



**Politecnico
di Torino**

Politecnico di Torino

Laurea Magistrale in Ingegneria del Cinema e dei Mezzi di Comunicazione

A.a. 2024/2025

Sessione di laurea Dicembre 2025

Design e sviluppo di un'applicazione mobile per la timbratura da remoto

Relatore:

Giovanni Malnati

Candidato:

Federica Masciangelo

Sommario

Negli ultimi anni, l'adozione sempre più diffusa di Internet e delle nuove modalità di lavoro, sempre più flessibili, ha profondamente trasformato il modo in cui vengono gestiti i processi di lavoro e di rilevazione presenze. Le organizzazioni necessitano quindi di strumenti digitali in grado di garantire accessibilità, sicurezza e semplicità d'uso, senza compromettere l'affidabilità dei dati.

In tale contesto, la presente tesi si propone di progettare e sviluppare un'applicazione mobile multiplatforma dedicata alla gestione delle timbrature aziendali, integrata con il sistema backend dell'azienda e orientata a un'esperienza utente intuitiva e coerente.

L'elaborato affronta l'intero ciclo di vita del progetto, partendo dalla fase di definizione e analisi dei requisiti fino ad arrivare alla fase di testing.

L'applicazione è stata sviluppata utilizzando React Native ed Expo, con un'architettura basata su componenti riutilizzabili, servizi API centralizzati e un sistema di traduzioni multilingue.

Nella sezione iniziale viene condotta un'analisi dello stato dell'arte e dei principali riferimenti teorici relativi ai sistemi di gestione delle presenze. Successivamente, viene illustrata la metodologia utilizzata per la raccolta dei requisiti e per l'analisi condotta sugli utenti.

La parte centrale del lavoro presenta il processo di progettazione e sviluppo dell'applicazione, ponendo particolare attenzione all'esperienza utente, alla definizione dell'interfaccia e alle funzionalità principali.

Infine, vengono discussi i risultati ottenuti attraverso la validazione e i test condotti, al fine di verificare la stabilità funzionale, la persistenza dei dati e la coerenza visiva tra le piattaforme, per poi concludere con le considerazioni finali e i possibili sviluppi futuri del progetto.

Nel complesso, il lavoro ha permesso di realizzare un'applicazione funzionante, in grado di semplificare le operazioni di timbratura e migliorare l'esperienza dei dipendenti. I risultati ottenuti confermano la validità dell'approccio adottato e pongono le basi per futuri miglioramenti.

Ringraziamenti

*Ringrazio il Professor Malnati e la Sirius,
per aver reso possibile questo progetto e per aver creduto in me.*

*Ai miei genitori, i miei pilastri e le mie radici,
per non avermi mai lasciato pensare di dover essere diversa
e per avermi fatto credere che tutto ciò che volevo fosse possibile.*

*Ai miei colleghi, che hanno alleggerito i giorni più pesanti
e non si sono mai tirati indietro quando avevo bisogno di una mano.*

*Ai miei amici di giù
che mi ricordano che certe distanze esistono solo sulla carta.*

*Ai miei amici di Torino
che hanno costruito con me una seconda casa.*

*A G., per esserci sempre stata,
e per aver custodito la nostra amicizia, anche nei silenzi.
Per non aver lasciato andare il filo, nemmeno quando lo tenevamo da due città
diverse.*

*Ad A., per le ore che non si contano, per i nodi sciolti senza far domande, per
ricordarmi di non arrendermi mai.*

*A te, che hai spento tutte le voci che dicevano che non ce l'avrei fatta.
Grazie per avermi salvata da me stessa,
per avermi aiutata quando non volevo chiedere aiuto,
per aver bilanciato il mio caos,
per aver creato un locus amoenus quando il mondo intorno a me sembrava troppo
rumoroso.*

A chi, nonostante l'ansia, riesce a fare tutto.

Indice

Elenco delle tabelle	V
Elenco delle figure	VI
Acronimi	VIII
1 Introduzione	1
1.1 Contesto e motivazioni	1
1.2 Problema di partenza	2
1.3 Obiettivi della tesi e del progetto	2
2 Stato dell'arte	4
2.1 Sistemi di gestione presenze: evoluzione dai metodi tradizionali a quelli digitali	4
2.2 Applicazioni mobile per gestire le presenze: analisi dei competitors .	5
2.3 Architetture client-server per sistemi mobile	7
3 Metodologia di ricerca e analisi preliminare	9
3.1 Approccio metodologico	9
3.2 Raccolta dei requisiti: obiettivi dell'azienda	10
3.3 Analisi degli utenti e definizione delle personas	11
3.4 Questionario agli utenti: metodologia e campione	14
3.5 Analisi dei feedback qualitativi e quantitativi	16
4 Progettazione e prototipazione	19
4.1 Modello di interazione e user flow	19
4.2 Wireframe e sketch iniziali	20
4.3 UI kit: colori, tipografia, componenti principali	22
4.3.1 Colori	22
4.3.2 Tipografia	24
4.3.3 Nome e identità visiva dell'applicazione	25

4.3.4	Componenti principali	25
4.4	Layout e design delle schermate principali	26
4.4.1	Home	27
4.4.2	Timbrature	28
4.4.3	Timetable	29
4.4.4	Impostazioni	29
4.5	Accessibilità e principi di usabilità adottati	30
5	Implementazione e sviluppo	32
5.1	Tecnologie utilizzate	32
5.1.1	Librerie principali	32
5.1.2	Test e ambienti di esecuzione	33
5.2	Architettura Software	33
5.3	Funzionalità principali sviluppate	34
5.3.1	Autenticazione sicura e gestione delle sessioni	34
5.3.2	Timbratura e gestione delle presenze	35
5.3.3	Calendario e visualizzazione storica delle presenze	36
5.3.4	Timbratura	37
5.3.5	Localizzazione e supporto multilingue	38
6	Test e validazione	40
6.1	Tipologia di test e metodologia adottata	40
6.2	Risultati e analisi dei test end-to-end	41
6.3	Test di usabilità e accettazione	43
6.3.1	Risultati quantitativi	43
6.3.2	Risultati qualitativi	44
6.4	Considerazioni e modifiche apportate	45
7	Conclusioni e sviluppi futuri	46
A	Componenti e varianti grafiche	48
B	Varianti grafiche delle schermate	55
	Bibliografia	58

Elenco delle tabelle

3.1	Requisiti funzionali e non funzionali	11
3.2	Risultati e implicazioni del questionario	18
6.1	Risultati dei test E2E	42
6.2	Risultati dei test di usabilità	44

Elenco delle figure

2.1	Architettura client-server	8
3.1	User Personas	14
3.2	Risultati in merito alla domanda sulle informazioni presenti nella home	16
3.3	Risultati in merito alla domanda su un widget personalizzato	17
4.1	User Flow	21
4.2	Wireframe Home	22
4.3	Wireframe Timbrature	22
4.4	Wireframe Timesheet	23
4.5	Wireframe Impostazioni	23
4.6	Identità visiva dell'app	25
4.7	Schermata Home (modalità chiara)	27
4.8	Schermata Timbrature (modalità chiara)	28
4.9	Schermata Timbrature (modalità chiara / trasferta)	28
4.10	Schermata Timetable (modalità chiara)	29
4.11	Schermata Impostazioni (modalità chiara)	30
A.1	Colori	49
A.2	Tipografia	50
A.3	Spacing	50
A.4	Textfields	51
A.5	Selectors	52
A.6	Buttons	52
A.7	Elements	53
A.8	Toast and feedbacks	54
B.1	Schermata Home (modalità scura)	55
B.2	Schermata Timbrature (modalità scura)	56
B.3	Schermata Timbrature (modalità scura / trasferta)	56
B.4	Schermata Timetable (modalità scura)	57

B.5	Schermata Impostazioni (modalità scura)	57
-----	---	----

Acronimi

ICT Information and Communication Technology

HTTPS Hypertext Transfer Protocol Secure

HR Human Resources

NFC Near Field Communication

GPS Global Positioning System

QR Quick Response

UX User Experience

UI User Interface

UCD User-Centred Design

API Application Programming Interface

REST Representational State Transfer

JSON JavaScript Object Notation

SaaS Software as a Service

IT Information Technology

WCAG Web Content Accessibility Guidelines

JWT JSON Web Token

OTP One-Time Password

PIN Personal Identification Number

IDE Integrated Development Environmen

E2E End-to-end

KPI Key Performance Indicator

OCR Optical character recognition

Capitolo 1

Introduzione

1.1 Contesto e motivazioni

Sirius è un'azienda italiana attiva nel settore ICT, con esperienza nello sviluppo di soluzioni digitali avanzate. La missione aziendale è quella di offrire ai propri clienti, strumenti tecnologici innovativi e personalizzati in grado di ottimizzare l'efficienza operativa.

All'interno di questo contesto, la gestione delle presenze rappresenta un aspetto cruciale per il funzionamento corretto delle dinamiche aziendali. Attualmente il sistema si basa su due strumenti:

- bollatrice fisica, che consente la timbratura tramite badge ai dipendenti presenti in sede;
- gestionale web interno, sviluppato in azienda, che permette di consultare e modificare le presenze, inserire ferie, permessi e note.

Questo modello è efficace per il lavoro in ufficio, ma presenta limiti nelle situazioni di smart working e in trasferta. In questi casi, infatti, i dipendenti non hanno la possibilità di effettuare una timbratura diretta e devono registrare manualmente le ore nel gestionale, con il rischio di commettere errori e riducendo l'affidabilità dei dati raccolti.

L'aumento della modalità di lavoro flessibile ha reso evidente la necessità di avere una soluzione che permetta la timbratura da remoto, garantendo sicurezza e compatibilità con il gestionale aziendale. Da qui, nasce l'idea di una applicazione mobile dedicata ai dipendenti, in grado di affiancarsi al sistema attuale e offrire un'esperienza utente semplice e immediata.

1.2 Problema di partenza

Il sistema di rilevazione presenze attualmente adottato dall'azienda, garantisce un funzionamento adeguato per i dipendenti che operano in sede, ma mostra limiti significativi in altri contesti lavorativi. In particolare, nelle situazioni di smart working o di trasferta, i dipendenti non dispongono di uno strumento per la timbratura diretta e devono inserire manualmente le ore lavorate all'interno del gestionale web.

Questa modalità di gestione comporta una serie di criticità:

- maggiore rischio di errori o dimenticanze, dovuti all'inserimento manuale dei dati;
- ridotta tracciabilità delle presenze, soprattutto nei casi di trasferta, dove i dipendenti devono anche registrare a mano i tempi di viaggio;
- perdita di efficienza, poiché il processo risulta macchinoso e richiede più tempo rispetto a una timbratura automatizzata.

La mancanza di un sistema uniforme e integrato per tutti i contesti lavorativi rappresenta quindi il problema principale che il progetto di tesi intende affrontare.

1.3 Obiettivi della tesi e del progetto

L'obiettivo principale di questo lavoro di tesi è lo sviluppo di un'applicazione mobile che consenta la timbratura da remoto per i dipendenti Sirius in smart working o in trasferta, integrandosi con il gestionale aziendale già in uso.

A partire da questa finalità generale, il progetto persegue una serie di ulteriori obiettivi:

- garantire la sicurezza e la protezione dei dati, attraverso l'utilizzo di protocolli standard (HTTPS), autenticazione biometrica e un backend che espone solo le informazioni necessarie;
- migliorare l'esperienza utente, progettando un'interfaccia chiara, intuitiva e accessibile, in grado di rendere le operazioni di timbratura rapide e immediate;
- assicurare la compatibilità con i sistemi esistenti, affiancandosi alla bollatrice fisica e al gestionale web senza sostituirli, ma integrandone le funzionalità;
- offrire una panoramica completa delle ore lavorate, tramite una dashboard che consenta di visualizzare timbrature, permessi, straordinari e note.

La tesi vuole adottare un processo di progettazione centrato sull'utente. Questo approccio mira a mettere in evidenza le esigenze dei dipendenti attraverso personas, questionari e validazioni iterative, con l'obiettivo di sviluppare uno strumento non solo funzionante ma anche utile e utilizzabile comodamente nella pratica quotidiana.

Il lavoro di tesi si concentra sullo sviluppo di un prototipo multiplatforma, accessibile su dispositivi Android e iOS.

Capitolo 2

Stato dell'arte

2.1 Sistemi di gestione presenze: evoluzione dai metodi tradizionali a quelli digitali

I sistemi di rilevazione presenze hanno seguito un'evoluzione progressiva, passando da strumenti puramente fisici a soluzioni digitali e più recentemente, a piattaforme mobile.

Questa trasformazione riflette le nuove modalità di lavoro, sempre più flessibili, e la necessità, da parte delle aziende, di garantire un monitoraggio accurato ed efficiente delle ore di lavoro.

Il modello tradizionale, basato su bollatrici fisiche e badge magnetici, rappresenta ancora oggi uno standard diffuso. Questo approccio funziona correttamente in sede, ma non è adatto a gestire attività svolte al di fuori dell'ufficio.

L'adozione diffusa di Internet, ha permesso la digitalizzazione delle presenze, consentendo l'accesso ai dati da remoto e in qualsiasi momento. In questo modo le aziende hanno potuto gestire le presenze in modo efficiente e sicuro e i dipendenti modificare i propri dati. Nonostante questo abbia migliorato la flessibilità, non risolve il problema della timbratura in modalità remota che resta affidata a inserimenti manuali.

Il nuovo livello di innovazione della rilevazione presenze è stato raggiunto con l'avvento delle applicazioni per smartphone che offrono la possibilità di timbrare da remoto in modo immediato e sicuro [1]. Oltre alla comodità, queste soluzioni permettono di integrare funzionalità aggiuntive come autenticazione biometrica, notifiche in tempo reale, geolocalizzazione e gestione delle ore lavorate, migliorando l'esperienza utente e la precisione del sistema.

Il mercato evidenzia i vantaggi dei sistemi di rilevazione automatica delle presenze, che rappresentano il principale driver di adozione di tali sistemi nelle aziende moderne:

- maggiore precisione nella registrazione dei dati, con riduzione degli errori dovuti all'inserimento manuale;
- riduzione delle frodi legate a timbrature improprie o a deleghe non autorizzate;
- automazione dei processi amministrativi, con conseguente risparmio di tempo per dipendenti e uffici HR;
- reportistica in tempo reale, che permette ai responsabili di avere una visione, aggiornata in tempo reale, delle presenze;
- integrazione con altri sistemi aziendali per semplificare il flusso dei dati.

Questi vantaggi, evidenziati anche in contributi recenti sull'argomento [2], ribadiscono la rilevanza dell'adozione di soluzioni digitali integrate.

2.2 Applicazioni mobile per gestire le presenze: analisi dei competitors

Negli ultimi anni, le applicazioni mobile per la gestione delle presenze hanno guadagnato una rilevanza sempre maggiore nel panorama aziendale HR. In questa sezione, vengono riportate alcune soluzioni commerciali significative - *Fluida* e *Factorial* - con l'intento di metterle a confronto con l'approccio progettuale adottato in questa tesi.

Fluida è una piattaforma HR mobile nata per semplificare la relazione tra lavoratore e azienda [3]. Essa integra funzioni di timbratura in smart, gestione delle ferie e dei permessi, note spese e foglio presenze con calendario, offrendo un sistema che integra funzionalità sia su dispositivi mobili che su piattaforme web.

Funzionalità rilevanti:

- Timbratura smart e geofencing: consente di confermare la presenza in remoto tramite dispositivi mobili. Inoltre, è possibile stabilire che le timbrature possano essere accettate esclusivamente in determinate aree predefinite, quali ad esempio l'abitazione del collaboratore;
- Dispositivi di timbratura hardware: prevede soluzioni fisiche come Beacon e Station. Il primo prevede l'impiego di un segnale Bluetooth per la timbratura

tramite smartphone, mentre il secondo sfrutta la tecnologia di comunicazione NFC per la gestione delle timbrature mediante l'utilizzo di un badge;

- Gestione presenze integrata: permette ai collaboratori di inserire le richieste di ferie, permessi o lavoro da casa. Attraverso l'uso del calendario è possibile visualizzare rapidamente le presenze;
- Nota spese: possibilità di raccogliere, inviare e approvare le spese sostenute in ambito lavorativo e consultare il riepilogo dall'app. È sufficiente caricare una fotografia dello scontrino per inserire i dati in modo automatico.

Limiti e considerazioni:

- Dipendenza da connettività e configurazione del geofencing: nel caso in cui il GPS non sia preciso, la timbratura potrebbe fallire e, in tale evenienza, sarebbe necessario ricorrere alla timbratura manuale;
- In contesti in cui l'uso dello smartphone non è consentito, o durante trasferite che implicano il cambiamento continuo del luogo di lavoro, diventa indispensabile ricorrere a un dispositivo hardware esterno;
- Interfaccia utente troppo articolata e poco intuitiva.

Factorial è un software HR per la gestione aziendale che permette di organizzare presenze e permessi, nonché controllare le spese [4].

Funzionalità rilevanti:

- Timbratura da mobile e desktop: gli utenti possono timbrare da applicazione mobile, con possibilità di attivare la geolocalizzazione e il geo-fencing, da web/desktop o tramite codice QR;
- Sincronizzazione con buste paga: i dati di timbratura possono integrarsi con il modulo payroll per automatizzare i calcoli;
- Gestione ferie, permessi e reportistica integrata;
- Interfaccia user-friendly, semplice, intuitiva e minimalista.

Limiti e considerazioni:

- Funzionalità offline: non è possibile timbrare offline - la connessione è un requisito per garantire l'operatività del sistema;

- Dipendenza da infrastruttura cloud: essendo un sistema SaaS, l'azienda cliente è vincolata alla piattaforma cloud del provider;
- Assenza di visualizzazione in tempo reale delle bollature effettuate.

A differenza delle soluzioni analizzate, l'applicazione sviluppata per Sirius introduce alcune funzionalità aggiuntive e una filosofia di design orientata all'utente, che la rendono più coerente con le esigenze interne dell'azienda.

L'interfaccia è progettata per essere intuitiva, permettendo di riconoscere a colpo d'occhio la modalità di lavoro corrente - smart working, ufficio o trasferta - attraverso un sistema di colori, icone e testi.

Una delle innovazioni più rilevanti è la possibilità di monitorare in tempo reale l'ultima timbratura eseguita, insieme a tutte le altre timbrature della giornata. In questo modo l'utente può consultare in ogni momento le ore lavorate aggiornate, e comprendere immediatamente se si trova in servizio, in pausa o in orario straordinario.

Queste scelte progettuali mirano a offrire un'esperienza utente più naturale e immediata rispetto ai competitors, in cui la grande quantità di funzioni tende spesso a ridurre la leggibilità, la semplicità d'uso e ad appesantire il carico cognitivo.

2.3 Architetture client-server per sistemi mobile

Le applicazioni mobile moderne si basano su un'architettura di tipo client-server, un modello di comunicazione che consente di separare un'interfaccia utente con la logica di presentazione (client) dalla gestione dei dati e delle operazioni di business (server). Questa logica di “*richiesta-risposta*” è alla base del funzionamento della maggior parte dei siti web e delle applicazioni.

Nel contesto di un sistema di rilevazione presenze, questa architettura risulta particolarmente efficace, poiché permette di garantire sicurezza, scalabilità e coerenza dei dati tra diversi dispositivi utilizzati dai dipendenti.

Nel modello client-server, l'applicazione mobile opera come client, inoltrando richieste al server centrale il quale si occupa della gestione delle informazioni aziendali, come i dati degli utenti, le timbrature e le note.

Il server a sua volta elabora la richiesta, accede al database e restituisce la risposta al client in un formato leggibile e standardizzato.

La comunicazione tra client e server avviene tramite API REST, uno standard ampiamente utilizzato per lo sviluppo di applicazioni distribuite. Nel caso specifico di questo progetto, le API vengono esposte attraverso il protocollo HTTPS, che garantisce la trasmissione cifrata dei dati e previene accessi non autorizzati.

Le informazioni scambiate sono codificate in formato JSON, leggero e facilmente interpretabile sia dal server che dal dispositivo mobile.

A questo primo livello di protezione si affiancano meccanismi di autenticazione sicura che comprendono l'uso di token di accesso e autenticazione biometrica per garantire che solo gli utenti autorizzati possano accedere alle funzioni di timbratura e consultazione dei dati.

Inoltre, l'app non interagisce direttamente con il gestionale aziendale, ma utilizza un backend dedicato, che espone esclusivamente le informazioni strettamente necessarie. Questo approccio riduce il rischio di vulnerabilità e consente un controllo più preciso sulle operazioni effettuate dal client.

Questo tipo di sistema si potrebbe quindi dividere in tre livelli principali:

- Client (app mobile): interfaccia utente per l'interazione diretta con il sistema;
- Application server (backend dedicato): gestisce la business logic e funge da intermediario con il gestionale aziendale;
- Database: conserva in modo sicuro le informazioni sulle presenze e sincronizza i dati tra i vari sistemi.

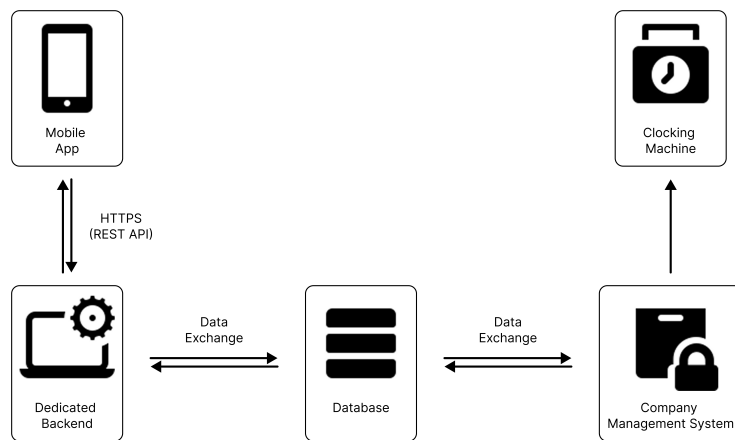


Figura 2.1: Architettura client-server

Capitolo 3

Metodologia di ricerca e analisi preliminare

3.1 Approccio metodologico

Il processo di sviluppo dell'applicazione mobile per la timbratura si è articolato in una serie di fasi, definite e verificate in collaborazione con l'azienda.

Il lavoro ha seguito un approccio iterativo e focalizzato sull'utente, con l'obiettivo di rispondere alle necessità operative reali dei dipendenti.

La fase iniziale ha previsto un'analisi preliminare delle esigenze aziendali, a partire dai requisiti generali individuati in collaborazione con il dipartimento IT. Successivamente è stata organizzata una riunione di confronto con il responsabile del reparto di ricerca e sviluppo e con il direttore tecnico, allo scopo di validare gli obiettivi iniziali e definire in modo più dettagliato i requisiti funzionali e non funzionali dell'applicazione.

Dalla definizione dei requisiti si è passati alla fase di progettazione concettuale, durante la quale sono stati realizzati i primi wireframe per individuare la struttura delle schermate principali e la disposizione delle funzionalità. Per questa attività sono stati impiegati strumenti di progettazione digitale quali Figma, utilizzato per la realizzazione degli sketch e dei prototipi, e Miro, impiegato per la realizzazione della mappa dei flussi utente e la rappresentazione visiva dello user flow.

La successiva fase di validazione ha previsto un confronto con il team aziendale, durante il quale sono emerse osservazioni utili per ottimizzare la chiarezza dell'interfaccia e per individuare con maggiore precisione i bisogni reali e le informazioni più rilevanti per gli utenti.

L'approccio può essere quindi suddiviso nelle seguenti fasi principali:

- analisi iniziale e raccolta dei requisiti;
- confronto con l'azienda e definizione delle priorità progettuali;
- progettazione tramite wireframe e user flow;
- revisione e validazione attraverso feedback.

3.2 Raccolta dei requisiti: obiettivi dell'azienda

La definizione dei requisiti consente di tradurre le esigenze aziendali in specifiche tecniche. Nel caso di questo progetto, i requisiti sono stati identificati attraverso tre attività:

- la proposta iniziale di tesi, che ha fornito gli obiettivi generali del progetto e una prima definizione delle funzionalità desiderate;
- la riunione tecnica con il reparto di ricerca e sviluppo e il direttore tecnico, nella quale sono stati chiariti i vincoli di integrazione con il gestionale interno;
- l'analisi dell'attuale sistema di rilevazione presenze, utile a comprendere le limitazioni esistenti.

Dalle attività di analisi e confronto sono emersi una serie di requisiti funzionali e non funzionali che hanno guidato lo sviluppo dell'applicazione. I requisiti funzionali definiscono le operazioni che l'app deve permettere all'utente di compiere, quelli non funzionali riguardano le caratteristiche di qualità del sistema, come prestazioni, sicurezza e usabilità.

Tipo	Codice	Descrizione	Priorità
Funzionale	F1	Permettere la timbratura da remoto per utenti in smart o in trasferta	Alta
Funzionale	F2	Visualizzare le ore lavorate giornaliere e lo stato operativo (ufficio, smart, trasferta)	Media
Funzionale	F3	Integrare la timbratura remota con il sistema gestionale aziendale esistente	Alta
Non funzionale	NF1	Garantire la sicurezza dei dati tramite protocollo HTTPS e autenticazione biometrica	Alta
Non funzionale	NF2	Assicurare accesso riservato esclusivamente ai dipendenti Sirius	Media
Non funzionale	NF3	Progettare un'interfaccia semplice, chiara e intuitiva, adatta a diversi profili di utenza	Alta
Non funzionale	NF4	Garantire la compatibilità multiplatforma (Android e iOS)	Media
Non funzionale	NF5	Mantenere la compatibilità con la bollatrice fisica già in uso	Alta

Tabella 3.1: Requisiti funzionali e non funzionali

3.3 Analisi degli utenti e definizione delle personas

L'analisi degli utenti permette di comprendere i bisogni, le aspettative e i comportamenti degli utilizzatori finali del sistema.

Nel contesto del progetto di tesi, l'obiettivo di questa fase è stato quello di

individuare i profili professionali predominanti tra i dipendenti destinatari dell'applicazione mobile, così da orientare la progettazione verso soluzioni realmente allineate alle modalità operative quotidiane.

Il target di riferimento è costituito dai dipendenti Sirius, i quali operano in contesti lavorativi differenti ma condividono la necessità di registrare le proprie presenze in modo preciso e immediato.

In particolare si distinguono tre categorie principali:

- dipendenti in sede, che utilizzano la bollatrice fisica e il gestionale web ma hanno bisogno di un modo veloce per consultare e monitorare le ore lavorate;
- dipendenti in smart working, che richiedono la possibilità di effettuare la timbratura da remoto in modo sicuro e immediato;
- dipendenti in trasferta, per i quali risulta fondamentale registrare le presenze e i tempi di viaggio anche al di fuori dell'ambiente aziendale.

Per rappresentare in modo sintetico e visivo i diversi tipi di utenti, sono state definite tre personas, ovvero modelli di riferimento che riassumono comportamenti, obiettivi e necessità comuni a gruppi reali di lavoratori.

Persona 1 - Martina (impiegata amministrativa)

- **Età:** 32 anni;
- **Ruolo:** impiegata amministrativa;
- **Modalità di lavoro:** prevalentemente smart working;
- **Dispositivo:** laptop aziendale e smartphone personale;
- **Obiettivi:** timbrare rapidamente da casa, controllare le ore giornaliere e lo stato lavorativo;
- **Complicazioni:** dimentica spesso di inserire manualmente le ore sul gestionale web;
- **Bisogni:** disporre di un'app semplice che consenta di registrare correttamente le presenze in smart working.

Persona 2 - Gabriele (tecnico)

- **Età:** 40 anni;

- **Ruolo:** tecnico che svolge attività e interventi presso clienti;
- **Modalità di lavoro:** principalmente trasferta;
- **Dispositivo:** laptop e smartphone aziendale;
- **Obiettivi:** registrare le timbrature in mobilità, segnare le ore di viaggio e avere la conferma che le bollature siano registrate correttamente;
- **Complicazioni:** l'inserimento manuale delle ore con la conseguenza di non avere visibilità immediata dei dati giornalieri;
- **Bisogni:** sistema rapido con possibilità di timbrare e aggiornare lo stato di lavoro.

Persona 3 - Luca (sviluppatore software)

- **Età:** 23 anni;
- **Ruolo:** sviluppatore software;
- **Modalità di lavoro:** principalmente ufficio;
- **Dispositivo:** laptop aziendale e smartphone personale;
- **Obiettivi:** accedere velocemente alle proprie timbrature;
- **Complicazioni:** l'impossibilità di visualizzare le ore lavorate durante la pausa;
- **Bisogni:** interfaccia intuitiva, ottimizzata per smartphone e integrata con l'autenticazione aziendale.

Le personas individuate hanno rappresentato la base per la definizione degli scenari d'uso e dei flussi di interazione dell'applicazione. Attraverso la loro analisi è stato possibile comprendere come differenziare la visualizzazione e le funzioni in base al contesto lavorativo, ponendo particolare attenzione alla chiarezza dell'interfaccia, alla rapidità dell'operazione di timbratura e alla visibilità immediata dello stato lavorativo.

USER PERSONAS



Figura 3.1: User Personas

3.4 Questionario agli utenti: metodologia e campione

Per approfondire le esigenze dei dipendenti Sirius e comprendere le principali criticità legate al sistema di timbratura attualmente in uso, è stato somministrato un questionario interno finalizzato a raccogliere dati qualitativi e quantitativi sulle aspettative verso una nuova applicazione mobile e sulle abitudini di utilizzo dell'attuale gestionale web.

L'obiettivo primario di questa indagine è stato quello di raccogliere feedback diretti da parte degli utilizzatori del sistema, per identificare le funzionalità più rilevanti da implementare nella nuova applicazione mobile e le criticità riscontrate nel flusso di timbratura e consultazione delle presenze.

Il questionario è stato progettato per:

- comprendere quali informazioni visualizzare nella schermata principale dell'app;
- valutare l'interesse verso funzioni aggiuntive (note, calendario, riepiloghi mensili, ecc.);
- analizzare le modalità di gestione delle trasferte e individuare i principali bisogni legati a questa attività;

- raccogliere suggerimenti qualitativi per la progettazione della nuova interfaccia.

Il questionario è stato realizzato tramite la piattaforma Microsoft Forms e distribuito a tutti i dipendenti e responsabili Sirius, così da raccogliere un punto di vista rappresentativo di tutte le aree aziendali.

Le domande sono state formulate in modo semplice, alternando quesiti a risposta multipla e domande aperte.

Nel complesso il questionario ha ottenuto 38 risposte su un totale di 50 dipendenti, garantendo una copertura quasi totale dell'organico aziendale. La compilazione è avvenuta online, in un tempo medio di circa 5 minuti.

Il modulo era composto da 10 domande, suddivise in tre aree tematiche principali:

1. Schermata principale dell'applicazione:

- (a) Quali tra queste informazioni vorresti visualizzare nella home?

(Allo scopo di definire la priorità visiva dei contenuti e comprendere le reali necessità operative degli utenti)

2. Funzionalità aggiuntive:

- (a) Sarebbe utile avere il campo note nella pagina delle timbrature anche da mobile?

- (b) Sarebbe utile avere un calendario con il riepilogo del mese?

- (c) Avendo la possibilità di aggiungere un widget nella pagina calendario, cosa ti piacerebbe visualizzare?

(Allo scopo di valutare l'interesse verso strumenti di annotazione e riepilogo personale)

3. Gestione delle trasferte:

- (a) Sei mai stato in trasferta?

- (b) Come comunichi di solito il tempo di viaggio?

- (c) Se potessi registrare il tempo di viaggio direttamente da un'app, come ti piacerebbe farlo?

- (d) In che altri modi l'app del gestionale ti potrebbe aiutare in trasferta?

(Allo scopo di ridurre i limiti del sistema attuale e sfruttare le potenzialità di una gestione mobile)

Infine, due domande aperte hanno raccolto suggerimenti liberi per migliorare la UX e introdurre eventuali nuove funzionalità non ancora previste.

3.5 Analisi dei feedback qualitativi e quantitativi

I dati raccolti hanno offerto una visione chiara delle priorità e delle difficoltà percepite dai dipendenti nell'utilizzo di un sistema di timbratura.

La maggior parte degli utenti ha espresso il desiderio di poter visualizzare in modo immediato l'ultima timbratura e le ore lavorate giornaliere, seguiti da un forte interesse verso un riepilogo mensile e la possibilità di inserire note direttamente da app.

Le sezioni dedicate alla gestione delle trasferte hanno inoltre evidenziato la necessità di automatizzare il tempo di viaggio, funzione che attualmente viene svolta in modo manuale o comunicata verbalmente.

Alla domande "Quali tra queste informazioni vorresti visualizzare nella home?", la maggioranza dei partecipanti ha indicato:

- l'ultima timbratura effettuata (25 voti) come elemento prioritario, da mostrare in modo chiaro e immediato;
- il totale delle ore giornaliere lavorate (22 voti) come informazioni secondarie ma essenziali;
- l'orario di uscita prevista (29 voti) come elemento aggiuntivo.

Quali tra queste informazioni vorresti visualizzare nella home?

- Ultima timbratura (25)
- Ore lavorate (22)
- Uscita prevista (29)
- Meteo (10)
- Percentuale di smart (17)

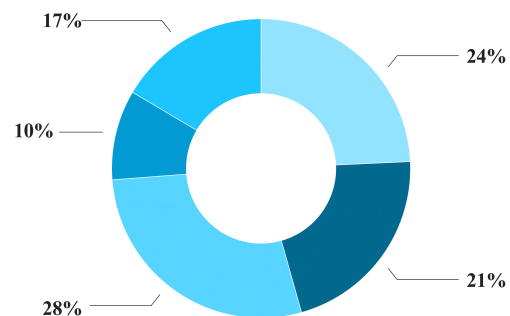


Figura 3.2: Risultati in merito alla domanda sulle informazioni presenti nella home

Dalle risposte aperte emerge un'esigenza trasversale: avere una dashboard compatta e leggibile che consenta, a colpo d'occhio, di capire la situazione lavorativa della giornata.

Per quanto riguarda l'aggiunta di funzionalità extra:

- il campo note risulta particolarmente apprezzato, con l'87% degli utenti che lo considera utile per annotare motivazioni o commenti legati alla timbratura;
- l'inserimento di un calendario mensile per la visualizzazione di anomalie, assenze e ore lavorate è stato ritenuto molto utile dal 76% dei dipendenti.

Le risposte confermano l'interesse verso una gestione integrata che unisca timbrature, note e riepiloghi in un unico strumento coerente.

Avendo la possibilità di aggiungere un widget nella pagina calendario, cosa ti piacerebbe visualizzare?

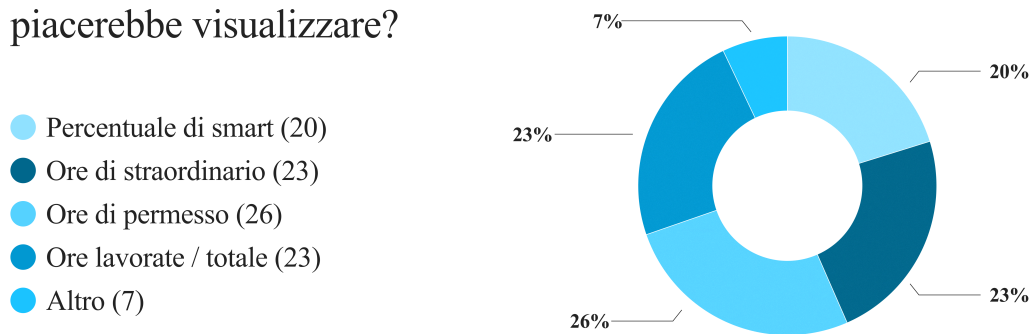


Figura 3.3: Risultati in merito alla domanda su un widget personalizzato

Alla domanda "Effettui trasferte per motivi lavorativi?", il 58% degli intervistati ha risposto affermativamente. Tuttavia la gestione delle trasferte risulta eterogenea:

- il 41% dichiara di inserire manualmente le ore di viaggio nel gestionale web;
- il 17% si affida ancora a comunicazioni verbali o via e-mail.

Da questi dati emerge una chiara opportunità di miglioramento: l'introduzione di una funzionalità dedicata alle trasferte che consenta la timbratura remota e l'indicazione automatica del tempo di viaggio.

In generale, i risultati del questionario confermano che i dipendenti Sirius attribuiscono grande importanza alla semplicità d'uso, alla chiarezza dell'interfaccia e alla trasparenza delle informazioni relative alle proprie timbrature.

Il bisogno di immediatezza e controllo personale rappresenta il punto di partenza per lo sviluppo del prototipo mobile, che dovrà ridurre al minimo le azioni necessarie per timbrare, visualizzare o modificare i dati di partenza.

Aspetto analizzato	Risultato	Implicazioni
Schermata principale	L'ultima timbratura e il totale ore giornaliere sono considerate le informazioni più rilevanti	La home deve evidenziare lo stato corrente e l'orario dell'ultima timbratura
Dashboard	Gli utenti desiderano una visione chiara delle timbrature giornaliere e settimanali	Inserimento di una sezione dedicata alla cronologia giornaliera e calendario mensile
Campo note	Considerato utile dall'87% dei partecipanti per commenti e motivazioni	Aggiunta di un campo testo opzionale collegato a ogni timbratura
Gestione trasferite	Il 58% dei dipendenti effettua trasferte, la gestione attuale è manuale e disomogenea	Prevedere una modalità "trasferta" in cui scrivere il tempo di viaggio
Usabilità dell'interfaccia	Gli utenti chiedono semplicità e rapidità d'uso	Struttura minimale, icone e colori distintivi
Feedback del sistema	Richieste di notifiche e conferme in tempo reale	Implementazione di messaggi visivi e toast di conferma per ogni azione critica
Accesso e sicurezza	Interesse verso autenticazione automatica o biometrica	Integrazione di accesso con credenziali aziendali e autenticazione biometrica

Tabella 3.2: Risultati e implicazioni del questionario

Capitolo 4

Progettazione e prototipazione

4.1 Modello di interazione e user flow

Il modello di interazione definisce il modo in cui l'utente entra in relazione con il sistema e rappresenta la base logica e funzionale dell'intera applicazione. Nel caso di questa tesi, si è voluto realizzare un flusso di navigazione chiaro, coerente e facilmente comprensibile.

Le linee guida di Material Design di Google sono state usate come riferimento per la definizione delle gerarchie visive e per la coerenza tra le diverse schermate [5].

Il flusso può essere descritto attraverso le seguenti fasi principali:

1. Accesso e autenticazione:

- (a) l'utente effettua il login utilizzando le stesse credenziali del gestionale aziendale (nel caso di credenziali non corrette compare un popup di errore);
- (b) per motivi di sicurezza, viene richiesto un codice OTP per completare la verifica dell'identità;
- (c) al primo accesso, l'utente imposta un PIN personale per semplificare gli accessi successivi. È possibile poi abilitare l'autenticazione biometrica.

2. Schermata Home:

- (a) rappresenta il centro dell'esperienza utente, offrendo una overview sintetica. È possibile visualizzare l'ultima timbratura effettuata, le ore lavorate giornaliere e la modalità di lavoro (ufficio, trasferta, smart working);

- (b) la parte inferiore della schermata include uno swipe interattivo per effettuare la timbratura;
- (c) dopo ogni azione di timbratura, il sistema fornisce un feedback visivo immediato tramite toast message, che conferma l'avvenuta registrazione o segnala eventuali errori;
- (d) sono presenti alcuni widget come il campo note, la percentuale di smart e il riepilogo del carico di lavoro.

3. Schermata Timbrature:

- (a) accessibile dalla home o dal calendario, consente di visualizzare le timbrature della giornata corrente.

4. Schermata Timetable:

- (a) permette di accedere alla cronologia delle presenze tramite un calendario mensile, evidenziando eventuali anomalie. La percentuale di smart working, le ore di straordinario e le ore di permesso sono visibili in un widget dedicato;
- (b) selezionando una determinata giornata, l'utente ha la possibilità di consultare i dettagli relativi alle timbrature registrate in quella data.

5. Schermata Impostazioni:

- (a) include le funzionalità di personalizzazione e gestione dell'app tra cui il riepilogo dei dati personali, la possibilità di abilitare o disabilitare l'uso della biometria, la selezione del tema e la scelta della lingua.

L'interfaccia risulta omogenea per tutti gli utenti, indipendentemente dal ruolo aziendale.

La struttura del flusso è stata progettata per ridurre al minimo i passaggi necessari, eliminando azioni ridondanti e mantenendo sempre visibili le funzioni principali.

La navigazione segue una logica bidirezionale e coerente. Ogni sezione consente di ritornare facilmente alla Home, senza interruzioni di flusso.

4.2 Wireframe e sketch iniziali

La fase di progettazione visiva è iniziata con la realizzazione dei wireframe, ossia rappresentazioni schematiche delle principali schermate dell'applicazione.

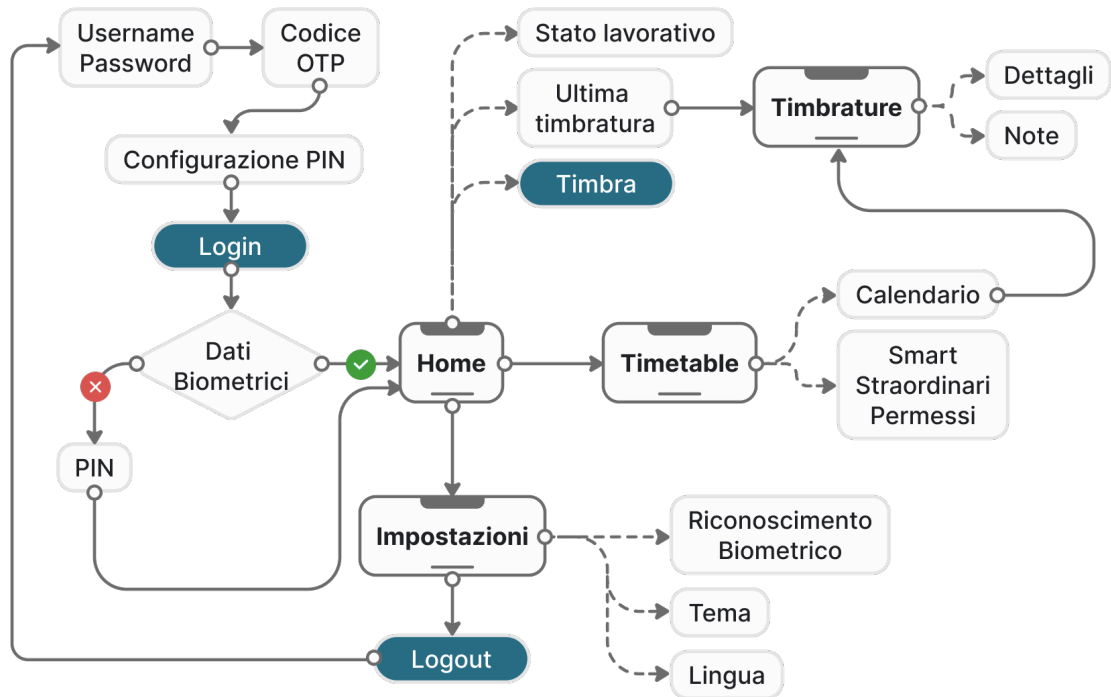


Figura 4.1: User Flow

Questa fase si è concentrata sulla progettazione della struttura di base dell'interfaccia, mirata a identificare la disposizione ottimale dei contenuti e degli elementi interattivi.

Gli sketch sono stati realizzati con Figma, uno strumento di progettazione che ha permesso di sviluppare in modo rapido le diverse soluzioni. In questa fase sono state rappresentate le schermate fondamentali del flusso utente.

I wireframe sono stati presentati al team aziendale durante una sessione di revisione, in cui sono stati valutati sia dal punto di vista funzionale che in termini di esperienza utente.

Dalle osservazioni emerse non sono state richieste modifiche sostanziali alla struttura: la proposta iniziale è rimasta coerente con le esigenze operative e con i flussi di utilizzo identificati.

Il mantenimento, quasi invariato, dei wireframe rispetto alla versione definitiva dimostra l'efficacia del processo di analisi preliminare e la solidità del modello concettuale scelto.



Figura 4.2: Wireframe Home



Figura 4.3: Wireframe Timbrature

4.3 UI kit: colori, tipografia, componenti principali

Per garantire coerenza visiva e uniformità grafica in tutte le schermate, è stato realizzato un UI Kit, ovvero un insieme di componenti e linee guida stilistiche che costituiscono la base del design dell'applicazione.

4.3.1 Colori

La palette cromatica è stata definita ispirandosi all'identità visiva aziendale Sirius, utilizzando tonalità coerenti con il contesto professionale, ma differenziate in base alla modalità di lavoro.

Ogni colore principale è stato associato a uno specifico stato operativo dell'utente:

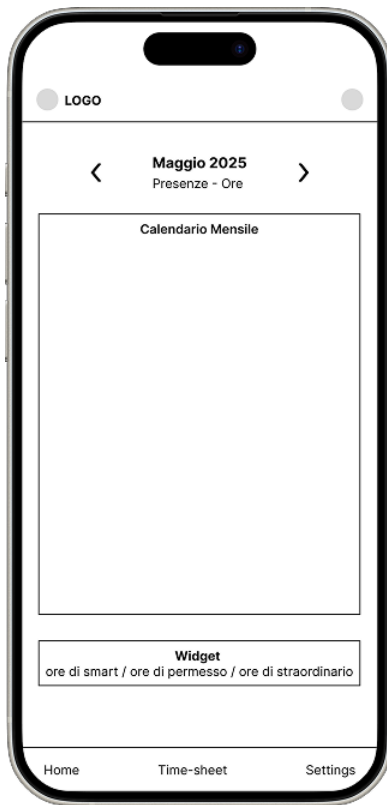


Figura 4.4: Wireframe Timesheet



Figura 4.5: Wireframe Impostazioni

- Cerulean (#286C84) - colore primario e modalità *ufficio*, corrispondente al colore istituzionale dell'azienda;
- Savoy blue (#6065C5) - modalità smart working;
- Pigment green (#419D3A) - modalità trasferta;
- Platinum (#E4E4E4) - bordi ed elementi di separazione;
- White (#FBFBFB) - colore principale delle card;
- Alabaster (#EBEBDF) - background delle schermate;
- Battleship gray (#989898) - pulsanti disabilitati;
- Eerie black (#222222) - testo primario;
- Indian red (#D9534F) - errore;

- Sandy brown (#F4A261) - warning;
- Pigment green (#419D3A) - success;

È stata implementata anche una dark mode, mantenendo la coerenza cromatica della versione chiara:

- Process cyan (#01BAEF) - colore primario e modalità ufficio;
- Tropical indigo (#7D83FF) - modalità smart working;
- Dark pastel green (#4CB944) - modalità trasferta;
- Delft blue (#2A3750) - bordi ed elementi di separazione;
- Raisin black (#181F2D) - colore principale delle card;
- Rich black (#040B1A) - background delle schermate;
- Dim gray (#666666) - pulsanti disabilitati;
- Baby powder (#FCFDF5) - testo primario;
- Light coral (#F27370) - errore;
- Sandy brown (#F4A261) - warning;
- Dark pastel green (#4CB944) - success;

L'utilizzo del colore segue la logica funzionale: a ciascuna modalità è stato attribuito un colore specifico, permettendo di individuare immediatamente lo stato del dipendente.

4.3.2 Tipografia

La tipografia scelta si ispira ai principi di Material Design, adottando i font di sistema predefiniti delle piattaforme su cui opera: San Francisco per iOS e Roboto per Android. Il titolo principale dell'app utilizza il font Vevey, selezionato per la personalità distintiva, che consente di dare riconoscibilità al brand e rafforzare l'identità visiva del prodotto.

4.3.3 Nome e identità visiva dell'applicazione

L'applicazione è stata denominata *ALYA*, nome ispirato a una stella situata nella costellazione del Serpente. Questa scelta riflette l'obiettivo di preservare una coerenza semantica con il nome dell'azienda, Sirius, anch'esso legato a un corpo celeste.

Il logo di *ALYA* rappresenta una stella stilizzata, realizzata mediante linee geometriche.

Dal punto di vista grafico, il logo adotta un design minimalista e un monocromo neutro, in linea con lo stile dell'interfaccia dell'applicazione.

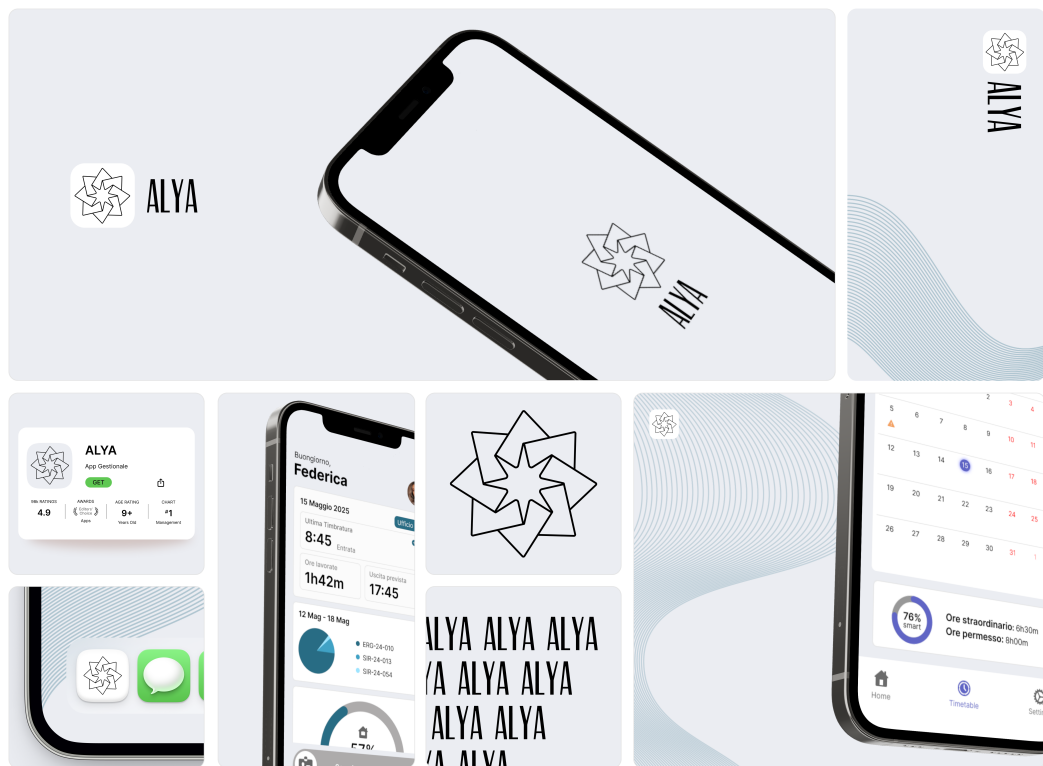


Figura 4.6: Identità visiva dell'app

4.3.4 Componenti principali

Il sistema di componenti comprende gli elementi ricorrenti dell'interfaccia utente:

- Pulsanti (button): principali, secondari e disabilitati con variazioni cromatiche tra modalità chiara e scura;

- Card: contenitori informativi per overview e widget;
- Widget: moduli dinamici che mostrano informazioni come note o carico di lavoro;
- Barra di navigazione: presente in tutte le schermate, permette un accesso rapido alle altre sezioni;
- Textfields: input di testo con annesso label;
- Selettori: dropdown che permette di scegliere un'opzione, toggle per attivare/-disattivare una modalità;
- Altri elementi: swipe button per la timbratura, progress bar/circle per mostrare una percentuale nei widget, pie charts per rappresentare il carico di lavoro e l'avatar;
- Toast e messaggi di feedback: notifiche brevi e temporanee che confermano azioni o segnalano errori.

L'intero set di componenti e varianti grafiche è riportato in Appendice A, dove sono illustrate le principali interfacce e i relativi stati di interazione, sia per la modalità chiara che per quella scura.

4.4 Layout e design delle schermate principali

Il layout dell'applicazione è stato organizzato seguendo una struttura modulare in cui gli elementi sono disposti in base a criteri di priorità e frequenza di utilizzo. Questa organizzazione consente di ridurre al minimo i passaggi necessari per accedere alle informazioni principali e completare le azioni più frequenti.

Il design utilizza una combinazione di colori, spaziature e tipografie pensata per ridurre il carico cognitivo e favorire una navigazione fluida.

Le schermate sono state realizzate in modalità chiara e scura per adattarsi alle preferenze degli utenti e migliorare la leggibilità in diversi contesti di utilizzo. Entrambe le modalità mantengono la stessa gerarchia visiva e palette cromatica adattandosi alle preferenze dell'utente.

Di seguito vengono presentate le schermate principali in modalità chiara, le varianti grafiche sono consultabili in Appendice B.

4.4.1 Home

La schermata Home rappresenta il punto di accesso principale dell'applicazione. Essa mostra:

- l'ultima timbratura effettuata (orario e tipologia di timbratura);
- ore lavorate durante la giornata;
- modalità di lavoro corrente;
- widget con carico di lavoro;
- widget con percentuale di smart;
- widget con note;

L'azione di timbratura con swipe è posizionata in basso per favorire l'interazione con una sola mano durante l'utilizzo su dispositivi mobile.

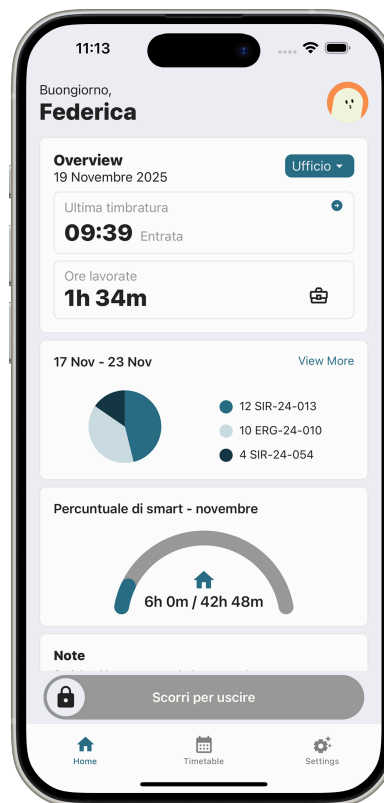


Figura 4.7: Schermata Home (modalità chiara)

Il widget del carico di lavoro permette all'utente di visualizzare l'avanzamento settimanale sulle commesse assegnate, con la possibilità di espandere il contenuto attraverso il pulsante *View More*, che apre un pannello informativo nella parte inferiore dello schermo. Questo pannello contiene l'elenco completo delle commesse assegnate all'utente, insieme alle ore lavorate su ciascuna di esse.

Il widget della percentuale di smart working mostra l'incidenza delle ore svolte in modalità smart sul totale mensile.

Infine, il widget delle note consente all'utente di annotare informazioni relative alla giornata; il testo inserito viene associato alla timbratura successiva.

4.4.2 Timbrature

La sezione timbrature elenca le registrazioni della giornata corrente. Ogni timbratura è visualizzata come elemento di una tabella contenente anche la tipologia di timbratura e l'orario. Questa sezione è accessibile dalla overview nella pagina Home.

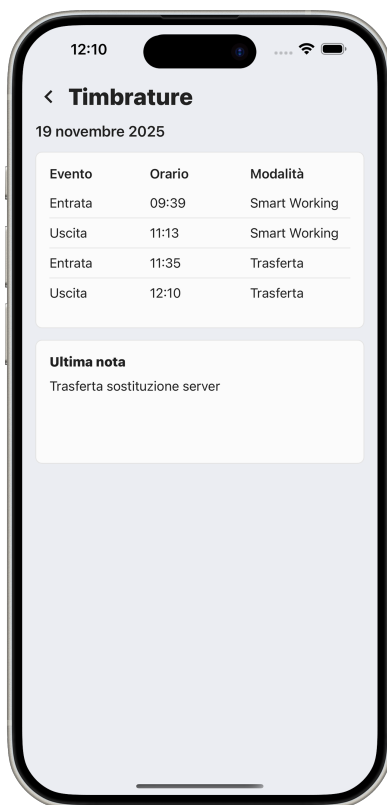


Figura 4.8: Schermata Timbrature (modalità chiara)

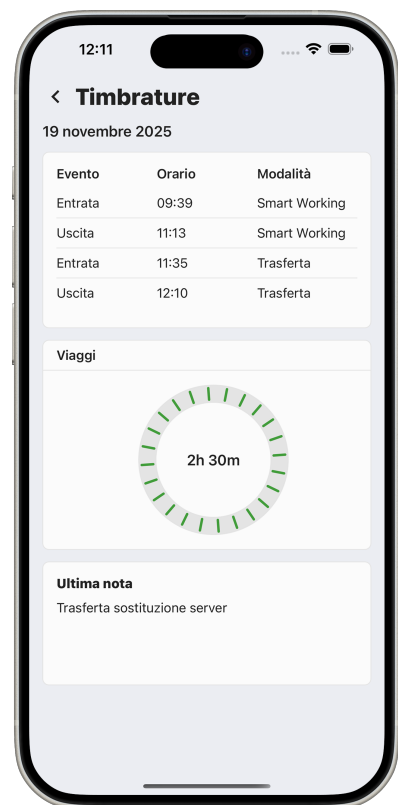


Figura 4.9: Schermata Timbrature (modalità chiara / trasferta)

Nel caso in cui l'utente si trovi in modalità trasferta, è possibile registrare il tempo di viaggio tramite uno spinner.

4.4.3 Timetable

La schermata Timetable fornisce una panoramica mensile delle presenze tramite calendario interattivo, evidenziando anomalie nella giornata. È inoltre presente un widget con percentuale di smart working, ore di straordinario e ore di permessi. Cliccando su un singolo giorno si può accedere alla pagina Timbrature del giorno selezionato.

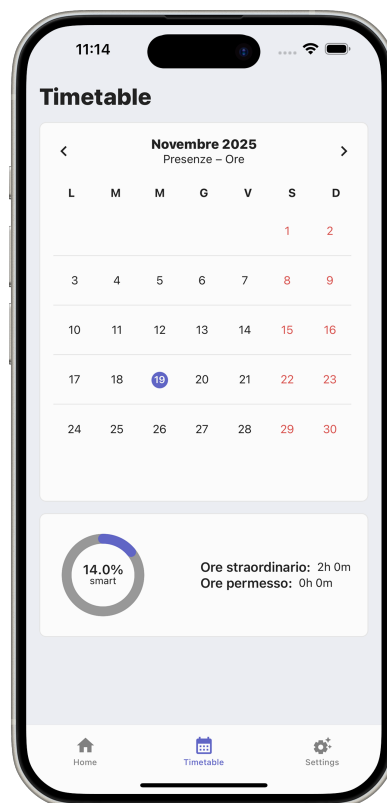


Figura 4.10: Schermata Timetable (modalità chiara)

4.4.4 Impostazioni

Questa sezione raccoglie le funzioni di personalizzazione e configurazione:

- riepilogo del profilo utente;
- abilitazione/disabilitazione del riconoscimento biometrico;

- selezione lingua tra italiano e inglese;
- selezione tema chiaro/scuro.

Da questa sezione si può anche effettuare il Logout, che reindirizza direttamente alla schermata iniziale di accesso.

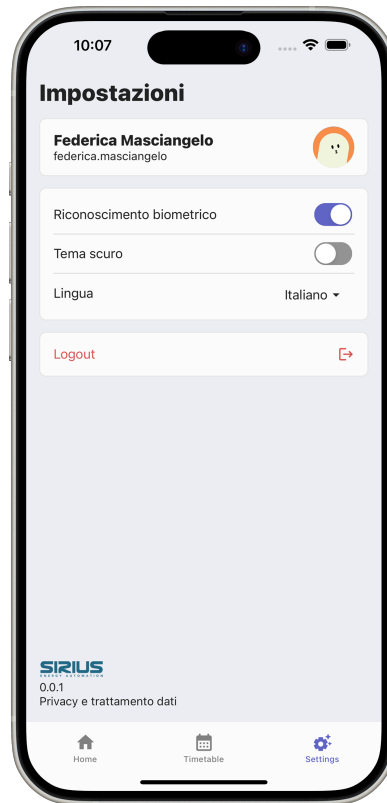


Figura 4.11: Schermata Impostazioni (modalità chiara)

4.5 Accessibilità e principi di usabilità adottati

Durante lo sviluppo dell'applicazione è stata posta particolare attenzione alle linee guida di accessibilità e usabilità, indipendentemente dal contesto di utilizzo o dal dispositivo.

Le scelte si basano sulle linee guida di Material Design [5] e sui principi delle Web Content Accessibility Guidelines (WCAG 2.1) [6], che definiscono criteri di leggibilità, contrasto e interazione.

L'interfaccia segue i principi fondamentali di usabilità definiti da Jakob Nielsen [7], adattandoli al contesto aziendale:

- Visibilità dello stato del sistema: dopo ogni timbratura effettuata viene mostrato un toast di conferma o errore;
- Corrispondenza tra il sistema e il mondo reale: le etichette e i testi utilizzano un linguaggio naturale, coerente con il lessico aziendale;
- Controllo e libertà dell'utente: l'utente può tornare alla schermata precedente in qualsiasi momento o annullare un'azione;
- Coerenza e standard: gli elementi interattivi mantengono lo stesso comportamento in ogni sezione dell'app;
- Prevenzione degli errori: operazioni critiche, come la timbratura, richiedono un gesto intenzionale (swipe) per evitare azioni accidentali.

Il design è stato validato utilizzando il plugin *Use Contrast* di Henrique Gusso su Figma [8], che permette di verificare i livelli di contrasto tra testo e sfondo secondo gli standard WCAG.

I risultati hanno evidenziato che tutti i componenti principali rispettano almeno il livello AA (rapporto di contrasto $> 4.5:1$), con valori medi rilevati pari a AA 5, garantendo buona leggibilità.

Capitolo 5

Implementazione e sviluppo

5.1 Tecnologie utilizzate

L'applicazione è stata sviluppata tramite l'utilizzo di React Native [9] e del framework Expo SDK 54 [10], il quale presenta numerosi vantaggi. Oltre a facilitare la gestione multiplatforma e a garantire un'ampia gamma di librerie e strumenti integrati, consente oggi di ottenere build significativamente più rapide grazie alla precompilazione per React Native su iOS e a un sistema di autolinking più avanzato ed efficiente.

L'ambiente di sviluppo è stato interamente configurato utilizzando TypeScript [11], che ha garantito una maggiore robustezza del codice e una gestione più precisa dei tipi in fase di compilazione. Per la gestione del versionamento è stato impiegato Git, con repository privato, mentre l'IDE principale è stato PHP Storm [12].

5.1.1 Librerie principali

Nel corso dello sviluppo sono state impiegate diverse librerie e moduli open-source per gestire in modo efficiente le principali funzionalità dell'applicazione:

- Navigazione: realizzata tramite **@react-navigation** [13] (stack e bottom-tabs) per garantire una navigazione fluida e coerente tra le schermate principali dell'applicazione;
- Comunicazione con il backend: implementata mediante l'uso di Axios [14], con l'impiego di chiamate REST su protocollo HTTPS verso il backend aziendale dedicato;

- Autenticazione e sicurezza: gestita attraverso `@expo-secure-store` [15] per il salvataggio locale dei token di accesso e `@expo-local-authentication` [16] per l'integrazione con la biometria (Face ID / Touch ID);
- Localizzazione: realizzata con `i18n-js` [17] ed `@expo-localization` [18], due librerie che consentono la traduzione dinamica dei testi e l'adattamento automatico della lingua in base alle impostazioni del dispositivo;
- Feedback visivi e interazione: la libreria `@react-native-toast-message` [19] è stata utilizzata per la visualizzazione di messaggi di conferma o errore, mentre `@rn-swipe-button` [20] consente la timbratura tramite gesto di swipe, garantendo maggiore intenzionalità nell'azione;
- Visualizzazione grafica dei dati: `@react-native-chart-kit` [21] è stato utilizzato per la rappresentazione di riepiloghi, come il carico di lavoro;
- Debug e monitoraggio: durante la fase di sviluppo sono stati impiegati Reactotron [22] e gli strumenti di debug di Expo per l'analisi dei log e la risoluzione degli errori.

5.1.2 Test e ambienti di esecuzione

L'applicazione è stata testata sia su emulatori Android e iOS forniti da Expo e Xcode [23], sia su dispositivi reali per verificare il corretto funzionamento in condizioni operative.

5.2 Architettura Software

L'architettura software dell'applicazione è stata progettata adottando un approccio modulare e scalabile per garantire una chiara distinzione delle responsabilità, promuovere la riusabilità dei componenti e assicurare una gestione sicura dei dati.

La logica è stata strutturata in modo tale da agevolare la leggibilità e la manutenzione del codice. Le directory principali sono:

- `/core` - contiene moduli di servizio, helpers e tipizzazioni, in particolare la gestione sicura dei dati con SecureStore e i servizi di interfaccia verso le API tra cui:
 - `api.ts`: configurazione dell'istanza Axios per la comunicazione con il backend, la gestione token e il refresh automatico;
 - `secureStore.ts`: gestione sicura dei dati tramite `@expo-secure-store`;

- `types.ts`: definizione tipi e interfacce comuni.
- `/locale` - include i file di localizzazione e traduzione multilingue;
- `/navigation` - definisce la struttura di navigazione, comprende `AppNavigator` e `BottomTabNavigator`;
- `/screens` - raccoglie le schermate principali e i relativi componenti interni;
- `/theme` - contiene la definizione centralizzata di colori, spaziature e tipografie, garantendo coerenza visiva.

Tutte le chiamate sono intercettate da un sistema di middleware che:

- aggiunge automaticamente il token JWT all'header `Authorization` di ogni richiesta;
- effettua un refresh automatico del token in caso di scadenza, tramite la richiesta `/auth/refresh`;
- forza il logout e la cancellazione dei dati locali se l'autenticazione non è più valida.

La gestione del token è centralizzata nel modulo `secureStore.ts`, che offre funzioni di salvataggio (`saveToken`), lettura (`getToken`), eliminazione (`Logout`) e decodifica del payload JWT (`retrieveTokenFromStorage`).

Il modulo `secureStore.ts` gestisce inoltre parametri aggiuntivi relativi alla personalizzazione dell'app, conservati in modo persistente e cifrato, come:

- uso dei dati biometrici (`biometricsKey`);
- pin di accesso (`userPin`);
- modalità di lavoro corrente (`workModeKey`);
- lingua selezionata (`languageKey`).

5.3 Funzionalità principali sviluppate

5.3.1 Autenticazione sicura e gestione delle sessioni

Il processo di autenticazione rappresenta uno degli elementi chiave dell'applicazione, in quanto garantisce l'accesso ai dati aziendali.

Il sistema implementato adotta un modello multi-step che combina tre livelli di sicurezza progressivi:

1. Login con credenziali aziendali

L'utente effettua l'accesso inserendo *username* e *password*, credenziali già utilizzate nel gestionale aziendale. In caso di errore (credenziali errate o server non raggiungibile), viene mostrato un messaggio modale costruito tramite un custom hook (`useCustomModal`), che fornisce un feedback all'utente;

2. Verifica tramite codice OTP

Dopo il primo accesso, l'app richiede l'inserimento di un codice One-Time Password, inviato all'utente tramite e-mail. Il codice viene validato attraverso la chiamata `await sendOtp(otp, signature)` che interagisce con il backend tramite `/auth/otp`;

3. Impostazione e verifica PIN

Al primo login, l'utente imposta un codice PIN composto da 4 cifre, salvato in modo cifrato nel `SecureStore` che, nelle sessioni successive, consente un accesso rapido senza dover reinserire le credenziali complete.

Il processo di validazione gestisce due casi distinti: la prima configurazione, che richiede una doppia conferma del PIN, e la verifica successiva con confronto rispetto al valore salvato;

4. Autenticazione biometrica opzionale

Al termine della configurazione del PIN, viene proposta l'attivazione dell'autenticazione biometrica. La scelta dell'utente viene memorizzata in modo persistente e consente il riconoscimento automatico all'avvio;

5. Gestione del token JWT

Dopo la validazione dell'OTP, il server restituisce automaticamente un token JWT associato a ogni richiesta API grazie agli interceptors di `Axios`. Il token viene decodificato per verificarne la scadenza e, in caso di errore 401, viene eseguito un refresh automatico o il logout.

5.3.2 Timbratura e gestione delle presenze

La schermata home rappresenta il punto di partenza per tutte le operazioni principali legate alla gestione delle presenze.

La logica principale è implementata nel componente `Overview.tsx` che gestisce il recupero, l'elaborazione e la visualizzazione delle timbrature giornaliere tramite API aziendali.

1. Recupero dati giornalieri: tramite la funzione `getClockings(date)`, l'applicazione interroga il backend - `/ore/giorno?date=` - ottenendo le timbrature registrate e il totale delle ore lavorate. I dati vengono aggiornati automaticamente, garantendo la sincronizzazione in tempo reale della schermata.

2. Calcolo delle ore lavorate: la funzione `calculateWorkingHours()` elabora le coppie di orari entrata/uscita e aggiunge la fascia oraria corrente in caso di sessione aperta.
3. Timbratura manuale tramite swipe: la funzione `handleClock()` è collegata al componente `swipeButton`. Al completamento dello swipe, viene invocata la funzione `await clockInOut(getWorkMode() ?? 0, note)`, che registra una nuova timbratura associata alla modalità di lavoro selezionata. Viene poi mostrato un toast di conferma o errore.
4. Gestione modalità di lavoro: memorizzata in locale, può essere selezionata tramite menù a tendina (`WorkSelector`). Ogni modalità corrisponde a un colore primario differente, generato dinamicamente tramite la funzione `getThemeWithPrimary()`.
5. Gestione e aggiornamento automatico dello stato: il sistema aggiorna in tempo reale l'ultima timbratura, la tipologia e il totale ore giornaliera.

5.3.3 Calendario e visualizzazione storica delle presenze

La schermata time-table consente all'utente di consultare lo storico delle presenze, visualizzare anomalie e monitorare l'andamento mensile delle ore di lavoro.

Il componente Calendar gestisce la visualizzazione di un calendario mensile dinamico, con funzionalità di navigazione e segnalazione anomalie.

- L'utente può spostarsi tra i mesi tramite due icone (Chevron left - Chevron right), aggiornando automaticamente la vista e i dati collegati;
- Ogni cella del calendario rappresenta un giorno del mese. L'elemento corrispondente alla data corrente è evidenziato da un cerchio colorato che riprende il colore principale del tema;
- Durante il caricamento dei dati (`getClockings()`), l'app verifica la presenza di anomalie tramite il campo `anomaliaBollature` restituito da backend. I giorni con anomalie vengono contrassegnati da un'icona di warning;
- Se si interagisce con una cella del calendario, viene aperta la schermata di dettaglio "Timbrature", passando come parametro la data selezionata:

```
navigation.navigate('Timbrature', { dateTimbrature: selectedDate });
```

L'aggiornamento dei dati avviene in modo asincrono attraverso `useEffect`, che richiama automaticamente le API ogni volta che cambia il mese visualizzato.

Lo stato locale mantiene le anomalie in un oggetto di mapping `Record<string, boolean>`.

Il componente `TimetableWidget` fornisce una panoramica sintetica delle ore di straordinario, dei permessi e della percentuale di ore di lavoro svolte in modalità smart working. Le informazioni vengono ottenute tramite l'endpoint `/ore/month`, che restituisce i dati aggregati per ciascun giorno del mese selezionato. Successivamente, il componente elabora le informazioni mediante l'esecuzione di tre funzioni principali:

- `getOvertime()`: calcola il totale delle ore di straordinario sommandole giorno per giorno. I risultati vengono formattati in ore e minuti per avere una maggiore leggibilità;
- `getPermit()`: recupera e somma le ore di permesso usufruite nel mese corrente;
- `getSmart()`: determina la percentuale di ore svolte in smart working rispetto al monte ore complessivo. La funzione calcola dinamicamente il limite massimo consentito (pari al 40% delle ore mensili totali) e restituisce un oggetto contenente:

```
{
  percentage: 100 * smart / maxSmart,
  label: `${hours}h ${minutes}m / ${maxHours}h ${maxMinutes}m`
}
```

I dati vengono poi rappresentati graficamente tramite il componente `CircleProgress` che mostra una barra di avanzamento circolare. In questo modo l'utente ha una rappresentazione immediata della propria distribuzione oraria mensile e dello stato rispetto al limite massimo di smart working.

5.3.4 Timbratura

La schermata Timbrature consente all'utente di visualizzare in modo dettagliato le timbrature giornaliere.

È possibile accedere a questa schermata da due punti dell'applicazione:

- dalla sezione overview della homepage selezionando il riquadro relativo all'ultima timbratura;
- dal calendario mensile, selezionando una data specifica.

In entrambi i casi, viene passata come parametro la data selezionata che consente di caricare dinamicamente i dati relativi a quel giorno.

I dati vengono recuperati tramite la funzione `getClocking(date)`, che interroga il backend e restituisce la lista delle timbrature per la data selezionata. Ogni voce viene poi formattata come:

```
{
  evento: tipologia // entrata o uscita
  orario: '08:45' // ora della timbratura
  modalità 'ufficio' // modalità di lavoro
  warning: false // presenza di eventuali anomalie
}
```

Il risultato è mostrato in una tabella dinamica, implementata nel componente `Clocking`.

5.3.5 Localizzazione e supporto multilingue

L'applicazione è stata progettata per essere multilingue, in modo da poter essere utilizzata anche in contesti internazionali.

Per la gestione delle traduzioni è stata adottata la libreria `i18n-js`, integrata con `expo-localization`, che consente di rilevare automaticamente la lingua impostata sul dispositivo e applicare le stringhe corrispondenti.

Il modulo di localizzazione è definito nel file `i18n.ts`, dove sono mappate le due lingue supportate: italiano e inglese. Ogni sezione dell'app ha un proprio namespace, così da mantenere il codice leggibile.

```
const i18n = new I18n({
  en: {
    login: { signIn: 'Sign in to continue' },
    homepage: { workedHours: 'Worked hours' },
    settings: { biometricRecognition: 'Biometric recognition' }
  },
  it: {
    login: { signIn: 'Accedi per continuare' },
    homepage: { workedHours: 'Ore lavorate' },
    settings: { biometricRecognition: 'Riconoscimento biometrico' }
  }
});
```

Il codice imposta automaticamente la lingua locale del dispositivo grazie a `expo-localization`:

```
i18n.locale = getLocales()[0].languageCode;
i18n.fallbacks = true;
```

Il parametro `fallback` garantisce che, in assenza di traduzione specifica, venga utilizzata la lingua di default.

Le traduzioni vengono richiamate direttamente nei componenti dell'interfaccia tramite la funzione `i18n.t()`:

```
<Text>{i18n.t('homepage.workedHours')}</Text>
```

Inoltre, è possibile modificare manualmente la lingua selezionata all'interno delle impostazioni, dove l'utente può scegliere tra italiano e inglese. La preferenza viene salvata in modo persistente nel `secureStore`, in modo che la lingua selezionata venga mantenuta anche nelle sessioni successive.

Capitolo 6

Test e validazione

La fase di testing ha svolto un ruolo cruciale nel miglioramento dell'applicazione. Lo scopo consisteva nell'assicurare che il sistema soddisfacesse i requisiti funzionali e, in particolare, nel rilevare eventuali anomalie nel comportamento, incongruenze logiche o problemi legati all'interfaccia e al funzionamento, così da correggerli prima del rilascio della versione definitiva.

I test sono stati condotti al termine della fase di progettazione su una versione operativa dell'applicazione, integrata con il backend aziendale.

Le attività di testing si sono sviluppate principalmente in due direzioni:

- Test end-to-end (E2E): volti a riprodurre scenari reali di utilizzo, simulando il comportamento dell'utente attraverso l'intero flusso dell'applicazione;
- Test di usabilità e accettazione: finalizzati alla raccolta di feedback qualitativi da parte di un gruppo di utenti selezionati con l'obiettivo di valutare la chiarezza dell'interfaccia, l'intuitività delle funzionalità e l'efficacia complessiva dell'esperienza utente.

6.1 Tipologia di test e metodologia adottata

I test E2E consentono di analizzare il comportamento dell'applicazione simulando il flusso di utilizzo reale.

Ogni test è stato eseguito seguendo scenari d'uso concreti, dalla fase di login iniziale fino alla gestione delle timbrature e delle impostazioni.

Le prove sono state condotte su una combinazione di dispositivi reali ed emulatori, per valutare il comportamento dell'applicazione in diversi ambiti operativi:

- emulatore iOS tramite Xcode;

- dispositivo iPhone aziendale;
- dispositivo Android aziendale.

Tale diversificazione ha consentito di analizzare la compatibilità tra le diverse piattaforme e di identificare eventuali variazioni di comportamento attribuibili al sistema operativo utilizzato.

I test E2E sono stati condotti da una risorsa del reparto di test aziendale, sotto la supervisione dello sviluppatore responsabile del progetto. La collaborazione ha permesso di integrare la visione tecnica con quella funzionale. Queste attività sono state svolte manualmente, senza l'impiego di strumenti automatizzati, questo per concentrarsi sull'esperienza utente complessiva, valutando in tempo reale la fluidità di navigazione, la coerenza visiva e la chiarezza dei messaggi di sistema.

Dall'analisi dei risultati ottenuti l'applicazione ha mostrato un comportamento complessivamente stabile e coerente con i requisiti di progetto. Le funzionalità principali risultano correttamente operative; tuttavia, sono emersi dei miglioramenti da apportare, legati principalmente alla chiarezza dei messaggi di sistema e alla formattazione di alcune notifiche.

6.2 Risultati e analisi dei test end-to-end

In totale sono stati effettuati 62 test E2E, coprendo tutte le principali funzionalità dell'applicazione. Questi test hanno permesso di verificare sia gli aspetti funzionali che quelli relativi all'interfaccia grafica, includendo il funzionamento dei temi e la gestione dei dati persistenti.

- Totale test eseguiti: 62
- Test superati: 56
- Test falliti: 6
- Tasso di successo: 90%

In particolare le discrepanze riscontrate non compromettono il funzionamento generale, ma riguardano aspetti migliorativi relativi all'interfaccia e alla messaggistica. Le aree che risultano completamente superate corrispondono ai test relativi a:

- login (T20, T23-T24, T26-T32);

- persistenza dei dati e dello stato (T3, T8);
- gestione delle timbrature e KPI giornalieri (T1, T4-T7, T34-T35, T37-T38, T40-T51);
- navigazione e sincronizzazione tra viste (T52-T57);
- temi, contrasto e lingua (T2, T9-T17, T19, T33);
- gestione lingua e Logout (T58-T62).

Sono stati identificati alcuni punti che necessitano di ottimizzazione:

Categoria	ID Test	Descrizione	Esito	Azione consigliata
Gestione temi	T18	Il tema selezionato non persiste dopo il riavvio in alcune condizioni	Fail	Verificare salvataggio del tema nel SecureStore
Login	T21-T22	Username con spazi iniziali/finali non gestiti	Fail	Applicare trim automatico all'input dello username
OTP	T25	Messaggi di errore non coerenti	Fail	Localizzare il messaggio di errore quando OTP sbagliato
Note Home	T36	Nota non persiste nella schermata Home dopo la timbratura	Fail	Verificare il salvataggio della nota
Note Home	T39	Testo multilinea non gestito	Fail	Limitare numero di caratteri

Tabella 6.1: Risultati dei test E2E

I test di accessibilità visiva hanno ottenuto un esito pienamente positivo, a conferma della corretta implementazione delle linee guida WCAG. Le funzionalità core (Login, OTP, PIN, timbratura, calendario, impostazioni) hanno dimostrato una buona stabilità, senza blocchi durante l'esecuzione. Le principali criticità emerse riguardano la gestione delle eccezioni e dei messaggi informativi.

6.3 Test di usabilità e accettazione

A seguito della fase di validazione funzionale, è stata condotta una sessione di test di usabilità al fine di valutare la chiarezza dell'interfaccia, la fluidità di utilizzo e la percezione complessiva dell'esperienza utente.

Le sessioni sono state organizzate dopo la conclusione dei test E2E, in presenza, con l'obiettivo di raccogliere osservazioni dirette sul comportamento dell'applicazione e identificare eventuali punti di miglioramento.

Il test ha coinvolto cinque utenti aziendali, appartenenti a ruoli diversi (sviluppatori, tester e personale amministrativo).

Le prove sono state condotte su dispositivi fisici iOS e Android aziendali, senza fornire credenziali dedicate: ciascun partecipante ha effettuato l'accesso utilizzando il proprio account aziendale, garantendo così un contesto realistico di utilizzo.

Ogni sessione è stata supervisionata direttamente dallo sviluppatore, il quale ha annotato i commenti e le difficoltà emerse. I partecipanti hanno completato una serie di dieci task rappresentativi delle principali funzionalità dell'applicazione, quali il login, l'OTP, l'impostazione del PIN, la timbratura, il cambio modalità, la navigazione tra le sessioni e le impostazioni.

Al termine di ogni sessione, è stato somministrato un questionario di feedback, comprendente una scala di valutazione a cinque punti e domande aperte per la raccolta di suggerimenti qualitativi.

6.3.1 Risultati quantitativi

L'analisi delle risposte rivela un livello di soddisfazione complessivamente elevato, con punteggi che si attestano su valori medio-alti.

Il punteggio medio complessivo derivante dalle valutazioni globali dei partecipanti è pari a 8.4 su 10, evidenziando un buon livello di accettazione e una percezione positiva dell'applicazione in termini di stabilità e facilità d'uso.

Aspetto	Valore medio (1-5)
Facilità di login e autenticazione	4.75
Chiarezza delle schermate e delle etichette	4.00
Fluidità di navigazione	4.75
Comprensibilità dei messaggi di errore	4.75
Aspetto grafico e leggibilità dei testi	5.00
Utilità delle informazioni in Home	4.00
Comprensione del calendario (Timetable)	3.25
Chiarezza della sezione Timbrature	4.50
Completezza delle impostazioni	4.75

Tabella 6.2: Risultati dei test di usabilità

6.3.2 Risultati qualitativi

Dall'analisi dei commenti e delle osservazioni raccolte emergono i seguenti punti principali:

- Punti di forza:
 - chiarezza del flusso di login e autenticazione;
 - semplicità di navigazione e coerenza grafica tra le sezioni;
 - feedback immediato durante la timbratura;
 - riconoscimento intuitivo delle modalità di lavoro tramite colori.
- Criticità riscontrate:
 - alcuni messaggi di errore generici nelle schermate di login e OTP;
 - difficoltà nel comprendere la logica del carico di lavoro;
 - richieste di maggiore personalizzazione della Home;
 - necessità di migliorare la visibilità del testo nel calendario nel tema scuro.

Nel complesso, i test di usabilità hanno confermato la coerenza dell'interfaccia e la stabilità delle funzionalità principali, con un grado di soddisfazione elevato da parte degli utenti.

Le osservazioni emerse sono state raccolte come base per ulteriori ottimizzazioni.

6.4 Considerazioni e modifiche apportate

In seguito alle sessioni di test E2E e di usabilità, sono emersi alcuni punti su cui è possibile intervenire per miglioramenti, sia dal punto di vista funzionale che in termini di esperienza utente. Le osservazioni raccolte dai tester si sono rivelate particolarmente utili per ottimizzare la chiarezza dei messaggi e migliorare la comprensibilità delle informazioni presentate in alcune sezioni.

Le modifiche introdotte hanno riguardato principalmente:

- miglioramento dei messaggi di errore, in particolare durante il login e la verifica OTP, rendendoli più esplicativi e coerenti con le azioni richieste all'utente;
- aggiunta di un messaggio informativo nel widget del carico di lavoro, per chiarire la condizione in cui i dati non risultano ancora disponibili, al fine di evitare confusione o interpretazioni errate;
- aggiornamento del testo nel widget delle note, specificando che il salvataggio della nota avviene al momento della timbratura, così da rendere più chiaro il comportamento dell'applicazione;
- introduzione di un collegamento diretto alle impostazioni tramite l'avatar presente nella schermata Home, per favorire una navigazione più intuitiva e veloce;
- revisione dei colori e dei contrasti grafici, in particolare nel calendario e nelle sezioni a tema scuro, per migliorare la leggibilità e la coerenza tra i diversi elementi dell'interfaccia.

Nel complesso, le modifiche apportate hanno contribuito a migliorare la chiarezza dell'interfaccia e a rendere più fluido il flusso di utilizzo, incrementando così la qualità complessiva dell'esperienza utente.

Capitolo 7

Conclusioni e sviluppi futuri

Lo sviluppo di questa applicazione ha rappresentato per me un'esperienza formativa, sia dal punto di vista tecnico che personale.

Fin dalle prime fasi di progettazione, è stato necessario considerare un contesto aziendale reale, con esigenze precise, tempi limitati e l'obiettivo di realizzare un prodotto che potesse essere effettivamente utilizzabile dai dipendenti. Questo mi ha permesso di comprendere l'importanza della pianificazione, della collaborazione con altre figure professionali e della capacità di adattarsi a esigenze che evolvono nel tempo.

La difficoltà principale è stata quella di integrare un'interfaccia moderna e intuitiva con un sistema backend preesistente, garantendo al contempo coerenza tra le piattaforme e compatibilità con sistemi eterogenei.

Una ulteriore sfida è stata la definizione di un'esperienza utente equilibrata: l'applicazione doveva mantenersi semplice e accessibile per tutte le tipologie di utente.

Sulla base dei feedback ricevuti durante la fase di test, sono state implementate diverse modifiche che hanno contribuito a rendere l'esperienza ancora più chiara e coerente. L'osservazione delle interazioni degli utenti con la piattaforma ha consentito di comprendere meglio le loro esigenze e di correggere aspetti inizialmente dati per scontato.

Questo progetto è destinato a evolversi oltre la semplice gestione delle timbrature. L'obiettivo è l'integrazione di un sistema di acquisizione immagini intelligente, in grado di interagire con altri moduli quali la gestione dei rimborsi spese, attraverso la digitalizzazione e la categorizzazione automatica dei documenti, e la gestione delle bolle di ingresso materiali per il magazzino, con riconoscimento e archiviazione strutturata delle informazioni. Questa evoluzione si baserà sull'utilizzo di tecniche

di riconoscimento automatico del testo (OCR), che consentiranno l'estrazione dei dati direttamente dai documenti cartacei.

Nel complesso, questa esperienza mi ha permesso di consolidare le mie competenze nello sviluppo mobile, nella progettazione di interfacce e nella gestione di un processo software completo, ma soprattutto mi ha insegnato l'importanza fondamentale del confronto con gli utenti e della capacità di trasformare il feedback ricevuto in miglioramenti concreti.

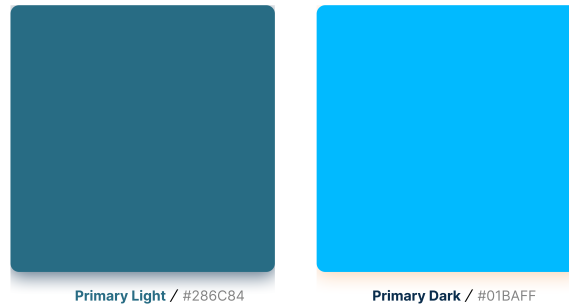
Appendice A

Componenti e varianti grafiche

La presente appendice raccoglie il set completo dei componenti grafici e degli elementi visivi progettati per l'applicazione mobile sviluppata nell'ambito di questa tesi. Il sistema di design è stato costruito in Figma.

Colors

Brand Colors



Primary colors



Other Colors

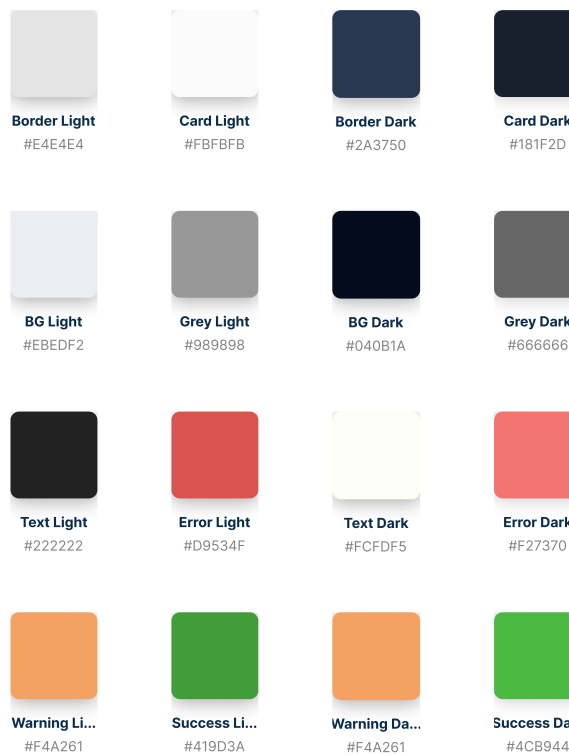


Figura A.1: Colori

Typography

Aa

Heading

Roboto / San Francisco

Name	Font size
Large Text Bold	22 px
Large Text Regular	
Medium Text Bold	18 px
Medium Text Regular	
Normal Text Bold	16 px
Normal Text Regular	
Small Text Bold	14 px
Small Text Regular	

Figura A.2: Tipografia

Spacing

Levels

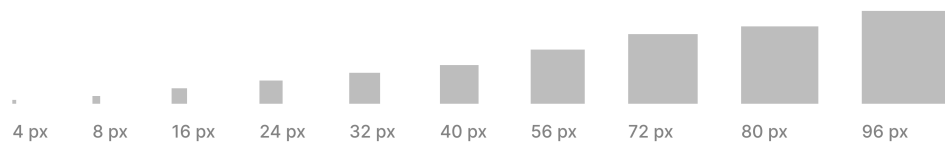


Figura A.3: Spacing

Textfields

Label , Status & Placeholder

Label

Placeholder



Input Rules:

Padding Left → 12px
Padding Top-Bottom → 8px
Padding Right(icon) → 12px

Input Form (No label & Status)

Input Text

Input Icon + Label

Label Sample

Input Text

Text Area

Label Sample

Input Form (No label & Status)

cognome@sirius-ea.com

Input Icon + Label

Label Sample

cognome@sirius-ea.com

Text Area

Label Sample

Figura A.4: Textfields

Selectors

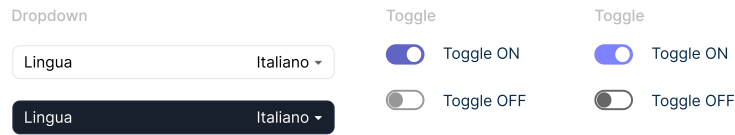


Figura A.5: Selectors

Buttons

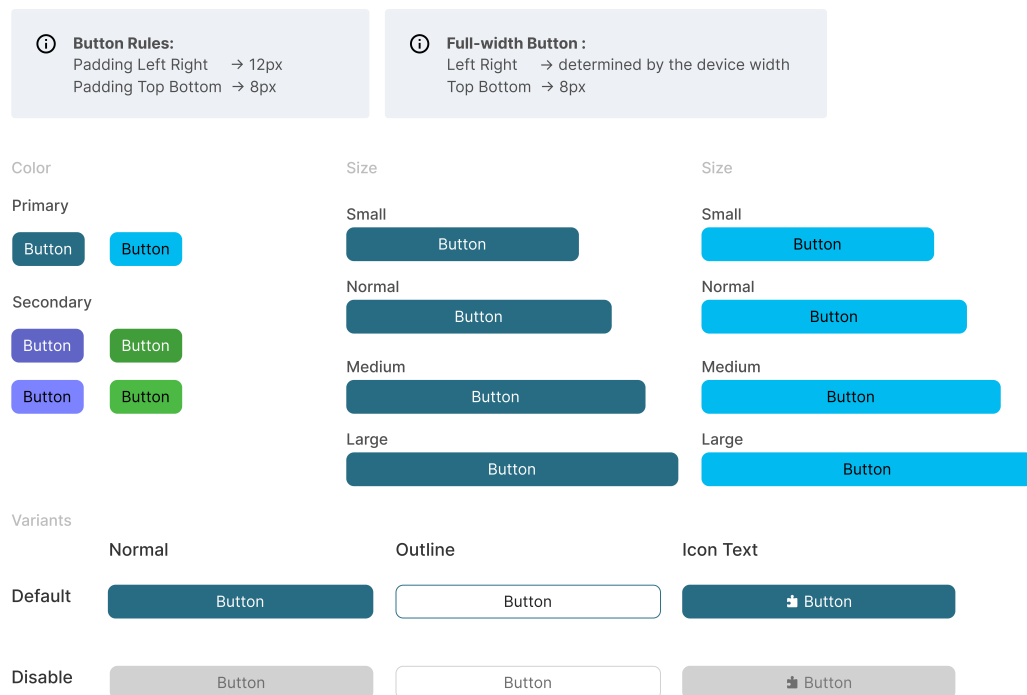


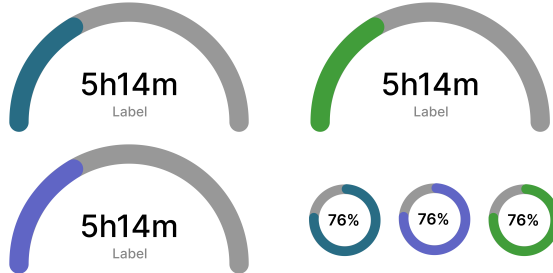
Figura A.6: Buttons

Elements

Swipe button Light



Progress bar Light



Pie charts Light



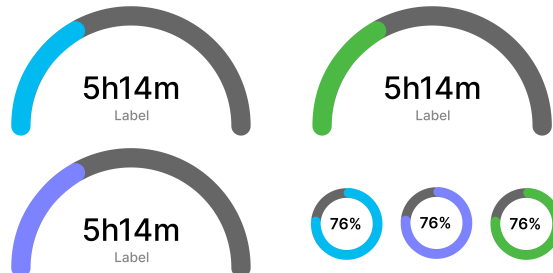
Avatar Light



Swipe button Dark



Progress bar Dark



Pie charts Dark



Avatar Dark



Bottom Navigation Bar

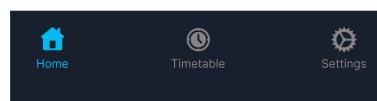
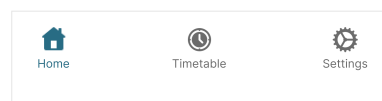


Figura A.7: Elements

Toast & Feedback

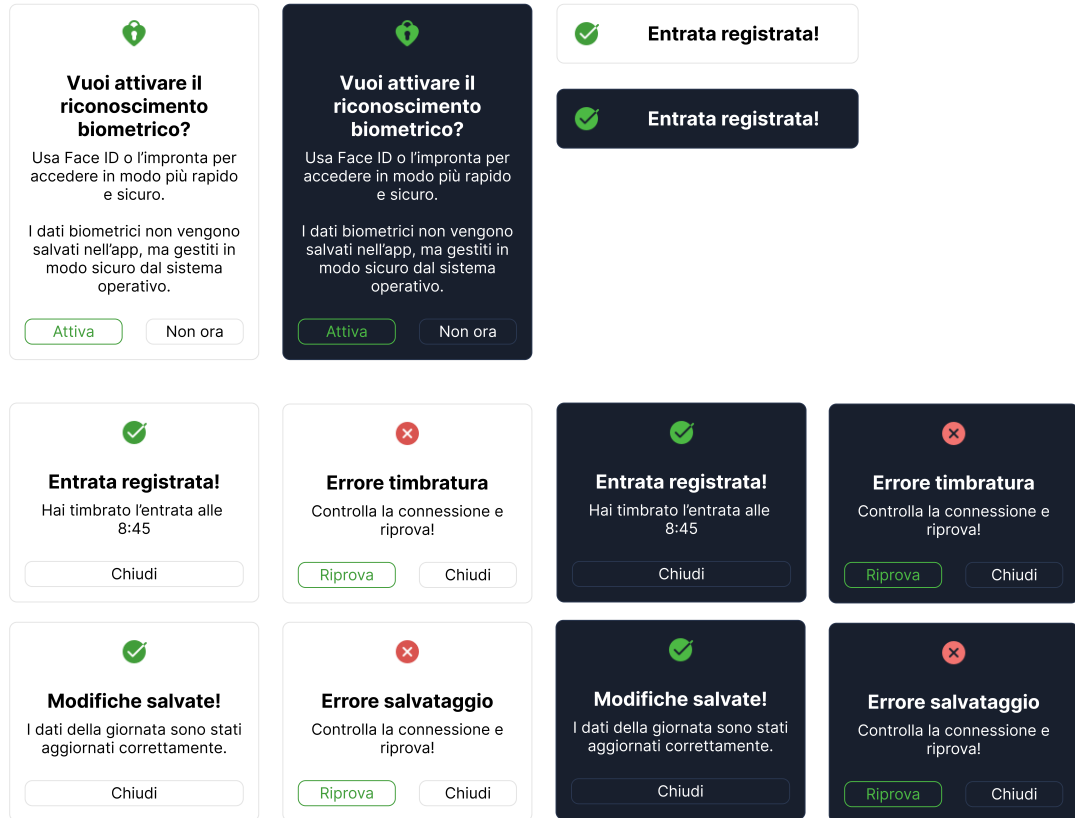


Figura A.8: Toast and feedbacks

Appendice B

Varianti grafiche delle schermate

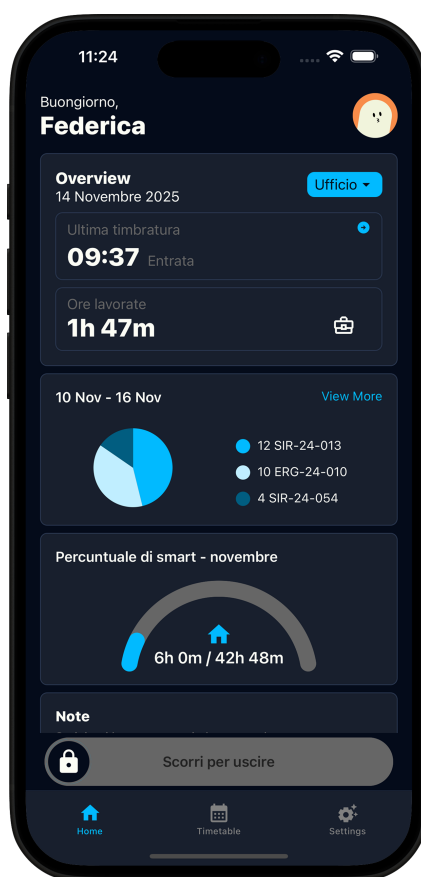


Figura B.1: Schermata Home (modalità scura)

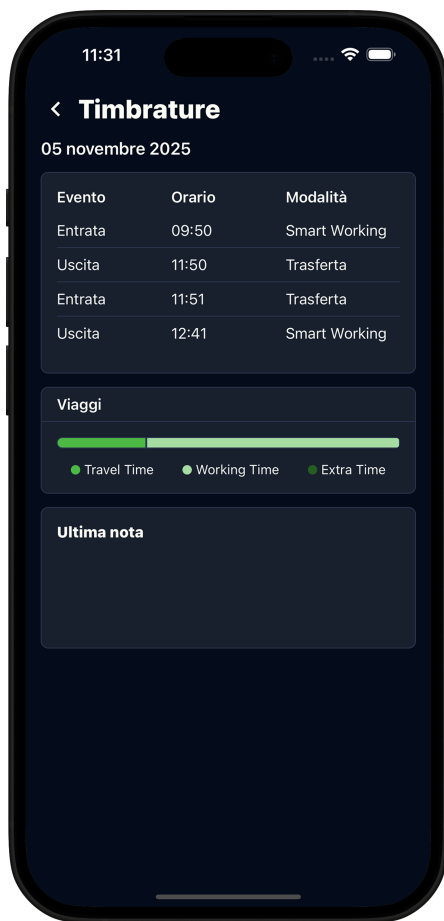


Figura B.2: Schermata Timbrature (modalità scura)



Figura B.3: Schermata Timbrature (modalità scura / trasferta)

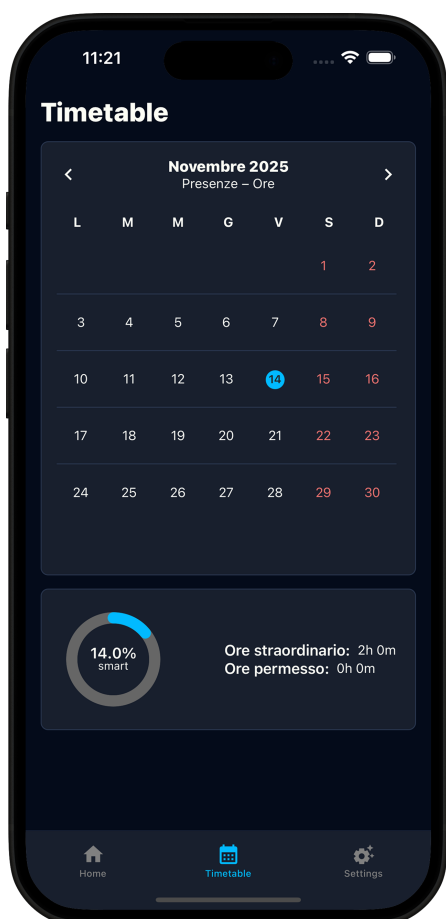


Figura B.4: Schermata Timetable (modalità scura)

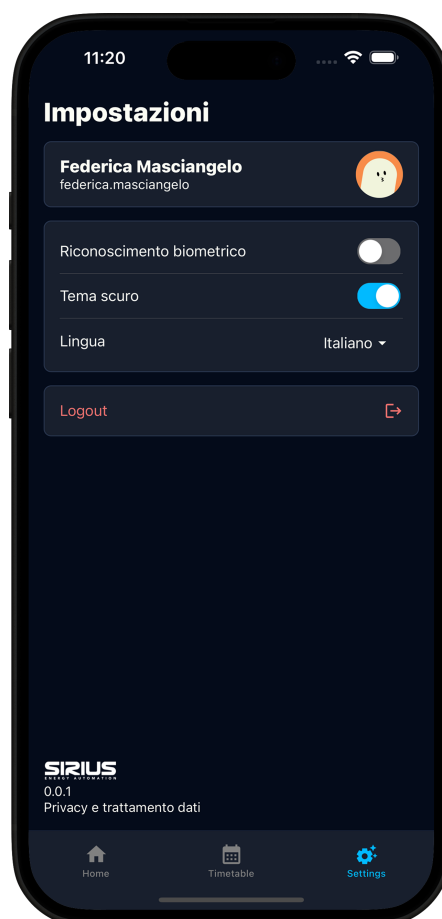


Figura B.5: Schermata Impostazioni (modalità scura)

Bibliografia

- [1] Libemax S.r.l. *La storia delle timbrature del cartellino: dalle prime meccaniche alle app moderne*. Accesso effettuato il 06 ottobre 2025. 2023. URL: <https://www.apprilevazionepresenze.com/it/news/135/la-storia-delle-timbrature-del-cartellino-dalle-prime-meccaniche-alle-app-moderne> (cit. a p. 4).
- [2] Compendium S.P.A. *Rilevazione presenze automatica: i vantaggi*. Accesso effettuato il 03 ottobre 2025. 2024. URL: <https://www.it-compendium.com/it/blog/rilevazione-presenze-automatica-i-vantaggi> (cit. a p. 5).
- [3] Fluida.io. *Rilevazione presenze con app mobile e dispositivi Bluetooth*. Accesso effettuato il 03 ottobre 2025. 2025. URL: <https://www.fluida.io/rilevazione-presenze> (cit. a p. 5).
- [4] Factorial.it. *Software rilevazione presenze: gestione orari e timbrature*. Accesso effettuato il 03 ottobre 2025. 2025. URL: <https://factorial.it/software-rilevazione-presenze> (cit. a p. 6).
- [5] Google. *Material Design Guidelines*. Accesso effettuato il 16 giugno 2025. 2025. URL: <https://m3.material.io/> (cit. alle pp. 19, 30).
- [6] W3C. *Web Content Accessibility Guidelines (WCAG) 2.1*. Accesso effettuato il 16 giugno 2025. 2018. URL: <https://www.w3.org/TR/WCAG21/> (cit. a p. 30).
- [7] Jakob Nielsen. *10 Usability Heuristics for User Interface Design*. Accesso effettuato il 20 giugno 2025. 1994. URL: <https://www.nngroup.com/articles/ten-usability-heuristics/> (cit. a p. 30).
- [8] Henrique Gusso. *Use Contrast – Figma Plugin*. Accesso effettuato il 26 settembre 2025. 2022. URL: <https://www.figma.com/community/plugin/1149686177449921115/use-contrast> (cit. a p. 31).
- [9] Meta Platforms Inc. *React Native Documentation*. Accesso effettuato il 15 maggio 2025. 2025. URL: <https://reactnative.dev/> (cit. a p. 32).
- [10] Expo. *Expo Documentation – SDK 54*. Accesso effettuato il 15 maggio 2025. 2025. URL: <https://expo.dev/changelog/sdk-54> (cit. a p. 32).

- [11] Microsoft. *TypeScript Documentation*. 2025. URL: <https://www.typescriptlang.org/> (cit. a p. 32).
- [12] JetBrains. *PhpStorm – Smart PHP IDE*. 2025. URL: <https://www.jetbrains.com/phpstorm/> (cit. a p. 32).
- [13] Meta Platforms Inc. *React Navigation - Routing and navigation for React Native and Web apps*. Accesso effettuato il 15 maggio 2025. 2024. URL: <https://reactnavigation.org/> (cit. a p. 32).
- [14] *Axios - Promise based HTTP client for the browser and node.js*. Accesso effettuato il 12 agosto 2025. 2020. URL: <https://axios-http.com/> (cit. a p. 32).
- [15] Expo. *Expo SecureStore*. Accesso effettuato il 12 agosto 2025. 2025. URL: <https://docs.expo.dev/versions/latest/sdk/securestore/> (cit. a p. 33).
- [16] Expo. *Expo LocalAuthentication*. Accesso effettuato il 12 agosto 2025. 2025. URL: <https://docs.expo.dev/versions/latest/sdk/local-authentication/> (cit. a p. 33).
- [17] fnando. *i18n-js*. Accesso effettuato il 22 settembre 2025. 2024. URL: <https://www.npmjs.com/package/i18n-js> (cit. a p. 33).
- [18] Expo. *Expo Localization*. Accesso effettuato il 22 settembre 2025. 2025. URL: <https://docs.expo.dev/versions/latest/sdk/localization/> (cit. a p. 33).
- [19] calintamas. *react-native-toast-message*. Accesso effettuato il 29 agosto 2025. 2025. URL: <https://www.npmjs.com/package/react-native-toast-message> (cit. a p. 33).
- [20] usk.kamineni. *rn-swipe-button*. Accesso effettuato il 20 giugno 2025. 2025. URL: <https://www.npmjs.com/package/rn-swipe-button> (cit. a p. 33).
- [21] hermanya - olegberman. *react-native-chart-kit*. Accesso effettuato il 23 ottobre 2025. 2021. URL: <https://www.npmjs.com/package/react-native-chart-kit> (cit. a p. 33).
- [22] Infinite Red. *Reactotron*. Accesso effettuato il 24 settembre 2025. 2025. URL: <https://docs.infinite.red/reactotron/> (cit. a p. 33).
- [23] Apple Inc. *Xcode*. 2025. URL: <https://developer.apple.com/xcode/> (cit. a p. 33).