Project Deadline-1 Rubric - Colors Switch Game

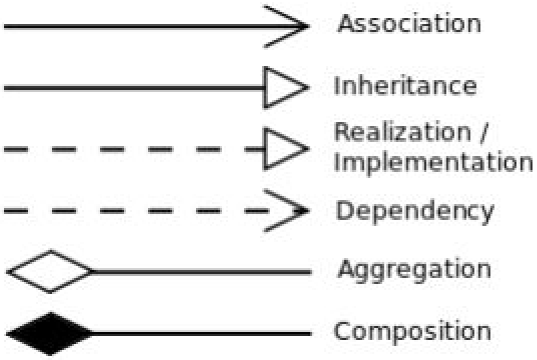
UML Use-case Diagram: (10 marks)

Category	Total marks	Remarks	Partial Marks
Use cases	2	User communicating with at least these four the system using use cases: a) start_game, b) restart_game, c)resume, and d) exit	0.5*4=2
Realationships	**4**	Meaningful demonstration of <<include>> relationship along with appropriate arrow	2
		Meaningful demonstration of <<extends>> relationship along with appropriate arrow	2
Multiplicity	**2**	Meaningful demonstration of multiplicity	2
Misc.	**2**	Detailed diagram, Student has gone the extra mile and made a very detailed use case diagram	1
		Actor with stick diagram and system boundary shown correctly	1
TOTAL	**10**		

1) Generalization: Solid arrow with hollow arrowhead.
2) Association: Solid arrow with regular arrowhead.
3) Include and Extends: Dashed arrow with regular arrowhead. The relationship name should be mentioned on these arrows. (e.g, <<include>>, <<extends>>)

UML Class Diagram (10 marks)

Classes	**2.5**	Classes: All these basic classes: a) Game, b) Player, c) Obstacle, d) Star e) ColorSwitch	0.5*5=2.5
Minimum Attributes (Variables and Functions)	**3.5**	Min. Attributes: Star class -- a) position :Int/Float	0.5
		Player class- a) current_color b) position:Int/Float	0.5
		ColorSwitch class -- a) position :Int/Float	0.5
		Obstacle: a) type b)position:Int/Float c)color	0.5
		Game: a) player: Player, b) Obstacles: List<Obstacle>, c) current_score: Integer d) StarPos: List<Star> e)ColorSwitch : List<ColorSwitch>	1
		App: functions - loadGame(), startGame(), saveGame(), exitGame()	0.5
Class Relationships	**1**	Player, Obstacle, ColorSwitch, Star should have composition relationship with Game	1
Multiplicity	**1**	Multiplicity shown appropriately, e.g., Player (1), Star (0..*), ColorSwitch (0..*), Obstacle(0...*).	1
Serializable Interface	**2**	Every Class except App Class should implement Serializable interface	2
TOTAL	**10**		



Arrow Definition in Class Diagrams