Project Name	...
Online team meeting	https://fau.zoom-x.de/j/61530257866?pwd=WUxZTWhpeHk2alBZTXJYYUFFNTVvZz09
Production system (if any)	...
Test system (if any)	...
GitHub repository	https://github.com/amosproj/amos2023ws01-ticket-chat-ai
GitHub feature board	https://github.com/orgs/amosproj/projects/25
GitHub impediments backlog	https://github.com/orgs/amosproj/projects/32
Team T-shirt (white)	-
Team T-shirt (black)	https://www.shirtinator.de/s/g18_eHiQRw-_7MpoGga0xg
Additional materials	...
Team email list	oss-amos-proj1@lists.fau.de

Last Name	First Name	GitHub User Name	Email Address
Murtaza	Sajjad	sajjadmurtaza	sajjad.murtaza@campus.tu-berlin.de
Hoxhallari	Irild	irhox	irild.hoxhallari@campus.tu-berlin.de
Konopka	Garvin	garvinkon	garvin.konopka@fu-berlin.de
Weber	Fabian	WebFa98	webef98@zedat.fu-berlin.de
Hirschbeck	Anna	AnnaH3003	anna.hirschbeck@fau.de
Gnagniko	Koffi Tino	Gnagniko	tino.gnagniko@gmail.com
Wegner	Thorben	thogebati	thorben.wegner@fau.de
Härtl	Marco Martin	M-HRL	marco.haertl@fau.de
Miltner	Jan	JMiltner97	jan.miltner@fau.de

#	Meeting Day	Product Owner	Software Developer	Release Manager	Scrum Master	Comment	Visits
1	2023-10-18	Anna	Everyone else	N/A	Jan Miltner		
2	2023-10-25	Thorben	Everyone else	Marco Martin Härtl	Jan Miltner		
3	2023-11-01			Koffi Tino Gnagniko			
3	2023-11-08	Thorben	Everyone else	Marco Martin Härtl	Jan Miltner		y
4	2023-11-15	Anna	Everyone else	Koffi Tino Gnagniko	Jan Miltner	**Build process review**	
5	2023-11-22	Thorben	Everyone else	Marco Martin Härtl	Jan Miltner		y
6	2023-11-29	Thorben	Everyone else	Koffi Tino Gnagniko	Jan Miltner		
7	2023-12-06	Anna	Everyone else	Marco Martin Härtl	Jan Miltner	**Mid-project review**	**y**
8	2023-12-13	Thorben	Everyone else	Koffi Tino Gnagniko	Jan Miltner		
9	2023-12-20	Anna	Everyone else	Garvin Konopka	Jan Miltner		
10	2024-01-10	Anna	Everyone else	Marco Martin Härtl	Jan Miltner		y
11	2024-01-17	Thorben	Everyone else	Koffi Tino Gnagniko	Jan Miltner		
12	2024-01-24	Anna	Everyone else	Marco Martin Härtl	Jan Miltner		
13	2024-01-31	Anna / Thorben	Everyone else	Koffi Tino Gnagniko	Jan Miltner		y
14	2024-02-07	Anna / Thorben	Everyone else	Marco Martin Härtl	Jan Miltner	Demo day!	
15	2024-02-14	Thorben	Everyone else	Koffi Tino Gnagniko	Jan Miltner	Retrospective	

Goals	Be respectful to each other. Be nice and try not to be too harsh to your teammembers	
	Make the teammeetings fun for every participant	
	Help each other	
	Be productive and work efficient	
	Deliver good working software that helps our customer	
Meeting norms	Everybody has to show up	
	Don't waste your teammates time. Let your team members know, if you should be late.	
	Everyone comes prepared to the meetings	
	Everyone is motivated and contributes actively	
	If possible show your face, even if it's just for a short time	
	Fokus on the meeting	
Working norms	Decisions are made democratically	
	We support each other	
	Work on features on individual branches	
	Use best practices when coding	
	Release code should always be commented	
Coordination norms	Everyone is responsible for his/her assigned tasks and has to deliver. If problems arise, tell the team in time	
	Every developer can pick backlog items for each sprint. Try to respect others' wishes	
	POs moderate the meeting, but everyone can always state his/her opinion	
	If certain topics should be addressed in meetings, let t he POs know in advance	
Communication norms	The main communication channel is discord	
	Respect everyone's opinion	
	Respond to messages at least on the next day if you are tagged	
	All public messages are written in English	
	In case of illness inform the team about the implications on your work	
Consideration norms	POs can always stop discussions when they deem them irrelevant or too specific for the whole team meeting	
	The scrum master intervenes whenever a discussion gets out of hand	
Cont. improvement norms	Try to improve the quality by giving constructive feedback	
	Respect that every developer has his/her own way of doing things	
	Try to find the underlying reason if the sprint plan fails and take respective measures	
Rewards	Play online game together	
Sanctions	We always try to solve problems immediatly as a team. If that does not work out we will sanction specific behavior or a member after a democratic discussion	
Signatures		
Scrum Master	Jan Miltner	
Product owner	Anna Hirschbeck	
Product owner	Thorben Wegner	
Software developer	Marco Martin Härtl	
Software developer	Irild Hoxhallari	
Software developer	Garvin Konopka	
Software developer	Sajjad Murtaza	
Software developer	Koffi Tino Gnagniko	
Software developer	Fabian Weber	

Product Vision	Project Mission
The reason of existence of the envisioned product (beyond this project). TalkTix empowers support teams to work more efficiently by eliminating short-term disruptions and thus enabling a focused way of working. It allows customers to easily report problems via free text or a simple voice message and automatically converts it into a support ticket without the support team actually having to create it manually.	*The mission of this particular project (in the context of the product vision).* The mission of this initiative is to develop a proof of concept initiated by Siemens, aimed at transforming unstructured free text and voice messages into IT support tickets. The fundamental capability of the system enables users to articulate their issues through either text or voice messages. Subsequently, an artificial intelligence component converts this input into a predefined support ticket format, and users are promptly notified upon its generation. The ultimate result is a JSON object representing the support ticket, to be seamlessly integrated into Siemens' ticketing system.

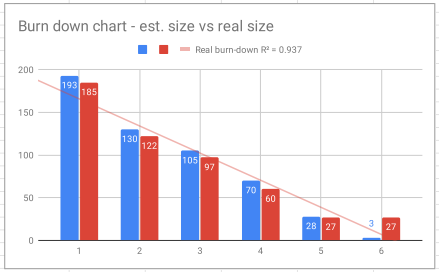
Term	Definition
IT support ticket	An IT support ticket is a formal request for help or assistance with a specific issue related to information technology (IT) or to requests services, following a certain structure containing specific form fields
title	the overarching topic of the IT support ticket
location	the location where the issue arose or where the service / help is to be executed
requestType	The type of request, for example: Incident, the user that opened has an issue of some kind and want it fixed or Service Request, the user requests some kind of service
description	The content of an IT support ticket, outlining the user's request
category	A few keywords that describte the over all topic of the ticket
customerPriority	How fast the ticket needs to be solved
priority	identifying number of customer priority object, indicating the urgency of the ticket solution in terms of business value
user	a person that wants to create an IT support ticket
IT support desk agent	a professional responsible for providing technical assistance and support to users through a helpdesk or support ticket system
ticket	identifying unique number of an IT support ticket

Sprint #	Sprint goal
1	None
2	None
3	None
4	Thorough integration of Email and Database
5	Refinement of code and completion of missing connections
6	Extend frontend by attachements and speech to text
7	Finalization of MVP + Improving UX
8	Preparation for feedback-loop + Initial AI Training
9	Foundation of feedback-loop + Improving accuracy of AI output
10	Create test data & train AI
11	Improving first draft of feedback-loop
12	Finalizing feedback-loop + Improving accuracy of AI output
13	Finishing touches to the use and feel
14	Preparing project retrospective

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Release						
Total			170	198	163	190
Sprints						
1	**Getting started and team building**		5	193	5	185
2	**Research and creating first simple components**		63	130	63	122
3	**First simple integration of AI, Email proxy and Database**		25	105	25	97
4	**Thorough integration of Email and Database**		35	70	37	60
5	**Refinement of code and completion of missing connections**		42	28	33	27
6	**Extend frontend by attachements and speech to text**		25	3	0	27
Features						
1	**Getting started and team building**					
		Decide on a team name	2		2	
		Design the team logo	2		2	
		Decide on the T-Shirt Design	1		1	
2	**Research and creating first simple components**	Decide on the software architecture	8		8	
		First Backend	5		5	
		[BE] Message from Backend to Frontend	3		3	
		Research into what kind of AI would make sense and how to train them	8		8	
		Evaluate T5 AI	8		13	
		Evaluate BART AI	8		5	
		Research how to store our data	5		3	
		Research into how to "get" the call and how to access the Email	5		5	
		Create SBOM	2		2	
		Create First Gmail Account	1		1	
		First WebApp / Front End	5		5	
		create super-linter workflow - WIP	5		5	
3	**First simple integration of AI, Email proxy and Database**	[BE] Add data base to backend	5		5	
		[BE] Create EmailProxy	5		5	
		create super-linter workflow - WIP	5		5	
		[FE+BE]Connection of Frontend (TalktixChatUI) and Backend (TalktixChatAPI)	5		5	
		[BE] Integration of AI into the backend	5		5	
4	**Thorough integration of Email and Database**	Creation of test data	8		8	
		[BE] Convert AI's output to a dictionary	3		3	
		Create good ChatGPT prompt	3		3	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
		Git Workflow License Test and Testing workflow	5		5	
		[BE] Mock Database in Tests	5		5	
		[BE] Created ticket is saved to the Database	3		5	
		[BE] Transferring email content to backend (TalktixChatAPI)	3		3	
		[BE] Establishing the connection between EmailProxy and TalktixChatAPI	5		5	
5	**Refinement of code and completion of missing connections**	Add dependency injection for production code and tests	8		5	
		[FE] [BE] Text box for the email	5		0	
		Create build process video	3		3	
		Create & Agree on project-specific definition of done	3		2	
		Create data	8		8	
		Good error handling	5		5	
		[BE] Ticket information from TalktixChatAPI to EmailProxy	5		5	
		[BE] proxy automatically reconnects to Email server in case of a disconnect	5		5	
6	**Extend frontend by attachements and speech to text**	Initialize user, (technical) design, and build/deploy documentation	5		3	
		[FE] [BE] Text box for the email	5		2	
		[FE] Attachement uploads in the frontend	5		8	
		[BE] Attachement from Proxy to TalktixChatAPI	5		3	
		[FE] Converting speech to text	5		5	
		[FE] Polishing WebApp	2		5	
		[BE] Expand backend to accept attachements	5		8	

Burn-down to Project - Mid Term



Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining		
Release								
Total			276	276				
Sprints								
7	Finalization of MVP + AI preliminaries		40	276	37	276		
8	Preparation for feedback-loop + Initial AI Training		37	236	32	239		
9	Foundation of feedback-loop + Improving accuracy of AI output		32	199	30	207		
10	Create test data & train AI		27	167	28	177		
11	Enhance AI accuracy& Usability		38	140	23	149		
12	Finish profile creation & start editing mode		39	102	22	126		
13	Finishing touches to the use and feel		46	63	19	104		
14	Preparing project retrospective		17	17	14	85		
7	**Finalization of MVP + AI preliminaries**							
		[BE] Attachement from Proxy to TalktixChatAPI	5		5			
		Refactor test data to include requestType	5		5			
		[FE] Creation of error messages	5		5			
		[FE] [BE] Text box for the email	5		5			
		AI Training preliminaries	8		5			
		Research "Confidence-Score" for NLPs	5		5			
		Create test data	5		5			
		Refactor test data by renaming location	2		2			
8	**Preparation for feedback-loop + Initial AI Training**							
		[EP] Proxy fixing	5		2			
		Update Chat GPT Prompt	2		2			
		[DB] Create User Database	3		5			
		[EP] Modifying Email Proxy to Recognize and Store .txt Files as Attachments	3		3			
		[FE] [BE] Create executable	8		8			
		[AI] First draft ticket creation	8		8			
		[BE] Set location based on user	3		1			
		Research on further possible suitable data sets	5		3			
9	**Foundation of feedback-loop + Improving accuracy of AI output**							
		[BE + AI] adjust ticket object and AI output	3		2			
		Adjust available data sets	5		5			
		Adjust voice recording	3		3			
		[AI] Improve accuracy of ticket output	8		5			
		[BE] Set location based on user	2		2			
		[EP] Proxy fixing	3		5			
		Prepare docker for industry partner	8		8			
10	**Create test data & train AI**							
		Expand data base by locations, services and category	5		3			
		Create test data	5		5			
		[FE+BE] Enquire for missing RequestType	8		8			
		Create Gmail Accounts	1		1			
		[AI] Improve accuracy of ticket creation by training AI	8		8			

		[EP] Proxy fixing	3		5			
		Translate testdata to English	5		5			
11	**Enhance AI accuracy& Usability**							
		Translate WebApp & error messages to English	2		1			
		[FE+BE] Log in & out for existing test users	8		8			
		Create more test data	3		1			
		[AI] Improve accuracy of service AI	5		5			
		[AI] Create category AI	5		5			
		[AI] Improve AI model for correct RequestType recognition	5		5			
		[AI] Create priority AI	5		5			
		[AI] create customerPriority AI	5		5			
		Ignore service emails from microsoft	3		3			
12	**Finish profile creation & start editing mode**							
		[FE+BE] Creation of profile for new users	8					
		[FE+BE] Edit existing user profile	8					
		[FE] Improve UI of the WebApp & e-mail	3					
		[BE] Creation of Draft-mode	5					
		[BE + FE] Editing mode	8					
		[AI] Adapt category AI to only show department	2					
		[Homework] Create demo day video	5					
13	**Finishing touches to the use and feel**							
		[EP] Bug fix	3					
		[BE/FE] Make the Title and Keyword AI output pretty	3					
		[Documentation] Update SBOM	1					
		[Documentation] Update architecture documentation	2					
		[FE] Make it pretty	5					
		[FE] Message when user sessions expires & chat clearing	5					
		[BE] Create more tests for the Backend	open end					
		[FE] Include profile icons	3					
		[FE] Adjust scaling of attachement field	5					
		Update demo video	3					
		[FE] Improve responsiveness of WebApp	8					
		[FE] Improve accessibility	8					
14	**Preparing project retrospective**							

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
	All acceptance criteria are satisfied	Project builds, deploys, and tests successfully	Project builds, deploys, and tests successfully
	Work products are uploaded to the Github repository and merged to the main branch	Existing features must remain operational	Tested on all major browsers
	If necessary, the bill of materials section of the planning documents is updated	Sprint release notes have been written	Release notes have been written
	Code/feature passes suited tests		
	Code is peer-reviewed		
	Help documentation (README) is updated		
	Fits into our architecture, where possible		

Type	Link / reference

#	Context	Name	Version	License	Comment
	angular/angular	Angular	16.2.9	MIT	Pulled from Node Package Manager (npm)
	microsoft/typescript	Typescript	5.2.0	Apache-2.0	Pulled from Node Package Manager (npm)
	python	Python	3.11 LTS	Python Software Foundation License Version 2	Installed via installer https://www.python.org/downloads/
	numpy	Numpy	1.26.1	BSD License (BSD-3)	Pulled from Python Package Index (pip)
	pytorch	Pytorch	2.1.0	BSD License (BSD-3)	Pulled from Python Package Index (pip)
	huggingface/transformers	Transformers	4.34.0	Apache-2.0	Pulled from Python Package Index (pip)
	fastapi	FastAPI	0.74.0	MIT	Pulled from Python Package Index (pip)
	pytest	PyTest	7.4.3	MIT	Pulled from Python Package Index (pip)
	encode/uvicorn	uvicorn	0.15.0	BSD License (BSD-3)	Pulled from Python Package Index (pip)
	pymongo	PyMongo Driver	4.6.0	Apache-2.0	Pulled from Python Package Index (pip)
	mongodb/mongodb-community-server	MongoDB Community Server	7.0.2	Server Side Public License Version 1	Installed via installer https://www.mongodb.com/try/download/community
	python-dotenv	python-dotenv	1.0.0	BSD License (BSD-3)	Pulled from Python Package Index (pip)
	python-formating	black	23.11.0	MIT	Pulled from Python Package Index (pip)
	protobuf	protobuf	4.25.0	BSD License (BSD-3)	Pulled from Python Package Index (pip)
	sentencepiece	sentencepiece	0.1.99	Apache-2.0	Pulled from Python Package Index (pip)
	pandas	Pandas	2.1.2	BSD License (BSD-3)	Pulled from Python Package Index (pip)
	python-multipart	Python Multipart	0.0.6	Apache-2.0	Pulled from Python Package Index (pip)
	beautifulsoup4	Beautiful Soup	4.12.2	MIT	Pulled from Python Package Index (pip)
	podman/podman-desktop	Podman Desktop	1.6.X+	Apache-2.0	Installed via installer https://podman.io
	openai	openai-python	1.7.2	Apache-2.0	Pulled from Python Package Index (pip)
	requests	requests	2.31.0	Apache-2.0	Pulled from Python Package Index (pip)

Last Name	First Name	Value					
Murtaza	**Sajjad**			**#DIV/0!**	**#DIV/0!**		
Hoxhallari	**Irild**						
Konopka	**Garvin**			0	No size		
Weber	**Fabian**			1	Trivial size		
Gnagniko	**Koffi Tino**			2	Small size		
Härtl	**Marco Martin**			3	Medium size		
				5	Large size		
				8	Very large size		
				13	Too large (size)		