

# Minimumeisen

Het FO is verzorgd (zie ook de ICA-controlekaart) en bevat voldoende informatie om het TO en de realisatie op te baseren.	Voldoet
Het TO is verzorgd (zie ook de ICA-controlekaart), bevat voldoende informatie om de realisatie op te baseren en is in lijn met het FO.	Voldoet
Alle zelfgebouwde publieke methoden en zelf toegevoegde publieke attributen zijn voorzien van documentatie (bij voorkeur met Javadoc, zie bijv. <a href="https://nl.wikipedia.org/wiki/Javadoc">https://nl.wikipedia.org/wiki/Javadoc</a> )	Voldoet
Er zijn minimaal acht eigen klassen gerealiseerd die voldoende verschillend zijn en die op z'n minst enige functionaliteit bevatten (klassen met vrijwel identiek gedrag dat ook met andere attribuutwaarden te realiseren zou zijn, tellen niet mee, en ook interfaces of volledig abstracte klassen tellen niet mee).	Voldoet <ul style="list-style-type: none"> <li>- DivingForTreasure</li> <li>- Aid</li> <li>- OxygenTank</li> <li>- Enemy</li> <li>- Bomb, Shark</li> <li>- Treasure</li> <li>- Coin, Diamond</li> <li>- InteractableObject</li> <li>- Spawner</li> <li>- Player</li> <li>- TreasureBag</li> <li>- Screen</li> <li>- PlayScreen, EndScreen</li> <li>- Clock</li> </ul>
Er wordt minimaal één van de interfaces uit de engine succesvol toegepast.	De klasse InteractableObject doet succesvol de interface ICollidableWithGameObjects toepassen
Er wordt overerving toegepast binnen de zelfgebouwde klassen (dus niet alleen overerven van objecten uit de engine).	<ul style="list-style-type: none"> <li>- De klasse Coin en Diamond overerven van Treasure</li> <li>- De klasse Bomb en Shark overerven van Enemy</li> <li>- De klasse OxygenTank overerft van Aid</li> <li>- De klasse PlayScreen en EndScreen overerven van Screen</li> </ul>
Er wordt minimaal eenmaal polymorfie toegepast binnen zelfgebouwde klassen (dus: er is de mogelijkheid van polymorfie ingebouwd, en deze wordt daadwerkelijk gebruikt).	Polymorfie wordt toegepast in de klasse Spawner. InteractableObjects hebben de methode SpawnObject() en deleteObject() Deze objecten worden bewaard in een ArrayList en met een For lus worden deze methodes op elk object uitgevoerd.

<p>Klassen en methoden hebben duidelijke verantwoordelijkheden:</p> <ul style="list-style-type: none"><li>• Naam komt overeen met de taak</li><li>• Namen van attributen dekken de lading</li><li>• Zo weinig mogelijk dubbele code</li><li>• Geen static variabelen, tenzij daar een goede reden voor is</li></ul>	<p>Voldoet</p> <ul style="list-style-type: none"><li>- De statische variabelen zijn de path naar media en sound (Zoals uitgelegd is in de tutorial van de game Engine)</li></ul>
---	--