

TimeFace: Sistema de control de temps laboral

Albert Marín Silvero

Grau d'Enginyeria Informàtica

TFG - Desenv.aplicacions dispositius mòbils (Android)

David Escuer Latorre

Carles Garrigues Olivella

01/01/2021



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>TimeFace: Sistema de control de temps laboral</i> <i>Sistema de control de temps laboral</i>
Nom de l'autor:	<i>Albert Marín Silvero</i>
Nom del consultor/a:	<i>David Escuer Latorre</i> <i>Jordi Almirall López</i>
Nom del PRA:	<i>Carles Garrigues Olivella</i>
Data de lliurament (mm/aaaa):	<i>01/2021</i>
Titulació o programa:	<i>Grau d'Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>TFG - Desenv.aplicacions dispositius mòbils (Android)</i>
Idioma del treball:	<i>Català i anglès</i>
Paraules clau	<i>Android, temps laboral, reconeixement facial</i>
Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i>	
La principal finalitat d'aquest treball de final de grau és desenvolupar i obtenir una aplicació per dispositius mòbils <i>Android</i> que proporcioni als usuaris la possibilitat d'enregistrat entrades i sortides de l'entorn laboral fent ús de la identificació i reconeixement facial per identificar-se. Donat que des del passat 12 de Maig de 2019 va entrar en vigor un decret que obliga a les empreses a tenir un control efectiu del número d'hores que realitzen els seus empleats registrant l'hora d'entrada i l'hora de sortida de cada un d'ells. I prenent consciència de les circumstàncies actuals en les que ens trobem, en plena pandèmia Covid 19, on infinitat de persones s'han vist obligades a teletreballar desplaçant el seu entorn laboral a la seva llar, ens trobem en una situació propícia per implantar aquest tipus de sistemes aprofitant la tecnologia actual. Com a model de desenvolupament s'ha optat per: Android Studio i el llenguatge Kotlin com a entorn de desenvolupament utilitzant llibreries encapsulades en l'app com Firebase, MLKit o Tensorflow per la detecció i reconeixement facial, amb bases de dades NoSQL de Firebase i Scrum com a metodologia de treball.	

Abstract (in English, 250 words or less):

The main purpose of this end-of-degree work is to develop and get an application for mobile Android devices that provide users with the functionality of record entries and exits from the work environment in order to identify and recognize themselves.

Since 12 May 2019, a decree has been aproved to force companies to control the number of hours their employees work by recording the time of entry and exit . We are aware of the current situation which we living, in the midst of the Covid 19 pandemic, with many people forced to move their work environment to their home, we are in a good moment to implement this kind of system, adopting the new technologys.

As a development model we have chosen: Android Studio and Kotlin language as development environment using encapsulated libraries in the app with Firebase, MLKit or Tensorflow for the detection and facial recognition, with NoSQL data bases of Firebase and Scrum as work methodology.

Índex

1. Introducció	1
1.1 Context i justificació del Treball:	1
1.2 Objectius del Treball:	4
1.2.1 Objectius fora de l'abast del projecte:	4
1.3 Enfocament i mètode seguit:	5
1.3.1 Entorns de desenvolupament:	5
1.3.2 Sistemes de reconeixement facial:	5
1.3.3 Bases de dades:	5
1.3.4 Metodologies de desenvolupament:	6
1.4 Planificació del Treball:	7
1.4.1 Recursos:	7
1.4.2 Activitats a realitzar, fites i estimació en hores:	7
1.4.3 Diagrama de Gantt:	8
1.5 Breu sumari de productes obtinguts:	9
1.6 Breu descripció dels altres capítols de la memòria:	9
2. Disseny	10
2.1 Disseny Centrat en l'Usuari (DCU):	10
2.1.1 Usuaris i context d'ús:	10
2.1.1.1 Indagació:	10
2.1.1.2 Perfils d'usuari identificats:	14
2.1.2 Disseny conceptual:	16
2.1.2.1 Escenaris d'ús:	16
2.1.2.2 Fluxos d'interacció:	20
2.1.3 Prototipatge:	21
2.1.3.1 <i>Sketches</i> :	21
2.1.3.2 Prototipus horitzontal d'alta fidelitat:	22
2.1.3.3 Flux d'interacció amb prototipus:	25
2.1.4 Avaluació:	26
2.1.4.1 Qüestionari dels usuaris de test:	26
2.1.4.2 Tasques dels usuaris de test:	26
2.1.4.3 Qüestionari sobre les tasques:	27
2.2 Disseny Tècnic:	28
2.2.1 Definició dels casos d'ús:	28
2.2.3 Disseny de l'arquitectura:	31
3. Implementació	33
3.1 : Presa de decisions durant el desenvolupament	33
3.2 : Eines i entorn de desenvolupament	34
3.3 : Frameworks, APIs i llibreries de tercers	34
3.4 : Arquitectura i flux d'informació de TimeFace	36
3.5 : Esquema i estructura de la solució TimeFace	37
3.6 : Estat del projecte	38
4. Proves	39
4.1 : Proves funcionals	39
4.1.1 : Iniciar TimeFace:	39
4.1.2 : Inici jornada:	40
4.1.3 : Pausa jornada:	41
4.1.4 : Fi jornada:	42

4.1.5 : Alta d'usuari:	43
4.1.6 : Finalitzar <i>TimeFace</i> :	44
4.1.7 : Control d'errors:	45
4.2 : Proves unitàries	46
5. Conclusions	48
6. Glossari	49
7. Bibliografia	51
8. Annexos	54
8.1 : Guia d'usuari:	54
8.1.1: Com iniciar <i>TimeFace</i> ?	54
8.1.2: Com finalitzar <i>TimeFace</i> ?	55
8.1.3: Com registrar un nou usuari?	56
8.1.4: Com Iniciar jornada?	57
8.1.5: Com Pausar la jornada?	58
8.1.6: Com Finalitzar la jornada?	60
8.2 : Guia d'usuari avançat:	61
8.2.1: Com compilar <i>TimeFace</i> ?	61
8.2.2: Com <i>debugar</i> <i>TimeFace</i> ?	62
8.2.3: Com executar <i>TimeFace</i> ?	63
8.2.4: Com instal·lar <i>TimeFace</i> en un dispositiu <i>Android</i> ?	63

Llista de figures

Figura 1: Captures de Bixpe Control horario	2
Figura 2: Captures Intratime Control horario	2
Figura 3: Captures Reloj Laboral.....	3
Figura 4: Exemple d'entorn de desenvolupament Android Studio.....	6
Figura 5: Exemple d'entorn de testing virtual Android	6
Figura 6: Diagrama de Gantt de la planificació.....	8
Figura 7: Fluxos d'interacció de TimeFace	20
Figura 8: Sketches d'inici i captura facial.....	21
Figura 9: Sketches d'alta d'usuari i motiu de la pausa	21
Figura 10: Prototipus d'inici i captura facial	22
Figura 11: Prototipus d'alta d'usuari i motiu de la pausa	23
Figura 12: Prototipus Confirmació i error d'alta d'usuari.....	24
Figura 13: Flux d'interacció amb prototipus.....	25
Figura 14: Diagrama UML de base de dades.....	31
Figura 15: Diagrama UML de base de dades Android local	32
Figura 16: Objectes JSON del servei Azure Cognitive Services	32
Figura 17: Política de preus de Microsoft Azure Cognitive Services	35
Figura 18: Arquitectura i flux d'informació de TimeFace	36
Figura 19: Esquema i estructura de la solució TimeFace.....	37
Figura 20: Proves funcionals: Iniciar TimeFace.....	39
Figura 21: Proves funcionals: Inici jornada.....	40
Figura 22: Proves funcionals: Pausa jornada	41
Figura 23: Proves funcionals: Fi jornada	42
Figura 24: Proves funcionals: Alta d'usuari	43
Figura 25: Proves funcionals: Finalitzar TimeFace.....	44
Figura 26: Proves funcionals: Control d'errors	45
Figura 27: Proves unitàries: MSTest	46
Figura 28: Proves unitàries: Funcions implementades.....	46
Figura 29: Proves unitàries: Exemple de test unitari	46
Figura 30: Proves unitàries: Resultats de les proves unitàries	47

1. Introducció

1.1 Context i justificació del Treball:

Donat que des del passat 12 de Maig de 2019 va entrar en vigor un decret que obliga a les empreses a tenir un control efectiu del número d'hores que realitzen els seus empleats registrant l'hora d'entrada i l'hora de sortida de cada un d'ells. I prenent consciència de les circumstàncies actuals en les que ens trobem, en plena pandèmia *Covid 19*, on infinitat de persones s'han vist obligades a teletreballar desplaçant el seu entorn laboral a la seva llar.

La meua proposta com a treball de final de grau, orientat al desenvolupament d'aplicacions per a dispositius mòbils *Android*, anirà enfocat a desenvolupar una aplicació per aquests tipus de dispositius, que faci ús de la tecnologia de reconeixement facial per identificar a les persones i registrar les hores d'entrada i sortida de l'entorn laboral.

Actualment existeixen multitud de sistemes que serveixen per obtenir el mateix propòsit, que van des de la típica fulla de firmes a la entrada de l'empresa, compartició de fulles d'Excel, sistemes d'enregistrament de números de personal, sistemes de fixatge amb targetes de personal per banda magnètica o *contactless MIFARE*, sistemes de control d'accés biomètric, etc. També existeixen moltes aplicacions per dispositius mòbils per a realitzar la mateixa acció, però amb simples interaccions amb la pantalla, no recordo haver-ne vist cap que utilitzi el reconeixement facial, tot i que molt possiblement sí que existeixen.

Els principals avantatges que pot aportar una aplicació amb aquestes característiques en primer lloc es la mobilitat, ja que permet instal·lar dispositius com mòbils o *tablets* a les entrades dels llocs de treball. Trobem dispositius de molts fabricants i econòmicament raonables, també permet a aquelles persones que estan desplaçades (comercials, teletreballadors, etc.) que puguin tenir instal·lada l'aplicació al seu propi dispositiu. Com a segon punt fort podem identificar la tecnologia de reconeixement facial, crec que és una tecnologia suficientment madura i ràpida (en alguns establiments ja s'utilitza com a mètode de pagament), que permet reconèixer inequívocament a les persones que estan realitzant les accions, evita els problemes d'haver de recordar números, portar a sobre sistemes d'identificació personal i redueix la possibilitat de frau.

A partir d'ara anomenarem l'app de treball com *TimeFace*.

Comparativa amb aplicacions similars:

- [Bixpe Control horario](#): És una app multiplataforma per *Android*, *iOS* i *Windows*, fa ús de la geolocalització del dispositiu per enregistrar els events, també fa ús de la càmera del dispositiu per enregistrar fotografies de l'usuari, però no utilitza aquestes fotos com a sistema d'identificació, sinó que fa servir un PIN d'usuari.

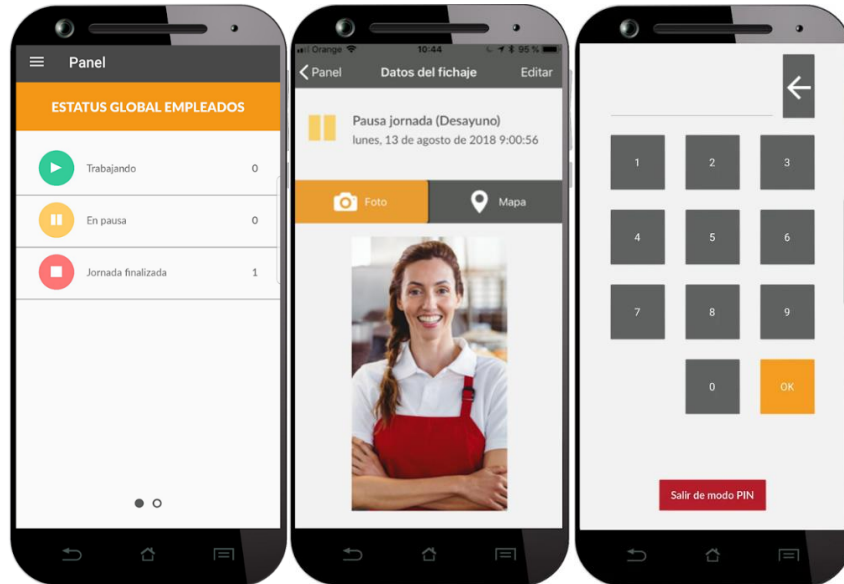


Figura 1: Captures de Bixpe Control horario

- [Intratype Control horario](#): És una app multiplataforma per *Android*, *iOS* i *Windows*, fa ús de la geolocalització del dispositiu per enregistrar els events, i utilitza usuari i contrasenya per identificar a l'usuari.

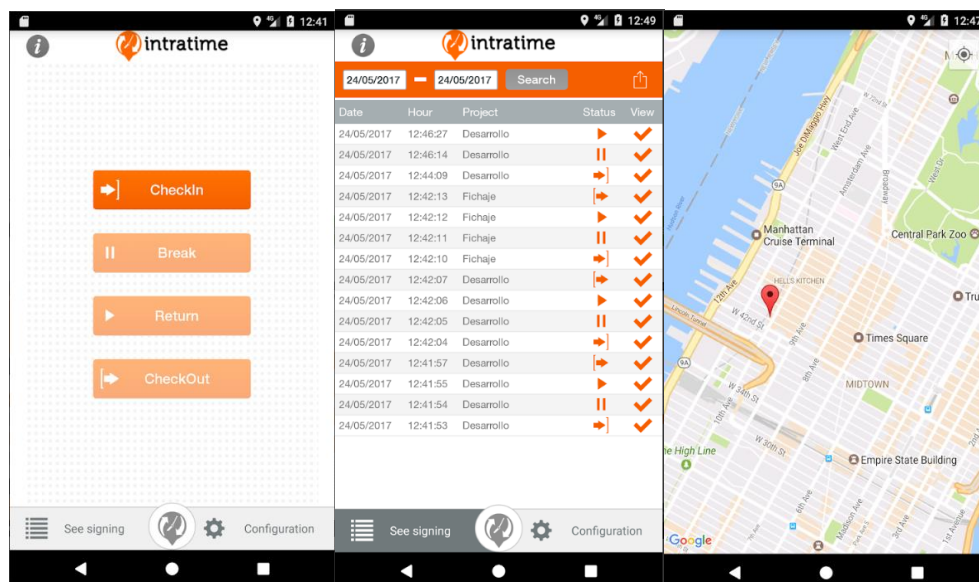


Figura 2: Captures Intratype Control horario

- Reloj Laboral: És una *app* multiplataforma per *Android*, *iOS* i *Windows*, fa ús de la geolocalització del dispositiu per enregistrar els events, i pot utilitzar els mètodes d'usuari i contrasenya per identificar a les persones o els sensors biomètrics del dispositiu.

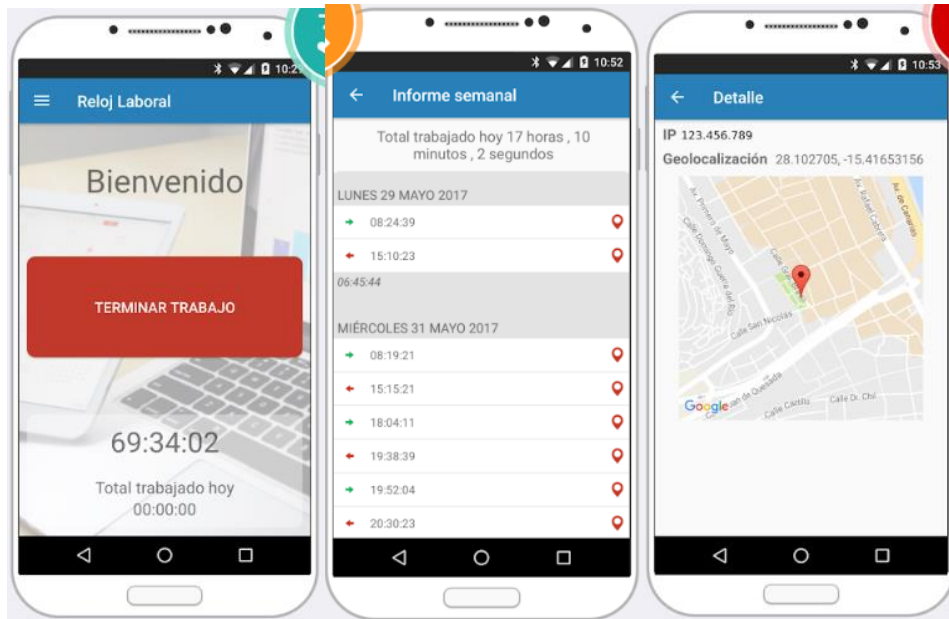


Figura 3: Captures Reloj Laboral

- *TimeFace*: Va adreçat a dispositius exclusivament *Android*, no utilitza la geolocalització del dispositiu ni sensors biomètrics per enregistrar els events, però la principal avantatge que aporta respecte la resta d'aplicacions és que utilitza la tecnologia de reconeixement facial per identificar l'empleat. Aquest fet és diferenciador de cara a l'usuari ja que no necessita recordar pins, contrasenyes o portar dispositius de reconeixement com targetes d'identificació, evita la suplantació de persones, etc.

1.2 Objectius del Treball:

El principal objectiu del treball és assolir els coneixements necessaris per obtenir una *app* nativa per a dispositius *Android* amb els següents requeriments funcionals:

- RF1: Operar amb la càmera del dispositiu *Android*.
- RF2: Detecció de cares capturades a partir de RF1.
- RF3: Identificació de les cares a partir d'una base de coneixement.
- RF4: Alta de nous usuaris *TimeFace*
- RF5: Registre d'entrada d'usuari
- RF6: Registre de sortida d'usuari
- RF7: Registre de pausa d'usuari
- RF8: Registre d'incidència d'usuari
- RF9: Captura de tots els events en BBDD

Requeriments no funcionals:

- RNF1: *TimeFace* ha de poder operar en telèfons o *tablets Android*.
- RNF2: Els dispositius requereixen de versions *Android 5.0 Lollipop* o superiors.
- RNF3: *TimeFace* ha de poder operar amb mode *online* o *offline*.
- RNF4: *TimeFace* ha de proporcionar una experiència d'usuari senzilla i simple.
- RNF5: *TimeFace* ha de complir totes les normes de RGPD.

1.2.1 Objectius fora de l'abast del projecte:

Per motius de temps, disponibilitat i recursos, *TimeFace* es limitarà a complir els requeriments funcionals i no funcionals descrits anteriorment. Queden fora de l'abast d'aquest projecte altres requeriments com per exemple tots aquells relacionats amb la gestió i anàlisi de les dades generades a partir de la captura d'events com ara notificacions, un *Backoffice* amb diferents *reportings*, *dashboards*, etc. Aquests requeriments seria necessari planificar-los i descriure'ls en versions més avançades del producte.

1.3 Enfocament i mètode seguit:

En relació a l'enfocament i el mètode a seguir, s'han valorat diferents possibilitats i estratègies que comentaré a continuació:

1.3.1 Entorns de desenvolupament:

- **Android Studio + Kotlin:** Es tracta de l'entorn de desenvolupament oficial de *Google* que integra tot l'SDK natiu per la plataforma *Android*. *Kotlin* és un llenguatge de tipat estàtic que corre a sobre de màquines virtuals de *Java* i també pot interoperar amb aquest llenguatge. També pot ser compilat com a codi font *Javascript*, elimina les referències nul·les i soluciona alguns defectes de *Java*. En l'actualitat *Kotlin* té el recolzament de *Google* i la comunitat de desenvolupadors i una sòlida reputació com a possible candidat per substituir a *Java* com a llenguatge per *Android*.

- **Microsoft Visual Studio + Xamarin:** És l'entorn de desenvolupament integrat de *Microsoft* per excel·lència per entorns *Windows* i compatible amb diversos llenguatges. Des de fa ja algunes versions pot incorporar *Xamarin* que permet junt amb el llenguatge *C#* desenvolupar aplicacions mòbils per *Android*, *iOS* i *Windows*.

1.3.2 Sistemes de reconeixement facial:

Existeixen llibreries i eines de desenvolupament que es poden instal·lar directament al dispositiu *Android* empaquetades junt amb l'app com per exemple: *Firebase*, *MLKit*, *Tensorflow*, etc.

Actualment existeixen multitud d'empreses tecnològiques que ofereixen el seu potencial de *cloud computing* com a servei web, i entre aquests serveis trobem eines de detecció i reconeixement facial. S'han valorat les següents:

- *Amazon Web Services (AWS)*
- *Microsoft Azure*
- *Google Cloud*

1.3.3 Bases de dades:

En relació a l'emmagatzematge de dades per sistemes *Android*, es contemplen dues possibilitats, les no relacionals com per exemple *Firebase* o les relacionals com *SQLite*.

1.3.4 Metodologies de desenvolupament:

En aquest cas optaré per metodologies *Scrum* ja que considero que és més flexible i s'adapta molt millor al canvi constant que pot patir un projecte d'aquestes característiques. A l'apartat 1.4 *Planificació del Treball* donaré més detalls sobre com aplicaré el conjunt de bones pràctiques.

Finalment i en aquesta fase del projecte optaré pel model: *Android Studio + Kotlin* com a entorn de desenvolupament utilitzant llibreries encapsulades en l'app com *Firebase*, *MLKit* o *Tensorflow* per la detecció i reconeixement facial, amb bases de dades *NoSQL* de *Firebase* i *Scrum* com a metodologia de treball i desenvolupament.

Personalment tinc més experiència amb entorns de desenvolupament de *Microsoft*, per aquest motiu he decidit moure'm de la meua zona de confort per encarar el repte amb tecnologia que desconec per intentar que l'experiència sigui el més enriquidora possible de cara al meu aprenentatge. També m'ha fet decantar-me per aquestes opcions el fet que la majoria de serveis de *cloud computing* tenen un pla d'entrada gratuït, però després passen a ser de pagament, aleshores he preferit explorar les llibreries *Firebase*, *MLKit* i *Tensorflow* ja que tenen versions *Open Source* i considero que aquestes llibreries són més adaptables per entorns de desenvolupament *Android* que no pas *Microsoft*.

Aquestes decisions es prenen en una fase molt inicial del projecte i a mida que avancin els esdeveniments es poden veure modificades, per aquest motiu durant la fase de planificació de treball inclouré fites per prendre aquest tipus de decisions per tal de que la tecnologia de desenvolupament sigui adaptable al cicle de vida del projecte i als requeriments funcionals.

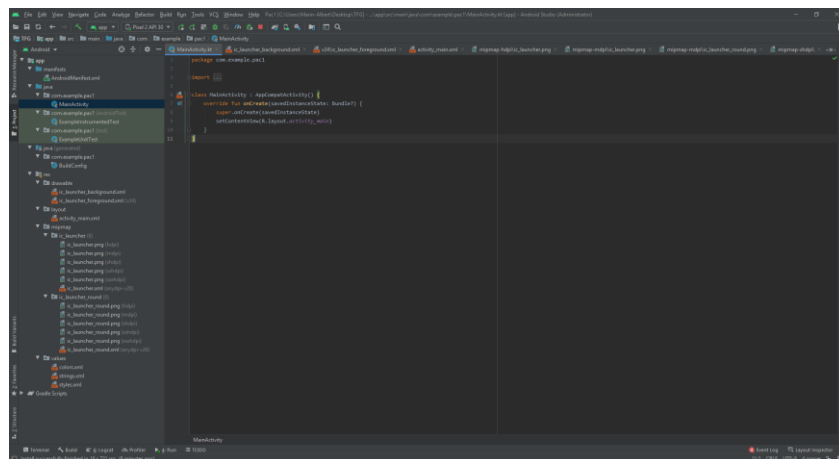


Figura 4: Exemple d'entorn de desenvolupament Android Studio

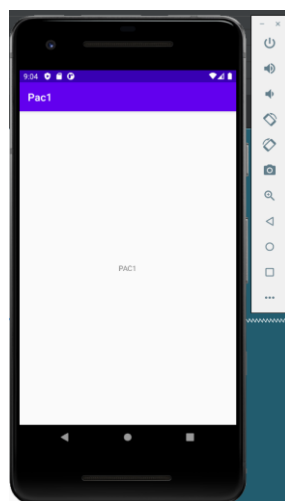


Figura 5: Exemple d'entorn de testing virtual Android

1.4 Planificació del Treball:

1.4.1 Recursos:

Per tal de descriure la planificació del projecte començaré parlant dels recursos, en aquest cas en concret es tracta d'un treball individual no col·laboratiu, aleshores l'únic recurs humà són els professors i consultors implicats i jo mateix (Albert Marín), però per aquest cas em centraré exclusivament en el recurs que puc controlar que sóc jo mateix.

L'estimació d'hores disponibles per al recurs Albert Marín es calcula de l'ordre de 6 hores entre setmana de dilluns a divendres i 6 hores els caps de setmana de dissabte a diumenge, el que fa un total de 12 hores setmanals. El total de la durada del projecte es preveu de 18 setmanes, el que faria un còmput total de **216 hores**.

Respecte a recursos materials, per l'entorn de desenvolupament i test es requereixen els següents:

- PC *Intel Core i5* amb 8 GB de memòria RAM i sistema operatiu *Windows 10*
- *Android Studio 4.0.1*
- Diferents dispositius *Android*

1.4.2 Activitats a realitzar, fites i estimació en hores:

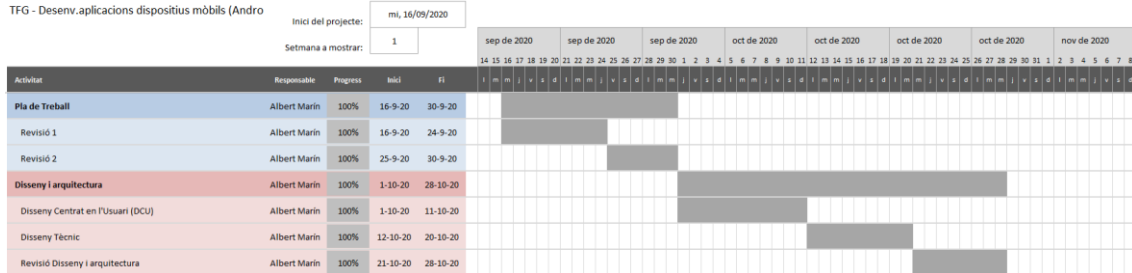
- Plà de Treball (15h)
 - o Obtenció del plà de treball del projecte.
- Disseny i arquitectura (50h)
 - o Disseny Centrat en l'Usuari (DCU) (35h)
 - o Disseny Tècnic (15h)
 - o Obtenció del disseny i arquitectura de l'app *Android* a implementar.
- Implementació (100h): Durant aquesta etapa del projecte, tot i no disposar d'un equip de treball col·laboratiu, intentaré aplicar la metodologia de desenvolupament de software *Scrum*, aplicant estratègies incrementals basades en la qualitat del resultat. Els *Sprints* es defineixen aproximadament de 2 setmanes (16h) amb fases de *review* i *retrospective* al final de cada *Sprint* per tal de valorar si el desenvolupament s'adequa al resultat esperat i canviant decisions de disseny i arquitectura si fos necessari.
 - o Sprint 1: Consolidació de l'entorn de desenvolupament, estructura principal de la solució, frameworks, APIs, llibreries de tercers i inici del disseny de la *UI*. (35h).
 - o Sprint 2: Consolidar la *UI* i implementació de funcionalitats i casos d'ús. (35h).
 - o Sprint 3: Finalitzar el desenvolupament i proves funcionals i unitàries. (35h).
 - o Obtenció de l'app *Android* en base als requeriments de disseny i arquitectura establerts.
- Lliurament Final (45h)
 - o Conclusions. (5h)
 - o Guia d'usuari. (5h)
 - o Glossari. (1h)
 - o Bibliografia. (2h)
 - o Consolidació de la memòria del treball de final de grau. (10h)
 - o Obtenció de la presentació del treball de final de grau. (15h)
 - o Elaboració d'un autoinforme de competències transversal. (8h)
- Defensa Virtual (6h)

1.4.3 Diagrama de Gantt:

Planificació: *TimeFace* - Sistema de control de temps laboral

Albert Marín Silvero

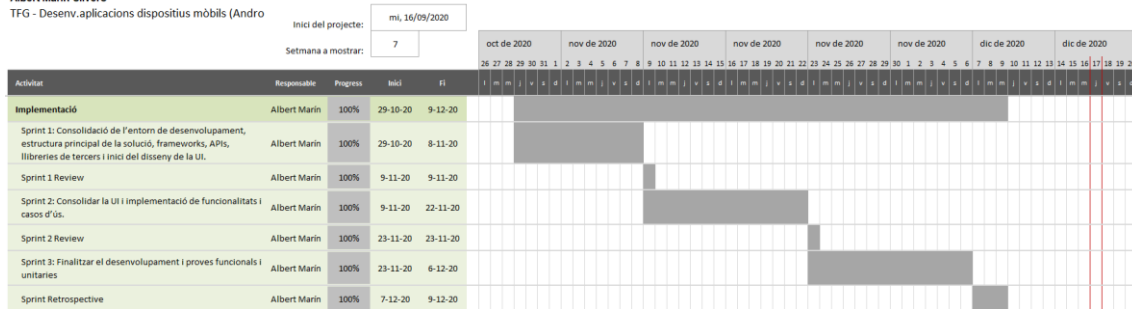
TFG - Desenv. aplicacions dispositius mòbils (Andro



Planificació: *TimeFace* - Sistema de control de temps laboral

Albert Marín Silvero

TFG - Desenv. aplicacions dispositius mòbils (Andro



Planificació: *TimeFace* - Sistema de control de temps laboral

Albert Marín Silvero

TFG - Desenv. aplicacions dispositius mòbils (Andro

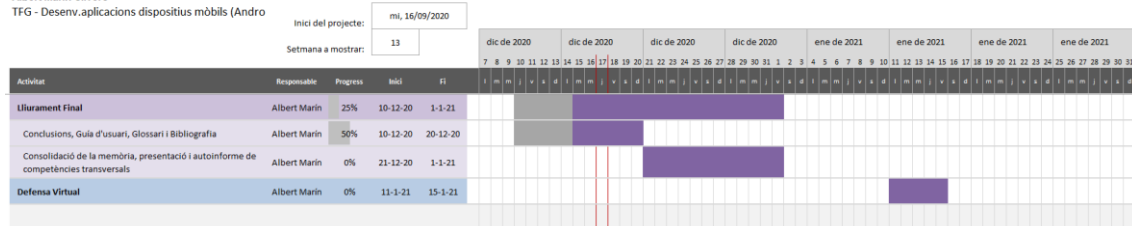


Figura 6: Diagrama de Gantt de la planificació

1.5 Breu sumari de productes obtinguts:

Els productes o entregables que s'obtindran d'aquest treball de final de grau són els següents:

- *App Android TimeFace*
- Memòria del treball de final de grau
- Presentació del treball de final de grau
- Projectes generats amb *Android Studio* i el codi font de l'*app*.

1.6 Breu descripció dels altres capítols de la memòria:

En aquest memòria del treball de final de grau s'inclouran posteriorment a mida que avanci el treball els següents capítols:

2. **Disseny:** Conté les especificacions, conceptes i requeriments per complir els objectius del *software* a implementar.
3. **Arquitectura:** Descriu l'estructura, funcionament i interaccions entre les diferents parts del *software*.
4. **Implementació:** Inclou tots els aspectes rellevants obtinguts durant l'etapa de desenvolupament de l'*app*.
5. **Presentació:** Descriu els aspectes més rellevants de la presentació realitzada per defensar el treball de final de grau.
6. **Conclusions:** Descripció i reflexió sobre els resultats obtinguts d'aquest treball.
7. **Glossari:** Definició dels termes i acrònims més rellevants utilitzats dins la Memòria.
8. **Bibliografia:** Llista numerada de les referències bibliogràfiques utilitzades dins la memòria.
9. **Annexos:** Llistat d'apartats que són massa extensos per incloure dins la memòria.

2. Disseny

En aquest capítol ens dedicarem a especificar els conceptes i requeriments per complir els objectius del *software* a implementar descrits durant la fase del Pla de treball.

2.1 Disseny Centrat en l'Usuari (DCU):

En aquest apartat situarem a l'usuari en el centre del procés de confecció de l'*app* per tal de satisfer les seves necessitats i objectius i identificar els diferents contextos d'ús.

2.1.1 Usuaris i context d'ús:

El principal objectiu d'aquesta fase és el de reconèixer les necessitats i objectius dels diferents perfils d'usuari de l'*app*.

Com ja hem esmentat en anteriors capítols, existeixen necessitats previsibles que per motius de temps i recursos queden fora de l'abast d'aquest projecte i que no analitzarem tampoc a la fase de disseny perquè seria necessari analitzar-los en versions més avançades del producte, com per exemple:

- *Backoffice* amb funcionalitats de perfil d'administrador de l'*app TimeFace*
- *Backoffice* amb funcionalitats de perfil de gestió i anàlisi de les dades, reportings, dashboards, etc.

2.1.1.1 Indagació:

Donada aquesta premissa, anem a esmentar els mètodes d'indagació triats per tal de reconèixer les diferents necessitats i perfils d'usuaris. Degut a que ja es preveu que la nostra població d'estudi seran exclusivament els usuaris finals de l'*app* i descartarem els usuaris amb perfil administrador o analític del sistema, els mètodes seleccionats han estat l'observació i investigació contextual (*Shadowing*) i entrevistes en profunditat, dinàmiques de grup i enquestes. Considero que són les que més s'adapten en centrar-se en l'usuari final i les seves necessitats d'un cas d'ús en particular dins d'una determinada empresa/corporació ja que els involucra directament.

Per tal d'aconseguir un model el més representatiu possible, s'han triat diferents perfils dins de la companyia que descriurem a continuació:

Empleat 1	
Nom:	Pilar
Edat:	41 anys
Sexe:	Dona
Formació:	Administrativa
Departament:	RRHH
Antiguitat:	12 anys
Entorn laboral:	Oficines Centrals
Altres observacions:	

Empleat 2	
Nom:	Daniel
Edat:	23 anys
Sexe:	Home
Formació:	CF Informàtica
Departament:	IT
Antiguitat:	Nova incorporació
Entorn laboral:	Oficines Centrals, esporàdicament a les instal·lacions del client final.
Altres observacions:	El Daniel, degut a les circumstàncies actuals de pandèmia, s'ha vist obligat a traslladar el seu entorn laboral a casa.

Empleat 3	
Nom:	Mario
Edat:	43 anys
Sexe:	Home
Formació:	Administració i direcció d'empresa
Departament:	Direcció
Antiguitat:	8 anys
Entorn laboral:	Oficines Centrals
Altres observacions:	El Mario, degut a les circumstàncies actuals de pandèmia, s'ha vist obligat a traslladar el seu entorn laboral a casa.

Empleat 4	
Nom:	Maite
Edat:	36 anys
Sexe:	Dona
Formació:	Administració i direcció d'empresa
Departament:	Comercial
Antiguitat:	3 anys
Entorn laboral:	Instal·lacions del client final i esporàdicament a les Oficines Centrals o a casa.
Altres observacions:	

Empleat 5	
Nom:	Santiago
Edat:	52 anys
Sexe:	Home
Formació:	Hosteleria
Departament:	Operacions (Cuiner)
Antiguitat:	16 anys
Entorn laboral:	Instal·lacions del client final.
Altres observacions:	

Observació i investigació contextual (*Shadowing*):

Aquest mètode ens permet obtenir dades qualitatives observant a un o més participants durant les seves activitats quotidianes, reconèixer les accions i patrons de comportament així com els seus hàbits o rutines. Per aplicar aquesta metodologia s'han triat persones amb diferents rols dins de l'empresa del cas d'estudi (sector de serveis d'alimentació i grans esdeveniments) als quals se'ls ha realitzat un seguiment i observació després de sol·licitar els permisos adequats per a realitzar aquestes tasques més intrusives amb la seva privacitat. Evidentment degut a la situació actual de distanciament social ha estat més complex del que hagués sigut en una situació habitual, ja que tan sols s'han pogut fer els seguiments a les oficines centrals de l'empresa i a les instal·lacions d'alguns clients. Per aquells empleats que han desplaçat el seu entorn laboral a les seves llars no ha estat possible i per aquest motiu ens recolzarem en les següents tècniques d'indagació per extreure conclusions.

Resultats:

1. El mètode actual per realitzar els controls del temps laboral de cada empleat es limita a una fulla de firmes on s'indica el dia, l'hora d'entrada i l'hora de sortida.
2. Aquest mètode l'utilitzen tant els empleats que treballen a les oficines centrals com aquells que estan desplaçats a les instal·lacions d'alguns clients.
3. No tots els empleats realitzen aquest control, actualment es limita a les entrades i sortides de les oficines centrals i de clients, però aquells empleats que treballen per exemple des de casa seva no realitzen cap tipus de control.
4. Tampoc s'enregistren les pauses de dinar o esmorzar, ni descansos, ni incidències, etc.

Entrevistes en profunditat, dinàmiques de grup i enquestes:

Donat que amb l'anterior mètode d'indagació no hem obtingut tota la informació necessària per extreure conclusions definitives, he optat per aplicar aquest segon mètode per complementar les observacions obtingudes fins ara. A la pràctica s'ha optat per realitzar una entrevista i una enquesta a cada un dels empleats de la mostra per reconèixer les necessitats i objectius dels diferents perfils d'usuari de l'*app*. S'han obviat alguns aspectes o preguntes de caire general ja que l'*app* està enfocada a un cas d'ús particular per a una única empresa:

Model d'entrevista:

- Quants anys tens?
- Sexe?
- Nivell d'estudis?
- Quant temps fa que treballes a la empresa?
- Quin càrrec o responsabilitat ocupes a dins de l'empresa?
- Quines són les teves funcions?
- A on acostumes a realitzar aquestes funcions?
- Quin sistema utilitzes actualment per realitzar el control d'hores laborals?
- Com es realitza aquest control de temps laboral?
- Quines millores proposes sobre l'actual sistema de control?
- Estàs familiaritzat amb l'ús de dispositius mòbils com *smartphones* o *tablets*?
- Tens algun d'aquests dispositius a nivell d'empresa o particular?

A part de les entrevistes als usuaris més representatius de cada perfil, s'ha decidit dissenyar un model d'enquesta que es pot fer extensible a un número més elevat d'empleats per tal d'obtenir resultats que cobreixin moltes necessitats i perfils d'usuari diferents:

Model d'enquesta:

Enquesta			
Edat:			
Sexe:	Home	Dona	
Formació:			
Departament:			
Funcions:			
Antiguitat:			
Entorn laboral habitual:	Oficina	Client	Teletreball
És possible que realitzis part del teu treball fora del teu entorn habitual?	Sí	No	NS/NC
Pots deixar el teu lloc de treball per realitzar descans, parades per esmorzar o dinar, assumptes personals, etc.?	Sí	No	NS/NC
Actualment realitzes algun control del teu temps laboral?	Sí	No	NS/NC
Estàs familiaritzat amb l'ús de dispositius mòbils com <i>smartphones</i> o <i>tablets</i> ?	Sí	No	NS/NC
Tens algun d'aquests dispositius a nivell d'empresa o particular?	Sí	No	NS/NC

Després de les entrevistes i d'analitzar les respostes de les enquestes podem extreure les següents conclusions o resultats:

1. El 70% dels empleats de l'empresa realitzen les seves funcions a les oficines centrals o a les instal·lacions dels clients. El mètode actual per realitzar els controls del temps laboral d'aquests empleats es limita a una fulla de firmes on s'indica el dia, l'hora d'entrada i l'hora de sortida.
2. Aproximadament el 30% restant del personal de l'empresa actualment es troba en règim de teletreball i aquests no realitzen cap mena de control de jornada laboral.
3. La majoria d'empleats pot deixar el seu lloc de treball per realitzar descans, parades per esmorzar, dinar o assumptes personals, però tampoc s'enregistren de cap manera aquets esdeveniments.
4. La majoria dels empleats tenen la possibilitat de realitzar el seu treball fora del seu entorn habitual.
5. La majoria dels consultats està familiaritzat amb l'ús de dispositius mòbils com *smartphones* o *tablets* i tenen accés a algun d'aquests, ja sigui a nivell d'empresa o a nivell personal.

2.1.1.2 Perfils d'usuari identificats:

Fent servir els resultats obtinguts amb els mètodes d'indagació identifiquem dos perfils genèrics d'usuaris:

- Perfil 1: Usuaris amb entorn laboral habitual fix
- Perfil 2: Usuaris amb entorn laboral variable o amb règim de teletreball

Esmentar que es preveu que la interfície d'usuari sigui la mateixa per els dos perfils, la única diferencia significativa que observem és que s'utilitzi amb diferents dispositius (*smartphones* o *tables*) i intentarem resoldre aquest dilema amb tècniques de *responsive design*. També comentar que qualsevol usuari de perfil 1 és un potencial usuari a comportar-se com un perfil 2, o a la inversa. A partir d'ara les següents fases del disseny es faran tenint en compte aquets dos perfils d'usuari.

Perfil 1: Usuaris amb entorn laboral habitual fix	
Característiques del perfil	
Edat:	Edat laboral
Sexe:	Indiferent
Formació:	Des de formació bàsica a formació superior
Departament:	Indiferent
Funcions:	Indiferent
Nivell tecnològic:	Ha d'estar familiaritzat amb l'ús de <i>smartphones</i> o <i>tablets</i> a un nivell molt bàsic.
Context d'ús	
Entorn laboral habitual:	A les oficines centrals de l'empresa o a les instal·lacions habituals d'un client
On farà ús de l'app?	La farà servir a la entrada de la seva oficina o a l'entorn laboral de les instal·lacions del client (cuina, cafeteria, oficina, etc.)
Quan farà ús de l'app?	Cada vegada que iniciï o finalitzi la seva jornada laboral, o cada vegada que realitzi una pausa de la jornada per esmorzar, dinar, assumptes personals, etc.
En quin entorn farà ús de l'app?	El supòsit en aquest cas és que a cada entrada/sortida dels entorns laborals, aquets perfils d'usuari trobin un o més dispositius fixats a un mobiliari d'oficina adequat per a que puguin realitzar les seves tasques necessàries de la manera més òptima i àgil possible.
Anàlisi de tasques	
- Registre d'inici de jornada laboral. - Registre de fi de jornada laboral. - Registre de pausa de jornada laboral. - Alta de nous usuaris (en el cas de no existir a la base de dades de coneixement del sistema)	
Llistat de característiques	
- <i>TimeFace</i> per aquest context d'ús sempre s'estarà executant en primer pla i amb la càmera sempre en mode obert per enregistrar esdeveniments en aquests dispositius dedicats exclusivament per aquesta funció. - La identificació de l'usuari serà automàtica amb reconeixement facial. - En el cas d'haver d'enregistrar un alta de nou usuari, <i>TimeFace</i> requerirà diferents captures facials de l'usuari i una sèrie de dades personals com Nom, Cognoms, Dni, etc. per verificar que efectivament aquella persona pertany a l'empresa. - Aquest perfil d'usuari, ocasionalment és possible que realitzi la seva jornada de treball fora del seu entorn laboral habitual, en aquest cas podrà fer ús de <i>TimeFace</i> de la mateixa manera que el Perfil 2.	

Perfil 2: Usuaris amb entorn laboral variable o amb règim de teletreball	
Característiques del perfil	
Edat:	Edat laboral
Sexe:	Indiferent
Formació:	Des de formació bàsica a formació superior
Departament:	Indiferent
Funcions:	Indiferent
Nivell tecnològic:	Ha d'estar familiaritzat amb l'ús de <i>smartphones</i> o <i>tablets</i> a un nivell molt bàsic i ha de disposar d'un dispositiu corporatiu o personal.
Context d'ús	
Entorn laboral habitual:	Teletreball o entorn laboral variable.
On farà ús de l'app?	A diferents llocs, des de casa seva, la oficina temporal, instal·lacions d'un client, etc.
Quan farà ús de l'app?	Cada vegada que iniciï o finalitzi la seva jornada laboral, o cada vegada que realitzi una pausa de la jornada per esmorzar, dinar, assumptes personals, etc.
En quin entorn farà ús de l'app?	Donat que l'usuari executarà <i>TimeFace</i> des d'un dispositiu mòbil, pot fer ús d'aquesta en qualsevol entorn.
Anàlisi de tasques	
- Iniciar <i>TimeFace</i> - Registre d'inici de jornada laboral. - Registre de fi de jornada laboral. - Registre de pausa de jornada laboral. - Alta de nous usuaris (en el cas de no existir a la base de dades de coneixement del sistema) - Tancar <i>TimeFace</i>	
Llistat de característiques	
- La identificació de l'usuari serà automàtica amb reconeixement facial. - En el cas de tenir que enregistrar un alta de nou usuari, <i>TimeFace</i> requerirà diferents captures facials de l'usuari i una sèrie de dades personals com Nom, Cognoms, Dni, etc. per verificar que efectivament aquella persona pertany a l'empresa. - Aquest perfil d'usuari, ocasionalment és possible que realitzi la seva jornada de treball en un entorn de treball fix, en aquest cas podrà fer ús de <i>TimeFace</i> de la mateixa manera que el Perfil 1.	

2.1.2 Disseny conceptual:

Amb aquesta fase, el que pretenem assolir és que amb la informació recopilada fins ara, elaborarem els escenaris d'ús dels diferents perfils d'usuaris, com utilitzaran *TimeFace* en un context concret.

2.1.2.1 Escenaris d'ús:

Ara descriurem els diferents escenaris d'ús de com persones de diferents perfils interactuen amb *TimeFace*.

Escenari d'ús 1	
Perfil d'usuari que intervé	
Nom:	Pilar
Edat:	41 anys
Sexe:	Dona
Formació:	Administrativa
Departament:	RRHH
Antiguitat:	12 anys
Funcions:	Tècnica de nòmines
Nivell tecnològic:	Està familiaritzada amb l'ús de <i>smartphones</i> o <i>tablets</i> a nivell d'usuari <i>Standard</i> , els utilitza amb regularitat per enviar i rebre trucades, missatgeria, xarxes socials i altres <i>apps</i> .
En quin context es troba	
La Pilar com cada dia es desplaça de casa a la feina, ella treballa cada dia en horari d'oficina (de 9:00 a 18:00) a les oficines centrals de l'empresa on treballa.	
Quins són els seus objectius	
Quan entra per la porta de l'oficina, el que vol es poder enregistrar els inicis, pauses i finals de la seva jornada laboral verificant que el procés ha sigut correcte, amb la major agilitat possible i sense haver de portar a sobre cap dispositiu d'identificació.	
Tasques que duu a terme	
- Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Les seves necessitats d'informació	
Principalment verificar que els esdeveniments (inici, fi i pausa) de jornada s'enregistren correctament amb una data i hora concreta.	
A posteriori també tenir accés a <i>reports</i> on visualitzar tots els successos (<i>aquesta necessitat queda fora de l'àmbit del projecte perquè es faria des d'un BackOffice web</i>).	
Funcionalitats que necessita	
- Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament veient el dia i la hora. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Com desenvolupa aquestes tasques	
Quan la Pilar entra per la porta de la seva oficina, es troba amb un o varis dispositius <i>Android</i> instal·lats en un mobiliari dissenyat per estar fixat adequadament. <i>TimeFace</i> està constantment en execució i amb la captura de vídeo oberta, en el moment que ella executa alguna de les funcions (Inici, fi o pausa), el dispositiu dispara una fotografia, reconeix la cara i identifica a la persona enregistrant l'acció seleccionada. En aquest moment <i>TimeFace</i> mostra un missatge a l'àrea de notifikacions informant de l'acció que s'acaba d'enregistrar. El mateix succeeix quan a l'hora de dinar va al menjador on torna a trobar aquests dispositius <i>Android</i> amb els que interactua per enregistrar la pausa de la seva jornada laboral. Finalment quan surt de l'oficina, enregistra la fi de la jornada amb els mateixos dispositius que ho ha fet al matí.	

Escenari d'ús 2	
Perfil d'usuari que intervé	
Nom:	Daniel
Edat:	23 anys
Sexe:	Home
Formació:	CF Informàtica
Departament:	IT
Antiguitat:	Nova incorporació
Funcions:	Programador al departament d'aplicacions Informàtiques.
Nivell tecnològic:	Està familiaritzat amb l'ús de <i>smartphones</i> o <i>tablets</i> a nivell d'usuari avançat, els utilitza amb regularitat per enviar i rebre trucades, missatgeria, xarxes socials i altres <i>apps</i> , però també coneix les opcions de configuració dels seus dispositius i com treballen internament.
En quin context es troba	
En Daniel normalment cada dia es desplaça de casa a la feina, treballa en horari d'oficina (de 9:00 a 18:00) a les oficines centrals de l'empresa on treballa, però ocasionalment teletreballa des de casa seva.	
Quins són els seus objectius	
Quan entra per la porta de l'oficina, el que vol es poder enregistrar els inicis, pauses i finals de la seva jornada laboral verificant que el procés ha sigut correcte, amb la major agilitat possible i sense haver de portar a sobre cap dispositiu d'identificació. El mateix succeeix quan està a casa, el que vol es notificar quan comença i acaba la seva jornada laboral de forma àgil i fiable sense dependre dels sistemes de la oficina central i amb una interfície d'usuari que li resulti familiar.	
Tasques que duu a terme	
- Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral. - El seu primer dia de feina, necessitarà donar d'alta el seu usuari amb una o varies captures amb la càmera del dispositiu <i>Android</i> perquè posteriorment <i>TimeFace</i> el reconegui.	
Les seves necessitats d'informació	
Principalment verificar que els esdeveniments (inici, fi i pausa) de jornada s'enregistren correctament amb una data i hora concreta. A posteriori també tenir accés a <i>reports</i> on visualitzar tots els successos (<i>aquesta necessitat queda fora de l'àmbit del projecte perquè es faria des d'un BackOffice web o des d'un apartat de reporting dins de la mateixa app TimeFace</i>).	
Funcionalitats que necessita	
- Iniciar <i>TimeFace</i> en el seu dispositiu <i>Android</i>. - Donar d'alta el seu usuari a <i>TimeFace</i>. - Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament veient el dia i la hora. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Com desenvolupa aquestes tasques	
Quan el Daniel entra per la porta de la seva oficina, es troba amb un o varis dispositius <i>Android</i> instal·lats en un mobiliari dissenyat per estar fixat adequadament. <i>TimeFace</i> està constantment en execució i amb la captura de vídeo oberta, en el moment que ell executa alguna de les funcions (Inici, fi o pausa), el dispositiu dispara una fotografia, reconeix la cara i identifica a la persona enregistrent l'acció seleccionada. En aquest moment <i>TimeFace</i> mostra un missatge a l'àrea de notifikacions informant de l'acció que s'acaba d'enregistrar. El mateix succeeix quan a l'hora de dinar va al menjador on torna a trobar aquests dispositius <i>Android</i> amb els que interacciona per enregistrar la pausa de la seva jornada laboral. Finalment quan surt de l'oficina, enregistra la fi de la jornada amb els mateixos dispositius que ho ha fet al matí. Ocasionalment quan treballa des de casa, té instal·lat <i>TimeFace</i> al seu dispositiu personal, quan s'asseu a treballar, executa la <i>app</i> que arrenca amb una interfície d'usuari equiparable a la que troba als dispositius de la seva oficina i des de la que pot realitzar exactament les mateixes funcionalitats.	

Escenari d'ús 3	
Perfil d'usuari que intervé	
Nom:	Maite
Edat:	36 anys
Sexe:	Dona
Formació:	Administració i direcció d'empresa
Departament:	Comercial
Antiguitat:	3 anys
Funcions:	Comercial del sector dels serveis d'alimentació.
Nivell tecnològic:	Està familiaritzada amb l'ús de <i>smartphones</i> o <i>tablets</i> a nivell d'usuari <i>Standard</i> , els utilitza amb regularitat per enviar i rebre trucades, missatgeria, xarxes socials i altres <i>apps</i> .
En quin context es troba	
La Maite cada dia es desplaça de casa a visitar clients, ella acostuma cada dia a treballar en horari d'oficina (de 9:00 a 18:00), però el seu horari és flexible en funció dels requeriments dels clients.	
Quins són els seus objectius	
Quan comença la seva jornada laboral arribant a les instal·lacions dels clients, el que vol es poder enregistrar els inicis, pauses i finals de la seva jornada verificant que el procés ha sigut correcte, amb la major agilitat possible i sense haver de portar a sobre cap dispositiu d'identificació.	
Tasques que duu a terme	
- Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Les seves necessitats d'informació	
Principalment verificar que els esdeveniments (inici, fi i pausa) de jornada s'enregistren correctament amb una data i hora concreta. A posteriori també tenir accés a <i>reports</i> on visualitzar tots els successos (<i>aquesta necessitat queda fora de l'àmbit del projecte perquè es faria des d'un BackOffice web o des d'un apartat de reporting dins de la mateixa app TimeFace</i>).	
Funcionalitats que necessita	
- Iniciar <i>TimeFace</i> en el seu dispositiu <i>Android</i>. - Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament veient el dia i la hora. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Com desenvolupa aquestes tasques	
Quan la Maite comença la seva jornada laboral, executa <i>TimeFace</i> que té instal·lat al seu dispositiu <i>Android</i> d'empresa. La aplicació arrenca amb la captura de vídeo oberta, en el moment que ella executa alguna de les funcions (Inici, fi o pausa), el dispositiu dispara una fotografia, reconeix la cara i identifica a la persona enregistrant l'acció seleccionada. En aquest moment <i>TimeFace</i> mostra un missatge a l'àrea de notifikacions informant de l'acció que s'acaba d'enregistrar.	

Escenari d'ús 4	
Perfil d'usuari que intervé	
Nom:	Santiago
Edat:	52 anys
Sexe:	Home
Formació:	Hostaleria
Departament:	Operacions
Antiguitat:	16 anys
Funcions:	Cap de cuina
Nivell tecnològic:	Està familiaritzat amb l'ús de <i>smartphones</i> o <i>tablets</i> a nivell bàsic, els utilitza amb regularitat per enviar i rebre trucades i missatgeria.
En quin context es troba	
En Santiago cada dia es desplaça de casa a la feina, ell treballa a les instal·lacions d'alimentació d'un dels clients de l'empresa de 10:00 a 17:00 però el seu horari és flexible, en funció dels requeriments dels clients.	
Quins són els seus objectius	
Quan comença la seva jornada laboral arribant a les instal·lacions dels clients, el que vol es poder enregistrar els inicis, pauses i finals de la seva jornada verificant que el procés ha sigut correcte, amb la major agilitat possible i sense haver de portar a sobre cap dispositiu d'identificació.	
Tasques que duu a terme	
- Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Les seves necessitats d'informació	
Principalment verificar que els esdeveniments (inici, fi i pausa) de jornada s'enregistren correctament amb una data i hora concreta. A posteriori també tenir accés a <i>reports</i> on visualitzar tots els successos (<i>aquesta necessitat queda fora de l'àmbit del projecte perquè es faria des d'un BackOffice web o des d'un apartat de reporting dins de la mateixa app TimeFace</i>).	
Funcionalitats que necessita	
- Iniciar <i>TimeFace</i> en el seu dispositiu <i>Android</i>. - Enregistrar l'inici de la seva jornada laboral. - Verificar que els esdeveniments s'han enregistrat correctament veient el dia i la hora. - Enregistrar pauses de la seva jornada laboral per esmorzar, dinar, etc. - Enregistrar la fi de la seva jornada laboral.	
Com desenvolupa aquestes tasques	
Quan el Santiago arriba a les instal·lacions del seu client, es troba amb un dispositiu <i>Android</i> que ha instal·lat la seva empresa demanant permís al client, en un mobiliari dissenyat per estar fixat adequadament. <i>TimeFace</i> està constantment en execució i amb la captura de vídeo oberta, en el moment que ell executa alguna de les funcions (Inici, fi o pausa), el dispositiu dispara una fotografia, reconeix la cara i identifica a la persona enregistrent l'acció seleccionada. En aquest moment <i>TimeFace</i> mostra un missatge a l'àrea de notifikacions informant de l'acció que s'acaba d'enregistrar. Finalment quan surt de l'oficina, enregistra la fi de la jornada amb els mateixos dispositius que ho ha fet al matí. Ocasionalment el Santiago treballa a les instal·lacions d'altres clients que no tenen dispositius <i>TimeFace</i> fixes dedicats. En aquestes ocasions ell té instal·lat <i>TimeFace</i> al seu dispositiu <i>Android</i> d'empresa, quan comença a treballar, executa la <i>app</i> que arrenca amb una interfície d'usuari equiparable a la que troba al seu entorn habitual de feina des de la que pot realitzar exactament les mateixes funcionalitats.	

2.1.2.2 Fluxos d'interacció:

En aquest capítol mostrarem de manera gràfica l'estructura general de l'aplicació.

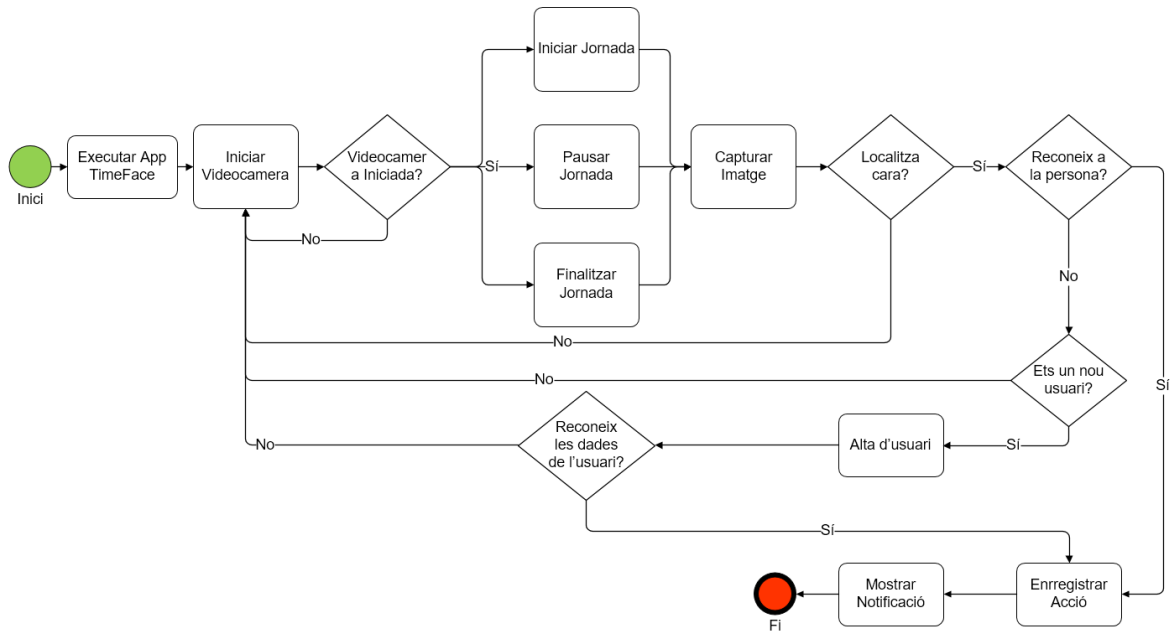


Figura 7: Fluxos d'interacció de TimeFace

2.1.3 Prototipatge:

En aquesta fase del projecte realitzarem una representació de l'aplicació que permeti comunicar decisions de disseny i avaluar-lo prenent com a punt de partida els fluxos d'interacció definits al capítol anterior.

2.1.3.1 Sketches:

Representen els primers esborralls de la interfície d'usuari que volem assolir.

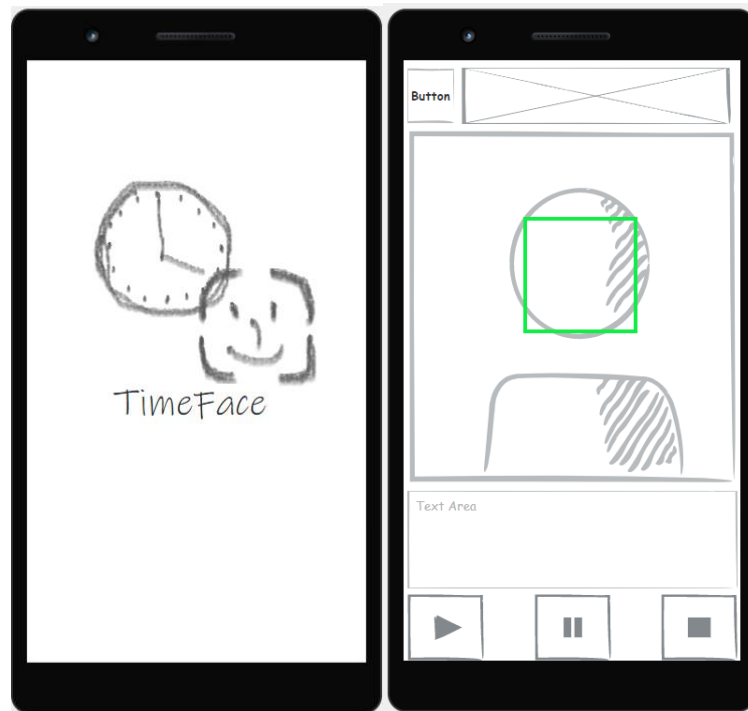


Figura 8: Sketches d'inici i captura facial



Figura 9: Sketches d'alta d'usuari i motiu de la pausa

2.1.3.2 Prototipus horitzontal d'alta fidelitat:

Amb aquest prototipus intentem representar el més aproximat possible l'aspecte final de *TimeFace* i que s'intentarà respectar durant les etapes del desenvolupament.

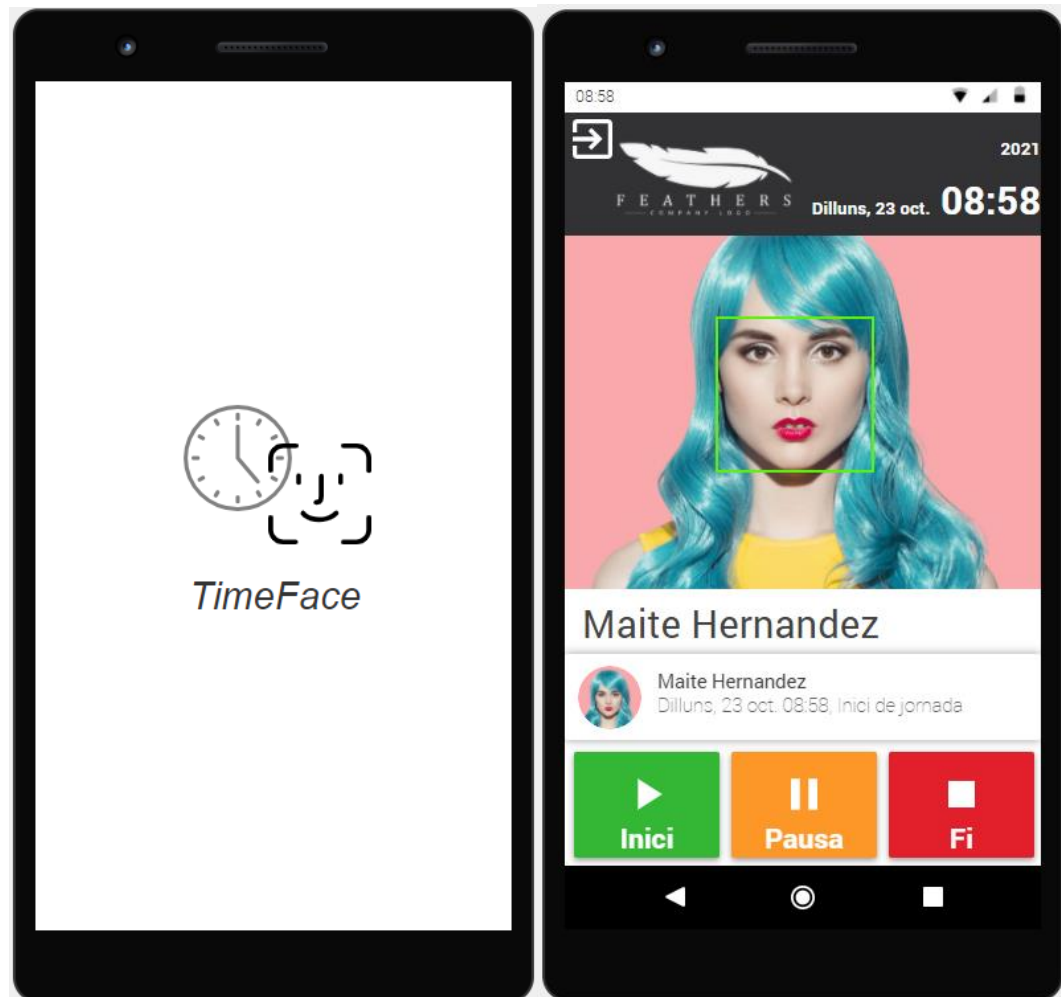


Figura 10: Prototipus d'inici i captura facial

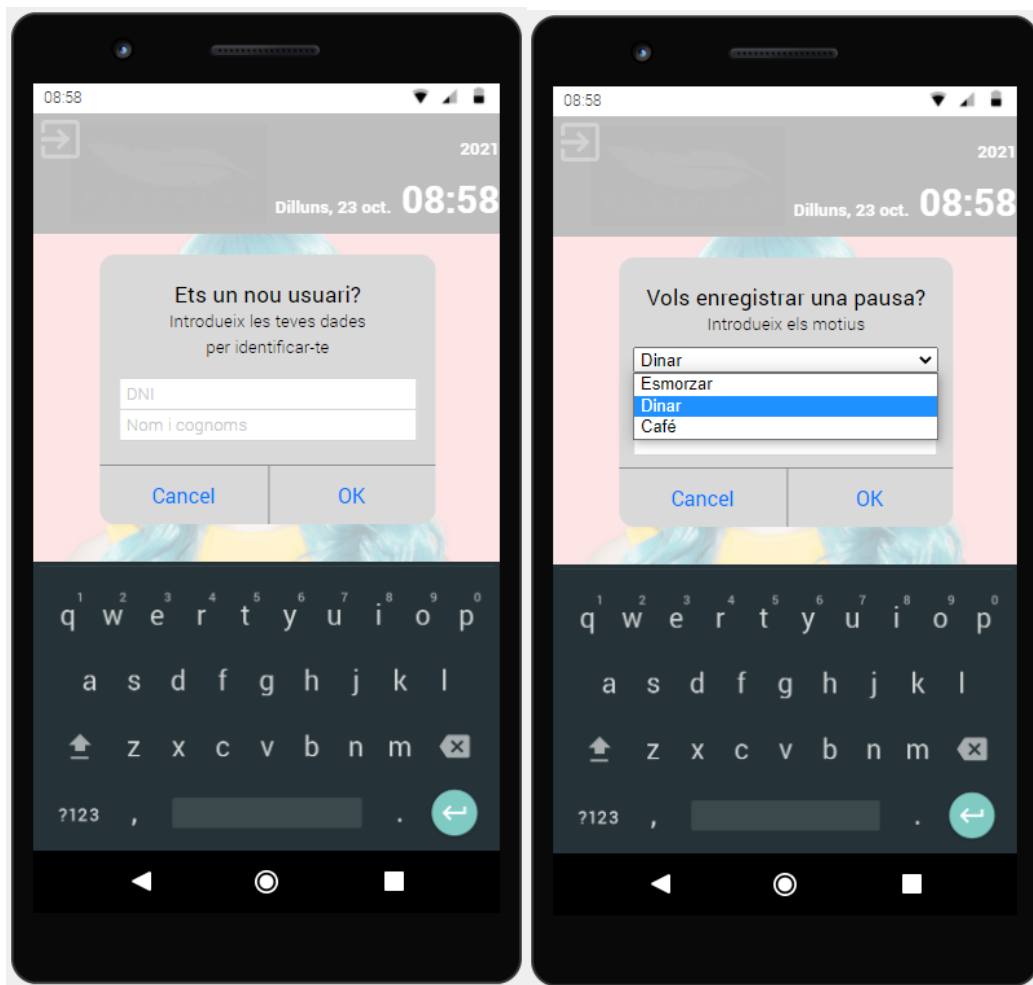


Figura 11: Prototipus d'alta d'usuari i motiu de la pausa

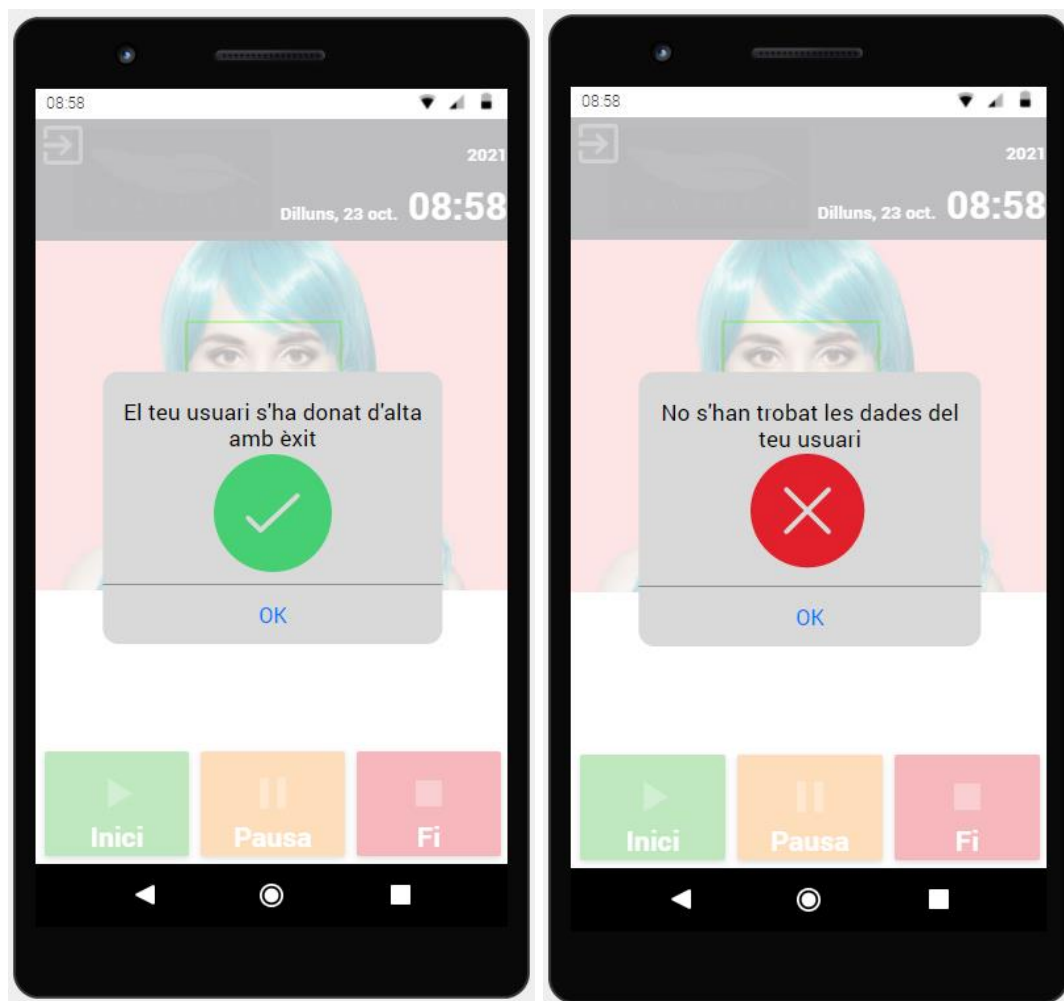


Figura 12: Prototipus Confirmació i error d'alta d'usuari

2.1.3.3 Flux d'interacció amb prototipus:

Amb aquesta representació s'intenta descriure el flux d'interacció de l'aplicació i amb quines interfícies d'usuari en format de prototip interactuaran en cada moment.

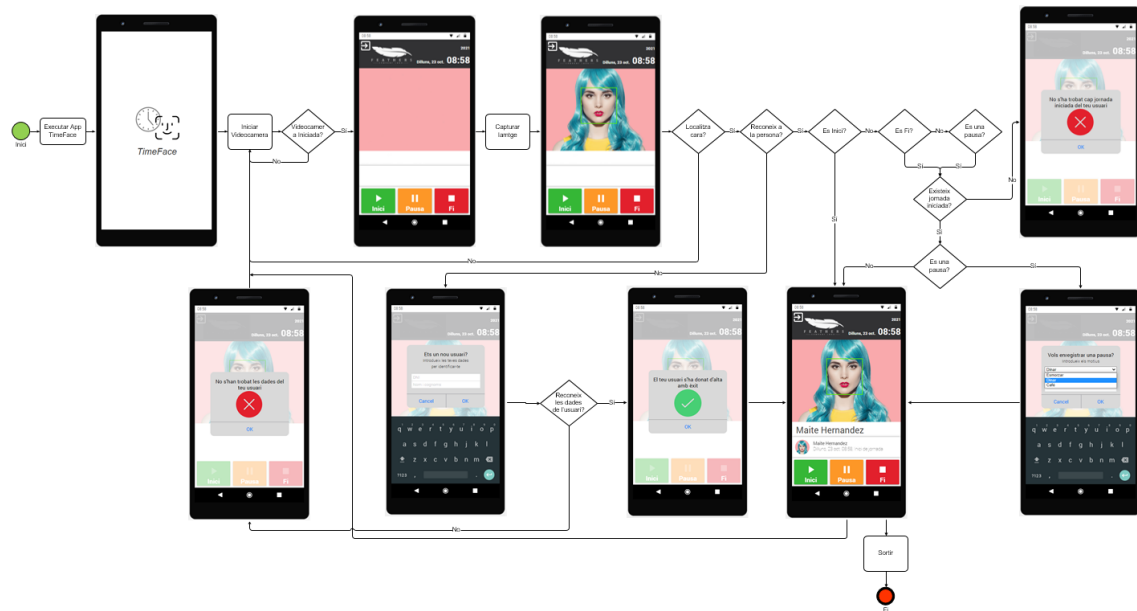


Figura 13: Flux d'interacció amb prototipus

2.1.4 Avaluació:

El propòsit d'aquest capítol és planificar l'avaluació dels prototipus de les fases anteriors per tal de detectar possibles millores o debilitats. Per aconseguir aquest objectiu optarem per les tècniques que descriurem a continuació.

2.1.4.1 Qüestionari dels usuaris de test:

Qüestionari usuari de test prototipus			
Edat:			
Sexe:	Home	Dona	
Formació:			
Departament:			
Funcions:			
Antiguitat:			
Entorn laboral habitual:	Oficina	Client	Teletreball
És possible que realitzis part del teu treball fora del teu entorn habitual?	Sí	No	NS/NC
Pots deixar el teu lloc de treball per realitzar descans, parades per esmorzar o dinar, assumptes personals, etc.?	Sí	No	NS/NC
Actualment realitzes algun control del teu temps laboral?	Sí	No	NS/NC
Estàs familiaritzat amb l'ús de dispositius mòbils com <i>smartphones</i> o <i>tablets</i> ?	Sí	No	NS/NC

2.1.4.2 Tasques dels usuaris de test:

En aquest apartat descriurem les tasques que han d'intentar realitzar els usuaris de test partint del prototipus d'alta fidelitat:

- Buscar *TimeFace* al dispositiu *Android*.
- Iniciar *TimeFace*.
- Enfocar la seva cara amb la captura d'imatge de la càmera del dispositiu.
- Realitzar un alta d'usuari.
- Enregistrar l'acció d'Inici de jornada.
- Enregistrar l'acció de Pausa de jornada.
- Enregistrar l'acció de Fi de jornada.
- Justificar una pausa.
- Finalitzar *TimeFace*.

2.1.4.3 Qüestionari sobre les tasques:

Per finalitzar la fase d'avaluació del prototipus, realitzarem el següent qüestionari a tots els usuaris de test per intentar obtenir el seu *feedback*.

Qüestionari sobre les tasques			
Has pogut iniciar <i>TimeFace</i> amb facilitat?	Sí	No	NS/NC
Has pogut realitzar totes les tasques amb facilitat?	Sí	No	NS/NC
T'ha resultat útil el mecanisme d'alta d'usuari	Sí	No	NS/NC
Et resulta útil i atractiu el sistema de reconeixement facial per identificar a l'usuari?	Sí	No	NS/NC
Has pogut tancar <i>TimeFace</i> amb facilitat?	Sí	No	NS/NC
Has trobat intuïtiu l'ús de <i>TimeFace</i> ?	Sí	No	NS/NC
Creus que els elements de la aplicació són fàcilment reconeixibles i interpretables?	Sí	No	NS/NC
Et sembla una aplicació adequada per controlar les jornades laborals?	Sí	No	NS/NC
Faries servir <i>TimeFace</i> durant el teu dia a dia per enregistrar les teves jornades?	Sí	No	NS/NC
Recomanaries <i>TimeFace</i> a altres persones?	Sí	No	NS/NC
Afegeix qualsevol comentari, suggeriment o recomanació addicional:			

2.2 Disseny Tècnic:

2.2.1 Definició dels casos d'ús:

A partir dels escenaris d'ús i el prototipus definits a la fase de disseny centrat en l'usuari, descriurem els casos d'ús per tal d'establir les funcionalitats de l'aplicació:

Identificador	CU-001
Nom	Iniciar <i>TimeFace</i>
Prioritat	Alta
Descripció	Permet iniciar l'app amb la finalitat de donar accés a la resta de funcionalitats de <i>TimeFace</i>
Actors	Qualsevol usuari
Pre-Condicions	El dispositiu <i>Android</i> ha de estar iniciat i operatiu
Iniciat per	Clic de l'Icona de <i>TimeFace</i>
Flux	1. Iniciar dispositiu <i>Android</i> 2. Clicar sobre de l'icona de <i>TimeFace</i>
Post-Condicions	1. S'inicia l'app <i>TimeFace</i> 2. S'inicia la captura de vídeo a través de la càmera frontal del dispositiu <i>Android</i>
Notes	

Identificador	CU-002
Nom	Iniciar jornada
Prioritat	Alta
Descripció	Permet a l'usuari enregistrar l'acció d'inici de jornada laboral
Actors	Qualsevol usuari
Pre-Condicions	1. <i>TimeFace</i> iniciat 2. Camara frontal del dispositiu <i>Android</i> iniciada
Iniciat per	Clic del botó "Iniciar jornada"
Flux	1. Capturar imatge amb la càmera 2. Localitzar cara a la captura 3. Identificar usuari de la captura 4. Enregistrar acció d'inici de jornada de l'usuari a la base de dades 5. Mostrar notificació a l'usuari
Post-Condicions	1. Que existeixi una cara a la captura de la càmera 2. Que la cara sigui identificada amb un dels usuaris 3. Que no existeixi una jornada de l'usuari iniciada 4. S'enregistra l'acció a la base de dades 5. <i>TimeFace</i> torna a l'estat per capturar imatges amb la càmera
Notes	

Identificador	CU-003
Nom	Pausar jornada
Prioritat	Alta
Descripció	Permet a l'usuari enregistrar l'acció de pausa de jornada laboral
Actors	Qualsevol usuari
Pre-Condicions	1. <i>TimeFace</i> iniciat 2. Camara frontal del dispositiu <i>Android</i> iniciada
Iniciat per	Clic del botó "Pausar jornada"
Flux	1. Capturar imatge amb la càmera 2. Localitzar cara a la captura 3. Identificar usuari de la captura
Post-Condicions	1. Que existeixi una cara a la captura de la càmera 2. Que la cara sigui identificada amb un dels usuaris 3. Que sí existeixi una jornada de l'usuari iniciada 4. S'inicia el cas d'ús: CU-005
Notes	

Identificador	CU-004
Nom	Finalitzar jornada
Prioritat	Alta
Descripció	Permet a l'usuari enregistrar l'acció de finalitzar la jornada laboral
Actors	Qualsevol usuari
Pre-Condicions	1. <i>TimeFace</i> iniciat 2. Camara frontal del dispositiu <i>Android</i> iniciada
Iniciat per	Clic del botó "Finalitzar jornada"
Flux	1. Capturar imatge amb la càmera 2. Localitzar cara a la captura 3. Identificar usuari de la captura 4. Enregistrar acció de fi de jornada de l'usuari a la base de dades 5. Mostrar notificació a l'usuari
Post-Condicions	1. Que existeixi una cara a la captura de la càmera 2. Que la cara sigui identificada amb un dels usuaris 3. Que sí existeixi una jornada de l'usuari iniciada 4. S'enregistra l'acció a la base de dades 5. <i>TimeFace</i> torna a l'estat per capturar imatges amb la càmera
Notes	

Identificador	CU-005
Nom	Justificar pausa de jornada
Prioritat	Alta
Descripció	Permet a l'usuari especificar el motiu de la pausa de jornada laboral
Actors	Qualsevol usuari
Pre-Condicions	1. Que l'usuari hagi realitzat l'acció de pausar la jornada laboral
Iniciat per	CU-003
Flux	1. Capturar els motius de la pausa 2. Enregistrar acció de pausa de jornada de l'usuari a la base de dades 3. Mostrar notificació a l'usuari
Post-Condicions	1. S'enregistra l'acció a la base de dades 2. <i>TimeFace</i> torna a l'estat per capturar imatges amb la càmera
Notes	

Identificador	CU-006
Nom	Alta d'usuari
Prioritat	Alta
Descripció	Permet a l'usuari donar d'alta un usuari no reconegut per el sistema de reconeixement facial de <i>TimeFace</i>
Actors	Qualsevol usuari
Pre-Condicions	1. Que l'usuari hagi realitzat l'acció d'iniciar la jornada laboral 2. Que existeixi una cara a la captura de la càmera 3. Que la cara no sigui identificada amb un dels usuaris
Iniciat per	CU-002
Flux	1. Capturar les dades de l'usuari 2. Enregistrar les dades del nou usuari a la base de dades de coneixement del sistema de reconeixement facial de <i>TimeFace</i> 3. Enregistrar acció d'inici de jornada del nou usuari a la base de dades 4. Mostrar notificació a l'usuari
Post-Condicions	1. S'enregistra l'acció a la base de dades 2. <i>TimeFace</i> torna a l'estat per capturar imatges amb la càmera
Notes	

Identificador	CU-007
Nom	Finalitzar <i>TimeFace</i>
Prioritat	Baixa
Descripció	Permet a l'usuari tancar l'aplicació <i>TimeFace</i>
Actors	Qualsevol usuari
Pre-Condicions	1. <i>TimeFace</i> iniciat
Iniciat per	Clic del botó "Sortir"
Flux	1. Finalitzar <i>TimeFace</i>
Post-Condicions	1. <i>TimeFace</i> es tanca adequadament
Notes	

2.2.3 Disseny de l'arquitectura:

En aquest apartat definirem l'arquitectura de *TimeFace* a nivell d'entitats, classes i bases de dades. En relació a l'emmagatzematge de dades per sistemes *Android*, es contemplen dues possibilitats, les no relacionals com per exemple *Firebase* o les relacionals com *SQLite*.

Cal esmentar que aquestes decisions es prenen en una fase molt inicial del projecte i a mida que avancin els esdeveniments es poden veure modificades, però l'enfocament inicial és treballar amb bases de dades *NoSQL* de *Firebase*.

També destacar que es tracta d'un projecte en mode experimental en molts sentits, especialment centrat en la tecnologia de reconeixement facial que es veurà recolzada per llibreries com *Firebase*, *MLKit* o *Tensorflow* i encara em generen molts dubtes en aquesta fase del desenvolupament com per tenir una visió clara del disseny de l'arquitectura final.

Inicialment la proposta per el model de dades seria el següent:

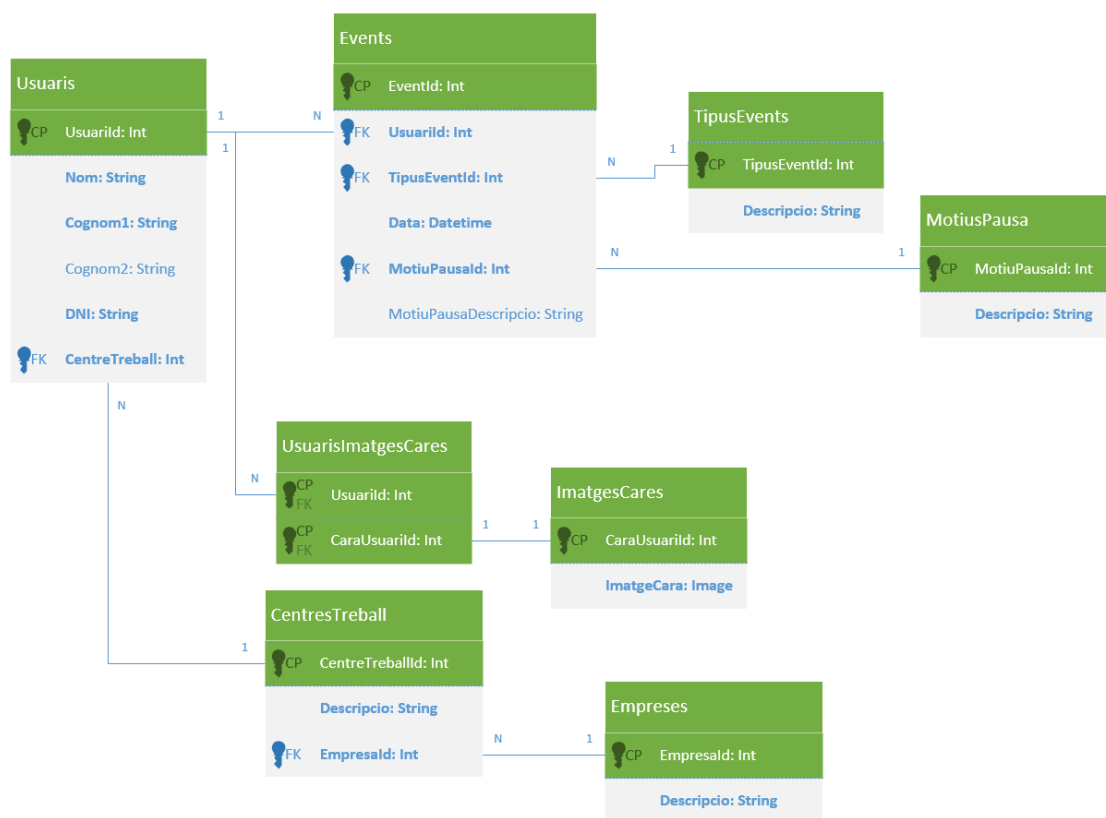


Figura 14: Diagrama UML de base de dades

Finalment, tal i com hem descrit a la secció d'Implementació i degut als problemes experimentats a la primera fase d'aquesta, es va optar per un canvi de model d'arquitectura i l'estructura de dades passa a estar en dos àmbits:

1. Base de dades relacional local *SQLite* al dispositiu *Android* per emmagatzemar les accions que realitza l'usuari i els events, fent servir el següent esquema:

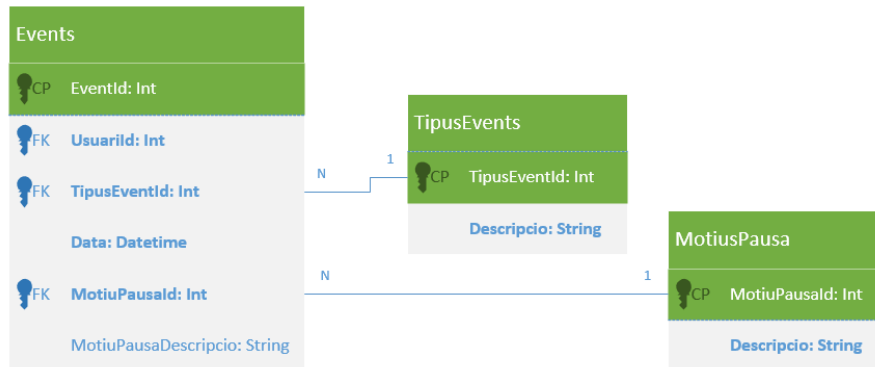


Figura 15: Diagrama UML de base de dades Android local

2. Una estructura de dades no relacional que es troba en el *cloud* d'*Azure Cognitive Services* que recuperem en forma d'objectes *JSON* a través de la *API* que proporciona aquest servei. Essencialment gestionem i emmagatzemem les dades relacionades amb els usuaris, les cares d'aquests i els seus atributs de la forma que es mostra a continuació:

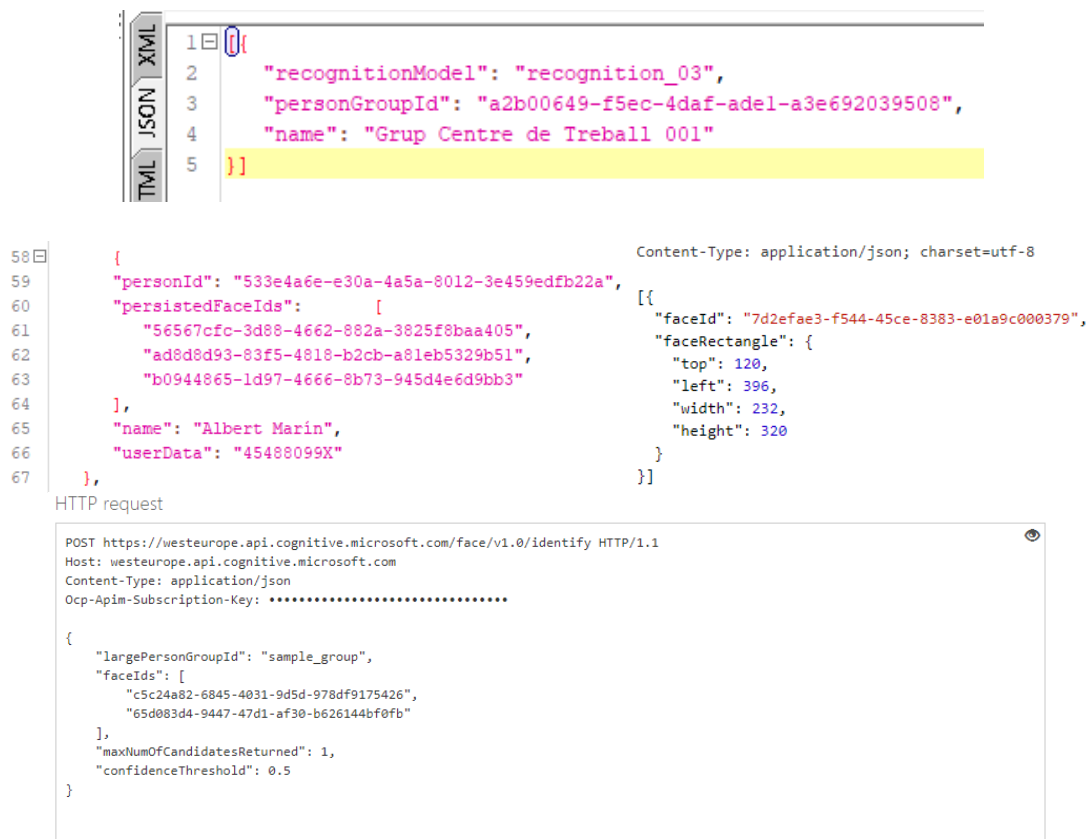


Figura 16: Objectes JSON del servei Azure Cognitive Services

3. Implementació

3.1 : Presa de decisions durant el desenvolupament

Tal i com vaig esmentar a les fases inicials del projecte 1.3.4 *Metodologies de desenvolupament*, l'enfocament inicial de la implementació es va plantejar sobre el model: *Android Studio + Kotlin* com a entorn de desenvolupament utilitzant llibreries encapsulades en l'app com *Firebase*, *MLKit* o *Tensorflow* per la detecció i reconeixement facial, amb bases de dades *NoSQL* de *Firebase*, però en el primer *Sprint Review* em vaig veure obligat a canviar d'estratègia pels següents motius que enumeraré a continuació donat que els resultats no s'adequaven als temps de dedicació:

1. Desconeixement de l'entorn de desenvolupament *Android Studio* i corba d'aprenentatge lenta degut al temps disponible de dedicació.
2. Desconeixement dels llenguatges de desenvolupament *Java/Kotlin*.
3. Amb les llibreries *Firebase* i *MLKit* es va assolir la fita de reconèixer els rostres d'una imatge, però no vaig ser capaç ni vaig trobar documentació per tal d'identificar els rostres a partir d'una font de coneixement. Aquest fet era un clar limitant per assolir els principals objectius del projecte i va ser el principal detonant que va fer que decidís canviar l'enfocament inicial.
4. No vaig preveure ni vaig saber dimensionar que aquestes eines de desenvolupament, principalment *Android Studio* i les màquines virtuals *Android*, requereixen d'un nivell alt de recursos de *hardware* dels que no disposava actualment, ja que el desenvolupament s'ha fet amb un ordinador amb processador *Intel Core i5-6300U* a 2.4GHz de primera generació amb 8 GB de memòria i sistema operatiu *W10 64 bits*. Aleshores em vaig trobar amb un coll d'ampolla on els temps no eren assumibles a l'hora d'iniciar les eines, compilar, *debugar*, *deploy*, fer córrer la solució amb emuladors *Android* locals, etc.

A partir d'aquesta situació em vaig trobar en un punt de no retorn i no volia córrer els riscos d'arribar al final de la fase d'implementació sense haver assolit les principals funcionalitats de l'app. Personalment un dels objectius d'aquest treball de fi de grau era enriquir els meus coneixements amb eines que desconec, i en part crec que l'he aconseguit, però finalment vaig prendre la decisió d'optar per un enfocament més conservador, amb eines de desenvolupament amb les que estic més familiaritzat, però que em permetessin aconseguir uns resultats satisfactoris. Aleshores he optat pel model de *Visual Studio 2019* com a entorn de desenvolupament integrat, amb *Xamarin* i *C#* com a llenguatges de programació i utilitzant *Cognitive Services* de *Azure* i *SQLite* perquè em permeten fer les operacions de *machine learning* que necessitava amb certa facilitat i de les que donaré més detalls en els següents capítols de la implementació.

3.2 : Eines i entorn de desenvolupament

Com hem explicat en capítols anteriors, per la implementació de *TimeFace* s'han utilitzat les següents eines de desenvolupament:

- **Microsoft Visual Studio Community 2019:** L'entorn de desenvolupament integrat de Microsoft per excel·lència, compatible amb diferents llenguatges com *C#*, *Visual Basic*, *Java*, *Python*, etc. Que permet desenvolupar aplicacions d'escriptori, webs, aplicacions per dispositius mòbils, etc.
- **Xamarin:** És un capa de desenvolupament que junt amb el llenguatge *C#* permet escriure aplicacions per dispositius mòbils *Android*, *iOS* i *Windows*.
- **C#:** Llenguatge de programació desenvolupat i estandaritzat per *Microsoft* com a part de la plataforma *.NET* basat en *C* i *C++*.
- **Azure Cognitive Services:** Servei de computació en *cloud* de l'empresa *Microsoft* que proporciona diferents funcionalitats en línia de *IA* i *Machine Learning*.
- **SQLite:** Sistema de gestió de bases de dades relacionals pensat per ser integrat amb aplicacions relativament petites i amb entorns locals.

3.3 : Frameworks, APIs i llibreries de tercers

- **NETStandard.Library**
 - o Versió: 2.0.3
 - o Autor: *Microsoft*
 - o Descripció: Formen un conjunt d'APIs que donen suport i usabilitat a totes les funcions del *framework .NET* estàndard.
- **Xamarin.Forms**
 - o Versió: 4.8.0.1451
 - o Autor: *Microsoft*
 - o Descripció: Formen un conjunt d'interfaces natives per *iOS*, *Android*, *UWP*, etc. a partir d'una única base de codi *C#* compartit.
- **Xamarin.Essentials:**
 - o Versió: 1.5.3.2
 - o Autor: *Microsoft*
 - o Descripció: Un *kit d'APIs* essencial per *apps* desenvolupades amb *Xamarin*.
- **Xam.Plugin.Media:**
 - o Versió: 5.0.1
 - o Autor: *James Montemagno*
 - o Descripció: És una *API* que proporciona a les *apps cross platform* el control sobre els recursos de la càmera de fotos i vídeo dels dispositius de les diferents plataformes.
- **Microsoft.Azure.CognitiveServices.Vision.Face:**
 - o Versió: 2.6.0-preview.1
 - o Autor: *Microsoft*
 - o Descripció: És una biblioteca que proporciona accés a les funcions de la *API* de reconeixement facial de *Microsoft Azure Cognitive Services*.
- **sqlite-net-pcl:**
 - o Versió: 1.7.335
 - o Autor: *SQLite-net*
 - o Descripció: És una biblioteca lleugera de codi obert que proporciona un sistema de bases de dades relacionals *SQL* incrustat dins de la mateixa *app*.

- **Plugin.Toast:**
 - o Versió: 2.2.0
 - o Autor: *Ishrak El Andaloussi*
 - o Descripció: *Plugin* per mostrar notificacions animades i personalitzades per dispositius mòbils *Android, iOS i Windows*.

Cal remarcar que les eines i *Apis* de *Microsoft Azure* actualment són gratuïtes durant el primer any d'avaluació, però a partir de llavors passen a ser de pagament per ús o serveis SaaS. Però a uns preus bastant assumibles per número de transaccions, a continuació adjunto una imatge de la política de preus actual de *Microsoft*:

Seleccione Oferta:

Face

Región:

Oeste de Europa

Divisa:

Euro (€)

Microsoft Face API utiliza los novedosos algoritmos de reconocimiento facial en la nube para detectar y reconocer caras humanas en imágenes. Esta funcionalidad incluye características como la detección y comprobación de caras y la agrupación de caras para organizarlas en grupos por su similitud visual.

Detalles de precios

INSTANCIA	TRANSACCIONES POR SEGUNDO (TPS) *	CARACTERÍSTICAS	PRECIO
Gratis - Web/Container	20 transacciones por minuto	Detección de caras Comprobación de caras Identificación de caras Agrupación de caras Búsqueda de caras parecidas	30,000 transacciones gratis por mes
Estándar - Web/Container	10 TPS	Detección de caras Comprobación de caras Identificación de caras Agrupación de caras Búsqueda de caras parecidas	De 0 a 1 millón de transacciones - €0,844 por 1000 transacciones De 1 a 5 millones de transacciones - €0,675 por 1000 transacciones De 5 a 100 millones de transacciones - €0,506 por 1000 transacciones Más de 100 millones de transacciones - €0,338 por 1000 transacciones

Figura 17: Política de preus de Microsoft Azure Cognitive Services

3.4 : Arquitectura i flux d'informació de *TimeFace*

Finalment i per simplificar el desenvolupament, s'ha triat un sistema d'intercanvi i emmagatzemament de dades híbrid. *TimeFace* utilitza la base de dades local *SQLite* per enregistrar els events i accions que realitzen els usuaris, però les dades dels usuaris, com l'identificador, el nom, DNI i valors facials, s'enregistren a l'estructura de dades d'*Azure Cognitive Services*.

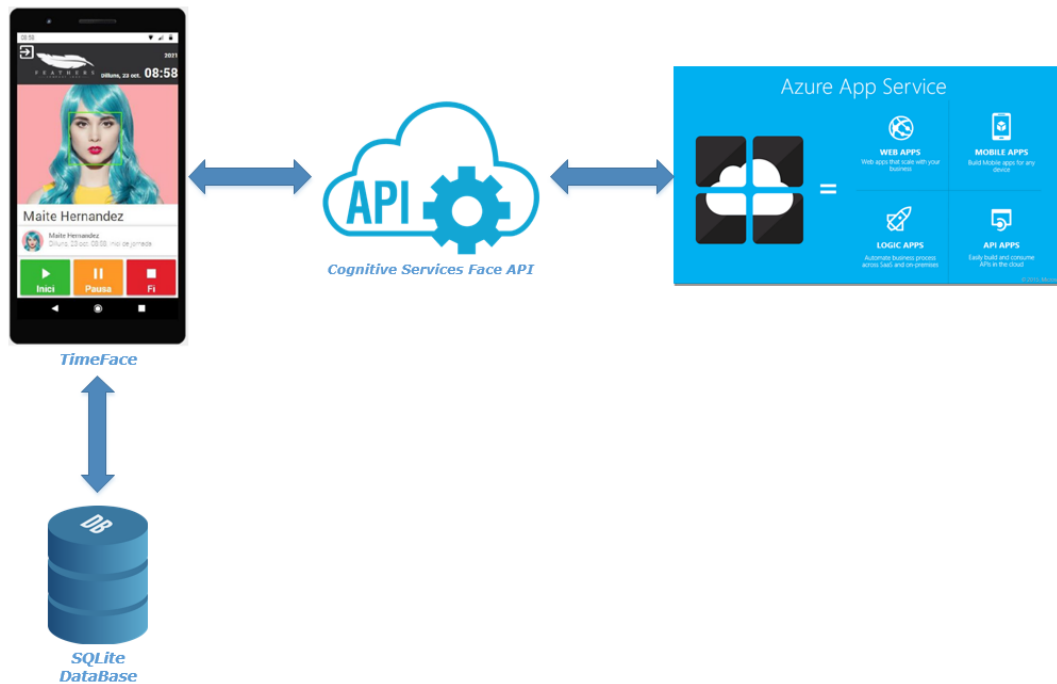


Figura 18: Arquitectura i flux d'informació de *TimeFace*

3.5 : Esquema i estructura de la solució *TimeFace*

Durant el desenvolupament s'ha mantingut el patró d'arquitectura de *software* *MVVM* (*Model-view-viewmodel*) per facilitar o mantenir el màxim possible separades la capa gràfica de presentació de l'usuari (*view*) de la lògica de negoci o capa de dades *back-end* (*model*) utilitzant la capa (*viewmodel*) com a intermediària entre els dos nivells d'abstracció, obtenint el següent esquema de la solució:

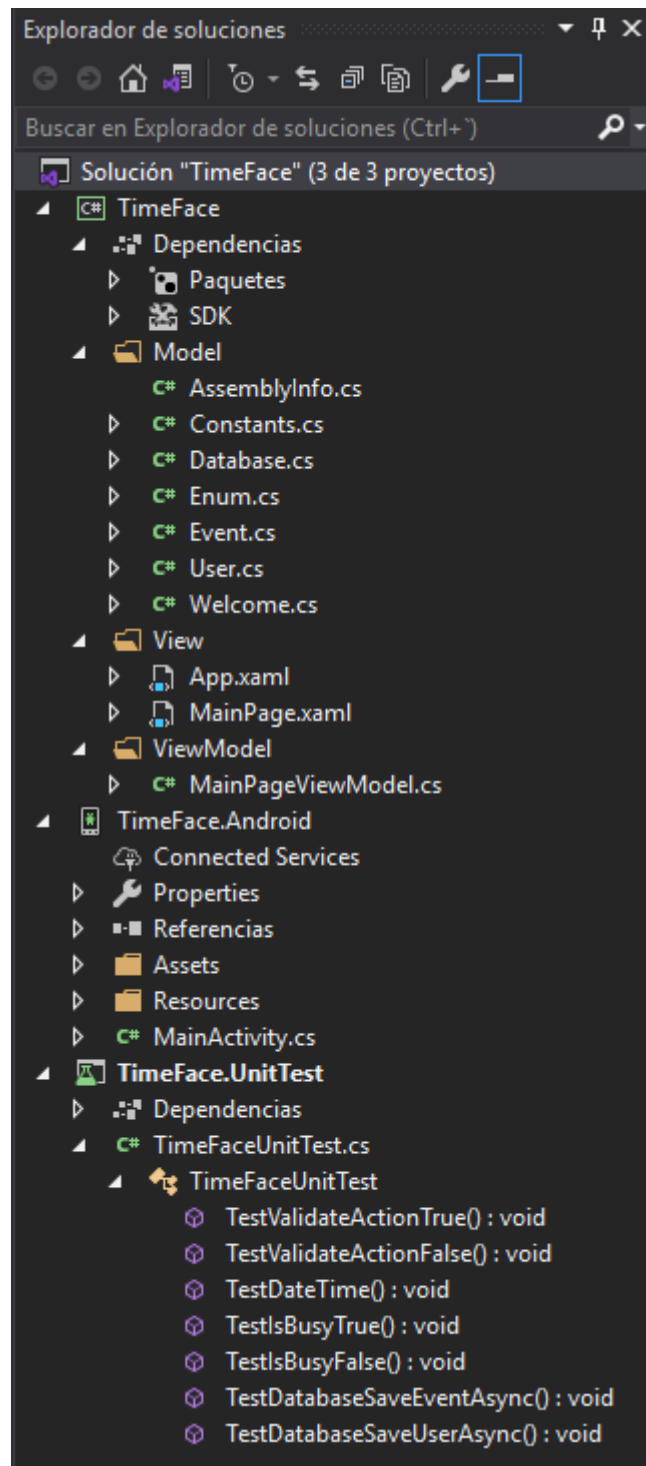


Figura 19: Esquema i estructura de la solució *TimeFace*

3.6 : Estat del projecte

L'estat de l'app *TimeFace* assolit durant el període d'implementació és una aplicació 100% funcional que satisfà totes les funcionalitats i casos d'ús descrits en els capítols anteriors de Disseny.

També destacar que degut als problemes descrits a l'apartat 3.1: *Presa de decisions durant el desenvolupament* la fase inicial del desenvolupament va patir una desviació de 10 dies aproximadament. Per tal de corregir aquestes desviacions, es van prendre mesures correctives per accelerar el desenvolupament com per exemple fer servir tecnologia amb la que estic més familiaritzat o fer servir serveis en *cloud* que es fan responsables de les tasques de *machine learning*. Algunes de les característiques que s'han omès o limitacions durant el desenvolupament són les següents:

- Amb el dispositiu físic que s'ha fet servir per a realitzar les proves, un *Samsung Galaxy A10*, no ha estat possible amb el *Xam.Plugin.Media* prendre el control de la càmera frontal del dispositiu i és necessari girar la càmera cada vegada que s'obra el control d'aquesta.
- Tot i que el servei *Microsoft.Azure.CognitiveServices.Vision.Face* retorna les coordenades de la quadrícula dels rostres detectats a la imatge no ha estat possible dibuixar i remarcar aquesta quadrícula tal i com es tenia previst sobre objectes del tipus *MediaFile* o *StreamImageSource*.
- Algunes de les dades dels usuaris s'han confiat a l'estructura de dades del servei *Microsoft Azure Cognitive Services* en comptes d'una estructura de dades pròpia tal i com es va plantejar inicialment. També el registre d'events es confia en una base de dades *SQLite* incrustada amb l'app i local al propi dispositiu. Una alternativa hagués sigut fer servir un sistema de bases de dades en *cloud* com per exemple *Azure SQL*, etc. d'aquesta manera desacobles la base de dades del dispositiu amb l'avantatge de poder compartir la mateixa base de dades amb varis dispositius, però estàs condicionat a la connexió a dades del dispositiu i perds l'opció d'operar *offline*, tot i que en aquest cas no té sentit operar *offline* degut a l'alta dependència que té l'app dels serveis en *cloud*.
- Com ja hem esmentat en anteriors capítols, existeixen necessitats previsibles que no s'han contemplat en aquesta fase d'implantació i que es podrien implementar en una fase més avançada del producte, com per exemple:
 - o Informes de registre d'events per usuari.
 - o *Backoffice* amb funcionalitats de perfil d'administrador de l'app *TimeFace*.
 - o *Backoffice* amb funcionalitats de perfil de gestió i anàlisis de les dades, *reportings*, *dashboards*, etc.

4. Proves

4.1 : Proves funcionals

Les proves funcionals s'han realitzat en un dispositiu *Samsung Galaxy A10* amb les següents característiques:

- Pantalla de 6.2 polsades.
- Resolució de 720 x 1520 píxels.
- Versió *Android 10*.
- 4GB RAM.
- Càmera principal de 13 MP i frontal de 5 MP.
- Connexió a Internet *Wi-Fi 802.11*

A continuació descrivim les proves funcionals realitzades per les principals funcionalitats i casos d'ús de *TimeFace*:

4.1.1 : Iniciar *TimeFace*:

L'usuari ha de ser capaç d'iniciar l'app *TimeFace* per començar a interactuar amb ella:

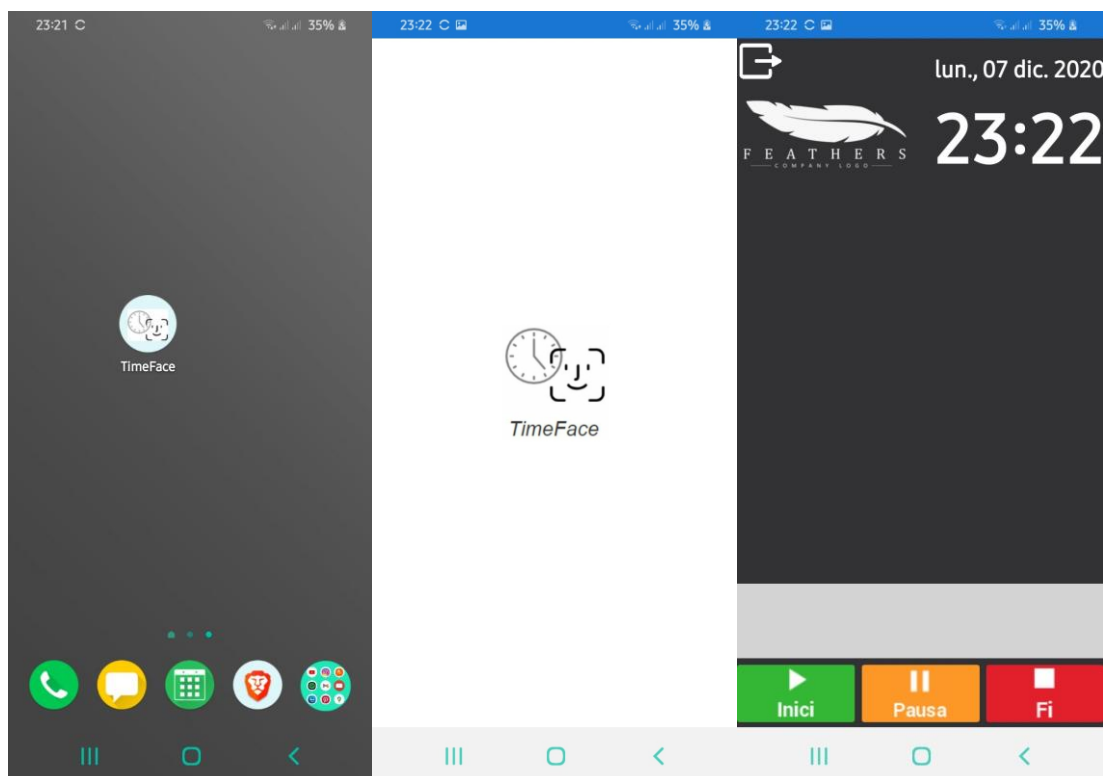


Figura 20: Proves funcionals: Iniciar *TimeFace*

4.1.2 : Inici jornada:

L'aplicació ha de ser capaç de reconèixer el rostre de la imatge, identificar-la i relacionar-la amb un usuari i enregistrar l'acció d'inici de jornada, així com identificar i controlar els possibles errors que puguin esdevenir d'aquesta acció:

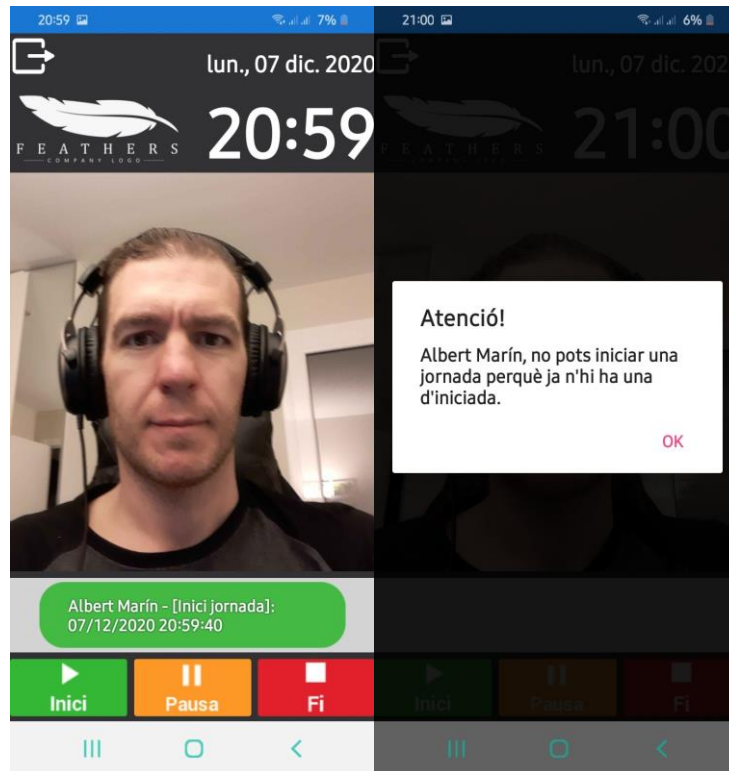


Figura 21: Proves funcionals: Inici jornada

4.1.3 : Pausa jornada:

L'aplicació ha de ser capaç de reconèixer el rostre de la imatge, identificar-la i relacionar-la amb un usuari i enregistrar l'acció de pausa de jornada amb el seu motiu, així com identificar i controlar els possibles errors que puguin esdevenir d'aquesta acció:

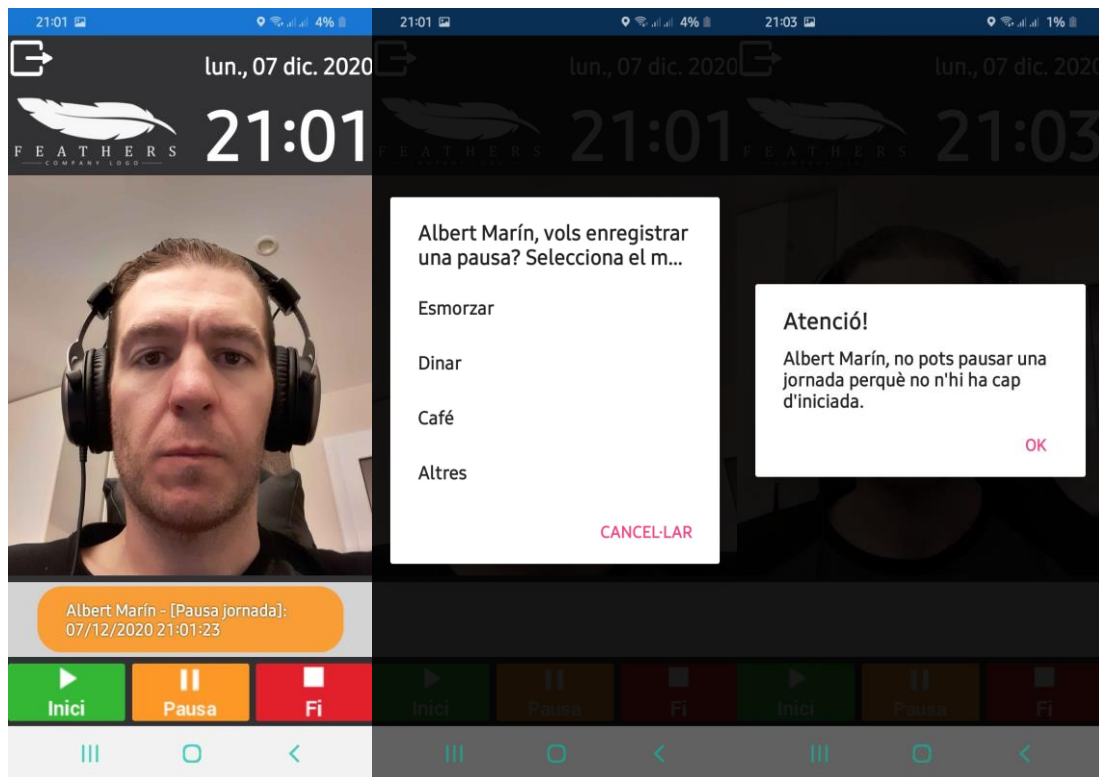


Figura 22: Proves funcionals: Pausa jornada

4.1.4 : Fi jornada:

L'aplicació ha de ser capaç de reconèixer el rostre de la imatge, identificar-la i relacionar-la amb un usuari i enregistrar l'acció de fi de jornada, així com identificar i controlar els possibles errors que puguin esdevenir d'aquesta acció:

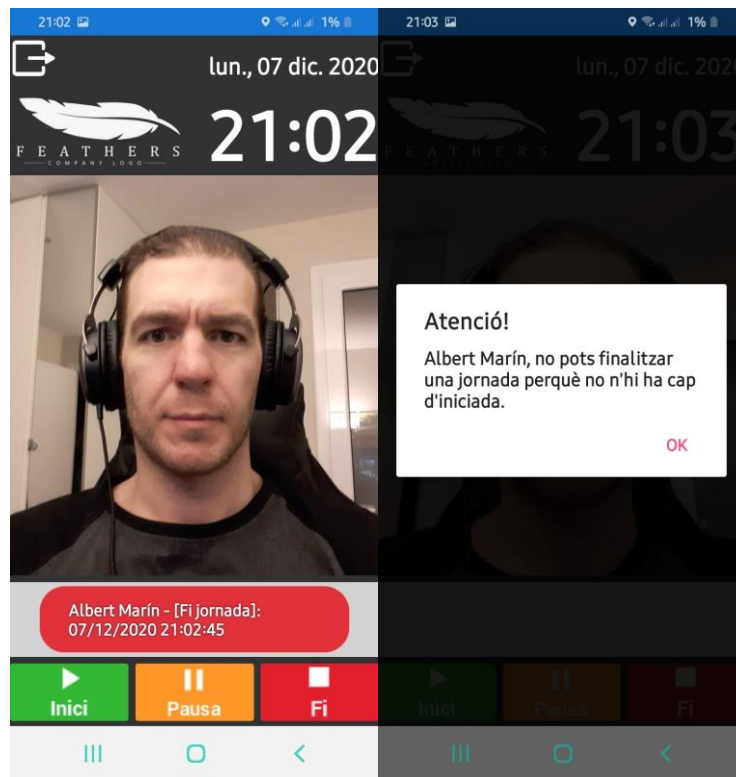


Figura 23: Proves funcionals: Fi jornada

4.1.5 : Alta d'usuari:

L'aplicació ha de ser capaç de reconèixer el rostres de la imatge, i identificar que aquella persona no correspon amb cap usuari reconegut per iniciar el procés d'alta d'usuari així com identificar i controlar els possibles errors que puguin esdevenir d'aquesta acció:

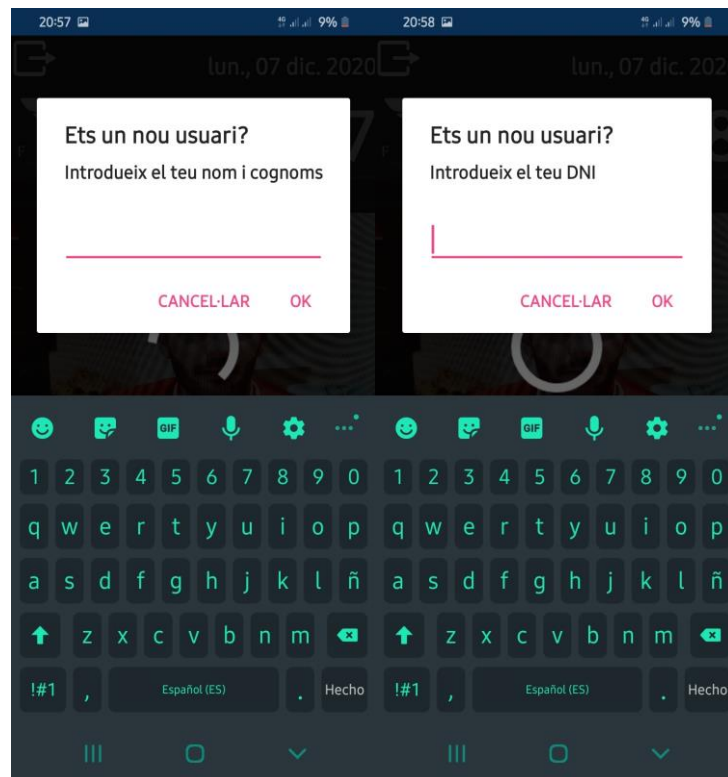


Figura 24: Proves funcionals: Alta d'usuari

4.1.6 : Finalitzar *TimeFace*:

L'aplicació ha de ser capaç de finalitzar *TimeFace*:

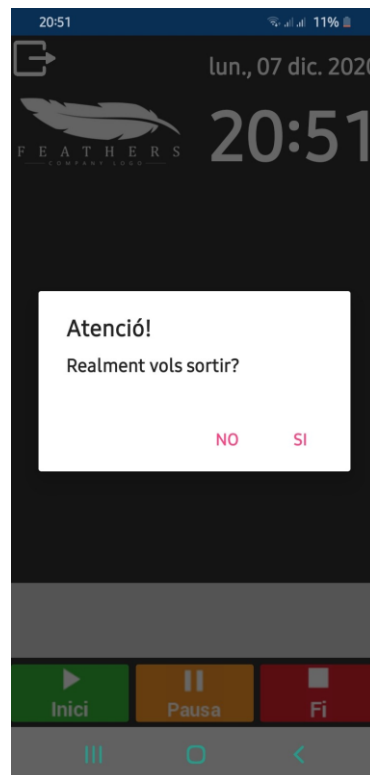


Figura 25: Proves funcionals: Finalitzar *TimeFace*

4.1.7 : Control d'errors:

L'aplicació ha de controlar certes situacions o casos d'ús per evitar que es produeixin comportaments no desitjats o errors informant a l'usuari:

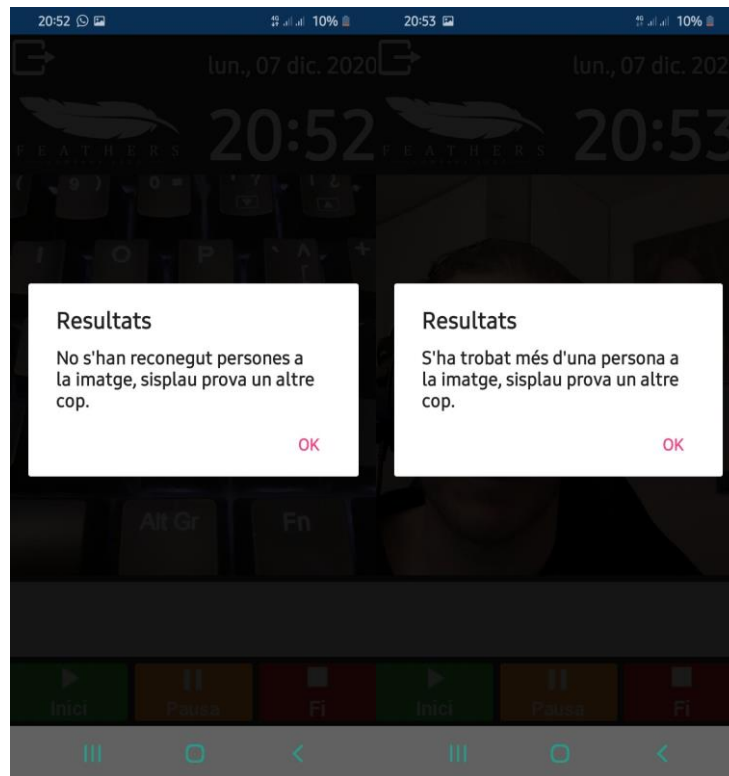


Figura 26: Proves funcionals: Control d'errors

4.2 : Proves unitàries

Amb la finalitat de realitzar les proves unitàries de la solució *TimeFace*, s'han utilitzat les pròpies eines *MSTest* que incorpora *Visual Studio 2019* per projectes *.NET* i *Core*:

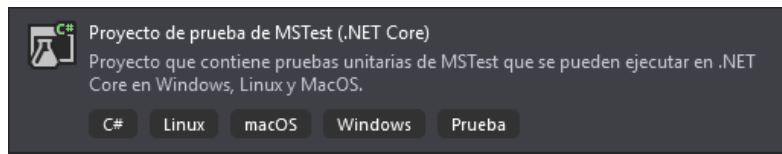


Figura 27: Proves unitàries: *MSTest*

El principal objectiu es comprovar el correcte funcionament d'una unitat aïllada de codi (funció, mètode, classe, etc.) per tal de garantir el correcte funcionament d'aquestes unitats per separat. A continuació es mostra un exemple d'algunes funcions candidates triades com a tests unitaris:

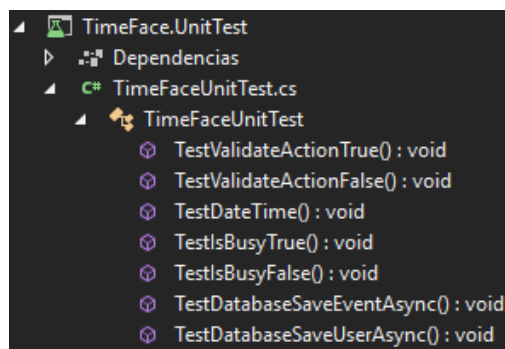


Figura 28: Proves unitàries: *Funcions implementades*

Per escriure els mètodes i les classes de test, s'ha utilitzat com a patró de *best practices* la convenció AAA (*Arrange, Act, Assert*). On primer es configuren els objectes de prova i es preparen els prerequisits per obtenir un resultat conegut, després es realitza el test i finalment s'avalua el resultat obtingut vs. l'esperat per decidir si el codi es comporta de manera satisfactòria:

```
[TestClass]
//Referencias
public class TimeFaceUnitTest
{
    [TestMethod]

    public async void TestValidateActionTrue()
    {
        //Arrange:
        Person person = new Person
        {
            Name = "Albert Marín",
            UserData = "45488099X"
        };
        eAction action = eAction.Start;
        var myClass = new MainPageViewModel();

        //Act:
        bool result = await myClass.ValidateAction(action, person);

        //Assert:
        Assert.IsTrue(result);
    }
}
```

Figura 29: Proves unitàries: *Exemple de test unitari*

Finalment mostrem el resultat de les proves unitàries implementades on es pot observar que ha estat satisfactori en tots els casos:

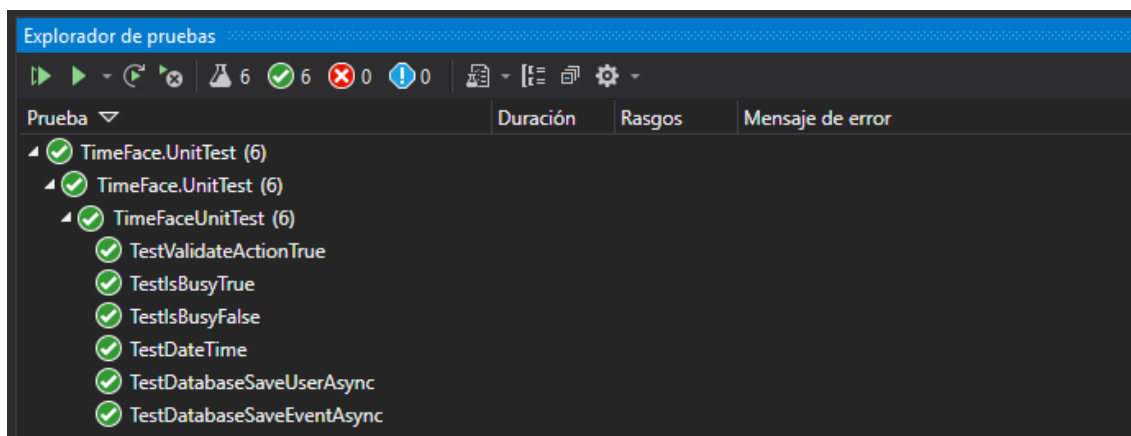


Figura 30: Proves unitàries: Resultats de les proves unitàries

5. Conclusions

El principal objectiu d'aquest treball de final de grau era desenvolupar una aplicació per a dispositius *Android* assolint tots els coneixements necessaris per tal d'obtenir un producte satisfactori que cobris els principals requeriments funcionals. Des d'aquest punt de vista, puc concloure que per a mi ha estat una experiència molt enriquidora i satisfactòria de cara a la meva formació i he ampliat els meus coneixements en l'àmbit del desenvolupament per dispositius *Android* i computació en *cloud*.

Durant el període d'implementació vaig aprendre que l'entorn de desenvolupament *Android Studio* i els seus *plugins*, *frameworks* i *APIs* requerien d'una corba d'aprenentatge i temps del que no disposava, també vaig trobar dificultats per trobar documentació o referències sobre certs aspectes i el recursos de *hardware* dels que disposava tampoc eren adequats. També he pres consciència de la complexitat de portar a bon terme un projecte d'aquesta naturalesa.

En relació als requeriments i objectius funcionals de l'app *TimeFace*, s'han assolit tots satisfactòriament, tot i que es podrien millorar alguns aspectes com per exemple dibuixar un marc al voltant de les cares de les imatges, traslladar tot el model de dades al *cloud* per desacoblar-lo del dispositiu, etc.

Un aspecte molt important del projecte ha estat la planificació i la dedicació de recursos i esforços. Gràcies a com està estructurada l'assignatura, és senzill fer inicialment una planificació bastant aproximada del que serà el pla de treball real, tot i que considero que va ser un encert triar *Agile* com a metodologia durant la fase d'implementació, perquè em va permetre ser més flexible i adaptar-me millor al canvi per modificar les decisions inicials d'arquitectura que vaig definir degut als problemes que em vaig anar trobant. Com ja vaig esmentar en el capítol d'Implementació per justificar els canvis d'arquitectura, el projecte inicialment va patir retards en el desenvolupament que es van anar recuperant a mida que vaig anar aplicant la nova arquitectura per tal d'evitar arribar al final de la fase d'implementació sense haver assolit part dels objectius funcionals.

Com a conclusió final diria que ha estat una experiència plenament satisfactòria a nivell acadèmic on he adquirit nous coneixements molt valuosos que podré aplicar en el futur en el camp laboral.

Línies de treball futur:

Com ja he esmentat anteriorment, l'estat actual de projecte *TimeFace* es va centrar premeditadament en les funcionalitats de reconeixement i identificació dels rostres d'una imatge i l'enregistrament d'events per tal d'assolir uns objectius finals sense plantejar altres objectius massa ambiciosos i evitar no complir objectius. Però seria molt interessant en una segona o tercera fase de desenvolupament de producte on es podrien plantejar i abordar les següents qüestions:

- Optimització del producte i codi actual amb model de dades desacoblat.
- Tota la part d'anàlisi d'events i *reporting* des de la mateixa app *TimeFace* de cara a l'usuari final.
- Multi idioma.
- Desenvolupar un *Backoffice web* que permetés totes les funcionalitats de gestió d'usuaris, perfils, rols, *dashboard* i *reporting* administrador.
- Campanyes de màrqueting i posicionament dins de les diferents *stores*.

6. Glossari

.NET: Framework de desenvolupament de Microsoft.

Act: Secció d'actuació de les proves unitàries.

Agile: Metodologia de desenvolupament.

Amazon Web Services (AWS): Col·lecció de serveis de computació a internet de l'empresa Amazon.

Android Studio: Entorn de desenvolupament integrat per el sistema operatiu *Android* de l'empresa Google.

Android: Sistema operatiu per dispositius mòbils de l'empresa Google.

API: Interfície de programació d'aplicacions.

App: Aplicació.

Arrange: Secció de preparació de les proves unitàries.

Assert: Secció d'afirmació de les proves unitàries.

Azure: Col·lecció de serveis de computació a internet de l'empresa Microsoft.

Back-end: Part del programari que processa i gestiona les dades.

Backoffice: Programari encarregat de donar suport, gestionar i configurar un altre programari.

BBDD: Base de dades.

Best practices: Bones pràctiques.

Biomètrics: Estudi del reconeixement inequívoc de les persones.

C#: Llenguatge de programació.

Cloud computing: Computació en servidors remots hostejats a Internet.

Cloud: Terme utilitzat per fer referència a serveis que es troben a Internet.

Cognitive Services: Serveis tecnològics que imiten sentits cognitius.

Contactless: Tecnologia d'intercanvi d'informació sense contacte físic.

Covid 19: Pandèmia global causada per la malaltia coronavirus 2019.

Cross platform: Programari que es pot executar en diferents plataformes.

Dashboard: Quadre de comandament.

DCU: Disseny centrat en l'usuari.

Debug: Mode d'execució del programari per depurar les instruccions pas a pas.

Deploy: Desplegament de programari.

Feedback: Obtenció de les impressions o opinions.

Firebase: Plataforma de desenvolupament d'aplicacions web o mòbils de l'empresa Google.

Framework: Conjunt d'artefactes, funcions i mòduls de programari.

Geolocalització: Capacitat d'obtenir la ubicació geogràfica d'un dispositiu.

Google: Empresa tecnològica basada en serveis a Internet.

IA: Intel·ligència artificial.

iOS: Sistema operatiu per dispositius mòbils de l'empresa Apple.

Javascript: Llenguatge de programació interpretat.

JSON: Format de text per intercanvi de dades.

Kotlin: Llenguatge de programació.

Lollipop: Una de les versions del sistema operatiu *Android*.

Machine learning: Aprenentatge automàtic.

Microsoft: Empresa tecnològica.

MIFARE: Marca registrada per fer referència a una tecnologia d'intercanvi d'informació sense contacte físic.

MLKit: Plataforma de desenvolupament d'aplicacions web o mòbils de l'empresa Google.

MVVM (Model-view-viewmodel): Metodologia de desenvolupament model-vista-vistamodel.

NoSQL: Estructures de dades que no es basen en el llenguatge estructurat de consultes.

Offline: Desconnectat de la xarxa de dades.

Open Source: Codi obert.

PC: Computadora personal.

PIN: Número d'identificació personal.

Python: Llenguatge de programació.

Reporting: Part del programari per obtenir dades i informes.

Responsive design: Disseny d'interfície d'usuari adaptable a diferents dispositius, pantalles i resolucions.

RGPD: Reglament general de protecció de dades.

SaaS: Programari com a part d'un servei.

Scrum: Metodologia de desenvolupament de programari.

SDK: Kit de desenvolupament de programari.

Shadowing: Mètode d'obtenció d'informació per part de l'usuari.

Sketch: Maqueta no funcional del programari.

Smartphone: Telèfon mòbil intel·ligent.

Software: Programari.

Sprint: Període de temps i activitats de les fases de desenvolupament.

SQLite: Gestor de bases de dades estructurat lleuger.

Stores: Botigues de les diferents plataformes de dispositius mòbils.

Tablet: Tauleta.

Tensorflow: Biblioteca de programari per aprenentatge automàtic.

TimeFace: Aplicació de control de temps laboral en la que està basat aquest treball de final de grau.

UI: Interfície d'usuari.

UML: Llenguatge unificat de modelatge de programari.

UWP: Plataforma universal de Windows.

Visual Studio: Entorn de desenvolupament integrat de l'empresa *Microsoft*.

Web: Pàgina electrònica hostejada a Internet.

Windows: Sistema operatiu de l'empresa *Microsoft*.

Xamarin: Llenguatge de programació per dispositius mòbils multiplataforma de *Microsoft*.

7. Bibliografia

- Roxtron. *MIFARE Cards*
[en línia] [Data consulta: 16/09/2020]
[Disponible a la URL: <https://www.roxtron.com/mifare-cards>]
- Bixpe. *Control horario*
[en línia] [Data consulta: 16/09/2020]
[Disponible a la URL: <https://play.google.com/store/apps/details?id=com.bixpe.chronos&hl=es>]
- Intratime. *Control horario*
[en línia] [Data consulta: 16/09/2020]
[Disponible a la URL: <https://play.google.com/store/apps/details?id=com.protectfive.intratime>]
- Duocom Europe. *Reloj Laboral*
[en línia] [Data consulta: 16/09/2020]
[Disponible a la URL: https://play.google.com/store/apps/details?id=es.duocom.reloj_laboral]
- Wikipedia. *Scrum*
[en línia] [Data consulta: 06/10/2020]
[Disponible a la URL: [https://es.wikipedia.org/wiki/Scrum_\(desarrollo_de_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software))]
- Jordi Almirall López. *Diseño Centrado en el Usuario para dispositivos móviles*
[en línia] [Data consulta: 06/10/2020]
[Disponible a la URL: <http://cv.uoc.edu/webapps/xwiki/wiki/matm1202es/view/Main/6.4+Inspecci%C3%B3+i+ests+amb+usuaris+%5Bavaluaci%C3%B3+5D>]
- Muriel Garreta Domingo i Enric Mor Pera. *Disseny centrat en l'usuari*
[PDF] [UOC: PID_00176048] [Data consulta: 06/10/2020]
- UOC. *Design Toolkit*
[en línia] [Data consulta: 06/10/2020]
[Disponible a la URL: http://materials.cv.uoc.edu/cdocent/design_toolkit/es/pdf/escenarios.pdf]
- CX INSIGHTS by Sabina Idler. *How user scenarios help to improve your UX*
[en línia] [Data consulta: 06/10/2020]
[Disponible a la URL: <https://usabilla.com/blog/how-user-scenarios-help-to-improve-your-ux/>]
- Google. *Introducción a Android Studio*
[en línia] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://developer.android.com/studio/intro>]
- Kotlin. *Learn Kotlin*
[en línia] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://kotlinlang.org/docs/reference/>]
- Firebase. *Agrega Firebase al proyecto de Android*
[en línia] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://firebase.google.com/docs/android/setup>]

- Firebase. *Kit de AA para Firebase*
[en línea] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://firebase.google.com/docs/ml-kit>]
- Firebase. *Detección de rostro*
[en línea] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://firebase.google.com/docs/ml-kit/detect-faces>]
- ML Kit. *ML Kit Guide*
[en línea] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://developers.google.com/ml-kit/guides>]
- TensorFlow. *Documentación esencial*
[en línea] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://www.tensorflow.org/guide>]
- Amazon. *Searching faces in a collection*
[en línea] [Data consulta: 26/10/2020]
[Disponible a la URL: <https://docs.aws.amazon.com/rekognition/latest/dg/collections.html>]
- Wikipedia. *Model-view-viewmodel*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93viewmodel>]
- Microsoft. *Uso de SQLite.NET con Android*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://docs.microsoft.com/es-es/xamarin/android/data-cloud/data-access/using-sqlite-orm>]
- Microsoft. *¿Qué es el servicio Azure Face?*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://docs.microsoft.com/es-es/azure/cognitive-services/face/overview>]
- Microsoft. Inicio rápido: *Uso de la biblioteca cliente de Face*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://docs.microsoft.com/es-es/azure/cognitive-services/Face/Quickstarts/client-libraries?pivot=programming-language-csharp&tabs=visual-studio>]
- Microsoft. *Face API - v1.0*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395236>]
- Microsoft. *Azure Cognitive Services*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://azure.microsoft.com/es-es/services/cognitive-services/>]
- Microsoft. *Documentación de Xamarin.Forms*
[en línea] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://docs.microsoft.com/es-es/xamarin/xamarin-forms/>]

- Xamarin Community Site. *Help*
[en línia] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://www.xamarinhelp.com/>]
- Nuget.org, James Montemagno. *Xam.Plugin.Media*
[en línia] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://www.nuget.org/packages/Xam.Plugin.Media/>]
- Nuget.org, Ishrak El Andaloussi. *Plugin.Toast*
[en línia] [Data consulta: 10/11/2020]
[Disponible a la URL: <https://www.nuget.org/packages/Plugin.Toast/>]
- Wikipedia. *Prueba unitaria*
[en línia] [Data consulta: 23/11/2020]
[Disponible a la URL: https://es.wikipedia.org/wiki/Prueba_unitaria]
- Kilt & Code, Steven McIntock.
Best Practices for Writing Unit Tests in C# for Bulletproof Code
[en línia] [Data consulta: 23/11/2020]
[Disponible a la URL: <https://www.kiltandcode.com/2019/06/16/best-practices-for-writing-unit-tests-in-csharp-for-bulletproof-code/>]

8. Annexos

8.1 : Guia d'usuari:

8.1.1: Com iniciar *TimeFace*?

1. Busca la icona *TimeFace* entre les aplicacions del teu dispositiu *Android*:

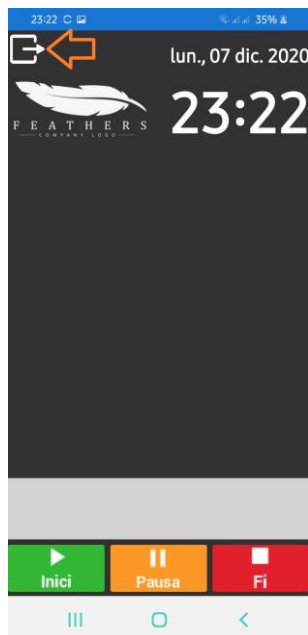


2. Fes clic a sobre la icona i executa l'aplicació, es mostrarà la pàgina d'inici indicant que *TimeFace* s'està iniciant:



8.1.2: Com finalitzar *TimeFace*?

1. Busca la icona de sortir dins de *TimeFace*:



2. Fes clic a sobre la icona de sortir, *TimeFace* et demanarà confirmació de si realment vols sortir:



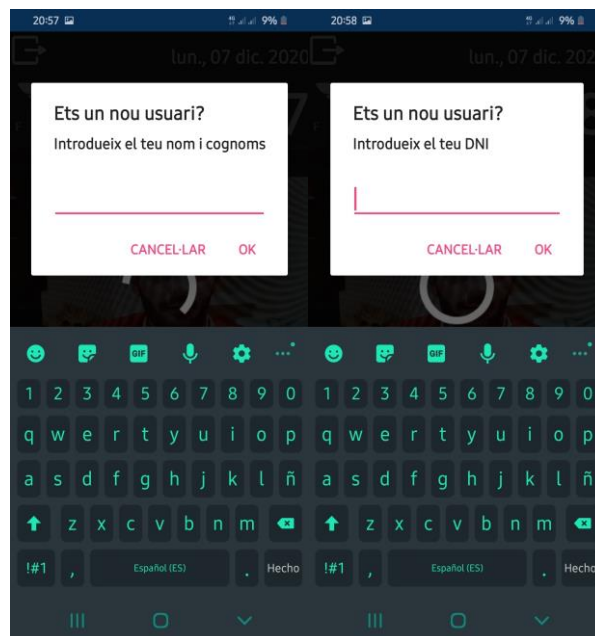
3. Si respos afirmativament *TimeFace* es tancarà.

8.1.3: Com registrar un nou usuari?

1. Busca el botó per iniciar la jornada:



2. Fes clic a sobre del botó per iniciar la jornada.
3. S'obrirà el control de la càmera frontal del teu dispositiu *Android*.
4. Dispara una foto de la teva cara.
5. Si ets un nou usuari, *TimeFace* et demanarà que introdueixis les teves dades:

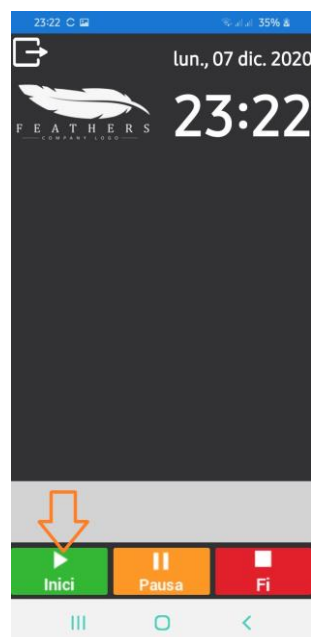


6. *TimeFace* enregistrarà les dades del teu usuari i per acabar indicarà que s'ha iniciat la teva jornada:



8.1.4: Com Iniciar jornada?

1. Busca el botó per iniciar la jornada:



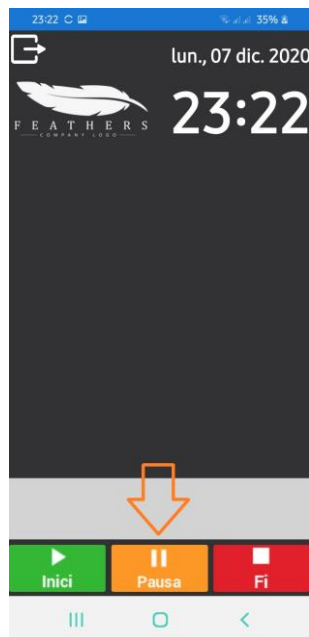
2. Fes clic a sobre del botó per iniciar la jornada.
3. S'obrirà el control de la càmera frontal del teu dispositiu *Android*.
4. Dispara una foto de la teva cara.

5. *TimeFace* enregistrarà les dades del teu usuari i per acabar indicarà que s'ha iniciat la teva jornada:



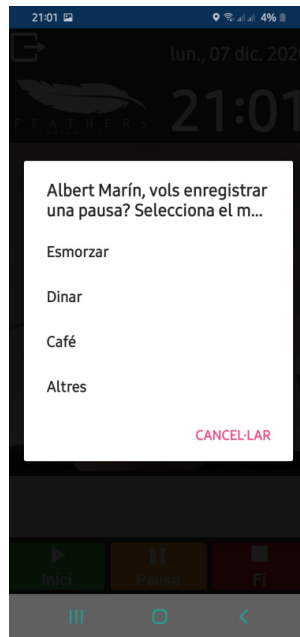
8.1.5: Com Pausar la jornada?

1. Busca el botó per pausar la jornada:



2. Fes clic a sobre del botó per pausar la jornada.
3. S'obrirà el control de la càmera frontal del teu dispositiu *Android*.
4. Dispara una foto de la teva cara.

5. *TimeFace* et sol·licitarà que indiquis el motiu de la pausa:

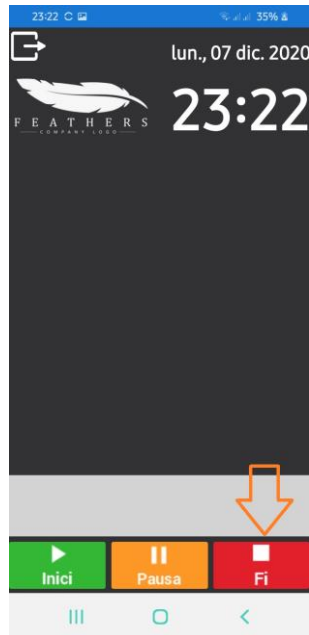


6. Finalment *TimeFace* indicarà que s'ha pausat la teva jornada:



8.1.6: Com Finalitzar la jornada?

1. Busca el botó per finalitzar la jornada:



2. Fes clic a sobre del botó per finalitzar la jornada.
3. S'obrirà el control de la càmera frontal del teu dispositiu *Android*.
4. Dispara una foto de la teva cara.
5. Finalment *TimeFace* indicarà que s'ha finalitzat la teva jornada:



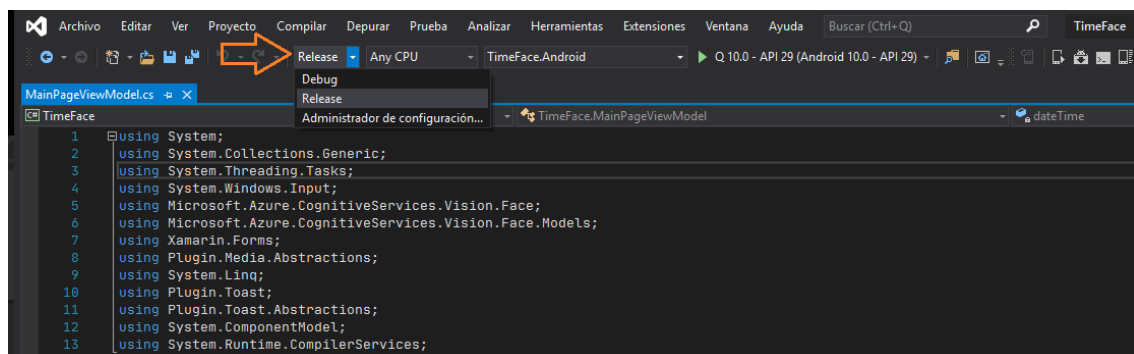
8.2 : Guia d'usuari avançat:

8.2.1: Com compilar *TimeFace*?

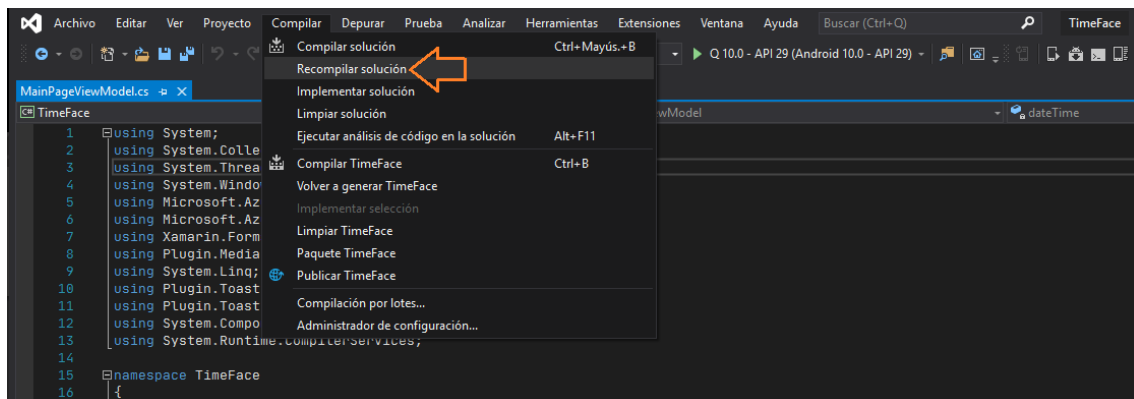
1. Obrir el fitxer de la solució *TimeFace* amb *Visual Studio 2019*:

Nombre	Fecha de modificación	Tipo	Tamaño
.vs	23/11/2020 20:27	Carpeta de archivos	
TestResults	09/12/2020 21:03	Carpeta de archivos	
TimeFace	23/11/2020 20:27	Carpeta de archivos	
TimeFace.UnitTest	09/12/2020 0:49	Carpeta de archivos	
TimeFace.sln	08/12/2020 23:31	Visual Studio Solu...	3 KB

2. Seleccionar la versió que volem compilar (*Release* o *Debug*) i la solució *TimeFace.Android*:

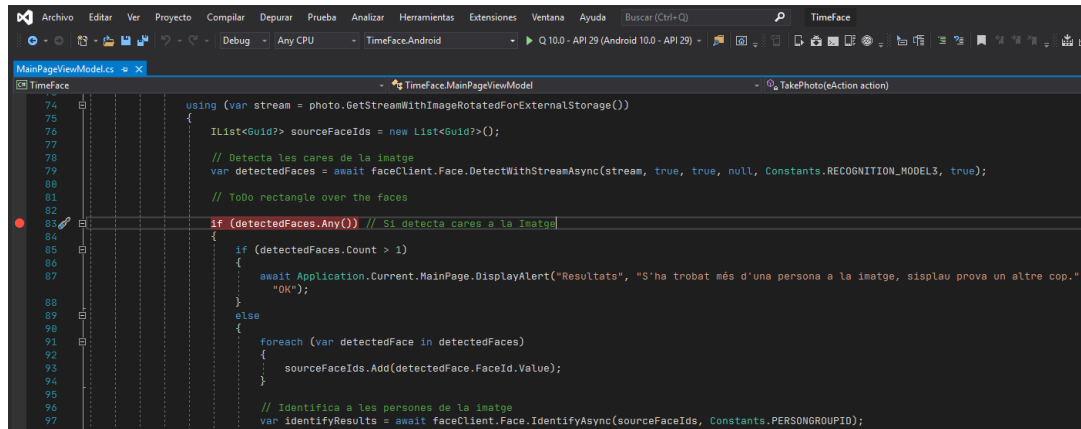


3. Anar al menú *Compilar > Recompilar solució* i executar:

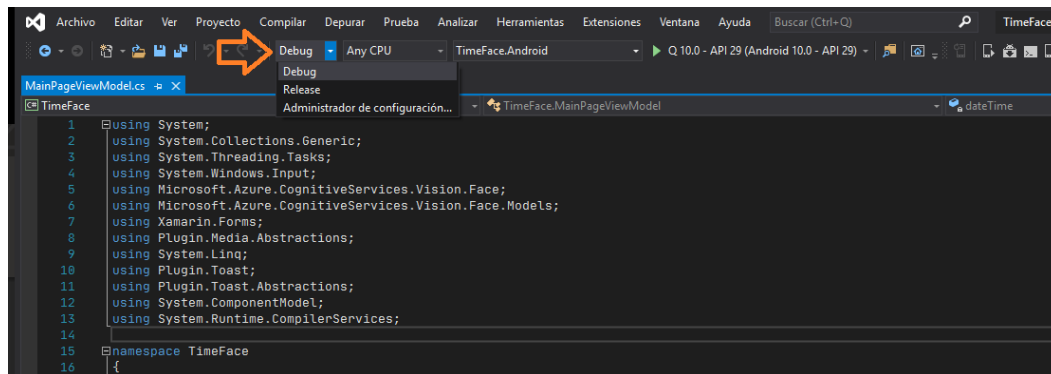


8.2.2: Com debugar TimeFace?

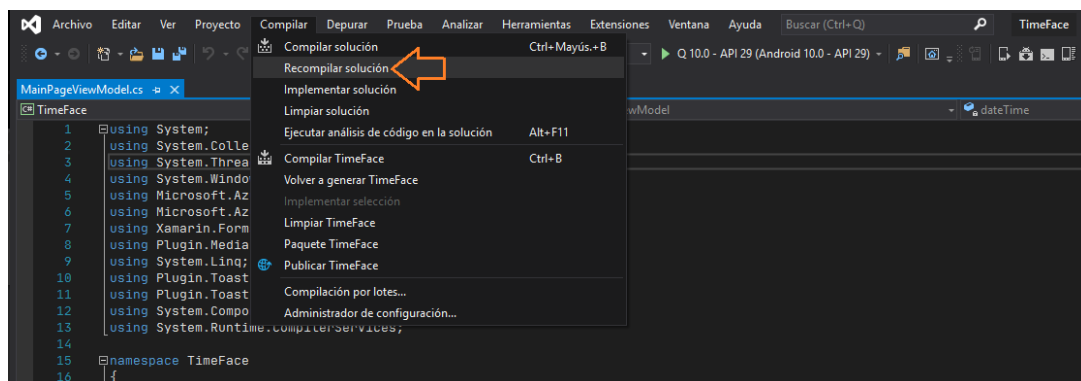
1. Introduir punts d'interrupció dins del codi de *TimeFace*:



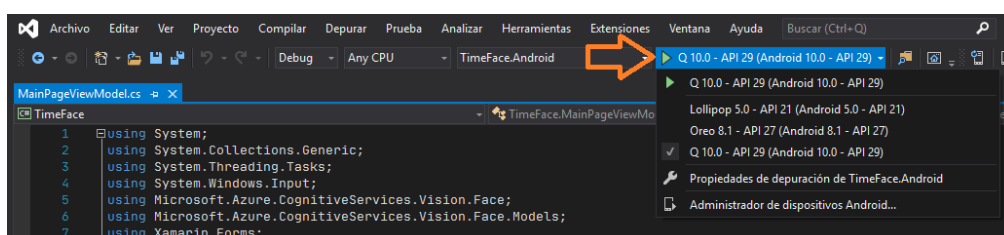
2. Seleccionar la versió *Debug* per compilar la solució *TimeFace.Android*:



3. Anar al menú *Compilar > Recompilar solució* i executar:

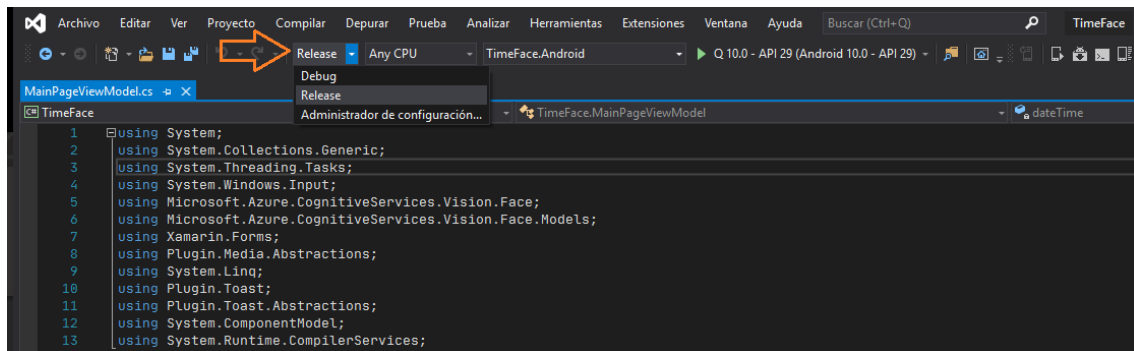


4. Executar la compilació seleccionant un dispositiu físic connectat al PC o un dispositiu virtual:

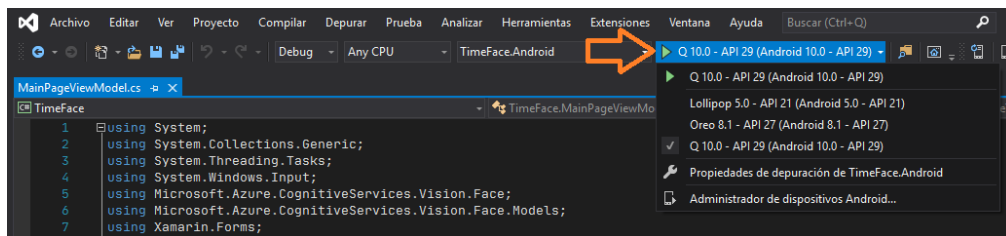


8.2.3: Com executar *TimeFace*?

1. Seleccionar la versió que volem executar (*Release* o *Debug*) i la solució *TimeFace.Android*:



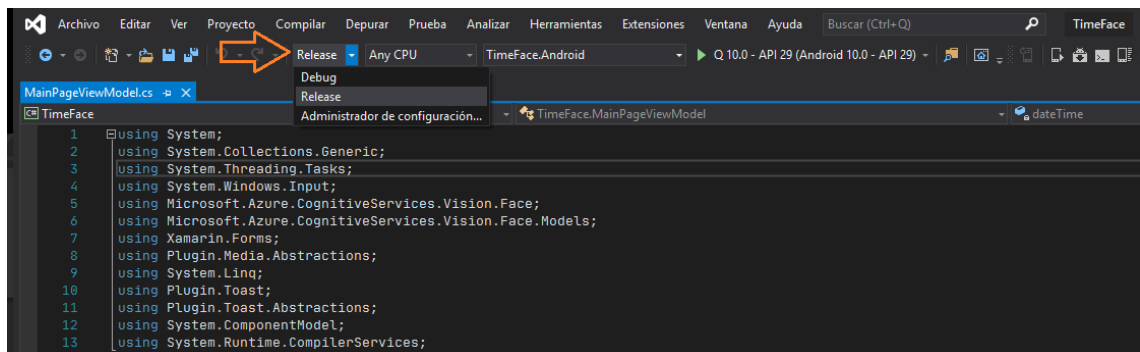
2. Executar la compilació seleccionant un dispositiu físic connectat al PC o un dispositiu virtual:



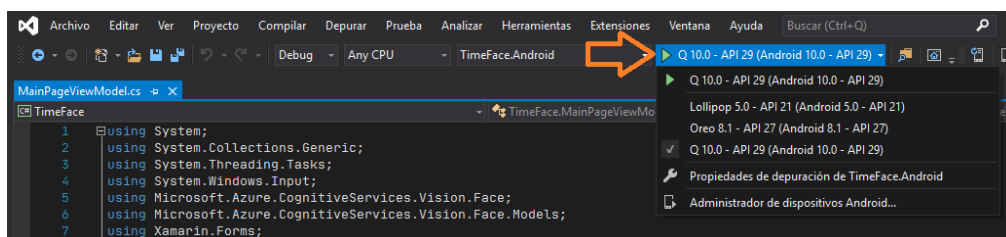
8.2.4: Com instal·lar *TimeFace* en un dispositiu *Android*?

Mètode 1:

1. Seleccionar la versió *Release* i compilar la solució *TimeFace.Android*:

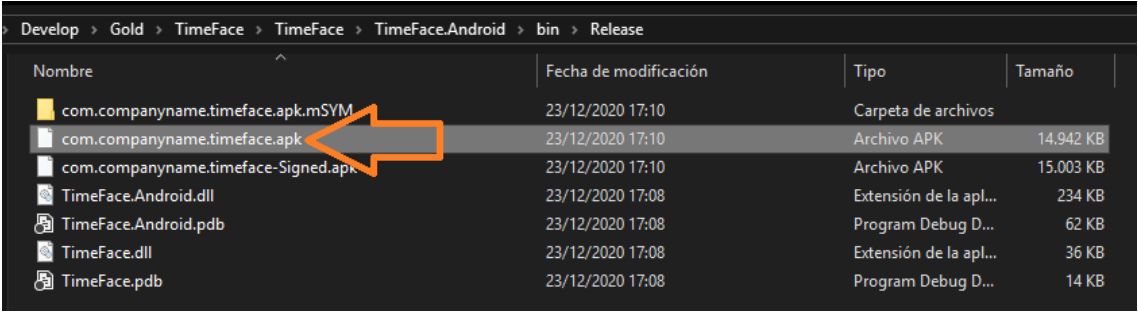


2. Desplegar la versió seleccionant el dispositiu físic i executant:



Mètode 2:

1. Buscar l'aplicació generada dins de la carpeta *Release* de la solució la aplicació amb extensió *.apk.



The screenshot shows a file explorer window with the path: Develop > Gold > TimeFace > TimeFace > TimeFace.Android > bin > Release. The table below lists the files in this directory.

Nombre	Fecha de modificación	Tipo	Tamaño
com.companyname.timeface.apk.mSYM	23/12/2020 17:10	Carpeta de archivos	
com.companyname.timeface.apk	23/12/2020 17:10	Archivo APK	14.942 KB
com.companyname.timeface-Signed.apk	23/12/2020 17:10	Archivo APK	15.003 KB
TimeFace.Android.dll	23/12/2020 17:08	Extensión de la apl...	234 KB
TimeFace.Android.pdb	23/12/2020 17:08	Program Debug D...	62 KB
TimeFace.dll	23/12/2020 17:08	Extensión de la apl...	36 KB
TimeFace.pdb	23/12/2020 17:08	Program Debug D...	14 KB

2. Copiar dins del dispositiu *Android* on es vol instal·lar *TimeFace*.
3. Donar permisos de seguretat al dispositiu per instal·lar aplicacions de fonts alternatives o desconegudes.
4. Fer clic a sobre de el fitxer *.apk del dispositiu per instal·lar *TimeFace*.