Rocq Game: A Gamified Web-Based Environment for Interactive Learning with a Proof Assistant

16th International Workshop on Theorem Proving Components for Educational Software

Khaled Jamal Suliman, Ján Perháč, Samuel Novotný

Department of Computers and Informatics
Faculty of Electrical Engineering and Informatics
Technical University of Košice
Slovak Republic

July 19, 2026
Lisbon, Portugal

Motivation: proof tools for theoretical courses

Our teaching context

Type theory **Logic** **Semantics** **Formal methods**

These courses benefit from executable, machine-checked proofs.

But a full-scale proof assistant is not always appropriate

Introducing its language, tactics, libraries, and development environment can consume time intended for the core concepts of the course.

Simple "calculator" tools

Easy to use, but limited to a narrow formalism or task.

$\longleftrightarrow$

Full-scale proof assistants

Expressive and authentic, but demanding for beginners.

Our long-term research direction

We develop systems between these two categories:
proof assistants for education. One of our newest tools is Rocq Game.

What should an educational proof assistant provide?

Lower the entry barrier

- start in one click, with no installation
- explain theory next to the exercise
- reveal hints when the learner is stuck

Preserve real proof practice

- execute genuine Rocq proofs
- show proof states and errors immediately
- unlock tactics through progressive tasks

Our goal

Bridge the gap with a **setup-free, guided, and gamified** environment— without hiding Rocq's interactive proof process.

The landscape of proof tools for education

Calculator-like tools

- Natural Deduction Tool
- Proof Tree Builder

Focused and accessible, but tied to a particular proof formalism.

Full-scale assistants

- Rocq, Lean
- Agda, Isabelle

Highly expressive, but their language and workflow require substantial onboarding.

Educational systems

- ProofBuddy
- OnlineProver
- Edukera

Add pedagogical structure while retaining meaningful proof construction.

Position of Rocq Game

Rocq Game belongs to the third category: an **educational layer over a full-scale proof assistant**.

Closest related systems

Lean Natural Number Game

Contributes: worlds and levels, progressive unlocking, and a game-like learning path.

Difference: proof checking is performed on a server.

`https://adam.math.hhu.de/#/g/leanprover-community/nng4`

Software Foundations + jsCoq

Contributes: genuine Coq in a browser, client-side execution, and an interactive proof state.

Difference: a textbook model with solutions visible.

`https://jscoq.github.io/ext/sf/`

Codewars for Coq

Contributes: exercise-based practice, challenges, rewards, and community-created content.

Difference: whole-proof submission without incremental subgoal feedback.

`https://www.codewars.com/kata/search/coq`

How Rocq Game combines these ideas

Engagement

Worlds, levels, XP, and progressive tactic unlocking

from learning games

+

Authenticity

Real Rocq execution and immediate proof-state feedback

from jsCoq

+

Teaching support

Theory, hints, structured tasks, and reusable content

for classroom use

Result

Setup-free client-side verification, guided progression, and instructor-authored content in one environment.

Rocq Game: start proving in the browser

rocq-game.kpi.fei.tuke.sk

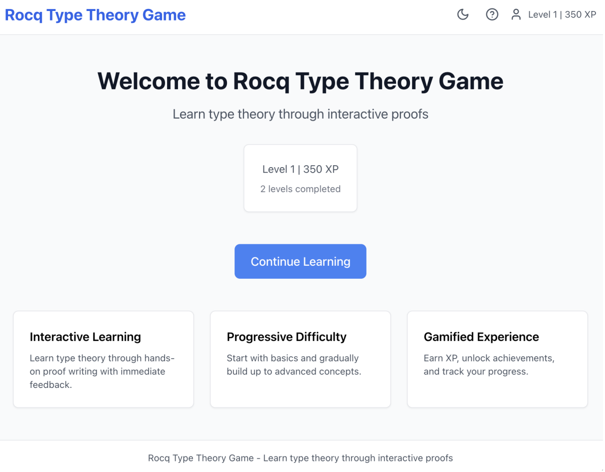
Runs directly in the browser
No installation or registration

What does Rocq Game provide?

- structured learning paths organized into **worlds and levels**,
- genuine **Rocq proofs** executed locally through jsCoq,
- immediate feedback through the live **proof state**,
- integrated **theory, hints, tactics, and available theorems**,
- gamified progression with **XP and unlockable content**,
- exportable progress data for learning and assessment.

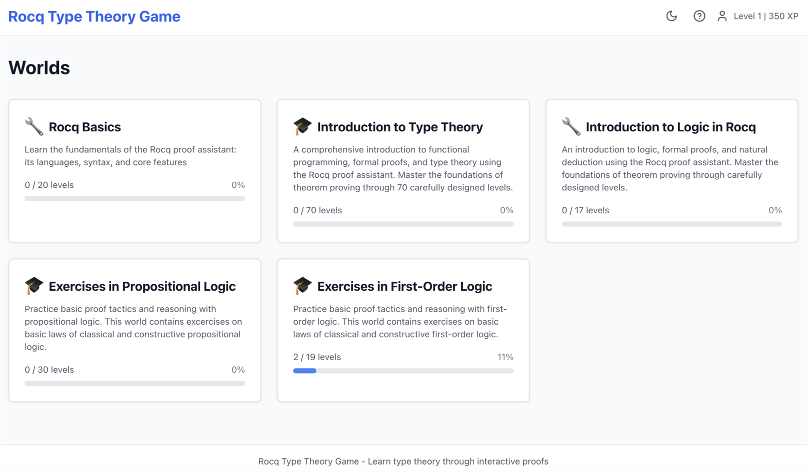
One environment connects explanation, guided practice, and authentic Rocq proofs.

Welcome page



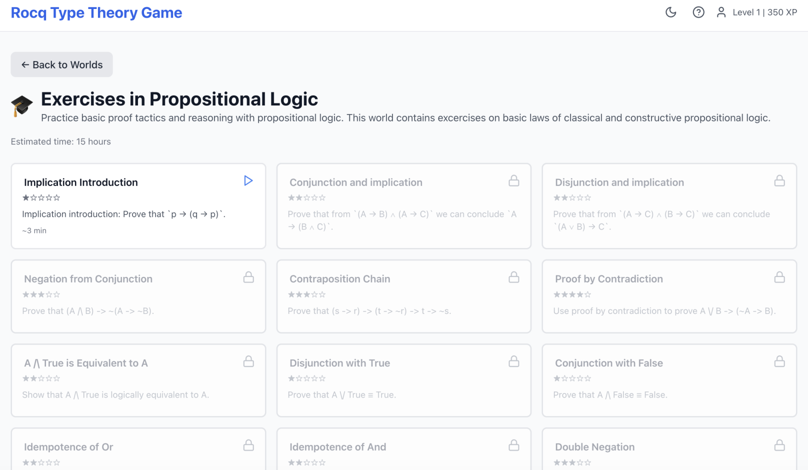
Resume existing progress or start a new learning path.

Worlds overview



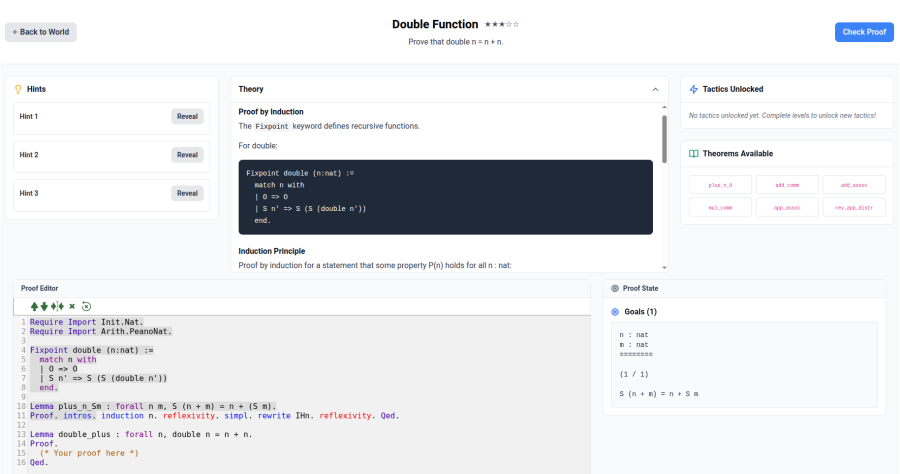
Choose a topic and see progress across the available worlds.

World overview



Open an available level; subsequent tasks unlock progressively.

A level brings everything into one view



The learning loop stays visible

Theory and hints ⟶ Proof editor ⟶ Live proof state ⟶ Check and progress

Pedagogical design: guided progression

Choose a topic world → Open the next available level → Read theory and reveal hints → Construct and check a proof → Earn XP and unlock content

Structure

Worlds divide the syllabus into topics; levels introduce concepts incrementally.

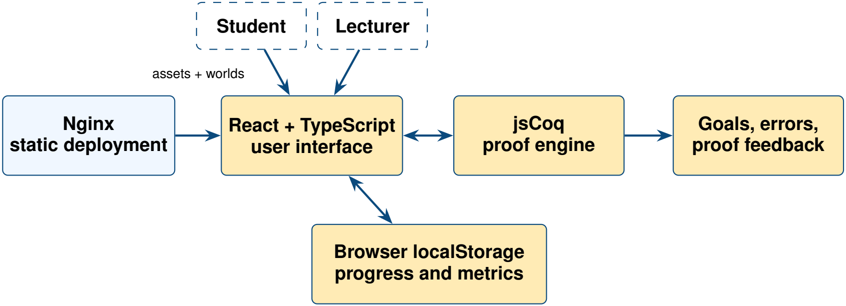
Scaffolding

Theory, hints, available tactics, and earlier theorems support the learner.

Authenticity

Students still write genuine Rocq code and inspect real proof states.

Architecture: proof checking stays in the browser



Setup-free

The browser receives a static single-page application.

Scalable

Core proof execution creates no server-side session.

Process-aware

Attempts, time, completions, and hint use can be exported.

Content authoring: from Rocq files to learning worlds

Annotated .v files Rocq + DSL → **DSL parser and validation** → **Generated JSON world definitions** → **Loaded by the Rocq Game SPA**

Content definitions

- **World definition**
- **Level definition**

Both definitions are compiled into JSON and loaded by the Rocq Game application.

Authoring workflow

- author content
- compile definitions
- generate JSON
- load into Rocq Game

Live demonstration

The authoring process and both definition types will be shown during the live demo.

Why separate content from the application?

- instructors work in a familiar Rocq environment,
- worlds can be added or enabled independently,
- learning content changes without changing the UI code.

Live demonstration

rocq-game.kpi.fei.tuke.sk

Open in a browser—no installation or account required.

Student workflow

1 Choose a world → **2** Open a level → **3** Construct a proof → **4** Check progress

Authoring workflow: add a new world

1 Add a `.v` file → **2** Compile → **3** Update config → **4** Rebuild → **5** Deploy

Two classroom deployments

Type Theory (TT)

- autumn semester 2025/2026
- 55 enrolled; **43 responses** (78.2%)
- Rocq used throughout the course
- Rocq Game introduced during the examination period
- comparison: Rocq vs. Rocq Game
- Rocq Game SUS: **81.1**

Logic for Informatics (LPI)

- spring semester 2025/2026
- 37 enrolled; **36 responses** (97.3%)
- tools introduced in the third lecture
- comparison: paper, OnlineProver, and Rocq Game
- OnlineProver SUS: **75.5**
- Rocq Game SUS: **54.8**

Two substantially different teaching contexts

The deployments should be interpreted separately rather than as direct replications.

First experiment: Setup and Context

Type Theory Course: `https://kurzy.kpi.fei.tuke.sk/tt/`

- 1st or 2nd year of Master's studies
- only 3 mathematics-oriented courses in Bachelor studies
- most students had no prior exposure to formal proofs or functional programming

About the course

- elective course
- based mainly on *Types and Programming Languages* and *Software Foundations*
- proof assistant Rocq
  - simple functional toy language, λ-calculus
  - simply typed λ-calculus with extensions
  - let polymorphism and System F
  - introduction to System $\lambda\underline{\omega}$ and System λP
  - Curry–Howard Correspondence

First experiment: procedure

Deployment during the 2025/2026 examination period

- 55 students were enrolled in the course.
- The first functional Rocq Game prototype was completed after teaching ended.
- Students had used Rocq during the course, but encountered Rocq Game only during the evaluation.

Group A

Set A in Rocq $\longrightarrow$ Set B in Rocq Game

Group B

Set B in Rocq $\longrightarrow$ Set A in Rocq Game

Post-task evaluation

43 anonymous questionnaires (**78.2% response rate**).

Evaluation measures and limitations

What we measured

- SUS for **Rocq Game**: 10 Likert-scale items converted to a 0–100 score
- self-reported theorem completion
- perceived time compared with Rocq
- open-ended feedback on both tools

Interpret with care

- fixed tool order: Rocq first, Rocq Game second
- different theorem sets in the two tools
- no objective timing
- no pre-test/post-test of learning
- unequal prior exposure to the tools

The evaluation measures perceived usability and experience, not learning gains.

Rocq Game received a high SUS score

81.1

mean SUS / 100

Grade A

using the interpretation reported in the paper

Score summary

Mean: 81.1

10% trimmed mean: 81.8

Interpretation

- above the commonly cited 68/100 reference,
- indicates **high perceived usability** in this deployment,
- does not by itself establish learning effectiveness or classroom readiness.

$n = 43$ **questionnaire responses**

Self-reported task outcomes

Theorem completion ($n = 42$)

- **21 (50.0%)** reported completing all five
- **4 (9.5%)** reported completing four
- **25 (59.5%)** reported completing at least four

Time compared with Rocq ($n = 43$)

- **22 (51.2%)** reported less time
- **18 (41.9%)** reported similar time
- **3 (7.0%)** did not complete the exercises

Interpretation

The time comparison is subjective. Because tool order was fixed and theorem sets differed, it does not demonstrate that Rocq Game objectively reduced solution time.

What students valued—and what needs improvement

Recurring benefits

- hints and integrated theory
- no installation and browser access
- task-oriented structure
- familiar Rocq interaction
- gamified presentation

Advantages reported for Rocq

- more workspace and autocomplete
- clearer goal separation and detailed errors
- standalone IDE preferred by some students

Requested improvements

- clearer errors and proof-state visualization
- complete, contextual, and adaptive hints
- fullscreen and resizable panels
- templates and tactics overview
- reliable proof checking and progress import

Takeaway

Students valued accessibility and guidance, while reliability and clearer feedback remained important design priorities.

Second experiment: study context

Logic for Informatics (LPI): `https://kurzy.kpi.fei.tuke.sk/lpi/`

- Propositional and first-order logic: language, semantics, and deduction calculi
- Resolution methods and the Boolean satisfiability problem (SAT)
- Horn clauses, SLD resolution, and logic programming in Prolog
- Intuitionistic, linear, and modal logics; Curry–Howard correspondence

Deployment and questionnaire sample

- Spring semester 2025/2026: **37 students enrolled**
- OnlineProver and Rocq Game were presented in the **third lecture**, after deduction calculi had been introduced
- **36 students completed the questionnaire** (97.3% response rate)
- 72.2% had no prior experience with proof assistants

Results presented here are based on self-reported questionnaire data.

Three approaches to constructing proofs

Pen-and-paper

- familiar notation
- unrestricted workflow
- no automated feedback

OnlineProver

- structured proof rules
- immediate validation
- structured feedback
- proof-tree visualization

Rocq Game

- tactic-based proofs
- authentic Rocq workflow
- integrated hints and theory

Comparison objective

Examine how usability and students' perceived learning differ across the three approaches.

OnlineProver: a tool for learning formal proofs

What is OnlineProver?

- A web-based educational environment for constructing and checking proofs in propositional and first-order logic
- Supports natural deduction.
- Represents a proof as an explicit tree of formulas and applied rules

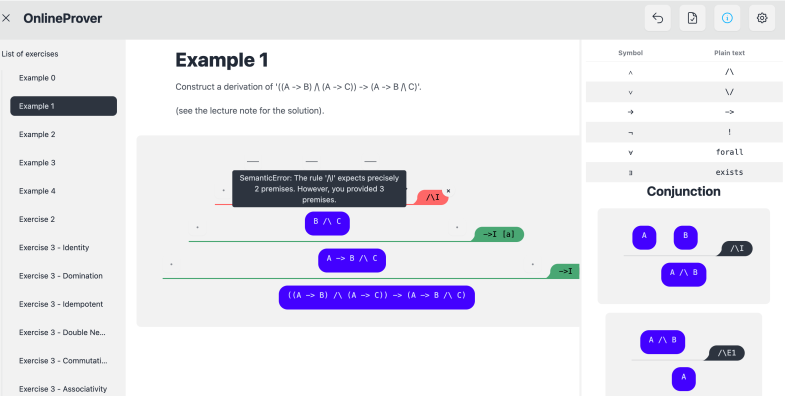
Student interaction

- choose and apply a rule
- enter the resulting formulas
- undo and revise proof steps

Educational support

- immediate step validation
- visible hypotheses and rules
- feedback without learning a tactic language

OnlineProver: a proof in progress



A partially constructed proof with immediate validation of individual steps.

Study design

Participants and methods

- 36 LPI students
- Pen-and-paper
- OnlineProver
- Rocq Game

Perceived learning

- understanding proof steps
- ability to explain a proof
- mental demand
- self-reported transfer

Usability

- SUS for both digital tools
- ease, speed, frustration
- direct method preference

Qualitative evidence

- benefits and barriers
- common mistakes
- suggested improvements

Interpret with care

These are **self-reported perceptions**, not objective learning gains. The study did not use a pre-test/post-test or controlled task assignment, and different levels of prior exposure may have influenced the comparison.

OnlineProver received a higher SUS score than Rocq Game

OnlineProver

75.5

mean SUS / 100

10% trimmed mean: 76.3

Rocq Game

54.8

mean SUS / 100

10% trimmed mean: 54.5

$n = 36$ **for both systems; trimmed** $n = 30$

The trimmed mean excludes the three lowest and three highest respondent-level SUS scores.

Direct comparison of proof methods

OnlineProver

- easiest: **44.4%**
- fastest: **47.2%**
- preferred for an exam: 41.7%

Pen-and-paper

- learned the most: **47.2%**
- preferred for an exam: **50.0%**
- fastest: 27.8%

Rocq Game

- most difficult: **52.8%**
- easiest: 16.7%
- learned the most: 8.3%

Main contrast

OnlineProver lowered the operational barrier; pen-and-paper remained important for perceived learning and independent performance.

Perceived learning and self-reported transfer

Mean rating on a 1–5 scale ($n = 36$)

Statement	Paper	OnlineProver	Rocq Game
Understand proof construction	**4.33**	4.06	2.39
Solve a similar task independently	**4.06**	3.75	2.19
Method helped me understand logic	**3.86**	**3.86**	2.61

Interpretation

OnlineProver combined high perceived understanding with immediate support, while pen-and-paper received the strongest self-reported transfer ratings.

Recurring benefits and barriers

Recurring benefits

- **OnlineProver:** immediate feedback, undo, visible rules, and clear proof structure
- **Pen-and-paper:** independent reasoning, flexibility, and preparation for manual proofs
- **Rocq Game:** hints, browser access, and visible proof state

Recurring barriers

- OnlineProver did not consistently check fresh-variable conditions
- Rocq syntax and tactics diverted attention from the logic
- Several students reported incorrect acceptance of empty or invalid Rocq Game submissions

Design priority

Validation must be trustworthy before hints and gamification can support learning.

Lessons learned and limitations

Lessons for educational proof tools

- Immediate feedback was the most frequently mentioned OnlineProver benefit.
- Syntax and onboarding were the dominant barriers for Rocq Game.
- A useful learning path may start with guided practice and then remove support to encourage independent proof construction.

Limitations

- voluntary sample from one course ($n = 36$)
- self-reported outcomes and unequal prior exposure
- reported validation defects may have affected students' ratings

Main takeaway

Combine immediate feedback with authentic proof construction—but keep the syntax barrier low and the validation reliable.

Cross-experiment interpretation

TT context

- students already had experience with Rocq
- Rocq Game extended a familiar workflow with hints and theory
- evaluation followed a semester of working with Rocq
- students reported high perceived usability

LPI context

- 72.2% had no prior proof-assistant experience
- OnlineProver provided a more structured alternative
- Rocq syntax and validation issues became important barriers
- OnlineProver was perceived as easier and more usable

Main interpretation

Rocq Game was well received when it extended an already familiar Rocq workflow. For novices, syntax, onboarding, and trustworthy validation became critical.

Both deployments were conducted within project KEGA 052TUKE-4/2025.

Future work

Rocq Game

- Finish features and fix bugs.
- Improve the interface and validation.
- Improve reliability of proof checking and error reporting.
- Reduce the syntax barrier with guided tactic onboarding.
- Extend content with new worlds.

Related projects

- Integrate lessons learned from our educational theorem-proving tools.
- Continue development of OnlineProver 2.0 (new interface, richer feedback, modular design, DSLs, etc.).
- We have/had two more talks at FLOC workshops:
  - *LogicProof: An Interactive Web-Based Educational Theorem Prover for Natural Deduction and Sequent Calculus across Classical and Constructive Logics* (TEAL),
  - *ProverSLD: An Interactive Visualizer for Teaching SLD Resolution in Prolog and First-Order Logic* (PEG).

Future work: cooperation

Pedagogy and evaluation

- Design stronger student-centered studies with clearer learning metrics.
- Collaborate with experts in pedagogy and experimental methodology.

International collaboration

- Use COIL (Collaborative Online International Learning) to connect courses.
- Pilot joint activities in type theory and logic for informatics.
- Build partnerships via the Ulysseus alliance and related grant calls.

Acknowledgments

Project **KEGA 052TUKE-4/2025**:
Modern approaches in educating IT professionals in the field of Type Theory

- National project.
- Duration: 1.1.2025 – 31.12.2027.
- Cultural and Educational Grant Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic.