

Akt von Daten: - Anfragen {auswählbar
- Berechnungsfähig}

Anfrage: fordert lange, kurze Frage oder Terminvereinbarung
Belegung: ein neues Datensatz wird erstellt falls noch nicht vorhanden
Datensatz muss aktualisierbar sein

Frage: - bestmögliche Information? (Suche nach Name, Telefonnummer, ...?)
- sind die Fälle historisierungsrelevant?
- Suchstrategien für unterschiedliche Belegungsstile: also für alle?
es geht in einem Datensatz die Belegungsstile mit aufzuheben
- für andere Belegungsstile: Email (privat & beruflich), ...
- dass Admin Prozessabläufe verbindet

Überprüfung ob alle Fälle ausgeführt werden → wenn nicht → Meldung → Wahl zwischen so lassen & weiter ausstellen
weitere Informationfelder müssen weitergeführt sein & andere Mitarbeiter sollen auf aktuelle Fälle zugreifen können (Textfeld, Zahlenfeld, Datum, ...)

Suchfeld für Fälle
im Datensatz: - Termin (wöchentlich, wöchentlich, wöchentlich, ...)
- email

Suchstrategie: - eine ausgewählte Suchkriterien (Mitarbeiter, Jahr, ...)
- eine virtuelle Kartensuche (z.B. wie viel Belegungen pro Suchkriterium)
Disjunkt, Suchstrategie: - die Suchkriterien werden in einer Suchkriterien-Struktur (z.B. Suchkriterien, Suchkriterien, Suchkriterien) ...
Suchkriterien, Suchkriterien, Suchkriterien
- Suchkriterien auswählbar & filterbar
- Filtermöglichkeiten sollen gegeben sein (mit Filter für Prozess)
- Prozess soll privat oder öffentlich sein
- wenn Prozess gegeben wird sollen Filtermöglichkeiten frei wählbar werden können
- Prozess hat Suchkriterien
- PDF, XLSX und CSV auswählbar
- optional mit Google

Belegung der Accounts: Prozess-gestütztes Konto
Basis: - wenn Suchkriterien anlegen, bestehende Konten löschen
- Suchkriterien löschen
- Suchkriterien Prozess löschen
- Suchkriterien Prozess erstellen
- Suchkriterien löschen
Erweitert: - wenn Felder an Formulare hinzufügen
- Suchkriterien Prozess erstellen
Admin: - Kontenverwaltung (was anlegen, Rechte zuordnen, löschen)
↳ es muss sein 2 Admin-Konten geben

streamlit
django

Technische Anforderungen: - Client-Server-Architektur (nicht zentrale Datenbank auf der ein Microservice (HUB))
- Microservice verbindet Admin & kommuniziert mit Datenbank
- Daten Import muss
- Rechte
- eine Version

UX Anforderungen: - schnelle Filtermöglichkeiten, optische Hinweise
- Benutzerfreundlichkeit