Technische
Universität
Braunschweig

Desigining a Curing System for the ISAN project

Project Thesis

Nikkel Lennart Heesen
Braunschweig, 10.11.2025

Advisors:	Viktor Sobotta
Supervisor:	Prof. Dr. Thomas M. Deserno
Matriculation Number:	4810083

Technische Universität Braunschweig
Carl-Friedrich-Gauß-Fakultät
Peter L. Reichertz Institut

PLRI
PETER L.
REICHERTZ INSTITUT
FÜR MEDIZINISCHE
INFORMATIK

Hello I am an abstract

Contents

1 Introduction

I am the introduction

2 Environment

Typescript: **type safety**, better maintainability, reducing common errors associated with dynamic typing. Sass introduces advanced styling capabilities ensuring scalable styling through reusable design components. Vue.js offers great reactivity with a component-based architecture ensuring great modularity. As a build tool using Vite accelerates development with fast build and its hot-module reloading, optimized bundling. The templating engine of Flask is Jinja good for flexible integration between the python back-end and the local client app.

Back-end relies on Python which is widely adopted and has an extensive ecosystem. Flask offers a minimal yet flexible framework for rapid service development, while Connexion enforces OpenAPI-driven design, enabling automatic validation, documentation and maintainable APIs.

Together, this stack provides strong reactivity for fast reactive visualization on the front end and robust workflows on the back end while enabling well-supported maintenance through comprehensive tooling.

Table 1: Technology stack

	Front-end	**Back-end**
Languages	HTML, SASS, TS, Jinja	Python
Frameworks	Vue.js (with Vite)	Flask/Connexion

2.1 Development setup

To streamline front-end development, hot reloading is an essential tool. Vite's development server can track changes and reload the webpage automatically. Frameworks like Symfony have integrations and project skeletons for development and production, which configures everything. For Flask I found the project flask-vite. This project provides only beta versions yet and is maintained by one person, with few contributions from other people. I decided to write my own helper functions for a better integration in the environment of the already existing ISAN environment and full transparency of what happens to make extending functionality easier.

Why not just start vite's development server and access it directly? That way the backend logic isn't used, requests send to server from the client side won't have valid responds. This could be solved by a proxy setting inside vite. That way requests matching a certain url pattern will get actually sent to the backend server on for example a different port. However this still doesn't account for template rendering via jinja. Creating DOM Nodes and contents happens to some degree on the backend. So instead of accessing the vite server directly we access the backend server hosted via flask. Hot reloading of changes inside the backend is now natively handled by the flask server. In order for hot reloading on the frontend to work the assets server by the vite dev server need to be included. But if no vite development server is detected the statically built assets are included instead as they should for production.

Elaborate why it makes sense to be on the backend

`asd`

app/vite_integration.py

```python
def is_vite_running() -> bool:
        try:
            requests.get(VITE_SERVER, timeout=0.2)
            return True
        except requests.RequestException:
            return False
```

Listing 1: Dynamically checking wether the development server is responsive

Figure 1: Returns the current known scripts either from the development server via crossorigin or with a direct path to the built assets.

3 Background

- research of design choices and standards in emergency software.
- analysis of other software projects similar to this

3.1 ISO Standards & Accessibility

DIN EN ISO 9241:

- key umbrealla standard for interactive systems (software and hardware)
- ISO 9241-110 specifically states it isn't for every situation

BITV 2.0:

- links to EN 301 549 which lists WCAG guidelines in chapter 9 and encourages AAA conformance if possible
- https://www.etsi.org/deliver/etsi_en/301500_301599/301549/03.02.01_60/en_301549v030201p.pdf
- contrast for example: https://www.w3.org/TR/WCAG21/#contrast-enhanced

3.2 Other Software

rescueTABLET:

- over 750 organisations in germany
- dashboard, sidebar menu, tiles

tablet command:

- https://www.fireengineering.com/fdic-coverage/tablet-command-introduces-two-new-features/
- more than 800 public safety agencies across us and canada
- managed over 170,000+ incidents (2024)
- purpose built for fire service
- more focus on unit management
- custom configurable
- web portal backend
  - 2.0 released in 2021
  - dashboard and sidebar

3.3 The ISAN project

- International Standard Accident Number: a master case index linking accident & emergency with medical data [1]
- Automatic alerting of accidents and emergencies: The international standard accident number and vital sign data embedded in future PACSa [2]
- Proposing an international standard accident number for interconnecting information and communication technology systems of the rescue chain [3]
- Automatic information exchange in the early rescue chain using the International Accident Number (ISAN) [4]

Bibliography

[1] H. P. Le, S. Hackel, A. Guenther, R. Goldschmidt, M. Daoud, and T. M. Deserno, “International Standard Accident Number: a master case index linking accident & emergency with medical data,” ICT for Health Science Research. IOS Press, pp. 120–124, 2019.

[2] R. Barakat and T. M. Deserno, “Automatic alerting of accidents and emergencies: The international standard accident number and vital sign data embedded in future PACS,” in Medical Imaging 2020: Imaging Informatics for Healthcare, Research, and Applications, 2020, pp. 252–258.

[3] N. Spicher et al., “Proposing an international standard accident number for interconnecting information and communication technology systems of the rescue chain,” Methods of information in medicine, vol. 60, no. S1, pp. e20–e31, 2021.

[4] M. Haghi et al., “Automatic information exchange in the early rescue chain using the International Standard Accident Number (ISAN),” in Healthcare, 2021, p. 996.

Statutory Declaration

I hereby declare in lieu of an oath that I have written this Master's Thesis "Desigining a Curing System for the ISAN project" independently and that I have cited all sources and aids used in full and that the thesis has not already been submitted as an examination paper.

Braunschweig, 10.11.2025

(signature with first- and lastname)