

# **newsR: A Responsive Newsreader Web App**

Elke Perner, Michael Pollak, Oliver Reichmann, Martin Sackl

Institute of Interactive Systems and Data Science (ISDS),  
Graz University of Technology  
A-8010 Graz, Austria

31 Jan 2021

## **Abstract**

Nowadays it is more important than ever to communicate over the internet in order to exchange information with others. One approach therefore is to use newsreaders which became popular in the 1980s. Newsreaders are tools, with which users are able to write news-articles, comment or reply to them - grouped in threads on news servers, the so-called newsgroups. In 2019, the newsreader newsR was developed as a Progressive Web App by students of the University of Technology, Graz. The main aim for this project is the further development of newsR and improving its usage as well as adding new features to it.

© Copyright 2021 by the author(s), except as otherwise noted.

This work is placed under a Creative Commons Attribution 4.0 International (CC BY 4.0) licence.



# Contents

<b>Contents</b>	<b>i</b>
<b>List of Figures</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Types of Newsreaders . . . . .	1
1.2 NewsR . . . . .	1
<b>2 Introduction to newsR</b>	<b>3</b>
2.1 Technology . . . . .	3
2.2 Features and Functionality . . . . .	3
2.2.1 Subscription View . . . . .	3
2.2.2 Newsgroup View . . . . .	5
2.2.3 Posting . . . . .	5
<b>3 Further Development of newsR</b>	<b>7</b>
3.1 Settings and News Server Configuration . . . . .	7
3.2 Newsgroup View . . . . .	8
3.3 Posting . . . . .	8
3.4 Mobile Views . . . . .	9
<b>4 newsR 2.0.0 User Guide</b>	<b>15</b>
4.0.1 Running the Application . . . . .	15
4.0.2 Settings . . . . .	15
4.0.3 Home and Subscription View . . . . .	15
4.0.4 Managing Newsgroups . . . . .	16
4.0.5 Newsgroup View . . . . .	16
4.0.6 Writing Posts and Responses . . . . .	16
<b>5 Future Work</b>	<b>17</b>
<b>Bibliography</b>	<b>19</b>



# List of Figures

1.1	Interface of Thunderbird newsreader . . . . .	2
2.1	Subscription View . . . . .	4
2.2	View of all newsgroups . . . . .	4
2.3	Newsgroup View . . . . .	5
2.4	Posting . . . . .	6
3.1	Settings View . . . . .	8
3.2	New Newsgroup View . . . . .	9
3.3	Messages in Columns . . . . .	9
3.4	New Posting View . . . . .	10
3.5	Thread Content . . . . .	10
3.6	The mobile home (or subscription) view. . . . .	11
3.7	The mobile settings view. . . . .	11
3.8	The mobile view of a subscribed newsgroup showing all posts. . . . .	12
3.9	The mobile posting view. . . . .	12
3.10	The mobile message view of a thread. . . . .	13
5.1	Threading . . . . .	18



# Chapter 1

## Introduction

Newsreaders are important tools in order to exchange information and knowledge over a platform with others. In general, newsreaders (or sometimes called news aggregators) are programs which allow their users to read, write and edit news-articles in so-called newsgroups over the Usenet. The Usenet is a network of newsgroups and is an individual service of the internet besides the World Wide Web. Each newsreader is connected to one (or multiple) news server via the Network News Transfer Protocol, or short NNTP. The original purpose of newsreaders is messaging, which was popular via newsreaders in the 1980s. Nowadays, the most common use of newsreaders is exchanging binary files.

### 1.1 Types of Newsreaders

There are several different newsreader in form of graphical user interfaces available nowadays, where some of them are used in email clients or web browsers, but also as stand-alone applications. Some of the most popular newsreaders are **Mozilla Thunderbird**, **Xnews**, **Forté Agent**, **Momentum** and **NewsBin**. Figure 1.1 shows the interface of the newsreader Thunderbird. Theses newsreaders mostly differ in design and functionality, but there are also several different types of newsreader clients:

- Text newsreaders
- Traditional newsreaders
- Binary grabber/plucker
- NZB downloader
- Binary posting client
- Combination client

### 1.2 NewsR

The newsreader program **newsR** is a responsive web application implemented as a progressive web application, making it possible to use it as a native app on several devices. The newsR project started in late 2019 and is implemented in React including TypeScript, SCSS and Newsie for the NNTP communication. Although the application was finished as a working prototype in 2020, there are still some issues, design decisions and missing functionalities which are mandatory for a goo newsreader. The goal for this project is the further development and improvement of newsR in order to gain a user-friendly and well-working newsreader.

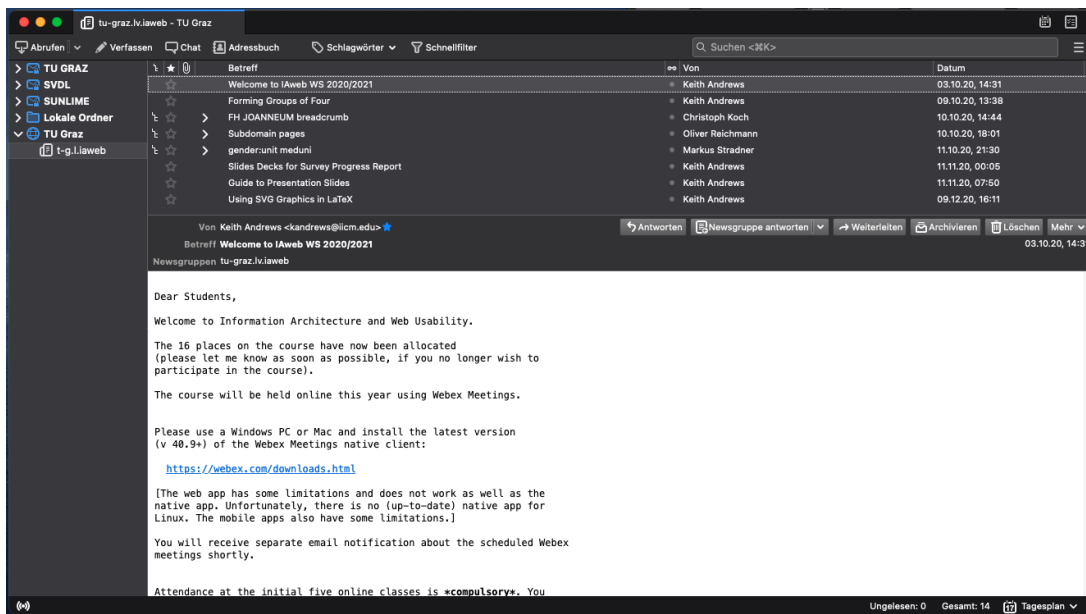


Figure 1.1: Thunderbird. [Image taken by the authors.]



## Chapter 2

# Introduction to newsR

This chapter will introduce and describe the newsreader **newsR** in a brief, but detailed enough way in order to gain an overview of its functionality. The deployed version of newsR 0.1.0 is provided on GitHub.io [Paul Ganster, Peter Grassberger, Magdalena Mayerhofer, Ana Lopez Camarero 2020].

### 2.1 Technology

NewsR is implemented as a progressive web application and uses the following technologies for its implementation:

- TypeScript
- React
- NNTP
- SCSS
- Progressive Web App (PWA)
- Web Sockets

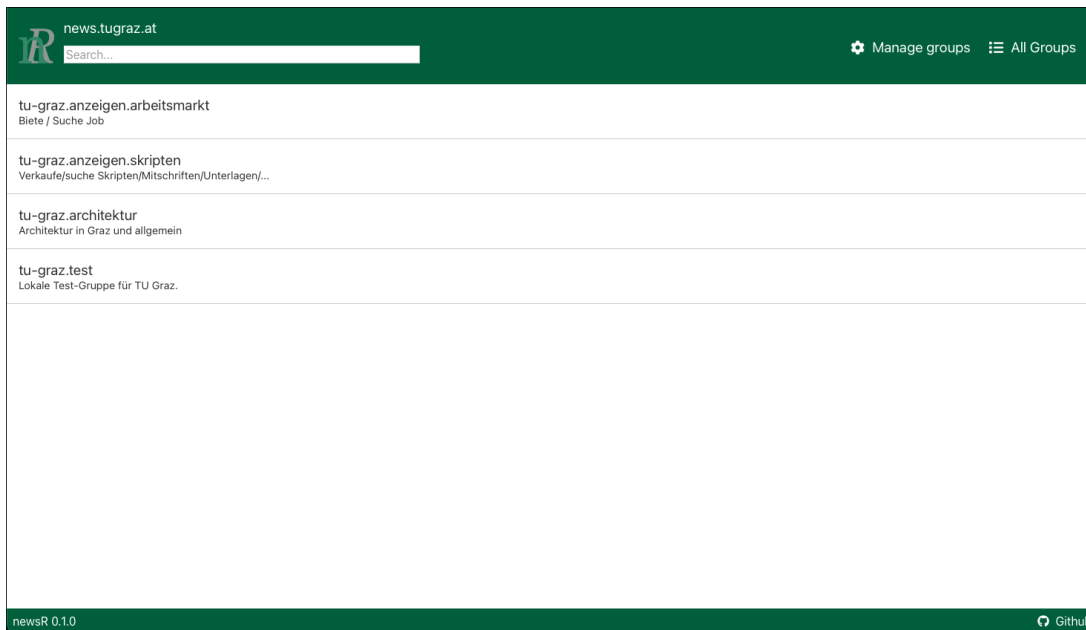
TypeScript is an open-source programming language, React (or ReactJS) is a JavaScript framework, used for creating user interfaces and SCSS is the metalanguage used for adding style in form of CSS (cascading style sheets) to the interface. NNTP is the communication language between clients (e.g. newsreaders) and news servers. Progressive web apps are basically web applications, developed for the native use on e.g. mobile devices like Android Smartphones or iPhones. Web Sockets are used for communicating between the react client and the server via NNTP since browsers can not open a TCP connection on their own.

### 2.2 Features and Functionality

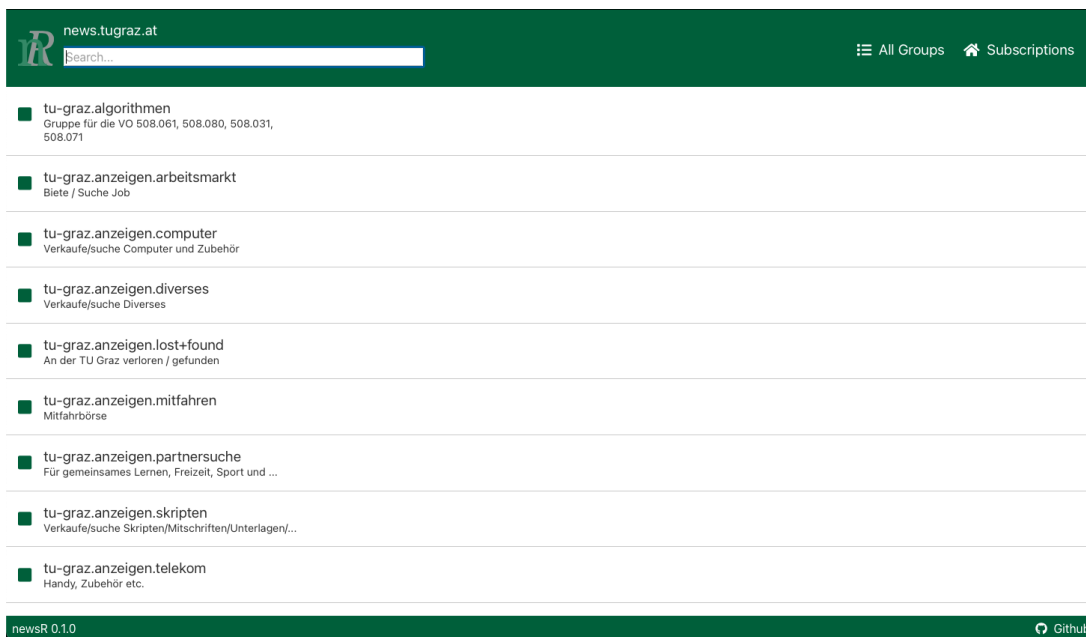
There are several features implemented in newsR, which are mentioned in this section.

#### 2.2.1 Subscription View

The first view a user sees when starting the application shows all subscribed newsgroups of the news server in a list-view. Users are able to toggle between a list of all available newsgroups and a list of subscribed newsgroups. When showing the complete newsgroup list, users can select (subscribe) or deselect (unsubscribe) from newsgroup. These informations are stored in the local storage of the browser. Figure 2.1 shows the interface in the subscription view, 2.2 shows the view, in which users can subscribe or unsubscribe to newsgroups.



**Figure 2.1:** Subscription view showing subscribed newsgroups. [Image taken by the authors.]



**Figure 2.2:** The view of all available newsgroups to manage (subscribe/unsubscribe). [Image taken by the authors.]

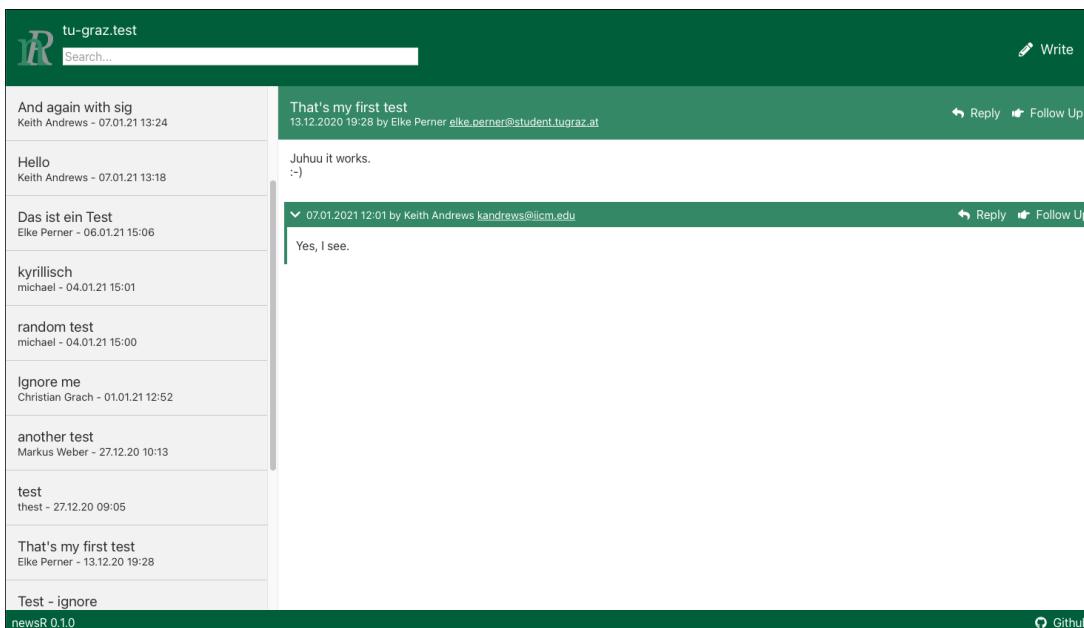


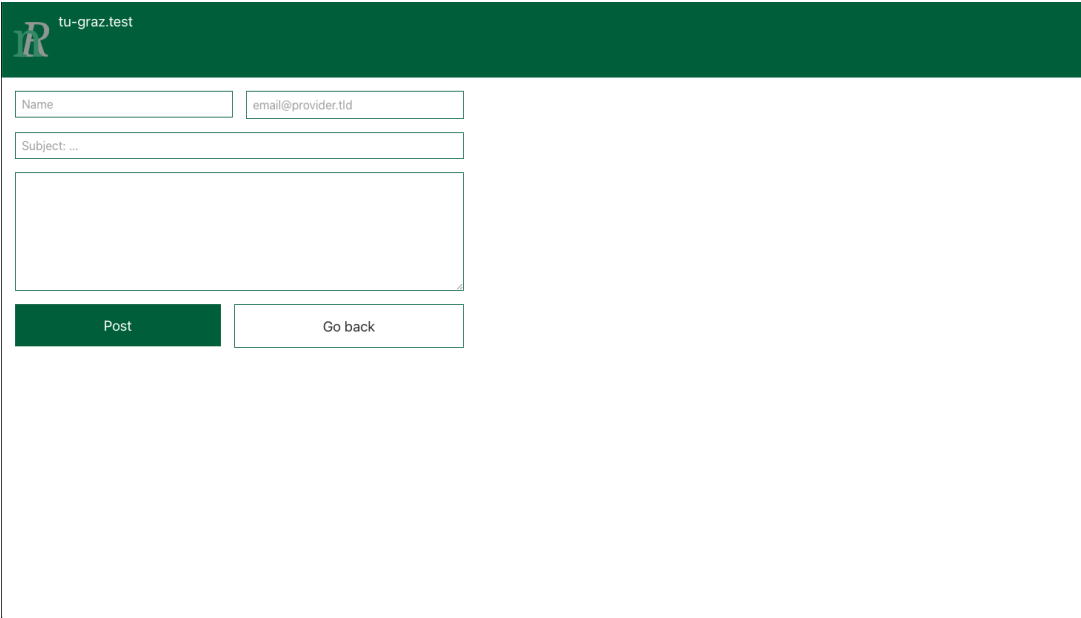
Figure 2.3: The view, in which the user sees the articles of a newsgroup. [Image taken by the authors.]

### 2.2.2 Newsgroup View

A news server has several newsgroups, which are collections of articles/messages and replies of users. The newsgroup view shows a left-side panel including a list of root articles of the thread and a bigger main area, in which the content and replies of those articles are shown (when clicking/selecting an article). Figure 2.3 shows this view.

### 2.2.3 Posting

To write a new article in a newsgroup users can click on *Write*, which opens another view including a form. This form includes the name and e-mail address of the user, a subject field and the content text area, as well as a *Post* and *Go back* button. Figure 2.4 shows the posting view.



The image shows a web form for posting an article to a newsgroup. The form is contained within a white rectangular area with a thin black border. At the top left of this area is a dark green header bar containing a white 'R' logo and the text 'tu-graz.test'. Below the header, the form consists of several input fields and two buttons. The first row contains two text input fields: the first is labeled 'Name' and the second contains the placeholder text 'email@provider.tld'. The second row contains a single text input field labeled 'Subject: ...'. The third row contains a large, empty text area for the article content. At the bottom of the form, there are two buttons: a dark green button labeled 'Post' and a white button with a black border labeled 'Go back'.

**Figure 2.4:** Form view to post an article to a newsgroup. [Image taken by the authors.]

## Chapter 3

# Further Development of newsR

Version 0.1.0 of the newsR PWA already implemented the most important functions in order to get a working newsreader. However, there are several features and design decisions, which have to be changed to gain a more user-friendly and better newsreader. The following list contains all necessary changes made for the new **newsR version 2.0.0**:

- Upgrading software to newest versions (e.g. React and components)
- Highlighting selected message in newsgroups
- GUI dialogue for news server configuration
- Rename button names and text field placeholders
- Table view with columns and specialised wide screen display (like Thunderbird)
- Sorting functionality in column headers
- Quoting on replies and follow-ups
- Fixing charset troubles Unicode, ASCII, etc.
- Changing the page logic
- Settings Page
- Indication of messages with attachments (by their size)
- Adding signatures

A part of the items in this list are self-explaining and do not need a deeper explanation of the implementation, whereas the following sections will describe the most important changes to newsR version 2.0.0 in detail. The old project was updated, since of course the use of the newest versions of all technologies, like react 16.14, was also an improvement task. The only library which could not be updated was newsie, the newest version 1.2.0 did not run.

### 3.1 Settings and News Server Configuration

Since the application allows users to connect to different news servers, it is sufficient to enable the option to set or edit server settings for the user. Therefore a settings view has been created and is shown to new users, who have not set server settings. These settings include the **server URL**, **server port** and a **group prefix**. By default, these are set to the connection parameters from the University of Technology Graz, so that new users only have to click the *Save* button in order to confirm those settings. This view is also

**Figure 3.1:** Settings view of the server connection settings. [Image taken by the authors.]

reachable from the main menu at any time for a user by clicking the new *Settings* menu item. Figure 3.1 shows the settings view.

The second part of the settings view are the user details, where a user has to enter his full name and a valid e-mail address. The third field is optional, which is the signature. This signature will be added at the bottom of messages the user writes automatically.

## 3.2 Newsgroup View

The newsgroup view has changed in several parts. The left-side panel now holds the subscribed newsgroups, not the included articles. The currently selected newsgroup is visually marked, so that users always know in which newsgroup they are.

The second panel on the right is now vertically divided into two parts, where the top container shows all articles. Displayed in columns, it shows the subject, author, date, size (in bytes) and the indication if the message contains attachments, of a post. Additionally, it allows a user to sort the posts by each column, simply by clicking on the column title. Here again, the selected post is highlighted. Figure 3.3 shows the top panel including the new columns, sorted by attachment indication.

Its content (original post and all replies) is shown in the second container underneath. On the top of the bottom panel is a message header, containing the author's name and e-mail address, the date and time of the message as well as a *Reply-* and *Follow up-* button. Clicking on the first button will open a mailing program and the user can write directly to the author, whereas the second button allows the user to reply directly in the current thread. This new layout leads to an easy and user-friendly use, see Figure 3.2 for the updated newsgroup view.

## 3.3 Posting

The posting view, in which users actually write messages, also changed. First of all, the user's name and e-mail address are automatically provided in the according fields. The user's signature is automatically included in the content text area, separated by an empty line followed by two dashes and a space like it is standard on news servers.

If the user is replying to another question, the subject is included with a *Re:* at the beginning and the

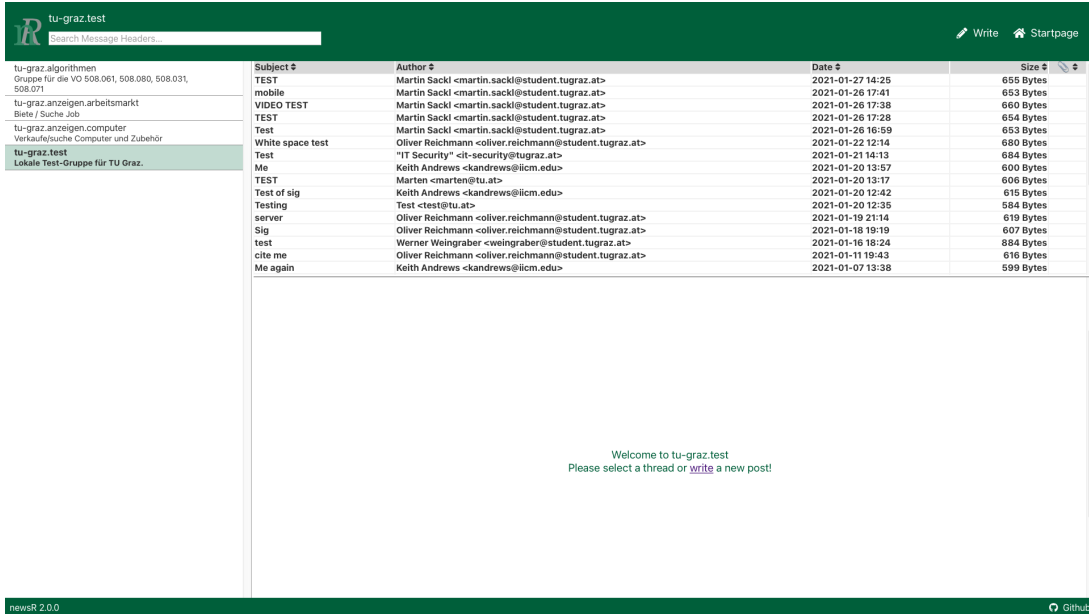


Figure 3.2: The updated newsgroup view with new structure. [Image taken by the authors.]

Subject	Author	Date	Size
Testing	Christian Grach <"christian.grach@alumni.tugraz.at">	2020-12-12 15:59	96658 Bytes
TEST	Martin Sackl <martin.sackl@student.tugraz.at>	2021-01-27 14:25	655 Bytes
mobile	Martin Sackl <martin.sackl@student.tugraz.at>	2021-01-26 17:41	653 Bytes
VIDEO TEST	Martin Sackl <martin.sackl@student.tugraz.at>	2021-01-26 17:38	660 Bytes
TEST	Martin Sackl <martin.sackl@student.tugraz.at>	2021-01-26 17:28	654 Bytes
Test	Martin Sackl <martin.sackl@student.tugraz.at>	2021-01-26 16:59	653 Bytes
White space test	Oliver Reichmann <oliver.reichmann@student.tugraz.at>	2021-01-22 12:14	680 Bytes
Test	"IT Security" <it-security@tugraz.at>	2021-01-21 14:13	684 Bytes
Me	Keith Andrews <kandrews@iicm.edu>	2021-01-20 13:57	600 Bytes
TEST	Marten <marten@tu.at>	2021-01-20 13:17	606 Bytes
Test of sig	Keith Andrews <kandrews@iicm.edu>	2021-01-20 12:42	615 Bytes
Testing	Test <test@tu.at>	2021-01-20 12:35	584 Bytes
server	Oliver Reichmann <oliver.reichmann@student.tugraz.at>	2021-01-19 21:14	619 Bytes
Sig	Oliver Reichmann <oliver.reichmann@student.tugraz.at>	2021-01-18 19:19	607 Bytes
test	Werner Weingraber <weingraber@student.tugraz.at>	2021-01-16 18:24	884 Bytes
cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>	2021-01-11 19:43	616 Bytes

Figure 3.3: Messages are now displayed in columns, here sorted by attachment indication. [Image taken by the authors.]

content text area includes the whole message, which the user wants to reply to, each line starting with a >. After the user submits the form by clicking on the send-button, the message will be posted and the user will be automatically taken back to the newsgroup view without any action by the user required. Figure 3.5 shows the bottom content panel in the newsgroup view including one initial post message and a response underneath. The > symbols from the posting form are not displayed in this view, but the message a user replied to is now quoted at the top of the response and marked with a blue left-side vertical bar.

### 3.4 Mobile Views

In order to deliver a responsive mobile view of newsR as well, no changes have been made to the mobile design to provide the same old layout which is better for small screens. In general, the mobile version of the application offers the same functionality as the desktop version. See the following figures for the mobile views of newsR. Figure 3.6 shows the home view, including all subscribed newsgroups, Figure 3.7 shows the settings view, Figure 3.8 shows a single newsgroup by displaying all posts of the selected newsgroup. Figure 3.9 shows the posting view and Figure 3.10 shows the mobile messages view.

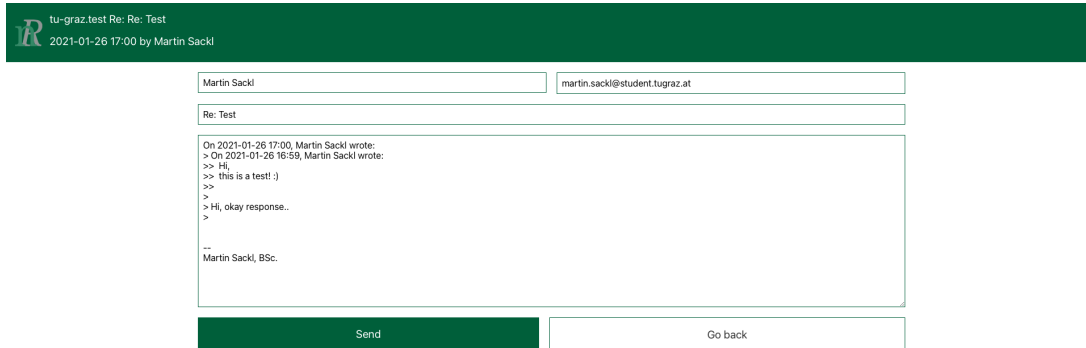


Figure 3.4: The updated posting view. [Image taken by the authors.]

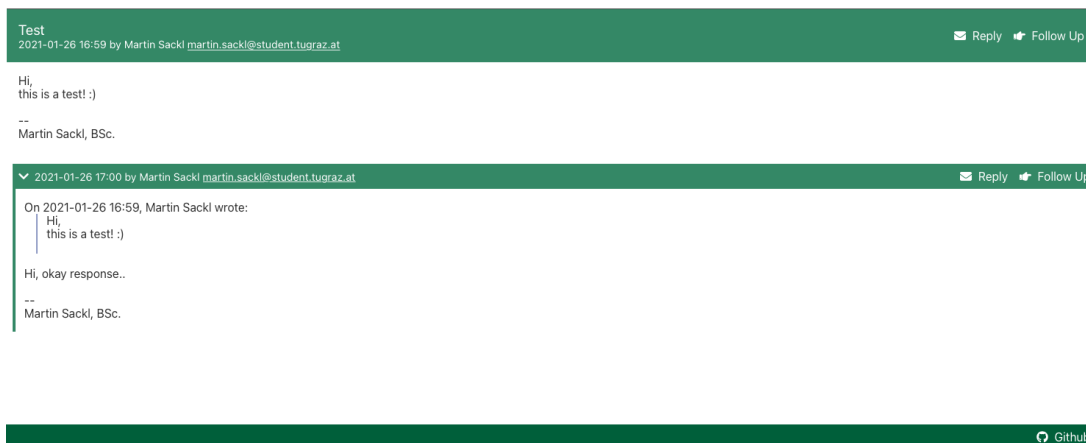
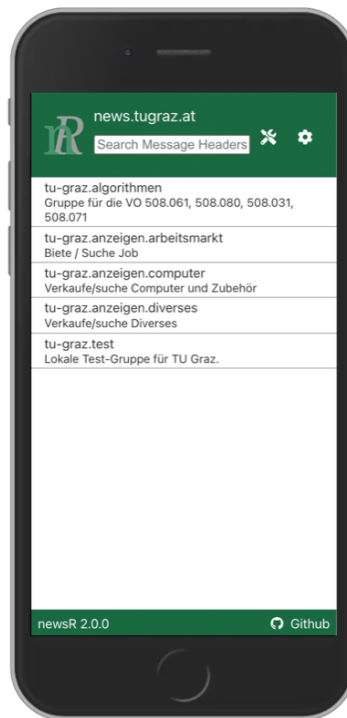
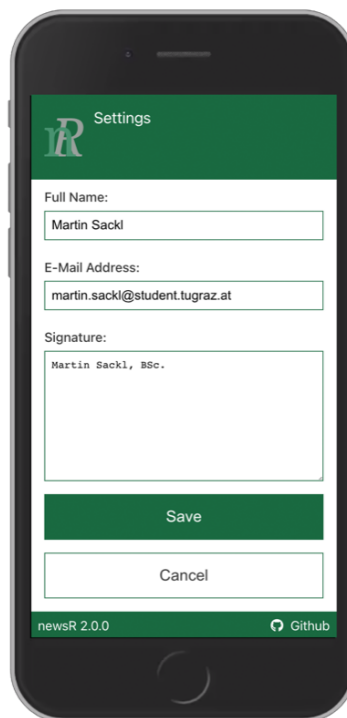


Figure 3.5: Content of a thread including the initial post and a response. [Image taken by the authors.]





**Figure 3.6:** The mobile home (or subscription) view.



**Figure 3.7:** The mobile settings view.

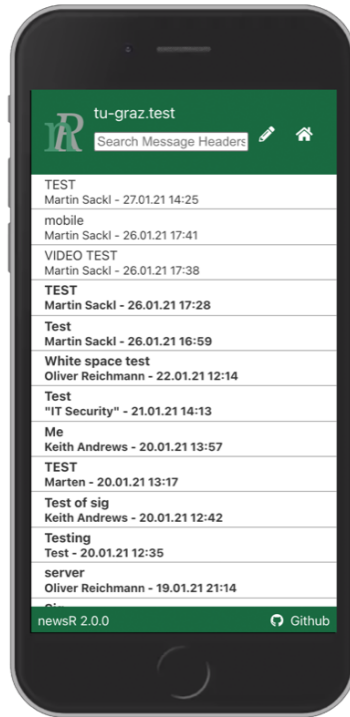


Figure 3.8: The mobile view of a subscribed newsgroup showing all posts.

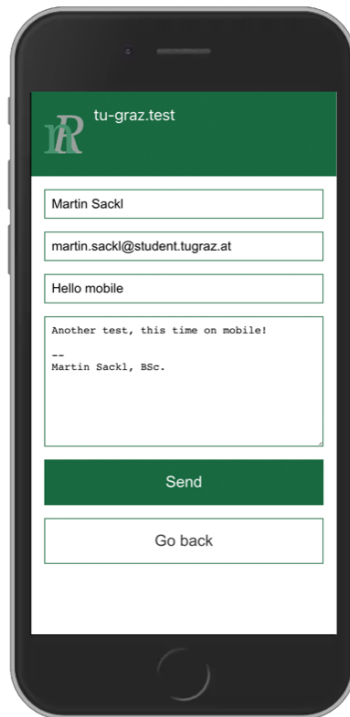
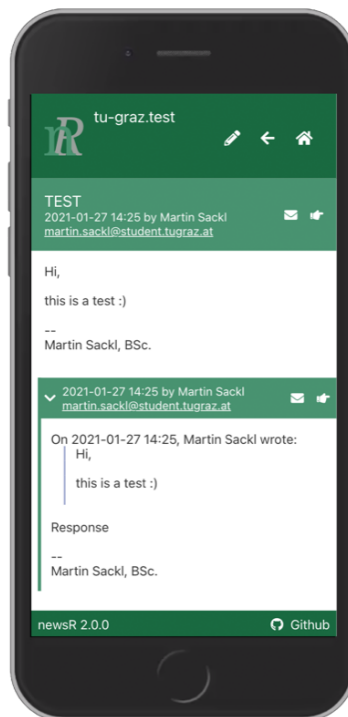


Figure 3.9: The mobile posting view.



**Figure 3.10:** The mobile message view of a thread.



## Chapter 4

# newsR 2.0.0 User Guide

This section will guide through newsR 2.0.0 and describe all user operations and functionality in order to connect to a news server, subscribe/unsubscribe to newsgroups and write posts and responses. A showcase video of this user guide can be found on Youtube [Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl 2020c].

### 4.0.1 Running the Application

For this user guide, one possibility to use newsR is to navigate to the deployed version from the authors on GitHub [Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl 2020a]. But there is also the option to download the public repository [Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl 2020b] from GitHub and run the application locally. Therefore, a user has to clone the repository and afterwards start the server as well as the client.

The server communicates with the news server via NNTP. In the *server* directory, run `npm ci` to install all necessary dependencies. Afterwards, run `npm run start` in order to start the NodeJS server.

The client side communicates with the just started NodeJS server and displays the in earlier sections introduces interface. Again, run `npm ci` to install all necessary dependencies, this time in the *client* folder and run `npm run start` to run the app in development mode. The running application can then be opened in a browser at the address `http://localhost:3000`.

To build the app for production into the *build* directory, run `npm run build`. This correctly bundles React in production mode and optimizes the build for the best performance. The build will be minified and the filenames will include the hashes.

### 4.0.2 Settings

The first thing a user will see is the settings page, shown in Figure 3.1. It is necessary to enter a **server URL**, **server port** and a **server prefix** as well as the users **full name** and a valid **e-mail address**. The **user signature** is optional. By default, the settings are set to the TU Graz news server:

- **Server URL:** news.tugraz.at
- **Server port:** 119
- **Server prefix:** tu-graz\*

### 4.0.3 Home and Subscription View

After entering and saving the settings, the user will be directed to the home view, in which all subscribed newsgroups will be shown. In the navigation bar, there is the newsR logo, which always takes a user back to this view if he or she clicks on it. Next to it there is a search input field, which filters, depending on the current view, either newsgroups or messages. In the top right corner there are two menu items *Settings* and *Manage Groups*. In order to manage subscriptions, a user has to click on *Manage Groups*.

#### 4.0.4 Managing Newsgroups

On the *Manage Groups* page, the user will be shown all available newsgroups from the connected news server, each displayed with a checkbox. This checkbox indicates whether the user has subscribed to a newsgroup or not. By simply clicking on the newsgroup, the checkbox toggles and the user manages his subscription. The search input field in the menu will filter the newsgroups by the input from the user. If a user has finished subscribing, he can go back by clicking on the newsR logo on the top left. The user then will see the subscribed newsgroups and can enter one newsgroup by clicking on it.

#### 4.0.5 Newsgroup View

The most important view is the newsgroup view, shown in Figure 3.2, which shows the user three panels. The left side panel includes all subscribed newsgroup, which the user is able to switch between. When selecting a newsgroup, the top right side panel will display all root messages of this newsgroup, displayed in columns, see Figure 3.3. The user is able to sort by these column, simply by clicking on them. If a message is clicked, the bottom right side panel will show the content of this message as well as all responses.

#### 4.0.6 Writing Posts and Responses

In order to write a new post, a user has to click on the *Write* button. This will open the same new view, when clicking on the *Follow Up* button in the header of a selected message. The user will see a form, already with his name, e-mail address and, if provided, his signature in it. If the user wants to write a new message, he or she has to enter a subject and of course content in the provided input fields. If the user writes a response to another message, the subject is already filled out and the content text area includes the quoted message the user responds to, see Figure 3.4.

## Chapter 5

# Future Work

Due to time limitations, there is still some functionality left, which has not been implemented yet (or which is not finished yet). The following list will show some possible future work or further development.

- **Crossposting**  
It should be possible to write a post to multiple newsgroups simultaneously.
- **User Login**  
A user management with login for users would be helpful, e.g. if servers need authentication or multiple users should be allowed on one device.
- **Multiple Servers**  
Settings to select multiple servers and the possibility to switch between them.
- **Column Chooser**  
Choose, which columns are shown and which are not.
- **Settings**  
Several additional settings like text colors, fonts, etc.
- **Attachment Checkbox**  
Checkboxes, if attachments should be displayed or not. (As a policy TU Graz does not allow attachments in newsgroups)
- **Time Format Selector**  
Possibility to select between different time formats.
- **Instant Reply**  
An instant reply method could be implemented, like a messenger app.

Another feature, which has already started to be implemented, is a better overview for threads similar to the one used in Thunderbird. The idea is to display all responses of a root post in the top panel of the newsgroup view. At the moment, follow-ups are only shown in the bottom panel after the original message. This should be changed to a kind of dropdown, which shows the responses if clicking on the root message. See Figure 5.1 for the wanted structure. If there exists interest in further developing this feature, the start of the development has already been done in the branch *threads* of the repository [Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl 2020b].

Threads ^	Subject ⇅	Author ⇅
	cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>
↳	Re: cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>
↳	Re: cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>
↳	Re: cite me	Keith Andrews <kandrews@iicm.edu>
↳	Re: cite me	Keith Andrews <kandrews@iicm.edu>
↳	Re: cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>
↳	Re: cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>
↳	Re: cite me	Oliver Reichmann <oliver.reichmann@student.tugraz.at>
↳	Re: cite me	Keith Andrews <kandrews@iicm.edu>
	Me again	Keith Andrews <kandrews@iicm.edu>

**Figure 5.1:** [Image taken by the authors.]



# Bibliography

- Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl [2020a]. *newsR 2.0.0 Deployed Online Version on GitHub*. GitHub.io. 27 Jan 2020. <https://olliwho.github.io/newsR> (cited on page 15).
- Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl [2020b]. *newsR 2.0.0 Repository on GitHub*. GitHub. 27 Jan 2020. <https://github.com/olliwho/newsR> (cited on pages 15, 17).
- Oliver Reichmann, Michael Pollak, Elke Perner, Martin Sackl [2020c]. *newsR 2.0.0 Showcase Video on Youtube*. youtube.com. 27 Jan 2020. <https://youtu.be/pia8mcRkYkA> (cited on page 15).
- Paul Ganster, Peter Grassberger, Magdalena Mayerhofer, Ana Lopez Camarero [2020]. *newsR 0.1.0 Repository on GitHub*. GitHub. <https://github.com/Elektropepi/newsR> (cited on page 3).