

Environment	(max)	Points	
Integrated development environment (IDE) can be used in a professional manner (short-cuts, importing APIs, etc.)	1 P	1	
Uses a CASE-Tool for UML (stand-alone, plugIn, etc.)	1 P	2	
Uses a version control tool (GitHub, GitLab)	1 P	3	
<b>Implementation</b>	<b>2 P</b>		
<b>Your own project has been implemented...</b> with use cases (written) with a sequence diagram with class diagrams (a rough design at beginning, detailed at end of project)	<b>+ 2 P</b> <b>+ 2 P</b> <b>+ 1 P</b> <b>+ 2 P</b>	12	
<b>Code is runnable</b> should run, but doesn't it runs for certain use cases, with others it crashes runs fully	1 P <b>-1P</b> <b>+ 1 P</b>	14	
<b>Exception-Handling is being used</b> own Exceptions implemented	2 P <b>+ 2 P</b>	18	
<b>Input and output</b> via console using files / data source very user friendly	1 P <b>+ 2 P</b> <b>+ 2 P</b>	23	
<b>JavaDoc</b> is used for min. 80% of classes and min. 80% of methods	2 P	25	
<b>JUnit</b> is applied for min. 1 class with min. 2 test methods for a higher test coverage	1 P <b>+ 2 P</b>	28	
<b>Object-orientated concepts in code</b>	1 P	29	
OO-concepts are shown with own written code			
Encapsulation / Information Hiding is applied	1 P		
Delegation is being used	1 P		
Aggregation / Composition is implemented	1 P		
Classes are well structured (ie. what they need to do with clearly defined methods)	<b>+ 1 P</b>	33	
Clean code (ie. main-method has no logic), distinction between input and logic (and data)	<b>+ 2 P</b>	35	
Project is briefly presented in class / or to teacher	<b>+ 2 P</b>	37	
<b>Documentation</b> covering design, planning and test cases (use cases) is in a repository (progress is documented), format in markdown	4 P	41	

Maximum points: 41; for grade 6 you need min.39pts.; for grade 5: min.31pts; for grade 4: min.24pts; for grade 3 min.16pts. If you have below 23pts the third column in the c-grid remains empty.