

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	8
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	9
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	8
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	7
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	9
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
2	80 <sup>18</sup>	17	20	18	20	93

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	10
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
4	20	20	20	20	20	100

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	10
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
6	20	20	20	20	20	100

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

### Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	10
	Reliability: Does the software perform consistently without failures or errors? 5 points	10
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	18
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	9
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	9
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	9

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
6	20	20	20	20	18	98

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	9
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	9
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
10	20	19	20	19	20	98

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

**Evaluation****Team #13**

Members: Emerzon Ruter Caranto &amp; John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>**Copy this per group**

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	10
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
12	20	20	20	20	20	100

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	4
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	9
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	19
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
14	19	19	19	20	20	97

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

### Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	10
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	18
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	18

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
16	20	20	18	20	18	96

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions
16	Very Nice, keep it up!

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	10
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	17
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	8
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	16
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
19	20	20	17	18	20	97

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions

## Evaluation

Team #13

Members: Emerzon Ruter Caranto & John Angelo Yap

Git Repo: <https://github.com/https-abakada/elevator-concurrency>

Copy this per group

Criteria	Functionality (20 POINTS)	Score
	Completeness: Does the software provide all the features and functionalities it claims to offer? 10 points	10
	Accuracy: Does the software produce correct and expected results? 5 points	5
	Reliability: Does the software perform consistently without failures or errors? 5 points	5
	Usability (20 POINTS)	
	Ease of Use: Is the software intuitive and easy to navigate for its intended users? 10 points	10
	Learnability: How quickly can a new user learn to use the software effectively? 10 points	10
	Performance (20 POINTS)	
	Speed: Does the software respond quickly to user inputs and requests? 20 points	20
	Maintainability (20 POINTS)	
	Code Quality: Is the code well-structured, readable, and documented? 10 points	10
	Error Handling: Does the software handle errors gracefully and provide useful feedback? 10 points	10
	. Documentation (20 POINTS)	

	User Documentation: Are user guides, manuals, and tutorials clear, comprehensive, and easy to follow? 10 points	10
	Technical Documentation: Is there sufficient documentation for developers (diagrams, process flow)? 10 points	10

### Summary

Group #	Functionality	Usability	Performance	Maintainability	Documentation	Total Score
20	20	20	20	20	20	100

### Comments and Suggestions Per Group

Group #	Findings / Comments and Suggestions
20	None So far