

Übung 07

1 Aufgaben

Bearbeiten Sie nun die folgenden Aufgaben.

Aufgabe 1

Erstellen Sie eine simple Blog Webseite mithilfe von Vue.js. Nutzen Sie hierfür die zur Verfügung gestellte `myBlog.html` und implementieren Sie folgenden Funktionalitäten:

- Ein Blogeintrag soll sowohl einen Titel, sowie einen Inhalt anzeigen. Erstellen Sie zum Testen einen Eintrag der beim Aufruf der Webseite immer angezeigt wird.
- Es soll eine Funktion implementiert werden, die einen neuen Blog mit Nutzerspezifiziertem Titel und Inhalt erstellt. Diese Option soll immer unter dem letzten Blogeintrag angezeigt werden.
- Weiterhin soll jeder Blogeintrag eine `Kommentarfunktion` beinhalten. Hierbei sollen die Kommentare unter dem dazugehörigen Blogeintrag chronologisch angezeigt werden.
- Ein Kommentar soll einen Nutzer und einen Inhalt beinhalten, die von dem Nutzer in ein Interface innerhalb des Blogeintrags eingegeben werden können (Hinweist: Ein Login ist hier nicht benötigt, der Name des Nutzers kann über das Interface angegeben werden).
- Die erstellten Kommentare sollen nur in dem dazugehörigen Blogeintrag angezeigt und nummeriert werden.

Aufgabe 2

Erstellen Sie eine To-Do-List Webseite mithilfe von Vue.js. Nutzen Sie hierfür die zur Verfügung gestellte `ToDoList.html` und implementieren Sie folgenden Funktionalitäten:

- Die erstellte Webseite soll mehrere To-Do Einträge besitzen, jeder To-Do Eintrag besitzt einen Titel, einen Inhalt, eine Deadline und eine Checkbox die anzeigt ob der To-Do Eintrag fertiggestellt wurde.
- Sollte der Eintrag fertiggestellt worden sein, soll der komplette Eintrag grün markiert werden.
- Jeder Eintrag soll weiterhin einen Entfernen-Knopf außerhalb des Eintrags besitzen. Der Knopf soll den Eintrag komplett aus der Liste aller Einträge entfernen.
- Um einen Eintrag zu erstellen, soll ein Interface zur Verfügung gestellt werden, welches dem Nutzer erlaubt einen To-Do Eintrag zu erstellen. Hierbei sollen Felder für den Titel, Inhalt und die Deadline implementiert werden.

Lösungen zu den Aufgabes

Lösung 1

Listing 1

```
var newComment = {
  data: function(){
    return{
      comment: {
        user: "",
        content: ""
      }
    }
  },
  template: '<div class="addNewComment">'+
    ' <table><tr><td>User</td><td><input v-model="comment.user"></td></tr>'+
    ' <tr><td>Inhalt</td><td><textarea'
    ↪ v-model="comment.content"></td></tr></table>'+
    ' <button v-on:click="$emit(`commit-new-comment`,'
    ↪ comment)`">Posten</button></div>'
  }
}

var commentSection = {
  props: ['comment'],
  template: '<div class="comment" style="border-style: solid; margin:'
  ↪ 2px;"><table>'+
    ' <tr><td><h3>Eintrag {{comment.id}}</h3></td></tr>'+
    ' <tr><td><p>{{comment.user}} schrieb:</p></td></tr>'+
    ' <tr><td><p>{{comment.content}}</p></td></tr>'+
    ' </table></div>'
  }
}

var myBlogMainBody = {
  props: ['blog'],
  data: function(){
    return{
      comments: [],
      commentCount: 0
    }
  },
  methods: {
    addNewComment: function(comment){
      this.commentCount++;
      let newCommentEntry = {
        user: comment.user,
        content: comment.content,
        id: this.commentCount
      }
      this.comments.push(newCommentEntry);
    }
  },
  components: {
    'comment-section': commentSection,
    'new-comment': newComment
  },
  template: '<div class="blogEntry" style="border-style: solid; margin:'
  ↪ 10px;"><table>'+
    ' <tr><td><h3>{{blog.title}}</h3></td></tr>'+
```

```

50     ' <tr><td><p>{{blog.content}}</p></td></tr></table>' +
    ' <comment-section v-for="comment in comments"
    ↪ :comment="comment"></comment-section>' +
    ' <new-comment @commit-new-comment="addNewComment"></new-comment></div>'
  }
}

55 var newBlogEntry = {
  data: function() {
    return {
      blogEntry: {
        title: "",
60        content: ""
      }
    }
  },
  template: ' <div class="newBlog">' +
65   ' <table><tr><td>Titel</td><td><input
    ↪ v-model="blogEntry.title"></td></tr>' +
    ' <tr><td>Inhalt</td><td><textarea
    ↪ v-model="blogEntry.content"></textarea></td></tr></table>' +
    ' <button v-on:click="$emit(`commit-new-blog`,
    ↪ blogEntry)`">Erstellen</button></div>'
  }
}

70 var vm = new Vue({
  el: "#app",
  data: {
    blogEntrys: [
      {
75        title: "Erster Eintrag",
        content: "Hier könnte dein Inhalt stehen"
      },
      {
        title: "Zweiter Eintrag",
        content: "Hier könnte noch mehr Inhalt stehen"
80      }
    ],
    methods: {
      addNewBlog: function(blog) {
        let newBlog = {
85          title: blog.title,
          content: blog.content,
        }
        this.blogEntrys.push(newBlog);
      }
    },
90    components: {
      'my-blog': myBlogMainBody,
      'create-blog-entry': newBlogEntry
    }
  }
});
95

```

Lösung 2

Listing 2

```

var mainToDoList = {
  data: function() {

```

```

    return {isDone: false}
  },
  methods: {
    deleteToDoItem: function() {
      this.isDone = false;
      this.$emit("delete-todo", this.item);
    }
  },
  props: ['item'],
  template: '<div class="main-todo"><table class="todo-list" :class="{finished : '
    ↪ isDone}">' +
    ' <tr><th class="todo-list">{{item.title}}</th></tr>' +
    ' <tr><td class="todo-list">{{item.content}}</td></tr>' +
    ' <tr><td class="todo-list">{{item.deadline}}</br>' +
    ' <input type="checkbox" id="checkbox" '
    ↪ v-model="isDone"><label>Fertig</label></td></tr>' +
    '</table><button v-on:click="deleteToDoItem">Entfernen</button></div>'
  }

  20 var toDoListController = {
    data: function() {
      return {
        newToDoItem :
        {
          25 key : 0,
            title : "",
            content : "",
            deadline : ""
          }
        }
      },
      30 template: '<div class="list-controller">' +
        ' <table><tr><td>Titel</td><td><input '
        ↪ v-model="newToDoItem.title"></td></tr>' +
        ' <tr><td>Beschreibung</td><td><textarea '
        ↪ v-model="newToDoItem.content"></td></tr>' +
        ' <tr><td>Deadline</td><td><input '
        ↪ v-model="newToDoItem.deadline"></td></tr></table>' +
        ' <button v-on:click="$emit(`commit-todo`, '
        ↪ newToDoItem)`">Hinzufügen</button></div>'
      }

    40 var vm = new Vue({
      el: "#app",
      data: {
        toDoItems: [{
          id: 0,
          title: "First Item",
          45 content: "Erstelle to-Do-List",
          deadline: "06.07.2020"
        }],
        idCounter : 0
      },
      50 methods: {
        addToDoItem: function(item) {
          this.idCounter++;
          item.id = this.idCounter;
          let newToDoItem = {
          55 id: item.id,
            title: item.title,
            content: item.content,
            deadline: item.deadline
          }
        }
      }
    })
  }

```

```

    }
    this.todos.push(newToDoItem);
  },

  deleteToDoItem: function(item){
    for(todoIndex in this.todos){
      if(this.todos[todoIndex].id == item.id){
        this.todos.splice(todoIndex, 1);
      }
    }
  },
},
components:{
  'main-to-do-list' : mainToDoList,
  'to-do-list-controller' : ToDoListController
}
});

```
