

Deliverables Milestone

1 - 4

Team 12

Use cases description

Registration: A new visitor of the site should be able to register and thus create an account. You can go to the registration page by clicking on the sign up link which is present on all the pages in the top-nav bar.

Login: The visitor of the site should be able to login in to the page with his created account using his username and password. This should be possible by typing in the login information in the input fields in the top-nav bar.

Profile page: The logged in user should be able to go to his account/profile page, where he sees all his registration information (Name, Surname, Adress). He should be able to change this information except the username. The password also should be changeable, but in a seperate process.

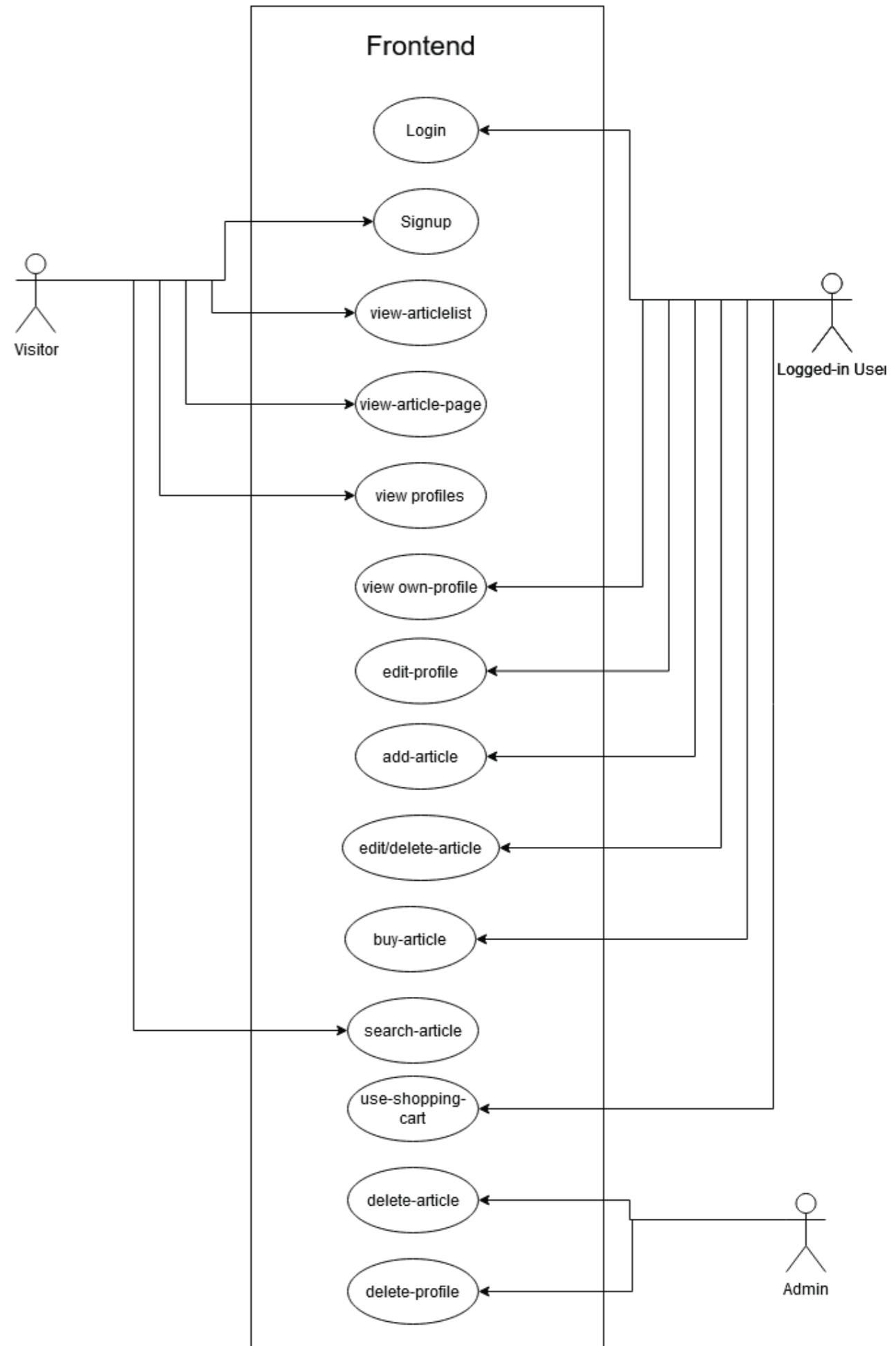
Add, update, remove goods/services: The logged in user should be able to add an article to sell, lend or provide as a service. This is done by creating a post which has a title, a description etc. The user should also be able to go into the created post and change things or should also be able to delete a post.

Search goods/services: The visittor should be able to use a search bar and by typing in searchterms search for specific articles. There should also be a filter function where you can specify the location, price and delivery otions for the sought articles. The sought articles should be shown on a separate page and by clicking on one article, you come to it's own page, with a more detailed insight to what it actually is.

Buy/sell goods/services: If a user wants to buy an article, he can do so on the specific article page. He needs to be logged in to do so. If an article gets bought, the article should dissapear from the page and also shouldn't be able to be found again by searching. If one wants to sell an article, he needs to be logged in and can do so by adding an article. After adding it, visitors should be able to see the newly added article.

Use a shopping cart: If a user wants to buy an article, he can do so on the specific article page. But if he wants to buy products, then he can use the shopping cart. By clicking on a buy button on the article, it gets added to the shopping cart. This way, he can add as many articles as he likes and when he's done with shopping, he can buy all the products in one go.

Use case diagram



User stories for the first milestone

Visitor: I'd like to see things on your page I might want to buy. If I want to, then I'd make an account and buy it!

Admin: If some user tries to sell things which are against the guidelines, I should be able to ban/delete/suspend his account.

Seller: If I sell things on your website, then people see where I come from. I should be able to change my personal information because I might move to another place of living.

IT: Every account should have the possibility to reset their password by letting them receive a reset password link via email, so I don't have to handle all the requests from people who forgot their password.

Grandma: Can I visit your website with my old mobile phone?

Impatient Visitor: If I'm only able to register and login, then I'll write a mail to the development team so they should speed up the coding!

User stories for the second milestone

Seller 2: I'd like to sell things, lend things and even provide a service. But if I want to change something in the description, I would like to be able to change it without creating a new post.

User stories for the third milestone

Seller 3: If I've added an article which I want to sell and it gets bought, I want the whole process to be handled by the page. So I shouldn't be the one who needs to delete the sold article now.

Seller 4: When I have added a new article, I want that the article is visible on the page and not buried somewhere only accessible with the specific search terms. Maybe put the newly added articles on the front page or something.

Buyer 1: If I go to this page, I want to find exactly what I'm looking for with the search function. If I can use some filters to narrow down the found articles, this would be awesome.

Buyer 2: When there is an article shown on the page as being available to buy, then I hope so it is. Otherwise I'll file a lawsuit.

User stories for the fourth milestone

Buyer 3: When I'm shopping, I don't want to do the payment process for every single product I want to buy. I want a shopping cart, where I can put everything into and then just pay once, so it makes my life easier.

List of functional and non-functional requirements for all milestones

Functional requirements

- Visitor can open up website --> **Visitor**
- Visitor can register himself to create account --> **Visitor**
- Website can check if username/email is already in use
- Accounts can be created in the backend
- Visitor with registered account can login on the page. --> **Visitor**
- Website checks if login credentials are correct
- Logged in user can look at his profile
 - See information --> **Seller**
 - Change information --> **Seller**
 - Change password --> **IT**
- Admin should be able to ban/suspend/delete accounts --> **Admin**

Non-Functional requirements

- Deadlines for different milestones and deliverables --> **Impatient visitor**.
- Website/App should be compatible also with older technologies e.g. older smartphones/computers.
- **Grandma**

List of functional and non-functional requirements for milestone 2

Functional requirements

- Users can sell/lend a product or provide a service --> **Seller 2**
- User can edit or delete these posts --> **Seller 2**

List of functional and non-functional requirements for milestone 3

Functional requirements

- Sold article gets removed automatically by the site --> **Seller 3**
- Newly added articles get shown on the frontpage/subpages at the top --> **Seller 4**
- Usable search function with possibility of using filters--> **Buyer 1**
- If article is shown, then it hasn't been bought yet --> **Buyer 2**

List of functional and non-functional requirements for milestone 4

Functional requirements

- Articles can be added to shopping cart and then the sum of them can be processed --> **Buyer 3**

CRC cards for identified user stories

Home / Front	
• Displays recently added articles • Contains link to displayed articles • Contains link to category-pages	• Articles • Category
Category	
• Displays all articles of this category (divided in Sell, Lend, Service) • Contains link to displayed articles	• Articles • Filter
Search-Tool	
• Takes input to search database • Conducts search • Routes to Search-Results	• Topbar
Search-Results	
• Displays articles according to the search term • Contains links to displayed articles	• Search-Tool • Article

Topbar	
<ul style="list-style-type: none"> Contains link to home Contains login form Contains signup link Contains search tool <p>If logged in:</p> <ul style="list-style-type: none"> Contains link to add article Contains link to profile Contains balance Contains logout button 	<ul style="list-style-type: none"> Home Profile Signup Add Article Search-Tool Balance

Article	
• Contains information about article (Specific for Sell, Lend or Service)	<ul style="list-style-type: none"> Article info service Controller

Article Info Service	
<ul style="list-style-type: none"> Able to fetch article info from backend Provides Dependency Injection for Article 	• Article

Userprofile	
<ul style="list-style-type: none"> Able to view own profile Able to change own profile Able to see old/current articles 	<ul style="list-style-type: none"> Articles Controller User Info Service

Signup	
<ul style="list-style-type: none"> Contains form to register a new user Checks if username/email is already in use Automatically logged in after signup 	<ul style="list-style-type: none"> Home Controller

Add-Article	
<ul style="list-style-type: none"> Contains add-article forms (for sell, lend and service) Creates a post depending on article-type 	<ul style="list-style-type: none"> Article User Controller

Balance	
<ul style="list-style-type: none"> Display current balance Ability to increase balance Contains link to terms of service 	<ul style="list-style-type: none"> Topbar User

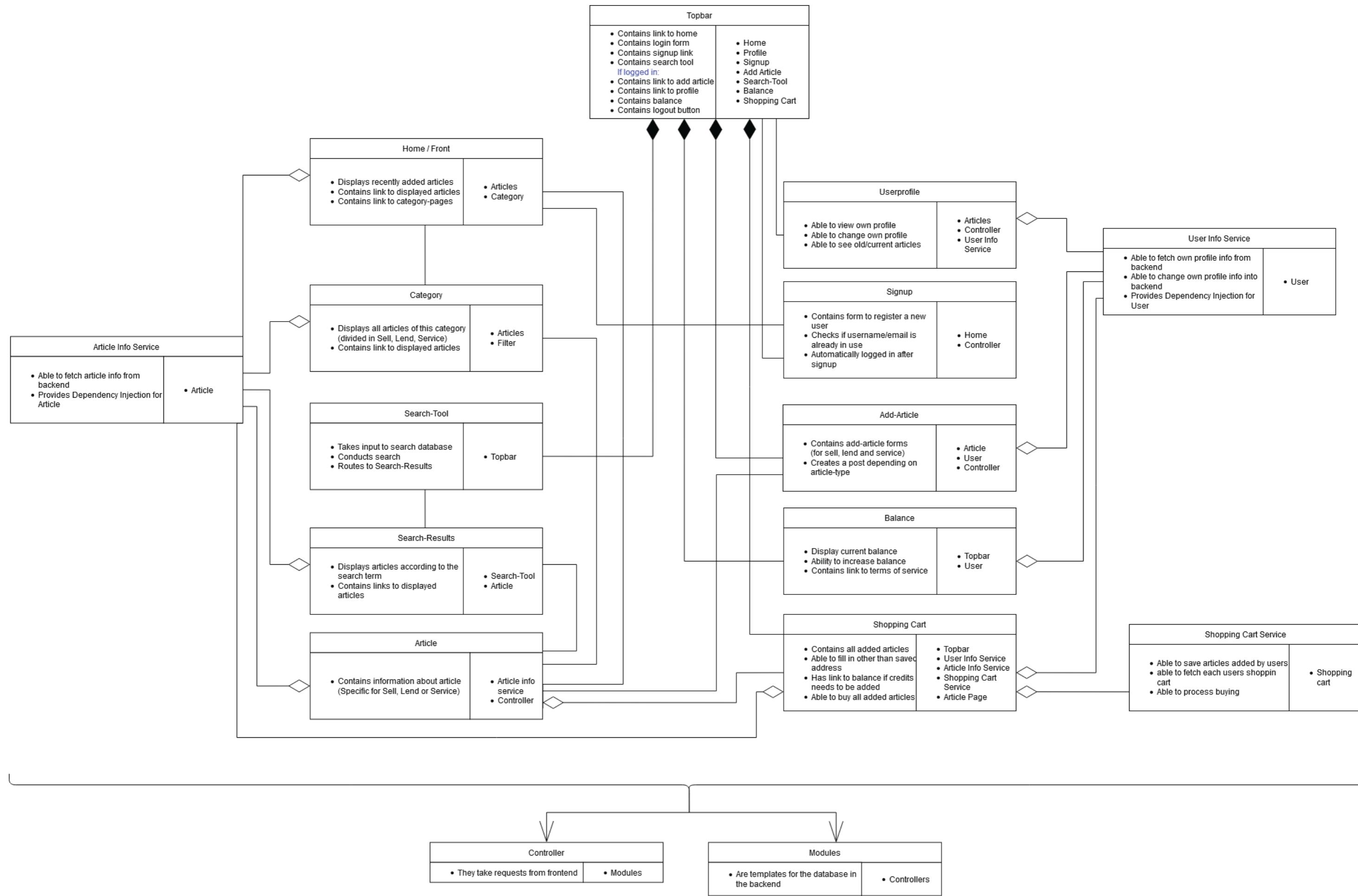
Controller	
• They take requests from frontend	• Modules
Modules	
• Are templates for the database in the backend	• Controllers

Shopping Cart	
<ul style="list-style-type: none"> Contains all added articles Able to fill in other than saved address Has link to balance if credits needs to be added Able to buy all added articles 	<ul style="list-style-type: none"> Topbar User Info Service Article Info Service Shopping Cart Service Article Page

User Info Service	
<ul style="list-style-type: none"> Able to fetch own profile info from backend Able to change own profile info into backend Provides Dependency Injection for User 	• User

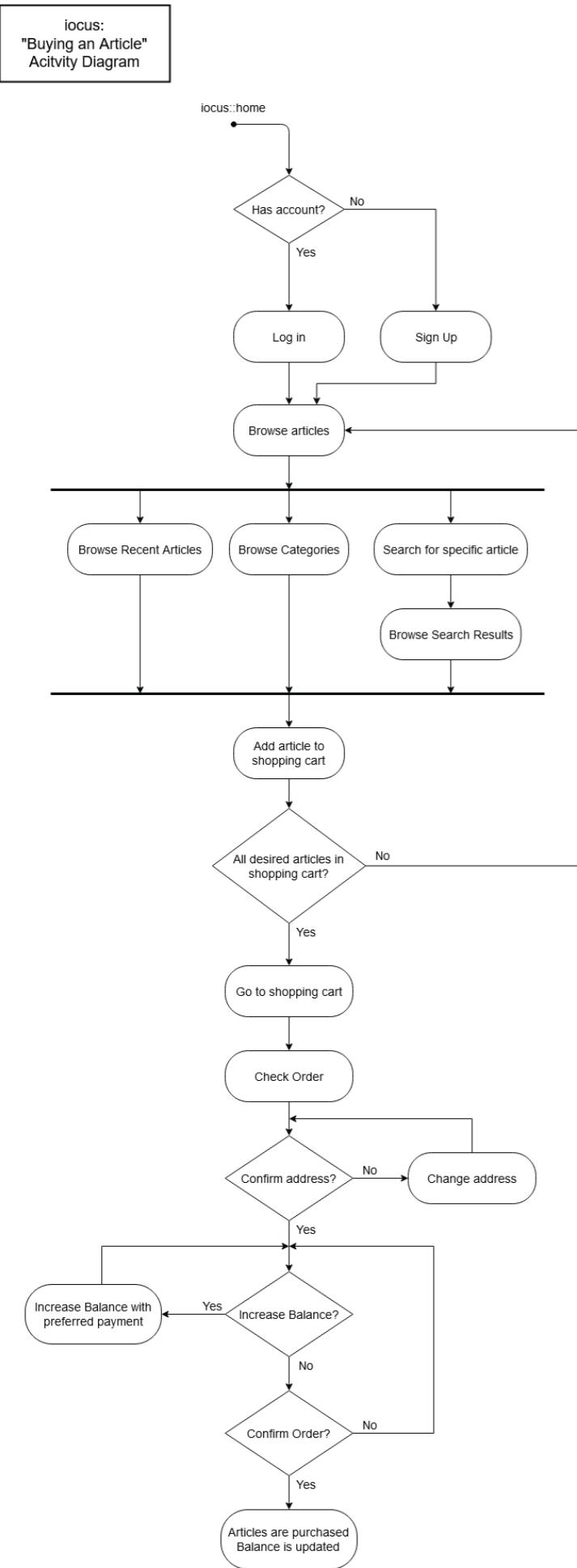
Shopping Cart Service	
<ul style="list-style-type: none"> Able to save articles added by users able to fetch each users shopping cart Able to process buying 	• Shopping cart

CRC card class diagram

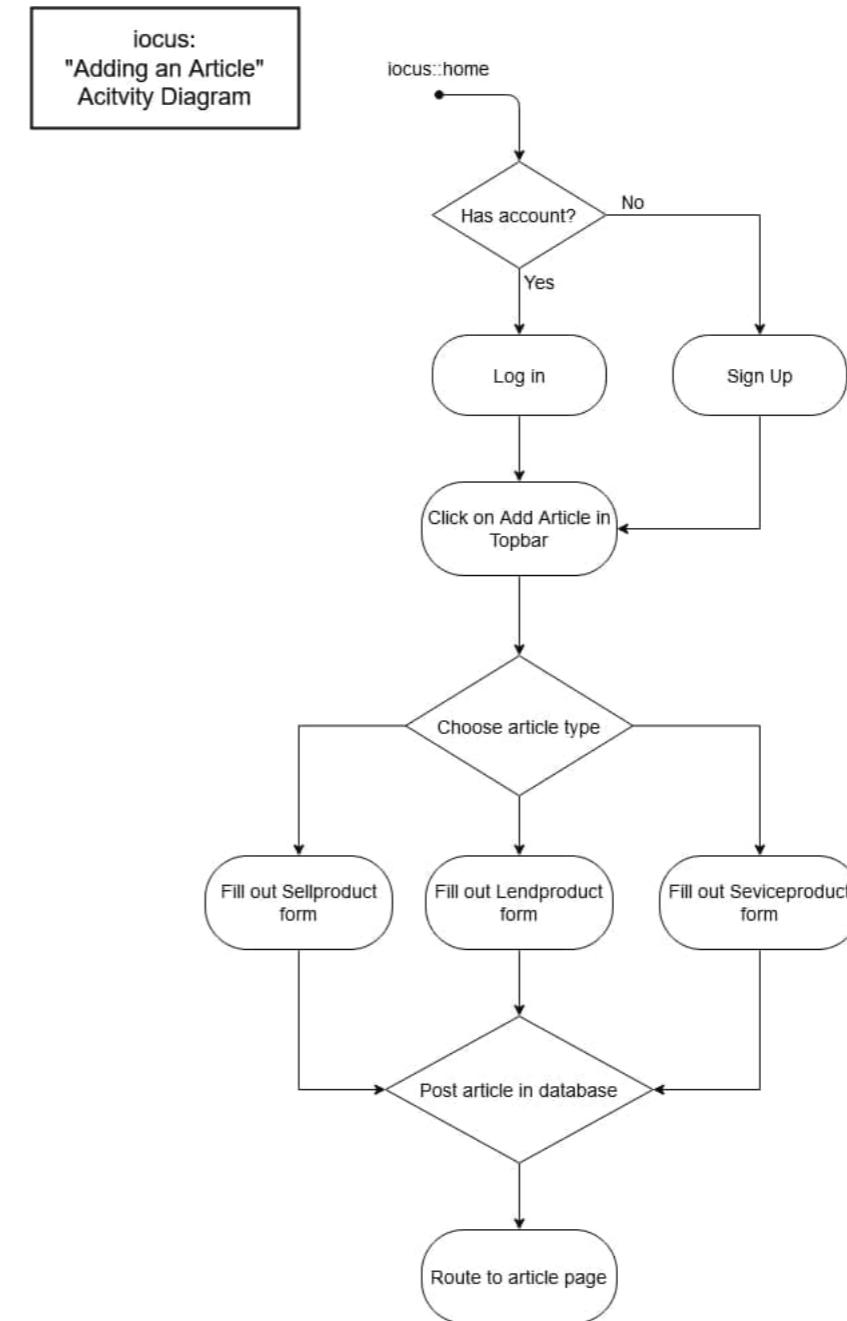


Activity diagrams

Add Article diagram



Buy Article diagram



Mockups

Mockup Mainpage v1

On the main page there is a top-navigation bar, where there is a logo of the website, and then there are two input fields, where you can login with your credentials. There is also a signup button and if you click onto it, you get redirected to /signup

If you are logged in, the top nav-bar changes and you don't see the input fields anymore, but your username and a logout button. If you click on your username, you get redirected to the /profile page.

A screenshot of the mainpage v1. It features a blue header bar with a black logo on the left. To the right of the logo are three input fields: 'username', 'password', and a purple 'login' button. To the right of the login button is a white 'sign up' button.

Mockup /profile logged in

If you're logged in and you go on your profile page, you'll see a page like this, where you can change your personal data. You can also add a profile picture and look up all your items which are already sold and are still selling. You can change these things and save the changes.

A hand-drawn sketch of the profile page. At the top, there's a blue header bar with a 'Logo' icon, an 'upload article' button, a 'username' dropdown, and a 'log out' button. Below the header, there's a large empty box labeled 'add profilepic.' In the center, there are several input fields with edit links: 'username' (with 'edit' link), 'password' (with 'show' and 'change password' links), 'address' (with 'edit' link), and 'email' (with 'edit' link). To the right of these fields is a note: 'Shows the users that are online for sale at the moment. Clicking on the items will lead to the store where you can buy the item. That's also how the user can remove his items'. Below the address field is a 'Bio:' section containing placeholder text: 'Here, the user can write about him, his work etc...'. To the right of the bio is a 'My items:' section with a list: '1. Item', '2. Item', and a '...' button. At the bottom right is a 'save changes' button.

Mockup /Signup

On the signup page, there is an input form, where most of the fields are required to fill out if you want to sign up and create an account. The password has a minimum strength which is required.

A screenshot of the signup page. It features a blue header bar. Below it is a form titled 'Sign up' with the following fields: Name, Surname, Username, E-Mail, Street, House number, City, Postal code, Password, and Repeat password. Each field has a small red asterisk indicating it is required. At the bottom is a white 'sign up' button.

Mockup add-article

If you're logged in, you're able to add articles. Here, you can choose three different options. Once sell article, once lend article and also provide a service. Depending on what you choose, you have different input fields.

A hand-drawn sketch of the add-article page. At the top, there's a blue header bar with a 'Logo' icon, a search bar, and a 'username' dropdown with a menu: 'Add article', 'Shopping cart', 'Balance: 100', and 'Logout'. Below the header is a 'Banner' section. At the bottom are three buttons: 'Sell a product', 'Lend a product', and 'Provide a service'. To the right of these buttons is a 'Selling a Product' form with the following fields: Title*, Price (CHF)*, Category*, Description*, Location*, Delivery*, and Delivery specification*. At the bottom right is a 'Add Product' button.

Logo search

username

-
-
-
-

Banner

Sell a product **Lend a product** **Provide a service**

Lending a Product

Title *

Price (CHF) *

Price *

Category *

Description *

Location *

Status *

Handling *

Add Product

Logo search

username

-
-
-
-

Banner

Sell a product **Lend a product** **Provide a service**

Providing a Service

Title *

Price (CHF) *

Price interval *

Category *

Description *

Location *

Expenses *

Expenses Cost (CHF) *

Add Service

Mockup mainpage v2

On the main page, the most recent products should be displayed as well as a choice of different categories.

Logo

Kategorien durchsuchen

Bekiebte Kategorien

Dienste durchsuchen

Bekiebte Dienste

Lending a product? locus!

Most recent:

Shiba Inu. Very Friend!

Buy for 12CHF

Remember

Shiba Inu. Very Friend!

Buy for 12CHF

Remember

Shiba Inu. Very Friend!

Buy for 12CHF

Remember

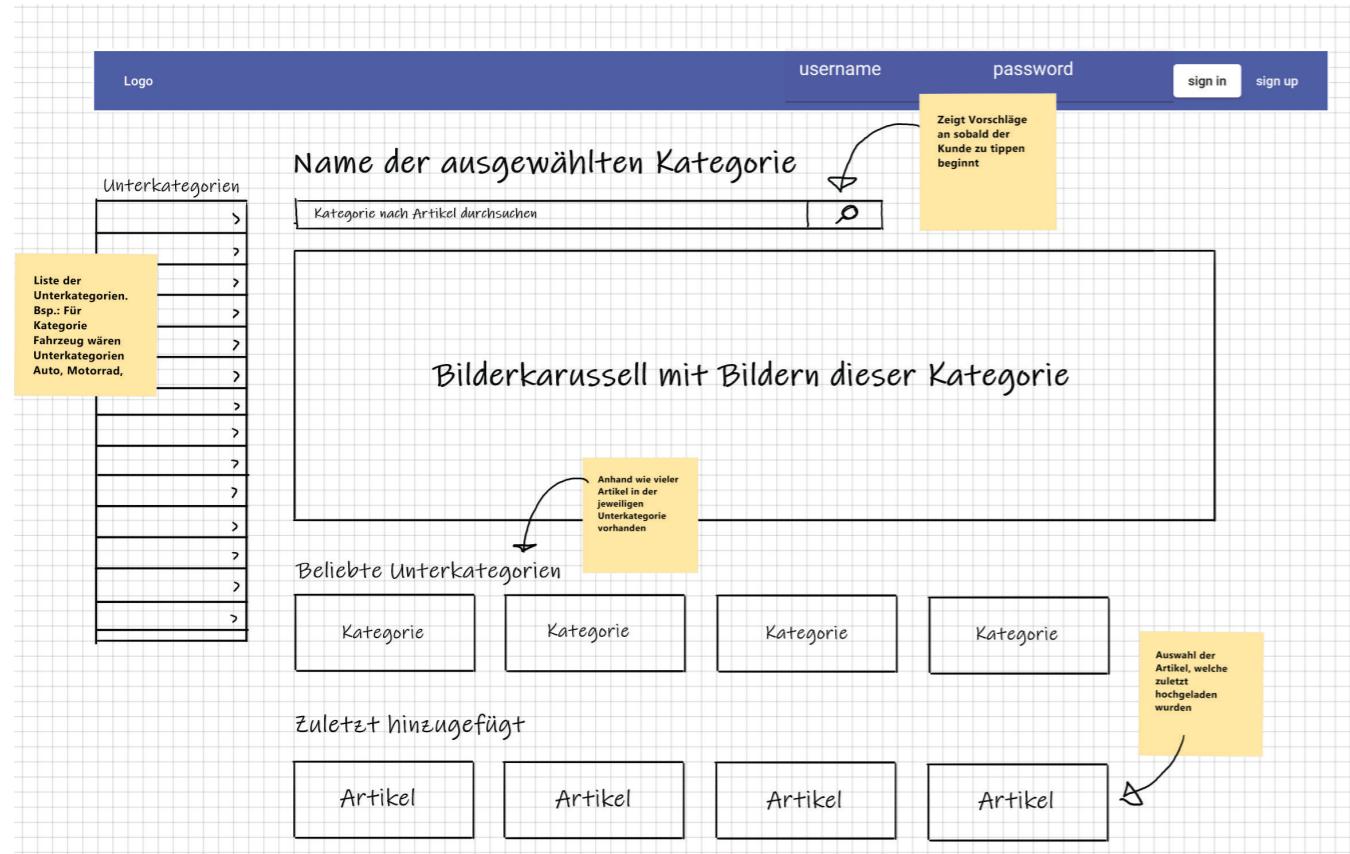
Entdecken Sie auf locus! angebotene Dienste!

Hier können die neusten Produkte beliebiger Kategorien angezeigt werden

Hier können die neusten Dienste angezeigt werden

Hier werden beliebte Dienste und beliebte Artikelkategorien angezeigt. Sie können ebenfalls durchsucht werden und in den jeweiligen Dropdown Möglichkeiten können alle verfügbaren Dienste/Kategorien angezeigt werden

Mockup categories

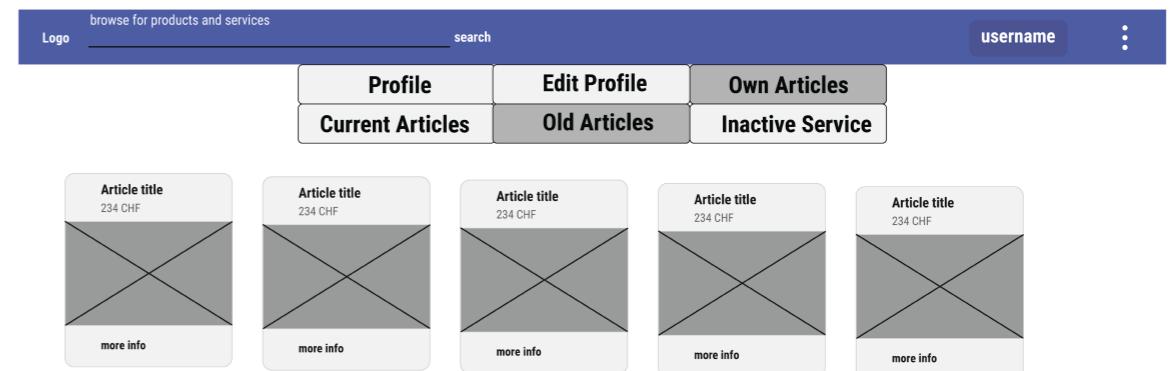
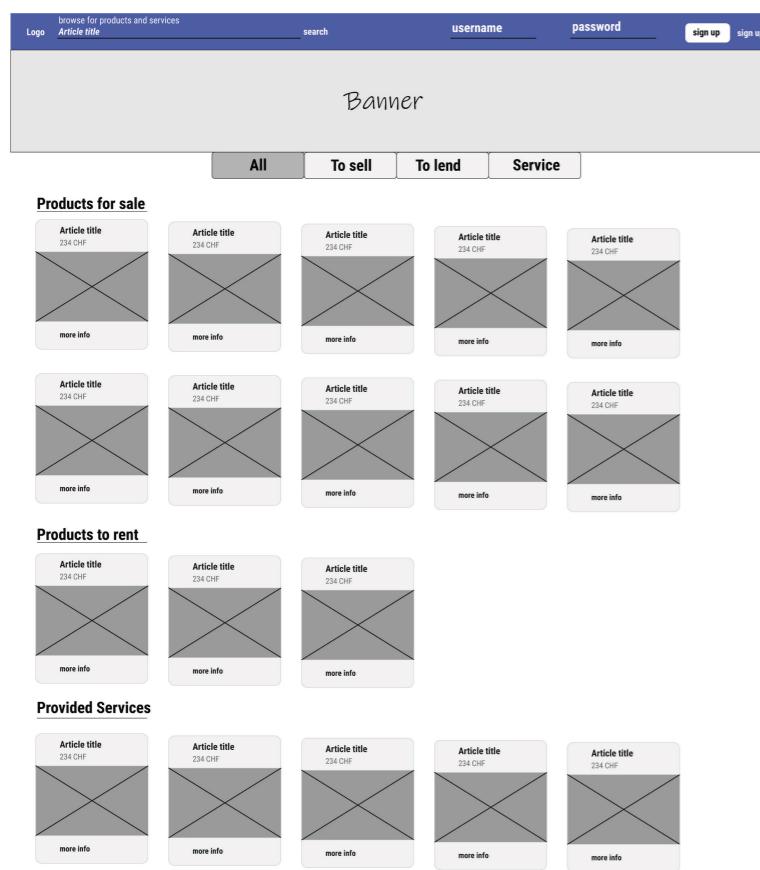


Mockup Profile v2

The profile page should have three different subpages, choosable by clicking on the different tabs. One should display your own profile, one should enable you to change your profile information and one should show you all your articles.

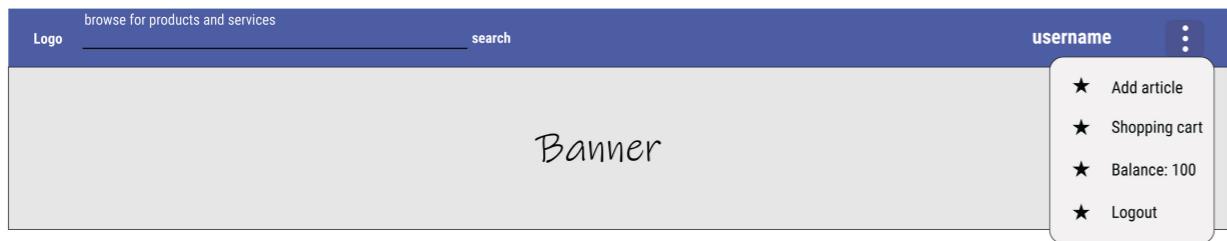
Mockup Search results

Normally, all the search results get displayed, but if you go to a separate tab, only the articles from this separate page get displayed.



Mockup Topbar logged in

Once you're logged in, the topbar changes. Your username is shown and it has a dropdown menu, where you can go to different pages only accessible if you're logged in.



Mockup Balance

To handle all the financial things, there should be a balance page, where you can add money to your account.



Handle your iocus! balance

Choose from where to withdraw money

bank account post account paypal account

How much do you want to add to your balance: (CHF)

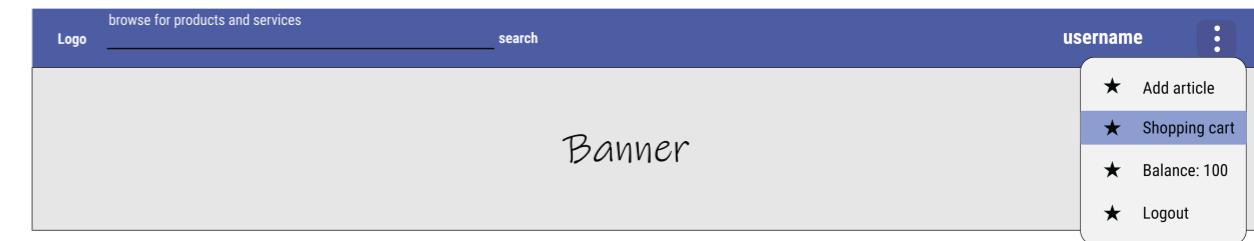
Amount to add

accept [terms of service](#)

Add Money

Mockup Shopping cart

To buy multiple articles, there is a shopping cart. You can add as many articles as you like to the shopping cart.



Articles in your shopping cart

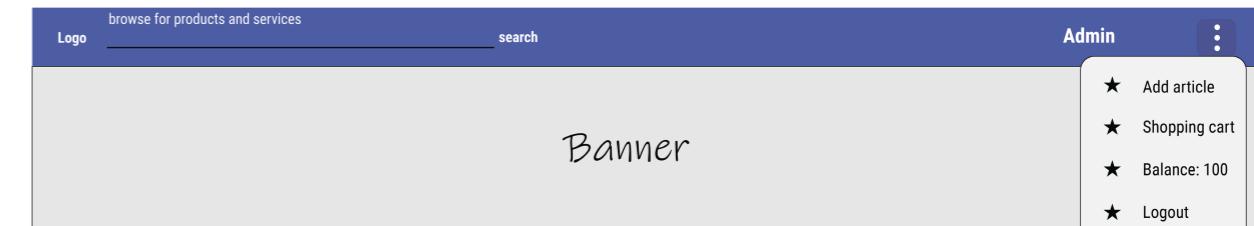
Article title 234 CHF	more info	Delete
Article title 234 CHF	more info	Delete
Article title 234 CHF	more info	Delete
Article title 234 CHF	more info	Delete
Article title 234 CHF	more info	Delete

Total amount:
1'170 CHF

Buy articles

Mockup Admin Panel

If you're logged in as an admin, you have an option in the topbar dropdown, to go to the admin panel. There you can see all users and also all of their products. You can delete users / articles.



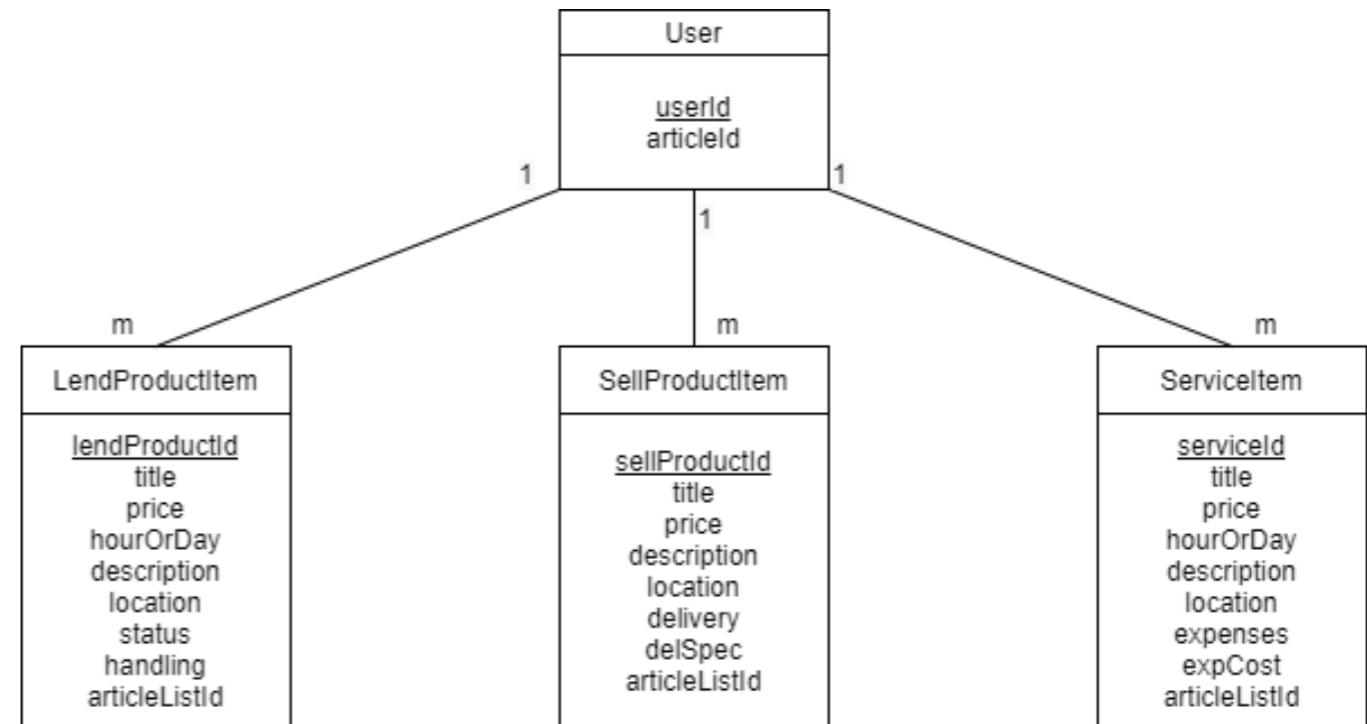
Users

username	show articles	delete user
hansmueller	show articles	delete user
peter233	show articles	delete user
derVerkäufer	show articles	delete user
Sellinator	show articles	delete user
...		

The mockup shows a top navigation bar with a logo, search bar, and a 'User' dropdown menu. Below is a 'Banner' section. The main content area is titled 'Articles of hansmueller' and contains two sections: 'Sell product' and 'Lend product'. Each section lists items like 'Finken', 'Putzlumpen', etc., with 'Show article' and 'Delete Article' buttons. A 'Service provided' section is also present.

Database

Add-Article database



Mockup Article page

If you click on a single article, you get to the articles page, where all the specific things get displayed for the currently chosen article.

The mockup shows a top navigation bar with a logo, search bar, and user authentication fields ('username', 'password', 'sign up'). Below is a 'Banner' section. The main content area includes a 'Popular categories' sidebar with a grid of links for 'electronics', 'clothes', 'hausehold', 'vehicles', 'sport', and 'real estate'. The main right section displays an article with 'Article title' '234 CHF', 'Description' (placeholder), 'Delivery specification' (placeholder), 'Seller and Location' (placeholder), and 'buy' and 'add to cart' buttons.

Retrospective on our project

LIKED – What did the team like?

We really liked how much we learned during the development. The experience of doing a practical project over the course of a whole semester was new for most of us. We had to figure out how to handle deadlines and find enough time to work on the project besides all our other lectures. Sometimes it was stressful, but also very interesting at the same time. And now in the end, we managed to build a whole website with front- and backend, just as we intended to do.

LEARNED – What did the team learn?

We learned how to manage time in a more efficient way, since all of us had other lectures beside ESE. To finish such a project, we had to use our time productively within the short timeframe. We were also just four people in our team, so we had to split up responsibilities and still help each other out. We learned, that not only coding takes up much of the time, but also the meetings, planning, and also the organisational or administrative work such as creating documents like mockups, diagrams etc.. We were confronted with HTML/CSS/Javascript/Typescript, which most of us didn't know well. We also had to learn Angular, Express and Bootstrap from scratch. With the help of the lecture we learned about all kinds of UML diagrams, we then created specifically for our project, so the learning effect was quite nice through the direct usage of the learned theory in practice. Our ability of troubleshooting and finding our way in solving different problems by efficiently using a limited amount of resources also increased. Since the whole semester was held remotely and we didn't see most of our teammates in person, we learned a lot about the importance of communication and how to work in such a project largely remotely and via zoom.

LACKED – What did the team lack?

We sometimes struggled with finding the right resources to help us solve problems with the frameworks in use. It was also not always easy to find enough time to work on the project, since other lectures took up a lot of time too. Nonetheless we weren't that advanced in web development technologies, so we needed to figure it out quite quickly. What clearly lacked were the meetings in person, where all the members stood in front of a whiteboard and could clearly see and discuss what needed to be done. Meetings via zoom isn't quite the same as meetings in person. Through the remote aspect of the whole project, we sometimes also lost a bit the overview over the small and bigger picture of the project.

LONGED FOR – What did the team long for?

We got help from our assistant in coding questions, but just like in the meetings with the team members, the meetings with the assistant would've also been more helpful, if they would have been in person. But here is nobody to blame except the virus. We also longed for a better introduction into the whole project. At the start we had quite the struggle, but then we figured it out how to handle the different things by ourselves. For a next time we would use way more unit tests, to automatically test our whole code. The branching of git will also be used in a next project, since we sometimes had to deal with problems, which wouldn't have been an issue if we used branching. The course is really interesting, but the effort we put into the whole project is more than 5 ET, if we consider 1 ECTS to be 25-30h worth of time. So it might be worth considering to reward the students with more ETCS in future semesters, so the effort would be justified a bit more.