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

García Sánchez, Óscar; Bartrina Rapesta, Joan, dir. Control Presencial amb Dispositiu Mòbil. 2023. (Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/280705>

under the terms of the  license

Control Presencial amb Dispositiu Mòbil

Oscar Garcia Sanchez

Resum— Aquest treball es centra en el desenvolupament d'una aplicació mòbil per registrar la presencialitat d'usuaris en un esdeveniment. L'objectiu és minimitzar la intervenció dels usuaris i garantir una integració funcional independent del sistema operatiu. S'explora l'ús de la tecnologia Bluetooth Low Energy (BLE)[1] per permetre la transmissió de dades de manera eficient i segura. L'aplicació ofereix la capacitat de funcionar en mode offline[2], mantenint les dades d'assistència al dispositiu fins que es recuperi la connexió a Internet. També es destaca la seva llibertat i personalització en la creació d'esdeveniments i la seva privadesa i seguretat, ja que les dades es queden al dispositiu i es transmeten localment a través de Bluetooth. Es preveu mantenir un registre de les dades d'assistència per utilitzar-les segons les necessitats de l'administrador.

Paraules clau— Bluetooth, Bluetooth Low Energy, React Native, Aplicació Mòbil, Adroid, IOS, Multi-Plataforma

Abstract— This paper focuses on the development of a mobile application to register the presence of users in an event. The goal is to minimize user intervention and ensure functional integration independent of the operating system. The use of Bluetooth Low Energy technology to enable efficient and secure data transmission is explored. The application offers the ability to work in offline mode, keeping attendance data on the device until the Internet connection is restored. Also highlighted is its freedom and customization in event creation and its privacy and security, as data is left on the device and transmitted locally via Bluetooth. It is planned to keep one of the usage data for use according to the registration needs of the administrator.

Keywords— Bluetooth, Bluetooth Low Energy, React Native, Mobile Application, Adroid, IOS, Cross-platform



1 INTRODUCCIÓ - CONTEXT DEL TREBALL

ACTUALMENT l'assistència/presencialitat a les classes i altres esdeveniments de la universitat no està sent registrada de cap manera, en part per què en la majoria dels casos és lliure i "opcional", però també es deu a la falta d'un mecanisme simple i segur per prevenir conductes fraudulentament en aquest àmbit.

Tot i així existeixen situacions com exàmens i pràctiques en les quals seria notable poder comptar amb la llista d'assistents. Algunes assignatures utilitzen un qüestionari al Campus Virtual[3] de la universitat per poder comptabilitzar l'assistència.

No obstant això, aquest mètode no resulta pràctic, ja que requereix la intervenció dels alumnes i, a causa de la seva transparència, és inevitable que es produeixin filtracions de

la metodologia, com la divulgació d'un codi d'accés a persones que no es troben físicament a la classe i que, per tant, poden marcar assistència sense estar presents.

De l'ambició de poder comptar amb un recurs o eina que ens permeti enregistrar l'assistència d'una manera consistent i segura neix la idea del control presencial a través del dispositiu mòbil i una tecnologia de curt rang, per tractar d'evitar així possibles falsos positius.

2 ESTAT DE L'ART

Els sistemes de control de presencialitat són eines essencials per gestionar l'assistència i la presència del personal en diverses àrees com ara l'educació, l'empresa i altres sectors. Aquests sistemes permeten recollir dades objectives sobre l'assistència dels empleats o alumnes, facilitant la presa de decisions, l'optimització dels recursos i el compliment de les normatives vigents.

En els últims anys, s'han produït importants avenços en la tecnologia de control de presencialitat[4], que han millorat la seva precisió, eficiència i facilitat d'ús. Algunes de les tendències destacades en aquest àmbit són Biome-

- E-mail de contacte: ogarsan@pm.me
- Menció realitzada: Tecnologies de la Informació
- Treball tutoritzat per: Joan Bartrina Rapesta (departament)
- Curs 2022/23

tria, Targetes d'accés i codis QR[5] o dispositius i aplicacions mòbils. Amb l'avanç de les tecnologies mòbils, s'han desenvolupat aplicacions específiques per gestionar la presencialitat. Aquestes aplicacions permeten als empleats o alumnes marcar la seva assistència des dels seus propis dispositius mòbils, evitant la necessitat de hardware[6] addicional. A més, aquestes aplicacions poden integrar altres funcionalitats com ara la gestió d'horaris, comunicació interna i notificacions.

Existeixen diferents opcions com QuickBooksTime[7] i AttendanceBot[8] que com s'ha exposat tenen més utilitat en l'àmbit de gestió de temps i control del personal d'acord amb l'activitat dins l'aplicació. Aquestes aplicacions funcionen a través d'internet per la qual cosa no ocupen el mateix lloc que el projecte dut a terme.

AttendAce presenta algun avantatge enfront aquestes aplicacions a causa de algunes de les seves característiques:

1. Capacitat de funcionament offline: Un altre avantatge de les aplicacions de control presencial basades en Bluetooth és la seva capacitat per funcionar en mode offline. Això significa que les dades d'assistència es poden registrar en el dispositiu fins que es restauri la connexió a Internet o es realitzi una sincronització posterior. Això és útil en situacions en què la connexió a Internet es perd per un període de temps i es vol garantir la integritat de les dades d'assistència.
2. Llibertat i personalització: Ofereix una llibertat total a l'hora de crear l'event a assistir, que pot ser des d'un examen d'universitat fins a una escapada a la muntanya.
3. Privadesa i seguretat: Com l'aplicació no requereix connexió a Internet, les dades de control presencial es queden al dispositiu i es transmeten localment a través de Bluetooth. Això pot ser considerat més segur per a aquells que tenen preocupacions sobre la privadesa i la seguretat de les seves dades. Les dades no es transmeten a través de xarxes externes, reduint el risc d'intercepció o accés no autoritzat.

3 OBJECTIUS

3.1 Escollir Tecnologia per les comunicacions

Es busca trobar mètode de transmissió de missatges més adients per a dur a terme el projecte. La tecnologia ens ha de permetre, verificar la localització dels assistents i poder enviar i rebre missatges curts eficientment per poder registrar als assistents ràpidament i còmodament. A més es busca la mínima intervenció possible de l'usuari, perquè el procés sigui transparent i invisible tant pel professor com per l'alumne.

3.2 Aconseguir la transmissió de dades

Per tal d'assolir la transmissió de dades en un sistema de control presencial basat en Bluetooth, és necessari obtenir el control de l'Application Programming Interface (API)[9] de la tecnologia seleccionada. Això permet iniciar la transmissió de dades visibles per altres dispositius mitjançant la mateixa tecnologia. És important definir el format de

les dades i assegurar la seva compatibilitat amb altres dispositius. Durant el procés de transmissió, cal supervisar i gestionar els errors per garantir la integritat de les dades. Aquesta comunicació efectiva és essencial per al sistema de control presencial.

3.3 Rebre les dades emeses per un altre dispositiu

Per tal de rebre les dades emeses pels dispositius en un sistema de control presencial basat en Bluetooth, és necessari establir un procés d'escaneig eficient per detectar els dispositius presents. A continuació, s'ha d'interpretar el contingut dels missatges (PAYLOAD) per obtenir informació rellevant com la identitat de l'usuari, la data i l'hora d'entrada/sortida. Una vegada interpretats, es realitzen les accions pertinents, com ara registrar l'assistència o actualitzar bases de dades. Cal tenir en compte la gestió de recursos, la sincronització adequada i les mesures de seguretat i privadesa de les dades tractades durant aquest procés.

3.4 Escollir estructura del missatge a transmetre

Per garantir el funcionament de l'aplicació, és crucial dissenyar l'estructura del missatge, decidir quin tipus de dades s'inclouran i com s'enviaran. Això implica establir un format de missatge coherent i estàndard, determinar les dades rellevants com la identitat de l'usuari i l'hora d'entrada/sortida, i assegurar-se que les dades es transmetin de manera segura. També cal tenir en compte les regulacions de privadesa i protecció de dades per garantir la confidencialitat i la integritat de les dades. A més, és important establir protocols per gestionar situacions de connexió fallida o errors de transmissió. Aquesta planificació curosa assegurarà el correcte funcionament de l'aplicació i la protecció de la informació dels usuaris.

3.5 Mantenir registre dels assistents amb les dades rebudes

Finalment es vol obtenir una persistència de les dades d'assistència registrades, per a ser utilitzades a discreció de l'administrador, ja sigui per imposar un control, com a mesura d'interès, o informació estadística.

4 METODOLOGIA

Aquest projecte s'està duent a terme fent servir la metodologia Agile. Es fa ús de GitHub com a eina de control de versions per carregar codi estable a mesura que avança el projecte. Després de cada sprint[12] definit com a períodes de dues setmanes, es duen a terme reunions amb els stakeholders[10] (interessats en el projecte) per revisar els canvis, les noves implementacions i funcionalitats aconseguïdes. D'aquesta manera es té en compte el feedback[11] del client per satisfer les expectatives.

4.1 Planificació

A la figura 1, es pot observar que inicialment el projecte es va dividir en cinc activitats principals organitzades

per sprints, incloent-hi el desenvolupament i la recerca d'informació. A causa de la incertesa sobre la durada de cada tasca, la càrrega de treball es va dividir en tasques amb càrregues de treball i durades similars. A cada tasca se li va assignar un període recomanat d'entre un o dos sprints, tenint en compte els quatre mesos de durada del projecte.

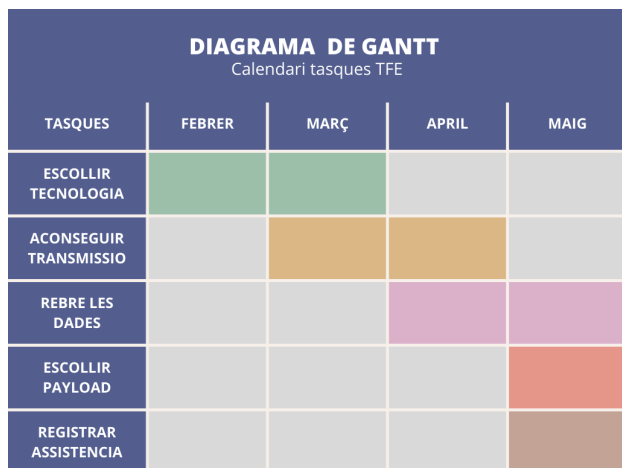


Fig. 1: Planificació inicial

5 DESENVOLUPAMENT

5.1 Preparar entorn de desenvolupament

Pel desenvolupament de l'aplicació es va escollir Expo[13], una plataforma i conjunt d'eines per a desenvolupar aplicacions mòbils utilitzant el framework[14] de React Native[15]. Proporciona una capa d'abstracció per simplificar el procés de desenvolupament i desplegament d'aplicacions React Native, eliminant la necessitat de configurar i gestionar aspectes tècnics complexos.

Expo inclou diverses característiques i avantatges que fan que sigui una opció atractiva per a desenvolupadors d'aplicacions mòbils, de fet des de Marc de 2023 és el nou estàndard recomanat per la documentació de React:

Expo ofereix una sèrie de components, llibreries i funcionalitats predefinides que faciliten el desenvolupament d'aplicacions mòbils com accés a API i funcionalitats del dispositiu, proporciona eines per a la gestió de dependències i un sistema de construcció simplificat.

A més es fa servir Expo Go, una aplicació complementària per a dispositius mòbils que permet executar i provar les aplicacions Expo directament en els dispositius mòbils.

Amb aquesta combinació d'eines es disposa d'un entorn de desenvolupament àgil per poder dur a terme la implementació de les funcionalitats requerides per l'aplicació i desplegar les solucions finals fent ús d' Expo CLI (Command Line Interface), que facilita el desplegament de l'aplicació en dispositius reals o en emuladors. Finalment, ofereix la possibilitat de generar fitxers binaris (instal·lables) d'aplicació (APK per a Android i IPA per a iOS) amb un sol clic.

5.2 Escollir Tecnologia per les comunicacions

A l'hora de triar una tecnologia, es van plantejar dues opcions inicials, i Near Field Communication (NFC)[16]. Es va discutir també la utilització d'una verificació a través de la connexió Wi-Fi del campus en combinació amb una geolocalització per direcció Internet Protocol (IP)[17], però la idea va ser descartada degut a la falta de precisió de la geolocalització, que permetia registrar assistència des d'altres espais de la facultat.

Pel que fa a BT es va trobar que tenia certes restriccions, com ara una connexió virtual d'un màxim de 7 dispositius simultanis[18], utilitzant diferents serveis, o només una amb cada servei. A més requereix la interacció dels assistents per permetre la connexió, el qual no disposaria del nivell de transparència desitjat. Amb NFC es requeriria que els alumnes s'apropessin a la taula del professor per entrar en contacte amb el dispositiu de registre, això suposava una millora, però no seria la situació òptima.

Finalment, es va decidir escollir BLE com a tecnologia per a les comunicacions, a causa de la simplicitat de transmetre informació a través de missatges d'anunci (advertising).

Els missatges d'advertising permeten als dispositius compartir dades de manera eficient i amb una càrrega baixa. A diferència dels mètodes tradicionals de transferència de dades, no es requereix establir una connexió prèvia entre els dispositius ni trencar-la després de finalitzar la transferència. En aquest context, hi ha dos rols principals: el dispositiu emissor, responsable de l'enviament de les dades, i el dispositiu central, encarregat d'escoltar i rebre aquestes dades.

El dispositiu central pot sol·licitar dades addicionals de l'emissor i obtenir més informació del que pot ser transmès en un sol paquet d'advertising. Aquestes dades poden incloure informació sobre els sensors de batecs del cor, sensors de temperatura o qualsevol altre tipus d'informació rellevant. Sovint, els serveis que ofereix el perifèric o emissor són anunciats mitjançant aquests missatges, indicant les capacitats de detecció i les dades específiques que estan disponibles. Cada servei té un identificador únic, conegut com a identificador universal (UUID)[19], que permet a la central identificar-lo i prendre decisions sobre la sol·licitud de més dades relacionades amb aquest servei.

En la figura 2 podem observar un exemple de la comunicació entre dos dispositius que implementen BLE, un desenvolupa el rol de perifèric, un dispositiu que recull informació i que l'emet fent servir Advertising i l'altre porta a terme el rol de Central, que escaneja l'entorn, troba els missatges d'anunci i els interpreta. Aquí tens el text corregit:



Fig. 2: Diagrama de Missatges Advertising

5.3 Aconseguir la transmissió de dades

En el desenvolupament d'aquest punt, es van fer evidents algunes de les limitacions de desenvolupar una aplicació mòbil amb Bluetooth (BT).

S'han d'utilitzar dispositius reals per a les proves, ja que els emuladors no permeten l'accés als controladors de Bluetooth de la màquina de desenvolupament. A més, la solució que ofereix la plataforma on es desenvolupa l'aplicació, Expo, no permet desplegar aplicacions directament a través d'Expo Go que requereixin de codi natiu. I pel fet que actualment Expo no disposa d'un mòdul de BT, no és possible provar el funcionament en dispositius amb sistema operatiu d'Apple (iOS) fàcilment, a causa de la manca d'un ordinador Apple i un compte del programa de desenvolupadors d'Apple.

Finalment, a través del kit de desenvolupament de programari (SDK) d'Android i un dispositiu físic, es va poder començar a provar l'aplicació i realitzar el desenvolupament.

Primerament, s'havia d'obtenir el control de l'API de Bluetooth del dispositiu, per la qual cosa es va buscar una llibreria per poder interactuar amb aquestes funcionalitats.

Es van tenir en compte les següents opcions: `react-native-ble-plx`[22][23] i `react-native-ble-advertiser`[24]. La primera disposa d'una àmplia documentació i ofereix una interfície senzilla i intuïtiva per interactuar amb els dispositius BLE. Proporciona mètodes i funcions ben definits per connectar-se a dispositius, llegir i escriure dades, i gestionar esdeveniments relacionats amb la connexió i les dades rebudes. Bàsicament, porta a terme el rol de Central a les connexions BLE, però careix de la capacitat d'emetre els missatges d'anunci que es requereixen per a l'aplicació. La segona és molt més limitada en quant a personalització i no disposa de documentació, però té la capacitat de crear un Advertiser (component emissor) per poder emetre els missatges i de crear un EventListener (component Central) per poder captar i llegir els missatges BLE, permetent així realitzar ambdós rols al mateix temps.

Després d'analitzar les possibilitats, es podia optar per una solució híbrida, utilitzant `react-native-ble-plx` per a realitzar el rol de Central i `react-native-ble-advertiser` per obtenir un emissor, o centrar-se únicament en la segona llibreria per al desenvolupament de tota l'aplicació. Es va decidir que l'enviament i la recepció de les dades es gestionarien amb la llibreria `react-native-ble-advertiser`, que implementa un Advertiser i un Scanner de BLE, per la facilitat que aportava utilitzar una única llibreria.

Per decidir el mètode d'enviament de les dades, es disposaven de diverses opcions:

1. Modificar el paràmetre del fabricant (MANUFACTURER_DATA) per contenir el missatge desitjat.
2. Modificar el nom de dispositiu dels paquets enviats.
3. Modificar l'ID del servei ofert pel dispositiu (SERVICE UUID).

Finalment, es va optar per l'opció de modificar l'UUID de servei que s'inclou a l'encapçalament del paquet de BLE, ja que era el mètode més simple que complia amb les restriccions, i l'UUID era prou gran per contenir les dades

requerides sense afectar l'eficiència. A més, això permetia un altre mètode d'escaneig basat en UUID de servei (scan-ByService).

A la següent imatge es pot observar la composició d'un paquet d'anunci de BLE. El camp PDU (Unitat de Dades de Protocol) és el camp que canvia en funció del tipus de paquet que s'està transmetent (dades o anunci), i és el que inclou el missatge que volem enviar.

Al camp Header es troba el tipus de paquet a enviar, el qual es va decidir que seria `ADV_SCAN_IND`. Aquest tipus de paquet indica que el dispositiu està fent Advertising, que no permet les connexions, és escanejable i que l'advertising no és dirigit.

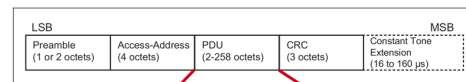


Figure 2.1: Link Layer packet format for the LE Uncoed PHYs



Figure 2.4: Advertising physical channel PDU

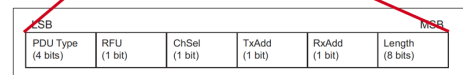


Figure 2.5: Advertising physical channel PDU header

Fig. 3: Estructura del paquet d'anunci de BLE

Dins del PAYLOAD, que està estructurat com una llista de diferents camps d'informació, on cada camp té un tipus i les seves dades definits, es troba el missatge que es vol transmetre, i que és un dels tipus disponibles d'Advertise Data[25]. Juntament amb l'UUID de servei, s'envien altres camps com el nom del dispositiu, el MANUFACTURER_DATA i la potència de transmissió (`txPowerLevel`)[26].

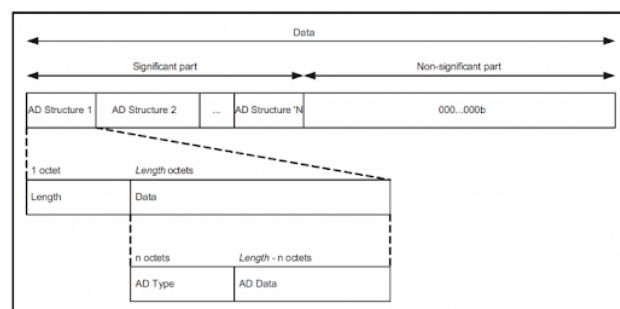


Fig. 4: Contingut del PAYLOAD

5.4 Rebre les dades emeses per un altre dispositiu

Per provar la recepció de les dades, es va requerir un segon dispositiu Android on desplegar l'aplicació de la mateixa manera que a l'emissor. El primer concepte tractava de reconèixer qualsevol dispositiu, independent del missatge enviat, recopilar els dispositius a una llista i poder identificar els diferents usuaris o assistents. Es van trobar problemes a l'hora d'obtenir alguns dels permisos al segon dispositiu

per diferències en la seguretat dels fabricants, però es van solucionar canviant la configuració per debugar del telèfon. S'ha observat que continuen existint limitacions amb dispositius Xiaomi. Els dispositius Xiaomi, juntament amb alguns altres dispositius Android, tenen restriccions addicionals i funcions d'estalvi d'energia agressives que poden interferir amb la funcionalitat del Bluetooth.

El setup pel desenvolupament era senzill, ambdós dispositius connectats a l'ordinador i desplegant l'aplicació al mateix temps en els dos clients. Un dels dispositius feia l'acció d'emetre els missatges estàndard i l'altre s'encarregava d'escanejar l'entorn per descobrir el dispositiu emissor. L'escaneig es realitza de manera contínua, per la qual cosa es poden actualitzar els paràmetres dels dispositius ja registrats, com ara la distància i la intensitat del senyal, a més d'afegir-ne de nous. Es va concloure sense més inconvenients i es va provar la funcionalitat a diferents distàncies per comprovar la resiliència de l'aplicació.

S'ha desenvolupat el component "EventAssistance" que permet difondre el propi dispositiu i escanejar altres dispositius per registrar l'assistència dels participants.

Primer, s'importen els mòduls i components necessaris de React Native, així com els mòduls relacionats amb la comunicació BLE. També s'importen el mòdul "Username-Context" i el mètode "calculateDistance" des de fitxers locals.

A continuació, es configura la constant "APPLEID" que representa l'identificador de l'empresa d'Apple per a la detecció d'iPhones mitjançant BLE.

El component "EventAssistance" rep la propietat "route" des de la qual obté la informació de l'event a registrar, que inclou, entre d'altres, la llista d'administradors de l'event, la qual és crucial per saber si l'usuari actual té permisos d'administrador o no.

Dins del component, es defineixen diversos estats. Per exemple, hi ha estats per controlar si s'està difonent o escanejant (isBroadcasting i isScanning) i altres, com "devicesFound", s'utilitzen per emmagatzemar la llista de dispositius trobats.

Hi ha funcions auxiliars com "add

Device", que s'encarrega d'afegir un dispositiu trobat a la llista de dispositius. També hi ha funcions per començar i aturar la difusió i l'escaneig, startBroadcast i startScan, respectivament.

En la part de la renderització, el component retorna una vista segura (SafeAreaView) que conté diverses seccions amb textos informatius, com ara el títol i el codi de l'event. Hi ha botons per començar/aturar la difusió i l'escaneig, depenent del rol de l'administrador. Si l'usuari és un administrador de l'esdeveniment, es mostra el botó per escanejar, una llista de dispositius trobats i botons per netejar la llista i desar l'assistència.

En resum, aquest component permet gestionar la difusió i l'escaneig de dispositius per registrar l'assistència a un esdeveniment. Proporciona funcionalitats per controlar la difusió del propi dispositiu, detectar dispositius propers i emmagatzemar l'assistència en una llista.

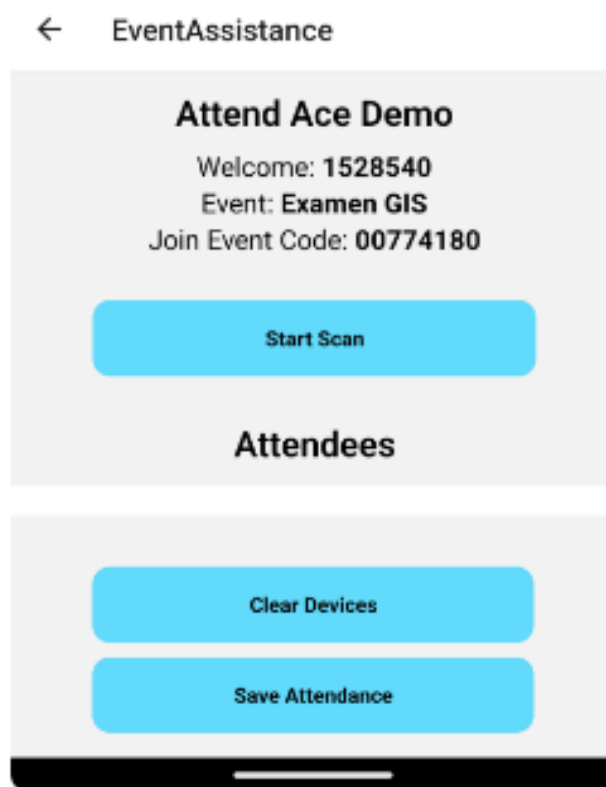


Fig. 5: Pantalla Registre assistència (Admin) (Captura Retallada)

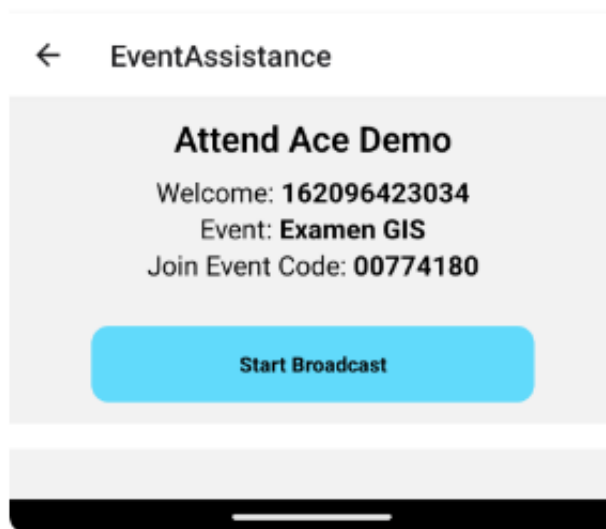


Fig. 6: Pantalla Registre assistència (Usuari) (Captura Retallada)

5.5 Escollir l'estructura del missatge a transmetre

Un cop demostrada la transmissió i recepció de dades, era imprescindible definir l'estructura del missatge. Es va decidir que havia de contenir dos factors importants: un identificador per a l'usuari i un per a l'esdeveniment, que permet a l'aplicació filtrar tant per usuaris registrats com per esdeveniment. Aquesta funcionalitat pot ser configurable en cas de disposar prèviament de la llista d'assistents o no. Com s'ha mencionat anteriorment, s'utilitza l'UUID del servei

per enviar aquests paràmetres, que segueix el format: uuid: "89537374-aa7c-4b87-bf25-000001528540".

Es va decidir generar-lo de la següent manera: primerament es genera un UUID aleatori fent servir la llibreria "react-native-uuid-generator". Els primers 8 caràcters serien l'ID de l'esdeveniment, generat aleatòriament per una funció a partir del nom de l'esdeveniment. L'identificador de l'usuari seria col·locat a la part final de l'UUID, amb una longitud màxima de 12 caràcters. A l'exemple proporcionat, podem trobar l'esdeveniment amb codi "89537374" i l'ID d'usuari "1528540", mentre que els caràcters del mig formen part de l'UUID aleatori generat a l'inici.

Tenint en compte la restricció autoimposada de no requerir una connexió a internet, s'han de definir totes les dades necessàries des de la mateixa aplicació. Per aquest motiu, es defineixen diferents components.

Per a la creació dels esdeveniments, disposem del component "EventCreation". Aquest component permet als usuaris introduir diversa informació, com ara el nom de l'esdeveniment, el tipus, la data d'inici, la durada, el nombre d'assistents i els administradors de l'esdeveniment.

En aquest component, s'utilitzen diverses llibreries i components de React Native, com el "DateTimePicker-Modal" per seleccionar dates i hores, el "Picker" per seleccionar el tipus d'esdeveniment i diversos components d'entrada de text per capturar les dades introduïdes pels usuaris.

Es declaren diversos estats utilitzant el hook useState per mantenir l'estat del component. Aquests estats s'utilitzen per emmagatzemar el nom de l'esdeveniment, el codi, el tipus, la data d'inici i finalització, la data límit per registrar-se, el nombre d'assistents, els administradors i per controlar la visibilitat d'alguns elements, com el selector de dates i l'entrada de l'administrador.

A més, s'utilitza el hook useContext per obtenir el context del nom d'usuari actual mitjançant el component UsernameContext.

El component també inclou diverses funcions auxiliars, com readAndModifyFile i createFileWithInitialContent, per gestionar l'escriptura i lectura d'un fitxer de base de dades local. A més, hi ha altres funcions, com handleEventCreation, handleAdminSubmission, showDatePicker, hideDatePicker, fixDate, handleConfirmStart, addTime, handleEventName i goToMainPage, per gestionar diverses accions i la navegació.

En la part de renderització, el component retorna una vista segura (SafeAreaView) que conté diversos components d'entrada de text i selecció, així com botons per interactuar amb el component. També es mostren informacions com la llista d'administradors i es controla la visibilitat d'alguns elements basant-se en certes condicions.

En resum, aquest component permet als usuaris crear nous esdeveniments mitjançant l'ompliment de formularis i la selecció de diverses opcions. Un cop s'ha introduït la informació necessària, es pot crear l'esdeveniment i realitzar altres accions, com afegir administradors o navegar a altres pàgines de l'aplicació.

Algunes de les dades que es poden especificar per a cada esdeveniment són les següents:

1. El títol de l'esdeveniment.

2. El tipus d'esdeveniment, que es selecciona dins d'un conjunt de tipus d'esdeveniments definit prèviament.
3. La data de començament de l'esdeveniment, especificant dia i hora en un selector intel·ligent.
4. La duració de l'esdeveniment en minuts.
5. El nombre d'assistents esperats.
6. El màxim de temps de retard en què està permès marcar assistència.
7. Llista d'administradors, on el creador es troba de manera predefinida.

Fig. 7: Pantalla creació d'esdeveniments

Quant al Registre d'esdeveniments pels usuaris, es realitza utilitzant el component "EventRegistration" que

s'utilitza per gestionar el registre d'esdeveniments en una aplicació. Aquest component és una pantalla de registre d'esdeveniments i inclou diverses funcions i mètodes per capturar les dades dels esdeveniments i gestionar-les.

Dins del component, es defineixen dues variables d'estat utilitzant el "useState" de React. Aquestes variables d'estat són "eventName" i "eventCode", que s'utilitzen per capturar i emmagatzemar el nom i el codi de l'esdeveniment que es vol registrar.

S'inclouen dos camps de text per capturar el nom i el codi de l'esdeveniment, i un botó "Join Event" que s'activa només quan els camps tenen dades vàlides. Per al registre, es requereix que el codi de l'esdeveniment hagi estat compartit amb l'usuari que es vulgui registrar.

Finalment, disposem del component "EventSelection" que s'utilitza per seleccionar els esdeveniments registrats o creats. Aquest component mostra una llista d'esdeveniments disponibles i proporciona opcions per registrar nous esdeveniments o eliminar la base de dades d'esdeveniments.

Dins del component, s'estableixen dues variables d'estat utilitzant el "useState" de React. La variable d'estat "events" s'utilitza per emmagatzemar una llista d'esdeveniments que es mostrarà en la llista. La variable "dbPath" s'estableix com la ruta del fitxer de base de dades utilitzant "FileSystem.documentDirectory" i el nom del fitxer "eventsDB.json".

A continuació, es defineixen dues funcions de navegació: "goToEventController" i "goToEventRegistration". La primera funció navega a la pantalla "EventAssistance" amb l'esdeveniment seleccionat com a paràmetre. La segona funció navega a la pantalla "EventRegistration" per registrar un nou esdeveniment.

La funció "getEventsDB" és una funció asíncrona que llegeix el contingut del fitxer de base de dades utilitzant "FileSystem.readAsStringAsync" i el converteix en un objecte JSON. A continuació, s'extreuen els objectes d'esdeveniments de la llista "eventList" i es crea un nou array d'esdeveniments "eventArray" amb una clau pròpia i l'objecte d'esdeveniment corresponent. Finalment, s'estableix l'array d'esdeveniments a través de la funció "setEvents".

A continuació, es defineix la funció "renderEventItem" que es crida per renderitzar cada esdeveniment a la llista. Aquesta funció mostra un botó amb el nom de l'esdeveniment i crida a la funció "goToEventController" quan es prem.

En resum, el component "EventSelection" mostra una llista d'esdeveniments disponibles, permet registrar nous esdeveniments i eliminar la base de dades d'esdeveniments. Utilitza el sistema de fitxers per accedir al fitxer de base de dades i actualitzar-lo amb les dades corresponents.

5.6 Mantenir registre dels assistents amb les dades rebudes

Per tal de mantenir un registre exhaustiu dels assistents a un esdeveniment, és necessari interpretar els missatges rebuts pels emissors. Per cada missatge escanejat, s'ha de determinar si pertany a l'esdeveniment en qüestió i extreure la identificació dels usuaris per a poder-los registrar adequadament. Una vegada finalitzat el procés de registre i tancada la

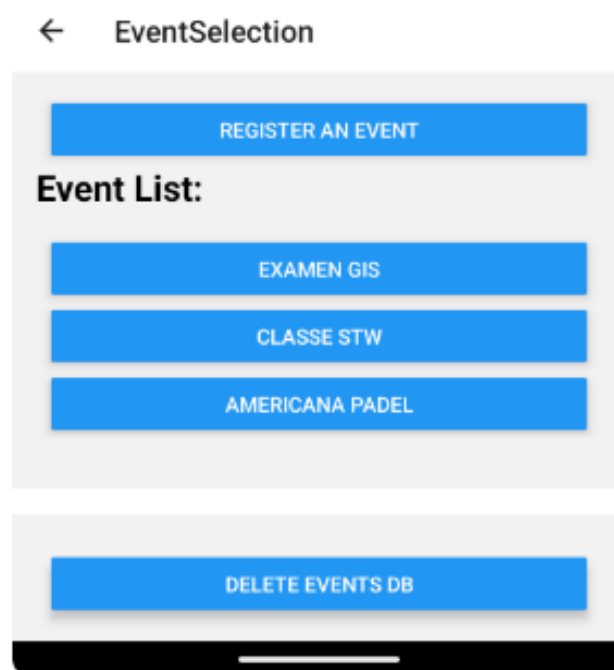


Fig. 8: Pantalla selecció d'esdeveniments (Captura Retallada)

recepció de nous usuaris, es proporciona una opció per exportar les dades dels assistents, incloent la seva identificació i l'hora de registre a l'esdeveniment.

Per a l'exportació de les dades, es posa a disposició la generació d'un arxiu en format de valors separats per comes (CSV)[33]. Aquest tipus d'arxiu permet guardar les dades tant en el dispositiu com compartir-les a través del mitjà desitjat. L'arxiu CSV és interpretable, el que significa que les dades poden ser llegides i processades de manera fàcil i senzilla per altres aplicacions o sistemes informàtics. Això facilita la seva integració amb altres plataformes o la realització d'anàlisis i tractament posterior de les dades registrades.

En resum, per mantenir un registre complet dels assistents a un esdeveniment, és necessari interpretar els missatges rebuts i extreure la informació rellevant per al registre dels usuaris. Una vegada finalitzat el procés de registre, es pot exportar la informació en un arxiu CSV, el qual és fàcilment llegible i interpretable per altres aplicacions o sistemes informàtics com Excel[34] o Google Sheets[35]. Aquesta funcionalitat permet una gestió més eficient i versàtil de les dades dels assistents a l'esdeveniment.

D'aquesta funcionalitat s'encarrega el component "AttendanceExporter" que rep com a paràmetre la ruta (route) que conté les dades d'assistència. A continuació, es defineix una funció "generateCSVString" que crea una cadena de text en format CSV a partir de les dades d'assistència rebudes. Aquesta cadena conté els noms dels assistents i l'hora d'entrada, separats per comes.

Seguidament, hi ha dues funcions, "saveCSVFile" i "shareCSVFile", que es fan servir per a guardar i compartir el fitxer CSV generat. Aquestes funcions utilitzen el mòdul "FileSystem" per escriure la cadena de text CSV en un fitxer amb l'extensió .csv i obtenir la seva ruta d'emmagatzematge. A continuació, es fa servir el mòdul "Share" per a compartir el fitxer amb altres aplicacions

mitjançant el títol del missatge, la cadena de text CSV i la seva ruta d'accés.

Finalment, el component retorna una vista (View) que conté dos botons ("Save CSV" i "Share CSV") per a l'usuari. Quan es prem un dels botons, s'executa la funció corresponent ("saveCSVFile" o "shareCSVFile") per a realitzar l'acció desitjada.

6 CONCLUSIONS

Després d'haver estudiat el projecte, la seva viabilitat i desenvolupat un prototip per a l'aplicació, s'han extret les següents conclusions:

- El treball realitzat demostra que el control presencial amb dispositiu mòbil és una solució viable i eficient per a la recopilació d'informació d'assistència, oferint avantatges com la simplicitat d'ús, la flexibilitat i la integració independent del sistema operatiu.
- L'ús de tecnologies com Bluetooth i Bluetooth Low Energy ofereixen una manera eficient de transmetre dades i comunicar-se amb els dispositius dels usuaris per a la presa de dades d'assistència.
- És important mantenir un registre precís de les dades d'assistència rebudes per a utilitzar-les en l'administració, el seguiment o les estadístiques.
- S'han assolit els objectius principals, però manca l'optimització de l'aplicació per requerir la menor intervenció de l'usuari possible, objectiu que necessita més investigació i temps de desenvolupament.

Actualment, l'aplicació és funcional, però encara requereix d'un disseny millorat i d'algunes optimitzacions en les funcionalitats principals.

AGRAÏMENTS

Vull expressar el meu profund agraïment al meu tutor d'aquest projecte, Joan Bartrina, per la seva inestimable assistència i orientació en cada moment. La seva contribució ha estat fonamental per al desenvolupament exitós d'aquest projecte, aportant una perspectiva experta i un suport constant. La seva dedicació i coneixement han estat una font d'inspiració i motivació durant tot el procés.

A més, desitjo expressar la meua gratitud al creador de la llibreria react-native-ble-advertiser, Victor Pamplona, per posar a la meua disposició les eines necessàries per dur a terme aquest projecte amb la tecnologia de BLE (Bluetooth Low Energy). La seva contribució en el desenvolupament d'aquesta llibreria ha estat essencial per permetre la implementació eficient de la funcionalitat de comunicació i recopilació de dades d'assistència.

Finalment, vull expressar el meu reconeixement a la meua estimada tieta, qui m'ha ajudat a formular de manera més acadèmica i precisa les idees que volia expressar en aquest document. La seva col·laboració i suport en el procés de redacció han estat inestimables i han contribuït a millorar la qualitat i la coherència de les meves expressions.

REFERENCES

- [1] The Bluetooth® Low Energy Primer (PDF v.1.1.0) — Bluetooth® Technology Website. (s.f.). Bluetooth® Technology Website. <https://www.bluetooth.com/bluetooth-resources/the-bluetooth-low-energy-primer/>
- [2] Off Line — Zorraquino. (n.d.). Zorraquino. <https://www.zorraquino.com/diccionario/internet/ques-off-line.html>
- [3] CAS - Central Authentication Service. (n.d.). Campus Virtual UAB. <https://cv.uab.cat/>
- [4] LA IMPORTANCIA DE LLEVAR UN CONTROL DE ASISTENCIA DE LOS COLABORADORES - VenturesSoft. (n.d.). VenturesSoft. <https://venturessoft.com/la-importancia-de-llevar-un-control-de-asistencia-de-los-colaboradores/>
- [5] QR Code Security: What are QR codes and are they safe to use? (n.d.). [www.kaspersky.com. https://www.kaspersky.com/resource-center/definitions/what-is-a-qr-code-how-to-scan](https://www.kaspersky.com/resource-center/definitions/what-is-a-qr-code-how-to-scan)
- [6] Hardware - Wikipedia, la enciclopedia libre. (2003, July 26). Wikipedia, la enciclopedia libre. <https://es.wikipedia.org/wiki/Hardware>
- [7] Time Tracking Software for Employees — QuickBooks Time. (n.d.). QuickBooks®. <https://quickbooks.intuit.com/time-tracking/>
- [8] AttendanceBot - Time tracking software. (n.d.). AttendanceBot - Attendance and Time Tracking. <https://www.attendancebot.com/time-tracking/>
- [9] Fernández, Y. (2019, August 23). API: qué es y para qué sirve. Xataka - Tecnología y gadgets, móviles, informática, electrónica. <https://www.xataka.com/basics/api-que-sirve>
- [10] Stakeholders: quiénes son, por qué son importantes y cómo gestionarlos. (n.d.). Thinking for Innovation. <https://www.iebschool.com/blog/stakeholders-quienes-son-digital-business/>
- [11] ¿Qué es el Feedback positivo y negativo? Errores a evitar - InboundCycle. (n.d.). Agencia de Inbound Marketing - InboundCycle. <https://www.inboundcycle.com/blog-de-inbound-marketing/feedback-positivo-y-negativo-9-errores-que-debes-evitar>
- [12] Scrum Sprints: Everything You Need to Know — Atlassian. (n.d.). Atlassian. <https://www.atlassian.com/agile/scrum/sprints>
- [13] Start a New React Project – React–Expo. (n.d.). React. <https://react.dev/learn/start-a-new-react-projectexpo>
- [14] Arimetrics - Que es Framework. (n.d.). Arimetrics. <https://www.arimetrics.com/glosario-digital/framework>

- [15] Introduction · React Native. (n.d.). React Native · Learn once, write anywhere. <https://reactnative.dev/docs/getting-started>
- [16] Penalva, J. (2011, January 25). NFC: qué es y para qué sirve en este 2023. Xataka - Tecnología y gadgets, móviles, informática, electrónica. <https://www.xataka.com/moviles/nfc-que-es-y-para-que-sirve>
- [17] Qué es una dirección IP: definición y explicación. (n.d.). latam.kaspersky.com. <https://latam.kaspersky.com/resource-center/definitions/what-is-an-ip-address>
- [18] Maximum number of BLE sensors that could be connected to a BLE gateway at a given instant? (s.f.). Stack Overflow. <https://stackoverflow.com/questions/39163772/maximum-number-of-ble-sensors-that-could-be-connected-to-a-ble-gateway-at-a-give/3917411539174115>
- [19] What is a UUID, and why should you care? (n.d.). Cockroach Labs. <https://www.cockroachlabs.com/blog/what-is-a-uuid/>
- [20] ¿Qué es iOS? — Gabit. (n.d.). gabit.org. <https://www.gabit.org/gabit/?q=es/que-es-ios>
- [21] ¿Qué es un SDK? (n.d.). Red Hat - We make open source technologies for the enterprise. <https://www.redhat.com/es/topics/cloud-native-apps/what-is-sdk>
- [22] Repositori react-native-ble-plx - Intent <https://github.com/dotintent/react-native-ble-plx>
- [23] react-native-ble-plx 2.0.2 — Documentation. (n.d.). <https://dotintent.github.io/react-native-ble-plx/introduction>
- [24] Repositori react-native-ble-advertiser - vitorpamplona <https://github.com/vitorpamplona/react-native-ble-advertiser>
- [25] AdvertiseData — Android Developers. (n.d.). Android Developers. <https://developer.android.com/reference/android/bluetooth/le/AdvertiseData>
- [26] AdvertiseSettings — Android Developers. (n.d.). Android Developers. <https://developer.android.com/reference/android/bluetooth/le/AdvertiseSettings>
- [27] SafeAreaView · React Native. (n.d.). React Native · Learn once, write anywhere. <https://reactnative.dev/docs/safeareaview>
- [28] Què és el NIU? - Departament de Biologia Cel·lular, de Fisiologia i d'Immunologia - UAB Barcelona. (n.d.). Universitat Autònoma de Barcelona - UAB Barcelona. <https://www.uab.cat/web/acollida-pdi-nova-incorporacio/que-es-el-niu-1345694076994.html>
- [29] Repositori react-native-modal-datetime-picker - mmazzarolo <https://github.com/mmazzarolo/react-native-modal-datetime-picker>
- [30] Repositori react-native-picker - React Native <https://github.com/react-native-picker/picker>
- [31] Introducing Hooks - React. (n.d.). React - A JavaScript library for building user interfaces. <https://legacy.reactjs.org/docs/hooks-intro.html>
- [32] JSON- Introduction. (n.d.). JSON. <https://www.json.org/json-es.html>
- [33] Contributors to Wikimedia projects. (2004, March 3). Comma-separated values - Wikipedia. Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Comma-separated_values
- [34] Excel- Microsoft 365. (n.d.). <https://www.microsoft.com/es-es/microsoft-365/excel>
- [35] Google Sheets: Online Spreadsheet Editor — Google Workspace. (n.d.). <https://www.google.com/.https://www.google.com/sheets/about/>

APÈNDIX

A.1 Diagrama de Casos d'us

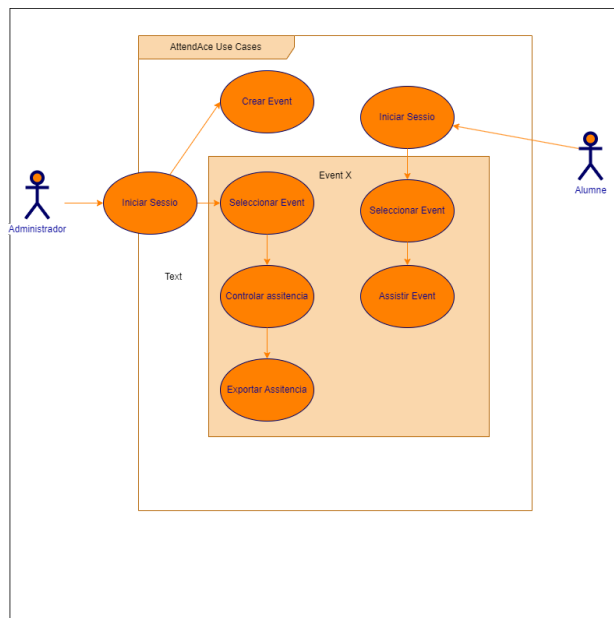


Fig. 9: Diagrama Casos d'us AttendAce