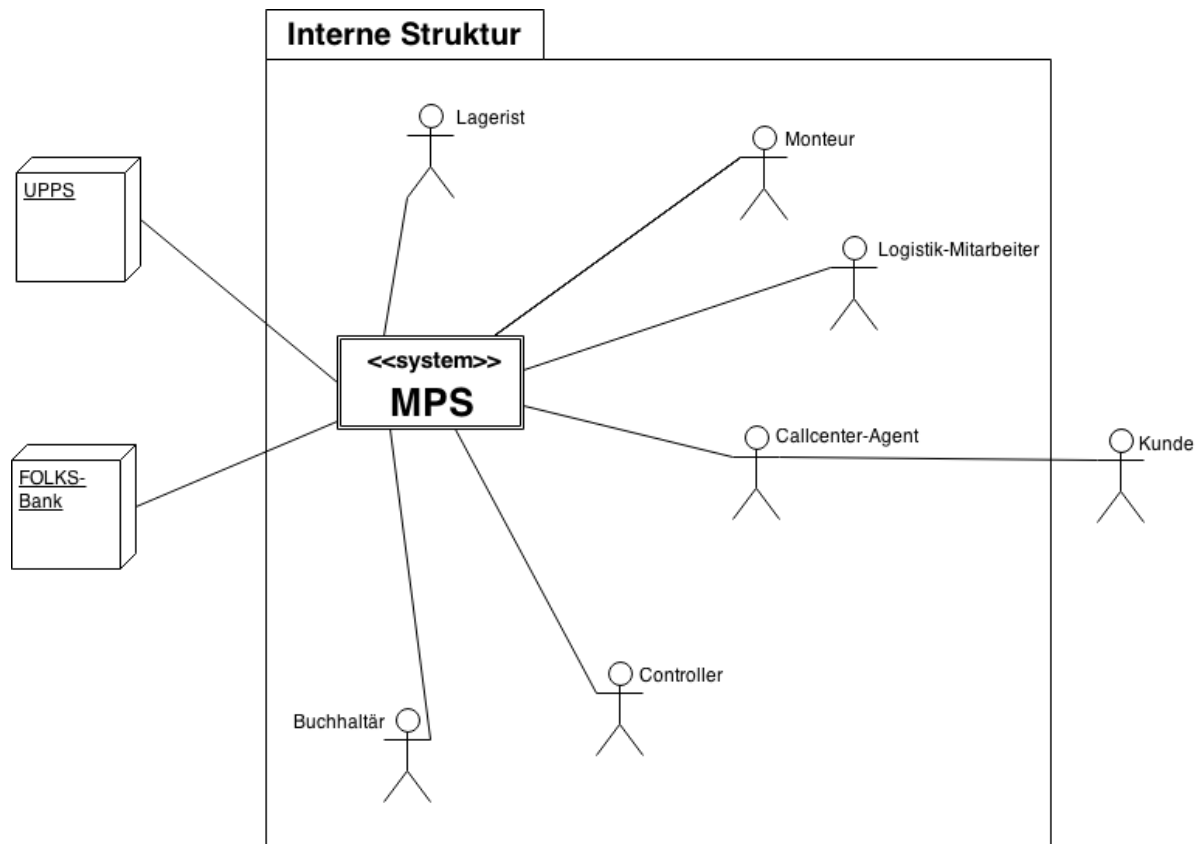


## Aufgabenblatt 2

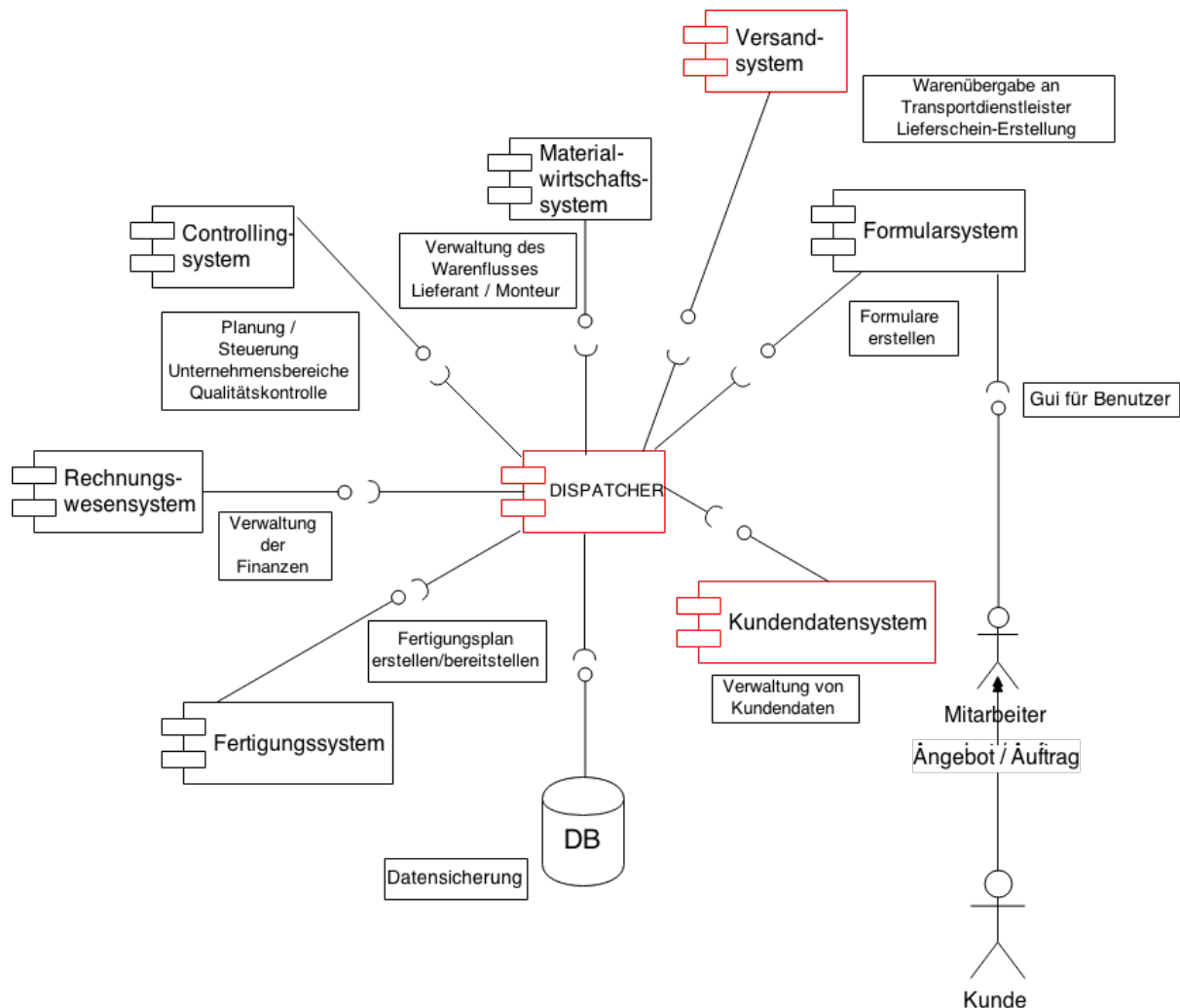
### Aufgabe 4



*Entscheidungen/Annahmen:* Jeder Mitarbeiter ist einer Komponente/Modul zugeordnet.  
Die Komponenten spiegeln auch die einzelnen  
Abteilungen/Bereiche des Unternehmens wieder.

- Zuordnung der Mitarbeiter zu den Komponenten/Modulen
  - Callcenter-Agent → FormularSystem
  - Monteur → FertigungsSystem
  - Lagerist → VersandSystem
  - Buchhaltär → RechnungswesenSystem
  - Controller → ControllingSystem
  - Logistik-Mitarbeiter → MaterialwirtschaftsSystem

## MPS

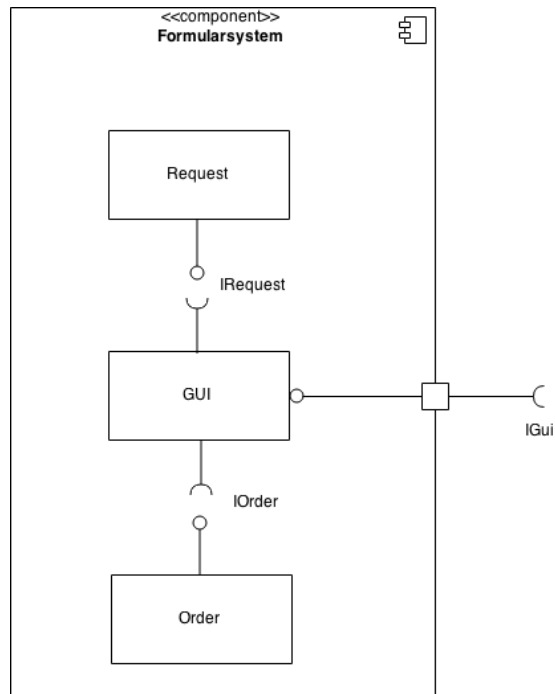


*Entscheidungen/Annahmen:* Im Mittelpunkt gibt es nun bei uns einen Dispatcher/Verteiler, der alle Komponenten miteinander verbindet. Ausserdem wurde das Versandsystem von der Material-Wirtschafts-System-Komponente getrennt und ist nun direkt am Dispatcher angebunden. Zusätzlich wurde noch die Kundendatensystem-Komponente hinzugefügt, da vorher noch keiner Verwaltung der Kundendaten zur Verfügung stand. Die Oberfläche-Komponente wurde entfernt, da der Mitarbeiter über eine GUI direkt mit dem Formularsystem interagiert.

- Anforderungen/Begründung der Mitarbeiter vom MPS

Mitarbeiter	Anforderung	Begründung
Callcenter-Agent	leichte Benutzung	-neue Mitarbeiter können schneller eingearbeitet werden -weniger Fehlerpotenzial
Monteur	Das System muss Fertigungspläne zur Verfügung stellen	Somit weiß der Mitarbeiter welche Teile er benötigt
Lagerist	Das System muss Lieferscheine zur Verfügung stellen	Damit kann gewährleistet werden, dass die richtige Ware auch beim richtigen Empfänger ankommt
Buchhaltär	100% genaue Berechnungen vom System	Falsche Rechnungen würden den Kunden irritieren und unnötigen Ärger erzeugen
Controller	Meldung vom System bei falschen/unzulässigen Eingaben von Benutzern	So können gravierende Fehler frühzeitig entdeckt werden
Logistik-Mitarbeiter	Meldung vom System bei geringen Bestand	Lieferverzögerungen können so vermieden werden

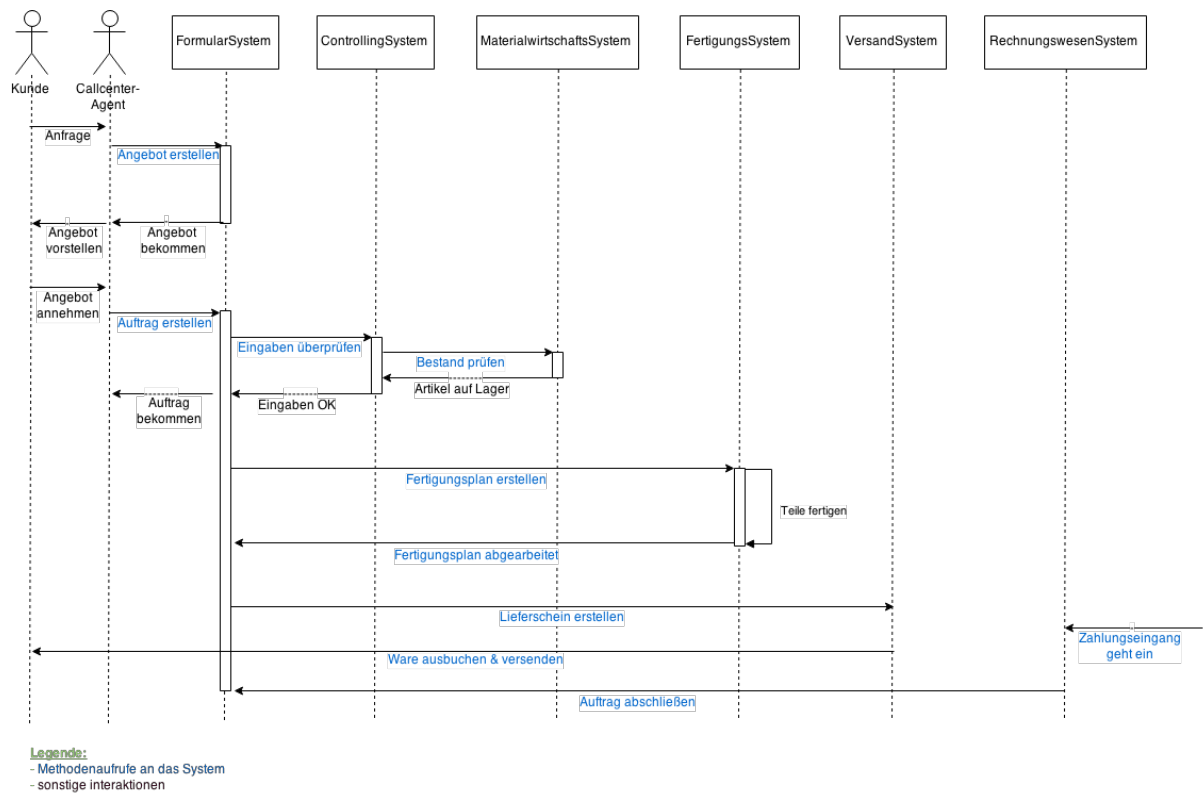
- Interne Aufbau (Innensicht) der Formularsystem-Komponente



#### Szenario für das Sequenzdiagramm

1. Kunde ruft den Callcenter-Agent an und will ein Angebot
2. Callcenter-Agent erstellt Angebot
3. Kunde entscheidet, ob er das Angebot wahrnimmt oder nicht
4. Callcenter-Agent erstellt aus Angebot einen Auftrag
5. MPS erstellt einen Fertigungsplan
6. Schleife zum Fertigen des Bauteils (komplexes;- oder Einfaches Bauteil)
7. Auftrag wird ausgeliefert. Der Versand vermerkt „Lieferung“ erfolgt.
8. MPS erhält Zahlungseingang vom Kunden
9. Buchhaltung: Rechnung bezahlt -----> MPS -> Auftrag abgeschlossen markieren

„Wir gehen davon aus, dass der Kunde schon im System existiert und ein Angebot haben möchte und sich für dieses entscheidet“.



### Parameter für das Sequenzdiagramm

1. `createRequest` (int customerNumber, int productNumber) -> int requestNumber
2. `createOrder` (int requestNumber) -> int orderNumber
  - 2.1 `checkParameter` (Request request) -> Bool
  - 2.2 `checkMaterialsAvailable` (Request request) -> Bool
3. `createManufacturingSchedule` (int orderNumber) -> int manufacturingNumber
4. `setManufacturingScheduleStatus` (int manufacturingNumber) -> int (0|1|-1)
5. `createDeliveryNote` (int orderNumber) -> int deliveryNoteNumber
6. `setPaymentStatus` (int orderNumber) -> int (0|1|-1)
7. `bookOutProduct` (Collection productList) -> int (0|1|-1)
8. `setOrderClose` (int orderNumber) -> int (0|1|-1)

### Objekte

- **Request**
  - I. int requestNumber
  - II. int customerNumber
  - III. int productNumber
- **Order**
  - I. int orderNumber
  - II. int requestNumber
  - III. Collection productList
  - IV. Bool paymentReceipt
- **ManufacturingSchedule**
  - I. int manufacturingNumber
  - II. Date deadline
  - III. int orderNumber

IV. Collection parts

- **DeliveryNote**

I. int deliveryNoteNumber

II. int orderNumber

III. int customerNumber

IV. Collection productList