

UNIVERSIDAD DE SANTIAGO DE
COMPOSTELA



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA

Ciclo completo de CI/CD con Dagger y Kubernetes

Autor/a:

Daniel Vieites Torres

Tutores:

Pichel

Francisco Maseda Muiño

Grado en Ingeniería Informática

2025

Trabajo de Fin de Grado presentado en la Escuela Técnica Superior de
Ingeniería de la Universidad de Santiago de Compostela para la obtención do
Grado en Ingeniería Informática

Resumen

Este trabajo aborda la gestión completa de un ciclo CI/CD (*Continuous Integration/Continuous Delivery*) con Dagger. Se desarrolla una aplicación de prueba, que consta de un frontend y un backend, junto con la infraestructura como código, y se comparan dos *pipelines*: uno implementado con Dagger y otro sin él. Los resultados demuestran que Dagger mejora el flujo de trabajo debido a la gestión que realiza de la caché, ejecutando todo el ciclo de CI/CD un 80 % más rápido que el *pipeline* sin Dagger. Como resultado, este trabajo propone un conjunto de módulos de Dagger que ofrece un enfoque práctico sobre cómo utilizar Dagger para acelerar el desarrollo despliegue de cualquier aplicación, minimizando el tiempo de espera y pasos manuales.

Índice general

1. Introducción	1
2. Especificación de Requisitos	3
3. Deseño	5
4. Probas	7
5. Exemplos (eliminar capítulo na versión final)	9
5.1. Un exemplo de sección	9
5.1.1. Un exemplo de subsección	9
5.1.2. Otro exemplo de subsección	9
5.2. Exemplos de figuras e cadros	10
5.3. Exemplos de referencias á bibliografía	10
5.4. Exemplos de enumeracións	11
6. Conclusións e posibles ampliacións	13
A. Manuais técnicos	15
B. Manuais de usuario	17
C. Licenza	19
Bibliografía	21

Índice de figuras

5.1. Esta é a figura de tal e tal.	10
--	----

Índice de cuadros

5.1. Esta é a táboa de tal e cal.	10
---	----

Capítulo 1

Introdución

Obxectivos Xerais, Relación da Documentación que conforma a Memoria, Descrición do Sistema, Información Adicional de Interese (métodos, técnicas ou arquitecturas utilizadas, xustificación da súa elección, etc.).

Capítulo 2

Especificación de Requisitos

Especificación dos requisitos máis relevantes do Sistema, xunto coa información que este debe almacenar e as interfaces con outros Sistemas, sexan hardware ou software, e outros requisitos (rendemento, seguridade, etc.).

Capítulo 3

Deseño

Debe describirse como se realiza o Sistema, a división deste en diferentes compoñentes e a comunicación entre eles. Así mesmo, determinarase o equipamento hardware e software necesario, xustificando a súa elección no caso de que non fose un requisito previo. Debe achegarse a un nivel suficiente de detalle que permita comprender a totalidade da estrutura do produto desenvolvido, utilizando no posible representacións gráficas.

Capítulo 4

Probas

Plan de probas (con evidencias) que verifica a funcionalidade e correctitude global do sistema, e se leva a cabo.

Capítulo 5

Exemplos (eliminar capítulo na versión final)

5.1. Un exemplo de sección

Esta é *letra cursiva*, esta é **letra negrilla**, esta é letra subrallada, e esta é **letra curier**. Letra tiny, scriptsize, small, large, Large, LARGE e moitas más. Exemplo de fórmula: $a = \int_0^\infty f(t)dt$. E agora unha ecuación aparte:

$$S = \sum_{i=0}^{N-1} a_i^2. \quad (5.1)$$

As ecuaciones se poden referenciar: ecuación (5.1).

5.1.1. Un exemplo de subsección

O texto vai aquí.

5.1.2. Otro exemplo de subsección

O texto vai aquí.

Un exemplo de subsubsección

O texto vai aquí.

Un exemplo de subsubsección

O texto vai aquí.

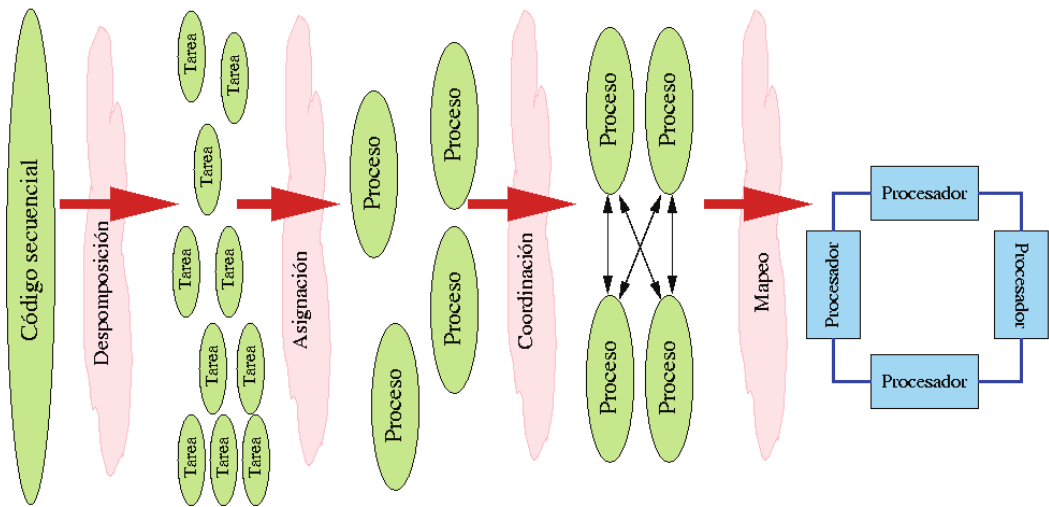


Figura 5.1: Esta é a figura de tal e cal.

Izquierda	Derecha	Centrado
ll	r	cccc
llll	rrr	c

Cuadro 5.1: Esta é a táboa de tal e cal.

Un exemplo de subsubsección

O texto vai aquí.

5.2. Exemplos de figuras e cadros

A figura número 5.1.
O cadro (taboa) número 5.1.

5.3. Exemplos de referencias á bibliografía

Este é un exemplo de referencia a un documento descargado da web [1]. E este é un exemplo de referencia a unha páxina da wikipedia [2]. Agora un libro [3] e agora unha referencia a un artigo dunha revista [4]. Tamén se poden por varias referencias á vez [1, 3].

5.4. Ejemplos de enumeraciones

Con puntos:

- Un.
- Dous.
- Tres.

Con números:

1. Catro.
2. Cinco.
3. Seis.

Exemplo de texto verbatim:

```
0 texto          verbatim
  se visualiza tal
    como se escribe
```

Exemplo de código C:

```
#include <math.h>
main()
{  int i, j, a[10];
   for(i=0;i<=10;i++) a[i]=i; // comentario 1
   if(a[1]==0) j=1; /* comentario 2 */
   else j=2;
}
```

Exemplo de código Java:

```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!"); // Display the string.
    }
}
```


Capítulo 6

Conclusións e posibles ampliacións

O traballo describe o grao de cumprimento dos obxectivos. Posibles vías de mellora.

Apéndice A

Manuais técnicos

En función do tipo de Traballo e metodoloxía empregada, o contido poderase dividir en varios documentos. En todo caso, neles incluírase toda a información precisa para aquelas persoas que se vaian encargar do desenvolvemento e/ou modificación do Sistema (por exemplo código fonte, recursos necesarios, operacións necesarias para modificacións e probas, posibles problemas, etc.). O código fonte poderase entregar en soporte informático en formatos PDF ou postscript.

Apéndice B

Manuais de usuario

Incluirán toda a información precisa para aquelas persoas que utilicen o Sistema: instalación, utilización, configuración, mensaxes de erro, etc. A documentación do usuario debe ser autocontida, é dicir, para o seu entendemento o usuario final non debe precisar da lectura doutro manual técnico.

Apéndice C

Licenza

Se se quere pór unha licenza (GNU GPL, Creative Commons, etc), o texto da licenza vai aquí.

Bibliografía

- [1] Nvidia CUDA programming guide. Versión 2.0, 2010. Disponible en <http://www.nvidia.com>.
- [2] Acceso múltiple por división de código. Artigo da wikipedia (<http://es.wikipedia.org>). Consultado o 2 de xaneiro do 2010.
- [3] R.C. Gonzalez e R.E. Woods, *Digital image processing*, 3ª edición, Prentice Hall, New York, 2007.
- [4] P. González, J.C. Cartex e T.F. Pellas, “Parallel computation of wavelet transforms using the lifting scheme”, *Journal of Supercomputing*, vol. 18, no. 4, pp. 141-152, junio 2001.