

## نرم افزار تحت ویندوز و وب و تفاوت آنها



**Web Apps**

VS



**Desktop Apps**

**نرم افزار تحت ویندوز چیست.** این نرم افزارها که برعکس نرم افزارهای تحت وب معمولا با زبان های برنامه نویسی خانواده C به ویژه C# سی شارپ نوشته می شوند. نرم افزارهایی هستند که برای اجرا نیاز به نصب دارند و با توجه به شرایط نصب و سازگاری آنها با ویندوزهای مختلف انتخاب می شوند. **نرم افزارهای تحت ویندوز** در جاهایی که سرعت و امنیت نرم افزار بسیار مهم باشد مانند فرو شگاه ها و سازمان ها بسیار مورد استفاده قرار می گیرند.

از ویژگی های بارز این نرم افزارها امکان دسترسی به رجیستری است، در صورتی که نرم افزارهای تحت وب این گونه نیست. این برنامه ها کاملا در سمت کاربر اجرا و نصب می شود و نیازی به پردازش یا تغییرات از سمت سرور نیست، به همین دلیل پاسخگویی و سرعت آنها بسیار بهتر از نرم افزارهای تحت وب است.

در نگاه اول در نوع دسترسی به این نرم افزارها است. شما برای استفاده از نرم افزار تحت وب نیاز به اینترنت و وب سایت نرم افزار دارید. به طور کلی به نرم افزارهایی که در محیط وب و در بستر اینترنت اجرا می شوند، نرم افزار تحت وب می گویند. وجه اشتراک همه نرم افزارهای تحت وب اجرای آنلاین آنها و نصب بر سرور است. نرم افزارهای تحت ویندوز به نرم افزارهایی گفته می شود که بر یک سیستم عامل اجرا می شوند. معمولا ویژگی های سخت افزار و نسخه ویندوز شما تعیین کننده نسخه نرم افزار سازگار با ویندوز است. معمولا نرم افزارهای تحت وب برای ارائه خدمات بوده و استفاده آنها رایگان است و بعضی از خدمات آنها هزینه دارد ولی بیشتر نرم افزارهای تحت ویندوز خریدنی است و شما در ابتدا هزینه آن را باید پرداخت کنید. در ایران به دلیل رعایت نشدن قانون کپی رایت، نسخه های غیرقانونی نرم افزارها به صورت رایگان در دسترس وجود دارد، به همین دلیل نسخه های تحت ویندوز بیشتر مورد استفاده قرار می گیرد

در نرم افزار های تحت ویندوز دسترسی به دیتا بیس و سیستم فایل فراهم است در صورتی که در نرم افزار تحت وب این دسترسی ها محدود شده است.

## مزایا و معایب برنامه های تحت وب و برنامه های تحت ویندوز

یکی از مهم ترین تفاوت نرم افزار تحت وب و ویندوز زمانی حس می گردد که نیاز به پردازش اطلاعات بیشتر می شود. با پیشرفت های ایجاد شده در صنعت سخت افزار و ایجاد قابلیت های بیشتر در سیستم عامل ها، نیاز است تا نرم افزار ها با سرعت و دقت بیشتری کار کنند. به همین دلیل نرم افزارهای تحت ویندوز به طور مرتب باید به روز رسانی و آپدیت شوند. ولی در نرم افزارهای تحت وب با به روز شدن نسخه نرم افزار بر روی سرور ، نسخه ارائه شده به تمامی کلاینت ها نیز به روز می گردد و مانند نرم افزارهای ویندوز نیازی به بروزرسانی نرم افزار به ازای هر کاربر نیست. همچنین توسعه نرم افزار های تحت وب سریع تر است به دلیل وجود جامعه بزرگتر برنامه نویس ها

**نرم افزارهای مبتنی بر وب** زیادی در دنیای اینترنت جهانی وجود دارد که معروف ترین آنها سرویس های ایمیلی مانند جیمیل، ایمیل یاهو و سرویس دهنده های ایمیل معروف است. این نرم افزارها معمولا با یکی از زبان های طراحی وب مانند PHP، JAVA، پایتون و ... نوشته می شوند. **نرم افزارهای مبتنی بر وب** معمولا برای ارائه خدماتی مانند خرید آنلاین، بانکداری، حسابرسی و حسابداری و امور مشابه به کار می روند. برای دسترسی به این نرم افزارها باید از طریق مرورگر خود به دنیای وب متصل شوید و کیفیت استفاده از این خدمات تا حدودی به کیفیت اینترنت شما بستگی دارد.

یکی از نکات برجسته این نوع نرم افزار، ربط نداشتن به سیستم عامل است و فرقی نمی کند شما چه سیستم عاملی استفاده می کنید. زیرا این نرم افزارها در محیط مرورگر اجرا می شوند. امکان استفاده بر روی اکثر سیستم عامل ها اجرا و قابلیت استفاده بر دستگاه های مختلف مانند تلفن های همراه هوشمند، تبلت و حتی ساعت های هوشمند در هر

ساعتی از شبانه روز بدون نیاز به نصب نرم افزار یا همراه داشتن سیستم کامپیوتری پیشرفته وجود دارد.

## **معایب نرم افزار تحت وب**

**معایب نرم افزار تحت وب** شامل پاسخگویی پایین تر نسبت به نرم افزارهای ویندوز است. در اکثر این نرم افزارها به دلیل قرار گرفتن پایگاه داده و واحد پردازنده اصلی در سرور، تمامی درخواست ها به سرور فرستاده می گردد و هرگونه اختلال یا تاخیر در ارتباط با سرور منجر به قطع نرم افزار خواهد شد. از دیگر معایب نرم افزار تحت وب پایین تر بودن میزان دسترسی برای تغییر رابط کاربری را نام برد. بیشتر تغییرات در نرم افزارهای تحت وب معمولاً در سمت سرور صورت می گیرد به همین دلیل ممکن است این امر نارضایتی کلاینت را به دنبال داشته و تصحیح آن سخت تر خواهد بود.

از نظر عملکرد، برنامه های دسکتاپ و وب به طور فزاینده ای از نظر امکانات به یکدیگر نزدیک می شوند. در بسیاری از موارد، محدودیت های عملکردی دیگر عاملی در انتخاب بین دسکتاپ و وب نیستند. با این حال، چند تفاوت مهم بین این دو وجود دارد.

## **1: نصب و راه اندازی**

با یک برنامه دسکتاپ، نرم افزار را روی رایانه شخصی (یا لپ تاپ) نصب می کنید یا یک میانبر برای برنامه روی سرور ایجاد می کنید. می توانید با مرورگری مانند کروم، فایرفاکس، اج یا سافاری به یک برنامه وب دسترسی داشته باشید. مزیت بسیاری از برنامه های کاربردی وب این است که بسته به تنظیمات امنیتی، می توانید از هر رایانه شخصی به برنامه دسترسی داشته باشید. اگر ایستگاه کاری یا لپ تاپ ثابت ندارید، این کار مفید است.

## 2: برنامه دسکتاپ بومی

یک برنامه دسکتاپ خوب یک برنامه بومی است. این بدان معنی است که از دستورالعمل های توسعه دهنده سیستم عامل مانند ویندوز یا macOS پیروی می کند. این در مواردی مانند نحوه عملکرد یک برنامه و میانبرهای صفحه کلید بیان می شود. وقتی از برنامه استفاده می کنید، لازم نیست همه آن ترفندها را دوباره یاد بگیرید. احساس آشنایی دارید و استفاده از آن را آسان تر می کند.

## 3: سرعت

مردم بی حوصله هستند و به همین دلیل مهم است که یک برنامه سریع باشد. به طور متوسط، برنامه های دسکتاپ همچنان سریعتر از برنامه های وب هستند. این به این دلیل است که اکثر برنامه های وب در هنگام تغییر صفحه زمان بارگذاری دارند.

از نظر تئوری، برنامه های دسکتاپ زمان بارگذاری ندارند - پس از راه اندازی. همه عناصر در دسترس هستند و تنها زمانی که ممکن است لازم باشد منتظر بمانید زمانی است که برنامه در حال دریافت مقادیر زیادی داده از یک منبع خارجی است.

برنامه های دسکتاپ می توانند اولین تجربه محلی را ارائه دهند که به شما امکان می دهد به صورت آفلاین، در هر کجا و هر زمان که می خواهید کار کنید.

## Electron PWA

در مقدمه اشاره کردیم که اپلیکیشن های دسکتاپ و وب از نظر عملکردی بیش از پیش معادل هستند. این همچنین در تکنیک هایی مانند PWA و Electron که توسعه دهندگان استفاده می کنند منعکس می شود.

PWA برنامه های وب پیشرو ، برنامه های وب هستند که شما از مرورگر خود به عنوان "برنامه دسکتاپ" ذخیره می کنید. این هنوز یک برنامه وب است، اما شما آن را طوری باز می کنید که گویی یک برنامه دسکتاپ است.

برنامه های الکترونیکی برنامه های دسکتاپ هستند که با فناوری های وب توسعه یافته اند. شما آنها را درست مانند برنامه های دسکتاپ نصب می کنید، و اغلب توسعه دهنده یک نسخه وب را نیز ارائه می دهد. برای توسعه دهنده، Electron یک راه کارآمد برای توسعه است زیرا شما برنامه ای را توسعه می دهید - نسبتاً سریع - که روی پلتفرم های مختلف مانند Windows، macOS و Linux کار می کند.

الکترون حد وسطی بین دسکتاپ و وب است. حد وسط معمولاً به معنای سازش است، و این برای بسیاری از برنامه های Electron نیز صدق می کند:

به طور کلی، برنامه های Electron کندتر از برنامه های دسکتاپ هستند. از نظر سرعت، آنها با برنامه های وب قابل مقایسه هستند. Visual Studio Code یک استثنا مثبت در این مورد هست.

دومین اشکال این است که برنامه های Electron معمولاً مانند برنامه های دسکتاپ بومی نیستند. با این کار، تشخیص و سهولت خاصی از استفاده را از دست می دهید.

برنامه های الکترونیکی کارایی کمتری دارند. تیم های میکروسافت نمونه ای از این موارد است. از قدرت پردازش، حافظه و انرژی زیادی استفاده می کند.

مزیت بزرگ Electron این است که یک برنامه اغلب در همه سیستم عامل های اصلی در دسترس است.

## SaaS و ابر

قبلاً اشاره کردیم که یک برنامه دسکتاپ را روی رایانه شخصی یا مک نصب می کنید. آن کاملاً درست نیست. چندین راه حل وجود دارد که به شما امکان می دهد از برنامه های دسکتاپ از راه دور استفاده کنید، درست مانند برنامه های وب. به عنوان مثال، در CE-IT، ما با Microsoft RDS کار می کنیم. با این کار مرورگر خود را باز می کنید - درست مانند برنامه های وب - و URL را وارد می کنید. از آنجا برنامه دسکتاپ را بدون نصب چیزی راه اندازی می کنید. از نظر راحتی، بنابراین شبیه یک برنامه وب است.

## پلت فرم های مخلوط

تا به حال، ما عمدتاً به انتخاب بین وب و دسکتاپ نگاه می کردیم، اما نباید آنقدر سیاه و سفید باشد. اگر چندین نوع کاربر دارید، ممکن است بخواهید راه حل های مختلفی ارائه دهید. شخصی که از همه عملکردها استفاده می کند - و هر روز از یک رایانه شخصی استفاده می کند - از یک برنامه دسکتاپ بهره می برد تا بتواند به طور بهینه از برنامه استفاده کند. همکار که گهگاه از برنامه با مجموعه ای از عملکردهای محدود استفاده می کند، ممکن است با یک برنامه وب بهتر باشد.

این ترکیب را در برنامه طراحی Sketch مشاهده می کنید. برنامه دسکتاپ همه ویژگی ها را ارائه می دهد و یک برنامه وب نیز وجود دارد. شما از دومی برای همکاری و به اشتراک گذاری طرح ها استفاده می کنید. برنامه وب ساده تر از نسخه دسکتاپ است که منحنی یادگیری را محدود می کند.

این رویکرد ترکیبی پلتفرم با دیدگاه ما که یک اندازه مناسب برای همه معمولاً کار نمی کند، همسو است. برای ما، برنامه دسکتاپ را در مرکز قرار می دهد و در عین حال آنچه دیگران نیاز دارند را منعکس می کند.

## تفاوت سرعت در c-javascript-c#

برای مثال تولید اعداد تصادفی را در این 3 زبان بررسی میکنیم.

مقایسه جاوا اسکریپت، سی شارپ و سی شارپ با 10 میلیون اعداد تصادفی با استفاده از بخش‌های کد زیر:

```
// C
srand(time(0));
for (int i = 0; i < 10000000; i++)
    rand();

// C#
var rand = new Random();
for (var i = 0; i < 10000000; i++)
    rand.Next();

// JavaScript
for (let i = 0; i < 10000000; i++)
    Math.random();
```

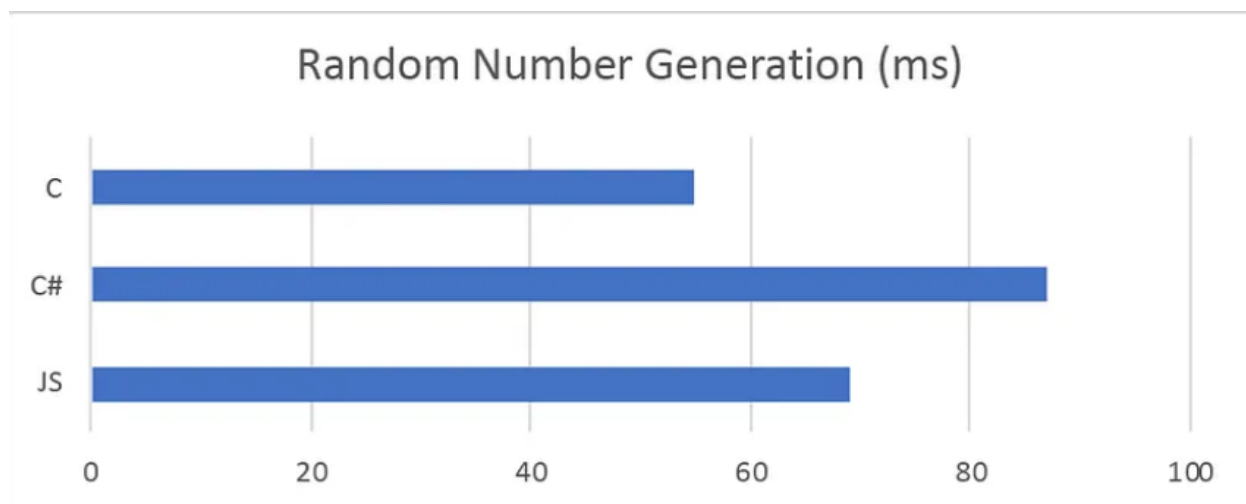
C: 54.738 milliseconds, ~18 per second

C#: 87 milliseconds, ~12 per second

JS: 69 milliseconds, ~15 per second



نتیجه به دست آمده :



### نتیجه گیری

هنگامی که نرم افزاری را توسعه می دهید، باید فراتر از آنچه می تواند به صورت کاربردی انجام دهد، نگاه کنید. شما نگاه می کنید که چه کسی از آن استفاده می کند، کسی کجا کار می کند و چگونه کار می کند. این به شما امکان می دهد برنامه هایی بسازید که افراد را بهره ورتر کنند و از استفاده از آن لذت ببرند.