Project Name	amos2022ss02-audit-chain
Online team meeting	https://fau.zoom.us/j/63420352046?pwd=TElTeFZoTDFXNVRYUWVCR1ZpQmVPdz09
Production system (if any)	...
Test system (if any)	...
GitHub repository	https://github.com/amosproj/amos2022ss02-audit-chain
GitHub kanban board (project)	https://github.com/amosproj/amos2022ss02-audit-chain/projects
Team T-shirt (white)	https://www.shirtinator.de/loadBasket/1CxTZ1RYcrz
Team T-shirt (black)	https://www.shirtinator.de/loadBasket/1CxTZ1RYcrz
Additional materials	https://drive.google.com/drive/folders/1Hf5l4I0L0jxXnNfANrMft7RYdwUBlvr4?usp=sharing

Last Name	First Name	GitHub User Name	Email Address
Schmidt	David	Lavicola	david.dav.schmidt@fau.de
Shanabhag	Gajanana	gdshanbhag	gajanana.shanabhag@fau.de
Mazzini	Francesco	francescomazzini	francesco.mazzini@fau.de
Linkies	Sebastian	jaRulez	sebastian.linkies@fau.de
Papadaki	Anastasia	annipap	anastasia.p.papadaki@fau.de
Rehm	Ronja	ronjarehm	ronja.rehm@fau.de
Khalid	Muhammad Ibrahim	ibs337	ibrahim.khalid@fau.de
Srikhaolan	Charinee	CharineeSrikhaolan	Charinee.Srikhaolan@fau.de
D'Ercoli	Chiara	cdercoli	chiara.dercoli@fau.de

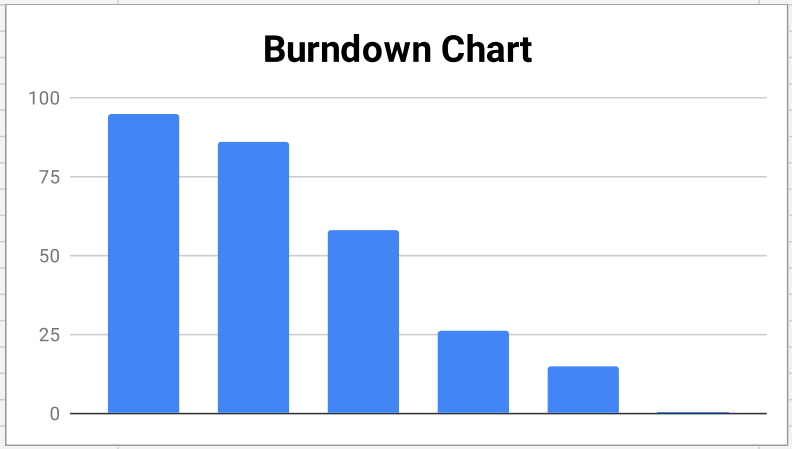
Goals	Learning objectives: Gaining knowledge about agile methods and continous improvement of required skill set (i.e. coding). Moreover, interpersonal relationships are also an important objective. To meet all team members with respect and working a focused but pleasure working atmosphere. Finally, we aim to achieve our defined project goals and work closely and efficient with our industry partner.
Meeting norms	Mandatory team meeting time is on wednesday at 12.30 for our sprint session, which takes 90 minutes. Additionally, we will meet weekly our industry partner to discuss our progress and requirements and meeting eventually a second time per week if required.
Working norms	Team members should discuss objectively and decisions have to be made unanimously. Overall attendance have to above > 80% and team members have to be punctual (not later than 5 minutes!). Criticism should be formulated in a constructive manner and and in case of different opinions, a compriss have to be find. We stick to our definded coding guidelines, our sprint sessions are conducted regarding to the SCRUM rules and in case of major changes in code we will notify all software developers. If team member face troubles with each others, the team is expected to support the individuals and find a solution, which fit to their opinions.
Coordination norms	The Scrum Master has the role of the moderator and support the team in lead us through the agenda of our meetings, the roles of Scrum, approach interpersonal problems and help us to deliver required artifacts on time. The Release manager is responsible for the technical part of every sprint - the software developers agree on the Release manager every sprint.
Communication norms	The sprint sessions on wednesday will be held in Zoom. Apart from our regular meeting, the team communicates via Discord, which has to be checked regularly - the response time should be within 24 hours. Personal communication, in case of urgent matters, is always be possible. Our internal files platform will be a Google Drive folder.
Consideration norms	In case of disagreements the team discusses openly but objectively, all opinions are welcome. The majority of votes has to be made upon the impact of the decision and the urgency, in case of not achieving a compromiss the Scrum master needs to be contacted.
Cont. improvement norms	The team progress is tracked how efficiently the team breaks down problems into tasks and if the team is able to solve all issues in the sprint session. SD's giving each other construtive feedback, if there is a more efficient solution or coding style. In case of good team collaboration (Happiness index) and productive working atmosphere we will go out for a beer.
Rewards	Going out for a beer and making each other compliments.
Sanctions	Recurring laxity in commitment of a team member have to be discussed in the team and the Scrum master tries to motivate the team member. In case of continous decline, we contact the assistants of Prof. Riehle. For acceptable excuses, it is encouraged to notify beforehand any problems are faced.
General Norms	Always be kind, respect each other and discuss problems openly. Ask for if assistance of Prof. Riehle if neccessary.
Signature	Sebastian Linkies, Ronja Rehm, David Schmidt, Gajanana Shanabhag, Francesco Mazzini, Anastasia Papadaki, Ibrahim Khalid, Khaled Saifullah, Charinee Srikhaolan

#	Meeting Day	Uni	Comment	Product Owner	Software Developer	Release Manager	Scrum Master
1	2022-04-27			Ronja Rehm Sebastian Linkies	Everyone else	N/A	Charinee Srikhaolan
2	2022-05-04			Ronja Rehm Sebastian Linkies	Everyone else	Anastasia Papadaki	Charinee Srikhaolan
3	2022-05-11	Yes		Ronja Rehm Sebastian Linkies	Everyone else	Gajanana Shanabhag	Charinee Srikhaolan
4	2022-05-18			Ronja Rehm Sebastian Linkies	Everyone else	Francesco Mazzini	Charinee Srikhaolan
5	2022-05-25	Yes		Ronja Rehm Sebastian Linkies	Everyone else	Francesco Mazzini	Charinee Srikhaolan
6	2022-06-01			Ronja Rehm Sebastian Linkies	Everyone else	Gajanana Shanabhag	Charinee Srikhaolan
7	2022-06-08	Yes	Mid-term due	Ronja Rehm Sebastian Linkies	Everyone else	Anastasia Papadaki	Charinee Srikhaolan
8	2022-06-15			Ronja Rehm Sebastian Linkies	Everyone else	Muhammad Ibrahim Khalid	Charinee Srikhaolan
9	2022-06-22			Ronja Rehm Sebastian Linkies	Everyone else	Francesco Mazzini	Charinee Srikhaolan
10	2022-01-13	Yes		Ronja Rehm Sebastian Linkies	Everyone else	Chiara D'Ercoli	Charinee Srikhaolan
11	2022-01-20			Ronja Rehm Sebastian Linkies	Everyone else	Anastasia Papadaki	Charinee Srikhaolan
12	2022-01-27			Ronja Rehm Sebastian Linkies	Everyone else	Gajanana Shanabhag	Charinee Srikhaolan
13	2022-02-03	Yes		Ronja Rehm Sebastian Linkies	Everyone else		Charinee Srikhaolan
14	2022-02-10		Demo day!	Ronja Rehm Sebastian Linkies	Everyone else		Charinee Srikhaolan
15	2022-02-17		Retrospective	Ronja Rehm Sebastian Linkies	Everyone else		Charinee Srikhaolan

Product Vision	Project Mission
The long-term goal of our project is to develop a new middleware based on blockchain data structures to guarantee the unchanged, compliant, in sequence, fault tolerant and buffered data flow between any kind of producers and consumers. Events of all kind (i.e. IoT or file systems) are meant to be transmitted securely via the network to enhance the security for end users through tamper-proof events that can be transmitted without any losses.	The mission is to deliver a MVP of a audit-proof recording of file system events to Grau Data. For this purpose there is a central event queue, which receives events from a producer and provides them a consumer, which is built on a underlying blockckain data structure. Finally, the event will be copied in a second consumer and will be deleted from the event queue.

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
	Total				92		95	95
1	**Initial Setup**				9		9	86
		Conductive administrative tasks and initiate project						
			Preparing Team Contract	2		2		
			Meeting with Industry Partner	1		1		
			Design Team Logo and T-Shirt	1		1		
			Organizing Release managers and discussing programming experience	1		1		
			Discussing communication strategy and preferred programming language	2		2		
			Create branch strategy in GitHub	2		2		
2	**Fundamentals**				26		28	58
		Programming prototypes of software components						
			Creating Software Architecture	3		5		
			Programm Consumer Prototype	5		5		
			Programm Producer Dummy in Java	3		3		
			Programm Producer Dummy in Python	3		3		
			Setup RabbitMQ as Event Queue	3		3		
			Research of suitable Blockchain technology	8		8		
			Provide Bill of Materials	1		1		
3	**Refinements**				31		32	26
		Evaluating message flow and refinement of software components						
			Facilitate to aggregate messages to one data package	3		5		
			Integrating Blockchain data structure in Consumer Dummy	5		5		
			Recovering of missing data in Producer Dummy	5		3		
			Evaluate secure data sources for Producer Dummy	2		3		
			Revising Blockchain	3		3		
			Check for message integrity	5		5		
			Creating message acknowledgement for Producer Dummy	8		8		
4	**Testing**							
		Creating Unit Tests for software components			11		11	15
			JUnit Test for Producer Dummy	5		5		
			JUnit Test for Consumer Dummy	3		3		
			JUnit Test for Blockchain	3		3		
5	**Documentation**							
		Creating documentation for software components			15		15	0
			Create documentation for repository	8		8		
			Create documentation for Producer Dummy	2		2		
			Create documentation for Consumer Dummy	2		2		
			Create documentation for Blockchain	3		3		

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down



#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
6	**Refactoring**							
		Refactoring Project code						
			Refactoring code for Producer Dummy					
			Refactoring code for Consumer Dummy					
			Refactoring code for Blockchain					
7	**Testing**							
		Testing Project Code						
			Improving Test Coverage Blockchain					
			Testing the whole System					

Term	Definition

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
1	Feature Code Review has been completed and passed	Feature Code Review has been completed and passed	Feature Code Review has been completed and passed
	a) Code is completly implemented	*a) Code is completly implemented*	*a) Code is completly implemented*
	b) Code is structured according to our coding guidelines and commented	*b) Code is structured according to our coding guidelines and commented*	*b) Code is structured according to our coding guidelines and commented*
	c) Code is checked into repository	*c) Code is checked into repository*	*c) Code is checked into repository*
	d) Documentation is updated	*d) Documentation is updated*	*d) Documentation is updated*
2	Cleanliness of Code	Cleanliness of Code	Cleanliness of Code
3	JUnit Tests have been written and passed (if required)	JUnit Tests have been written and passed (if required)	JUnit Tests have been written and passed (if required)
4	No critical bugs are open	All known bugs are fixed	All known bugs are fixed
5	Feature branch has been tagged and merged	Code has been inclued into the release (candidate)	Software prototype passes external review
6	Feature Code has been included into the release (candidate)		User documentation passes external review
7	Product Owners accept Feature		Developer documentation is available
8	Code Coverage:	Code Coverage:	Code Coverage:
	60% for Features	70% for Sprint Release	80% for Product Release

Type	Link / reference

\	Context	Name	Version	License	Comment
	Backend	Python	3.10.4	MIT	software is programmed in python
	Backend	Java	8.0 Update 333	MIT	second Producer Dummy
	Message Broker	RabbitMQ	3.10.0	MIT	implements Advanced Message Queuing Protocol

Last Name	First Name	Value					
Schmidt	**David**	2		**2.00**	**OK**		
Shanabhag	**Gajanana**	2					
Mazzini	**Francesco**	2					
Papadaki	**Anastasia**	2					
D'Ercoli	**Chiara**	2		0	No size		
Khalid	**Muhammad Ibrahim**	2		1	Trivial size		
				2	Small size		
				3	Medium size		
				5	Large size		
				8	Very large size		
				13	Too large (size)		