FlutterCV: Flutter App development with Computer Vision modules (provisional)

Juan Carlos Soriano Valle

Resum—Resum del projecte, màxim 10 línies. ....
....
....
....
....
....
....
....
....
....

Paraules clau—Paraules clau del projecte, màxim 2 línies. ....
....

Abstract—Versió en anglès del resum. ....
....
....
....
....
....
....
....
....
....

Index Terms—Versió en anglès de les paraules clau. ....
....

1. Introducció - Context del treball

En els darrers anys, el sector de l'oci té un component que creix de forma exponencial: el videojoc. Aquest fet té moltes causes: van ser un salvavides per la gent durant la pandèmia, cada vegada apareixen més en tot tipus d'anuncis, està creixent la capacitat de fer videojocs més reals o més òptims per poder jugar en dispositius de baix-mig nivell, etc.

Un dels grans videojocs que van ser claus durant la pandèmia va ser l'*Animal Crossing: New Horizons* [1]. Aquest és un videojoc que va sortir just va començar la quarantena global, el 20 de Març del 2020, i gràcies a aquesta situació juntament amb l'estil de videojoc i el temps que portaven els jugadors esperant una nova entrega d'aquesta saga, va acabant sent un exit en vendes [2]. Tant és així que va aconseguir vendre un total de 11,7 milions d'unitats durant els primers 10 dies al mercat a tot el mon. A data de febrer de 2022, es situa en les 37,62 milions d'unitats, havent venut 32,63 milions només en el primer any (fins març de 2021). A més de totes les vendes, va ser nominat a tots els premis de millor joc de l'any 2020 i 2021 [3], guanyant premis com "Japan Game Awards – Game of the Year", "The Game Awards 2020 – Best Family Game", entre d'altres.

Com que aquest és un videojoc que tracta de tenir el teu personatge que va fent la seva vida dia a dia amb diferents tasques, objectius i col·leccionables, arriba un moment en el progrés del joc que el jugador no pot tenir memoritzat l'estat de totes les tasques i catàleg del seu inventari. És per aquest fet que ja fa un temps s'han posat de moda unes aplicacions o programes que t'ajuden a complir aquest objectiu afegint al jugador eines concretes per a cada videojoc. Aquestes eines reben el nom de "Companion". En el cas de l'*Animal Crossing*, una app *Companion* ajudaria l'usuari a mantenir un estat actualit-

- *E-mail de contacte: juancarlos.soriano@autonoma.cat*
- *Menció realitzada: Enginyeria de Computació*
- *Treball tutoritzat per: Felipe Lumbreras Ruiz (departament de Ciències de la Computació)*
- *Curs 2021/22*

zat de la seva col·lecció d'animals i obres d'art que pot recopilar al museu de la seva ciutat, tenir informació detallada de cada animal i quan es poden obtenir (meteorològicament, estació de l'any, hora del dia, etc.).
Aquestes tasques ja les fan aplicacions que s'han publicat, però des d'una perspectiva d'un jugador "veterà", trobo que hi ha moltes tasques que no s'han fet encara i que millorarien molt l'experiencia d'usuari del videojoc. Entre elles, ajudaria a l'usuari a reconèixer si l'obra d'art que un personatge determinat del videojoc intenta vendre és verdadera o és una imitació quasi perfecta. Aquest personatge té un gran nombre d'obres d'art per vendre, però només apareix 1 vegada una o dues setmanes, i de les 5 peces que et ven, normalment només 1 és verdadera.
Aquest projecte doncs implementarà funcions que no s'han implementat fins ara en les apps Companion d'aquest joc amb l'ajuda de mòduls de visió per computació dintre de l'app mòbil.

2. OBJECTIUS DEL PROJECTE

Tal com s'ha comentat breument en l'anterior punt, aquest projecte té com a objectiu aprofitar funcions de la Visió per Computació per ajudar a l'experiència de joc de l'"Animal Crossing". El desenvolupament tindrà una forma seqüencial, a on a cada iteració s'implementarà una funcionalitat diferent, i no es començarà la implementació de les següents funcions fins que l'actual no arribi a un estat de *Minimum viable product*. Aquests subobjectius són:

2.1 Reconeixement de l'obra d'art

El primer objectiu que tindrà aquest projecte serà el reconeixement de l'obra d'art que té el jugador en la pantalla de la seva consola.

Fig. 1. Vista prèvia abans de comprar l'obra "Dama amb un ermini"

En aquest cas el videojoc és per a la consola Nintendo Switch [4], que té una pantalla de 6,2" amb tecnologia LCD. L'usuari haurà de fer una fotografia de la seva pantalla amb el telèfon mòbil, que a través de l'aplicació li mostrarà quina obra és i si està falsificada o no. Aquesta casuística fa que les imatges que es processaran no siguin iguals, canviant aspectes com la il·luminació, la perspectiva i la posició de la pantalla de la consola en la fotografia. A la figura 1 es mostra un exemple de com es veuria dintre de la sessió del joc.

2.2 Reconeixement de l'insecte o peix (OCR)

En aquesta funcionalitat, el programa tractarà de reconèixer quina espècie acaba d'atrapar l'usuari, en una pantalla similar a la que es mostra en la figura 2. Les espècies poden ser peixos, criatures marines i insectes. Quan s'hagi esbrinat de quina espècie es tracta, el programa et dirà si l'usuari ja té aquest animal registrat al museu. Aquesta funcionalitat es basa en processar el text que diu el personatge quan atrapa a qualsevol animal.

Fig. 2. Frase de pesca d'una tonyina

2.3 Reconeixement de l'insecte o peix (Object Recognition)

En aquest punt es treballarà sobre el mateix problema que el punt anterior, amb la diferència que en comptes d'interpretar el missatge de captura, es basarà en la pantalla de l'enciclopèdia que té el jugador. Aquesta funcionalitat estarà pensada en la situació en què un jugador té una espècie en concret que o no surt en l'estació actual entre molts exemples. També és idoni per un jugador que està en un punt molt avançat del progrés del joc, i prefereix registrar tots els espècimens que té directament des de la seva col·lecció.

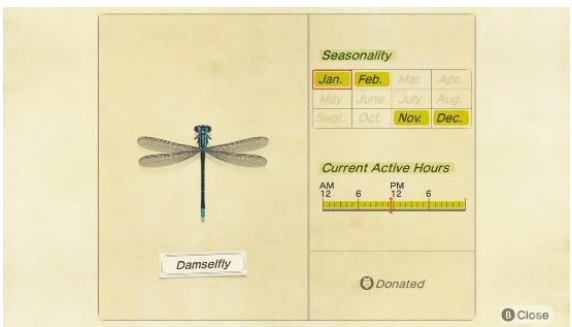


Fig. 3. Mostra de l'entrada de l'enciclopedia d'un insecte

A la pantalla de cada animal figura una imatge la qual es posiciona de manera diferent per a cada animal, apareixent informació addicional tal com els mesos on apareix, a quines hores apareix, tal com es mostra a la figura 3.

2.4 Mòduls VC en un entorn de producció

Tots els mòduls que es volen desenvolupar ho faran des d'un ordinador, però l'objectiu és incorporar-los a un entorn gràfic i interactiu com és una aplicació mòbil. Aquest fet té les dificultats del fet que un telèfon mòbil no té la potència d'un ordinador. A més, quan es desplega algun programa d'aquest estil, no sol tenir cap interfície gràfica, sinó que s'executen directament des de consola o amb alguna comanda especial.

Per tant, un propòsit d'aquest projecte és implementar aquests mòduls en un entorn amigable per a un usuari final en un entorn de producció, fora de l'entorn de desenvolupament.

2.5 Desenvolupament en entorns diferents

Tal com s'ha exposat a l'anterior punt, el dispositiu final on es visualitzarà tot el programa i que servirà com a punt d'introducció de l'input és un dispositiu mòbil, a on s'executarà l'aplicació que es vol desenvolupar. Per això, s'ha de tenir en compte la potència de còmput entre un ordinador i un telèfon mòbil. En termes dels mòduls de visió per computació que es volen implementar en aquest projecte, això es tradueix en el fet que es necessita reduir al màxim l'impacte computacional del programa.
L'aplicació s'executarà en Flutter, component que es profunditzarà més endavant, però la part de backend de l'aplicació, on estarà tota la massa computacional, tindrà 2 variants. El desenvolupament es dividirà en 2 "sistemes":

- **Model entrenat prèviament a l'ordinador i importat a l'aplicació mòbil.** Aquesta variant es basa a entrenar un model pel mòdul de deep learning per la part d'image recognition, per després importar aquest model al codi de l'aplicació perquè només s'encarregui de fer les prediccions d'una forma òptima.
- **Part backend a un entorn Cloud.** Aquesta variant utilitzarà la computació al núvol tal com Google Cloud o AWS per l'execució de tots els mòduls de Visió per Computació. D'aquesta manera l'aplicació només s'encarregarà de fer trucades a funcions i rebre la seva resposta.

Per tant, es durà a terme un desenvolupament en paral·lel de dues aproximacions per a un mateix problema, com és el d'incorporar funcions de visió per computació a un entorn tan poc tractat en la menció de computació com és la d'una aplicació final, ja sigui una app mòbil, un programa gràfic d'escriptori, etc. Però en un entorn de producció, lluny de consola de comandes il·legibles pels usuaris finals.

2.6 Preparació dels datasets

Per últim però no menys important, els models a entrenar ho han de fer sobre un conjunt de dades. En aquest projecte el tema és tant concret i tancat que no hi ha cap data set categoritzat i etiquetat. Per tant, una de les primeres tasques a l'hora de fer la part de desenvolupament del codi i els models serà la creació d'un dataset per a cada model que s'hagi de crear.
Principalment hi haurà un per les obres d'art i un altre per els diferents animals que hi ha en el joc.

3. Estat de l'Art

Els diferents mòduls de Visió per Computació que conformen el projecte són de temàtiques diferents, d'aquesta manera, en aquesta secció es tractaran diferents temes en comptes de centrar-se en un de sol.

3.1 Object Detection

Aquesta tasca es basa en detectar instàncies d'objectes d'una classe determinada en una imatge o un vídeo. Dintre dels mètodes més coneguts i utilitzats es poden trobar 2 tipus:

- **Mètodes "One-Stage":** Aquests mètodes es basen en prioritzar la velocitat de la inferència per tal de tenir resultats més ràpids, sacrificant la precisió del mètode. Entre els mètodes que entren dintre d'aquest rang es troben el YOLO [5], SSD [6] i RetinaNet [7].
- **Mètodes "Two-**Stage": Aquests mètodes es basen en donar més passos ja sigui de transformacions a les dades o passades extres dels models, el que fa que augmentin la precisió de la detecció per tenir uns millors resultats finals. Exemples de mètodes d'aquest tipus són: Faster R-CNN [8], Mask R-CNN [9] i Cascade R-CNN [10].

A part d'aquests mètodes més coneguts, n'hi ha un que recentment s'està consolidant com el mètode que millors resultats aconsegueix en aquest camp. Aquest és el SwinV2-G [11], que es basa en escalar la capacitat i la resolució màxima possible amb la que es pot entrenar el model Swin Transformer. Amb aquesta nova versió s'ha aconseguit encapçalar les puntuacions dels datasets més populars en aquest camp (ADE20K [12], COCO minival i COCO test-dev [13], ImageNet [14], entre d'altres).

3.2 OCR

En aquest camp hi ha diferents aproximacions que intenten interpretar el text que conté la imatge a processar. Hi ha dues fases per tal de llegir un text en una imatge. El primer pas és detectar el text en la imatge. Entre aquestes tècniques es troben la "sliding window" que es basa en anar desplaçant una finestra per tota la imatge. Un exemple de mètode que utilitza aquesta tècnica és una CNN [15]. D'altres tècniques que existeixen són les "single shot" com la YOLO [16] a la qual es passa només una vegada la imatge i detecta el text en una regió determinada i per últim l'"EAST" [17], un mètode molt avançat i precís que fins i tot pot detectar text en un vídeo a 13fps en qualitat 720p.

L'altre pas de l'OCR és reconèixer el text que hi ha a la regió. Per això es fan servir Xarxes Neuronals tals com la CRNN [18], una combinació de xarxa convunacional i xarxa recurrent.

3.3 Aplicacions Mòbils

Aquest és un sector que evoluciona molt ràpid, on surten noves tecnologies cada poc temps i les que existeixen s'actualitzen de forma molt freqüent si no volen estancarse i desaparèixer. Entre aquestes plataformes [19] s'ha de tenir en compte el factor més important: el SO. Les que més s'utilitzen actualment en el món d'Android són:

- **JavaScript**: Aquest llenguatge de programació va ser un dels més utilitzats gràcies al gran nombre de frameworks que l'utilitzen com poden ser Angular, React, etc. Destaca la seva facilitat per crear una estructura de full-stack molt còmodament. En el cas de React, també ens permetrà programar apps per iOS.
- **Java**: D'igual manera que JavaScript, aquest és

un llenguatge molt polivalent que també permet fer una estructura full-stack.
- **Kotlin**: Al tenir suport natiu d'Android, Kotlin és la millor opció si es vol desenvolupar una aplicació en Android i es volen aprofitar totes les característiques tant del sistema operatiu com del telèfon mòbil.

En canvi, si es vol desenvolupar una app en iOS, les millors opcions són:
- **Swift**: És el llenguatge oficial d'iOS. Per aquest fet, té suport directe d'Apple, una gran avantatge al ser un sistema operatiu tan tancat. Al ser natiu, utilitza totes les funcions possibles del telèfon mòbil.

Fora de l'àmbit natiu, el framework que està creixent d'una forma exponencial és Flutter [20]. Aquest framework que ha creat Google fa servir el llenguatge Dart. La principal avantatge d'aquest sistema és que d'un mateix codi es pot crear una aplicació d'Android, d'iOS, d'escriptori de Windows i fins i tot una pàgina web.

4. METODOLOGIA

En aquest projecte s'utilitzaran diferents mètodes i eines per tal de portar una correcta gestió i desenvolupament de tot el conjunt de fases del projecte.

Pel que fa a metodologies, es farà servir una metodologia àgil tal com és SCRUM [21], amb una configuració d'sprints. La seva durada estarà determinada per les diferents entregues que hi hagi en el transcurs del TFG, amb l'excepció d'alguna funcionalitat gran que tingui moltes subtasques o períodes molt grans entre lliuraments. Això fa que els sprints tinguin una durada d'entre 2-3 setmanes, una durada òptima per tal de treballar amb aquesta metodologia.

Pel que fa a les eines que s'utilitzaran, aquestes seran:
- **Jira:** S'utilitzarà aquesta eina per fer la gestió i el seguiment de la metodologia SCRUM. És una eina pensada per a projectes en un entorn col·laboratiu, però aquest projecte és una oportunitat ideal per utilitzar-la en un entorn quasi real. A més, aquesta eina ens permet generar una sèrie d'informes dels sprints que ajudaran a tenir una millor perspectiva del progrés del projecte [22].
- **Notion:** Aquesta eina és un programa de gestió de projectes i per a prendre notes, entre d'altres [23]. En general, és una eina molt versàtil on el límit el marca la teva imaginació. El seu paper en aquest projecte és el d'elaborar panells i pàgines per les diferents entregues i sprints, a on s'aniran documentant totes les fases del projecte. Aquest és un software que permet un entorn col·laboratiu d'igual manera que el Jira, però també es pot utilitzar per a un ús personal, depenent de com es configurin els panells. També té una funcionalitat d'exportar el teu espai en forma d'una pàgina web perquè altres usuaris la puguin consultar a través d'un link.
- **Github:** S'utilitzarà aquesta eina de control de versions [24] per tenir un repositori de tots els materials del projecte, tant en la fase de documentació com la de desenvolupament. Amb aquesta web es construirà el dossier final del TFG que s'ha d'entregar a l'entrega final.
- **Toggl:** Aquest programa s'utilitzarà principalment de forma personal, sense repercussió en el projecte. Toggl és un software que s'encarrega de comptar el temps que l'usuari consumeix en les tasques en les quals treballa [25]. Es poden agrupar les diferents tasques en projectes, que en aquest cas seran els diferents sprints o tasques grans del TFG. La funció principal serà tenir una aproximació real del temps que consumeix cada tasca del projecte per registar-les al Jira i per poder predir aproximadament els temps estimats en els següents sprints.

La majoria d'aquestes eines i metodologies estàn pensades per aplicar-les en projectes col·laboratius amb un equip de persones. En aquest cas, el projecte només es desenvoluparà per una persona, però és una molt bona oportunitat per tenir un primer contacte en un àmbit de projecte real.

5. PLANIFICACIÓ DEL PROJECTE

Les diferents tasques del projecte es basen en la redacció de l'informe del treball, la documentació dels diferents temes que es tractaran en el seu transcurs i les tasques de desenvolupament de les diferents funcionalitats, tant de l'aplicació com dels mòduls de Visió per Computació.

Per tant, s'haurà de tenir en compte que en algunes etapes del projecte es faran algunes d'aquestes tasques de manera simultània.

D'una manera molt preliminar i sense tenir cap referència del temps de les tasques la planificació inicial seria aquesta:

Data	Output desitjat
06-03-22	Primer Informe i documentació tècniques de VC inicials
20-03-22	Primera versió mòdul image recognition
27-03-22	Millores mòdul Img Rec i primera versió de l'App
10-04-22	Informe progrés I i millores mòdul Img Rec
24-04-22	Img Rec acabat, primera versió OCR i millora de l'aplicació
08-05-22	OCR acabat
22-05-22	Informe progrés II i primera versió Img Rec animals
05-06-22	Desenvolupament app i Img Rec animals acabat
12-06-22	Proposta Informe Final
26-06-22	Proposta Presentació
27-06-22	Lliurament Dossier

Taula 1. Planificació inicial del proyecte

Per aprofundir més en aquesta planificació, consultar a l'Apèndix 1 el Diagrama de Gantt corresponent.

Cal aclarir que aquesta planificació és aproximada i pot

canviar segons el transcurs del projecte i del seu progrés. També existeix el fet que la part *backend* de l'aplicació es desenvolupara tant d'una forma "local" amb un model prèviament entrenat i importat al telèfon móvil i una altra versió que tota la potència de còmput es fa a un servei Cloud.

BIBLIOGRAFIA

[1] "Animal Crossing: New Horizons - Wikipedia". Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Animal_Crossing:_New_Horizons (accedit el 17 de febrer de 2022).

[2] "Unit Sales of Animal Crossing: New Horizons worldwide as of December 2021". Statista. https://www.statista.com/statistics/1112631/animal-crossing-new-horizons-sales/ (accedit el 4 de març de 2022).

[3] "Animal Crossing awards and nominations". IMDb https://www.imdb.com/title/tt10476972/awards/ (accedit el 4 de març de 2022).

[4] "Technical Specs - Nintendo Switch™ - System hardware, console specs - Nintendo - Official Site". Nintendo. https://www.nintendo.com/switch/tech-specs/#switch-section (accedit el 18 de febrer de 2022).

[5] Chien-Yao Wang, Alexey Bochkovskiy, Hong-Yuan Mark Liao, "YOLOv4: : Optimal speed and accuracy of object detection" [Online]. Disponible: https://arxiv.org/abs/2004.10934

[6] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg, "SSD: Single Shot MultiBox Detector" [Online]. Disponible: https://arxiv.org/abs/1512.02325

[7] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár, "Focal Loss for Dense Object Detection" [Online]. Disponible: https://arxiv.org/abs/1708.02002

[8] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks" [Online]. Disponible: https://arxiv.org/abs/1506.01497

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick, "Mask R-CNN" [Online]. Disponible: https://arxiv.org/abs/1703.06870

[10] Zhaowei Cai, Nuno Vasconcelos, "Cascade R-CNN: Delving into High Quality Object Detection" [Online]. Disponible: https://arxiv.org/abs/1712.00726

[11] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, Baining Guo, "Swin Transformer V2: Scaling Up Capacity and Resolution" [Online]. Disponible: https://arxiv.org/abs/2111.09883

[12] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso and Antonio Torralba, "ADE20K Dataset". https://groups.csail.mit.edu/vision/datasets/ADE20K/ (accedit el 25 de febrer de 2022).

[13] Tsung-Yi Lin, Genevieve Patterson, Matteo R. Ronchi, Yin Cui, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, Larry Zitnick, Piotr Dollár, "COCO Dataset". https://cocodataset.org/#home (accedit el 25 de febrer de 2022).

[14] Li Fei-Fei, Jia Deng, Olga Russakovsky, Alex Berg, Kai Li, "ImageNet Dataset". https://www.image-net.org/index.php (accedit el 25 de febrer de 2022).

[15] Mikey Taylor, "Computer Vision with Convolutional Neural Networks". https://medium.com/swlh/computer-vision-with-convolutional-neural-networks-22f06360cac9 (accedit el 27 de febrer de 2022).

[16] Grace Karimi, "Introduction to YOLO Algorithm for Object Detection", https://www.section.io/engineering-education/introduction-to-yolo-algorithm-for-object-detection/ (accedit el 27 de febrer de 2022).

[17] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, Jiajun Liang, "EAST: An Efficient and Accurate Scene Text Detector" [Online]. Disponible: https://arxiv.org/abs/1704.03155v2

[18] Chandra Churh Chatterjee, "An Approach Towards Convolutional Recurrent Neural Network", https://towardsdatascience.com/an-approach-towards-convolutional-recurrent-neural-networks-a2e6ce722b19 (accedit el 27 de febrer de 2022).

[19] Javinpaul, "Top 5 Programming languages for Mobile App Development in 2022", Medium. https://medium.com/javarevisited/top-5-programming-languages-for-mobile-app-development-in-2021-19a1778195b8 (accedit el 27 de febrer de 2022).

[20] "What is Scrum?" Scrum.org. https://www.scrum.org/resources/what-is-scrum (accedit el 20 de febrer de 2022).

[21] "Jira | Issue & Project Tracking Software | Atlassian". Atlassian. https://www.atlassian.com/software/jira (accedit el 20 de febrer de 2022).

[22] "Flutter Documentation". Flutter - Build apps for any screen. https://flutter.dev/learn (accedit el 17 de febrer de 2022).

[23] "Notion - One workspace. Every team". Notion. https://www.notion.so/product (accedit el 20 de febrer de 2022).

[24] "Build software better, together". GitHub. https://github.com/about (accedit el 20 de febrer de 2022).

[25] "Meet Toggl. Three producs; One mission". https://toggl.com (accedit el 20 de febrer de 2022).

[26]

[27]

[28]

[29]

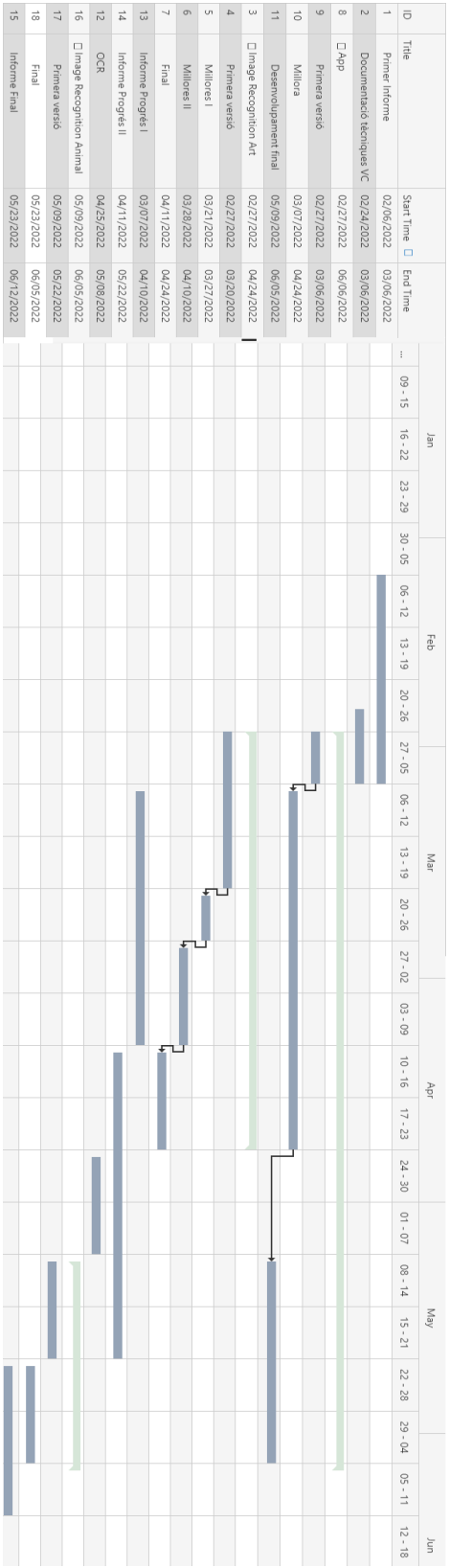
[30]

[31]

[32] Etc.

APÈNDIX

A1. DIAGRAMA DE GANTT INICIAL



ID	Title	Start Time	End Time
1	Primer informe	02/06/2022	03/06/2022
2	Documentació tècniques VC	02/24/2022	03/06/2022
8	App	02/27/2022	06/06/2022
9	Primera versió	02/27/2022	03/06/2022
10	Millora	03/07/2022	04/24/2022
11	Desenvolupament final	05/09/2022	06/05/2022
3	Image Recognition Art	02/27/2022	04/24/2022
4	Primera versió	02/27/2022	03/20/2022
5	Millores I	03/21/2022	03/27/2022
6	Millores II	03/28/2022	04/10/2022
7	Final	04/11/2022	04/24/2022
13	Informe Progrés I	03/07/2022	04/10/2022
14	Informe Progrés II	04/11/2022	05/22/2022
12	OCR	04/25/2022	05/08/2022
16	Image Recognition Animal	05/09/2022	06/05/2022
17	Primera versió	05/09/2022	05/22/2022
18	Final	05/23/2022	06/05/2022
15	Informe Final	05/23/2022	06/12/2022