

Web Programming

AJAX and Vue.js

Leander Jehl | University of Stavanger

AJAX Recap

Example

🔗 examples/ajax/no_ajax

The screenshot shows a web browser window with the title "Exercise #1: Add JSON file". The address bar displays "127.0.0.1:5000" and the page is zoomed to 120%. The web application has two input fields: "Song name" and "Band name", followed by an "Add Song" button. Below these is a table with two rows of song data. The first row has "Song" and "band" as headers, with "Played 2" and buttons for "play" and "Delete". The second row has "Another song" and "Another band" as headers, with "Played 0" and buttons for "play" and "Delete".

Song	band	Played 2	play	Delete
Another song	Another band	Played 0	play	Delete

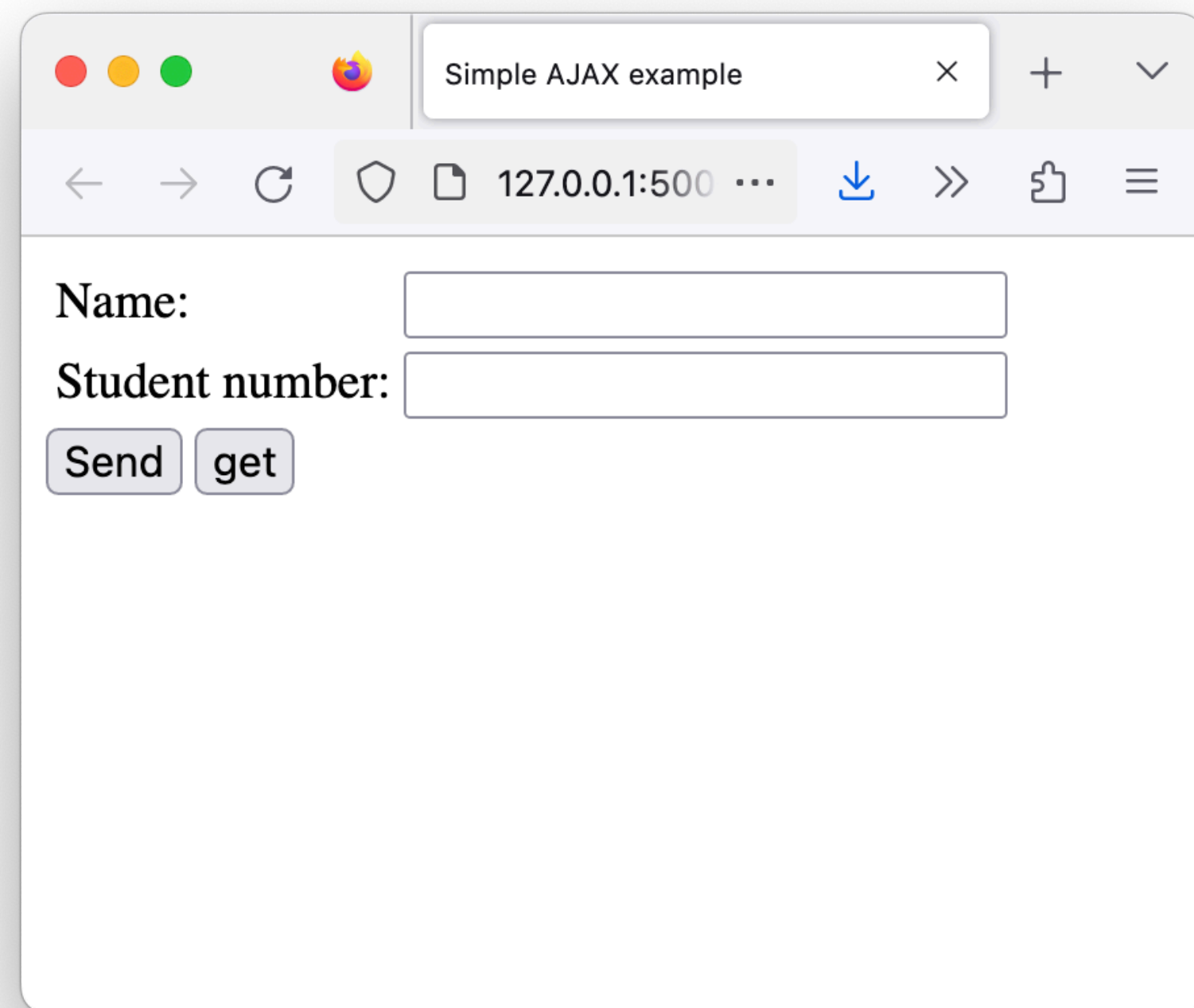
Example



[https://github.com/dat310-2023/info/tree/master/](https://github.com/dat310-2023/info/tree/master/examples/ajax/json/student)
examples/ajax/json/student

Example

🔄 examples/ajax/json/student



A screenshot of a web browser window titled "Simple AJAX example". The address bar shows "127.0.0.1:5000". The page content includes two text input fields: "Name:" and "Student number:". Below these fields are two buttons: "Send" and "get".

static/index.html

```
<input type="button" value="Send" onclick="sendStudent();" />  
<input type="button" value="get" onclick="getStudent();" />
```

Example

🔗 [examples/ajax/json/student](#)

student.js

```
async function getStudent(){  
  let reply = await fetch("/get_data");  
  if (reply.status == 200){  
    let result = await reply.json();  
    ...  
  }
```

Send GET request using fetch

student.js

```
async function sendStudent(){  
  let student = { name: name, student_no: student_no };  
  
  let reply = await fetch("/post_data",{  
    method: "POST",  
    headers: {  
      "Content-Type": "application/json",  
    },  
    body: JSON.stringify(student)  
  });  
  ...  
}
```

Send POST request

Example

🔗 [examples/ajax/json/student](#)

app.py

```
@app.route("/get_data", methods=["GET"])
def check_license():
    DATA = {
        "name": "John Doe",
        "student_no": 111111
    }
    return json.dumps(DATA)
```

Reply with json data.

```
@app.route("/post_data", methods=["POST"])
def print_data():
    print(request.get_json())
    return "OK"
```

Receive json data.

Example

🔗 [examples/ajax/json/student](#)

student.js

```
async function getStudent(){
  let reply = await fetch("/get_data");
  if (reply.status == 200){
    let result = await reply.json();
  }
  ...
}
```

Receive json data.

student.js

```
async function sendStudent(){
  let student = { name: name, student_no: student_no };

  let reply = await fetch("/post_data",{
    method: "POST",
    headers: {
      "Content-Type": "application/json",
    },
    body: JSON.stringify(student)
  });
  ...
}
```

Include JSON data in request

AJAX and Vuejs

- Vuejs and Flask
 - How to combine Vuejs and Flask
- AJAX calls in Components
 - Lifecycle hooks and watchers
 - Pitfalls
- Cross origin requests
- Caching

Lecture example on **course-info**
examples/ajax/vue/lecture

Flask and Vuejs

- Serve JS files and index.html from /static folder.

```
@app.route("/")
def index():
    return app.send_static_file("index.html")
```

- Include /static/ for resources in index.html

```
<!-- vuejs development version -->
<script src="https://unpkg.com/vue@3.0.5/dist/vue.global.js"></script>

<script src="/static/mycomponent.js"></script>
<link rel="stylesheet" href="/cstatic/style.css">
```

Flask and Vuejs

- It is not recommended to use jinja templates together with
 - Should `{{ xxx }}` be replaced by jinja or vue?

AJAX and Vuejs

- Use **created** lifecycle hook to load data via AJAX
- Perform AJAX calls in **methods**
- Load data in components using **watch**

AJAX and Vuejs

- Use **created** lifecycle hook to load data via AJAX

```
let app = Vue.createApp({
  data: function(){
    return {
      playlist: []
    }
  },
  created: function(){
    // function is run after component is created.
    // can access data, methods, ... but not DOM
  },
  methods: {
    ...
  }
});
```

AJAX and Vuejs

- Perform AJAX calls in **methods**
- Pitfalls when performing state updates:
 - AJAX requests do not happen immediately
 - AJAX request may fail
 - AJAX requests may be accidentally send twice
 - Should be idempotent

Example

🔗 [examples/ajax/vue/playlist](#)

Add Song

My favorite
This band



Second favorite
This other band



Exercises #1



[github.com/dat310-2023/info/tree/master/
exercises/ajax/vue](https://github.com/dat310-2023/info/tree/master/exercises/ajax/vue)

AJAX and Vuejs

- Load data in components using **watch**

```
props: ['title'],
  watch: {
    title: function(){
      // executed every time title changes
      this.load();
    }
  },
```

Example

🔗 [examples/ajax/vue/blog](#)

- Retrieve blog text only when opened.

My Story	<h2>My Story</h2> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi at aliquet sapien. Pellentesque bibendum felis quis nulla pulvinar, luctus lobortis turpis venenatis. Phasellus imperdiet neque neque, in condimentum nisi fringilla et. Nam aliquet justo nunc, et semper nunc hendrerit in. Suspendisse potenti. Nam lectus purus, ullamcorper aliquet aliquet eu, malesuada nec nulla. Etiam dui eros, interdum a diam non, tempus vestibulum lorem. Nulla facilisi. Pellentesque et purus non dolor semper eleifend eget eget sem. Sed gravida pharetra varius. Aenean elementum sapien in mi finibus dictum.</p> <p>In dictum ipsum id nisi pretium. a interdum magna blandit.</p>
New Story	
Yesterday	
Tomorrow	

SPA

Single Page Application

- Only loads a single HTML document
- All dynamic content is retrieved using AJAX and displayed using JS
- Can use different routes, e.g. using Vue Router

Flask and Vite decoupled

- It is also possible to run a Vite application independently.
 - I.e. open index.html or run development server, (live server VSCode extension)

- Send AJAX requests using full URL

```
let response = await fetch("http://127.0.0.1:5000/playlist");
```

- Need to enable cross origin requests in Flask:

```
from flask import Flask, request
from flask_cors import CORS
```

```
app = Flask(__name__)
CORS(app)
```

Example

🔗 [examples/ajax/vue/playlist-cors](#)

Add Song

My favorite
This band



Second favorite
This other band



References

- Vue lifecycle hooks
<https://alligator.io/vuejs/component-lifecycle/>
- Cros origin requests
<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>
- Some free loading gifs
<http://ajaxload.info/>