




Webframeworks


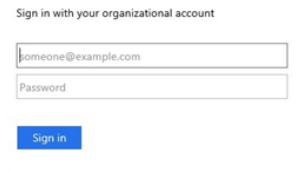
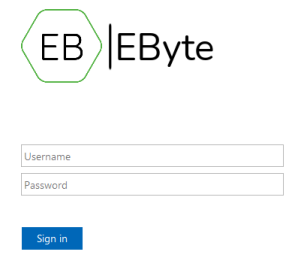
	ASP.NET Core 	Django	Ruby on Rails	Next.js 
Kurzbeschreibung	<p>ASP.NET Core ist ein Webframework von Microsoft. Für ASP.NET Core Projekte gibt es verschiedenste Vorlagen, z.B. für APIs oder Webapps mit MVC (Model View Controller) Pattern. ASP.NET Core ist vor allem für Businesskritische Anwendungen beliebt, da Microsoft Support dazugekauft werden kann. StackOverflow, als Beispiel, baut auf ASP.NET Core.</p>	<p>Django ist ein Webframework welches dem MVP (Model View Presenter) Pattern folgt. Django wirbt mit einfacher Entwicklung. Auf der Website heisst es; «Das Webframework für Perfektionisten mit Deadlines». Das Disqus Backend ist mit Django realisiert worden.</p>	<p>Ruby on Rails ist ein Webframework welches dem MVC (Model View Controller) Pattern folgt. Ruby on Rails wirbt damit, dass alles was ein Developer anfangs braucht, schon vorhanden ist (z.B. Module wie der Action Mailer). Über die Jahre wurde Ruby on Rails von vielen grossen Websites verwendet, darunter sind Twitter, GitHub und Twitch.</p>	<p>Next.js ist ein React Frontend-Framework, dass vor allem dadurch bekannt geworden ist, dass es hybride SSG (Static Site Generation) und SSR (Server Side Rendering) unterstützt. Hybride SSR ist sehr interessant, weil so Seiten zum Beispiel durch Events neu generiert werden können. Für einen Webshop wäre das interessant, weil die Seiten generiert werden könnten, wenn ein Produktattribut ändert. Next.js wird zum Beispiel von Nike, AT&T und Hulu verwendet.</p>
Unterstützte 'Programmiersprachen'	Theoretisch alle .NET (d.h. eigentlich alle CLR kompatiblen) Sprachen, in der Praxis vor allem C#, F# oder VB.	Python	Ruby	JavaScript, TypeScript
Einfach erweiterbar / 'Plug-In' Marketplace	Mit nuget Paketen	Mit Django Packages	Mit Gems	Mit NPM Packages
Was ich sonst noch interessant finde	Mir war ja bekannt, dass ASP.NET performant ist, nicht aber, dass es fast 10 mal schneller als Node.js ist.	-	-	Next.js unterstützt Server-Side-Rendering, damit möchte ich irgendwann noch herumspielen.
Quellcodelizenzmodell	Open-Source mit MIT Lizenzmodell.	Open-Source mit eigenem Lizenzmodell.	Open-Source mit MIT Lizenzmodell.	Open-Source mit MIT Lizenzmodell.
Plattform	Plattformunabhängig			

 Möchte ich vorschlagen.

Plug-Ins

	componentspace.saml2 ✓	prometheus-net.aspnetcore	amazon.lambda.aspnetcoreserver	aspnetcore.ratelimit ✓
Erkenntnis	<p>componentspace.saml2 ist ein Plug-In, welches in ASP.NET Applikationen verwendet werden kann. Es erlaubt das Einbinden von SAML IdPs (Identity Providern) und SPs (Service Provider) für SSO (Single Sign-On).</p>	<p>prometheus-net.aspnetcore ist ein Plug-In für ASP.NET, dass es erlaubt Metriken für Prometheus zu exportieren. Die gesammelten Daten können dann mit Prometheus visualisiert und analysiert werden. Mit dem Prometheus Alertmanager kann man sich dann benachrichtigen lassen, wenn die gesammelten Daten nicht der Erwartung entsprechen, zum Beispiel wenn die Latenz eines bestimmten API-Aufrufes höher als sonst ist.</p>	<p>amazon.lambda.aspnetcoreserver erlaubt es, ASP.NET Core Web APIs über AWS Lambda Funktionen gesteuert laufen zu lassen. So können Requests von einem ELB (Elastic Load Balancer) oder einem AAG (Amazon API Gateway) abgezweigt, in der ASP.NET Core Web API verarbeitet und dann an den ELB oder das AAG zurückgeliefert werden.</p>	<p>aspnetcore.ratelimit ermöglicht es, Ressourcen in ASP.NET Applikationen mit einem rate-limit (Durchsatzbegrenzung) zu versehen. So könnte man zum Beispiel festlegen, wie viele Requests in einer bestimmten Zeit auf eine Ressource erfolgen dürfen, bevor die Requests nicht mehr beantwortet werden. Rate-limiting kann zum Beispiel vor DoS (Denial of Service) Angriffen schützen.</p>

Scripts

	AD FS Login Seite abändern	Werbung rotieren	Boterkennung 	Automatischer Darkmode
Kurzbeschreibung	<p>Um die Login-Nachricht und die Placeholder für die Eingabefelder von meiner ADFS Login-Seite zu abzuändern, habe ich ein kleines Script geschrieben, was nach den Elementen im DOM sucht, und sie, sofern sie gefunden werden, ersetzt/abändert.</p> <p>Beispiel Login-Nachricht:</p> <pre>// Check whether the loginMessage element is present on this page. if (loginMessage) { // loginMessage element is present, modify its properties. loginMessage.innerHTML = ""; }</pre> <p>Das Resultat ist ein Login-Form ohne Login-Nachricht:</p> <div> <div> <p>Vorher:</p>  </div> <div> <p>Nachher:</p>  </div> </div>	<p>Ein weiteres Beispiel für die Verwendung von Scripts wäre, das Rotieren von Werbeanzeigen (nicht mit Google Ads oder anderen vorgefertigten Werbesystemen) auf Grund von bestimmten Bedingungen.</p> <p>So könnte man eine Bestimmte Anzeige anzeigen, wenn der Seitennutzer einen vordefinierten User-Agent nutzt, sich länger als 30 Sekunden an der gleichen Position aufhält, ein bestimmten Client-Typ nutzt, etc.</p>	<p>Es wäre auch möglich, mit JavaScript an die Client IP-Adresse zu kommen und je nach Herkunftsland/User-Agent/etc. eine Captcha anzuzeigen, bevor der User die Seite nutzen kann.</p>	<p>Theoretisch könnte man ein Script schreiben, welches durch die IP des Clients die Geolocation herausfindet und dann bei Sonnenauf- und untergang zwischen dem Light- und Darkmode wechselt. Sinnvoll nein, machbar ja.</p>

Ich konnte ehrlich gesagt sehr schwer einschätzen, was hier der gefragte Umfang ist. Vielleicht wäre ein kleines Beispiel (auf diesen Auftrag bezogen und nicht irgendein Programm) ganz hilfreich gewesen.