

Integrace DotVVM a .NET MAUI

Tomáš Mikeš*

Abstrakt

Cílem této práce je zajištění integrace technologií DotVVM a .NET MAUI, která má umožnit jejich vzájemnou komunikaci a možnost vykreslení webových stránek vyvinutých v DotVVM uvnitř cross-platformní aplikace .NET MAUI. Součástí je popis implementovaného řešení probíhajícího pomocí komponenty WebView a také vzorové aplikace demonstrující hybridní způsob vývoje. Implementační část zahrnuje analýzu způsobů vývoje mobilních aplikací, aplikačního rámce DotVVM a .NET MAUI. Výsledky této práce umožňují aplikacím vyvinutým v DotVVM fungovat uvnitř cross-platformní aplikace a přistupovat tak k nativním funkcionalitám daných platform

*xmikes15@vutbr.cz, Faculty of Information Technology, Brno University of Technology

1. Úvod

Představme si, že vyvíjíme software pro zákazníka, který si dle analýzy žádá rozsáhlou desktopovou aplikaci skládající se z komplexních grafických komponent. Tyto komponenty vytvoříme pomocí knihovny určeného pro vývoj uživatelského rozhraní, jenž je specifický pro vývoj desktopových aplikací. Později se však zákazník rozhodne, že by potřeboval zmíněné komponenty využít i ve své webové aplikaci a my jsme nuceni všechny komponenty uživatelského rozhraní vytvořit znovu pomocí webové technologie. Takový vývoj stojí spoustu času, režie a také financí.

Zde přichází pojem hybridní aplikace, jejímž základem jsou webové technologie (HTML, CSS a JavaScript). Taková aplikace je definována jako cross-platformní, jelikož sdílí stejný zdrojový kód a může fungovat na více platformách [1]. Kompatibilitu s více operačními systémy zajišťuje nasazení aplikace do nativního kontejneru využívající komponenty WebView, která umožňuje zobrazovat webové stránky a oproti prohlížeči neobsahuje žádnou z funkcí, jako je navigační menu či vstupní pole s URL. V porovnání s progresivními webovými aplikacemi (PWA) dokáže využít i nativních funkcionalit, které webový prohlížeč nenabízí. Může mezi ně patřit například připojení k bluetooth, snímače přiblížení nebo také využití biometrické autentizace, jako je Touch ID, nebo Face ID u iOS.

Tato práce se zabývá poskytnutím rozšíření DotVVM takovým způsobem, aby mohla webová aplikace vyv-

inutá v DotVVM fungovat cross-platformně. K tomu může být využit cross-platformní framework, v němž bude webová aplikace hostována hybridním způsobem. Toho lze docílit vlastní modifikací komponenty WebView, která je typicky součástí cross-platformních frameworků, a následnou tvorbou komunikačního kanálu mezi webovým frameworkem, cross-platformním frameworkem a WebView. Aplikační rámec .NET MAUI se jeví jako vhodný kandidát pro integraci s DotVVM zejména kvůli platformě .NET, na které jsou oba frameworky postaveny, a také použitím stejného návrhového vzoru MVVM.

Principiálně blízkým řešením je implementace integrace webového aplikačního rámce Blazor a .NET MAUI společností Microsoft, která byla v rámci analýzy prozkoumána a vychází z ní značná část navrženého řešení.

2. Implementace

DotVVM je framework uživatelského rozhraní, který umožňuje tvorbu webových aplikací postavených na technologii ASP.NET pomocí návrhového vzoru MVVM [2]. Přenos dat mezi serverem a klientem (prohlížečem) je zajištěn bez nutnosti tvorby a údržby REST API či mapování klientských a serverových dat. Díky tomu je možné vytvářet komplexní webové aplikace s použitím čistě jazyka C# a DotHTML, jehož syntax je pouze rozšířením běžného HTML.

.NET MAUI je nově vzniklá technologie sloužící ke

cross-platformnímu vývoji aplikací a zároveň je nástupcem velmi známého využívaného frameworku Xamarin Forms¹. Ten je rozšířen o vývoj desktopových aplikací a vyšší výkon a rozšířitelnost UI komponent (tzv. *controls*), která je v této práci využita k modifikaci komponenty *WebView*. Jelikož jsou *DotVVM* i *.NET MAUI* založeny na platformě *.NET*, běží ve stejném procesu.

Hlavní součástí implementovaného řešení, popsaného diagramem **Figure 1**, je komponenta *WebView*, která slouží k vykreslování webového obsahu a figuruje jako prostředník komunikace mezi *.NET MAUI* a klientské části *DotVVM*. Její specifická implementace pro každou z platform je schopna odchyťvat požadavky tak, aby namísto odeslání požadavku na daný vzdálený webový server určený URL, zpracovala požadavky lokálně. Zpracování zajišťuje služba nazvaná v diagramu jako *DotVVM Middleware*.

Každá implementace *WebView* na platformě nějakým způsobem umožňuje spouštět skripty v jazyce JavaScript, odesílat do něj zprávy ve formátu JSON a také je přijímat. Těchto možností je využito pro komunikaci s klientskou částí *DotVVM*, konkrétně pomocí implementovaného rozhraní *Messaging*. Přeposílané zprávy jsou využity ze směru *.NET MAUI* například k získání nebo modifikaci aktuálního stavu *DotVVM* a z opačného směru ke zpracování HTTP požadavků metody POST nebo k volání metod na straně *.NET MAUI*.

HTTP požadavky metody GET jsou odchyťvány komponentou *WebView* napřímo, nicméně to není možné u metod POST na platformě Android. Proto bylo nutné tento typ požadavků zahrnout do komunikačního rozhraní a odesílat jej tak jako obsah v komunikační zprávě.

Pro otestování funkčnosti řešení byla naimplementovaná responzivní vzorová aplikace (**Figure 2**) pomocí jazyků *DotHTML*, *TypeScript*, *SCSS* a *C#*. Umožňuje zobrazení seznamu podcastů, z nichž je možné přejít na stránku detailu podcastu s dalšími informacemi a seznamem přehrávatelných epizod. Součástí uživatelského rozhraní byla vytvořena komponenta reprezentující přehrávač audia, který umožňuje zapnutí, pozastavení, změnu hlasitosti a rychlosti právě přehrávané epizody. Epizody i podcasty je možné uložit do seznamu oblíbených a zobrazit si je na separátní stránce. Rovněž je umožněno získat reálná dynamická data z několika vybraných zdrojů RSS. Vzorová aplikace byla otestovaná na platformách

Windows i Android, kde byla rovněž prokázána použitelnost pro vývojáře webových aplikací v aplikačním rámci *DotVVM*.

3. Závěr

Cílem této práce bylo zajistit integraci technologií *DotVVM* a *.NET MAUI* tak, aby spolu dokázaly vzájemně komunikovat, a aby mohla být webová aplikace vyvinutá v *DotVVM* vykreslena pomocí komponenty *WebView* v cross-platformní aplikaci *.NET MAUI*. Předcházela tomu analýza existujících přístupů tvorby mobilních aplikací a interní fungování aplikačního rámce *DotVVM*.

Jelikož jsou veškeré zdrojové kódy obou integrovaných aplikačních rámců open-source pod licencí Apache License 2.0 a MIT, je projekt s řešením a vzorovou aplikací rovněž zveřejněn na platformě *GitHub*².

Součástí této práce nebylo provedení výkonostního měření běhu aplikace na jednotlivých platformách, což se jeví jako vhodné rozšíření této práce. Další prací do budoucna je řádné otestování výsledné aplikace na platformě *iOS*, jelikož se zde vyskytla komplikace při vykreslování obsahu a nebylo tedy možné ověřit případné nedostatky v uživatelském rozhraní aplikace na této platformě.

Poděkování

Poděkování patří vedoucímu mé práce Ing. Jiřímu Hynkovi, PhD. za hodnotnou zpětnou vazbu v průběhu zpracování této práce a Bc. Tomáši Hercegovi za odborné konzultace, zejména u rozšiřování aplikačního rámce *DotVVM*.

Literatura

- [1] Kotlin. What is cross-platform mobile development? online, 2022.
- [2] RIGANTI s.r.o. Introduction — *dotvvm* documentation. online, 2022.

¹<https://dotnet.microsoft.com/en-us/apps/xamarin/xamarin-forms>

²<https://github.com/tomasmikes/dotvvm-maui-podcast>