

		Project	
Development		BE	Comments
Task / Feature		Time, hours	
FINAL ESTIMATION --->		1531	
Risks & additional activities		543	
1	Estimations errors (5%)	49	
2	Communications (15%)	148	
3	Unit/Integration Tests(15%)	148	
4	Stabilization (15%)	148	
5	Customer interactions (delay) (5%)	49	306
Functionality		988	sum of all estimates of all features below
	AWS infrastructure setup and configuration.	306	
	Basic setup	16	Create users, roles, pick up regions, zones, etc..
	Set up Dev environment	8	region, AZ, IAM, gateway, VPC, EC2 etc..
	Set up Prod environment	24	regions, AZs, IAM, gateways, VPC(pub, private), EC2, ASG, Eureka etc..
	Set up Kinesis dev	8	open ports, add to VPC
	Setup KDF	8	Kinesis Data Firehouse
	Set up User Database dev	8	
	Set up Log Database dev	8	adjust Clickhouse, open ports, add to VPC
	Set up Cache dev	8	
	Set up Log Consumer dev	24	Lambda
	Set up MyLog dev	8	
	Set up MyLog FE dev	8	
	Set up Kinesis prod	16	open ports, add to VPC
	Set up Log Database prod	10	adjust RDS, open ports, add to VPC, replication, sharding
	Set up Log Databases prod	24	adjust Redshift, open ports, add to VPC, replication, sharding
	Set up Cache prod	16	
	Set up Log Consumer dev	8	Lambda + logic
	Set up MyLog BE prod	16	ASG
	Set up MyLog FE prod	8	
	Set up Stage env with all components	24	after dev env, copy components and basic configuration
	Set up CloudWatch and Alerts	16	
	Set up CI/CD	24	
	Set up CDN	16	
	Auth and SSO provider integration	136	

	Documentation investigation	8	tech design, communication with SSO team
	Integrate with SSO provider	32	get token pairs by code, refresh tokens, add HTTP service to work with it
	User registration	40	creating models, flow, possible OTP, design FE components
	User log out	16	log out from SSO, from the app+ redirect + QA
	User sign in	40	authorization, security filters, JWT, design FE components
	Processing, storing and displaying logs	168	
	Set up a project	8	repository, git, initial commit, code style, README
	Logs processing	48	consumers groups to get logs from Kinesis, creating model from that data, store logs(and aggregated data) to a database and cache
	Logs filtration real-time	40	Filtration for WebSocket(which logs show now) by set of parameters + FE + QA
	Logs aggregation real-time	32	get aggregated values by param + refresh charts based on new data (BE, FE, QA)
	Local logs aggregation	16	create local cache bean to send to DB each 2 sec + risks with evicting it
	Real-time logs displaying	24	adjust WebSockets, send new logs based on filters + FE part + QA
	Admin console	128	
	User registration	40	creating models, flow, possible OTP, design FE components, roles
	Application management	40	add/delete an application from the list, settings change
	User log out	8	from the app + redirect + QA
	User sign in	40	authorization, security filters, JWT, design FE components
	MyLog application implementation	250	
	Set up a project BE	8	repository, git, initial commit, code style, README
	Set up a project FE	8	repository, git, initial commit, code style, README
	Set up a project Mobile	8	
	List of available applications	48	get apps by param(BE, FE, Mob, QA)
	Real-time logs displaying	24	real-time data via websocket(BE, FE, Mob, QA)
	Custom charts creation	50	integration with QuickSight
	Language support	8	translations, architecture to easily adding a new one
	Logs filtration	72	Filtration by set of parameters, managing requests between cache and DB (BE, FE, Mob, QA)
	Logs aggregation	24	get aggregated values by param(BE, FE, Mob, QA)