Semester Project 2

Project   ▲

Semester Project 2 brief

Table of Contents

- Goal

- Brief

- Requirements

- API

- Approved resources

- Delivery

Goal

To take the skills learned over the past three semesters and create an auction website.

Brief

An auction site is looking to launch a website where users can add items to be bid on and bid on items other users have put up for auction.

When a new user joins the website, they are given 1000 credits to use on the site. They can get credits by selling items and use credit by buying items. Non-registered users can search through the listings, but only registered users can make bids on listings.

Requirements

> All API functionality is managed by an existing application. This project only covers the front-end application for the API.

API

The API you are using for this project can be found under Auction Endpoints in the Noroff API documentation ⤴.

Resources

API Swagger ⤴

User stories

The client has specified the following requirements in the form of *User Stories*:

1. A user with a `stud.noroff.no` email may register

2. A registered user may login

3. A registered user may logout

4. A registered user may update their avatar

5. A registered user may view their total credit

6. A registered user may create a `Listing` with a title, deadline date, media gallery and description

7. A registered user may add a `Bid` to another user's `Listing`

8. A registered user may view `Bid`s made on a `Listing`

9. An unregistered user may search through `Listing`s

Technical restrictions

The company CTO has set the following technical restrictions:

1. Must use an approved `CSS Framework`

2. Must be hosted on an approved `Static Host`

3. Must use an approved `Design Application`

4. Must use an approved `Planning Application`

Required links

The Product Owner has requested links to the following:

1. A Gantt chart for project timing

2. A design prototype

3. A style guide

4. A kanban project board

5. A repository link

6. A hosted application demo link

Approved resources

This list covers libraries and services that have been vetted by the company and approved for use.

CSS processors

- SASS/SCSS

- PostCSS

CSS frameworks

- Bootstrap (version >5.0.1)

- Tailwind (version >3.0.0)

- MUI (version >5.11.8)

Hosting services

- GitHub Pages

- Netlify

Design applications

- Adobe XD

- Figma

- Sketch

Planning applications

- Trello

- GitHub Projects

Delivery

Include the required links in the Moodle delivery window using this template format (delivery-template.html).

All final changes must be merged into the default branch `main` or `master`. Other branches will not be checked.

Ensure that the `readme.md` file describes your project thoroughly, including how to setup and run the project locally and any special instructions for testers.