



بسمه تعالی

دانشکده فنی و مهندسی باهنر شیراز

**درس: مهندسی نرم افزار**

**عنوان پروژه: دواپس (DevOps)**

**گردآورنده:**

پوریا سمیرانی نژاد

**ش.د: ۰۰۱۲۱۰۸۵۷۰۵۰۱۵**

**استاد مربوطه: آقای مهندس سهرابی**

پاییز ۱۴۰۰

# فهرست مطالب

صفحه

عنوان

- ۱ ..... مقدمه
- ۲ ..... دواپس چیست؟
- ۳ ..... اجرای برنامه‌ها با دواپس
- ۴ ..... فرآیند دواپس چیست؟
- ۶ ..... دلیل اهمیت دواپس چیست؟
- ۷ ..... تفاوت اجایل و دواپس چیست؟
- ۸ ..... اصول دواپس (DevOps)
- ۱۰ ..... منابع

# فهرست اشکال

صفحه

عنوان

شکل ۱ - فرایند دواپس..... ۱

شکل ۲ - مزایای دواپس..... ۲

شکل ۳ - مراحل مختلف دواپس..... ۳

شکل ۴ - تفاوت اجایل و دواپس..... ۷

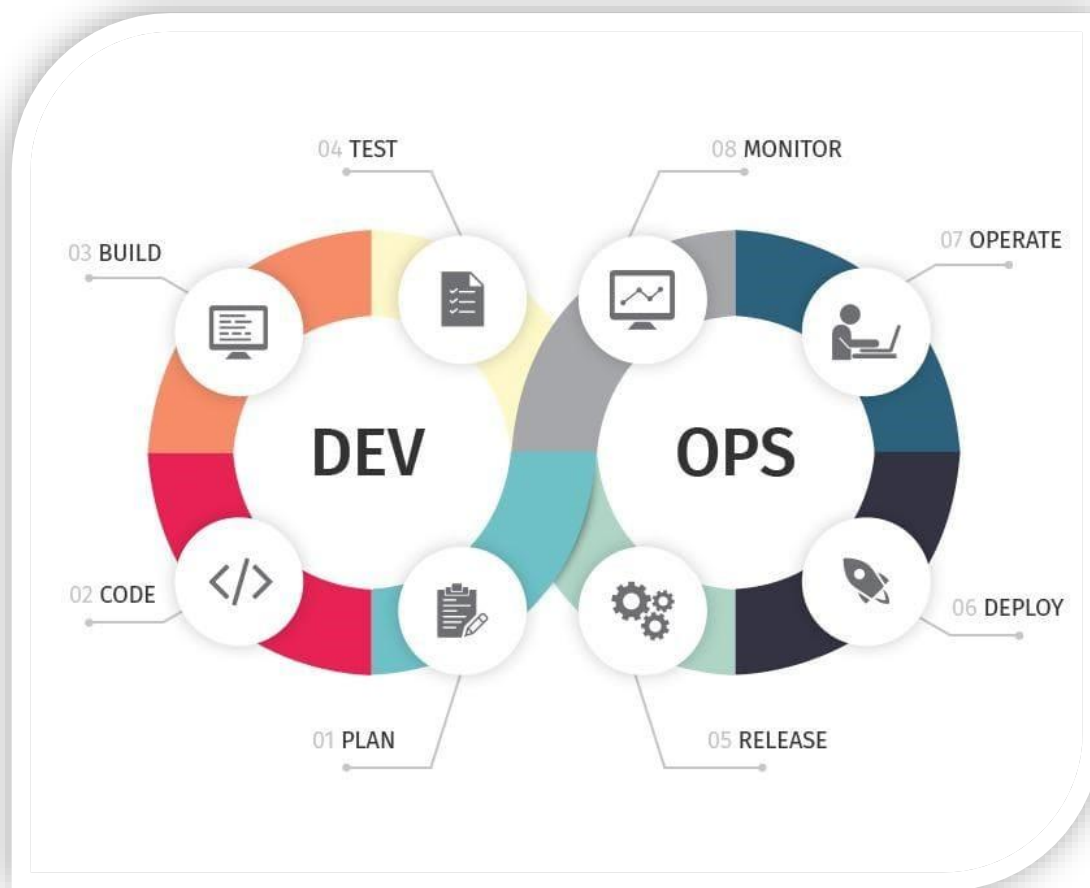
شکل ۵ - اصول دواپس..... ۹

## مقدمه:

روند توسعه برنامه‌های نرم‌افزاری یا برنامه‌های وب از ایده‌پردازی آن‌ها تا استقرارشان، پیچیده است. با گذر از چندین مرحله پیچیده توسعه، یک برنامه وب یا نرم‌افزار قبل از تولید در چندین سطح آزمایش می‌شود.

در بیشتر موارد، توسعه برنامه‌های نرم‌افزاری به دلیل مشخصات و پیچیدگی‌های آن زمان‌بر می‌شوند. به منظور ارائه برنامه در یک بازه زمانی کوتاه‌مدت، توسعه‌دهندگان نرم‌افزار مجموعه‌ای جهانی را دنبال می‌کنند که چرخه عمر دواپس (DevOps Lifecycle) نامیده می‌شود.

بنابراین، دواپس در دنیای توسعه برنامه‌های نرم‌افزاری چیست؟ در این مقاله به معنی، کاربردها و همچنین در شکل (۱) هر مرحله حساس در چرخه عمر دواپس خواهیم پرداخت.



شکل ۱- فرایند دواپس

## دواپس چیست؟

دواپس یک اصطلاح ساده و مختصر می‌باشد که از کلمات “توسعه (Development)” و “عملیات (Operations)” گرفته شده است. این مجموعه شامل تمرین چندین وظیفه توسعه برنامه به روشی خاص می‌شود. به عبارت گسترده‌تر، ترکیبی از توسعه نرم‌افزار و عملیات فناوری اطلاعات عنوان با عنوان دواپس نام‌گذاری شده است.

قبل از دواپس، برنامه‌های نرم‌افزاری یا با مدل آبشاری (Waterfall) یا با مدل توسعه چابک (Agile) ساخته شده بودند:

- آبشار یا مدل پی در پی، یک رویکرد پیشگامانه در چرخه عمر توسعه نرم‌افزار (SDLC) بود. با استفاده از این مدل، توسعه نرم‌افزار به فرایندی خطی تبدیل می‌شود که در آن مراحل و وظایف مختلف به طور متوالی قرار می‌گیرند.
- از طرف دیگر، توسعه نرم‌افزار چابک شامل استفاده از رویکردهای مختلف و همکاری چندین تیم در چرخه عمر توسعه نرم‌افزار بود.



شکل ۲- مزایای دواپس

خطی بودن مدل آبشار و عملکرد متقابل توسعه نرم‌افزار چابک قادر به اطمینان از تحویل سریع و مداوم برنامه‌های نرم‌افزاری بی‌نقص نبودند.

در شکل (۲) خدمات دواپس با هدف کاهش زمان صرف‌شده در چرخه عمر توسعه نرم‌افزار، برای تحویل سریع و اطمینان از تولید مداوم برنامه‌های نرم‌افزاری باکیفیت، وارد عمل شدند.

## اجرای برنامه‌ها با دواپس:

اثر بخشی دواپس در ایجاد رابطه چابک بین عملیات فناوری اطلاعات و توسعه نرم‌افزار توسط چندین عامل پشتیبانی می‌شود. با امکان ایجاد ارتباطات بهتر در چندین واحد کسب و کار در بخش توسعه نرم‌افزار و فناوری اطلاعات، دواپس با اعمال تغییرات زیر، عملیات تولید نرم‌افزار را به طور کلی بهبود بخشیده است:

- محیط عملیاتی پایدار
- تحویل فوق‌العاده سریع
- همکاری ثابت قدم
- بهینه‌سازی زمان، به ویژه در مراحل تعمیر
- نوآوری مداوم



شکل ۳- مراحل مختلف دواپس

## فرآیند دواپس چیست؟

در طی این فرآیند DevOps، عملیات توسعه نرم افزار و اعمال تغییرات زیرساختی به صورت خودکار انجام می‌شود. این فرآیند به ۵ مرحله اصلی تقسیم می‌شود.

### ۱. برنامه ریزی (Planning)

در مرحله اول، تیمی اهداف پروژه را ترسیم کرده و ساختار کلی نرم افزار را تعیین می‌کنند. متخصص دواپس باید از دانش فنی اعضای تیم استفاده کرده و بررسی کند چگونه می‌توان سازمان را در قالب یک سیستم یکپارچه به تمام اهداف مورد نظر رساند. پس از برنامه ریزی‌های اولیه، متخصص دواپس باید به دنبال راهکارهایی برای انجام خودکار فرآیندهای مختلف باشد. به بیان دیگر، باید قادر باشد تیم را در جهت افزودن قابلیت‌ها و تکنولوژی‌های تولید شده به سیستم، هدایت کند. در این مرحله مهندس devops باید برای سوالات زیر پاسخ مناسب پیدا کند:

دو سرویس مختلف چگونه با هم تعامل می‌کنند؟

برای مرتبط کردن این دو سرویس از چه راهکاری باید استفاده کرد؟

چه چیزهایی را باید تولید کنیم و چه چیزهایی را باید خریداری کنیم؟

آیا یک وظیفه خاص در سیستم می‌تواند به صورت خودکار انجام شود؟

### ۲. توسعه (Development)

اکنون که چارچوب کلی کار مشخص شده، نوبت توسعه دهندگان یا Developer ها است تا کد بنزند و فیچرهای پیش‌بینی شده را برای نرم افزار توسعه دهند. در این مرحله مهندس دواپس باید راهی پیدا کند تا توسعه دهندگان بتوانند بهترین فعالیت را در کوتاه‌ترین زمان ممکن انجام دهند. در واقع او به دولوپرها می‌گوید که از کدام ابزارها استفاده کنند و این ابزارها را در اختیارشان قرار می‌دهد. همچنین بخش‌های مختلف کدهایی که به وسیله دولوپرها در محیط توسعه نوشته شده‌اند را کنار هم قرار داده و آن‌ها را با محیط نهایی نرم افزار سازگار و هماهنگ می‌کند. سوالاتی که متخصص دواپس در این مرحله ممکن است با آن‌ها روبرو شود، عبارتند از:

چگونه می‌توانیم کارایی و بهره‌وری دولوپرها را افزایش دهیم؟

چگونه محیط نهایی نرم افزار را برای توسعه دهندگان ترسیم کنیم؟

توسعه دهندگان در چه شرایطی می توانند از ابزارهای مورد علاقه خود استفاده کنند؟

### ۳. آزمایش کردن (Testing)

در این مرحله کدها به وسیله توسعه دهندگان و مسئولان کنترل کیفیت (QC) تست شده و برای یکپارچه شدن با سورس کد اصلی آماده می شود. اما توجه داشته باشید که برای اجرای دستی کدها بر روی سیستم های داخلی سازمان هنوز به حضور دولوپرها و مسئولان کنترل کیفیت نیاز داریم.

اکنون مهندس دواپس با استفاده از ابزارهای (Continuous Integration) CI مانند Jenkins ، Bamboo یا Drone باید برای تکرار خودکار تست ها راهی پیدا کند تا به این وسیله تست مداوم کدها آسان تر شود. موضوعاتی که یک متخصص دواپس در این مرحله با آن ها روبرو است عبارتند از:

چگونه تاریخچه تست ها را بررسی کنیم و با استفاده از آن ها روندهای موجود را پیدا کنیم؟

بعد از تست کدها، چگونه مشکلات احتمالی را به توسعه دهندگان اعلام کنیم؟

چطور می توانیم چندین محیط مشتری (Client Environment) تکرارپذیر ایجاد کنیم؟

### ۴. استقرار (Deployment)

در این مرحله باید تعیین کنید کدهای نوشته شده چگونه و با چه نظمی در محصول نهایی قرار داده شوند تا کاربران نرم افزار بتوانند از سرویس شما استفاده کنند. در واقع Deployment در اینجا به معنای آپلود کدها بر روی سرور اصلی نرم افزار است. در اینجا نیز متخصصان دواپس از ابزارهای CI استفاده می کنند. سوالاتی که در این مرحله باید پاسخ داده شوند، عبارتند از:

چگونه می توانیم فرآیند Deploy شدن یا استقرار را به صورت خودکار در آوریم؟

چطور مطمئن شویم سرویسی که به تازگی مستقر شده، اختلال و مشکلی ایجاد نمی کند؟

چگونه بدون اینکه کاربر متوجه شود، سرویسی را Deploy کنیم؟

نسخه نهایی نشده از نرم افزار، چه زمانی برای Deploy شدن آماده است؟

چگونه می توانیم فرآیند استقرار را با روش های تکرارپذیر انجام دهیم؟



## ۵. نگهداری (Maintenance)

این مرحله در مورد انجام تمام فعالیت‌هایی است که باعث در دسترس قرار گرفتن یک سیستم و حفظ کارایی آن می‌شود. توجه داشته باشید که این مرحله بیشترین زمان یک مهندس دواپس را به خود اختصاص می‌دهد. موضوعات و سوالات این مرحله را در ادامه بیان کرده‌ایم:

چگونه می‌توانیم از باگ‌ها و مشکلات موجود در محصول مطلع شویم؟

چگونه باگ‌های زیرساختی موجود در محصول را برطرف کنیم؟

چطور می‌توانیم از سلامت و کارایی همه سرویس‌ها مطمئن شویم؟

چگونه مشکلات مختلف موجود در سرویس را به تیم مناسب ارجاع دهیم؟

## دلیل اهمیت دواپس چیست؟

تا اینجا دانستیم دواپس چیست و فرآیند آن شامل چند مرحله است. در این بخش اهمیت اصول دواپس را بررسی خواهیم کرد.

همانطور که می‌دانید، تغییر بر اساس تحولات بازار و نیاز مشتریان از ویژگی‌های اصلی کسب و کارها در حوزه فناوری اطلاعات محسوب می‌شود. قبل از معرفی DevOps، ارتباط ضعیفی بین تیم‌های تضمین کیفیت، اجرا، توسعه و... وجود داشت، اکثر اوقات آن‌ها مشکلات ایجاد شده را به گردن هم می‌انداختند و یکدیگر را محکوم می‌کردند. از سوی دیگر، به دلیل نزدیک بودن مرز کارها، امکان نظارت دقیق فراهم نبود.

سیستم DevOps مرز میان دو تیم را از میان برداشت و آن‌ها را به هم نزدیک‌تر کرد. همچنین با خودکارسازی بسیاری از روال‌های تکراری، فرآیند تحویل ارزش به مشتریان را سرعت بخشید.

## تفاوت اجایل و دواپس چیست؟

- اجایل، نوعی متدولوژی چابک است که بر تکرارهای متناوب در تمامی مراحل تولید، تست و رسیدن به خروجی تاکید دارد. در این راهکار توسعه نرم افزار، محصول به بخش‌های کوچک تقسیم و در تست نهایی یکپارچه می‌شود. خوب است بدانید این متدولوژی به روش‌های متعدد مانند اسکرام و کانبان قابل انجام است. اما DevOps یک روش توسعه نرم افزار و نوعی فرهنگ با تمرکز بر ارتباط و همکاری بین تیم‌های سازمان برای تولید سریع تر نرم افزار است. این متد بر تولید سریع و خودکار تاکید دارد.
- فرآیندهای رویکرد Agile بر تغییرات مداوم تمرکز دارد، در حالی که رویکرد DevOps مبتنی بر تست و انتشار مداوم است.
- با کمک Agile می‌توانید پروژه‌های پیچیده را مدیریت کرده و در نهایت یک نرم افزار تولید کنید. در حالی که DevOps به دنبال راه حلی برای مدیریت ارتباطات کاری با هدف انتشار سریع و امن است.
- متد Agile با چارچوب‌هایی مانند اسپرینت و اسکرام قابل اجرا است. اما راهکار DevOps مبتنی بر همکاری است و نمی‌توانیم یک چارچوب معمولی برای اجرای آن در نظر بگیریم.



شکل ۴- تفاوت اجایل و دواپس

## اصول دواپس (DevOps) :

شرکت‌های بیشتری در حال انتقال به دواپس هستند. پیاده‌سازی دواپس مزایای بسیاری دارد، مانند استقرار سریع و آسان یکپارچه نرم افزار.

انتقال به این فرهنگ جدید بدون درک ارزش‌های اساسی که آن را پیش می‌برد، غیرممکن است. این مسئله نیاز به تغییر طرز فکر در گروه توسعه و تیم عملیاتی دارد، که آنها را به کار می‌گیرد تا به عنوان یک جبهه متحد کار کنند. اصول زیر پایه‌هایی هستند که روند مهندسی را در یک محیط دواپس هدایت می‌کند.

- کنترل نسخه

توسعه دهندگان چندین بار در روز تغییرات کد را به یک مخزن مرکزی ارسال می‌کنند. قبل از ارسال کد به مخزن اصلی (شاخه اصلی)، همه کد باید تأیید شود. برای تسهیل همکاری، سایر توسعه دهندگان می‌توانند تغییرات را پیگیری کنند.

- ادغام مداوم

اعضای تیم توسعه، کد خود را در یک مخزن مشترک، چندین بار در روز ادغام می‌کنند. هر توسعه دهنده کار را به چند کد کوچک و قابل مدیریت تقسیم می‌کند و اختلافات احتمالی ادغام و اشکالات را سریعتر تشخیص می‌دهد.

- تحویل مداوم

از آنجا که کد به طور مداوم ادغام شده است، به طور مداوم به کاربر نهایی نیز تحویل داده می‌شود. مشارکت‌های کوچک اجازه انتشار سریع‌تر به روزرسانی را می‌دهد، که عامل مهمی برای رضایت مشتری است.

- استقرار مداوم

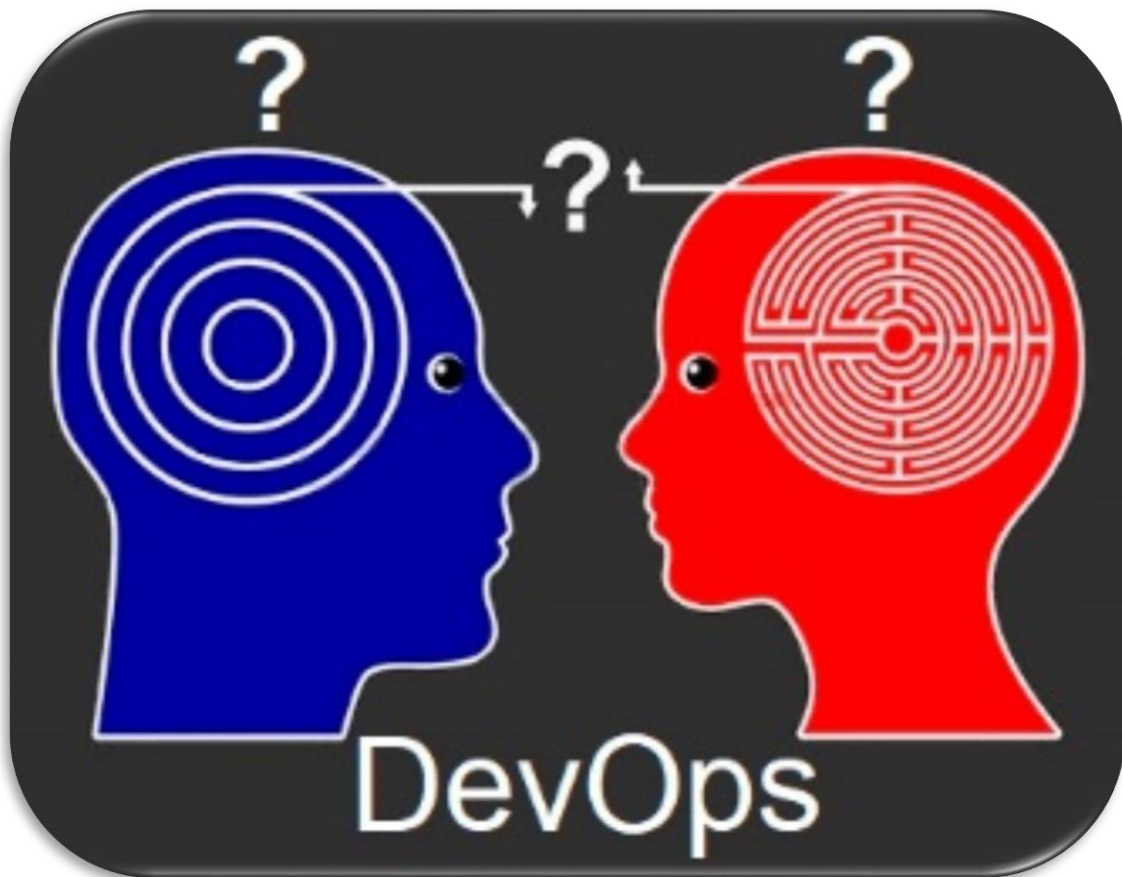
بخش عمده ای از دواپس، خودکارسازی فرایندها برای سرعت بخشیدن به تولید است. استقرار مداوم شامل انتشار خودکار به روزرسانی‌های جزئی است که تهدید قابل توجهی برای معماری موجود نیست.

- تست مداوم

چنین استراتژی شامل آزمایش هرچه بیشتر در هر مرحله از توسعه است. آزمایشات خودکار بازخورد ارزشمند و ارزیابی ریسک فرآیند موجود را ارائه می‌دهند.

• عملیات مداوم

تیم دواپس همیشه در حال کار بر روی ارتقا نرم افزار با نسخه‌های کوچک اما مکرر است. به همین دلیل دواپس نیاز به نظارت مداوم بر عملکرد دارد. هدف اصلی آن جلوگیری از خرابی و مشکلات موجود در هنگام انتشار کد است.



شکل ۵- اصول دواپس

## منابع:

- سایت مرجع علمی کار و کسب ([karokasb.org](http://karokasb.org))
- سرویس ابری آموزش مجازی ([www.irandnn.ir](http://www.irandnn.ir))
- ویکی پدیا ([fa.wikipedia.org](http://fa.wikipedia.org))
- مرجع علمی و پژوهشی ([sid.ir](http://sid.ir))
- شبکه فناوری و نوآوری ایران ([tinet.ir](http://tinet.ir))