



PROGETTAZIONE E SVILUPPO DI UN'APPLICAZIONE ANDROID PER LA GESTIONE DEGLI ORARI DI LAVORO

Dipartimento di Scienze Matematiche, Informatiche e Fisiche
Corso di Laurea Triennale in Informatica

Laureando: Alberto Caliman

Relatore: Dott. Stefano Burigat

Introduzione

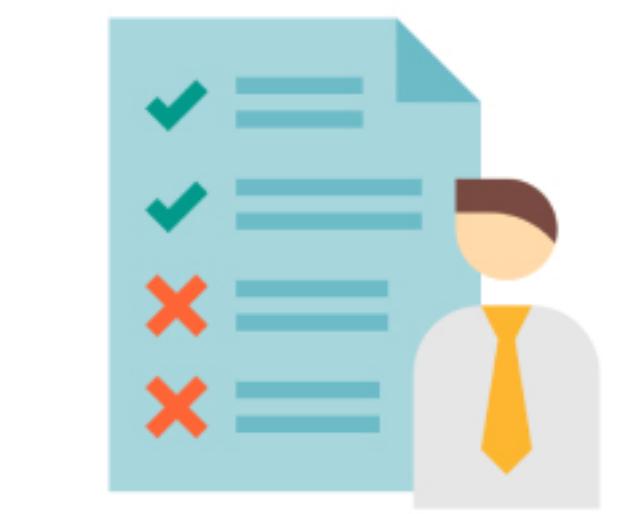
Il tirocinio svolto presso Sinesy srl ha previsto lo sviluppo di un'applicazione mobile nativa Android. L'idea è nata dalla necessità di offrire una soluzione moderna e intuitiva, che permetta ai dipendenti dell'azienda di gestire le presenze e gli orari di lavoro in qualsiasi momento, eseguendo il minimo numero di operazioni per inserire i dati. L'applicazione affianca un sito web esistente, il quale permette di svolgere tali attività in modo più complesso e non chiaro a utenti non tecnici. Si vuole quindi modificare alcune operazioni presenti nel sito web, rendendo le operazioni di inserimento delle ore di lavoro e delle commesse più semplici, intuitive e veloci. Il tirocinio della durata di cinque mesi ha permesso di integrarmi in un team di sviluppo strutturato e ben consolidato. I principali strumenti utilizzati per lo sviluppo sono: AdobeXD per la realizzazione dei prototipi e il linguaggi di programmazione Java e Kotlin per la parte di implementazione e testing, arricchiti da librerie come: Room, LiveData contenute all'interno degli Android Architecture Components.

Analisi dei requisiti

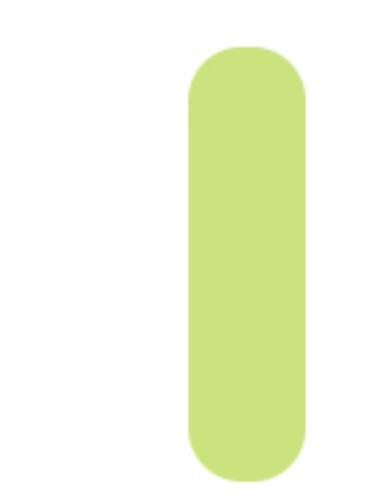
La progettazione dell'applicazione è iniziata con la fase di analisi e raccolta dei requisiti. Esaminando il funzionamento della versione web sono state definite le funzionalità principali e compreso il loro funzionamento. Successivamente tramite il brainstorming, che ha coinvolto 15 dipendenti (tecnicci e non), sono state evidenziate le principali proprietà che il sistema finale doveva offrire. Infine i requisiti e le idee proposte nella fase di analisi sono state definite in modo chiaro ed esaustivo, i requisiti raggruppati tra loro, in modo che fossero coerenti e correlati producendo infine un documento che raccoglie i principali requisiti.

Design

La fase di progettazione del design ha previsto inizialmente la realizzazione di prototipi a basso livello. Lo scopo principale di tale strumento è di esplorare velocemente e in modo economico quali sono le idee principali per le differenti finestre che verranno successivamente implementate. Tramite lo sketching sono state disegnate le idee iniziali delle varie schermate. Mediante raffinamenti successivi, si è giunti alla realizzazione di prototipi ad alta fedeltà in cui sono stati definiti font, colori, dimensione del testo e immagini. Questi prototipi sono stati testati sugli utenti, i quali hanno evidenziato delle criticità sulla funzionalità di cancellazione delle commesse e della mancanza di feedback in fase di salvataggio. Grazie ai feedback ottenuti dagli utenti, le schermate sono state modificate e infine nuovamente fatte testare. Dopo le modifiche gli utenti non hanno evidenziato ulteriori criticità. Il team che si è occupato della parte grafica, nel quale ho contribuito, ha inoltre definito uno User Flow Diagram che ha definito la navigazione che l'utente doveva avere durante la navigazione. Infine i prototipi ad alta fedeltà sono stati consegnati al team di sviluppo che ha implementato le schermate in Android.



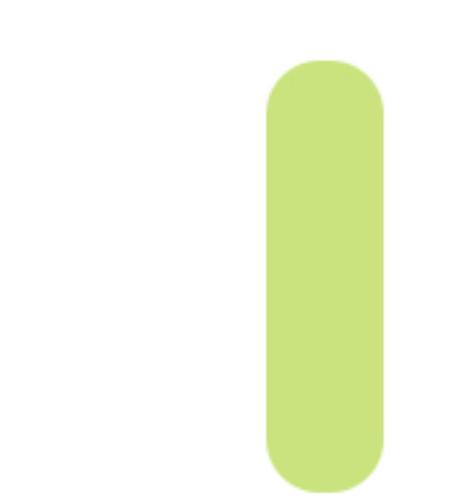
Analisi dei
requisiti



UX e UI Design



Implementazione
in Android



Verifica automatica
delle interfacce



Dettagli implementativi

La progettazione dell'applicazione è iniziata con lo studio dei diversi modelli architetturali comunemente utilizzati in ambito mobile, evidenziandone pregi e difetti di ognuno. Successivamente è stato scelto il modello MVVM in quanto il team aveva già esperienza nell'utilizzo di tale architettura. Inoltre Google ha rilasciato un insieme di librerie chiamate Android Architecture Components che permettono di strutturare un'applicazione secondo il modello MVVM in modo semplice e intuitivo.

Il progetto inoltre, include un database SQLite per tenere in memoria alcune commesse, ed è stato realizzato utilizzando la libreria Room. Per ottenere le commesse e i dati relativi ad ogni dipendente vengono utilizzati dei Json Web Service. Questo formato testuale permette di mettere in comunicazione tramite il protocollo HTTP/1.1 l'applicazione con il server. L'utilizzo dei LiveData ha permesso di gestire i flussi di dati provenienti dai vari endpoint forniti dai WebService e mostrare all'utente sempre i dati più aggiornati, senza che l'utente dovesse eseguire operazioni di aggiornamento.

Verifica

L'ultima fase del processo di progettazione e implementazione si è concentrata nello studio dei vari framework in commercio utilizzati per testare in modo dinamico le viste. Lo scopo dei test delle interfacce è di verificare il corretto funzionamento delle viste realizzate, verificando che l'utente riesca ad interagire con i componenti che caratterizzano l'interfaccia e che l'ui corrisponda a quanto progettato nello user flow diagram. Lo studio ha portato il team a utilizzare il framework Espresso in quanto l'applicazione è stata sviluppata utilizzando strumenti nativi e quindi sembrava la scelta più appropriata. Inoltre, si voleva apprendere a fondo il funzionamento del framework per valutarne l'utilizzo in un'applicazione di media complessità. I test realizzati sono stati eseguiti su cinque dispositivi fisici, con diverse versioni del sistema Android e diverse dimensioni del display. L'esecuzione sistematica e controllata dei test ha permesso di verificare che l'utente riuscisse ad interagire con i componenti che caratterizzano una vista e che l'ui corrispondesse a quanto progettato nello user flow diagram.

Conclusioni

L'applicazione sviluppata risulta complessivamente aderente agli obiettivi stabiliti in fase di progettazione. Dal rilascio è stato possibile notare un particolare apprezzamento da parte dei dipendenti dell'azienda. Molti hanno iniziato ad utilizzare solo lo smartphone per gestire le presenze e orari di lavoro, trovando il suo utilizzo più veloce e immediato rispetto al sito web. Lo studio e la realizzazione di UI testing ha permesso di introdurre questa attività nel lavoro quotidiano del team in modo più strutturato, considerando che questa tipologia di test non veniva svolta finora.

Gli sviluppi futuri del seguente progetto prevedono la realizzazione della stessa applicazione per dispositivi iOS, utilizzando il framework 4WSPlatform.

