DOCUMENTO DI DESIGN

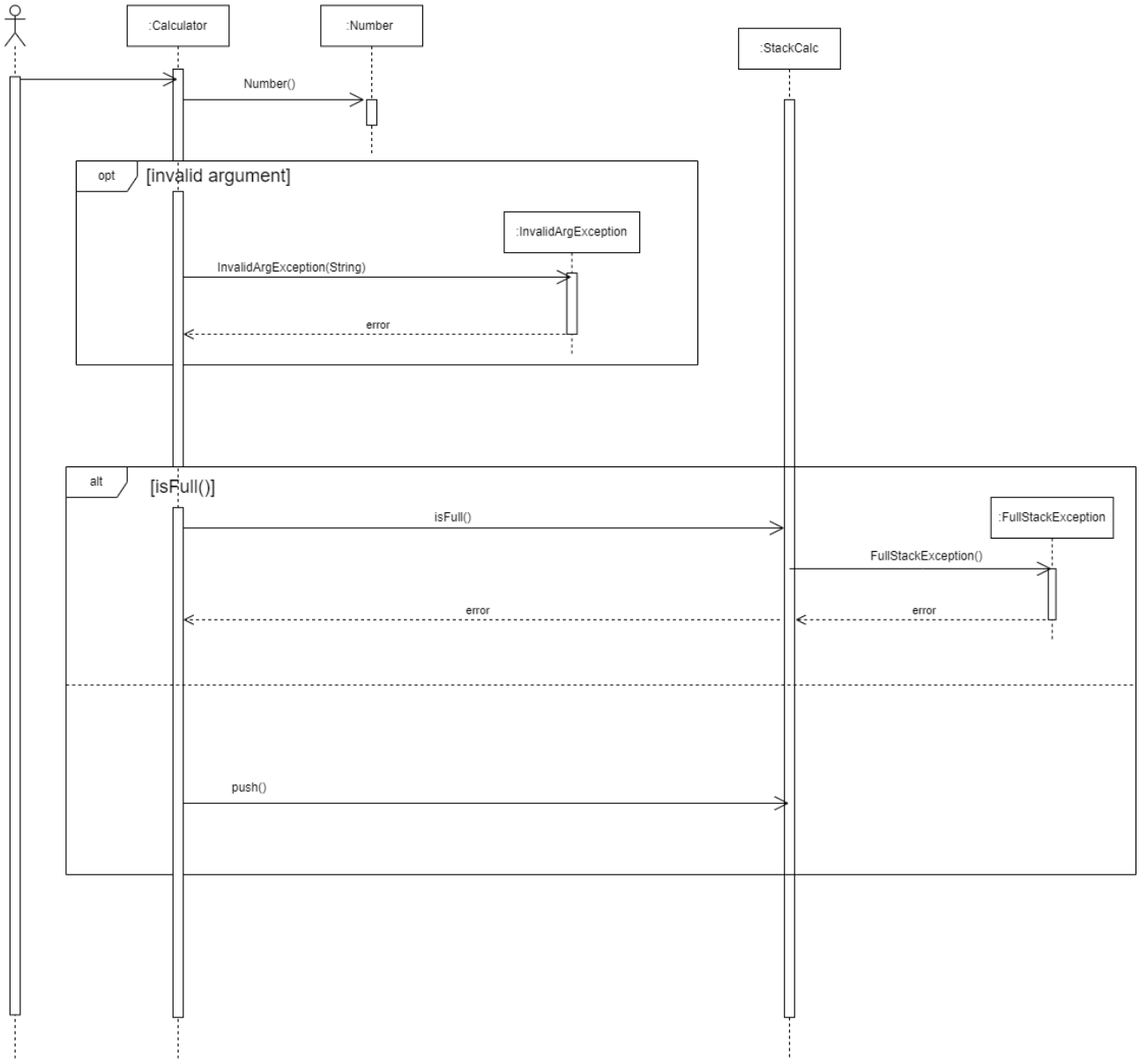
Durante questa fase, nella matrice di tracciabilità è stata aggiunta la colonna che mette in relazione i requisiti ed è stata aggiornata la colonna dedicata alla fase di Design.

Per la realizzazione del class diagram e di ogni sequence diagram è stato utilizzato il tool: draw.io.
È stato preferito il suo utilizzo per la praticità e per la possibilità di condivisione del lavoro in tempo reale con ogni membro del team, facilitando il completamento delle varie task e subtask in totale collaborazione.
Inoltre, si è fatto uso della piattaforma di Google Drive come supporto al tool precedentemente citato.

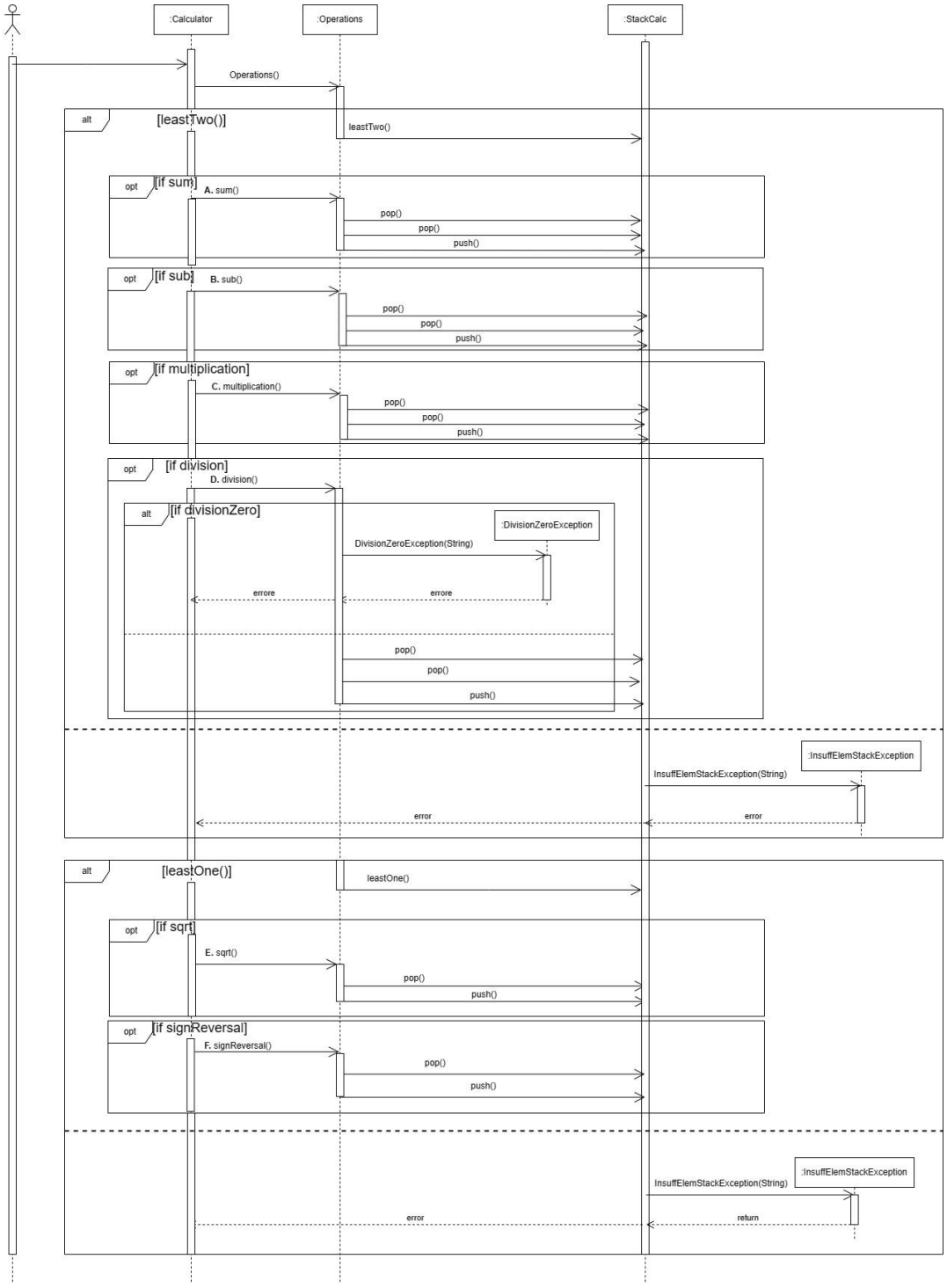
Traceability Matrix (Update)

ID	Requisiti	Design	Implementazione	Testing	Requisiti Connessi
1	implementazione stack	classe StackCalc			
2	supporto della notazione cartesiana per i numeri complessi	classe Number			
3.1	visualizzazione di 12 elementi sullo stack	classe CalculatorViewController			1
3.2	visualizzazione input utente	classe CalculatorViewController			
3.3	visualizzazione variabili	classe CalculatorViewController			
3.4	bottoni manipolazione stack	classe CalculatorViewController			1
3.5	bottoni di: numeri, variabili, operazioni binarie, "j"	classe CalculatorViewController			4.1, 4.2, 4.3, 4.4, 7
3.6	bottoni operazioni unarie	classe CalculatorViewController			4.5, 4.6
4.1	addizione	classe Operations			1, 2, 5, 3.5
4.2	sottrazione	classe Operations			1, 2, 5, 3.5
4.3	moltiplicazione	classe Operations			1, 2, 5, 3.5
4.4	divisione	classe Operations			1, 2, 5, 3.5
4.5	radice quadrata	classe Operations			1, 2, 5, 3.6
4.6	inversione del segno	classe Operations			1, 2, 5, 3.6
5	esecuzione operazioni in ordine di inserimento input	classe CalculatorViewController			
6.1	clear	classe Commands			1, 2, 3.4
6.2	drop	classe Commands			1, 2, 3.4
6.3	dup	classe Commands			1, 2, 3.4
6.4	swap	classe Commands			1, 2, 3.4
6.5	over	classe Commands			1, 2, 3.4
7	supporto di 26 variabili	classe Variables			
8.1	>x	classe Variables			1, 2, 7, 3.5
8.2	<x	classe Variables			1, 2, 7, 3.5
8.3	+x	classe Variables			1, 2, 7, 3.5
8.4	-x	classe Variables			1, 2, 7, 3.5

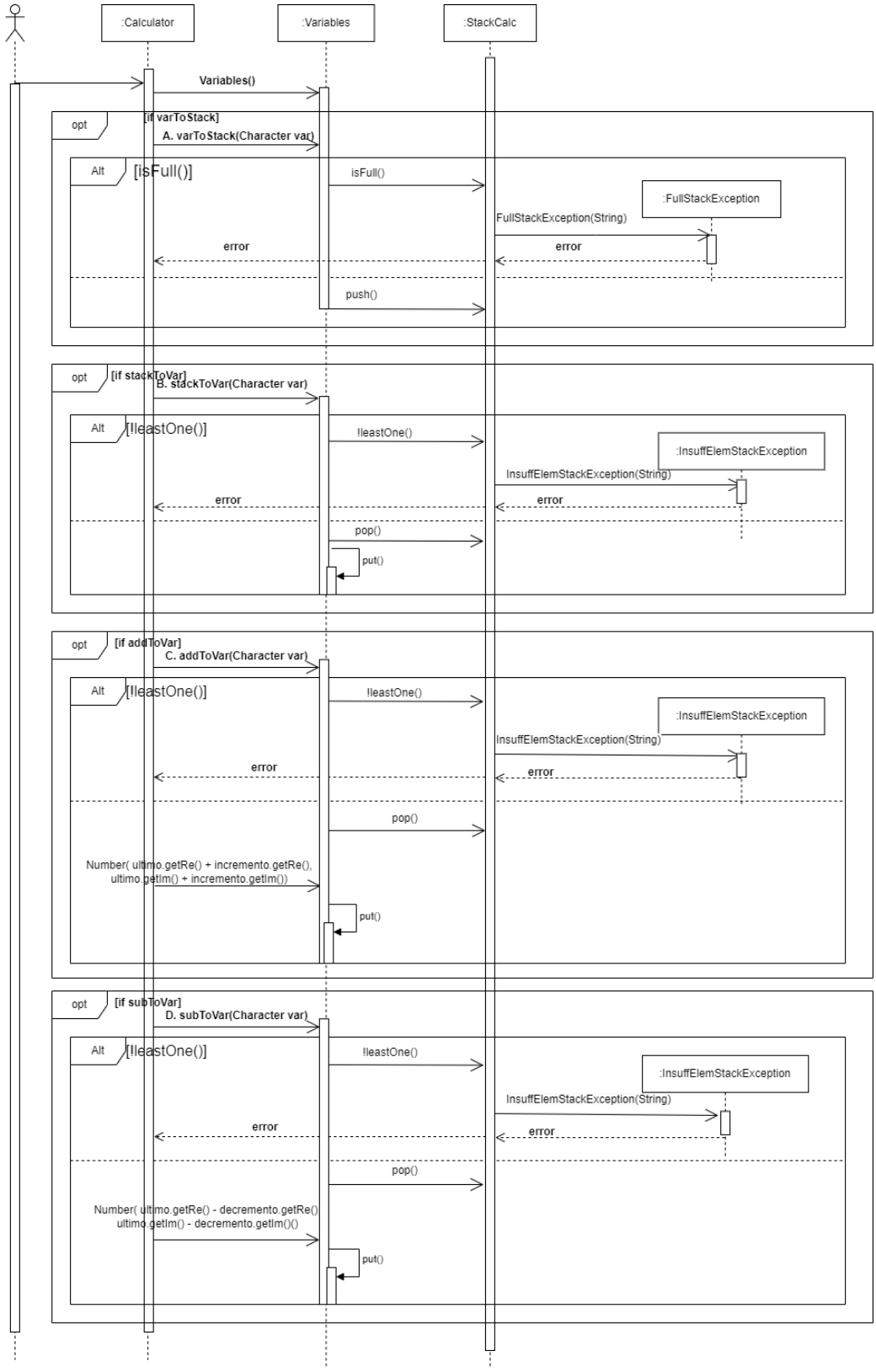
Sequence Diagrams



1. In questo sequence diagram è stato utilizzato un frammento condizionale del tipo “option” per rappresentare la possibilità di dover lanciare un’eccezione nel caso in cui si cerchi di inserire un argomento non valido.
È stato utilizzato un frammento condizionale del tipo “alternative” per gestire la condizione di stack pieno durate l’inserimento di un valore da parte dell’utente.
Se la condizione è verificata, si verifica una eccezione, altrimenti si prosegue con l’inserimento del valore nello stack.

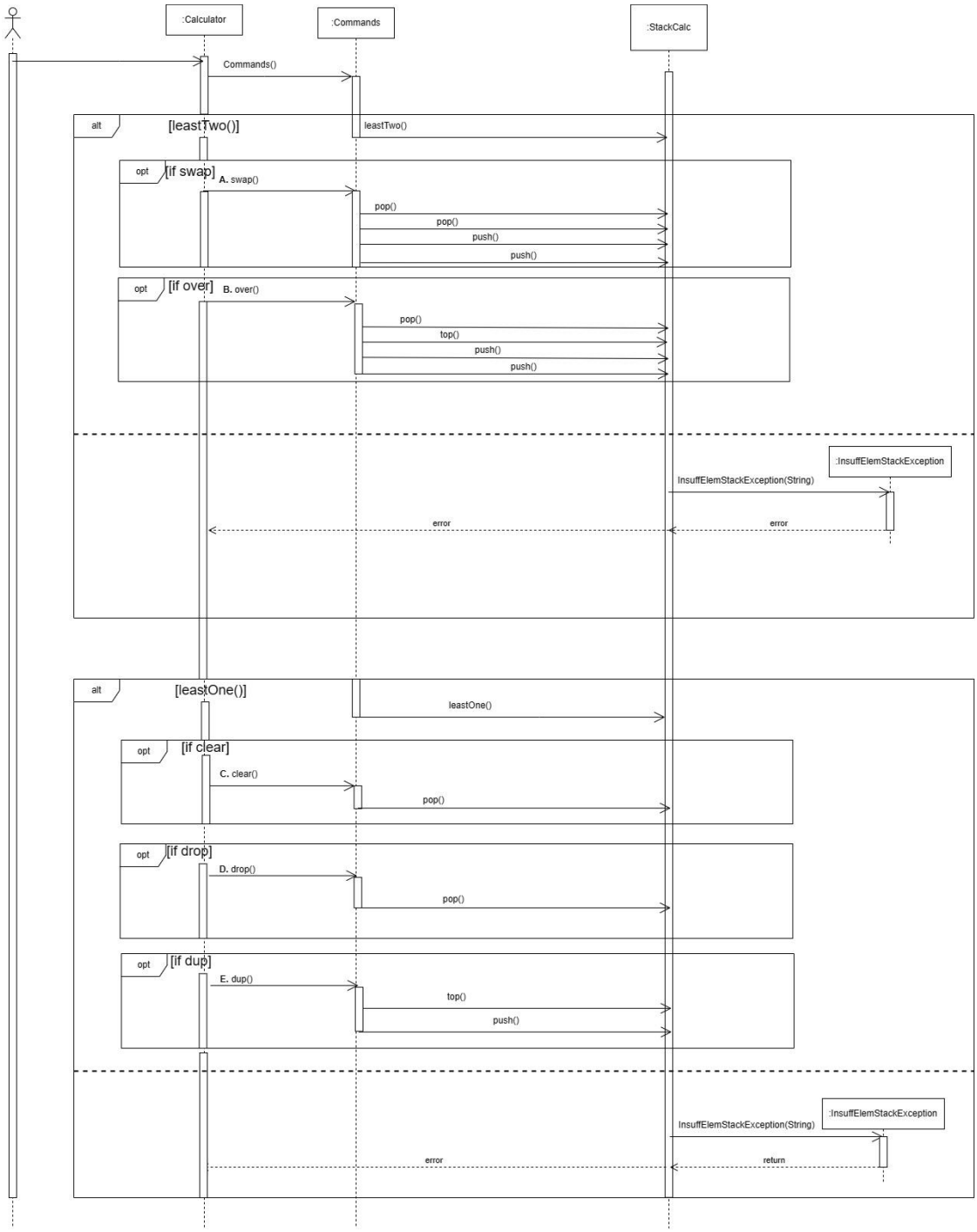


2. Questo sequence diagram rappresenta tutte le operazioni che avvengono sugli elementi dello stack. Può verificarsi solo una delle sei operazioni per volta contrassegnate attraverso dei frammenti condizionali di tipo “option”.
Per le quattro operazioni elementari è fatto un controllo per verificare se sono presenti almeno due elementi nello stack, in caso contrario verrà lanciata un’eccezione.
Inoltre, per l’operazione della divisione è possibile che venga lanciata una eccezione se si cercherà di svolgere una divisione per zero.
Per le operazioni di radice quadrata e inversione del segno, sarà fatto un controllo per verificare che sia presente almeno un elemento all’interno dello stack, altrimenti verrà lanciata una eccezione.



3. Il corrente sequence diagram si occupa della gestione delle variabili, delle operazioni principali che possono essere effettuate su di esse, tramite la comunicazione con la classe StackCalc, che rappresenta lo stack.

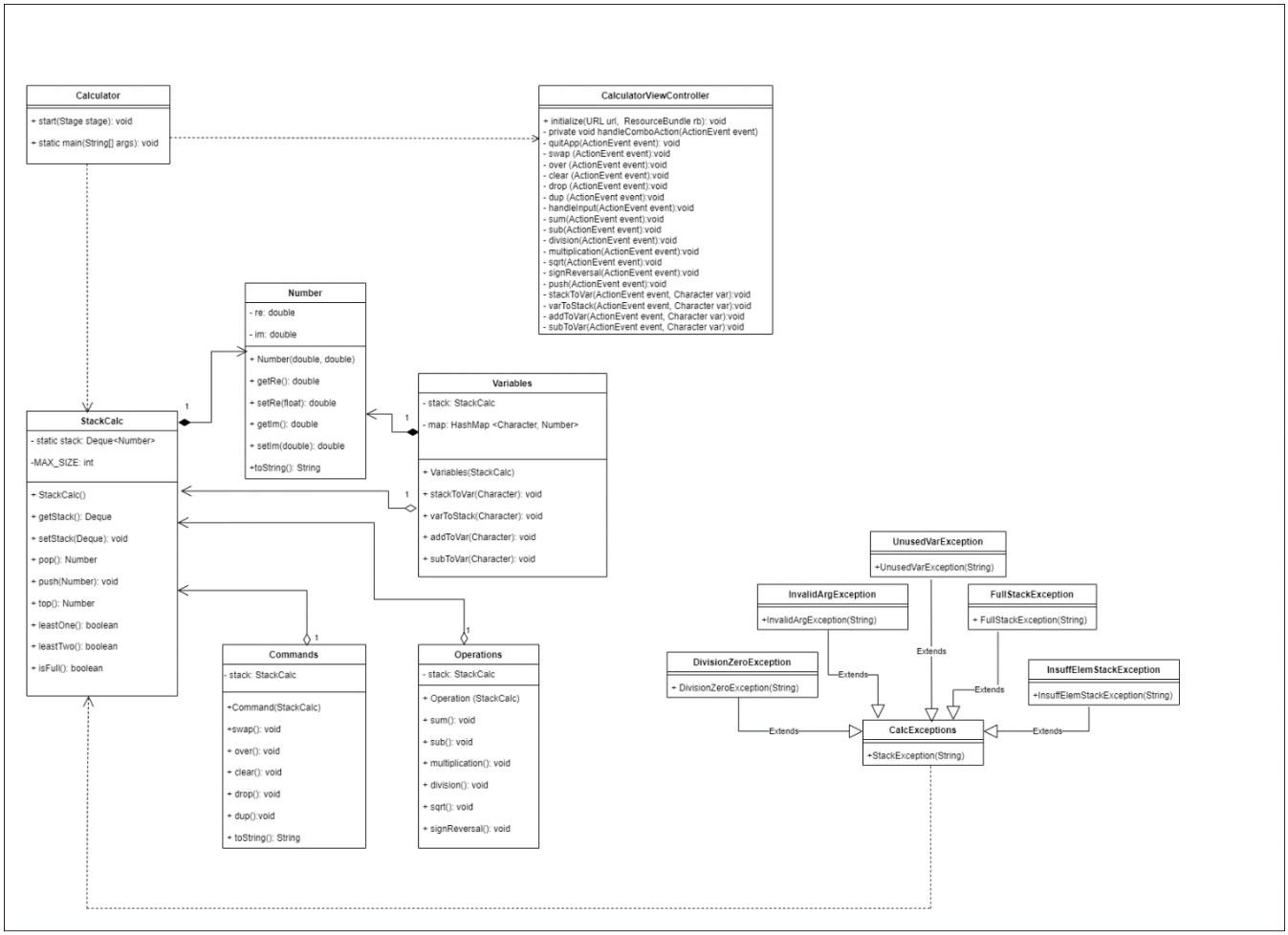
Sono stati utilizzati i frammenti condizionali del tipo “alternative” per gestire i possibili metodi richiesti da utente e due tipi di errori: la condizione di stack pieno durate l’inserimento di una variabile da parte dell’user, e la condizione di elementi non sufficienti per l’operazione richiesta.



4. Questo sequence diagram rappresenta tutti i comandi che avvengono sugli elementi dello stack.
Può verificarsi solo uno dei cinque comandi per volta contrassegnati attraverso dei frammenti condizionali di tipo “option”.
Per i comandi swap e over è fatto un controllo per verificare se sono presenti almeno due elementi nello stack, in caso contrario verrà lanciata una eccezione.
Per i comandi: clear, drop e dup, sarà fatto un controllo per verificare che sia presente almeno un elemento all’interno dello stack, in caso contrario verrà lanciata una eccezione.

*non sono state riportate le eccezioni che possono essere lanciate da ogni metodo presente all’interno della classe StackCalc per motivi di visibilità

Class Diagram



Il diagramma mostra le classi del software e come sono intercollegate, l'obiettivo principale è ottenere una prospettiva esaustiva, che copra tutte le operazioni principali che effettuerà il programma.
La scelta di rappresentazione del Class Diagram è stata di crearne solo uno che desse una visione completa dell'intero progetto software, compresa sia di parte implementativa che di GUI.