

# Web Engineering

Summer Semester 2025

Professor: Jürgen Cito

Assistants: Michael Schröder, Samuel Pilz, Nathanael Nussbaumer, Markus Böck

Tutors: Nicolas Hrubec, Jakob Jeschke, Valentino Lazarevic, Lukas Schatzer, Si Yu Sun, Marin Toma

# Web Engineering

Mandatory (Pflichtmodul), 4th Semester:

033 526 **Wirtschaftsinformatik**

Elective (Wahlmodul):

033 521 **Informatik**

# ~~Module „Development of Web Applications“ (Entwicklung von Web-Anwendungen)~~

Course

**3.0h/6 ECTS VU 188.951 Web Engineering**

~~To successfully complete the module, you have to  
successfully complete both courses~~

**Both Web Engineering and  
Semi-structured data are now their  
own modules going forward  
(with 6 ECTS each).**

**There will be no option to take a 3  
ECTS version.**

**Have a look at the  
“Übergangsbestimmungen”  
to see how this may affect you**

# Course Structure - Theoretical Part

## Flipped Classroom

Students watch lectures online and do pre-reading before attending live lectures.

Live lectures will go over “katas” (small examples to deepen understanding) and answer student questions to theoretical aspects

05.03. Kick-off / HTTP

12.03. HTML/CSS

02.04. JavaScript

07.05. Backend Abstractions + Sockets (Node.js)

28.05. Frontend Abstractions (Vue.js)

11.06. Guest Lecture (TBD)

# Project “Artmart”

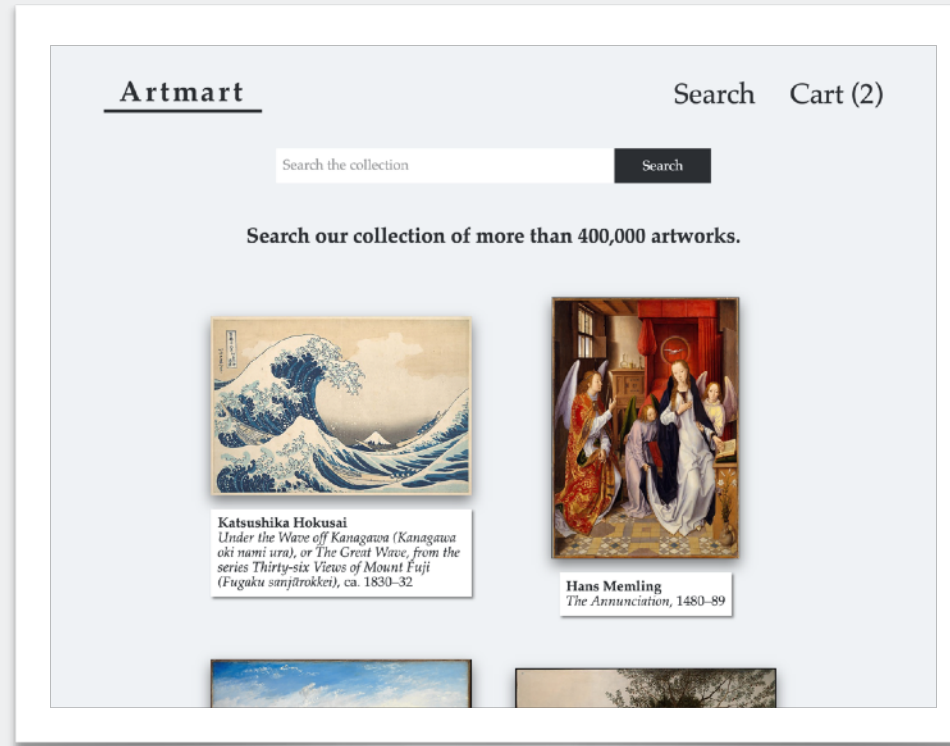
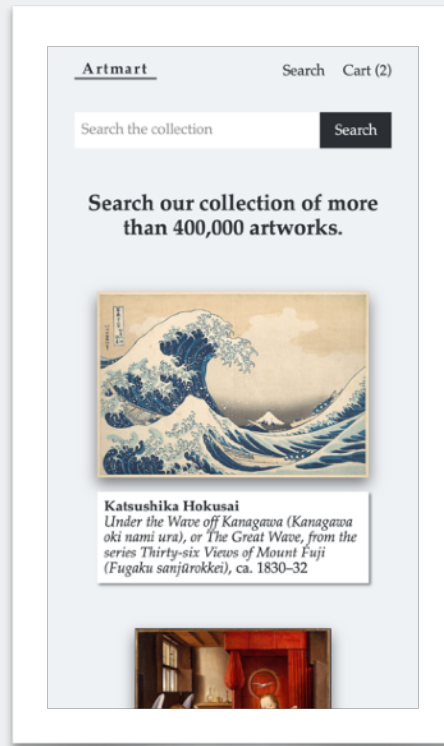
You are tasked with building the website for Artmart, a web shop for fine art prints.

The screenshot displays the Artmart website interface. At the top left, the logo "Artmart" is underlined. At the top right, there are links for "Search" and "Cart (2)". Below the logo is a search bar with the placeholder text "Search the collection" and a "Search" button. A message below the search bar reads "Search our collection of more than 400,000 artworks." The main content area features a grid of ten art prints, each with a title, artist name, and a brief description.

Artist	Artwork Title	Approximate Date
Katsushika Hokusai	Under the Wave off Kanagawa (Kanagawa oki nami urai), or The Great Wave, from the series Thirty-six Views of Mount Fuji (Fugaku sanjūrokkei)	ca. 1830–32
Hans Memling	The Annunciation	1480–89
Joseph Mallord William Turner	Venice, from the Porch of Madonna della Salute	ca. 1835
Pieter Bruegel the Elder	The Harecooters	1563
Vincent van Gogh	Wheat Field with Cypresses	1889
Albrecht Dürer	Melencolia I	1514
Ferdinand Hodler	The Dream of the Shepherd (Der Traum des Hirten)	1896
Hans Holbein the Younger	Erasmus of Rotterdam	ca. 1532
Paul Cézanne	Still Life with Apples and a Pot of Primroses	ca. 1890
Thomas Eakins	[Thomas Eakins and John Laurie Wallace on a Beach]	ca. 1883

# Project “Artmart”

You are tasked with building the website for Artmart, a web shop for fine art prints.



## Assignment 1:

### Static HTML/CSS

- HTML5
- WAI
- CSS3
- Responsive Design

# Automated Grading AI — Component Differencing

✘ 302 — Looks correct at 1440px width

-1

The page does not look correct: 4 components are in the wrong place, there is 1 unexpected component and 1 expected component is missing.

Your Page

Artmart Search Cart (2)

Hans Memling  
Portrait of a Young Man, ca. 1472-75

Vincent van Gogh  
Wheat Field with Cypresses 1889  
Medium print in a 3.3 cm natural frame with a 1.7 cm mint mat.  
€ 114.50

Total: € 194.50  
Checkout

- matching
- moved
- unexpected
- expected components

Expected

Artmart Search Cart (2)

Hans Memling  
Portrait of a Young Man, ca. 1472-75  
Small print in a 5 cm classic frame.  
€ 80.00

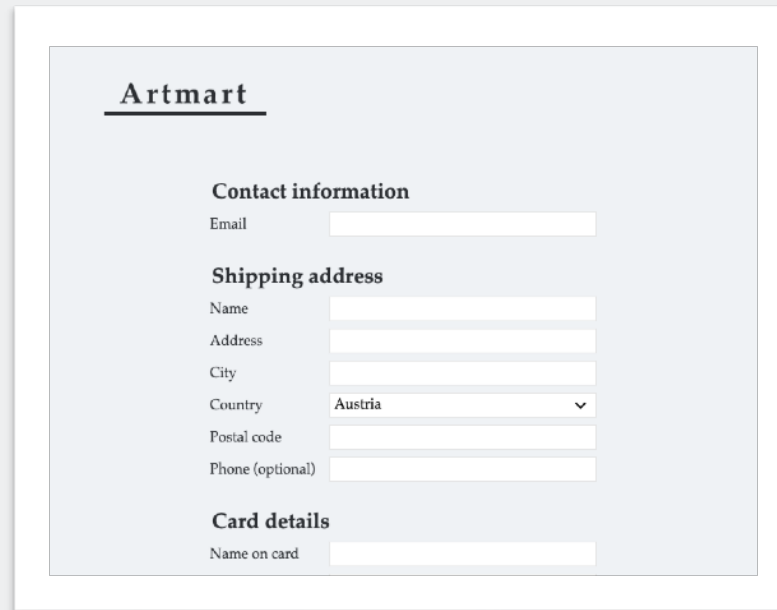
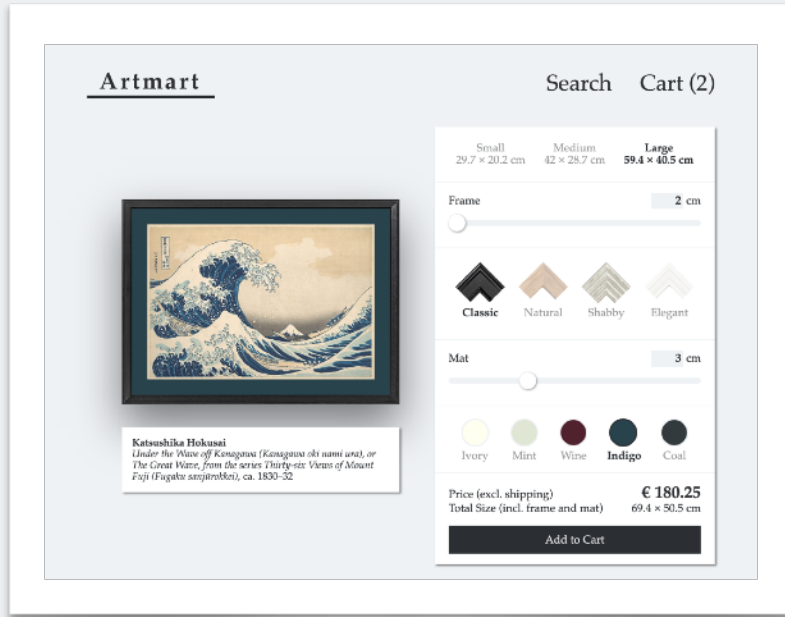
Vincent van Gogh  
Wheat Field with Cypresses, 1889  
Medium print in a 3.3 cm natural frame with a 1.7 cm mint mat.  
€ 114.50

Total: € 194.50  
Checkout

- matching
- moved
- missing
- your components

# Project “Artmart”

You are tasked with building the website for Artmart, a web shop for fine art prints.



## Assignment 2:

### Web Interactivity

- JavaScript/DOM
- Service Interaction (Metropolitan Museum)

## Assignment 3:

### Backend

- Node.js
- REST API
- Server Sockets

## Assignment 4:

### Frontend Abstraction

- Vue.js
- Web Sockets



# Automated Grading A2-A4 — Integration Testing

## x 209 — Show number of items in cart

### start intercepting HTTP requests [less info](#)

- GET requests to relevant parts of the Met API will return **random** responses.
- GET requests to `images.example.com` will return a test image.
- GET requests to `localhost` will continue unaltered.
- All other requests will be aborted.

### put **two** items in the cart [less info](#)

```
localStorage.setItem('cart', JSON.stringify([
  {
    "objectID": 747867,
    "printSize": "M",
    "frameStyle": "classic",
    "frameWidth": 29,
    "matColor": "mauve",
    "matWidth": 9
  },
  {
    "objectID": 52391,
    "printSize": "L",
    "frameStyle": "elegant",
    "frameWidth": 23,
    "matColor": "mint",
    "matWidth": 8
  }
]))
```

### navigate to `http://localhost:4444/framing.html?objectID=99925`

expect `innerText` of `nav > a[href="cart.html"]` to be `Cart (2)`

Expected `innerText`: "Cart (2)"

Actual `innerText`: "Cart"

## x 109 — Cache Met objects

### start intercepting HTTP requests [less info](#)

- GET requests to relevant parts of the Met API will return **random** responses.
- GET requests to `images.example.com` will return a test image.
- GET requests to `localhost` will continue unaltered.
- All other requests will be aborted.

### navigate to `http://localhost:4444/search.html?q=vejturvo`

expect Met API requests for objects **878746, 854085, 200039**

### Expected requests:


```
https://collectionapi.metmuseum.org/public/collection/v1/objects/878746
https://collectionapi.metmuseum.org/public/collection/v1/objects/854085
https://collectionapi.metmuseum.org/public/collection/v1/objects/200039
```

### Actual requests:

```
https://collectionapi.metmuseum.org/public/collection/v1/objects/39799
https://collectionapi.metmuseum.org/public/collection/v1/objects/459055
https://collectionapi.metmuseum.org/public/collection/v1/objects/437853
https://collectionapi.metmuseum.org/public/collection/v1/objects/435809
https://collectionapi.metmuseum.org/public/collection/v1/objects/436535
https://collectionapi.metmuseum.org/public/collection/v1/objects/360018
https://collectionapi.metmuseum.org/public/collection/v1/objects/634108
https://collectionapi.metmuseum.org/public/collection/v1/objects/459080
https://collectionapi.metmuseum.org/public/collection/v1/objects/435882
https://collectionapi.metmuseum.org/public/collection/v1/objects/271890
https://collectionapi.metmuseum.org/public/collection/v1/objects/459054
```

(The order of requests does not matter.)

# Web Engineering Diary Study — Bonus Points

- for each assignment, **upload a short diary of your experience** solving it
- we are interested in the use of **external sources and tools**
  - AI assistants: GitHub Copilot, ChatGPT, Bing AI chat, LLaMA,...
  - Q&A sites and discussion forums: StackOverflow, TUWEL, Discord,...
- you can get **up to 2 bonus points per assignment**
  - be as detailed as possible — screenshots are welcome!
  - contents of diaries do not affect the grading of your assignment
- more information in TUWEL  [A1 diary \(bonus points\)](#)