With the support of the
Erasmus+ Programme
of the European Union

Questions

Kahoot! for formative assessment
Kahoot! Summary

Rank	Player	Total Score (points)	Q1	How a class is drawn?	Q2	What is a class diagram?	Q3	The main purpose of non-functional requirements is to ensure that the system is usable, effective and efficient	Q4	What is UML?
1	A	7683	0	A class is drawn as a rectangle with two compartments !		A class diagram is a type of implementation diagram.	977	True	0	It is a tool for representing, designing, and documenting software
2	B	7539	725	A class is drawn as a rectangle with three compartments !	689	A class diagram is a type of implementation diagram.	783	True	0	It is a tool for representing, designing, and documenting software
3	C	7237	862	A class is drawn as a rectangle with three compartments !	914	It describes the structure of a system.	0	False	0	It is a tool for representing, designing, and documenting software
4	D	7141	524	A class is drawn as a rectangle with three compartments !	0	A class diagram is a type of implementation diagram.	790	True	0	UML stands for Unified Mapping Language
5	E	6418	0		759	It describes the structure of a system.	588	True	0	It is a tool for representing, designing, and documenting software
6	F	6248	0		0	A class diagram is a type of implementation diagram.	822	True	0	It is a tool for representing, designing, and documenting software
7	G	6209	729	A class is drawn as a rectangle with three compartments !	884	It describes the structure of a system.	0	False	0	It is a tool for representing, designing, and documenting software
8	H	5817	0	A class is drawn as a rectangle with two compartments !	608	A class diagram is a type of implementation diagram.	852	True		It is a tool for representing, designing, and documenting software
9	I	5573	705	A class is drawn as a rectangle with three compartments !	0		956	True	0	It is a tool for representing, designing, and documenting software
10	L	5279	0		680	It describes the structure of a system.	0	False	0	It is a tool for representing, designing, and documenting software
11	M	4939	654	A class is drawn as a rectangle with three compartments !	697	It describes the structure of a system.	797	True	0	UML stands for Unified Mapping Language
12	N	4747	0	A class is drawn as a rectangle with two compartments !	790	It describes the structure of a system.	965	True	0	It is a tool for representing, designing, and documenting software
13	O	4628	0		0		796	True	0	UML stands for Unified Mapping Language
14	P	4429	0	A class is drawn as a rectangle with two compartments !	0	A class diagram is a type of implementation diagram.	954	True	847	It is used for designing, and documenting software
15	Q	4193	0		0	A class diagram is a type of implementation diagram.	707	True	0	It is a tool for representing, designing, and documenting software
16	R	4021	0		564	It describes the structure of a system.	655	True	0	It is a tool for representing, designing, and documenting software
17	S	3849	0	It is drawn as a rectangle and a set of required and provided interfaces!	0	A class diagram is a type of implementation diagram.	764	True	787	It is used for designing, and documenting software
18	T	3839	0	A class is drawn as a rectangle with two compartments !	0	It is a type of behavioural diagram the describes how the system behaves.	951	True	0	UML stands for Unified Mapping Language
19	U	3639	0		0		823	True	0	It is a tool for representing, designing, and documenting software
20	W	3618	542	A class is drawn as a rectangle with three compartments !	0	A class diagram is a type of implementation diagram.	838	True	0	UML stands for Unified Mapping Language
21	X	3279	0	A class is drawn as a rectangle with two compartments !	0	It is a type of behavioural diagram the describes how the system behaves.	526	True	0	UML stands for Unified Mapping Language
22	Y	2986	0	It is drawn as a rectangle and a set of required and provided interfaces!	838	It describes the structure of a system.	889	True	0	It is a tool for representing, designing, and documenting software
23	Z	2917	0	It is drawn as a rectangle and a set of required and provided interfaces!	0	A class diagram is a type of implementation diagram.	789	True	618	It is used for designing, and documenting software
24	AA	537	0		537	It describes the structure of a system.	0		0	UML stands for Unified Mapping Language
25	BB	0	0		0		0		0	

	Q5	What is meant by a sequence diagram?	Q6	Enumerations ...	Q7	You are given the following clipping of a UML2 class diagram. Which of the following statements is true?	Q8	Which of the following statements is true?	Which of the following statements ... compositions (composite aggre...)
It is a tool for representing, designing, and documenting software	0	It implements the sequence of actions that a system must execute.	875	... have the keyword <<enumeration>>.	0	One object of B is associated with 1..* objects of D.	820	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	808	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0	One object of B is associated with 1..* objects of D.	668	Operations may have parameters and return values.	The multiplicity of a composi...
It is a tool for representing, designing, and documenting software	950	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	529	One object of D is associated with at least one object of B.	921	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	507	One object of D is associated with at least one object of B.	771	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	0	It implements the sequence of actions that a system must execute.	901	... have the keyword <<enumeration>>.	0		555	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	896	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	518	One object of D is associated with at least one object of B.	695	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	797	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0	One object of B is associated with 1..* objects of D.	780	Operations may have parameters and return values.	The multiplicity of a composi...
	908	It is a type of UML diagram that shows the order of execution of a system.	0		0		0		When the composite eleme...
It is a tool for representing, designing, and documenting software	0	It implements the sequence of actions that a system must execute.	521	... have the keyword <<enumeration>>.	547	One object of D is associated with at least one object of B.	777	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	585	One object of D is associated with at least one object of B.	708	Operations may have parameters and return values.	When the composite eleme...
	925	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	667	One object of D is associated with at least one object of B.	618	Operations may have parameters and return values.	When the composite eleme...
	946	It is a type of UML diagram that shows the order of execution of a system.	0	Skip this question	649	One object of D is associated with at least one object of B.	704	Operations may have parameters and return values.	
	851	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0		651	Operations may have parameters and return values.	When the composite eleme...
	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	0	One object of B is associated with 1..* objects of D.	752	Operations may have parameters and return values.	When the composite eleme...
It is a tool for representing, designing, and documenting software	894	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0	One object of B is associated with 1..* objects of D.	631	Operations may have parameters and return values.	
It is a tool for representing, designing, and documenting software	778	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0	Direct instances of A exist.	543	Operations may have parameters and return values.	In a composition, a part does...
	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	0		658	Operations may have parameters and return values.	When the composite eleme...
	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	631	One object of D is associated with at least one object of B.	765	Operations may have parameters and return values.	In a composition, a part does...
It is a tool for representing, designing, and documenting software	865	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0	Direct instances of A exist.	593	Operations may have parameters and return values.	In a composition, a part does...
	783	It is a type of UML diagram that shows the order of execution of a system.	0	Skip this question	0		682	Operations may have parameters and return values.	
	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	0	One object of B is associated with 1..* objects of D.	567	Operations may have parameters and return values.	In a composition, a part does...
It is a tool for representing, designing, and documenting software	0	It implements the sequence of actions that a system must execute.	0	... are data type which instances form a list of named literal values.	0	One object of B is associated with 1..* objects of D.	733	Operations may have parameters and return values.	The multiplicity of a composi...
	778	It is a type of UML diagram that shows the order of execution of a system.	0	... are data type which instances form a list of named literal values.	0		734	Operations may have parameters and return values.	In a composition, a part does...
	0	It implements the sequence of actions that a system must execute.	0		0		0		
	0		0		0		0		

Questions

Table 1

Q9	Which of the following statements about compositions (composite aggregations) is true?	Q10	Which are the desired results of modeling use cases?	Q11	Which of the following statements about the given diagram clipping is true?	Q12	The use case diagram ...
830	When the composite element is deleted, the parts also die.	0		576	Z is a specialised Y and can be executed by sub actors of X.	603	specifies who performs which tasks with a system is developed
0	The multiplicity of a composite aggregation may be ≥1.	0	A model of the systems' application flows.	588	Z is a specialised Y and can be executed by sub actors of X.	548	specifies who performs which tasks with a system is developed
589	When the composite element is deleted, the parts also die.	506	A detailed textual description of every use case.	522	Z is a specialised Y and can be executed by sub actors of X.	0	... comprises actors, use cases and functional requirements.
544	When the composite element is deleted, the parts also die.	559	A detailed textual description of every use case.	0		746	specifies who performs which tasks with a system is developed
716	When the composite element is deleted, the parts also die.	0	A model of the systems' application flows.	0		549	specifies who performs which tasks with a system is developed
509	When the composite element is deleted, the parts also die.	0	A model of the systems' application flows.	532	Z is a specialised Y and can be executed by sub actors of X.	573	specifies who performs which tasks with a system is developed
0	The multiplicity of a composite aggregation may be ≥1.	0	A model of the systems' application flows.	523	Z is a specialised Y and can be executed by sub actors of X.	719	specifies who performs which tasks with a system is developed
740	When the composite element is deleted, the parts also die.	0	A model of the systems' application flows.	560	Z is a specialised Y and can be executed by sub actors of X.	0	... comprises actors, use cases and functional requirements.
742	When the composite element is deleted, the parts also die.	0	A model of the systems' application flows.	0	A or B can not execute Z.	758	specifies who performs which tasks with a system is developed
532	When the composite element is deleted, the parts also die.	0	Only one use case diagrams.	697	Z is a specialised Y and can be executed by sub actors of X.	0	... comprises actors, use cases and functional requirements.
581	When the composite element is deleted, the parts also die.	0	Only one use case diagrams.	0	A and B can not execute Y seperately.	0	... comprises actors, use cases and functional requirements.
0		0	Only one use case diagrams.	0		0	... comprises actors, use cases and functional requirements.
523	When the composite element is deleted, the parts also die.	0		511	Z is a specialised Y and can be executed by sub actors of X.	0	... comprises actors, use cases and functional requirements.
523	When the composite element is deleted, the parts also die.	0		0	A and B execute Y together.	634	specifies who performs which tasks with a system is developed
0		0		0		716	specifies who performs which tasks with a system is developed
0	In a composition, a part does not belong to only one composite at a time.	561	A detailed textual description of every use case.	0	A or B can not execute Z.	0	... comprises actors, use cases and functional requirements.
520	When the composite element is deleted, the parts also die.	0	Only one use case diagrams.	618	Z is a specialised Y and can be executed by sub actors of X.	0	... comprises actors, use cases and functional requirements.
0	In a composition, a part does not belong to only one composite at a time.	0	Only one use case diagrams.	0		0	... comprises actors, use cases and functional requirements.
0	In a composition, a part does not belong to only one composite at a time.	0	A model of the systems' application flows.	0		683	specifies who performs which tasks with a system is developed
0		0	A model of the systems' application flows.	0		0	
0	In a composition, a part does not belong to only one composite at a time.	599	A detailed textual description of every use case.	550	Z is a specialised Y and can be executed by sub actors of X.	0	
0	The multiplicity of a composite aggregation may be ≥1.	0	A model of the systems' application flows.	526	Z is a specialised Y and can be executed by sub actors of X.	0	... comprises actors, use cases and functional requirements.
0	In a composition, a part does not belong to only one composite at a time.	0	A model of the systems' application flows.	0	A and B execute Y together.	0	... specifies the functional decomposition of a system.
0		0		0		0	
0		0		0		0	

Which of the following statement... diagram clipping is true?

Table 2

	Q13	Which of the following statements about the given diagram clipping is true?	Q14	You are given the following sequence diagram. Which traces is possible?	Q15	You are given the following sequence diagram. Which traces is possible?	Q16	In a State Diagram, What does the syntax for labeling a transition look like?
specifies who performs which tasks with a system is developed	744	B can execute the same use cases as A.	726	a → b → c	762	a → b → c	770	event[guard]/effect
specifies who performs which tasks with a system is developed	731	B can execute the same use cases as A.	565	a → b → c	691	a → b → c	743	event[guard]/effect
... comprises actors, use cases and functional requirements.	0	A can execute the same use cases as B.	569	a → b → c	0	no one	875	event[guard]/effect
specifies who performs which tasks with a system is developed	605	B can execute the same use cases as A.	553	a → b → c	766	a → b → c	774	event[guard]/effect
specifies who performs which tasks with a system is developed	538	B can execute the same use cases as A.	543	a → b → c	745	a → b → c	524	event[guard]/effect
specifies who performs which tasks with a system is developed	652	B can execute the same use cases as A.	522	a → b → c	0	no one	529	event[guard]/effect
specifies who performs which tasks with a system is developed	623	B can execute the same use cases as A.	645	a → b → c	709	a → b → c	0	[guard]effect/event
... comprises actors, use cases and functional requirements.	703	B can execute the same use cases as A.	536	a → b → c	910	a → b → c	0	[guard]effect/event
specifies who performs which tasks with a system is developed	0	A can execute the same use cases as B.	567	a → b → c	0	c	0	[effect]event/guard
... comprises actors, use cases and functional requirements.	735	B can execute the same use cases as A.	632	a → b → c	0	a → b → c → a → b	730	event[guard]/effect
... comprises actors, use cases and functional requirements.	0	A inherits all of B's associations.	0		0	no one	0	[effect]event/guard
... comprises actors, use cases and functional requirements.	0	B inherits part of A's associations.	0	b → a → c	693	a → b → c	0	effect[guard]/event
... comprises actors, use cases and functional requirements.	661	B can execute the same use cases as A.	0		0	a → b → c → a → b	635	event[guard]/effect
specifies who performs which tasks with a system is developed	0	B inherits part of A's associations.	0	b → c → a	719	a → b → c	0	[guard]effect/event
specifies who performs which tasks with a system is developed	528	B can execute the same use cases as A.	0	b → a → c	0	c	717	event[guard]/effect
... comprises actors, use cases and functional requirements.	0	B inherits part of A's associations.	0		920	a → b → c	0	[guard]effect/event
... comprises actors, use cases and functional requirements.	0	A inherits all of B's associations.	502	a → b → c	0	c	0	[effect]event/guard
... comprises actors, use cases and functional requirements.	641	B can execute the same use cases as A.	0	b → a → c	851	a → b → c	0	[guard]effect/event
specifies who performs which tasks with a system is developed	0	B inherits part of A's associations.	675	a → b → c	0	a → b → c → a → b	0	[effect]event/guard
	773	B can execute the same use cases as A.	0		0		0	
... comprises actors, use cases and functional requirements.	507	B can execute the same use cases as A.	0		0	a → b → c → a → b	530	event[guard]/effect
	0	B inherits part of A's associations.	0	c → a → b	0	a → b → c → a → b	0	[effect]event/guard
	0	B inherits part of A's associations.	0	b → c → a	0	c	0	[effect]event/guard
	0		0		0		0	
	0		0		0		0	