



مدرس: دکتر خامس پناه

طراحان: [سید بارسا حسینی نژاد](#)، [امین عارف زاده](#)، [علی الهی](#)

تامین کننده‌ی منابع و حامی: ستون

مهلت تحویل: جمعه ۲۱ خرداد ۱۴۰۰، ساعت ۲۳:۵۵



## بیلد سٹان!

### مقدمه

هدف از این پروژه، آشنایی CI/CD و مدیریت کانتینرها به وسیله‌ی ابزار کوپرنیتیس<sup>۱</sup> است.

یکپارچه‌سازی پیوسته<sup>۲</sup> فرایندی است که در آن به طور مداوم تغییرات تایید شده‌ی برنامه‌نویسان در یک خط اصلی ادغام شود. این کار به وسیله‌ی یک روند اتوماتیک انجام شده و کد جدید پس از بیلد شدن و اجرای آزمون‌ها به نسخه‌ی قابل اعتمادتری تبدیل می‌شود و اگر در این روند خطایی داشته باشد، تیم توسعه دهنده به سرعت از آن باخبر می‌شود.

تحویل پیوسته<sup>۳</sup> فرایندی است که در آن پس از اطمینان از صحت کد جدید، طی فرایندی اتوماتیک یک نسخه‌ی جدید، قابل انتشار باشد. در واقع نسخه‌ی آماده‌سازی شده در آن، در یک محیط آزمایشی کوچکتر نسبت به محیط

<sup>۱</sup> Kubernetes

<sup>۲</sup> Continuous Integration

<sup>۳</sup> Continuous Delivery

عملیاتی اصلی منتشر می‌شود تا عملکرد تغییرات جدید در محیطی واقعی مورد ارزیابی قرار بگیرد اما بسته به سیاست‌های بهره‌بردار، روال انتشار آن در محیط عملیاتی به صورت خودکار نیست.

استقرار پیوسته<sup>4</sup>، علاوه بر مراحل فوق، نسخه‌ی جدید منتشر شده از برنامه را در محیط عملیاتی اصلی به طور خودکار منتشر می‌کند. در این روش نیاز به سیستم‌های نظارتی بسیار محسوس است تا در صورت بروز مشکل، تیم از آن مطلع شود. کوپرنیتیس نیز ابزاری برای مدیریت سیستم‌های کانتینری در بستر ابری است که مدیریت، آپدیت دادن، مقیاس‌پذیری و ... را آسان و سریع می‌کند. البته همانطور که در کلاس درس نیز گفته شده، مفاهیم CI/CD مستقل از داکر و کوپرنیتیس تعریف شده‌اند اما کاربردهای این دو ابزار پیاده‌سازی مفاهیم CD/CI را بسیار راحت و مقیاس‌پذیر کرده‌اند.

---

<sup>4</sup> Continuous Deployment

### راه‌اندازی CI

در این بخش شما باید در محیط مخزن خود، روال CI را که در بخش مقدمه توضیح داده شد پیاده‌سازی کنید. با توجه به اینکه شما از تکنولوژی داکر استفاده می‌کنید، تنها کافی است که پس از هر تغییر در کد روی شاخه‌ی مستر، از روی داکر فایل<sup>5</sup> پروژه، یک داکر ایمج<sup>6</sup> جدید بسازد. سپس آن را در یک رجیستری پوش کند تا در بخش‌های بعدی از آن استفاده کنید. برای این کار می‌توانید از [Quay.io](https://quay.io) یا رجیستری‌های دیگر استفاده کنید. برای این قسمت شما باید فایل `gitlab-ci.yml` را که در پروژه‌های شما اضافه شده تحویل دهید. پیشنهاد ما برای پیاده‌سازی CI استفاده از ابزار گیت‌لب است اما اگر از ابزارهای دیگری برای این قسمت استفاده کنید مشکلی ندارد.

### راه‌اندازی پروژه در محیط کوبرنتیس

در صورت تمایل به دست‌یابی به اطلاعاتی بیش از آنچه درباره کوبرنتیس در درس گفته شده می‌توانید این [لینک](#) را مطالعه کنید و این [ویدیو](#) را ببینید. ابتدا مطمئن شوید که به مفاهیم Namespace، Pod، Deployment و Service در مورد کوبرنتیس مسلط هستید (می‌توانید از این [لینک](#) استفاده کنید). برای شروع کار با کلاستران<sup>7</sup> باید کلاينت کوبرنتیس (kubectl) را با تنظیماتی که در اختیارتان قرار داده می‌شود راه‌اندازی کنید (برای دریافت فایل کانفیگ گروه خود به [aminarefzadeh1376@gmail.com](mailto:aminarefzadeh1376@gmail.com) ایمیل بزنید).

بعد از تنظیم کردن کلاينت، به namespace خودتان برای ایجاد و تغییر منابع دسترسی خواهید داشت. سپس باید سه Deployment بسازید. یکی برای بک‌اند (دو Pod) و یکی برای فرانت‌اند (یک Pod) و یکی برای دیتابیس (یک Pod). برای اینکه پادهای بک‌اند و فرانت‌اند شما بالا بیايند باید کلاستر کوبرنتیس بتواند ایمج شما را از رجیستری‌ای که ایمج پروژه را در بخش قبل در آن پوش کردید دریافت کند.

دقت کنید که با هر بار ریستارت شدن پاد دیتابیس، داده‌های ذخیره شده در آن نباید از بین بروند (برای این کار مفاهیم volume persistent و claim volume persistent را مطالعه کنید).

<sup>5</sup> Dockerfile

<sup>6</sup> Docker Image

<sup>7</sup> Cluster

قبل ساخته شده‌اند و شما فقط باید claim volume persistent بسازید و به دیپلویمنت دیتابیس متصل کنید). در این قسمت شما هر بار که می‌خواهید نسخه‌ی جدید کد را روی محیط کوپرنیتیس قرار دهید، باید نسخه‌ی ایمج مربوطه را به صورت دستی در Deployment تغییر دهید. دقت کنید که بخش‌های داخلی کوپرنیتیس می‌توانند در شبکه‌ی داخلی همدیگر را ببینند اما برای دسترسی به آنها از اینترنت خارجی، باید در کوپرنیتیس یک Service از نوع NodePort ایجاد کنید که برای هر Pod روی پورتهای که تعیین می‌کنید، دسترسی خارجی ایجاد شود. در این قسمت شما حتما باید در هنگام ساختن یک دیپلویمنت برای استفاده از memory و cpu محدودیت تعیین کنید، در غیر این صورت پادهای شما ساخته نخواهند شد. برای این قسمت شما باید فایل خروجی با فرمت yml تمام ریسورس‌هایی که در کوپرنیتیس ساخته‌اید را در پروژه‌های بک‌اند و فرانت‌اند قرار دهید.

---

## راه‌اندازی CD (امتیازی)

در قسمت‌های قبل، شما باید برای آپدیت پروژه در محیط کوپرنیتیس، ایمج پروژه را به صورت دستی آپدیت می‌کردید. در این قسمت باید به CI پروژه، یک stage جدید اضافه کنید تا بعد از هربار پوش کردن روی شاخه‌ی مستر و موفقیت آمیز بودن Build پروژه، آخرین نسخه‌ی کد به طور اتوماتیک روی محیط پروداکشن قرار گیرد (به دلیل مشکلاتی که رانرهای گیت‌لب در ارتباط با سرورهای داخل ایران دارند، ممکن است در این بخش دچار مشکل شوید و وقت زیادی از شما بگیرد).

## نکات تکمیلی و راهنمایی

- پروژه‌های زیرساختی آمیخته با مشکلات فراوان و پیشبینی نشده است. در صورتیکه در هر یک از قسمت‌های پروژه بیش از حد گیر کردید و به بن بست رسیدید از دستیاران آموزشی کمک بگیرید.

---

## نکات پایانی

- کافی است که یکی از اعضای گروه Hash مربوط به آخرین کامیت پروژه سمت سرور و سمت کاربر را در سایت درس آپلود کند. در هنگام تحویل، پروژه روی این کامیت مورد ارزیابی قرار می‌گیرد.
- هدف این تمرین یادگیری شماسست. لطفاً تمرین را خودتان انجام دهید. در صورت مشاهده‌ی مشابهت بین کدهای دو گروه، از نمره هر دو گروه مطابق سیاستی که در کلاس گفته شده است کسر خواهد شد.
- سوالات خود را تا حد ممکن در فروم درس مطرح کنید تا سایر دانشجویان نیز از پاسخ آنها بهره‌مند شوند. در صورتی که قصد مطرح کردن سوال خاص‌تری داشتید، از طریق ایمیل با طراحان این فاز پروژه ارتباط برقرار کنید. توجه داشته باشید که دیگر شبکه‌های اجتماعی مانند تلگرام راه ارتباطی رسمی با دستیاران آموزشی نیست و دستیاران آموزشی موظف به پاسخگویی در محیط‌های غیررسمی نیستند.
- ایمیل طراحان پروژه:

[hoseininejad1999@gmail.com](mailto:hoseininejad1999@gmail.com)

[aminarefzadeh1376@gmail.com](mailto:aminarefzadeh1376@gmail.com)

[ae.561999@gmail.com](mailto:ae.561999@gmail.com)