Project Name	Explainable Similarity Detector
Production system (if any)	...
Test system (if any)	...
GitHub repository	https://github.com/amosproj/amos2021ws06-exp-similarity-detector
GitHub kanban board (project)	https://github.com/amosproj/amos2021ws06-exp-similarity-detector/projects/1
Team T-shirt (white)	...
Team T-shirt (black)	https://www.shirtinator.de/loadBasket/OLYtK-gJU9h
Additional materials	...
Zoom-Link:	https://fau.zoom.us/j/65358072167?pwd=UFd4MFBHaU5iT3AwdVlMdnVxbXZwUT09

Last Name	First Name	GitHub User Name	Email Address	Roles (preliminary)
	Max	hemnemne		Backend
	Tim	t99-i		Frontend + PO
	Ronny	georgir20		Frontend
	René	Re-Krass		Backend
	Claudia	TuCl		Backend
	Hannes	h4nn3s94		Frontend
	Jasper	jasperjulius		Backend
	Simon	B4rtware		Backend

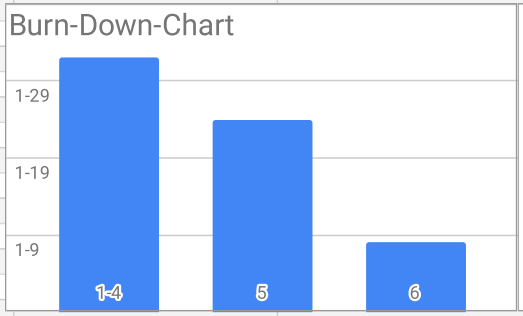
#	Meeting Day	Comment	Coach	Product Owner	Software Developer	Release Manager	Scrum Master
1	2021-10-21		Yes	Tim	Everyone else	N/A	Coach
2	2021-10-28		Yes	Tim	Everyone else	Claudia	Coach
3	2021-11-04		Yes	Tim	Everyone else	Max	Coach
4	2021-11-11		Yes	Tim	Everyone else	Jasper	Coach
5	2021-11-18		Yes	Tim	Everyone else	Tim	Coach
6	2021-11-25		Yes	Tim	Everyone else	Simon	Coach
7	2021-12-02	Mid-project relea	Yes	Tim	Everyone else	Hannes	Coach
8	2021-12-09			Tim	Everyone else	René	Coach
9	2021-12-16			Tim	Everyone else	Ronny	Coach
10	2022-01-13		Yes	Tim	Everyone else	Claudia	Coach
11	2022-01-20			Tim	Everyone else	Max	Coach
12	2022-01-27			Tim	Everyone else	Jasper	Coach
13	2022-02-03		Yes	Tim	Everyone else	Tim	Coach
14	2022-02-10	Demo day / final release		Tim	Everyone else	Simon	Coach
15	2022-02-17	Project retrospective due		Tim	Everyone else	Hannes	Coach

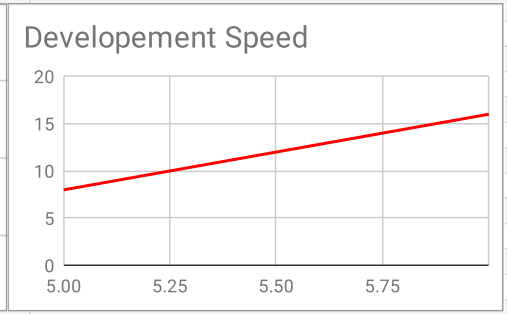
Goals	Das Ziel ist die Sicherstellung einer effektiven, verlässlichen aber auch gleichzeitig flexiblen Teamkommunikation.
	Jeden im Projekt involviert halten, indem man sich für die aktuellen Aufgaben/Probleme der anderen interessiert.
Meeting norms	Wir treffen uns alle jeden Donnerstag spätestens um 12:35 Uhr.
	Optionales 30-minütiges wöchentliches Meeting kann nach individueller Vereinbarung dazukommen.
	Mehr als 10 Minuten Verspätung müssen kommuniziert werden.
Working norms	Jeder zeigt gegenseitigen Respekt und übernimmt volle Verantwortung für ihre/seine Aufgabenbereiche.
	Features im Team besprechen, Umsetzung in den Kompetenzbereichen / Unterteams.
	Es ist ok manchmal seine eigenene Kompetenzen zu überschätzen und sich rechtzeitg Hilfe zu suchen.
	Bei einer Diskussion von mehr als (ungefähr) 15 Minuten, dann wird abgestimmt. Tie Breaker ist der PO.
Coordination norms	Product Owner sollte Telegram moderieren.
	Meeting Moderation nach Scrum Norm. Darauf achtet der Scrum Master.
	Alternative Modelle werden akzeptiert, solange sie nicht Scrum wiedersprechen.
Communication norms	Für die externe Kommunikation sollte Telegram genutzt werden
	Zoom für weekly calls: (https://tu-berlin.zoom.us/j/68376196208?pwd=b0N6NUFXcnFxQXhaSVB6TXFwM25aQT09)
Consideration norms	Jeder darf einmal ohne besonderen Grund, aber mit vorheriger Ansage, fehlen.
Cont. improvement norms	Wichtige Entscheidungen werden im Meeting Protokoll vom PO abgelegt.
	Jeder darf und soll ehrliches Feedback äußern.
Rewards	Gemeinsames Bier bei Projektabschluss.
	Wir wissen uns gegenseitig für unsere Arbeit wertzuschätzen.
Sanctions	2 Euro Spende an Wohltätige Organisation, wenn man etwas "verhauen" hat.
	Wahlweise Bierspende ans Team.
	xx X x x x X x }{

Product Vision	Project Mission
The Explainable Similarity Detector should give all developers who work with electronic components a noticeable offer simplification in everyday work. Through the use of machine learning and an easily understandable surface, the time-consuming search for suitable components should be faster, more convenient and clearer.	The mission of this project is to utilize the machine learning algorithm given by Siemens for the find functionally similar electronic components. This is supposed to be given on the basis of one or more Components are done with the help of filters.
The reason of existence of the envisioned product (beyond this project).	The mission of this particular project (in the context of the product vision).

Term	Definition
(electronic) components	semiconductors, LEDs, etc.
admin	An administrator in the company that uses the product
CD	Shorthand for Continous Deployment
CI	Shorthand for Continous Integration
Continous Deployment	Software releases that successfully pass the automated tests will also be deployed automatically
Continous Integration	Practice of merging all developers working copies to the same shared main repositry
Dummy	A function that has not yet been fully implemented, but is used to test basic functionalities
IaC	Shorthand for Infarstructure as Code
Industry-Partner-Software-Developer	Software developers on the part of the industrial partner who will work on the software product
Infarstructure as Code	Managment of the infarstructure in a descriptive model (using versioning like the source code)
Machine learning model	The machine learning model given by the industrial partner
ML-model	Shorthand for machine learning model
Project-Developer	Project side software developer
Scheduled-for-Split	It is likely that this feature will have to be broken down into subtasks
SFS	Shorthand for Scheduled-for-Split
Developer	a developer who works with electronic components (industry partner)
Procurement responsible	

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
1-4	Organisation, clearing questions with the Industry-Partner and familiarize with the material				/		/	33
		Allocation of roles, laying the foundations, setting up software and holding first industrial partner meetings (mainly non-code)						
				/		/		
5	Groundwork				11		8	25
		Create first Domain Model to work with (Frontend), testing requests and understanding/working with the ML-Code, data etc.						
			Create first Domain Model with most important entities in mendix	3		3		
			Get-Request (azure-Model) (Dummy)	8		5		
6	Dummy-Prototype (Part 1)				22		16	9
		Deliver first prototype to test basic interaction between frontend and backend and lay the foundation for the Basic-Protoype						
			create JSON-Format	3		5		
			Request function for data(Dummy)	5		0		
			Function for adding component(capacitor)(Dummy)	3		5		
			Create database-scheme(SQL)	3		3		
			Backend-API connection with the database (Dummy)	5		0		
			User feedback function for openAPI	3		3		

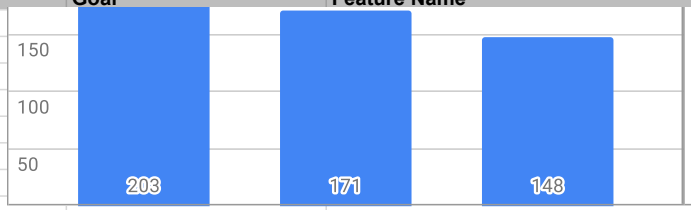


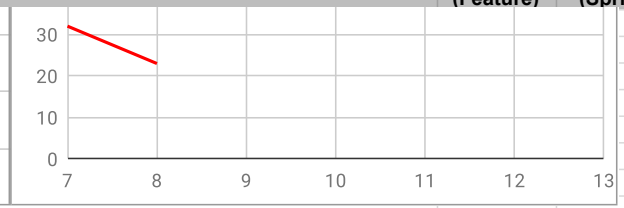


#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
7	**Dummy-Prototype (Part 2)**				**34**		**32**	**203**
		Finish all Dummy function and testing basic interaction for implementation of real functions						
			Backend-API (Dummy)	**5**		**5**		
			Function for adding component(resistor)(Dummy)	**2**		**2**		
			Host-Open-API-Specification (Swagger)	**5**		**5**		
			Backend-API connection with the database (Dummy)/Implementation of azure functions relying on database access (for capacitors)	**5**		**5**		
			Create userrole: admin	**2**		**2**		
			Request function for data (Dummy)	**5**		**8**		
			Implementation fo azure function for ML-model inference (Dummy)	**5**		**0**		
			Host Mendix Docker container on azure	**5**		**5**		
8	**Basic-Version (Part 1)**				**22**		**23**	**171**
		Deliver a prototype that can already receive and process input						
			Implementation of azure functions for ML-model inference(Dummy)	**5**		**8**		
			(restructuring / adapting) of data and mapping	**5**		**5**		
			Split endpoint for retraining and adding of new components	**5**		**5**		
			Create new users	**3**		**0**		
			Backend-API connection with the database (Dummy)/Implementation of azure functions relying on database access (for resistors)	**2**		**2**		
			Fix the 500 Server Error on the Backend-API commection when accessing endpoints	**2**		**3**		
9	**Basic-Version (Part 2)**				**40**		**TBD**	**148**
		Deliver a prototype that implements required functionality						
			Post request for single capacitors	**5**		**TBD**		
			Generalising Backend-API connection with the database (Dummy) for attributes	**3**		**TBD**		
			Parse POST Body-Content in ML API	**5**		**TBD**		
			Use hash function for returning Mock ML comments in API	**2**		**TBD**		
			Add suffix 'schema' to pydantic models	**1**		**TBD**		
			Improve error handling in CreateSimilarities endpoint	**3**		**TBD**		
			Create new users	**3**		**TBD**		
			Add CI pipeline	**5**		**TBD**		
			Change Framework	**5**		**TBD**		
			Add tests for Azure Functions	**8**		**TBD**		
10	**Basic-Version (Part 3)**				**47**		**TBD**	

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
		Deliver a prototype that implements required functionality						
			Outsource the filter serilization into a seperate function which is global available for the Azure Functions	**5**		**TBD**		
			Prefilter endpoint for search filter results	**5**		**TBD**		
			Include MIN MAX value for each attribute through an endpoint	**3**		**TBD**		
			Error-handling(missing attributes)	**3**		**TBD**		
			Use IaC concept (e.g. Terraform) to easily deploy resources on Azure	**3**		**TBD**		
			Add CD pipeline	**5**		**TBD**		
			Machine-Learning model connection (azure)	**5**		**TBD**		
			Create OpenAPI specification for ML modell	**5**		**TBD**		
			Deserialize Return Message from ML-model Inference	**5**		**TBD**		
			Creation of a validation error object for POST operations in OpenAPI and implementation	**2**		**TBD**		
			Get request for a list of components	**3**		**TBD**		
			Subsequent editing of entries in the list	**3**		**TBD**		
11	**Advanced-Version (Part 1)**				**16**		**TBD**	
		deliver a prototype which already contains advanced(non-basic) functionalities						
			Tab system	**8**		**TBD**		
			Show data sheet	**3**		**TBD**		
			Filter results	**5**		**TBD**		
12	**Advanced-Version (Part 2)**				**31**		**TBD**	
		deliver a prototype which already contains advanced(non-basic) functionalities						
			Filter based search	**13**		**TBD**		
			Allocation of results in classes	**13**		**TBD**		
			Refresh-Switch	**5**		**TBD**		
13	**Advanced-Version (Part 3)**				**13**		**TBD**	
		deliver a prototype which already contains advanced(non-basic) functionalities						
			price and property comparison of similar components	**8**		**TBD**		
			Search result counter	**5**		**TBD**		

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down





Last Name	First Name	Value					
	Max			#DIV/0!	#DIV/0!		
	Tim						
	Ronny						
	René						
	Claudia			0	No size		
	Hannes			1	Trivial size		
	Jasper			2	Small size		
	Simon			3	Medium size		
				5	Large size		
				8	Very large size		
				13	Too large (size)		

Sprint	Status	Source	Impediment	Resolution
1	Resolved	Industry partner	Fehlende Informationen zu Spezifikation und Technik	Warten auf User-Journey-Meeting
1	Resolved	Industry partner	Nicht die versprochenen Unterlagen geschickt	Unterlagen bekommen.
1	Resolved	Industry partner	User-Journey Meeting erst in zwei Wochen angesetzt.	Später Termin ist in Ordnung.
5	Resolved	Siemens	Frage bezügliche des Patents/Copyrights für ML-Modell	Mit Riehle gesprochen, ist für den Moment in Ordnung.
5	Resolved	SD	Azure-Experte hat erst "irgendwann" Zeit (sehr spät)	Termin steht für den 30.11
5	Resolved	Prof. Riehle	Privates Git-Repository vs. Benotung	Mit Riehle gesprochen, ist für den Moment in Ordnung.
5	Resolved	Siemens	Copyright Frage bezgl. des Mendix-Templates	Mit Riehle gesprochen, ist für den Moment in Ordnung.
5	In-work	Siemens (Emre)	ML-Modell läuft nicht entgegengesetzt der Ansage	
6	Unsolvable	Maria	Maria ist im Urlaub und nicht erreichbar	Erstmal kein Hinderniss
6	Resolved	Siemens	Ansprechpartner für ML-Modell nicht erreichbar.	Ansprechpartner Emre erreicht und in Kontakt
7	Resolved	Siemens (Tejashri)	Noch keine Lizenz für Mendix von Siemens bekommen	
7	Resolved	SD	Die Students-Credits für Azure reichen nicht mehr lange	Falls Credits drohen leer zu gehen, einfach bei Dirk melden
8				

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
1	- Code has been reviewed by developer	- no bugs that affect the functionality	- Basic functionality and front end is ready for use
2	- Code has been reviewed by the responsible developer Team (Frontend/Backend)	- small bugs are documented for rectification	- User documentation is ready **(subject to change)**(used and checked by AT LEAST 3 off-project persons (Industry-Partner-User))
3	- individual acceptance criteria are met	- The user documentation is updated according to the changes	- Developer documentation is ready **(subject to change)**(used and checked by AT LEAST 3 off-project persons (Industry-Partner-Software-Developer)
4		- The developer documentation is updated according to the changes	
5		- (Optimal): 80-90% of the assigned tasks were completed successfully	
6		- (Minimum): At least 50% of all assigned tasks have been completed (discussion in the meeting)	
			Appendix: Subject-to-change Expect further information from the industry partner, how many people are available for this

\	Context	Name	Version	License	Comment
1	Backend (Azure)	SQLAlchemy	1.4.27	MIT License	
2	Backend (Azure)	azure-functions	1.8.0	MIT License	
3	Backend (Azure)	click	8.0.3	BSD License	
4	Backend (Azure)	colorama	0.4.4	BSD License	
5	Backend (Azure)	greenlet	1.1.2	MIT License	
6	Backend (Azure)	mypy	0.91	MIT License	
7	Backend (Azure)	mypy-extensions	0.4.3	MIT License	
8	Backend (Azure)	pathspec	0.9.0	Mozilla Public License 2.0 (MPL 2.0)	
9	Backend (Azure)	platformdirs	2.4.0	MIT License	
10	Backend (Azure)	psycopg2-binary	2.9.2	GNU Library or Lesser General Public License (LGPL)	
11	Backend (Azure)	pydantic	1.8.2	MIT License	
12	Backend (Azure)	pyodbc	4.0.32	MIT License	
13	Backend (Azure)	regex	2021.11.10	Apache Software License	
14	Backend (Azure)	sqlalchemy-stubs	0.4	MIT License	
15	Backend (Azure)	toml	0.10.2	MIT License	
16	Backend (Azure)	tomli	1.2.2	MIT License	
17	Backend (Azure)	typing-extensions	4.0.0	Python Software Foundation License	
18	Backend (Development)	black	21.11b1	MIT License	
19	Backend (Machine Learning)	Markdown	3.3.6	BSD License	
20	Backend (Machine Learning)	Pillow	8.4.0	Historical Permission Notice and Disclaimer (HPND)	
21	Backend (Machine Learning)	Werkzeug	2.0.2	BSD License	
22	Backend (Machine Learning)	absl-py	1.0.0	Apache Software License	
23	Backend (Machine Learning)	certifi	2021.10.8	Mozilla Public License 2.0 (MPL 2.0)	
24	Backend (Machine Learning)	charset-normalizer	2.0.7	MIT License	
25	Backend (Machine Learning)	click	8.0.3	BSD License	
26	Backend (Machine Learning)	cycler	0.11.0	BSD License	
27	Backend (Machine Learning)	dataclasses	0.8	Apache Software License	
28	Backend (Machine Learning)	fasttext	0.9.2	MIT License	
29	Backend (Machine Learning)	gensim	4.1.2	LGPL-2.1-only	
30	Backend (Machine Learning)	grpcio	1.42.0	Apache Software License	
31	Backend (Machine Learning)	idna	3.3	BSD License	
32	Backend (Machine Learning)	importlib-metadata	4.8.2	Apache Software License	
33	Backend (Machine Learning)	joblib	1.1.0	BSD License	
34	Backend (Machine Learning)	kiwisolver	1.3.1	BSD License	
35	Backend (Machine Learning)	matplotlib	3.3.4	Python Software Foundation License	
36	Backend (Machine Learning)	mlxtend	0.19.0	BSD License	
37	Backend (Machine Learning)	nltk	3.6.5	Apache Software License	
38	Backend (Machine Learning)	numpy	1.19.0	BSD	
39	Backend (Machine Learning)	pandas	0.25.3	BSD	
40	Backend (Machine Learning)	protobuf	3.19.1	3-Clause BSD License	
41	Backend (Machine Learning)	pyarrow	6.0.1	Apache Software License	
42	Backend (Machine Learning)	pybind11	2.8.1	BSD License	
43	Backend (Machine Learning)	pyparsing	3.0.6	MIT License	
44	Backend (Machine Learning)	python-dateutil	2.8.2	Apache Software License; BSD License	

\	Context	Name	Version	License	Comment
45	Backend (Machine Learning)	pytz	2021.3	MIT License	
46	Backend (Machine Learning)	regex	2021.11.10	Apache Software License	
47	Backend (Machine Learning)	requests	2.26.0	Apache Software License	
48	Backend (Machine Learning)	sacremoses	0.0.46	MIT License	
49	Backend (Machine Learning)	scikit-learn	0.24.2	new BSD	
50	Backend (Machine Learning)	scipy	1.5.4	BSD License	
51	Backend (Machine Learning)	six	1.16.0	MIT License	
52	Backend (Machine Learning)	smart-open	5.2.1	MIT License	
53	Backend (Machine Learning)	stop-words	2018.7.23	BSD License	
54	Backend (Machine Learning)	tensorboard	2.0.0	Apache Software License	
55	Backend (Machine Learning)	threadpoolctl	3.0.0	BSD License	
56	Backend (Machine Learning)	torch	1.10.0	BSD License	
57	Backend (Machine Learning)	tqdm	4.62.3	MIT License; Mozilla Public License 2.0 (MPL 2.0)	
58	Backend (Machine Learning)	typing-extensions	4.0.0	Python Software Foundation License	
59	Backend (Machine Learning)	urllib3	1.26.7	MIT License	
60	Backend (Machine Learning)	zipp	3.6.0	MIT License	
61	Frontend	Mendix	9.6.1	Free License	

Type	Link / reference
User Documentation	https://github.com/Re-Krass/amos2021ws06-exp-similarity-detector/blob/main/Documentation/user/README.md
Build Documentation (Frontend)	https://github.com/Re-Krass/amos2021ws06-exp-similarity-detector/blob/main/Documentation/build/frontend/docker/README.md
Build Documentation (Backend)	https://github.com/Re-Krass/amos2021ws06-exp-similarity-detector/blob/main/Documentation/build/backend/azure/README.md
Software-Architecture	https://github.com/Re-Krass/amos2021ws06-exp-similarity-detector/blob/main/Deliverables/2021-11-30_sprint-06-software-architecture.pdf