



Benefits

In this lesson, we'll discuss the benefits of frontend integration.

We'll cover the following



- Loose coupling
- Logic and UI in one microservice
- Free choice of frontend technologies

Frontend integration offers a number of benefits which make the approach attractive.

Loose coupling

Integration in the frontend creates loose coupling. For example, if links are used for integration, only the URL has to be known for integration to occur. What is behind the URL and how the information is displayed doesn't matter and can be changed without impacting other frontends. So, a change can be limited to one frontend, even if the page looks completely different.

Logic and UI in one microservice

This is advantageous from the architecture's point of view. All logic for a

certain functionality is implemented in a single microservice. For example, a microservice can be responsible for maintaining and



displaying a to-do list, even if the list is displayed as integrated in the UI of another microservice. If you want to display additional information in the to-do list, such as a priority, you can implement the logic, persistence, and UI by changing only one microservice, even if another microservice integrates the rendered to-do list.

Free choice of frontend technologies

Another plus for frontend integration are the frontend technologies. Frontend technologies especially are subject to many innovations; there are constantly new JavaScript frameworks and new ways to create beautiful and easy-to-use interfaces being made.

An important advantage of microservices is the freedom of technology. Each microservice can choose its own technologies. If technology freedom should also apply to the frontend, then **each microservice must have its own frontend that can use its own technology.**

For this, **the frontends must be integrated accordingly.** In particular, care must be taken to ensure that integration does not restrict the use of frontend technologies.

For example, if the integration forces a certain JavaScript library, this can limit the technology selection as it is not possible to use another version of this library in parallel.

For example, the client-side integration in chapter 4 (<https://www.educative.io/collection/page/10370001/5441945024331776/4860469534785536>) requires that each frontend uses a certain jQuery version or provides its own custom code.



QUIZ

Z

1 Frontend integration **always** leads to loose coupling _____. (need author's opinion here)

☐ A) True

☐ B) False

Submit Answer



Question 1 of 2
0 attempted



Reset Quiz ↺

In the next lesson, we'll discuss some variations to the approaches we've already discussed.

← Back

Next →

Challenges



Mark as Completed



Report an Issue