Wouter ten Brinke

Enschede region, the Netherlands

✉ wouter.tenbrinke@icloud.com | ⌂ github.com/wtb04
in/wtb | 🌐 woutertenbrinke.nl

INTRODUCTION

I am a Master's student in Embedded Systems at the University of Twente. My background covers embedded software, computer vision, IoT systems, 5G and modern networking technologies, as well as full-stack and mobile development. I have worked on projects ranging from real-time Linux components and OpenCL-based GPU/SoC optimizations to cloud-connected applications. My main interests are in embedded software engineering, connected intelligent systems, IoT and computer vision.

EDUCATION

• **MSc Embedded Systems [🌐]**   *Sept. 2025 - Jul. 2027*
University of Twente   Enschede, the Netherlands

Two-year master's programme in Embedded Systems covering computer vision, IoT, embedded optimization, cloud-connected systems and embedded AI.

◦ Developed and modified a Python-based real-time face-detection system.

◦ Optimized matrix multiplication by moving a CPU implementation to both GPU (OpenCL) and an SoC platform, with part of the computation offloaded to an FPGA.

• **Erasmus+ Computer Engineering and Robotics Engineering [🌐]**   *Sept. 2024 - Feb. 2025*
Università degli Studi di Genova   Genova, Italy

Erasmus+ exchange semester in Computer Engineering and Robotics, Courses in Cloud Networking, 5G and IoT, Computer Vision, Real-Time Operating Systems and AR/VR.

◦ Developed a Linux kernel driver for a real-time task execution environment.

◦ Created an AI-driven interactive game in Unity using Convai.

◦ Set up automated deployment of a full monitoring and logging stack using Docker and Ansible.

◦ Achieved A1 level proficiency in Italian through a language course offered by the university.

• **BSc Technical Computer Science [🌐]**   *Sept. 2022 - Jul. 2025*
University of Twente   Enschede, the Netherlands

Three-year programme combining theoretical computer science with practical engineering. Covered all main areas of software engineering in both a theoretical and hands-on manner.

◦ Applied theoretical concepts such as algorithms, operating systems and automata in practical assignments.

◦ Engaged in interdisciplinary team projects, collaborating with students from various technical disciplines.

◦ Built a self-made programming language and compiler (ByteBender) as a course project.

◦ Bachelor thesis: FlexiTeX, a tool for structuring LaTeX projects.

• **Voorbereidend Wetenschappelijk Onderwijs (High School) [🌐]**   *Sept. 2016 - Jul. 2022*
Het Erasmus   Almelo, the Netherlands

Dutch pre-university high school program (VWO), completed with a science-oriented profile and aimed at preparing students for university studies.

◦ Completed the Science & Engineering and Science & Health subject tracks.

◦ Participated in the Technasium, working on real-world assignments for external companies.

◦ Nominee for the (national) Technasium Top Awards 2019 with a project on converting a 1930s school building to be gas-free.

◦ Graduation project on harvesting energy from water flow in household systems (e.g., rainwater piping).

Work experience

- **Software Developer (Part-time) [🌐]** *Feb. 2023 - Present*
Cloudwise B.V. Hengelo, the Netherlands

Full-stack development in an English-first, multi-location development team. Focussing on the development of the Klasbord Ouderapp, a platform used by schools and parents to share updates, photos, videos and announcements across web and mobile.

○ Backend development in ASP.NET for new functionality and platform maintenance.

○ Mobile app development in Xamarin and MAUI, including the migration after Xamarin's end-of-life.

○ Web development in Angular for user-facing features and interface behaviour.

○ Full OpenTelemetry integration for background jobs and platform-wide tracing.

○ Continuous bug fixing to keep the application stable and reliable.

Projects
🌐 = Portfolio, 🔗 = Demo, ⌂ = GitHub, 🔓 = UTwente

- **FlexiTeX (Bachelor thesis) [🌐, 🔗, ⌂, 🔓]** *University project*
LaTeX Collaboration Without Giving Up Personal Project Structure

LaTeX gives users a lot of freedom in how they structure their projects, but this becomes a problem when working with others. Popular collaboration tools assume that all users follow the same project structure, which does not reflect how people actually prefer to organize their work. This thesis introduces FlexiTeX, a system that allows each user to keep their own project structure while still collaborating on the same content. The system works by flattening a LaTeX project, parsing it into an abstract tree that captures the logical structure. It then applies transformation rules to rebuild the project structure based on a configuration file. The transformation is designed to be reversible, idempotent and preserve the ability to compile the document. A proof of concept shows how this approach can be used in a collaborative setup where each user works in a personal branch and changes are synced through a shared internal version. An evaluation on real-world projects shows that the system preserves content in most cases, although some limitations remain due to parser behavior. Overall, FlexiTeX makes it possible to collaborate on LaTeX projects without forcing everyone to adopt the same structure.

- **CalendarChanger [🌐, ⌂, 🔗]** *Personal project*
Smart ICS Proxy for Structured and Customizable Calendars

CalendarChanger is a self-hosted iCalendar (ICS) proxy that fixes and enriches exported calendar feeds. The project began when the University of Twente adopted TimeEdit in 2024. While TimeEdit provided technically valid ICS files, it stored all details inside the description field instead of using the correct iCalendar fields such as LOCATION and ATTENDEE. This made the data unreadable in most calendar apps. CalendarChanger acts as a middle layer between TimeEdit and the calendar client, extracting and reformatting event data to fully leverage standard ICS semantics. It cleans titles, assigns teachers as attendees, and separates lectures, exams, and practicals into individual feeds so users can color-code them in their calendar.

It also adds richer context by inserting Apple Maps locations and linking building names to a custom Mazemap redirect endpoint that resolves rooms to Mazemap locations. The system runs continuously, fetching updates from the original feed and serving always up-to-date subscriptions.

The web interface lets users hide or modify events such as cancelled classes or changed times even though the source is read-only. Each event is hashed by its contents so when TimeEdit updates an event it reverts to its original form ensuring consistency. The backend is built with ASP.NET Core and Razor Pages, containerized with Docker, and deployed automatically with Ansible to a private cloud server.

- **ByteBender Language and Compiler [🌐]** *University project*
Designing and Implementing a Language for the Sprockell CPU

ByteBender is a custom programming language and compiler developed as part of a university project. It translates high-level code into machine instructions for the Sprockell processor, a simple Haskell-based CPU. The project includes a frontend for parsing and type-checking, a backend for code generation, and integration with the Sprockell simulator for execution. The language supports multithreading, synchronization primitives, and shared memory operations.

Publications
T = Thesis   C = Conference paper

[T.1] Wouter ten Brinke. FlexiTeX: LaTeX Collaboration Without Giving Up Personal Project Structure. Bachelor thesis, University of Twente, June 2025. http://purl.utwente.nl/essays/107262

[C.1] Wouter ten Brinke, Bart Griepsma, Aleksandra Ignatovič, Nhat, Vadim Zaytsev. On the Structuring of LaTeX Projects. Conference paper, Pre-proceedings of the 24th Belgium-Netherlands Software Evolution Workshop (BENEVOL), pp. 39-45, 2025. https://benevol2025.github.io/pre/paper10.pdf

Additional Information

Languages: Dutch (Native), English (C1 (Fluent)), German (B2 (Intermediate)), Italian (A1 (Basic))