Lecture 10: final steps & static sites

Olivier Liechti
TWEB

heig-vd
Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

Final steps & evaluation

Evaluation of the final project delivery

heig-vd
Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud

- **Demonstration**
  - I will ask each group to make a **10’ demonstration**.
  - You should **use the online version (on heroku) for the demo**, and I should be able to use it afterwards (keep the same version online).
  - You should have a **well defined scenario** to demonstrate the final version of your product.
  - During the demo, I will look at the **UI** (does it look like the original template or did you do some customization), the **UX** (are the features easy to use, is the interface convenient in a class) and the **robustness** of the implementation (are the features working).
- **Landing page / product description**
  - I will evaluate the **content** and the **presentation** of your landing page. You goal should be to convince people to try your product.

Final test

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- I will evaluate the **technical documentation** of your project as the final test.
- The main goal is for you to take a step back and **write up what you have learned** about the various frameworks, libraries and techniques during the project. I would like to see that you have understood how the pieces fit together.
- One particular thing I would like to see is a **sequence diagram**, with clear and complete explanations, which describe the implementation of the slideshow control system (i.e. what happens when the teacher clicks on “next”). You should consider the entire chain, including the REST API layer, socket.io and the persistence layer.
- For the different frameworks, including the build system, I would like you to write a description about what it does, how it is used in your project, what are particular things to take care of. I would like to see **commented code samples**.
- Please **document the issues** that you have encountered during the project and how you have solved them.
- Please write the documentation as **markdown pages** in your GitHub repo.

Planning

- **Monday, January 5th**
  - **Group work on the project**.
  - If you have not defined a clear demo scenario yet, start with that. Make a status check about what is working, not working and what you still need to do in order to finish the implementation.
  - I would suggest that you also (at least) prepare a plan (table of contents) for the technical documentation.
- **Monday, January 12th**
  - **Group work on the project**.
  - In your planning, you should allocate time to write the technical documentation.
  - At the end of the day, your application should work (no more issue) and you should only have to implement “nice to have” features and do fine tuning on the UI and content.
- **Monday, January 19th**
  - I will spend time with each group, to see the demo and ask questions.
  - During that session, you will have time to work on your technical documentation. I will ask you to submit the final version of your technical documentation until Friday, January 23rd at 8 PM.

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

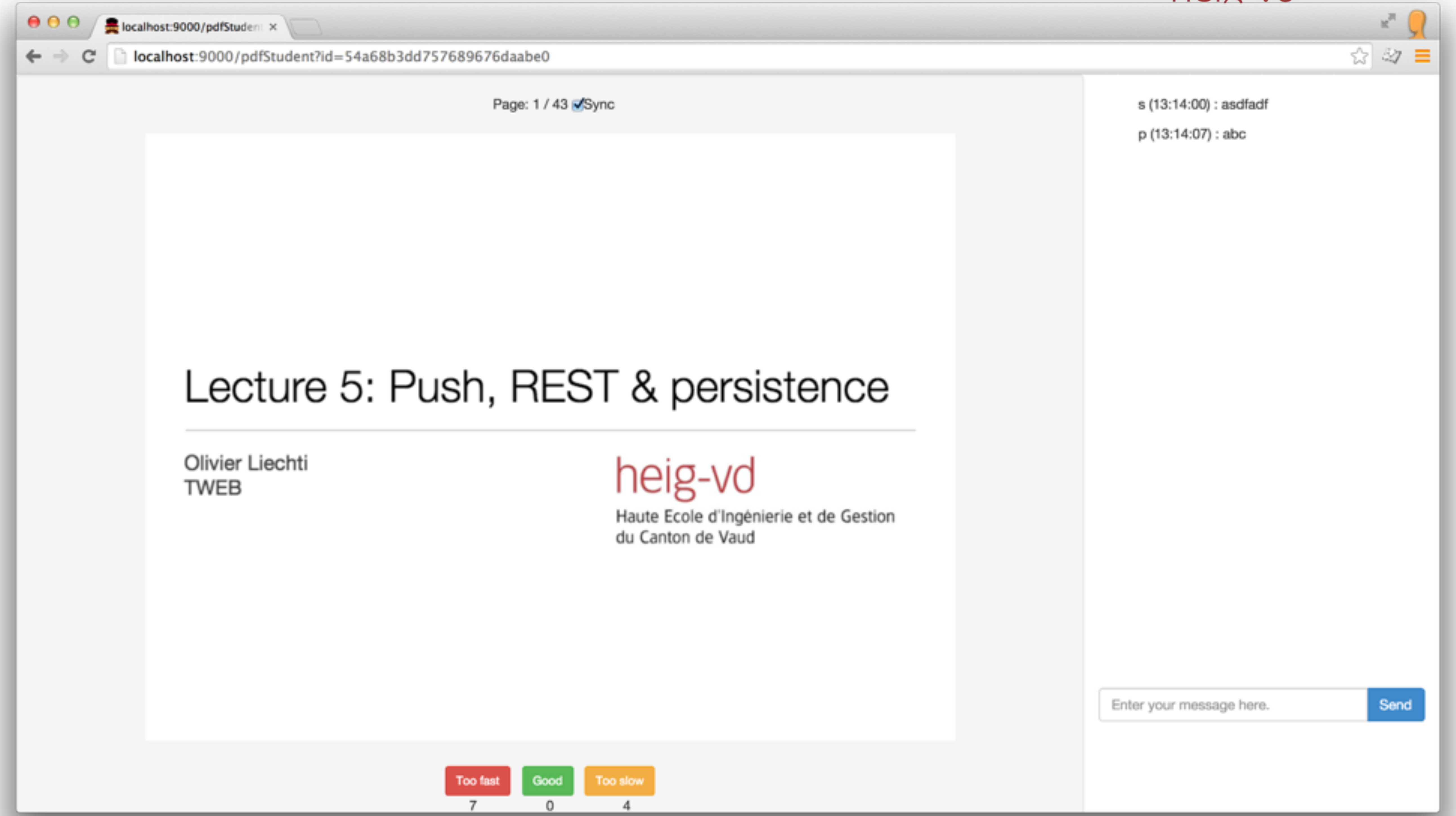
PDF.js / Firefox bug

The issue

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- When I tried your projects, I made the comment to almost every group that the PDF files were rendered correctly in Chrome, but not in Firefox and Safari.
- Finding the root cause of the issue was a bit tricky, but the fix is simple. Here is a summary...

heig-vd


localhost:9000/pdfStudent?id=54a68b3dd757689676daabe0

Page: 1 / 43 ✓Sync

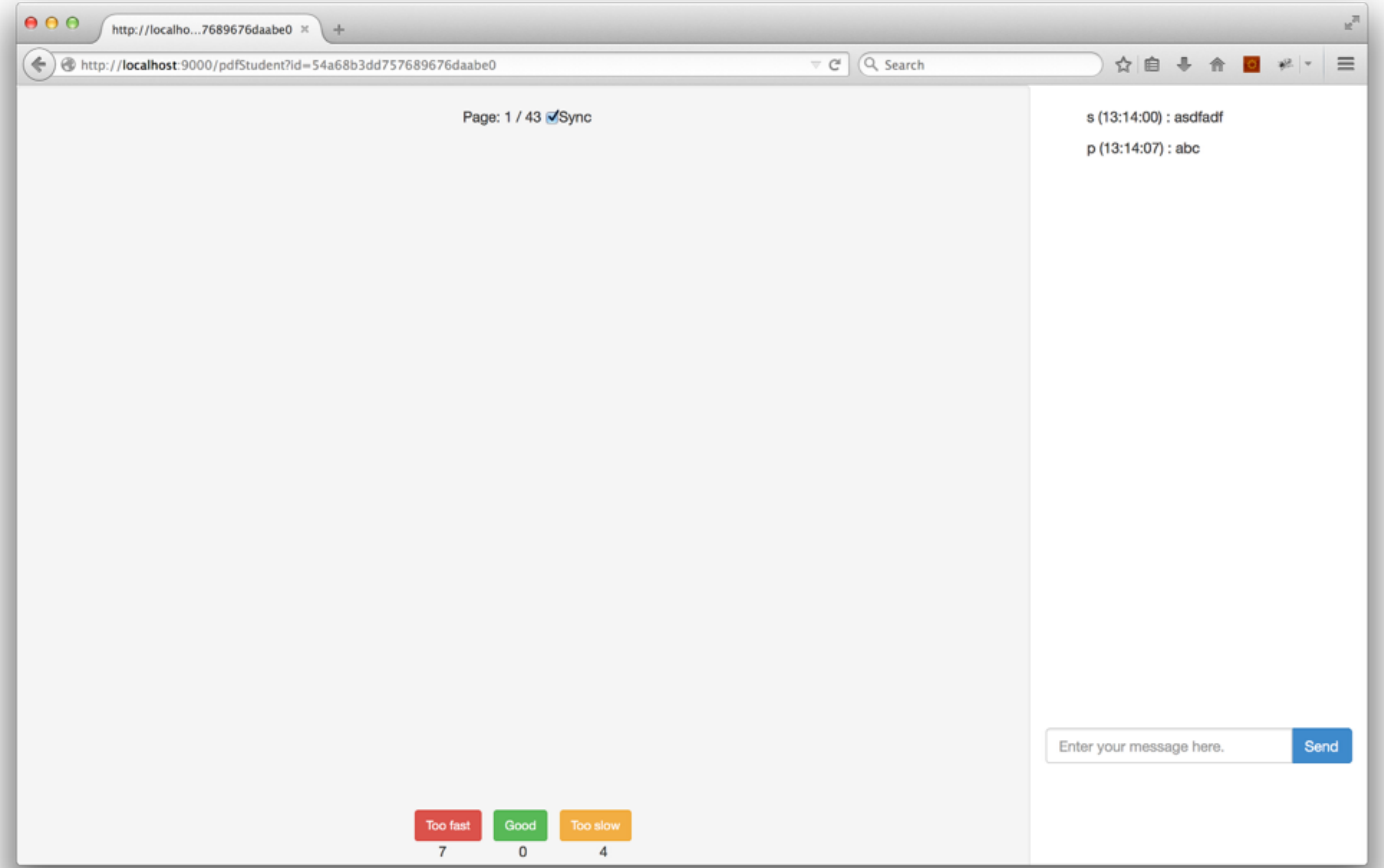
Lecture 5: Push, REST & persistence

Olivier Liechti
TWEB

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

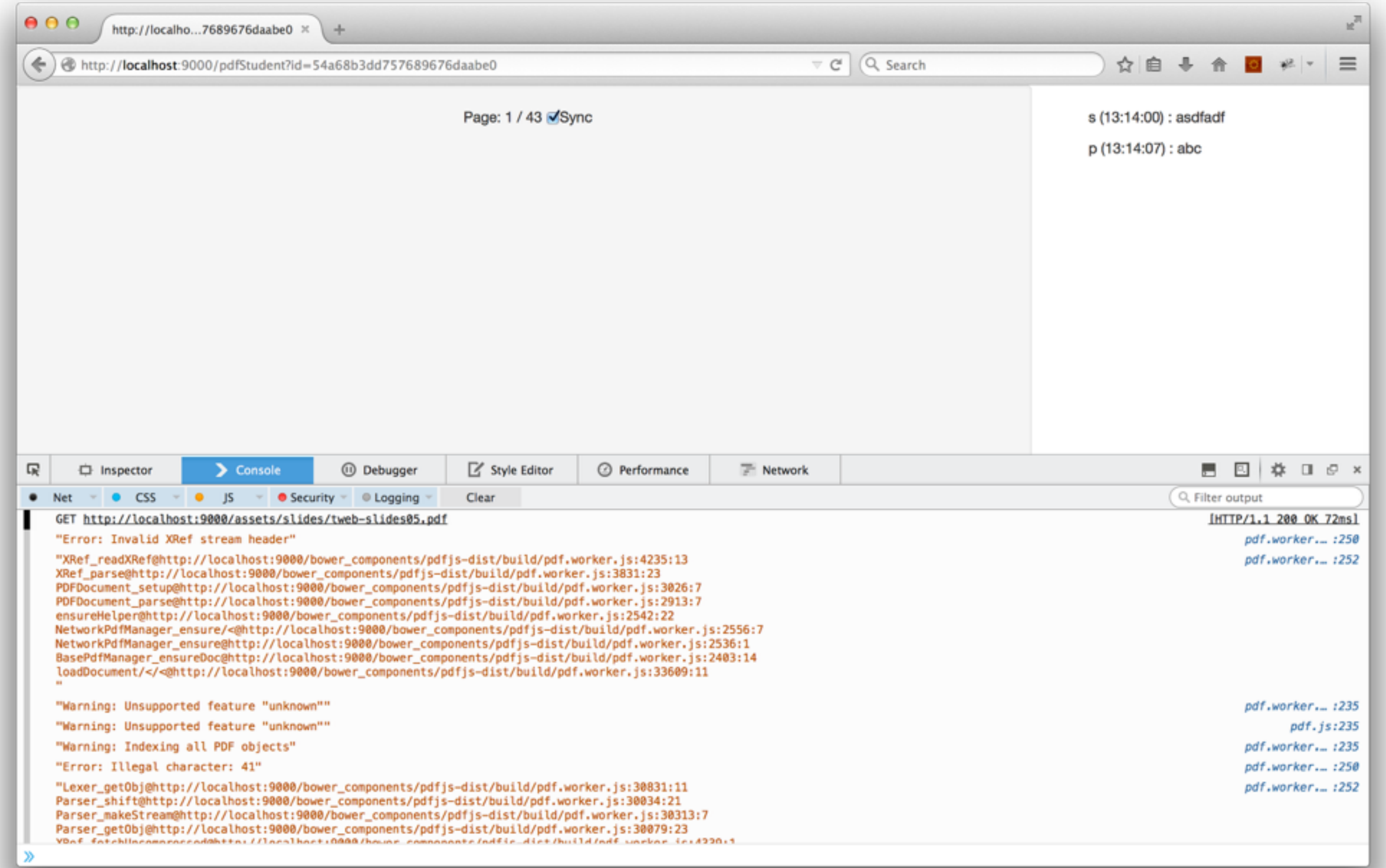
s (13:14:00) : asdfadf

p (13:14:07) : abc


http://localho...7689676daabe0
http://localhost:9000/pdfStudent?id=54a68b3dd757689676daabe0
Search
Page: 1 / 43 ✓Sync
Too fast
Good
Too slow
7
0
4
Enter your message here.
Send

s (13:14:00) : asdfadf

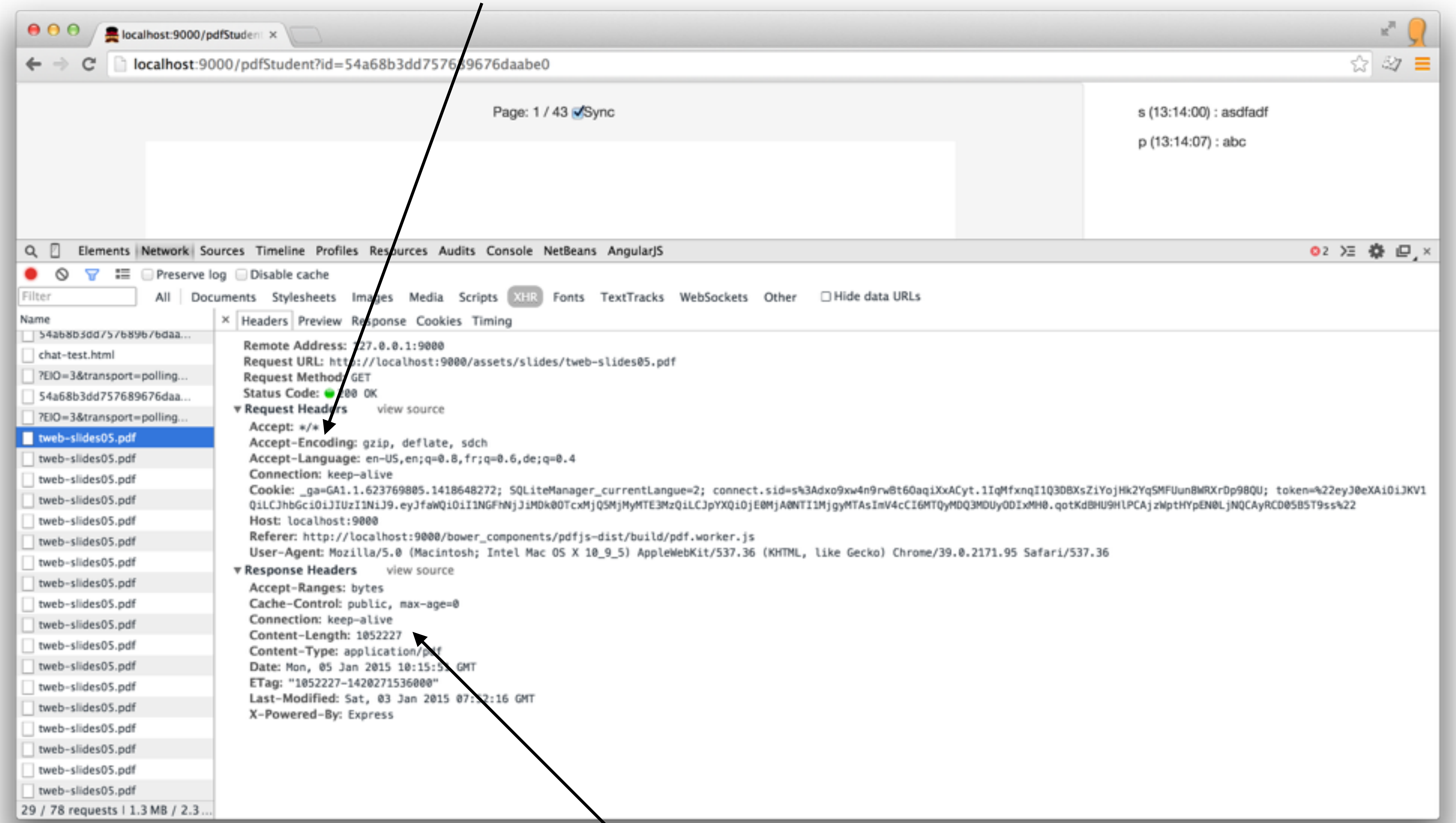
p (13:14:07) : abc



s (13:14:00) : asdfadf

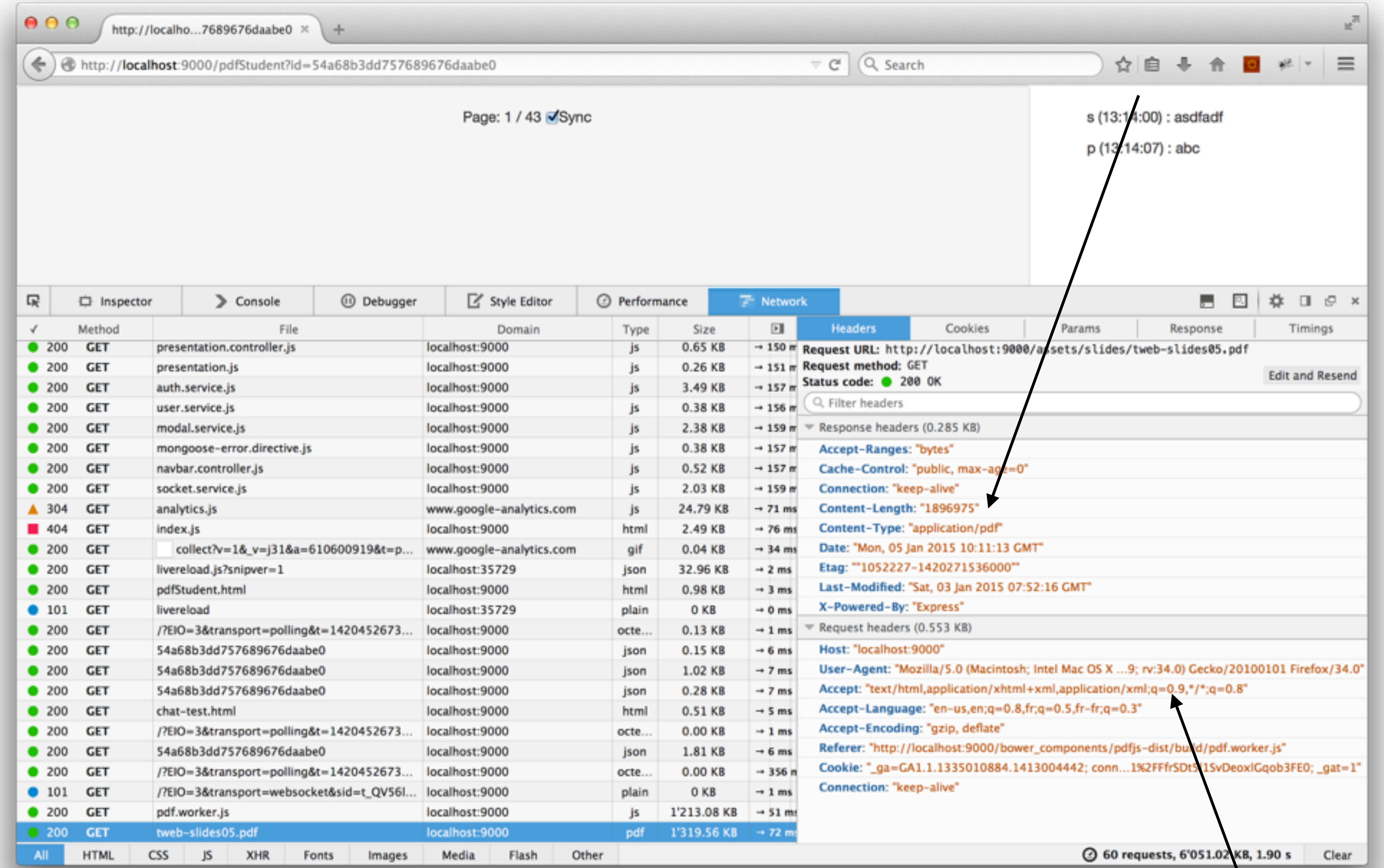
p (13:14:07) : abc

```
GET http://localhost:9000/assets/slides/tweb-slides05.pdf                                   [HTTP/1.1 200 OK 72ms]
"Error: Invalid XRef stream header"                                                          pdf.worker.… :250
"XRef_readXRef@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:4235:13 pdf.worker.… :252
XRef_parse@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:3831:23
PDFDocument_setup@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:3026:7
PDFDocument_parse@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:2913:7
ensureHelper@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:2542:22
NetworkPdfManager_ensure/<@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:2556:7
NetworkPdfManager_ensure@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:2536:1
BasePdfManager_ensureDoc@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:2403:14
loadDocument/</<@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:33609:11
"
"Warning: Unsupported feature "unknown""                                                     pdf.worker.… :235
"Warning: Unsupported feature "unknown""                                                          pdf.js:235
"Warning: Indexing all PDF objects"                                                          pdf.worker.… :235
"Error: Illegal character: 41"                                                               pdf.worker.… :250
"Lexer_getObj@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:30831:11 pdf.worker.… :252
Parser_shift@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:30034:21
Parser_makeStream@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:30313:7
Parser_getObj@http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js:30079:23
```


localhost:9000/pdfStuden
localhost:9000/pdfStudent?id=54a68b3dd757689676daabe0
Page: 1 / 43 ✓Sync
s (13:14:00) : asdfadf
p (13:14:07) : abc
Elements Network Sources Timeline Profiles Resources Audits Console NetBeans AngularJS
Preserve log
Disable cache
Filter
All Documents Stylesheets Images Media Scripts XHR Fonts TextTracks WebSockets Other
Hide data URLs
Name
Headers Preview Response Cookies Timing
29 / 78 requests | 1.3 MB / 2.3...

```
54a68b3dd757689676daa...
chat-test.html
?EIO=3&transport=polling...
54a68b3dd757689676daa...
?EIO=3&transport=polling...
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
tweb-slides05.pdf
```

```
Remote Address: 127.0.0.1:9000
Request URL: http://localhost:9000/assets/slides/tweb-slides05.pdf
Request Method: GET
Status Code: ● 200 OK
▼ Request Headers     view source
  Accept: */*
  Accept-Encoding: gzip, deflate, sdch
  Accept-Language: en-US,en;q=0.8,fr;q=0.6,de;q=0.4
  Connection: keep-alive
  Cookie: _ga=GA1.1.623769805.1418648272; SQLiteManager_currentLangue=2; connect.sid=s%3Adxo9xw4n9rwBt6OaqiXxACyt.1IqMfxnqI1Q3DBXsZiYojHk2YqSMFUunBWRXrDp98QU; token=%22eyJ0eXAiOiJKV1
QiLCJhbGciOiJIUzI1NiJ9.eyJfaWQiOiI1NGFhNjJiMDk0OTcxMjQ5MjMyMTE3MzQiLCJpYXQiOjE0MjA0NTI1MjgyMTAsImV4cCI6MTQyMDQ3MDUyODIxMH0.qotKdBHU9HlPCAjzWptHYpEN0LjNQCAyRCD05B5T9ss%22
  Host: localhost:9000
  Referer: http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js
  User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/39.0.2171.95 Safari/537.36
▼ Response Headers     view source
  Accept-Ranges: bytes
  Cache-Control: public, max-age=0
  Connection: keep-alive
  Content-Length: 1052227
  Content-Type: application/pdf
  Date: Mon, 05 Jan 2015 10:15:5[illegible] GMT
  ETag: "1052227-1420271536000"
  Last-Modified: Sat, 03 Jan 2015 07:52:16 GMT
  X-Powered-By: Express
```


http://localho...7689676daabe0

http://localhost:9000/pdfStudent?id=54a68b3dd757689676daabe0

Search

Page: 1 / 43 ✓Sync

s (13:14:00) : asdfadf

p (13:14:07) : abc

Inspector | Console | Debugger | Style Editor | Performance | Network

✓	Method	File	Domain	Type	Size	
● 200	GET	presentation.controller.js	localhost:9000	js	0.65 KB	→ 150 m
● 200	GET	presentation.js	localhost:9000	js	0.26 KB	→ 151 m
● 200	GET	auth.service.js	localhost:9000	js	3.49 KB	→ 157 m
● 200	GET	user.service.js	localhost:9000	js	0.38 KB	→ 156 m
● 200	GET	modal.service.js	localhost:9000	js	2.38 KB	→ 159 m
● 200	GET	mongoose-error.directive.js	localhost:9000	js	0.38 KB	→ 157 m
● 200	GET	navbar.controller.js	localhost:9000	js	0.52 KB	→ 157 m
● 200	GET	socket.service.js	localhost:9000	js	2.03 KB	→ 159 m
▲ 304	GET	analytics.js	www.google-analytics.com	js	24.79 KB	→ 71 ms
🟥 404	GET	index.js	localhost:9000	html	2.49 KB	→ 76 ms
● 200	GET	collect?v=1&_v=j31&a=610600919&t=p...	www.google-analytics.com	gif	0.04 KB	→ 34 ms
● 200	GET	livereload.js?snipver=1	localhost:35729	json	32.96 KB	→ 2 ms
● 200	GET	pdfStudent.html	localhost:9000	html	0.98 KB	→ 3 ms
● 101	GET	livereload	localhost:35729	plain	0 KB	→ 0 ms
● 200	GET	/?EIO=3&transport=polling&t=1420452673...	localhost:9000	octe...	0.13 KB	→ 1 ms
● 200	GET	54a68b3dd757689676daabe0	localhost:9000	json	0.15 KB	→ 6 ms
● 200	GET	54a68b3dd757689676daabe0	localhost:9000	json	1.02 KB	→ 7 ms
● 200	GET	54a68b3dd757689676daabe0	localhost:9000	json	0.28 KB	→ 7 ms
● 200	GET	chat-test.html	localhost:9000	html	0.51 KB	→ 5 ms
● 200	GET	/?EIO=3&transport=polling&t=1420452673...	localhost:9000	octe...	0.00 KB	→ 1 ms
● 200	GET	54a68b3dd757689676daabe0	localhost:9000	json	1.81 KB	→ 6 ms
● 200	GET	/?EIO=3&transport=polling&t=1420452673...	localhost:9000	octe...	0.00 KB	→ 356 m
● 101	GET	/?EIO=3&transport=websocket&sid=t_QV56l...	localhost:9000	plain	0 KB	→ 1 ms
● 200	GET	pdf.worker.js	localhost:9000	js	1'213.08 KB	→ 51 ms
● 200	GET	tweb-slides05.pdf	localhost:9000	pdf	1'319.56 KB	→ 72 ms

All | HTML | CSS | JS | XHR | Fonts | Images | Media | Flash | Other

Headers | Cookies | Params | Response | Timings

Request URL: http://localhost:9000/assets/slides/tweb-slides05.pdf
Request method: GET
Status code: ● 200 OK

Edit and Resend

Filter headers

Response headers (0.285 KB)

```
Accept-Ranges: "bytes"
Cache-Control: "public, max-age=0"
Connection: "keep-alive"
Content-Length: "1896975"
Content-Type: "application/pdf"
Date: "Mon, 05 Jan 2015 10:11:13 GMT"
Etag: ""1052227-1420271536000""
Last-Modified: "Sat, 03 Jan 2015 07:52:16 GMT"
X-Powered-By: "Express"
```

Request headers (0.553 KB)

```
Host: "localhost:9000"
User-Agent: "Mozilla/5.0 (Macintosh; Intel Mac OS X ...9; rv:34.0) Gecko/20100101 Firefox/34.0"
Accept: "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8"
Accept-Language: "en-us,en;q=0.8,fr;q=0.5,fr-fr;q=0.3"
Accept-Encoding: "gzip, deflate"
Referer: "http://localhost:9000/bower_components/pdfjs-dist/build/pdf.worker.js"
Cookie: "_ga=GA1.1.1335010884.1413004442; conn...1%2FFfrSDt5l1SvDeoxlGqob3FE0; _gat=1"
Connection: "keep-alive"
```

60 requests, 6'051.02 KB, 1.90 s

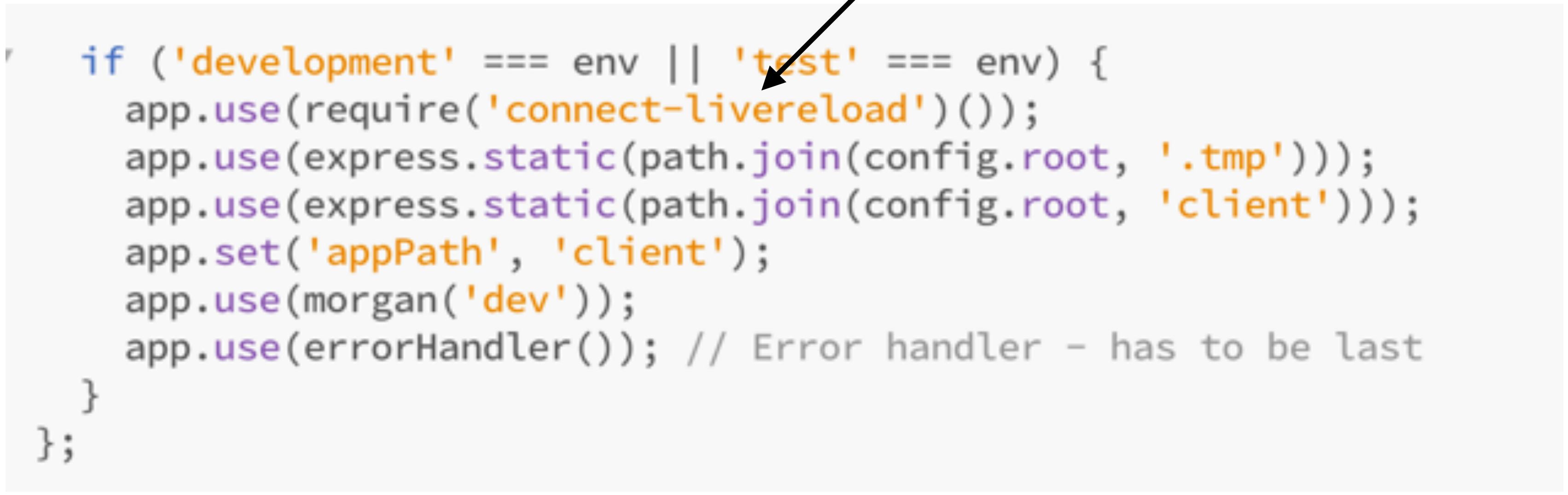
Clear

The issue

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- So... Firefox/Safari do not the same “Accept” header and as a result do not get the same number of bytes back.
- Behavior confirmed by doing a “telnet localhost 9000” on a terminal.
- Something is wrong on the server side, time to dig into **express** and **static-serve middleware**...

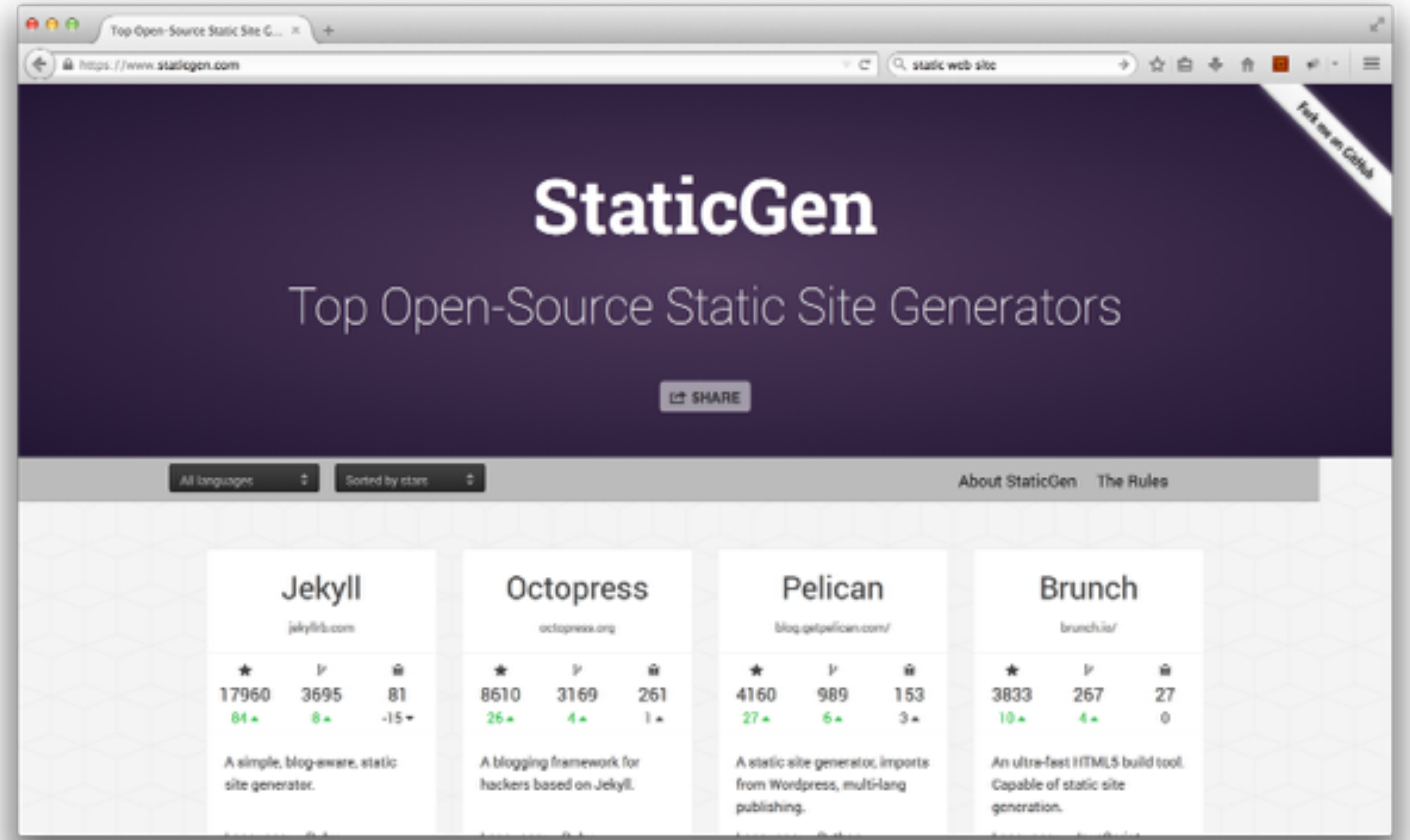
this is the troublemaker!



```
  if ('development' === env || 'test' === env) {
    app.use(require('connect-livereload')());
    app.use(express.static(path.join(config.root, '.tmp')));
    app.use(express.static(path.join(config.root, 'client')));
    app.set('appPath', 'client');
    app.use(morgan('dev'));
    app.use(errorHandler()); // Error handler - has to be last
  }
};
```

server/config/express.js

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud



StaticGen

Top Open-Source Static Site Generators

	★		
Jekyll (jekyllrb.com)	17960 (84▲)	3695 (8▲)	81 (-15▾)
Octopress (octopress.org)	8610 (26▲)	3169 (4▲)	261 (1▲)
Pelican (blog.getpelican.com/)	4160 (27▲)	989 (6▲)	153 (3▲)
Brunch (brunch.io/)	3833 (10▲)	267 (4▲)	27 (0)

Jekyll: A simple, blog-aware, static site generator.

Octopress: A blogging framework for hackers based on Jekyll.

Pelican: A static site generator, imports from Wordpress, multi-lang publishing.

Brunch: An ultra-fast HTML5 build tool. Capable of static site generation.

Static web sites

Static web sites

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- There is a trend in using tools that **generate static html pages** by merging template files and data. The resulting html can be deployed on any web server, without the need for a scripting language or a database.
- It is an **alternative to using a dynamic web site**, where templates and data are merged at runtime.
- I am personally not completely convinced of the approach, although it has merits for some sites. Landing pages, technical documentation and technical blogs are relevant use cases.
- What is really nice is to be able to **write articles in markdown**. Much better that writing code snippets in Wordpress.
- There are lots of tools available (see http://www.staticgen.com). A popular one is named **Jekyll** and what is interesting, is that it is supported by GitHub pages.
- The combination makes it a nice addition to your “web toolbox”.

Quick intro

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

- Installation is easy on Mac OS (you need a ruby environment). Windows is not officially supported, but there is a special page in the documentation. Maybe a linux VM is a safer bet...
- Once you have installed jekyll, you can generate (scaffold) a new site with a single command (jekyll new demo).

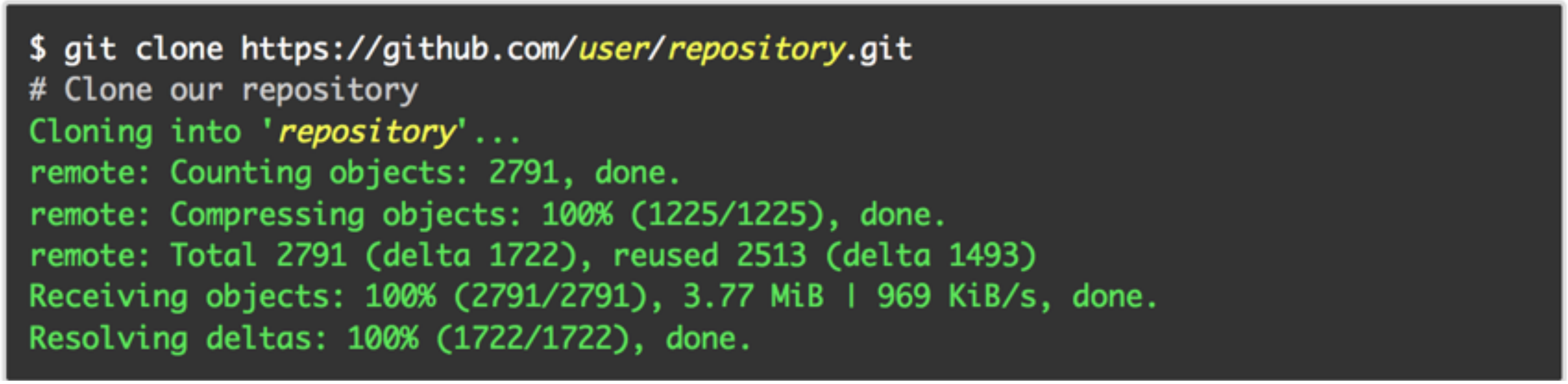
```
.
./demo
./demo/.gitignore
./demo/_config.yml
./demo/_includes
./demo/_includes/footer.html
./demo/_includes/head.html
./demo/_includes/header.html
./demo/_layouts
./demo/_layouts/default.html
./demo/_layouts/page.html
./demo/_layouts/post.html
./demo/_posts
./demo/_posts/2015-01-05-welcome-to-jekyll.markdown
./demo/_sass
./demo/_sass/_base.scss
./demo/_sass/_layout.scss
./demo/_sass/_syntax-highlighting.scss
./demo/about.md
./demo/css
./demo/css/main.scss
./demo/feed.xml
./demo/index.html
```

Quick intro

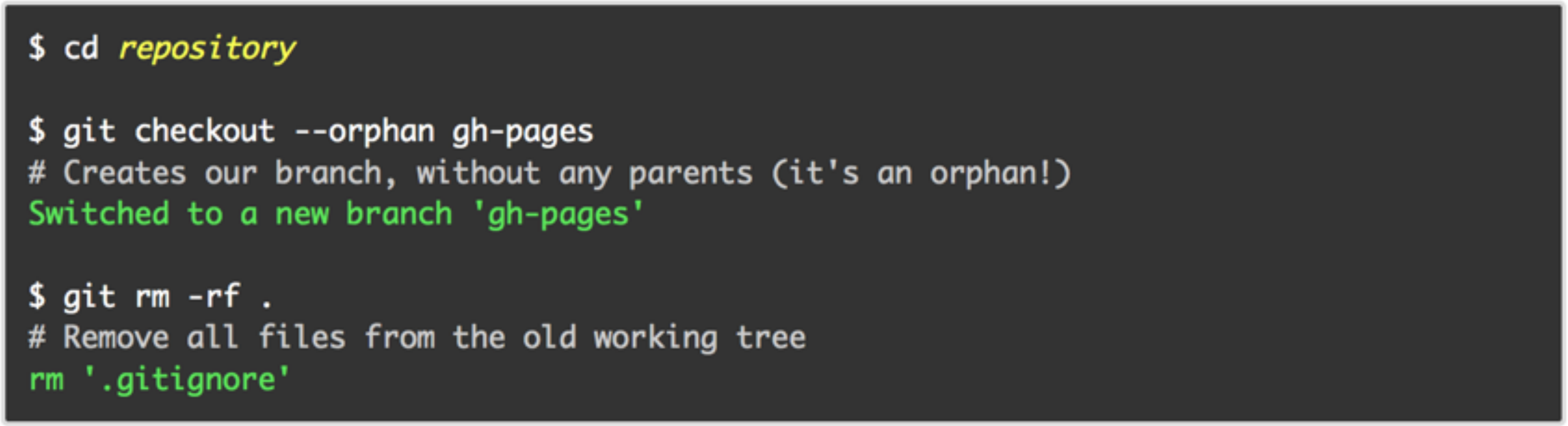
- There is a special markup language for writing the templates.
- Remember that everything is generated at compile time (for instance, if you use pagination to go through the blog posts, there is one html document generated for each page.
- You can use “jekyll build” and “jekyll serve”
- If you use GitHub pages, you can simply push your source files to the “gh-pages” branch. GitHub will generate the static files and serve them.

Quick intro

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud



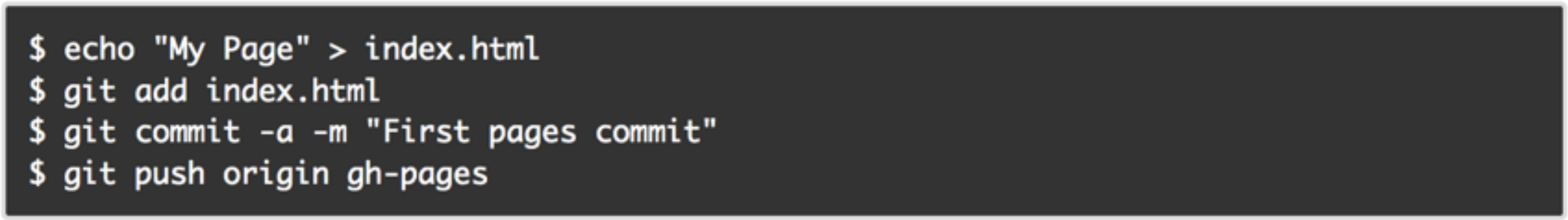
```
$ git clone https://github.com/user/repository.git
# Clone our repository
Cloning into 'repository'...
remote: Counting objects: 2791, done.
remote: Compressing objects: 100% (1225/1225), done.
remote: Total 2791 (delta 1722), reused 2513 (delta 1493)
Receiving objects: 100% (2791/2791), 3.77 MiB | 969 KiB/s, done.
Resolving deltas: 100% (1722/1722), done.
```



```
$ cd repository

$ git checkout --orphan gh-pages
# Creates our branch, without any parents (it's an orphan!)
Switched to a new branch 'gh-pages'

$ git rm -rf .
# Remove all files from the old working tree
rm '.gitignore'
```

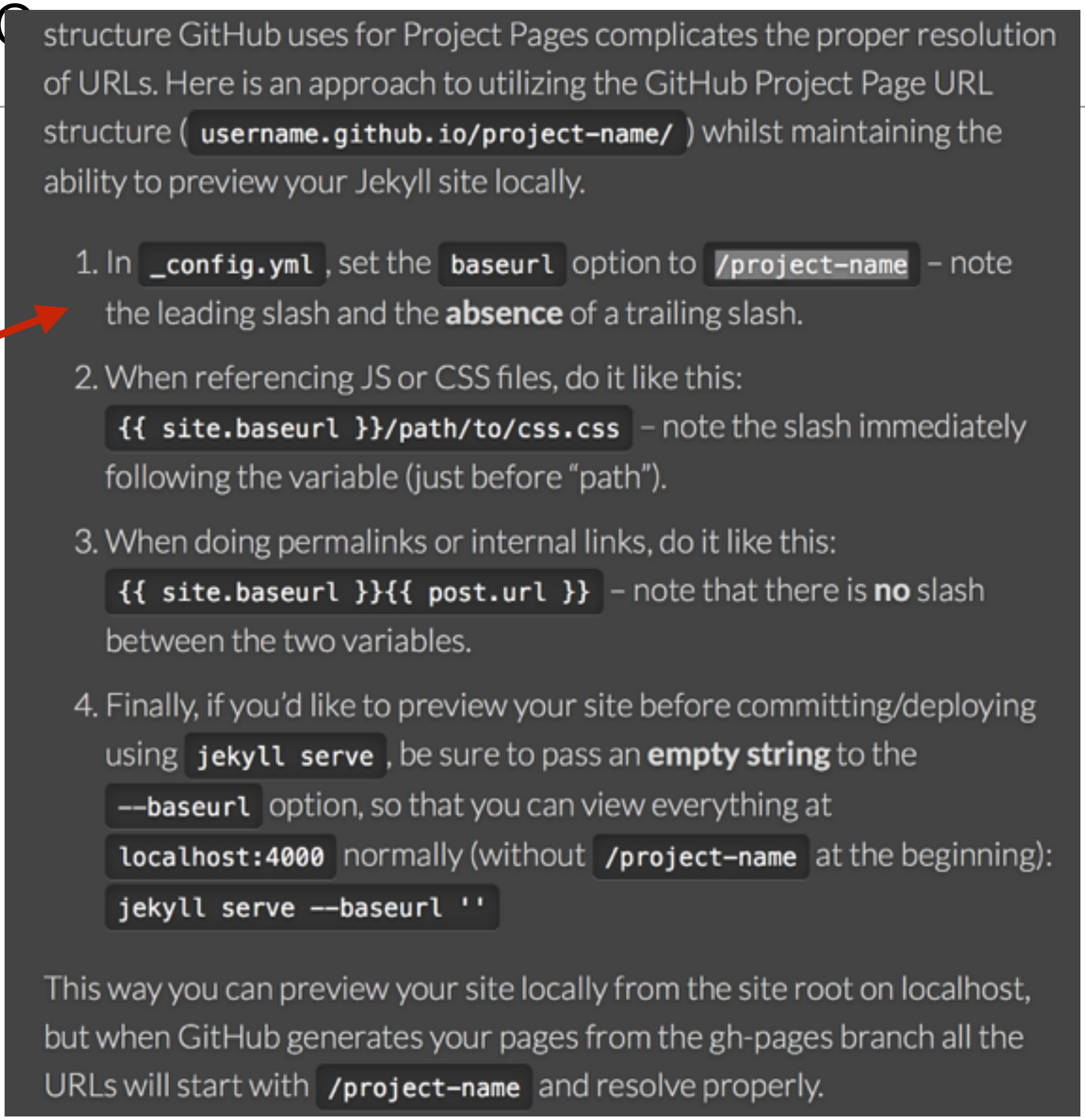


```
$ echo "My Page" > index.html
$ git add index.html
$ git commit -a -m "First pages commit"
$ git push origin gh-pages
```

https://help.github.com/articles/creating-project-pages-manually/

Quick intro

heig-vd
Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud



structure GitHub uses for Project Pages complicates the proper resolution of URLs. Here is an approach to utilizing the GitHub Project Page URL structure (`username.github.io/project-name/`) whilst maintaining the ability to preview your Jekyll site locally.

1. In `_config.yml`, set the `baseurl` option to `/project-name` – note the leading slash and the **absence** of a trailing slash.
2. When referencing JS or CSS files, do it like this: `{{ site.baseurl }}/path/to/css.css` – note the slash immediately following the variable (just before "path").
3. When doing permalinks or internal links, do it like this: `{{ site.baseurl }}{{ post.url }}` – note that there is **no** slash between the two variables.
4. Finally, if you'd like to preview your site before committing/deploying using `jekyll serve`, be sure to pass an **empty string** to the `--baseurl` option, so that you can view everything at `localhost:4000` normally (without `/project-name` at the beginning): `jekyll serve --baseurl ''`

This way you can preview your site locally from the site root on localhost, but when GitHub generates your pages from the gh-pages branch all the URLs will start with `/project-name` and resolve properly.

http://jekyllrb.com/docs/github-pages/