

AEM

AEM compon

History

date

author

1.3.17

stefan
meyer

4.2.17

stefan
meyer

3.11.16 stefan
 meyer

1 Introduc

The goal of the AEM ([A](#) javascript ui library by f

Why React and A

React components are content. This project bri

Features

- [Universal React](#)
- High performance
- Nesting React components
- Converting vanilla JS to React
- SPA based on react

Projects

AEM project

The [aem-react](#) project contains

- osgi bundle container
- maven archetype
- demo content package

Maven artifact is available

Javascript project

The aem-react projects

Maven archetype project

Lastly a [maven-archetype](#)

archetype.

Maven artifact is available

Version

This documentation

Prerequisites

-  Java 8 (Oracle)

-  AEM 6.0

2 Getting S

In this guide we will use

Requirements

- Adobe Experience
- Apache Maven (:

The example react com
use another dialect you

2.1 First proj

To quickly get started w

1. create maven p

```
mvn archetype:genera  
-DarchetypeGroupId=  
-DarchetypeArtifactId=  
-DarchetypeVersion=1
```

Versions

Get the latest [archetype](#)
in [ui.apps/src/main/t](#)

You will then be asked
find detailed explanation

Available properties

groupid

groupid

artifactId

version

package
group

appsFolderName

artifactName

componentGroupName

contentFolderName

cssId

```
packageGroup  
groupId
```

```
siteName
```

2. start AEM

AEM should now be running

3. install demo

```
mvn install -PautoInstall
```

If your AEM instance is

parameter

-Daem.port

-Daem.host

4. Open browser

To check what was dep

- go to the page /c
- find react compo
- find components



NEW PI

Service C

React text comp

React Panel

React text inside

Standard text co

HelloWorldMode
Hello
This is
Resour

Lorem ips

Lorem ipsum de
voluptua. At ver



Lorem ipsum ac
voluptua. At ver

5. disable author

To check that we have

- Disabled the auth
- Install your [react](#)
- Have a look at re



NEW PROJ

Service Compo

React text component

React Panel

React text inside a react

Standard text componen

```
HelloWorldModel says:  
    Hello World!  
    This is instan  
    Resource type
```

Lorem ipsum

Lorem ipsum dolor sit ar
eirmod tempor invidunt l
voluptua. At vero eos et
kasd gubergren, no sea
Lorem ipsum dolor sit ar
eirmod tempor invidunt l
voluptua. At vero eos et
kasd gubergren, no sea

Duis autem vel eum iriur
consequat, vel illum dolc
et iusto odio dignissim q
dolore te feugait nulla fa
adipiscing elit sed diam

There are two *RootCon*
Panel which is a vanilla
another *Text* composer

2.2 First com

This part assumes that

1. Start watch tas

Start the watch task wh

Open console to folder

If your AEM instance is

```
npm config set demo:c
```

Alternatively the config

2. create file

Create a file MyComponent

```
import {ResourceCom  
import * as React from
```

```
import Text from "../tex
```

```
export default class M
```

```
  public renderBody():
```

```
    let label: string = t
```

```
    return (
```

```
      <div>
```

```
        <span>Hello
```

```
        <Text path="
```

```
      </div>
```

```
    );
```



```
}  
  
}
```

3. Register comp

The component needs to be registered in the application's main component file.

`/ui.apps/src/main/ts/components`

```
// add this line at the top  
import MyComponent  
...  
// add this line after component  
componentRegistry.register
```

4. Create component

Create the component at
`my-component.jsx`. The

- `/apps/${appsFolder}`

- .content.xr
- my-compo
- dialog.xml

5. Synchronize s

The component configu
the javascript file.

6. Open browser

find the new react comp

7. Continuously imp

2.3 Demo

The aem-react project c

To run the demo

1. Clone git repo

```
git clone https://
```

2. Install into runnin

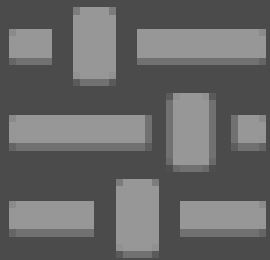
```
mvn install -Pau
```

Currently it is not possible to install the engine via the web browser.

SPA with a two level p

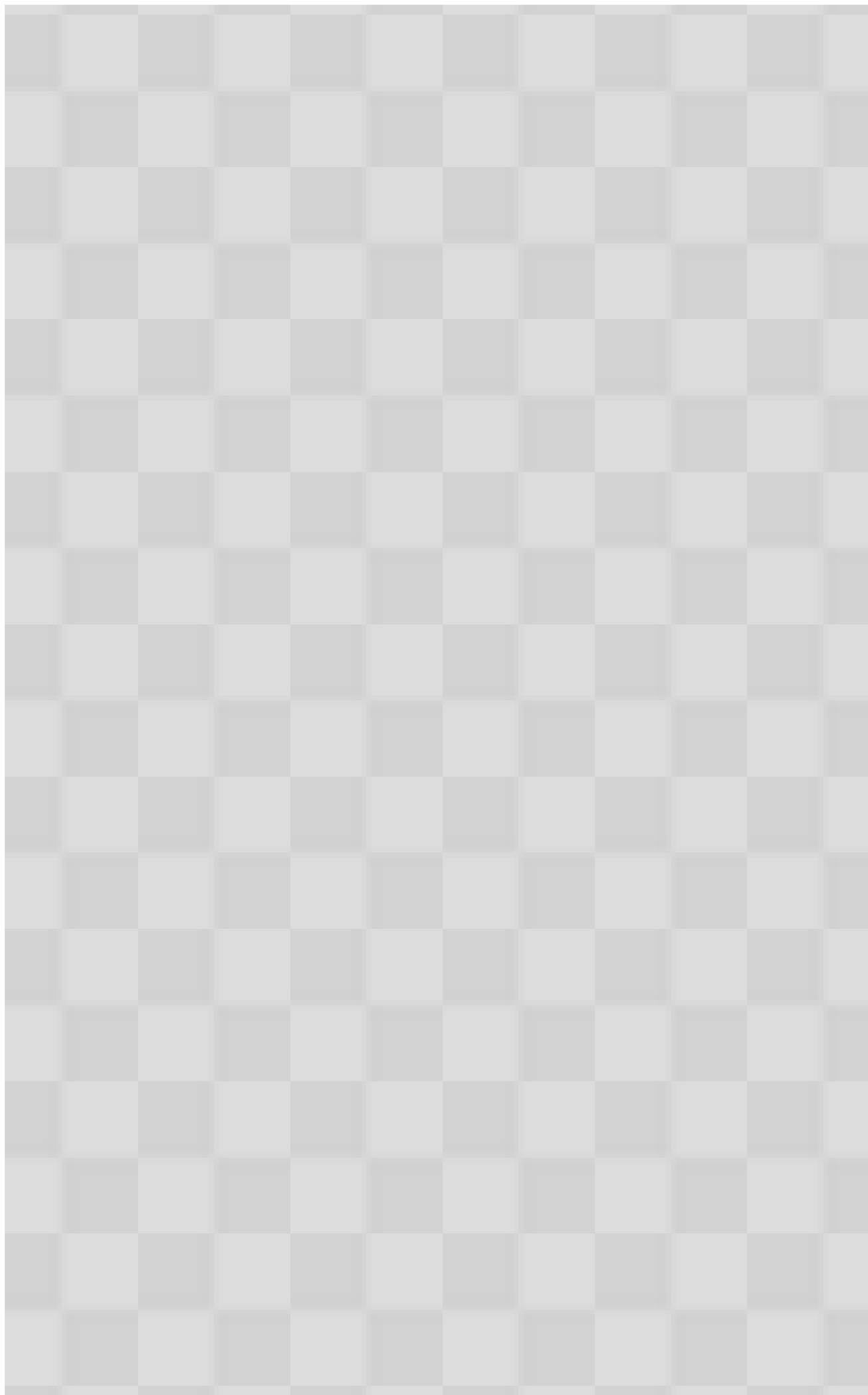
Local url: </content/react>

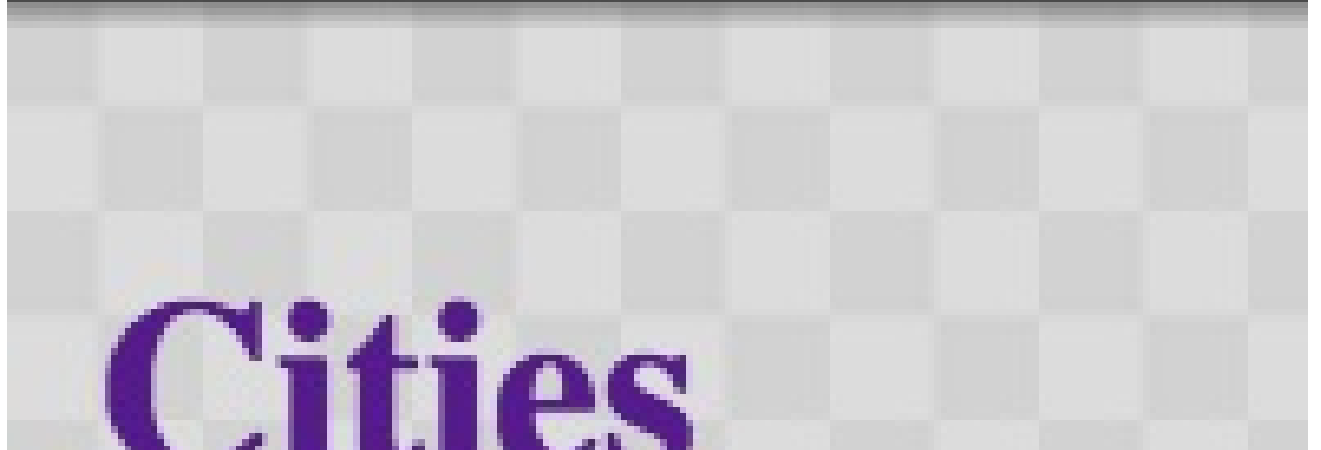
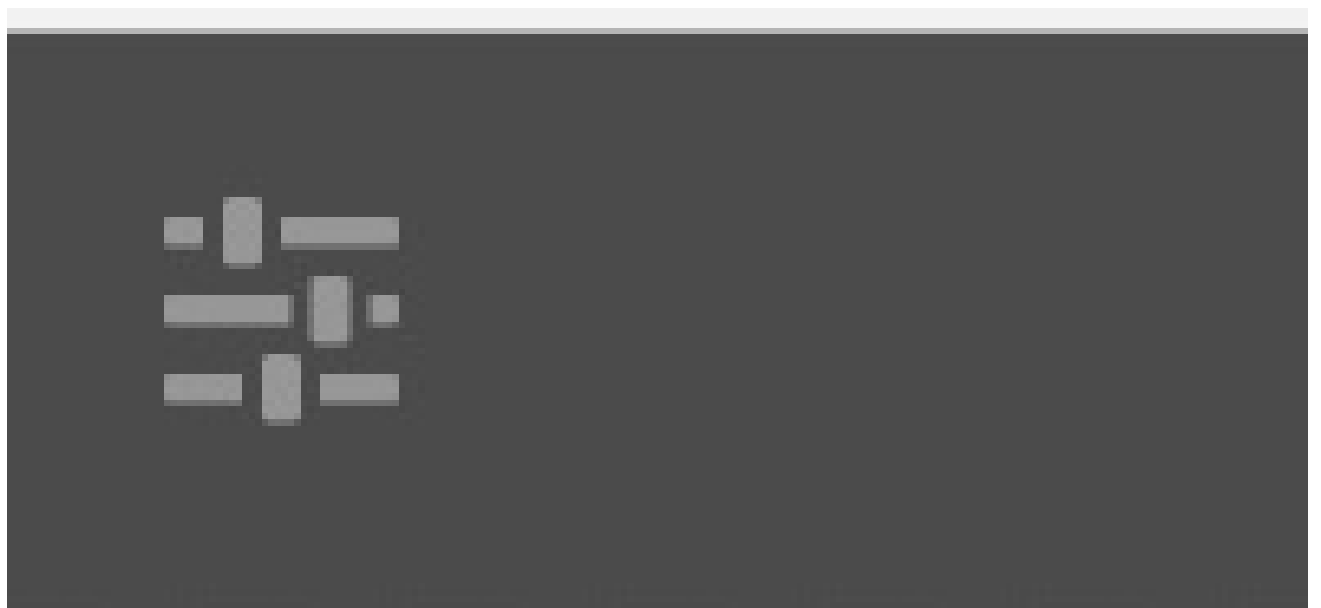
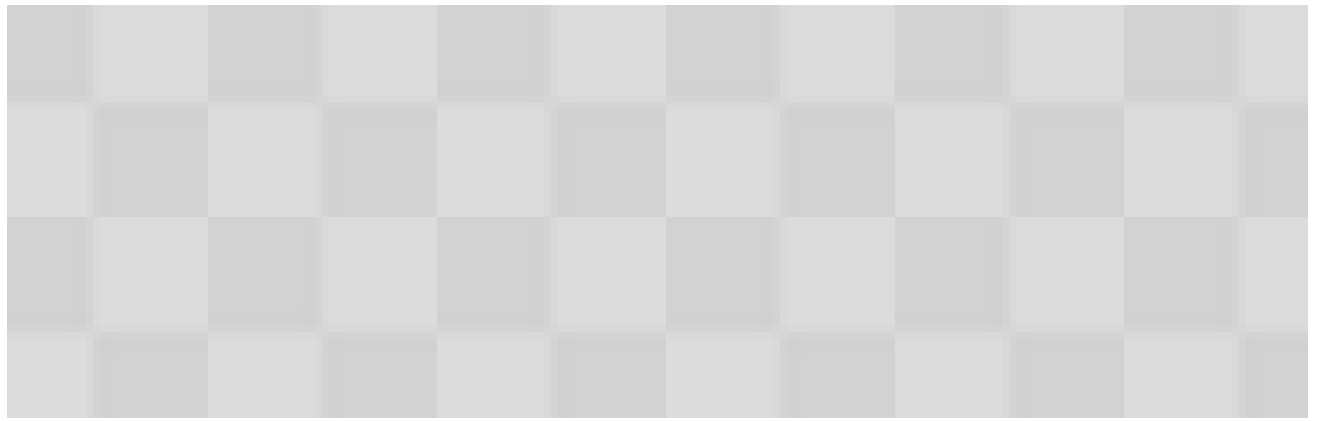
The SPA (single page application) </cities/hamburg.html>).



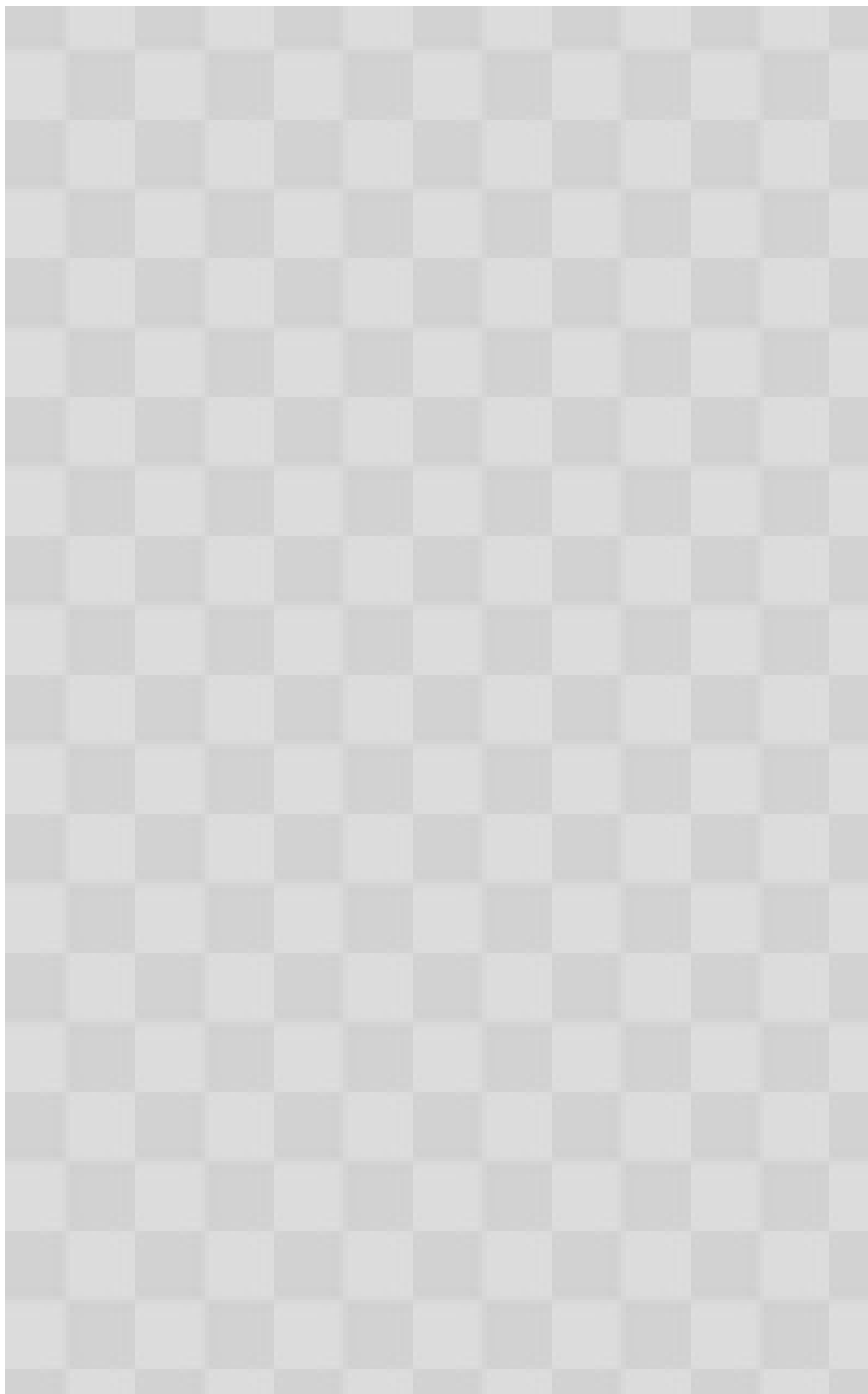
Cities

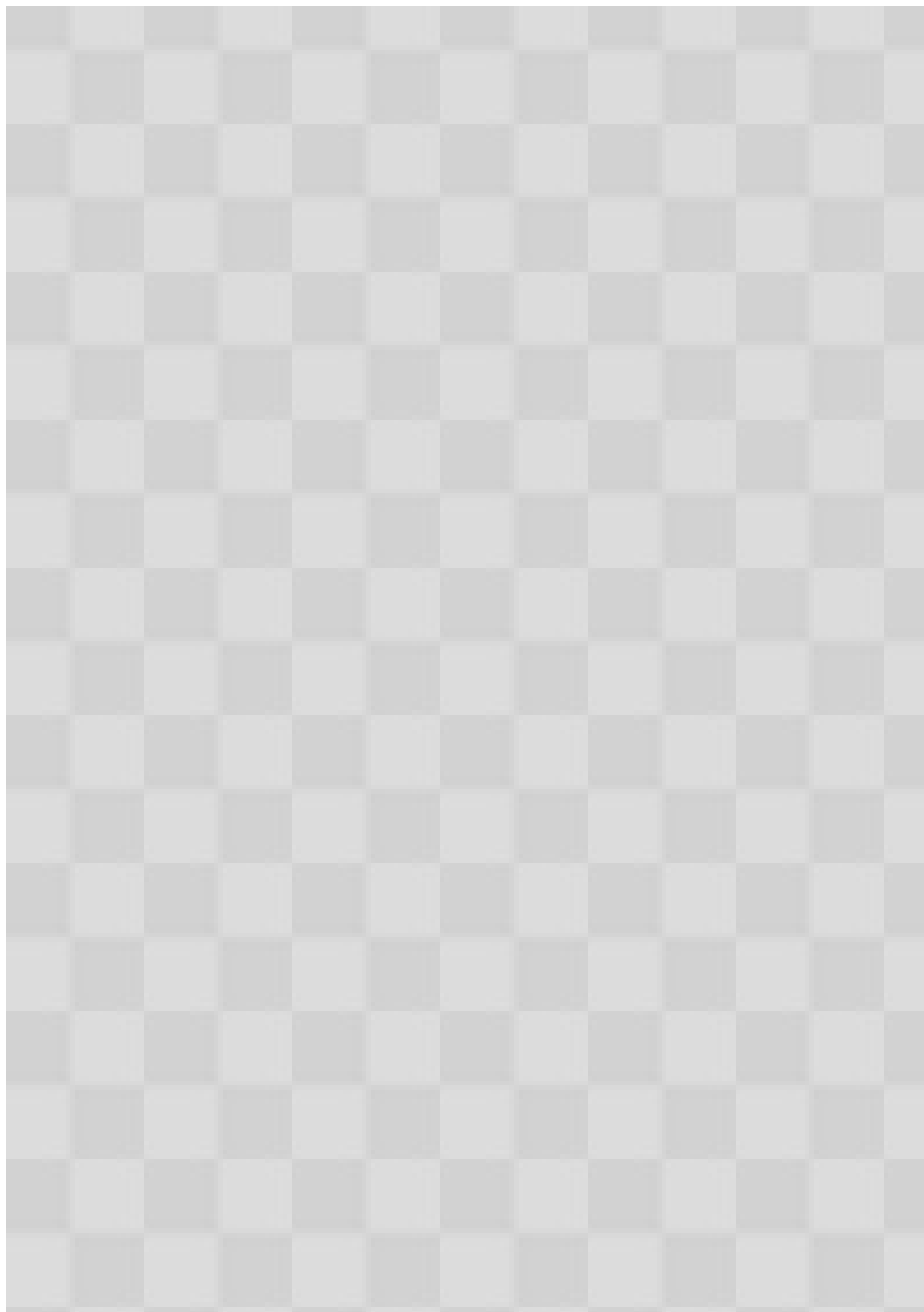
- Hamburg
- Bremen
- München





- Hambur
- Bremen
- Münche





The main AEM react component is displayed on all views of a single AEM page. Each page has a `CityFinder` and `CityView` component.

A simplified version of the `CityView` component is shown below.

```
export default class CityView extends React.Component {  
  renderBody() {  
    // ...  
  }  
}
```

```
return (
```

```
  <Router history={his
```

```
    <Route path="citi
```

```
      <IndexRoute co
```

```
      <Route path="c
```

```
    </Route>
```

```
  </Router>
```

```
);
```

```
}
```

```
}
```

The purpose of the gen

Component list

Local url: </content/react>

The list of components

- simple text comp
- embedded comp
- accordion compc
- vanilla composer
- vanilla panel whic



local



Apps



ume-aem



Text



Embedded

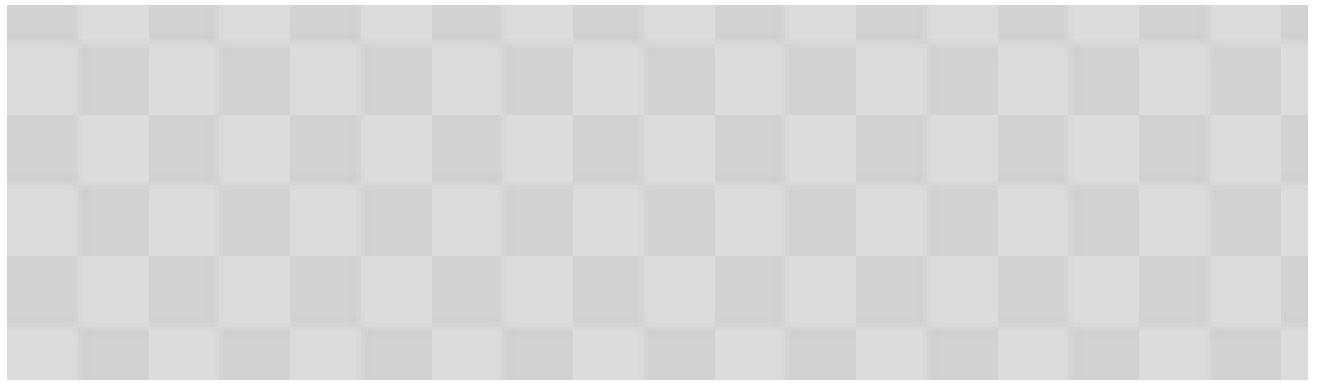
^ Parsys

This is a react text con

This is the **sightly** tex

A React Parsys can in

√ Vanilla



3 Developi

3.1 Resource

ResourceComponent is
so that the component c

The main method to im

The resource is availab

```
import {ResourceComp  
import * as React from  
import Text from "../tex
```

```
export default class M
```

```
  public renderBody():
```

```
    let label: string = t
```

```
    return (
```

```
      <div>
```

```
        <span>Hello
```

```
      </div>
```

```
    );
```

```
  }
```

```
}
```

Container

If the component is a container-like container for any children

```
public renderBody():
```

```
    let label: string = this.props.label
```

```
    let children: React.ReactNode = this.props.children
```

```
    return ()
```

```
        <div>
```

```
            <span>Hello World</span>
```

```
        <span>Hello {id}</span>
        {children}
    </div>
);
}
```

3.2 Registering

There must be one instance of the component registered with one of the

```
import ComponentReg  
import RootComponer  
import MyComponent ·
```

```
let registry: Component  
registry.register(MyCon
```

```
let rootComponentReg  
rootComponentRegistry  
rootComponentRegistry  
AemGlobal.registry = rc
```

If your project was created with Gradle, you can use the `ComponentRegistry` class to register the resource.

3.3 Loading Resources

The resource will be loaded by the `ResourceLoader` class, which is overriding the method `loadResource()` in the `ResourceLoader` interface.


```
public getDepth():  
    return 2;  
}
```

Lazy Loading

If the resource is not fetched
Asynchronous loading
the client. This is often t

```
public renderLoading
```

```
    return (  
        <span>Loading  
    );
```

```
}
```

3.4 Author m

React components are

means that they need to
on the server to detect t

3.5 Java Api

Presentation logic is oft
passed to appropriate n
the actual java method

method

getResourceModel(class
method

getRequestModel(class

getOsgiService(classM

```
import ServiceProxy
...
public renderBody(
    let model: Service
    let label: string =
    return (
        <div>
            <span>Hello
        </div>
    );
}
```

The Java API methods created during server re the same parameters. I

If a service is needed to

3.6 Vanilla cc

A vanilla react compone

```
registry.registerVanilla(  
registry.registerVanilla(  

```

All resource properties

For a simple component
class that should be reg

| |
|------------------|
| parameter |
| depth |

props?
parameter

parsys?

parsys.path

parsys.className?

parsys.elementName?

parsys.childElementN


```
parsys.childClassName  
parameter
```

```
transform?
```

Container

If the **parsys** property is
attribute:

```
<?xml version="1.0"
<jcr:root xmlns:sling="http://sling.apache.org/jcr/sling/1.0"
  jcr:primaryType="cq:Page"
  jcr:title="My Title"
  cq:isContainer="true"
/>
```

Resource transfc

If the resource's structu
passed the resource an

In this example a sling i

```
let transform: any = (re  
    let model: ServicePro  
    let newProps: any =  
    newProps["imageSrc"  
    return newProps;  
};
```

```
registry.registerVanilla(  
    component: myComp  
});
```

4 Configui

The configuration consi
is a single javascript file

4.1 OSGI

The main OSGI service

| |
|-------------|
| name |
| |

**scripts.paths
name**

pool.total.size

scripts.reload

subServiceName

4.2 Javascript

There must be two sepa

which are the same for

Server

For the server the boots

The global variable **Aer**
rendered.

```
declare var Cqx: any;  
declare var AemGlobal  
  
let rootComponentRegi  
AemGlobal.registry = rc  
  
AemGlobal.renderReac  
  
...  
}
```

The implementation of

the ScriptEngine. Cqx is
and the sling implemen

```
let container: Container
container.register("ja")
let cache: Cache = ...
let serverSling = new ...
container.register("sl")
container.register("ca")

let serverRenderer: ServerRenderer
return serverRenderer
```

Client

The javascript for the cl
initReactComponents or
which is basically the s
the client uses the cach

```
interface MyWindow {  
    AemGlobal: any;  
}  
  
declare var window: M  
  
if (typeof window === '  
    throw "this is not the
```

```
}
```

```
let rootComponentRegistry  
rootComponentRegistry  
rootComponentRegistry
```

```
let container: Container  
let cache: Cache = new  
let clientSling: ClientSli  
container.register("sling  
container.register("cach
```

```
let componentManager
```

```
let componentManager  
componentManager.init
```

```
window.AemGlobal = {c
```

5 Single P

This section explains ho

content to the user spread
SPA should be bookmarkable
crawlable by search bots

5.1 Example

Usually a single react component
dynamic parts, which are

The example application

Welcome view

The welcome view pres

A large blue L-shaped graphic that starts near the top left and extends downwards and to the right, framing the text below.

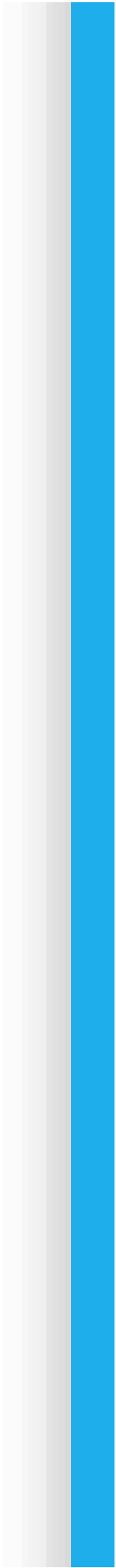
Ne

Two vertical bars on the left side of the page. Each bar consists of a thin light gray line and a thicker blue line to its right.

RIC

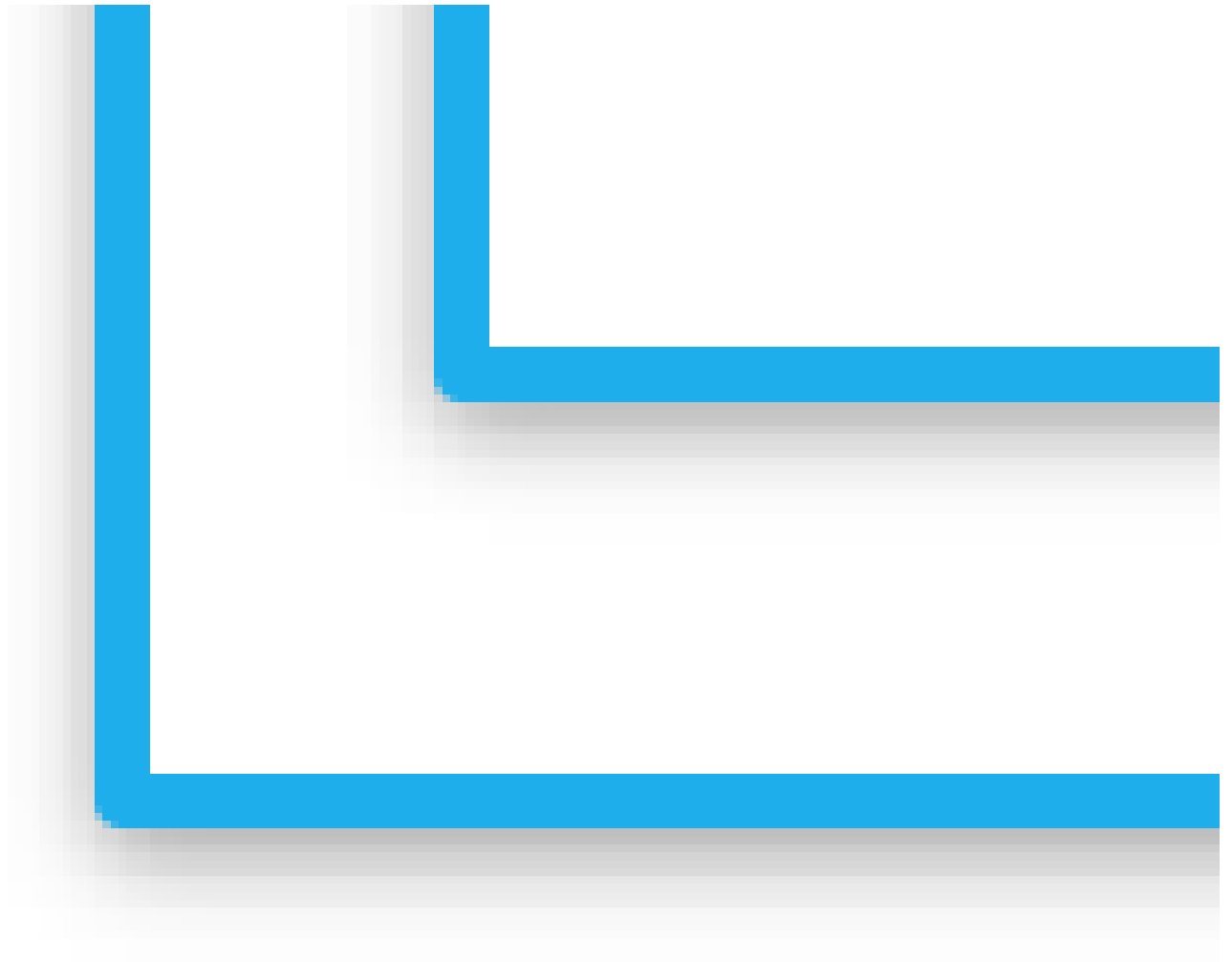
TOK

PAR

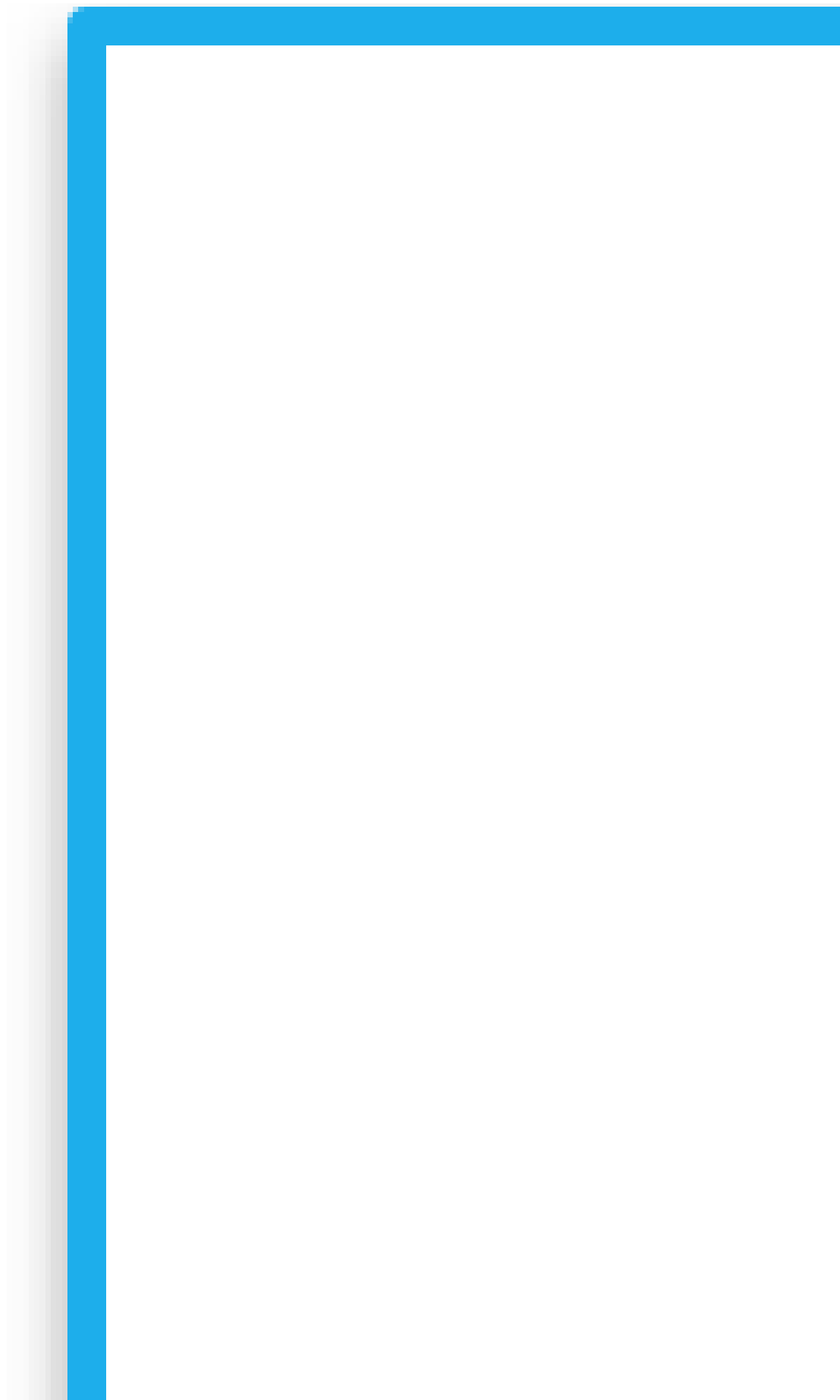


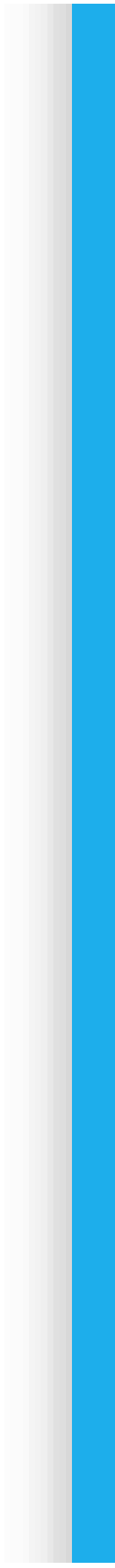
Lor

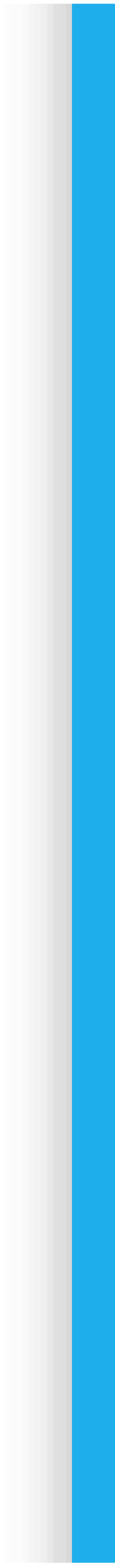
Bei



The **welcome** compone









City view

The **city** view presents



Ne

Rio

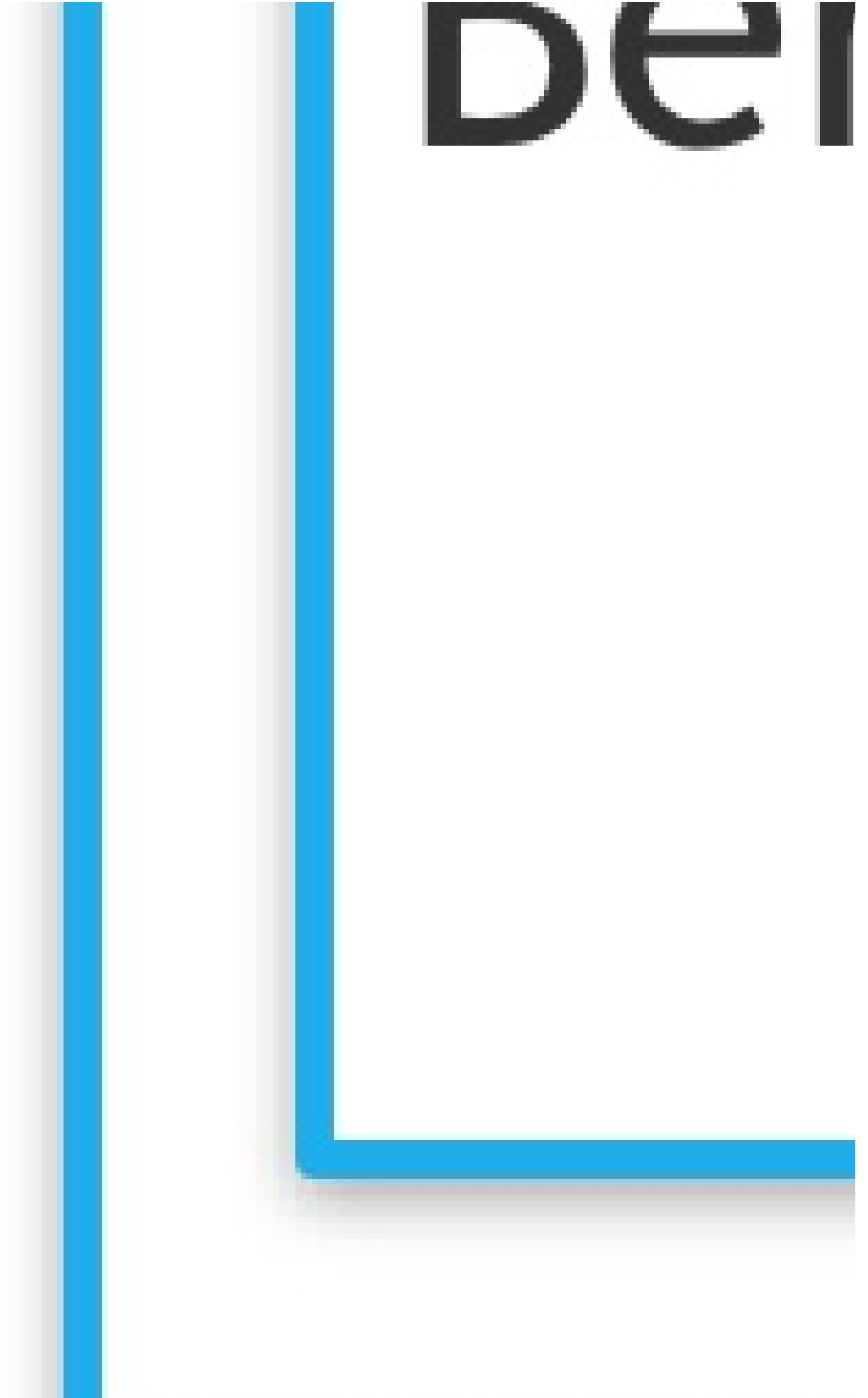
Tok

Par

Lor

D

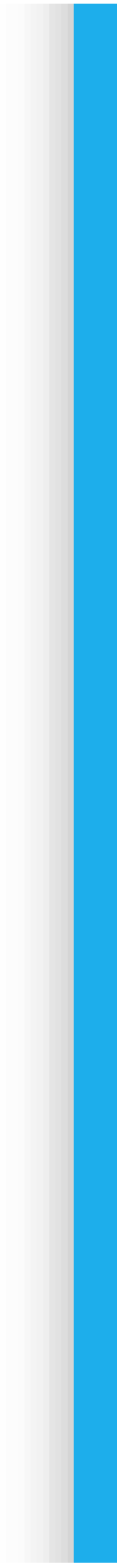
DEI

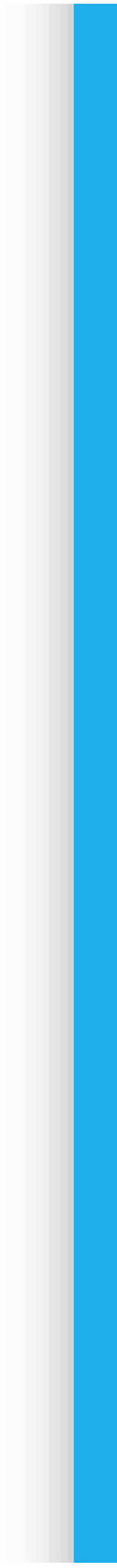


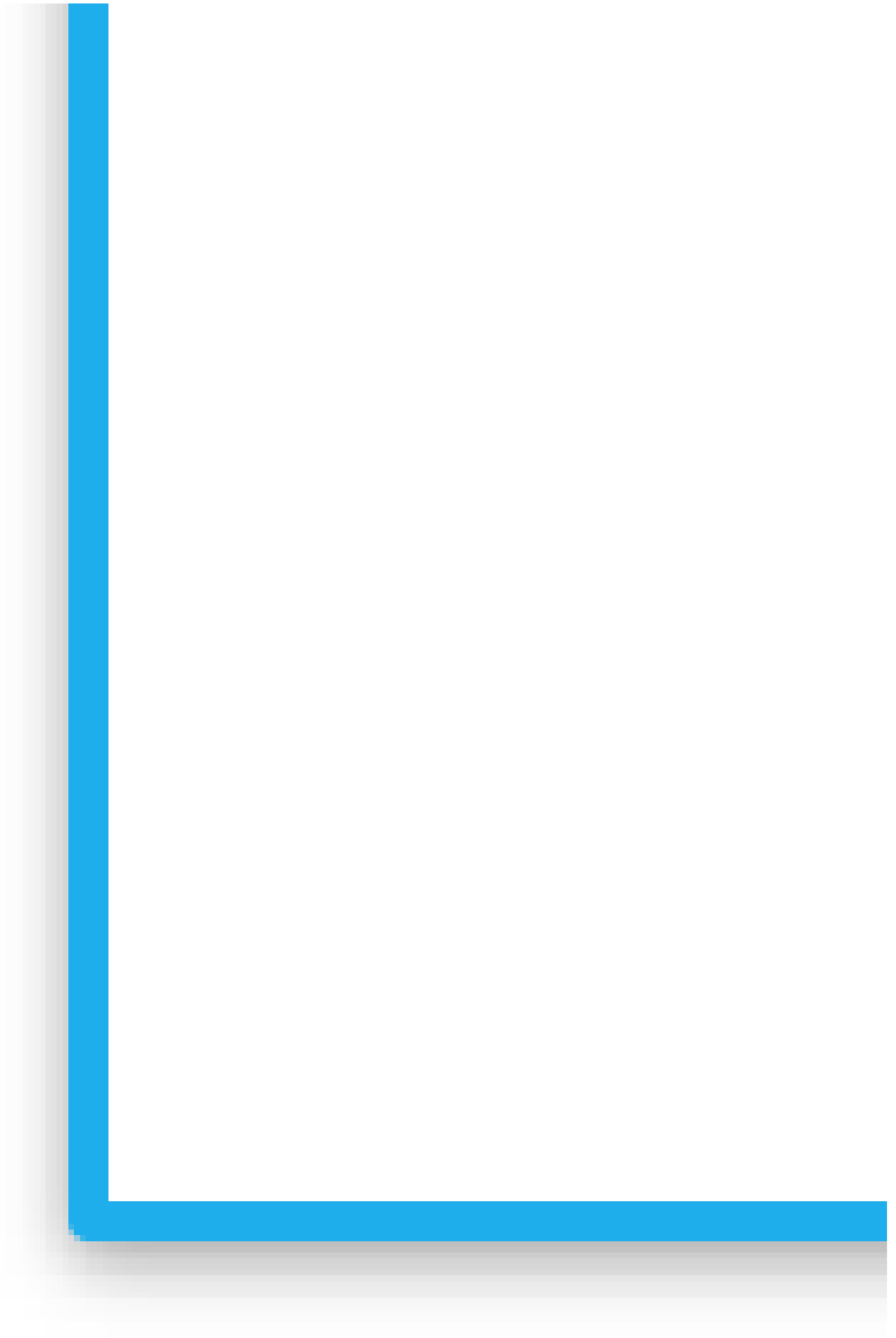


The welcome compone









Router

There are two routes in

1. The route to the v
2. The second route

Another aspect of the r
individual route. The ro

```
<Router history={history}
  <Route path='/cities.'
    <IndexRoute comp
  <Route path='/cities
  </Route>
</Router>
```

Note that the component
a resourcePath and pas

To use this component

- /cities.html
 - /cities/ham
 - /cities/mür

Important

The router component

6 Tools

This section describes the tools that are installed in the directory

6.1 Build Task

The build task transpiles the project `ui.apps/target/classes`

- `reactClient.js`

- reactServer.js

```
npm run start
```

6.2 Watch Task

The watch task will watch the source files and
configure the address of the server.

```
npm config set demo:config
```

All typescript files are w
`ui.apps/target/classes/etc`

generated on 1.3.17
by `count-docu`