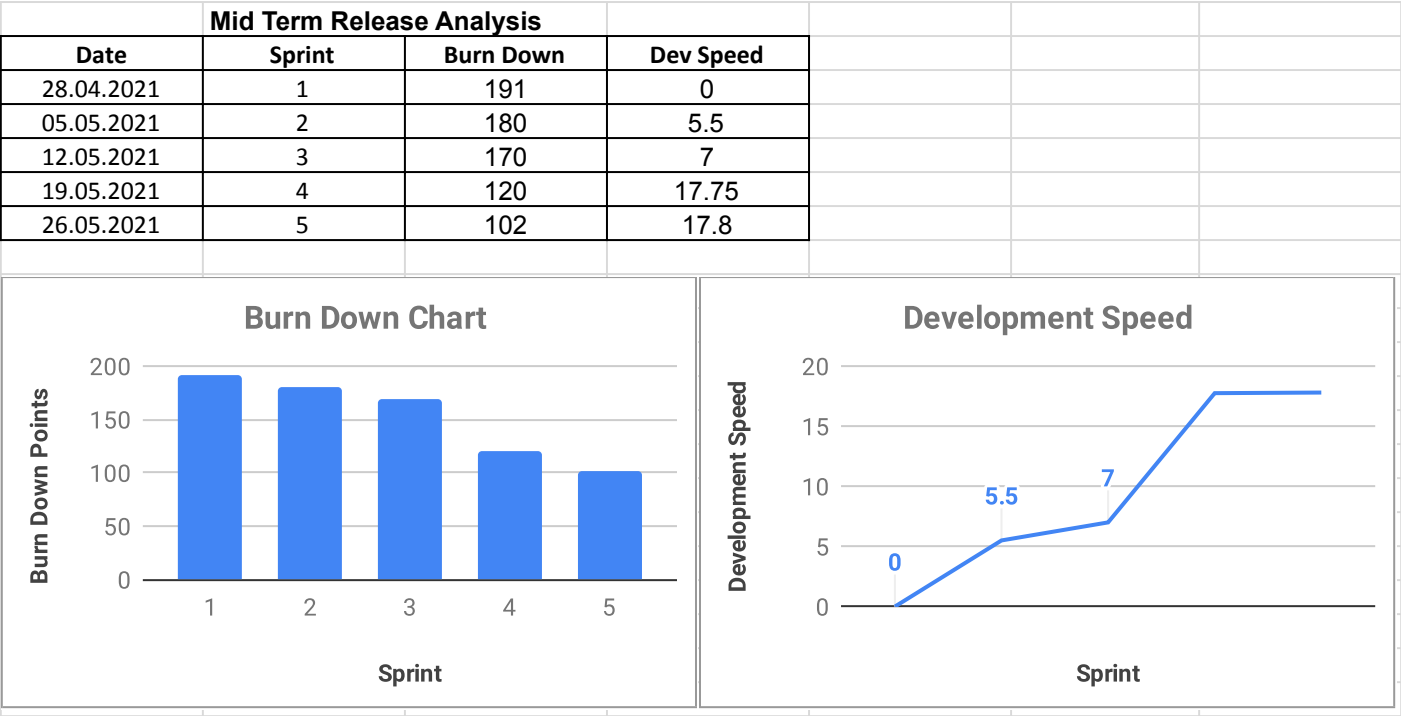| Project Name	ACIDIC - Autonomous Continuous Inner-source Development and Integrity Checker
Production system (if any)	Docker: https://hub.docker.com/u/amoslinter
Test system (if any)	Local/Docker: https://hub.docker.com/u/amoslinter
GitHub repository	https://github.com/amosproj/amos-ss2021-is-project-linter
GitHub kanban board (project)	https://github.com/amosproj/amos-ss2021-is-project-linter/projects/1
Team T-shirt (white)	Yes
Team T-shirt (black)	No
Additional materials	T-Shirt Link: https://www.shirtinator.de/loadBasket/GZ7K_ZV95U2 (already ordered)

Last Name	First Name	GitHub User Name	Email Address
Harutyunyan	Nick (coach)	nickharutyunyan	nikolay.harutyunyan@fau.de (for Zoom) / nikolay.harutyunyan@gmail.com (Discord)
Wohlwend	Christoph	WohlwendGitHub	c.wohlwend@gmx.de / chris.wohlwend@gmail.com / christoph.cw.wohlwend@studium.fau.de
Groth	Patrick	patrickgroth	patrick.groth@logatik.de / patrick.groth@fau.de
Böhm	Lukas	lukasbm	boehm-lukas@outlook.de / lukas.l.boehm@gmail.com / lukas.l.boehm@fau.de
Pfister	Jonas	zokto	jonas.pfister@fau.de
Karri	Veera Venkata Satyanarayana	karri2909	veera.karri@fau.de
Neuerer	Christian	Auron2402	christian.neuerer@fau.de
Asani	Amir	AmirAsani	amir.asani@fau.de
Krieg	Nicolas	nico-628	nicolas.krieg@fau.de

Goals	Fulfilling all business requirements with few or no defects with final user acceptance criteria
Meeting norms	Regular frequent meetings. Mandatory for everyone to join in time. Wednesday 12:30 - 01:45 PM (Mandatory meeting).
Working norms	Every member has a defined responsibility. Every member can collaborate/help others as well.
Coordination norms	Product owners are Scrum masters initially. Software developers also act as scrum masters on rotational basis. Scrum master moderates the team meetings/allocates assignments/lead team meetings
Communication norms	Discord. Group communication. Team members are expected to respond promptly. Acknowledge messages with in one or two hours.
Consideration norms	Side conversations are allowed as required. Decisions are finalized based on team's overall agreeement (via. group voting etc)
Cont. improvement norms	Track the progress of tasks in scrum board. Team members are expected to update the status promptly with comments. Contructive feedback on each others coding style to maintain uniformity in the code.
Rewards	Completion of big tasks as per acceptance criteria.
Sanctions	Team member needs to do 30 push ups as a punishment on camera
Signatures	Lukas Böhm, Christian Neuerer, Christoph Wohlwend, Nicolas Krieg, Amir Asani, Jonas Pfister, Patrick Groth, Veera Karri

#	Meeting Day	Comment	Coach	Product Owner	Software Developer	Scrum Master	Release Manager
1	2021-04-14	Project Initiation	Yes	Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
2	2021-04-21		Yes	Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
3	2021-04-28			Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
4	2021-05-05		Yes	Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
5	2021-05-12			Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
6	2021-05-19		Yes	Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
7	2021-05-26	Mid-term due	Yes	Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
8	2021-06-02			Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
9	2021-06-09			Patrick Groth/Veera Karri	Everyone else	Veera Karri	Lukas Böhm
10	2021-06-16		Yes	Patrick Groth/Veera Karri	Everyone else	Veera Karri	Jonas Pfister
11	2021-06-23			Patrick Groth/Veera Karri	Everyone else		
12	2021-06-30			Patrick Groth/Veera Karri	Everyone else		
13	2021-07-07		Yes	Patrick Groth/Veera Karri	Everyone else		
14	2021-07-14	Demo day!		Patrick Groth/Veera Karri	Everyone else		
15	2021-07-21	Retrospective		Patrick Groth/Veera Karri	Everyone else		

#	Theme	Feature Name	Goal	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
3	Backend	Query a gitlab repository	To get details of a repository and display in the dashboard	3	3	5	5	
5	Back end	REST API	To query the linting state of the projects	5	5	3	3	
6	Front End	Feature to enter link to repository in the front end screen	In order for backend to check the repository entered in front end screen	3	3	3	3	
7	Backend	Crawler	To check the repository from the front end for criteria being defined later	5	5	1	1	
11	Backend	JPA Objects	JPA objects to store linting results in a persistent data base	3	3	3	3	
12	Front end	Visibility Check	To check in the database if the visibility of the project is public and display the result	0	0	0	0	
13	Backend	Visibility Check	To check if the visibility of a project given by the frontend is set to "public" and store the result in the database	0	0	0	0	
14	Front end	Merge Request Activation Check	"Merge Requests" in settings should be activated/enabled "Everyone with access" should be allowed to use "Merge Requests"	3	3	3	3	
15	Backend	Merge Request Activation Check	"Merge Requests" in settings should be activated/enabled. "Everyone with access" should be allowed to use "Merge Requests"	3	3	2	2	
16	Front end	Forking Activation Check	"Forking" in settings should be activated/enabled "Everyone with access" should be allowed to use "Forking"	0	0	0	0	
17	Backend	Forking activation check	"Forking" in settings should be activated/enabled "Everyone with access" should be allowed to use "Forking"	2	2	2	2	
18	Front end	Issues Activation Check	"Issues" in settings should be activated/enabled "Everyone with access" should be allowed to use "Issues"	0	0	0	0	
19	Backend	Issues Activation Check	"Issues" in settings should be activated/enabled "Everyone with access" should be allowed to use "Issues"	2	2	1	1	
20	Front end	Yaml / XML configuration file	To edit a yaml/xml file to edit what checks should be displayed to the user.	2	2	2	2	
21	Backend	Mandatory Files Check	To check whether the files (README.md, CONTRIBUTING.md, MAINTAINERS.md) are included	0	0	1	1	
27	Database	Database Change	To store information of each test result in a object linked to lintingResult	2	2	2	2	
32	Front end	Show when a requested repository is not found	To see an error if my entered URL is wrong so that i can check if the entered URL is correct	1	1	1	1	
33	Backend	Visibility Check - Request access	To check the whether the "Request Access" provided by the front end is enabled or not and store the result in the database.	2	2	1	1	
35	Backend	Visibility Check	To check the whether the Gitlab wiki pages feature (provided by the front end) is enabled or not and store the result in the database	2	2	1	1	
36	Backend	Cleanliness - Project description - Backend	To check the whether the project description exists or not	2	2	1	1	
39	Backend	Cleanliness - Project Avatar - Back end	To check the whether the project avatar check provided by the front end is added or not	1	1	2	2	
40	Backend	Severity of Checks	To display the severity of the check	1	1	2	2	
42	Backend	JSON file update	To assign the severity level, description as well as the error message	2	2	3	3	
45	Backend	Check - Squash commits are not allowed	To ensure squash commits are not allowed	2	2	2	2	
46	Backend	Additional Check - README.md-	To check whether README.md file contains atleast one link at a specified URL area	3	3	3	3	
47	Backend	Additional Check - Badges	To check whether atleast one badge exist either in the README.md file or in the badge bar (at the top of the project)	2	2	1	1	
48	Backend	Additional Check - CONTRIBUTING.md-	to check whether CONTRIBUTING.md file contains link to other CONTRIBUTING.md file and display a warning message " CONTRIBUTING.md file does contain link to other CONTRIBUTING.md file" based on the check Acceptance criteria	2	2	3	3	

Mid - Term	No Sprints	6					
	Due Date	19.05.2021 - 26.05.2021					
Sprint Review	Sprint	Sprint Theme	User Stories	Est. Size	Real Size	Burn-Down	Dev Speed
21.04.2021	0	-	-			228	
28.04.2021	1	Project Preparation and Analysis	-	0	0	228	0
05.05.2021	2	Basic Functionality	6, 20, 15, 5, 7, 27	17	11	217	5.5
12.05.2021	3	Frontend and Backend	21, 12, 18, 16, 32, 14, 13, 33, 19, 17, 35	12	10	207	7
19.05.2021	4	Frontend and Backend	43, 39, 36, 45, 48, 3 , 46, 42, 40, 47, 11	55	50	157	17.75
26.05.2021	5	Mid Term Release	54, 59, 55, 60, 41, 57, 58, 26, 53, 49	18	24	133	19
	Total			102	95		
Final	No Sprints	7					
	Due Date	14.07.2021					
Sprint Review	Sprint	Sprint Theme	User Stories	Est. Size	Real Size	Burn-Down	
02-06-2021	6	Frontend and Backend	70, 74, 75, 73, 72, 8	16	15	118	18.33333333
09-06-2021	7	Frontend and Backend	79, 82, 80, 64, 66, 81, 71	0	0	118	15.71428571
16-06-2021	8			0	0	118	13.75
23-06-2021	9			0	0	118	12.22222222
30-06-2021	10			0	0	118	11
07-07-2021	11			0	0	118	10
14-07-2021	12			0	0	118	9.166666667
	Total			16	15		

Mid Term Release Analysis			
Date	Sprint	Burn Down	Dev Speed
28.04.2021	1	191	0
05.05.2021	2	180	5.5
12.05.2021	3	170	7
19.05.2021	4	120	17.75
26.05.2021	5	102	17.8

Burn Down Chart

Burn Down Points vs Sprint

Development Speed

Development Speed vs Sprint

Term	Definition
Crawler	The crawler extracts all repositories from a gitlab instance and sends them to the linter
Gitlab instance	A GitLab System, installed for a domain
Linter	The linter executes all checks for a repository and saves the results in the database
REST API	The rest api handles the communication between the frontend and the database. It also handles the communication between the frontend and the linter
Fehler	Hohes Schwierig
Warnung	Warning message
Info	Information message
Angular	a TypeScript-based open-source web application framework
Wireframes	A wireframe is a layout of a web page that demonstrates what interface elements will exist on key pages
YAML	a recursive acronym for "YAML Ain't Markup Language", is a human-readable data-serialization language. It is commonly used for configuration files and in applications where data is being stored or transmitted.
XML	It is an extensible markup language file and it is used to structure data for storage and transport
Spring (Boot)	It is an application framework and inversion of control container for the Java platform
Postgres SQL	It is an open-source relational database management system emphasizing extensibility and SQL compliance
Docker	Docker is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers
Docker-Compose	Docker Compose is a tool for running multi-container applications on Docker defined using the Compose file format
Compose file	It is used to define how the one or more containers that make up your application are configured.
GitLab4J™ API	It provides a full featured and easy to consume Java library for working with GitLab repositories via the GitLab REST API
OpenJDK	OpenJDK is an open-source implementation of the Java Platform, Standard Edition
VueJS	Vue.js is an open-source model–view–viewmodel front end JavaScript framework for building user interfaces and single-page applications
config file	All Checks and Settings are defined here
database	Contains only results of the linting process
README.MD	A README file contains information about other files in a directory or archive of computer software. A form of documentation, it is usually a simple plain text file called Read Me
CONTRIBUTING.MD	It is a software contribution guidelines text file which project managers include in free and open-source software packages or other open media packages for the purpose of describing how others may contribute user-generated content to the project.
MAINTAINERS.MD	It is a markdown file intended for maintainers with commit access to one or more repositories.
Test	Specific check that tests the characteristic of a Projekt
Projekte	Gitlab repository/project

Product Vision	Project Mission
Helping Open-/Inner source initiatives improve their work and raise awareness for great projects.	Developing an awesome app without hating each other afterwards. Having a clear overview over all the different inner source projects within Datev and improve collaboration within the company. The goal is also to give improvement suggestions to maintainers about the quality of their project.

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
3	Backend	Query a gitlab repository	To get details of a repository and display in the dashboard	3	3	5	5	
5	Back end	REST API	To query the linting state of the projects	5	5	3	3	
6	Front End	Feature to enter link to repository in the front end screen	In order for backend to check the repository entered in front end screen	3	3	3	3	
7	Backend	Crawler	To check the repository from the front end for criteria being defined later	5	5	1	1	
11	Backend	JPA Objects	JPA objects to store linting results in a persistent data base	3	3	3	3	
12	Front end	Visibility Check	To check in the database if the visibility of the project is public and display the result	0	0	0	0	
13	Backend	Visibility Check	To check if the visibility of a project given by the frontend is set to "public" and store the result in the database	0	0	0	0	
14	Front end	Merge Request Activation Check	"Merge Requests" in settings should be activated/enabled "Everyone with access" should be allowed to use "Merge Requests"	3	3	3	3	
15	Backend	Merge Request Activation Check	"Merge Requests" in settings should be activated/enabled. "Everyone with access" should be allowed to use "Merge Requests"	3	3	2	2	
16	Front end	Forking Activation Check	"Forking" in settings should be activated/enabled "Everyone with access" should be allowed to use "Forking"	0	0	0	0	
17	Backend	Forking activation check	"Forking" in settings should be activated/enabled "Everyone with access" should be allowed to use "Forking"	2	2	2	2	
18	Front end	Issues Activation Check	"Issues" in settings should be activated/enabled "Everyone with access" should be allowed to use "Issues"	0	0	0	0	
19	Backend	Issues Activation Check	"Issues" in settings should be activated/enabled "Everyone with access" should be allowed to use "Issues"	2	2	1	1	
20	Front end	Yaml / XML configuration file	To edit a yaml/xml file to edit what checks should be displayed to the user.	2	2	2	2	
21	Backend	Mandatory Files Check	To check whether the files (README.md, CONTRIBUTING.md, MAINTAINERS.md) ar	0	0	1	1	
27	Database	Database Change	To store information of each test result in a object linked to lintingResult	2	2	2	2	
32	Front end	Show when a requested repository is not fo	To see an error if my entered URL is wrong so that i can check if the entered URL is co	1	1	1	1	
33	Backend	Visibility Check - Request access	To check the whether the "Request Access" provided by the front end is enabled or not and store the result in the database.	2	2	1	1	
35	Backend	Visibility Check	To check the whether the Gitlab wiki pages feature (provided by the front end) is enabled or not and store the result in the database	2	2	1	1	
36	Backend	Cleanliness - Project description - Backend	To check the whether the project description exists or not	2	2	1	1	
39	Backend	Cleanliness - Project Avatar - Back end	To check the whether the project avatar check provided by the front end is added or not	1	1	2	2	
40	Backend	Severity of Checks	To display the severity of the check	1	1	2	2	
42	Backend	JSON file update	To assign the severity level, description as well as the error message	2	2	3	3	
45	Backend	Check - Squash commits are not allowed	To ensure squash commits are not allowed	2	2	2	2	
46	Backend	Additional Check - README.md-	To check whether README.md file contains atleast one link at a specified URL area	3	3	3	3	
47	Backend	Additional Check - Badges	To check whether atleast one badge exist either in the README.md file or in the badge bar (at the top of the project)	2	2	1	1	
48	Backend	Additional Check - CONTRIBUTING.md-	to check whether CONTRIBUTING.md file contains link to other CONTRIBUTING.md file and display a warning message " CONTRIBUTING.md file does contain link to other CONTRIBUTING.md file" based on the check Acceptance criteria	2	2	3	3	
70	Backend	URL area check	To change the domains to check for within the readme.md in the JSON file without the need for hardcoding	2	2	1	1	
74	Front end	Actionable items front end	To know the top 3 actions I can take to improve my repository	3	3	2	2	

#	Theme	Goal	Feature Name	Est. Size (Feature)	Est. Size (Sprint)	Real Size (Feature)	Real Size (Sprint)	Burn-Down
75	Backend	Actionable items backend	To prioritize checks in regards to a ranking and give the user feedback on what to fix next.	2	2	1	1	
73	Documentation	Status of Crawler	To fix the error if the crawler is not taking action	2	2	5	5	
72	Backend	Sorting Backend	Additional information to sort repositories in the front end	5	5	3	3	
8	Automate Tests	Automate Tests (Github actions)	Automated testing environment to get quick overview of the project	2	2	3	3	

Sprint	Status	Source	Impediment	Resolution
2	Resolved	Team Logo	Incorporating DATEV logo into T-shirt	Sought additional approval to to include DATEV logo on Tshirt both from FAU ajd DATEV
3	Resolved	Creation of adhoc user stories	Unforeseen new user stories based on developer identification of new user story based on/related to a already created user story that is being worked upon	Whenever a developer identifies requirement of adding new stories then the same will be communicated to product owner(s) to be added in the Kanban board based on customer requirements
3	Resolved	Coordination	Coordination between Front end and Backend	Increse communication and coordination between front end and backend developers.
3	Resolved	Duplication of user stories	User stories are created holistically in sprint number 3,2	Avoid Duplication and review the user stories with developers. More coordination between product owners required. Seperate discord channel to discuss new stories before the sprint meeting on wednesday.
4	Resolved	Feedback from Customer	Delay in receiving response	Increase more communication with customer. Inform the coach if in case, we did not receive update at the end of the week.
5	Resolved	Closing completed issues	Completed issues in Kanban board are not closed	Product owners to close the completed issues. Done
6	In-work	Creation of user stories on sprint day	Less time to review new stories before the sprint review meeting	Product owners to create new user stories by Sunday before sprint review meeting so that software developers can review them and ask for clarifications (if any)
7	In-work	User stories are technical	The technical details may not be accurate or mislead SD's	1) Product owners to create user stories by Sunday. 2) We need to create user stories objectively. 3) SDs will review and come back for clarifications. 4) Based on the need, Contact Max if we disagree with requirement or for any changes before sprint review meeting.
8	Resolved	User stories created by software developers	Miscommunication issue whether to contact PO or SD first	Whenever a Product owner finds a user story created by software developer, PO/SD informs and discuss with SD/PO. It is a collective responsibility.

#	Feature Definition of Done	Sprint Release Definition of Done	Project Release Definition of Done
1	Approved by product owner	Demo approved by team	Demo approved by team
2	Code review by release manager	Code review by release manager	User Documentation
3	50% test coverage	50% test coverage	Guide on deploying the application
4		Approved by product owner	Code Documentation
5		Deployable via docker	70% test coverage
6			Code review by release manager
7			Apporved by product owner
8			Deployable via docker
9			User acceptance testing
10			Approved by customer

Type	Link / reference
Overview README	https://github.com/amosproj/amos-ss2021-is-project-linter/blob/staging/README.md
Server README	https://github.com/amosproj/amos-ss2021-is-project-linter/blob/staging/server/README.md
Frontent README	https://github.com/amosproj/amos-ss2021-is-project-linter/blob/staging/angular-frontend/README.md
Full Documentation	https://github.com/amosproj/amos-ss2021-is-project-linter/wiki

\	Context	Name	Version	License	Comment
1	Frontend Framework	Angular	11.2.11	MIT	
2	Backend Framework	Spring (Boot)	2.4.5	Apache License 2.0	
3	Backend Database	Postgres SQL	12	PostgreSQL Licence	Similar to MIT
4	Backend runtime	Docker	20.10.6	Apache License 2.0	
5	Backend orchestrator	Docker-Compose	1.29.1	Apache License 2.0	
6	Backend Framework	GitLab4J-API	4.8.0	MIT	
7	Backend testing	JUnit	5	Eclipse Public License	
8	Backend scheduler	awaitility	3.1.2	Apache License 2.0	
9	Testing Database	H2 in-memory datbase	1.0.200 61217	Eclipse Public License or Mozilla Public License 2.0	
10	Http Testing	HttpClient	4.5.13	Apache License	
11	Java runtime	OpenJDK	11.0.11	GNU GPL v2	with linking exception
12	Frontend UI Components	Angular Material	11.2.12	MIT-like	
13	Frontend Framework	VueJS	2.6.12	MIT	
14	Frontend UI Components	Vuetify		MIT	
15	HTTP client	Axios		MIT	(frontend)
16	Event Bus	Tiny-Emitter		MIT	(frontend)
17	Frontend Date Library	DayJS	1.10.4	MIT	
18	Software Initializer	dumb-init	1.2.5	MIT	(docker)
19	Graph Visualisation	chart.js	2.9.4	MIT	
20	WebServer and ReverseProxy	nginx	stable	2-clause BSD - like	(docker)
21	Exporting CSV	OpenCSV	5.4	Apache License 2.0	
22	Logger	slf4J	1.7.30	MIT	

Last Name	First Name	Value					
Böhm	Lukas	0					
Groth	Patrick	0		**0.00**	**OK**		
Wohlwend	Christoph	0					
Pfister	Jonas	0					
Karri	Veera Venkata Satyanarayana	0		0	No effort		
Neuerer	Christian	0		1	Minimal effort		
Asani	Amir	0		2	Small effort		
Krieg	Nicolas	0		3	Medium effort		
				5	Large effort		
				8	Very large effort		
				13	Too large effort		