

Università degli Studi di Salerno

Corso di Ingegneria del Software

HairBeautyNow

TP: Test Plan

Versione 2.0



Data: 11/02/2025

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

Coordinatore del progetto:

Nome	Matricola

Partecipanti:

Nome	Matricola
Aniello Cirillo	0512117775
Christian Pio Lanziero	0512117931

Scritto da:	
-------------	--

Revision History

Data	Versione	Descrizione	Autore
10/11/2024	0.1	Creazione del documento, indice	Tutto il team
13/12/2024	0.2	Panoramica del sistema, pass/fail criteria	Tutto il team
16/12/2024	1.0	Approccio, Sospensione e ripresa, Testing materials, Testing schedule e Prima release	Tutto il team
11/02/2025	2.0	Revisione dell'intero documento, aggiornate le caratteristiche da testare, aggiornata la Testing Schedule.	Tutto il team

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

Indice

1. INTRODUZIONE.....	4
2. RELAZIONE CON ALTRI DOCUMENTI.....	4
3. PANORAMICA DEL SISTEMA.....	4
4. CARATTERISTICHE DA TESTARE / NON TESTARE.....	5
4.1 Caratteristiche da testare.....	5
4.2 Caratteristiche da non testare.....	6
5. PASS / FAIL CRITERIA.....	6
6. APPROCCIO.....	6
7. SOSPENSIONE E RIPRESA.....	7
8. TESTING MATERIALS.....	8
9. TEST CASES.....	8
10. TESTING SCHEDULE.....	8
11. GLOSSARIO.....	9

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

1. INTRODUZIONE

Il presente **Test Plan** definisce le strategie, le metodologie e le risorse necessarie per testare il sistema **HairBeautyNow**. L'obiettivo principale è verificare che il software soddisfi tutti i requisiti funzionali e non funzionali descritti nei documenti [RAD_HairBeautyNow](#), [SDD_HairBeautyNow](#) e [ODD_HairBeautyNow](#). Il piano di test mira a identificare eventuali errori, anomalie o malfunzionamenti attraverso test unitari, di integrazione e di sistema, garantendo così un prodotto finale stabile e conforme alle esigenze del cliente.

HairBeautyNow è un sistema progettato per la gestione dei servizi e delle prenotazioni nei saloni di bellezza, oltre all'amministrazione della rete di franchising. Per assicurare il corretto funzionamento delle sue funzionalità, verranno eseguite attività di test mirate a valutarne precisione ed efficienza. I test seguiranno approcci bottom-up e black-box, consentendo di esaminare sia i singoli moduli sia le loro interazioni all'interno del sistema.

2. RELAZIONE CON ALTRI DOCUMENTI

- [RAD_HairBeautyNow](#): Descrive le funzionalità individuate in fase di analisi, fornendo dettagli sui requisiti funzionali e non funzionali (Utili per il Test Plan)
- [SDD_HairBeautyNow](#): Descrive l'architettura completa del sistema, dei sottoinsiemi individuati e dei servizi offerti da questi ultimi. Bisogna quindi effettuare dei test per ogni servizio offerto dai sottoinsiemi.
- [ODD_HairBeautyNow](#): Descrive le classi, i metodi e le interfacce. Nel Test Plan si andranno a verificare tutte le classi effettuando dei test unitari di integrazione con i sottosistemi.

3. PANORAMICA DEL SISTEMA

Il sistema **HairBeautyNow** è stato progettato per gestire le operazioni relative alla prenotazione dei servizi offerti da una catena di saloni di bellezza, alla gestione delle sedi e alla pianificazione degli appuntamenti. L'architettura proposta, come delineato nel **System Design Document (SDD)**, adotta un modello **three-tier**, che separa in modo chiaro i livelli di presentazione, logica di business e persistenza dei dati. Questa struttura favorisce una maggiore manutenibilità e modularità del sistema, rendendo più semplici i test e gli aggiornamenti.

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

Il **Presentation Layer** del sistema **HairBeautyNow** è dedicato all'interazione con l'utente. Tramite un'interfaccia grafica responsive, sviluppata con tecnologie come **JSP**, **HTML5**, **CSS** e **JavaScript**, gli utenti possono accedere alle funzionalità principali del sistema, tra cui: autenticazione, selezione dei saloni, scelta dei professionisti e orari disponibili, visualizzazione dello storico delle prenotazioni, inserimento di nuove programmazioni e creazione di nuove sedi. Le **JSP**, denominate seguendo il formato **camelCase**, sono progettate per interagire con il livello applicativo e gestire dinamicamente i dati visualizzati.

L'**Application Layer** rappresenta la logica di business del sistema, fungendo da intermediario tra il **Presentation Layer** e il **Data Layer**. Questo livello si occupa dell'elaborazione delle richieste degli utenti e del coordinamento delle operazioni. Include **controller** che gestiscono operazioni principali, come la gestione delle prenotazioni e dei profili utenti. Inoltre, questo livello comprende il **Facade**, che centralizza le operazioni di validazione, come la verifica della correttezza dei dati inseriti (ad esempio, formato della password, disponibilità dei saloni, ecc.). La connessione con il database è ottimizzata tramite un **DataSource Singleton**, che garantisce un accesso efficiente alle risorse del database.

Il **Data Layer**, infine, è responsabile della persistenza dei dati. Questo livello utilizza un database relazionale **MySQL** per archiviare in modo strutturato le informazioni relative agli utenti, alle prenotazioni, ai saloni, ai professionisti e ai servizi offerti. Le operazioni di lettura e scrittura sono gestite da classi **DAO**, come **PrenotazioneDAO**, **UtenteDAO**, **ServizioDAO**, ecc., che si interfacciano direttamente con il database per assicurare che i dati siano correttamente salvati e recuperati. Il **Presentation Layer** invia le richieste al **Application Layer**, che le elabora e interagisce con il **Data Layer** per accedere o salvare i dati richiesti.

4. CARATTERISTICHE DA TESTARE / NON TESTARE

4.1 Caratteristiche da testare

- ❖ **UtenteAcquirente**
 - Autenticazione
 - Prenotazione
- ❖ **UtenteGestoreSede**
 - Aggiunta di un professionista.
- ❖ **UtenteGestoreCatena**
 - Aggiunta di una sede.
- ❖ **UtenteGuest**
 - Registrazione

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

Per motivi di priorità e risorse limitate, alcune caratteristiche minori (indicate con priorità media/bassa nei requisiti funzionali del RAD) non saranno oggetto di test specifici. Le caratteristiche non testate non influiscono sulle operazioni principali del sistema e saranno eventualmente prese in considerazione in fasi successive di miglioramento del progetto.

5. PASS / FAIL CRITERIA

L'obiettivo del **testing** nel sistema **HairBeautyNow** è assicurarsi che tutte le funzionalità implementate rispettino i requisiti funzionali e non funzionali, garantendo che il software operi correttamente e generi risultati coerenti con gli input ricevuti. Il processo di testing consiste nel confrontare i risultati attesi con quelli effettivamente ottenuti durante l'esecuzione delle operazioni. Se i due valori coincidono, il test sarà considerato superato; in caso contrario, verrà classificato come fallito.

Gli errori vengono individuati confrontando i risultati con un oracolo, ossia un riferimento che definisce il comportamento atteso del sistema. Qualora il software non rispetti i criteri prestabiliti, sarà necessario correggere il codice, rieseguire i test e verificare che tutti i requisiti siano soddisfatti.

6. APPROCCIO

L'approccio al **testing** per **HairBeautyNow** è suddiviso in tre fasi principali:

1. **Testing unitario:** La prima fase è focalizzata sul controllo delle singole classi e dei metodi del sistema. In questa fase si adotta una metodologia di testing **black-box**, concentrandosi esclusivamente sugli input e sugli output delle componenti, senza entrare nel dettaglio della loro implementazione interna. Per gestire la varietà dei possibili input, viene utilizzata la tecnica di **category partition**, che suddivide i dati in categorie strutturate, ognuna delle quali rappresenta una classe di equivalenza. Ogni classe viene testata con un caso di test specifico, garantendo così una copertura completa senza ridondanze. Questo approccio permette di identificare eventuali anomalie nelle singole componenti, che vengono quindi corrette e ri-testate per garantire il rispetto dei requisiti.
2. **Testing di integrazione:** La seconda fase si concentra sulla verifica delle interazioni tra le diverse componenti del sistema. In questa fase, l'obiettivo è assicurarsi che i moduli collaborino correttamente tra loro, senza generare errori. Le interazioni tra le principali classi, come quelle che gestiscono le prenotazioni, gli utenti e i professionisti, vengono testate insieme. Anche in questa fase viene applicata la tecnica di **category partition**, per identificare le combinazioni significative di

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

input e ottimizzare i casi di test.

3. **Testing di sistema:** L'ultima fase prevede una verifica complessiva del sistema **HairBeautyNow**. Questa fase simula scenari reali e comportamenti degli utenti, considerando sia le funzionalità principali (come la registrazione, la prenotazione dei servizi e la gestione delle promozioni) sia gli aspetti non funzionali, come le prestazioni e la sicurezza. Il **testing di sistema** ha lo scopo di confermare che il prodotto finale rispetti tutti i requisiti stabiliti nei documenti di analisi e progettazione (RAD, SDD, ODD). Durante questa fase, il sistema viene testato in un ambiente che replica quello di produzione, in modo da rilevare eventuali problemi prima del rilascio definitivo.

7. SOSPENSIONE E RIPRESA

La fase di testing del sistema **HairBeautyNow** sarà sospesa nel caso si verifichino condizioni che impediscono il proseguimento delle attività di test.

Le principali motivazioni per la sospensione includono:

- **Rilevazione di bug critici o limitanti:** Qualsiasi malfunzionamento che comprometta le funzionalità principali, come il crash del sistema o anomalie nella gestione delle prenotazioni o delle informazioni degli utenti, che impediscono l'esecuzione dei test programmati.
- **Modifiche al sistema non integrate:** Eventuali aggiornamenti o modifiche al codice che non siano stati sufficientemente testati, integrati o documentati.

Per riprendere le attività di testing sarà necessario:

- **Correggere i bug identificati:** Una volta risolti i problemi, si procederà con test di regressione per verificare che le correzioni non abbiano introdotto nuove anomalie o compromesso altre funzionalità del sistema.

8. TESTING MATERIALS

Per il testing del sistema **HairBeautyNow**, saranno utilizzati strumenti software specifici per garantire la qualità delle funzionalità implementate e per assicurarsi che i requisiti funzionali e non funzionali siano soddisfatti. Di seguito sono descritti gli strumenti principali che verranno impiegati durante il processo di testing:

❖ Selenium

➤ **Selenium** sarà utilizzato per l'automazione dei test dell'interfaccia utente (UI). Questo strumento permetterà di simulare le azioni dell'utente, come il login, la registrazione, la prenotazione dei servizi e la

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

gestione delle informazioni dei professionisti. I test automatizzati garantiranno che le interazioni tra l'utente e il sistema siano corrette e che siano ripetibili su diversi browser. Selenium sarà particolarmente utile per individuare eventuali anomalie nell'interfaccia grafica e per assicurare che il flusso dell'utente sia sempre coerente.

❖ JUnit

➤ **JUnit** sarà il framework principale per i test unitari e di integrazione. Con **JUnit**, verranno testati i metodi delle classi DAO, i controller e le logiche di business, come la gestione delle prenotazioni, degli utenti e dei professionisti. I test unitari assicureranno che ogni singolo componente funzioni come previsto, mentre i test di integrazione verificheranno il corretto funzionamento delle interazioni tra le varie componenti del sistema. Inoltre, **JUnit** fornirà report dettagliati, utili per analizzare i risultati dei test e individuare eventuali errori o incongruenze rispetto ai requisiti definiti.

❖ Mockito

➤ **Mockito** sarà utilizzato per il testing di unità e di integrazione. Grazie a Mockito, sarà possibile validare il corretto funzionamento delle interazioni tra diverse componenti del sistema, come ad esempio tra le classi DAO e il database. L'utilizzo di Mockito renderà più semplice isolare e analizzare eventuali errori nei moduli integrati.

9. TEST CASES

9.1 Panoramica

La Sezione 9, cuore del piano di test, descrive i casi di test utilizzati durante il processo di validazione. Ogni caso di test è dettagliatamente descritto in un documento separato di [Test Case Specification](#).

10. TESTING SCHEDULE

Il programma di testing per il sistema **HairBeautyNow** è stato pianificato con l'obiettivo di assicurare un processo di verifica completo, strutturato e allineato alle tempistiche di sviluppo. I test saranno suddivisi in tre fasi principali, ciascuna focalizzata su un livello diverso del sistema:

- **Testing Unitario (1 settimana):** Verifica delle singole classi e metodi implementati, utilizzando JUnit per garantire che ogni unità funzioni correttamente in base ai requisiti. I test saranno eseguiti in parallelo con i test di Integrazione, per velocizzare i tempi di sviluppo.
- **Testing di Integrazione (1 settimana):** Test delle interazioni tra i moduli, inclusi i collegamenti tra i DAO e il database tramite il DataSource, e le interazioni tra le classi di gestione e le JSP, verificando le dipendenze tra i vari componenti del sistema.

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025

- **Testing di Sistema (2-3 giorni):** Verifica dell'intero sistema, comprendendo test su funzionalità e prestazioni. Con Selenium, saranno simulati scenari reali per operazioni come la registrazione degli utenti, la prenotazione dei servizi e la gestione delle prenotazioni..

Una volta completato lo sviluppo, tutti i test saranno rieseguiti come verifica finale per garantire il corretto funzionamento del sistema e per produrre i documenti conclusivi. Ogni fase di test sarà seguita da una revisione approfondita per correggere eventuali errori prima del passaggio alla fase successiva, assicurando così un rilascio privo di bug critici.

11. GLOSSARIO

Utente Generale	Rappresenta un utente non autenticato nel sistema, sono presenti quindi restrizioni nell'utilizzo di quest'ultimo.
Utente Acquirente	Rappresenta un utente autenticato nel sistema, che quindi può sfruttare le funzionalità riguardanti le prenotazioni e lo storico degli ordini.
Utente Gestore Sede	Rappresenta un utente registrato con il compito di gestire una singola sede dell'intera catena, esso può gestire le prenotazioni e i professionisti in salone.
Utente Gestore Catena	Rappresenta un utente registrato, che gestisce l'intera catena HairBeautyNow, ha il compito di amministrare le sedi, gli Utenti Gestori Sede e di creare e attivare Promozioni in tutta la catena.
Prenotazione	Rappresenta l'acquisto di un servizio con relative informazioni riguardo la sede, il professionista scelto e il servizio.
Franchise	Un sistema di collaborazione tra un produttore e un distributore, ove il primo cede al secondo la facoltà di distribuire il servizio con alcune definizioni contrattuali.

Progetto: HairBeautyNow	Versione: 2.0
Documento: Test Plan	Data: 11/02/2025