Project Name	Financial Product Portfolio Quick-Check
Production system (if any)	...
Test system (if any)	...
GitHub repository	https://github.com/amosproj/amos2021ws05-fin-prod-port-quick-check
GitHub kanban board (project)	https://github.com/amosproj/amos2021ws05-fin-prod-port-quick-check/projects/1
Team T-shirt (white)	https://www.shirtinator.de/loadBasket/TVvtvi-d3Fa
Team T-shirt (black)	https://www.shirtinator.de/loadBasket/TVvtvi-d3Fa
Additional materials	...
Happinessindex	https://happy-amos.appspot.com/Project?project=6194017020674048&course=6014071715397632
Team T-shirt (white)	Ivan Butron Sossa
Team T-shirt (black)	Maximilian Bartels, Ines Vogel, Lily Hügerich, Alexander Werner, Maximilian Rodiek, Tobias Bernhard, Andreas Ronellenfitsch

Last Name	First Name	GitHub User Name	Email Address
Bartels	Maximilian	Batogami	m.bartels@campus.tu-berlin.de
Bernhard	Tobias	tobbber	to-tb@gmx.de.
~~Butron Sossa~~	~~Ivan Antonio~~	~~ibutrons~~	~~ivanbutronsossa@gmail.com~~
Hügerich	Lily	lilyhuegerich	lilyhueg@gmail.com
Rodiek	Maximilian	m4xrdk	rodiek@campus.tu-berlin.de
Ronellenfitsch	Andreas	aron4	a.ronellenfitsch@campus.tu-berlin.de
Vogel	Ines	InesVogel	ine.vogel@gmail.com
Werner	Alexander	AlexanderW1996	werner.3@campus.tu-berlin.de
Groth	Patrick		patrick.groth@fau.de

Goals	Be Helpful & Respectful
	We and our customers are satisfied with the project results
	Not only satisfied with the product, also with the moral of and atmosphere in the team
	Improve efficiency over the semester
Meeting norms	Team Meetings are mandatory (Thursday, 12:30pm)
	Be on time
	Turn your camera on if possible
	Come prepared
Working norms	Primate of consensus, else voting, majority wins
	Support your team
	Task assignments will be made clear and agreed upon by the end of weekly group meetings.
	Task can solved independently or in groups (pair programming)
	Give constructive criticism, dont make it personal
Coordination norms	Team meetings will be lead by the POs
	Assignments can be chosen based on preference if assignments are not voluntarily chosen the Product owner can assign tasks based on workloads
Communication norms	Our communication channel will be slack and we write respectfully
	Communicate over the corresponding channels
	Response at least in 24h
	Whatsapp as urgent channel
	Any discussion should be viewable to anyone that the discussion pertains.
	If you cannot participate or arrive late, write a message in slack beforehand if possible at least an hour before
	Ask for help if necessary
Consideration norms	Side conversations in separate slack channel
	Don't interrupt each other in the zoom calls
	Be considerate of others people time and communicate at your earliest convience
	If two people have a conflict choose mediator, for more people or fail group discussion
Cont. improvement norms	Kanban board, closed tickets and weekly deadlines are used to track progress
	Test and code metrics
	Assign default reviewer
	Master should always be runnable

	Anyone can start a discussion about outcomes or notices from their assignments.
Rewards	Cake
	Drinks
Sanctions	Write Documentation
	Last place for task assignment
	Sanctions only by vote

#	Meeting Day	Comment	Coach	Product Owner	Software Developer	Release Manager	Scrum Master
1	2021-10-21		Yes	Max R & Andreas	Everyone else	N/A	Coach
2	2021-10-28		Yes	Max R & Andreas	Everyone else	N/A	Coach
3	2021-11-04		Yes	Max R & Andreas	Everyone else	Ines Vogel	Coach
4	2021-11-11		Yes	Max R & Andreas	Everyone else	Ines Vogel	Coach
5	2021-11-18		Yes	Max R & Andreas	Everyone else	Ines Vogel	Coach
6	2021-11-25		Yes	Max R & Andreas	Everyone else	Ines Vogel	Coach
7	2021-12-02	Mid-project release	Yes	Max R & Andreas	Everyone else		Coach
8	2021-12-09			Max R & Max B	Everyone else		tbd
9	2021-12-16			Max R & Max B	Everyone else		tbd
10	2022-01-13		Yes	Max R & Max B	Everyone else		tbd
11	2022-01-20			Max R & Max B	Everyone else		tbd
12	2022-01-27			Max R & Max B	Everyone else		tbd
13	2022-02-03		Yes	Max R & Max B	Everyone else		tbd
14	2022-02-10	Demo day / final release		Max R & Max B	Everyone else		tbd
15	2022-02-17	Project retrospective due		Max R & Max B	Everyone else		tbd

Product Vision	Project Mission
With growing opportunities of derivating and customizing financial products the product range of financial institutions is emerging rapidly. Hence, the difficulty of good consulting is rising equally while high complexity products drive the largest costs which are outgrowing benefits and revenue of conventional consulting. The Financial Product Portfolio Quick-Check simplifies financial consulting by providing an information-rich project-related overview of financial products. This overview helps banking and other types of financial institutions to understand the risks and opportunities of different financial products, while it tries to maximize profits by identifying overly complex products, which do not contribute to the business volume. Additionally, the product provides useful visualisations annotated with source documents to ensure an increase of a consultants efficiency in assessing the benefit of a product. Published under the MIT license, the aim is to make it accessible for our business partner as well as for any other organization and enable an enormous amount of other use cases.	Our project mission is to provide a web application which allows structured data entry for financial products, comparison and evaluation among products and product variants as well as with source documents annotated visualization. The core feature of the software is a calculation of benchmarks regarding complexity combined with annotated result charts. To meet the objectives mentioned above, we will maintain a helpful and friendly team atmosphere.

Term	Definition
Projekt	Ein Projekt hat ein oder mehrere Produkte; Jedes Projekt hat 1-n Produktbereiche
Produkt	Jedes Produkt hat ein oder mehrere Produktvarianten; Produkte werden in Produktbereiche unterteilt; *Beispiel: Dispokredit*
Produktvariante	Ein oder mehrere Produktvarianten bilden das Produkt; *Beispiele: Sofortdispo, Dispo Fix & Dispo Variabel*
Produktbereiche	Jeder Produktbereich hat zwei Unterkategorien (Privatkundenbereich oder Firmenkundenbereich); Jeder Produktbereich hat 1 bis n Produkte; *Beispiele: Kunde (Retail & Corporate), Kredit (Firmenkunde & Privat), Payments (Firmenkunde & Privat), Wealth Management und Investmentbanking;*
Produktschlüssel	Die ProduktID
Kundenschlüssel	Die KundenID
Beurteilungskriterium	
Komplexitätskriterium/-treiber	Werden zur Unterscheidung von Produktbereichen (z.B. Kunden) eingesetzt
Piechart (äußerer Ring)	Zur Unterscheidung von Kundensegmenten *(Beispiel ApoBank: 20% angestellte Heilberufler; 70% selbständige Heilberufler; 10% Studenten)*
Piechart (innerer Ring)	
Piechart (Größe)	Bezieht sich auf Produktvariante, also: "wie groß ist der Kreis" und "wie viel wird als hellblau, dunkelblau und grün bewertet"
Produktbereich Kunde	Produkte im Produktbereich Kunde wären: 1) Standardsegmente 2) Zusätzliche a) Retailkunde mit bis 50k, b) Mittelschichtkunde 50k-150k, c) Wealthmanagementkunden <150k, d) High Net Worth Individuals <500k; Zudem kann unterschieden werden zwischen Inlander/Ausländer oder Ehepaare/Alleinständig
Kundensegmente	*Beispiele: 1) Standardsegmente 2) Zusätzliche a) Retailkunde mit bis 50k, b) Mittelschichtkunde 50k-150k, c) Wealthmanagementkunden <150k, d) High Net Worth Individuals <500k; Zudem kann unterschieden werden zwischen Inlander/Ausländer oder Ehepaare/Alleinständig*
User/Usermanagement/Berechtigungen	Erster "Bereich": Login, Userrechte, Passwort vergessen
Verschachtelter Fragebogen	Zweiter "Bereich": Fragt Produktbereich, Produkt, Produktvarianten und wirtschafliche Bewertung sowie Komplexitätsbewertung ab
Anmerkungs/-Dokumentzuordnungsbereich	Dritter "Bereich": Zu jeder einzelnen Kategorie können Anmerkungen & Dokumentzuordnungen (z.B. Quelle: Geschäftsbericht 2017; Interview 19.08...) hinzugefügt werden
Bewertungen	Im Kern: Hoch, Mittel & gering (als Farben: rot, gelb & grün bzw hellblau, dunkelblau, grün); Aber auch Prozentberechnungen; Die Bewertung erfolgt durch einen Consultant, denn nur er weiß z.B. Mittelständler ist gelb, GmbH ist gelb, oder Verein ist rot; Die Bewertungen (=Fragen) sind für jedes Produkt einer Kategorie gleich, wobei es sein kann das bestimmte Fragen nicht beantwortet werden können

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
			TOTAL					
1	Initial Setup	Getting to know & project setup			10		10	83
			Additional team meeting	2		2		
			#3 Team Logo	1		2		
			#1 T-Shirt Logo	3		2		
			#2 Team Contract	2		1		
			Meeting with industry partner	2		3		
2	Initial Setups & Research	Understand the project & Implement first requirements from industry partner (Docker)			24		18	73
			#12 Define Initial REST-API	5		3		
			#13 Create Initial Containerized Architecture	5		5		
			#17 Define Minimal Requirements (User Stories)	5		5		
			#15 Upload Research for WebUI	3		1		
			#14 Upload Research for Database Types	3		1		
			#11 Conduct Research for User Management	3		3		
3	Frontend and Backend Functionalities & Architecture	Backend Architecture & first WebUI is implemented			18		12	55
			#36 Refactor project API	2		1		
			#34 Setup Backend REST API	5		3		
			#30 Set Up Relational Database	3		2		
			#32 Set Up Initial WebUI	5		3		
			#41 Use Case Diagram	1		1		
			#42 Flowchart Diagram	1		1		
			#40 ER-Diagram	1		1		
4	First Frontend Technical Breakthrough	Clear understanding of Project Mission, Product Vision & Definition of done & working on frontend			9		9	43
			#37 Product Vision	1		1		
			#38 Project Mission	1		1		
			#59 Definitons of Done	1		2		
			#44 Update Glossary	1		2		
			#46 Create View "Project Overview"	5		3		
5	Main Frontend and Backend Functionalitites	Getting a breakthrough with the main funcionalitites of the software			24		18	34
			#45 Create View "Manage Project"	5		3		
			#47 Create View "Manage Project Members"	5		5		
			#48 Create "Product Overview"	5		5		
			#61 Create "Enter Financial Product Data"	3		1		
			#63 Create "Add Product View"	3		1		
			#64 Create "Enter Complexity Data"	3		3		
6	Mid Term Release & Testing	Getting the last frontend functionalities for a first breakthrough of the software and continue testing the backend			21		16	16
			#65 Create View "Complexity Evaluation"	5		3		
			#62 Create View "Economic Evaluation"	5		5		
			#79 Testing of "Controller" Folder (Backend)	3		2		

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
			#81 Testing of Data Transfer Object ("dto") Folder (Backend)	3		2		
			#83 Testing of "Service" Folder (Backend)	3		3		
			#82 Testing of "Exceptions" Folder (Backend)	2		1		
	Release	**Mid Term**						
	No Sprints	6	.					
	Due Date	02.12.2021						

Sprint	Sprint Theme	User Stories	Est. Size	Est. Burndown	Real Size	Real Burndown	
0				106		83	
1	Initial Setup	3,2,1	10	96	10	73	
2	More Initial Setups & Research	12, 13, 17, 14, 15, 11	24	72	18	55	
3	Frontend and Backend Functionalities & Architecture	36, 34, 30, 32, 41, 42, 40	18	54	12	43	
4	First Frontend Technical Breakthrough	37, 38, 59, 44, 46	9	45	9	34	
5	Main Frontend and Backend Functionalitites	45, 47, 48, 61, 63, 64	24	21	18	16	
6	Mid Term Release & Testing	65, 62, 79, 81, 83, 82	21	0	16	0	
Total			**106**		**83**		

Burn-down Chart

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
			TOTAL					
1	Initial Setup	Getting to know & project setup			10		10	83
			Additional team meeting	2		2		
			#3 Team Logo	1		2		
			#1 T-Shirt Logo	3		2		
			#2 Team Contract	2		1		
			Meeting with industry partner	2		3		
2	Initial Setups & Research	Understand the project & Implement first requirements from industry partner (Docker)			24		18	73
			#12 Define Initial REST-API	5		3		
			#13 Create Initial Containerized Architecture	5		5		
			#17 Define Minimal Requirements (User Stories)	5		5		
			#15 Upload Research for WebUI	3		1		
			#14 Upload Research for Database Types	3		1		
			#11 Conduct Research for User Management	3		3		
3	Frontend and Backend Functionalities & Architecture	Backend Architecture & first WebUI is implemented			18		12	55
			#36 Refactor project API	2		1		
			#34 Setup Backend REST API	5		3		
			#30 Set Up Relational Database	3		2		
			#32 Set Up Initial WebUI	5		3		
			#41 Use Case Diagram	1		1		
			#42 Flowchart Diagram	1		1		
			#40 ER-Diagram	1		1		
4	First Frontend Technical Breakthrough	Clear understanding of Project Mission, Product Vision & Definition of done & working on frontend			9		9	43
			#37 Product Vision	1		1		
			#38 Project Mission	1		1		
			#59 Definitons of Done	1		2		
			#44 Update Glossary	1		2		
			#46 Create View "Project Overview"	5		3		
5	Main Frontend and Backend Functionalitites	Getting a breakthrough with the main funcionalitites of the software			24		18	34
			#45 Create View "Manage Project"	5		3		
			#47 Create View "Manage Project Members"	5		5		
			#48 Create "Product Overview"	5		5		
			#61 Create "Enter Financial Product Data"	3		1		
			#63 Create "Add Product View"	3		1		
			#64 Create "Enter Complexity Data"	3		3		
6	Mid Term Release & Testing	Getting the last frontend functionalities for a first breakthrough of the software and continue testing the backend			21		16	16
			#65 Create View "Complexity Evaluation"	5		3		
			#62 Create View "Economic Evaluation"	5		5		
			#79 Testing of "Controller" Folder (Backend)	3		2		
			#81 Testing of Data Transfer Object ("dto") Folder (Backend)	3		2		
			#83 Testing of the first half of "Service" Folder (Backend)	3		3		
			#82 Testing of "Exceptions" Folder (Backend)	2		1		
7	Refactoring, Tests & Refinement	Code Cleanup for Developers & do leftovers			24			
			#95 Testing of the second half of the "Service" Folder (Backend)	3				
			#97 Testing of the second half of the "Controller" Folder (Backend)	5				
			#101 Define custom UI theme	3				
			#84 Refactor "Score", "ProductAreaController", & "RatingController" functionalities dynamically	5				
			#99 Connect Product Overview View with Backend	3				
			#98 Connect Manage Project View with Backend	2				

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
			#102 Basic layout for Product Rating View	3				
8	Product Ratings	Implement the possibility to rate the products with scores			32			
			#103 Fill Product Rating View with Mock Data	3				
			#31 Initialize Document Database	5				
			#104 Implement Bearing Point Theme	5				
			#105 Write code documentation for backend	2				
			#107 Write code documentation for frontend	2				
			#110 Catalogue for Scores (Backend)	5				
			#106 Make the Basic Layout of the Product Rating View to a Functional Site	3				
			#109 Make the Functional Site of the Product Rating View Dynamic and Connect it to Backend	5				
			#100 Add Functionality for Product Overview to attach files	2				
9	Evaluation of results	Implement basis for visualisation of results and defining first user roles for later security mechanisms			26			
			#114 Convert evaluation to file	5				
			#108 Evaluate project in backend	3				
			#111 Basic layout for Evaluation View	8				
			#112 Make the Basic Layout of the Evaluation View to a Functional Site	5				
			#112 Make the Functional Site of the Evaluation View Dynamic and Connect it to Backend	5				
10	Visualisation of Results and begin User Management Implementation	Start with implementation of the core of the software, the visualisation of results in different graphs			27			
			#116 Make the Basic Layout of the Result View to a Functional Site	8				
			#117 Restricit access to services to a user account	8				
			#88 User Login View	3				
			#66 Basic Layout for Result View	3				
			#115 Fill the Basic Layout of the Result View with Mock Data	5				
11	Further User Management	Specify & define security mechanisms more and give permissions to different user			8			
			#118 Add multiple user to system	5				
			#90 View Forgot Password	3				
12	Enhance Security and Finish Login	Finish Security Mechnisms, be able to Login into the application & give permissions to different user roles			18			
			#120 Add account roles to system	3				
			#119 Make the Functional Site of the Result View Dynamic and Connect it to Backend	5				
			#121 Make the Result View Pretty	5				
			#125 Reach DoD Test Coverage	5				
13	Fine Tuning & Preparation for Demo Day	Improvement of system features & doing leftovers from last sprints			12			
			#124 Prepare Demo Day Video	5				
			#126 Prepare Demo Day Slides	5				
			#123 Write code documentation for frontend (final code)	2				
14	Final Release (no real sprint)	Refinement of the final product			2			
			#128 Finish documentation and wiki	1				
			#127 GitHub Clean Up	1				

Release	Mid Term
No Sprints	6
Due Date	02.12.2021

Sprint	Sprint Theme	User Stories	Est. Size	Est. Burndown	Real Size	Real Burndown
0				106		83
1	Initial Setup	3,2,1	10	96	10	73
2	More Initial Setups & Research	12, 13, 17, 14, 15, 11	24	72	18	55
3	Frontend and Backend Functionalities & Architecture	36, 34, 30, 32, 41, 42, 40	18	54	12	43
4	First Frontend Technical Breakthrough	37, 38, 59, 44, 46	9	45	9	34

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
5		Main Frontend and Backend Functionalitites	45, 47, 48, 61, 63, 64	24	21	18	16	
6		Mid Term Release & Testing	65, 62, 79, 81, 83, 82	21	0	16	0	
	Total			106		83		

Release	Final Project
No Sprints	8
Due Date	10.02.2022

Sprint	Sprint Theme	User Stories	Est. Size	Est. Burndown	Real Size	Real Burndown
7	Refactoring, Tests & Refinement					
8	Product Ratings					
9	Evaluation of results					
10	Visualisation of Results begin User Management Implementation					
11	Further User Management					
12	Enhance Security and Finish Login					
13	Fine Tuning & Preparation for Demo Day					
14	Final Release (no real sprint)					
	Total					

Sprint	Status	Source	Impediment	Resolution
1	Resolved	Maximilian R	Not all members could use the happiness tool	All members posted a screenshot of the happiness tool to Slackto verify the properly used the hapiness tool
1	Resolved	Maximilian R	Late appointment with the industry partner (one day before sprint meeting because of their vacation)	Early communication because they are not on vacation anymore
1	Resolved	Andreas Ronellenfitsch	Chaotic settling-in phase	Defined a working process
1	Resolved	Andreas Ronellenfitsch	Use of designated Slack servers	Definied a Slack server for each topic
2	Resolved	Maximilian R	The feature creation process was a little unstructured	Feature creation process at least 3 days before group meeting
3	Resolved	Maximilian R	External requirements regarding the exact definition of "product" & "product area" was unclear and blocked some tasks	Meeting with Theo on 10.11.2021 solved those terminologies (Glossary)
4	Resolved	Tobi	Code review proces is currently unclear	Was resolved by scrum implicitly
4	Resolved	Ines	Clarification needed on product area definition	See answer from Andreas in slack
4	In-work	Maximilian R	1 Software Developer (Frontend) dropped out	Splitting the team in fixed frontend and backend teams in the next meeting (18.11)
6	In-work	Patrick G.	Tickets could not be finished	- Ivan dropped out -> POs support with smaller development tasks - unforseen complexity as well as sickness of a teammembers - watch for one more week and maybe split tickets into frontend/backend if it's not resolved
6	In-work	Team	Complexity of frontend tickets still uncertain	- Split frontend tickets further (functional -> fancy|different components) - Max is going to support front-end from now on
			Communication between front- and backend	- additional (optional) problem specific meetings to solve issues on monday

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
	Approved by product owner	Approved by product owner	Approved by product owner
	Tests have been written (e.g. Unit test, integration test etc..)	100% tests passed	No critical bugs
	Code has been peer reviewed and approved	No critical bugs	User documentation is written
	No syntax or runtime errors emerged	Merge conflicts solved	75% test coverage
	Code has to be included in the release candidate	Demo approved by team	Demo approved by team
	minimum 85% test coverage	Code is deployable using docker	Code is deployable using docker
		50%-60% test coverage	Approved by business partner
		Code in a clean state	Code documentation is written
			Build & deployment documentation is written
			Documentation has been peer reviewed

Type	Link / reference
Wiki	https://github.com/amosproj/amos2021ws05-fin-prod-port-quick-check/wiki

\	Context	Name	Version	License	Comment
	REST API	Spring Boot	2.5.6	Apache License, Version 2.0	Used for REST API and microservice architecture
	Deployment	Github Actions		-	CI/CD Tool for running automated tests, building docker images and deployment.
	Deployment	Docker Engine	v20.10	Apache License 2.0	Docker engine is used to run applications in a container
	Deployment	Docker CLI	v20.10	Apache License 2.0	Tool to build containerized applications as docker images.
	Database	HyperSQL	2.6.2	BSD License	Deployment of database applications
	Frontend framework	React	17.0.1	MIT License	UI Framework and handler for the view layer of web and mobile apps

Last Name	First Name	Value						
Bartels	Maximilian	5						
Bernhard	Tobias	5		**5.00**	**OK**			
Butron Sossa	Ivan Antonio	5						
Hügerich	Lily	5						
Rodiek	Maximilian	5		0	No size			
Ronellenfitsch	Andreas	5		1	Trivial size			
Vogel	Ines	5		2	Small size			
Werner	Alexander	5		3	Medium size			
				5	Large size			
				8	Very large size			
				13	Too large (size)			