

Abgabe Projektarbeit Paul Friedrich Vierkorn

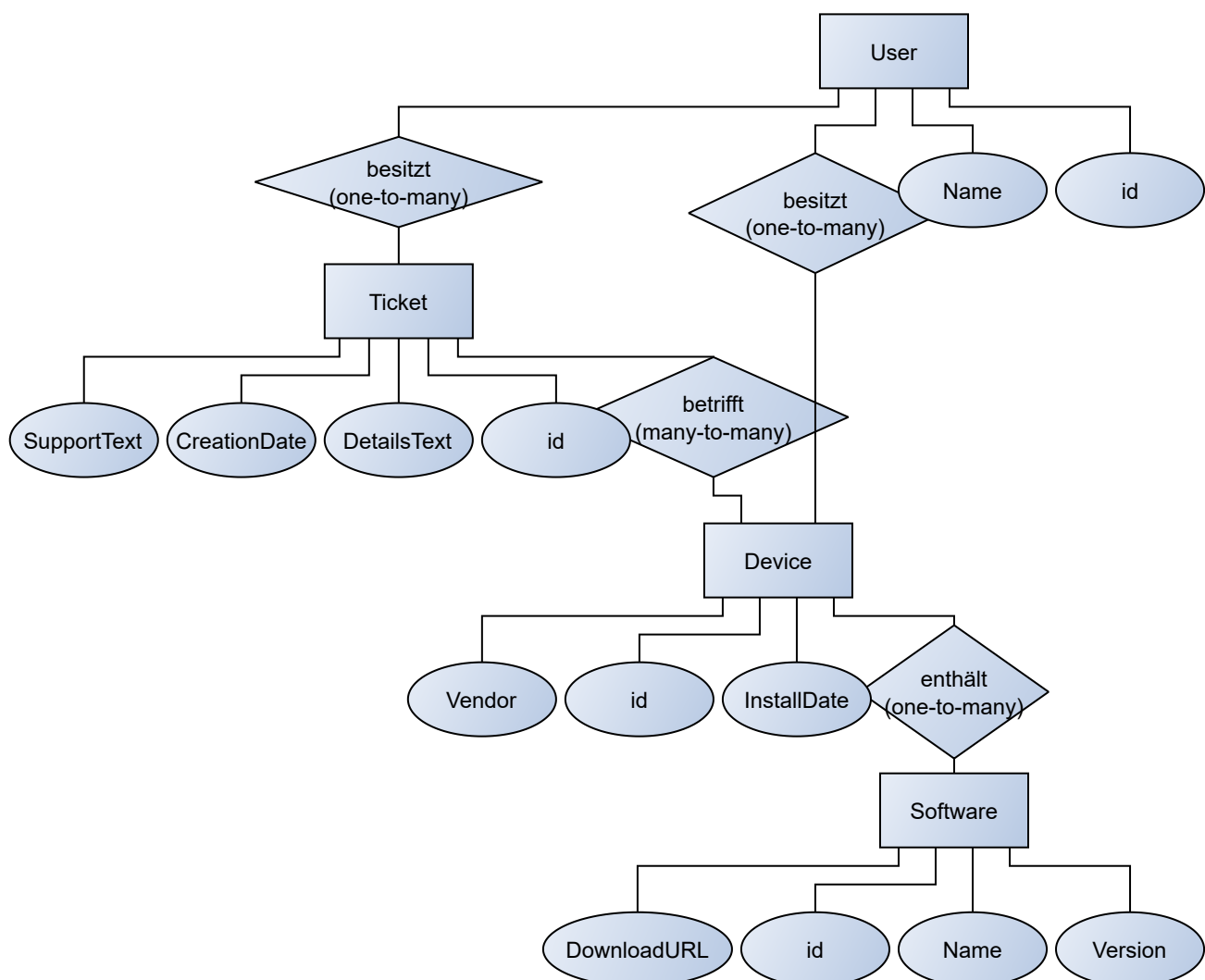
Aufgabe 1

In der Firma Biotronik gibt es ein automatisiertes Ticketsystem, welches Probleme mit der technischen Ausstattung der Mitarbeiter aufnimmt und für den IT-Support aufbereitet.

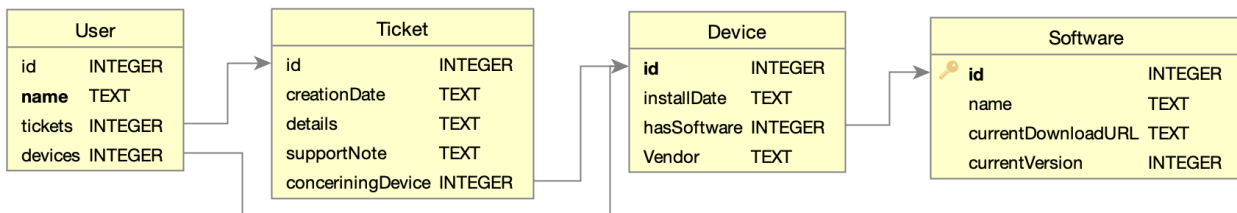
Das Ticketsystem der Biotronik basiert auf einem kommerziellen System, welches ebenfalls an das Inventarsystem angebunden ist. Mithilfe dieser Anbindung können Geräte direkt verschiedenen Mitarbeitern zugewiesen werden. Die verschiedenen Geräte haben verschiedenste Eigenschaften, zum Beispiel die Liste von installierter Software und Daten zur Hardware.

Durch diese Zusatzinformationen, die direkt an jedem Ticket hängen, kann der Support mögliche Rückfragen minimieren und die Effizienz erhöhen.

Aufgabe 2



Aufgabe 3



Aufgabe 4

```

DROP TABLE IF EXISTS "Software";
DROP TABLE IF EXISTS "Device";
DROP TABLE IF EXISTS "User";
DROP TABLE IF EXISTS "Ticket";
DROP TABLE IF EXISTS "hasSoftware";
DROP TABLE IF EXISTS "hasDevice";
DROP TABLE IF EXISTS "hasTicket";

CREATE TABLE IF NOT EXISTS "Software" (
    "id"                INTEGER,
    "name"              TEXT,
    "currentDownloadURL" TEXT,
    "currentVersion"    INTEGER,
);
CREATE TABLE IF NOT EXISTS "Device" (
    "id"                INTEGER,
    "installDate"       TEXT,
    "hasSoftware"       INTEGER,
    "Vendor"            TEXT,
);
CREATE TABLE IF NOT EXISTS "User" (
    "id"                INTEGER,
    "name"              TEXT
);
CREATE TABLE IF NOT EXISTS "Ticket" (
    "id"                INTEGER,
    "creationDate"      TEXT,
    "details"           TEXT,
    "supportNote"       TEXT,
    "concerningDevice"  INTEGER
);
CREATE TABLE IF NOT EXISTS "hasTicket" (
    "userid"  INTEGER,
    "ticketid" INTEGER
);
CREATE TABLE IF NOT EXISTS "hasSoftware" (
    "deviceid"  INTEGER,
    "softwareid" INTEGER
);
CREATE TABLE IF NOT EXISTS "hasDevice" (
    "userid"  INTEGER,

```

```
    "deviceid" INTEGER
);

COMMIT;
```

Version 2 (mit Foreign Key Constraints)

```
DROP TABLE IF EXISTS "Software";
DROP TABLE IF EXISTS "Device";
DROP TABLE IF EXISTS "User";
DROP TABLE IF EXISTS "Ticket";
DROP TABLE IF EXISTS "hasSoftware";
DROP TABLE IF EXISTS "hasDevice";
DROP TABLE IF EXISTS "hasTicket";

CREATE TABLE IF NOT EXISTS "Software" (
    "id"                INTEGER,
    "name"              TEXT,
    "currentDownloadURL" TEXT,
    "currentVersion"    INTEGER,
);
CREATE TABLE IF NOT EXISTS "Device" (
    "id"                INTEGER,
    "installDate"       TEXT,
    "hasSoftware"        INTEGER,
    "Vendor"            TEXT,
);
CREATE TABLE IF NOT EXISTS "User" (
    "id"                INTEGER,
    "name"              TEXT
);
CREATE TABLE IF NOT EXISTS "Ticket" (
    "id"                INTEGER,
    "creationDate"       TEXT,
    "details"           TEXT,
    "supportNote"        TEXT,
    "concerningDevice"   INTEGER,
    FOREIGN KEY("concerningDevice") REFERENCES "Device" ("id")
);
CREATE TABLE IF NOT EXISTS "hasTicket" (
    "userid"            INTEGER,
    "ticketid"          INTEGER,
    FOREIGN KEY("ticketid") REFERENCES "Ticket" ("id"),
    FOREIGN KEY("userid") REFERENCES "User" ("id")
);
CREATE TABLE IF NOT EXISTS "hasSoftware" (
    "deviceid"          INTEGER,
    "softwareid"        INTEGER,
    FOREIGN KEY ("deviceid") REFERENCES "Device" ("id"),
    FOREIGN KEY ("softwareid") REFERENCES "Software" ("id")
);
```

```
CREATE TABLE IF NOT EXISTS "hasDevice" (  
  "userid" INTEGER,  
  "deviceid" INTEGER,  
  FOREIGN KEY ("userid") REFERENCES "User" ("id"),  
  FOREIGN KEY ("deviceid") REFERENCES "Device" ("id")  
);  
  
COMMIT;
```

Aufgabe 5

```
SELECT * FROM User, Device WHERE name LIKE '%Vierkorn' and Device.id LIKE  
User.devices  
SELECT name FROM Software, Ticket WHERE Software.currentDownloadURL LIKE  
'http://%' AND Ticket.supportNote LIKE '%INSTALLATION%'
```