**Prostředí pro web MZK.cz**

v 2023.09.04

**Tohle je návrh technického provozu webu na Drupalu v prostředí MZK.**

Vychází z prostředí MZK a best-practices, které se MZK zatím osvědčily. Návrhy na

vylepšení / zjednodušení vítáme.

**Drupal v MZK**

Finální stav je nasazení produkčního webu www.mzk.cz na infrastrukturu dostupnou v MZK,

primárně kontejnery v prostředí Kubernetes clusteru MZK. Nasazení nového webu by se s

výjimkou technicky odůvodněných rozdílů nemělo lišit od nasazení v Docker/Podman

kontejnerech na počítačích u vývojářů.

Nový web/weby poběží na Drupalu. Předpokládá, že finální kontejner s Drupalem bude kvůli

rozložení zátěže provozovat ve více instancích kontejnerů vedle sebe. Drupal by měl

zvládnout obsluhovat požadavky z více nezávislých instancí PHP.

Jako databázi předpokládá MZK MariaDB nebo PostgreSQL, jako úložiště pro sessions

Redis nebo KeyDB. Jako úložiště cache pak opět Redis/KeyDB. Pro ukládání souborů je

možné použít sdílený diskových svazek nebo Minio (S3-like storage).

Pro zvýšení výkonu je možné nasadit Varnish jako předřazenou vyrovnávací keš.

Podle zkušeností MZK by toto nasazení Drupalu mělo být přiměřeně jednoduché pro

nasazení a nemělo by se příliš lišitt od nasazení na klasickém LAMP hostingu. Zároveň

poskytuje možnost škálování výkonu v případě potřeby.

**Uživatelské testování před nasazením**

Poběží tři kopie celého stacku (web, db, cache), pravděpodobně v sestavě:

● **produkční web** (prod, https://www.mzk.cz/) - sem chodí uživatelé z venku knihovny

a na tomto webu provádí správce obsahu MZK změny - vkládání článků, obrázků,

atd.

● **ověřovací web** (staging) - na tomto webu je možné otestovat změny, které se mají

posléze provést na produkčním webu na první pokus.

● **vývojový web** (devel) - tady se provádí vývoj. Je možné připravit více kopií

vývojového webu.

Pro každou instanci Drupalu bude připravena samostatná databáze, Redis a storage, čímž

se zjednoduší testování při nasazování nových verzí a updatů.

Pro vývojáře webu připraví wftech i nástroje pro přístup k databázím, úložištím i keši.

Poskytne i nástroj pro jednoduché okopírování dat z produkčního webu na testovací.

**Vývoj**

Zdrojové kódy budou uložené v privátním git repozitáři v Gitlabu, který MZK provozuje na

https://gitlab.mzk.cz/ . Ze zdrojových kódů se budou pomocí Gitlab CI/CD pipeline

sestavovat kontejnery, které se budou po schválení vývojářem nasazovat do clusteru.

**Předpokládaný postup vývoje**

● na stroji u vývojáře se připraví nová verze aplikace

● po odeslání revize do Gitlabu se automaticky sestaví obraz kontejner s novou verzí

aplikace

● obraz se připraví k nasazení a kliknutím (nebo automaticky) se nasadí do vývojového

prostředí

● vývojář otestuje změny na vývojovém prostředí.

● pokud je potřeba, upraví změny v databázi, apod.

● opakovat od bodu 1

Pokud se vývojové prostředí rozbije, budou k dispozici nástroje, které obnoví databázi ze

zálohy, vyprázdní cache, vyprázdní objekt storage, apod.

wftechzřídí přístup k testovacímu prostředí, tak aby se dalo testovat pohodlně.

Pro testování bude možné obnovit testovací databázi ze zálohy produkčního webu. .

V případě, že tento předpokládaný postup vývoje zhotovitel vyhodnotí jako nevhodný, je povinen na toto upozornit MZK ve smyslu smlouvy o dílo)

Ověrovací prostředí (staging)

Pokud bude aplikace v testovacím prostředí v pořádku, bude možné ji nasadit do ověřovacího prostředí.

Ověřovací prostředí by mělo být nastaveno stejně jako produkční prostředí, jen s menšími vyhrazenými prostředky (CPU, RAM) a s jinými URL.

Pro ověření upgrade z předchozí verze aplikace je k dispozici nástroj, který do databází v ověřovacím prostředí nasadí data z produkčního prostředí.

Nasazení verze aplikace do testovacího prostředí dělá robot na vyžádání.

Poté je možné provést testy na ověřovacím prostředí. Pokud je vše v pořádku, aplikace je pravděpodobně funčkní a připravena k nasazení do produkčního prostředí.

Pokud aplikace nebude fungovat, je možné aplikaci v ověřovacím prostředí zastavit nebo se vrátit se k předchozí verzi a ověřovat znovu.

Produkční prostředí

Po ověření aplikace v ověřovacím prostředí je možné aplikaci nasadit do produkčního prostředí stejně jako v pro

**Co zajistí wf tech**

● Vzdálený repozitář na Gitlabu MZK

● Přístup k vývojovému prostředí

● Uživateleké účty a přístup k repozitářům v Gitlabu

● Sestavení výchozího obrazu kontejneru pro Drupal

● Nainstalovaný provozní cluster - společný pro produkci, test i devel

● Oddělené prostředí pro staging a devel, aby se vývojáři při testech nemuseli

bát, že přetíží produkční web.

● Oddělené MariaDB pro každé prostředí (prod, staging, devel)

● Oddělené Redis / KeyDB pro každé prostředí (prod, staging, devel)

● Minio pro každé prostředí (prod, staging, devel)

● kešující CDN pro Minio (bude-li třeba)

● nebo sdílený diskový svazek pro data

● Nástroje pro přístup k databázím (webový adminer, nebo port-forward databáze)

● Přístup ke Redisu a Minio ve všech prostředích

● Prostředí pro sledování logů z kontejnerů - Grafana, Loki

● Prostředí pro sledování zatížení kontejnerů - Grafana, Prometheus

● Prostředí pro sledování stavu nasazení + připojení do terminálu kontejnerů - ArgoCD

● pravidelné zálohy databází a objektového úložiště

● možnost zálohy databází na vyžádání přes webové UI

● možnost obnovy databáze ze zálohy z přes webové UI

**Co se bude upravovat**

Dle odborného posouzení dodavatele na základě písemné výzvy dodavatele wftech připraví

* CI/CD pipelines pro automatické sestavení kontejnerů ze zdrojových kódů
* pokud budou ve zdrojových kódech připraveny jednoduché testy, tak i testování

zdrojových kódů

**Povinnosti dodavatele webu**

* Připravit zdrojové kódy webu tak, aby bylo možné je sestavit do kontejneru
* Připravit zdrojové kódy webu tak, aby bylo možné je nasadit do Kubernetes clusteru

MZK. Zajistíme podporu.

* Změny ve schématu databáze dělat ideálně pomocí Drupal Migrations či podobné

technologie, aby se mohlo automaticky nasazovat i verze se změnou databázového

schématu.

* využití Redis keše pro urychlení webu, pokud to bude možné (dodavatel je povinen

MZK sdělit písemně v průběhu provádění díla, zda je toto možné, pokud nesdělí, má

se za to, že je to možné)

* připravené kontejnery by měly poskytovat healthcheck URL, např. /healthcheck pro

zjištění stavu kontejneru při přepnutí provozu

**Plánované technické řešení**

● PHP8+, fpm worker

● webserver Apache nebo Nginx (dle dohody)

● MariaDB nebo PostgreSQL (dle dohody)

● KeyDB jako storage

● Minio nebo sdílený diskový svazek pro ukládání souborů

● volitelně: Elasticsearch pro fulltext search

● volitelně: varnish jako kešující load balancer

● pro nasazení Kubernetes Deployments + Services

● stavba kontejner image - Gitlab CI, kaniko + docker build

● nasazení do clusteru - Argo CD + gitops

● přístup do kontejnerů - Argo CD Terminal

● monitoring výkonu - Grafana, Prometheus

● sběr logů a vizualzace - Loki, Grafana

● živé logy Argo CD

● automatizace obnovy a záloh - Argo Workflows + vlastní skripty

● přístup k databázím - Adminer