

AMBIENTE WEB

Atualização e Deploy para PHP 8 (Linux + Portal CP)

Conteúdo

1	Atualização do Ambiente Web para PHP 8	3
1.1	Configuração do Arquivo config.json (WebSocket)	3
1.2	Parametrização do Diretório PHP 8	4
1.3	Modificação do Arquivo teknisa.conf	4
1.4	Atualização da Porta no Arquivo Nginx (proxy.conf)	4
1.5	Download e Subida dos Containers PHP 8	5
2	Configuração de Ambiente no CP para Deploy	7
2.1	Pré-Requisitos	7
2.2	Configuração do Servidor de Deploy	7
2.3	Vinculação do Produto ao Servidor	8
2.4	Configuração dos Mnemônicos	10
2.4.1	Banco de Dados BIRT	11
2.4.2	Banco de Dados (Oracle)	11
2.4.3	Banco de Dados MongoDB	11
2.4.4	Configurações de E-mail (SMTP)	11
2.4.5	Identificação e QR Code	12
2.4.6	FileServer	12
2.5	Deploy do Produto	12
2.6	Parametrização Zeedhi Next (TecFood e ERP)	14
2.7	Finalização: Reinicialização e Validação do Ambiente	16

Atualização do Ambiente Web para PHP 8

Este documento descreve o procedimento para migração do ambiente web Linux da versão PHP 7 para PHP 8, incluindo a configuração do WebSocket, parametrização dos arquivos de ambiente e subida dos containers Docker.

1.1 CONFIGURAÇÃO DO ARQUIVO CONFIG.JSON (WEBSOCKET)

O arquivo **config.json** é responsável pela conexão do WebSocket da aplicação. Antes de iniciar a migração, é necessário criar e configurar esse arquivo no servidor.

1. Acesse o servidor via terminal e crie o diretório de configuração do WebSocket:

```
mkdir -p /var/www/wsconf/wsphp8
```

2. Acesse o diretório criado:

```
cd /var/www/wsconf/wsphp8
```

3. Crie o arquivo de configuração:

```
touch config.json
```

4. Abra o arquivo para edição:

```
vim config.json
```

5. Insira o conteúdo abaixo no arquivo, substituindo o valor do campo **email** pelo e-mail do cliente.

Exemplo: *nomecliente_prod_php8@teknisa.com*:

```
{
  "email": "client_update@teknisa.com",
  "server": "wss://ws.zeedhi.com/",
  "workfolder": "",
  "path": "/.",
  "windowsService": "",
  "commandReset": "./reset.sh",
  "waitTime": "",
  "logExpirationDay": "",
  "waitTimeProcess": "",
  "watcherLog": "",
  "clientLog": "logs/ws.log"
}
```

1.2 PARAMETRIZAÇÃO DO DIRETÓRIO PHP 8

Após configurar o WebSocket, crie o diretório de trabalho destinado à versão PHP 8.

1. Execute o comando abaixo para criar o diretório:

```
mkdir /var/www/php8
```

1.3 MODIFICAÇÃO DO ARQUIVO TEKNISA.CONF

O arquivo **teknisa.conf** contém referências à versão do PHP utilizada pelo servidor Apache. É necessário substituir todas as ocorrências de `php7` por `php8` neste arquivo.

1. Abra o arquivo com o editor **vim** (ou outro editor de sua preferência) e insira o comando:

```
vim /var/www/apacheconf/teknisa.conf
```

2. Selecione **Shift+:** e digite:

```
:%s/php7/php8/g
```

3. Salve e saia do arquivo com **ESC + Shift+:** e em seguida **x!**.

ATENÇÃO:

O comando `:%s/php7/php8/g` substitui **todas** as ocorrências de `php7` no arquivo. Verifique o conteúdo antes de salvar para garantir que nenhuma referência indevida foi alterada.

1.4 ATUALIZAÇÃO DA PORTA NO ARQUIVO NGINX (PROXY.CONF)

O arquivo **proxy.conf** do Nginx precisa ser atualizado para redirecionar o tráfego para a porta correspondente ao container PHP 8.

1. Abra o arquivo com o editor **vim** (ou outro editor de sua preferência):

```
vim /teknisa/files/nginx/proxy.conf
```

2. Pressione a tecla **i** para entrar no modo de edição.
3. Localize o parâmetro **proxy_pass** e altere a porta de 82 para 83.
Exemplo: proxy_pass http://000.111.222.333:83;
4. Salve e saia do arquivo com **ESC + Shift+:** e em seguida **x!**.

1.5 DOWNLOAD E SUBIDA DOS CONTAINERS PHP 8

Com os arquivos de configuração ajustados, realize o download do **docker-compose.yml** correspondente à versão PHP 8 e suba os containers.

1. Acesse o diretório que contem o arquivo docker-compose.yml:

```
cd /teknisa/files/mobile
```

2. Faça um backup do arquivo atual, modificando o nome para que seja substituído pelo novo arquivo atualizado:

```
mv docker-compose.yml docker-compose.yml-bkp
```

3. Realize o download do arquivo **docker-compose-php8.zip**:

```
wget https://drive.teknisa.com/public.php/dav/files/qYqaRrFBG3iQaxC/?accept=zip --  
output-document=docker-compose-php8.zip
```

4. Descompacte o arquivo baixado:

```
unzip docker-compose-php8.zip
```

5. Suba os containers em modo background:

```
docker-compose up -d
```

 NOTA:

Com os containers ativos, o próximo passo é configurar o servidor e o deploy no portal CP, descrito no Capítulo 2. Ao final desse capítulo, retome este procedimento na Seção 2.7 para reiniciar e validar o ambiente.

Configuração de Ambiente no CP para Deploy

Este capítulo descreve os procedimentos necessários para configurar o servidor de deploy PHP 8 no portal CP, parametrizar os mnemônicos do produto e executar o deploy.

2.1 PRÉ-REQUISITOS

Antes de iniciar a configuração no CP, certifique-se de que os itens abaixo já foram concluídos no ambiente do cliente:

- Criação do ambiente onde o produto será instalado (containers Docker, Apache etc.);
- Criação do WebSocket responsável pela comunicação entre o ambiente Teknisa e o servidor (o endereço de e-mail definido durante a instalação do ambiente no servidor Linux);
- Configuração dos domínios que serão utilizados;
- Configuração das variáveis (mnemônicos) dos produtos a serem instalados, incluindo dados de banco de dados, IPs e domínios.

! ATENÇÃO:

A criação do WebSocket e o e-mail gerado durante a instalação do ambiente no servidor Linux são obrigatórios antes de prosseguir. Sem essas informações não é possível registrar o servidor no CP.

2.2 CONFIGURAÇÃO DO SERVIDOR DE DEPLOY

O servidor de deploy centraliza as informações do ambiente onde o produto será instalado e viabiliza a comunicação entre o CP e os containers do cliente.

1. Acesse a tela **Servidorx** através do caminho (Figura 1):

Parametrizações → Deploy → Servidor

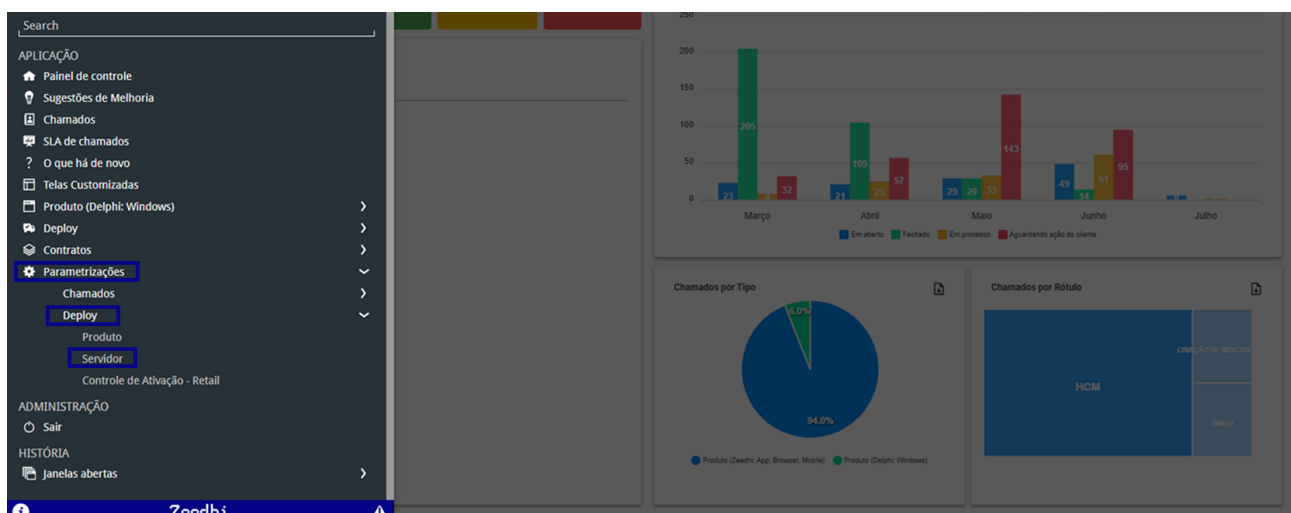


Figura 1: Caminho

2. Clique em **Adicionar** (Figura 2) para registrar um novo servidor.

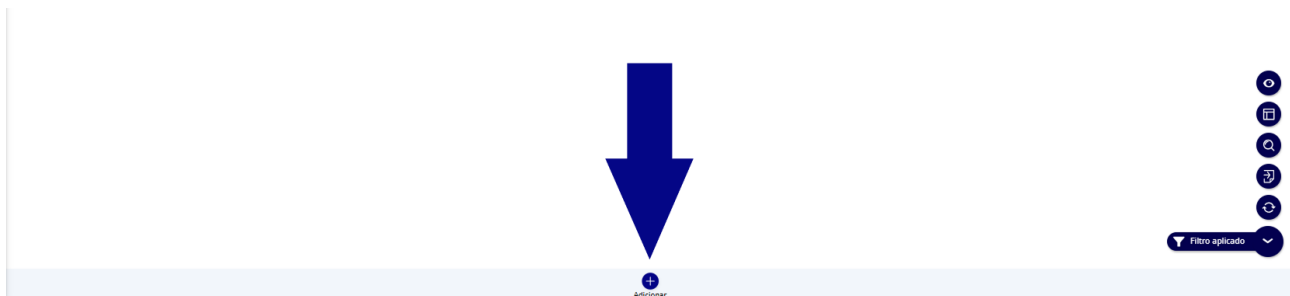


Figura 2: Adicionar

3. Preencha os campos do formulário (Figura 3):

- **Nome:** Identificação do servidor. *Exemplo: Produção PHP 8 Cliente X.*
- **Id do Servidor:** E-mail configurado no WebSocket durante a instalação dos containers. *Exemplo: nomecliente_prodphp8@teknisa.com.*
- **Limite de Backup:** Defina o valor **1** para evitar consumo excessivo de espaço em disco no servidor.

4. Clique em **Salvar** para concluir o registro do servidor.

Figura 3: Parametrização de Servidor

2.3 VINCULAÇÃO DO PRODUTO AO SERVIDOR

Após registrar o servidor, é necessário associar o produto que receberá o deploy PHP 8 a esse servidor e informar o diretório de instalação.

1. Acesse a tela **Produto** através do caminho (Figura 4):

Parametrizações → Deploy → Produto

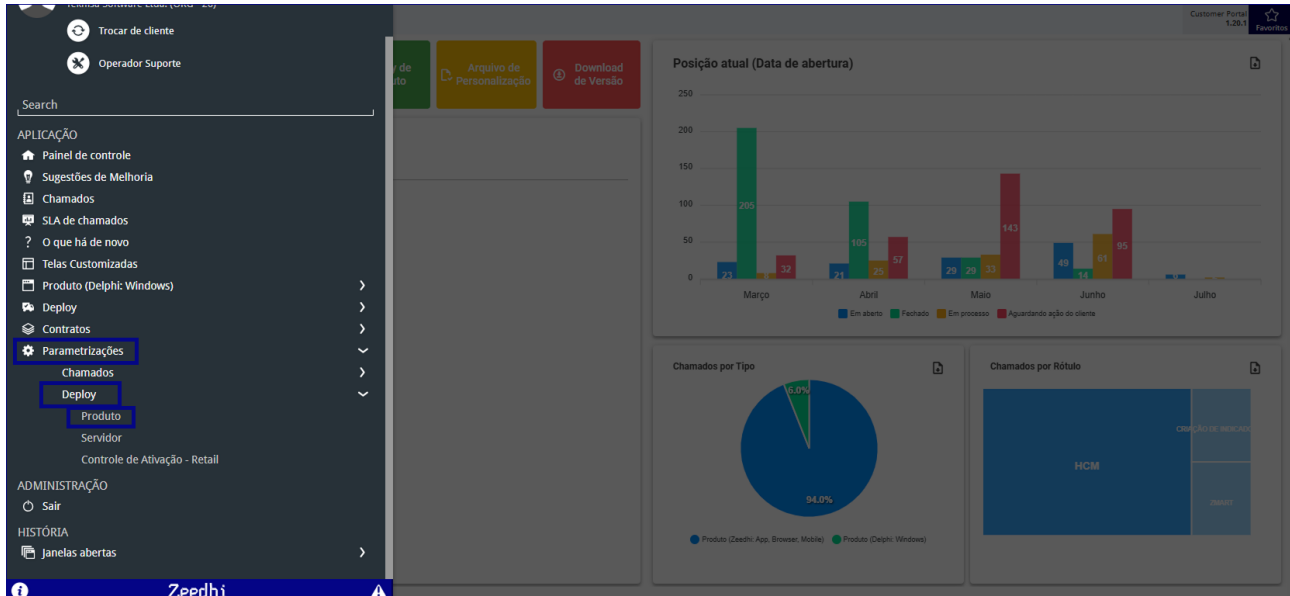


Figura 4: Caminho tela Produto

2. Selecione o produto que será atualizado para PHP 8.

3. Navegue até a aba **Servidor** (Figura 5) e clique em **Adicionar**.

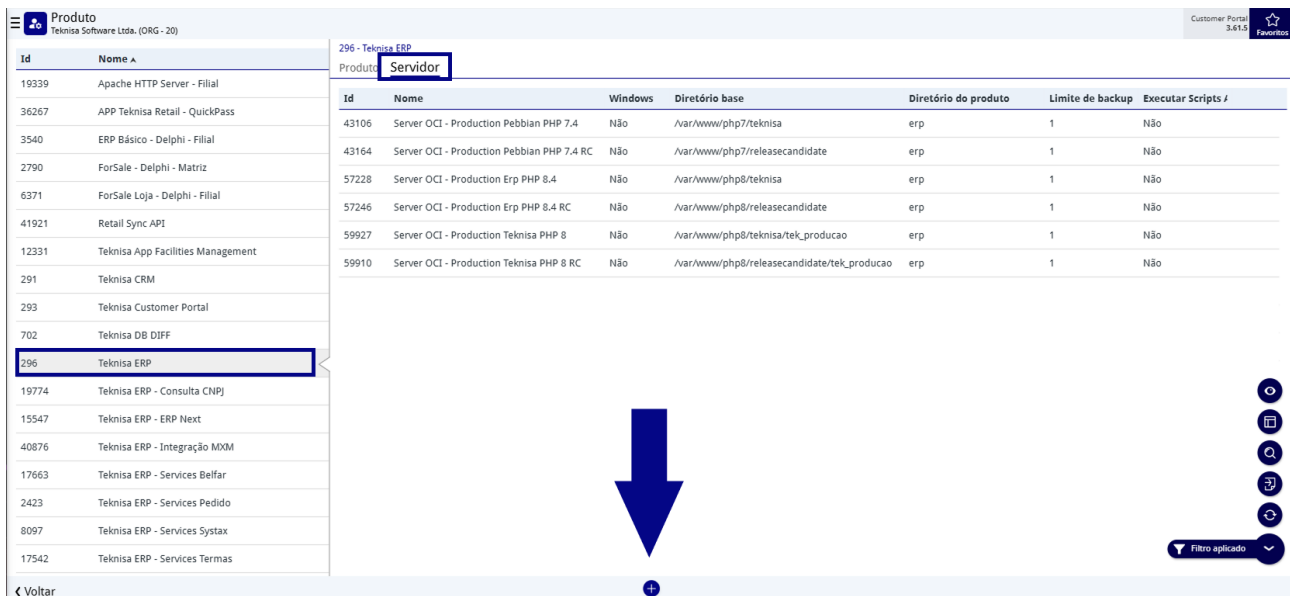


Figura 5: Tela Produto — aba Servidor, botão Adicionar

4. Preencha os campos (Figura 6) exibidos:

- **Nome:** Selecione o servidor PHP 8 criado na etapa anterior.
- **Diretório:** Informe o caminho de instalação do produto (/var/www/php8)

5. Clique em **Salvar** para registrar a vinculação.

Figura 6: Formulário de parametrização de Servidor

2.4 CONFIGURAÇÃO DOS MNEMÔNICOS

Os mnemônicos são variáveis de ambiente que definem as configurações do produto, como conexão com banco de dados, URLs de serviço, configurações de e-mail e integrações externas. Devem ser preenchidos para o servidor PHP 8 recém-criado.

1. Na tela **Servidor**, selecione o servidor PHP 8 criado.
2. Acesse a aba **Mnemônicos** (Figura 7).

Logo

Nome

3517 - Server HCM - Production Pebbian

3517 - Server HCM - Production Pebbian

Mnemônico

Id	Nome
10637	Server AWS - Production Teknisa PHP7
10636	Server AWS - Production Teknisa PHP7 RC
6016	Server AWS Teknisa - Pebbian Production PHP7
9496	Server AWS Teknisa - Pebbian Production PHP7 RC
3517	Server HOM - Production Pebbian
6177	Server HOM - Production Pebbian PHP 7.4
6256	Server HOM - Production Pebbian php7 2
17078	Server OCI - Production Erp PHP 8.4
17079	Server OCI - Production Erp PHP 8.4 RC
13059	Server OCI - Production Pebbian PHP 7.4
13077	Server OCI - Production Pebbian PHP 7.4 RC
17796	Server OCI - Production Teknisa PHP 8
17797	Server OCI - Production Teknisa PHP 8 RC
4357	Server Pebbian - Production Nsgroup AWS
9056	Server Pebbian AWS - Production Facilities Sensores
3856	Server TED(78) - Apresentacao Mercado
2316	Server TEK (107) - Pebbian Production
10656	Server TEK (107) - Pebbian Production PHP 7.4 RC

Mnemônico	Valor	Descrição	Grupo	Produto	Criptografado
DB_MONGO_HOST	hom-mongo.teknisa.com	MONGO DB HOST	ERP - Cadastros de Gestão ...	Não	
DB_MONGO_PORT	27020	MONGO DB PORT	ERP - Cadastros de Gestão ...	Não	
DB_ORA_DBNAME	USR_ORG_53	ORACLE DB SERVICE NAME	ERP - Cadastros de Gestão ...	Não	
DB_ORA_HOST	null	ORACLE DB HOST	ERP - Cadastros de Gestão ...	Não	
DB_ORA_PASSWORD	null	ORACLE DB PASSWORD	ERP - Cadastros de Gestão ...	Sim	
DB_ORA_PORT	null	ORACLE DB PORT	ERP - Cadastros de Gestão ...	Não	
DB_ORA_USER	/	ORACLE DB USER	ERP - Cadastros de Gestão ...	Não	
DB_VPD_WALLET_BOOL	true	Database VPD use WALLET identifier	ERP - Cadastros de Gestão ...	Não	
FILE_SERVER_ADMIN_API_KEY	*****	Chave do administrador do File Server	Teknisa License Manageme...	Sim	
FILESERVER_BASE_URL	null	http://midia.teknisa.com	Pebbian FM Backend,Tekni...	Não	
FILESERVER_BASE_URL_MAN	https://midia.teknisa.com/index.php	FILESERVER_BASE_URL_MAN		Não	
FILE_SERVER_CUSTOM_URL	null	URL apontando para o index.php do ...	Teknisa CRM Next,Teknisa ...	Não	
FILE_SERVER_FACTORY_METHOD	null	Factory do Fileserver	Teknisa CRM Next,Teknisa ...	Não	
FILE_SERVER_FILES_ALIAS	/files	File server files folder alias		Não	
FILE_SERVER_FILES_URL	null	FILE_SERVER_FILES_URL	TecFood Corp - MyOrder	Não	
FILE_SERVER_FILES_URL_MYO	https://myorder.teknisa.com/files/	MyOrder file server files URL	Teknisa TecFood MyOrder ...	Não	

Filtro aplicado

Figura 7: Aba Mnemônico

3. Edite e preencha os mnemônicos conforme os grupos descritos abaixo.

2.4.1 Banco de Dados BIRT

- **BIRT_ORA_DBNAME**: Nome do banco de dados.
- **BIRT_ORA_PASSWORD**: Senha do banco de dados.
- **BIRT_ORA_USER**: Usuário do banco de dados.
- **BIRT_ORA_PORT**: Porta do banco de dados.
- **BIRT_ORA_HOST**: IP do servidor de banco de dados.
- **BIRT_URL**: URL do produto.

2.4.2 Banco de Dados (Oracle)

- **DB_ORA_SESSION_MODE**: Preencher com `null`.
- **DB_ORA_DBNAME**: Nome do banco de dados.
- **DB_ORA_PORT**: Porta do banco de dados.
- **DB_ORA_HOST**: IP do servidor de banco de dados.
- **DB_ORA_PASSWORD**: Senha do banco de dados.
- **DB_ORA_USER**: Usuário do banco de dados.

2.4.3 Banco de Dados MongoDB

- **DB_MONGO_PORT**: 27081
- **DB_MONGO_HOST**: IP da máquina Linux onde o MongoDB está instalado.

2.4.4 Configurações de E-mail (SMTP)

- **SMTP_AUTH**: Preencher com `true`.
- **SMTP_NAME**: Nome do usuário.
- **SMTP_HOST**: IP ou endereço do servidor.
- **SMTP_FROM**: Endereço de e-mail.
- **SMTP_SECURE**: Tipo de segurança utilizado.
- **SMTP_PORT**: Porta do servidor.
- **SMTP_PASSWORD**: Senha da conta de e-mail.
- **SMTP_USER**: Usuário da conta de e-mail.

2.4.5 Identificação e QR Code

- **LOGIN_MAC_ID**: ID da empresa no sistema.
- **QRD_PORT**: Porta do socket configurado na máquina Windows. *211 ou 212, conforme configuração do ambiente.*
- **QRD_TYPE**: Preencher com PDF.
- **QRD_URL**: URL da máquina ou endereço configurado para o serviço de QR Code.
Exemplo: http://IP_DA_MÁQUINA/ ou http://ENDEREÇO_CONFIGURADO.

2.4.6 FileServer

- **FILE_SERVER_KEY_ACL**: Preencha com o valor f3cebebc01f9ad77a8b064d5e65af30a.
- **FILE_SERVER_URL**: URL do FileServer.
- **TEK_FILE_SERVER_BASE_URL**: URL do FileServer.
- **TEK_FILESERVER_BASE_URL**: URL do FileServer.
- **TEK_FILESERVER_FILES_URL**: URL do FileServer.
- **TEK_FILESERVER_KEY**: NRORG da tabela Empresa.
- **TEK_FILESERVER_MODE**: Preencher com Production.
- **TEK_FILESERVER_TOKEN**: Token gerado no container PHP 5 do ambiente.

ATENÇÃO:

Os mnemônicos **FILE_SERVER_URL**, **TEK_FILE_SERVER_BASE_URL**, **TEK_FILESERVER_BASE_URL** e **TEK_FILESERVER_FILES_URL** devem receber a mesma URL do FileServer. Divergências entre esses campos podem causar falha na integração.

4. Clique em **Salvar** após preencher todos os mnemônicos.

2.5 DEPLOY DO PRODUTO

Com o servidor configurado e os mnemônicos preenchidos, execute o deploy do produto.

1. Acesse a tela **Deploy do Produto** (Figura 8) e selecione o produto que receberá a atualização.

Deploy → Deploy do Produto

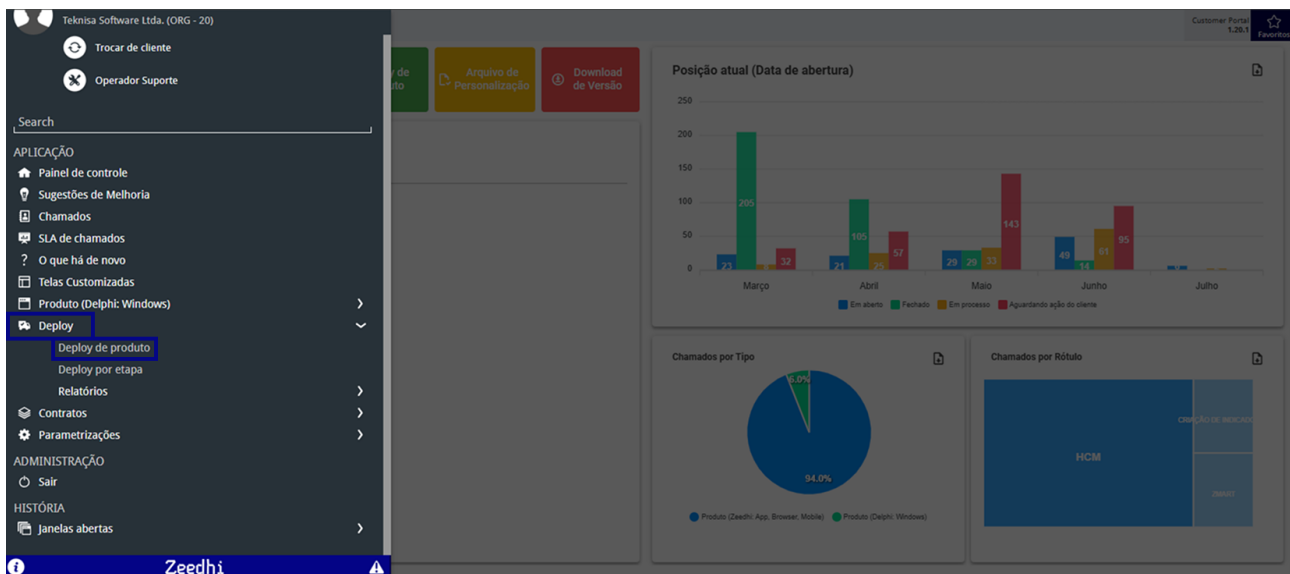


Figura 8: Caminho Deploy de Produto 1

2. Selecione a **versão** (Figura 9) desejada.
3. Selecione **Servidor** (Figura 9).
4. Clique em **Aplicar Deploy** (Figura 9).

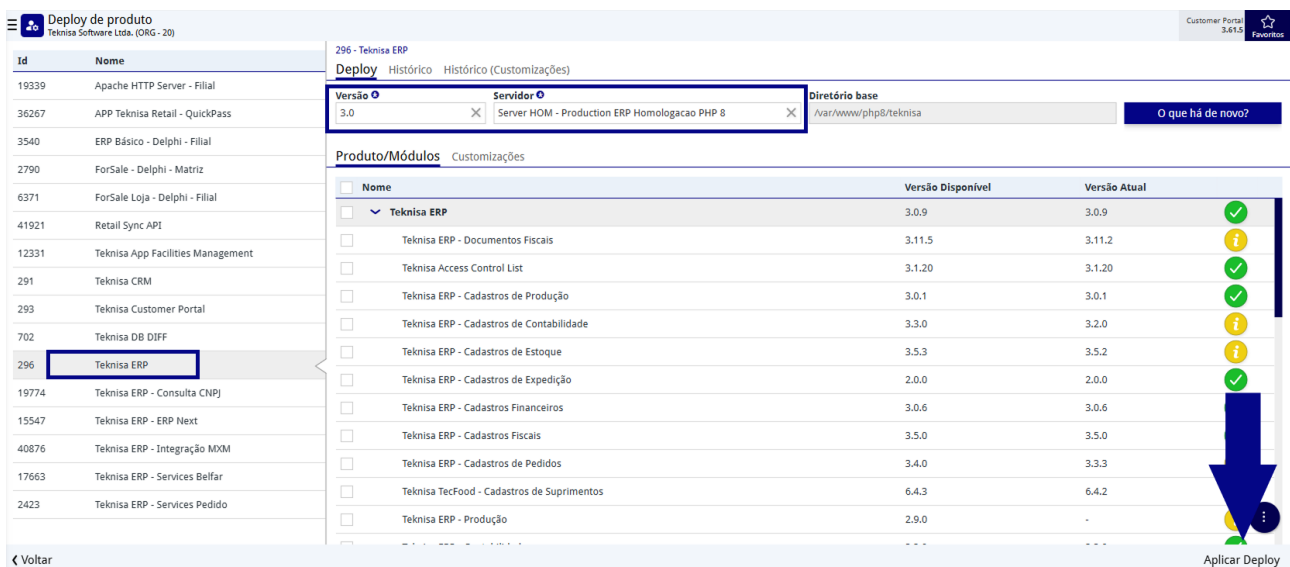


Figura 9: Configuração e aplicação de deploy

NOTA:

Caso o sistema exiba uma mensagem informando que foi gerado um script de banco de dados, faça o download do arquivo e aplique-o no banco de dados antes de prosseguir com a validação do ambiente.

2.6 PARAMETRIZAÇÃO ZEEDHI NEXT (TECFOOD E ERP)

Para os produtos **TecFood Next** e **ERP Next**, é obrigatório realizar a parametrização do Zeedhi Next no portal DevOps antes de aplicar o deploy. Sem essa configuração, o deploy dos produtos Next não será executado.

1. Acesse o portal DevOps pelo endereço <https://devops.teknisa.com/>.
2. Navegue até a tela **Product** (Figura 10).

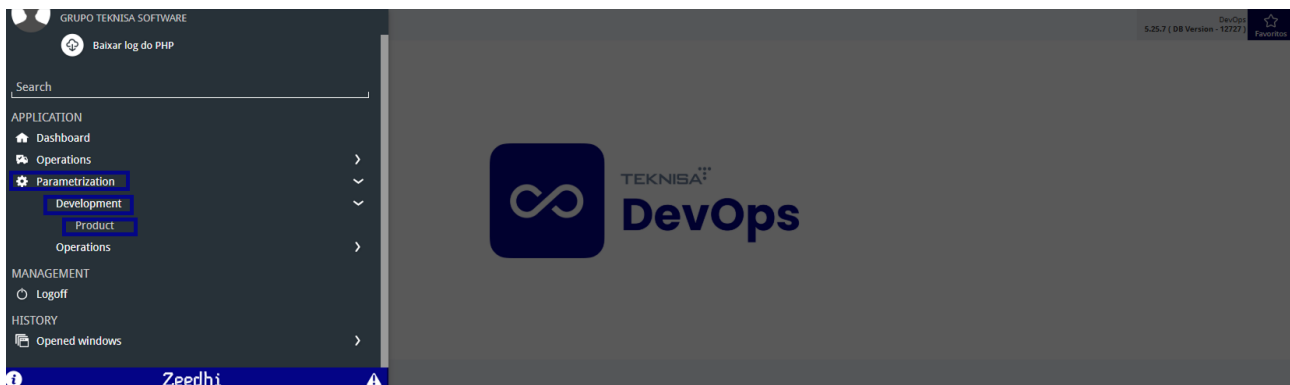


Figura 10: Caminho Product

3. Pesquise pelo nome do produto: **ERP Next** e/ou **TecFood Next** no filtro da tela (Figura 11).

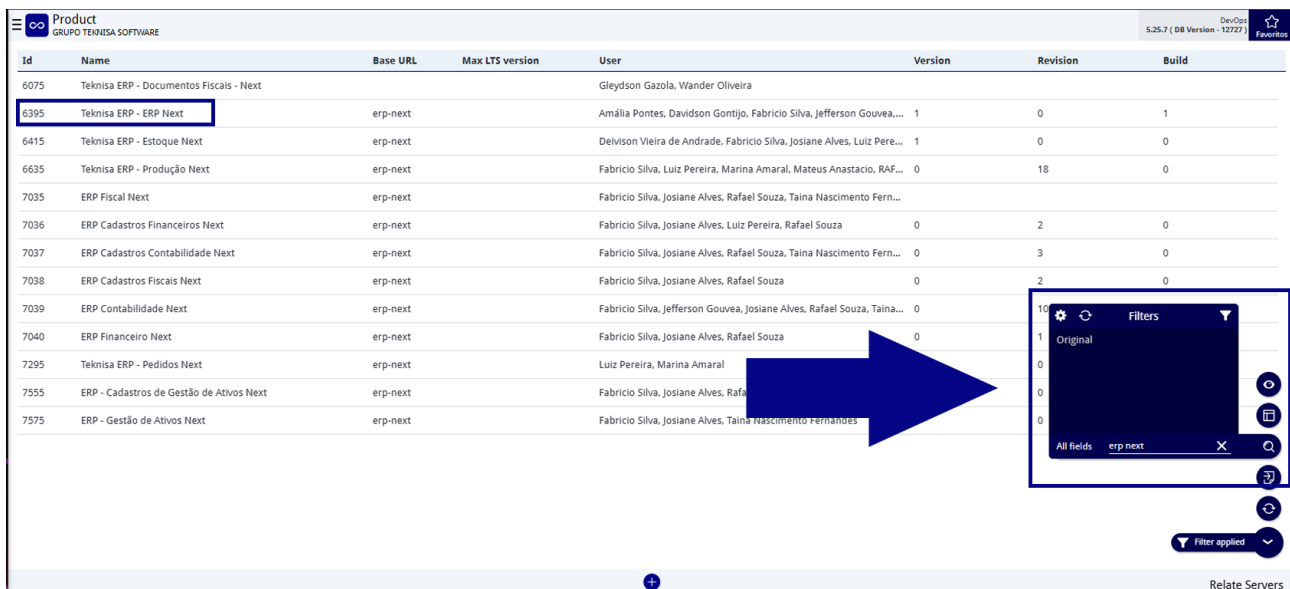


Figura 11: Filtragem de produto

- Selecione o produto e acesse a aba **Server**.
- Clique em **Adicionar** (Figura 12) no rodapé central da tela.

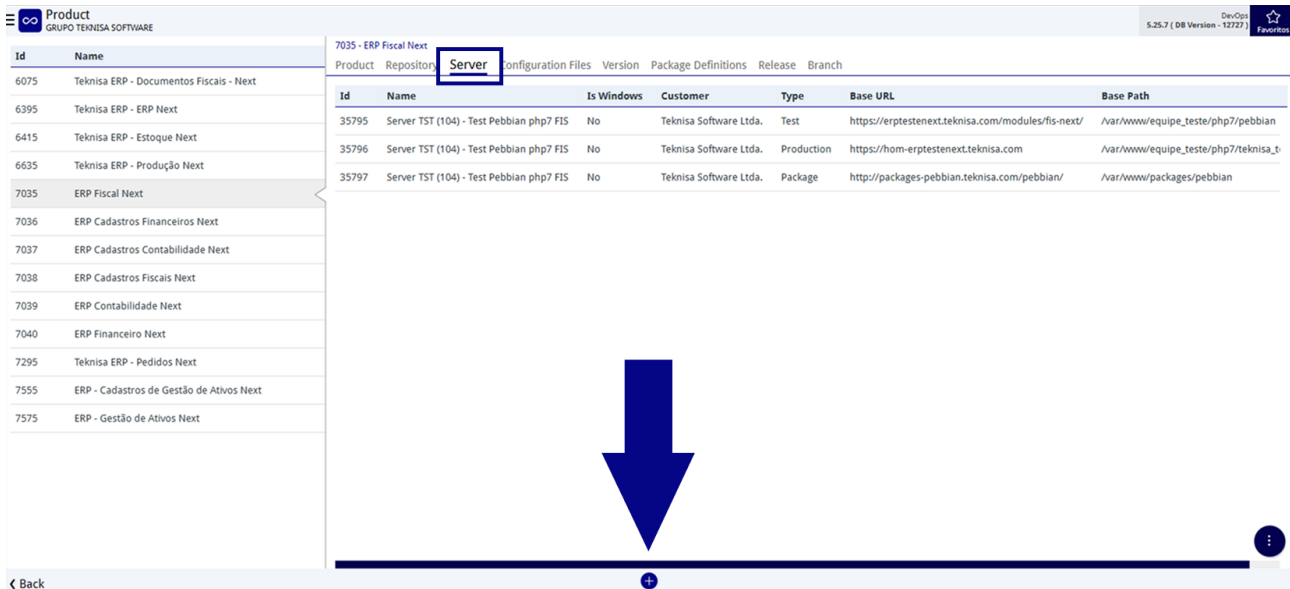


Figura 12: Server Adicionar 1

- Preencha os campos **Name**, **Type** e **Base Path** (Figura 13) com os valores correspondentes ao ambiente do cliente.

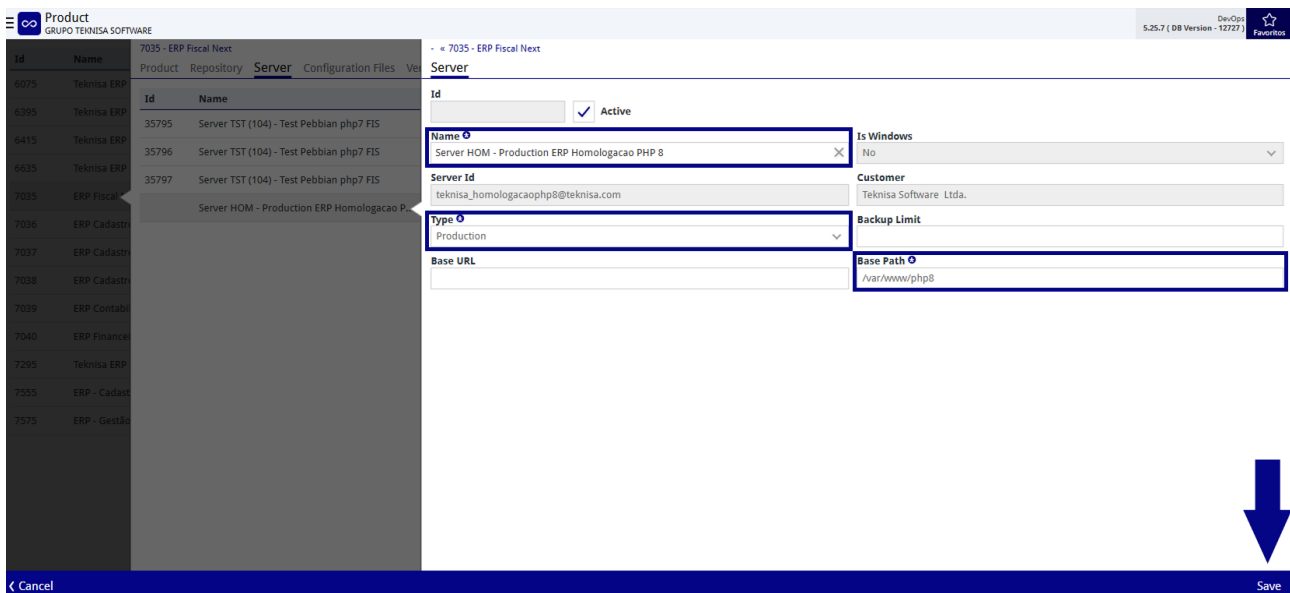


Figura 13: Parametrização Server

- Acesse o portal CP pelo endereço <https://cp.teknisa.com/> e verifique se o mnemônico **NEXT_URL** (Figura 14) está preenchido com a URL criada para o produto Next.

17136 - Server HOM - Production ERP Homologacao PHP 8

Servidor

Mnemônico

17136

Server HOM - Production ERP Homologacao PHP 8

17078

Server OCI - Production Erp PHP 8.4

17079

Server OCI - Production Erp PHP 8.4 RC

17639

Server TST (104) - Test Erp PHP 8 - POSTGRES

Mnemônico	Valor	Descrição	Grupo	Produto	Criptografado
ERP_NEXT_URL	https://hom-erp-next.teknisa.com	Url para o erp-next		ERP - Cadastros de Gestão ...	Não
PME_NEXT_URL	https://hom-pme.teknisa.com	URL do produto Planejamento de Ali...		Teknisa TecFood Planejam...	Não

Filtros

Original

Todos os ... next_url

Filtro aplicado

Voltar

Figura 14: Conferência de mnemônico

8. Aplique o deploy do produto conforme descrito na Seção 2.5.

⚠️ ATENÇÃO:

Caso seja exibida uma mensagem informando a geração do script de banco, realize o download do arquivo e execute-o no banco de dados.

2.7 FINALIZAÇÃO: REINICIALIZAÇÃO E VALIDAÇÃO DO AMBIENTE

Após concluir o deploy do produto (Seção 2.5) — e a parametrização do Zeedhi Next —, retorne ao servidor Linux para reiniciar o serviço Docker e validar o ambiente.

1. Reinicie o serviço Docker com o comando:

```
service docker restart
```

2. Verifique o status dos containers com o comando abaixo e confirme que todos estão com o status **Up**:

```
docker ps
```

3. Com os containers ativos, realize o acesso ao sistema para validar o ambiente.