



Test Summary Report

C01 - GameHub

Riferimento	C01_TSР_V1.0
Versione	1.0
Data	20/01/2021
Destinatario	Top Management
Presentato da	Gerardo Brescia, Domenico D'Alessandro, Roberto Esposito, Francesco Mattina, Francesco Maria Rastelli, Andrea Terlizzi
Approvato da	



Revision History

Data	Versione	Descrizione	Autori
20/01/2021	1.0	Prima Stesura e Revisione	TUTTI



Sommario

1 Introduzione	1
2 Testing di Unità	1
2.1 Classi Testate	1
2.2 Package Model.bean	3
2.3 Package Model.dao	4
2.4 Package Controller.UserManagement.AdminManagement	4
2.5 Package Controller.UserManagement.UserProfileManagement	5
2.6 Package Controller.ShopManagement.....	5
2.7 Package Controller.UserManagement.OperatorManagement.....	6
2.8 Riassunto dei risultati del testing di unità	6
3 Testing di Sistema	7
3.1 Test Case eseguiti	7
3.2 Riassunto dei risultati del testing di sistema	10
5. Glossario	10



1 Introduzione

Nel seguente documento vengono riportati i risultati relativi all'esecuzione dei test case di unità delle componenti della piattaforma GameHub. Per quanto riguarda i package model e controller è stato utilizzato JUnit, framework di unit testing per il linguaggio di programmazione Java. Invece, per quanto riguarda il package view è stato utilizzato il framework Katalon il quale fa riferimento alla libreria Selenium.

Il testing delle servlet, attraverso le richieste e risposte http, è effettuato mediante l'utilizzo del framework Mockito.

2 Testing di Unità

Di seguito sono riportati i risultati del testing di Unità, effettuato con JUnit sulle classi Java appartenenti ai packages Controller e Model, quest'ultimo è costituito dai packages Bean e Dao.

2.1 Classi Testate

Il testing di unità si è concentrato sulle seguenti classi:

- **Package Bean:**
 1. Admin
 2. Cart
 3. Category
 4. Order
 5. Product
 6. Tag
 7. User
- **Package Dao:**
 1. AdminDAO
 2. CartDAO
 3. CategoryDAO
 4. DigitalProductDAO
 5. ModeratorDAO
 6. OperatorDAO
 7. OrderDAO
 8. PhysicalProductDAO
 9. TagDAO
 10. UserDAO



- **Package Controller:**
 - **Package CommunityManagement:**
 1. ShowCommunityPage;
 - **Package ShopManagement:**
 1. CartServlet;
 2. GetMoreCategoriesServlet;
 3. GetMoreProductsServlet;
 4. GetMoreTagsServlet;
 5. ProceedCheckoutServlet;
 6. ShowCategoryPageServlet;
 7. ShowCheckoutPageServlet;
 8. ShowProductsServlet;
 9. ShowShopPageServlet;
 - **Package UserManagement:**
 - ◆ **Package AdminManagement:**
 1. GetMoreAdminsServlet
 2. GetMoreModeratorsServlet
 3. GetMoreOperatorsServlet
 4. GetMoreUsersServlet
 5. ManageAdminsServlet
 6. ManageCategoriesServlet
 7. ManageModeratorsServlet
 8. ManageOperatorsServlet
 9. ManageProductsServlet
 10. ManageTagsServlet
 11. ManageUsersServlet
 12. ShowAdminAreaServlet
 - ◆ **Package OperatorManagement:**
 1. ApproveOrderServlet
 2. DeclineOrderServlet
 3. ShowOperatorAreaServlet
 - ◆ **Package UserProfileManagement:**
 1. CheckMailServlet
 2. CheckUsernameServlet
 3. LoginServlet
 4. LogoutServlet
 5. ShowLoginPageServlet
 6. ShowReservedAreaPageServlet
 7. ShowSignUpPageServlet
 8. SignUpServlet
 9. UpdateUserServlet
 - ◆ HomeServlet



Viene riportata di seguito la branch coverage ottenuta testando l'intero progetto:

gamehub

gamehub

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
controller.userManagement.adminManagement		56%		51%	135	283	532	1.221	0	36	0	12
model.dao		94%		92%	20	213	130	1.614	1	88	0	11
model.bean		86%		91%	30	209	46	500	22	163	0	13
controller.userManagement.userProfileManagement		93%		88%	19	111	30	359	0	28	0	9
controller.shopManagement		94%		93%	10	125	25	422	0	30	0	9
controller.userManagement.operatorManagement		85%		78%	4	18	7	84	1	11	0	3
controller		78%		50%	6	9	12	39	5	8	0	2
controller.communityManagement		100%		n/a	0	3	0	6	0	3	0	1
Total	3.322 of 18.829	82%	302 of 1.208	75%	224	971	782	4.245	29	367	0	60

2.2 Package Model.bean

Viene riportata di seguito la branch coverage ottenuta testando il package **model.bean**:

gamehub > [model.bean](#)

model.bean

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
Admin		50%		87%	5	14	9	29	4	10	0	1
Category		51%		100%	3	13	5	25	3	11	0	1
Moderator		66%		66%	5	11	9	29	3	8	0	1
Order		92%		80%	7	36	4	75	2	23	0	1
Operator		66%		n/a	2	9	2	23	2	9	0	1
PhysicalProduct		64%		n/a	1	8	4	19	1	8	0	1
Tag		61%		100%	2	9	2	17	2	7	0	1
Cart		94%		100%	3	25	5	58	3	14	0	1
User		96%		100%	0	30	4	93	0	28	0	1
Product		97%		100%	1	36	1	78	1	27	0	1
EmailUtility		96%		n/a	1	2	1	17	1	2	0	1
DigitalProduct		100%		n/a	0	14	0	36	0	14	0	1
EmailUtility.new Authenticator().(...)		100%		n/a	0	2	0	2	0	2	0	1
Total	328 of 2.424	86%	8 of 92	91%	30	209	46	500	22	163	0	13



2.3 Package Model.dao

Viene riportata di seguito la branch coverage ottenuta testando il package **model.dao**:

gamehub > model.dao

model.dao

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
ModeratorDAO		84%		82%	5	23	20	122	0	9	0	1
OrderDAO		95%		94%	3	34	16	237	0	8	0	1
OperatorDAO		92%		92%	1	16	15	139	0	9	0	1
PhysicalProductDAO		96%		100%	0	26	12	252	0	12	0	1
AdminDAO		87%		88%	2	16	12	92	0	7	0	1
CategoryDAO		87%		91%	1	13	11	84	0	7	0	1
CartDAO		94%		84%	7	29	11	138	0	6	0	1
UserDAO		95%		100%	0	17	14	190	0	10	0	1
DigitalProductDAO		97%		100%	0	26	10	278	0	12	0	1
TagDAO		90%		100%	0	10	8	66	0	6	0	1
ConPool		94%		100%	1	3	1	16	1	2	0	1
Total	368 of 6.274	94%	19 of 250	92%	20	213	130	1.614	1	88	0	11

2.4 Package Controller.UserManagement.AdminManagement

Viene riportata di seguito la branch coverage ottenuta testando il package **controller.userManagement.adminManagement**:

gamehub > controller.userManagement.adminManagement

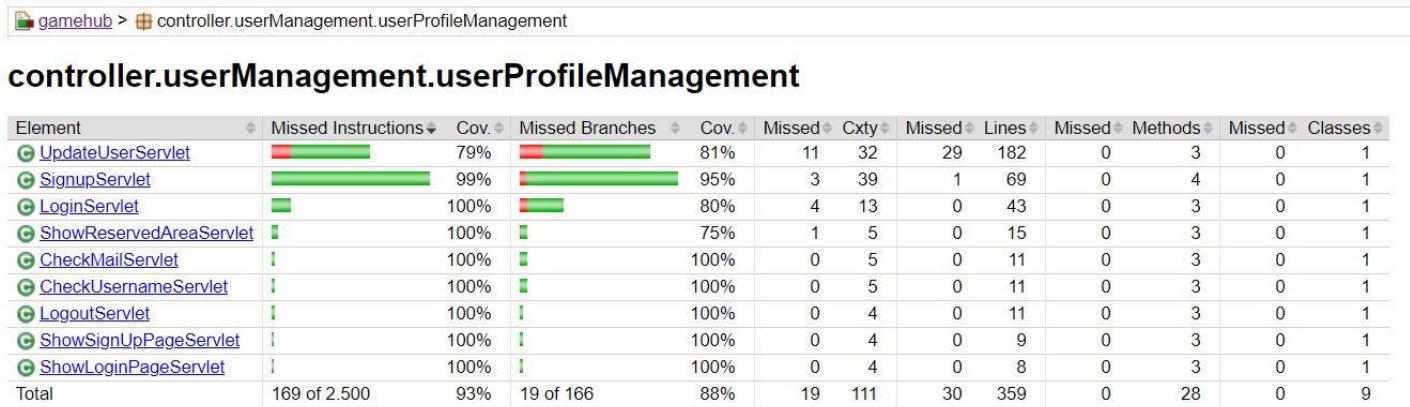
controller.userManagement.adminManagement

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
ManageProductServlet		23%		26%	97	126	428	571	0	3	0	1
ManageCategoryServlet		25%		22%	24	32	84	118	0	3	0	1
GetMoreAdminsServlet		72%		50%	3	6	5	26	0	3	0	1
ManageTagServlet		91%		97%	1	20	5	76	0	3	0	1
ManageUserServlet		76%		87%	1	7	4	24	0	3	0	1
ShowAdminAreaServlet		93%		55%	9	13	6	65	0	3	0	1
ManageAdminsServlet		100%		100%	0	20	0	92	0	3	0	1
ManageOperatorServlet		100%		100%	0	22	0	87	0	3	0	1
ManageModeratorServlet		100%		100%	0	19	0	81	0	3	0	1
GetMoreUsersServlet		100%		100%	0	6	0	28	0	3	0	1
GetMoreOperatorsServlet		100%		100%	0	6	0	27	0	3	0	1
GetMoreModeratorsServlet		100%		100%	0	6	0	26	0	3	0	1
Total	2.272 of 5.176	56%	239 of 494	51%	135	283	532	1.221	0	36	0	12



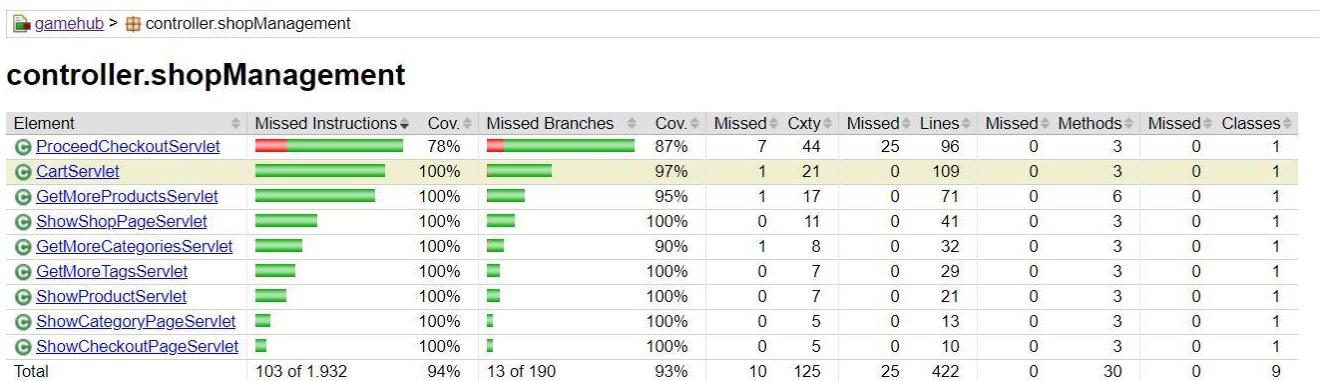
2.5 Package Controller.UserManagement.UserProfileManagement

Viene riportata di seguito la branch coverage ottenuta testando il package **controller.userManagement.userProfileManagement**:



2.6 Package Controller.ShopManagement

Viene riportata di seguito la branch coverage ottenuta testando il package **controller.shopManagement**:





2.7 Package Controller.UserManagement.OperatorManagement

Viene riportata di seguito la branch coverage ottenuta testando il package **controller.userManagement.operatorManagement**:

gamehub > controller.userManagement.operatorManagement

controller.userManagement.operatorManagement

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cqty	Missed	Lines	Missed	Methods	Missed	Classes
C ApproveOrderServlet		78%		75%	3	9	5	44	1	5	0	1
C DeclineOrderServlet		93%		100%	0	4	2	26	0	3	0	1
C ShowOperatorAreaServlet		100%		75%	1	5	0	14	0	3	0	1
Total	51 of 361	85%	3 of 14	78%	4	18	7	84	1	11	0	3

2.8 Riassunto dei risultati del testing di unità

Componenti Testate		Errori Trovati		Errori Corretti		Componenti non Testate	
Model	Controller	17	0	0	3	5	2
Model	Personalization	0	0	0	0	4	

- All'interno del package controller non sono stati effettuati i test delle classi ShowReccomendedProductServlet e SetPersonalizationVote e del package personalization in quanto appartenenti al modulo di Intelligenza Artificiale.
- Non è stato possibile testare le classi ManageCategory e ManageProduct nella loro interezza poiché contengono istanze della classe "Part" non testabili con gli strumenti a nostra disposizione. Per la stessa motivazione appena illustrata non è stato possibile ottenere un valore molto al di sopra del 75% di branch coverage.



3 Testing di Sistema

Di seguito vengono riportati i risultati ottenuti dal testing di sistema, effettuato con il plugin Katalon Recorder.

Il testing di sistema viene effettuato seguendo i test case individuati all'interno del Test Case Specification Document.

3.1 Test Case eseguiti

Sono stati eseguiti i seguenti Test Case:

♦ **Registrazione Utente TC_1**

- TC_1.1_1
- TC_1.1_2
- TC_1.1_3
- TC_1.1_4
- TC_1.1_5
- TC_1.1_6
- TC_1.1_7
- TC_1.1_8
- TC_1.1_9
- TC_1.1_10
- TC_1.1_11
- TC_1.1_12
- TC_1.1_13
- TC_1.1_14
- TC_1.1_15
- TC_1.1_16
- TC_1.1_17
- TC_1.1_18
- TC_1.1_19
- TC_1.1_20
- TC_1.1_21
- TC_1.1_22
- TC_1.1_23
- TC_1.1_24
- TC_1.1_25
- TC_1.1_26
- TC_1.1_27

♦ **Login TC_2**

- TC_2.1_1
- TC_2.1_2



- TC_2.1_3

- TC_2.1_4

◆ **Modifica Dati Utente - Info - TC_3a**

- TC_3a.1_1

- TC_3a.1_2

- TC_3a.1_3

- TC_3a.1_4

- TC_3a.1_5

- TC_3a.1_6

- TC_3a.1_7

- TC_3a.1_8

- TC_3a.1_9

- TC_3a.1_10

- TC_3a.1_11

- TC_3a.1_12

◆ **Modifica Dati Utente - Password - TC_3b**

- TC_3b.1_1

- TC_3b.1_2

- TC_3b.1_3

◆ **Modifica Dati Utente – Other Info - TC_3c**

- TC_3c.1_1

- TC_3c.1_2

- TC_3c.1_3

- TC_3c.1_4

- TC_3c.1_5

- TC_3c.1_6

- TC_3c.1_7

- TC_3c.1_8

- TC_3c.1_9

- TC_3c.1_10

- TC_3c.1_11

- TC_3c.1_12

- TC_3c.1_13

◆ **Checkout Whitout Login – TC_4**

- TC_4.1_1

- TC_4.1_2

- TC_4.1_3

- TC_4.1_4

- TC_4.1_5

- TC_4.1_6

- TC_4.1_7

- TC_4.1_8

- TC_4.1_9

- TC_4.1_10



- TC_4.1_11
- TC_4.1_12
- TC_4.1_13
- TC_4.1_14
- TC_4.1_15
- TC_4.1_16
- TC_4.1_17
- TC_4.1_18
- TC_4.1_19
- TC_4.1_20
- TC_4.1_21
- TC_4.1_22
- TC_4.1_23
- TC_4.1_24
- TC_4.1_25

◆ **Checkout Whit Login**

- TC_5.1_1
- TC_5.1_2
- TC_5.1_3
- TC_5.1_4
- TC_5.1_5
- TC_5.1_6
- TC_5.1_7
- TC_5.1_8
- TC_5.1_9
- TC_5.1_10

◆ **Aggiungi Tag TC_8**

- TC_8.1_1
- TC_8.1_2
- TC_8.1_3

◆ **Modifica Tag TC_9**

- TC_9.1_1
- TC_9.1_2
- TC_9.1_3

◆ **Aggiungi Operatore TC_10**

- TC_10.1_1
- TC_10.1_2
- TC_10.1_3
- TC_10.1_4
- TC_10.1_5
- TC_10.1_6

◆ **Modifica Operatore TC_11**

- TC_11.1_1
- TC_11.1_2



- TC_11.1_3
- TC_11.1_4

3.2 Riassunto dei risultati del testing di sistema

Grazie al testing di sistema è stato possibile individuare 3 failure, e di conseguenza sono stati risolti i relativi faults. Tutte le failure individuate sono state riportate nel documento "Incident Report".

5. Glossario

- RAD (Requirement Analysis Document): Documento ottenuto dalla fase di raccolta ed analisi dei requisiti;
- SDD (System Design Document): Documento ottenuto dalla fase di modellazione del sistema proposto;
- ODD (Object Design Document): Documento all'interno del quale si definiscono le scelte relative all'implementazione del sistema, scegliendo le possibili soluzioni ed analizzandone eventuali costi/benefici;
- TCD (Test Case Document): documento all'interno del quale vengono testate le singole funzionalità, osservando l'input fornito e l'output atteso, specificati prima dell'esecuzione del test;