AMOS P4 - Planning Document Project Data

Project Name	Kubernetes Inventory Taker
Online team meeting	https://fau.zoom.us/j/66067362855
Production system (if any)	not available
Test system (if any)	not available
GitHub repository	https://github.com/amosproj/amos2023ss04-kubernetes-inventory-taker
GitHub feature board	amos2023ss04-feature-board (github.com)
GitHub impediments backlog	amos2023ss04-impediments-backlog (github.com)
Team T-shirt (white, male)	https://www.shirtinator.de/t-shirts/gestalten/t-shirt-bedrucken#/load/share/87425b24-a18c-4528-b35d-81df816ca277
Team T-shirt (black, male)	https://www.shirtinator.de/t-shirts/gestalten/t-shirt-bedrucken#/load/share/42541114-1bf4-4ca6-89c2-2040abfcfe20
Team T-shirt (white, female)	https://www.shirtinator.de/t-shirts/gestalten/t-shirt-bedrucken#/load/share/05d839cc-d89e-4063-a1e2-a2ddb9deff98
Team T-shirt (black, female)	https://www.shirtinator.de/t-shirts/gestalten/t-shirt-bedrucken#/load/share/077e16f6-e925-4331-9371-fb165b676607
Additional materials	

AMOS P4 - Planning Document Project Team

Last Name	First Name	GitHub User Name	Email Address	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Degen	Jan	jandegen	jan.degen@fau.de							
Kramer	Philipp	PhlppKrmr	philipp.kramer@fau.de							
Zabka	Stefan	vringar	zabka@campus.tu-berlin.de	Unavailable	Unavailable	Actively Working	Actively Working			
Jochens	Nikolas	njochens	nj@andaco.de							
Ali	Muhammad Fahad	muhdfahadali	mfahad.ali@fau.de	Actively Working	Actively Working	Actively Working	Unavailable	Unavailable		
Gehrlein	Lucas	lifesbest23	I.gehrlein@campus.tu-berlin.de	Actively Working				Unavailable		Actively Working
Wiesend	Simon	smnws	simon.wiesend@fau.de							
Kwiatek	Anna	IceFlovver	anna.kwiatek@fau.de							
Aziri	Ali	freefreep	ali.aziri@campus.tu-berlin.de							

AMOS P4 - Planning Document

Role Assignments

#	Meeting Day Product Owner	Software Developer	Release Manager	Scrum Master	Comment
1	2023-04-12 Philipp Kramer, Jan Degen	Everyone else	N/A	Anna Kwiatek	
2	2023-04-19 Philipp Kramer, Jan Degen	Everyone else	Lucas Gehrlein	Anna Kwiatek	
3	2023-04-26 Philipp Kramer, Jan Degen	Everyone else	Stefan Zabka	Anna Kwiatek	
4	2023-05-03 Philipp Kramer, Jan Degen	Everyone else	Stefan Zabka	Anna Kwiatek	
5	2023-05-10 Philipp Kramer, Jan Degen	Everyone else	Ali Aziri	Anna Kwiatek	
6	2023-05-17 Philipp Kramer, Jan Degen	Everyone else	Nikolas Jochens	Anna Kwiatek	
7	2023-05-24 Philipp Kramer, Jan Degen	Everyone else	Nikolas Jochens	Anna Kwiatek	Mid-term due
8	2023-05-31 Philipp Kramer, Jan Degen	Everyone else	Fahad Ali	Anna Kwiatek	
9	2023-06-07 Philipp Kramer, Jan Degen	Everyone else	Simon Wiesend	Anna Kwiatek	
10	2023-06-14 Philipp Kramer, Jan Degen	Everyone else	Lucas Gehrlein	Anna Kwiatek	
11	2023-06-21 Philipp Kramer, Jan Degen	Everyone else	Simon Wiesend	Anna Kwiatek	
12	2023-06-28 Philipp Kramer, Jan Degen	Everyone else	tbd.	Anna Kwiatek	
13	2023-07-05 Philipp Kramer, Jan Degen	Everyone else	tbd.	Anna Kwiatek	
14	2023-07-12 Philipp Kramer, Jan Degen	Everyone else	tbd.	Anna Kwiatek	Demo day!
15	2023-07-19 Philipp Kramer, Jan Degen	Everyone else	tbd.	Anna Kwiatek	Retrospective

AMOS P4 - Planning Document Team Contract

	Cat in tauch with Kuhamatan Dadien
	- Get in touch with Kubernetes, Docker
	- Everyone feels comfortable working with it
	- Earning something
	- Deliver working application
	- Satisfy the client
Goals	- We succeed as a team or we fail as a team
Goals	- We succeed as a team of we fail as a team
	- Create meeting protocol, maintainer is the current release manager
	- Zoom for Teammeeting 12:30
	- Dont use Zoom Chat, instead use Mattermost channel <u>Teammeeting</u>
Meeting norms	- Stay within meeting schedule, dont extend meetings heavily
	- Help each other
	- Main branch is runnable
	- No direct pushes to main
	- Code is reviewed by other software developers
	- Everyone picks up open active topics proactivly
Working norms	- Every developer has to do at least 1 story point per sprint
Working norms	- Every developer has to do at least 1 story point per sprint
	- Decline meetings if you're not able to joiin
	- Inform team if you're late
Coordination norms	- Use comments on GitHub issues for coordination on user stories
	Note the second
	- Mattermost for team internI communication
	- We interact polity and respectfully with each other
	- Everyone can always speak up and state their mind
	- In case of problems talk to each other
Communication norms	- Cameras are always on with happy faces
	Camerate and annually of the manual paper in the camerate and the camerate
	- Less tasks for people with exams
	- Majority vote for hard decisions
Consideration norms	- Every opinion is weighted equally
Consideration norms	- Every opinion is weighted equally
	- Perform retrospective after sprints
Cont improvement nerms	
Cont. improvement norms	- Talk openly about what went wrong respectful manner
Rewards	Cood aprinto divo recognish hudgets which is apont an recognish project related stuff
Newalus	- Good sprints give research budgets which is spent on reasearch project related stuff
Sanctions	- Sad smileys for late people
Janoudia	- Sau sittieys for late people
Signature	Jan Degen
Olgilatule	oan begen

AMOS P4 - Planning Document Team Contract

Stefan Zabka
Muhammad Fahad Ali
Lucas Gehrlein
Nikolas Jochens
Simon Wiesend
Philipp Kramer
Anna Kwiatek

AMOS P4 - Planning Document Product Goal

Product Vision	Project Mission
Our vision is to create a tool that empowers DevOps teams to manage their Kubernetes clusters with ease, providing them with a single platform for inventory visibility, deep insights, real-time analytics, and intelligent monitoring and alerting. With KIT, teams can optimize their resources and streamline their operations, resulting in faster deployment times, better performance, and happier customers.	Product Mission for KIT: Simplifying Kubernetes Management and Monitoring for DevOps Teams At KIT (Kubernetes Inventory Taker), our mission is to simplify Kubernetes management and monitoring for DevOps teams. We want to provide a user-friendly tool that gives you a clear, real-time view of your Kubernetes resources, from deployments and pods to containers, so you can easily manage and optimize your infrastructure. Here's what we're all about: - Inventory Visibility: We'll provide you with a comprehensive view of your Kubernetes inventory, showing you the state, health, and configuration of your deployments, pods, and containers in a single, easy-to-navigate interface. No more complex logs or guesswork - we'll make it simple and intuitive for you to understand your resources Insights and Analysis: We'll help you gain deep insights into your Kubernetes resources with powerful analytics and analysis. Get real-time information on resource utilization, health status, and configuration changes, so you can quickly identify and resolve issues, track changes over time, and optimize your resources for better performance User-friendly Web Frontend: We believe in making KIT easy to use and visually

AMOS P4 - Planning Document Product Glossary

Term	Definition
	Project Name short for Kubernetes Inventory Taker
KIT	short for Kubernetes Inventory Taker

Sprint	Theme	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Releas	e						
	Total			79	49	75	
Sprints							
1	Project kick-off	Get in touch with the team, project and AMOS		0	79	0	75
2	Project Setup	Create application architecture and setup project components		21	79	22	75
3	Application Domain Modelling	Model the application domain as well as create drafts for the UI		19	58	17	53
4	Application Domain Model Implementation & CICD architecture	Implement the first components of the application domain and prepare CICD pipeline		23	39	23	36
5	CICD, Testing and UI design			16	16	13	13
Feature	98						
1	Project kick-off	Get in touch with the team, project and AMOS					
			Create & agree on team contract Initialize planning documents				
			Fill happiness index				
			Design team logo				
			Submit team T-Shirt size preference				
2	Project Setup	Create application architecture and setup project components					
			Initialize base Website	8		8	
			Initialize database	2		1	
			Initialize proxy container	3		5	
			Introduction to used technologies	8		8	

Corint	Thoma	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Sprint	Theme	Goal	Create software architecture	ESt. Size	Remaining	Real Size	Remaining
			description				
			Initialize software bill of materials				
3	Application Domain Modelling	Model the application domain as well as create drafts for the UI					
			Create wire draft for UI	5		8	
			Design database scheme	8		5	
			Apply database scheme to database	3		1	
			Initialize project docker-compose	3		3	
4	Application Domain Model Implementation & CICD architecture	Implement the first components of the application domain and prepare CICD pipeline					
			Create base implementation for resource information collection	5		5	
			Create base container detail page	8		8	
			Create coding guidelines	5		5	
			Decide on CICD architecture	5		5	
			Create product vision				
			Create product mission				
			Create definition of done				
5	CICD, Testing and UI design	Setup CICD and testing framework					
			Update wire draft	3		3	
			Adjust database scheme to history approach	5		5	
			Implement CICD	8		5	
			Create mid-release plan				
			Create build process video				

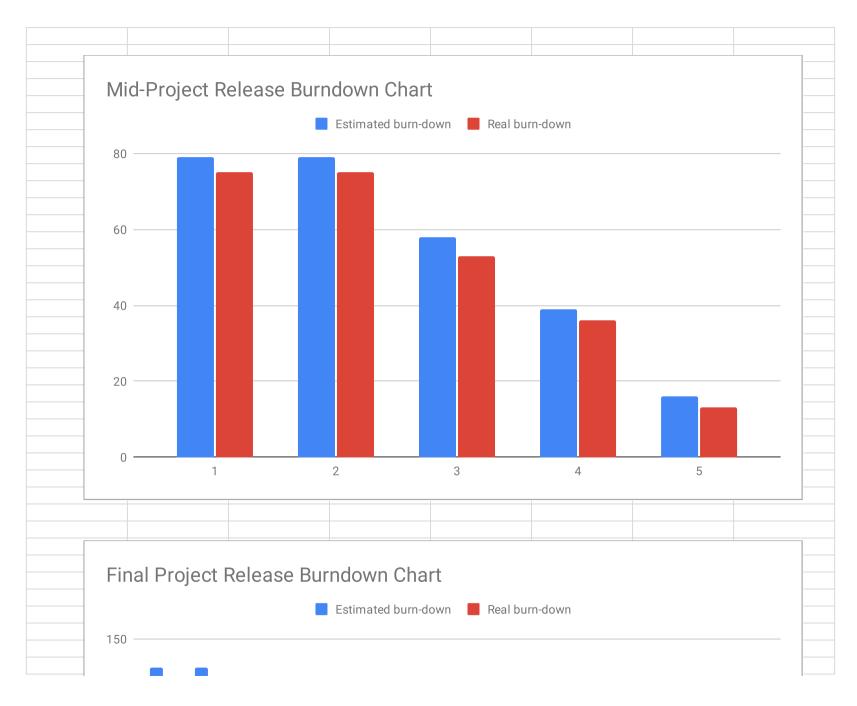
Sprint	Theme	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
Dalasa							
Release	e 						
	Total			136	49	132	57
Sprints							
1	Project kick-off	Get in touch with the team, project and AMOS		0	136	0	132
2	Project Setup	Create application architecture and setup project components		21	136	22	132
3	Application Domain Modelling	Model the application domain as well as create drafts for the UI		19	115	17	110
4	Application Domain Model Implementation & CICD architecture	Implement the first components of the application domain and prepare CICD pipeline		23	96	23	93
5	CICD, Testing and UI design	Setup CICD and testing framework		16	73	13	70
6	Mid Release	Prepare mid release and documentation documents		18	57	16	57
7	Containers & Navigation	Container list detail views shall be implemented and dynamically loaded from database		0	39	0	41
8	Containers & Navigation 2	Container list detail views shall be implemented and dynamically loaded from database		23	39	25	41
9	Containers	Fine-tune containers in proxy and explorer		16	16	16	16
10	Pods	Introduce Pods to the application			0		0
11					0		0
12	Final Release				0		0
13 14	Project Retrospective				0		0
14	rioject Retrospective				U		U
Feature	98						
1	Project kick-off	Get in touch with the team, project and AMOS					
			Create & agree on team contract				

Sprint	Theme	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
			Initialize planning documents				
			Fill happiness index				
			Design team logo				
			Submit team T-Shirt size preference				
			preference				
2	Project Setup	Create application architecture and setup project components					
	<u> </u>		Initialize base Website	8		8	
			Initialize database	2		1	
			Initialize proxy container	3		5	
			Introduction to used technologies	8		8	
			Create software architecture description				
			Initialize software bill of materials				
3	Application Domain Modelling	Model the application domain as well as create drafts for the UI					
			Create wire draft for UI	5		8	
			Design database scheme	8		5	
			Apply database scheme to database	3		1	
			Initialize project docker-compose	3		3	
4	Application Domain Model Implementation & CICD architecture	Implement the first components of the application domain and prepare CICD pipeline					
			Create base implementation for resource information collection	5		5	
			Create base container detail page	8		8	
			Create coding guidelines	5		5	
			Decide on CICD architecture	5		5	
			Create product vision				
			Create product mission				
			Create definition of done				
5	CICD, Testing and UI design	Setup CICD and testing framework					
			Update wire draft	3		3	

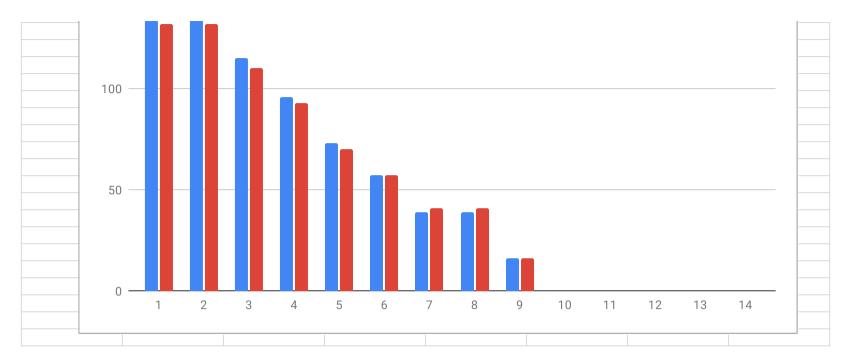
Sprint	Theme	Goal	Feature Name	Est. Size	Est. Remaining	Real Size	Real Remaining
			Adjust database scheme to history approach	5		5	
			Implement CICD	8		5	
			Create mid-release plan				
			Create build process video				
6	Mid Release	Prepare mid release and documentation documents					
			Setup frontend testing	5		5	
			Create health indicator	3		3	
			Generate test data for container & pods within database	5		3	
			Create list with static data (container names)	5		5	
			Create user documentation				
			Create build & deploy documentation				
			Create technical design documentation				
7	Containers & Navigation	Container list detail views shall be implemented and dynamically loaded from database		0		0	
8	Containers & Navigation 2	Container list detail views shall be implemented and dynamically loaded from database					
			Update database schema	5		5	
			Track changes with watchers	8		13	
			Display all containers as list	8		5	
			GHA and Pre-Commit use different configs for golangcilint	2		2	
9	Containers	Fine-tune containers in proxy and explorer					
			Collect metadate about containers	5		5	
			Display health state in list views	5		5	
			Rename tables to lower case	1		1	
			Create sidebar navigation	5		5	
10	Bugs & Container Finalization		Install ESLint when building next. js application	2			

					Est.		Real
Sprint	Theme	Goal	Feature Name	Est. Size	Remaining	Real Size	Remaining
			Use /api instead of marking				
			pages as force-dynamic	0			
			Search containers by name	5			
			Display container metadata in				
			container detail page	5			
			Container Sorting Functionality				
			Not Working Properly After				
			Implementing Search Feature				
			Pre-commit Doesn't Pass All				
			Relevant File Paths To Eslint				
			Add code documentation	3			
			Run all tests in CI	3			

AMOS P4 - Planning Document Burndown Chart



AMOS P4 - Planning Document Burndown Chart



AMOS P4 - Planning Document Definition of Done

Feature Definition of	Done	Sprint Release Definition of Done	Project Release Definition of Done
		sprint-XX-release-candidate release created and	
Colours match with co	olor scheme	marked as pre-release	All required features are implemented
		New Docker images available at Docker	
Implementation is uni	t tested	Container Registry	User manual is ready
Code documentation u	updated (jsdoc or godoc?)	Sprint Review completed	Technical manual is ready
Changes to database of	locumented at Confluence- Database scheme	Sprint Retrospective completed	No bugs
		Pull request to "main" branch merged & closed if	
Changes to architectur	re documented at Confluence - Architecture	there is a new release	
		Real size set for each closed user story	
Feature has been fully	implemented		
All acceptance criteria	a were met		
Product owner approv	red features		
All tests are passing			
Pull request to "develo	op" branch is merged & closed		
Bill of materials is up	dated with new dependencies		

AMOS P4 - Planning Document Documentation

Type	Link / reference

1 F		Name	Version	License	Comment
	Proxy	Kubernetes Client Library	v0.27.1	Apache-2.0	https://github.com/kubernetes/client-go/
2 F	Frontend	Types	multiple	MIT	https://github.com/DefinitelyTyped/DefinitelyTyped/
3 F	Frontend	Next.js	13.3.4	MIT	https://github.com/vercel/next.js
4 F	Frontend	typescript	5.0.4	Apache-2.0	https://github.com/microsoft/TypeScript
5 F	Frontend	tailwindcss	3.3.2	MIT	https://github.com/tailwindlabs/tailwindcss
6 F	Frontend	swr	2.1.5	MIT	https://github.com/vercel/swr
7 F	Frontend	react	18.2.0	MIT	https://github.com/facebook/react
8 F	Frontend	postcss	8.4.23	MIT	https://github.com/postcss/postcss
9 F	Frontend	eslint	8.39.0	MIT	https://github.com/eslint/eslint
10 F	Frontend	autoprefixer	10.4.14	MIT	https://github.com/postcss/autoprefixer
11 F	Frontend	cypress	12.13.0	MIT	https://github.com/cypress-io/cypress
12 F	Frontend	eslint-plugin-cypress			
13 F	Proxy	klog	v1.0.0	Apache-2.0	https://github.com/kubernetes/klog
14 [Proxy	utils	v0.0.0- 202302091 94617- a36077c30 491	Apache-2.0	https://github.com/kubernetes/utils
	Proxy	apimachinery	v0.27.2	Apache-2.0	https://github.com/kubernetes/apimachinery
16 F	Proxy	api	v0.27.2	Apache-2.0	https://github.com/kubernetes/api
17 F	Proxy	go-yaml/yaml.v2	v2.4.0	MIT and Apache-2.0	https://github.com/go-yaml/yaml
18 F	Proxy	uptrace/bun	v1.1.14	BSD 2-Clause	https://github.com/uptrace/bun
19 F	Proxy	golang	v1.20	BSD 3-Clause	https://github.com/golang/go
20 F	Proxy	golangci-lint	v1.53.3	GPL-3.0	https://github.com/golangci/golangci-lint

#	Context	Name	Version	License	Comment

#	Context	Name	Version	License	Comment

#	Context	Name	Version	License	Comment

AMOS P4 - Planning Document Planning Poker

a	First Name	Value			
Degen	Jan		#UIV/	#DIV/	
Kramer	Philipp				
Zabka	Stefan		01	0!	
Jochens	Nikolas				
Ali	Muhammad Fahad		0	No size	
Gehrlein	Lucas		1	Trivial size	
Wiesend	Simon		2	Small size	
Kwiatek	Anna		3	Medium size	
Aziri	Ali		5	Large size	
			8	Very large size	
			13	Too large (size)	