

MatchDay
Test Cases
Versione 0.1



MatchDay

Data: 13/12/2024

Progetto: Nome Progetto	Versione: 0.1
Documento: Titolo Documento	Data: 13/12/2024

Nome	Matricola

Nome	Matricola
Vincenzo Vitale	0512113542
Nicola Moscufo	0512114886
Francesco Moscufo	0512115027

Scritto da:	Vincenzo Vitale, Nicola Moscufo, Francesco Moscufo
--------------------	--

[illegible]

Sommario

1. TEST CASES.....5

1.1. Gestione Prenotazioni5

1.2. Gestione Campo6

1.3. Gestione Slot8

1.4. Gestione Utente.....10

1. TEST CASES

1.1. Gestione Prenotazioni

○ Test Case Specification Identifier

ID	Nome del Test Case	Descrizione
TCS-001	Creazione Prenotazione	Testare la creazione di una prenotazione cliccando su uno slot disponibile
TCS-002	Modifica Prenotazione	Testare la modifica di una prenotazione esistente selezionando un nuovo slot
TCS-003	Cancellazione Prenotazione	Testare la cancellazione di una prenotazione esistente inviando una richiesta all'admin

○ Test Item

- ✚ **Creazione di una prenotazione:** L'utente seleziona uno slot disponibile e invia la richiesta di prenotazione all'admin
- ✚ **Modifica di una prenotazione:** L'utente accede alla lista delle prenotazioni, seleziona una prenotazione, sceglie un nuovo slot dal calendario e invia una nuova richiesta
- ✚ **Cancellazione di una prenotazione:** L'utente accede alla lista delle prenotazioni, seleziona la prenotazione da cancellare e invia una richiesta di cancellazione all'admin

○ Input Specification

Test Case	Input Dati
TCS-001	Slot disponibile selezionato, clicca su Prenota
TCS-002	Selezione della prenotazione della lista, clicca su Modifica , scelta del nuovo slot, clic su Prenota
TCS-003	Selezione della prenotazione della lista, clic su Cancella

○ Output Specifications

Test Case	Oracle
TCS-001	Messaggio di conferma: "La tua richiesta di prenotazione è stata inviata all'admin"
TCS-002	Messaggio di conferma: "La tua richiesta di modifica è stata inviata all'admin"

TCS-003	Messaggio di conferma: “cancellazione effettuata con successo”
---------	--

○ **Enviromental Needs** ➦ Software:

1. Flutter SDK
2. Firebase Firestore

○ **Special Procedural Requirements**

Casi da validare	
TCS-001	-Cliccando su uno slot prenotato non verrà inviata la richiesta di prenotazione all’admin -Verifica che se si clicca correttamente su prenota, arrivi correttamente la richiesta di prenotazione all’admin
TCS-002	-Impossibile modificare la prenotazione il giorno stesso della prenotazione originale -impossibile selezionare slot che presentato orari già superati -verificare se la richiesta di modifica della prenotazione arrivi correttamente all’admin
TCS-003	-Verificare se la cancellazione e’ stata effettuata con successo

○ **InterCase Dependencies**

Dipendenze	Descrizione
TCS-001 -> TCS-002	La modifica della prenotazione richiede una prenotazione creata con successo
TCS-001-> TCS-003	La cancellazione di una prenotazione richiede una prenotazione esistente

1.2. Gestione Campo

○ **Test Case Specification Identifier**

ID	Nome del Test Case	Descrizione
TCS-004	Aggiunta Campo	Testare l’aggiunta di un nuovo campo da parte dell’admin
TCS-005	Rimozione Campo	Testare la rimozione di un campo sportivo aggiunto da parte dell’admin

TCS-006	Modifica Campo	Testare la modifica delle informazioni di un campo sportivo esistente da parte dell'admin
---------	----------------	---

○ **Test Item**

- ✚ **Aggiunta Campo:** L'admin clicca sul pulsante “+” nella sua interfaccia, inserisce i dati richiesti e conferma l'aggiunta
- ✚ **Rimozione Campo:** L'admin seleziona un campo sportivo aggiunto e lo elimina cliccando su “Elimina”
- ✚ **Modifica Campo:** L'admin seleziona un campo esistente, modifica i dati e conferma le modifiche

○ **Input Specification**

Test Case	Input Dati
TCS-004	Clicca sul “+”, aggiunge i dati e clicca salva
TCS-005	L'admin seleziona il campo e clicca su elimina
TCS-006	L'admin seleziona il campo, clicca Modifica , clicca salva

○ **Output Specification**

Test Case	Oracle
TCS-004	Messaggio di conferma: “il campo è stato aggiunto con successo”
TCS-005	Messaggio di conferma: “il campo è stato modificato con successo”
TCS-006	Messaggio di conferma: “il campo è stato rimosso con successo”

○ **Environment Needs** ✚ Software:

1. Flutter SDK
2. Firebase Firestore

○ **Special Procedural Requirements**

Casi da validare	
------------------	--

TCS-004	-Tentativo di aggiungere un campo con dati incompleti o vuoi mostra messaggi di errore - Inserimento di dati non validi nei campi mostra messaggi di errore
TCS-005	-Rimozione di un campo: il campo scompare dalla lista dei campi disponibili
TCS-006	-Modifica di un campo con dati non validi: il campo non viene modificato e compare un
	messaggio di errore

○ **Intercase Dependencies**

Dipendenze	Descrizione
TCS-004->TCS-005	Un campo deve esistere prima di poter essere rimosso
TCS-004->TCS-006	La modifica di un campo richiede che il campo sia stato aggiunto correttamente
TCS-004->TCS-001	I campi aggiunti devono essere visibili e prenotabili dagli utenti

1.3. Gestione Slot

○ **Test Case Specification Identifier**

ID	Nome del Test Case	Descrizione
TCS-007	Aggiunta Slot	Testare l'aggiunta di uno slot orario associato a un campo
TCS-008	Modifica Slot	Testare la modifica degli orari di uno slot esistente
TCS-009	Rimozione Slot	Testare la rimozione di uno slot esistente del calendario

TCS-010	Modifica Stato Slot	Testare il cambio di stato di uno slot passando da prenotato quindi “false” a disponibile “true”
---------	---------------------	--

○ **Test Item**

- ✚ **Aggiunta Slot:** L'admin seleziona il calendario associato a un campo, sceglie un giorno e aggiunge uno slot cliccando sul pulsante "+" e inserendo orari specifici.
- ✚ **Modifica Slot:** L'admin seleziona uno slot esistente e modifica l'orario associato.
- ✚ **Rimozione Slot:** L'admin seleziona uno slot esistente e lo rimuove.
- ✚ **Modifica Stato Slot:**
 - Uno slot passa in stato “false” quando viene prenotato da un utente
 - Uno slot è in stato “true” quando uno slot è prenotabile e torna in questo stato quando l'admin accetta la modifica o cancellazione

○ **Input Specifications**

Test Case	Input Dati
TCS-007	L'admin seleziona il campo, seleziona il giorno nel calendario, clicca il “+” e aggiunge lo slot
TCS-008	L'admin seleziona lo slot poi modifica gli orari o il giorno e salva la modifica
TCS-009	L'admin seleziona il slot e clicca su elimina
TCS-010	L'utente prenota lo slot e quindi lo stato passa a “false”, dopo l'approvazione della modifica lo stato passa a “true”

○ **Output Specification**

Test Case	Oracle
TCS-007	Messaggio di conferma “slot aggiunto con successo”, slot visibile e prenotabile all'utente

TCS-008	Messaggio di conferma “slot modificato con successo”, slot visibile all’utente con le modifiche effettuate
TCS-009	Messaggio di conferma “slot rimosso con successo”, slot non più visibile all’utente
TCS-010	Cambio di stato nel database e cambio di stato all’interno degli slot presenti nel calendario

○ **Enviromental Needs ➦ Software:**

1. Flutter SDK
2. Beckend configurato per: aggiunta, modifica, rimozione e aggiornamento dello stato degli slot
3. Firebase Firestore

○ **Special Procedural Requirements**

Casi da validare	
TCS-007	-Provare a inserire slot con orari non validi mostra un messaggio di errore -Impossibile inserire slot in giorni o orari passati
TCS-008	-Modifica degli slot con orari che sono stati già superari porta messaggi di errore - Verificare l’impossibilità di modificare lo slot se risulta prenotato
TCS-009	-Rimozione di uno slot: lo slot scompare dal calendario
TCS-010	-Cambiando lo stato dello slot da “true” a “false”, lo slot non può più essere prenotato

○ **Intercase Dependencies**

Dipendenze	Descrizione
TCS-007->TCS-008	Uno slot deve esistere prima di poter essere modificato
TCS-007->TCS-009	Uno slot deve esistere prima di poter essere rimosso
TCS-007->TCS-010	Uno slot deve essere “true” prima di poter esser prenotato

1.4. Gestione Utente

- **Test Case Specification Identifier**

ID	Nome del Test Case	Descrizione
TCS-011	Registrazione Utente	Testare la registrazione di un nuovo utente nel sistema
TCS-012	Login Utente	Testare la funzionalità di login per un utente già registrato

- **Test Items**

- ✦ **Registrazione Utente:** Un nuovo utente si registra creando un account con le proprie credenziali (email, password, nome, cognome, numero di telefono)
- ✦ **Login Utente:** Un utente esistente inserisce email e password per accedere

- **Input Specifications**

Test Case	Input Dati
TCS-011	L'utente inserisce una email valida, password secondo i parametri, nome, cognome e numero di telefono
TCS-012	L'utente utilizza email e password utilizzati precedentemente per la registrazione

- **Output Specification**

Test Case	Oracle
TCS-011	L'utente ha inserito correttamente i suoi dati e verrà reindirizzato alla home page
TCS-012	L'utente ha inserito correttamente i suoi dati utilizzati in precedenza nella registrazione e verrà reindirizzato alla home page

- **Environments Needs ✦ Software:**

1. Flutter SDK
2. Firebase Firestore
3. Firebase Auth
4. ○

Special procedural Requirements

Casi da validare	
------------------	--

TCS-011	<ul style="list-style-type: none"> -Verificare che la password soddisfi i requisiti minimi -Verificare che la email non sia già associata a un altro account -Verificare che l'account venga aggiunto correttamente al database con tutte le informazioni -Verifica che l'utente venga reindirizzato alla home page dopo la registrazione
TCS-012	<ul style="list-style-type: none"> - Verifica che l'utente possa accedere solo con email e password corretti -Verifica che se l'utente inserisce credenziali errate, venga mostrato il messaggio di errore "Email o password errati"

○ **Intercase Dependencies**

Dipendenze	Descrizione
TCS-011->TCS-012	La registrazione deve essere stata effettuata con successo prima di poter effettuare il login
TCS-012->TCS-011	Il login è possibile solo se l'account è stato registrato correttamente