Competency Type	Competency	Description
Technical	OOP Principles	- Understanding and ability to explain basic principles of OOP.
Technical	Design pattern/SOLID	- Provide explanation what patterns are, when we need to use them, give examples; - Formulate and explain SOLID design principles.
Technical	Source Control Systems	- GIT, commands.
Technical	Deployment (CD/CI)	- Explain the main stages of CI/CD process.
Technical	Architecture Types	
Technical	Software Development Process	- Name and explain the development phases; - Provide example how to work one of methodology of software development.
Technical	Testing Types Unit Testing	- Formulate goals and principles of UT; - How to estimate UT coverage; - Explain using equivalence classes for improving coverage; - Types of auto testing; - What should be tested and in what way; - Tools.
Technical	JavaScript	Fundamentals - data types - loops - functions - objects - classes - promise, async/await - modules ES6 +
Technical	TypeScript	- What is Typescript; - Advantages; - Core principles, Types, Interfaces, syntax sugar perks.
Technical	Store/State Management	- Understanding of core principles - state, store, reducer.
Technical	Network / HTTP / REST	- Formulate the rules and principles for the implementation of the REST API - How HTTP works, HTTP methods
Technical	HTML5, CSS3, HTML5 API	HTML engines, CSS Preprocessors, modern features, DOM, Browser features
Technical	Refactoring	- ability to analyze the code problems, code smells - approaches
Technical	Frameworks	- Angular (depends on the project) - React (depends on the project) - Vue.js (depends on the project)
Technical	Build tools	- gulp / grant / Webpack
Technical	Debugging	- dev tools - debug approaches
Technical	Angular	- Core modules, DI, Change Detection, Data flow
Technical	React	-
Soft Skill	English	
Soft Skill	Communication	
Soft Skill	Self-organization/Self-management	
Soft Skill	Self-education and professional growing	
Soft Skill	Managment(Agile)	