**Initial meeting with the customer**

* Purpose/vision: customer creates app for digitally signed documents
	+ The credibility of the signature is given by an X.509 certificate and a CA
	+ Adobe Acrobat verifies the certificate automatically -> using CA hierarchy and CRL/OCSP -> validity of the certificate at the time of signing
* Problem:
	+ In the production, certificates issued by global CA are used. These are however very expensive – YOSO’s customer cannot provide spare certificates for testing
	+ To test the application, YOSO needs to deploy the app in a testing environment with production certificates (need of almost production-like environment) or to test it in the production environment while disabling end-users to access the app (downtime)
	+ Managing testing certificates is rather complicated for the developers
* Possible certificate use-cases:
	+ Digital signature
	+ Authentication
	+ HTTPS (SSL/TLS)
* Requirements (Minimum Acceptance Criteria):
	+ A single lightweight application
	+ Linux server application
	+ No vertical scaling required
	+ No external database (embedded database or FS can be used)
	+ Creating root/intermediate CAs (tree of CAs)
	+ Creating end X.509 certificates issued by one of the created CAs
	+ Possibility to revoke the certificate
	+ CRL + OCSP endpoint for each of the CA for verifying the validity of the certificates
	+ Web GUI
* Extension requirements:
	+ OpenSSL user templates (in case OpenSSL is used)
	+ Console API
	+ WS interface (for automated test tool – script requests a new certificate just for the test)
	+ Docker
	+ Authentication (would be nice, but the tool will run in the intranet)
	+ Categorization of the certificates – eg. possibility to choose the SQLite database file
* Actors:
	+ Few administrators (1-3)
	+ Automated test script
* Technologies:
	+ Crypto library of our choice or OpenSSL
* Delivery:
	+ Packaged SW (.jar, .war)
	+ In case Python is chosen – Linux systemd script
	+ DEMO meetings: delivery 1-2 days before the meeting – in our environment accessible to the customer.
	+ Zipped GIT repo
* Toolchain:
	+ The team is allowed to use the tool infrastructure provided by the ASWI lecturers (KIV GitLab and Redmine)
* Documentation:
	+ Diagrams (use-cases, components, classes)
	+ Code comments
	+ User & Administration documentation – including step-by-step screenshots
		- Administration – installation, restart, updates, configuration, database, backups
* Communication: Via email
* Demo meetings: Bi-weekly on Friday at 2pm (starting 12.03. 2021)