

Candidate version

# Supporters - News Feed exercise

## Ruby/Rails Version

## Introduction

### Context:

We want to launch a new feature that lets users get informed about what recently happened. We are building a Petitions newsfeed.

As part of the team, you'll contribute to the delivery of this feature: a Petitions newsfeed.

We are asking you to build the first MVP of this feature.

### About the Newsfeed:

*A newsfeed is a timeline of multiple events that happened and that you might be interested in.*

Our Newsfeed will display:

- Latest 3 created petitions
- Latest petitions that created from your network (petition starters you follow)

### Guidelines:

You will use the bootstrapped Rails 6.5.3 [available here](#).

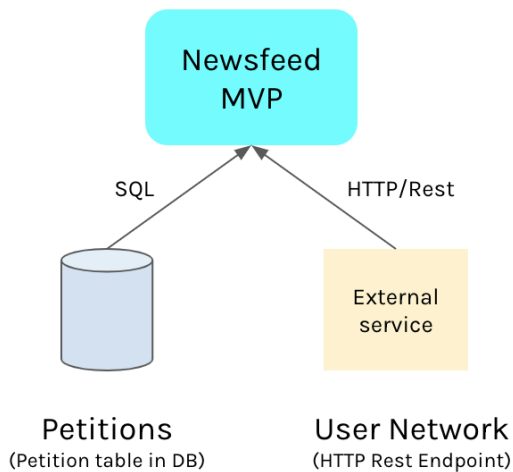
You can use any IDE.

You can use any browser / you can connect to the internet and browse for documentation.

Please share your screen (IDE, command line, browser)

### General architecture:

You will compose a NewsFeed by fetching data from different sources.



Your *Newsfeed MVP* will connect to those two systems:

- **Petitions:** Store all petition information
  - This is a table within a database (SQLite in this exercise) - table name: Petition
- **User Network service:** Serve data about your network (petitions starter you follow).
  - This service is a REST endpoint

## Exercise

### **Sprint 1: Build a newsfeed that displays the 3 most recently created petitions from the past week**

- Network is out of scope here, you can display any petitions.
- Display the most recent ones from the past week
- Order by creation date desc

The database has already been created and seeded with some data. The table/model to use is: Petition

### **Sprint 2: Improve your newsfeed, make it personalized: Only displays petitions that have been created by users you are following (ie: from your network)**

- Use the *Network Service* endpoint to fetch the network

/!\ The endpoint is currently being developed by another team, and it is not ready yet.

Below is the contract provided by the external team.

**Sprint 3: Good job, your newsfeed has a lot of success, you now get thousands of users using it.** But the newsfeed is getting slower. You are investigating. This is not the DB but the Network Rest Endpoint which is under pressure. It sometimes takes several seconds to retrieve user network information.

You want to ensure your newsfeed can be displayed within 100ms.

**Write / update your code to *fallback* to the first use case (display 5 latest created petitions) in case the external endpoint is too slow.**

## Appendices

### Network Rest Endpoint documentation

#### Network object

Retrieve a list of users followed by a specific user.

#### Root URL

<https://api.change.org> (this is a fictive URL, only for the exercise)

#### REST method

```
GET /v1/network/:id
```

#### Parameter

##### **Id**

An integer representing the user id.

To keep it simple, we will skip any authentication/authorization constraints.

#### Returns

Returns a network object (as JSON) if a valid identifier was provided.

The network object contains:

- `User_id` : The user id passed as a parameter

- `Followed_petition_starters`: an array of URNs (Ids) of followed users.

To keep it simple, we will skip any content negotiation. By default, JSON format is returned.

### In practice

#### *Request*

curl <https://api.change.org/v1/network/1234>

#### *Response*

```
{
  "user_id":1234,
  "followed_petition_starters": [
{"starter_urn":"urn:changeorg:starter:6757e68c-8ac4-11eb-8dcd-0242ac130003"},
{"starter_urn":"urn:changeorg:starter:c6e83ee8-151b-4ac0-8e4a-b0dcc11a7f0d"}
  ]
}
```

In the above example, the user with id `1234` is following two petition starters with URN

`urn:changeorg:starter:6757e68c-8ac4-11eb-8dcd-0242ac13000` and

`urn:changeorg:starter:c6e83ee8-151b-4ac0-8e4a-b0dcc11a7f0d`