

MyBookShelf App - Report



Lorenzo Cavallucci

0000915669

lorenzo.cavallucci4@studio.unibo.it

Sviluppo di una app iOS per la digitalizzazione della libreria personale.

Indice

1	Introduzione	2
2	Funzionalità e utilizzo	3
2.1	La mia libreria	3
2.2	Aggiunta di libri	4
2.3	Dettagli di un libro	6
2.4	Home Page	7
2.5	Profilo/Impostazioni	8
3	Tecnologie	10
4	Apple Watch Companion App	11
5	Mockup	12

1 Introduzione

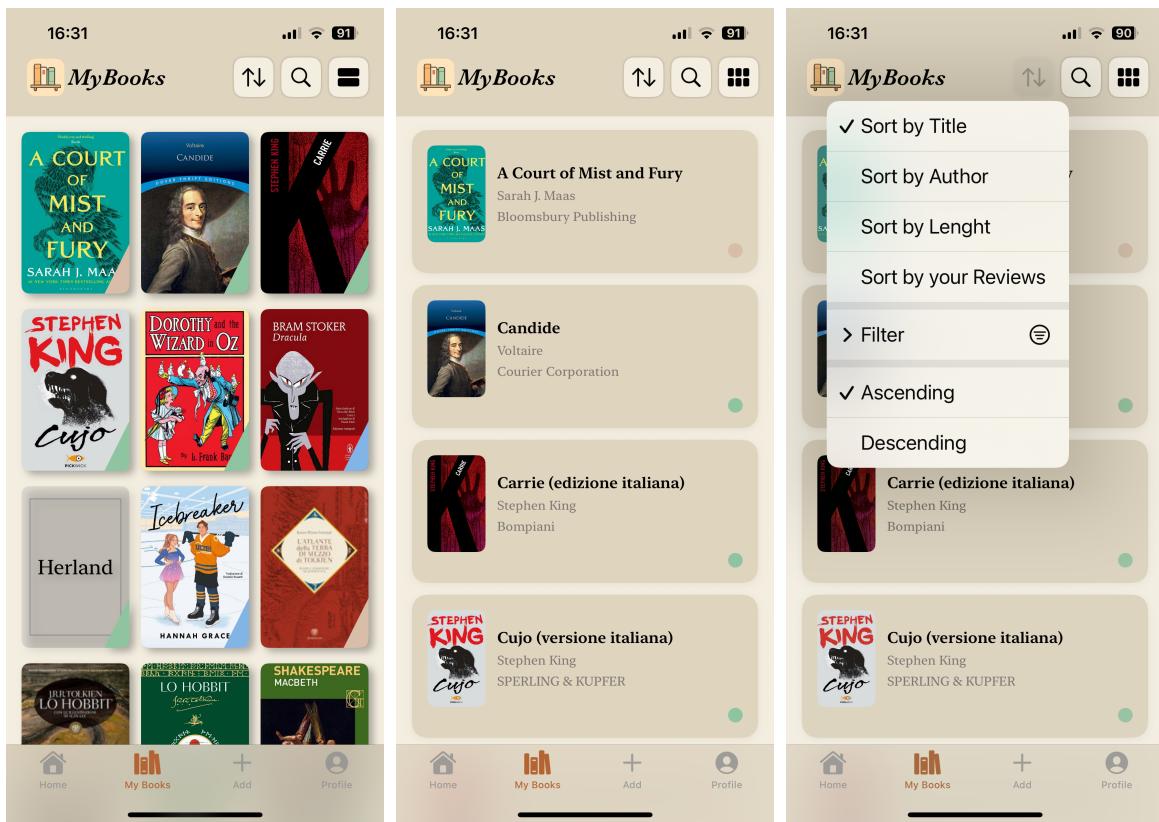
MyBookShelf nasce come applicazione nativa iOS con l'obiettivo di digitalizzare la libreria fisica di un utente, fornendo, in aggiunta, funzionalità varie per invogliare l'eventuale aspetto social di una app simile e fornire comodità per svariati casi d'uso.

L'idea è quindi quella di mettere a disposizione dell'utente una vera e propria libreria virtuale, su cui tenere traccia dei libri di cui si è in possesso, offrendo anche funzionalità quali la scoperta di nuovi libri da leggere, dettagli sui libri e tanto altro.

2 Funzionalità e utilizzo

2.1 La mia libreria

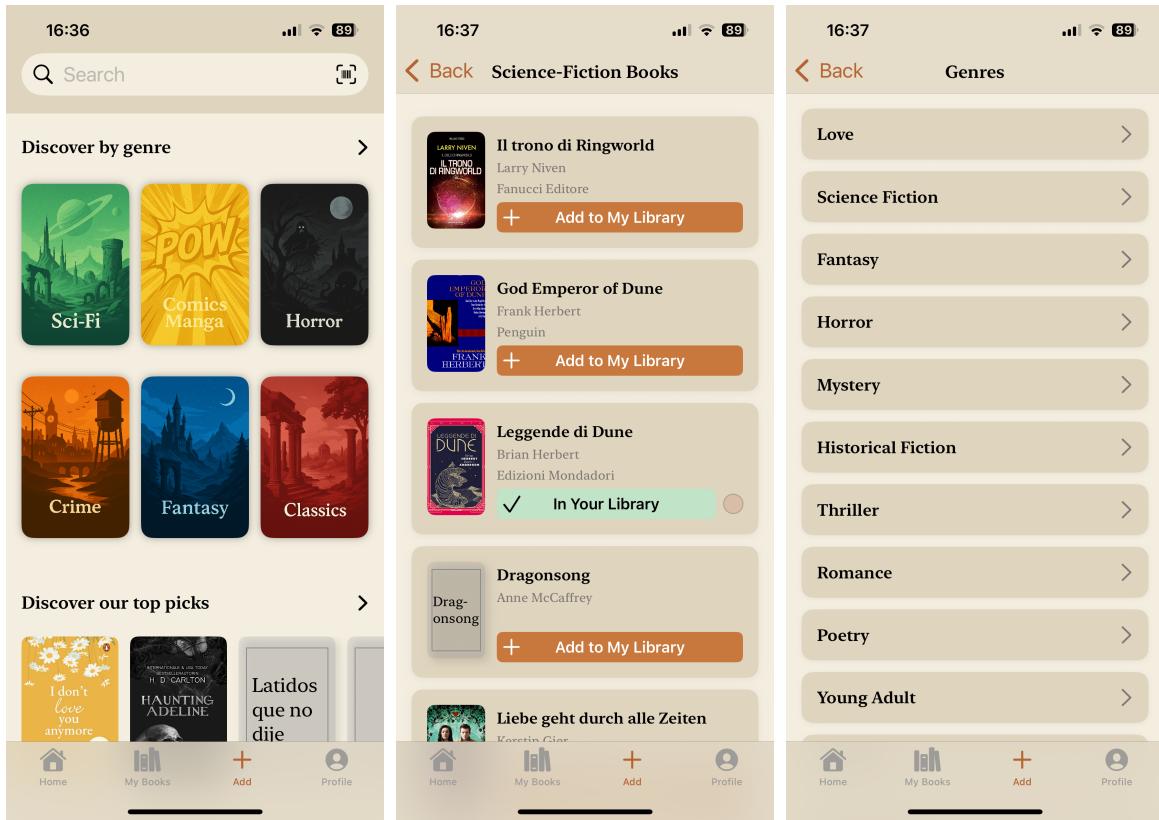
L'utente ha a disposizione la propria libreria virtuale (accessibile dal menu "My Books"), con due modalità di visualizzazione: una dedicata ai dettagli quali copertina, titolo, autore/i e editore; mentre una seconda "a griglia" che presenta solo la copertina dei libri, pensata per offrire un'estetica accattivante e soddisfacente dei libri posseduti.



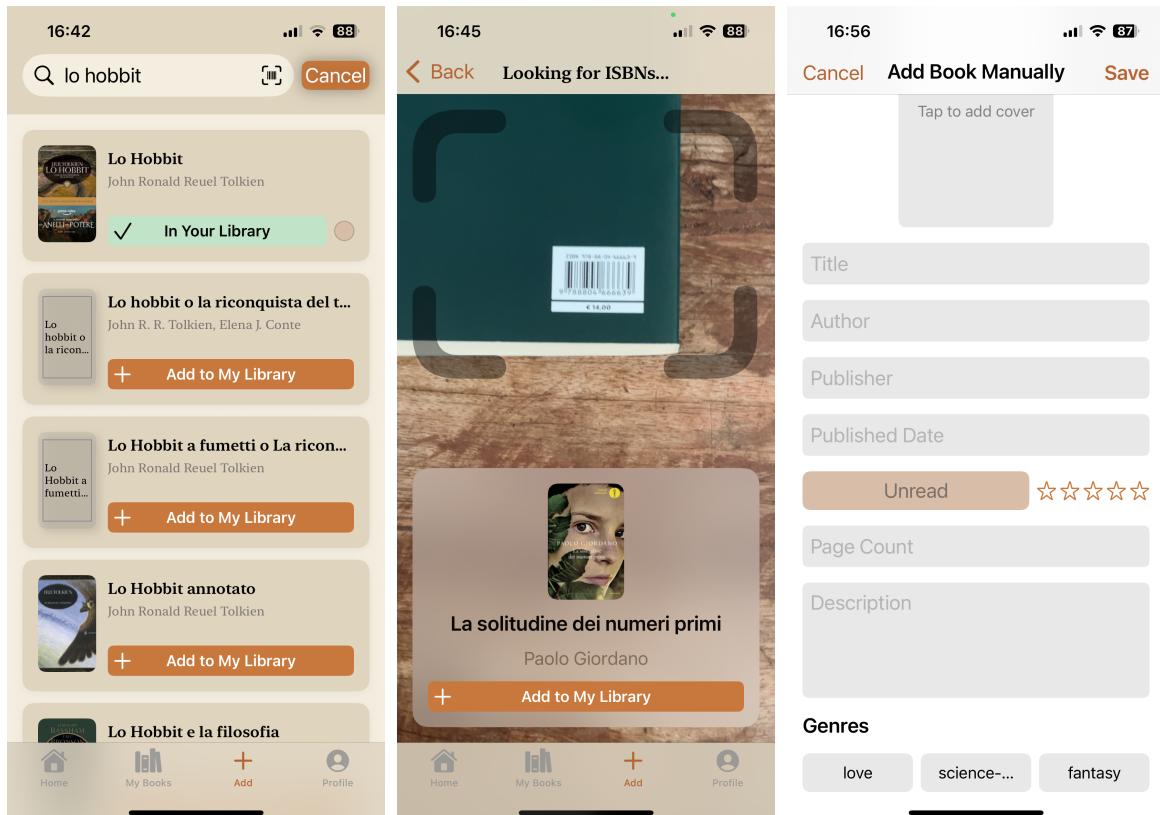
Qui, i libri possono essere visualizzati impostando determinati filtri (genere, stato di lettura, preferiti, ecc...) e ordinamenti (per titolo, autore, lunghezza, ecc...) per permettere una comodità maggiore all'utente in cerca di libri specifici; in aggiunta a una funzione di ricerca all'interno della libreria personale stessa.

2.2 Aggiunta di libri

L'aggiunta di libri alla raccolta personale avviene principalmente dalla pagina "Add" del menù, nella quale è possibile cercare libri in diversi modi:

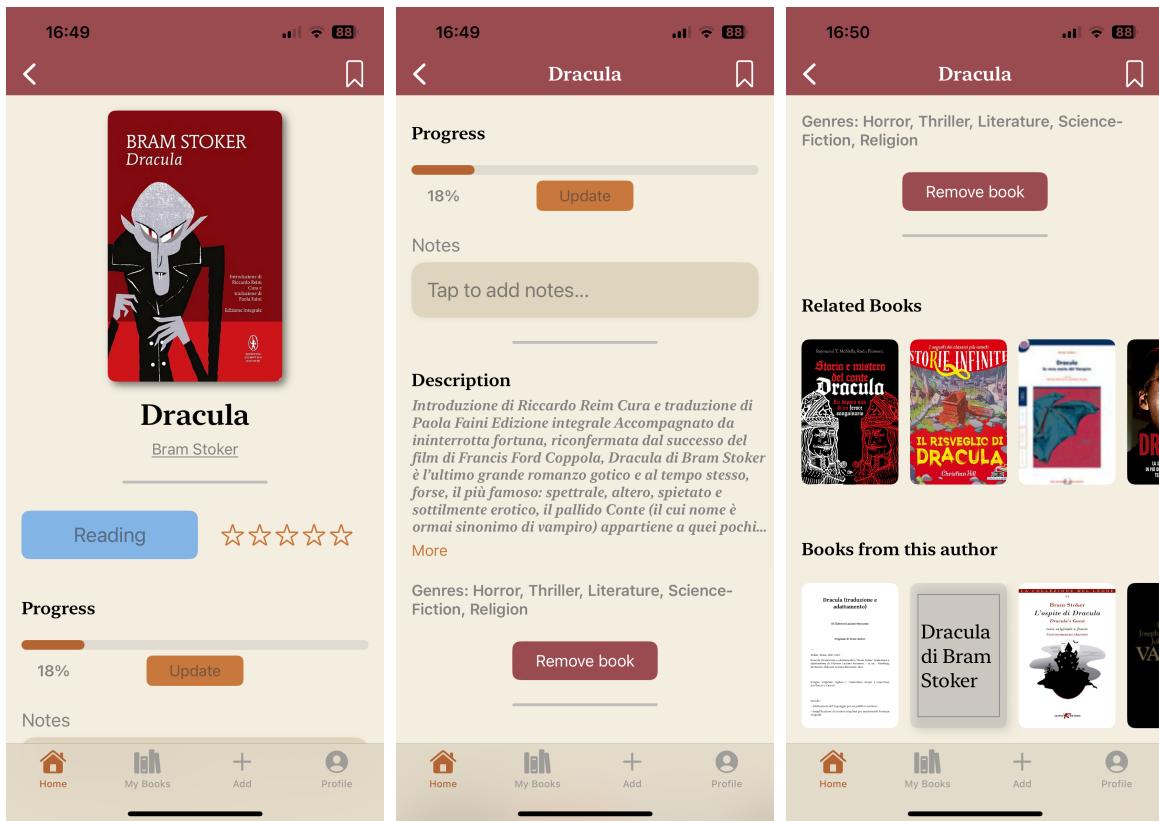


- **Aggiunta per genere** - Si può scegliere da una vasta gamma di generi e scoprire nuovi libri inerenti, salvandoli direttamente dalla lista di risultati;
- **Top Picks** - Vengono forniti una serie di libri "trending" al momento, o da una selezione di Best Seller;
- **Ricerca per nome, autore, ISBN** - Tramite una ricerca testuale è possibile trovare i libri desiderati e aggiungerli direttamente alla libreria privata;
- **ISBN Scanner** - E' disponibile anche la funziona di ricerca tramite ISBN scannerizzando con la fotocamera del dispositivo i bar code identificativi presenti nei libri fisici;
- **Aggiunta Manuale** - Si fornisce anche, all'utente, la possibilità di aggiungere manualmente un libro e i suoi dettagli nel caso in cui il database di libri fornito non dovesse restituire risultati coerenti, o semplicemente per inserire libri non mappati tradizionalmente.



2.3 Dettagli di un libro

Selezionando un libro, sia dalla libreria personale, che dai risultati di una ricerca di libri dal database, vengono forniti i dettagli dello stesso. Qui, abbiamo una visualizzazione diversa sulla base della presenza o meno di un libro tra quelli salvati: nel caso di un libro salvato, saranno mostrate informazioni esaustive, quindi copertina, titolo, stato di lettura (modificabile), eventuale recensione (numerica) dei libri, presenza o meno nei preferiti, aggiunta di una sessione di lettura (inizio e fine del periodo di lettura del libro), descrizione, autori, generi, note dell'utente e libri inerenti/ dello stesso autore; nel caso in cui non sia invece presente, vi sarà la possibilità di salvarlo, con in aggiunta informazioni basilari quali titolo, copertina, autori, descrizione e generi, con sempre libri correlati mostrati in fondo.



2.4 Home Page

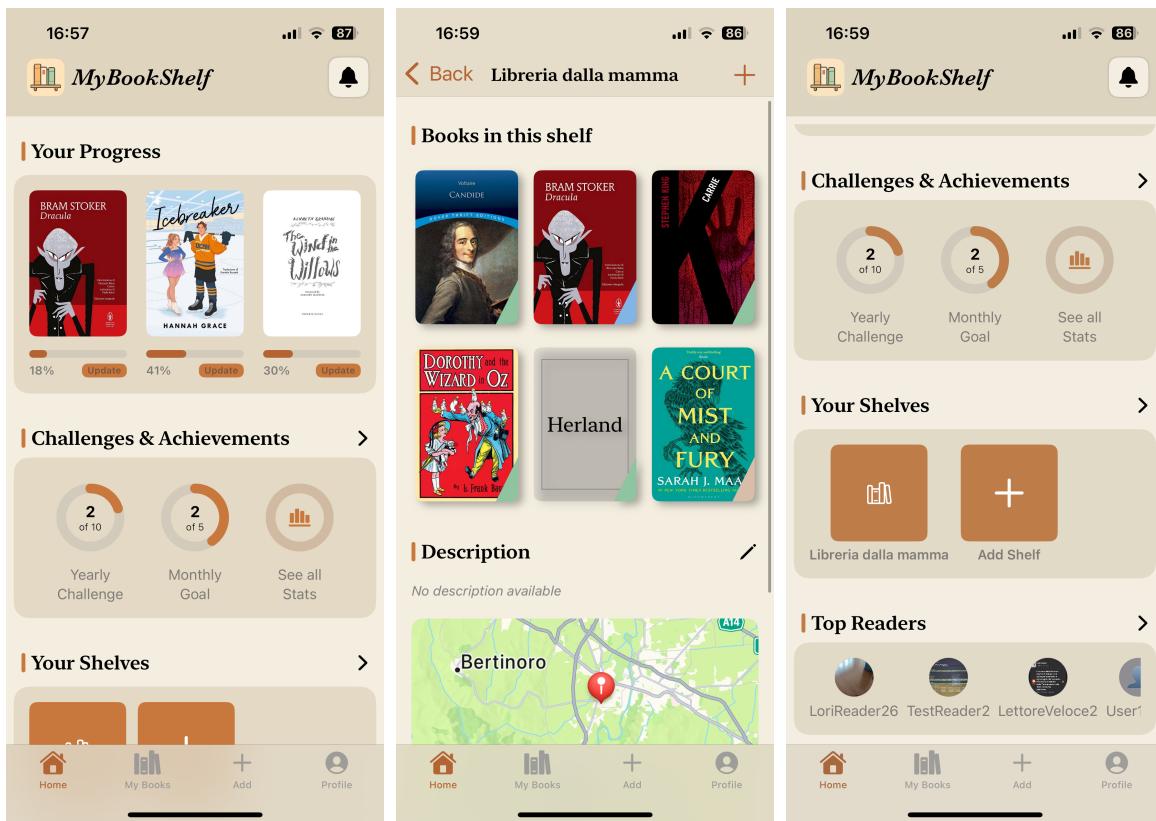
La home page della app fornisce comodità all'accesso di diverse funzioni, oltre che a dare il benvenuto all'utente al lancio, fornendo quindi una overview delle possibilità offerte dalla app.

Vengono quindi mostrati i libri cui stato di lettura è "Reading", ovvero che l'utente sta leggendo correntemente, con visualizzazione di una barra del progresso - cui click porta direttamente all'aggiornamento del progresso di lettura di un libro dalla sua pagina di dettaglio.

E' anche presente un'anteprima delle statistiche e obiettivi dell'utente, la cui navigazione ne porta a una visualizzazione dettagliata, compresa di Goal periodici (mensili/annuali), livello del profilo con XP, statistiche varie e grafici di andamento.

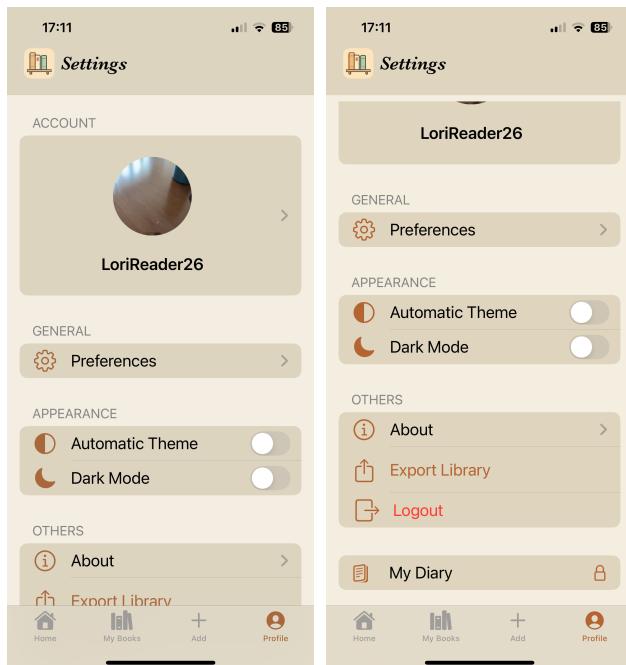
Altra funzionalità a disposizione dell'utente è quella delle "Shelves", ovvero delle raccolte identificate da un nome e una posizione, contenenti alcuni dei libri (ovviamente selezionabili) salvati. L'idea è quella di creare delle librerie virtuali aggiuntive in cui, a discrezione dell'utente, segnare quali libri sono dove, magari ubicati in abitazioni diverse.

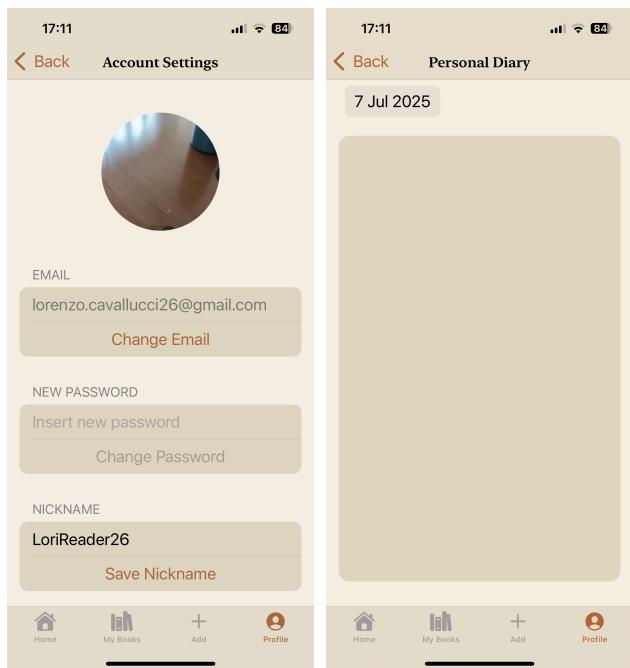
E' infine possibile avere una vista sulla classifica dei migliori lettori, ovvero gli utenti ordinati per livello.



2.5 Profilo/Impostazioni

E' ovviamente presente per l'utente l'accesso ad una pagina di impostazioni, dove è possibile modificare i dati dell'account, impostazioni di visualizzazione (tema chiaro/scuro) e permessi della app (con possibilità di accedere direttamente alle impostazioni per modificarli). E' inoltre presente una funzione di export della libreria, con scelta di formato in cui esportare, e una di diario personale ("My Diary"), nella quale si ha a disposizione un vero e proprio diario privato (accessibile solo tramite faceID/touchID) su cui scrivere qualsiasi tipo di pensiero, giorno per giorno.





3 Tecnologie

Nella realizzazione della app (interamente in Swift), sono state utilizzate diverse librerie e API. Innanzitutto, per il fetch dei libri, la app sfrutta principalmente Google Books API per l'ottenimento di risultati da una ricerca e per i vari dati di dei libri (copertina, autore, titolo, ecc). Durante lo sviluppo sono state svelate delle debolezze nella API di cui sopra, ovvero l'essere poco dettagliata per quanto riguarda il tag dei libri tramite generi e presentare delle immagini di copertina di qualità non accettabile in diversi casi. Per questo è stato implementato un sistema a doppio fetch, sfruttando la API offerta da Open Library, permettendo di avere una opzione di fallback nel caso in cui una delle due non desse risultati, oltre che a garantire immagini di copertina di qualità maggiore e un sistema di tagging per genere dettagliato e che ha poi permesso anche la realizzazione della scoperta di libri "top picks".

Sono inoltre state sfruttate, per la persistenza dei dati, sia Swift Data, che FireBase. Anche in questo caso le due tecnologie vengono usate in maniera combinata: la prima, permette il salvataggio in locale di determinati dati (come magari aggiornamento di dettagli relativamente a un libro salvato) che vengono poi propagati a FireBase firestore, nel quale è presente una collezione di utenti, ciascuno con delle sottocollezione contenenti i dati necessari per permettere ad un user di loggare col proprio profilo nella app e caricare tutti i suoi dati già presenti. L'uso di Swift Data garantisce anche casi di aggiornamento dei dati offline, di modo tale che, nel momento in cui l'utente torna online queste modifiche vengono riportate anche sul database online.

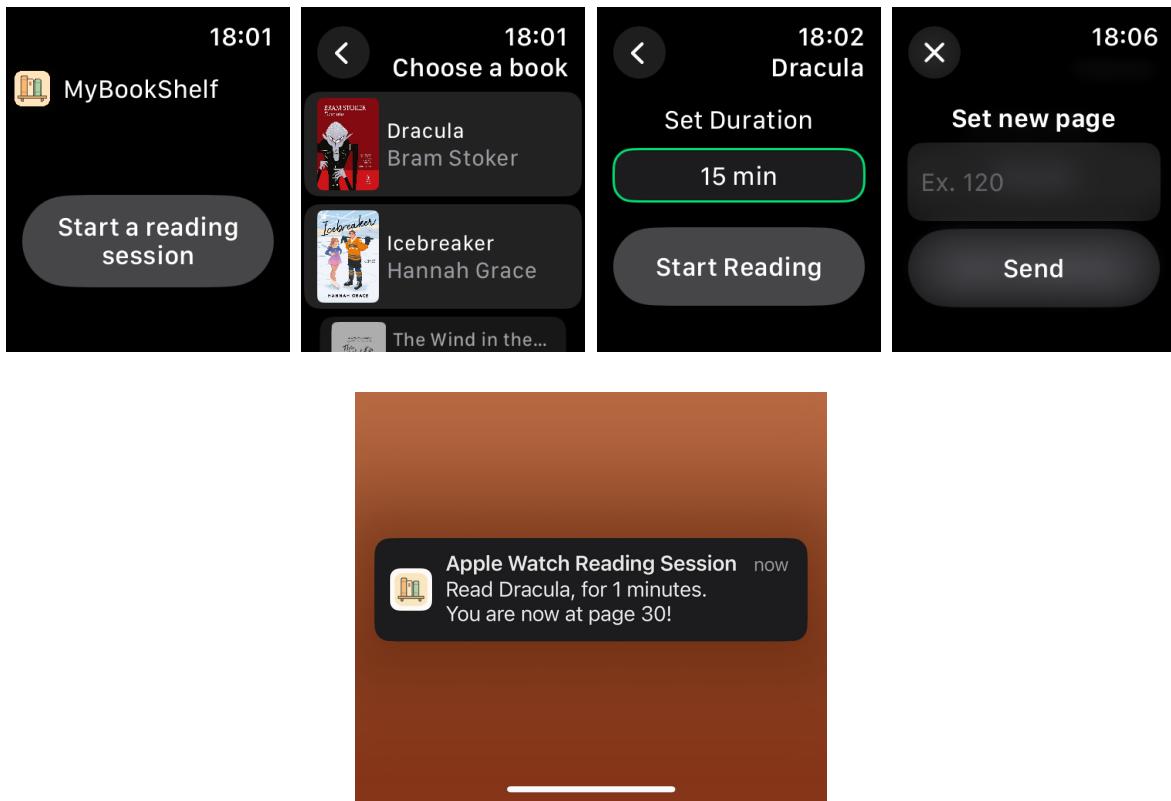
La comodità aggiuntiva offerta dal FireBase si trova nel non dover salvare localmente ogni singolo dato, alleggerendo lo spazio occupato dalla app sul dispositivo.

Anche l'autenticazione è gestita tramite FireBase, con sistema a coppia di email - password, permettendo però anche il login tramite Sign in con account Google.

Sono state usate poi altre librerie come Charts (per la realizzazione di grafici in merito alle statistiche), MapKit (per gestire la posizione delle Shelves, con posizione in tempo reale dell'utente) e WatchConnectivity per la connessione alla companion App per Apple Watch (vedi sezione 4).

4 Apple Watch Companion App

E' possibile sfruttare una app Companion pensata per Apple Watch, la cui funzione è quella di permettere di avviare delle sessioni di lettura in relazione a un libro (il cui stato è "Reading", presi dalla app iOS) per un determinato periodo di tempo. Al termine del timer, viene richiesto di aggiornare le pagine lette, e all'invio viene, appunto, inviata una notifica alla app iOS, la quale aggiornerà lo stato di lettura del libro in questione, sincronizzandosi poi con FireBase Firestore oltre che in locale con Swift-Data.



5 Mockup

