

Titolo

Autori

12 luglio 2024

# Indice

<b>1</b>	<b>Specifiche Informali</b>	<b>3</b>
<b>2</b>	<b>Analisi e Specifica dei Requisiti</b>	<b>4</b>
2.1	Analisi Nomi-Verbi . . . . .	4
2.2	Revisione dei Requisiti . . . . .	4
2.2.1	Requisiti Aggiuntivi . . . . .	5
2.3	Glossario dei Termini . . . . .	5
2.4	Classificazione dei Requisiti . . . . .	6
2.4.1	Requisiti Funzionali . . . . .	6
2.4.2	Requisiti sui Dati . . . . .	6
2.4.3	Vincoli/Altri Requisiti . . . . .	6
<b>3</b>	<b>Modellazione dei Casi d'Uso</b>	<b>7</b>
3.1	Attori e Casi d'Uso . . . . .	7
3.2	Diagramma dei Casi d'Uso . . . . .	8
3.3	Scenari . . . . .	9
3.4	Diagramma delle Classi . . . . .	14
3.5	Diagrammi di Sequenza . . . . .	14
3.6	Verifica Completezza dei Requisiti . . . . .	20
<b>4</b>	<b>Piano di Test Funzionale</b>	<b>21</b>
4.1	RegistraCliente . . . . .	21
4.2	LoginUtente . . . . .	23
4.3	CercaFarmaco . . . . .	24
4.4	ModificaFarmaco . . . . .	24
4.5	AggiungiFarmaco . . . . .	25
4.6	EliminaFarmaco . . . . .	26
4.7	RegistraConsegnaOrdineAcquisto . . . . .	27
4.8	RitiraOrdine . . . . .	28
4.9	GeneraOrdineAcquistoFarmacista . . . . .	28
4.10	CreaOrdine . . . . .	29
<b>5</b>	<b>Progettazione</b>	<b>29</b>
5.1	Diagramma delle classi . . . . .	29
5.2	Modello Entity-Relationship del Database . . . . .	30
5.3	Diagrammi di sequenza . . . . .	30
<b>6</b>	<b>Implementazione</b>	<b>31</b>
<b>7</b>	<b>Testing</b>	<b>32</b>
7.1	Test strutturale . . . . .	32
7.1.1	creaOrdine() . . . . .	32

## Capitolo 4

# Piano di Test Funzionale

### 4.1 RegistraCliente

#### Category Partition Testing

Nome	Cognome	Username	Password	Email	DataNascita
<ul style="list-style-type: none"><li>• Stringa di lunghezza <math>\leq 45</math></li><li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li><li>• Stringa vuota [ERROR]</li></ul>	<ul style="list-style-type: none"><li>• Stringa di lunghezza <math>\leq 45</math></li><li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li><li>• Stringa vuota [ERROR]</li></ul>	<ul style="list-style-type: none"><li>• Stringa di lunghezza <math>\leq 45</math></li><li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li><li>• Stringa vuota [ERROR]</li><li>• Stringa già presente nel sistema [ERROR]</li></ul>	<ul style="list-style-type: none"><li>• Stringa di lunghezza compresa tra 8 e 45</li><li>• Stringa di lunghezza <math>&lt; 8</math> [ERROR]</li><li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li></ul>	<ul style="list-style-type: none"><li>• Stringa di lunghezza <math>\leq 45</math></li><li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li><li>• Stringa vuota [ERROR]</li><li>• Stringa già presente nel sistema [ERROR]</li></ul>	<ul style="list-style-type: none"><li>• Data con formato valido (gg-mm-aaaa)</li><li>• Data con formato non valido [ERROR]</li></ul>

Il numero di test da effettuarsi senza particolari vincoli è:  $3 \cdot 3 \cdot 4 \cdot 3 \cdot 4 \cdot 2 = 864$ .

Introduciamo i vincoli [ERROR]. Il numero di test da eseguire per testare singolarmente i vincoli è 13 (2 per Nome, 2 per Cognome, 3 per Username, 2 per Password, 3 per Email, 1 per DataNascita).

Il numero di test risultante è 14:  $(1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1) + 13 = 14$ .

#### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti gli input validi	Nome, Cognome, Username, Password, Email, DataNascita validi	Il cliente non è ancora registrato nel sistema	Nome: Mario Cognome: Rossi Username: mariorossi Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Registrazione effettuata	Il cliente è stato correttamente registrato nel sistema
2	Nome $> 45$ caratteri	Nome $> 45$ caratteri [ERROR], Cognome, Username, Password, Email, DataNascita validi	–	Nome: ... Cognome: Rossi Username: mariorossi Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Nome troppo lungo	–
3	Nome assente	Nome assente [ERROR], Cognome, Username, Password, Email, DataNascita validi	–	Nome: Cognome: Rossi Username: mariorossi Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Inserire un nome	–

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
4	Cognome > 45 caratteri	Cognome > 45 caratteri [ERROR], Nome, Username, Password, Email, DataNascita validi	–	Nome: Mario Cognome: ... Username: mariorossi Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Cognome troppo lungo	–
5	Cognome assente	Cognome assente [ERROR], Nome, Username, Password, Email, DataNascita validi	–	Nome: Mario Cognome: Username: mariorossi Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Inserire un cognome	–
6	Username > 45 caratteri	Username > 45 caratteri [ERROR], Nome, Cognome, Password, Email, DataNascita validi	–	Nome: Mario Cognome: Rossi Username: ... Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Username troppo lungo	–
7	Username assente	Username assente [ERROR], Nome, Cognome, Password, Email, DataNascita validi	–	Nome: Mario Cognome: Rossi Username: Password: miapassword Email: mariorossi@gmail.com DataNascita: 22-06-1989	Inserire un username	–
8	Password < 8 caratteri	Password < 8 caratteri [ERROR], Nome, Cognome, Username, Email, DataNascita validi	–	Nome: Mario Cognome: Rossi Username: mariorossi Password: prova Email: mariorossi@gmail.com DataNascita: 22-06-1989	Password troppo corta	–
9	Password > 45 caratteri	Password > 45 caratteri [ERROR], Nome, Cognome, Username, Email, DataNascita validi	–	Nome: Mario Cognome: Rossi Username: mariorossi Password: ... Email: mariorossi@gmail.com DataNascita: 22-06-1989	Password troppo lunga	–
10	Email > 45 caratteri	Email > 45 caratteri [ERROR], Nome, Cognome, Username, Password, DataNascita validi	–	Nome: Mario Cognome: Rossi Username: mariorossi Password: miapassword Email: ... DataNascita: 22-06-1989	Email troppo lunga	–
11	Email assente	Email assente [ERROR], Nome, Cognome, Username, Password, DataNascita validi	–	Nome: Mario Cognome: Rossi Username: Password: miapassword Email: DataNascita: 22-06-1989	Inserire un'e-mail	–
12	Formato DataNascita non valida	Formato DataNascita non valida [ERROR], Nome, Cognome, Username, Password, Email validi	–	Nome: Mario Cognome: Rossi Username: mariorossi Password: miapassword Email: mariorossi@gmail.com DataNascita: 1989-06-22	Formato data non valido	–

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
13	Username già presente nel sistema	Username già esistente [ERROR], Nome, Cognome, Password, DataNascita ed Email validi	Username “pippo2002” già presente nel sistema	Nome: Pippo Cognome: Baudo Username: pippo2002 Password: miapassword Email: pippo@gmail.com DataNascita: 1989-06-22	Username già utilizzato	–
14	Email già presente nel sistema	Email già esistente [ERROR], Nome, Cognome, Password, DataNascita e Username validi	Email “pippo@gmail.com” già presente nel sistema	Nome: Pippo Cognome: Baudo Username: pippo2002 Password: miapassword Email: pippo@gmail.com DataNascita: 1989-06-22	Email già utilizzata	–

## 4.2 LoginUtente

### Category Partition Testing

Username	Password
<ul style="list-style-type: none"> <li>Stringa di lunghezza <math>\leq 45</math></li> <li>Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>Stringa vuota [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Stringa di lunghezza compresa tra 8 e 45 e corrispondente alla Password memorizzate per l'utente che cerca di effettuare il login</li> <li>Stringa di lunghezza <math>&lt; 8</math> [ERROR]</li> <li>Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>Stringa non corrispondente alla Password memorizzata per l'utente che cerca di effettuare il login [ERROR]</li> </ul>

Il numero di test da effettuarsi senza particolari vincoli è:  $3 \cdot 4 = 12$ .

Introduciamo i vincoli [ERROR]. Il numero di test da eseguire per testare singolarmente i vincoli è 5 (2 per Username, 3 per Password).

Il numero di test risultante è 6:  $(1 \cdot 1) + 5 = 6$ .

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti gli input validi	Username, Password validi	L'utente deve essere correttamente registrato nel sistema	Username: mario Password: mia-password	Login effettuato	L'utente è entrato correttamente nel sistema
2	Username $> 45$ caratteri	Username $> 45$ caratteri [ERROR], Password valida	–	Username: ... Password: mia-password	Username troppo lungo	–
3	Username assente	Username assente [ERROR], Password valida	–	Username: Password: mia-password	Inserire un username	–
4	Password $< 8$ caratteri	Password $< 8$ caratteri [ERROR], Username valido	–	Username: mario Password: prova	Password troppo corta	–
5	Password $> 45$ caratteri	Password $> 45$ caratteri [ERROR], Username valido	–	Username: mario Password: ...	Password troppo lunga	–
6	Password valida ma errata	Password errata [ERROR], Username valido	L'utente esiste nel sistema e ha come password 'passwd'	Username: mario Password: ciao	Password errata	–

## 4.3 CercaFarmaco

### Category Partition Testing

Nome
<ul style="list-style-type: none"> <li>• Stringa di lunghezza <math>\leq 45</math></li> <li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>• Stringa vuota [ERROR]</li> <li>• Nome di un farmaco non presente nel sistema [ERROR]</li> </ul>

Essendo previsto un solo input, il numero di test da effettuarsi è pari a 4.

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Nome del farmaco valido	Nome del farmaco valido	Il farmaco è presente nel catalogo	Nome: Tachipirina	Il farmaco viene mostrato a video	–
2	Nome del farmaco $> 45$ caratteri	Nome del farmaco $> 45$ caratteri [ERROR]	–	Nome: ...	Nome del farmaco troppo lungo	–
3	Nome del farmaco assente	Nome del farmaco assente [ERROR]	–	Nome:	Inserire il nome di un farmaco	–
4	Nome di un farmaco che non esiste	Nome di un farmaco che non esiste [ERROR]	Il farmaco con nome “Tachipirina” non esiste	Nome: Tachipirina	Il farmaco non esiste	–

## 4.4 ModificaFarmaco

### Category Partition Testing

Nome	Prezzo	Prescrizione	Scorte
<ul style="list-style-type: none"> <li>• Stringa di lunghezza <math>\leq 45</math></li> <li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>• Stringa vuota [ERROR]</li> <li>• Stringa non presente nel sistema [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>• Numero reale <math>&gt; 0</math> (€)</li> <li>• Numero reale <math>\leq 0</math> (€) [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>• True (necessaria) / False (non necessaria)</li> </ul>	<ul style="list-style-type: none"> <li>• Numero intero <math>\geq 0</math></li> <li>• Numero intero <math>&lt; 0</math> [ERROR]</li> </ul>

Il numero di test da effettuarsi senza particolari vincoli è:  $4 \cdot 2 \cdot 2 \cdot 2 = 32$ .

Introduciamo i vincoli [ERROR]. Il numero di test da eseguire per testare singolarmente i vincoli è 5 (3 per Nome, 1 per Prezzo, 1 per Scorte).

Il numero di test risultante è 7:  $(1 \cdot 1 \cdot 2 \cdot 1) + 5 = 7$ .

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti gli input validi, prescrizione True	Nome, Prezzo, Scorte e Prescrizione (True) validi	Il farmaco è presente nel sistema	Nome: Tachipirina Prezzo: 11.44 € Scorte: 100 Prescrizione: True	Modifica effettuata	Il farmaco viene correttamente modificato

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
2	Tutti gli input validi, prescrizione <b>False</b>	Nome, Prezzo, Scorte e Prescrizione ( <b>False</b> ) validi	Il farmaco è presente nel sistema	Nome: Tachipirina Prezzo: 11.44 € Scorte: 100 Prescrizione: <b>False</b>	Modifica effettuata	Il farmaco viene correttamente modificato
3	Nome > 45 caratteri	Nome > 45 caratteri [ERROR], Prezzo, Scorte e Prescrizione (sia <b>True</b> che <b>False</b> ) validi	–	Nome: ... Prezzo: 22.00 € Scorte: 50 Prescrizione: <b>boolean</b>	Nome troppo lungo	–
4	Nome assente	Nome assente [ERROR], Prezzo, Scorte e Prescrizione (sia <b>True</b> che <b>False</b> ) validi	–	Nome: Prezzo: 9.99 € Scorte: 44 Prescrizione: <b>boolean</b>	Inserire un nome	–
5	Prezzo $\leq 0$ (€)	Prezzo $\leq 0$ (€) [ERROR], Nome, Scorte e Prescrizione (sia <b>True</b> che <b>False</b> ) validi	–	Nome: Augmentin Prezzo: -4.00 € Scorte: 10 Prescrizione: <b>boolean</b>	Inserire un prezzo $> 0$ €	–
6	Scorte $< 0$	Scorte $< 0$ , Nome, Prezzo e Prescrizione (sia <b>True</b> che <b>False</b> ) validi	-	Nome: Neoduplamox Prezzo: 7.50 € Scorte: -8 Prescrizione: <b>boolean</b>	Inserire scorte $\geq 0$	–
7	Il nome del farmaco non esiste	Nome non presente nel sistema [ERROR], Prezzo, Scorte e Prescrizione validi	Il farmaco “Tachipirina” non esiste nel sistema	Nome: Tachipirina Prezzo: 11.44 € Scorte: 100 Prescrizione: <b>False</b>	Il farmaco non esiste	–

## 4.5 AggiungiFarmaco

### Category Partition Testing

Nome	Prezzo	Prescrizione	Scorte
<ul style="list-style-type: none"> <li>Stringa di lunghezza <math>\leq 45</math></li> <li>Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>Stringa vuota [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li>Numero reale <math>&gt; 0</math> (€)</li> <li>Numero reale <math>\leq 0</math> (€) [ERROR]</li> </ul>	<ul style="list-style-type: none"> <li><b>True</b> (necessaria) / <b>False</b> (non necessaria)</li> </ul>	<ul style="list-style-type: none"> <li>Numero intero <math>\geq 0</math></li> <li>Numero intero <math>&lt; 0</math> [ERROR]</li> </ul>

Il numero di test da effettuarsi senza particolari vincoli è:  $3 \cdot 2 \cdot 1 \cdot 2 = 12$ .

Introduciamo i vincoli [ERROR]. Il numero di test da eseguire per testare singolarmente i vincoli è 4 (2 per Nome, 1 per Prezzo, 1 per Scorte).

Il numero di test risultante è 5:  $(1 \cdot 1 \cdot 1 \cdot 1) + 4 = 5$ .

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti gli input validi	Nome, Prezzo, Scorte e Prescrizione (sia <b>True</b> che <b>False</b> ) validi	Il farmaco non è presente nel sistema	Nome: Rocefin Prezzo: 15.00 € Scorte: 60 Prescrizione: <b>boolean</b>	Farmaco aggiunto	Il farmaco viene correttamente aggiunto al catalogo

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
2	Nome > 45 caratteri	Nome > 45 caratteri [ERROR], Prezzo, Scorte e Prescrizione (sia True che False) validi	–	Nome: ... Prezzo: 16.10 € Scorte: 23 Prescrizione: boolean	Nome troppo lungo	–
3	Nome assente	Nome assente [ERROR], Prezzo, Scorte e Prescrizione (sia True che False) validi	–	Nome: Prezzo: 9.99 € Scorte: 10 Prescrizione: boolean	Inserire un nome	–
4	Prezzo ≤ 0 (€)	Prezzo ≤ 0 (€) [ERROR], Nome, Scorte e Prescrizione (sia True che False) validi	–	Nome: Brufen Prezzo: -8.00 € Scorte: 200 Prescrizione: boolean	Inserire un prezzo > 0 €	–
5	Scorte < 0	Scorte < 0, Nome, Prezzo e Prescrizione (sia True che False) validi	-	Nome: Macladin Prezzo: 11.50 € Scorte: -36 Prescrizione: boolean	Inserire scorte ≥ 0	–

## 4.6 EliminaFarmaco

### Category Partition Testing

Nome
<ul style="list-style-type: none"> <li>• Stringa di lunghezza ≤ 45</li> <li>• Stringa di lunghezza &gt; 45 [ERROR]</li> <li>• Stringa vuota [ERROR]</li> <li>• Stringa non presente nel sistem [ERROR]</li> </ul>

Essendo previsto un solo input, il numero di test da effettuarsi è pari a 4.

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Nome del farmaco valido	Nome del farmaco valido	Il farmaco è presente nel catalogo	Nome: Tachipirina	Farmaco cancellato	Il farmaco viene cancellato dal catalogo
2	Nome del farmaco > 45 caratteri	Nome del farmaco > 45 caratteri [ERROR]	–	Nome: ...	Nome del farmaco troppo lungo	–
3	Nome del farmaco assente	Nome del farmaco assente [ERROR]	–	Nome:	Inserire il nome di un farmaco	–
4	Il farmaco non esiste	Farmaco non presente nel sistema [ERROR]	Non esiste il farmaco chiamato “Tachipirina”	Nome: Tachipirina	Non puoi eliminare un farmaco che non esiste	–



## 4.7 RegistraConsegnaOrdineAcquisto

### Category Partition Testing

Id dell'ordine
<ul style="list-style-type: none"> <li>• Stringa di lunghezza <math>\leq 45</math></li> <li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>• Stringa vuota [ERROR]</li> <li>• Stringa non presente nel sistema [ERROR]</li> </ul>

Essendo previsto un solo input, il numero di test da effettuarsi è pari a 4.

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Id dell'ordine valido	Id dell'ordine valido	–	Id: 5ea930bc-f0a5-427a-8ca1-f9a2a6146948	Ordine ritirato	Lo stato dell'ordine è stato cambiato correttamente e le scorte in magazzino sono state correttamente incrementate
2	Id > 45 caratteri	Id > 45 caratteri [ERROR]	–	Id: ...	Id troppo lungo	–
3	Id assente	Id assente [ERROR]	–	Id:	Inserire l'id di un ordine	–
4	Id inesistente	Id inesistente [ERROR]	Non esiste un farmaco con id "5ea930bc-f0a5-427a-8ca1-f9a2a6146948"	Id: 5ea930bc-f0a5-427a-8ca1-f9a2a6146948	L'ordine non esiste	–

## 4.8 RitiraOrdine

### Category Partition Testing

idOrdine
<ul style="list-style-type: none"> <li>• Stringa di lunghezza <math>\leq 45</math></li> <li>• Stringa di lunghezza <math>&gt; 45</math> [ERROR]</li> <li>• Stringa vuota [ERROR]</li> <li>• Stringa non presente nel sistema [ERROR]</li> </ul>

Essendo previsto un solo input, il numero di test da effettuarsi è pari a 4.

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	ID dell'ordine valido	ID dell'ordine valido	–	ID: 5ea930bc-f0a5-427a-8ca1-f9a2a6146948	Stato ordine cambiato con successo	Lo stato dell'ordine è stato cambiato con successo
2	ID dell'ordine $> 45$ caratteri	ID dell'ordine $> 45$ caratteri [ERROR]	–	ID: ...	ID dell'ordine troppo lungo	–
3	ID dell'ordine assente	ID dell'ordine assente [ERROR]	–	ID:	Inserire l'ID di un ordine	–
4	Id inesistente	Id inesistente [ERROR]	Non esiste un farmaco con id "5ea930bc-f0a5-427a-8ca1-f9a2a6146948"	Id: 5ea930bc-f0a5-427a-8ca1-f9a2a6146948	L'ordine non esiste	–

## 4.9 GeneraOrdineAcquistoFarmacista

Farmaci-Quantità
<ul style="list-style-type: none"> <li>• Lista di coppie (nomeFarmaco, quantità) non vuota, in cui per ogni nomeFarmaco esiste un corrispondente farmaco nel sistema.</li> <li>• Lista vuota [ERROR]</li> </ul>

Essendo previsto un solo input, il numero di test da effettuarsi è pari a 2.

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Ordine valido	Farmaci-Quantità valido	Esistono nel sistema i farmaci 'Tachipirina' e 'Fluifort'	[('Tachipirina', 5), ('Fluifort', 10)]	Ordine di acquisto generato	Un ordine di acquisto viene correttamente creato
2	Ordine vuoto	Lista vuota [ERROR]	–	–	Non puoi creare un ordine vuoto	–

## 4.10 CreaOrdine

Farmaci-Quantità
<ul style="list-style-type: none"> <li>• Lista di coppie (nomeFarmaco, quantità) non vuota, in cui per ogni nomeFarmaco esiste un corrispondente farmaco nel sistema e la quantità richiesta è &lt; alle scorte in magazzino (per ogni farmaco).</li> <li>• Lista di coppie (nomeFarmaco, quantità) non vuota, in cui per ogni nomeFarmaco esiste un corrispondente farmaco nel sistema e ed esiste almeno un farmaco la cui quantità richiesta è esattamente uguale alle scorte in magazzino.</li> <li>• Lista vuota [ERROR]</li> <li>• Lista di coppie (nomeFarmaco, quantità) in cui per ogni nomeFarmaco esiste un corrispondente farmaco nel sistema e in cui esiste almeno un farmaco la cui quantità richiesta è &gt; delle scorte in magazzino [ERROR].</li> </ul>

Essendo previsto un solo input, il numero di test da effettuarsi è pari a 4.

### Test Suite

Test Case ID	Descrizione	Classi di Equivalenza Coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Ordine valido, l'ordine non esaurisce le scorte di nessun farmaco	Farmaci-Quantità valido	Esistono nel sistema i farmaci 'Tachipirina' e 'Fluifort' con scorte rispettivamente di 80 e 120	[('Tachipirina', 5), ('Fluifort', 10)]	Ordine generato	Un ordine viene correttamente creato e le scorte in magazzino vengono correttamente decrementate
2	Ordine valido, l'ordine esaurisce le scorte di un farmaco	Farmaci-Quantità valido	Esistono nel sistema i farmaci 'Tachipirina' e 'Fluifort' con scorte rispettivamente di 80 e 120	[('Tachipirina', 80), ('Fluifort', 10)]	Ordine generato	Un ordine viene correttamente creato e le scorte in magazzino vengono correttamente decrementate. Un ordine di fornitura viene generato con il farmaco esaurito e una quantità di default
3	Ordine vuoto	Lista vuota [ERROR]	–	–	Non puoi creare un ordine vuoto	–
4	Ordine invalido per scorte insufficienti	Scorte insufficienti [ERROR]	Esistono nel sistema i farmaci 'Tachipirina' e 'Fluifort' con scorte rispettivamente di 80 e 120	[('Tachipirina', 5), ('Fluifort', 150)]	Ordine non creato per scorte insufficienti	–

# Capitolo 7

## Testing

### 7.1 Test strutturale

#### 7.1.1 creaOrdine()

Codice Java

```
1  public String creaOrdine(Map<Integer, Integer> farmaciQuantita) throws
   ↳ OrderCreationFailedException {
2      if (farmaciQuantita.isEmpty())
3          throw new OrderCreationFailedException("Ordine vuoto");
4
5      EntityCatalogo catalogo = EntityCatalogo.getInstance();
6      if (!catalogo.checkScorte(farmaciQuantita)) {
7          throw new OrderCreationFailedException("Ordine non creato per mancanza scorte");
8      }
9      EntityOrdine ordine = new EntityOrdine(this.getId());
10     try {
11         EntityOrdineAcquisto ordineAcquisto = new EntityOrdineAcquisto();
12         for (Map.Entry<Integer, Integer> entry : farmaciQuantita.entrySet()) {
13             int id = entry.getKey();
14             int quantita = entry.getValue();
15             EntityFarmaco farmaco = catalogo.cercaFarmacoById(id);
16             ordine.aggiungiOrdineFarmaco(farmaco, quantita);
17             int scorteResidue = catalogo.decrementaScorte(id, quantita);
18             if (scorteResidue == 0) {
19                 ordineAcquisto.aggiungiOrdineAcquistoFarmaco(farmaco,
20                     ↳ EntityOrdineAcquisto.QUANTITA_ORDINE_DEFAULT);
21             }
22             if (!ordineAcquisto.getQuantitaFarmaci().isEmpty()) {
23                 ordineAcquisto.salvaInDB();
24             }
25             ordine.salvaInDB();
26             storicoOrdini.add(ordine);
27             return ordine.getId();
28         } catch (FarmacoNotFoundException e) {
29             throw new OrderCreationFailedException("Errore creazione ordine, farmaco non
30                 ↳ trovato");
31         } catch (DBException e) {
32             throw new OrderCreationFailedException(e.getMessage());
33         }
34     }
```

## Control Flow Graph

Il numero di cammini linearmente indipendenti è detto *numero cicломatico* di McCabe, e può essere calcolato equivalentemente in uno dei modi seguenti. Sia  $G$  il grafo della funzione, allora risulta:

1.  $V(G) = E - N + 2$  in cui  $E = \# \text{archi in } G$ ,  $N = \# \text{nodi in } G$
2.  $V(G) = P + 1$  con  $P = \# \text{predicati in } G$
3.  $V(G) = R + 1$  con  $R = \# \text{regioni chiuse in } G$

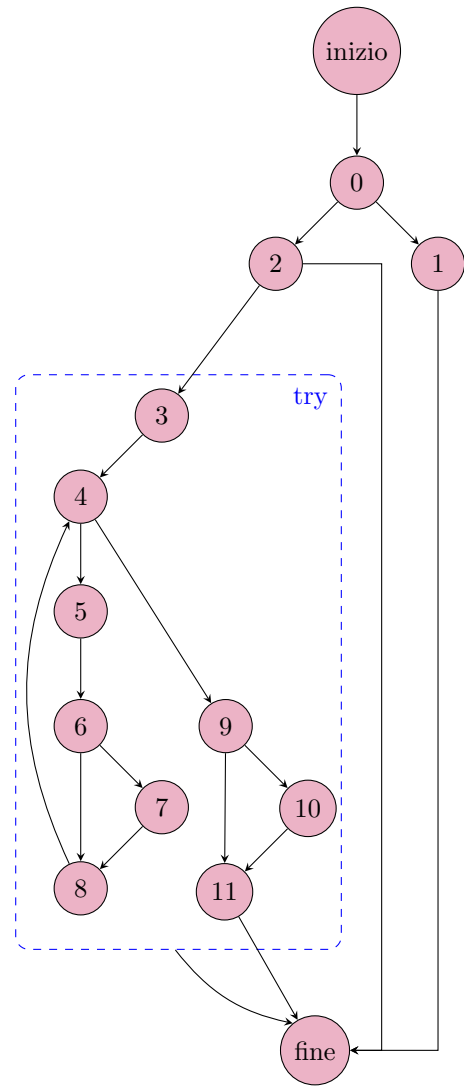
Nel nostro caso:

- $E = 16$
- $N = 12$
- $P = 5$
- $R = 5$

1.  $V(G) = E - N + 2 = 16 - 12 + 2 = 6$
2.  $P + 1 = 5 + 1 = 6$
3.  $R + 1 = 5 + 1 = 6$

I cammini di base sono:

1. 0-1
2. 0-2
3. 0-2-3-4-9-11
4. 0-2-3-4-9-10-11
5. 0-2-3-4-5-6-8-4-9-11
6. 0-2-3-4-5-6-7-8-4-9-10-11



**Test suite strutturale**

Test Case ID	Descrizione	Cammino Coperto	Pre-condizioni	Input	Esito
1	Ordine vuoto	0-1	–	[]	Viene lanciata l'eccezione <code>OrderCreationFailedException("Ordine vuoto")</code>
2	Scorte insufficienti	0-2	Sono presenti 10 scorte del farmaco 'Tachipirina'	[(⟨idTachipirina⟩, 15)]	Viene lanciata l'eccezione <code>OrderCreationFailedException("Ordine non creato per mancanza scorte")</code>
3	Ordine contemporaneamente vuoto e non vuoto	0-2-3-4-9-11	–	[]	Cammino non percorribile
4	Ordine vuoto, viene generato un ordine d'acquisto	0-2-3-4-9-10-11	–	[]	Cammino non percorribile
5	Ordine creato, senza generare un ordine d'acquisto	0-2-3-4-5-6-8-4-9-11	Sono presenti 10 scorte del farmaco 'Tachipirina'	[(⟨idTachipirina⟩, 5)]	Viene creato un ordine
6	Ordine creato, si genera un ordine d'acquisto	0-2-3-4-5-6-7-8-4-9-10-11	Sono presenti 10 scorte del farmaco 'Tachipirina'	[(⟨idTachipirina⟩, 10)]	Ordine creato, parte una richiesta di fornitura per 50 'Tachipirina'

L'id dei farmaci viene scelto dal DB all'atto dell'aggiunta (indice autoincrementale). Pertanto, nel test l'id del farmaco viene ricavato con una funzione esterna.