

Reliable

PWAs



Detta verk är licensierad under en
[Creative Commons Erkännande 4.0](https://creativecommons.org/licenses/by/4.0/)
[Internationell licens](https://creativecommons.org/licenses/by/4.0/).

Content

- ✓ Routing
 - State
 - Push State
- ✓ History
- ✓ Offline
 - LieFi

Routing

/artist/654d

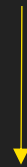
/artist/543dd

/playlist/6542d

/playlist/7653fgf

/album/hgff5s

/album/75635d



Spotify

Hem

Sök

Ditt bibliotek

SPELLISTOR

+ Skapa spellista

♥ Gillade låtar

Scoutdisco

140bpm running

Florida vibes

The Umbrella Academy - ...

🌟 Nyår 2018 Dancef... 🌟

metropolis

Rock'n'Roll Racing

The Great Plains

Hot Country

Classic Road Trip Songs

Roadtrip USA - Classics (6...

Korridorspartaj 🌟

Vi som älskar 90-talet 2016

Imagine Dragons

33 021 198 lyssnare i månaden

FÖLJER

Populära

1		Believer	1 551 074 850	3:24
2		Thunder	1 403 823 743	3:07
3		Bad Liar	535 332 240	4:20

Spotify

Hem

Sök

Ditt bibliotek

SPELLISTOR

+ Skapa spellista

♥ Gillade låtar

Scoutdisco

140bpm running

Florida vibes

The Umbrella Academy - ...

🌟 Nyår 2018 Dancef... 🌟

metropolis

Rock'n'Roll Racing

The Great Plains

Hot Country

Classic Road Trip Songs

Roadtrip USA - Classics (6...

Korridorspartaj 🌟

Vi som älskar 90-talet 2016

SPPELLISTA

The Great Plains

Garrett Forbes • 4 gillar

• 104 låtar, ungefär 6 tim 30 min

TITEL ALBUM

1		Freeze Frame T... Brandon Rhyder, ...	Conviction	4:59
2		Goodbye Girl Whiskey Myers	Road of Life	5:38
3		Ballad Of A So... Whiskey Myers	Firewater	3:40
4		Song For You Whiskey Myers	Firewater	7:47
5		Lonely East TX ... Whiskey Myers	Firewater	4:21

Spotify

Hem

Sök

Ditt bibliotek

SPELLISTOR

+ Skapa spellista

♥ Gillade låtar

Scoutdisco

140bpm running

Florida vibes

The Umbrella Academy - ...

🌟 Nyår 2018 Dancef... 🌟

metropolis

Rock'n'Roll Racing

The Great Plains

Hot Country

Classic Road Trip Songs

Roadtrip USA - Classics (6...

Korridorspartaj 🌟

Vi som älskar 90-talet 2016

ALBUM

Random Access Memories

Daft Punk • 2013

• 13 låtar, 1 tim 14 min

TITEL

1		Give Life Back to Music Daft Punk	4:35
2		The Game of Love Daft Punk	5:22
3		Giorgio by Moroder Daft Punk	9:04
4		Within Daft Punk	3:48
5		Instant Crush (feat. Julian Casablancas) Daft Punk, Julian Casablancas	5:37

Routing - location

<http://www.test.com:8080/index.html?id=1432&name=Tim#anchor>

<code>window.location.hash</code>	<code>// #anchor</code>
<code>window.location.host</code>	<code>// www.test.com</code>
<code>window.location.hostname</code>	<code>// www.test.com:8080</code>
<code>window.location.href</code>	<code>// http://www.test.com:8080/index.html?id=1432&name=Tim#anchor</code>
<code>window.location.pathname</code>	<code>// /index.html</code>
<code>window.location.port</code>	<code>// 8080</code>
<code>window.location.protocol</code>	<code>// http:</code>
<code>window.location.search</code>	<code>// ?id=1432&name=Tim</code>

URL and searchParams

<http://www.test.com:8080/index.html?id=1432&name=Tim#anchor>

```
let params = (new URL(document.location)).searchParams;  
let name = params.get('name'); // is the string "Tim".  
let age = parseInt(params.get('id')); // is the number 1432
```

Fragment identifier routing

```
<a href="#!/account/47435675">Account 1 details</a>  
<a href="#!/account/65443234">Account 2 details</a>  
<a href="#!/account/12675334">Account 3 details</a>
```

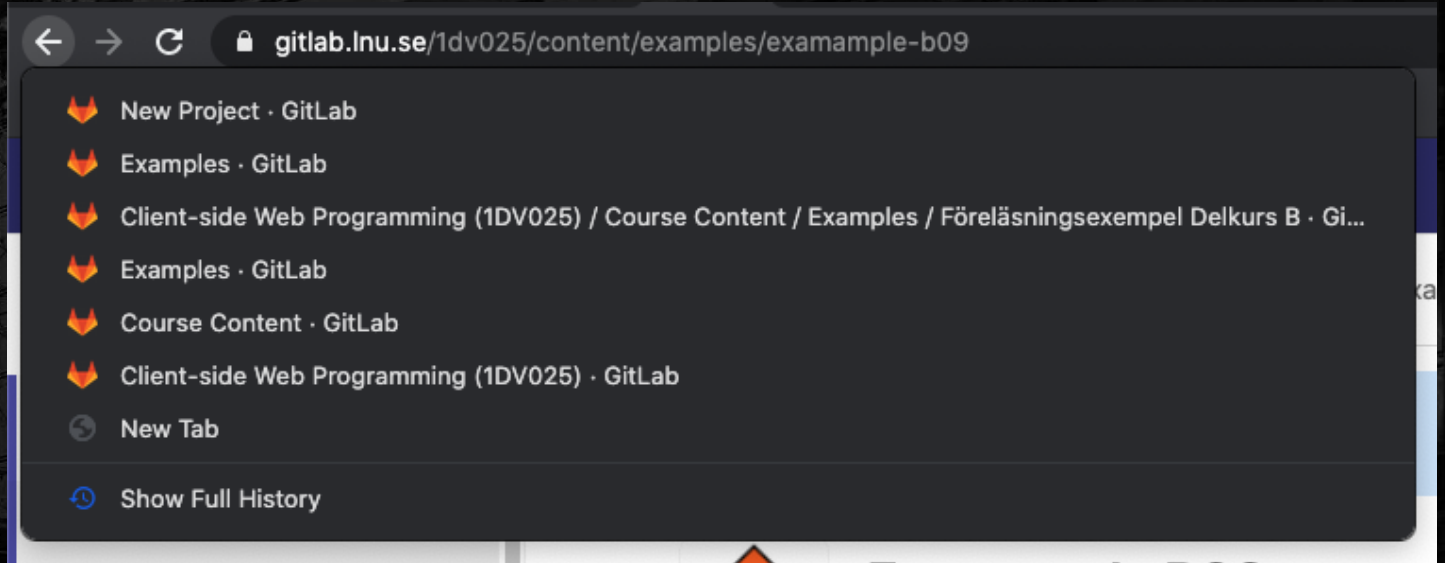
```
// Router  
window.addEventListener('hashchange', event => {  
  let hash = window.location.hash // #!/account/47435675  
  
  if(hash.substr(2,9) === '/account/'){  
    let account = hash.substr(11) //47435675  
  
    // Fetch account information from server  
    // Display account information  
  }  
})
```

History in the browser

✓ window.history

- .back()
- .forward()
- .go(-3)
- .length

window.history.back()



Pushstate

```
<ul id="links">  
  <li><a href="#!/account/47435675" data-at="Account 1" data-a="47435675">Account 1</a></li>  
  <li><a href="#!/account/65443234" data-at="Account 2" data-a="65443234">Account 2</a></li>  
  <li><a href="#!/account/12675334" data-at="Account 3" data-a="12675334">Account 3</a></li>  
</ul>
```

.html

```
document.querySelector('#links').addEventListener('click', event => {  
  event.preventDefault()
```

.js

```
  const stateObj = {  
    accountnbr: event.target.getAttribute('data-account'),  
    account: event.target.getAttribute('data-account-title')  
  }
```

```
  window.history.pushState(stateObj,  
    `Account information: ${stateObj.account}`,  
    `!/account/${stateObj.accountnbr}`)
```

```
  // Load account information and show it
```

```
  console.log(`Push state: ${stateObj.account} with account number ${stateObj.accountnbr}`)  
})
```

Popstate

```
window.addEventListener('popstate', event => {  
  // When an old state is triggered.  
  if (event.state) {  
    console.log(`Pop state: ${event.state.account} with account number ${event.state.accountnbr}`)  
  }  
})
```

Popstate is not called when the user reloads the page!

Read state onload

```
window.addEventListener('load', event => {  
  if (window.history.state) {  
    console.log(`Pop state: ${window.history.state.account} with account number ${window.history.state.accountnbr}`)  
  }  
})
```

This require all routes to be redirected to index.html

In snowpack 3 this can be done by adding:

```
"experiments": {  
  "routes": [  
    {  
      "src": ".*",  
      "dest": "/index.html",  
      "match": "routes"  
    }  
  ]  
}
```

! This must also be replicated in the production environment!

Offline – Problems to solve

- ✓ How to know if the user is online?
- ✓ How to cache resources in the browser?
- ✓ How to inform the user?
- ✓ How to handle data on client?
- ✓ How to sync data to server?
- ✓ How to handle conflicts?

Online / Offline?

```
if(window.navigator.onLine){  
  console.log('You are online! Or.....?')  
}
```

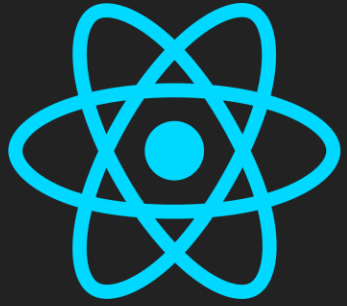
```
windows.addEventListener('online', event =>  
  console.log('Online?'))
```

```
windows.addEventListener('offline', event => {  
  console.log('Offline?')  
})
```

It is not that simple....:(

Lie-Fi

Client Side Libraries and Frameworks



React



