se a duração da visita é valida.

depois fiz testes para ver se a regra que valida no visita se o limite de 480 minutos no dia foi atingido corretamente, fiz um teste passando formularios e produtos que ultrapassavam 480 minutos, validando o statuscode 400 e o aviso do return onde alerta o limite de horas atingido.

depois fiz o teste para fechar um dia de trabalho, seguindo a regra de enviar para o proximo dia as visitas não completadas, e não vai dar mais tempo de continuar, pois tenho o front end para fazer.