1. **INTRODUZIONE**

Il Testing di integrazione è una delle fasi di testing più importanti. Essa consiste nel trovare fault integrando differenti componenti insieme, fault che non sono stati trovati nello “unit testing”. Questo documento ha il compito di identificare la strategia di testing di integrazione per il sistema ***Gestione Prenotazione.***

1. **RIFERIMENTI**

Per verificare la corretta integrazione dei sottosistemi del sistema sono stati predisposti dei test case basati sulla divisione in sottosistemi proposta in fase di System Design, nel documento *GESTIONEPRENOTAZIONE\_SDD*.

1. **TEST DI INTEGRAZIONE**

**3.1 Approccio di Integration Testing**

La strategia che verrà adottata per il testing di integrazione è quella “Bottom-up”. Tale strategia prevede che i sottosistemi nel layer più in basso della gerarchia vengano testati individualmente, successivamente sono testati i sottosistemi testati in precedenza congiuntamente a quelli che fanno uso tali sottosistemi. Si ripete questo passo finchè tutti i sottosistemi sono testati. Tale metodologia fa uso dei test driver. Un test driver è una implementazione parziale di una componente che dipende dalla componente.

**3.2 Componenti da testare**

I componenti da testare sono stati scelti seguendo la strategia di testing Bottom-up.

* Per quanto riguarda il data access layer verrà testato:

Database.

* Per quanto riguarda l’Application Layer verrà testato:

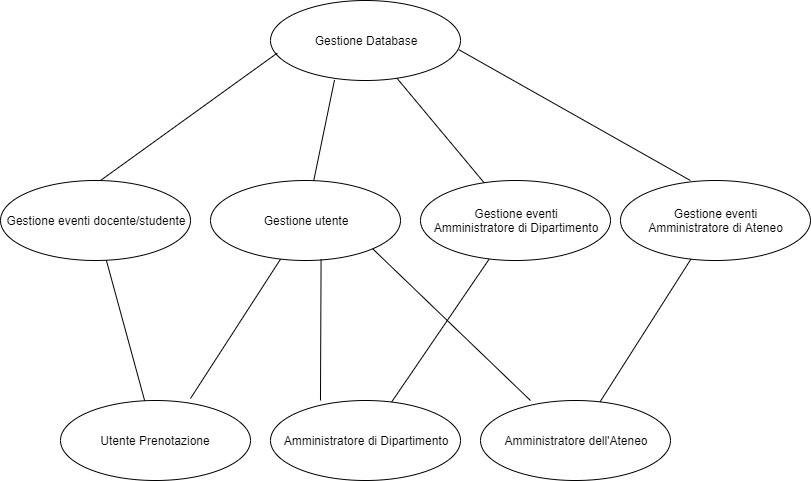
Gestione utente,

Gestione eventi amministratore di dipartimento,

Gestione eventi amministratore di ateneo,

Gestione eventi docente/studente

* E verrà testato il Presentation Layer per l’interazione con l’utente.



1. **CRITERI DI SUCCESSO**

Il testing ha successo quando il comportamento osservato dal sistema implementato è diverso comportamento atteso specificato attraverso il modello del sistema. Avremo, quindi, un successo se il test individuerà una failure. In tal caso, questa verrà analizzata e si procederà alla sua eventuale correzione, dopo una correzione la fase di testing verrà reiterata per verificare che tale modifica non abbia impattato su altre componenti del sistema. Se il test non riesce ad individuare un errore parliamo di fallimento.