Produktová dokumentace se přesouvá do modernější a rozšířené podoby na docs2.feedyou.ai. Stávající již nebude rozšiřována a časem bude nahrazena.

Jak nasadit WebChat na web

Možnosti nasazení

Je několik možností, jak WebChat do webu nasadit, které se liší podle toho, jak moc je potřeba jeho vzhled nebo chování přizpůsobovat.

a) Rozbalovací chat

Nejrychlejší možnost je jen vložit před koncovou značku `</body>` jednoduchý kód, který dostanete od tvůrce virtuálního asistenta (VA). Bude vypadat podobně jako v ukázce no-config.html, jen v něm bude vyplněné ID bota. Pro nasazení je samozřejmě možné využít i služby jako např. Google Tag Manager pro jeho vložení bez nutnosti ručního zásahu do kódu.

Design a chování webchatu je pak možné nastavit ze strany Feedyou (nebo v nástroji Designer pokud máte přístup). Všechny domény, kde VA poběží je nutné ručně povolit ze strany Feedyou. Pokud je VA v testovacím režimu, nebude na webu zobrazen, pokud se na konec URL nepřidá `#feedbot-test-mode`.

b) Chat do vlastního elementu

Pokud je žádoucí upravit vzhled nebo chování chatu, je možné ho vložit do vlastního HTML elementu, který je možné nastylovat po svém. Daný DOM element vložte jako druhý parametr funkce `BotChat.App`. Chat se pak vyrenderuje do něj. Element nemusí být nutně vyskakovací okno, klidně může jít přímo o součást stránky.

> Pokud je chat při načtení stránky skrytý a zobrazuje se až na základě akce uživatele, je důležité, aby se `BotChat.App` volalo až při jeho zobrazení a předešlo se tím zbytečným voláním VA.

Ukázky možného použití jsou k dispozici na GitHubu, a to např.:

- samples/feedyou/expandable.html - vyskakovací okno ve vlastním elementu
- samples/feedyou/embed.html - bot vložený do konkrétního elementu stránky
- samples/feedyou/persistent.html - speciální varianta, která si ukládá do prohlížeče ID uživatele a konverzace, aby na ni mohla navázat při příští návštěvě stránky

c) Pokročilé způsoby nasazení

WebChat je otevřená knihovna postavená na Reactu, takže je možné ji snadno zapojit do stávající React aplikace nebo jakýmkoli způsobem upravovat její chování. Pro více informací prosím kontaktujte Feedyou.

d) Vyzkoušení na libovolné stránce

Pokud chcete jen vyzkoušet, jak bude VA na libovolné stránce vypadat, je možné do následujícího kódu nastavit odpovídající údaje a celý ho vložit do konzole ve webovém inspektoru (většinou stačí pravý klik kamkoli na stránku a pak *Inspect* nebo *Prozkoumat*). Tento kód NEPOUŽÍVEJTE v produkčním prostředí, k tomu jsou určeny možnosti a) - c). Pokud je bot stále v testovacím režimu, je nutné nejprve na konec URL dané stránky přidat `#feedbot-test-mode`

```
(function (f,y,b,o,t,s){(t = y.createElement(b)), (s = y.getElementsByTagName(b)[0]);t.async = 1;t.src = o;t.onload=function(){
BotChat.App({
  bot: { id: 'feedbot-...' }, // najdeme na konci "Code" odkazu v "Channels settings" sekci Designeru
  channel: { id: '...' }, // je 6ti pismenny kod ktery najdeme na konci "Preview" odkazu u daneho WebChatu v "Channels settings" sekci Designeru
})
};s.parentNode.insertBefore(t, s);
})(window,document,"script","https://feedyou.azureedge.net/webchat/latest/botchat-es5.js");
(function (f,y,b,o,t,s){(t = y.createElement(b)), (s = y.getElementsByTagName(b)[0]);t.rel = 'stylesheet';t.type = 'text/css';t.href = o;s.parentNode.insertBefore(t, s);
})(window,document,"link","https://feedyou.azureedge.net/webchat/latest/botchat.css");
```

Propojení s botem

Nehledě na to, kterou metodu vložení zvolíte, je nutné nastavit několik základních nastavení pro propojení se správným botem. **Tyto údaje jsou v ukázkách vyznačeny pomocí `...` a dostanete je od Feedyou** - konkrétně jde o `directLine.secret` a `bot.id`. Během testování je možné využít hodnoty uvedené v komentáři v dané ukázce.

Další nastavení

Existují i další nastavení, která je možné využít pro úpravy chování WebChatu:

- `locale` - umožní změnit jazyk textů v WebChat komponentě z výchozí češtiny (např. en, sk ,sr, hu

...)

- `disableInputWhenNotNeeded` - pokud nastaveno na `true`, skryje vstup pro text, pokud se bot zrovna uživatele na nic neptá (vhodné pro virtuální asistenty bez NLP modelu)
- `user.id` a `user.name` - v případě že známe přihlášeného uživatele, můžeme botovi předat jeho ID nebo jméno
- `userData` - objekt, kterým je možné předat libovolné informace, které víme o uživateli, botovi (např. email, telefon, apod.)
- `startOverTrigger` - umožňuje navázat událost na libovolný element, která zapříčiní restart konverzace - např. tlačítko "Začít znovu" (více info na GitHubu i na v README NA GitHubu)

```
directLine: { secret: '...' },
bot: { id: '...', name: 'Chatbot' },
theme: { mainColor: '#d83838' },
locale: 'en', // Zkratka jazyka, ve kterém mají být prvky webchatu, např. text inputu pro text
disableInputWhenNotNeeded: true // Skrýt input pro text, dokud není uživatel vybídnut, např. pomocí promptu
```

Úpravy vzhledu

Vzhled chatu je pak možné libovolně přizpůsobit pomocí CSS stylů - ideálně pravidla všech selectorů začínat pomocí `.wc-app`. Díky otevřenosti k takovýmto možnostem customizace se může na druhou stranu stát, že bude WebChat nepříznivě ovlivněn stylováním webu (např. obecná pravidla na základních elementech jako `<button>` apod.) - v tomto případě je nutné tyto problémy vyresetovat upřesňujícími zacílenými pouze na potomky `.wc-app`.

URL parameters

WebChat je možné ovlivnit následujícími parametry v URL vhodnými např. během testování:

- `#feedbot-test-mode` nastavuje `testMode: true` do nastavení `userData`, což v botovi zapíná testovací režim pro daného uživatele
- `#feedbot-intro-dialog` přepisuje výchozí dialog, kterým bot začíná konverzaci (může být také nastaven pomocí parametru `introDialog.id` v nastavení webchatu)
- logiku předávání parametrů z části URL za `#` do bota je možné udělat i vlastní, a to dvěma způsoby
  - pokud např. rozesíláme odkaz uživatelům, které známe, můžeme podle ní nastavit `user.id`, díky čemuž bude možné dohledat, kterou konverzaci provedl který uživatel a navíc bude možné poznat a reagovat, pokud se na daný odkaz vrátí v budoucnu
  - podle parametru nastavit libovolnou proměnnou v `userData` a tu v komunikaci bota pak využít, nebo se podle ní rozhodnout (např. jestli přistupuje přímo, tzn. bez parametru v URL nebo z QR kódu, kde nějaký parametr bude nebo např. z kiosku, kde je taky nějaký parametr

přednastaven)
    - podle parametru vybrat, který dialog se spustí místo výchozího dialogu `main`

```
{
  // ...
  userData: { zdroj: location.hash.substr(1) }, // vše za # v URL se předá do proměnné "zdroj"
  introDialog: { id: location.hash.substr(1) === 'xyz' : 'abc' : 'main' } // pokud je v URL #xyz, spustit dialog "abc"
}
```

Programové otevření chatu

Pokud je potřeba chat otevřít při nějaké akci na stránce, např. kliku na tlačítko, je k tomu možné využití čistý JS emulující kliknutí na hlavičku zavřeného chatu.

```
document.querySelector(".feedbot-wrapper.collapsed .feedbot-header").click()
```

Komunikace na pozadí

WebChat může s VA kromě přenosu klasických zpráv zpráv komunikovat i na pozadí pomocí událostí (tzv. back-channel).

Bot → WebChat

Následuje ukázka, jak zareagovat na libovolnou událost nadefinovanou kdekoli v komunikačním stromu přesně ve chvíli, kdy daný uživatel místem prochází:

```
// místo klasického nastavování secretu v propertou directLine je potřeba vytvořit
// objekt BotConnection a ten předat do nastavení
var botConnection = new BotChat.DirectLine({
  secret: "...",
});

BotChat.App({
  botConnection: botConnection,
  bot: { id: "feedbot-..." },
});

// na něm je pak možné zaregistrovat callbacky na určité události a z activity.value
```

```
// použít libovolnou hodnotu předanou z bota
botConnection.activity$.filter(function (activity) {
  return (
    activity.type === "event" && activity.name === "set-reload-timeout"
  );
}).subscribe(function (activity) {
  setTimeout(function () {
    location.reload();
  }, parseInt(activity.value || 60000));
});
```

WebChat → Bot

Naopak stránka, na které WebChat běží může v botovi vyvolat některou z následujících událostí pomocí jednoduše pomocí triggeru události na `window` objektu:

- `feedbot: trigger-dialog` starts selected dialog specified in dialog property of CustomEvent (for example `window.dispatchEvent(new CustomEvent('feedbot: trigger-dialog', { detail: 'package-status' }))`)
- `feedbot: start-over` restarts conversation, which is the same behavior as startOverTrigger callback in config above (for example `window.dispatchEvent(new CustomEvent('feedbot: start-over'))`)

If support of Internet Explorer is required, please provide use fallback to support custom event creation there.

Revision #18
Created 19 February 2020 09:38:10
Updated 21 July 2022 08:06:32 by Jan Dvořák