aitronic

aiBrowser

Edition 17.01.2018

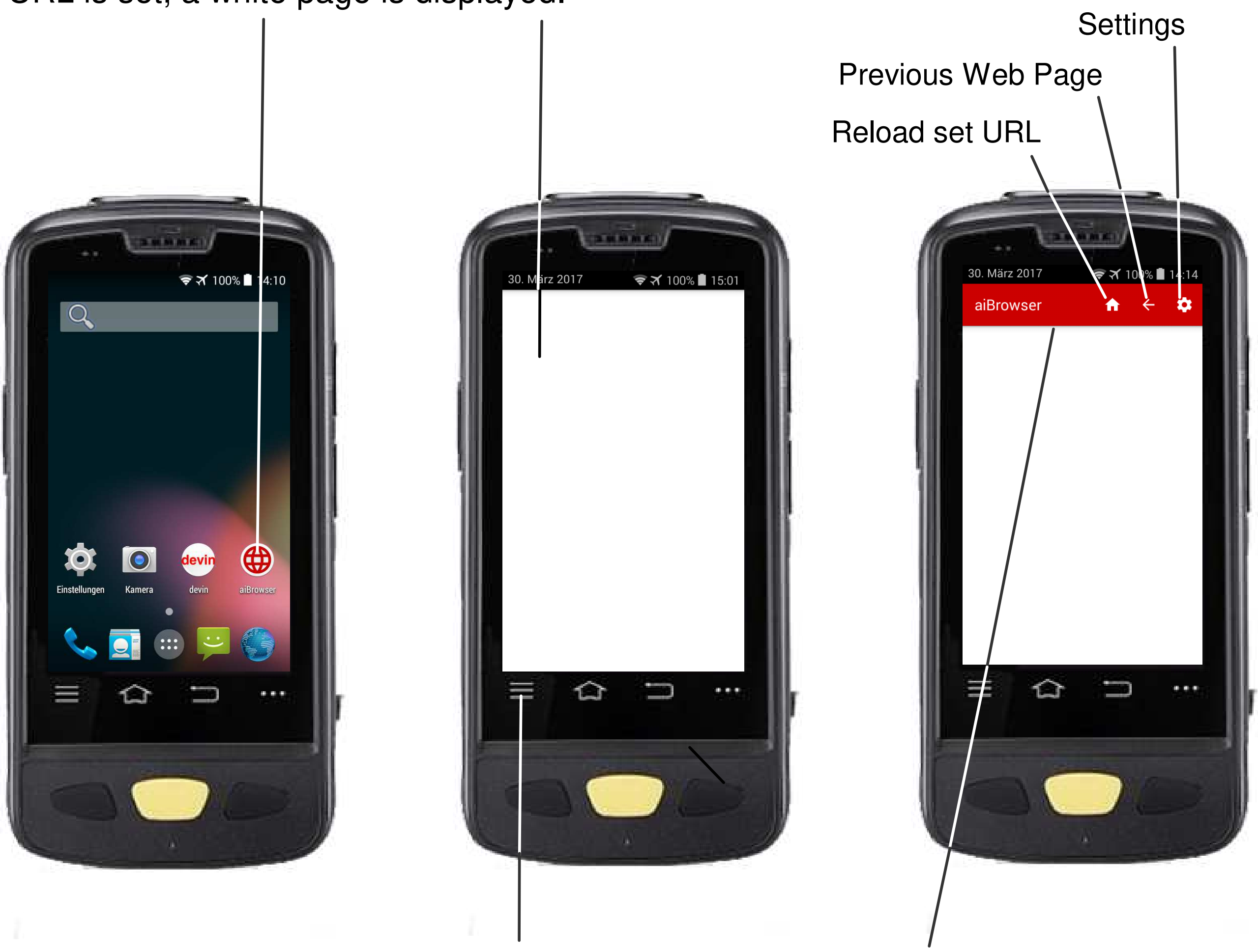
Manual

The Android app **aiBrowser** together with the Android app **devin** enables the data scanned by barcode or RFID readers to be transferred to the input fields of apps and web-based applications in an elegant and simple way. The following overview lists the options and the settings to be made in devin.

Type of application	**Possibilities of barcode/RFID data transfer by appropriate configuration of devin**
Existing app i.e. the source code is not available and therefore there is no way to change the app.	Data transfer must be done in the **keyboard buffer**. **devin** must be configured manually. Further settings such as, e.g. the barcode types to be accepted must be set manually in devin.
Self developed app i.e. the source code is available and therefore the possibility exists to change the app.	The data can be transferred in the **keyboard buffer** (no additional programming required) or via **broadcast**. Broadcast provides e.g. The ability to manipulate or intercept the received data before it is inserted into an input field. **devin** can be configured by the app. For example, The barcode types to be accepted can be set.
Web-based application is executed in a browser by calling a URL.	The **aiBrowser** from aitronic was developed for the barcode/RFID data transfer in web-based applications. Data transfers must be done by **broadcast**. **devin** must be manually configured for this. Further settings such as, e.g. the barcode types to be accepted must be set manually in devin.

Start and Menu Bar

After launching the **aiBrowser**, the web page set under the default URL is displayed. By default, the default URL is `about: blank`, i.e. As long as no other URL is set, a white page is displayed.

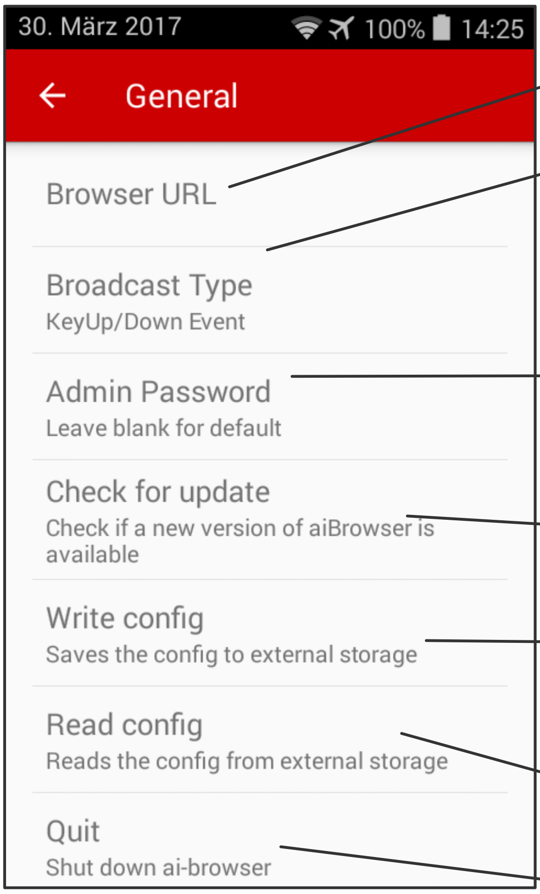


After tapping the hardware button menu (for Android 5, press and hold), the menu bar will appear

Settings

General

Only available after entering Password. The default Password is `ai1500`.



30. März 2017 100% 14:25

General

Browser URL

Broadcast Type
KeyUp/Down Event

Admin Password
Leave blank for default

Check for update
Check if a new version of aiBrowser is available

Write config
Saves the config to external storage

Read config
Reads the config from external storage

Quit
Shut down ai-browser

Default URL that is called when aiBrowser is started.

Here, the administrator can set how scan data from devin is passed to the web page.

Admin password to access these settings. If the admin password is left blank, the default password `ai1500` is active. If the password is forgotten, a master password that is valid only for the day can be requested via the aitronic support. After entering the master passwords, the password is reset to `ai1500`.

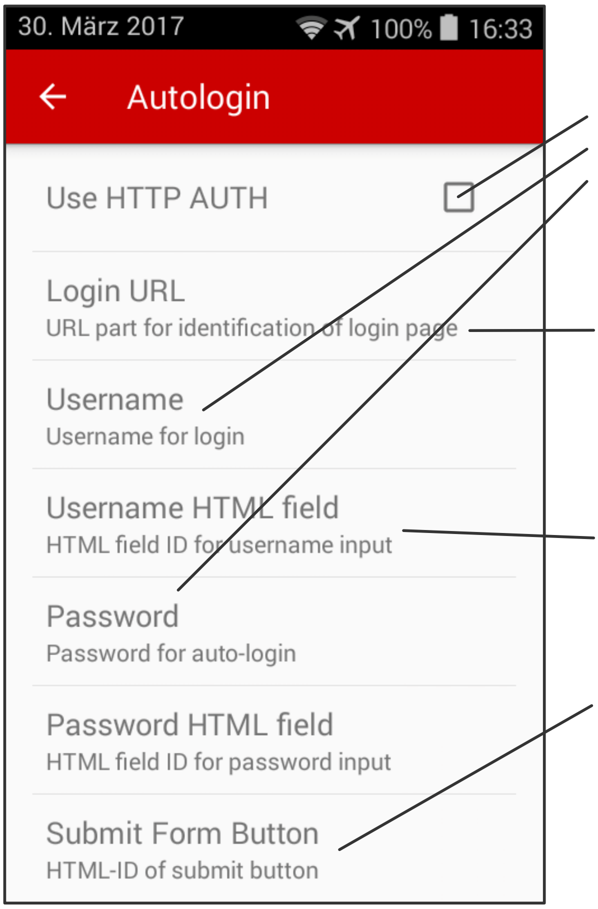
Checks if a new version of the app is available on the server.

Write a config file to `/sdcard/aitronic/aiBrowser/config.ini` for easy distribution of configurations to multiple devices.

Reads the config file at `/sdcard/aitronic/aiBrowser/config.ini`

Ends the app to access the Android system..

Autologin



30. März 2017 100% 16:33

Autologin

Use HTTP AUTH □

Login URL
URL part for identification of login page

Username
Username for login

Username HTML field
HTML field ID for username input

Password
Password for auto-login

Password HTML field
HTML field ID for password input

Submit Form Button
HTML-ID of submit button

If this checkbox is activated, HTTP authentification is used for automatic login. The Username and Password field must be filled. If the checkbox is deactivated, it is possible to fill defined form fields automatically.

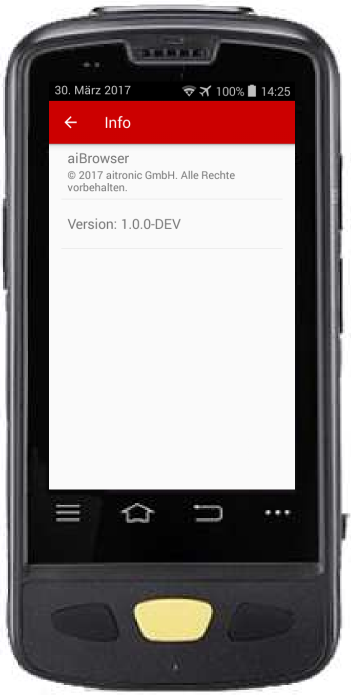
Here, a part of the login URL must be entered so that the aiBrowser can identify the login page on which the login data is to be entered. A short portion of the URL is sufficient, e.g. "SignIn.aspx" or "login.php".

Here the HTML-IDs of the fields for the user name and the password can be entered.

You can specify the HTML ID of the field on which JavaScript-Click is to be executed after completing the Username and Password fields. However, this works only if both Username and Password are stored.

Info

About the copyright and version of the app.



aiBrowser

© 2017 aitronic GmbH. Alle Rechte vorbehalten.

Version: 1.0.0-DEV

Transfer of the Scan Data to the Browser

In order for the scan data from devin to be transferred to the aiBrowser, the devin settings in the "Processing" section must be set to "Broadcast de.aitronic.SCAN_DATA".

Method 1: JavaScript Function Call

Here, after a successful scan, the data are passed to a JavaScript function on the web page.

This function must be implemented by the customer on the website. The call is:

```
Aitronic.scanDataReceived('barcode')
```

Example:

```
var Aitronic = {
       scanDataReceived: function(dat){
       document.getElementById("idcode").value=dat;
       return true;
       }
};
```

Method 2: KeyUp/KeyDown Events

Keyboard input is generated from the scanned data. A minimum of KeyDown and a KeyUp event is generated for each character in the scanned code.

Example:

The character 'a' generates the events

- `KeyDown KeyCode: 65`
- `KeyUp KeyCode: 65`

The character 'A' generates the events

- `KeyDown KeyCode: 16` (Up / Shift key)
- `KeyDown KeyCode: 65`
- `KeyUp KeyCode: 65`
- `KeyUp KeyCode: 16`

since uppercase letters are generated using the Shift key.

These events can then be intercepted by JavaScript event listeners.

Playing Sounds with JavaScript

Sounds can be stored in the format wav, ogg or mp3 on the SD card of the device. For the aiBrowser to process these, they must be stored in folder `/aitronic/aiBrowser/sounds`. After changing the sound files, the aiBrowser must be restarted to read the new files. The files can then be played over the provided interface.

Example:

```
<html>
<body>
<button
onclick="javascript:AiBrowser.playSound('demo_error.ogg')">
Play Sound</button>
</body>
</html>
```

As a parameter, the file name of the sound file to be played must always be passed to the function `AiBrowser.playSound ()`.

Disable / enable devin with JavaScript

For example, You must disable devin. This can be done out of JavaScript via the function

```
AiBrowser.deactivateDevin()
```

devin is then temporarily disabled and the use of the camera is possible. When you press the scan button, devin is reactivated and starts scanning after a short initialization time. This reactivation can also be done using JavaScript with the function

```
Aibrowser.activateDevin()
```