

# Project Proposal – Twin page: the user interface to digital twin data

## 1. Introduction

Despite all Internet of Things and eXtended Reality development efforts, the information gap between physical and digital worlds is wide. A newish concept called “digital twin” is here to close that gap, but unfortunately most digital twins are built with legacy systems that are not properly interoperable with each other. ([Google digital twin](#) if it is not familiar. We concentrate on data transfer from one organization to another rather than simulation and 3D models.)

Twinbase uses “digital twin description documents” to combine (meta)data and services from several systems into one digital twin entity. This combined digital twin should contain all digital information about the corresponding real-world entity. A “twin document” can be thought of as an “HTML document for digital twins”. However, there is currently no winning standard for twin document, so we pick best practices from the suitable standards and fill in gaps where necessary. The overall approach is called “Digital Twin Web”, which is like the World Wide Web, but aims to combine the physical and digital worlds into a *phygital* world.

We are on the way to start a startup around the methodology and are currently in the middle of a Research to Business project “Twinbase”, funded by Business Finland. Twinbase is also the name of the software platform that we are developing. The primary function of Twinbase is to store and distribute twin documents, supporting functions include searching and visualizing the documents.

The task assigned in this project proposal is to develop a web user interface for the twin documents, in other words, develop a “twin page” for Twinbase. This web UI should become a part of the Twinbase platform to help human users understand the contents of the twin documents more intuitively. This user interface is generated automatically for each twin document based on the global data models (=semantic web vocabularies). (Twinbase uses Semantic Web technologies to build a solution that is interoperable and scalable across organizations.)

End users are ordinary people who are interested in the information the twin document contains. They are not experts on Semantic Web concepts or technologies, so the jargon must be translated to ordinary language. The users are often domain experts in the actual content of the twins.

## 2. Project goals

The main goal of the project is to build a user interface that shows the contents of twin documents in a way that is intuitive and understandable for domain experts. The user interface solution must adapt to all twin documents stored in Twinbase and to a specified set of data types used in those documents. Each twin document contains different information, and the twin page should present each data type in an optimal way. We select the supported data types during the project.

Examples for features of the user interface:

- Coordinates should be presented on a map.
- Live data source should be shown in a dynamic time-series graph.
- Relationships to other twins could be shown in a visual graph.
- 3D model should be shown as a rotatable 3D object ([example](#)).

- Swagger-style data fetcher for REST APIs.
- The order of data blocks should align with what is most important data for the user.
- Authentication and granular access control to the twin page is a requested extra feature.

Inspiration for what the contents of twin page are:

- Simple prototype of twin page for [crane](#) and [windmill rotor](#). (Too boring and technical.)
- The amount of unnecessary data presented in a suboptimal way on a [Wikidata page](#) makes it difficult to form a quick overview of the item.

### Extra assignment ideas

In addition to this straightforward service that we need, there are several extra development topics to support the Twinbase development and the general ambition to build the Digital Twin Web. Some of these include:

- Generating (thousands of) twin documents from publicly available databases ([example](#)).
- Built-in web editor for twin description documents.
- Enhancing the documentation of existing vocabularies for users of Twinbase.
- Vocabulary publisher service (transform [.ttl vocabulary](#) into a [web page](#))
- Built-in web editor for vocabularies.
- Generating vocabularies from other formats, such as OpenAPI/Swagger documents.
- Visual vocabulary mapper tool to help users map types to other types.
- Case-specific extension development with our pilot customer(s).

These extra assignments are quite open-ended and should be discussed and specified with the team.

## 3. Technologies

The student team may select their preferred technologies, although it should be possible to integrate the twin page feature into the Twinbase platform that we are developing.

We currently use:

- Languages and frameworks: Python, FastAPI, Typescript, React (Next.js)
- Data modeling / semantics: [RDF](#), [Turtle](#), [JSON-LD](#), [WoT TD](#), [RDFLib](#)
- Other infra: Keycloak, Docker, Azure, Ubuntu, Git

The assignment probably can be implemented with the technologies we already use, but the team may find better ones and many new libraries probably need to be introduced for the features.

## 4. Requirements for the students

### Skills

Ability to create modular solutions is a crucial for succeeding in this assignment.

Frontend is the primary focus of the project and students are expected to have skills in that, for example in the React framework. We are not able to support much in frontend technologies.

Existing knowledge of Semantic Web technologies ([see course](#)) is an advantage in this project, although the concepts and technologies can also be learned during the project.

Familiarity with the digital twin concept is considered a plus. Ability to think in abstract concepts helps in understanding the Twinbase solution and methodology.

### Difficulty

Creating a simple twin page that covers a minimum amount to data types is considered easy.

Building a truly intuitive solution and large data type coverage require empathizing with the end user and efficient work ethics. Therefore, difficulty of building a fine quality service is considered moderate.

Ideating, specifying, and implementing one or more extra feature or service requires deep understanding of the topic, rigorous work ethics, and is considered demanding.

## 5. Legal Issues

Intellectual Property Rights (IPR): A. All IPRs to all Results will be transferred to the Client.

Confidentiality: A. The client will share some confidential information with the students.

We may choose to publish some of the project results under an open-source software license. This is a strategic decision for us and will be discussed with the team members during the project.

## 6. Client

Twinbase is a Research to Business project, hosted by Department of Mechanical Engineering at Aalto University, and funded by Business Finland. The goal of the project is to research methodologies, develop technologies, and find business opportunities so that the research group can start a startup after the project in summer 2024. We are looking for co-founders and first employees.

Client representatives:

Project manager (Postdoctoral researcher)

**Juuso Autiosalo**

Expertise in the digital twin concept and the high-level architecture of our solutions

e-mail: [juuso.autiosalo@aalto.fi](mailto:juuso.autiosalo@aalto.fi)

+358 50 340 9550

Room I406a, Otakaari 5, 02150 Espoo

Software developer (Project advisor)

**Jani Hietala**

Expertise in IoT programming, technical architecture, and implementation

e-mail: [jani.hietala@aalto.fi](mailto:jani.hietala@aalto.fi)

Each of us is prepared to support the team 4 hours a week, more if needed and possible. We use the same office space so on-demand guidance should be straightforward. The beginning includes an introduction to the Twinbase platform and Semantic Web technologies.

We have an office room at A Grid, Otaniemi, which the project team can use. We can try to arrange some programming parties with refreshments as long as we stay in line with university policies.

## 7. Additional information

We are looking for employees for the project and co-founders for the startup.