
This is the **published version** of the bachelor thesis:

Bello Marès, Daniel. *Desenvolupament d'una aplicació web per l'avaluació d'algoritmes avançats de procés d'imatges*. Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/334456>

under the terms of the  license.

Desenvolupament d'una aplicació web per l'avaluació d'algoritmes avançats de procés d'imatges

Daniel Bello Marès

Resum—Aquest projecte estudia les possibilitats d'extensió i reutilització de funcionalitats de la plataforma Codabench amb l'objectiu de millorar la seva aplicabilitat en benchmarks de processament i restauració d'imatges. El treball s'ha centrat en l'anàlisi de l'arquitectura, el funcionament intern i les característiques ja disponibles a la plataforma, identificant aquelles funcionalitats susceptibles de ser reutilitzades per donar resposta als requisits plantejats sense necessitat de desenvolupar noves solucions des de zero. Aquesta tasca ha permès aprofitar mecanismes existents i reduir la complexitat associada a la implementació de noves funcionalitats. Paral·lelament, s'han dissenyat i implementat millores en el sistema de visualització de resultats, incorporant noves formes de representació i exploració de les mètriques generades durant l'avaluació dels algoritmes. Finalment, s'han identificat línies de continuació sobre la feina feta i possibles millores que es podrien afegir en un futur per fer el resultat presentat més complet.

Paraules clau—Codabench, competició, repte, codi obert, mètriques d'avaluació, processament d'imatges

Summary—This project studies the possibilities of extending and reusing functionalities of the Codabench platform with the aim of improving its applicability in image processing and restoration benchmarks. The work has focused on the analysis of the architecture, internal operation and features already available on the platform, identifying those functionalities that can be reused to respond to the requirements raised without the need to develop new solutions from scratch. This task has allowed to take advantage of existing mechanisms and reduce the complexity associated with the implementation of new functionalities. In parallel, improvements have been designed and implemented in the results visualization system, incorporating new forms of representation and exploration of the metrics generated during the evaluation of the algorithms. Finally, lines of continuation have been identified on the work done and possible improvements that could be added in the future to make the presented result more complete.

Key words— Codabench, competition, challenge, open source, evaluation metrics, image processing



1 CONTEXT DEL TREBALL

Al llarg dels últims anys, el creixement de disciplines com la IA, l'aprenentatge automàtic o la visió per computador ha fet necessari disposar de mecanismes estandarditzats per avaluar i comparar algoritmes de manera objectiva. La reproductibilitat dels experiments i la comparabilitat dels resultats s'han tornat aspectes clau, especialment en entorns on múltiples investigadors treballen sobre els mateixos conjunts de dades i amb les mateixes mètriques d'avaluació.

Amb l'objectiu de facilitar aquest procés van sorgir les plataformes d'avaluació automàtica, que permeten executar, validar i comparar solucions sota condicions controlades. Aquestes plataformes proporcionen infraestructures capaces de gestionar conjunts de dades, programes d'avaluació, mètriques i classificacions, reduint de manera significativa l'esforç necessari per organitzar competicions científiques i experiments comparatius.

En aquest context s'emmarca aquest Treball de Final de Grau. L'objectiu principal ha estat analitzar les plataformes d'avaluació automàtica de codi obert existents i seleccionar la que millor s'adaptés a les necessitats del *Grup de Recerca d'Anàlisi d'Imatges Multiespectrals del Centre de Visió per Computador*. Un cop escollida la plataforma, s'han identificat les funcionalitats que ja donaven resposta als requisits del grup i, posteriorment, s'han desenvolupat les modificacions necessàries per cobrir aquells casos d'ús que no estaven suportats de manera nativa. D'aquesta manera, el treball combina tant una fase d'estudi i avaluació de solucions existents com una fase d'adaptació i extensió de la plataforma seleccionada.

2 OBJECTIU DEL TREBALL

L'objectiu d'aquest treball és dotar el grup de recerca esmentat anteriorment d'una plataforma que permeti l'avaluació d'algoritmes de processament d'imatges de manera eficient i estandarditzada. Per assolir aquest objectiu, no s'ha optat pel desenvolupament d'una solució des de zero, sinó per l'estudi i l'aprofitament de plataformes d'avaluació automàtica de codi obert ja existents. Aquesta aproximació permet reutilitzar funcionalitats consolidades i

-
- E-mail de contacte: Daniel.BelloM@autonoma.cat
 - Menció realitzada: Enginyeria de Software
 - Treball tutoritzat per: Daniel Ponsa Mussarra (Àrea de Ciències de la Computació i Intel·ligència Artificial)
 - Curs 2025/26

àmpliament provades, alhora que redueix l'esforç de desenvolupament necessari. El treball se centra, per tant, en la identificació de la plataforma més adequada, l'anàlisi de les funcionalitats que ofereix de manera nativa i l'adaptació o extensió d'aquells components necessaris per satisfer els requisits específics del grup de recerca.

Per tal d'assolir aquest objectiu general, s'han definit diversos objectius específics. En primer lloc, s'ha dut a terme una anàlisi de les plataformes d'avaluació automàtica disponibles en l'ecosistema de programari lliure, amb l'objectiu d'identificar-ne les característiques principals, els avantatges i les limitacions. Aquesta anàlisi ha permès seleccionar la solució que millor s'ajusta als requisits funcionals i tècnics del projecte.

Un cop seleccionada la plataforma, el següent objectiu ha consistit a estudiar-ne l'arquitectura i el funcionament intern per comprendre quines funcionalitats podien ser reutilitzades directament i quines requerien modificacions o ampliacions. Aquest procés ha estat fonamental per minimitzar els canvis necessaris i mantenir la compatibilitat amb el projecte original.

A continuació, s'han implementat les adaptacions necessàries per donar suport als casos d'ús específics plantejats pel grup de recerca. Entre aquests requisits destaquen la possibilitat d'avaluar algoritmes de processament d'imatges sobre diferents conjunts de dades, gestionar múltiples tasques d'avaluació i proporcionar una visualització adequada dels resultats obtinguts.

3 METODOLOGIA

El treball ha portat uns mesos de desenvolupament (de febrer a juliol aproximadament), durant els quals haurien pogut sorgir problemes o noves sol·licituds per part del grup de recerca. Per això, ha sigut necessari utilitzar una metodologia que ens permeti ser flexibles a l'hora de dur a terme canvis sobre el que ja portem fet. Però, a l'hora d'escollir una metodologia adient al treball hem hagut de considerar que només hi ha una persona rere del treball, per la qual cosa no hem pogut considerar metodologies de treball en equip, com per exemple *Scrum* [12]. Tot i que existeixen metodologies de treball que són adaptables per a que les utilitzi només una persona (per exemple, *Kanban* [11]), vaig considerar que en el meu cas no era necessari cenyir-se a una metodologia concreta, sinó que seria millor utilitzar una de pròpia adaptada a les meves necessitats.

La metodologia que he seguit durant el desenvolupament del projecte s'ha basat en un enfocament iteratiu i de seguiment continu, pensat per garantir una validació periòdica dels objectius assolits. Per això, es van establir reunions setmanals amb el meu tutor a les quals es presentaven els resultats obtinguts o problemes trobats durant la setmana. Aquest format ha permès poder anar incorporant millores i arreglar error de manera progressiva.

4 PLANIFICACIÓ

La realització d'aquest Treball de Fi de Grau s'ha desenvolupat seguint una planificació incremental, dividida en diferents fases que han permès abordar progressivament l'estudi, desenvolupament i validació de les millores incorporades a la plataforma seleccionada [Veure Taula 1]. Aquestes fases han estat les següents:

- **Fase 1. Elicitació de requisits del sistema:** Durant aquesta primera etapa el principal objectiu va ser determinar quins eren els requisits que el grup de recerca necessitava que tingués la plataforma.
- **Fase 2. Elecció de plataforma i recerca sobre aquesta:** En aquesta fase vam centrar-nos en estudiar i entendre les diferents plataformes de codi obert que ja es troben al mercat i que podríem reaprofitar. Un cop vam escollir la plataforma ens vam centrar en determinar quines característiques de la qual ens són d'ús per als nostres objectius i quines li falten i hem hagut d'implementar.
- **Fase 3. Tasques de desenvolupament de les millores i modificació dels protocols d'ús:** Un cop identificades a la fase prèvia les millores o casos d'ús que ens ofereix la plataforma i quins no de manera predeterminada, en aquesta fase hem hagut d'implementar les característiques que falten i trobar una manera de satisfer alguns dels requisits del grup de recerca mitjançant mecanismes nadius de la plataforma.
- **Fase 4. Elaboració de material de suport a l'usuari:** En aquesta etapa ens hem centrat en elaborar un manual que permeti als usuaris finals, és a dir el grup de recerca, tenir una guia de com utilitzar la plataforma per a dur a terme les seves investigacions.
- **Fase 5. Fase final:** Aquesta darrera fase del treball ha consistit en elaborar l'informe final i el dossier. Així com preparar la presentació final de cara a la defensa davant del tribunal.

A l'apèndix A1 es pot veure un diagrama de Gantt generat amb PlantUML amb la planificació del treball.

Taula 1. Principals objectius del treball i les seves fites d'assoliment

Objectius i fites d'assoliment	
Recopilació de requisits	12/03/2026
Elecció de plataforma i recerca sobre aquesta.	02/04/2026
Recerca per poder dur a terme reptes de visió a la plataforma triada.	16/04/2026
Implementar millores de visualització qualitativa dels resultats	10/05/2026
Implementar exportació de resultats quantitativus en format CSV	20/05/2026
Redacció manual d'usuari	13/06/2026
Informe final i dossier	25/06/2026
Elaboració presentació	30/06/2026

5 ESTAT DE L'ART

En els darrers, la creixent rellevància de la Intel·ligència Artificial y el Aprenentatge Automàtic ha impulsat el desenvolupament de plataformes especialitzades per a la organització de competicions i avaluacions reproduïbles. Aquestes plataformes permeten comparar algoritmes de manera objectiva i fomentar la investigació col·laborativa.

Algunes de les principals plataformes que podem trobar són les següents:

- **Middlebury Computer Vision Pages:** Llançada el 2002, aquesta plataforma proposa datasets i protocols estandarditzats en tasques de visió per computador com stereo matching i optical flow. Tot i que no constitueix una plataforma genèrica per crear noves competicions, ha tingut un paper fonamental en la estandardització de mètriques i en la comparació objectiva d'algoritmes. [1]
- **Kaggle:** Va sortir el 2010. Kaggle és una plataforma que ofereix competicions, conjunts de dades i un entorn col·laboratiu per científics de dades, analistes i desenvolupadors d'aprenentatge automàtic. Proporciona infraestructura completa per la gestió de datasets, avaluació automàtica i generació de rankings, facilitant la participació en massa de la comunitat. Tot i això, al ser una plataforma propietària (el seu codi font no és públic), el seu nivell de personalització i control es limita des de la perspectiva de l'organitzador. [2]
- **Codalab:** Codalab va veure la llum el 2013, i ha sigut durant anys una referència en l'organització de competicions associades a conferències científiques. El seu model basat en lots i execució reproducible mitjançant contenidors ha influït directament en el desenvolupament de plataformes posteriors. [2]
- **Alcrowd:** Alcrowd, publicada el 2018, és una plataforma que ofereix suport per a competicions patrocinades i reptes tècnics de diversa complexitat. Permet a organitzacions i comunitats coordinar competicions col·laboratives que acceleren el desenvolupament i avaluació de models predictius en entorns reals. A diferència d'altres opcions com Codabench, Codalab o EvalAI aquesta no és una plataforma de codi obert.
- **EvalAI:** EvalAI va ser llançada oficialment el 2019, és una plataforma dissenyada per suportar avaluacions complexes, incloent interaccions humanes i entorns dinàmics, com agents d'aprenentatge per reforç o avaluacions amb intervenció humana (human-in-the-loop). La seva arquitectura basada en contenidors permet executar codi enviat pels participants en entorns aïllats. [4]
- **Codabench:** Originada com evolució conceptual de Codalab, Codabench va sortir al públic el 2023 i és una plataforma amb un fort èmfasi en la reproducibilitat, la modularitat i la definició explícita de tasques (tasks), datasets i mètriques. Codabench facilita tant els reptes tradicionals (algoritmes davant

d'un dataset) com configuracions més avançades, incloent reptes invertits (avaluació de datasets davant algoritmes de referència). [5][7]

5 SELECCIÓ DE PLATAFORMA

A l'hora de tenir que escollir quina plataforma utilitzar com a base per dur a terme aquest treball la qüestió sempre ha estat entre plataformes de codi obert que ens deixessin poder fer les modificacions necessàries per arribar a implementar tots els casos d'ús desitjats. Després d'haver fet recerca sobre EvalAI i Codabench (principals plataformes de codi obert d'avaluació d'algoritmes), i haver comparat ambdues plataformes, he pres la decisió d'utilitzar Codabench com a plataforma base pel desenvolupament d'aquest treball. Aquesta decisió és deguda a diversos motius pels quals he considerat que Codabench és una millor opció.

En primer lloc, Codabench té una arquitectura més modular que no pas EvalAI. Codabench destaca per la seva arquitectura basada en contenidors de Docker que permeten desplegar la plataforma sense gaire dificultat, mentre que EvalAI posseeix una arquitectura més monolítica i complexa.

Per una altra banda, l'organització de la documentació oficial de Codabench m'ha semblat molt intuïtiva i fàcil de poder trobar allò que necessites. Tot sigui dit, EvalAI també té una documentació força rica però en la meua opinió era més superficial i no tan fàcil de seguir com la de Codabench.

Per últim, m'ha semblat que la manera d'organitzar reptes a Codabench és més clara que no pas a EvalAI.

A la Taula 2 podem veure una taula comparativa amb els principals aspectes que he tingut en compte a l'hora de triar la plataforma [2].

Taula 2. Comparativa d'EvalAI i Codabench

Aspecte	EvalAI	Codabench
Codi obert	Sí	Sí
Bona documentació de la plataforma	No és dolenta però la he trobat més complexa d'entendre i pitjor organitzada.	Al meu parer molt bona documentació i fàcil d'entendre i trobar el que necessites.
Gestió integrada de recursos (conjunts de dades, programes)	No	Sí
Possibilitat de desplegar la plataforma en servidors propis	Sí	Sí
Comunitat activa	No tant com Codabench	Més que EvalAI

6 RECOPILACIÓ DE REQUISITS

Pertal de recollir els casos d'ús [Veure Taula 3] per extreure els requisits es va dur a terme a principis del mes de març del 2026 una sessió de *story mapping* [10] amb membres del grup de recerca per tal de que exposessin les seves necessitats. El resultat de dita sessió va ser un tauler on es recullen els casos d'ús desitjats pel sistema, tal i com es mostra a l'apèndix A2.

Amb aquest tauler el següent pas va ser realitzar el diagrama de casos d'ús que es pot observar a l'apèndix A3.

Taula 3. Taula amb casos d'ús més importants i si ja els ofereix la plataforma

Cas d'ús	L'ofereix ja Codabench?
Visualització qualitativa de resultats	No
Exportació de resultats en format CSV	No, de manera nativa només es pot descarregar el rànquing de la competició en format CSV
Descarregar imatges resultants	No
Casos d'ús relacionats amb l'administració del sistema	Sí
Casos d'ús relacionats amb la gestió d'usuaris	Sí
Veure resultats detallats d'un participant	No
Pujar resultats/ algoritme per avaluar-ho	Sí
Veure l'avaluació d'un algoritme	Sí, si només ens referim a veure el resultat de les mètriques d'avaluació utilitzades

7 ANÀLISIS DEL FUNCIONAMENT DE CODABENCH

Per tal de contextualitzar les contribucions realitzades en aquest treball, és necessari comprendre com funciona Codabench. Codabench és una plataforma de codi obert orientada a la creació i execució de competicions d'avaluació, especialment dins de l'àmbit de la intel·ligència artificial i l'aprenentatge automàtic. El seu objectiu és proporcionar un entorn estandaritzat per comparar de manera objectiva les solucions proposades pels usuaris finals a diferents reptes amb un o més conjunts de dades i mètriques.

La plataforma estructura les competicions a partir de diversos elements interrelacionats, com ara fases, tasques, conjunts de dades, programes d'ingestió i d'avaluació. Quan un participant envia la seva proposta de solució, aquesta s'executa en un entorn controlat amb els components necessaris per generar els resultats i les mètriques corresponents.

En les subseccions següents es descriuen amb més detall els components i els processos que utilitza Codabench per al processament de les solucions dels usuaris.

7.1 TIPUS DE COMPETICIONS A CODABENCH

A Codabench les competicions són el principal element de la plataforma i cadascuna constitueix un contenidor on podem trobar tot allò que necessitem per dur avaluació les solucions pujades pels participants d'aquestes. Una competició defineix el problema que es vol resoldre, la documentació associada, les fases d'execució, les tasques d'avaluació i els rànquings amb els resultats.

Tot i que de cara als objectius d'aquest treball el tipus de competició o repte més important a tenir en compte és un del tipus *code submission*, en el qual nosaltres pugem un model i la plataforma ho executa sobre un o més conjunts de dades i mesura els resultats amb unes mètriques determinades, Codabench permet tenir altres tipus de competicions. [6]

A banda del tipus esmentat abans on el participant puja el seu propi codi, l'altre principal tipus de competició a Codabench és aquella on el participant ha de pujar únicament els resultats obtinguts a partir de l'execució en local del seu model. En aquest cas, Codabench utilitza el *scoring program* proporcionat per l'organitzador de la competició per comparar els resultats del participant amb les dades de referència.

Un altre mecanisme (no és un tipus de competició per si mateix) interessant que ens ofereix Codabench són les competicions amb format multi tasca, és a dir, competicions que tenen més d'una tasca a executar. Això és útil quan volem executar i avaluar un mateix model però volem que utilitzi múltiples conjunts de dades, per això hem de definir tantes tasques com conjunts de dades volem utilitzar i fer referència a aquestes tasques dins el *competition.yaml*.

7.2 FUNCIONAMENT DE LA PLATAFORMA

Durant aquesta fase del projecte s'ha dut a terme una recerca sobre el funcionament a nivell intern de Codabench per tal de comprendre com es gestionen les competicions, i com funciona l'arquitectura que permet la seva execució.

Organització de competicions

A Codabench, una competició es podria entendre com un entorn d'avaluació dissenyat per comparar el rendiment de diferents algoritmes o perspectives sobre un problema concret. Per organitzar una competició a Codabench ho podem fer a través de la plataforma web, o a través de la pujada a la plataforma d'un arxiu comprimit on es descriu l'estructura de la competició que volem crear. Aquest fitxer comprimit ha de tenir una estructura concreta [Veure Taula 4].

Taula 4. Estructura del fitxer zip per crear una competició

Contingut del fitxer comprimit de creació de competició	
Competition.yaml	En aquest fitxer es defineixen les característiques bàsiques i l'estructura per la nostra competició.
Logo.png	Un logo per la nostra competició, que tot i que no ho sembli és obligatori afegir-ho.
Reference_data.zip	Un fitxer comprimit on hi posarem les nostres dades de referència (en anglès <i>ground truth</i>) per poder comparar els resultats obtinguts.
Input_data.zip	En aquest fitxer posarem les dades amb les que s'executarà el model del participant per produir resultats.
Ingestion_program.zip	Dins d'aquest zip hi hauran d'haver-hi dos arxius; un amb el codi del programa d'ingestió, encarregat d'executar el nostre model i produir les dades de resultat, i un meta-data.yaml, el contingut del qual serà la comanda necessària per executar el nostre codi.
Scoring_program.zip	En aquest fitxer es trobarà el programa per calcular les mètriques d'avaluació i es generen les visualitzacions detallades de resultats que hem implementat. Igual que al d'ingestió també hi haurà un meta-data.yaml.
pages/	Directorio on es troben diferents arxius <i>markdown</i> on es defineixen els termes i condicions de la competició, les fases, com s'avaluen els models, etc...

Al fitxer *competition.yaml* és on es defineix tota l'estructura i disposició de la nostra competició, a més d'altres detalls també importants. Però, aquesta estructura mostrada a la Taula 3 no és constant, és a dir, pot variar segons el tipus de competició que vulguem organitzar. L'organitzador també ha de proporcionar una programa d'ingestió (encarregat d'executar el codi del participant), el *scoring* (les mètriques que avalua les prediccions del participant amb les dades de referència i el posiciona en el rànquing), les dades d'entrada (amb les que el participant genera les seves prediccions) i les

dades de referència (el resultat ideal).

Participació en competicions

Per prendre part en una competició cada participant haurà de pujar un fitxer comprimit amb el necessari per poder participar. Generalment, si la competició consisteix en pujar codi (tipus de competicions més comuns) el fitxer comprimit del participant haurà de tenir el fitxer de codi necessari per poder executar-lo i extreure els resultats pertinents per avaluar el rendiment. Si pel contrari la competició consisteix en que els participants pugin els seus resultats per compararlos amb un resultat ideal, en aquest cas hauran de pujar un arxiu amb els seus resultats per tal de que el sistema els pugui comparar amb els de l'organitzador. A diferència del cas dels organitzadors, no hi ha un estàndard de com hauria de ser el fitxer comprimit que pugen els participants, sinó que han de pujar un que s'adapti a les necessitats especificades per l'organitzador per poder participar al repte.

Estructura d'execució

Probablement l'aspecte més important en una plataforma de *benchmarking* és com s'executen i avaluen els models dels participants. A Codabench, cada participació s'executa dins d'un entorn *Docker* aïllat, garantint la reproductibilitat dels resultats. El flux d'execució comença amb el participant enviant el seu model en un arxiu comprimit. Un cop validat, Codabench assigna la seva execució a un *compute worker*, que crea un contenidor temporal utilitzant la imatge *Docker* especificada en el *competition.yaml*. Dins el contenidor es creen diversos directoris amb funcions específiques i s'afegeixen els programes de ingestió i avaluació, responsables de generar les prediccions i calcular les mètriques finals. El primer pas és executar el *ingestion.py*; aquest programa rep les dades d'entrada i la solució del participant, executa l'algoritme proposat i genera els resultats corresponents. A continuació s'executa el *scoring.py*, que compara les prediccions amb les dades de referència. A partir d'aquesta comparació es calculen les mètriques d'avaluació definides per l'organitzador, els resultats s'emmagatzemen en un fitxer *scores.json*, les claus del qual han de coincidir amb les claus de les columnes del rànquing. Finalment, Codabench recull les puntuacions obtingudes i actualitza automàticament la classificació de la competició. Si s'ha activat l'opció a l'hora de crear la competició, és en aquest darrer pas quan també es genera el fitxer *detailed_results.html* amb les millores de visualització de resultats implementades. En el diagrama que es mostra a continuació [Veure Figura 4], podem veure una esquematització del funcionament de l'estructura d'execució de Codabench; es pot veure com el *Ingestion* crida al nostre model, el *Scoring* fa una comparació entre el resultat de l'execució del nostre codi i el resultat ideal que hem proporcionat, per a posteriorment enviar la informació al rànquing i mostrar-ho a la pàgina de la competició.

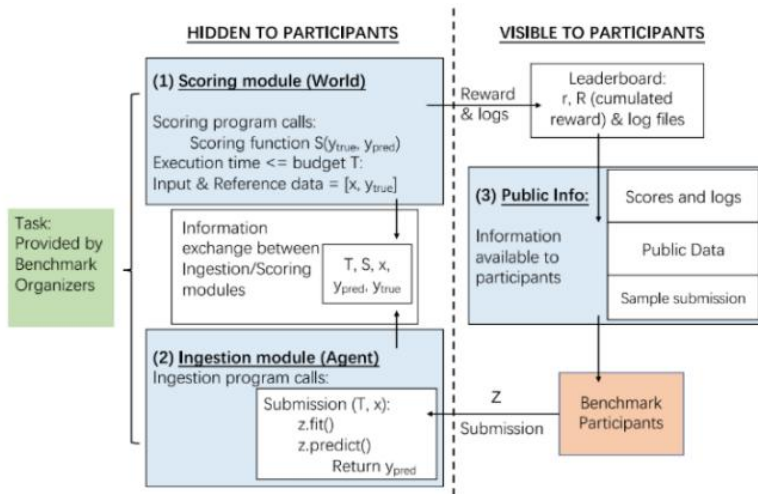


Fig. 1. Diagrama de l'estructura d'execució de Codabench [9]

Sistema de tasques

Un aspecte clau que ofereix la plataforma i amb el que he treballat força al llarg del treball ha sigut el sistema de tasques. Una tasca representa una unitat d'avaluació independent dins d'una competició. Cada tasca encapsula tots els elements necessaris per avaluar una solució: les dades d'entrada, les de referència, el programa d'avaluació i el d'ingestió. Aquesta estructura modular permet definir diversos problemes dins d'una mateixa competició i reutilitzar components d'avaluació de manera eficient. Per això, durant el desenvolupament d'aquest treball s'ha fet un ús intensiu d'aquesta funcionalitat per tal d'ampliar les capacitats d'avaluació de la plataforma. En comptes de limitar-nos a un únic conjunt de dades i mètriques per competició, s'ha utilitzat el sistema de tasques per poder avaluar diferents escenaris de visió per computador de manera independent. A més, aquesta arquitectura basada en tasques ens ha permès disposar d'avaluacions independents per cada conjunt de dades que hem volgut avaluar, d'aquesta manera s'han pogut generar les estadístiques, gràfics i comparatives d'una manera més detallada, superant les limitacions que ens ofereixen els rànquings tradicionals com el que proposar la plataforma de manera predeterminada.

9 SISTEMA DESENVOLUPAT

Partint de l'anàlisi fet de les necessitats exposades pel grup de recerca i de les capacitats de la plataforma ja presents de manera nativa, al llarg del treball s'han hagut de desenvolupar modificacions dels protocols d'ús de la plataforma orientats a facilitar la gestió dels recursos necessaris per dur a terme competicions, avaluar i explotar els resultats generats durant les nostres competicions.

Les modificacions introduïdes tenen com objectiu augmentar la flexibilitat de la plataforma i proporcionar una millor experiència als organitzadors i participants de les competicions. En concret s'han incorporat mecanismes per millorar la visualització dels resultats, facilitar la incorporació de nous conjunts de dades així com de noves mètriques a competicions ja creades prèviament i permetre l'exportació

dels resultats obtinguts a formats estàndard.

Millores en la visualització de resultats

Un dels principals objectius del projecte ha sigut millorar la qualitat de les visualitzacions de resultats dins la plataforma. Per això, s'ha utilitzat la funcionalitat de *detailed_results.html*, aquest és un fitxer que tenen associades totes les participacions en un repte. Dins de dit arxiu es pot generar contingut HTML dinàmic des de el *scoring program* i mostrar-ho directament a la interfície web de la competició.

D'aquesta manera hem pogut fer que en comptes tenir només una avaluació basada en valors numèrics agregats al rànking de la competició, ara tenim una pàgina HTML dinàmica on podem tenir una experiència més detallada i visual dels resultats del nostre model, facilitant la interpretació dels resultats.

Per generar aquesta pàgina HTML, el funcionament ha sigut implementat dins del fitxer *scoring.py* (arxiu de codi on es calculen les mètriques). Dins de dit fitxer es calculen una sèrie de mètriques sobre la qualitat de la imatge generada, s'emmagatzemen els resultats dins de *scores.json* i amb aquest JSON s'alimenta el rànking, i simultàniament es construeix l'arxiu *detailed_results.html*.

Un dels aspectes més rellevants és que tots els elements visuals com gràfics o imatges que es veuen dins del fitxer *detailed_results.html* han sigut inserits com imatges codificades en *Base64* degut a les restriccions del *backend* de Codabench referents a l'accés d'arxius estàtics dins de l'entorn aïllat d'execució. Per resoldre aquest problema, es va implementar una conversió de totes aquestes figures cap a cadenes *Base64*, inserint-les posteriorment mitjançant etiquetes *img* dins del HTML. Segons la documentació oficial de Codabench, aquesta és la manera adient per inserir figures dins del fitxer *detailed_results.html* [7], tot i que d'aquesta manera la solució és poc escalable en el cas de que un conjunt de dades amb el que vulguem treballar contingui centenars o milers d'imatges, ja que el fitxer HTML generat seria força gran en quant a mida de l'arxiu es refereix.

A través de les millores implementades mitjançant el fitxer *detailed_results.html*, ara cada participació en un repte de visió per computador té associada una pàgina on podem veure una taula amb els resultats numèrics de les mètriques calculades, per cada imatge dins del conjunt de dades amb el que s'ha treballat s'ha implementat també un *slider* [Veure Figura] amb el qual podem sobreposar les imatges de referència (resultat ideal) i les imatges generades pel nostre model, i també un mapa d'error generat utilitzant la llibreria *matplotlib* [8], amb aquest mapa d'error podem veure les "zones calentes" on hi ha una major concentració d'errors a la imatge generada respecte la imatge perfecta [Veure Figura].

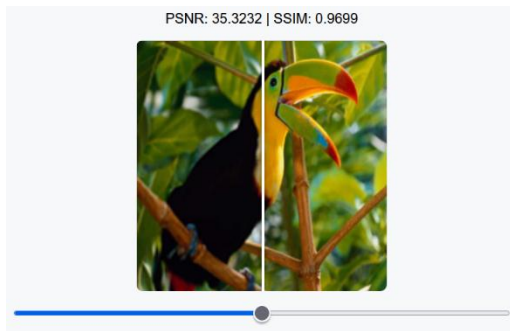


Fig. 2. Slider

per comparar el resultat obtingut i el resultat perfecte

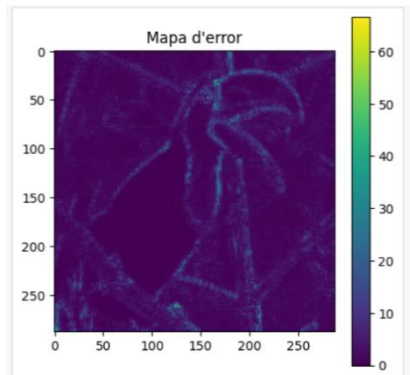


Fig. 3. Heatmap amb els

punts amb major error de la imatge obtinguda

Exportació de resultats

A banda de les millores implementades de cara a la visualització i interpretació dels resultats, també s'han afegit noves maneres de descarregar els resultats d'una determinada execució del nostre codi.

Per una banda, s'ha incorporat la funció de poder descarregar les imatges (en format png) resultants amb el fi de poder tenir una millor comprensió del funcionament del nostre model. D'aquesta manera s'han afegit dues maneres de descarregar dites imatges; la primera i més típica és descarregar només la imatge resultant, la segona manera afegida consisteix en descarregar una imatge que està formada per un costat la imatge de referència (el resultat ideal) i per l'altre la imatge que hem obtingut [Veure Figura 5], d'aquesta forma podem veure de manera més senzilla i directa la comparació entre el que és lo ideal i lo que produeix el nostre algoritme.

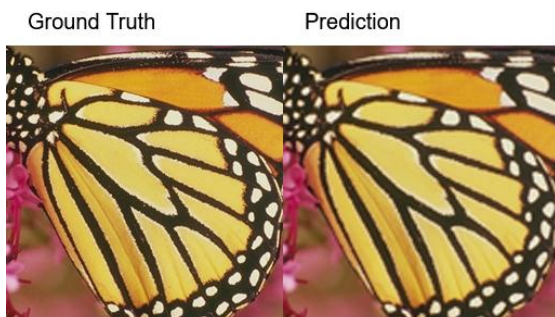


Fig. 4. Comparació entre resultat ideal i el nostre resultat

Aquesta nova imatge s'ha generat mitjançant la utilització de la etiqueta *canvas* [13] de HTML; es genera un de nou i dins d'aquestes "dibuixen" les dues imatges, la de referència a l'esquerra i la de resultat a la dreta, i s'escriu el text que podem veure a sobre.

A banda de poder descarregar les imatges, s'ha afegit l'opció de poder descarregar en format CSV la taula que podem veure dins la pàgina que genera el fitxer *detailed_results.html*, així tenim una forma de poder exportar els resultats quantitatius de manera més detallada i amb el resultat d'imatge per imatge.

Addició de noves mètriques i conjunts de dades

Un altre dels aspectes que hem pogut resoldre durant el projecte ha sigut el fet de poder afegir noves mètriques i dades a una competició ja existent, de manera que ens assegurem de que un cop tenim una competició creada podem treballar amb noves dades i/o mètriques en el cas de que sigui necessari.

Per poder incorporar aquesta millora no s'ha hagut d'implementar res de zero gràcies a que "jugant" amb el sistema de tasques (prèviament explicat a la secció 7.2 d'aquest informe) hem pogut definir un procés mitjançant el qual es poden modificar competicions existents sense necessitat de crear de nou una competició amb els nous paràmetres desitjats.

Aquest nou procés consisteix en modificar el nostre programa d'avaluació amb les noves mètriques que volem que es calculin, aquesta nova versió de l'arxiu es puja al sistema de Codabench que tenim en la nostra màquina i se'l fa referència dins de la tasca que tenim en el nostre repte o competició. Ara si tornem a enviar la nostre solució els resultats generats seran avaluats amb les noves mètriques.

Pel cas dels conjunts de dades el procediment és força similar, però en aquest cas el que haurem de definir serà una nova tasca al sistema, on les dades de referència i d'entrada siguin les que volem afegir, i des del menú de configuració de la nostra competició afegir aquesta nova tasca a la llista de tasques que utilitza el repte. I finalment, de la mateixa manera que abans, tornar a pujar la nostra solució per a que s'executi amb les noves dades.

10 CONCLUSIONS

Amb aquest treball s'ha treballat per poder donar resposta a les necessitats plantejades pels investigadors del grup de recerca, a més s'ha estudiat en profunditat la plataforma de Codabench per poder analitzar la seva viabilitat per a ser utilitzada com a eina complementària per a tasques d'investigació relacionades amb la visió per computador. Ens hem trobat amb que Codabench, tot i ser una plataforma adequada ja que satisfà de manera nativa molts dels requisits proposats, també hem hagut d'implementar millores i inventar-nos nous protocols d'ús amb les seves eines ja existents per poder satisfer requisits sense necessitat de desenvolupar noves mecàniques des de zero.

Un dels aspectes més rellevants observat al llarg del projecte ha sigut la flexibilitat que ofereix Codabench per definir competicions mitjançant arxius YAML, tasques reutilitzables i múltiples conjunts de dades. Aquesta estructura permet crear competicions complexes i adaptables a diferents dominis.

A més, gràcies a l'ús de *Docker* la plataforma permet executar *submissions* de forma aïllada, reproducible i escalable. Aquest aspecte és especialment important dins de contextos acadèmics o d'investigació, on factors com la reproductibilitat o l'estandardització dels resultats són aspectes fonamentals.

Un altre aspecte clau en aquest treball ha sigut la possibilitat que ens dona Codabench de poder mostrar una visualització personalitzada mitjançant l'arxiu *detailed_results.html*, gràcies al qual tenim la possibilitat de mostrar gràfics, mètriques i altra informació específica de manera visual, oferint a l'usuari una molt millor comprensió dels resultats obtinguts que nos pas amb només un rànquing convencional.

A nivell tècnic també ens hem trobat amb dificultats, les quals no es que siguin limitacions de la plataforma ja que hem implementat casos d'ús pels quals Codabench no estava pensat des d'un inici. La més important ha sigut la causada pel fet de que no es pugui accedir a arxius externs dins del sistema d'execució de la plataforma, fet pel qual hem hagut de fer una conversió a *Base64* de totes les figures que volíem mostrar dins de l'arxiu de *detailed_results.html*.

En conjunt, els resultats obtinguts a partir de la recerca i el desenvolupament fets en aquest treball ens demostren que Codabench, tot i tenir algunes mancances, és una plataforma robusta per dur a terme diverses tasques dins el món acadèmic i de la investigació.

10.1 LÍNIES DE CONTINUACIÓ

De cara a continuar amb l'adaptació de Codabench a les necessitats del grup de recerca el primer a fer hauria de ser aconseguir modificar el *backend* de la plataforma de manera que la inserció d'imatges en el fitxer *detailed_results.html* sigui feta de manera directa, és a dir, carregant les imatges directament des de l'entorn d'execució. Tot i que s'ha pogut "descapar" la plataforma per a que es puguin carregar arxius externs, el *backend* és massa hermètic i no permet carregar res a part de fitxers HTML. Tot i això, la solució oferta de carregar les imatges en *Base64* és una bona opció per a un producte mínim viable, però bé és cert que el fet de carregar imatges d'aquesta manera es pot tornar una cosa difícil de mantenir a llarg termini si passem a tenir conjunts de dades de centenars o milers d'imatges degut a la necessitat de carregar tantes imatges diferents d'aquesta manera.

Una altra opció que es podria afegir com a millora afegida podria ser una opció per exportar la taula que es genera a la pàgina de *detailed_results.html* en format LaTeX per

poder-la incloure dins d'algun article. I també es podria de mirar com fer per afegir una opció similar però que ens permetés compartir el nostre rànquing o la taula esmentada anteriorment a les xarxes socials.

ÚS DE LA IA GENERATIVA

Durant el treball he fet ús d'eines d'intel·ligència artificial com a mitjà de suport per a determinades tasques de recerca o redacció. El seu ús s'ha limitat a funcions de consulta, mantenint en tot moment el control i la validació dels resultats per part meua.

AGRAÏMENTS

Aquest treball ha suposat el final d'una etapa molt important de la meua vida, no només a nivell acadèmic sinó també a nivell personal. És per això, que vull aprofitar aquesta darrera secció per expressar la meua gratitud a totes aquelles persones que m'han acompanyat al llarg d'aquests mesos.

En primer lloc, vull agrair a la meua família, especialment els meus pares Juan Antonio i Ana, el seu suport incondicional, el seu esforç constant i la confiança que sempre han tingut en mi. Gràcies per haver estat al meu costat en cada etapa i haver fet possible que hagi arribat fins aquí.

A la meua parella, Júlia (qui també ha fet el seu TFG al mateix temps que jo), per la seva paciència, comprensió i suport durant tot aquest procés. Gràcies per haver-me acompanyat durant els moments d'incertesa, per celebrar amb mi cada avenç i per recordar-me la importància de seguir endavant quan les coses es compliquen.

Al meu tutor, Daniel Ponsa Mussarra, per la seva orientació, disponibilitat i consells al llarg d'aquest projecte. Gràcies per haver estat disposat a ajudar-me en tot moment i haver sigut comprensiu quan he comès errors.

Finalment, gràcies als meus companys i amics de la universitat, amb els que he compartit tants moments i hores d'estudi.

Gràcies.

BIBLIOGRAFIA

- [1] Scharstein, D., Szeliski, R. The Middlebury Computer Vision Pages, <https://vision.middlebury.edu/> Recuperat 24 de març de 2026.
- [2] Zhou, Yimin (2024). *Estudio de plataformas para la creación de competiciones aplicado a la tarea de clasificación de imágenes*. Trabajo Fin de Grado / Proyecto Fin de Carrera, E.T.S. de Ingenieros Informáticos (UPM), Boadilla del Monte.
- [3] *Competition Bundle Structure - CodaBench Docs*. (s. f.). <https://docs.codabench.org/latest/Organizers/Benchmark-Creation/Competition-Bundle-Structure/> Recuperat 25 de març de 2026.
- [4] *Introduction – EvalAI 1.1 documentation*. (n.d.). <https://evalai.readthedocs.io/en/latest/01-getting->

[started/introduction.html](#) Recuperat 24 de març de 2026.

[5] *Codabench Docs*. (s. f.). <https://docs.codabench.org/latest/> Recuperat 9 d'abril de 2026.

[6] *Competition examples - CodaBench Docs*. (s. f.). <https://docs.codabench.org/latest/Organizers/Benchmark-Creation/Benchmark-Examples/> Recuperat 17 de maig de 2026

[7] *Detailed Results and Visualization - Codabench Docs*. (s. f.). <https://docs.codabench.org/latest/Organizers/Benchmark-Creation/Detailed-Results-and-Visualizations/#how-to-include-figures> Recuperat 18 de maig de 2026

[8] *Matplotlib documentation - Matplotlib 3.10.9 documentation*. (s. f.). <https://matplotlib.org/stable/> Recuperat 18 de maig de 2026

[9] Zhen Xu, Sergio Escalera, Adrien Pavão, Magali Richard, Wei-Wei Tu, Quanming Yao, Huan Zhao, Isabelle Guyon (2022). *Codabench: Flexible, easy-to-use, and reproducible meta-benchmark platform*. Patterns, Volume 3, Issue 7

[10] Admin_Donetonic. (2024, 21 octubre). User Story Mapping, qué es y cómo crearlo. DoneTonic. <https://donetonic.com/es/user-story-mapping/>

[11] ¿Qué es un tablero de Kanban? (s. f.). Atlassian. Recuperat 19 de juny de 2026, <https://www.atlassian.com/es/agile/kanban/boards>

[12] ¿Qué es scrum? (s. f.). Scrum.org. Recuperat 19 de juny de 2026, <https://www.scrum.org/resources/blog/que-es-scrum>

[13] *canvas - HTML | MDN*. (2025, 20 junio). Recuperat 19 de juny de 2026, <https://developer.mozilla.org/es/docs/Web/HTML/Reference/Elements/canvas>

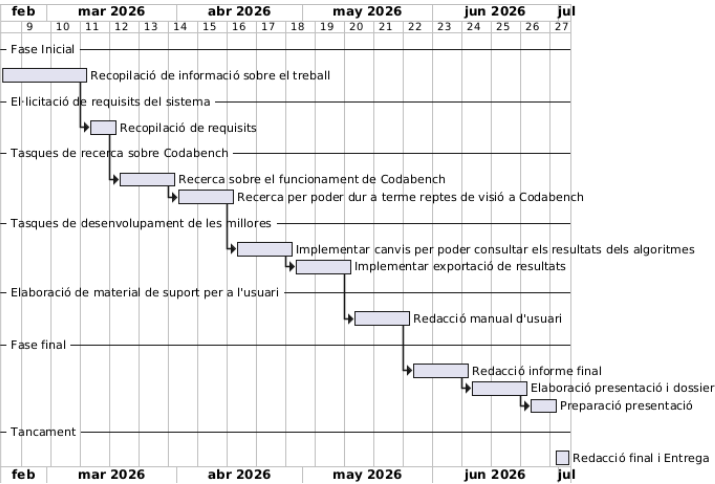
	• Diagrama UML casos d'ús actualitzat per fer-ho més vertical i no tant horitzontal.
07/06/2026	• Context adaptat. • Secció 10.1 afegida. • Agraïments afegits. • Secció Desenvolupament ara es diu Resultats. • Fonts no utilitzades eliminades. • Planificació actualitzada.
19/06/2026	• Objectius modificats. • A la bibliografia fonts 3 i 4 eliminades, fonts 11 i 12 afegides. • Planificació modificada per no fer referència a Codabench. • Figura 3 ara és Taula 3. • Seccions 7 i 8 fusionades en una sola secció. • Conclusions modificades. • Secció 9 ara es diu Sistema desenvolupat. • S'ha eliminat les mencions a súper resolució per visió per computador. • Primer paràgraf de la secció de Conclusions modificat. • El contingut de la secció 9 s'ha dividit en subseccions. • Figures 2 i 3 mogudes de l'apèndix a dins del text. • Font 13 afegida. • Canviat el color de les barres de la planificació.

HISTORIAL DE CANVIS

30/03/2026	Primera versió de l'arxiu
09/04/2026	• Context i Estat de l'art separats en dues seccions diferents. • Plataformes esmentades a l'Estat de l'Art ordenades de manera cronològica. • Canvi de metodologia escollida per dur a terme el treball. • Figures mogudes a l'Apèndix. • Secció de motius per haver escollit Codabench afegida. • Fitxers explicats a la secció d'organització de competicions. • Darrera visita a les web afegida.
18/05/2026	• Motius elecció plataformes de codi obert afegits al punt 5. • Fonts número 9, 10, 11 afegides a la bibliografia. • Seccions 8, 9 i 10 afegides. • Figures 4 i 5 afegides a l'apèndix. • Secció Ús de la IA generativa actualitzada.

APÈNDIX

A1: Planificació del treball



A2: Tauler resultat de la sessió de Story Mapping



A3: Diagrama de casos d'ús

