





TSR-TEST SUMMARY REPORT

Versione	1.0
Data	23/01/2022
Destinatario	Prof.ssa F. Ferrucci, Prof. Fabio Palomba
Presentato da	Tutti i team members
Approvato da	Salvatore Amideo
	Alice Vidoni





Revision History

Data	Versione	Descrizione	Autori
22/01/2022	0.1	Prima stesura	Maria Rosaria Giudice
23/01/2022	1.0	Completamento documento con aggiunta test di sistema	Maria Rosaria Giudice

Role	Name	Position	Contact
Top manager	Filomena Ferrucci	Cliente	fferrucci@unisa.it
Project Manager	Salvatore Amideo	Project Manager	s.amideo@studenti.unisa.it
Project Manager	Alice Vidoni	Project Manager	a.vidoni@studenti.unisa.it
Team Member	Claudio Buono	Team Member	c.buono20@studenti.unisa.it
Team Member	Nicola Cappell o	Team Member	n.cappello@studenti.unisa.it
Team Member	Alessandro Cavaliere	Team Member	a.cavaliere41@studenti.unisa.i t
Team Member	Carmine Citro	Team Member	c.citro23@studenti.unisa.it
Team Member	Maria Rosaria Giudice	Team Member	m.giudice12@studenti.unisa.it
Team Member	Alessio Salzano	Team Member	a.salzano32@studenti.unisa.it
Team Member	Gerardo Sessa	Team Member	g.sessa56@studenti.unisa.it





Team composition

Sommario

Revisio	on History	2
Team c	composition	3
1. Intro	oduzione	4
1.1	Identificativo del documento	4
1.2	Scopo	
	Riferimenti	
1.3		
2. Risul	ltati test di unità	4
2.1	Features testate	5
2.2	Panoramica dei risultati dei test	6
2.2	2.1 Gestione Tesserino	6
2.2	2.2 Gestione Personale ADISU	8
2.2	2.3 Gestione Operatore Mensa	8
2.2	2.4 Gestione Chat	9
2.2	2.5 Gestione FAQ	9
2.2	2.6 Gestione Ticket	10
2.2		
2.2	2.8 Gestione Autenticazione	11
3. Risul	ltati test di integrazione	12
3.1	Panoramica dei risultati dei test	12
4. Risul	ltati test di Sistema	12
4.1	Panoramica dei risultati dei test	13
4.1.	.1 Lato Cliente	
4.1.	Lato Operatore Mensa	14
4.1.	Lato Personale Adisu	
4.1.	.4 Lato Admin	
5. Riep	oilogo del testing	15
6 Glos	esario	15





1. Introduzione

Questo documento mostra i risultati dell'esecuzione dei test case di unità delle varie componenti del software web Unisa Eat. In particolare, sono stati testati i controller e i model del software con l'utilizzo di Mocha&Chai, e successivamente è stato effettuato il testing di sistema con Selenium.

1.1 Identificativo del documento

Il documento in questione è caratterizzato da alcune proprietà che lo definiscono univocamente; tali proprietà riguardano la data di stesura, i nomi degli autori del documento e l'approvazione da parte dei project manager.

1.2 Scopo

Il Test Summary Report fornisce una presentazione dei casi di test di unità, di integrazione e di sistema per il software Unisa Eat. I membri del team si sono impegnati nel verificare che le singole unità del sistema (classi e metodi) abbiano avuto il comportamento atteso.

1.3 Riferimenti

- UE_RAD_V_2.0.docx
- UE_SDD_V_2.0.docx
- UE_ODD_V_1.1.docx
- UE_TCS_V_1.0.docx
- B. Bruegge, A.H.Dutoit, ObjectOrientedSoftwareEngineering—UsingUML, Patterns and Java, Prentice Hall.
- Slides del corso presenti sulla piattaforma.

2. Risultati test di unità

Di seguito vengono riportati i risultati dei test di unità in base alle varie gestioni del sistema.





2.1 Features testate

Il testing d'unità si propone di effettuare il test dei control presenti all'interno della propria gestione. In particolare:

- 1. GT GESTIONE TESSERINO:
 - richiestaTesserino
 - ricaricaTesserino
 - rinnovoTesserino
- 2. GLA GESTIONE PERSONALE ADISU
 - inserimentoPersonaleAdisu
- 3. GAR GESTIONE OPERATORE MENSA
 - inserimentoOperatoreMensa
- 4. GN GESTIONE CHAT
 - invioMessaggio
 - modificaMessaggio
- 5. GP GESTIONE FAQ
 - inserimentoFaq
 - modificaFaq
- 6. GR GESTIONE TICKET
 - compilazioneTicket
 - risoluzioneTicket
- 7. GD GESTIONE PROFILO
 - modificaPassword
- 8. GC GESTIONE AUTENTICAZIONE
 - login





2.2 Panoramica dei risultati dei test

2.2.1 Gestione Tesserino

Classe richiesta tesserino





Classe ricarica Tesserino

Classe rinnovo Tesserino

UE_MU_V_1.0 Pag. 7 | 16





2.2.2 Gestione Personale ADISU

Classe inserimentoPersonaleAdisu

2.2.3 Gestione Operatore Mensa





Classe inserimentoPersonaleAdisu

2.2.4 Gestione Chat

Classe invioMessaggio

```
Field test for invio messaggio

√ TC_CM_1.1 (38ms)

√ TC_CM_1.2

2 passing (69ms)
```

Classe modificaMessaggio

```
Field test for modifica messaggio

VTC_CM_2.1
VTC_CM_2.2

2 passing (58ms)
```

2.2.5 Gestione FAQ





Classe inserimentoFaq

Classe modificaFaq

```
Field test for faq

√ TC_FM_2.1 (132ms)

√ TC_FM_2.2

√ TC_FM_2.3

√ TC_FM_2.4

√ TC_FM_2.5 (41ms)

5 passing (301ms)
```

2.2.6 Gestione Ticket

Classe compilazioneTicket

Classe risoluzioneTicket

 $UE_MU_V_1.0$ Pag. 10 | 16





2.2.7 Gestione Profilo

Classe modificaPassword

```
Field test for profilo

TC_PM_1.1 (145ms)

TC_PM_1.2

TC_PM_1.3

TC_PM_1.4

TC_PM_1.5

TC_PM_1.6

TC_PM_1.6

TC_PM_1.7

TC_PM_1.8

TC_PM_1.9 (636ms)
```

2.2.8 Gestione Autenticazione

Classe login

 $UE_MU_V_1.0$ Pag. 11 | 16





3. Risultati test di integrazione

3.1 Panoramica dei risultati dei test

Lo scopo del test di integrazione, data l'architettura software impiegata, è di verificare il corretto funzionamento degli endpoint dell'applicazione, controllando il risultato atteso dalla response. Esso rappresenta una delle fasi di testing più importanti, in quanto consiste nella verifica delle interazioni tra 2 o più componenti. Per eseguirlo è stato utilizzato Mocha. Elenchiamo qui i risultati:

Coverage:



File	% Stmts	% Branch	% Funcs	% Lines
All files	77.67	75.04	64.22	78.75
UnisaEAT_ProjectBE	93.93	100	66.66	93.93
server.js	93.93	100	66.66	93.93
UnisaEAT_ProjectBE/app/config	100	100	100	100
db.config.js	100	100	100	100
UnisaEAT_ProjectBE/app/controller	75.91	74.95	61.76	77.26
controller_faq.js	90.62 79.59	78.57 75	71.42 100	90 84.78
controller_login.js controller_messaggio.js	66.66	76.66	44.44	71.79
controller_personale.js	93.47	80.72	75	93.47
controller_profile.js	75.86	75	60	78.57
controller_tesserino.js	69.63	72.95	53.57	70.84
controller_ticket.js	82.25	72.72	66.66	82.25
hash.js	100	100	100	100
UnisaEAT_ProjectBE/app/models	72.09	100	53.84	72.09
admin.js	62.5	100	50	62.5
cliente.js	66.66	100	50	66.66
conversazione.js	66.66	100	50	66.66
faq.js	62.5	100	50	62.5
index.js	100	100	100	100
menu.js	62.5	100	50	62.5
messaggio.js	100	100	100	100
notifica.js	66.66	100	50	66.66
ordine.js pasto.js	66.66 62.5	100 100	50 50	66.66 62.5
personale.js	62.5	100	50	62.5
statistiche.js	62.5	100	50	62.5
tesserino.js	62.5	100	50	62.5
ticket.js	62.5	100	50	62.5
UnisaEAT_ProjectBE/app/routes	100	100	100	100
routes conversazione.is	100	100	100	100
routes_faq.js routes_login.js	100	100	100	100
routes_login.js	100	100	100	100
routes_menu.js	100	100	100	100
routes_messaggio.js	100	100	100	100
routes_notifica.js	100	100	100	100
routes_ordine.js	100	100	100	100
routes_personale.js	100	100	100	100
routes_profilo.js	100	100	100	100
routes_statistiche.js	100	100	100 100	100 100
routes_tesserino.js routes_ticket.js	100 100	100 100	100	100
routes_treket.js	100	100	100	100

4. Risultati test di Sistema

 $U E_M U_V_1.0$ Pag. 12 | 16

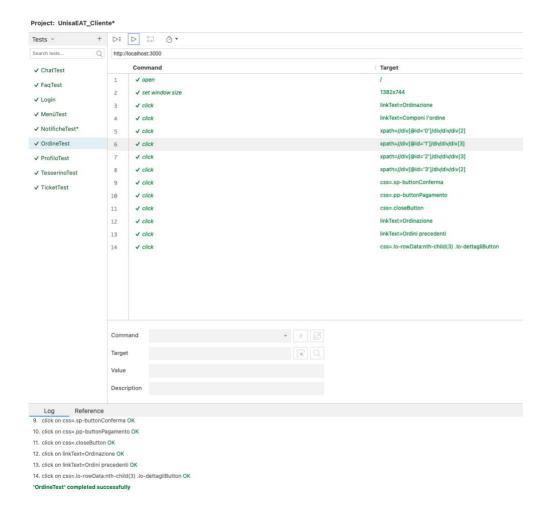




Di seguito vengono riportati i risultati dei test di sistema in base agli attori del sistema:

4.1 Panoramica dei risultati dei test

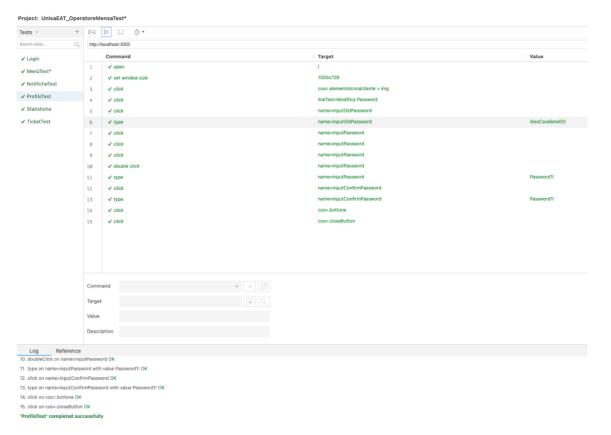
4.1.1 Lato Cliente



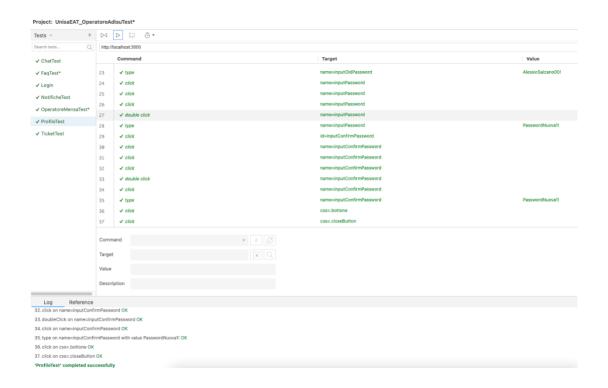




4.1.2 Lato Operatore Mensa



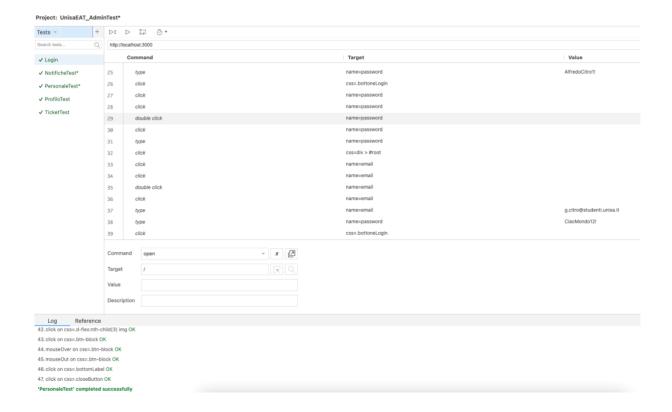
4.1.3 Lato Personale Adisu







4.1.4 Lato Admin



5. Riepilogo del testing

	Numero di componenti testate	Numeri di errori trovati	Numero di errori corretti	Numero di componenti non testate
Control	7	0	0	0
Model	14	0	0	0
Integration Testing	1	0	0	0

6. Glossario

 RAD (Requirement Analysis Document): Documento di Raccolta e analisi dei Requisiti; contienel'elenco dei requisiti funzionali enon funzionali individuati in fase di individuazione degli stessi e la loro analisi sotto forma di scenari e casi d'uso. I mock-up mostrano una possibile implementazione dell'interfaccia del sistema.





- SDD (System Design Document): Documento all'interno del quale viene riportata la progettazione del sistema come risultato di una prima fase di modellazione: contiene una suddivisione ad alto livello del sistema nei sottosistemi che lo comporranno.
- ODD (Object Design Document): Documento che riporta e analizza gli oggetti che compongono il sistema analizzando le componenti a più basso livello, riportandole così come saranno implementate.
- TCS (Test Case Specification): Documento che riporta i casi di test pianificati.