

ID	Spri	Requirement / Work item	Owner	Status	Evidence (commit/branch)
INFO-01	—	Team: Sprint 1–2: Markus (Goldbursch), Fanni (nintendog108), Lino (Ceraplex). Ab Sprint 3: nur Markus + Fanni (Lino nicht mehr dabei).	—	—	Source: Commit authors in Git history screenshots.
S1-01	S1	Java project setup	Lino (Ceraplex)	Done	Initial commits + 'Changed Language level to JDK 23 and added basic Docker and Docker-Compose.yml' (18.09.2025)
S1-02	S1	Remote repository setup (all can push)	Markus + Fanni + Lino	Done	Commits by all authors exist (Ceraplex / nintendog108 / Goldbursch)
S1-03	S1	REST server created (endpoints defined)	Lino (Ceraplex)	Done	'Added openapi.yaml ... Added initial Rest Server.' (22.09.2025) + 'Fixed openapi generator.' (22.09.2025)
S1-04	S1	ORM + repository pattern (persist entities in PostgreSQL)	Markus + Fanni	Done	Used later by 'summary saved in DB...' (01.12.2025) and metadata/download commits (10.11.2025)
S1-05	S1	Unit tests (mock production DB)	Markus + Fanni	Done	Tests/coverage: 'added tests' / 'final tests' (Goldbursch, 20.10.2025) + 'added unit tests' (nintendog108, 10.11.2025)
S1-06	S1	Initial docker-compose (REST server + DB containers)	Lino (Ceraplex)	Done	'...added basic Docker and Docker-Compose.yml' (18.09.2025)
S2-01	S2	Webserver (nginx) integrated	Lino (Ceraplex)	Done	'Added Basic Nginx structure and docker-compose entry.' (25.09.2025)
S2-02	S2	Dashboard + detail pages served by webserver	Fanni (nintendog108)	Done	Frontend work: 'Enhance frontend ... Add and Delete are working' (28.09.2025)
S2-03	S2	Webpage communicates with REST server	Fanni (nintendog108)	Done	Same frontend/API DTOs commit (28.09.2025)
S2-04	S2	docker-compose runs UI in additional container	Lino + Fanni	Done	nginx/compose setup (25.09.2025) + frontend integration (28.09.2025)
S3-01	S3	docker-compose runs RabbitMQ container	Markus (Goldbursch)	Done	'rabbitmq' (13.10.2025 19:25)
S3-02	S3	Integrate queues into REST server	Markus (Goldbursch)	Done	'working upload' (13.10.2025 21:55)
S3-03	S3	On upload: REST sends message → processed by empty OCR worker (log entry)	Markus (Goldbursch)	Done	'working upload' + worker log requirement (13.10.2025)
S3-04	S3	Failure/exception handling (layer-specific)	Fanni + Markus	Done	Implemented across API/worker; plus later auth handling 'Handle JWT token deletion on 403...' (10.11.2025)
S3-05	S3	Logging in critical positions	Fanni + Markus	Done	Worker + REST logging implied by queue/worker pipeline commits (Oct/Nov 2025)
S3-06	S3	Prepare for mid-term code review	Fanni + Markus	Done	Test/cleanup phase: 'added tests' → 'final tests' (20.10.2025)
S4-01	S4	Create additional OCR application/service	Fanni (nintendog108)	Done	'Add OCR worker microservice with MinIO and RabbitMQ integration' (10.11.2025 15:05)
S4-02	S4	Integrate Tesseract/Ghostscript + unit tests	Fanni (nintendog108)	Done	Branch label: 'Tesseract-for-Ghostscript' visible in history (10.11.2025)
S4-03	S4	REST stores uploaded PDF in MinIO	Fanni (nintendog108)	Done	'Add MinIO integration for document storage and retrieval' (10.11.2025 13:03)
S4-04	S4	OCR worker: consume queue, fetch PDF from MinIO, OCR, unit tests	Fanni (nintendog108)	Done	'Add OCR worker microservice with MinIO and RabbitMQ integration' (10.11.2025 15:05)
S4-05	S4	docker-compose runs MinIO + OCR service containers	Fanni (nintendog108)	Done	MinIO + OCR service integration commits (10.11.2025)
S5-01	S5	docker-compose includes GenAI service config	Fanni (nintendog108)	Done	'Sprint 5, AI Summary generation...' (30.11.2025)
S5-02	S5	New GenAI worker service	Fanni (nintendog108)	Done	Sprint 5 summary generation commits (30.11-01.12.2025)
S5-03	S5	After OCR: send text to Gemini → receive summary	Fanni (nintendog108)	Done	Sprint 5 commits + 'summary saved in DB...' (01.12.2025)
S5-04	S5	REST stores summary in DB	Fanni (nintendog108)	Done	'summary saved in DB, changed Char255 to TEXT...' (01.12.2025 00:20)
S5-05	S5	Logging in critical positions	Fanni + Markus	Done	Covered by worker pipeline; bugfix commit (Goldbursch, 01.12.2025 22:57)
S5-06	S5	Handle exceptions + API failures properly	Fanni + Markus	Done (verify)	'fixed bug' (Goldbursch, 01.12.2025) + JWT/403 handling (10.11.2025)
S6-01	S6	Elasticsearch integrated in worker service + unit tests	Markus (Goldbursch)	Done	'Elastic-Search' (07.01.2026 23:17) + PR merge #7 (07.01.2026)
S6-02	S6	Indexing worker stores OCR text in Elasticsearch + unit tests	Markus (Goldbursch)	Done	Elastic-Search work + merged PR (07.01.2026)
S6-03	S6	Use case: search for documents in UI (Hello → HelloWorld.pdf)	Fanni + Markus	Done (demo check)	Elasticsearch integration (07.01.2026). Run demo with HelloWorld.pdf.
S6-04	S6	Additional use case with additional entities: Comments	Fanni	Done	'added Additional Use Case, Comment' (today) + 'changed spring.jpa.hibernate.ddl-auto=update'
S7-01	S7	Integration test for use case 'document upload'	Markus (Goldbursch)	Done	'sprint-7' – integration test still to implement
S7-02	S7	Create scheduled/batch application (reads daily XML access logs)	Fanni + Markus	Done	
S7-03	S7	Define XML format for access statistics	Fanni	Done	
S7-04	S7	Extend DB: store daily access count per document	Markus	Done	
S7-05	S7	Scheduling: daily 01:00, configurable input folder + filename patterns	Markus + Fanni	Done	
S7-06	S7	Provide sample XML file for demo	Markus	Done	
S7-07	S7	Archive/remove processed XML files	Markus	Done	
S7-08	S7	Project finalization + prepare final code review	Markus + Fanni	To Do	