

Web Design
& Development

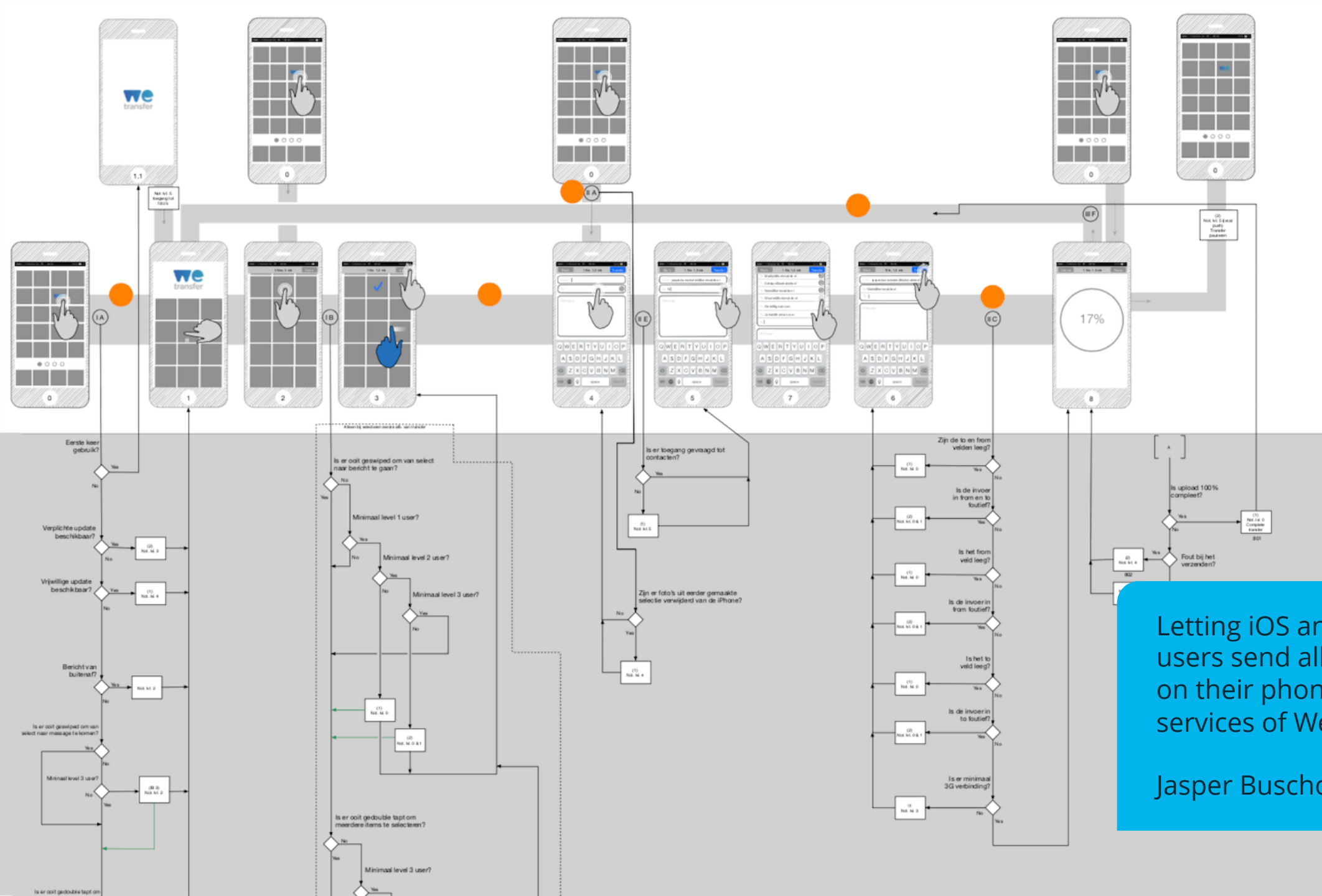
Refactor / Design Code Logics

Planning

1. Activity Diagram

Activity Diagram

When I was working at Blizzard I was making UI flow diagrams and I noticed that a few engineers were hanging the documents on their walls for reference.



Letting iOS and Android users send all the pictures on their phone through the services of WeTransfer.

Jasper Buschoten

Design the code logics of your SPA in an Activity Diagram

1. Sketch all states of your SPA and create a Wireflow
2. Add a swimlane with url/hash for the different states
3. Add a swimlane for the Control Flow, use pseudo code for all modules and methods
4. Add your Activity Diagram to the Readme
5. Code ... code ... code ...

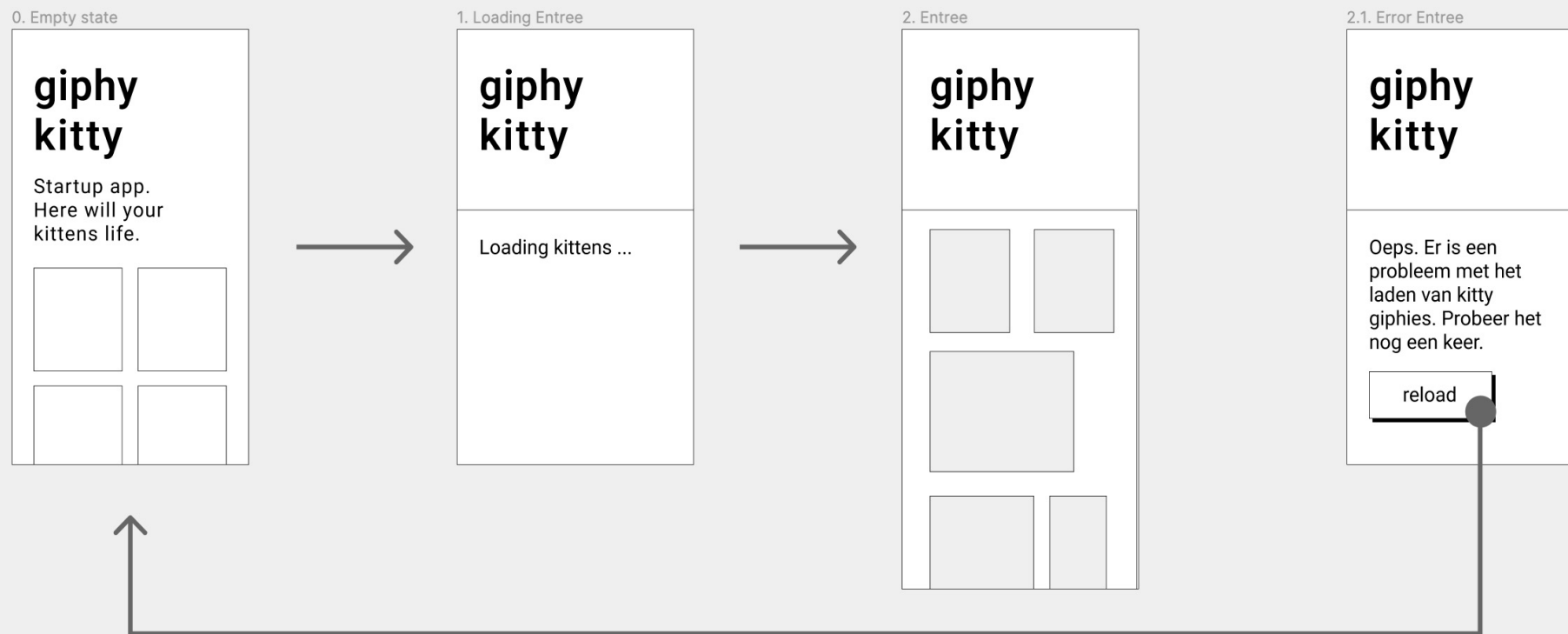
Bespreek dit met een collega

Activity Diagram

Swimlane →

Swimlane →

Swimlane →



URL/URL/Hash

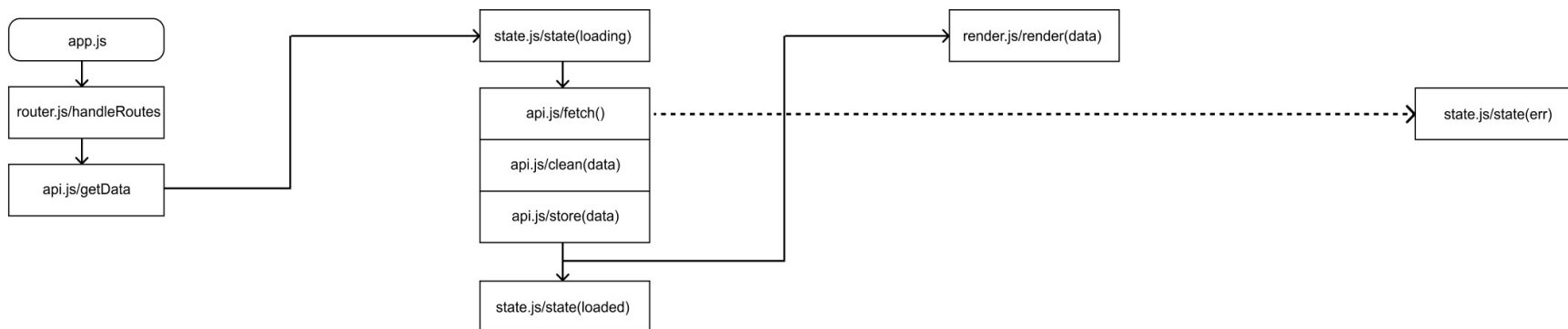
/

#gifs

#gifs

#gifs

Control Flow





EXPLORER



index.html

JS app.js

JS router.js

JS router.js



> OPEN EDITOR... 1 UNSAVED

WEB-A...

static

> css

js

modules

> vendor

JS api.js

JS render.js

JS router.js

JS state.js

JS ui.js

JS app.js

★ favicon.ico

Modules

> spa

> team

> visitekaartje

.gitignore

LICENSE

> OUTLINE

> TIMELINE

examples > modules > static > js > JS app.js

1 import { handleRoutes } from './api.js'

2

3 handleRoutes()

4

examples > modules > static > js > modules > JS router.js > ...

1 import { getData } from './api.js'

2 import { render } from './render.js'

3 import { updateUI } from './ui.js'

4 import './vendor/routie.min.js'

5

6 export function handleRoutes() {

7 routie(

8 {

9 'gifs': () => {

10 getData().then(data => {

11 render(data)

12 updateUI('gifs')

13 });

14 },

15 'gifs/:id': id => {

16 getData(id).then(data => {

17 render(data, id)

18 updateUI('giphy')

19 });

20 },

21 'about': () => {

22 updateUI('about')

23 }

24 })

25 }

26

27

28

