EAIE

Esports as Accelerator for Social Inclusive Education

LESSON

THE DEBUGGING MARATHON

AUTHOR

SERGIO ELVIRA PEREZ

CODEARTS SOLUTIONS

THE DEBUGGING MARATHON

The Debugging Marathon is a timed troubleshooting challenge in which teams repair faults across a small full-stack training project. Learners must gather evidence, prioritise issues and avoid introducing regressions while the codebase changes around them.

Subject and level

Intermediate and advanced VET developers who already work with PHP, JavaScript and development tools and need a systematic approach to diagnosis and debugging.

Connection to curriculum

The activity reinforces debugging, log interpretation, browser and IDE tools, version control, configuration analysis and regression checking. It also develops prioritisation and technical communication under realistic project pressure.

Didactic and pedagogic considerations

The exercise rewards method rather than random trial and error. Teams must state a hypothesis before major changes, record evidence and verify fixes. Controlled interruptions simulate real project pressure while remaining safe and recoverable.

Time estimate

2 hours

THE DEBUGGING MARATHON

Group size

16-24 students

Overall introduction to exercise

Teams receive a training application containing front-end, backend and configuration defects. They choose the order of investigation and earn points for verified fixes, clear diagnosis notes and avoiding regressions.

Because new information is introduced during the session, learners experience debugging as a dynamic professional activity. The marathon makes persistence, calm communication and evidence-based reasoning visible alongside technical knowledge.

THE DEBUGGING MARATHON

Exercise as presented to the students

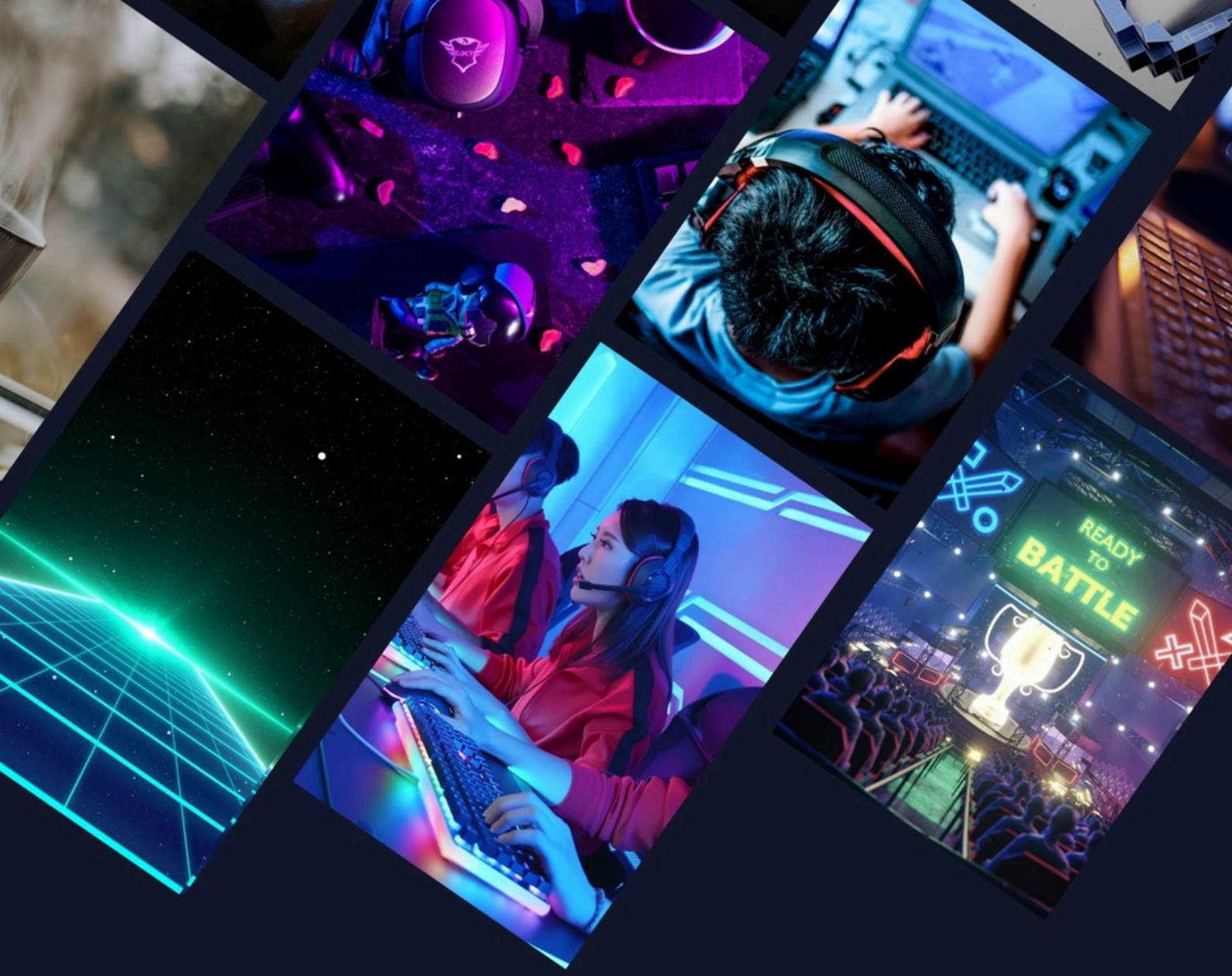
The classroom was arranged like a short hackathon and teams of four cloned a repository containing three fault categories: a PHP backend bug, a JavaScript interface problem and a Docker configuration issue. A projected countdown established the scenario: the application had to be stabilised before a fictional client demonstration.

Before editing code, each team opened an incident sheet and recorded symptoms, likely causes and the evidence they intended to collect. Students used logs, browser DevTools, breakpoints, console output and container status commands to narrow each problem. A verified diagnosis earned as much recognition as a fast patch.

After 30 minutes, the teacher pushed a small update that changed one dependency and forced teams to pull, re-test and decide whether previous assumptions still held. This controlled interruption recreated the uncertainty of a live project while version control ensured that every change could be recovered safely.

The final ten-minute challenge introduced a high-impact configuration error. Teams had to decide whether to stop current work and investigate it or complete a nearly finished fix first. Every repaired issue required a quick regression check before it was accepted on the scoreboard.

When the timer ended, groups presented one successful debugging path and one false lead that had cost time. The class compared strategies and identified the value of reproducing faults, reading evidence and making small reversible changes. The activity assessed technical debugging while also revealing resilience, prioritisation and team communication.



EASIE
Esports as Accelerator for Social Inclusive Education

E-SPORT AS ACCELERATOR FOR SOCIAL INCLUSIVE EDUCATION

WWW.EASIE.GG

Co-funded by
the European Union

FUNDED BY THE EUROPEAN UNION. VIEWS AND OPINIONS EXPRESSED ARE HOWEVER THOSE OF THE AUTHOR(S) ONLY AND DO NOT NECESSARILY REFLECT THOSE OF THE EUROPEAN UNION OR THE EUROPEAN EDUCATION AND CULTURE EXECUTIVE AGENCY (EACEA). NEITHER THE EUROPEAN UNION NOR EACEA CAN BE HELD RESPONSIBLE FOR THEM.