Universitat Autònoma de Barcelona

Dipòsit digital de documents de la UAB

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

Cañas Martin, Ivan; Martinez Carrascal, Juan Antonio, dir. Entorn d'analítiques acadèmiques. 2023. (958 Enginyeria Informàtica)

This version is available at https://ddd.uab.cat/record/272814

under the terms of the BY-NC-ND license

Entorn d'analítiques acadèmiques

Ivan Cañas Martin

Resum– El següent article tracta de la implementació d'un entorn d'analítiques acadèmiques en format aplicació web. El projecte engloba dos temes principalment, la programació d'aplicacions web amb la creació d'un servidor web funcional, la implementació d'un client per introduir, demanar i visualitzar dades del servidor i una lleu introducció al tractament de dades amb l'estudi i desenvolupament d'una eina de predicció.

Paraules clau– Aplicació web, dades acadèmiques, visualització de dades, Python, Flask, Pandas, JavaScript, React, ciència de dades, sklearn, models de predicció, Decission Tree

Abstract– The following article is about the implementation of an academic analytics environment in web application format. The project covers two main topics, the programming process of a web application with the creation of a functional web server, implementation of a client to enter, request and view data from the server and a slight introduction to data science with the study and development of a prediction tool.

Keywords– Web App, Academic Data, Data Visualization, Python, Flask, Pandas, JavaScript, React, Data Science, sklearn, Prediction Models, Decision Tree

1 Introducció

1.1 Motivació del projecte

AQUEST document recull tota la informació del treball fet sobre la creació d'un entorn de dades analítiques acadèmiques, un treball que neix a partir de les situacions socials viscudes en la recent actualitat amb la covid-19, en la qual tots els estudiants tant universitaris com d'estudis obligatoris, hem estat una mitja d'un any i mig cursant estudis de forma no presencial. Encara que la normalitat ha tornat a la metodologia de classes presencials aquest esdeveniment ha impulsat l'ús i creació d'estudis en línia. Tot i que es coordina de la millor manera possible també es perd una mica el tracte personal entre professor i alumne on el professor pot veure fàcilment degut al treball constant a classe quins alumnes porten millor l'assignatura i quins tenen més dificultats o bé falta d'interès, poden fer-se així una idea general de les seves classes.

Aquest treball proposa la creació d'una eina que sigui d'ajuda als professors per poder identificar la situació de la seva classe dia a dia, fent que amb aquesta eina es puguin consultar les dades dels alumnes del seu curs per poder visualitzar un conjunt de dades rellevants sobre el curs com per exemple: qui ha entregat treballs, quina nota mitjana porten i un seguit més d'estadístiques dels alumnes.

1.2 Objectius

Com s'ha comentat prèviament al resum de l'article, l'objectiu principal del projecte és el de crear una eina, una aplicació web, que té com a propòsit permetre analitzar el comportament acadèmic dels estudiants en un entorn de docència virtual per poder millorar la visió i el seguiment del professor respecte als seus alumnes. Pel que el principal objectiu és el de desenvolupar una eina de control de les dades tant acadèmiques com personals dels alumnes a temps real.

Com la idea principal del projecte implica poder ajudar als alumnes, la visualització de dades ha de poder ser accessible tant per alumnes com per professors. Aquest punt de vista implica la creació d'un mecanisme que permeti identificar a l'usuari per així poder donar una visualitzaci de ´les dades personalitzada tant a alumnes com professors. Això permetrà poder ajudar també a l'estudiant amb una eina per fer-se un autoseguiment.

Per millorar la qualitat del seguiment acadèmic als alumnes es planteja la incorporació d'un model de predicció al projecte, la funció d'aquest model serà la de poder predir quins alumnes estan en risc de suspendre segons factors tant acadèmics com socials i poder ajudar a l'alumne amb temps. S'ha decidit treballar amb una base de dades ja defi-

• E-mail de contacte: 1417496@uab.cat
• Menció realitzada: Enginyeria en Tecnologies de la Informació
• Treball tutoritzat per: Juan Antonio Martinez Carrascal
• Curs 2022/23

nida i existent i no amb una amb ús continu que es va actualitzant a temps real, per així poder fer proves del model de predicció. El desenvolupament comptarà amb una eina de selecció de temps per simular el que seria un comportament real amb actualitzacions de dades.

Un cop exposats els objectius es poden veure els requisits funcionals de l'aplicació:

- Creació d'una aplicació executable i interactiva.
- Creació d'un entorn d'analítiques acadèmiques general, amb la capacitat d'integrar i visualitzar les dades provinents d'una base de dades oberta.
- Creació d'un sistema de login per accedir-hi a l'aplicació i identificar a l'usuari per personalitzar l'ús de l'aplicació. Això permet distingir entre alumnes i professors.
- Filtrar els cursos i assignatures que l'usuari pot seleccionar a l'aplicació segons si ha estat matriculat o ha impartit.
- Creació d'una pantalla pel tipus d'usuari "professor"on es mostri les dades dels seus alumnes a l'assignatura i curs seleccionats, les seves entregues fetes i dades generals del curs com persones matriculades, mitjana de la classe i d'altres que puguin ser rellevants pel curs.
- Creació d'una pantalla pel tipus d'usuari "estudiant", on es mostri les dades de l'estudiant al curs i assignatura seleccionats i una comparativa amb la mitjana de la classe i les dades de la resta d'assignatures que està cursant.
- Creació un widget que permet seleccionar fins a quina data del curs volem actualitzar la informació que es mostra.
- Creació d'una barra d'aplicació que permet tant tornar a l'inici com tancar sessió.

  Tot i no ser un requisit funcional estrictament, la millora en la qualitat del producte associada al possible desenvolupament dels següents punts fa que l'estudi de les seves implementacions es tingui com a requisit.
- Estudiar la possibilitat de crear una eina que implementi un algorisme de predicció, per preveure si els estudiants és troben en risc de suspendre o no.
- Estudiar la implementació de l'aplicació a un entorn exportable per permetre el seu ús a diferents sistemes operatius i sense necessitat de descarregar manualment les possibles dependències.

2 ESTAT DEL ART

Està clar que avui en dia al segle de la informació hi ha tota classe d'aplicacions i tot tipus de tractaments de dades i a l'ambient acadèmic no és una excepció.

En el cas d'aplicacions per mostrar les dades i analítiques d'alumnes podem trobar diversitat d'opcions, totes les opcions normalment proporcionen unes dades generals com: quantitat d'alumnes, nota mitja de l'assignatura, nota de cada estudiant, etc. D'una altra banda, cadascuna de les opcions a l'àmplia varietat que hi ha proporcionen dades més específiques on creuen que hi ha més importància, o dependent també de l'àmbit dels estudis hi ha dades úniques per aquests estudis. Per exemple un cas de dades úniques del tipus d'estudi podria ser la separació de notes mitjanes dels estudiants en listening, reading i writing a una escola d'anglès.

Fent una cerca d'aplicacions de l'estil s'han trobat 3 que ens han servit com a exemples per la creació de la nostra. La motivació per crear una aplicació pròpia d'aquest tipus és per la dificultat que hi ha en l'ús d'altres, ja bé o pels seus alts preus o per la impossibilitat d'obtenir-los. Aquestes 3 són: OUAnalyse [1],BoldBI[2] i SchoolAnalytics [3].

- OUAnalyse creat per OULA (Open University Learning Analytics), aquesta és l'aplicació que més es tindrà en compte a l'hora de crear la nostra, ja que utilitzem una base de dades estàtica que ens proporcionen, un dataset. Aquesta eina no té cap tipus de distribució actua només com a exemple del que és l'anàlisi de dades acadèmiques. Es troba enfocat una mica a estudis en línia, perquè no existeixen dades de presencialitat a classe, en canvi, té dades del temps de connexió dels alumnes als materials del campus virtual.
- BoldBI, aquesta aplicació a diferència de l'anterior sí que es distribueix i es pot personalitzar segons el comprador, al ser software de pagament. Té una prova gratuïta d'un mes on t'has de registrar primerament i després de la prova gratuïta el seu cost és de 495 dollars, uns 508,21 euros, a la seva versió estàndard. BoldBI té una àmplia gamma de dashboards al no només presentar serveis al món de l'educació, però focalitzant l'educació trobem un dashboard que representa tota una escola, és a dir hi ha una diferenciació entre estudiants de diferents anys, hi ha mitjanes de totes les assignatures a l'hora i trobem que tots els estudiants fan totes les matèries però en diferents anys acadèmics.
- SchoolAnalytics, és una altra aplicació que es distribueix, però en aquest cas si vols provar-la en comptes d'una prova gratuïta compta amb una prova amb un dataset fixat. El seu preu són 295 libres mensualment, uns 335,78 euros. En aquest cas SchoolAnalytics als seus dashboards filtra els estudiants per anys i té dades generals molt interessants com el nombre de persones absents constants, el nombre total de llengües diferents que parlen els alumnes o un mapa de les seves localitats.

Com s'ha mencionat s'ha decidit utilitzar un model semblant al d'OULA pel fet d'estar utilitzant el dataset[4] que ens proporcionen com a base de dades, tot i així també aplicarem a la nostra aplicació elements que hem vist a les altres opcions, com a la vista de l'usuari posar-hi les seves diferents mitjanes a assignatures del mateix curs que hagi seleccionat per fer-li un resum de com va als estudis en general i a la vista del professor una targeta que mostri el nombre d'alumnes que es connecten menys temps de la mitja.

3 METODOLOGIA

La metodologia que s'ha utilitzat a aquest treball és una de caràcter agile[8]. El desenvolupament agile consisteix en un desenvolupament iteratiu on a cada iteració el desenvolupament passa per 5 fases, aquestes són:

- Anàlisis dels requisits i planificació.
- Disseny del qual es farà a alt nivell.
- Codificació i proves de qualitat.
- Documentació dels objectius aconseguits i errors trobats.
- Mostra al professor de les implementacions aconseguides en aquesta iteració.

Aquest projecte es dividirà en 7 iteracions a les quals anomenarem sprints. Cada sprint durarà dues setmanes i tindrà un seguit de tasques associades. Aquests sprints es poden veure a l'apèndix 1.

Dintre de la metodologia agile es farà servir Scrumban[9], una metodologia que combina dues altres, la metodologia Scrum i la Kanban. S'ha utilitzat aquesta per ser la més adient a aquest treball, a l'utilitzar iteracions/sprints, però sense l'organització jeràrquica de l'equip que implica Scrum i per part de Kanban una gestió de tasques dividides en targetes explicatives, que per gestionar-les s'aniran arrossegant entre columnes d'una taula que determinen l'estat de la tasca.

Pel tractament i seguiment de les tasques a realitzar s'ha fet servir Jira[10], un software privat creat per la gestió de projectes de manera agile, permetent separar les tasques en Sprints i posicionar les tasques segons prioritats, si n'estan per començar, en procés o acabades. Tot i ser un producte privat ofereix el seu software de manera gratuïta a persones per un ús no comercial.

Per la creació de mockups[11] s'ha emprat una eina en línia també d'ús gratuït anomenada Moqups . Aquests Mockups són un dels passos previs a la creació de qualsevol mena d'aplicació amb un component visual, que permet materialitzar visualment les ideés i les funcionalitats de l'aplicació i millora l'estructuració i el desenvolupament d'aquesta. Els Mockups es poden veure a l'apèndix 2.

Les dades acadèmiques que s'han utilitzat són les que ens dona l'Open University Learning Analytics que de manera també gratuïta ens proporciona un conjunt de dades, dataset, dels alumnes, assignatures i entregues. Això fa que la nostra base de dades sigui fixa i no s'utilitzen dades recollides periòdicament, és a dir, no és una base de dades viva sinó que simularem el pas del temps a través d'un selector de data en forma de barra de progrés.

Per la codificació de l'aplicació web s'ha utilitzat l'eina Visual Studio Code en un entorn Linux. Visual Studio Code és un IDE desenvolupat per Microsoft d'ús gratuït, s'ha escollit aquest per ser un dels IDE més utilitzats avui dia pel fet que, és fàcil d'utilitzar, té molts plugins amb diverses utilitats i funciona molt bé per diferents llenguatges de programació alhora, com és el cas d'aquest treball en el qual s'utilitzen diferents llenguatges per implementar el Backend (Python) i per implementar el Frontend (JavaScript). Com s'acaba de veure es treballarà amb dos llenguatges, aquests els farem servir a partir de dos frameworks. Per Python serà Flask[5] un framework que enfocat a la creació simple d'aplicacions web, també s'utilitzarà pandas[6] una llibreria utilitzada pel tractament de dades, i per JavaScript serà React[7] un framework que es focalitza a la creació d'interfícies d'usuari.

4 DESENVOLUPAMENT

A continuació s'esmentarà el desenvolupament fet fins data d'avui amb un llenguatge amb un caràcter més tècnic.

Primer de tot cal informar que el desenvolupament s'està aplicant sobre un sistema operatiu basat en Linux i de distribució gratuïta com és Ubuntu.

Del projecte podem dividir el seu desenvolupament en dos subdesenvolupaments principals, aquests són, per una banda, el desenvolupament de l'aplicació web i per l'altre l'estudi i generació d'un model de predicció. Començant amb la creació d'una aplicació web tenim que el seu desenvolupament com la gran majoria d'aplicacions d'aquest estil té una estructura basada en el model Client-Servidor. Aquest model implica que l'emmagatzemament de dades necessàries i el seu tractament es trobaran a un servidor que proporcionarà les dades a l'aplicació client per poder mostrar el contingut sobre la pantalla del client. El model Client-Servidor s'ha implementat sobre la mateixa màquina actualment, utilitzant diferents ports de la màquina per donar el servei i per rebre-ho.

4.1 Servidor

El servidor o backend de l'aplicació s'ha escollit implementar-lo en llenguatge de programació Python pel fet que l'objectiu principal del servidor és la lectura i tractament de dades, i sobre aquest llenguatge tenim la llibreria Pandas que és àmpliament utilitzada pel processament de dades.

4.1.1 Preparació de l'entorn

La part del servidor com ja s'ha comentat s'ha implementat en Python i utilitzen un parrell de llibreries externes, això provoca una dependència del projecte sobre aquestes llibreries que s'ha de solucionar. Primer per permetre una execució segura del servidor es fa servir un entorn virtual[12], l'entorn virtual permet la instal·lació de llibreries només en aquest projecte no en tota la màquina i això protegeix al projecte de possibles incompatibilitats amb llibreries que estiguessin instal·lades prèviament, l'entorn virtual s'ha creat utilitzant una llibreria d'Ubuntu anomenada "python-virtualenv". Finalment, s'han descarregat fent servir el gestor de paquets 'pip' utilitzant les comandes:

- "pip3 install pandas"
- "pip3 install Flask"

4.1.2 Base de dades

S'ha optat per la utilització d'una base de dades acadèmica oberta, la base de dades en qüestió és el dataset d'OULA, on hi han les dades de diferents cursos, anys i estudiants ja

definits. Pel que fa a la implementació de l'obtenció de dades no varia gaire del comportament habitual que implicaria fer ús d'una base de dades no predefinida com és la que tenim. La diferència principal es troba a on i com es guarda la base de dades, ja que en el nostre cas al ser una base de dades predefinida formada per diversos arxius csv relacionats entre ells com taules SQL es troba emmagatzemada en una carpeta al servidor que no s'actualitza. Només es fan operacions de lectura.

4.1.3 Inici del servidor

Flask permet la creació de servidors de manera ràpida i poc complexa, aquest fet permetrà contestar a les trucades al servidor que farà el client i gestionar-les segons el tipus d'informació que demana. Flask funciona fent servir la definició de rutes per gestionar les peticions que li arriben, cada ruta té tot seguit al codi una funció que dicta que farà el servidor per aquesta petició i quines dades retornarà. Al principi es genera l'objecte de Flask servidor, després es declaren totes les rutes possibles al servidor i finalment es declara el main de l'arxiu python, on per evitar la lectura constant dels arxius csv que podria provocar errors, tots es llegeixen abans d'executar el servidor i es guarden com a objectes Pandas a una classe creada només per guardar-hi la base de dades. Cada atribut de la classe és un objecte Pandas d'un arxiu csv i els seus únics mètodes són per obtenir aquests atributs, i per últim s'executa el servidor perquè ocupi un port de l'ordinador i comenci a escoltar peticions.

4.1.4 Funcionament de les rutes

Les rutes declarades al desenvolupament fet fins ara contesten a les peticions d'identificació d'usuaris, identificació dels cursos als quals està matriculat en cas de ser un alumne, identificació dels cursos que imparteix en cas de ser professor, les dades acadèmiques de tots els alumnes i de la classe en general i finalment les dades acadèmiques de l'alumne. Les rutes són crides al servidor que fa servir el client per abastir-se d'informació i totes aquestes crides manen variables al servidor perquè pugui fer el processament de les dades pertinents. Un exemple seria la identificació d'usuari que envia el nom i contrasenya que l'usuari a escrit per comprovar si es troben a la base de dades.

4.2 Client

El client o frontend de l'aplicació s'ha escollit implementarlo en llenguatge de programació JavaScript, més concretament amb un framework anomenat React que és un del més estesos i amb més popularitat a la creació d'aplicacions web. Aquest framework permet la creació de les vistes de l'aplicació fent servir objectes, el qual vol dir que implementem programació orientada a objectes, en comptes de crear una pàgina html per a cada vista que volem a l'aplicació es generen objectes que es mostren, o no o es mostren modificats dependent del comportament programat i desitjat.

4.2.1 Preparació de l'entorn

React al ser una llibreria de JavaScript, necessita d'aquest per funcionar i el primer pas per poder crear el frontend és la descàrrega de NodeJs, una eina que permet la creació de llocs web dinàmics de manera molt eficient. En entorns Linux, com és el cas del projecte la seva instal·lació és fàcil, des de la pàgina web de NodeJs[13] es pot descarregar gratuïtament i amb qualsevol gestor de paquets Debian instal·lar-ho, però en entorns Windows es complica i per aquest motiu un dels objectius del projecte és estudiar la possibilitat d'encapsular el projecte amb la intenció de ferho fàcilment exportable tot i no ser un requisit funcional.

Un dels motius pels quals React és una eina molt estesa és la facilitat que proporciona per crear un projecte, amb el gestor de dependències npm es pot crear una aplicació base i les seves dependències dels objectes que utilitza React amb la comanda:

- "npx create-react-app client"

Aquesta execució ens crea una aplicació amb un index.js, un index.html i un index.css, com l'aplicació està formada per objectes que es representen amb codi html, l'arxiu index.html ha sigut eliminat per evitar arxius innecessaris.

Per acabar la configuració hem de definir-li al nostre client que quan faci crides al servidor aquest es trobarà en l'entorn local a un cert port, per fer-ho s'ha declarat un proxy amb el valor localhost/5000, port de la nostra màquina que executa el servidor, aquest port s'ha decidit per ser el port per defecte, però es podria utilitzar un altre qualsevol, a l'arxiu package.json que s'ha generat a l'hora de crear l'aplicació client.

El client es pot executar fent servir:

- "npm start"

4.2.2 Memòria cau

Per poder utilitzar dades i donar-lis una pertinença quan deixem d'utilitzar alguns objectes i no perdre dades que són importants tenir pel bon funcionament de la pàgina i que poder tornar enrere sense perdre totes les dades dels objectes de la vista actual s'implementa l'ús de la memòria cau. Permet guardar dades sense que persisteixin un cop tancada la pàgina.

4.2.3 Components

El projecte està programat de manera orientada a objectes, aquests objectes es diran components, dintre de l'aplicació client s'ha creat una carpeta per guardar tots els components que s'implementaran. A continuació es farà una descripció de les funcionalitats dels components dissenyats sense entrar en detalls molt tècnics.

HomePage

Aquest component és un dels més bàsics la seva funció és la de comprovar si hi ha algun usuari logejat i en cas d'existir un usuari logejat proporciona la pàgina de Login perquè es pugui identificar, en canvi, si es troba un usuari logejat es passa a mostrar la pàgina bàsica d'usuari.

LogPage

El component LogPage com el seu nom indica és l'encarregat de mostrar el formulari d'inici de sessió de l'usuari,

el patró d'aquesta vista és molt estàndard demanant només nom i contrasenya de l'usuari i en cas de no ser correcte mostra un missatge d'error per avisar a l'usuari que no s'ha trobat cap usuari amb aquesta combinació de nom i contrasenya.

BaseUserPage

Un cop ja tenim la informació de quin usuari ha iniciat sessió el primer que es veu serà la pàgina bàsica d'usuari, aquesta com sap qui és l'usuari, però no quin curs vol consultar només consta d'un títol que diu si és estudiant o professor, un desplegable amb quins cursos està cursant o impartint, dependent del tipus d'usuari, i un panell explicatiu de què un cop seleccionat el curs es mostrarà la informació.

TeacherPage

Per la representació de dades dels usuaris professors s'està desenvolupant el component TeacherPage, aquest actualment només mostra per pantalla una taula amb les dades acadèmiques de cada estudiant que hi ha al curs i implementa els widgets AppBarr i Slider.

StudentPage

Per la representació de dades dels usuaris alumnes s'està desenvolupant el component StudentPage, aquest actualment només mostra per pantalla una taula amb les dades acadèmiques de l'estudiant sobre el curs seleccionat i implementa els widgets AppBarr i Slider.

StudentsTable

Aquest component s'encarrega de mostrar les dades tretes del servidor sobre els alumnes en un cert temps del curs que ve donat pel component Slider, a la vista de StudentPage o TeacherPage. Mostra dades fixes com en quina regió viuen i número de crèdits matriculats, dades dinàmiques com quines tasques s'han fet fins a la data desitjada i quines ha fet cada alumne i finalment si aprovaran o no.

Assessment

Petit component que s'encarrega de facilitar la mostra de quines tasques han sigut fetes per l'alumne i quines no a la taula d'estudiants.

AppBar

Un aspecte bàsic de la usabilitat de l'aplicació passa per poder recórrer les diferents opcions que proporciona aquesta i per poder desconnectar-se quan vulguin, l'AppBar implementa aquestes dues funcions proporcionant tant un botó per poder tornar a la pàgina base d'usuari com un botó que permeti fer logout, sortir de la sessió d'usuari.

Slider

El component Slider és una barra horitzontal dinàmica que té com a funció la selecció del dia del curs que es vol mirar de manera àgil, fent servir el cursor es pot moure de manera horitzontal un punt que determina fins a quin dia mirar i va mostrant a un costat a quin número pertany el percentatge de la barra que estàs marcant, per facilitar el seguiment.

4.3 Model de predicció

A continuació s'esmenta el segon desenvolupament principal que ha sigut la implementació d'una eina predictiva per l'aplicació web que funcioni amb les dades dels alumnes obtingudes segons el temps seleccionat del curs. Per portar a terme aquest desenvolupament primer s'ha fet un estudi dels models predictius, s'ha seleccionat un i finalment s'ha generat al servidor el seu funcionament.

4.3.1 Estudi de possibilitats

Dintre del món de la predicció hi ha una àmplia varietat d'opcions i això passa pel que es coneix com a machine learning[14]. Els tres principals mètodes dintre d'aquest àmbit i els més utilitzats són:

- Classificació: La classificació tracta de segons una sèrie de paràmetres i normalment utilitzant un historial previ poder classificar els objectes nous que es volen estudiar en algun d'uns grups predefinits.
- Regressió: La regressió consisteix en igual que l'anterior utilitzant una sèrie de paràmetres dels objectes i normalment un historial previ tractar de prevenir el valor específic d'un atribut de l'objecte
- Agrupament: Al contrari que els anteriors aquest procés no utilitza normalment un historial, ja que el seu funcionament principal és el d'agrupar els objectes en un número variable o fix de grups no predefinits a diferència del classificador.

Després de veure les característiques i usos de cada un dels anteriors s'ha decidit utilitzar el mètode de classificació pel nostre model de predicció, ja que el nostre objectiu principal és el de fer seguiment dels alumnes i poder predir si un alumne té risc de suspendre l'assignatura per poder donar-li ajuda amb temps. Amb el model de classificació podem classificar els alumnes dintre de dos grups principals, que serien si suspendrà o no, dintre del suspens incloem deixar els estudis, perquè és una manera de no aprovar l'assignatura i és molt possible que una ajuda a temps pugui canviar això.

Dintre de la classificació també existeix una àmplia varietat de metodologies a seguir per poder aconseguir el resultat, com per exemple: Nearest Neighbors, Random Forest, Decision Tree, etc. En aquest cas al tenir les dades de resultats finals per utilitzar una base de dades preexistent, ser un mètode senzill d'entendre i aplicar i tenir una bona tassa d'encert s'ha decidit utilitzar el mètode de Decision Tree, arbre de decisions.

4.3.2 Decision Tree

Un arbre de decisió és com el seu nom indica una estructura en forma d'arbre on els seus nodes són condicions o decisió a prendre per seguir els arcs de l'arbre cap a les seves fulles. Les fulles finalment són la classificació que rebrà l'objecte. L'arbre pot tenir tants nivells de profunditat com

es vulguin, però cada nivell de profunditat implicarà una major demanda de recursos i temps per part del sistema a predir les classificacions dels objectes tenint en compte que el nombre de nodes a tenir en compte serà de n^2 sent 'n' el nivell de profunditat.

4.3.3 Desenvolupament de l'arbre

Per generar l'arbre de decisió utilitzem els Datasets de pandas, que vam mencionar a l'informe anterior, ja que pandas és una llibreria molt estesa al món del machine learning i funciona perfectament amb la llibreria 'sklearn'. Aquesta llibreria és una llibreria enfocada expressament al machine learning, tenint funcions definides tant per classificació, regressió, agrupament, etc. En concret aquesta llibreria té definit un objecte arbre de decisions que només necessita com a paràmetre el grau de profunditat que vols al teu arbre i això generarà una estructura en forma d'arbre, però amb condicions i fulles buides. Per emplenar els nodes i fulles de l'arbre s'ha d'entrenar a l'arbre, això es fa donant-li un percentatge de les dades que vols classificar amb el seu resultat de classificació final i en la funció train (entrenar) buscarà dintre de les dades que li has donat i els resultats els patrons que generin una major taxa d'encert, tenint totes les dades inicialment el mateix pes a les decisions.

Abans de fer l'entrenament de l'arbre, s'ha de generar el Dataset que se li donarà a l'arbre per entrenar. El Dataset general amb el qual es comença són les dades personals dels alumnes filtrades per assignatura i curs. Tot i que hi ha dades molt interessants dintre de l'aspecte social que poden condicionar el rendiment de l'estudiant només són dades personals i no té en compte cap dada acadèmica, per això hem de processar aquest Dataset per afegir-li algunes dades acadèmiques. Les dades acadèmiques agafades han sigut: la nota mitjana en els treballs entregats i exàmens fets, el nombre d'entregues que ha fet l'estudiant i finalment el nombre de dies diferents que ha consultat el material d'estudi l'alumne.

Amb totes aquestes dades només ens queda netejar el Dataset, la funció train de l'arbre de decisió no accepta valors en intervals ni valors 'NaN'[15], per això el primer que es fa és netejar tots els valors 'NaN' i substituir-los per '0' i després canviar els valors que indiquen intervals a valors binaris representats per diferents columnes noves. A continuació un exemple on la taula 1 que conté valors representats amb intervals és transformada amb una nova taula 2, representant la mateixa informació però amb valors binaris.

Fig. 1: Taula d'intervals

Persona	Edat
Persona 1	20-30
Persona 2	30-40
Persona 3	40-50

Fig. 2: Taula binària

Persona	Edat 20-30	Edat 30-40	Edat 40-50
Persona 1	1	0	0
Persona 2	0	1	0
Persona 3	0	0	1

Un cop tenim el Dataset vàlid creat separarem el resultat que esperem, que es troba al Dataset en un nou Dataset, de manera que quedaran el Dataset de dades i el nou Dataset de resultats. Normalment, aquests Datasets es tornen a partir en dos cadascun per entrenar l'arbre amb una tupla de Dataset dades i Dataset resultat i testejar el funcionament de l'arbre de decisions amb l'altra tupla per verificar que no hi hagi corrupció de dades. En el nostre cas sí que agafarem només una part de la tupla de Datasets per entrenar l'arbre per, en canvi, provem l'arbre directament amb la tupla de Datasets sencera per poder donar una predicció de tots els estudiants que es troben al Dataset de la classe.

4.3.4 Generació de l'arbre

Inicialment, la idea al tenir totes les dades de tots els cursos i assignatures a l'abast per estar utilitzant una base de dades preexistent era la d'inicialitzar i entrenar l'arbre just després que es guardin totes les dades de la base de dades a Datasets pandas, en iniciar el servidor perquè a l'entrenar l'arbre amb un volum tan elevat de dades generaria un nivell de fiabilitat a la predicció molt elevat, però al tenir una base de dades amb un volum massiu de dades no era una opció adient al ser poc eficient, entrenar l'arbre complet podia fer que el servidor trigués en estar operatiu diversos minuts i això en una aplicació web és un punt important a tenir en compte. Per això la lògica es va canviar per inicialitzar l'arbre i entrenar-ho i donar les prediccions en el moment que tenim totes les dades filtrades, és a dir només tenim els alumnes que continuen estudiant, que cursen l'assignatura i curs seleccionada i també tenim la informació de quin dia del curs ens trobem. Això millora el rendiment fent que les dades per entrenar l'arbre siguin menors però encara amb un número suficient gran com per dona valors elevats a la predicció.

Per aquesta nova lògica implementada no és possible generar prediccions per la vista de l'estudiant, ja que al tenir les dades filtrades al Dataset només es troba un estudiant i amb tan poca informació no es pot entrenar l'arbre.

4.3.5 Diferents Opcions

Com s'ha explicat hi ha diferents factors que influeixen en la qualitat de la predicció del model, factors com a quines variables utilitzem, el grau de profunditat o el nombre de dades que tenim tant per entrenar l'arbre com per classificar.

Per això a continuació es mostraran els resultats obtinguts de proves fetes sobre la mateixa assignatura amb diferents variables i en diferents temps del curs per veure que la quantitat d'informació acadèmica que tenim dels alumnes és important.

Accuracy, és el percentatge d'encert resultant del model de predicció

1. Només dades personals
    - Dia 0
        - Accuracy = 62,03
    - Dia 50
        - Accuracy = 62.26
    - Dia 100
        - Accuracy = 62.58
2. Dades personals i d'entregues fetes (percentatge d'entregues i nota mitja)
    - Dia 0
        - Accuracy = 62.03
    - Dia 50
        - Accuracy = 74.36
    - Dia 100
        - Accuracy = 78.56
3. Dades personals i d'entregues fetes i dies estudiats
    - Dia 0
        - Accuracy = 68,10
    - Dia 50
        - Accuracy = 78.58
    - Dia 100
        - Accuracy = 80.79

Com es pot veure en els resultats obtinguts, tot i que totes les prediccions comencen amb el mateix nivell d'accuracy, amb més quantitat d'informació acadèmica més millora el nivell d'accuracy obtingut passat el temps, això és degut al fet que hi ha més informació sobre el comportament de l'alumne, i per això al projecte s'utilitza l'última opció de les 3 mostrades.

5 Resultats

Un cop vist el procés de desenvolupament dut a terme al projecte, es poden veure els resultats finals aconseguits. Per poder avaluar els resultats obtinguts es comprovarà si aquests compleixen els objectius marcats a l'inici d'aquest document i com de bé el compleixen.

5.1 Entorn de visualització de dades

Els resultats obtinguts són majoritàriament exitosos, ja que els objectius inicials han estat satisfactòriament aconseguits. S'ha aconseguit desenvolupar una aplicació web totalment funcional, amb servei d'identificació d'usuari, navegable i que representa totes les dades inicialment plantejades com a requisits i es pot interaccionar amb elles de forma temporal seleccionant la data en què es vol consultar la informació.

5.2 Sistema de predicció

Dintre de les dades a representar es troba la predicció de la possibilitat de suspendre o no de l'estudiant, aconseguida utilitzant un model de predicció basat en les dades dels alumnes segons els dies transcorreguts del curs que dona un nivell de fiabilitat alt d'un 75 percent d'encert.

5.3 Resultats millorables

Com s'ha dit abans s'han aconseguit de manera satisfactòria la majoria i no tots els objectius. Això és perquè alguns tot i haver estat assolits no han sigut de la millor manera, hi ha dos objectius concretament:

- La pàgina d'estudiant, on l'estudiant veu tota seva la informació. Un dels objectius era no només poder donar suport al professor amb el seguiment dels alumnes, sinó que també se li volia donar a l'estudiant el màxim d'informació pròpia perquè es pogués fer autoseguiment també, tot i que s'ha aconseguit mostrar-li la informació mínima requerida, com dades personals i les entregues que s'han fet i que ha pujat, no s'ha aconseguit mostrar-li dades que a la vista del professor sí que hi són com la predicció de si està en perill de suspendre.

- La portabilitat de l'aplicació, un dels objectius era el d'estudiar la possibilitat de fer que l'aplicació sigui més fàcilment exportable i instal·lable, encara que l'estudi s'ha dut a terme i s'ha determinat que això es podria fer amb l'eina Docker no s'ha aconseguit crear un Docker exportable. Això fa que l'objectiu estigui complert, però no hagi aportat valor al projecte finalment.

6 Discussió

En aquesta secció es vol comentar conceptes que han tingut bastant pes en el desenvolupament del projecte, començant per la definició dels objectius.

Al principi el plantejament de la solucio a donar una eina d'ajuda academica, era només la creació de la aplicacio per poder visualitzar totes les dades tant academiques com personals dels alumnes necesaries, donades per la base de dades i aquestes visualitzarles de la millor manera posible desde el punt de vista 'UserFriendly'. Es a dir, nomes era nocesari la visualització de les dades, pero per millorar l'eficiencia de la eina en el seu objectiu principal que es el de crear una eina d'ajuda en entorns academics i per motius d'interes personal en el camp de tractament de dades, es va decidir ampliar el projecte i implementar un model de predicció basat en dades personals i academiques dels alumnes, això ha implicat un estudi del camp de models predictius, fer un tractament intensiu de les dades a l'abast, crear l'eina de predicció i testejar-la. Finalment s'ha aconseguit que l'aplicació compti amb una eina molt util desde el punt de vista de poder detectar estudiants amb dificultats amb temps pero que ha implicat molt de temps. Per aixo el aspecte visual i la comoditat del usuari en la aplicació no ha sigut una prioritat i tot i que se l'hi ha dedicat temps tambe clarament es un aspecte que es podria millorar.

Un altre aspecte a comentar sobre el projecte han sigut problemes amb la planificació emprada, d'aquí podem parlar sobre dos problemes d'eficiència que han succeït durant el desenvolupament del projecte. El primer ha sigut no dedicar-li més cura al plantejament inicial de tasques i el seu desenvolupament, això ha conduït a un allargament de temps del projecte de 5 sprints inicials a 7 finalment i també al no plantejar una estructura precisa de dependència de tasques/funcions ha aparegut un problema on la funció

d'entrenar l'arbre i predir els resultats estava només enfocada les dades que el professor té d'un curs i això ha impedit poder utilitzar aquesta funció per la vista d'estudiant, creant així una mancança de valor al producte que es podria haver solucionat amb una millor planificació. La segona ha sigut la ineficiència de la utilització de l'eina Jira per tenir un control de les tasques: quines fer, quines s'han fet i quines estan boquejades. Aquesta ineficiència bé del fet que només hi ha hagut una persona treballant en el projecte i en tot moment sabia que estava fent i quines tasques estaven ja fetes, Jira és una eina molt útil però per la gestió de tasques en un equip de persones.

Finalment, un punt remarcable és la representació de la fiabilitat de les dades de l'eina de predicció per escollir les dades a utilitzar. El mètode que s'utilitza per veure la precisió de la predicció és l'accuracy, això és un mètode molt simple a la vegada que molt útil, ja que mira el nombre d'encerts entre el total de prediccions fetes. Però s'hauria d'estudiar també diferents representacions de precisió de resultats, com per exemple la matriu de confusió, aquesta matriu mostra el número de en el nostre cas: aprovats reals, suspensos reals, aprovats falsos, suspensos falsos. Com per nosaltres volem que el nombre més gran de persones aprovi i donar suport als que tenen possibilitats de suspendre si dos DecissionTree donen el mateix accuracy s'hauria de mirar aquesta mètrica i escollir l'arbre que doni més resultats aprovats falsos que suspensos falsos.

7 Conclusions

Per concloure aquest document i el treball, un cop vist els resultats d'aquest i el compliment dels objectius marcats es pot dir que ha sigut un treball tant exitós com profitós des de punt de vista acadèmic i personal, ja que en aquest projecte he après i consolidat coneixement que no només ha sigut útil pel projecte de la carrera sinó que són conceptes als quals tenia un interès personal per continuar la meva formació.

L'eina resultant d'aquest treball pot ser molt profitosa pels entorns acadèmics des del punt de vista del professor, però també de l'alumne, ja que el seu punt principal és millorar els seus resultats. Aquesta també treball té potencial per millorar la seva usabilitat com aplicació i les seves prestacions de tractament de dades.

Agraïments

Vull Agrair al professor Juan Antonio Martinez Carrascal la llibertat, ajuda i guia a donar aquests primers passos al món del Data Science amb aquesta introducció a models de predicció.

Referències

[1] OU Analyse — Knowledge Media Institute — The Open University. (n.d.). Retrieved October 8, 2022, from https://analyse.kmi.open.ac.uk/AnalyseDashboard-demo/module-overview/AAA/2015J

[2] Student Performance Dashboard — Education Dashboards — Bold BI. (n.d.). Retrieved October 8, 2022, from https://www.boldbi.com/dashboard-examples/education/student-performance-dashboard

[3] Home - School Analytics Ltd. (n.d.). Retrieved October 8, 2022, from https://www.schoolanalytics.co.uk/

[4] Conjunto de datos - Wikipedia, la enciclopedia libre. (n.d.). Retrieved January 22, 2023, from https://es.wikipedia.org/wiki/Conjunto-de-datos

[5] Tutorial — Flask Documentation (2.2.x). (n.d.). Retrieved October 8, 2022, from https://flask.palletsprojects.com/en/2.2.x/tutorial/

[6] pandas documentation — pandas 1.5.0 documentation. (n.d.). Retrieved October 8, 2022, from https://pandas.pydata.org/docs/index.html

[7] Tutorial: Introducción a React – React. (n.d.). Retrieved October 8, 2022, from https://es.reactjs.org/tutorial/tutorial.html

[8] Desarrollo ágil de software - Wikipedia, la enciclopedia libre. (n.d.). Retrieved October 8, 2022, from https://es.wikipedia.org/wiki/Desarrollo-

[9] Scrumban - Wikipedia. (n.d.). Retrieved October 8, 2022, from https://en.wikipedia.org/wiki/Scrumban

[10] Jira — Software de seguimiento de proyectos e incidencias. (n.d.). Retrieved October 8, 2022, from https://www.atlassian.com/es/software/jira

[11] Herramienta de Maquetas, Esquemas and Prototipos UI En Línea Moqups. (n.d.). Retrieved October 8, 2022, from https://moqups.com/es/

[12] venv — Creation of virtual environments — Python 3.11.0 documentation. (n.d.). Retrieved November 14, 2022, from https://docs.python.org/3/library/venv.html

[13] Node.js. (n.d.). Retrieved November 14, 2022, from https://nodejs.org/en/

[14] Aprendizaje automático - Wikipedia, la enciclopedia libre. (n.d.). Retrieved December 18, 2022, from https://es.wikipedia.org/wiki/Aprendizaje-autom

[15] NaN - Wikipedia, la enciclopedia libre. (n.d.). Retrieved December 18, 2022, from https://es.wikipedia.org/wiki/NaN

Apèndix

A.1 Sprints



Fig. 3: Sprint 1

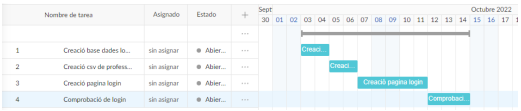


Fig. 4: Sprint 2

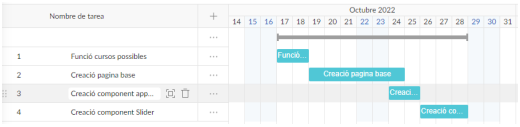


Fig. 5: Sprint 3

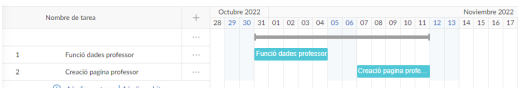


Fig. 6: Sprint 4

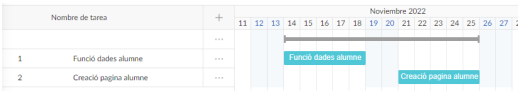


Fig. 7: Sprint 5

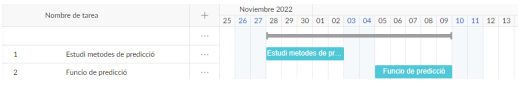


Fig. 8: Sprint 6

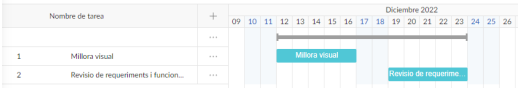


Fig. 9: Sprint 7

A.2 Mockups

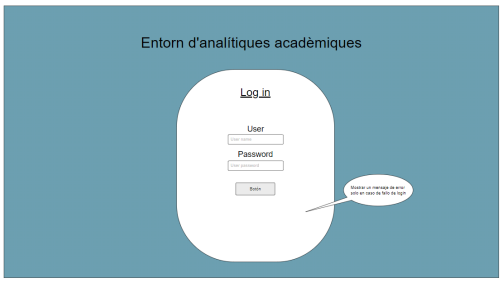


Fig. 10: Mockup Login

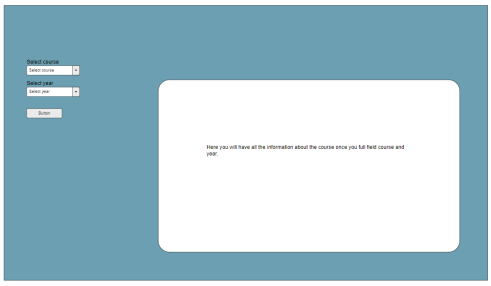


Fig. 11: Mockup base page

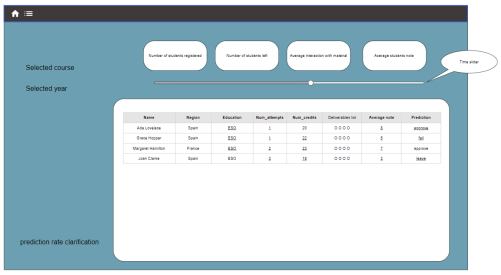


Fig. 12: Mockup teacher page

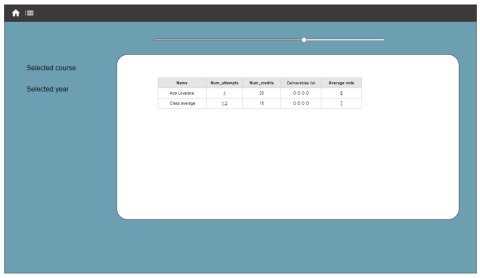


Fig. 13: Mockup student page