Rubric Prog 6 v1.4 - Jan 2026

Proficiency level:	Advanced	Proficient	Emerging	Novice
Competencies				
Analyze: Problem	*Case was very well connected to a case from the Wilhelmina Children's Hospital (WKZ). Relevant user stories were defined. WKZ/Health experts are really enthusiastic about the solution(s).*	*Case was decently well connected to a case from the Wilhelmina Children's Hospital (WKZ). Case could have been more apt to the wishes of WKZ.*	*Case could have been more apt to the wishes of WKZ.*	*Case was either not given or not connected to a realistic problem, also no real-world examples were given.*
Requirements	*SMART Requirements. Both FURPS+ and MoSCoW were used perfectly.*	*Somewhat SMART Requirements. Both FURPS+ and MoSCoW were used to some degree.*	*Requirement could be much more SMART . FURPS+/MoSCoW were used but not very well.*	*Nothing SMART. No FURPS+ and/or MoSCoW.*
Design	*There was a SOLID design that helped to turn the problem into a proof-of-concept.*	*Overall design was good, some details could have been better, it was a solid starting point for the proof-of-concept.*	*The design could have been better but there was a basic design that incorporates key elements.*	*There was no design at all or the design lacked key elements needed for successful implementation.*
Functional	*The use cases themselves and their explanation are clear cut.*	*There are still a few issues with the use cases and their explanation.*	*There are still a lot of issues with the use cases and their explanation.*	*No use cases and/or no elaboration.*
Technical	*At least three types (e.g. Class, Sequence, States) of technical designs were used and they were flawless.*	*At least two types technical designs were used.*	*There was at least one technical design and/ or it/they still had problems.*	*No adequate technical designs used.*
Manage: Code Management	*You used Git at a professional level including submits, pull-requests, blames and code reviews throughout the assignment.*	*You showed good competence of Git. Some submits, a few pulls-request.*	*You used Git but only at a very basic level.*	*You did not use Git or used it in a less than professional way.*
Control: Testing	*You tested at professional level.*	*You did decent testing.*	*You did some tests but they were lacking.*	*You did no or virtually no testing.*
Realization: Proof-of-Concept	*Problem was turned into a working proof-of-concept (PoC) that worked well and helped to prove the case.*	*Working example was created, it still had some issues but the overall proof-of-concept was successful.*	*There was a basic solution for the problem, but it did not work well.*	*There was no working proof-of-concept at all or it was not relevant for the problem.*
Realization: Application of OO principles	*Classes, encapsulation, inheritance, abstract classes and/ or the principle of interface were used to perfection.*	*More advanced OO principles were applied.*	*The basic OO principles were applied.*	*No OO principles were applied.*
Good programming: SOLID / DRY / KISS	*Both details and overall process were managed very well.*	*Details and overall process were managed well, but could still improve.*	*Process like a bumpy road, but at least good efforts were made to reach the end.*	*Utter chaos.*
Professionalize: Reflection on your work	*A solid advice was given based on analysis (use-case), PoC. This was translated into a convincing presentation / report / video.*	*Given advice was sufficient, but there were still holes in argumentation.*	*The presented advice was not really based on use-case and/or PoC and/or not convincing at all.*	*There was no advice.*
Professionalize: personal level	*Perfect performance. Great communication.*	*Good performance, reasonable communication.*	*Okay performance, communication should have been better.*	*Lacking performance, communication is poor.*
	Note: make sure all files and documents are in a readable format without the need for additional installers (apart from any basic IDE and/or image viewer): Diagrams should be PNG/PDF, Source in .c(pp)/.h(pp)/.py/other text formats. In case you want to demo something hand it in as a video. Documentation in PDF format.			
	Teacher name:	Date:		
	Johan Korten			
	Name:	ID:		Final mark:
Student:				

Proficiency level:	Advanced	Proficient	Emerging	Novice