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

Alfonso Secanella, Marc; Carreras Sala, Sergi, dir. App (Android + iOS) de panell de reproducció de dades de sensors diversos: GNSS, acceleració, velocitat, inclinació. 2024. (Grau en Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/298909>

under the terms of the  license

App (Android + iOS) de panell de reproducció de dades de sensors diversos: GNSS, acceleració, velocitat, inclinació...

Marc Alfonso Secanella

25 de juny de 2024

Resum– Actualment, un dels dispositius IoT que fa servir Atlantis IT per geolocalitzar motocicletes disposa d'un conjunt de dades extretes de diversos sensors GNSS juntament amb un acceleròmetre intern, dades com la velocitat, acceleració, inclinació, frenada, etc. Aquestes dades ara per ara no les estan processant i és aquí on apareix aquesta proposta de projecte. La idea és analitzar i processar les diferents dades que ofereix el dispositiu i representar-les en un format d'aplicació mòbil dirigit tant a dispositius Android com iOS, a través de gràfics, rellotges, marcadors, i diferents elements que puguin ser visualment interessants pels usuaris que acabin fent servir l'aplicació.

Paraules clau– Sensors, GNSS, Acceleròmetre, Aplicació, Mòbil, IoT, Flutter, Dart, Geolocalització, Motocicleta, Android, iOS

Abstract– Currently, one of the IoT devices used by Atlantis IT to geolocate motorcycles has a set of data extracted from various GNSS sensors together with an internal accelerator, data such as speed, acceleration, inclination, braking, etc. These data are not currently being processed and this is where this project appears. The idea is to analyze and process the different data offered by the device and represent them in a mobile application format aimed at both Android and iOS devices, through graphics, clocks, markers, and different elements that can be visually be interesting for users who end up using the application.

Keywords– Sensors, GNSS, Accelerometer, App, Mobile, IoT, Flutter, Dart, Geolocation, Motorcycle, Android, iOS



1 INTRODUCCIÓ - CONTEXT DEL TREBALL

ACTUALMENT existeixen moltes tecnologies IoT[1] que d'una manera o altra poden facilitar la vida a les persones. Entre tota aquesta varietat es troben els dispositius enfocats a la geolocalització[2] de persones, bicicletes, automòbils, etc. L'empresa col·laboradora[3] s'encarrega, entre altres coses, a fer dispositius IoT per a motocicletes per tal de, a través d'un dispositiu que l'usuari pot instal·lar a la seva motocicleta, tenir-la localitzada en tot moment fent ús d'una aplicació mòbil on pot obser-

var una gran varietat d'informació sobre el vehicle associat i configurar-ho al gust d'aquest. Informació com l'estat de la bateria, consum de gasolina, rutes fetes, etc. Aquests dispositius IoT envien una sèrie d'informació als servidors, la qual prové de diversos sensors i GNSS, que posteriorment és tractada i mostrada a l'aplicació.

Una part d'aquesta informació rebuda pels dispositius, actualment no es processa i d'aquí sorgeix la necessitat de l'empresa en processar-les i representar-les a través d'elements gràfics. Degut a la problemàtica esmentada, apareix aquest projecte de treball per dur a terme tot el procés necessari amb l'objectiu principal de processar i representar aquestes dades en una nova aplicació per a dispositius mòbils.

Aquest article pretén mostrar tot el procés per satisfer la necessitat identificada a través d'una sèrie de seccions que aniran descrivint tot el procés que es seguirà. L'organització del document constarà primerament d'una secció dedi-

• E-mail de contacte: malfonso621@gmail.com
 • Menció realitzada: Enginyeria del Software
 • Treball tutoritzat per: Sergi Carreras Sala (Microelectrònica i Sistemes Electrònics)
 • Curs 2023/24

cada als objectius, seguidament la metodologia i la planificació a seguir, a continuació l'estat de l'art on es mostraran aplicacions similars a la proposta d'aquest projecte. A més d'això, es dedicarà una secció a l'anàlisi de les dades i el seu tractament associat. Feta l'anàlisi es passarà a explicar el desenvolupament del projecte, i finalment els resultats i conclusions que es puguin extreure.

2 OBJECTIUS

L'objectiu principal del projecte és fer una aplicació mòbil per a dispositius Android i iOS per tal de representar visualment les dades d'una varietat de sensors que es poden trobar en un dels dispositius IoT de l'empresa col·laboradora.

Fent un desglossament de l'objectiu principal es pot extreure la següent informació:

OB1 - Processament i anàlisi de les dades GNSS i de l'acceleròmetre: El primer pas per assolir l'objectiu principal és realitzar un estudi de les dades GNSS i les de l'acceleròmetre amb els seus tres eixos. L'estudi de les mateixes es farà a través de l'eina Postman[7] que permet fer peticions API, en aquest cas apuntant als servidors de l'empresa, i rebre dades reals dels dispositius. Amb això es pretén obtenir un millor coneixement sobre com s'obtenen, en quin format venen i com s'ha de procedir per tractar-les. Una anàlisi exhaustiva permetrà posteriorment saber quines dades són les que interessarà acabar mostrant a l'aplicació.

OB2 - Representació de la ruta: Aquest objectiu dona resposta a les necessitats de pintar la ruta sobre un mapa i que aquesta es pugui anar reproduint, també a pintar la ruta segons aquesta es filtri per la velocitat, l'acceleració o la inclinació de la motocicleta. I finalment, mostrar una motocicleta de cara i que aquesta es vagi inclinant en funció d'aquesta inclinació en els diversos punts de la ruta. Es pot resumir, en què aquest segon objectiu dona l'abast a totes les necessitats relacionades amb la ruta i el mapa digital.

OB3 - Eines de visualització: La inclusió de diferents rellotges, gràfics i d'altres elements gràfics que permetin representar diverses dades és una part important de l'aplicació, per tant, aquest objectiu englobarà tant la part d'investigació d'eines que puguin ser interessants i útils per mostrar tota la informació pertinent, com la part d'utilització de les eines triades i la manera en les que es mostraran.

3 METODOLOGIA

La metodologia triada per a la realització d'aquest projecte va ser una metodologia àgil, en aquest cas va ser l'Scrum. Aquesta decisió es va deure al fet que Scrum és una metodologia que durant el transcurs dels estudis vaig aplicar a diversos projectes i va funcionar molt bé. Aquesta, es caracteritza per ser una metodologia incremental i iterativa. L'aplicació de la metodologia va consistir en dividir el projecte en Sprints de dues setmanes on a cada Sprint es realitzaven un conjunt de tasques específiques. Això va permetre posar un ordre al projecte i anar separant-lo per etapes, tot seguint la planificació indicada, la qual s'explica en detall a la secció 4.

La divisió de la feina va quedar de forma que cada Sprint seria de dues setmanes fins a arribar a la finalització del projecte. La decisió de fer Sprints de dues setmanes sobretot es va deure al fet que les reunions de seguiment amb l'empresa eren també cada dues setmanes. Durant aquestes reunions es feia una revisió de les tasques planificades per aquell Sprint, s'assignaven noves tasques i es proposaven idees o suggeriments. Aquest seguiment va permetre anar obtenint comentaris constants de l'empresa sobre els elements que s'anaven desenvolupant.

Pel que fa a la gestió de tasques i del seu progrés, es va fer ús de l'eina Trello[4], que permet a través del seu taulell, visualitzar ràpidament l'estat de les diverses tasques i permet filtrar-les segons una sèrie de paràmetres.

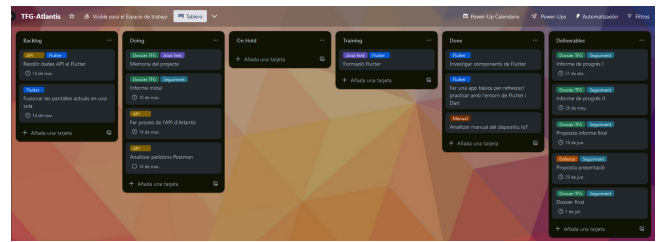


Fig. 1: Fase inicial del Trello

3.1 Eines i tecnologies

El projecte, al ser una aplicació mòbil, presenta un conjunt més limitat d'opcions tecnològiques. Les opcions més comunes inclouen Kotlin, Flutter amb Dart JS[6], i Swift, utilitzat principalment en aplicacions iOS. Donat que un dels requisits és que l'aplicació sigui compatible amb dispositius tant Android com iOS, per tant, Swift es pot descartar, deixant les opcions de Kotlin i Flutter. Considerant això, i tenint en compte que l'empresa col·laboradora acostuma a utilitzar Flutter, es va decidir seguir aquest camí tecnològic per no diferir gaire de cara a la integració del producte final. A continuació es mostra un breu resum de les tecnologies i eines utilitzades per al projecte:

- **Flutter:** És un framework de codi obert fet per Google enfocat a fer aplicacions mòbils, especialment multi-plataforma, on amb un únic codi l'aplicació pot funcionar en dispositius Android, iOS, Windows, Web, etc.
- **Postman:** És una eina gratuïta que permet fer proves de peticions API als endpoints especificats. Aquesta funcionalitat resulta molt útil per analitzar les dades reals abans de passar al processament d'aquestes sense necessitat de programació.
- **Android Studio[8]:** És l'Entorn de Desenvolupament Integrat (IDE) seleccionat per desenvolupar el projecte. Aquest programari està dissenyat específicament per a la creació d'aplicacions Android, encara que l'exportació acaba sent a les plataformes indicades, inclòs iOS. La decisió d'utilitzar Android Studio ve donada a què dispo de cert coneixement previ amb aquesta eina i això pot agilitzar el procés de treball.
- **Git:** És un sistema de control de versions molt popular arreu del món. En aquest projecte, s'ha utilitzat tant en

l'aplicació inicial com en Atlantis Moto. L'ús de Git permet desfer canvis gràcies als commits, el qual és útil en cas de problemes. A més, Atlantis utilitza Git en tots els seus projectes, per la qual cosa era necessari tenir-ho en compte.

3.2 Llibreries

Per tal d'implementar certes funcionalitats de l'aplicació, s'ha fet ús d'algunes llibreries sigui per components estètics o per aspectes més concrets de l'aplicació. En aquesta subsecció s'explica en detall les utilitzades i com s'han fet servir.

- **google_maps_flutter[11]:** És la llibreria oficial de Google per fer ús de la seva API de mapes a Flutter. Cal disposar d'un compte de Google Cloud activat per poder fer servir les diferents crides API segons es necessiti. Aquest plugin és el més important, ja que ha permès disposar dels mapes en temps real i poder tractar-los segons convinguí.
- **syncfusion_flutter_gauges[13][16]:** Llibreria principal per a la representació visual de dades en format radial (circular) o en forma lineal. A través d'aquest element s'han implementat tots indicadors circulars els quals són l'acceleròmetre, inclinòmetre, velocímetre i una brúixola.
- **syncfusion_flutter_charts[12]:** Aquesta tercera llibreria està enfocada a tot el tema relacionat amb gràfics i diagrames. En aquest cas, s'ha fet servir per la representació dels diferents gràfics implementats.
- **syncfusion_flutter_sliders[15]:** Llibreria enfocada a fer sliders o lliscadors. L'ús d'aquesta llibreria s'ha degut a la necessitat de fer un lliscador vertical per adaptar la pantalla a la disposició en horitzontal. La llibreria ofereix un ampli ventall de paràmetres que permeten una millor personalització.
- **http[9]:** És la llibreria oficial de Dart per realitzar peticions HTTP. El seu ús ha permès la comunicació amb les API de l'empresa i obtenir dades reals dels sensors, les quals s'han pogut tractar.
- **odometer[17]:** Llibreria específica per fer un component hodòmetre o comptakilòmetres, gràcies al seu ús ha permès una fàcil implementació d'un comptador visual amb l'animació peculiar que té a l'anar sumant kilòmetres al total.

4 PLANIFICACIÓ

La planificació del projecte s'ha dividit en quatre fases que separen grups de tasques concretes, a continuació s'explica breument en què consisteixen:

Fase-01 Concepció i Proves: Fase inicial del projecte on es duen a terme les reunions inicials amb l'empresa per definir el concepte del projecte i què es vol assolir amb ell. També s'estableixen uns requisits bàsics del que es demana i es sintetitzen els objectius que cal satisfer. Paral·lelament, es fan diverses proves amb Flutter[5], per

tal d'anar agafant el fil al funcionament d'aquest. Els requisits del sistema es poden trobar a l'apèndix A.6. Termini: setmanes 1-3

Fase-02 Desenvolupament: La segona fase constitueix la part més extensa del projecte, centrant-se principalment en el desenvolupament de l'aplicació mòbil. Aquesta etapa implica no només la programació necessària de les funcionalitats requerides, sinó també la recerca d'eines de visualització i la seva implementació. Termini: setmanes 4-14

Fase-03 Comprovacions: Com a tercera fase i abans d'entrar en la recta final del projecte, es fan una sèrie de comprovacions de qualitat de l'aplicació per validar que s'han satisfet tots els requisits i s'han assolit els objectius inicials del projecte. A part, es verifica que l'aplicació funciona correctament i no esdevé en cap problema o anomalia a l'hora d'executar-la, en cas de trobar-ne algun, es dedica el temps necessari per a resoldre-ho i continuar amb el cicle de comprovacions. Per tant, aquesta fase es pot resumir en el fet que es prova tota l'aplicació amb l'objectiu de deixar-la polida i en perfecte estat, assolint així un alt grau de qualitat i un producte preparat per ser entregat. Termini: setmanes 15-17

Fase-04 Finalització: Aquesta fase engloba tota la part de tancament del projecte. Es redacta l'informe explicant els resultats obtinguts, es defineixen les conclusions i posteriorment es fa una revisió general de tot l'informe abans de procedir a la seva entrega. Per altra banda, es prepara l'aplicació per a ser entregada segons el format que demani l'empresa col·laboradora. Un cop s'han entregat tots els lliuraments, es dona per finalitzat el projecte. Termini: setmanes 18-20

A l'apèndix A.2 es mostra un Diagrama de Gantt amb tota la planificació de les fases en funció de les setmanes.

5 ESTAT DE L'ART

Aquest projecte respon a les necessitats específiques de l'empresa col·laboradora, per tant, es parla d'estat de l'art respecte a aplicacions similars a l'aplicació Atlantis Moto, o aplicacions que disposin de característiques que puguin servir com idea per a realitzar aquest projecte.

Primerament, s'ha analitzat l'aplicació Wikiloc[18], la qual és molt popular en el context de rutes de qualsevol mena ja sigui fent senderisme, anar a córrer, anant en cotxe, motocicleta i moltes més opcions. Pel cas concret que interessa la representació de dades, s'ha agafat de referència la manera com presenta l'altitud mitjançant un graf que mostra les diferents altures durant tot el transcurs del recorregut.

Com a segona opció, l'aplicació Relive[19], de manera similar a l'anterior, també ofereix característiques de cara a la representació de rutes i s'enfoca sobretot a persones que facin senderisme, ciclisme i diverses activitats a l'aire lliure. La característica principal de Relive recau en el fet que recopila les dades de les rutes gravades pels

usuaris i genera vídeos 3D del recorregut.

Per acabar, el producte Komobi[21] és la competència directa d'Atlantis Moto. Komobi es centra especialment en temes de seguretat com la detecció d'accidents i evitar robatoris de motocicletes. Ofereixen també una aplicació mòbil que disposa d'un mapa digital amb diferents mètriques com la inclinació, velocitat, etc.

6 ANÀLISI

La part d'anàlisi i processament de les dades és també una part important del projecte, ja que prèviament a fer la seva representació cal entendre quina informació arriba del dispositiu i com s'ha de procedir per al seu tractament. Per aquest motiu, aquesta secció es centrarà a desglossar tot el procés que s'ha dut a terme per processar les dades. L'explicació constarà de dues subseccions, la primera centrada a explicar les API cedides per l'empresa i el seu funcionament, mentre que la segona subsecció es centrarà en tot el desglossament de les dades i el seu processat.

Cal destacar que la informació delicada tant de les API, com de les figures i de l'explicació, s'evita mostrar per temes de privacitat amb l'empresa.

6.1 Explicació i funcionament de les API

En aquesta primera subsecció es dona protagonisme a les quatre peticions API proporcionades per l'empresa per obtenir dades reals i continuar amb el desenvolupament del projecte.

Login Moto: Aquest primer endpoint respon a la sol·licitud per iniciar sessió a l'aplicació Atlantis Moto. Com es pot observar a la figura 2, rep com a body, un nom d'usuari, una contrasenya i finalment el tipus d'aplicació. La petició retorna entre d'altres, un token d'autorització del tipus Bearer Token i també retorna un refresh token, ambdós tokens són necessaris per poder cridar a la resta de peticions API.

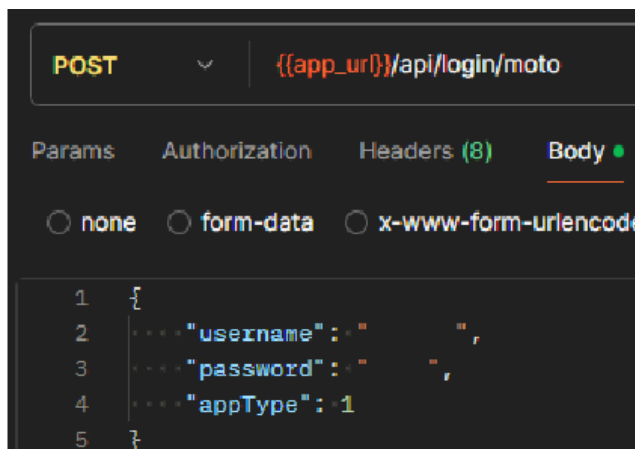


Fig. 2: Petició API Login

Refresh Token: Aquesta segona petició, s'encarrega de refrescar una clau d'inici de sessió el qual hagi expirat. Rep com a paràmetre del body el refresh token obtingut a l'haver fer prèviament la petició d'inici de sessió.

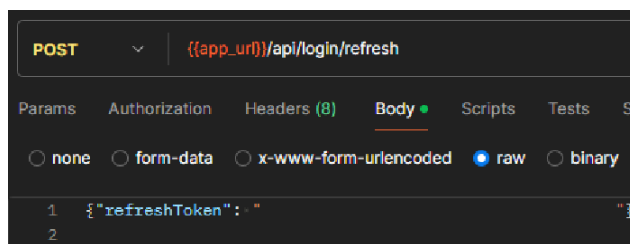


Fig. 3: Petició API Refresh

Get days trips: Com a tercer endpoint, es troba la petició per obtenir d'un usuari tots els dies on hagi fet viatges. La petició rep l'IMEI del dispositiu lligat a l'usuari i retorna un llistat amb totes les dates que disposin de viatges. Aquestes dates que retorna poden ser utilitzades per la petició de Get trips info i veure detalls del viatge fet.

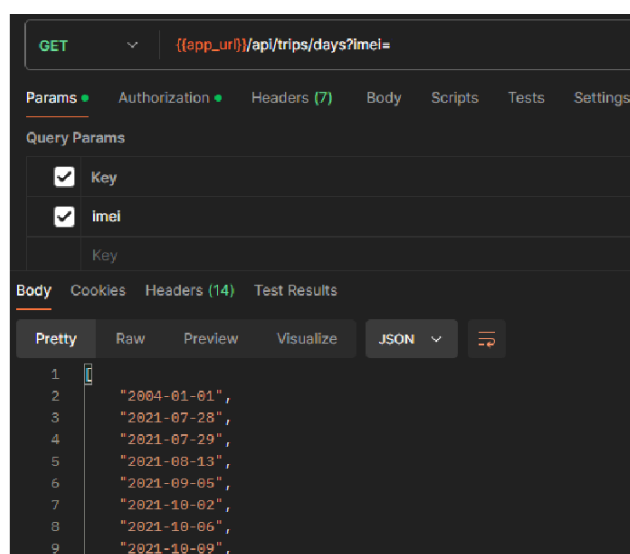


Fig. 4: Petició API Days

Get trips info: Per últim, el Get trips info és la petició més important i la que retorna totes les dades del viatge sol·licitat. La petició rep l'IMEI del dispositiu juntament amb la data d'inici i finalització del viatge a tractar.

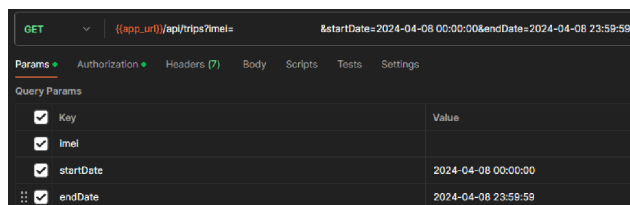


Fig. 5: Petició API Info

6.2 Extracció i processament de les dades

Pel que fa a aquesta segona subsecció es mostra el desglossament de tota la petició Get trips info i les dades que retornen els diversos sensors, les quals seran processades i finalment mostrades a l'aplicació.

Tota l'explicació es fa agafant de referència un viatge fet per un usuari el dia 8 d'abril del 2024. Les figures de la resposta de l'API es poden trobar a l'apèndix A.3.

Part 1: Camps principals

La figura de referència és la 10.

- **initialPosition i finalPosition:** Són camps que contenen informació tant de la posició inicial com el final del viatge fet el dia indicat. Informació com ara la velocitat, les coordenades, altitud, direcció, entre d'altres.
- **Camps de totals:** Són un conjunt de camps que resumeixen de manera global tot el viatge. Mostren dades com tota la distància recorreguda, el consum de combustible, la velocitat mitjana, etc.
- **Trips:** Aquest camp és un llistat que conté els diferents subviatges fets durant el període de temps indicat. S'entén com a subviatge aquell recorregut on s'encén i s'apaga el motor de la motocicleta.

Part 2: Contingut del camp trips

La figura de referència és la 11 i pel llistat de Tracks és la 12.

- **initialPosition i finalPosition:** Segueixen la mateixa lògica que els camps explicats en la primera part, però en aquest cas són la posició inicial i final del subviatge en qüestió.
- **Camps de trip:** Actuen de la mateixa manera que els camps totals de l'apartat anterior, en aquest cas mostren dades globals del subviatge.
- **Tracks:** És un camp de tipus llistat o array que conté totes les posicions del subviatge seleccionat. Cada posició engloba un conjunt d'informació que recull el dispositiu IoT i envia cada X temps al servidor. Conté informació com ara la velocitat, coordenades, altitud, direcció, etc.

Part 3: Processament de les dades

Un cop s'han identificat totes les dades disponibles ja només queda escollir quines es faran servir per a la representació i segons calgui, fer el tractament que convingui. Les dades que interessen són les següents:

- El llistat de trips amb tots els seus tracks de cada subviatge, necessari per poder representar tota la ruta al mapa.
- Els camps initialPosition i finalPosition, per poder identificar on comença i on acaba el viatge.
- La majoria de dades informatives dins dels camps initialPosition/finalPosition i dels tracks. Les dades s'utilitzaran per tota la representació de dades a través dels indicadors i dels gràfics.

7 DESENVOLUPAMENT

El projecte, pel que fa al desenvolupament, es va dividir en dues fases. La primera fase es va centrar en la creació d'una aplicació des de zero per tal d'anar investigant i fent proves amb els diferents components i llibreries que ofereix Flutter. Un cop es van identificar els components i les llibreries a utilitzar es va passar a la implementació de totes les funcionalitats. La segona fase del desenvolupament va consistir

en migrar el codi desenvolupat a l'aplicació Atlantis Moto per tal d'aplicar l'estètica final i preparar el lliurament d'aquesta. Cal destacar que mentre es feia la migració es van anar fent noves implementacions i millores del codi.

7.1 Aplicació inicial

L'inici del desenvolupament va consistir en anar fent proves amb llibreries i components de Flutter per tal de trobar els ideals per fer les funcionalitats demanades. Vaig iniciar el desenvolupament fent diverses pantalles on vaig provar coses específiques. Es van desenvolupar cinc pantalles inicials:

- A la primera pantalla es van fer proves d'inclinació agafant una imatge d'una motocicleta i fent un slider o lliscador que anava modificant els graus d'inclinació. La primera pantalla enfocava el tema de la inclinació de la motocicleta.
- La segona pantalla va consistir en la incorporació del mapa de l'API de Google Maps amb un marcador personalitzat en forma de motocicleta.
- La tercera pantalla era una prova de canviar el color d'un contenidor en funció del tipus de ruta (acceleració, velocitat o inclinació). Això ho vaig fer amb un selector de 3 opcions, un slider per modificar el valor i un contenidor amb el color de fons que anava canviant segons en quin rang estigués el valor seleccionat.
- La quarta era un indicador bàsic[16], en aquest cas un velocímetre, fet amb valors aleatoris que s'actualitzaven cada segon. Aquí ja vaig incorporar una llibreria nova que era la Syncfusion Flutter Gauges.
- Per acabar, una pantalla on vaig provar el tractament d'API dins de Flutter, fent ús de la llibreria http per fer la petició a l'API especificada i poder obtenir totes les dades. Primerament, vaig agafar una API[14] gratis d'Internet per fer proves, i un cop rebut l'accés a les API d'Atlantis vaig provar amb aquestes.

Passades les setmanes inicials del desenvolupament, vaig començar a ajuntar totes les pantalles en una única la qual acabaria formant la pantalla final. A partir d'aquest moment, tot el desenvolupament es va centrar en una única pantalla amb tots els elements visuals. La pantalla inclou el mapa, tots els indicadors, el canvi de ruta, el panell de reproducció, etc. A continuació es detalla totes les funcionalitats desenvolupades durant el transcurs de la primera part del desenvolupament i l'ús d'aquest component s'ha degut a la necessitat d'aprofitar al màxim l'espai disponible.

7.1.1 Informació prèvia

Cal destacar que per a la representació de tota la informació, en especial els gràfics i els indicadors, s'ha fet servir el component PageView[10] de Flutter, el qual permet fer un paginador de components. Per aquest projecte s'ha fet el paginador de 3 i 4 pàgines segons si l'orientació del dispositiu era en vertical o en horitzontal respectivament. El paginador funciona amb el lliscament del dit, per tant, les pàgines van variant segons la direcció del lliscament. Cada pàgina conté certs components.

7.1.2 Panell de reproducció

Aquesta és una de les funcionalitats principals de l'aplicació. La idea és que donada una ruta, aquesta es pugui reproduir com si fos un vídeo, de manera que el marcador es va movent a mesura que la ruta va avançant. S'ha implementat un panell bàsic amb 3 opcions, les quals són reproduir/aturar la ruta i situar-se a la primera o a l'última posició de la ruta directament. La reproducció es produeix cada segon, és a dir, el marcador avança cap endavant cada segon que passa sempre i quan s'hagi clicat a la icona de reproduir.

A la figura de sota, es pot observar al centre el botó de reproduir/aturar, a l'esquerra el botó per anar directament a la posició inicial, i a la dreta per situar-se a la posició final de la ruta.



Fig. 6: Botons del panell de reproducció

7.1.3 Lliscador de posicions

Aquesta funcionalitat és un complement al panell de reproducció. Fent ús de la llibreria Syncfusion Flutter Sliders, s'ha implementat un slider o lliscador que permet situar-se a qualsevol posició de la ruta amb el simple moviment del slider. El lliscador és de la mida del llistat de tracks, per tant, permet situar-se en qualsevol punt. Tots els rellotges indicadors i els gràfics es veuen actualitzats amb el moviment del lliscador, això fa que aquesta característica també sigui útil si es vol veure alguna dada específica en algun de la ruta. A les imatges de l'apèndix A.4 es pot observar el lliscador de posicions tant horitzontal com vertical en diverses imatges situat a sota del panell de reproducció

7.1.4 Representació de la ruta

Aquesta funcionalitat respon directament a l'objectiu OB2 que es pot trobar a la secció 2. A continuació es detallen les funcionalitats desenvolupades per representar la ruta.

El pintat de la ruta sobre el mapa digital es va assolir fent ús del paràmetre `polyline` que ofereix el component Google Maps. El paràmetre rep un llistat d'elements i els uneix per formar les línies que conformen la ruta. D'aquesta manera es va poder mostrar la totalitat d'una ruta amb el seu inici i final.

La selecció del tipus de ruta va ser el següent pas un cop ja es mostrava la ruta sencera al mapa. Es va utilitzar el widget `PopupMenuButton` que permet definir un llistat d'opcions i l'acció que es vol fer amb cadascuna d'elles. Les seleccions disponibles van ser velocitat, acceleració i inclinació. Prèviament a això, es van generar variables locals que emmagatzemaven llistats de totes les dades que s'havien escollit, aleshores segons l'opció seleccionada es calculava la ruta en funció de la dada triada. Cada tipus de ruta disposava d'una escala de colors i cada color corresponia a un rang del valor en qüestió. Les figures de referència són la 16 i 20. A les figures es pot veure el selector d'opcions, el qual apareix clicant a la icona dels

tres punts situada a l'appbar.

Finalment, es va fer que el marcador fos una icona d'una motocicleta i s'inclinés, simulant el grau d'inclinació en cada punt de la ruta. Aquesta característica s'ha vist alterada i s'explica en la secció 10.

7.1.5 Indicadors de dades

Aquesta funcionalitat s'enfoca en l'objectiu OB3, dedicat a les eines de visualització. S'ha assolit amb la creació de quatre indicadors.

Un velocímetre similar als que es pot trobar en els vehicles de l'actualitat. S'ha implementat amb la llibreria Syncfusion Flutter Gauges que permet fer una gran varietat d'elements en format rellotge o cercle. El velocímetre mostra la velocitat en el format km/h i la dada real utilitzada correspon al camp `speed`.

Un acceleròmetre que utilitza la fórmula de l'acceleració mitjana [20] entre dos punts. Es fa una divisió entre les velocitats i els temps donats dos punts. En detall, el càlcul que hi ha darrere consisteix en agafar el punt actual de la ruta i d'aquest s'agafa la velocitat= v_1 i el temps= t_1 . A continuació, s'accedeix a la següent iteració, és a dir, posició actual+1 i en aquest cas es fa velocitat= v_2 i temps= v_2 . Finalment, es divideix $(v_2-v_1)/(t_2-t_1)$ per obtenir l'acceleració entre aquests dos punts. El resultat d'aquest càlcul pot ser tant negatiu com positiu, en els casos que ha donat negatiu s'ha considerat que és la frenada entre els dos punts i és la dada que s'ha considerat com a frenada de la motocicleta. Finalment, destacar que l'acceleració s'ha mostrat en unitats del tipus forces G. Les dades utilitzades corresponen a la `speed` i `gpsTime` o `lastGPSTime` en funció del punt de la ruta.

Una brúixola que mostra la dada de heading, és a dir, la direcció de la motocicleta en cada punt de la ruta.

Un inclinòmetre, encara que el seu desenvolupament està aturat a causa de la falta de dades reals i s'estava mirant amb l'empresa com procedir.

Per acabar, s'ha fet un hodòmetre emprant la llibreria `Odometer` que és un component de Flutter amb l'animació típica que té un comptakilòmetres. S'ha fet ús de la dada `odometer` obtinguda a cada posició de la ruta.

A la figura de sota es poden veure tots els rellotges o indicadors de dades. D'esquerra a dreta, a la primera fila està el velocímetre i l'acceleròmetre. Al centre es troba l'hodòmetre i a la segona fila d'esquerra a dreta, una rèplica del velocímetre que en el futur serà un inclinòmetre, i a la dreta una brúixola.

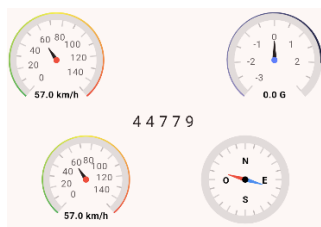


Fig. 7: Indicadors de dades

7.1.6 Gràfics de dades

Aquesta funcionalitat es centra també en l'objectiu OB3, però en la part de representació de gràfics. Es van desenvolupar dos gràfics, un gràfic en funció de l'acceleració/frenada en cada punt de la ruta, i un segon gràfic d'altitud. Els dos gràfics tenen com a eix X el temps en format hh:mm:ss i en l'eix Y l'acceleració/frenada en el primer i l'altitud en el segon. Figures 14 i 15 de l'apèndix A.4.

7.1.7 Layout horitzontal

Aquesta funcionalitat va sorgir com a proposta en una de les reunions de seguiment amb l'empresa. Va consistir a oferir la possibilitat a l'usuari d'orientar el mòbil en mode apaïsat i que la pantalla es vegi adaptada a la nova orientació. Això va comportar una sèrie de proves per identificar quina seria la millor manera de posicionar tots els elements per tal que la interfície fos agradable a la vista. La part més difícil va ser adaptar les mides de certs components i acabar de trobar la millor posició per cadascun d'ells.

Figures 17, 18, 19 i 20.

7.2 Migració a Atlantis Moto

L'inici d'aquesta segona fase es va produir un cop es van finalitzar i validar les últimes funcionalitats introduïdes a l'aplicació. En aquell punt ja estaven fets tots els objectius del projecte. La migració va contemplar diversos elements, a part del mateix traspàs del codi d'un projecte a un altre.

1. Millores del codi en general i la implementació de noves característiques.
2. Identificació del layout definitiu dels elements. Això era un problema pendent de la primera part del desenvolupament, es van anar fent diverses proves per aprofitar tot l'espai disponible i fer la interfície el més agradable possible.

7.2.1 Entorn de migració

Per tal de traspassar tot el contingut d'un projecte a l'altre, Atlantis em va proporcionar un programa per accedir a la seva VPN i al seu repositori Git on tenen el projecte Atlantis Moto que és on es va afegir la meua aplicació / pantalla desenvolupada. Primer de tot es va crear una nova branca al repositori on es va anar traspassant el codi del meu projecte inicial i afegint de nou a l'anar fent millores i noves funcionalitats. El traspàs de codi es va fer als fitxers indicats pel tutor de l'empresa.

7.2.2 Modificacions realitzades

Durant el transcurs de la migració es van anar aplicant millores tant de codi com de funcionalitats específiques. A continuació es detallen les més rellevants:

1. S'ha fet que els gràfics mostrin visualment la posició actual i no tots els punts. Abans costava identificar quina era la posició actual i visualment es veien multitud de punts, ara s'ha deixat únicament un punt per representar la posició actual.
2. S'ha fet que el contenidor que conté els indicadors i els gràfics es puguin amagar i així veure el mapa més gran i més detallat. Per fer-ho s'han creat dos botons nous, que mostren/amaguen la informació. Corresponen a les icones de + i - ubicades a sota del lliscador. A les figures 21 i 22 es poden observar aquestes icones.

7.2.3 Noves funcionalitats

- **Modes de càmera:** Aquesta funcionalitat millora l'experiència de reproduir la ruta. Es van afegir tres tipus de càmeres per gaudir de diferents perspectives de la ruta. La primera era el mode de seguiment que anava seguint constantment al marcador o punter, per tant, aquest sempre estava centrat en pantalla. La segona era el mode bloc que exclusivament movia la càmera quan el marcador sortia del focus d'aquesta. I un tercer mode, anomenat mode lliure, on l'usuari tenia total llibertat per fer zoom i moure's pel mapa digital. Figures 33, 34 i 35 de l'apèndix A.5.
- **Gràfic d'altura complet:** En addició al gràfic d'altitud que ja hi havia anteriorment, se'n va desenvolupar un de nou però per mostrar l'altitud de tota la ruta sencera. De cara l'usuari, aquest disposa de dos botons que segons a quin cliqui es mostra el gràfic actual o el gràfic complet. Figures 24 i 25 de l'apèndix A.5.
- **Noves funcions al panell de reproducció:** S'han creat dos botons nous al panell de reproducció els quals permeten moure el marcador d'1 en 1 tant cap enrere com cap endavant. De manera similar al lliscador de posicions, la idea és facilitar la cerca de posicions específiques de la ruta. Els nous botons corresponen a les dues fletxes de color taronja que es troben als costats del botó de reproduir/pausar. A les figures 21 i 27 es poden observar els botons en els dos tipus de layout.

- **Inclinòmetre:** Es va desenvolupar un nou indicador, en la primera aplicació ja es va fer l'esquelet, però faltava pactar amb l'empresa si es disposaria de les dades reals o caldria buscar una alternativa. Al final es va continuar fent l'indicador, però es van utilitzar dades aleatòries obtingudes dins d'un rang de valors. A la secció 10 es detalla la problemàtica, entorn aquesta implementació. La figura de referència és la 36, també es pot observar en les figures 29 on correspon al rellotge situat a l'esquerra, i a la figura 22 correspon al rellotge de l'extrem esquerre de la segona fila.

8 VALIDACIÓ DE REQUISITS

A l'inici del projecte es van definir i sintetitzar una sèrie de requisits funcionals i no funcionals que englobaven certs aspectes del projecte. Els requisits es poden trobar a l'apèndix A.6. De manera general s'han pogut assolir tots encara que el requisit RNF4 enfocat al funcionament de l'aplicació tant a Android com iOS, no s'ha pogut validar en iOS per falta d'equipament Apple. De totes maneres, Flutter al permetre el desenvolupament multiplataforma, pot donar peu a considerar que en iOS també funcionaria encara que no es pogués validar amb una prova real.

9 RESULTATS

Els resultats del projecte han estat bastant bons, ja que s'ha pogut desenvolupar totes les funcionalitats plantejades i s'han fet de noves per tal d'aprofitar més dades que venien del dispositiu IoT. Els resultats respecte a material es poden traduir en, primerament l'aplicació inicial o bàsica on s'han desenvolupat tots els elements gràfics i el panell de reproducció. I segon, la migració de contingut a l'aplicació Atlantis Moto.

Respecte a resultats sobre els objectius del projecte, s'han pogut assolir els tres. Primer de tot, s'ha fet una anàlisi exhaustiva de les dades, la qual es pot trobar en la secció 6, que ha permès entendre quines dades venen i fer el seu processat pertinent. Feta la part d'anàlisi del projecte i obtingudes totes les dades a utilitzar, s'ha donat per completat aquest primer objectiu.

Pel que fa al segon i tercer objectiu, a continuació s'explica de manera molt resumida totes les funcionalitats fetes que comprenen:

- **Visualització del recorregut:** S'ha pintat la ruta a sobre del mapa digital i s'ha afegit un filtre que permet triar el tipus de ruta a visualitzar, segons la velocitat, acceleració o inclinació amb una escala de colors per cada opció.
- **Panell de reproducció:** Creació d'un panell amb diverses opcions per reproduir la ruta com ara reproduir/aturar, anar endavant o enrere d'1 a 1 i anar directament a la primera o l'última posició.
- **Modes de càmera:** Per complementar la reproducció de la ruta, s'han fet tres tipus de càmera que permeten tenir diferents perspectives de la ruta.
- **Lliscador de posicions:** És una forma alternativa de reproduir la ruta, s'ha fet un lliscador de posicions que permet situar-se a qualsevol punt de la ruta ràpidament.
- **Indicadors de dades:** Per la part de visualització de dades, s'han fet 4 rellotges. Un velocímetre, un acceleròmetre, un inclinòmetre i una brúixola. A més, s'ha fet un hodòmetre per representar el comptador total de kilòmetres.
- **Gràfics de dades:** S'han generat tres gràfics, un per mostrar l'acceleració en funció del temps i dos més per representar l'altitud.

- **Layout horitzontal:** Tant la meua aplicació com la d'Atlantis estaven fetes principalment en mode vertical o de retrat. S'ha ofert l'opció de poder orientar el dispositiu en mode apaïsat o horitzontal amb tota la interfície adaptada a la pantalla desenvolupada.

Totes les característiques desenvolupades s'expliquen amb més detall a la secció 7.

A l'apèndix A.5 es poden veure imatges de totes les funcionalitats fetes amb la versió final, a l'aplicació d'Atlantis Moto.

10 CONCLUSIONS

L'objectiu principal d'aquesta proposta de projecte era construir una aplicació mòbil per tal de representar les dades de diversos sensors d'un dispositiu IoT d'Atlantis IT. Es considera que s'ha assolit completament i, per tant, que el projecte ha estat un èxit.

Com a tal, l'aplicació mòbil va ser la primera fase del desenvolupament, per posteriorment traspasar tot el contingut a l'aplicació oficial d'Atlantis durant la segona fase i així disposar ja de la integració al producte final. De certa forma es pot considerar que la meua aplicació es va transformar en una pantalla nova d'Atlantis Moto.

Les sensacions a l'haver fet aquest projecte han estat positives, encara que la corba d'aprenentatge de Flutter hagi estat complicada, perquè hi ha hagut certes funcionalitats que al no haver fet mai res similar, es feien una mica difícil, però al final acabaven sortint. Sobretot la dificultat del projecte s'ha notat durant les primeres setmanes de desenvolupament, posteriorment la corba s'anava equilibrant i el projecte anava agafant forma.

Encara que els objectius s'hagin assolit completament, sí que hi ha certes característiques del projecte que s'han vist alterades:

- La representació de les dades d'inclinació, sigui amb una motocicleta inclinant-se o amb l'inclinòmetre, s'ha implementat utilitzant dades aleatòries dins d'un rang de -45 a 45 graus, rang que correspon a la inclinació mitjana d'una motocicleta. Aquesta alternativa s'ha degut al fet que Atlantis no ha pogut finalitzar la recollida de dades reals per problemes externs.
- Similarment, al cas anterior per l'absència de dades reals, en aquest cas les dades de frenada de la motocicleta s'han considerat com els valors negatius obtinguts en calcular l'acceleració segons la fórmula de l'acceleració mitjana[20].

El projecte ha adquirit un grau de maduresa elevat amb totes les funcionalitats desenvolupades, però encara es considera que té potencial per poder seguir en desenvolupament de cara al futur.

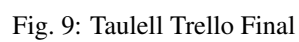
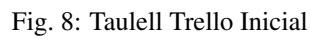
AGRAÏMENTS

M'agradaria aprofitar aquesta secció per agrair al meu tutor Sergi Carreras Sala per tota l'ajuda proporcionada durant el transcurs del projecte, tant resolent dubtes com guiant-me en el procés d'avançar el projecte. Per altra banda, també m'agradaria expressar el meu agraïment a Lluís Freixas, tutor per part d'Atlantis IT, que ha estat de gran ajuda durant tot el projecte conjuntament amb el personal de l'empresa que m'han anat proporcionant noves idees i millores a implementar i també ajudant en aspectes tècnics del projecte. Finalment, vull agrair a la meua família i amics pel suport i la motivació en moments difícils.

REFERÈNCIES

- [1] “¿Qué es IoT (Internet Of Things)?,” Deloitte Spain. <https://www2.deloitte.com/es/es/pages/technology/articles/IoT-internet-of-things.html> (accessed Feb. 20, 2024).
- [2] Izertis, “Encontrando cosas a través de la tecnología IoT,” www.izertis.com. <https://www.izertis.com/es/-/blog/encontrando-cosas-a-traves-de-la-tecnologia-iot> (accessed Feb. 20, 2024).
- [3] “Inicio - Atlantis IT,” www.atlantisioe.com, Dec. 03, 2019. <https://www.atlantisioe.com/es/> (accessed Feb. 21, 2024).
- [4] Atlassian, “Trello,” Atlassian. <https://www.atlassian.com/es/software/trello> (accessed Feb. 29, 2024).
- [5] Flutter, “Flutter - Beautiful native apps in record time,” Flutter.dev, 2019. <https://flutter.dev/> (accessed Feb. 17, 2024).
- [6] Dart, “Dart programming language,” dart.dev. <https://dart.dev/> (accessed Feb. 17, 2024).
- [7] “Postman API Platform,” www.postman.com. <https://www.postman.com/home> (accessed Mar. 02, 2024).
- [8] Android Studio, “Download Android Studio and SDK tools,” Android Developers, 2019. <https://developer.android.com/studio> (accessed Mar. 02, 2024).
- [9] “http — Dart Package,” Dart packages. <https://pub.dev/packages/http> (accessed Mar. 14, 2024).
- [10] “PageView class - widgets library - Dart API,” api.flutter.dev. <https://api.flutter.dev/flutter/widgets/PageView-class.html> (accessed Mar. 15, 2024).
- [11] “Cómo agregar Google Maps a una app creada con Flutter,” Google Codelabs. <https://codelabs.developers.google.com/codelabs/google-maps-in-flutter?hl=es-419#0> (accessed Mar. 04, 2024).
- [12] “Getting started with Flutter Cartesian Charts widget — Syncfusion,” help.syncfusion.com. <https://help.syncfusion.com/flutter/cartesian-charts/getting-started> (accessed Mar. 04, 2024).
- [13] “Getting started with Flutter Radial Gauge widget — Syncfusion,” help.syncfusion.com. <https://help.syncfusion.com/flutter/radial-gauge/getting-started> (accessed Mar. 10, 2024).
- [14] “Working with REST APIs — Flutter,” Codemagic blog, Feb. 24, 2022. <https://blog.codemagic.io/rest-api-in-flutter/> (accessed Mar. 03, 2024).
- [15] “Getting started with Flutter Range Slider widget — Syncfusion,” help.syncfusion.com. <https://help.syncfusion.com/flutter/range-slider/getting-started> (accessed Apr. 11, 2024).
- [16] “How to Design a Speedometer Using Flutter Radial Gauge — Syncfusion Blogs,” Feb. 03, 2020. <https://www.syncfusion.com/blogs/post/speedometer-using-flutter-radial-gauge.aspx> (accessed Feb. 22, 2024).
- [17] “odometer — Flutter package,” Dart packages. <https://pub.dev/packages/odometer> (accessed Apr. 2, 2024).
- [18] “Wikiloc — Rutes del Món,” Wikiloc — Rutes del Món. <https://ca.wikiloc.com/> (accessed Mar. 10, 2024).
- [19] “Relive: Corre, Pedalea y más,” Relive. <https://www.relive.cc/?hl=es> (accessed Mar. 10, 2024).
- [20] “Aceleración Media,” www.fisicalab.com. <https://www.fisicalab.com/apartado/aceleracion-media> (accessed Mar. 10, 2024).
- [21] “Seguridad e innovación: el mejor localizador GPS del mercado.” <https://www.komobimoto.com/> (accessed Jun. 07, 2024).

A.1 Taulell Trello

[illegible]

A.3 Resposta API Get trips info

```
{
  "initialPosition": {
    "latitude": 41.6208381652832,
    "longitude": 2.6650679111480713,
    "speed": 0.0,
    "altitude": 0,
    "heading": 0,
    "gpsTime": "2024-04-08T07:07:51.000Z",
    "gpsFix": false,
    "odometer": 44833.70783125,
    "addr": "Carrer de Joan Coromines i Vigneaux|Caletlla|Caletlla|08370|Barcelona|Espanya"
  },
  "finalPosition": {
    "latitude": 41.620849609375,
    "longitude": 2.664547920227051,
    "speed": 11.0,
    "altitude": 42,
    "heading": 328,
    "gpsTime": "2024-04-08T16:31:03.000Z",
    "gpsFix": true,
    "odometer": 45163.125,
    "addr": "Carrer de Saula i Pujals|Caletlla|Caletlla|08370|Barcelona|Espanya"
  },
  "totalTripDistance": 329.41796875,
  "totalTripFuel": 18.11798828125,
  "totalTripCost": 30.438219362173694,
  "totalTripTime": 368.8833333333333,
  "totalAverageSpeed": 53.588838002078345,
  "trips": [
    {
      ...
    }
  ]
}
```

Fig. 10: Get Info Part 1

```
"trips": [
  {
    "initialPosition": {
      "latitude": 41.6208381652832,
      "longitude": 2.6650679111480713,
      "speed": 0.0,
      "altitude": 0,
      "heading": 0,
      "gpsTime": "2024-04-08T07:07:51.000Z",
      "gpsFix": false,
      "odometer": 44833.70783125,
      "addr": "Carrer de Joan Coromines i Vigneaux|Caletlla|Caletlla|08370|Barcelona|Espanya"
    },
    "finalPosition": {
      "latitude": 41.622901916503906,
      "longitude": 2.6737849712371826,
      "speed": 0.0,
      "altitude": 39,
      "heading": 98,
      "gpsTime": "2024-04-08T07:19:48.000Z",
      "gpsFix": true,
      "odometer": 44834.8125,
      "addr": "Carrer del Garbí|el Poblenou|Pineda de Mar|08397|Barcelona|Espanya"
    },
    "tripTime": 11.95,
    "tripDistance": 1.10546875,
    "tripCost": 0.10214530931087211,
    "tripFuel": 0.060800781250000005,
    "averageSpeed": 5.550470711297072,
    "tracks": [
      ...
    ]
  }
]
```

Fig. 11: Get Info Part 2

```
"tracks": [
  {
    "deviceTime": 1712560347000,
    "routeId": 0,
    "lastGPSDateTime": 1712560347000,
    "speed": 0,
    "heading": 6,
    "longitude": 2.664989,
    "latitude": 41.62041,
    "altitude": 19,
    "satellite": 5,
    "vehicleMileage": 44833.707,
    "sensors": "[5.0|22.0|1712560076000|MOTION_STATUS|22888|ANALOG|0|199084",
    "actuators": "[false|1|1712559643000|DIGITAL_OUTPUT_1|22894|DIGITAL|0|",
    "gpsFix": false
  }
]
```

Fig. 12: Get Info Part 3

A.4 Pantalles de l'aplicació inicial



Fig. 13: Pantalla Principal

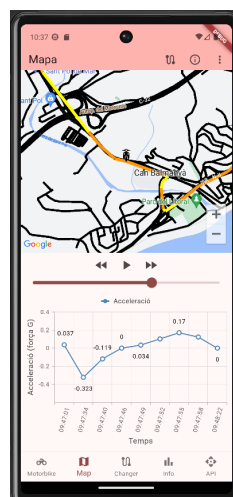


Fig. 14: Gràfic d'acceleració

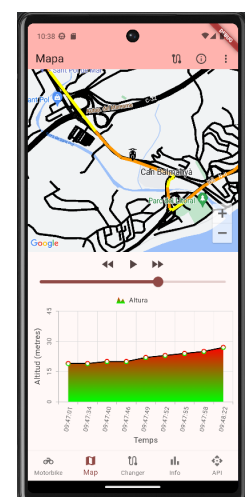


Fig. 15: Gràfic d'altitud

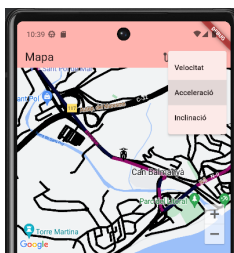


Fig. 16: Selector de Ruta

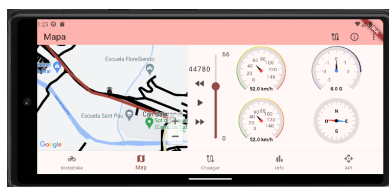


Fig. 17: Pantalla Principal apaïsat

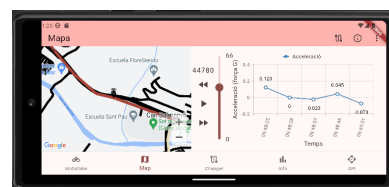


Fig. 18: Gràfic d'acceleració apaïsat

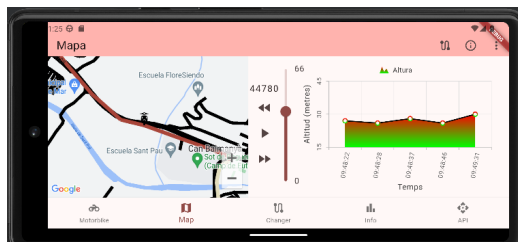


Fig. 19: Gràfic d'altitud apaïsat



Fig. 20: Selector Ruta Apaïsat

A.5 Pantalles de l'aplicació final

Les sigles AM fan referència a l'aplicació d'Atlantis Moto.

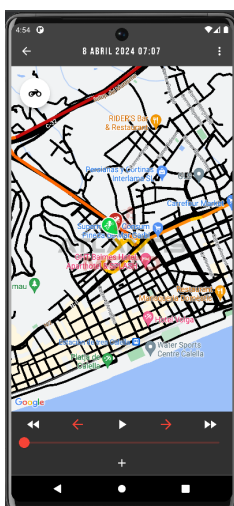


Fig. 21: AM - Pantalla Principal

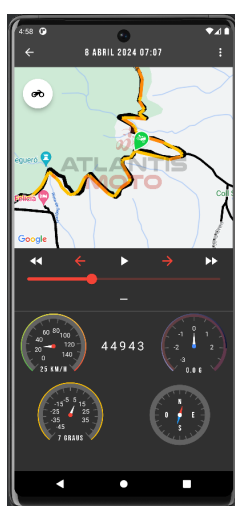


Fig. 22: AM - Indicadors de dades

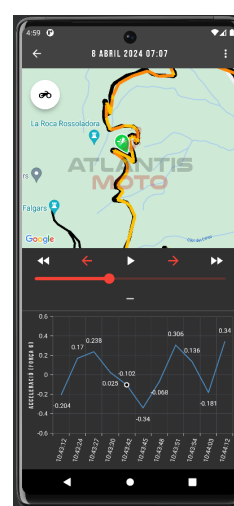


Fig. 23: AM - Gràfic d'acceleració

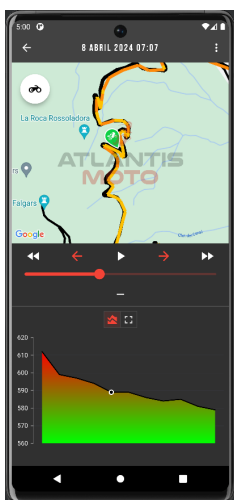


Fig. 24: AM - Gràfic d'altitud actual

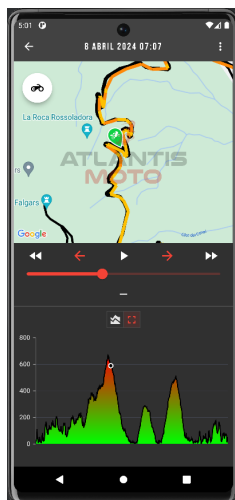


Fig. 25: AM - Gràfic d'altitud sencer

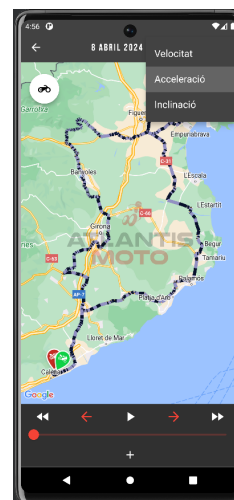


Fig. 26: AM - Selector Ruta



Fig. 27: AM - Main Landscape

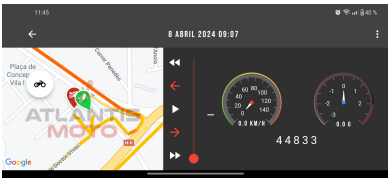


Fig. 28: AM - Indicadors 1

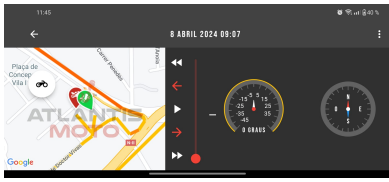


Fig. 29: AM - Indicadors 2

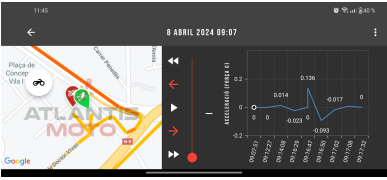


Fig. 30: AM - Gràfic Acceleració

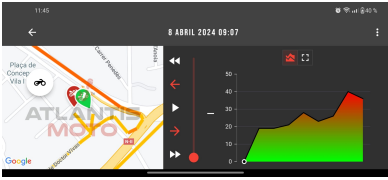


Fig. 31: AM - Gràfic Altitud Actual

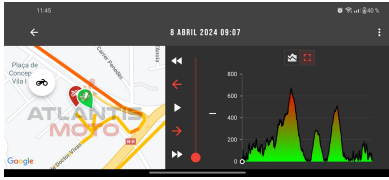


Fig. 32: AM - Gràfic Altitud Sencer



Fig. 33: Mode Seguint



Fig. 34: Mode Bloc

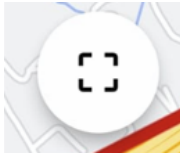


Fig. 35: Mode Lliure



Fig. 36: Inclinòmetre

A.6 Requisits del sistema

Llista de requisits funcionals:

ID	Descripció
RF1	El sistema ha de permetre pintar el recorregut de la ruta sobre el mapa digital
RF2	El sistema ha de permetre reproduir la ruta afegint un punter que sobre el mapa digital.
RF3	El sistema ha de pintar la ruta en funció d'un filtre.
RF4	El sistema ha de contenir, com a mínim, dues eines de visualització.
RF5	El sistema ha de mostrar al costat de la ruta, indicadors de velocitat, acceleració i frenada.
RF6	El sistema ha de permetre visualitzar un gràfic que indiqui l'acceleració en cada punt de la ruta.
RF7	El sistema ha de representar la velocitat en un velocímetre.

Llista de requisits no funcionals:

ID	Descripció
RNF1	El sistema ha de mostrar tres opcions per filtrar la ruta: velocitat, acceleració i inclinació.
RNF2	El sistema ha de mostrar un menú de reproducció de la ruta que consti com a mínim de tres opcions: Enrere, Reproduir/Aturar i Avançar.
RNF3	El sistema ha de calcular el gràfic d'acceleració en funció de l'acceleració i frenada de cada punt.
RNF4	El sistema ha de funcionar en dispositius Android i iOS.
RNF5	El sistema ha de tenir una interfície d'usuari senzilla i intuïtiva.
RNF6	El sistema ha de ser desenvolupat amb el framework Flutter i el llenguatge Dart JS.