**Karte 1: Titelfolie**

* Begrüßung
* Vorstellung des Projekts:
  + Odoo-Modul für die ZIP-komprimierte Dateiverwaltung im Dokumentenmanagementsystem (DMS)

**Karte 2: Kurzvorstellung**

* Firma: Sirum GmbH -> SIRUM Logistics Suite -> modular, webbasiert Softwarelösung für digitale Logistik.
* Ich: Mark Albers – Entwicklung bon ZIP-komprimierten Dokumentenexport im DMS

**Karte 3: Überblick**

* Themenüberblick:
  + Ausgangslage
  + Modulübersicht
  + Qualitätssicherung
  + Zeitersparnis
  + Aktueller Stand
  + Fazit

**Karte 4: Problem & Ziel**

* Problem:
  + Kein Massen-Download -> Zeitintensiv und anfällig für Fehler.
  + Moderne Browser blockieren Mehrfachdownloads
* Ziel:
  + Komfortable Filterfunktion + gezielte Dokumente für Massenexport
  + Schnell (Beschleunigt), sicher, komfortabel

**Karte 5: Technische Anforderungen**

* Nutzer setzt Filter (Datum Pflicht, Modelle optional)
* System erstellt ZIP → im DMS gespeichert
* Export im Anhang und jederzeit abrufbar

**Karte 6: Modul-Funktionalität**

* Filter: Datum + Modell
* Status: Draft → Open → Done
* Automatischer Exportname
* ZIP-Erstellung per Button
* ZIP mit passenden Dokumenten -> dauerhaft gespeichert

**Karte 7: Revisionssicherheit**

* Eindeutige Dateinamen -> file.txt, file1.txt
* Markierung exportierter Dokumente
* Protokolliert:
  + Name, Datum, Filter, Datei größe

**Karte 8: Benutzeroberfläche**

* Nicht zu tief in das Thema
* Testsystem
* Übersicht der Kategorien aller Exporte:
  + Name, Filter, Datum, Datei größe, Anzahl Dokumente

**Karte 9: Ablauf Schritt für Schritt**

1. „Erstellen“ → Status „Draft“
2. Filter setzen (Datum Pflicht, Modell optional) -> Modelle leer = Alle berücksichtig
3. „Check Export“
4. Dokumente gefunden? -> Nein = Error -> Ja = Zwischengespeichert +

Status „Open“

1. Dokumente prüfen -> Details (Name, etc) -> Dokumente die beteiligt sind
2. „Pack as ZIP“ → Status „Done“
3. ZIP im hintergrund erstellt
4. ZIP dauerhaft im DMS verfügbar

**Karte 10: Qualitätssicherung**

* Unit-Tests
* GitLab CI:
  + Linting -> Einhaltung von Code-Standards
  + Deploy -> Erfolgreiche Bereitstellung sicherstellen
  + Translate-Tests -> Vollständigkeit und Korrektheit von Übersetzung prüfen
* Code-Review:
  + Lesbarkeit, Sicherheit, Einhaltung von Best Practices

**Karte 11: Zeitersparnis**

* Beispiel: 3 Vorgänge + 20 Arbeitstage
* Vorher: 20 Min/Vorgang → 20 Std/Monat
* Nachher: 2 Min/Vorgang → 2 Std/Monat
* Einsparung: 18 Std/Monat -> Zeitfresser nahzu entfernt
* Balken Diagram vorher, nacher pro Export

**Karte 12: Aktueller Stand**

* Entwicklung & Tests: abgeschlossen
* Offene Punkte: Feinschliff & Review
* Einführung steht bevor nach Freigabe
* Erweiterungen möglich (z. B. wöchentlicher Auto-Export)

**Karte 13: Fazit**

* Anforderungen erfüllt
* ZIP-Export löst Browser-Problematik
* Automatisiert, zeitsparend (18 Std/Monat)
* Modular erweiterbar

**Karte 14: Abschluss**

* Danke!
* Fragen oder Anregungen?