

**Università degli Studi di Salerno
Corso di Ingegneria del Software**

**BirdComics
Test Plan
Versione 2.0**

BirdComics

Data: 23/03/2025

Coordinatore del progetto:

Nome	Matricola

Partecipanti:

Nome	Matricola
De Rosa Giuseppe	0512110683
Buccino Adriano	0512120484

Scritto da:	
--------------------	--

Revision History

Data	Versione	Descrizione	Autore
23/03/2025	2.0	Revisione finale per la consegna	De Rosa Giuseppe Buccino Adriano
16/12/2024	1.0	Prima Versione del Test Plan	De Rosa Giuseppe Buccino Adriano

1. Introduzione	4
2. Relazione con gli altri documenti	4
3. Panoramica del sistema	4
4. Funzionalità da testare	4
• Gestione Profilo	4
5. Criteri di successo/fallimento	5
6. Approccio	5
Unit Testing	5
Integration Testing	5
System Testing	5
7. Sospensione e ripresa	6
8. Casi di test	6
9. Test Schedule	16

1. Introduzione

Nel documento viene definito il piano di testing, con l'obiettivo di verificare il corretto funzionamento attraverso l'analisi delle funzionalità da testare e i relativi test cases. Descriviamo inoltre come viene eseguito il testing del sistema, per identificare eventuali bug, errori o comportamenti indesiderati e garantire un'esperienza utente ottimale e un sistema affidabile.

2. Relazione con gli altri documenti

La relazione tra il Test Plan e il RAD riguarda i requisiti funzionali e verranno svolti test su queste funzionalità, tenendo conto delle specifiche espresse. Nella fase di testing si deve tener conto della suddivisione in sottosistemi presentati nell'SSD.

3. Panoramica del sistema

Il sistema che svilupperemo si basa sull'architettura **MVC (Model-View-Controller)** e utilizza la **decomposizione del sottoproblema** come metodologia progettuale. Per garantire modularità e semplicità di gestione, il sistema è stato suddiviso in sottosistemi, organizzati per gestioni specifiche. I sottosistemi identificati sono:

- Gestione Assistenza (Non implementato)
- Gestione Profilo
- Gestione Catalogo
- Gestione Spedizione
- Gestione Magazzino
- Gestione Finanza (Non implementato)
- Gestione Carrello

Questa struttura modulare consente una più facile implementazione, manutenzione e testing di ciascun sottosistema, migliorando l'efficienza complessiva del sistema.

4. Funzionalità da testare

Funzionalità introdotte nel sistema che saranno sottoporre a test:

- Gestione Profilo
 - Autenticazione
 - Registrazione
- Gestione Carrello
 - Svuota Carrello
 - Aggiunta Fumetto al carrello
 - Rimozione Fumetto dal carrello
- Gestione Catalogo
 - Aggiunta Fumetto al catalogo
 - Rimozione Fumetto dal catalogo
- Gestione Ordine
 - Effettua acquisto

5. Criteri di successo/fallimento

I criteri di successo e fallimento si basano sull'identificazione di faults nel sistema, per consentire la loro correzione prima del rilascio del software. L'**oracolo** del test è il risultato atteso dell'esecuzione.

Un test case sarà considerato superato (**Pass**) se il risultato ottenuto corrisponde esattamente a quello atteso, come definito dall'oracolo del test. Se, invece, il risultato ottenuto differisce da quello atteso, il test avrà esito negativo (**Fail**).

L'obiettivo principale del testing è l'identificazione e la correzione delle faults nel sistema. Una volta che una fault è risolta, si procederà con il test di regressione per garantire che la modifica non abbia introdotto nuovi problemi nel sistema.

6. Approccio

Unit Testing

Nella fase di Unit Testing, ci concentreremo sul test delle singole componenti del sistema in modo isolato, per garantire che ogni unità funzioni correttamente prima di essere integrata. In questa fase, applicheremo la tecnica white-box. I test saranno implementati utilizzando il framework JUnit e Mockito che consentiranno di automatizzare e validare i test delle singole unità, assicurandoci che le componenti eseguano correttamente le operazioni senza errori.

Integration Testing

Nell'Integration Testing, testeremo le componenti del sistema/sottosistema aggregate in gruppi, per verificare che interagiscano correttamente tra di loro. Adotteremo un approccio Bottom-Up, iniziando dal test delle componenti di accesso ai dati (DAO) già verificate durante il Unit Testing, per poi procedere con l'integrazione delle Servlet e infine verificare che il Layer di Presentazione visualizzi correttamente i dati dal Layer di Logica di Business. L'obiettivo è garantire che i vari moduli, dopo essere stati testati singolarmente, funzionino correttamente insieme. Useremo JUnit per i test e Mockito per simulare le interazioni tra i componenti.

System Testing

Nella fase di System Testing, testeremo l'intero sistema per assicurarci che soddisfi i requisiti funzionali, non funzionali e gli obiettivi di design definiti nel RAD e nel SDD. In questa fase, testeremo il comportamento complessivo del sistema in scenari d'uso realistici, per garantire che tutte le funzionalità siano implementate correttamente e che l'esperienza dell'utente finale sia ottimale. I test saranno progettati utilizzando la tecnica del Category-Partitioning, che ci permetterà di coprire un ampio spettro di funzionalità con un numero ridotto di test. Per eseguire i test, utilizzeremo Selenium IDE, uno strumento che consente di automatizzare i test delle interazioni dell'utente con l'interfaccia del sistema.

7. Sospensione e ripresa

La sospensione dell'esecuzione dei casi di test avverrà nei seguenti casi:

- Difetti critici che impediscono il proseguimento dei test, come ad esempio crash del sistema o altri problemi gravi.
- Cambiamenti significativi nei requisiti del sistema
- Limitazioni delle risorse

La ripresa del testing sarà possibile solo quando:

- I difetti critici sono stati risolti e non impediscono più l'esecuzione dei test.
- I cambiamenti nei requisiti sono stati adeguatamente compresi e integrati.
- Le Risorse necessarie sono nuovamente disponibili

Inoltre, al momento della ripresa, sarà necessario eseguire un test di regressione per assicurarsi che le modifiche non abbiano introdotto nuovi malfunzionamenti nel sistema.

8. Casi di test

8.1. Registrazione

PARAMETRI	Email, password, Nome, Cognome, Data di nascita, Indirizzo, Numero di telefono, e Codice postale
OGGETTI DELL'AMBIENTE	Database
CATEGORIE	C1: email valida (rispetta il formato della specifica) C2: email presente nel database C3: password valida (rispetta il formato della specifica) C4: Nome valido(rispetta il formato della specifica) C5: Cognome valido(rispetta il formato della specifica) C6: Data di nascita valida(rispetta il formato della specifica) C7: Indirizzo valido(rispetta il formato della specifica) C8: Numero di telefono valido(rispetta il formato della specifica) C9: Codice postale valido(rispetta il formato della specifica)
SCELTE E VINCOLI RELATIVI	<ul style="list-style-type: none">- SCELTE PER C1:<ul style="list-style-type: none">- EV1: email valida [<i>property email_valida</i>]- EV2: email NON valida- SCELTE PER C2:<ul style="list-style-type: none">- EP1: email presente nel database [<i>if email_valida</i>]- EP2: email NON presente nel database [<i>property email_non_presente</i>]

	<ul style="list-style-type: none"> - SCELTE PER C3: <ul style="list-style-type: none"> - PV1: password valida [<i>property password_valida</i>] - PV2: password NON valida [<i>if email_valida and email_non_presente</i>] - SCELTE PER C4: <ul style="list-style-type: none"> - NV1: Nome valido [<i>property nome_valido</i>] - NV2: Nome NON valido [<i>if email_valida and email_non_presente and password_valida</i>] - SCELTE PER C5: <ul style="list-style-type: none"> - CV1: Cognome valido [<i>property cognome_valido</i>] - CV2: Cognome NON valido [<i>if email_valida and email_non_presente and password_valida</i>] - SCELTE PER C6: <ul style="list-style-type: none"> - DV1: Data di nascita valida [<i>property dataNascita_valida</i>] - DV2: Data di nascita NON valida [<i>if email_valida and email_non_presente and password_valida</i>] - SCELTE PER C7: <ul style="list-style-type: none"> - IV1: Indirizzo valido [<i>property indirizzo_valido</i>] - IV2: indirizzo NON valido [<i>if email_valida and email_non_presente and password_valida</i>] - SCELTE PER C8: <ul style="list-style-type: none"> - TV1: Numero di telefono valido [<i>property telefono_valido</i>] - TV2: Telefono NON valido [<i>if email_valida and email_non_presente and password_valida</i>] - SCELTE PER C9: <ul style="list-style-type: none"> - AV1: Codice postale valido [<i>property codPostale_valido</i>] - AV2: codice postale NON valido [<i>if email_valida and email_non_presente and password_valida</i>]
TEST FRAME	<ul style="list-style-type: none"> - TF1: EV2 ORACOLO: Messaggio di errore accanto al campo email: "email errata, non rispetta la sintassi" - TF2: EV1, EP2 ORACOLO: Messaggio di errore accanto al campo email: "email già utilizzata" - TF3: EV1, EP1, PV2 ORACOLO: Messaggio di errore accanto al campo password: "password non rispetta la sintassi" - TF4: EV1, EP2, PV1, NV2 ORACOLO: Messaggio di errore accanto al campo nome: "nome non può contenere caratteri speciali" - TF5: EV1, EP2, PV1, NV1, CV2 ORACOLO: Messaggio di errore accanto al campo cognome: "cognome non può contenere caratteri speciali" - TF6: EV1, EP2, PV1, NV1, CV1, DV2 ORACOLO: Messaggio di errore accanto al campo data di nascita: "la data di nascita non rispetta il formato standardizzato gg/mm/aaaa" - TF7: EV1, EP2, PV1, NV1, CV1, DV1, IV2 ORACOLO: Messaggio di errore accanto al campo indirizzo: "indirizzo non può contenere caratteri speciali" - TF8: EV1, EP2, PV1, NV1, CV1, DV1, IV1, TV2 ORACOLO: Messaggio di errore accanto al campo numero di telefono: "il numero di telefono deve

	<p>essere composto da 10 caratteri solo numerici”</p> <ul style="list-style-type: none">- TF9: EV1, EP2, PV1, NV1, CV1, DV1, IV1, TV2 , AV2 ORACOLO: Messaggio di errore accanto al campo codice postale: “il codice postale deve essere composto da 5 caratteri solo numerici”- TF10: EV1, EP2, PV1, NV1, CV1, DV1, IV1, TV1, AV1 ORACOLO: Registrazione effettuata con successo
TEST CASE	<ul style="list-style-type: none">- TC1: email = “ap%\$uapu@gmail.com” ORACOLO: La pagina di registrazione mostra il messaggio: “email sintatticamente non corretta contiene caratteri speciali oppure è un’email aziendale”- TC2: email = “apuapu@gmail.com”- in DB già presente email = “apuapu@gmail.com” ORACOLO: La pagina di registrazione mostra il messaggio: “email già utilizzata”- TC3: email = “apuapu@gmail.com”, password = “apu” ORACOLO: Messaggio di errore accanto al campo password: “password non rispetta la sintassi”- TC4: email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco@” ORACOLO: Messaggio di errore accanto al campo nome: “nome non può contenere caratteri speciali”.- TC5: Email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco”, Cognome: “Rossi!” ORACOLO: Messaggio di errore accanto al campo cognome: “cognome non può contenere caratteri speciali”.- TC6: Email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco”, Cognome: “Rossi”, Data di nascita: “15-03-1990” ORACOLO: Messaggio di errore accanto al campo data di nascita: “la data di nascita non rispetta il formato standardizzato gg/mm/aaaa”.- TC7: Email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco”, Cognome: “Rossi”, Data di nascita: “15/03/1990”, Indirizzo: “Via Milano#15” ORACOLO: Messaggio di errore accanto al campo indirizzo: “indirizzo non può contenere caratteri speciali”.- TC8: Email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco”, Cognome: “Rossi”, Data di nascita: “15/03/1990”, Indirizzo: “Via Milano 15”, Numero di telefono: “12345abcde” ORACOLO: Messaggio di errore accanto al campo numero di telefono: “il numero di telefono deve essere composto da 10 caratteri solo numerici”.- TC9: Email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco”, Cognome: “Rossi”, Data di nascita: “15/03/1990”, Indirizzo: “Via Milano 15”, Numero di telefono: “1234567890”, Codice postale: “1234a” ORACOLO: Messaggio di errore accanto al campo codice postale: “il codice postale deve essere composto da 5 caratteri solo numerici”.- TC10: Email: “apuapu@gmail.com”, Password: “Apu2024”, Nome: “Marco”, Cognome: “Rossi”, Data di nascita: “15/03/1990”, Indirizzo: “Via Milano 15”, Numero di telefono: “1234567890”, Codice postale: “12345” ORACOLO: Registrazione effettuata con successo.

8.2. Autenticazione

PARAMETRI	Email, password
OGGETTI DELL'AMBIENTE	Database
CATEGORIE	C1: email presente nel database C2: password associata a email nel database
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none">- EP1: email presente nel database [<i>property email_presente</i>]- EP2: email NON presente nel database SCELTE PER C2: <ul style="list-style-type: none">- PA1: password associata a email nel database [<i>if email_presente</i>]- PA2: password NON associata a email nel database
TEST FRAME	<ul style="list-style-type: none">- TF1: EP1, PA1 ORACOLO: Utente autenticato e ridiretto alla sua homepage- TF2: EP1, PA2 ORACOLO: Messaggio di errore: "Credenziali errate"- TF3: EP2, PA2 ORACOLO: Messaggio di errore: "Credenziali errate"
TEST CASE	<ul style="list-style-type: none">- TC1: email = "homersimpson7@gmail.com", password = "donuts7G?" In DB presente la coppia (email = "homersimpson7@gmail.com", password = "donuts7G?") – ORACOLO: Utente autenticato con successo e ridiretto alla sua homepage- TC2: email = "homersimpson7@gmail.com", password = "1234" – In DB presente la coppia (email = "homersimpson7@gmail.com", password = "donuts7G") – ORACOLO: La pagina di autenticazione mostra il messaggio: "credenziali errate"- TC3: email = "homer@gmail.com", password = "1234" – In DB NON presente la coppia (email = "homer@gmail.com", password = "1234") ORACOLO: La pagina di autenticazione mostra il messaggio: "credenziali errate"

8.3. Aggiunta Fumetto al carrello

PARAMETRI	id_fumetto, id_Carrello
OGGETTI DELL'AMBIENTE	Database, File System
CATEGORIE	C1: fumetto aggiunto al carrello
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none">- IP1: id_fumetto presente su Catalogo [<i>property id_fumetto_presente_catalogo</i>]- IP2: id_fumetto NON presente su Catalogo- IP3: quantità selezionata è disponibile [<i>property quantita_selezionata_disponibile</i>]- IP4: quantità selezionata NON disponibile- IP5: id_fumetto già presente nel carrello [<i>property id_fumetto_presenteCarrello</i>]- IP6: id_fumetto NON presente nel carrello
TEST FRAME	<ul style="list-style-type: none">- TF1: IP1, IP3, IP6<ul style="list-style-type: none">- ORACOLO: Il cliente ora ha il fumetto selezionato aggiunto al carrello- TF2: IP2<ul style="list-style-type: none">- ORACOLO: Nessun fumetto viene aggiunto al carrello- TF3: IP4<ul style="list-style-type: none">- ORACOLO: Messaggio di errore: "Quantità selezionata non disponibile"- TF4: IP1, IP3, IP5<ul style="list-style-type: none">- ORACOLO: Il cliente ora ha la quantità del fumetto nel carrello aggiornata- TF4: IP1, IP4, IP5<ul style="list-style-type: none">- ORACOLO: Messaggio di errore: "Quantità selezionata non disponibile"
TEST CASE	<ul style="list-style-type: none">- TC1: id_fumetto=5, quantità=3 – Su DB presente fumetto con id=5 dal titolo "Naruto" e quantità = 10<ul style="list-style-type: none">- ORACOLO: Il fumetto è stato aggiunto al carrello e l'utente viene notificato con il messaggio "Fumetto Aggiunto al carrello"

	<ul style="list-style-type: none"> - TC2: id_fumetto=5 quantità = 3– Su DB presente fumetto con id=5 dal titolo “Naruto” <ul style="list-style-type: none"> - ORACOLO: Il fumetto è già presente nel carrello, la quantità di fumetti nel carrello viene aggiornata
--	---

8.4. Rimozione Fumetto dal carrello

PARAMETRI	id_fumetto, id_Carrello
OGGETTI DELL'AMBIENTE	Database, File System
CATEGORIE	C1: fumetto rimosso dal carrello
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none"> - IP1: id_fumetto presente sul Carrello[<i>property id_fumetto_presente_carrello</i>] - IP2: id_fumetto NON presente sul Carrello - IP3: più di un prodotto con lo stesso id_prodotto presente nel carrello [<i>property_quantita</i>] - IP4: un solo fumetto con id_fumetto presente nel carrello
TEST FRAME	<ul style="list-style-type: none"> - TF1: IP1,IP4 <ul style="list-style-type: none"> - ORACOLO: Il fumetto associato all'id è stato rimosso dal carrello - TF2: IP2 <ul style="list-style-type: none"> - ORACOLO: Messaggio di errore: “Fumetto non trovato” - TF3: IP1, IP3 <ul style="list-style-type: none"> - ORACOLO: La quantità del fumetto associato all'id del fumetto è stata modificata
TEST CASE	<ul style="list-style-type: none"> - TC1: presente fumetto nel carrello con id=5 dal titolo “Naruto” e quantità = 1 <ul style="list-style-type: none"> - ORACOLO: Il fumetto è stato rimosso dal carrello e l'utente viene notificato con il messaggio “Fumetto Rimosso dal carrello”

	<ul style="list-style-type: none"> - TC2: presente sul carrello un fumetto con id=6 e quantita = 3 dal titolo "DragonBall" <ul style="list-style-type: none"> - ORACOLO: Il fumetto è già presente nel carrello, ma non nel db quindi il prodotto viene cancellato - TC3: presente sul carrello un fumetto con id=6 e quantita = 3 dal titolo "DragonBall" <ul style="list-style-type: none"> - ORACOLO: Il fumetto è già presente nel carrello, la quantita del fumetto a carrello viene aggiornata diminuendo di un fumetto
--	---

8.5. Svuota Carrello

PARAMETRI	id_fumetto, id_Carrello
OGGETTI DELL'AMBIENTE	Database, File System
CATEGORIE	C1: carrello vuoto
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none"> - IP1: id_fumetto presente sul Carrello[<i>property id_fumetto_presente_carello</i>] - IP2: Nessun id_fumetto presente sul Carrello - IP3: quantità selezionata di id_fumetto è minore o uguale a quella disponibile [<i>property quantita_selezionata_disponibile</i>]
TEST FRAME	<ul style="list-style-type: none"> - TF1: IP1,IP3 <ul style="list-style-type: none"> - ORACOLO: carrello svuotato con successo - TF2: IP2 <ul style="list-style-type: none"> - ORACOLO: Messaggio di errore: "Carrello già vuoto"
TEST CASE	<ul style="list-style-type: none"> - TC1: presente fumetti nel carrello con id=5 dal titolo "Naruto" e quantità = 1 e id=6 dal titolo "Dragon ball" <ul style="list-style-type: none"> - ORACOLO: i fumetti sono stati rimossi dal carrello e l'utente viene notificato con il messaggio "Fumetti Rimosso dal carrello"

	<ul style="list-style-type: none"> - TC2: il carrello non ha nessun fumetto - ORACOLO: l'utente viene notificato con il messaggio "Carrello già vuoto"
--	--

8.6. Effettua Acquisto

PARAMETRI	id_fumetto, id_carrello, id_ordine
OGGETTI DELL'AMBIENTE	Database
CATEGORIE	C1: fumetti aggiunti all'ordine
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none"> - IP1: id_fumetto presente sul carrello [<i>property id_fumetto_presente_carrello</i>] - IP2: Nessun id_fumetto presente sul Carrello - IP3: quantità selezionata di id_fumetto è minore o uguale a quella disponibile [<i>property quantita_selezionata_disponibile</i>]
TEST FRAME	<ul style="list-style-type: none"> - TF1: IP1,IP3 ORACOLO: Il fumetto associato all'id è stato rimosso dal carrello e aggiunto all'ordine - TF2: IP2 ORACOLO: Messaggio di errore: "Nessun fumetto trovato"
TEST CASE	<ul style="list-style-type: none"> - TC1: id_fumetto = "12345", id_carrello = "67890" ORACOLO: Il fumetto associato all'id è stato rimosso dal carrello e aggiunto all'ordine - TC2: id_fumetto = "54321" ORACOLO: Messaggio di errore: "Nessun fumetto trovato"

8.7. Aggiunta Fumetto al catalogo

PARAMETRI	id_fumetto, titolo, descrizione, prezzo
OGGETTI DELL'AMBIENTE	Database
CATEGORIE	C1: titolo valido (rispetta il formato della specifica) C2: descrizione valida (rispetta il formato della specifica) C3: prezzo valido (rispetta il formato della specifica)
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none"> - LT1: titolo valido [property titolo_valido] - LT2: titolo NON valido SCELTE PER C2: <ul style="list-style-type: none"> - DV1: descrizione valida [<i>property descrizione_valida</i>] - DV2: descrizione NON valida SCELTE PER C3: <ul style="list-style-type: none"> - PV1: prezzo valido [property prezzo_valido] - PV2: il prezzo NON valido
TEST FRAME	<ul style="list-style-type: none"> - TF1: LT2 ORACOLO: Messaggio di errore accanto al campo titolo: "titolo non puo contenere caratteri speciali" - TF2: LT1, DV2 ORACOLO: Messaggio di errore accanto al campo descrizione: "il campo non può contenere caratteri speciali" - TF3: LT1, DV1, PV2 ORACOLO: Messaggio di errore accanto al campo prezzo: "il prezzo puo contenere solo numeri" - TF4: LT1, DV1, PV1 ORACOLO: "Fumetto aggiornato correttamente"

TEST CASE	<ul style="list-style-type: none"> - TC1: id_fumetto = "125" titolo = "Fumetto!@#" ORACOLO: Messaggio di errore accanto al campo titolo: "titolo non può contenere caratteri speciali" - TC2: id_fumetto = "125" titolo = "paperino" descrizione = "Storia avvincente!!" ORACOLO: Messaggio di errore accanto al campo descrizione: "il campo non può contenere caratteri speciali" - TC3: id_fumetto = "125" titolo = "paperino" descrizione = "Storia avvincente" prezzo = "19,99ss" ORACOLO: Messaggio di errore accanto al campo prezzo: "il prezzo può contenere solo numeri" - TC4: titolo = id_fumetto = "125" titolo = "paperino" descrizione = "Storia avvincente" prezzo = "19,99" ORACOLO: "Fumetto aggiornato correttamente"
-----------	--

8.8. Rimozione Fumetto dal catalogo

PARAMETRI	id_fumetto
OGGETTI DELL'AMBIENTE	Database
CATEGORIE	C1: id_fumetto presente nel catalogo
SCELTE E VINCOLI RELATIVI	SCELTE PER C1: <ul style="list-style-type: none"> - IP1: id_fumetto presente su Catalogo [<i>property id_fumetto_presente_catalogo</i>] - IP2: id_fumetto NON presente sul Carrello
TEST FRAME	<ul style="list-style-type: none"> - TF1: IP1 ORACOLO: fumetto rimosso dal catalogo - TF2: IP2 ORACOLO: fumetto non trovato
TEST CASE	<ul style="list-style-type: none"> - TC1: presente fumetto nel catalogo con id=5 dal titolo "Naruto" ORACOLO: Il fumetto è stato rimosso dal catalogo - TC1: nessun fumetto presente con id=5 nel catalogo ORACOLO: Il fumetto non è stato trovato

9. Test Schedule

La scrittura dei casi di test sarà effettuata parallelamente allo sviluppo del codice. Il test di ogni unità verrà eseguito immediatamente dopo la sua implementazione, mentre il test di integrazione e il test di sistema saranno avviati solo una volta completati con successo i test di unità precedenti.