Project Name	International Data Space Station
Online team meeting	https://fau.zoom-x.de/j/64245120479
Production system (Private)	https://github.com/projectamoscd/flux
Test system (if any)	(none yet)
GitHub repository	https://github.com/amosproj/amos2024ss02-international-dataspace-station
GitHub feature board	https://github.com/orgs/amosproj/projects/60
GitHub impediments backlog	https://github.com/orgs/amosproj/projects/59
Team T-shirt (black) (women)	https://www.shirtinator.co.uk/s/3Wt7FEo7RM23NZZT4qwLfw
Team T-shirt (black) (men)	https://www.shirtinator.co.uk/s/3WAORhs4QQ63NQPrUpa0tQ
Additional materials	https://github.com/projectamoscd
Team maling list	oss-amos-proj2@lists.fau.de
AMOS Happy	https://happy-amos.appspot.com/Project?project=5875167674761216&course=6219429234868224

Last Name	First Name	GitHub User Name	Email Address
Zhang	Jin	jinzhangfau	jin.zhang@fau.de
Kurtz	Daniel	daku-de	daniel.kurtz@fau.de
Kanatova	Sezim	skanatova	kanatova.sezim@fau.de
Sanyoto	Matthew Jason	msanyoto	sanyoto@campus.tu-berlin.de
Cosgun	Esra	esracosgun	esra.cosgun@campus.tu-berlin.de
Kröcker	Timo	timoKroecker	t.kroecker@web.de
Wysokinska	Xemena	xenia1w	x.wysokinska@campus.tu-berlin.de
Ivanishcheva	Ekaterina	EkaterinaIvanishcheva	ekaterii39@zedat.fu-berlin.de
~~Hirschpeck~~	~~Leon~~	~~leones18~~	~~leon.hirschpeck@fau.de~~
~~Rameshkumar~~	~~Rathujan~~	~~rathu2712~~	~~rathujan.rameshkumar@fau.de~~
~~Frieß~~	~~Tobias~~	~~Freeze-FF~~	~~tobias.friess@fau.de~~

	Meeting Day	Product Owners	Software Developer	Release Manager	Scrum Master	Comment
1	2024-04-17	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	N/A	Tobias Frieß	
2	2024-04-24	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Xemena Wysokinska	Jin Zhang	
3	2024-05-01	N/A	N/A	N/A	N/A	
4	2024-05-08	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Sezim Kanatova	Jin Zhang	
5	2024-05-15	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Ekaterina Ivanishcheva	Jin Zhang	
6	2024-05-22	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Daniel Kurtz	Jin Zhang	
7	2024-05-29	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Esra Cosgun	Jin Zhang	Mid-term due
8	2024-06-05	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Xemena Wysokinska	Jin Zhang	
9	2024-06-12	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Sezim Kanatova	Jin Zhang	
10	2024-06-19	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Ekaterina Ivanishcheva	Jin Zhang	
11	2024-06-26	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Daniel Kurtz	Jin Zhang	
12	2024-07-03	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Esra Cosgun	Jin Zhang	
13	2024-07-10	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Xemena Wysokinska	Jin Zhang	
14	2024-07-17	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Sezim Kanatova	Jin Zhang	Demo day!
15	2024-07-24	Timo Kröcker & Matthew Jason Sanyoto	Everyone else	Ekaterina Ivanishcheva	Jin Zhang	Retrospective

Goals	Successful Product, High customer satisfication, Good Teamwork
Meeting norms	Be on time on agreed meetings
Working norms	Transparency, Clear Documentation, Openess, Trust
Coordination norms	Tell the POs if something comes up (organisational/meetings)
Communication norms	Check communication channels daily (WhatsApp, Discord)
Consideration norms	POs: are responsible for the order and relevance of the to be implemented features, SDs: are responsible for the implementation and all technical aspects of the project, SM: is responsible for the organization of the SCRUM
Cont. improvement norms	SDs: we will be having a main branch where the working code from the previous sprints will be located. For the current sprint of branch dev will be created. SDs working on features will create new branches from the dev one: at the end of the sprint these branches will be merged into dev again and after reviewing it finally into main
Rewards	-
Sanctions	-
Signatures	
Scrum Master	Jin Zhang
Product owner	Timo Kröcker
Product owner	Matthew Jason Sanyoto
Software developer	Daniel Kurtz
Software developer	Xemena Wysokinska
Software developer	Sezim Kanatova
Software developer	Esra Cosgun
Software developer	Ekaterina Ivanishcheva
~~Software developer~~	~~Leon Hirschpeck~~
~~Software developer~~	~~Rameshkumar Rathujan~~

Product Vision	Project Mission
With the increase of data exchange between organizations and other independent instances such as finance, legal, healthcare, government, etc the need of ease of data interoperability while still adhering to data usage, policies, and compliance to local legalizations is becoming significant. Dataspace is the envisioned solutions to tackle these challenges.	Explore the feasibiltiy of dataspace usage with regards to data sovereignty. This includes the testing the maturity of dataspace, which components are important and ease of deployment

Term	Definition
Data Sovereignity	The collecting and processing of data should be subjected to the laws of the country of which the data are being generated.
Dataspace	Data ecosystem that is built upon commonly agreed policies
(EDC) Connector	Entry port to the dataspace as well as means to exchange data
Metadata Broker	
Catalog	
Policy	
Contract	

Sprint #	Sprint goal
1	Understanding the concept of Dataspace and it's components
2	Run samples of EDC Connector (Team 1) and Gaia-X framework samples (Team 2)
3	Build the Kubernetes pipeline for automatic deployment in collaboration with DATEV
4	Start establising frontend framework early
5	Continue with the frontend framework and make the application user friendly
6	Prepare for mid-project release
7	Documentation & refactoring
8	Decentralising the connector
9	Create web frontend for each connector
10	Connect the connector with a database
11	Transfer data between 2 connectors
12	UI improvement, bug fixes, deployment to cloud

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Release						
Total			88	88		
Sprints						
1	Understanding the concept of Dataspace and it's components		10	88	10	88
2	Run samples of EDC Connector (Team 1) and Gaia-X framework samples (Team 2)		13	78	13	78
3	Build the Kubernetes pipeline for automatic deployment in collaboration with DATEV		16	65	12	65
4	Start establising frontend framework early		8	49	13	53
5	Continue with the frontend framework and make the application user friendly		15	41	15	40
6	Prepare for mid-project release		26	26	26	25
	Sum					
Features						
1	**Understanding the concept of Dataspace and it's components**					
	Gain understanding of Gaia-X		5		5	
	Research connector concepts and the EDC connector repository		5		5	
2	**Run samples of EDC Connector (Team 1) and Gaia-X framework samples (Team 2)**					
	Eun EDC connector samples		8		8	
	Gain more understanding of Gaia-X		5		5	
3	**Build the Kubernetes pipeline for automatic deployment in collaboration with DATEV**					
	Create two connector instances		3		3	
	Transfer data on localhost		5		3	
	Create docker image output		5		3	
	Create documentation for the localhost connection process		3		3	
4	**Start establising frontend framework early**					
	Establish a frontend framework		8		13	
5	**Continue with the frontend framework and make the application user friendly**					
	Establish a frontend framework		8		13	
	Create build process video		2		2	
	Create a button in the frontend which runs the CLI automatically		5			
6	**Prepare for mid-project release**					
	Create a fuctioning button to establish a two-connector-connection		5		3	
	Open three ports for three connectors		2		2	
	Create UI design for login page		3		3	
	Create UI design for connector		5		5	
	Create UI frontend of the connector page		5		5	
	Create / Update the Dockerfile		3		5	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
	Test connector configurations		1		1	
	Response - bug fix		2		2	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Release						
Total			121	121		
Sprints						
1	Understanding the concept of Dataspace and it's components		10	121	10	121
2	Run samples of EDC Connector (Team 1) and Gaia-X framework samples (Team 2)		13	111	13	111
3	Build the Kubernetes pipeline for automatic deployment in collaboration with DATEV		16	98	12	98
4	Start establising frontend framework early		8	82	13	86
5	Continue with the frontend framework and make the application user friendly		15	74	15	73
6	Prepare for mid-project release		26	59	26	58
7	Documentation & refactoring		9	33	9	32
8	Decentralising the connector		6	24	10	23
9	Create web frontend for each connector		18	18	10	13
10	Connect the connector with a database			0		3
11	Transfer data between 2 connectors			0		3
12	UI improvement, bug fixes, deployment to cloud			0		3
Features						
1	**Understanding the concept of Dataspace and it's components**					
	Gain understanding of Gaia-X		5		5	
	Research connector concepts and the EDC connector repository		5		5	
2	**Run samples of EDC Connector (Team 1) and Gaia-X framework samples (Team 2)**					
	Eun EDC connector samples		8		8	
	Gain more understanding of Gaia-X		5		5	
3	**Build the Kubernetes pipeline for automatic deployment in collaboration with DATEV**					
	Create two connector instances		3		3	
	Transfer data on localhost		5		3	
	Create docker image output		5		3	
	Create documentation for the localhost connection process		3		3	
4	**Start establising frontend framework early**					
	Establish a frontend framework		8		13	
5	**Continue with the frontend framework and make the application user friendly**					
	Establish a frontend framework		8		13	
	Create build process video		2		2	
	Create a button in the frontend which runs the CLI automatically		5			
6	**Prepare for mid-project release**					
	Create a fuctioning button to establish a two-connector-connection		5		3	
	Open three ports for three connectors		2		2	
	Create UI design for login page		3		3	
	Create UI design for connector		5		5	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
	Create UI frontend of the connector page		5		5	
	Create / Update the Dockerfile		3		5	
	Test connector configurations		1		1	
	Response - bug fix		2		2	
7	**Documentation & refactoring**					
	Create build documentation		1		1	
	Create design documentation		3		2	
	Create user documentation		1		2	
	Bugfix: connector status checking		3		3	
	Refactor & delete unused branches		1		1	
8	**Decentralising the connector, creating web frontend for each connector, and start transfering the data**					
	Bugfix: "Execute command"-button		3		5	
	Uploadable files in the UI		3		5	
9	**Create web frontend for each connector**					
	Bugfix: Cloud UI accessibility		3			
	Text file transfer between connectors		5			
	Create web frontend for the bank connector		3		5	
	Create web frontend for tax advisor and company		2		1	
	Update docker file to accomodate three seperate web frontends		2		2	
	Check feasibility of establishing a connection with object storage		3		2	
10	Connect the connector with a database					
	Create Identity hub		5			
	Create JSON files for all scenarios		5			
	Link a role-exclusive login page to each port		3			
	Transfer a text file between two connectors - frontend		3			
	Transfer a text file between two connectors - backend		5			
	Connector status checking 2.0 Cloud		3			
11	Transfer data between 2 connectors					
12	UI improvement, bug fixes, deployment to cloud					

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining

Type	Link / reference
GitHub DATEV repo	https://github.com/projectamoscd
GitHub Wiki	https://github.com/amosproj/amos2024ss02-international-dataspace-station/wiki

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
	Product Owners approved of the features	There is no major regression compared to previous release	Same as sprint release
	Software Developers agree to release the features	All Software Developers agree to release	Relevant features have been documented and put into the wiki
	All acceptance criteria need to be met	Release Manager set the tag to release	Bugs should be kept as minimal as possible
	Pull request has been reviewed, approved, and merged into the main branch	Application should be able to be executed	

#	Context	Name	Version	License	Comment
1	org.eclipse.edc	EDC Connector	0.6.3	Apache-2.0	
2	Docker	Docker	26.1.4	Apache-2.0	
3	Gradle	Gradle	8.7.0	Apache-2.0	
4	@heroicons/react	JSON Package	2.1.3	JSON license	
5	axios	JSON Package	1.7.2	JSON license	
6	clsx	JSON Package	2.1.1	JSON license	
7	express	JSON Package	4.19.2	JSON license	
8	js-cookie	JSON Package	3.0.5	JSON license	
9	net	JSON Package	1.0.2	JSON license	
10	next	JSON Package	14.2.3	JSON license	
11	react	JSON Package	18.0.0	JSON license	
12	react-dom	JSON Package	4.7.5	JSON license	
13	socket.io	JSON Package	latest	JSON license	

Last Name	First Name	Value					
Kurtz	**Daniel**			**3.00**	**OK**		
Kanatova	**Sezim**						
Cosgun	**Esra**	3					
Wysokinska	**Xemena**	3					
Ivanishcheva	**Ekaterina**			0	No size		
				1	Trivial size		
				2	Small size		
				3	Medium size		
				5	Large size		
				8	Very large size		
				13	Too large (size)		
How to play planning poker							
1. Everyone type their number into their value field, don't hit return yet							
2. Someone, perhaps a product owner, count down 3.. 2.. 1..							
3. Then, everyone hit return to submit their value							