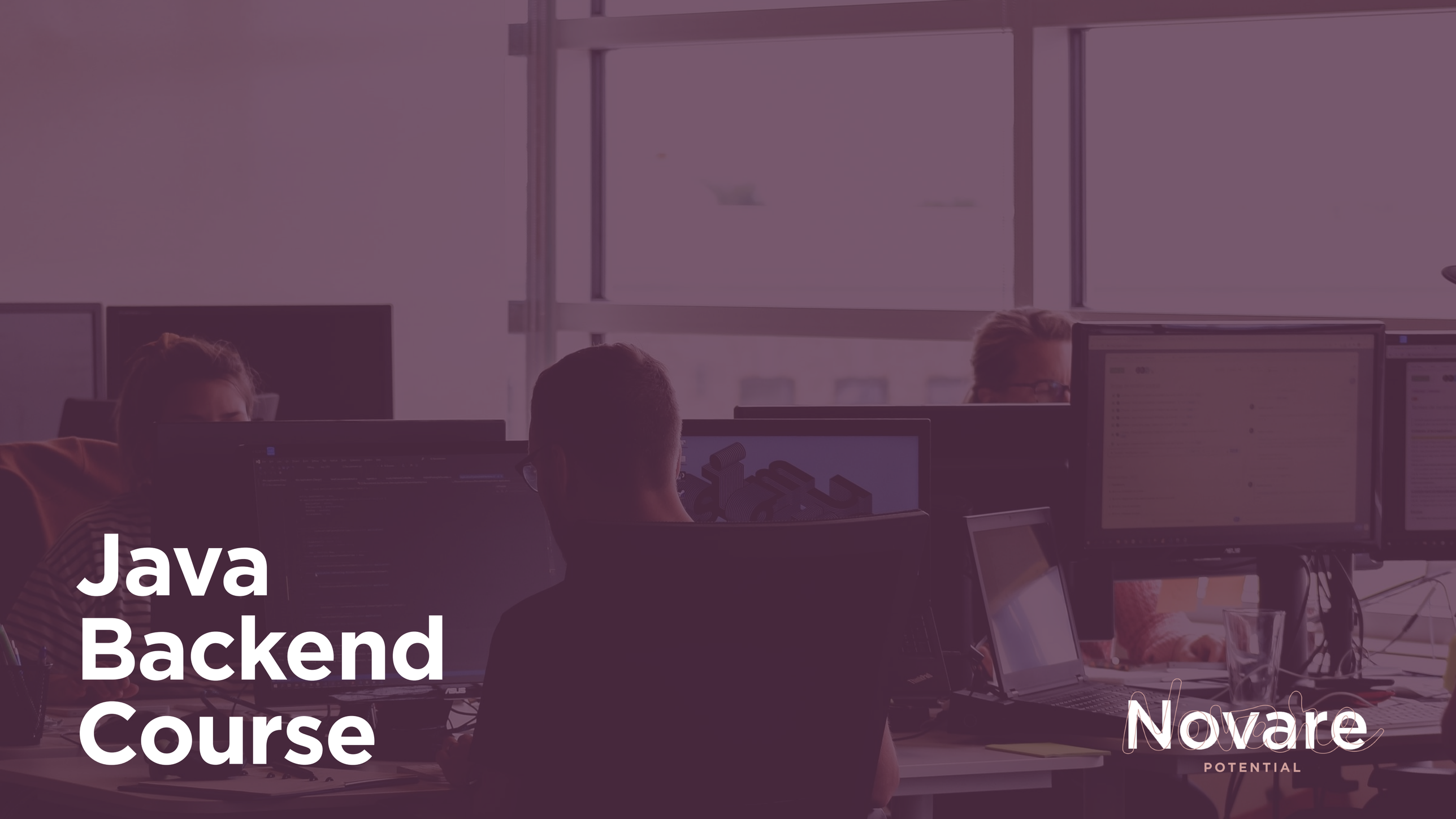

Java Backend Course

Novare
POTENTIAL

Methodology

Learning style

Project based

Students will code their projects in an agile way using incremental iterations instead of a waterfall approach using a constantly evolving Prioritized Product backlog from SCRUM.

Close monitoring

Our grading system considers the working code and the correct understanding of the project requirements to produce high code quality with low coupling, avoiding side effects.

Mentor support

Students will have access to qualified staff that co-create the projects before the course starts to answer any students' doubts.

Course structure

What will be teached

Project
Management

Java
basics

Java
advanced

Full-stack
development

Backend
development

Large scale
projects

Course structure

What will be teached

Project Management

This module starts the whole course by giving the correct mentality to our candidates that projects succeed or fail before the first line of code is ever written.

Java basics

Teaches how to code using Java as the primary language of the course.

Java advanced

Teaches how to utilize Java to interact with the rest of the computer using the built-in functions of the Java SDK.

Full-stack development

In this part of the course, the students built graphical interfaces for the web by utilizing a backend created by our staff.

Backend development

In this stage, the students create the backend themselves.

Large scale projects

At this part of the course, the students combine their previous modules' knowledge to build a full-stack app while applying continuous integration.

Course structure

Curricula details

1. Course introduction	2. Java basics	3. Java advanced	4. Full-stack development	5. Backend development	6. Large scale projects
Requirement gathering	Good coding practices (3R's in software architecture)	Testing	Familiarize with the frontend tech stack	Database management using MySQL	Dev ops
Prioritized Product Backlog (SCRUM)	Java SDK (how to use FileStream to manipulate files, etc)	Design patterns (factory, MVC, etc)	REST API's with Spring boot to connect a existing backend to a frontend	Docker container	Deploy to Amazon Web Services (Cloud)
Bring ideas to life (UML diagrams, whiteboard discussions)					