



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA



Machine Learning Operations (MLOps) Clase 1

Leticia Rodríguez

Septiembre 2024 - 2do Cuatrimestre - 4to. Bimestre

Universidad de Buenos Aires - FCEyN - Departamento de Computación

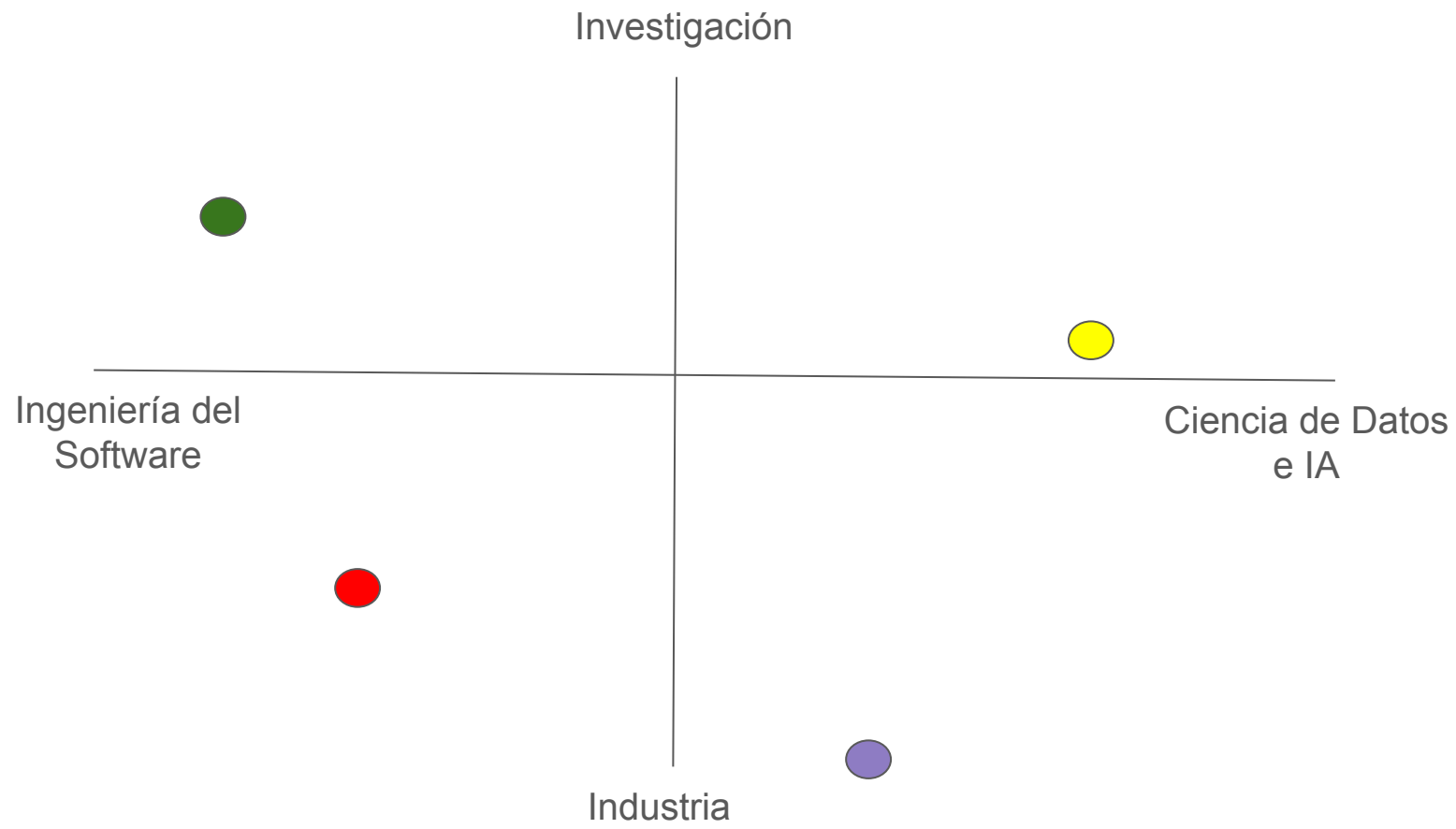
Sobre la materia

- La materia propone una mirada sobre la automatización de la construcción y monitoreo de modelos de Inteligencia Artificial en el contexto de la investigación y del desarrollo de software
- 7 clases: 5 teórica-prácticas y 2 evaluación. Asistencia requerida 80%.
- Aprobación:
 - **Promocionable**
 - Evaluación es un trabajo en clase - 11 de Noviembre
 - Recuperatorio: coloquio - 25 de Noviembre
- Metodología de trabajo:
 - teórico-práctico con actividades en el aula que habilitan a la reflexión alrededor de cada uno de los temas.
 - es un espacio para compatir las experiencias profesionales, acompañadas por reflexión y pensamiento de oportunidades futuras.
 - la participación en kahoots y las actividades grupales suma en la nota final
- Material en inglés y español
- Doctorado: 1.5 puntos - Optativa Cs. Computacion y Cs. Datos: 2 puntos

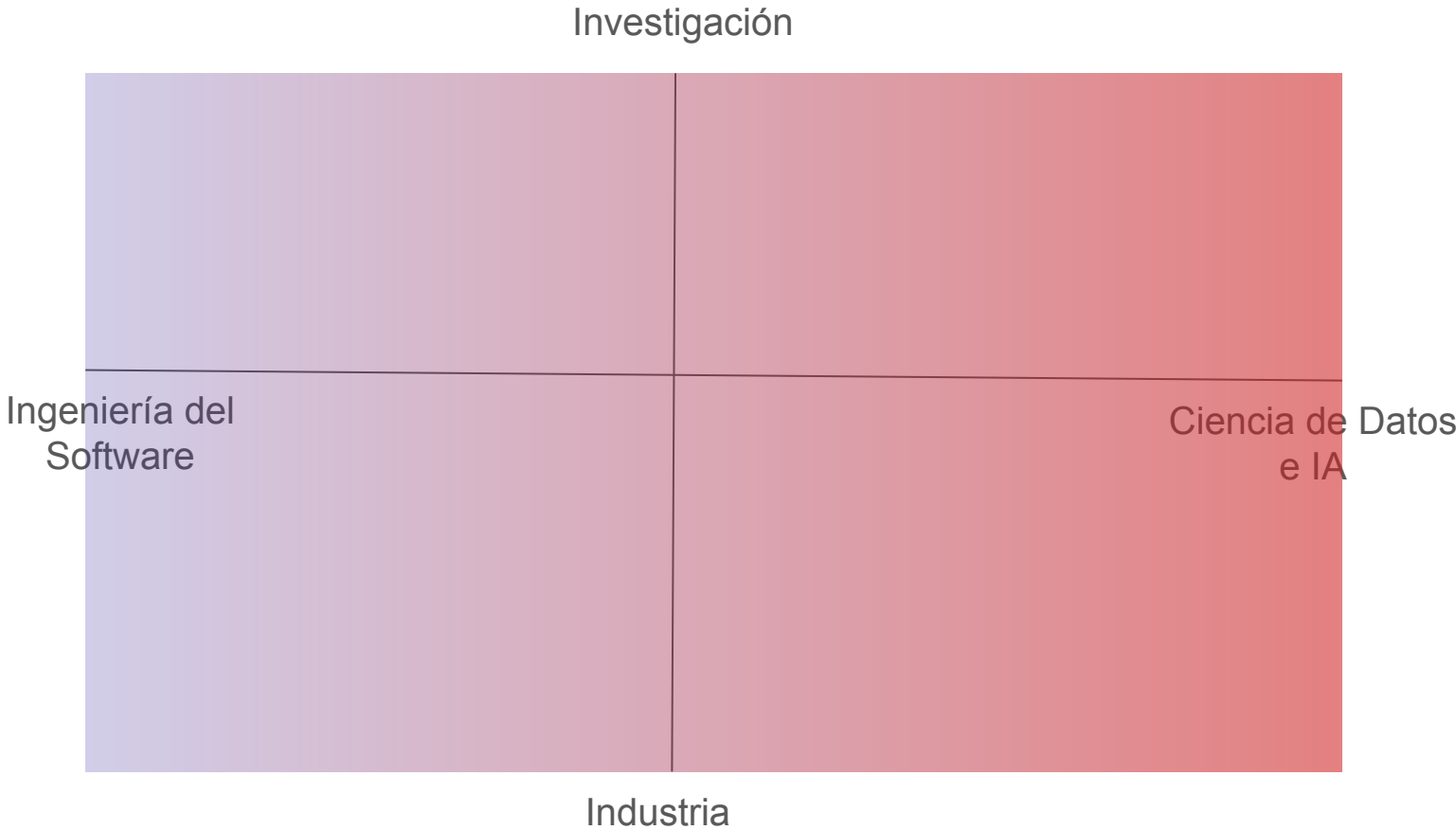
Cronograma

- 7/10 - Clase 1 - Introducción: ¿Qué es MLOps?
 - 14/10 - Clase 2 - MLOps Pipeline: Alcance del proyecto y procesamiento de Datos
 - 21/10 - Clase 3 - MLOps Pipeline: Procesamiento de Datos
 - 28/10 - Clase 4 - MLOps Pipeline: Modelado
 - 4/11 - Clase 5 - MLOps Pipeline: Deploy
 - 11/11 - Evaluación - Trabajo en Clase
 - 18/11 - Feriado
 - 25/11 - Recuperatorio de Evaluación (modalidad a definir)
- Clase 6 - LLMOps - Ejemplos en Kubeflow - Cloud - Cierre de la materia

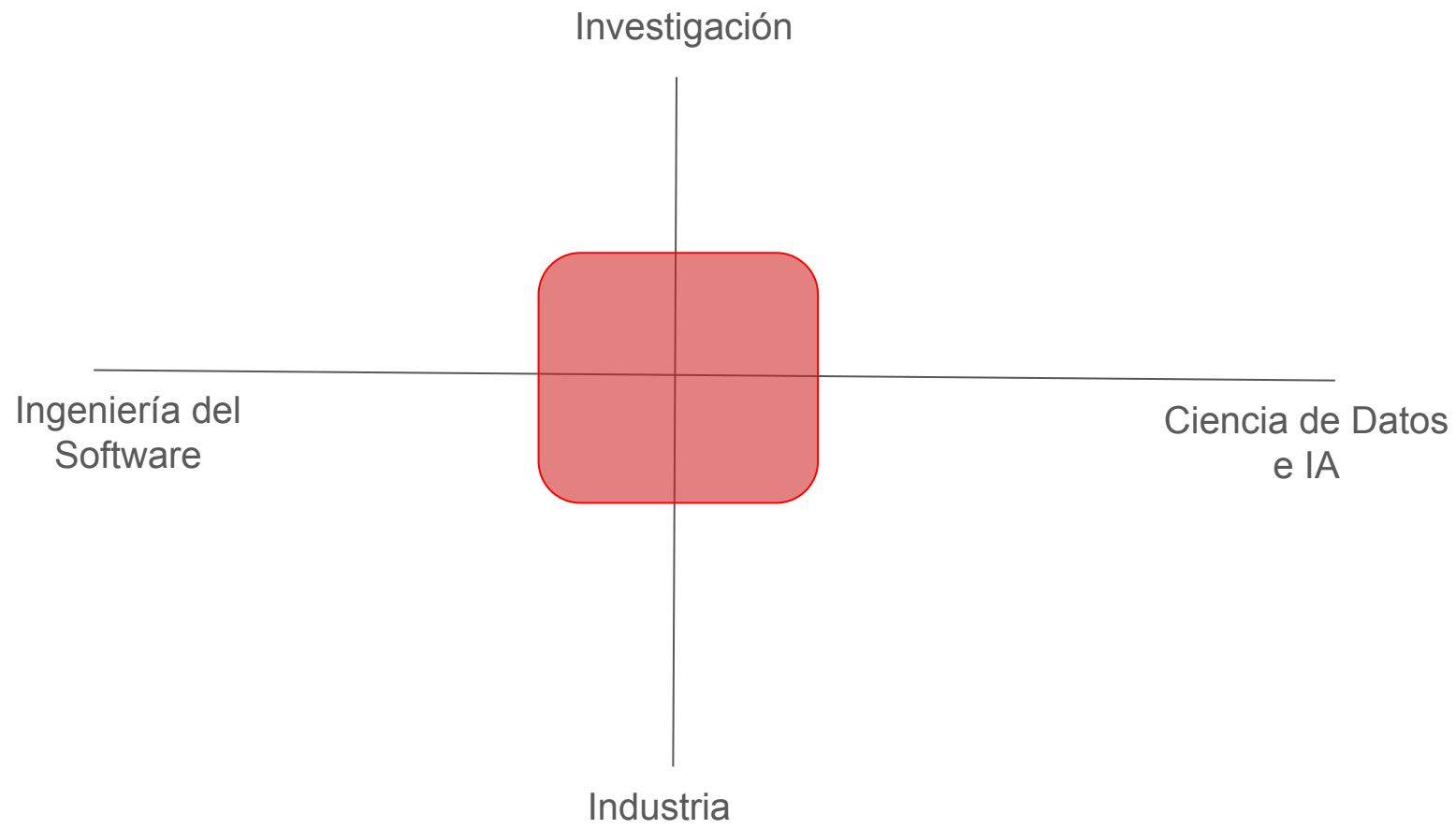
Investigación vs. Industria - Cs. de Datos vs. Ing. del Software



Investigación vs. Industria - Cs. de Datos vs. Ing. del Software



MLOps en todas sus dimensiones



Investigación vs. Industria - Cs. de Datos vs. Ing. del Software



Encuesta: Experiencias en Computación, Ingeniería del Software y Ciencia de Datos

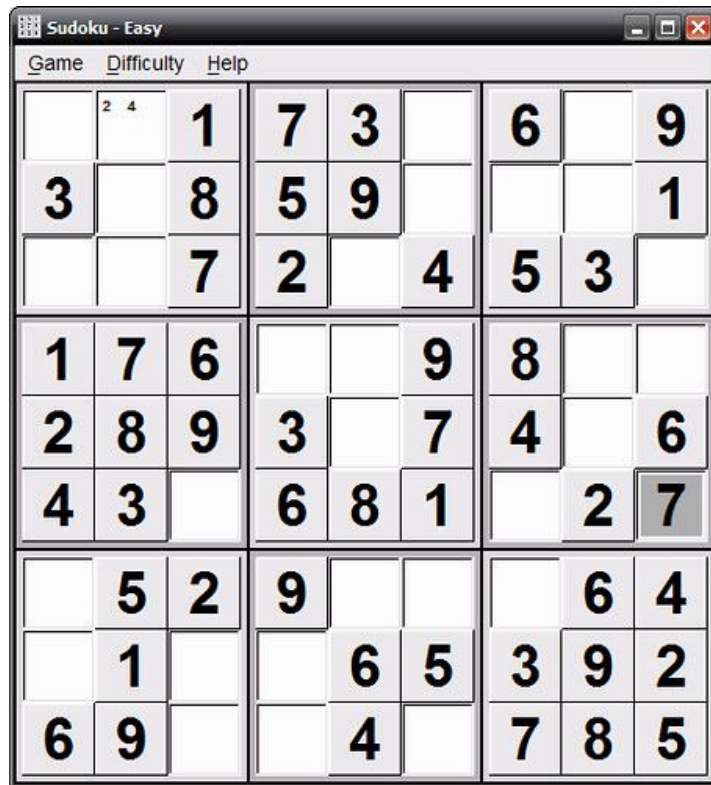
En el campus!
Sección material del curso



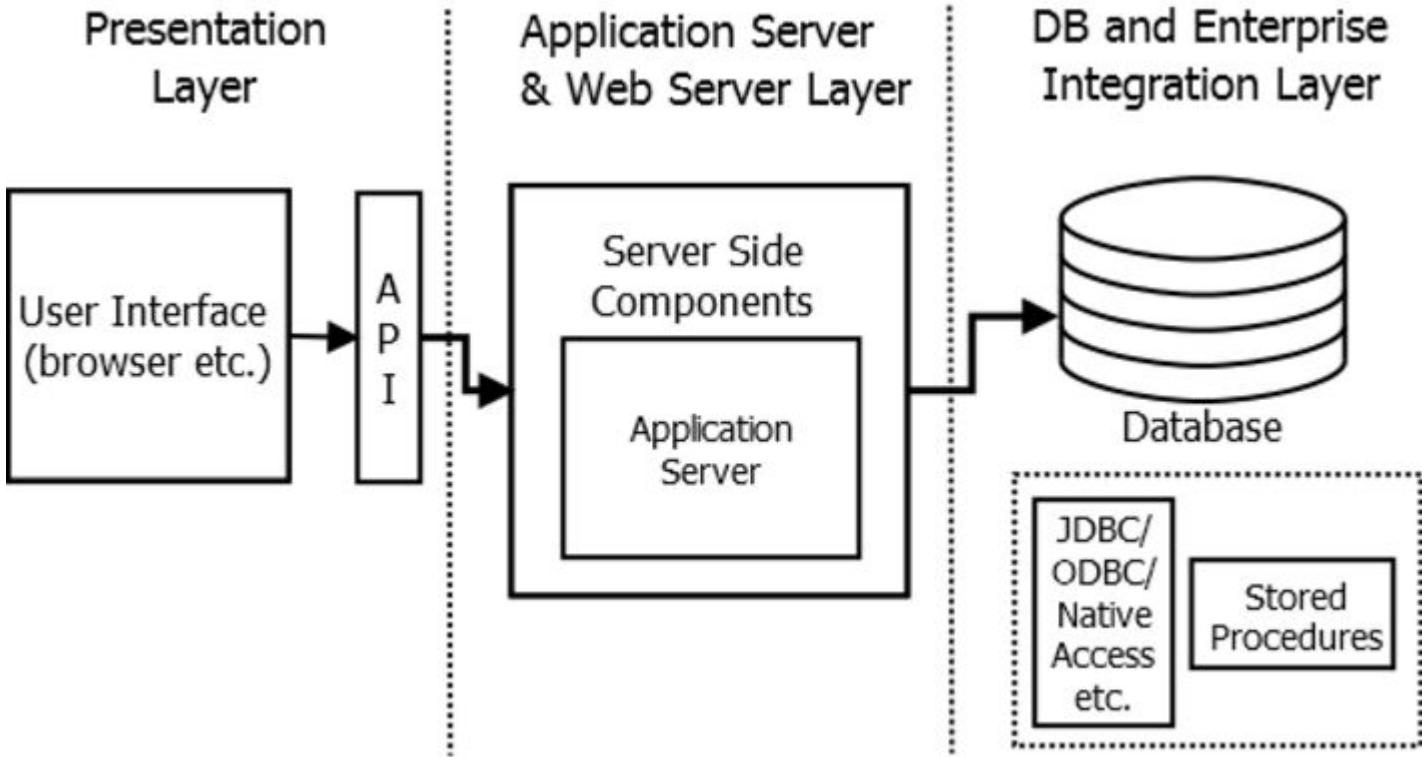
Desarrollo de Software

Code Blame 66 lines (52 loc) · 1.95 KB

```
1 import sys, os, random, pygame
2 sys.path.append(os.path.join("objects"))
3 import SudokuSquare
4 from GameResources import *
5
6 digits = '123456789'
7 rows = 'ABCDEFGH'I'
8
9
10 def play(values_list):
11     pygame.init()
12
13
14     size = width, height = 700, 700
15     screen = pygame.display.set_mode(size)
16
17     background_image = pygame.image.load("./images/sudoku-board-bare.jpg").convert()
18
19     clock = pygame.time.Clock()
20
21     # The puzzleNumber sets a seed so either generate
22     # a random number to fill in here or accept user
23     # input for a duplicatable puzzle.
24
25     for values in values_list:
26         pygame.event.pump()
27         theSquares = []
28         initXLoc = 0
29         initYLoc = 0
30         startX, startY, editable, number = 0, 0, "N", 0
31         for y in range(9):
32             for x in range(9):
33                 if x in (0, 1, 2): startX = (x * 57) + 38
34                 if x in (3, 4, 5): startX = (x * 57) + 99
35                 if x in (6, 7, 8): startX = (x * 57) + 159
36
37                 if y in (0, 1, 2): startY = (y * 57) + 35
38                 if y in (3, 4, 5): startY = (y * 57) + 100
39                 if y in (6, 7, 8): startY = (y * 57) + 165
40                 col = digits[x]
41                 row = rows[y]
42                 string_number = values[row + col]
43                 if len(string_number) > 1 or string_number == '' or string_number == '.':
44                     number = None
45                 else:
46                     number = int(string_number)
47             theSquares.append(SudokuSquare.SudokuSquare(number, startX, startY, editable, x, y))
```



Arquitectura Web



Un ejemplo práctico de la industria

Testimonio de Gabriel, trabajaba en MercadoLibre escribía:

- El equipo de **Infraestructura & Ingeniería (50 personas)** es una mezcla entre Arquitectura, Desarrollo, Infraestructura y Networking.
- Proyectos:
 - **Fury:** esta plataforma le permite a todo **desarrollo crear, desarrollar y mantener sus aplicaciones en producción**. Con más de **1000 aplicaciones corriendo** y más de **1000 deploys por día**, esta plataforma se encarga de la gestión tanto de los deploys, como de los sistemas de alarmas, logs, troubleshooting y la integración de las aplicaciones con el resto de los servicios (bases de datos, key value store, cache etc). Esta plataforma está construida íntegramente in-house utilizando, Docker y AWS (si querés conocer mas: <http://bit.ly/2vBYd97>)
 - **BigQ:** éste es el bus de comunicaciones de Mercadolibre para la difusión de novedades entre las distintas aplicaciones de Meli de forma asincrónica. Este servicio está construido sobre pulsar (<https://github.com/apache/incubator-pulsar>) siendo de los primeros en usarlo, hoy uno de los líderes de BigQ también es committer de pulsar. Por esta plataforma hoy se envían **mas de mil millones de mensajes por día (unos 12000 por segundo)**
 - **Document Search:** en pos de simplificar y robustecer las búsquedas de las distintas aplicaciones de mercadolibre estamos en el medio de la construcción de un “buscador as a service”, montado sobre elasticsearch. Este servicio le agrega la posibilidad de replicación, almacenamiento sobre múltiples índices en paralelo, monitoreo y control, backups y por sobre todas las cosas brindar un servicio de busquedas 100% administrado en el cual el usuario final del servicio no tenga que preocuparse por crear y mantener clusters de elastic (tarea no menor). Hoy en este servicio **se crean más de 200 millones de documentos nuevos por día y se realizan más de 1000 millones búsquedas por día**

<https://www.linkedin.com/pulse/infraestructura-ingenier%C3%ADa-meli-gabriel-eisbruch>

¿Qué desafíos tiene el desarrollo y mantenimiento del Software a lo largo del tiempo?

- Crecimiento en la cantidad de usuarios finales
- Más código, distintas herramientas y software (más desarrollo)
- Más datos
- Más pedidos (requests web)
- Más requerimientos y modificaciones
- Más personal (desarrolladores, lideres, etc)
- Infraestructura y actualizaciones
- Mayores riesgos
- Requerimientos de Seguridad
- Requerimientos de Privacidad de Datos
- Requerimientos de Disponibilidad (reliabilidad, SLA)
- Monitoreo / Logs
- Costos
- Otros

Los estándares: Ciclo de Desarrollo del Software



Los estándares: roles y manejo del Proyecto



Product Owner
PO



Arquitecto



QA
Testeo



Project Manager
PM



Desarrolladores
Front-end



UX
User experience



Stakeholders



Desarrolladores
Backend



DevOps



Líderes



Científicos de
Datos



Ingenieros de
Machine Learning



Managers



MLOps

Los estándares: Planificación - Kanban

The screenshot shows a Jira Kanban board for the 'Teams in Space' project. The board is organized into four columns: TO DO, IN PROGRESS, CODE REVIEW, and DONE. Each column contains several tasks, each with a title, a team label, a status icon, a count, a due date, and an assignee.

TO DO 5

- Engage Jupiter Express for outer solar system travel
SPACE TRAVEL PARTNERS
5 TIS-25
- Create 90 day plans for all departments in the Mars Office
LOCAL MARS OFFICE
9 TIS-12
- Engage Saturn's Rings Resort as a preferred provider
SPACE TRAVEL PARTNERS
3 TIS-17
- Enable Speedy SpaceCraft as the preferred
TIS-17

IN PROGRESS 5

- Requesting available flights is now taking > 5 seconds
SEESPACEEZ PLUS
3 TIS-8
- Engage Saturn Shuttle Lines for group tours
SPACE TRAVEL PARTNERS
4 TIS-15
- Establish a catering vendor to provide meal service
LOCAL MARS OFFICE
4 TIS-15
- Engage Saturn Shuttle Lines for group tours
SPACE TRAVEL PARTNERS
TIS-15

CODE REVIEW 2

- Register with the Mars Ministry of Revenue
LOCAL MARS OFFICE
3 TIS-11
- Draft network plan for Mars Office
LOCAL MARS OFFICE
3 TIS-15

DONE 8

- Homepage footer uses an inline style - should use a class
LARGE TEAM SUPPORT
TIS-68
- Engage JetShuttle SpaceWays for travel
SPACE TRAVEL PARTNERS
5 TIS-23
- Engage Saturn Shuttle Lines for group tours
SPACE TRAVEL PARTNERS
TIS-15
- Establish a catering vendor to provide meal service
LOCAL MARS OFFICE
TIS-15

Los estándares: Planificación - Tareas por sprint

JIRA

Dashboards ▾ Projects ▾ Issues ▾ Agile ▾ Service Desk ▾ Capture ▾ Portfolio ▾ Create

Search 🔍 ? ⚙️ 👤

Scrum: Teams in Space

Plan Work Report Board ↕

Q QUICK FILTERS: Product UI Server Only My Issues Recently Updated

Versions

EPICS

All issues

SeeSpaceEZ Plus ▾

Space Travel Partners ▾

Summer Saturn Sale ▾

Afterburner Plus ▾

Large Team Support ▾

Local Mars Office ▾

Hyper-speed shuttles ▾

New launch platforms ▾

Sprint 6 24 issues

24/Nov/14 11:05 PM • 08/Dec/14 11:05 PM

Linked pages

TIS-46 Update LocalTransportContrc [3.0] Large Team Support 6

TIS-42 Extend booking experience in [3.0] Large Team Support 9

TIS-43 Extend booking experience in [3.0] Large Team Support 9

TIS-40 Update FlightController to har [3.0] Large Team Support 6

TIS-44 Reward Customers an extra £ [3.0] Large Team Support 3

TIS-41 Update LodgingController to f [3.0] Large Team Support 6

TIS-39 Update UI controls on travel t [3.0] Large Team Support 8

TIS-45 Email non registered users to [3.0] Large Team Support 2

TIS-49 Draft network plan for Mars O [2.1] Local Mars Office 5

TIS-68 Homepage footer uses an inltyle - should use a class

Teams in Space / TIS-56

Add pointer to main css file to instruct users to create child themes

Details

Status: CLOSED

Component/s: Web Site

Labels: None

Affects Version/s: 2.0

Fix Version/s: 2.0

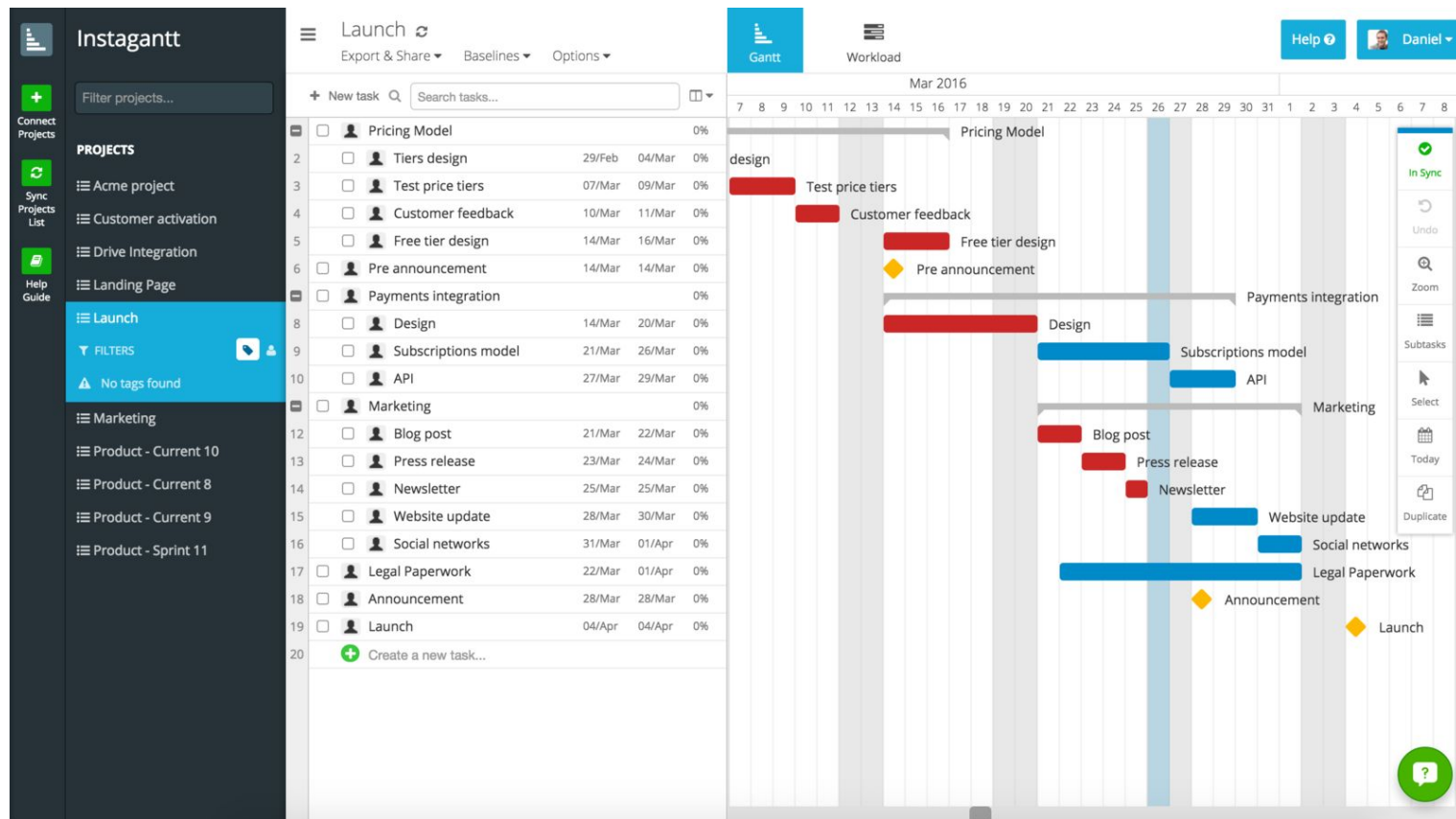
Epic: Large Team Support x

People

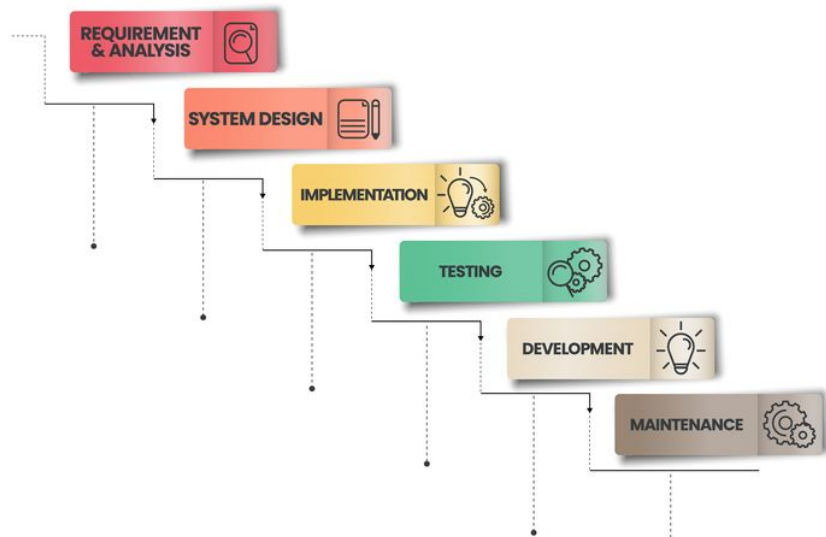
Reporter: Jennifer Evans

Assignee: Jennifer Evans

Los estándares: Planificación - GANTT

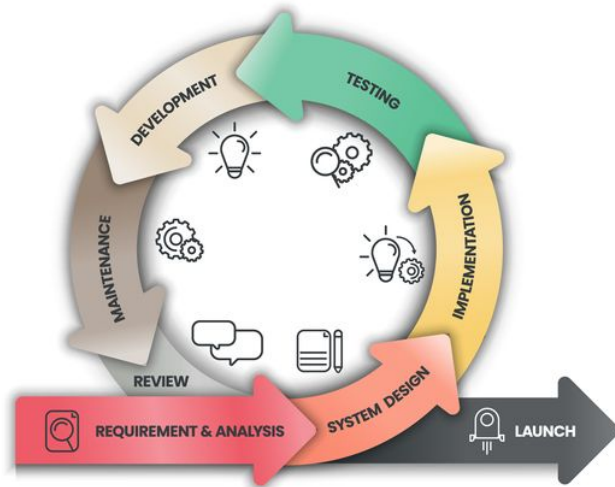


Los estándares: Metodologías del desarrollo del Software



WATERFALL
MODEL

VS



AGILE
MODEL

Los estándares: Scrum

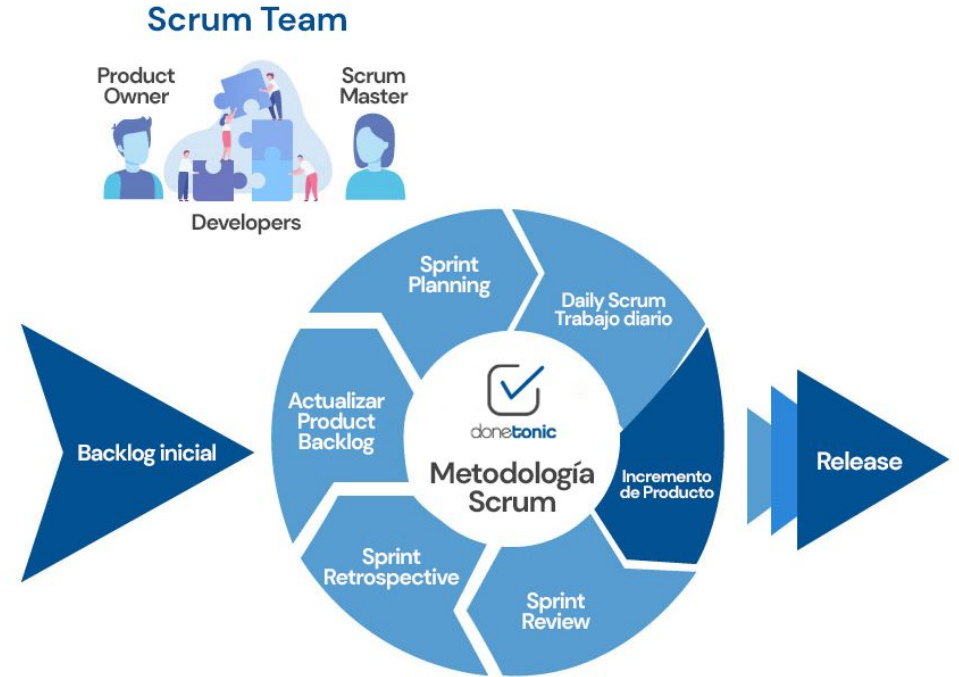
Metodología de trabajo Ágil

Permite organizar el equipo en roles

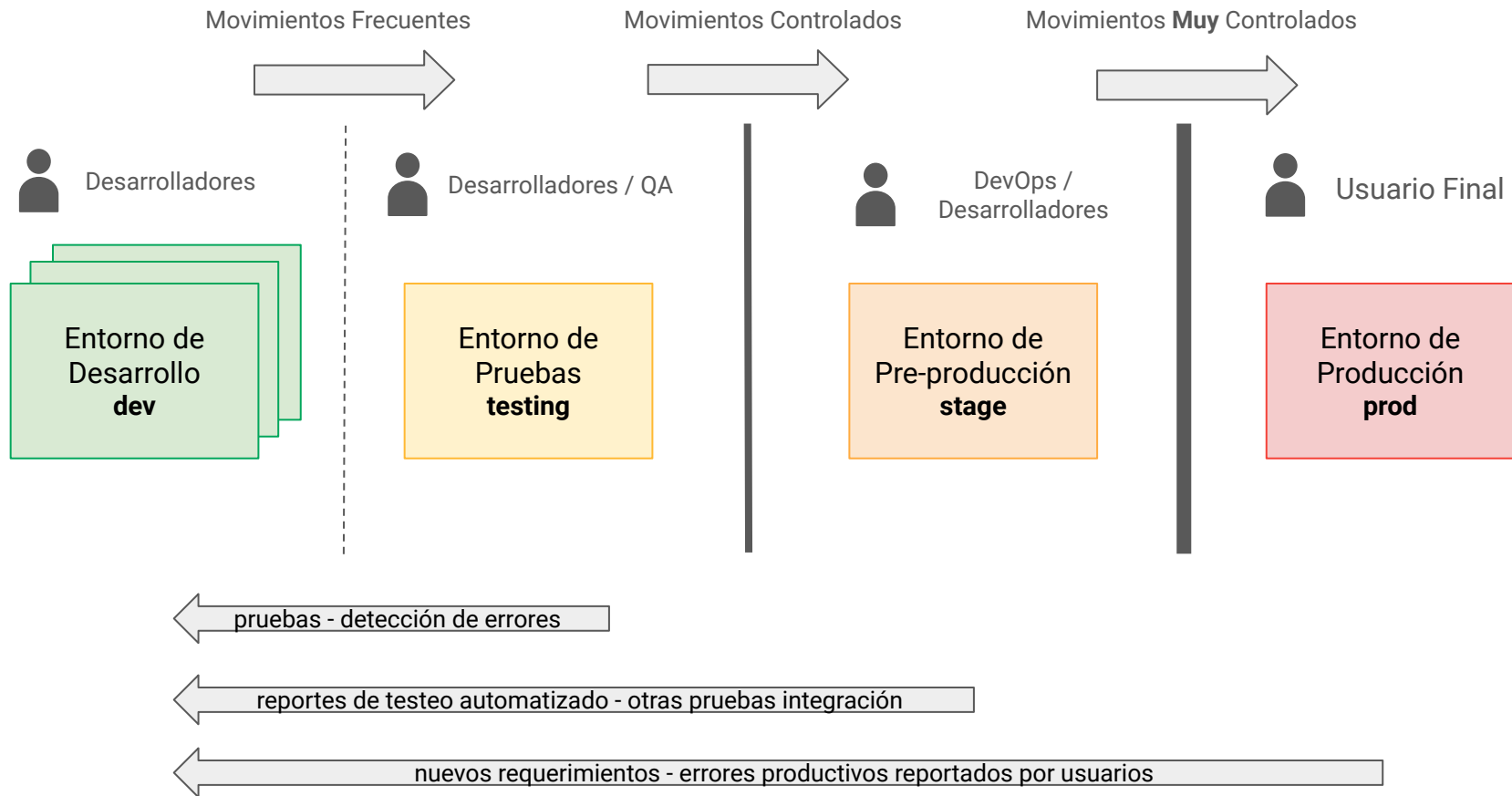
La descripción de tareas en backlogs y herramientas.

Organiza el tiempo del proyectos en sprints (periodos de 2 o 3 semanas)

Define las reuniones del equipo: diaria o stand-up (15 mins), planning por sprint y retrospectiva del sprint

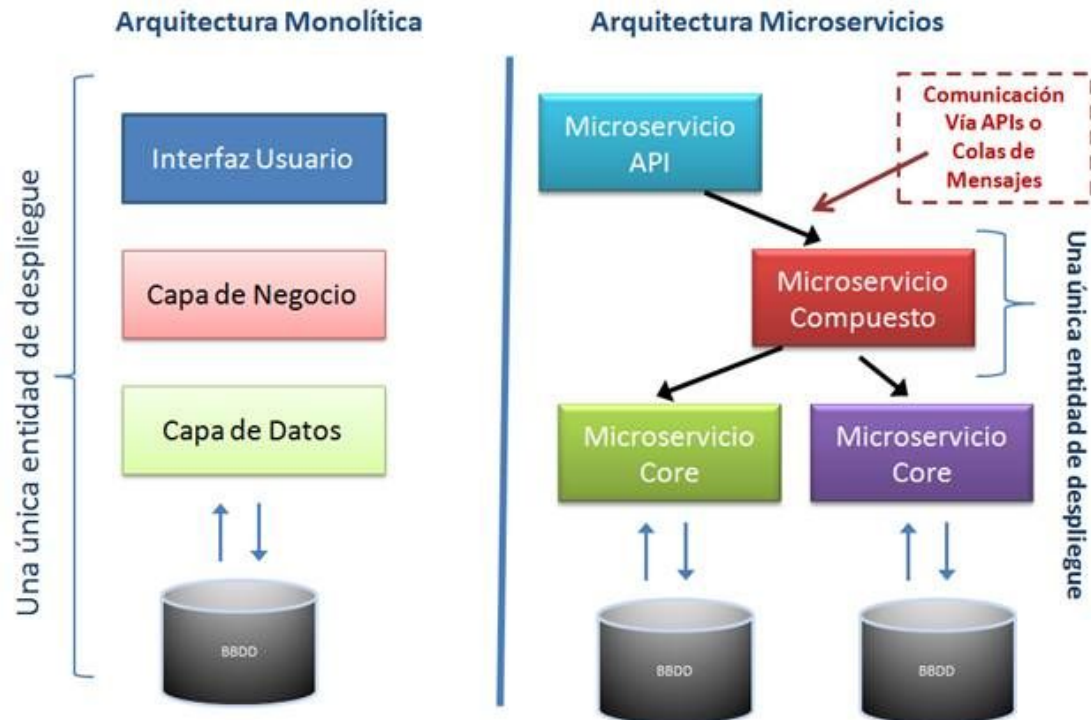


Los estándares: Ambientes del desarrollo de Software



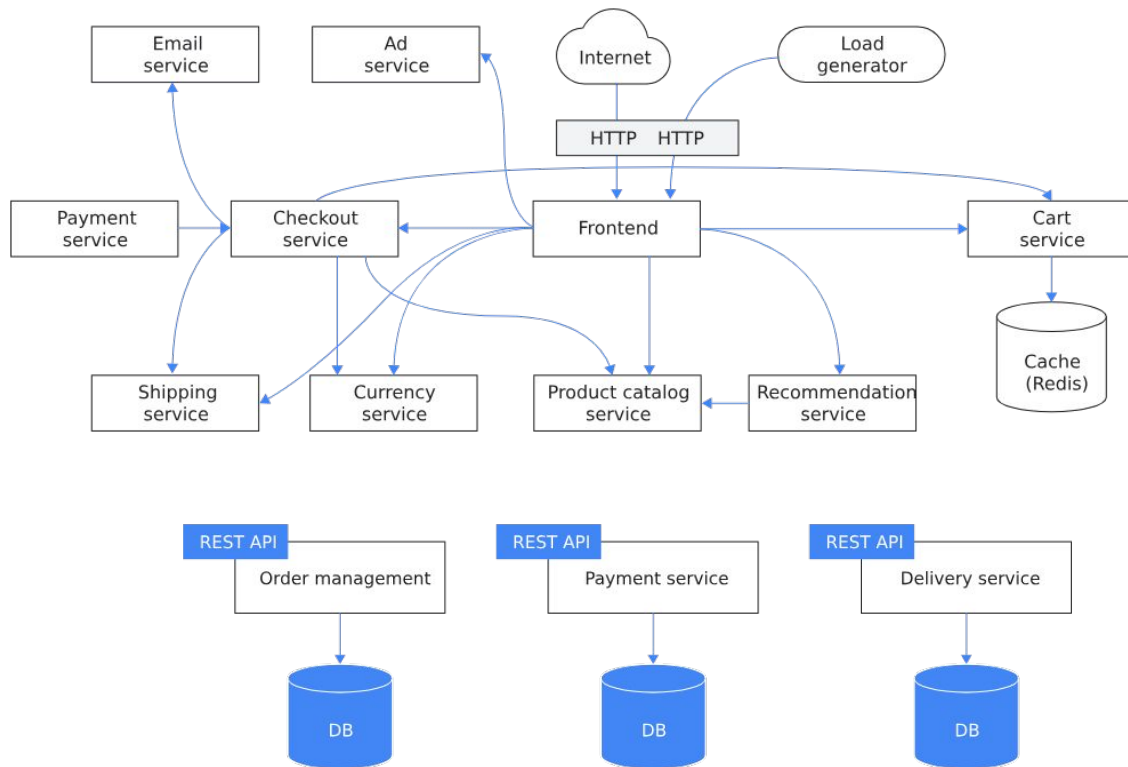
Los estándares: Diseño y Arquitectura

Arquitectura Monolítica vs. Microservicios



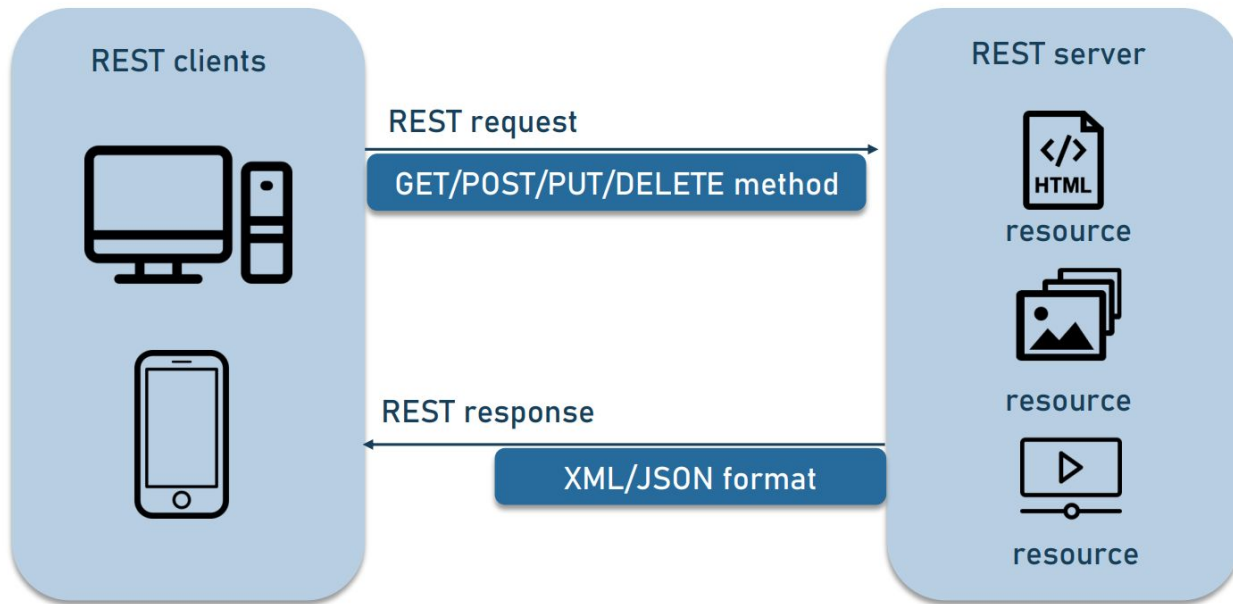
Los estándares: Diseño y Arquitectura

Arquitectura Microservicios Ejemplo



Los estándares: API REST

REST API IN ACTION



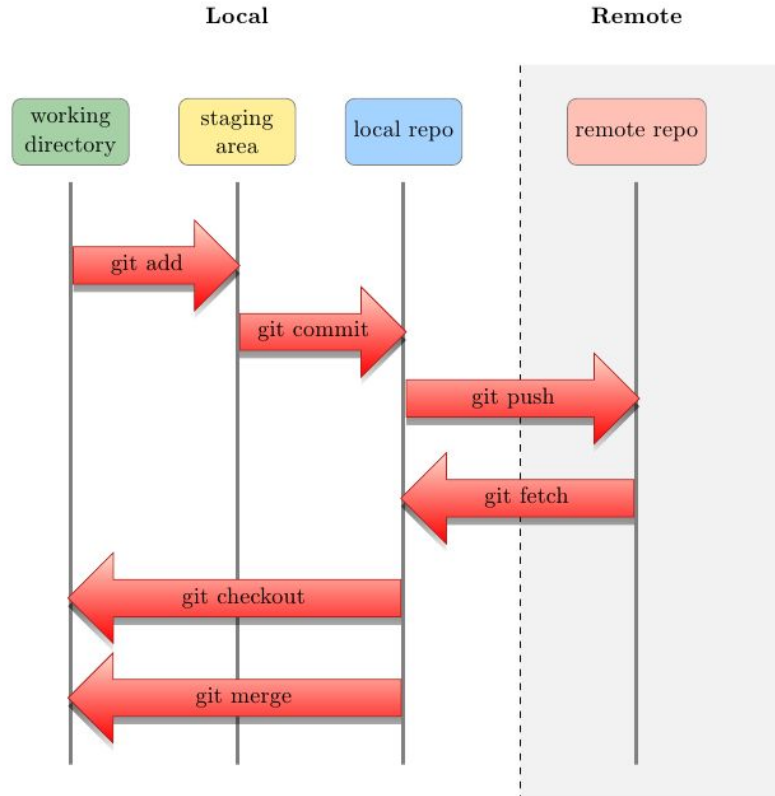
Los estándares: API REST

Endpoint	<code>https://apiurl.com/review/new</code>
HTTP Method	<code>POST</code>
HTTP Headers	<code>content-type: application/json</code> <code>accept: application/json</code> <code>authorization: Basic abase64string</code>
Body	<pre>{ "review" : { "title" : "Great article!", "description" : "So easy to follow.", "rating" : 5 } }</pre>

Métodos HTTP

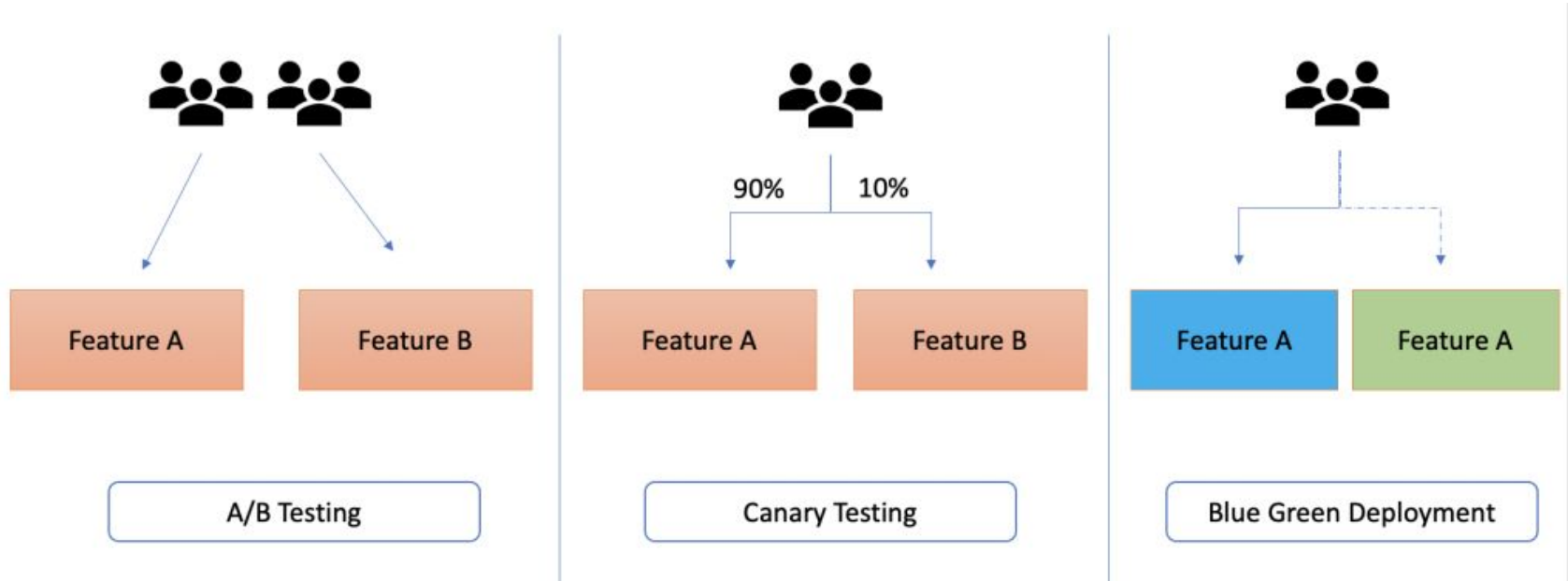
- POST - Crear
- UPDATE - Actualizar
- DELETE - Borrar
- GET - Obtener

Los estándares: Repositorios



- git / svn
- Permiten el versionado de código y el trabajo colaborativo.
- Facilitan CI/CD

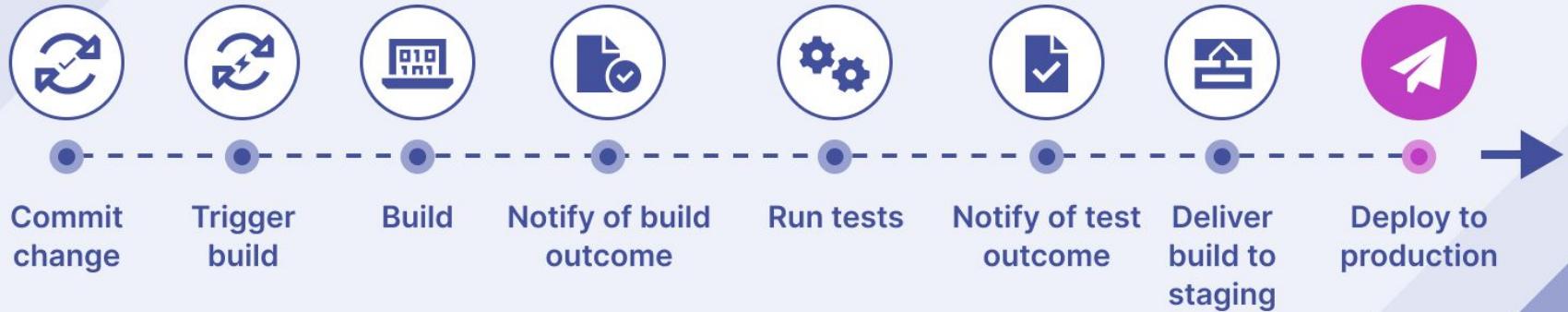
Los estándares: Metodologías para el encendido en producción



Deploy: Pasar el desarrollo de código y datos al entorno de producción. Esta operación se hace luego que esté completo y probado en el entorno de Testeo. Puede provenir de un entorno Stage.

CI /CD - Continuous integration - Continuos Deployment

CI/CD PIPELINE



DevOps

DevOp es un conjunto de prácticas que agrupan el desarrollo de software (*Dev*) y las operaciones de IT (*Ops*). Su objetivo es hacer más rápido el ciclo de vida del desarrollo de software y proporcionar una entrega continua de alta calidad. DevOps es una práctica complementaria al desarrollo de software ágil .

La principal característica del movimiento DevOps es defender activamente la automatización y el monitoreo en todos los pasos de la construcción del software, desde la integración, las pruebas, el despliegue, hasta la implementación y la administración de la infraestructura. DevOps apunta a ciclos de desarrollo más cortos, mayor frecuencia de implementación, lanzamientos más efectivos, en estrecha alineación con los objetivos comerciales.



<https://www.techtarget.com/searchitoperations/definition/DevOps>

Testeos

Testear es probar que el software funcione de la manera adecuada. Se realiza en diferentes entornos y en distintas etapas del proyecto. Es una tarea iterativa.

Puede haber un equipo especializado en realizar las pruebas y herramientas que permiten el trackeo de errores con flujos de comunicación entre los equipos de desarrollo y qa.

Generalmente, se acuerda un proceso y formas de integración en los distintos ambientes, tiempos de resolución de errores esperados (SLA), diferentes criticidad y prioridades de errores reportados, etc.

Los testeos que generalmente se realizan son:

- **unitarios** - testeo individuales de pequeñas porciones de código que generalmente son automatizados y escritos por el desarrollador
- **integración** - testeo que prueban más de una funcionalidad pueden estar hechos por el equipo de qa y los desarrolladores. También pueden estar automatizados.
- **aceptación** - son testeos E2E que performan los stakeholders o usuarios finales de sistema

Otros tipos son: pruebas de stress, pruebas de usabilidad, pruebas de performance, pruebas de seguridad, etc

Actividad: Ingeniería del Software en sus ambientes de trabajo y en la facultad

En grupos, cada integrante puede contar alguna anécdota o su experiencia en algunos de los siguientes puntos:

- Desarrollo de software en grupo de la facultad, en el laboratorio o equipo de trabajo en la industria. ¿Cuáles son los acuerdos del trabajo en equipo? ¿Cómo se distribuye el trabajo? ¿En qué ambientes se trabaja y cuáles son los entregables?
- ¿Han utilizado o interactuado con equipos que utilice alguna de las técnicas vistas anteriormente? Agile/Scrum, Waterfall, DevOps, distintos ambientes de desarrollo de software, arquitecturas y diseños de software de cualquier tipo. Comenten sus experiencias
- Relaten alguna experiencia significativa en la cual se hayan enfrentado a los problemas que surgen de la producción y mantenimiento de software.
- Si se dedican a la ciencia de datos o machine learning, relacionen los problemas y particulares anteriores en el contexto de la creación de sistemas que usen Inteligencia Artificial o el análisis de datos. ¿Qué herramientas usa? ¿Cómo se organiza el trabajo? ¿Cómo se generan los entregables?

Intercambien opiniones, puntos de vistas y relacionarlo con lo visto hasta ahora.

Break
15 minutos
y charlamos sobre ML

Roles de AI - Ciencia de Datos



Product Owner
PO



Arquitecto



QA
Testeo



Project Manager
PM



Desarrolladores
Front-end



UX
User experience



Stakeholders



Desarrolladores
Backend



DevOps



Líderes



Científicos de
Datos



Ingenieros de
Machine Learning




Managers



MLOps





Programación en ciencia de datos

 AI - Neural Networks - House Pricing.ipynb ☆


Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda [Se editó por última vez: 9 may 2019](#)

+ Código + Texto

Conectar



En esta notebook, estamos buscando predecir el precio de casas de en Boston utilizando información inmobiliaria.

 houses


Se trata de un problema típico de Machine Learning.

Se utiliza información pre-diseñada y cuya descripción se puede encontrar on-line.

▼ Los datos a usar

```
[ ] 1 import pandas as pd
    2
    3 columnNames = ["CRIM","ZN","INDUS","CHAS","NOX","RM","AGE","DIS","RAD","TAX","PTRATIO","B","LSTAT","MEDV"]
    4 url="https://archive.ics.uci.edu/ml/machine-learning-databases/housing/housing.data"
    5
    6 # Leemos el dataset usando pandas read_csv https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read
    7
    8 dataframe =pd.read_csv(url, names=columnNames, delim_whitespace=True, header=None)
    9 dataset = dataframe.values
```

```
[ ] 1 print (dataset)
    2 print(len(dataset))
    3 print(len(dataset[0]))
    4
```



```
[[6.3200e-03 1.8000e+01 2.3100e+00 ... 3.9690e+02 4.9800e+00 2.4000e+01]
 [2.7310e-02 0.0000e+00 7.0700e+00 ... 3.9690e+02 9.1400e+00 2.1600e+01]
 [2.7290e-02 0.0000e+00 7.0700e+00 ... 3.9283e+02 4.0300e+00 3.4700e+01]
 ...
 [6.0760e-02 0.0000e+00 1.1930e+01 ... 3.9690e+02 5.6400e+00 2.3900e+01]
 [1.0959e-01 0.0000e+00 1.1930e+01 ... 3.9345e+02 6.4800e+00 2.2000e+01]
 [4.7410e-02 0.0000e+00 1.1930e+01 ... 3.9690e+02 7.8800e+00 1.1900e+01]]
506
14
```

También, está disponible en Sklearn.

Sklearn load Boston Housing Prices: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_boston.html

IA en el software

Código Fuente del Software

Código Fuente
de Machine Learning

ML

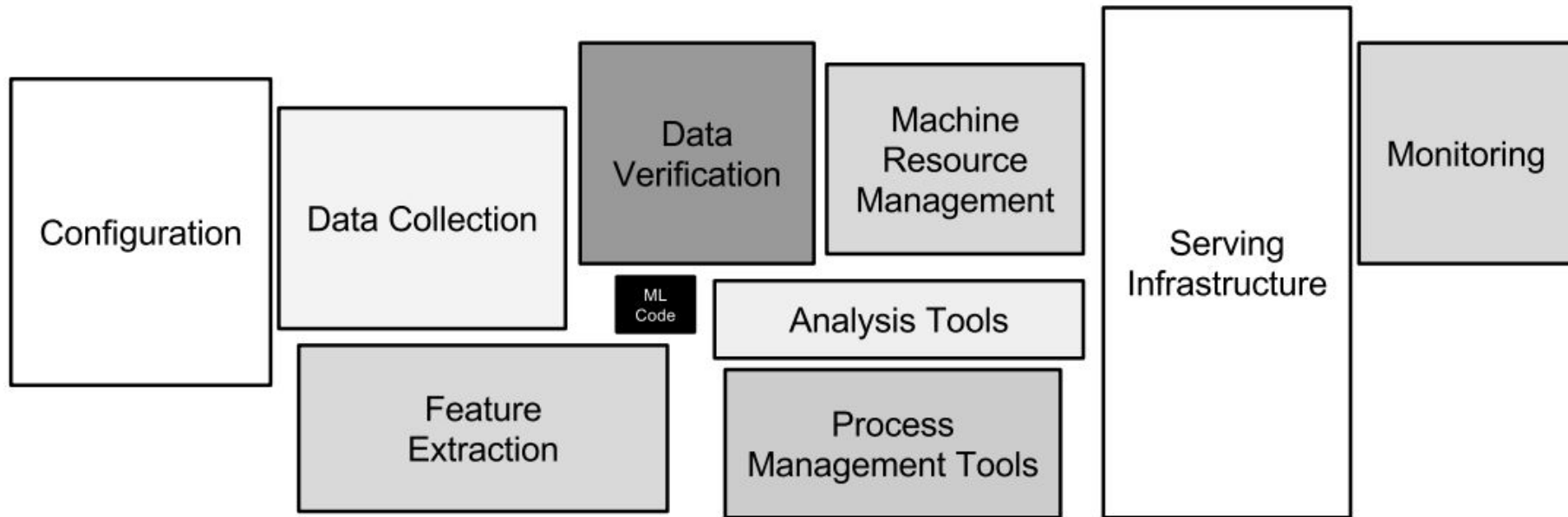
- Backend
- Procesos que corren en background
- Bases de Datos Relacionales - NonSQL
- Mensajería (Kafka, pub/sub)
- Código de Interfaces de Usuario
- Código de Infraestructura
- Datos estáticos - imágenes - configuraciones
- Scripts
- Testeo automatizado

Actividad: Deuda Técnica Oculta en Sistemas de ML

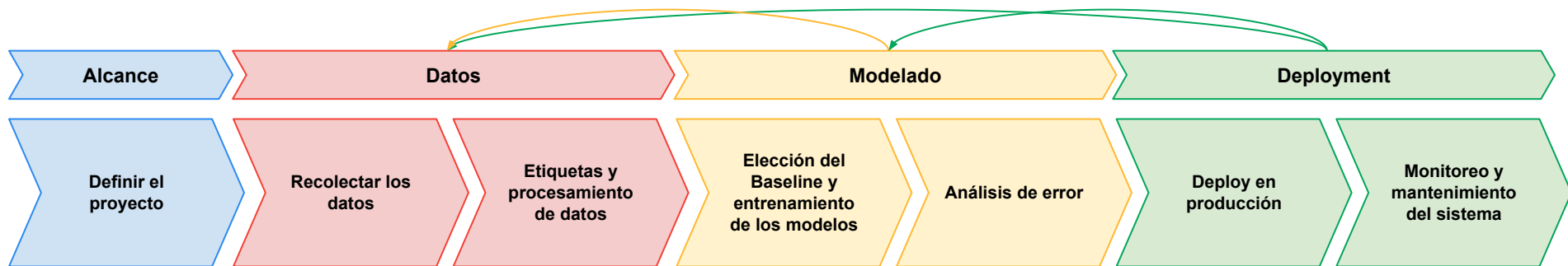
En grupos analicen el paper, Hidden Technical Debt in Machine Learning Systems - D. Sculley at el. y respondan:

1. De la sección 1 Introducción: ¿A qué se llama Technical Debt en Ingeniería del Software?
2. ¿Cómo se relaciona el Technical Debt con Machine Learning?
3. ¿A qué llaman los autores como Erosión de Bordes (Erode Bounderies) en los modelos complejos de ML y que relación tiene respecto a las afirmaciones tradicionales de la ingeniería del Software?
4. Expliquen a que se refieren los autores con la dependencia de datos de los sistemas de ML y el impacto que tiene en el correcto funcionamiento de sistema al largo plazo
5. ¿Qué soluciones proponen los autores para la Dependencia de Datos?
6. Averigüen a que se llama Design Patterns en Ingeniería del Software. Luego, lean la sección ML System Anti-Patterns. Describan brevemente lo que exponen los autores.
7. Extraigan las principales ideas de las secciones que hablan con lidiar con los cambios externos y otras areas con deudas en ML (7 Dealing with Changes in the External World y 8 Other Areas of ML-related Debt)
8. Discutan en grupo lo que han aprendido de la lectura del paper y relaciónenlo con su actualidad o su futuro como profesionales en AI (tanto en investigación como en industria).

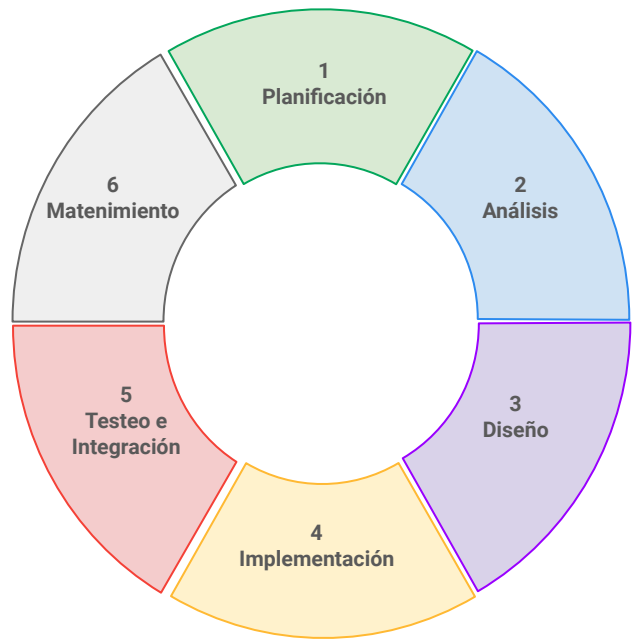
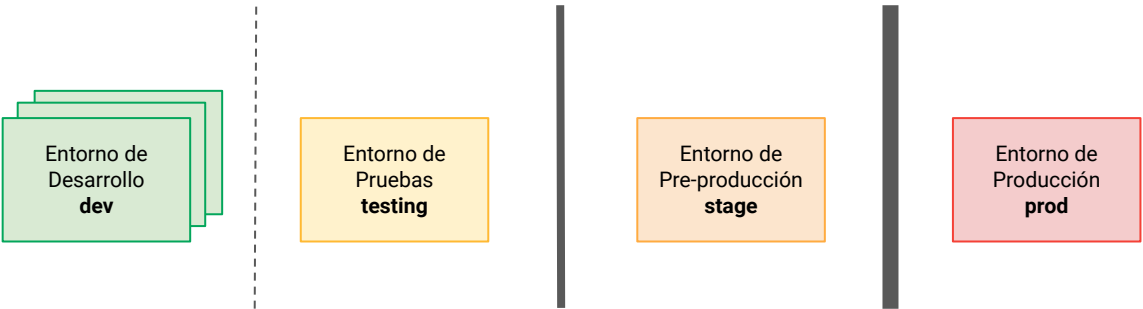
El ecosistema de un sistema de ML



El flujo de creación de un modelo de ML



Los estándares: Ambientes del desarrollo de Software

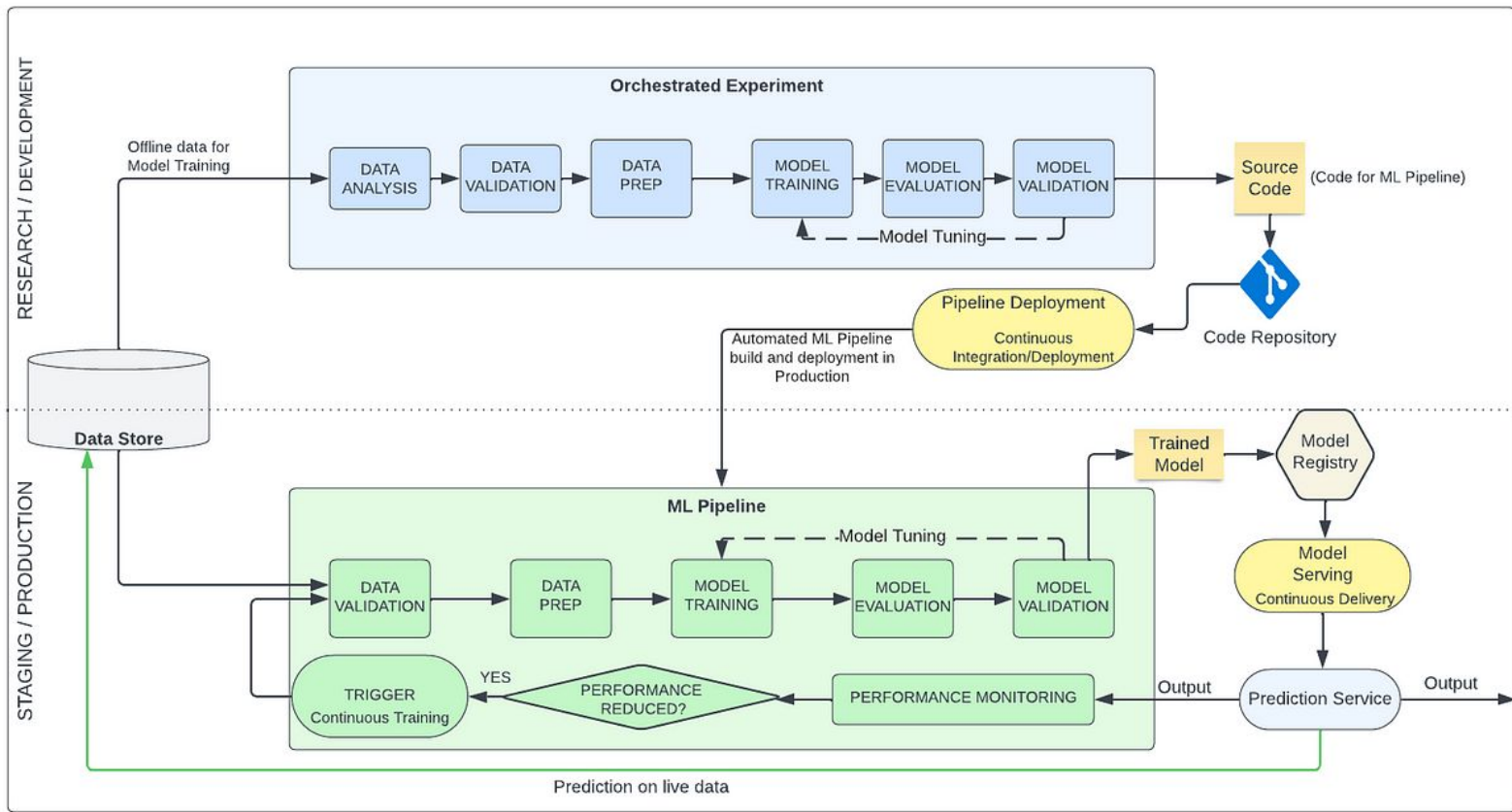


**Ingenieria del
Software
(DevOps)**

**Aprendizaje
Automático
(Flujo de ML)**



El flujo de creación de modelos



MLOps

MLOps (Machine Learning Operations) es un paradigma que incluye aspectos como las mejores prácticas, conjuntos de conceptos y una cultura de desarrollo en lo que respecta a la conceptualización, implementación, monitoreo, despliegue y escalabilidad de extremo a extremo de productos de aprendizaje automático. Sobre todo, es una práctica de ingeniería que aprovecha tres disciplinas contribuyentes: aprendizaje automático, ingeniería de software (especialmente DevOps) e ingeniería de datos. MLOps tiene como objetivo producir sistemas de aprendizaje automático al cerrar la brecha entre el desarrollo (Dev) y las operaciones (Ops). Básicamente, MLOps tiene como objetivo facilitar la creación de productos de aprendizaje automático al aprovechar estos principios: automatización de CI/CD, orquestación de flujo de trabajo, reproducibilidad; control de versiones de datos, modelos y códigos; colaboración; capacitación y evaluación continuas de ML; seguimiento y registro de metadatos de ML; monitoreo continuo; y bucles de retroalimentación.

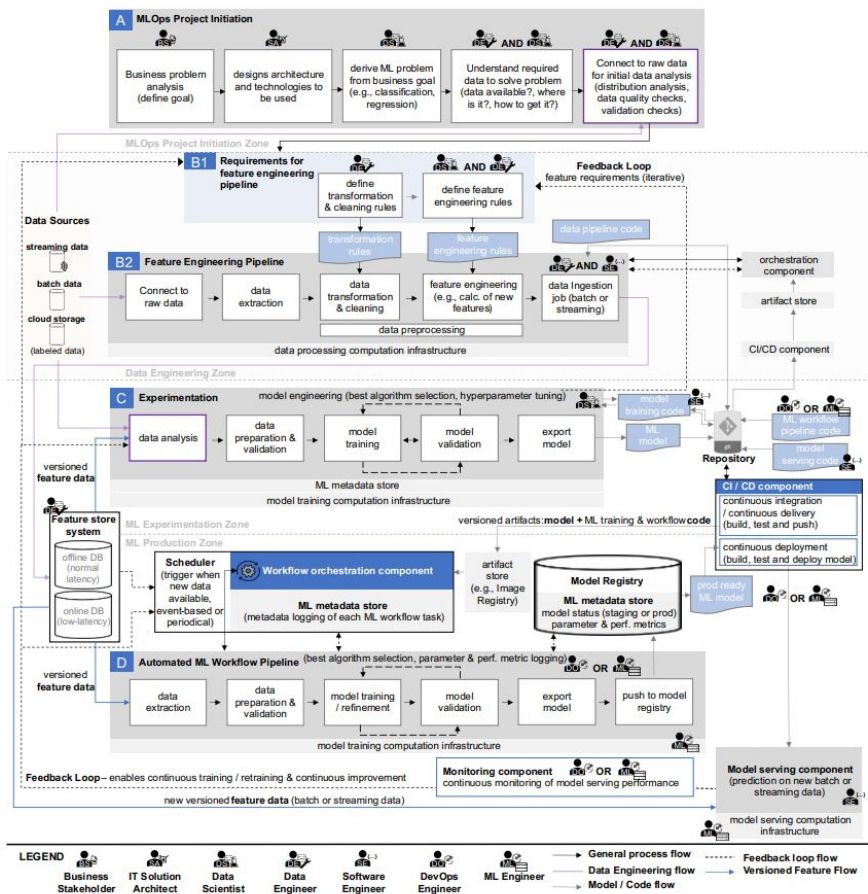
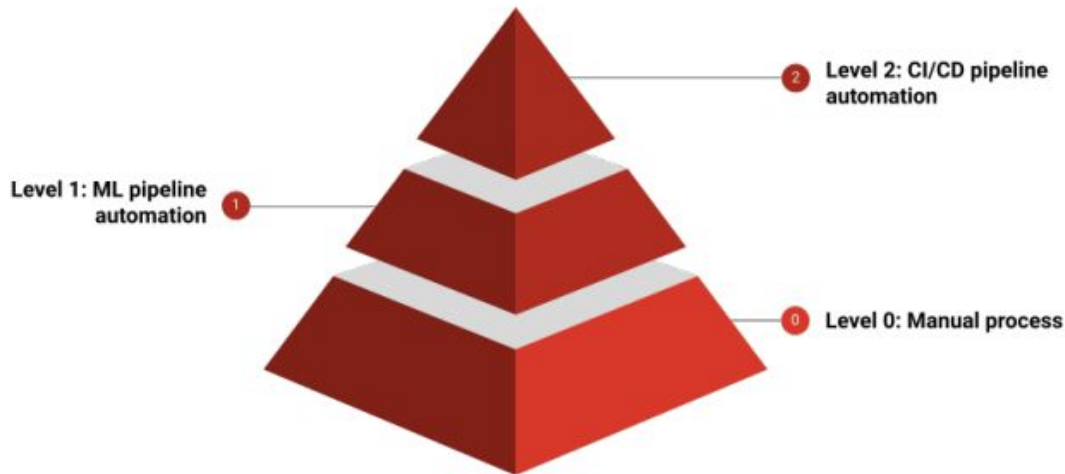


Figure 4. End-to-end MLOps architecture and workflow with functional components and roles

Niveles de Maduración de MLOps

Acorde al grado de automatización / maduración de los distintos pipelines se definen niveles de maduración:

- Nivel 0:
Procesamiento Manual
- Nivel 1:
ML Pipeline automatizado
- Nivel 2:
CI/CD pipeline automatizado



- Los proveedores de cloud pueden definir otros niveles: [Symeonidis et al. MLOps - Definitions, Tools and Challenges, 2022](#)
- Hay trabajos que buscan definirlos: [John et al., Towards MLOps: A Framework and Maturity Model, 2021](#)

MLOps

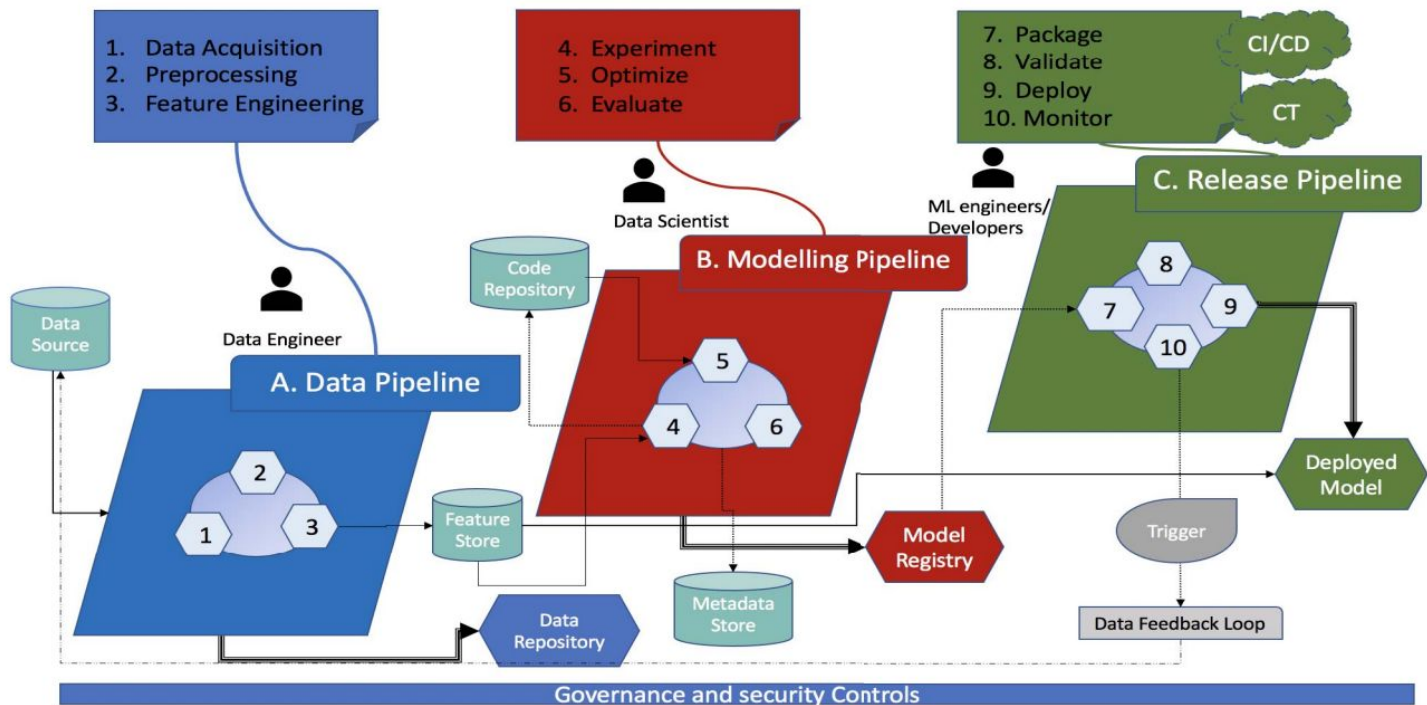


Fig. 2. MLOps Framework

Material Recomendado de esta semana

[Introduction to Agile Methodologies](#) o cualquier video que les sirva para entender más sobre Scrum

[DevOps en 5 minutos](#) o cualquier video que les sirva para entender más sobre DevOps

[CI/CD](#) o cualquier video que les sirva para entender más de CI/CD

Chip Huyen, Designing Machine Learning Systems, 2022 - Chapter 1: Overview of Machine Learning Systems - Machine Learning in Research Versus in Production - Pág. 12-21

Chip Huyen, Designing Machine Learning Systems, 2022 - Chapter 1: Overview of Machine Learning Systems - Pág. 22-23

Otros Recursos

Libro Glenford Myers, "The Art of Software Testing", 2004