

Frontend Requirements

Milestone 4 & 5 Frontend

-Requirements analysis:

- The Landing page should contain clear and concise information about the services provided, a short description and a link for user support as well as an introduction to the project, including a link to a more detailed description of all its versions and functionalities.
- The navbar should be displayed on all pages and should contain links to all services, including a dropdown menu for further navigating user account services. The deckbuilder_archive icon should always route back to the landing page if clicked.
- An 'About Page' should contain all the motivations behind the project, as well as information about the current version, previous versions and future versions.
- The 'Imprint' page will contain legal information about the creators and the services provided.
- The card archive services will contain a searchbar and several filter options for the different properties of database objects, to avoid confusion, when browsing database entries.
- Card objects will be displayed in a separate container and users will be able to switch to a detailed view of the object by clicking the card item created for each database entry.
- The deck archive service will also provide a searchbar and several options to filter between decks of different formats.
- The deck objects will be displayed in a separate container and clicking them will show the full contents of the list, with card names, mana values and quantity in separate mainboard, sideboard and maybeboard containers in correspondence with their database values. Their creator, format and total number of cards in each container will also be displayed, either next to their respective containers or at the top of the page, in case of the format and the username.
- The login page will simply consist of two input fields, one for the email address of the user and the other for their corresponding password, as well as a login button. Logging in will disable the login option and will replace it by the logout option. The account creation option will be replaced by the account settings option and the create deck option will be unlocked.
- The account creation page will contain three input fields for email address, password and a confirmation field for the password. The submit button will only unlock once the checkboxes for the TOS and data security agreement have been checked. These will also contain links to the 'Imprint' Page, so that users can read up on the terms of service and the data security agreement.
- The logout button will simply remove the user token from the current user and will unlock the option to login again, will replace the account settings option with the account creation option and will lock the create deck option.
- The user account service will provide a sidebar that switches between account settings and currently owned decklists. While the account settings option contains the abilities to change the username or the user's password, the decklists option will display all currently owned decklists, that were created by the currently logged in user. If no user is logged in, the option for account settings will not be displayed.
- The deck creation service will only be available to a user who is currently logged into an account and will link to the deck creation page. This will contain radio buttons for a chosen format as well as an input field for a deckname. A submit button will light up as soon as a format has been selected and a name has been

entered. Clicking the submit button will not only create a new database entry, but also link to the sandbox service.

- The sandbox service can also be accessed through the user account service→decklists option, by clicking on one of the item cards created for a deck owned by the currently logged in user. Once the sandbox page is up, it will display three containers, one for mainboard, one for sideboard and one for maybeboard. These will display the total quantity of all objects currently inside each of their respective containers. A sidebar will contain a searchbar and a container, that is used to display card objects as a part of the search result. The searchbar will only allow the card name search option from the card archive service and card objects created based on the search result will have a button next to their entry that allows them to add that card to either of the three ‘...board’-containers. Once clicked a dropdown option will appear when clicking the add button to make adding the desired object to the desired container simpler. Once a card object has been added to the desired container, the card object will display its mana value, its full name, the current quantity and two buttons. An add-button that will increment the quantity by one and a remove-button that will decrease the quantity of the card by one. Should a card object inside one of the ‘...board’-containers reach 0 it is automatically removed from their respective container. A third container is placed at the top of the sandbox page, right below the navbar, that will contain information about the deck, in particular, the deckname and format, and a submit/save button to update/save the contents of the current decklist in the database.

-Planning MS 4:

◦Landing Page:

1. Setup index.js with Bootstrap, pinia, and vue-Router.
2. Setup vue Router with Landing Page, Imprint and About Us Page.
3. Create a navbar inside the App.vue component.
4. Add a logo to the navbar.
5. Add links for Imprint and About Us to the navbar.
6. Create content for Landing Page in a separate component: card archives, deck archives, deckbuilder and about the project, including a call to contact us, right near the bottom of the Landing Page Component.
7. Create a footer for the App.vue Component that holds contact information.
8. Include the <RouterView/> component between the navbar and the footer of the App.vue Component.
9. Create endcaps for the Landing Page and Imprint.

◦About Us Page:

1. Create the following contents for the About Us Page:
 - What is the deckbuilder_archive?
 - Why is it useful?
 - Why did we create it?
 - What is the current version capable of?
 - What were previous versions capable of?
 - What can be expected of future versions?

°Imprint:

1. Create Disclaimer.
2. Create Terms of Service.
3. Data Privacy Agreement.
4. Create contact data.

°Card Archives:

1. Extend the navbar with the Card Archives option.
2. Create a new Card Archive Component and include it in the Vue Router.
3. Create a searchbar with two input fields. One for text the other as a radio button, with the options api.
4. Create a pinia Card Archive Store that communicates with the backend.

°Login:

1. Create a component that is included in the Vou-Router.
2. Create a User Account Store that communicates with the backend.
3. Setup Firebase for communicating user tokens.
4. Extend the navbar with a dropdown menu, that currently includes login and logout.
5. Extend the App.vue component so it differentiates between possessing a user token and possessing no user token.
6. Add Input fields to the Login component.

-Planning MS 5:

°Deck Archives:

→ Use a similar approach to the Card Archives.

°Account Creation:

1. Extend the Vue Router and the navbar with the Account Creation component.
2. Extend the User Account store with the account creation service.
3. Create input fields and checkboxes, that have links.
4. Submit routes to the Landing Page component.
5. Clicking a link routes to the Imprint component.

°User Account:

1. Extend the vue Router and the navbar.
2. Create a new Deckbuilder store.
3. Extend the User Account store with options for changing personal data.
4. Extend the Deckbuilder store with search options for owned database objects.

°Deck Creator:

1. Create and include a Deck Creator component into the Vue Router and the navbar, also as a part of the User Account dropdown option.
2. Extend the Deckbuilder store with create as well as add and remove functions.
3. Extend the Deckbuilder store with options for saving and deleting user created content.

<A comprehensive list of all User Stories can be found in the Kanban Board as well as in the deckbuilder_archive_vue_frontend_US.pdf file>