

Project Name	Cloud Native LLM
Online team meeting	https://fau.zoom-x.de/j/67390211804?pwd=SzlsTVZ1WjkzOHU1ZXVmVFM1V0NzUT09
Production system (if any)	...
Test system (if any)	...
GitHub repository	https://github.com/amosproj/amos2024ss08-cloud-native-llm
GitHub feature board	https://github.com/orgs/amosproj/projects/57
GitHub impediments backlog	https://github.com/orgs/amosproj/projects/52
Team T-shirt (white)	...
Team T-shirt (black) (female)	https://www.shirtinator.de/en/s/LCIK9Uk5SPe2dkCDPs2zOQ
Team T-shirt (black) (male)	https://www.shirtinator.de/en/s/w8SArOIS0a8qh96I_E8aw
Additional materials	...
Team mailing list	oss-amos-proj8@lists.fau.de

Last Name	First Name	GitHub User Name	Email Address
Li	Jiaao	grayJiaaoLi	ljaxmcx@gmail.com
Schäffer	Daniel	dnsch	dnsch@mailbox.org
Perez	Julio	julio-c-p	perezduranjulio@gmail.com
Fischer	Dominic	dominic0df	fischer.df@web.de
Wegner	Thorben	thogebati	thogebati@thogebati.de
Rezaei	Ahmad Anosh	anosh-ar	ahmadanosh.ar@gmail.com
Tiwary	Poorvi Pankaj	poorvi0311	poorvipankajtiwary@gmail.com
Pansuriya	Yashodhar	YashodharPansuriya	yashodhar.pansuriya@gmail.com
Wielenberg	Christian	christianwielenberg	chrissi_8@gmx.de

Goals	Be respectful to each other. Be nice and try not to be too harsh to your teammates
	Make the team meetings 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
	Focus on the meeting
Working norms	Decisions are made democratically
	We support each other
	Work on feature 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 the 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
	Overvoting of SM and PO possible if relevant
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	Thorben Wegner
Product owner	Jiaao Li
Product owner	Dominic
Software developer	Daniel Schäffer
Software developer	Poorvi Pankaj Tiwary - Dropped out
Software developer	Ahmad Anosh Rezaei
Software developer	Julio Perez
Software developer	Yashodhar Pansuriya
Software developer	Christian Wielenberg

#	Meeting Day	Product Owners	Software Developer	Release Manager	Scrum Master	Comment
1	2024-04-17	Jiaao Li, Dominic Fischer	Everyone else	N/A	COACH student	
2	2024-04-24	Jiaao Li, Dominic Fischer	Everyone else		COACH student	
3	2024-05-01	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
4	2024-05-08	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
5	2024-05-15	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
6	2024-05-22	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
7	2024-05-29	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	Mid-term due
8	2024-06-05	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
9	2024-06-12	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
10	2024-06-19	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
11	2024-06-26	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
12	2024-07-03	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
13	2024-07-10	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	
14	2024-07-17	Jiaao Li, Dominic Fischer	Everyone else	Julio Perez	COACH student	Demo day!
15	2024-07-24	Jiaao Li, Dominic Fischer	Everyone else	N/A	COACH student	Retrospective
Product owners, software developers, and Scrum Master are set and ideally don't change over time; the critical part is the Release Manager role you need to define here						

Product Vision	Project Mission
<p>This project aims to simplify the Cloud Native ecosystem by resolving information overload and fragmentation within the CNCF landscape. Our vision is a future where developers and users can effortlessly obtain detailed, context-aware answers about CNCF projects, thereby boosting productivity and enhancing comprehension.</p>	<p>1. Automated Dataset Preparation: Develop an automated system that prepares structured datasets from CNCF landscape documentation.</p> <p>2. NER Integration: Utilize Named Entity Recognition (NER) to extract key entities from the CNCF documentation, categorizing the data into predefined groups, establishing specific entities unique to the CNCF landscape and construct the relationships between entities.</p> <p>3. LLM Fine-Tuning: Enhance a Large Language Model specifically tailored to the prepared dataset for precise and relevant responses.</p> <p>4. Open Source Contribution: Release and open-source the fine-tuned model, dataset preparation tools, and NER integration methodologies to the community, fostering collaboration and innovation.</p>

[illegible]

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Release						
Total			136	136		
Sprints						
1-2	Implement ETL process		26	136	22	136
3	Research and Identify the suitable LLM		19	110	19	114
4	Develop Q&A Pairs Transforming Pipeline for model training		28	91	24	95
5	Enhance the Quality of Q&A Dataset and Research for NER		34	63	27	71
6	Documentations and Training LLM		29	29	25	44
Features						
1-2	Implement ETL process	Identify Webpages for the Kubernetes LLM dataset	5		3	
		Extract Content from CNCF sources	1		5	
		Transform extracted data into a unified Format	5		5	
		Load data to Hugging Face	2		2	
		Monitoring of the ETL process by Logging	3		2	
		Identify Repositories for the Kubernetes LLM dataset	5		5	
		Create Software architecture draft	5		5	
3	Research and Identify the suitable LLM	Research Open-Source LLMs	3		3	
		Conduct Large Language Base Model Selection Process	5		5	
		Create a Report on LLM Research Findings	3		3	
		Enhance Software Architecture Draft - Detail Machine Learning Pipeline	5		5	
		Extract the dataset from Hugging Face	3		3	
4	Develop Q&A Pairs Transforming Pipeline	CI/CD pipeline for project	3		3	
		Research Q&A Pairs Pipeline	3		5	
		Implement an Initial Question Extraction Function	5		5	
		Implement an Initial Answer Matching Function	5		2	
		Implement an Initial Generator for enriching Q&A Pairs	5		5	
		Store Q&As in Hugging Face	5		2	
		Exclude Non-English Content from Raw Dataset	2		2	
5	Enhance the Quality of Q&A Dataset and Research for NER	Improve the Q&A Pairs Transformation Pipeline	5		5	
		Improve the Performance of Data Unifying Script	3		3	
		Research and Select a Suitable NER Tool or Library	5		3	
		Learn How to Utilise the HPC	5		3	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
		Make whole project able to be build with a build tool	8		8	
		Include Stackoverflow Q&As related to CNCF into our Q&A dataset	8		5	
6	Documentations and Training LLM	Create design documentation	3		3	
		Create User documentation	3		2	
		Create build/deploy documentation	3		2	
		Utilise HPC for Data Generation Scripts	5		5	
		Preprocessing of the data of the Q&A dataset	5		5	
		Implementation of the Retraining to get a CNCF LLM	5		5	
		Train LLM on FAUs HPC	5		3	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Release						
Total			260	260		
Sprints						
1-2	Implement ETL process		26	260	22	260
3	Research and Identify the suitable LLM		19	234	19	238
4	Develop Q&A Pairs Transforming Pipeline for model training		28	215	24	219
5	Enhance the Quality of Q&A Dataset and Research for NER		34	187	27	195
6	Documentations and Training LLM		29	153	25	168
7	Deploy the initially trained LLM, create a chatbot with it and start evaluating the generated LLM		29	124	29	143
8	Prepare the evaluation of the generated LLM and its hyperparameter tuning, refactor ETL and data transformation scripts		30			
9						
10						
Features						
1-2	Implement ETL process	Identify Webpages for the Kubernetes LLM dataset	5		3	
		Extract Content from CNCF sources	1		5	
		Transform extracted data into a unified Format	5		5	
		Load data to Hugging Face	2		2	
		Monitoring of the ETL process by Logging	3		2	
		Identify Repositories for the Kubernetes LLM dataset	5		5	
		Create Software architecture draft	5		5	
3	Research and Identify the suitable LLM	Research Open-Source LLMs	3		3	
		Conduct Large Language Base Model Selection Process	5		5	
		Create a Report on LLM Research Findings	3		3	
		Enhance Software Architecture Draft - Detail Machine Learning Pipeline	5		5	
		Extract the dataset from Hugging Face	3		3	
4	Develop Q&A Pairs Transforming Pipeline	CI/CD pipeline for project	3		3	
		Research Q&A Pairs Pipeline	3		5	
		Implement an Initial Question Extraction Function	5		5	
		Implement an Initial Answer Matching Function	5		2	
		Implement an Initial Generator for enriching Q&A Pairs	5		5	
		Store Q&As in Hugging Face	5		2	
		Exclude Non-English Content from Raw Dataset	2		2	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
5	Enhance the Quality of Q&A Dataset and Research for NER	Improve the Q&A Pairs Transformation Pipeline	5		5	
		Improve the Performance of Data Unifying Script	3		3	
		Research and Select a Suitable NER Tool or Library	5		3	
		Learn How to Utilise the HPC	5		3	
		Make whole project able to be build with a build tool	8		8	
		Include Stackoverflow Q&As related to CNCF into our Q&A dataset	8		5	
6	Documentations and Training LLM	Create design documentation	3		3	
		Create User documentation	3		2	
		Create build/deploy documentation	3		2	
		Utilise HPC for Data Generation Scripts	5		5	
		Preprocessing of the data of the Q&A dataset	5		5	
		Implementation of the Retraining to get a CNCF LLM	5		5	
		Train LLM on FAUs HPC	5		3	
7	Deploy the initially trained LLM, create a chatbot with it and start evaluating the generated LLM	Train LLM on FAUs HPC	3		3	
		Deploy the trained model to Huggingface	5		5	
		Create a Chatbot on HuggingFace	5		5	
		Select suitable Benchmark metrics	5		5	
		Run Stackoverflow extraction to get more data	3		3	
		Extract and Store Text Data From CNCF Project Webpages	8		8	
8	Prepare the evaluation of the generated LLM and its hyperparameter tuning, refactor ETL and data transformation scripts	Run Q&A Generator with the Latest Collected Data	5		3	
		Study and Research on LLM Hyper-parameters	5		5	
		Refactor the ETL and Data Transformation Scripts	5		5	
		Prepare a small test set for evaluation	5		3	
		Collect previous research work in our Wiki	3		2	
		Create a Chatbot on HuggingFace	5		3	
		Run Stackoverflow extraction to get more data	2		2	
9	Use new Hetzner infrastructure to fine-tune (hyperparameter tuning) our LLM and continue with NER	Run Stackoverflow extraction to get more data	1			
		Refactor the ETL and Data Transformation Scripts	3			
		Run Q&A Generator with the Latest Collected Data	3			
		Prepare a small test set for evaluation	1			
		Implement a Script for LLM fine-tuning	8			
		Utilise hetzner for LLM Fine-tuning	5			
		Fine-tune LLM on Hetzner	5			
		Implement Quantitative Evaluation Script	5			

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
1	Bill of Materials in the planning document has been updated	Project builds, deploys, and tests successfully	User documentation has been updated
2	The requirement.txt is updated	HuggingFace data Repo update scripts succeed	Architecture documentation has been updated
3	All feature branches have been merged and closed	Sprint release notes have been written	Release plan has been updated
4	New feature code has been documented	change log is updated	Sprint Release DoD is fulfilled
5	Potential new licenses have been checked		
6	All GitHub Actions are passing		

Type	Link / reference
Build/Deploy documentation	https://github.com/amosproj/amos2024ss08-cloud-native-llm/wiki/Build-Deploy-documentation
Design documentation	https://github.com/amosproj/amos2024ss08-cloud-native-llm/wiki/Design
User documentation	https://github.com/amosproj/amos2024ss08-cloud-native-llm/wiki/User-Documentation

#	Context	Name	Version	License	Comment
1	roboto font used for team logo	ttf-roboto	2.138-4	Apache-2.0	taken from https://archlinux.org/packages/extra/any/ttf-roboto/
2	handle yaml files	PyYAML	6.0.1	MIT License	
3	make requests	requests	2.31.0	Apache-2.0	
4	cache reponses of requests	requests_cache	1.2.0	BSD License	
5	mock for unit tests	mock	5.1.0	BSD License	
6	get and push huggingface datasets	datasets	2.19.1	Apache Software License	
7	Used to generate pdf files for testing	reportlab	4.2.0	BSD License	
8	Used for Authentication to huggingface	huggingface_hub	0.23.0	Apache Software License	
9	read and write json	ijson	3.2.3	BSD License	
10	read and write csv	pandas	2.2.2	BSD License	
11	read and write pdf	PyPDF2	3.0.1	BSD License	
12	scrape data from web	Scrapy	2.11.1	BSD License	
13	log length of loops	tqdm	4.66.2	MIT License, Mozilla Public License 2.0 (MPL 2.0)	
14	array manipulation	numpy	1.23.2	BSD License	
15	language identification	langid	1.1.6	BSD License	
16	QA generation	lmqg	0.1.1	MIT License	
17	load NLP models	spacy	3.7.4	MIT License	
18	abstract boilerplate code for GPUs/TPUs	accelerate	0.30.1	Apache Software License (Apache)	
19	Loading transformer models, tokenizers & and additional helper packages	transformers	4.41.1	pache Software License (Apache 2.0 License)	
20	general machine learning library	torch	2.3.0	BSD License (BSD-3)	
21	Lora and Peft Configurations for Finetuning an LLM	peft	0.11.1	Apache Software License (Apache)	
22	make training more efficient	bitsandbytes	0.43.1	MIT	
23	Supervised Fine-tuning	trl	0.8.6	Apache Software License (Apache 2.0)	
24	Detect language of a text file	langid	1.1.6	Costum lic, Redistribution Allowed	
25					
26					

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

#	Context	Name	Version	License	Comment

Sprint #	Sprint goal
1	Project Initialisation and Meeting with Kubernetes
2	Implement ETL process
3	Research and Identify the suitable LLM
4	Develop Q&A Pairs Transforming Pipeline for model training
5	Enhance the Quality of Q&A Dataset and Research for NER
6	Initial training of the CNCF LLM
7	Deploy the initially trained LLM, create a chatbot with it and start evaluating the generated LLM
8	Prepare the evaluation of the generated LLM and its hyperparameter tuning, refactor ETL and data transformation scripts
9	Use new Hetzner infrastructure to fine-tune (hyperparameter tuning) our LLM and continue with NER
10	
11	
12	
13	
14	
15	

Last Name	First Name	Value					
Schäffer	Daniel	5		5.00	OK		
Perez	Julio	5					
				0	No size		
Rezaei	Ahmad Anosh	5		1	Trivial size		
				2	Small size		
Pansuriya	Yashodhar	5		3	Medium size		
Wielenberg	Christian	5		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							

