

<b>Project Name</b>	RTDIP Timeseries Forecasting
<b>Online team meeting</b>	<a href="https://fau.zoom-x.de/j/65502405196?pwd=8H73lyixQfqKrnO8Eb47kawnuKChGp.1">https://fau.zoom-x.de/j/65502405196?pwd=8H73lyixQfqKrnO8Eb47kawnuKChGp.1</a>
<b>Production system (if any)</b>	
<b>Test system (if any)</b>	
<b>GitHub repository</b>	<a href="https://github.com/amosproj/amos2025ws03-rtdip-timeseries-forecasting">https://github.com/amosproj/amos2025ws03-rtdip-timeseries-forecasting</a>
<b>GitHub feature board</b>	<a href="https://github.com/orgs/amosproj/projects/91/">https://github.com/orgs/amosproj/projects/91/</a>
<b>GitHub imp-squared backlog</b>	<a href="https://github.com/orgs/amosproj/projects/96">https://github.com/orgs/amosproj/projects/96</a>
<b>Team T-shirt (white)</b>	<a href="https://www.shirtinator.de/s/QWcDXTKGS72ISnrRNZq1hg">https://www.shirtinator.de/s/QWcDXTKGS72ISnrRNZq1hg</a>
<b>Team T-shirt (black)</b>	<a href="https://www.shirtinator.de/s/0yx3duRJSzW-5hP32KmZ1w">https://www.shirtinator.de/s/0yx3duRJSzW-5hP32KmZ1w</a>
<b>Additional materials</b>	<a href="https://discord.gg/KJXGmjcs">https://discord.gg/KJXGmjcs</a>
<b>Team mailing list</b>	oss-amos-projX@lists.fau.de
<b>Happines Index App</b>	<a href="https://happy-amos.appspot.com/Courses">https://happy-amos.appspot.com/Courses</a>



#	Meeting Day	Review	Planning	Software Developer	Release Manager	Scrum Master	Comment
1	2025-10-15	Patrick Meusling	Hafidz Arifin	Everyone else	N/A	Luca	
2	2025-10-22	Hafidz Arifin	Patrick Meusling	Everyone else	Christoph	Luca	
3	2025-10-29	Patrick Meusling	Hafidz Arifin	Everyone else	Mehdi Khabouze	Luca	
4	2025-11-05	Hafidz Arifin	Patrick Meusling	Everyone else	Rene Jokiel	Luca	
5	2025-11-12	Patrick Meusling	Hafidz Arifin	Everyone else	Abdul Haseeb	Luca	
6	2025-11-19	Hafidz Arifin	Patrick Meusling	Everyone else	Hannes Pohnke	Luca	
7	2025-11-26	Patrick Meusling	Hafidz Arifin	Everyone else	Simon Selbig	Luca	Mid-term due
8	2025-12-03	Hafidz Arifin	Patrick Meusling	Everyone else	Rene Jokiel	Luca	
9	2025-12-10	Patrick Meusling	Hafidz Arifin	Everyone else	Mehdi Khabouze	Luca	
10	2026-01-11	Hafidz Arifin	Patrick Meusling	Everyone else	Abdul Haseeb	Luca	
11	2026-01-18	Patrick Meusling	Hafidz Arifin	Everyone else	Hannes Pohnke	Luca	
12	2026-01-25	Hafidz Arifin	Patrick Meusling	Everyone else	Simon Selbig	Luca	
13	2026-02-01	Patrick Meusling	Hafidz Arifin	Everyone else	Mehdi Khabouze	Luca	
14	2026-02-08	Hafidz Arifin	Patrick Meusling	Everyone else	Christoph	Luca	Demo day!
15	2026-02-15	Patrick Meusling	Hafidz Arifin	Everyone else	Hannes Pohnke	Luca	Retrospective

<b>Goals</b>	Aquire new skills Produce a functioning and valuable product (a 1.0)
<b>Meeting norms</b>	We respect the opinions of others and we show up on time Summary of Partner Meeting required
<b>Working norms</b>	Clean, testable code with clear commit messages, code comments We respect other people's work
<b>Coordination norms</b>	Seperate branches for issues, second person review We balance workload among the team
<b>Communication norms</b>	Daily check the corresponding channels (discord) We communicate constructively, make sure to communicate possible problems (absence, etc.)
<b>Consideration norms</b>	We discuss issues openly We vote in case we can't reach a consensus
<b>Cont. improvement norms</b>	We encourage critique and improvement efforts Encourage reaching out for help (use the strength/skills of others)
<b>Rewards</b>	We treat ourselfes to a sweet of choice for good work
<b>Sanctions</b>	Push Ups (amount decided on the case)
<b>Signatures</b>	
Scrum Master	Luca Böhm
Product owner	Patrick Meusling
Product owner	Hafidz Arifin
Software developer	Abdul Haseeb
Software developer	Rene Jokiel
Software developer	Simon Selbig
Software developer	Hannes Pohnke
Software developer	Christoph Huy
Software developer	Mehdi Khabouze

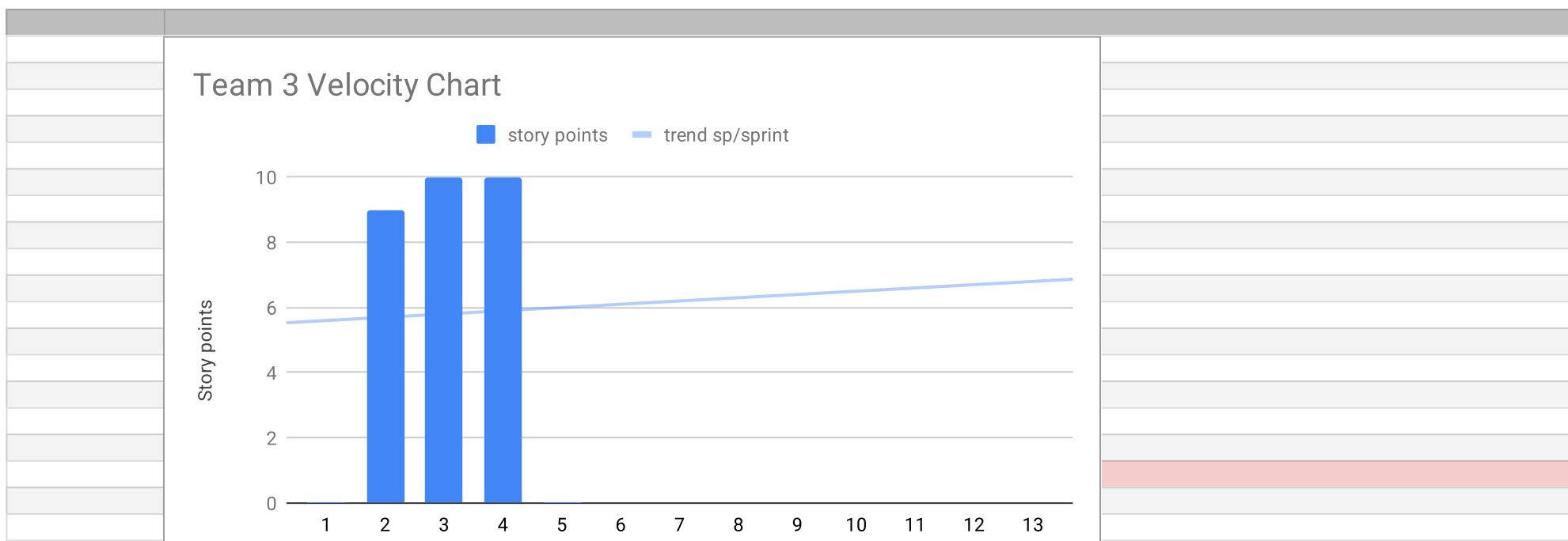
Product Vision	Project Mission
<p>To empower organizations with open-source, scalable, and transparent forecasting capabilities that enable data-driven decision-making across industries. By extending RTDIP with advanced forecasting and anomaly detection components as well as sector specific datasets, we envision a future where time series insights such as trends, seasonality, and predictive patterns are seamlessly integrated into real-time data pipelines. This will allow businesses to understand historical data, anticipate future behavior, optimize operations, and improve efficiency using accessible, community-driven, and production-ready forecasting tools.</p>	<p>To design, develop, and contribute open-source forecasting components for the RTDIP platform that enable trend analysis, anomaly detection, and predictive modeling on time series data. Our mission within this project is to research and implement forecasting techniques using Python and Apache Spark, validate and enrich them with real-world datasets, and ensure they meet RTDIP's modular, tested, and well-documented standards. By doing so, we will enhance RTDIP's functionality and provide the open-source community and industry users such as Shell with reliable, scalable forecasting capabilities that integrate seamlessly into existing data pipelines.</p>

Term	Definition
RTDIP (Real-Time Data Ingestion Platform)	An open-source project under the Linux Foundation designed to simplify the ingestion, transformation, and storage of data from various sources using scalable cloud technologies.
Forecasting	The process of predicting future values or trends based on historical time series data.
Time Series Data	Data collected over time at regular intervals, often used to analyze patterns such as trends and seasonality.
Apache Spark	An open-source distributed computing system that provides fast data processing for large-scale datasets.
PySpark	The Python API for Apache Spark, allowing developers to write Spark applications using Python.
Delta Lake	A storage layer that brings reliability, consistency, and performance to data lakes by supporting ACID transactions and schema enforcement.
Anomaly Detection	The identification of unusual patterns or outliers in data that do not conform to expected behavior.
Predictive Modeling	A statistical or machine learning approach that uses historical data to predict future outcomes.
Time Series Decomposition	The process of breaking down a time series into its core components such as trend, seasonality, and residuals.
ETL (Extract, Transform, Load)	A data pipeline process that extracts data from sources, transforms it into a suitable format, and loads it into a storage system.
Open Source	Software that is freely available to use, modify, and distribute, typically developed collaboratively by a community.
Linux Foundation	A nonprofit organization that supports open-source software projects and fosters collaboration across industries.
EasyCLA (Easy Contributor License Agreement)	A system used by the Linux Foundation to manage contributor license agreements, ensuring legal compliance for open-source contributions.
Modular Architecture	A software design principle that divides a system into separate, interchangeable components that can be developed and maintained independently.
Unit Testing	The practice of testing individual units or components of software to ensure they work as intended.
Documentation	Written descriptions and guides that explain the design, usage, and functionality of a software system.
Data Pipeline	A set of processes that move, transform, and store data from one system to another in a structured and automated way.
Scalability	The ability of a system to handle increasing amounts of work or data by adding resources.
Contribution Guidelines	A set of rules and best practices that contributors must follow to ensure consistency and quality in open-source projects.

Sprint #	Sprint goal
1	None
2	None
3	None
4	Optional
5	Establish a working foundation for the forecasting pipeline
6	Integrate the pipeline into the existing software
7	Advance the pipeline
8	Add more Datasets to the pipeline
9	Testing and Analysing the new Datasets
10	Adding new and different models to the pipeline
11	Testing and Analysing the new Models
12	Smaller features for more/different preprocessing etc.
13	Finishing the product so it's ready
14	Merging the product with the partner, coordinating the merge (they said this might take a bit of time)
15	Backup Week

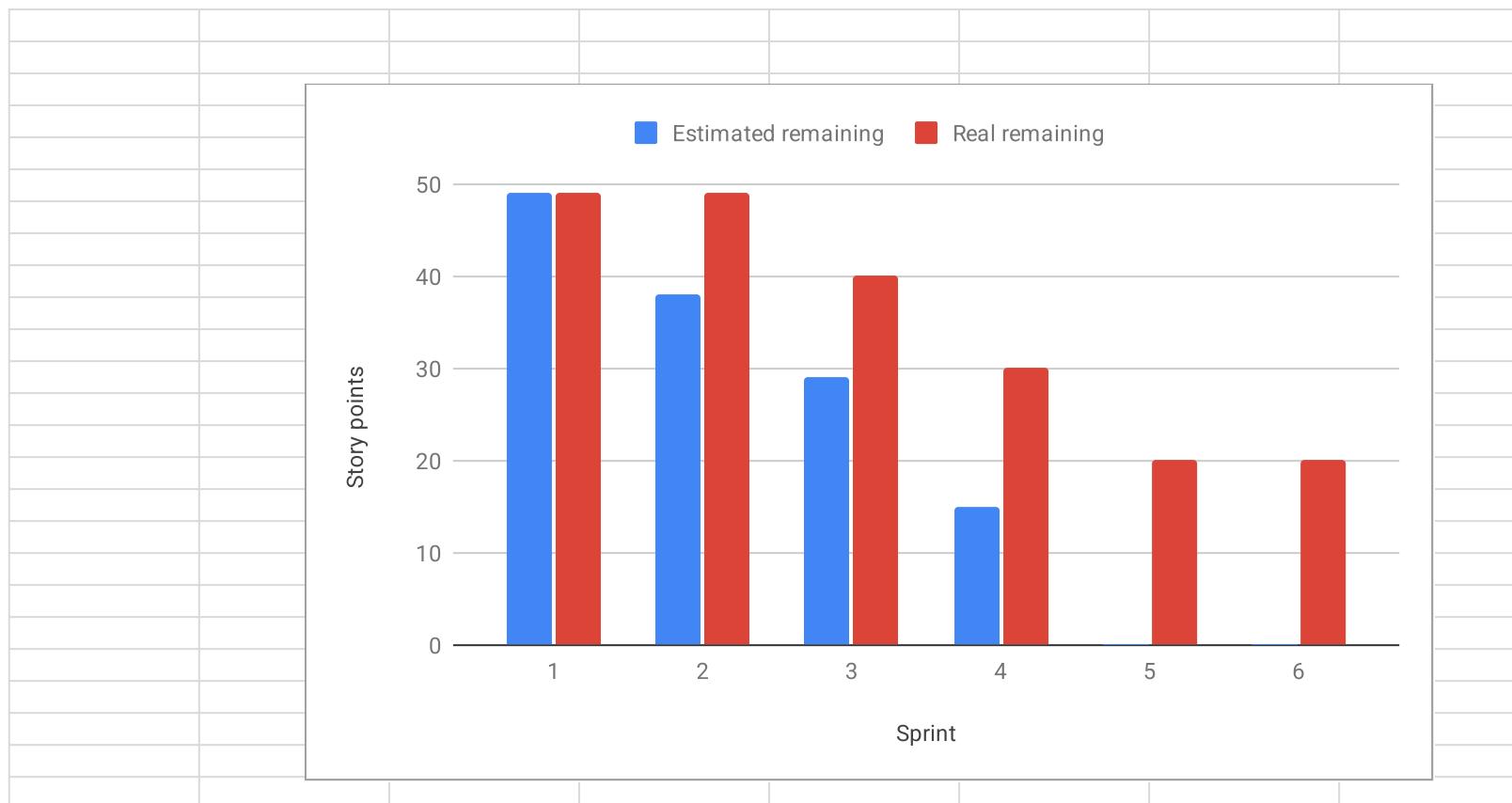
Last Name	First Name	Value	#DIV/0!	#DIV/0!
Böhm	Luca			
Meusling	Patrick			
Arifin	Hafidz			
Huy	Christoph			
Selbig	Simon		0	No size
Haseeb	Abdul		1	Trivial size
Pohnke	Hannes		2	Small size
Jokiel	Rene		3	Medium size
Khabouze	Mehdi		5	Large size
			8	Very large size
			13	Too large (size)
<b>How to play planning poker</b>				
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				

Sprint #	Story Points Realized
1	0
2	9
3	10
4	10
5	0
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
<b>PLEASE CREATE THE VELOCITY CHART ON A NEW TAB USING THE DATA FROM THIS TAB</b>	



Sprint	Goal	Feature Name	Est. size	Est. remaining	Real size	Real remaining
	<b>Release</b>					
	<b>Total</b>		49	49		
	<b>Sprints</b>					
1	None		0	49	0	49
2	None		11	38	9	49
3	None		9	29	10	40
4	Optional		14	15	10	30
5	Establish a working foundation for the forecasting pipeline		15	0	0	20
6	Integrate the pipeline into the existing software		0	0	0	20
	<b>Features</b>					
1	<b>None</b>					
2	<b>None</b>	Run the software on your own machine	1		1	
		Understanding the rtdip software	2		2	
		Software Architecture	2		2	
		Find open source time series datasets	2		2	
		Initialize software bill of materials	2		2	
		Research about time series forecasting fundamentals	2		NotCompleted	
3	<b>None</b>	Research Possible Models Useful for Shell Dataset	2		2	
		Update our Forked Repository with Latest RTDIP Release	1		1	
		Prepare and Demo Project Build Process	1		2	
		Perform Data Exploration on Shell Dataset	3		3	
		Research about time series forecasting fundamentals	2		2	
4	<b>Optional</b>	Getting to Know HPC of AMOS	2		NotCompleted	
		Perform Data Exploration on Opensource Dataset 1	2		3	
		Preprocess Shell Dataset	3		3	
		Implementing a basic Model pipeline	2		2	
		Research Possible Models Useful fo Opensource Dataset	2		NotCompleted	
		CI/CD Pipeline fail bug	3		2	
5	<b>Establish a working foundation for the forecasting pipeline</b>	Choosing Models, Datasets, Error Measure Metrics	2			
		Create Build Process Video	2			
		Model Training and Evaluation with Shell Dataset	3			
		Visualization of Results	2			
		Preprocess Opensource Dataset 1	3			
		PO Homework	3			

Sprint	Goal	Feature Name	Est. size	Est. remaining	Real size	Real remaining
6	<b>Integrate the pipeline into the existing software</b>	Model Deployment Pipeline into RTDIP				
		User/Design/Build Documentation				
		Repository Structure Refactor				
		Update Software Architecture				
		Set Up Unified Development Environment				
		Collect and Document Customer Feedback on Project Progress				
		Model Optimization and Improvement				
		Model Training and Evaluation with Opensource Dataset 1				
		<b>PLEASE CREATE THE BURNDOWN CHART ON A NEW TAB USING THE DATA FROM THIS TAB</b>				



Sprint	Goal	Feature Name	Est. size	Est. remaining	Real size	Real remaining
<b>Release</b>						
<b>Total</b>			0	0		
<b>Sprints</b>						
1			0	0	0	0
2			0	0	0	0
3			0	0	0	0
...				0		0
<b>Features</b>						
1						
2						
3						
		<b>PLEASE CREATE THE BURNDOWN CHART ON A NEW TAB USING THE DATA FROM THIS TAB</b>				

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
	Acceptance criteria are met.		
	Work are pushed to the Github repository.		
	Create a branch for each backlog items (coding)		
	A pull request is created for each related branch.		
	The work products in the pull requests are reviewed.		
	The corresponding branches are merged and closed.		
	The bill of materials section of the planning documents is updated.		
	Work needs to be documented in the corresponding wiki section		
	For new features unit test have to be written.		
	Update our forked repository with the latest RTDIP release.		
	If the task involves coding, the implementation is integrated into the RTDIP framework and verified to function correctly within it.		
		Release candidates with a working and meaningful update to the previous sprint is tagged.	
		Previously established features must continue to work.	
			The RTDIP forecasting components can be successfully built and deployed within the RTDIP environment.
			All automated tests and validation checks for the developed components pass without errors.
			The implemented forecasting features have been tested end-to-end and verified through a basic user workflow.
			Developer documentation is complete, clearly describing the architecture, setup, and contribution process for RTDIP.
			User documentation (usage examples, configuration steps, expected outputs) is finalized and up to date.
			The final release has been reviewed and approved by all project team members and the industry partner (Shell).



Tools	Name	Category	Version	Description
git	GitHub - GitHub	Code	2.43.0	Open source version control system used for tracking changes in source code over time.
VSTS Code	Visual Studio Code - Microsoft	Development	1.65.1	Integrated development environment (IDE) for writing, editing, and debugging code.
Confluence	Atlassian Confluence - Atlassian	Development / Build	255.1	Managing reusable builds and dependencies.