

Framework

Framework (v překladu *aplikační rámeček*) je standardizovaný soubor předpřipraveného kódu, knihoven, nástrojů a nejlepších praktik, který slouží jako základ a opěrná kostra pro vývoj softwaru. Namísto toho, aby programátor psal celou aplikaci od naprosté nuly, poskytuje mu framework hotovou architekturu, do které pouze doplňuje specifickou logiku své aplikace.

Framework si lze představit jako **hrubou stavbu domu**. Máte hotové základy, nosné zdi, rozvody elektřiny a potrubí. Nemusíte pálit vlastní cihly ani vymýšlet, jak funguje gravitace. Vaším úkolem je pouze dům zařídit, vymalovat a určit, jaký bude mít účel (zda to bude nemocnice, nebo rodinný dům).

Framework vs. Knihovna (Library)

Nejčastějším omylem v IT je zaměňování pojmů framework a knihovna. Klíčovým rozdílem je technický princip zvaný **Inverze řízení (Inversion of Control - IoC)**:

- **Knihovna (Library):** Vy máte plnou kontrolu nad během programu. Váš kód určuje strukturu a v případě potřeby si „zavolá“ knihovnu, aby pro něj udělala nějakou specifickou práci (např. knihovna pro zobrazení kalendáře nebo převod měn).
- **Framework:** Framework má kontrolu nad během programu. On určuje celkovou strukturu aplikace, její životní cyklus a bezpečnost. Framework „volá“ váš kód ve chvíli, kdy je to potřeba. Vaše aplikace žije uvnitř frameworku, nikoliv naopak.

Výhody a nevýhody používání frameworků

Výhody

- **Rychlost vývoje:** Zabraňuje „znovuvynalézání kola“. Rutinní úkoly (připojení k databázi, zabezpečení hesel, routování URL) jsou již vyřešeny.
- **Bezpečnost:** Populární frameworky jsou testovány tisíci vývojáři po celém světě, což minimalizuje riziko běžných bezpečnostních děr (jako je SQL Injection nebo XSS).
- **Standardizace:** Pokud do firmy přijde nový vývojář, který zná daný framework, může okamžitě začít pracovat, protože struktura složek a kódu je všude stejná.

Nevýhody

- **Křivka učení:** Naučit se pracovat s komplexním frameworkem (jakým je např. Angular nebo Spring) může trvat měsíce.
- **Omezení (Opinionated design):** Framework vám často striktně diktuje, jak máte věci dělat. Pokud chcete udělat něco nestandardního, můžete narazit na tvrdý odpor systému.
- **Zátěž (Overhead):** Pro extrémně jednoduchou webovou stránku je použití velkého frameworku jako jít na komára s kanónem – aplikace bude zbytečně velká a pomalá.

10 populárních frameworků a příklady jejich použití

Svět IT nabízí stovky frameworků. Následující tabulka představuje 10 nejvýznamnějších rozdělených podle zaměření, včetně konkrétního příkladu, kdy po nich sáhnout.

Název frameworku	Jazyk / Ekosystém	Zaměření	Typický příklad použití (Kdy jej zvolit)
1. Django	Python	Backend Web	Tvorba robustního e-shopu, zpravodajského portálu nebo interního informačního systému s důrazem na extrémní bezpečnost a rychlost nasazení.
2. Laravel	PHP	Backend Web	Vývoj moderní webové aplikace (např. rezervační systém nebo CRM), kde je kladen důraz na elegantní kód a snadné napojení na e-mail a databáze.
3. Spring Boot	Java	Backend / Enterprise	Tvorba vysoce zabezpečených a škálovatelných mikroslužeb pro bankovníctví, pojišťovny nebo obří nadnárodní korporace.
4. ASP.NET Core	C# (.NET)	Backend Web / API	Výkonné backendové API pro komunikaci s velkými podnikovými databázemi (SQL Server), často využívané v podnikovém prostředí Microsoftu.
5. Ruby on Rails	Ruby	Backend Web	Rychlé prototypování startupových projektů a platform (např. původní verze Airbnb nebo Twitteru). Filozofie „konvence má přednost před konfigurací“.
6. Angular	TypeScript	Frontend (SPA)	Komplexní „Single Page Applications“ (SPA) jako je webový e-mailový klient, internetové bankovníctví nebo složitý firemní dashboard.
7. Vue.js	JavaScript	Frontend (UI)	Tvorba interaktivního uživatelského rozhraní. Ideální pro postupné nasazování do již existujících aplikací, které potřebují modernizovat vzhled a dynamiku.
8. Flutter	Dart	Mobilní (Cross-platform)	Vývoj mobilní aplikace pro iOS i Android současně z jednoho jediného zdrojového kódu (např. e-shop aplikace, fitness tracker).
9. React Native	JavaScript	Mobilní (Cross-platform)	Tvorba nativně vypadající mobilní aplikace pro existující webovou službu s využitím znalostí JavaScriptu a týmu webových vývojářů (např. aplikace Facebook, Instagram).
10. Bootstrap	CSS / HTML / JS	Frontend (Styling)	Rychlé nastavení webu nebo administrátorského rozhraní tak, aby bylo responzivní (dokonale se přizpůsobilo na monitory, tablety i displeje mobilních telefonů).

Poznámka pro hnidopichy: Některé nástroje (jako např. React nebo knihovny v ekosystému JavaScriptu) balancují na tenké hranici mezi knihovnou a frameworkem. V širším IT diskurzu se však často označují jako frameworky díky robustnímu ekosystému, který je obklopuje.

From:

<http://www.serviceit.cz/> - **IT ENCYKLOPEDIA**

Permanent link:

<http://www.serviceit.cz/doku.php?id=framework>

Last update: **2026/06/06 11:39**

