netcompany



Version 0.2

Status Approved Approver

Author Daniel Joachim Larsen

KOMBIT

AULA – WIDGET LOCAL DEVELOPMENT

T0150 – Widget Local Development

© Copyright 2025 Netcompany. All rights reserved.

Neither this document nor any part thereof may be passed on to others, copied or reproduced in any form or by any means, or translated into another language without the express prior permission in writing from Netcompany.

Document history

Version	Date	Author	Status	Comments
0.1	22-06-2023	Daniel Joachim Larsen	Draft	
0.2	23-06-2023	Andreas Laursen	Approved	

References

Reference	Title	Author	Version
[WidgetGuide]	T0150 – Widget guide	Lasse Poulsen	0.1

Table of contents

1	Ba	ackground	3
	1.1	Target group	3
2	Se	etup	4
	2.1	Installation	4
	2.2	Git	4
	2.2.1	Clone the project	. 4
	2.2.2	Remote origins	. 4
	2.2.3	Updates	. 5
3	w	/idget development	5
	3.1	Getting Started	5
	3.2	SSO Widgets	5
	3.3	Iframe Widgets	6
	3.4	Other Widgets	7
	3.5	Widget props	8
4	Тс	polkit	10
	4.1	Compress	10
	4.2	Applying for new or change widget	. 11
	4.3	Review Widget Code	11
5	Li	mitations	12
	5.1	No use of v-html	12
	5.2	Usage of the aula folder	12
	5.3	Imports	12

1 Background

To make Aula successful as the communication platform for all parties involved in schools and daycare institutions, the system will be able to both

- Display user interfaces from other systems alongside the relevant communication tools within Aula
- Perform single-sign-on on web links into other systems, enabling automatic login in with the end-users UNI-Login Id

Such user interfaces and single-sign-on links from other systems are referred to as Widgets, and this guide describes how a Widget Supplier can develop & test Widgets to be used in Aula.

Examples of possible Aula Widgets:

- Widgets from the individual Municipalities' Learning Platforms (e.g. Homework or links to digital learning tools)
- Widget from absence registration solution to register student absence.
- Single-sign-on links that can be placed as a toolbox on relevant Aula Dashboards

1.1 Purpose of this document

Testing that a widget works with Aula is, of course, necessary ahead of deployment. Classically, the way to test a widget was to use the Widget previewer as described in [WidgetGuide]. With the addition of the Local Development Environment there is now an alternative way of testing widgets.

This document serves the purpose of providing instructions to Widget suppliers on setting up, testing, and deploying their widgets using the local development environment. This document is specifically designed for widget developers responsible for creating and updating widgets for Aula within the supplier's organization, the document aims to guide them through the necessary steps of developing widgets for Aula locally.

1.1.1 Relation to Widget previewer

Given the enhancement of the local widget development environment, which now supports multiple widget instances and the inclusion of test users, it is advised that widget suppliers utilize this environment for their development needs.

1.2 Target group

This guide is primarily intended for developers responsible for developing Widgets for Aula. Readers are assumed to have knowledge of web development, in particular HTML, JavaScript, CSS and REST Web API's.

In addition, the guide describes the process of setting up a local development environment for developing Aula Widgets.

2 Setup

This section describes the technical details for setting up local development of Widgets for Aula. If you are not yet an Authorized Widget Supplier, refer to [WidgetGuide] section 3.1 to become a Widget Supplier. Note that to gain access to the resources below it is necessary to subsequently create a case on this in the toolkit

2.1 Installation

To ensure that Widget Suppliers can run the development environment locally. The Widget Supplier must download and install Node JS and Docker on their machines.

Node.js is utilized for hosting the frontend of the local Widget development environment.

Docker is used to host the backend API, which generates the Aula token and lists all the test users.

Installing Node JS:

- 1. Go to the official Node JS website (<u>https://nodejs.org/</u>) and download the recommended version for your operating system.
- 2. Follow the installation instructions for your operating system.
- 3. Verify that Node JS is installed by opening a terminal or command prompt and typing "node -v". If Node JS is installed properly, it should display the version number.

Installing Docker:

- Go to the official Docker website (<u>https://www.docker.com/</u>) and download the appropriate version for your operating system.
- 2. Follow the installation instructions for your operating system.
- 3. After the installation is complete, verify that Docker is installed by opening a terminal or command prompt and typing "docker --version". If Docker is installed properly, it should display the version number.

To get the development environment up and running. Refer to the instructions described in the README.md file.

2.2 Git

This section describes the technical details of cloning and maintaining the local widget development repository.

2.2.1 Clone the project

Go to the Azure DevOps and clone the WidgetDevelopmentEnvironment repository.

2.2.2 Remote origins

To ensure continuous updates to the cloned project, assign a new remote origin name to the aula Azure DevOps repository.

Example

git remote rename origin aula

The widget supplier should add their own remote origin to push their changes towards.

```
git remote add origin {url}
```

Set the upstream to the new origin

```
git branch --set-upstream-to origin/master
```

2.2.3 Updates

All Widget Suppliers will receive an Email when the WidgetDevelopmentEnvironment repository is updated.

Typically, an update will occur when a new release update of Aula is up.

Regular updates are always cosmetic, in case there are more substantial changes, an email notice will be sent out to all widget suppliers regarding those changes.

After the Widget Supplier have received the Email notification, the widget supplier can update their copy of the repository with the following command

Example:

git pull aula master # pulls changes from the remote origin aula

3 Widget development

This Section describes the technical details of developing a widget for Aula.

3.1 Getting Started

Widget suppliers can use their preferred IDE to develop the widget source code. All code examples will be illustrated in the Jetbrains IDE.

To understand the technologies used in Aula, refer to [WidgetGuide] section 4 Technical guide.

3.2 SSO Widgets

To begin configuring SSO widgets the widget supplier can start the development environment and configure the SSO widget in the settings menu by setting the name, URL and icon, the SSO widget will be ready and configured.

Aula						SETTINGS	TEST USERS	GROUPS		? HELP
	NORMAL Min nye widge	SECURE	IFRAME	INFOTAVLE	SSO		Leveret af XXX	Min nye widge	t n nye widget	Leveret af XOO
KALENDER	🥼 мі	n nye widget								
BESKEDER										
2 MIN NYE WIDGET		Settings						×		
		Widget	settings					~		
		WIDGET NA	ME							
		Min nye w	idget							
		WIDGET DE	SCRIPTION							
		WIDGET UR	ı.							
		http://127	.0.0.1:5173/							
		WIDGET ICO	N							
		NO FILE C	HOSEN					BROWSE		
		GROUP ID (UNI GROUP ID)							
		INSTITUTIO	NCODE							
		00000								
	l					 		Reset Settings		

3.3 Iframe Widgets

When developing iframe widgets. It is recommended that the Widget supplier create a separate project in their preferred language or Javascript framework that the Widget development environment can communicate with.

There is a sample setup in the Widget development environment at src/IframeWidget Aula recommends that this folder is treated as an example for how to setup a Iframe widget, however widget suppliers can also develop in the IframeWidget folder as well.

An Iframe widget will be treated the same as a secure widget. Refer to [WidgetGuide] section 4.2 for information about available parameters for iframe widgets.

In the settings menu, the widget supplier can point towards the widget's hosted URL by setting the Widget URL.

Example

SECURE IFRAME INFOTAVLE SSO
WIDGET DESCRIPTION WIDGET URL http://127.0.0.1:5173/ WIDGET ICON NO FILE CHOSEN BROWSE GROUP ID (UNI GROUP ID)

3.4 Other Widgets

There is an example of how-to setup a widget for Widget vendors in src/WidgetSample. This folder is only meant to serve as a template example, any custom widgets should be created in a separate directory fx. src/MyCustomWidget/Main.vue.

N. Davids understand and de land de landeren de la		props: {	~ ~
auta-widget-test-portal C\Users\djr(NetcompanyProj		widgetID: String,	
> U.vscode		axios: {	
> 🛄 api		type: Function as PropType <axiosinstance></axiosinstance>	
> 🗈 aula			
> 🗅 dist		moment: {	
> D node_modules library root		type: Object as PropType <moment></moment>	
> 🗈 public			
✓ ➡ src		sessionUVID: String,	
> 🗀 IframeWidget	46	placement: String,	
WidgetSample		currentWeekNumber: String,	
V Main.vige	48	ISROBILEADP: BOOLEAN,	
10 vite-env.d.ts	49	UserProfile: String,	
≡ .env	50	group: String,	
■ envproduction	52	source, string,	
Ø gitignore	53	tristorizationale - Seiner Jan	
📥 docker-compose vml	54	assuranceLevel: Number.	
- dockerfile		childFilter: {	
		type: Array as PropType <string[]></string[]>	
A and and and			
O packagejson		institutionFilter: {	
O package-lockjson		type: Array as PropType <string[]></string[]>	
■ private_key.pem			
M4 README.md		9 getAulaToken: Function,	
O tsconfig.json		aulaToken: String,	
O tsconfig.node.json		notifications: Array,	
O tsconfig.webpack.json		deleteNotifications: Function,	
vite.config.ts	65		
🔲 vite-aula.config.ts	66 OT	setup(props: any) {	
🕫 webpack.config.cjs	67	const propkeys = computed (generit) => {	
s webpack.production.cjs	60	1).	1.1
由 External Libraries	70	const propsi ist = computed (peter () => 1	
Scratches and Consoles	71	pattern proprieter stranger (stranger) => proprieter (stranger) => f	
	72	return {	
	73	kau	

Customizing the widget type and placement can be done through settings.

netcompany

			SETTINGS	TEST USERS	
et	+				Normal widg
w_					This is your test platfor
en s	Settings				×
em	WIDGET DESCRIPTION				^
. Ap	Widget description				
g bø grø	WIDGET ICON				
r vil	NO FILE CHOSEN			BR	OWSE
	WIDGET URL				
rsb	http://localhost:5173				
. Ap	WIDGET TYPE				
olet	Secure				\$
mec	PLACEMENT				
	Default				÷
	Default Noticeboard				
	Widget Props				
	WidgerTops				•
	GROUP ID (UNI GROUP ID)				
		Remove Widget	Peset Widget Dr		otifications
		Keniove wiuger	Keset Widget Pi	Clear no	

3.5 Widget props

Refer to [WidgetGuide] section 4.2 for information about available widget props for any widget types.

A widget supplier can manipulate props sent to a widget through the settings button in the top menu.

+ ettings Widget Props GROUP ID (UNI GROUP ID) INSTITUTION CODE 00000 DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee ASSURANCE LEVEL Normal institution Flitter (COMMA SEPARATED)			کېک Settings	TEST USERS	GROUPS		
ettings Widget Props GROUP ID (UNI GROUP ID) Institution code 0000 0000 DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee ASSURANCE LEVEL Normal INSTITUTION FILTER (COMMA SEPARATED)							
ettings Widget Props GROUP ID (UNI GROUP ID) INSTITUTION CODE 00000 DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee ASSURANCE LEVEL Normal •							
Widget Props Image: Additional addi	ttings				×		
GROUP ID (UNI GROUP ID) INSTITUTION CODE 00000 DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee ASSURANCE LEVEL Normal *	Widget Props				× ^		
INSTITUTION CODE 00000 DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee \Rightarrow ASSURANCE LEVEL Normal \Rightarrow INSTITUTION FILTER (COMMA SEPARATED)	GROUP ID (UNI GROUP ID)				r	elt i tråd med komm	nuner
0000 DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee ASSURANCE LEVEL Normal INSTITUTION FILTER (COMMA SEPARATED)	INSTITUTION CODE				1	ug af juletræer i gru	ipper
DATE 04/13/2023 UNI-LOGIN benj359c PROFILE TYPE Employee \diamond ASSURANCE LEVEL Normal \diamond INSTITUTION FILTER (COMMA SEPARATED)	00000						
04/13/2023 me gruppe som PDF. UNI-LOGIN benj35% PROFILE TYPE Employee ASSURANCE LEVEL Normal INSTITUTION FILTER (COMMA SEPARATED)	DATE						
UNI-LOGIN benj359c PROFILE TYPE Employee \diamond ASSURANCE LEVEL Normal \diamond INSTITUTION FILTER (COMMA SEPARATED)	04/13/2023				n	ne gruppe som PDF.	
benj359c PROFILE TYPE Employee ASSURANCE LEVEL Normal INSTITUTION FILTER (COMMA SEPARATED)	UNI-LOGIN						
PROFILE TYPE Employee ASSURANCE LEVEL Normal INSTITUTION FILTER (COMMA SEPARATED)	benj359c						
Employee ASSURANCE LEVEL Normal INSTITUTION FILTER (COMMA SEPARATED)	PROFILE TYPE						
ASSURANCE LEVEL Normal \$ INSTITUTION FILTER (COMMA SEPARATED) •	Employee				÷		
Normal Normal Norma	ASSURANCE LEVEL						
INSTITUTION FILTER (COMMA SEPARATED)	Normal				÷		
	INSTITUTION FILTER (COMMA SEPARATED)				•		
Remove Widget Reset Widget Props Clear potifications		Remove Widget	Reset Widget Pr	ons Clear no	tifications		

Alternatively, widget props can be setup for specific purposes by choosing a test user in the top menu.

SETTINGS	TEST USERS	GROUPS		? HELP
		Test users		
		Lone Andersen		>
		Villads Kruse		>
		Jonas Kruse	>	
orvandles til flis, d om vinterfugle og n ud!".	er kan anvendes ; fuglefodring, og	Freja Hansen	>	
		Benjamin Jeppe	esen	>
skolen". Den er lige	ledes sendt ud			

NOTE: To use test users, the docker container in the widget development environment must be running.

4 Toolkit

Toolkit works as the central unit for anything widget related. This page will contain a document library with guides and documentation, a Cases page to create applications for new/updating/deleting widget, incidents and other service requests and the Widgets source code.

4.1 Compress Widget

When a Widget suppliers widget is ready, compress the folder to a zip file. In Windows this can be done by right clicking the folder and compress to zip file.

Name			Date modified		іуре	Size		
🔤 lframeWi	dget		25-01-2023 18:34		File folder			
Standard		Open			File folder			
🖬 vite-env.c		Open in new window			TS File	1 KB		
		Pin to Quick access						
	<u> </u>	Add to VLC media player's P	Playlist					
	×	Open with Visual Studio						
	-	Git GUI Here						
	-	Git Bash Here						
	<u> </u>	Play with VLC media player						
	× 1	Open with Code						
		Scan with Microsoft Defend	er					
		Give access to						
		Restore previous versions						
		Include in library						
		Pin to Start						
		Send to	>	8	Bluetooth device			
		Cut		2	Compressed (zipped) fo	lder		
		Сору			Desktop (create shortcu	t)		
		Create shortcut		<u>_</u>	Documents			
		Delete			Fax recipient			
		Rename			Mail recipient			
		Dramatian		<u> </u>	Apple iPhone			
		Properues		=	Shared (S:)			
1				2	Department (Y:)			
				-	dji (\\ncrite.nclan.netco	mpany.dk\Pers	sonal) (Z:)	LCJpc1BpbG90ljpmYWxzZ5wiaWF0lj

4.2 Applying for new or change widget

Refer to [WidgetGuide] section 4.3 for general information on how to add new widgets in Toolkit.

4.3 Review Widget Code

Widget Code can be uploaded in the Widget Code menu.

At the top of this menu page, the upload can be performed by clicking a new element (Nyt element)

GENNEMSE G	SETORGANIZED E	ELEMENTER LISTE							
Nyt element + My	Vis Rediger element element	Versionshistorik Delt med Slet element	Vedhæft Notificér Kopier fil	r direkte ink Giv mig besked +	Arbejdsprocesser Godkend/afv	/is			
Ny	Admir	inistrer	Handlinger	Del og overvåg	Arbejdsprocesser				
Nyt element		t (GTO)	> AULA Ekstern Test2 (K	(MTEXTST) Widget	code				
Føj et nyt elem Dokumentation	ent til denne liste n	+ nyt ele	ement eller redig	ger denne liste					
Dokument	oibliotek	Alle elemente	r ··· Find et eler	ment P					
Processer		Midaat/Mi	idaatld WidaatVarrian	Nama CanAccoss	aMabila CanRoBlacadOnC	roups CanRoDiacodiacidoModulo	~		
Sager		• Widget.Wi	Die entering en in der deutsche enteringen deutsche einer deutsche						
Udviklingsi	déer	Din søgning	gav ingen resultate	r.					
Kalender		Nogle elemen	ter kan være skjult. Med	dtag disse i søgninger	1				
Widgets									
Widget code									
Seneste									
Produktions ce	ertifikat								

Attach the Widget zip file.

GENNEMSE REDIGER				
Gem Annuller Satt	ABC Stawekontrol			
Anvena Galo pina der	and inger Stavekontrol			
Netcompany GOTO > GetOrganized	Vedhæft fil TST) :	Widget code Nyt element		
Dokumentation	Vedhæfter en fil i dette element.			
Dokumentbibliotek	Name *			
Processer	WidgetVue			
Saper				
Ush Disputition				
oundingsideer				
Kalender				
Widgets				
Widget code				
Seneste				
Produktions certifikat				
			A	
	WidgetNotificationVue			

5 Limitations

You should read all limitations in the [WidgetGuide], however, the listed limitations here are specifically relevant for when using the local development environment.

5.1 No use of v-html

The v-html directive in Vue.js allows you to render dynamic HTML content on your web page by binding a string of HTML to an element's innerHTML. While v-html can be a useful feature, it can also introduce a security risk if used improperly.

v-html can allow an attacker to inject malicious code into your web page, leading to a potential XSS attack. If the data that you're binding to v-html is not properly sanitized or validated, an attacker can inject script tags or other HTML elements that execute malicious code in the context of your page.

If you must render content Aula has made a custom component that sanitizes your input.

Example

<widget-html :html="strHtml"></widget-html>

****NOTE**: Links and inline javascript will be stripped from the content.

5.2 Usage of the aula folder

The Aula folder is for Aula developers only any changes in this folder will result in future merge conflicts when updating your cloned repo.

Aula recommends that you do not alter or make use of anything in the aula folder.

5.3 Imports

For non- iframe widgets, imports are limited to the following dependencies:

- element-ui
- bootstrap-vue
- lodash
- momentJS

- vue