

Università degli Studi di Salerno

Corso di Ingegneria del Software

HairBeautyNow

TP: Test Plan

Versione 1.0



Data: 10/11/2024

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

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

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

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.....	10

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

1. INTRODUZIONE

Il presente documento di **Test Plan** descrive le strategie, i metodi e le risorse necessarie per effettuare il testing del sistema HairBeautyNow. L'obiettivo primario è assicurare che il software soddisfi tutti i requisiti funzionali e non funzionali specificati nei documenti [RAD_HairBeautyNow](#), [SDD_HairBeautyNow](#) e [ODD_HairBeautyNow](#). Il piano si propone di individuare eventuali difetti, anomalie o malfunzionamenti nel sistema attraverso test unitari, test di integrazione e test di sistema, al fine di garantire un prodotto finale stabile e conforme alle aspettative del cliente.

Il sistema HairBeautyNow, progettato per gestire i servizi e le prenotazioni nei saloni di bellezza e amministrare la rete del franchising, includerà attività di testing mirate a verificare l'accuratezza e l'efficienza delle funzionalità implementate. I test saranno condotti seguendo approcci **bottom-up** e **black-box**, al fine di valutare sia i singoli componenti sia le loro interazioni.

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.

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

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

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

- ❖ **Registrazione e autenticazione:**
 - Autenticazione basata su ruoli.
 - Verifica del formato email e della crittografia delle password.
- ❖ **Prenotazioni:**
 - Selezione del servizio, professionista e orari.
 - Finalizzazione del pagamento con validazione dei dati della carta.
 - Rimozione prenotazione.
- ❖ **Gestione delle sedi:**
 - Aggiunta e rimozione professionisti
- ❖ **Gestione della catena:**
 - Aggiunta e rimozione sedi
 - Aggiunta e rimozione gestore sede
 - Aggiunta e rimozione promozione

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

4.2 Caratteristiche da non testare

❖ Prenotazioni attive:

- Modifica delle prenotazioni attive

❖ Sede esistente:

- Modifica di una sede attiva

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 garantire che il sistema soddisfi pienamente i requisiti funzionali e non funzionali, verificando che ogni funzionalità implementata funzioni correttamente e produca risultati coerenti con gli input forniti. Il processo di testing si basa sul confronto tra il risultato atteso e quello effettivo ottenuto dall'operazione eseguita. Se il risultato effettivo corrisponde a quello previsto, il test sarà considerato superato. In caso contrario, se si riscontrano discrepanze, il test sarà classificato come fallito.

Gli errori vengono identificati durante i test confrontando i risultati con un oracolo, che rappresenta il comportamento atteso del sistema. Se il sistema implementato non soddisfa i criteri stabiliti, sarà necessario correggere il codice, ripetere i test e verificare che tutti i requisiti siano correttamente rispettati.

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 tempestivamente

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

eventuali anomalie nelle singole componenti, che vengono quindi corrette e ritestate 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 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 bloccanti:** 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 bloccanti 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.

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

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 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.

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**([aggiungere link](#)).

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

HairBeautyNow	Ingegneria del Software	Pagina 8 di 10
---------------	-------------------------	----------------

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

parallelo con lo sviluppo del codice per ottimizzare i tempi.

- **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.
- **Testing di Sistema (1 settimana):** 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 programmazioni.

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.

Progetto: HairBeautyNow	Versione: 1.0
Documento: Test Plan	Data: 16/12/2024

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.
------------------	---