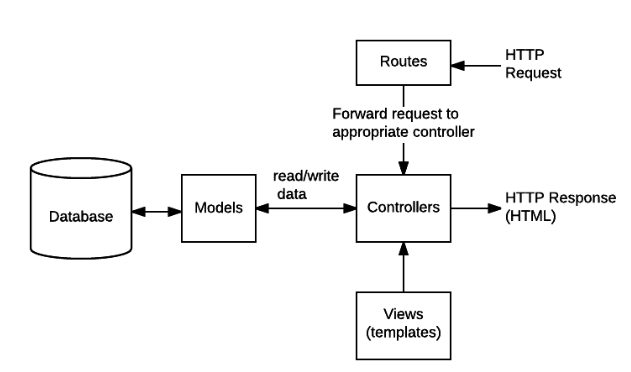
Describe how you would handle updates of the given Data origin



explain my flow on this chart.

Testing agenda: what are the most important parts that need tests

**Testing agenda**

When coming to test my service, I came into a conclusion that the testing needs to be in 2 main aspects – **validity** of the data, and his **correctness**, Why?

At first, the main thinking is that this service will might be part of a bigger application, hence, the validity is top concern, to make sure that if it will become part of bigger application, there won’t be any situation that the service making the the application crash.

This led me to write the first test – the response that being sent for each request. I added test to situation where he is valid, and when there was some error on the way

Afterwards, I added tests that derived from this tests, which will keep the data in the same format all the way through the service flow.

Another main important concern, which is also critical to the service is of course the correctness of the data. Pretty clear why – there is no use to this service if we cant relay on it to behave as we expect it. For this testing, ive added tests that go through the filtering behavior in the app.

An explanation about how you would handle updates  of the given Data origin

The way I managed the data handling, manipulation and reading, is a way in which the data will always be converted into array of matches no matter what the source is.

My purpose was that the data reading will be “black box” for the rest of the service.

The file in charge for this is the DataTransformingHandler, which gets as input from the config file the data origin format, process the data origin, and returns array of matches.

For example, if we decide to change the data origin to remote 3rd party service, the changes that needs to be done are:

1. Change the file type in the config file to http/Json etc
2. Implement function that will get the new data format, parse and manipulate it, and return an array of matches.