

Project Name	Ailixir - AI Agent Definition and Generation Framework
Online team meeting	https://fau.zoom-x.de/j/66967665829
Production system (if any)	
Test system (if any)	Github Actions: https://github.com/amosproj/amos2024ss06-health-ai-framework/tree/7f68876bb8cb0176c2791674ac83e60a27d3d715/.github
Github repository	https://github.com/amosproj/amos2024ss06-health-ai-framework
Github feature board	https://github.com/orgs/amosproj/projects/53
Github impediments backlog	https://github.com/orgs/amosproj/projects/64
Team T-shirt (white)	https://www.shirtinator.de/s/nDefN-9-QpugOXEh5oJIOA
Team T-shirt (black)	https://www.shirtinator.de/s/Xeu6K1B5RrSuf5lw05rzSg
Additional materials	
Team mailing list	oss-amos-proj6@lists.fau.de

Last Name	First Name	GitHub User Name	Email Address	Comment
Enescu	David	EnescuDavid	david.enescu@fau.de	
Karategos	Georgios	tubamos	amosproject@proton.me	Email used ONLY for GitHub and the Happy-AMOS tool.
Negasa	Gemechis	Ghemechis	gemechismelkamu@gmail.com	
Potthoff	Jan	potthoffjan	potthoff.jan@googlemail.com	
Sandt	Eloi	eloinoel	eloi.sandt@campus.tu-berlin.de	
Varga	Lukáš	lukas-varga	lukas.varga128@gmail.com	
Zimmermann	Simon	Tims777	tim.simon.zimmermann@fau.de	
Gupta	Manik	manikg08	manik.gupta@fau.de	
Vadaliya	Preet	preetvadaliya	preet.vadaliya@fau.de	

#	Meeting Day	Product Owners	Software Developer	Release Manager	Scrum Master	Comment
	2024-04-17	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	N/A	Simon	
1	2024-04-24	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	David	Simon	irr
	2024-05-01	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Manik	Simon	irr
2	2024-05-08	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Preet	Simon	-
3	2024-05-15	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Jan	Simon	-
4	2024-05-22	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Eloi	Simon	irr
5	2024-05-29	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Lukáš	Simon	irr
6	2024-06-05	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Preet	Simon	Mid-term due
7	2024-06-12	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Manik	Simon	-
8	2024-06-19	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	David	Simon	-
9	2024-06-26	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Jan	Simon	-
10	2024-07-03	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Eloi	Simon	irr
11	2024-07-10	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Lukáš	Simon	irr
12	2024-07-17	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Jan	Simon	Demo day!
13	2024-07-24	George, Gemechis	David, Jan, Eloi, Lukáš, Manik, Preet	Preet	Simon	Retrospective

Goals	<ul style="list-style-type: none"> - learn and share learning on tech and collaboration tasks - keep Andi Zink happy - finish tasks for each sprint on time (or communicate early why it isn't possible or if you need help)
Meeting norms	<ul style="list-style-type: none"> - be on time - be informed on topics relevant for the meeting (eg. lecture) - if you are late or can't make it, give a heads up (eg. on slack) - active participation - let's not digress too much while key points haven't been gone through yet
Working norms	<ul style="list-style-type: none"> - ask for help - decisions made in mutual agreement - deliver constructive criticism with the goal of helping others and yourself grow - for code: settle on conventions - good comments/documentation of work - be considerate of varying sets of experiences and skills
Coordination norms	<ul style="list-style-type: none"> - the Scrum Master keeps the meeting on track and PO leads the meeting - the POs (at least, devs are always welcome to join) meet with the partner at regular intervals and communicate / clarify feature requests to the rest of the team.
Communication norms	<ul style="list-style-type: none"> - be welcoming and nice - don't interrupt others - communicate illnesses and expected work downtime and what it means for others - communication through slack student channel
Consideration norms	<ul style="list-style-type: none"> - if I disagree, I should think of an alternative/provide constructive reasoning - both sides should be heard and find a consensus together - prefer to speak in public over speaking privately
Cont. improvement norms	<ul style="list-style-type: none"> - performance evaluated based on weekly sprint target compared with git pull requests - automated tests?
Rewards	<ul style="list-style-type: none"> - after successful release, we can go to mensa (separately in Erlangen / Berlin) - consider meeting in person after the project is completed (in Berlin or Erlangen)
Sanctions	<ul style="list-style-type: none"> - Start with a warning. Persistent issues can lead to report.
Signatures	
Scrum Master	Simon Zimmermann
Product owner	Georgios Karategos
Product owner	Gemechis Negasa
Software developer	Manik Gupta
Software developer	David Enescu
Software developer	Eloi Sandt
Software developer	Lukáš Varga

Software developer	Preet Vadaliya
Software developer	Jan Potthoff

Product Vision	Project Mission
<p>Ailixir empowers users to create and incrementally refine custom AI agents that are specialized in specific domains. It assists users in gaining useful, reliable and timely answers to domain specific questions they are interested in, eventually becoming a dependable companion in the journey to navigate effectively within the field of their choice.</p>	<p>The mission of Ailixir is to produce an MVP for a prototyping tool that allows users, who are entrepreneurs / developers to create, refine and compare the results of custom AI agents. Ailixir can be thought of a combination of three pieces that come together to achieve its goals:</p> <ul style="list-style-type: none">-- The first one aims to create an automated, modular pipeline of acquiring, storing and generating current contextual information from handpicked knowledge sources.-- The second one uses the acquired data and by utilizing a data pipeline which is modular enough to change various parameters, it produces answers independently of the underlying components. This allows the user to tweak parameters or replace components with the aim of finding an optimum combination that produces scientifically accurate and useful results. Important at this step is the ability to trace the sources that were used to generate the results.--The final piece of the project is the creation of a user-facing app that allows users to interact with the data pipeline via modalities such as text and voice.

Sprint #	Sprint goal
1	None
2	None
3	None
4	Progress data acquisition and storage functionality, start with Langchain and app framework
5	Finalise data acquisition functionality, progress langchain and set up the basics for app
6	Finalise data acquisition, finalise basic LangChain pipeline and continue setting up the app
7	Develop basic app components
8	Develop app components
9	Develop advanced app components
10	Prompt engineering & Finetuning
11	Implementation of multiple LLMs
12	Voice modality stability, handoff
13	AMOS Closing

Term	Definition
Config	a way to configure a handpicked list of sources to be used as targets for data scrapping or access to 3rd party APIs
Orchestrator	A piece of logic that coordinates / orchestrates the data acquisition process from the scrapping targets / API providers based on the configurator
Embeddings	numerical representations of data (such as words) that capture their semantic meanings and relationships, enabling their processing and analysis by machine learning algorithms
Text Chunking	the process of dividing a body of text into smaller pieces, like sentences or paragraphs, to facilitate the processing by NLP models
VectorStore	a specialized database that stores high-dimensional vectors, enabling efficient similarity searches and retrievals based on the semantic content of the data
RAG	(Retrieval-Augmented Generation): a technique that combines retrieval of relevant documents from a database with generative capabilities to improve the accuracy and relevance of generated text
LangChain	a framework for developing applications powered by language models, enabling the integration of these models with various data sources and computational workflows.
Hugging Face	a platform that hosts tools, pre-trained models and datasets useful for building AI enabled applications

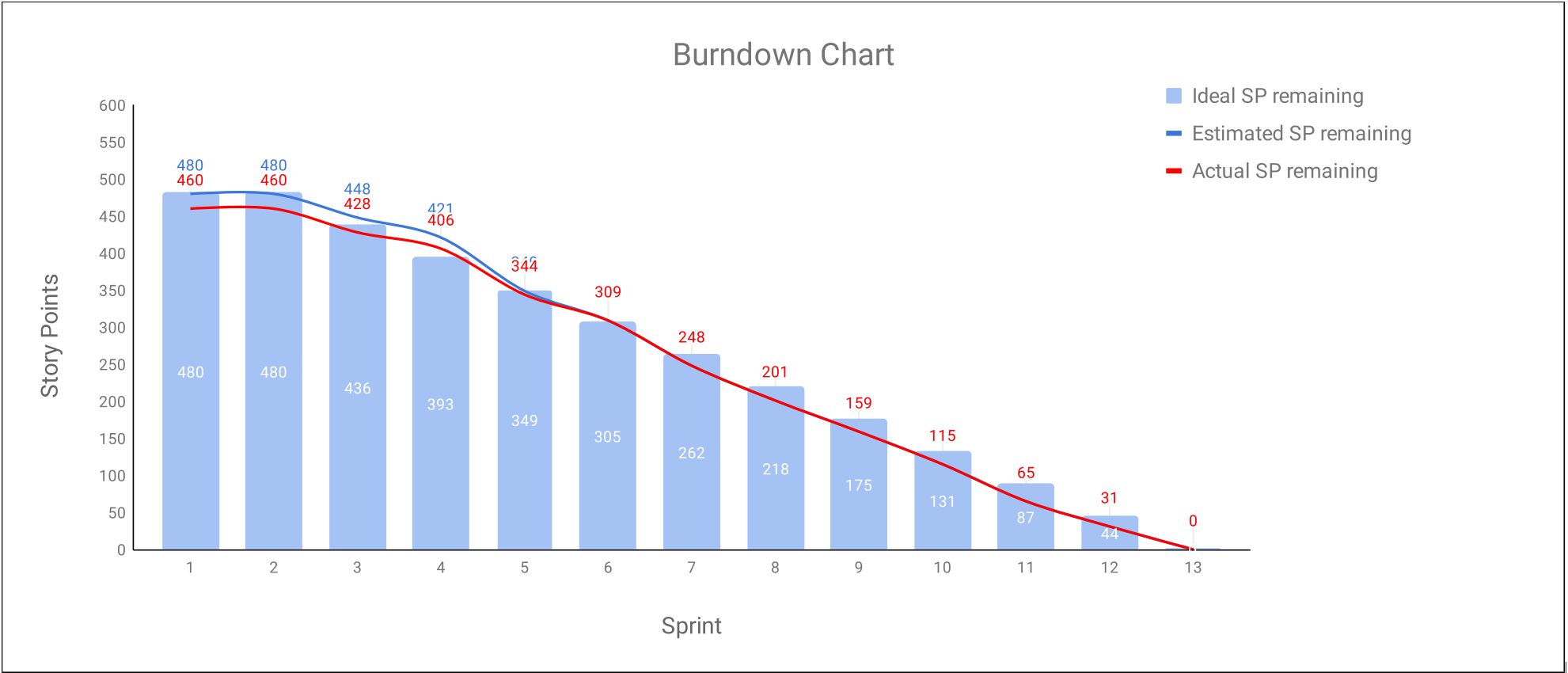
					Est. Size	Est. Remaining	Real Size	Real Remaining
Mid Project Release					232	232	212	212
Sprints 01-06					232	232	212	212
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est. BD	Est. BD	Real BD	Real BD
1	2024-04-17	2024-04-24	AMOS Kickoff		0	232	0	212
2	2024-04-24	2024-05-08	Project set-up and start with data acquisition		32	232	32	212
3	2024-05-08	2024-05-15	CI/CD pipeline set-up, progress with data acquisition and local storage		27	200	22	180
4	2024-05-15	2024-05-22	Progress data acquisition and storage functionality, start with Langchain and app framework		72	173	62	158
5	2024-05-22	2024-05-29	Finalise data acquisition functionality, progress langchain and set up the basics for app		40	101	35	96
6	2024-05-29	2024-06-05	Finalise data acquisition, finalise basic LangChain pipeline and continue setting up the app		61	61	61	61
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est.Size	Real Size		
1	2024-04-17	2024-04-24	AMOS Kickoff					
				No items / commits	0		0	
					0		0	
2	2024-04-24	2024-05-08	Project set-up and start with data acquisition					
				Extract transcripts from podcasts	8		13	
				Extract transcript from a single YouTube video locally	8		3	
				Extend transcript extraction from YouTube	5		5	
				Create concept for the sources input functionality (config)	5		5	
				Setup Development Environment	3		3	
				Project Setup	3		3	
3	2024-05-08	2024-05-15	CI/CD pipeline set-up, progress with data acquisition and local storage					
				Create storage solution for extracted data in the server of the FAU HPC server	8		3	
				Create CI/CD pipeline for Github	3		3	
				Extract data from academic repository PubMed Central (PMC) of the NIH	8		8	
				Implement the "orchestrator" functionality	8		8	
4	2024-05-15	2024-05-22	Progress data acquisition and storage functionality, start with Langchain and app framework					
				Enhance Allrecipes scraper to include the latest orchestrator methods	3		3	
				Migration of the YouTube data extraction pipeline to GC	8		3	
				LangChain - Query Contruction	8		5	
				LangChain - get on the same page	5		5	
				Extract recipe data from recipe website Allrecipes	8		8	
				Extract text content from academic repository: Arxiv	8		5	
				Create the DB for extracted data in Google Cloud	8		5	
				Enhance Arxiv scraper to include the new orchestrator methods	3		5	
				Enhance YouTube scraper to include the new orchestrator methods	3		5	
				Familiarise with the RAG technique and the LangChain framework	5		5	
				Extract blog posts from the website Nutritionfacts	8		8	
				Create system architecture for the app	5		5	
5	2024-05-22	2024-05-29	Finalise data acquisition functionality, progress langchain and set up the basics for app					
				LangChain - Basic retrival and generation	8		8	
				Create frontend interaction wireframes for the user app	8		8	
				Finalize LangChain query construction	5		3	
				Create build process video	5		5	

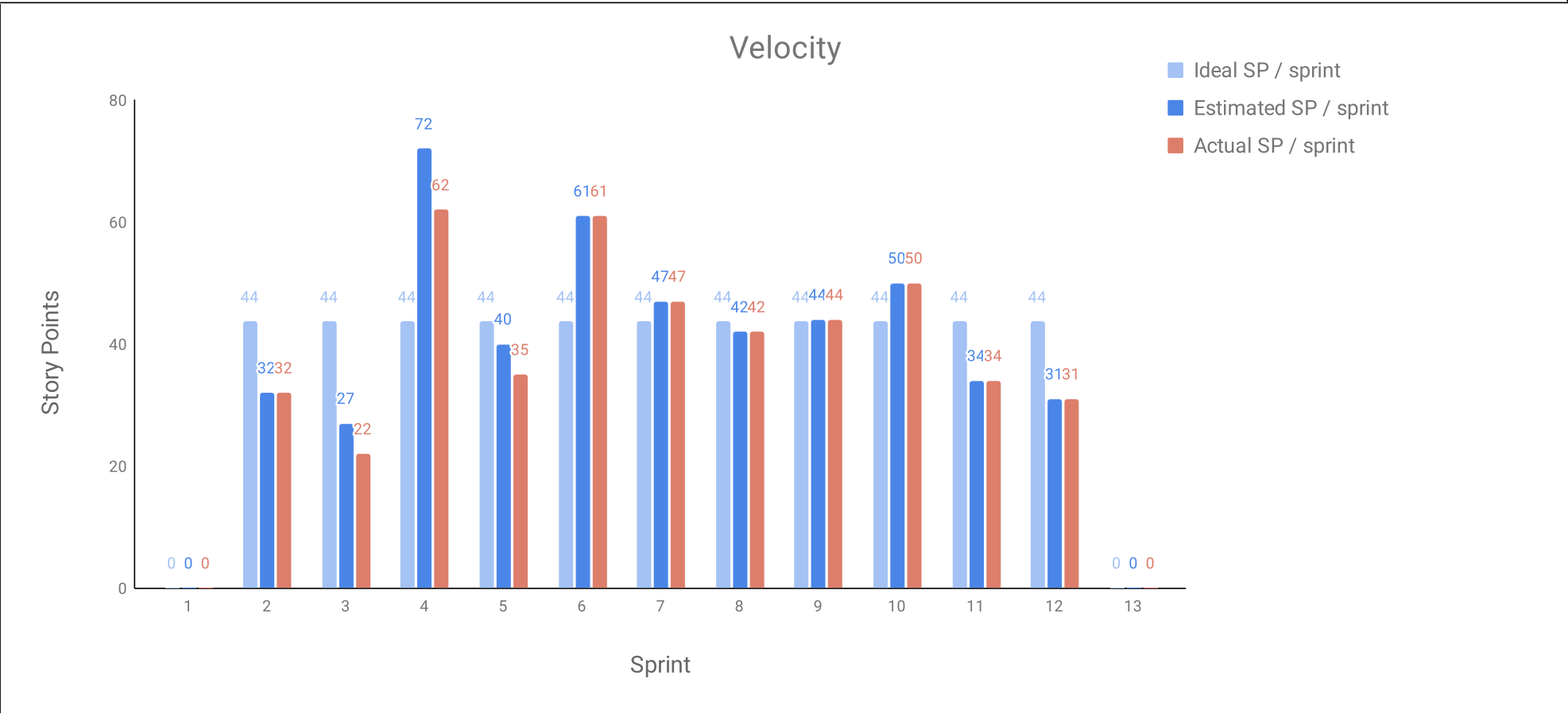
				Est. Size	Est. Remaining	Real Size	Real Remaining
Mid Project Release				232	232	212	212
Sprints 01-06				232	232	212	212
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est. BD	Real BD	
1	2024-04-17	2024-04-24	AMOS Kickoff		0	232	0
2	2024-04-24	2024-05-08	Project set-up and start with data acquisition		32	232	32
3	2024-05-08	2024-05-15	CI/CD pipeline set-up, progress with data acquisition and local storage		27	200	22
4	2024-05-15	2024-05-22	Progress data acquisition and storage functionality, start with Langchain and app framework		72	173	62
5	2024-05-22	2024-05-29	Finalise data acquisition functionality, progress langchain and set up the basics for app		40	101	35
6	2024-05-29	2024-06-05	Finalise data acquisition, finalise basic LangChain pipeline and continue setting up the app		61	61	61
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est.Size	Real Size	
				Error handling in the orchestrator	5		3
				Extend the "save" functionality of the base-scraper	5		2
				Integrate the podcast scraper into the latest version of the orchestrator	3		5
				Create automated PR	1		1
6	2024-05-29	2024-06-05	Finalise data acquisition, finalise basic LangChain pipeline and continue setting up the app				
				Create short video as status update for IP and the Mid-Project product demo	3		3
				Implement the LangChain pipeline in Typescript	8		8
				Complete the Wireframe task to include the document download functionality	3		3
				Setup a Firestore within Firebase	8		8
				Write unit tests for orchestrator/config functionality	5		5
				(data acquisition) Integrate the nutritionfacts blog posts scraper into the new orchestrator	3		3
				Setup react-native dev-environment	8		8
				Implement the appropriate JSON chunking technique	5		5
				Integrate the NIH-PubMed scraper into the new orchestrator structure	5		5
				Patch for the PubMed Scraper pdf downloading problems	3		3
				Catch errors from missing api-keys in .env file	1		1
				Orchestrator: automatically create data/ folder and config.json	1		1
				Setup and write project documentation in GitHub Wiki	8		8

				Est. Size	Est. Remaining	Real Size	Real Remaining
Complete Project				480	232	460	460
Final Project Release				248	248	248	248
Sprints 07-12				248	248	248	248
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est. BD	Real BD	Real BD
7	2024-06-05	2024-06-12	Develop basic app components		47	248	47
8	2024-06-12	2024-06-19	Develop app components		42	201	42
9	2024-06-19	2024-06-26	Develop advanced app components		44	159	44
10	2024-06-26	2024-07-03	Prompt engineering & Finetuning		50	115	50
11	2024-07-03	2024-07-10	Implementation of multiple LLMs		34	65	34
12	2024-07-10	2024-07-17	Voice modality stability, handoff		31	31	31
13	2024-07-17	2024-07-24	AMOS Closing		0	0	0
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est. Size	Real Size	Real Size
7	2024-06-05	2024-06-12	Develop basic app components				
				Add a logging and reattempt functionality for failed scrapping tasks	5		5
				Save question-answer pair as a formatted txt	5		5
				Setup simple user authentication	8		8
				Create components for login with email	8		8
				Create components for basic visual navigation	8		8
				Create components for the chat screen	8		8
				Scrum spike: Research prompt engineering techniques	5		5
8	2024-06-12	2024-06-19	Develop app components				
				Develop the components for the settings interface	8		8
				Develop the components for presenting the references in the answers	13		13
				Develop functionality to export conversation history as a text file	8		8
				Develop the components for conversation history	8		8
				Update user / wiki documentation with new app functionality	5		5
9	2024-06-19	2024-06-26	Develop advanced app components				
				Develop components for multiple assistants	8		8
				Develop search feature	13		13
				Develop Issue logging for internal auditing	8		8
				Research Spike: Fine tuning techniques	5		5
				Update user / wiki documentation with new app functionality	5		5
				Create functionality for storing custom context coming from manual user input	5		5
10	2024-06-26	2024-07-03	Prompt engineering & Finetuning				
				Implement basic CoT prompt engineering technique	8		8
				Implement basic reACT prompt engineering technique	8		8
				Implement basic DSP prompt engineering technique	8		8
				Combine prompt engineering techniques	13		13
				Answer tagging for good / bad answers and retraining based on the tag	5		5

				Est. Size	Est. Remaining	Real Size	Real Remaining
Complete Project				480	232	460	460
Final Project Release				248	248	248	248
Sprints 07-12				248	248	248	248
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est. BD	Real BD	Real BD
7	2024-06-05	2024-06-12	Develop basic app components		47	248	47
8	2024-06-12	2024-06-19	Develop app components		42	201	42
9	2024-06-19	2024-06-26	Develop advanced app components		44	159	44
10	2024-06-26	2024-07-03	Prompt engineering & Finetuning		50	115	50
11	2024-07-03	2024-07-10	Implementation of multiple LLMs		34	65	34
12	2024-07-10	2024-07-17	Voice modality stability, handoff		31	31	31
13	2024-07-17	2024-07-24	AMOS Closing		0	0	0
Sprint	Start Date	End Date	Sprint Goal	Feature Name	Est. Size	Real Size	Real Size
				Improve responses by optimizing parameters in the embeddings	8		8
11	2024-07-03	2024-07-10	Implementation of multiple LLMs				
				Develop functionality for selecting different assistants	8		8
				Develop functionality for switching the LLM behind the assistant	8		8
				Switch prompt setups based on Agent / LLM selection	13		13
				Apply a saved question to a different agent / LLM setup	5		5
12	2024-07-10	2024-07-17	Voice modality stability, handoff				
				Develop components for voice interaction	8		8
				Update user / wiki documentation with new app functionality	5		5
				Bug fixing	13		13
				Update user / wiki documentation for Handoff	5		5
13	2024-07-17	2024-07-24	AMOS Closing				
				Project Report			
				Project Retrospective			

Sprint	Sprint Goal	Sprint Start Date	Sprint Finish Date	Ideal SP / sprint	Ideal SP remaining	Estimated SP / sprint	Estimated SP remaining	Actual SP / sprint	Actual SP remaining
1	AMOS Kickoff	2024-04-17	2024-04-24	0	480	0	480	0	460
2	Project set-up and start with data acqui	2024-04-24	2024-05-08	44	480	32	480	32	460
3	CI/CD pipeline set-up, progress with dat	2024-05-08	2024-05-15	44	436	27	448	22	428
4	Progress data acquisition and storage f	2024-05-15	2024-05-22	44	393	72	421	62	406
5	Finalise data acquisition functionality, p	2024-05-22	2024-05-29	44	349	40	349	35	344
6	Finalise data acquisition, finalise basic	2024-05-29	2024-06-05	44	305	61	309	61	309
7	Develop basic app components	2024-06-05	2024-06-12	44	262	47	248	47	248
8	Develop app components	2024-06-12	2024-06-19	44	218	42	201	42	201
9	Develop advanced app components	2024-06-19	2024-06-26	44	175	44	159	44	159
10	Prompt engineering & Finetuning	2024-06-26	2024-07-03	44	131	50	115	50	115
11	Implementation of multiple LLMs	2024-07-03	2024-07-10	44	87	34	65	34	65
12	Voice modality stability, handoff	2024-07-10	2024-07-17	44	44	31	31	31	31
13	AMOS Closing	2024-07-17	2024-07-24	0	0	0	0	0	0





Type	Link / reference
Product Wiki	https://github.com/amosproj/amos2024ss06-health-ai-framework/wiki

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
1	The feature has been fully implemented.	All features planned for the sprint are integrated and functioning as expected.	The app functions correctly and responds appropriately to user inputs.
2	The feature has been manually tested and works as expected without critical bugs.	A manual check confirms that all integrated features work together without critical issues.	Design documentation is updated to reflect the final architecture and system design.
3	The feature code is documented with clear explanations of its functionality and usage.	The app builds successfully and can be deployed.	User documentation is updated to provide guides and instructions on using the app.
4	The feature code has been reviewed and approved by at least one team member.	Database update scripts are successful.	The project release is approved by the product owners and industry partner.
5	The feature branches have been merged into the main branch and closed.	Sprint release notes are written, summarising the new features, changes, and improvements.	
6		The changelog is updated to reflect all modifications made during the sprint.	

#	Context	Name	Version	License	Comment
1	development tooling	ruff	0.4.2	MIT	linter and code formatter
2	development tooling	pre-commit	3.7.0	MIT	pre-commit hooks
3	development tooling	python-dotenv	1.0.1	BSD License (BSD-3-Clause)	reads key-value pairs from a .env file
4	data pipeline development	python-youtube	0.9.4	MIT	youtube data scrapping, wrapper around for YouTube Data API V3.
5	data pipeline development	bs4	0.0.2	MIT License	Website scraping. Parses html and provides functionality to process the html
6	data pipeline development	pydub	0.25.1	MIT License (MIT)	To work on audiofiles
7	data pipeline development	ffmpeg	1.4	LGPL 2.1+/ GPL 2+	Needed to convert mp3 to wav
8	data pipeline development	wave	0.0.2	GNU Library or Lesser General Public License (LGPL)	work with wav files
9	data pipeline development	vosk	0.3.45	Apache Software License	Provides models for speech recognition and transcription
10	data pipeline development	paperscraper	0.2.11	MIT	Scraping text data from pdfs
11	data pipeline development	biopython	1.83	Freely Distributable	Pubmed API access, search queries, metadata scraping
12	data pipeline development	pypdf	4.2.0	BSD	Downloading pdfs from doi links (pubmed)
13	data pipeline development	requests	2.32.2	Apache 2.0	Requests allows you to send HTTP requests for download vosk transcription model
14	data pipeline development	arxiv	2.1.0	MIT	Webscraping tool API for Arxiv
15	deployment tool	apptainer	1.3.1	BSD	container tool for HPC (potentially) or Google Cloud
16	retrieval augmented generation	openai	1.30.1	Apache 2.0	Access to the OpenAI REST API
17	retrieval augmented generation	langchain	0.2.0	MIT	Framework for developing LLM
18	retrieval augmented generation	langchain-community	0.2.0	MIT	Framework for developing LLM
19	retrieval augmented generation	langchain-google-genai	1.0.4	MIT	Google integrations with LangChain
20	retrieval augmented generation	chromadb	0.5.0	Apache 2.0	AI-native open-source embedding database
21	retrieval augmented generation	vertexai	1.49.0	Apache 2.0	Platform for data science and machine learning
22	retrieval augmented generation	tiktoken	0.7.0	MIT	Tokeniser for OpenAI's models
23	retrieval augmented generation	sentence-transformers	2.7.0	Apache 2.0	Multilingual Sentence & Image Embeddings with BERT