

# *Waterproof*

## *Workshop & Experience report*

**Pim Otte**  
PhD Candidate

# Outline

Motivation

Context

Workshop

Results

Conclusion

# What is Waterproof?

- Mathematics is about proofs.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.
- A proof assistant is software that helps you write proofs.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.
- A proof assistant is software that helps you write proofs.
- Waterproof is a proof assistant using controlled natural language.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.
- A proof assistant is software that helps you write proofs.
- Waterproof is a proof assistant using controlled natural language.
- Why?
  - Goal: To teach writing paper proofs.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.
- A proof assistant is software that helps you write proofs.
- Waterproof is a proof assistant using controlled natural language.
- Why?
  - Goal: To teach writing paper proofs.
  - Forces precision.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.
- A proof assistant is software that helps you write proofs.
- Waterproof is a proof assistant using controlled natural language.
- Why?
  - Goal: To teach writing paper proofs.
  - Forces precision.
  - Immediate feedback.

# What is Waterproof?

- Mathematics is about proofs.
- Learning to prove is hard, being precise is hard.
- A proof assistant is software that helps you write proofs.
- Waterproof is a proof assistant using controlled natural language.
- Why?
  - Goal: To teach writing paper proofs.
  - Forces precision.
  - Immediate feedback.
  - Transferable to paper.

# Context

- Developed in Eindhoven by Jelle Wemmenhoven, Jim Portegies and many others [WAB<sup>+</sup>24].

# Context

- Developed in Eindhoven by Jelle Wemmenhoven, Jim Portegies and many others [WAB<sup>+</sup>24].
- Used in teaching Analysis 1 (since 2019, obligatory last this academic year).

# Context

- Developed in Eindhoven by Jelle Wemmenhoven, Jim Portegies and many others [WAB<sup>+</sup>24].
- Used in teaching Analysis 1 (since 2019, obligatory last this academic year).
- Used an Utrecht University in “Bewijzen in de Wiskunde”

# Workshop

`https://dikiedick.github.io/waterproof-browser/`

# Participants

Study Programme	Waterproof	Control
Mathematics (no double bachelor)	7	14
Mathematics and applications	0	9
Mathematics/Computer Science	13	0
Mathematics/Economics	1	5
Mathematics/Physics	9	14
Other:	4	4

# Performance on first quiz

Table: Mean Quiz 1 results by study programme and condition

Study Programme	WP (active)	WP (inactive)	Control
Mathematics (no double bachelor)	6.12	5.75	5.96
Mathematics and applications			4.61
Mathematics/Computer Science	8.00	6.78	
Mathematics/Economics	5.50		6.65
Mathematics/Physics	7.20	4.12	6.75
Other:	5.75	4.50	5.62



# Proof analysis

Table: Part of analyzed proof of participant 11 (no indentation)

Proof step	Code
...	...
- Case $q(14)$ .	signpost-case
Assume that $q(14)$ .	continued
Choose $y = 14$ .	$\exists$ -intro
{Indeed, $14 \in \mathbb{R}$ .}	domain-check
It holds that $\exists y \in \mathbb{R}, q(y)$ .	continued
We conclude that $(\neg p(x)) \vee (\exists y \in \mathbb{R}, q(y))$ .	signpost-concl
- Case $\neg p(x)$ .	signpost-case
Assume that $\neg p(x)$ .	continued
It holds that $\neg p(x)$ .	continued
We conclude that $(\neg p(x)) \vee (\exists y \in \mathbb{R}, q(y))$ .	or-intro, signpost-concl
End of proof.	continued

# Difference in proof steps

Table: Average occurrences of selected logical steps per student, by group

Code	Control	WP (Inactive)	WP (Active)
or-elim	0.70	0.65	1.44
signpost-case	2.78	3.12	4.31
exists-intro	1.46	1.41	2.31

# Interaction with generative AI

- Automation

# Interaction with generative AI

- Automation is algorithmic.

# Interaction with generative AI

- Automation is algorithmic.
- What is the main problem with Generative AI?

# Interaction with generative AI

- Automation is algorithmic.
- What is the main problem with Generative AI?  
Correctness!

# Interaction with generative AI

- Automation is algorithmic.
- What is the main problem with Generative AI?  
Correctness!
- Using a proof assistant means the computer can automatically check the proof.

# Interaction with generative AI

- Automation is algorithmic.
- What is the main problem with Generative AI?  
Correctness!
- Using a proof assistant means the computer can automatically check the proof.
- Positive feedback loop: Generative AI  $\Leftrightarrow$  Formalized mathematics.

# Conclusion

- Software solutions can help in teaching analog skills.

# Conclusion

- Software solutions can help in teaching analog skills.
- Look for checks in times of uncertainty.

**Thanks for your attention!**

# References



Jelle Wemmenhove, Dick Arends, Thijs Beurskens, Maitreyee Bhaid, Sean McCarren, Jan Moraal, Diego Rivera Garrido, David Tuin, Malcolm Vassallo, Pieter Wils, and Jim Portegies, *Waterproof: Educational software for learning how to write mathematical proofs*, Proceedings 12th International Workshop on Theorem proving components for Educational software. (ThEdu'23) (Julien Narboux, Walther Neuper, and Pedro Quaresma, eds.), 2024.



Utrecht  
University

Sharing science,  
*shaping tomorrow*

Joint work with Jim Portegies, Jelle Wemmenhoven and various teams of student programmers

---

TU/e  
LINDHOVEN  
UNIVERSITY OF  
TECHNOLOGY



[pim.otte.dev](https://pim.otte.dev)