



GEFÖRDERT VOM

Bundesministerium  
für Bildung  
und Forschung

IBM®

# The Frontend Overview

## COMPASS NUM-APP

20.04.2021



# Objectives

After this session, you should be able to answer these questions:

- What Features does the NUM-App Frontend provide?
- What does the App Flow look like?
- Where are the different customization- and development options located?
- How can the UI be customized?





netzwerk  
universitäts  
**medizin**

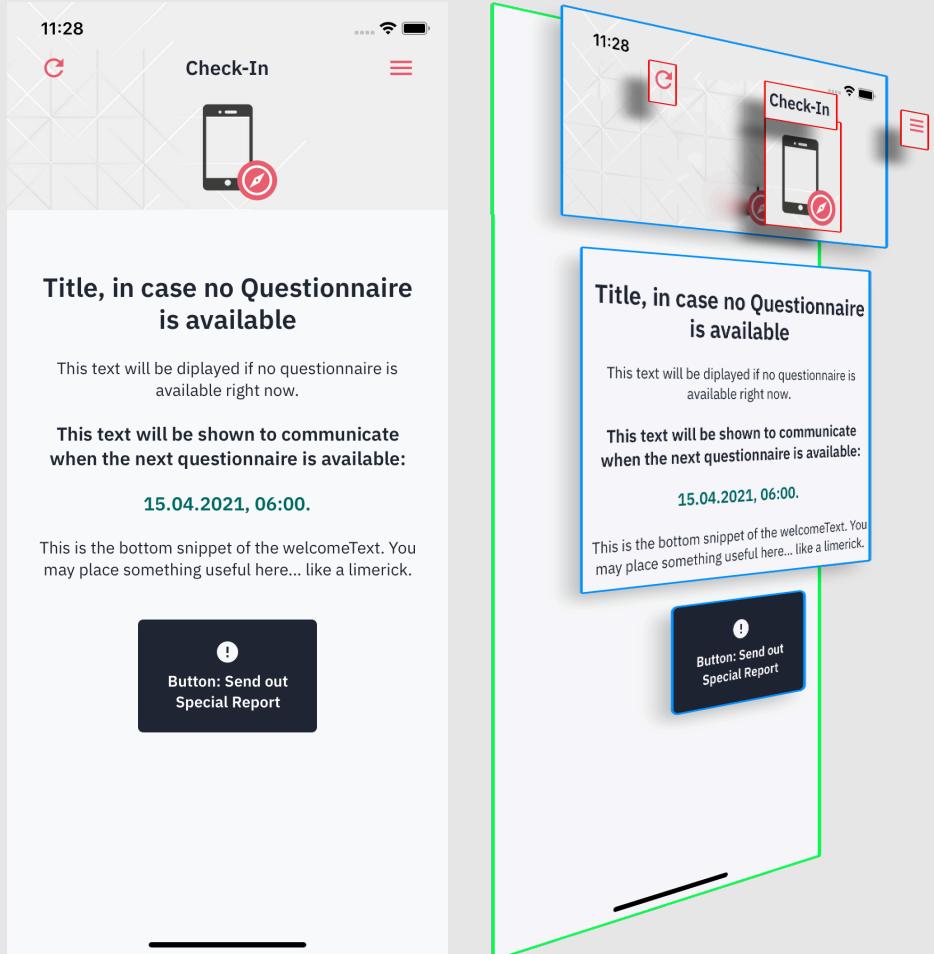
GEFÖRDERT VOM

Bundesministerium  
für Bildung  
und Forschung

IBM®

# Recap Previously on the NUM-Frontend Track...

# Recap UI Components



React Native allows the developer to create simple components that can be stacked on top of (or into) each other.

Each component consists of:

- A visual representation layer (Template)
- A logic layer (Controller)
- **A data layer (App State or Props)**

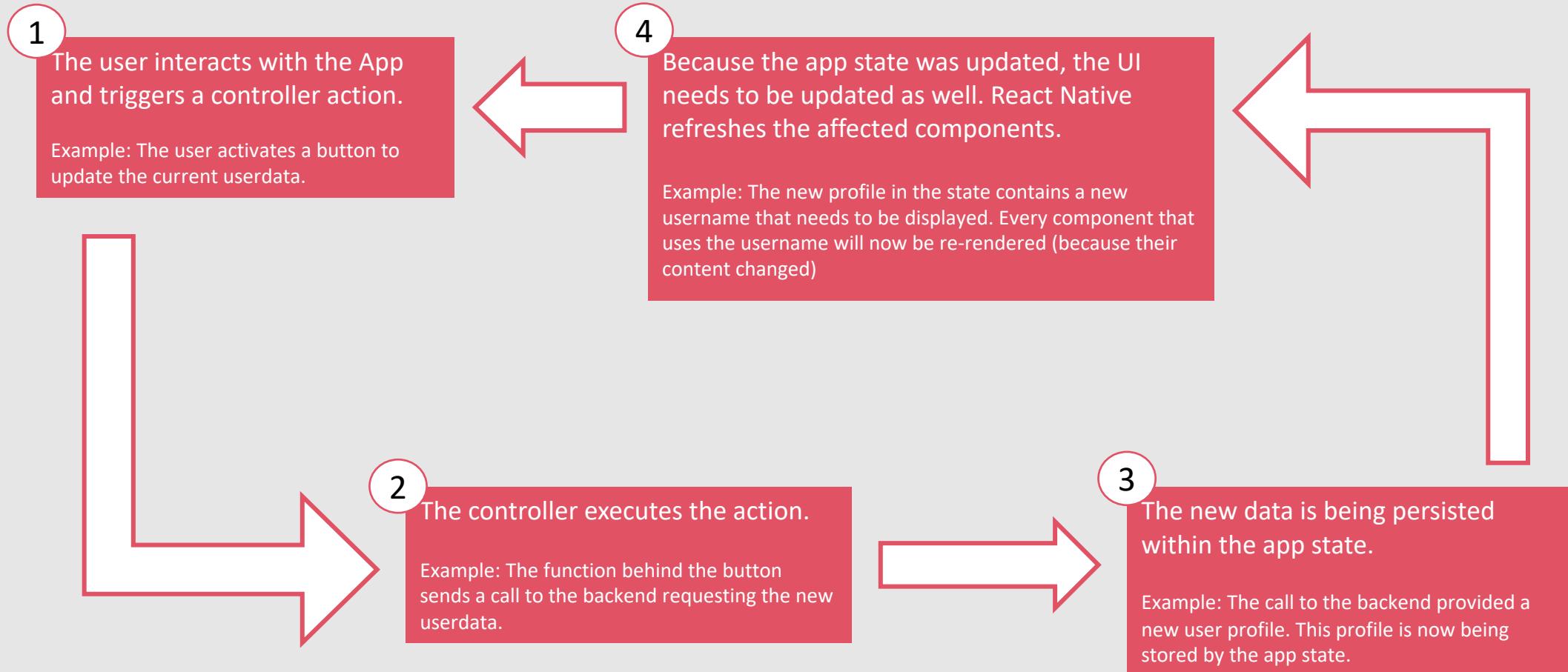
The app state basically consists of all mutable data that affects a component. For example:

- The string that is being used on a button label
- The color of a button
- The size of an UI-Element

**If the state changes, the UI will be updated automatically!**

# Recap

# The basic Lifecycle of a Component

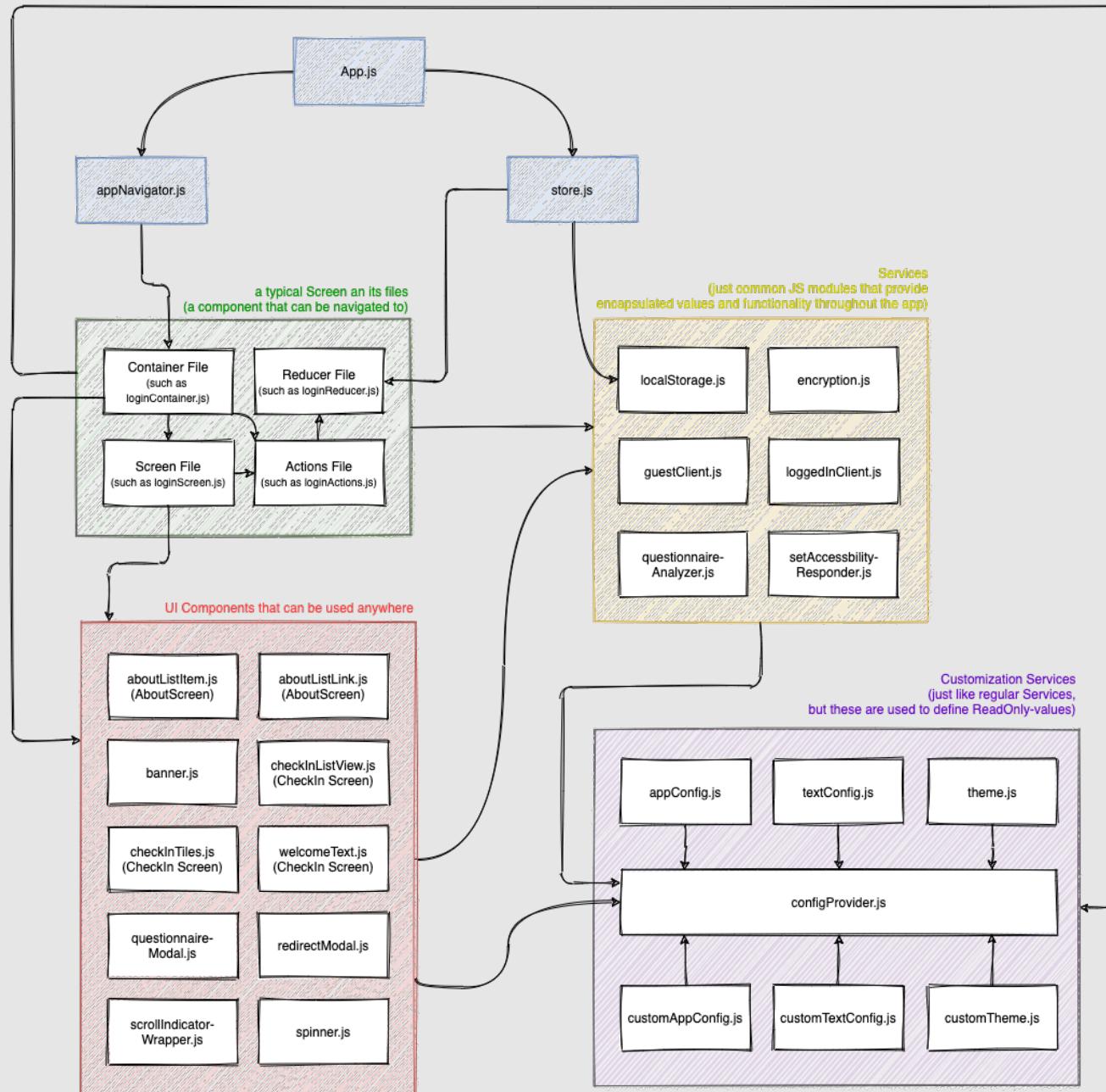


# Recap

# A map of the Frontend

The frontend consists of various files that are being used for specific and common purposes.

- **Blue:** defining the basic structure of the app
- **Green:** represents the screens that can be navigated to
- **Red:** simple UI components that can be used anywhere (for example in Screen-Components)
- **Yellow:** Modules that provide encapsulated functions
- **Purple:** used to set configuration- and styling options





GEFÖRDERT VOM

Bundesministerium  
für Bildung  
und Forschung

IBM®

# The basic Features

# Features Overview

These are the main features of the NUM-App Frontend:

- Login & User Management
- FHIR Questionnaire Retrieval
- FHIR Questionnaire Rendering & Interaction
- FHIR Questionnaire Response Encryption & Transmission
- Transmitting a Report (if no questionnaire is available)
- Displaying static or dynamic content



# Features

## Login & User Management

- Users can authenticate using a QR-Code, which contains the ID for his/her participation. This ID will be persisted on the device to automatically log the user in the next time the app is opened.
- Every time the user triggers an action that enables a connection to the backend, the user will be updated, and the frontend refreshed (with the data of the now updated user).
- There is **no logout functionality** in production mode.
- The user information is available from the **Login-State** and from the **CheckIn-State** (with some additional attributes)

Login State before the login-action

```
▼ Login:
  loading: false
  loggedIn: false
  loginError: null
  loginUnauthorized: false
  session: null
  userId: null
```

Login State after the login-action

```
▼ Login:
  loading: false
  loggedIn: true
  loginError: null
  loginUnauthorized: false
▼ session:
  accessToken: "7bfc3b07-a97d-4e11-8ac6-b970c1745476"
  recipientCertificatePemString: "-----BEGIN CERTIFICA
► __proto__: Object
userId: "7bfc3b07-a97d-4e11-8ac6-b970c1745476"
```

# Features

## Login & User Management

```

▼ CheckIn:
  categories: null
  categoriesLoaded: false
  currentCategoryIndex: null
  currentPageIndex: 1
  error401: false
  firstTime: false
  loading: false
  noNewQuestionnaireAvailableYet: true
  questionnaireError: null
  questionnaireItemMap: null
  questionnaireResponseError: null
  showDatePicker: false
  showQuestionnaireModal: false

▼ user:
  additional_iterations_left: 0
  current_instance_id: "181caeb9-1d7b-4b9d-b285-e558c2b94c24"
  current_interval: 7
  current_questionnaire_id: "fedaf08b-aae2-4d9b-8306-8a3de02d2f74"
  due_date: "2021-04-23T16:00:00.000Z"
  firstTime: false
  pushAppGUID: "f98fc829-1d4d-418b-b05b-a147fdb9c0f3"
  pushClientSecret: "ab4b7d62-1fde-4a46-a566-2d1ada678df3"
  recipient_certificate_pem_string: "-----BEGIN CERTIFICATE-----MIIE0zC0
  start_date: "2021-04-20T04:00:00.000Z"
  study_id: "7bfc3b07-a97d-4e11-8ac6-b970c1745476"
  userId: "7bfc3b07-a97d-4e11-8ac6-b970c1745476"

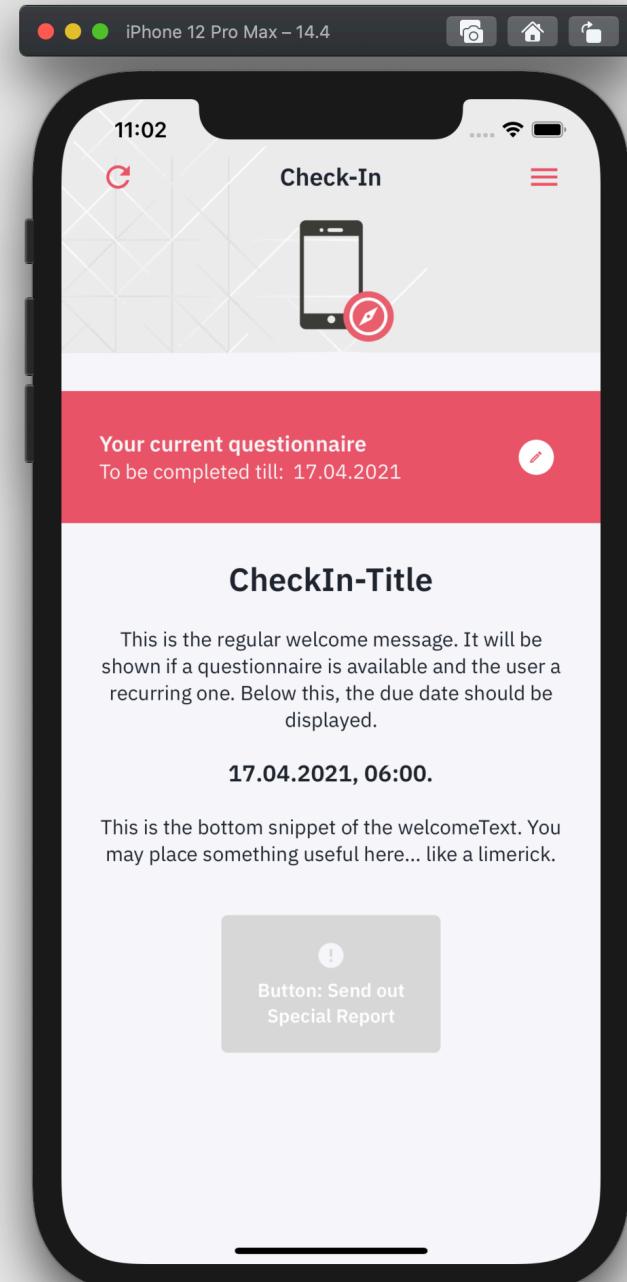
```

- The received user contains additional information about the current state of the profile.
- This includes the ID of the current questionnaire, the due- and start-date of the current interval.
- This data is used to procure the questionnaire or check if an already downloaded one is still valid.
- Data necessary to initiate the Push Service is also transmitted.

# Features

## Questionnaire Retrieval

- The application will download a questionnaire as soon as an updated user suggests that a new one is available.
- When restarting the app, the questionnaire (and the given answers) of the last active user will be loaded from the local storage.
- Should the ID of that questionnaire not match the one transmitted with the user object, the local one will be deleted and a new one from the backend will be procured.
- The user will always be informed when a given questionnaire is due as well when the next one is available as this information is displayed on the **CheckIn-Screen**.



# Features

## Questionnaire Retrieval

- Two objects are generated within the CheckIn-State that hold are used to work with the received questionnaire
- The `categories` array holds the top-level questions of the questionnaire
- The `questionnaireItemMap` object holds the content and the state of every single question provided by the questionnaire

```

▼ categories: Array(3)
  ▷ 0: {linkId: "0", text: "Slider Test 2", type: "group", required: true, item: Array(2)}
  ▷ 1: {linkId: "2", text: "Kommunikation", type: "group", required: true, item: Array(4)}
  ▷ 2: {linkId: "3", text: "Schlafen", type: "group", required: true, item: Array(1)}

▼ questionnaireItemMap:
  ▷ 0: {linkId: "0", done: false, answer: null, text: "Slider Test 2", type: "group", ...}
  ▷ 0.1: {linkId: "0.1", done: false, answer: null, text: "Wollen sie den Snap Slider sehen?", type: "choice", ...}
  ▷ 0.1.1: {linkId: "0.1.1", done: false, answer: null, text: "Snap Slider", type: "integer", ...}
  ▷ 0.2: {linkId: "0.2", done: false, answer: null, text: "Wollen sie den Smooth Slider sehen?", type: "choice", ...}
  ▷ 0.2.1: {linkId: "0.2.1", done: false, answer: null, text: "Smooth Slider", type: "integer", ...}
  ▷ 2: {linkId: "2", done: false, answer: null, text: "Kommunikation", type: "group", ...}
  ▷ 2.1: {linkId: "2.1", done: false, answer: null, text: "Haben Sie körperliche oder funktionelle Einschränkungen?", ...}
  ▷ 2.1.1: {linkId: "2.1.1", done: false, answer: null, text: "Was für Einschränkungen haben Sie?", type: "group", ...}
  ▷ 2.1.1.1: {linkId: "2.1.1.1", done: false, answer: null, text: "Sehschwäche", type: "boolean", ...}
  ▷ 2.1.1.1.1: {linkId: "2.1.1.1.1", done: false, answer: null, text: "Blind", type: "boolean", ...}
  ▷ 2.1.1.1.2: {linkId: "2.1.1.1.2", done: false, answer: null, text: "Auf welcher Seite haben Sie diese Einschränkung?", ...}
  ▷ 2.1.1.1.2.1: {linkId: "2.1.1.1.2.1", done: false, answer: null, text: "Links", type: "boolean", ...}
  ▷ 2.1.1.1.2.2: {linkId: "2.1.1.1.2.2", done: false, answer: null, text: "Rechts", type: "boolean", ...}
  ▷ 2.1.1.2: {linkId: "2.1.1.2", done: false, answer: null, text: "Hörschwäche", type: "boolean", ...}
  ▷ 2.1.1.2.1: {linkId: "2.1.1.2.1", done: false, answer: null, text: "Taub", type: "boolean", ...}
  ▷ 2.1.1.2.2: {linkId: "2.1.1.2.2", done: false, answer: null, text: "Auf welcher Seite haben Sie diese Einschränkung?", ...}
  ▷ 2.1.1.2.2.1: {linkId: "2.1.1.2.2.1", done: false, answer: null, text: "Links", type: "boolean", ...}
  ▷ 2.1.1.2.2.2: {linkId: "2.1.1.2.2.2", done: false, answer: null, text: "Rechts", type: "boolean", ...}
  ▷ 2.1.1.3: {linkId: "2.1.1.3", done: false, answer: null, text: "Sprachstörungen", type: "boolean", ...}
  ▷ 2.1.1.3.1: {linkId: "2.1.1.3.1", done: false, answer: null, text: "Stottern", type: "boolean", ...}
  ▷ 2.1.1.3.2: {linkId: "2.1.1.3.2", done: false, answer: null, text: "Stumm", type: "boolean", ...}
  ▷ 2.1.1.4: {linkId: "2.1.1.4", done: false, answer: null, text: "Wortfindungsstörungen", type: "boolean", ...}
  ▷ 2.1.1.4.1: {linkId: "2.1.1.4.1", done: false, answer: null, text: "Verwaschene Sprache", type: "boolean", ...}
  ▷ 2.1.1.4.2: {linkId: "2.1.1.4.2", done: false, answer: null, text: "Aphasie", type: "boolean", ...}

```

# Features

## Questionnaire Rendering & Interaction

- The application can render a version of the FHIR Questionnaire standard that obliged with a certain ruleset for numbering items and a few other contextual additions.
- The rendering is done in a modal that separates topics by id and that way provides the questions on several pages - for the convenience of the user.
- The questionnaire as a whole, as well as single categories and single questions, are constantly checked for their completion state. This is visible due to several colors used in the application.
- The state of the questionnaire, as well as the state of single questions, can be accessed through the `questionnaireItemMap` object, provided by the `CheckIn-State`.

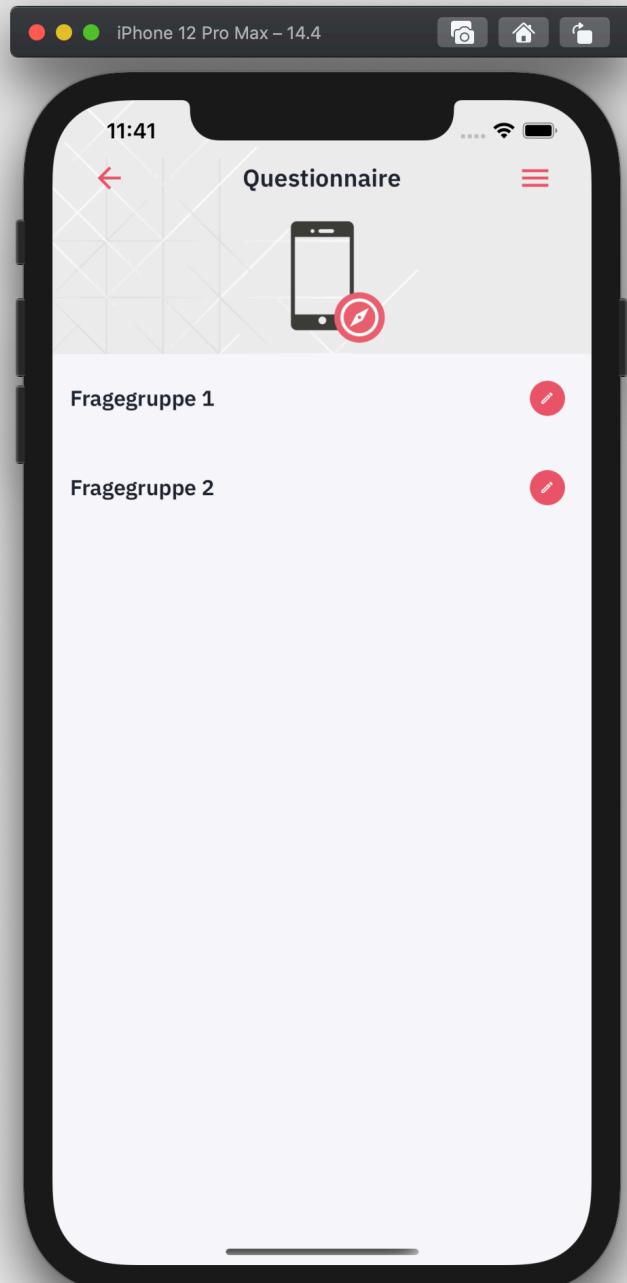


# Features

## Questionnaire Rendering & Interaction

- First level items are being displayed as the main categories
- Subsequent items need to be numbered in a specific way to group them in to separate modal pages

```
"publisher": "IBM",
"purpose": "Abbildung der aktuell in der von IBM entwickelten App unterstützten FHIR Funktionalitäten",
"item": [
    {
        "linkId": "1",
        "text": "Fragegruppe 1",
        "type": "group",
        "required": true,
        "item": [...]
    },
    {
        "linkId": "2",
        "text": "Fragegruppe 2",
        "type": "group",
        "required": true,
        "item": [...]
    }
]
```



IBM®

# Features

## Questionnaire Rendering & Interaction

```
{  
    "linkId": "1.15",  
    "text": "Bedingte Abfrage mit answerDecimal",  
    "type": "group",  
    "required": true,  
    "item": [  
        {  
            "linkId": "1.15.1",  
            "text": "Abfrage Dezimalzahl (erwartet = 1.5)",  
            "type": "decimal",  
            "required": true  
        },  
        {  
            "linkId": "1.15.2",  
            "text": "Diese Frage wird nur bei erwarteter Eingabe angezeigt",  
            "type": "string",  
            "required": true,  
            "enableWhen": [  
                {  
                    "question": "1.15.1",  
                    "operator": "=",  
                    "answerDecimal": 1.5  
                }  
            ]  
        }  
    ],  
},  
}
```



# Features

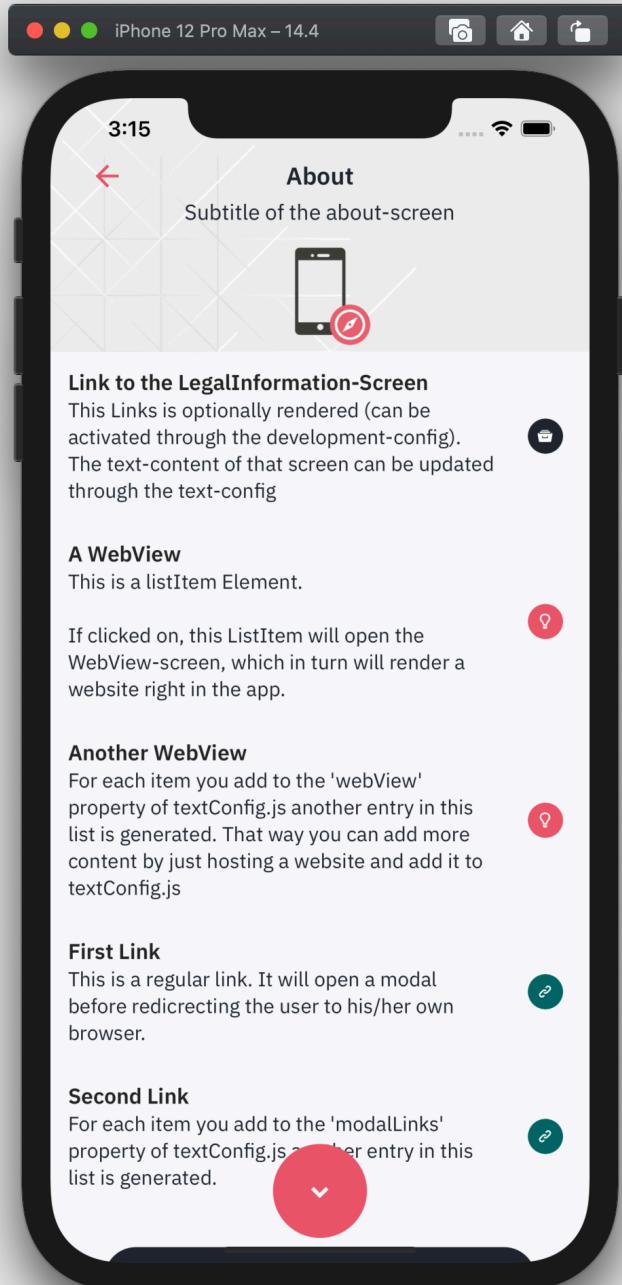
## Questionnaire Response Encryption & Transmission

- A completed questionnaire can be sent to the backend.
- Based on a predefined ruleset (provided by the backend or through a local definition) a few parameters are determined and after that the Questionnaire Response is created and encrypted.
- This encrypted response, plus the ascertained parameters are then send to the backend.
- The parameters are used to determine the update values for the user (as the response is now encrypted and not accessible for the backend) - Meaning when the next questionnaire will be available, what ID that questionnaire will have, etc.

# Features

## Displaying Content

- The burger menu on the top right will open the About-Screen.
- This Screen contains a list of various entries:
  - **Link to the Legal Information Screen**
    - Holds static text, provided by textConfig.js.
    - Used to display legal information
  - **Webview Links**
    - Weblink that will open in a Webview within the app
    - The Webview is located on the WebView-Screen
    - **These links are defined in the file textConfig.js**
  - **Browser Links**
    - A regular link that will be opened in the local browser
    - **These links are defined in the file textConfig.js**





GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung

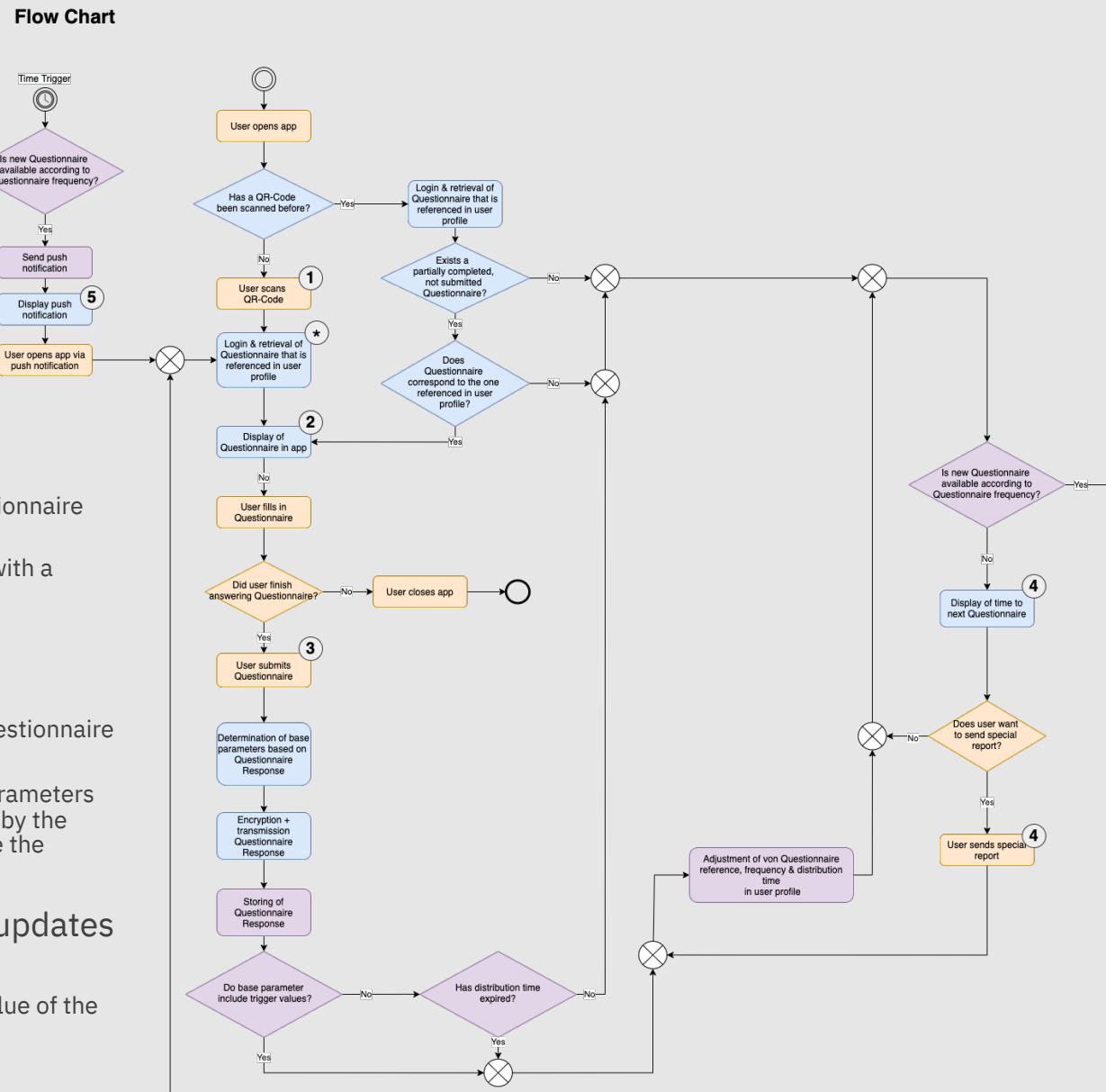
IBM®

# The App Flow

# App Flow

## An Overview

1. The user triggers an update, by:
  - Logging in / Opening the app
  - Manual User Refresh
  - Sending out a Questionnaire Response / Sending out a report
2. The app loads the questionnaire (if available), either from:
  - The backend (should there be no locally persisted questionnaire whose ID matched the currently active one)
  - The local storage (in case there is one locally persisted with a matching ID)
3. The user completes the questionnaire and sends it to backend
  - Alternatively: The user sends out a report (in case no questionnaire is available)
  - If the user sends out a questionnaire response, a few parameters are determined (based on a predefined ruleset provided by the backend or through a local definition) and sent alongside the encrypted questionnaire response.
4. The backend persists the received data and updates the user.
  - The user update is based on various factors, like the value of the additional parameters, the currently active interval, etc.

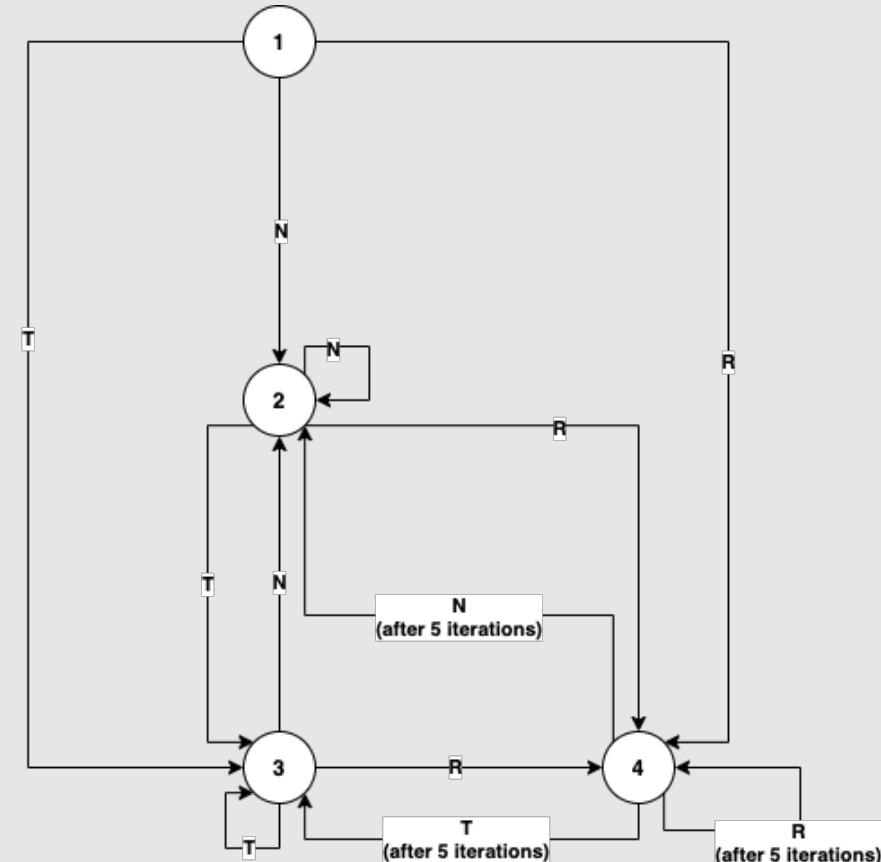


# App Flow Updating the User (server side)

This graph displays how the backend determines what questionnaire the user will receive next.

- 1 Questionnaire 1, Starting Point, will be sent out once per user
- 2 Questionnaire 2, default, 7 days interval
- 3 Questionnaire 3, after hitting a Trigger, 2 days interval
- 4 Questionnaire 4, after sending out a report, 2 days interval at 5 iterations

N	Neutral (no Trigger activated, next questionnaire: 2)
T	Trigger (trigger was activated, next questionnaire: 3)
R	Report (report was sent out, next questionnaire: 4; for 5 iterations, after that: reassessment)



# NUM-App: Configuration files & options

## textConfig.js

*(defines strings and entries on the About-Screen)*

```
/** strings for accessibility features (i.e. VoiceOver) */
"accessibility":{

  "types": {
    "button": "button",
    "header": "header",
  },

  "refresh": "refresh",
  "refreshHint": "Re-fetches the questionnaire and status. The current page will be",

  "back": "back",
  "backHint": "Jumps to previous page.",

  "close": "close",
  "closeHint": "Closes current page or modal.",

  "accept": "accept",
  "cancel": "cancel",
  "acceptHint": "Switches to default browser application, and opens link there.",

  "menu": "Additional Information",
  "menuHint": "Opens new page with additional informations and links.",

  "logoutHint": "Starts logout process and redirects to login page.",
  "loginHint": "Starts login process and redirects to QRCode scanner.",

  "questionnaire": {
    "middleButtonFinished": "Next question, question is answered.",
    "middleButtonUnfinished": "Next question, even though question is not yet ans",
    "middleButtonHint": "Jumps to the next questions, or closes the modal if this

    "rightButtonHint": "Closes the modal.",
    "leftButtonHint": "Jumps to previous question.",
    "alertButtonHint": "Closes the alert.",

    "questionnaireCellHint": "Opens new page to see the questionnaire's categorie",
    "categoryCellHint": "Opens modal to fill out this questionnaire entry.",
    "category": "This Category",
    "questionnaire": "This questionnaire",
    "notstarted": "has not been answered yet.",
    "notfinished": "has already been started, but isn't completely filled in yet.",
    "finished": "is fully complete."
  }
}
```

## theme.js

*(provides global style options)*

```
const defaultUiSettings = {
  /* set to TRUE if you want ot use your own logo in the banner-component. for this to
   * work, you need to place a new logo.png file into 'src/CUSTOMIZATION/images'
   */
  useCustomLogo: false,

  /* set to TRUE if the banner component should use a full-width background, that is
   * behind the app-logo. if set to false, the banner will display its own background
   * the colors for the banner can be chaged through the theme
   */
  useBannerBackground: true,

  /* set to TRUE if you want ot use your own logo in the banner-component. for this t
   * work, you need to place a new defaultLogoBackground.png file into 'src/CUSTOMIZAT
   */
  useCustomLogoBackground: false,

  /* ui-option
   * allows the use of an scrollwrapper that permanently shows the scroll indicator
   (this is only used on the screens, not in the modals)
   */
  allowScrollIndicators: true
}

const defaultColors = {
  /* the primary color of the scheme. most buttons are of this color by default, as i
  primary: '#1F2532',
  /* highlighting for a not fully completed questionnaire (yellow-ish by default) */
  secondary: '#FFC107',
  /* highlighting for completed questionnaires, as well as some icons (green-ish by d
  success: '#009684',
  /* mostly used for "abort"-buttons and error messages (red-ish by default) */
  alert: '#EB576A',
  /* the color primarily used for disabled elements */
  accent1: '#919191',
}
```

## appConfig.js

*(impacts the functionality of the app)*

```
/*
 * this object contains the configuration for the application. some options are only valid in
 * development environment.
 */
const conf = {

  // DEVELOPMENT OPTIONS
  /*-----*/
  /* these options are meant for development only.

  /** dev-option:
   * skips the QR login after 500ms */
  automateQRLogin: __DEV__ && true,
  /* dev-option:
   * user-id for automated login */
  automateQRLoginUserId: {'AppName': 'COMPASS', 'AppID': '7bfc3b07-a97d-4e11-8ac6-b970c1745476'},

  /** dev-option:
   * shows a button to erase all data (in the about-menu) - dev-only */
  showEraseAll: __DEV__ && true,
  /* dev-option:
   * shows a button to log out the user (in the about-menu) - dev-only */
  showLogout: __DEV__ && true,
  /* dev-option:
   * logs out the response-json parsed as an object in the developer console
   */
  logPureResponse: __DEV__ && false,
  /* dev-option:
   * logs out the response-json in the developer console
   */
  logPureResponseJSON: __DEV__ && false,
  /* dev-option:
   * logs out the encrypted response-json in the developer console
   */
  logEncryptedResponse: __DEV__ && false,
}
```



GEFÖRDERT VOM

Bundesministerium  
für Bildung  
und Forschung

IBM®

# Live Demo

## Editing up some values



netzwerk  
universitäts  
**medizin**

GEFÖRDERT VOM

Bundesministerium  
für Bildung  
und Forschung

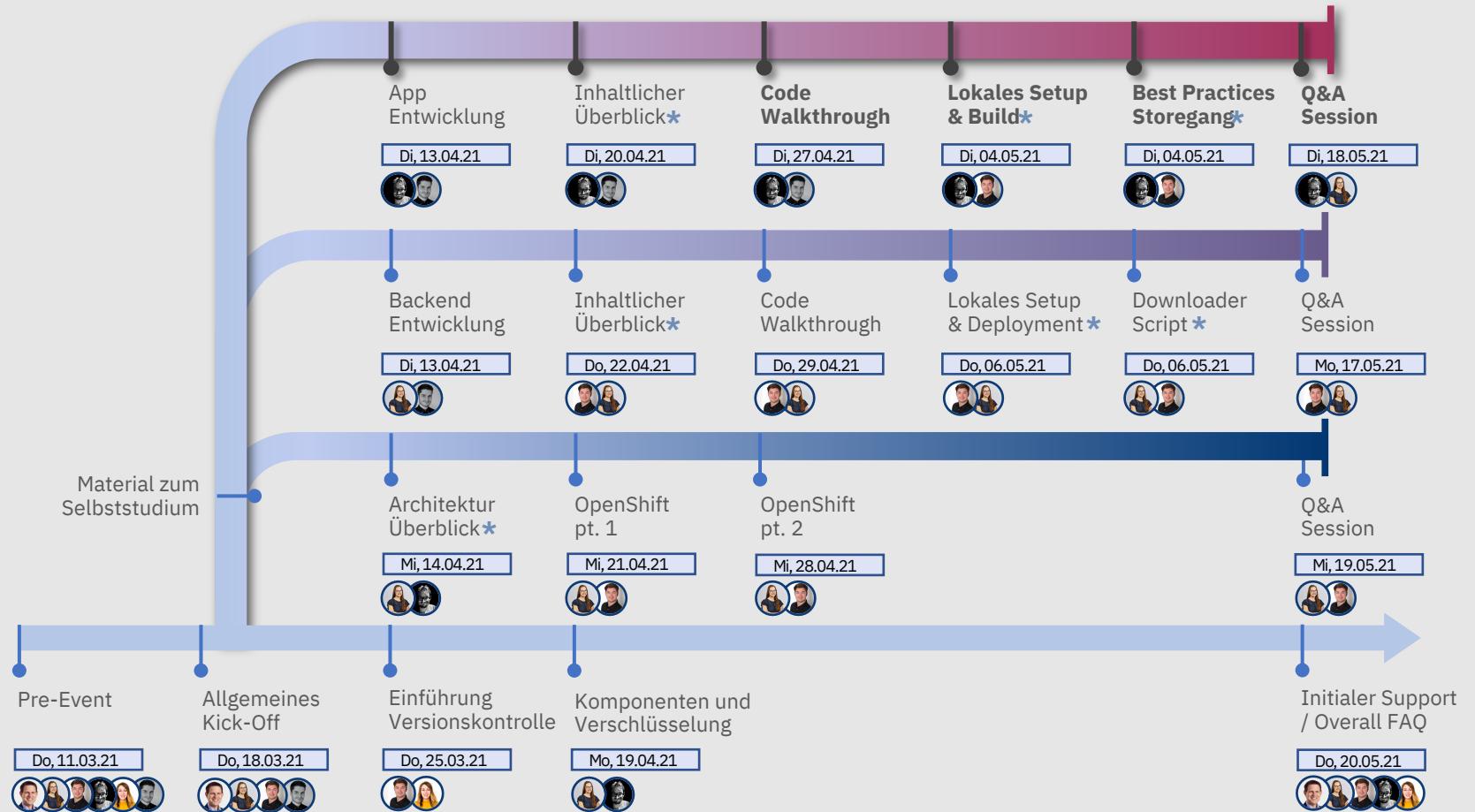
IBM®

# Outlook Future Meetups

Frontend Contributor  
Backend Contributor  
Platform Contributor  
Overall Generalist



# Frontend Contributor Track





GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung

IBM®

# Links

# Links

NUM-App Frontend Repository:

- <https://github.com/NUMde/compass-numapp-frontend>

NUM-App Frontend Documentation:

- <https://github.com/NUMde/compass-numapp-frontend/blob/main/docs/setup>
- <https://github.com/NUMde/compass-numapp-frontend/blob/main/docs/appConfiguration>
- <https://github.com/NUMde/compass-numapp-frontend/blob/main/docs/theming>
- <https://github.com/NUMde/compass-numapp-frontend/blob/main/docs/content>
- <https://github.com/NUMde/compass-numapp-frontend/blob/main/docs/development>
- <https://github.com/NUMde/compass-numapp-frontend/blob/main/docs/build>



GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung

IBM®

# Q&A

# Q&A

*What questions  
do you have?*

