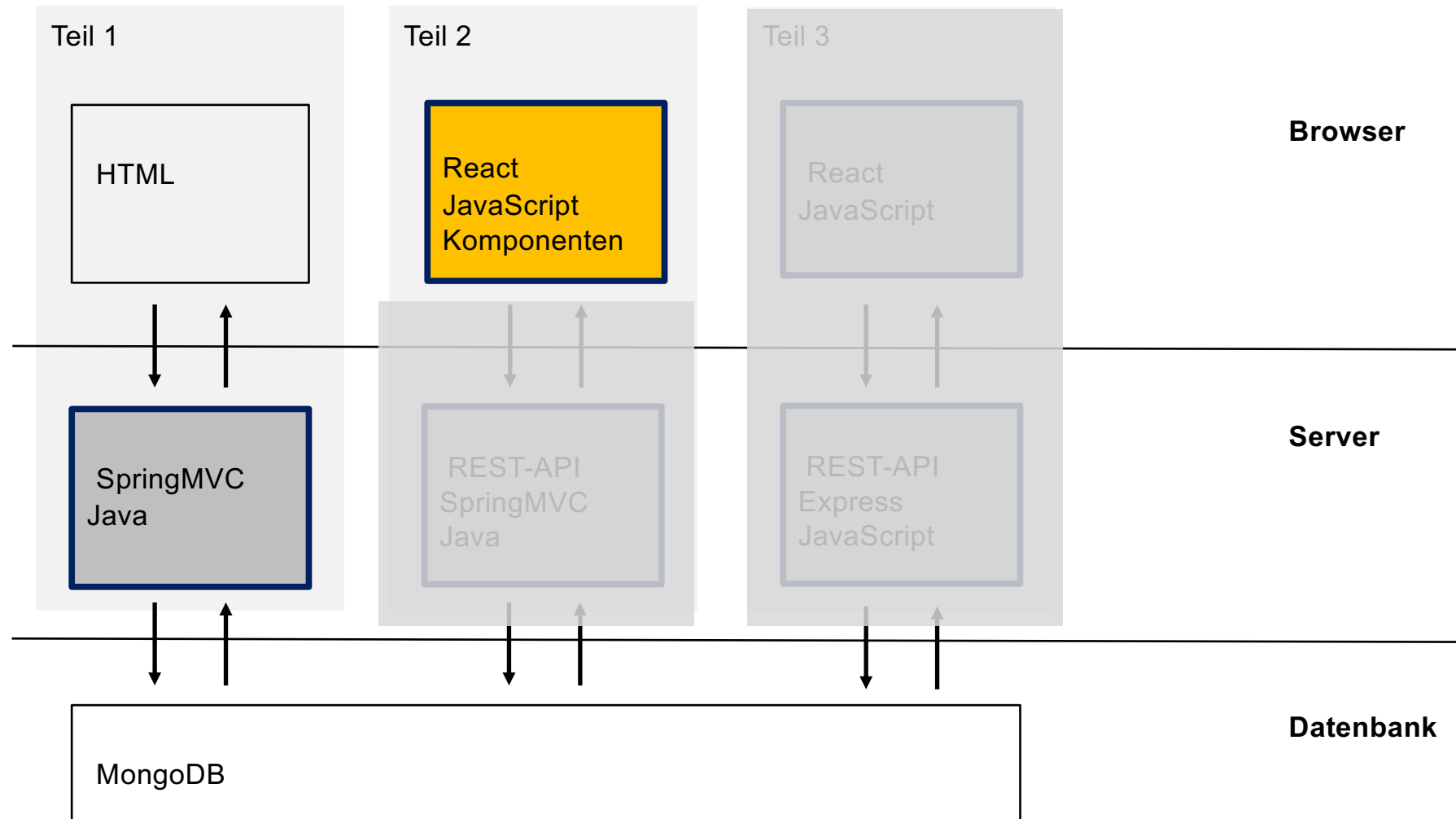


React Interaktionen

Themen heute

- Besprechungen
 - Übung 7
- Interaktionen mit React
 - Event Handling

Lab "flashcard": Setup



Besprechung Übung 7 (1/3)

Aufgabe 1: Komponente "QuestionnaireContainer"

```
class QuestionnaireContainer extends React.Component {  
  render() {  
    return (  
      <div>  
        <h3>Questionnaires</h3>  
        <QuestionnaireTable  
          questionnaires={this.props.qs}/>  
      </div>  
    )  
  }  
}
```

ES6 Klasse

Kein State!
(read-only)

Übergabe der "Questionnaires" an
"QuestionnaireTable" über die Properties

```
QuestionnaireContainer.defaultProps = {  
  qs: [  
    { 'id': 1, 'title': 'Test Title 1', 'description': 'Test Description 1' },  
    { 'id': 2, 'title': 'Test Title 2', 'description': 'Test Description 2' },  
    { 'id': 3, 'title': 'Test Title 3', 'description': 'Test Description 3' },  
    { 'id': 4, 'title': 'Test Title 4', 'description': 'Test Description 4' },  
    { 'id': 5, 'title': 'Test Title 5', 'description': 'Test Description 5' }  
  ]  
}  
  
export default QuestionnaireContainer;
```

Default-Werte zur Initialisierung des Array "qs"

Besprechung Übung 7 (2/3)

■ Aufgabe 2: Komponente "QuestionnaireTable"

Functional Component

```
const QuestionnaireTable = ({questionnaires}) => (  
  <Table hover>  
    <tbody>  
      {questionnaires.map(questionnaire =>  
        <QuestionnaireTableElement  
          key={questionnaire.id}  
          questionnaire={questionnaire}  
        />  
      )}  
    </tbody>  
  </Table>  
)  
  
export default QuestionnaireTable;
```

"Table" Komponente vor "reactstrap"

map() erstellt einen neuen Array

"key" Attribute ist ein wichtiges und spezielles String Attribute, das bei jeder React-Liste gebraucht wird und innerhalb der Liste eindeutig sein muss. Der Key wird von React benutzt, um feststellen zu können, welches Listenelement sich verändert hat.

<https://facebook.github.io/react/docs/lists-and-keys.html#keys>

Besprechung Übung 7 (3/3)

■ Aufgabe 3: Komponente "QuestionnaireTableElement"

```
const textStyle = {  
  verticalAlign: 'middle'  
};
```

Definition eines Style.

```
const QuestionnaireTableElement = ( {questionnaire } ) => (  
  <tr>  
    <td colSpan="1" style={textStyle}>  
      {questionnaire.id}  
    </td>  
    <td colSpan="3" style={textStyle}>  
      {questionnaire.title}  
    </td>  
    <td colSpan="10" style={textStyle}>  
      {questionnaire.description}  
    </td>  
  </tr>  
)
```

Functional Component

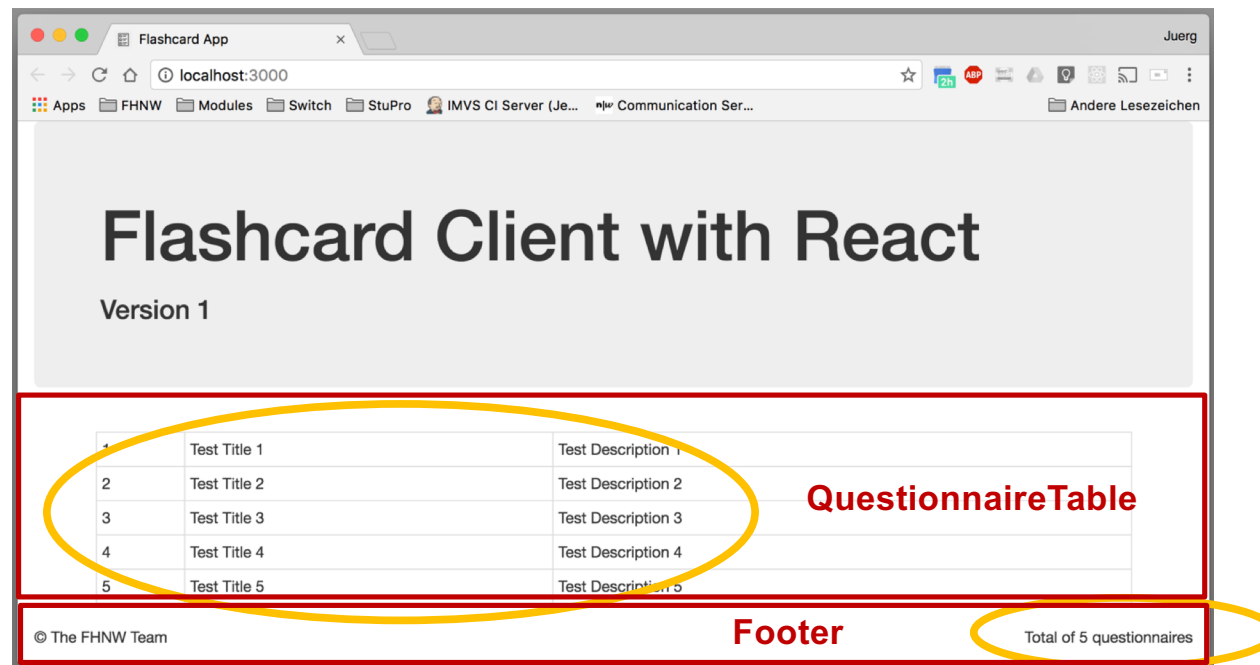
Normale "Table"-Zeile

Referenz auf den Style.

```
export default QuestionnaireTableElement;
```

Managing Application State

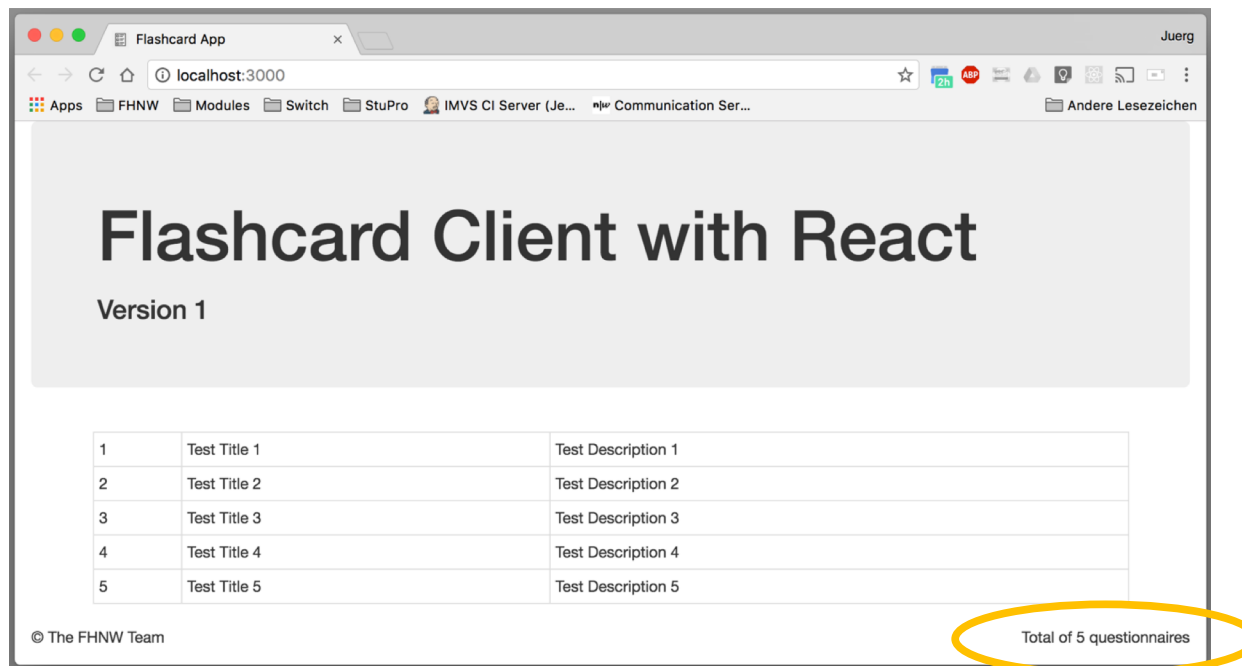
- Der State wird immer innerhalb einer Komponente verwaltet.
- Wird ein State von mehreren Komponenten verwendet so muss dieser State durch eine gemeinsame Parent-Komponente verwaltet werden.
- Die State wird über die Properties an die Child-Komponenten übergeben.
- Beispiel: State "questionnaires" in "Table" und "Footer". Parent-Component?



Arbeitsblatt 16

■ "Lifting State Up" Pattern

- Wo muss der State verwaltet werden, der für mehrere Komponenten relevant ist?



Arbeitsblatt 16: Zusammenfassung

- ... "Single Responsibility Principle" ist zentral
- ... klare Zuordnung der Verantwortungen eines "State"
- ... Komponenten können in der Hierarchie einfach verschoben werden
- ... gerichteter Datenfluss => "Owner to Child"
- ... Daten werden mittels Properties an Child übergeben
- ... unterstützt kein "Two-Way Data Binding"

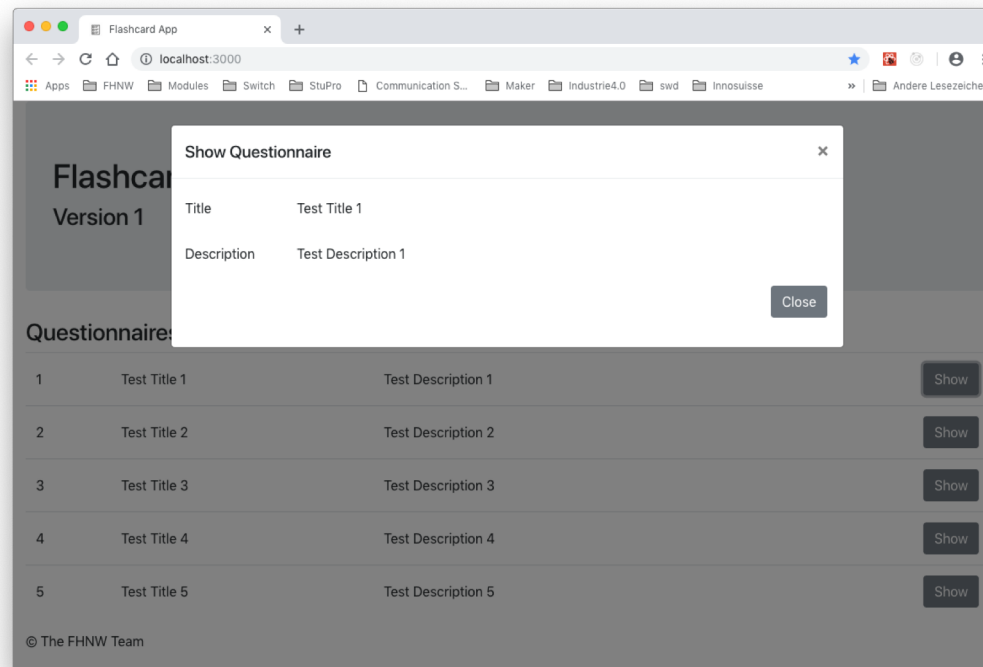
Arbeitsblatt 17

■ READ Dialog

Aufgabe 1: Modal Dialog von Bootstrap

Aufgabe 2: React Komponente "QuestionnaireShowDialog"
State?

Aufgabe 3: "QuestionnaireTableElement" erweitern



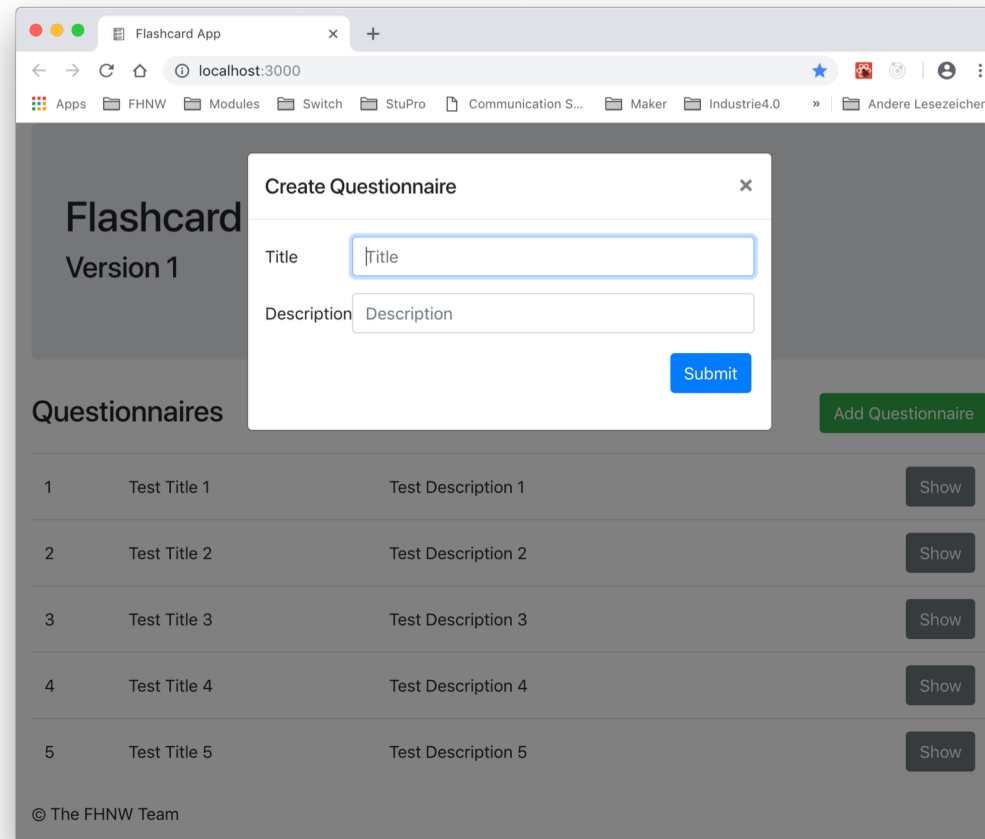
Arbeitsblatt 17: Zusammenfassung

- ... Handler Methoden sind normale JavaScript Methoden
- ... "State" auf das Minimum reduzieren
- ... Modal Dialog von Bootstrap ist einfach integrierbar
- ... Anzeigen des Dialogs wird über State gesteuert

Arbeitsblatt 18

■ "Inverse Data Flow"

- Aus Child-Komponente "QuestionnaireCreateDialog" die Komponente "QuestionnaireContainer" aktualisieren



Arbeitsblatt 18: Zusammenfassung

- ... Modal Dialog von Bootstrap ist einfach integrierbar
- ... Callback-Methode vom Owner über Property an Child übergeben
- ... submit() nutzt Callback, um einen Questionnaire an "Owner" hochzureichen
- ... im Owner wird der Questionnaire in die Liste eingefügt
- ... jede Tasteneingabe wird in einer entsprechenden Handler-Methode verarbeitet
- ... über Event kann Wert aus Input-Feld ausgelesen werden

Übung 8 als Hausaufgabe

- CRUD in "Flashcard App" vervollständigen

