



Test Case Integration Document

EV English Validation

Riferimento	
Versione	0.5
Data	14/12/2018
Destinatario	Top Management
Presentato da	Ivan Buccella, Alessandro Bacco, Giuseppe Cirino, Alfonso Ingenito, Angelomaria Macellaro, Luigi Melchionno, Vincenzo Passariello
Approvato da	Giammaria Giordano, Valeria Pontillo



Revision History

Data	Versione	Cambiamenti	Autori
11/12/2018	0.1	Inizio stesura	Alessandro Bacco, Angelomaria Macellaro
13/12/2018	0.2	Componenti da testare	Vincenzo Passariello
13/12/2018	0.3	Test integrazione	Alessandro Bacco
14/12/2018	0.4	Completamento	Angelomaria Macellaro
14/01/2019	0.5	Revisione e modifica	Alfonso Ingenito



Sommario

1.Introduzione	1
2. Riferimenti.....	1
3. Test di integrazione	1
3.1 Approccio di Integration Testing	1
3.2 Componenti da testare.....	1
4. Pass/fail criteri.....	3

1.Introduzione

Il testing serve a rilevare la presenza di errori e rappresenta una delle fasi più importanti. L'obiettivo del testing è quello di testare il maggior numero di funzioni, il più spesso possibile, in modo da rilevare le possibili anomalie.

2. Riferimenti

EV_SDD_v.0.8.

3. Test di integrazione

3.1 Approccio di Integration Testing

La strategia adottata per il testing di integrazione è quella di tipo “Bottom-up” la cui strategia prevede che i sottosistemi dei layer del livello più basso della gerarchia dopo essere state testate individualmente e vengono testati congiuntamente ai layer di livello superiore, cioè il layer della logica; al passo successivo vengono testati i due strati sottostanti con lo strato di presentazione. Per effettuare il testing di integrazione è stato adottato il modello “Bottom-up”, la sua strategia prevede che i sottosistemi dei layer del livello più basso vengono testati con i layer di livello superiore. Per testare i layer dei dati è sufficiente l'esecuzione delle query, mentre per testare i layer di logica è sufficiente l'esecuzione dei metodi implementati.

3.2 Componenti da testare

La scelta delle componenti da testare segue la decisione di eseguire la strategia di testing Bottom-up.

Per quanto riguarda il layer Model, le componenti da testare sono:

- Admin
- Attached
- Ente
- Request
- Secretary
- State
- Student
- SystemAttribute

Per quanto riguarda il layer Control, le componenti da testare sono:

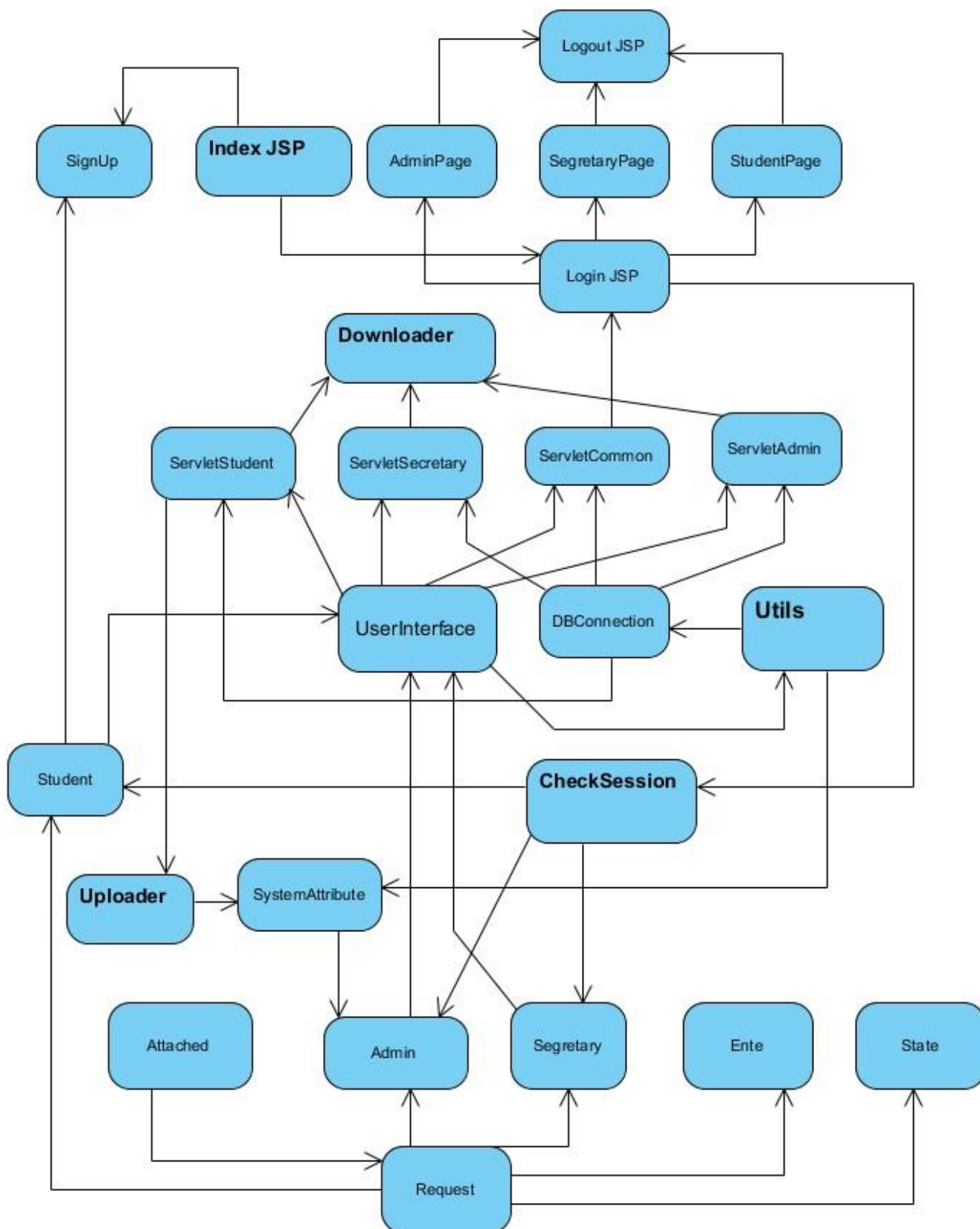
- CheckSession
- DBConnection
- Downloader
- ServletAdmin
- Servlet Common
- ServletSecretary
- ServletStudent
- Uploader
- Utils

Per quanto riguarda il layer View, le componenti da testare sono:

- UserInterface



- AdminPage
- SecretaryPage
- StudentPage
- Login JSP
- Logout JSP
- Index JSP



4. Pass/fail criteri

Se l'output osservato risulta essere diverso dall'output atteso, il testing ha successo.



Parleremo, quindi, di SUCCESSO se verranno individuate delle failure. In tal caso verranno analizzate e, se legate a dei fault, si procederà con le dovute correzioni. Infine, sarà iterata la fase di testing per verificare che le modifiche apportate non abbiano avuto impatto su altri componenti del sistema.

Si parlerà, invece, di FALLIMENTO se il test non riesce ad individuare un errore.