	Knowledge Graph Extractor (Hella)
Online team meeting	https://fau.zoom-x.de/j/67111681334?pwd=LzdBM3lXeXhPTEtWL3lGUnFqbTAzZz09
Production system (if any)	n/a. Everything is built locally from the GitHub repo
Test system (if any)	n/a. Everything is built locally from the GitHub repo
GitHub repository	https://github.com/amosproj/amos2024ss05-knowledge-graph-extractor
GitHub feature board	https://github.com/orgs/amosproj/projects/56/views/2
GitHub impediments backlog	https://github.com/orgs/amosproj/projects/69
Team T-shirt (white)	...
Team T-shirt (black)	https://www.shirtinator.de/s/pYjJO4qcR3u9lSKbgQdyiw
Additional materials	...
Team maling list	oss-amos-proj5@lists.fau.de
Single Demo Day Slides	https://docs.google.com/presentation/d/117Dtbkm4HWCBunBCTG7MC_yQGgf65zF7/edit#slide=id.g2ea86f049f2_0_0
Demo video slides	https://docs.google.com/presentation/d/1_LUVofRksDbKnRNJpvj6N50Crg2OXrlO/edit#slide=id.p1
Demo day slides	https://docs.google.com/presentation/d/1SZt8DKno8YjCOdPmgsX_dqkMbXW8nkFi/edit#slide=id.p1

Last Name	First Name	GitHub User Name	Email Address
Kuo	Irene	kuoirene	kuo.irene.y@gmail.com
Greiner	Rebecca	RebeccaGreiner	rebecca.greiner@fau.de
Rauscher	Nikolas	nikolas-rauscher	nikolas.rauscher@gmail.com
Ozseker	Irem	iremozs	iremozseker@gmail.com
Müller	Hanna	hanna-212	hanna.mueller@fau.de
Fabian Borges	Filipe Alexandre	borges-filipe	filipe.af.borges@gmail.com
Kotini	Kristi	kristikotini	kristi.kotini@fau.de
Bhesaniya	Yash	yashbhesaniya	yashbhesaniya1999@gmail.com
Ramesh	Sandeepkumar	Sandeep-kumar-Ramesh	sandeepkumar.ramesh@fau.de
Hoffmann	Florian	get4flo	f.hoffmann@campus.tu-berlin.de

#	Meeting Day	Product Owners	Software Developer	Release Manager	Scrum Master	Comment
1	2024-04-17	Irene Kuo, Rebecca Greiner	Everyone else	n/a	Hanna Müller	
2	2024-04-24	Irene Kuo, Rebecca Greiner	Everyone else	n/a	Hanna Müller	
3	2024-05-01	Irene Kuo, Rebecca Greiner	Everyone else	n/a	Hanna Müller	
4	2024-05-08	Irene Kuo, Rebecca Greiner	Everyone else	n/a	Hanna Müller	
5	2024-05-15	Irene Kuo, Rebecca Greiner	Everyone else	Kristi Kotini	Hanna Müller	
6	2024-05-22	Irene Kuo, Rebecca Greiner	Everyone else	Nikolas Rauscher	Hanna Müller	
7	2024-05-29	Irene Kuo, Rebecca Greiner	Everyone else	Sandeepkumar Ramesh	Hanna Müller	Mid-term due
8	2024-06-05	Irene Kuo, Rebecca Greiner	Everyone else	Yash Bhesaniya	Hanna Müller	
9	2024-06-12	Irene Kuo, Rebecca Greiner	Everyone else	Florian Hoffmann	Hanna Müller	
10	2024-06-19	Irene Kuo, Rebecca Greiner	Everyone else	Filipe Borges	Hanna Müller	
11	2024-06-26	Irene Kuo, Rebecca Greiner	Everyone else	Kristi Kotini	Hanna Müller	
12	2024-07-03	Irene Kuo, Rebecca Greiner	Everyone else	Nikolas Rauscher	Hanna Müller	
13	2024-07-10	Irene Kuo, Rebecca Greiner	Everyone else	Irem Ozseker	Hanna Müller	
14	2024-07-17	Irene Kuo, Rebecca Greiner	Everyone else	Florian Hoffmann	Hanna Müller	Demo day!
15	2024-07-24	Irene Kuo, Rebecca Greiner	Everyone else		Hanna Müller	Retrospective

Goals	1. Finish tasks for each sprint on time.	
Meeting norms	1. Be on time! (send a msg in WhatsApp if you'll be late) 2. Show up (unless deathly sick) 3. Try to participate actively	
Working norms	1. Good comments/documentation of work so everyone can follow easily. 2. Don't do everything the day before it's due. 3. Reach out if you have questions, help each other out!	
Coordination norms	1. Make it clear on the feature board what you're working on. 2. If you're overwhelmed, communicate so we can reassign tasks.	
Communication norms	1. Create WhatsApp group and reach out for questions and concerns there first (informal quick chats) 2. Discord for screenshots, code concerns, one point of reference for project items.	
Consideration norms	1. Be kind to each other.	
Cont. improvement norms	1. Have a retrospective after each sprint.	
Rewards	Everyone bring your own treat and we can have a celebratory meeting at the end!	
Sanctions	If you're more than 5min late without notice, 1pushup per minute late is owed.	
Signatures		
Scrum Master	Hanna Müller	
Product owner	Irene Kuo	
Product owner	Rebecca Greiner	
Software developer	Nikolas Rauscher	
Software developer	Irem Ozseker	
Software developer	Yash Bhesaniya	
Software developer	Filipe Borges	
Software developer	Kristi Kotini	
Software developer	Florian Hoffmann	
Software developer	Sandeepkumar Ramesh	

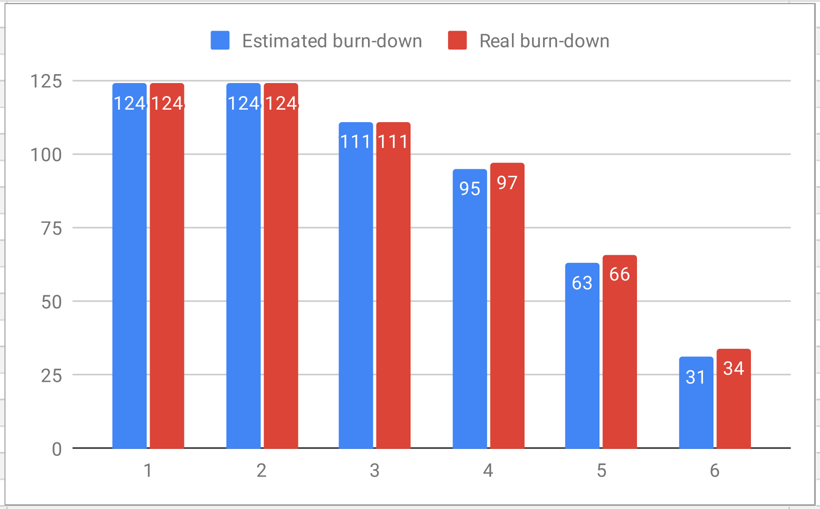
Product Vision	Project Mission
An AI-powered chatbot that helps any user query and extract knowledge from uploaded document(s). Through generating knowledge graphs from a corpus of text, information and knowledge is organized in a smarter way that is able to reveal different insights that may not have been noticed before. The knowledge graph will include communities of concepts and can be used to uncover insights and links between seemingly disconnected concepts. Through querying knowledge graphs, users can more quickly gather the correct information and potentially gain additional understandings that are not noticeable without the graph communities.	The mission of this project is to create a MVP for the knowledge graph generation in order to visually see clusters of information and how they're linked. The knowledge graph will include a basic search function to query information. Core functionality will be ingesting user document(s), processing the data and extracting relationship entities through the use of LLMs, building and storing the knowledge graph, an interactive visual representation of the knowledge graph, and a basic search function for entities in the knowledge graph.

	Definition
ASPICE	An industry-standard guideline for evaluating and improving software development processes in the automotive industry.
Barnes Hut	A hierarchical algorithm that approximates forces in n-body simulations to reduce computational complexity.
Edges	Connections between nodes in the knowledge graph, indicating relationships or associations between entities.
Embeddings	Vector representations of the entities and relationships within the graph. They capture the semantic meaning and structural properties of the graph in a continuous vector space, allowing for more efficient computation and analysis.
Entities	Key concepts, objects, or subjects extracted from text or data, forming the nodes of the knowledge graph.
ForceAtlas2	A force-directed layout algorithm for graphs, balancing attractive and repulsive forces for visualization.
Hierarchical	A layout style that organizes nodes in a tree-like structure with levels of hierarchy.
Hierarchical repulsion	A method that arranges nodes hierarchically, applying repulsive forces to avoid overlap and improve readability.
Knowledge Graph (KG)	A knowledge base that uses a graph structure to represent the data with nodes as objects and edges as relationships between the nodes.
Large Language Model (LLM)	An advanced artificial intelligence model trained on vast amounts of text data to understand and generate human-like language.
Layout algorithms	Methods used to arrange the positions of nodes in a graph.
Nodes	Representations of entities within the knowledge graph, each node encapsulates information about a specific entity.
Physics Options	Settings that control the physical simulation in graph layout algorithms.
Repulsion	A force that pushes nodes away from each other to prevent overlap in graph layouts.
Technical document	A piece of written content that provides detailed information, instructions, or explanations about a specific technical subject, product, or process.
Component	A set of nodes that are connected by edges form a component

Sprint #	Sprint goal
1	None
2	None
3	None
4	Optional
5	Finish all basic components/functions in preparation for connecting them all for the end-to-end functionality (upload -> knowledge graph visualization).
6	MVP for mid-project with upload through 1st basic visualization of knowledge graph
7	Streamline UX and work on additional knowledge graph generation tasks
8	Update UI and knowledge graph fine-tuning
9	Enhance graph visualization and LLM-usage
10	Graph search functionality and UI improvements
11	Finalize graph search and graph visualization
12	Finish final project release and prepare for demo day

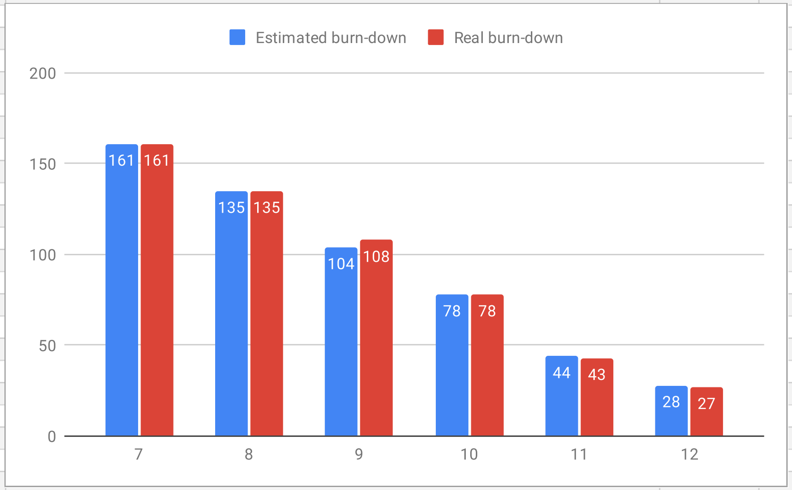
Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Release						
Total			124	124		
Sprints						
1	**Getting started**		0	124	0	124
2	**Define technologies, create software architecture and user interface design**		13	124	13	124
3	**Setup project environment**		16	111	14	111
4	**Ingestion of documents and LLM setup with POC of data processings**		32	95	31	97
5	**Preparation of individual functions to connect for the MVP**		32	63	32	66
6	**MVP for mid-project with upload through 1st basic visualization of knowledge graph**		31	31	31	34
	Sum			0		3
Features						
1	**Getting started**	Setup feature board	n/a		n/a	
2	**Define technologies, create software architecture and user interfa**	Team logo	n/a		n/a	
		Create software architecture overview	5		5	
		Design user interface	8		8	
3	**Setup project environment**	Set up initial project environment (backend excluding LLM container)	8		8	
4	**Ingestion of documents and LLM setup with POC of data processings**	PDF parsing into text	5		3	
		Text to .json chunks	3		3	
		Interface setup	3		3	
		Allow user to upload PDF document(s)	5		5	
		POC: Graph visualization	5		5	
		Setup Mistral locally: documentation	3		2	
		POC: Prompt template for LLM	3		5	
		Syntax checking for JSON and converting to graph format	5		5	
5	**Preparation of individual functions to connect for the MVP**	Update software architecture diagram and documentation	1		1	
		Prepare LLM setup for dev team	2		2	
		POC: combine graph pieces with LLM	8		8	
		Create record in database	5		5	
		LLM function	3		3	
		Generate graph button	5		5	
		CORS implementation	3		3	
		CI/CD improvements	2		3	
		HW: Build process video	3		2	

Sprint	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
6	**MVP for mid-project with upload through 1st basic visualization of knowledge graph**	Save graph data	8		8	
		User documentation	3		3	
		Connecting components from documents to entities	5		5	
		Deploy documentation	3		3	
		Connect entities to graph data	5		5	
		Generate graph visualization from database	5		5	
		Technical/design documentation	2		2	



	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining		
Release								
Total			161	161				
Sprints								
7	**Streamline UX and work on additional knowledge graph generation tasks**		26	161	26	161		
8	**Update UI and knowledge graph fine-tuning**		31	135	27	135		
9	**Enhance graph visualization and LLM-usage**		26	104	30	108		
10	**Graph search functionality and UI improvements**		34	78	35	78		
11	**Finalize graph search and graph visualization**		16	44	16	43		
12	**Finish final project release and prepare for demo day**		28	28	33	27		
	Sum			0		-6		
Features								
7	**Streamline UX and work on additional knowledge graph generation tasks**	Linting/Formatting	3		3			
		Create landing page	3		3			
		View list/table of existing knowledge graphs	5		5			
		Create new knowledge graph button (link to current user flow)	2		2			
		Delete uploaded document from upload screen	3		3			
		Refine .json extraction from LLM results	3		3			
		Refine graph connections	3		3			
		Ordering size of nodes for graph visualization	3		3			
		Remove JanusGraph	1		1			
8	**Update UI and knowledge graph fine-tuning**	Update user interface pages to design theme	3		3			
		Allow users to delete knowledge graph	2		2			
		POC: Query knowledge graph (to help with evaluating it)	8		5			
		Improve visualization based on different node sizes	3		3			
		Experiment with different approaches	5		5			
		View knowledge graph from table list	2		1			
		Graph display text/node color + less overlapping of nodes	8		8			
9	**Enhance graph visualization and LLM-usage**	Clustering of nodes / topic modeling: attributes	5		5			
		Link entities to page	3		3			
		Run linting and fix any errors	2		2			
		Finetuning of prompt template and ontology - make it more abstract and more concise	3		3			
		Look deeper into centrality measures for making network more concise	3		3			
		Finetune force-based algorithm for node positions	2		5			
		Split view - to show more information on left side, graph on right	3		3			
		“Generate” button to link to generate graph for documents that only have been uploaded	2		3			
		Refactoring: "delete uploaded document" button	3		3			
10	**Graph search functionality and UI improvements**	POC: Graph search with embeddings	8		8			
		Hover over node, return page numbers	3		3			
		Display of most extracted entities	5		3			
		Clustering of nodes / topic modeling: coloring	3		3			
		Support multiple document formats	3		3			
		Find way to improve performance time	5		5			
		After LLM results, eliminate duplicate entities	5		8			
		Responsive web design	2		2			
11	**Finalize graph search and graph visualization**	Demo day slide	2		2			

	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining		
		Demo day video	3		3			
		Bug: missing panel on graph visualization and layout algorithm 4 & 5	5		5			
		Bug: Page numbers start at 0 instead of 1	1		1			
		Notification for delete from graph list	2		2			
		Work on graph clustering/coloring ambiguity issue	3		3			
12	**Finish final project release and prepare for demo day**	Cache graph visualization	5		5			
		Follow up on graph search w/ embeddings POC	5		8			
		Minimize topics legend	2		2			
		Bug: page numbers don't always show on hover	3		3			
		Bug: wrong file type error unclear and too long	2		2			
		Clean-up codebase	1		3			
		HW: Finalize user, (technical) design, and build/deploy documentation	3		3			
		End-to-end testing of application and features	1		1			
		Finalize demo day workflow	3		3			
		"Home, Upload, About" - either have it or not. Also, if we keep it, it should be center-aligned	2		2			
		Time formatting	1		1			



#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
	Acceptance criteria is satisfied	Release tag candidate builds and deploys properly	Project builds and deploys properly
	Pull request to dev branch	All previous working features should still work properly	Proper documentation on how to use and build the project is done
	Code-reviewed by peer		
	Approve code and merge into dev branch		
	Automated tests are run and passed		
	When necessary, update software architecture diagram/documentation and bill of materials		

Type	Link / reference
User Documentation	https://github.com/amosproj/amos2024ss05-knowledge-graph-extractor/wiki/User-Documentation
Design Documentation	https://github.com/amosproj/amos2024ss05-knowledge-graph-extractor/blob/main/Documentation/design-documentation.pdf
Build/Deploy Documentation	https://github.com/amosproj/amos2024ss05-knowledge-graph-extractor/blob/main/Documentation/user-documentation.pdf

	Context	Name	Version	License	Comment
1	Splitting text into chunks	LangChain	v0.1.17	MIT	Extract text from input and chunks
2	Working with the data	pandas	v2.2.2	new BSD	
3	Generating graph from data	NetworkX	v3.3	new BSD	python package, this version requires Python 3.10, 3.11, or 3.12.
4	Upload documents	Filepond	4.31.1	MIT	
5	Visualization	Vis.js	v9.1.9.	Apache 2.0 / MIT	
6	Operational database	Postgres	16.2	PostgreSQL license (similar to MIT)	
7	LLM (more powerful option)	Gemini	1.5	Google API Terms of Service	might switch to this LLM from the original one
	LLM (more powerful option)	llama3	llama3-8b-8192	Groq API Terms of Service	
8	Topic modeling	bertopic	0.16.2	OSI Approved :: MIT License	
9	semantic search	SBERT.net	Model: all-mpnet-base-v2	Apache 2.0	SentenceTransformer
10	Vector store	Faiss	1.7.3	MIT	

Last Name	First Name	Value					
Ramesh	**Sandeepkumar**	2		**2.00**	**OK**		
Hoffmann	**Florian**	2					
Rauscher	**Nikolas**	2					
Ozseker	**Irem**	2					
Bhesaniya	**Yash**	2		0	No size		
Fabian Borges	**Filipe Alexandre**	2		1	Trivial size		
Kotini	**Kristi**	2		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