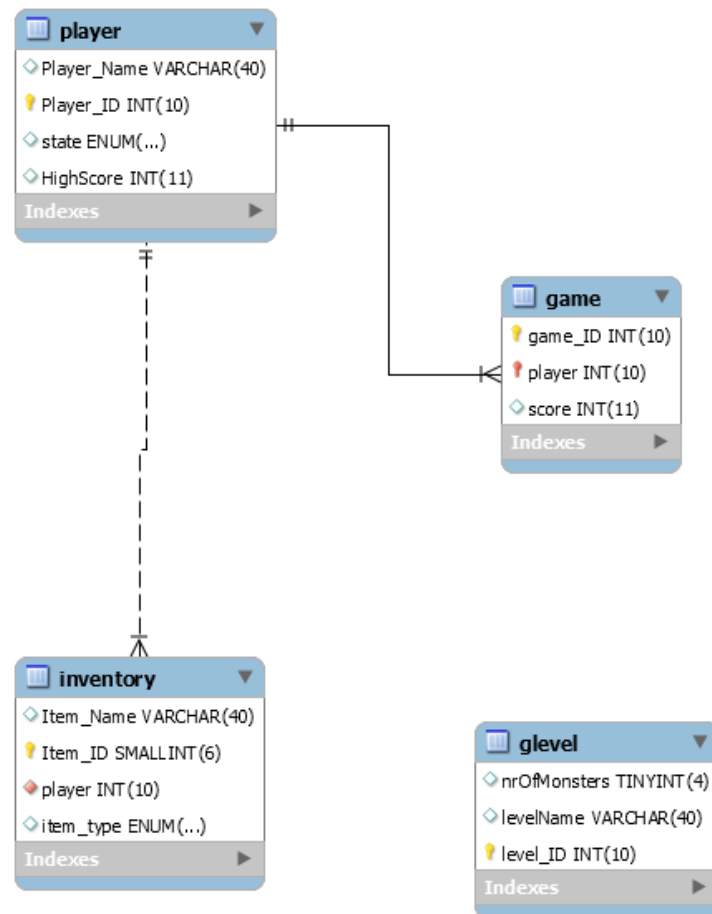
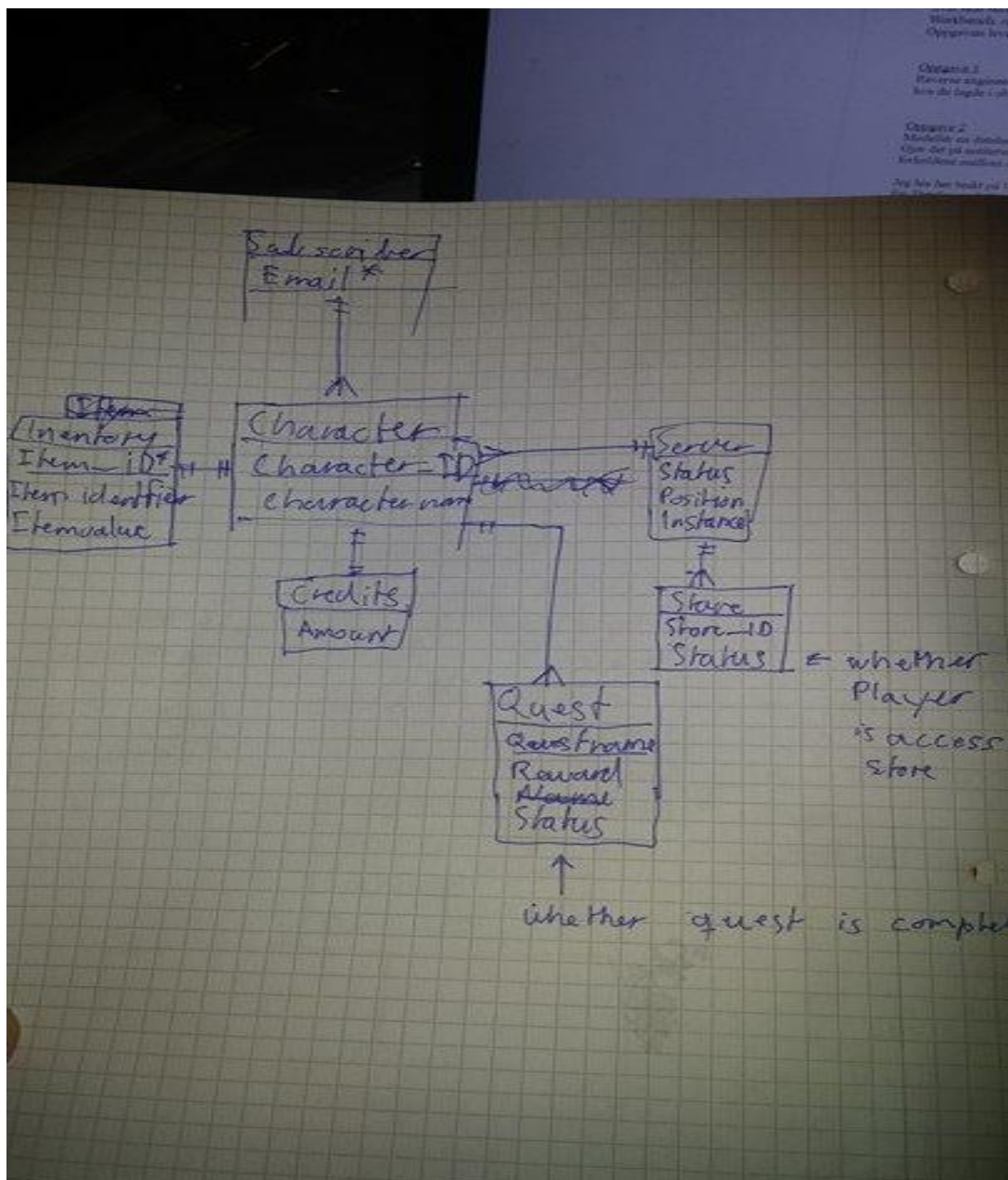


Rapport, Oblig 2 databaser og nettverk, Adrian Ihle



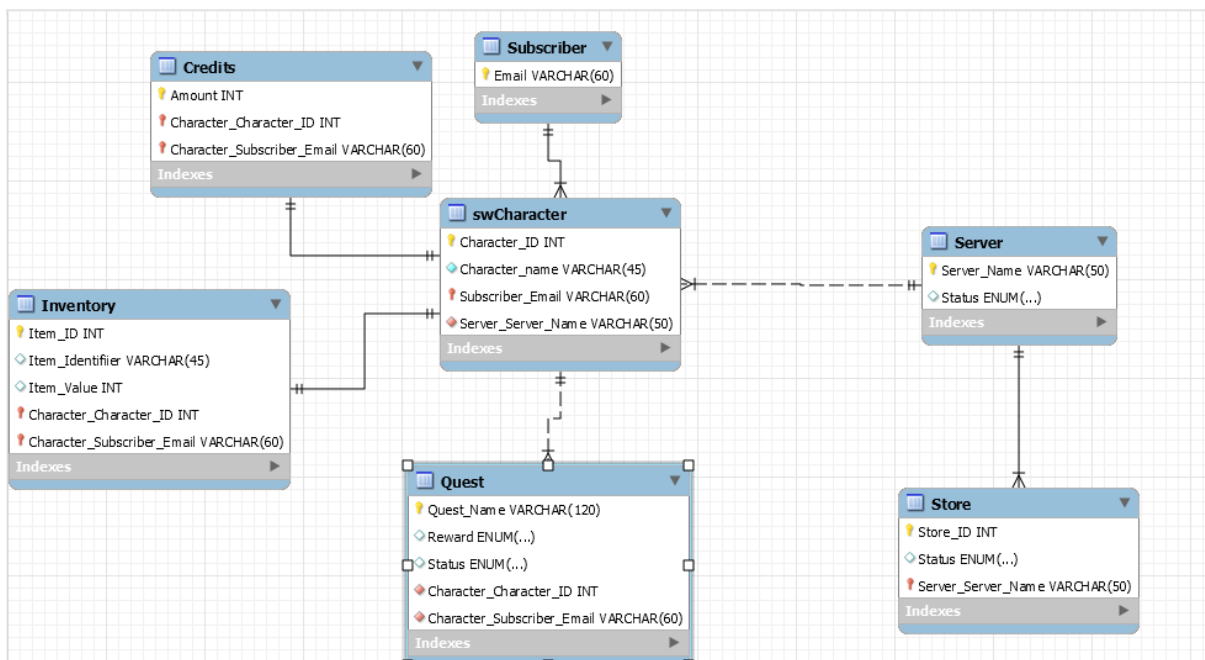
ER av oblig 1.



ER av oblig 2 på papir

### Oppgave 3 normalisering

Ettersom modellen ikke har noen komposittnøkler i seg, samt alle Fremmednøkler kun fungerer som pekere vil alle disse nodene kunne operere helt uavhengig hverandre med ingen redundans.



## Workbench modell

Her ser vi at jeg har lagt til en del ekstra for å få ting til å være mer fungerende. Det som er tydelig her at Modellen mener at det er viktig og ta med seg videre epost adressen til Subscriber uten at det burde egentlig være nødvendig, men jeg har latt det forbli sånn.

## Kode Generert av modell

```
-- MySQL Workbench Forward Engineering
```

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
```

```
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
```

```
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';
```

```
-- Schema mydb
```

```
-- -----  
-- Schema mydb  
-- -----
```

```
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;  
USE `mydb` ;
```

```
-- -----  
-- Table `mydb`.`Subscriber`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Subscriber` (  
  `Email` VARCHAR(60) NOT NULL,  
  PRIMARY KEY (`Email`))  
ENGINE = InnoDB;
```

```
-- -----  
-- Table `mydb`.`Server`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Server` (  
  `Server_Name` VARCHAR(50) NOT NULL,  
  `Status` ENUM('Online', 'Offline', 'Full') NULL,  
  PRIMARY KEY (`Server_Name`))  
ENGINE = InnoDB;
```

```
-- -----  
-- Table `mydb`.`Character`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`swcharacter` (  
  `Character_ID` INT NOT NULL AUTO_INCREMENT,  
  `Character_name` VARCHAR(45) NOT NULL,
```

```

`Subscriber_Email` VARCHAR(60) NOT NULL,
`Server_Server_Name` VARCHAR(50) NOT NULL,
PRIMARY KEY (`Character_ID`, `Subscriber_Email`),
UNIQUE INDEX `Character_name_UNIQUE` (`Character_name` ASC),
INDEX `fk_Character_ID_Subscriber_idx` (`Subscriber_Email` ASC),
INDEX `fk_Character_Server1_idx` (`Server_Server_Name` ASC),
CONSTRAINT `fk_Character_ID_Subscriber`
    FOREIGN KEY (`Subscriber_Email`)
    REFERENCES `mydb`.`Subscriber` (`Email`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
CONSTRAINT `fk_Character_Server1`
    FOREIGN KEY (`Server_Server_Name`)
    REFERENCES `mydb`.`Server` (`Server_Name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

-----

-- Table `mydb`.`Credits`

-----

```

CREATE TABLE IF NOT EXISTS `mydb`.`Credits` (
    `Amount` INT NOT NULL,
    `Character_Character_ID` INT NOT NULL,
    `Character_Subscriber_Email` VARCHAR(60) NOT NULL,
    PRIMARY KEY (`Amount`, `Character_Character_ID`, `Character_Subscriber_Email`),
    INDEX `fk_Credits_Character1_idx` (`Character_Character_ID` ASC, `Character_Subscriber_Email`
    ASC),
    CONSTRAINT `fk_Credits_Character1`
        FOREIGN KEY (`Character_Character_ID`, `Character_Subscriber_Email`)

```

```
REFERENCES `mydb`.`swcharacter` (`Character_ID`, `Subscriber_Email`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-- Table `mydb`.`Inventory`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Inventory` (
  `Item_ID` INT NOT NULL AUTO_INCREMENT,
  `Item_Identifiier` VARCHAR(45) NULL,
  `Item_Value` INT UNSIGNED NULL,
  `Character_Character_ID` INT NOT NULL,
  `Character_Subscriber_Email` VARCHAR(60) NOT NULL,
  PRIMARY KEY (`Item_ID`, `Character_Character_ID`, `Character_Subscriber_Email`),
  UNIQUE INDEX `Item_Identifiier_UNIQUE` (`Item_Identifiier` ASC),
  INDEX `fk_Inventory_Character1_idx` (`Character_Character_ID` ASC, `Character_Subscriber_Email`
  ASC),
  CONSTRAINT `fk_Inventory_Character1`
    FOREIGN KEY (`Character_Character_ID`, `Character_Subscriber_Email`)
    REFERENCES `mydb`.`swcharacter` (`Character_ID`, `Subscriber_Email`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-- Table `mydb`.`Quest`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Quest` (
```

```

`Quest_Name` INT NOT NULL,

`Reward` ENUM('Item', 'Credits', 'Other') NULL,

`Status` ENUM('Active', 'Completed') NULL,

`Character_Character_ID` INT NOT NULL,

`Character_Subscriber_Email` VARCHAR(60) NOT NULL,

PRIMARY KEY (`Quest_Name`),

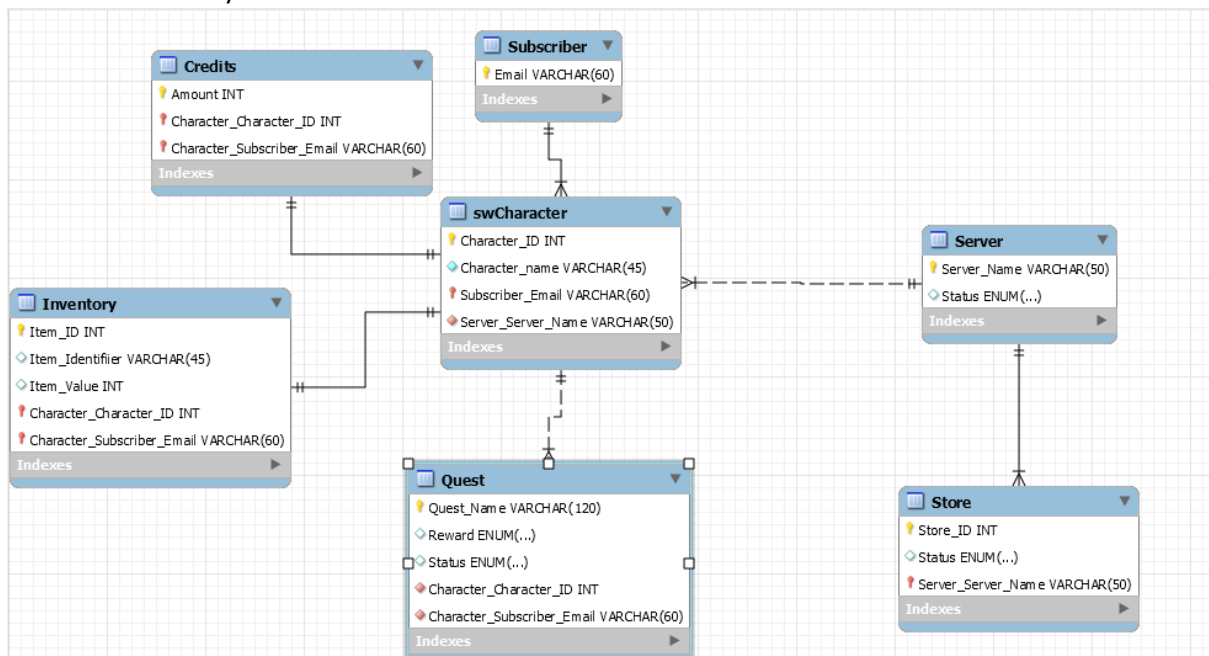
INDEX `fk_Quest_Character1_idx` (`Character_Character_ID` ASC, `Character_Subscriber_Email`
ASC),

CONSTRAINT `fk_Quest_Character1`

FOREIGN KEY (`Character_Character_ID`, `Character_Subscriber_Email`)

REFERENCES `mydb`.`swc

```



```

haracter` (`Character_ID`, `Subscriber_Email`)

```

```

ON DELETE NO ACTION

```

```

ON UPDATE NO ACTION)

```

```

ENGINE = InnoDB;

```

```

-- Table `mydb`.`Store`

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Store` (

```

```

`Store_ID` INT NOT NULL,
`Status` ENUM('PlayerUsing', 'NotInUse') NULL,
`Server_Server_Name` VARCHAR(50) NOT NULL,
PRIMARY KEY (`Store_ID`, `Server_Server_Name`),
INDEX `fk_Store_Server1_idx` (`Server_Server_Name` ASC),
CONSTRAINT `fk_Store_Server1`
    FOREIGN KEY (`Server_Server_Name`)
    REFERENCES `mydb`.`Server` (`Server_Name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

### **Kode of inputting av data.**

```

USE mydb;

INSERT INTO Subscriber VALUES('db@db.com'), ('SW@db.com'), ('AD@sql.com');

INSERT INTO server VALUES('Theograd', 'Online'), ('Durothan', 'Full');

SET FOREIGN_KEY_CHECKS=0;

INSERT INTO swcharacter VALUES(NULL, 'Debra', 'db@db.com', 'Theograd'),(NULL, 'Derek',
'SW@db.com', 'Theograd'),(NULL, 'Dory', 'AD@sql.com', 'Theograd'),(NULL, 'Dina', 'AD@dsq.com',
'Durothan');

INSERT INTO inventory VALUES
    (NULL, 'gun', 100, 17, 'db@db.com'),
    (NULL, 'cannon', 100, 18, 'db@db.com'),
    (NULL, 'companion', 100, 17, 'db@db.com'),
    (NULL, 'shield', 100, 17, 'db@db.com'),

```



```
(NULL,'lightsaber',100, 19, 'db@db.com'),  
(NULL,'agadean crystals',100, 19, 'db@db.com'),  
(NULL,'scrap',100, 19, 'db@db.com'),  
(NULL,'quest item',100, 20, 'db@db.com');
```

INSERT INTO credits VALUES

```
(6000, 17, 'db@db.com'),  
(7843, 18, 'SW@db.com'),  
(23300, 19, 'AD@sql.com'),  
(100000, 20, 'db@dsq1.com');
```

INSERT INTO quest VALUES

```
('Lore of Mandalore', 'Item', 'Completed',17, 'db@db.com'),  
('For the Sith', 'Credits', 'Completed', 18, 'SW@db.com'),  
('Black Sun Blackout', 'Other', 'Completed', 19, 'AD@sql.com'),  
('Force Ghosts', 'Item', 'Completed', 20, 'db@dsq1.com'),  
( 'New Haven', 'Item', 'Active',17, 'db@db.com'),  
( 'Sharu Nan', 'Other', 'Completed', 18, 'SW@db.com'),  
( 'Sure Fire', 'Item', 'Active', 19, 'AD@sql.com'),  
( 'Truth on Ordos', 'Credits', 'Completed', 20, 'db@dsq1.com');
```

INSERT INTO store VALUES(1,'NotInUse','Theograd'), (2,'NotInUse','Theograd'),  
(3,'NotInUse','Theograd'), (4,'NotInUse','Durothan');

SET FOREIGN\_KEY\_CHECKS=1;

Av en eller annen grunn har jeg fått feilen:

Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails  
(`mydb`.`swcharacter`, CONSTRAINT `fk\_Character\_ID\_Subscriber` FOREIGN KEY (`Subscriber\_Email`) REFERENCES `subscriber` (`Email`) ON DELETE NO ACTION ON UPDATE NO ACTION)

Ettersom Modellen har lagd koden og jeg ikke greyer å finne feilen så har jeg bare mått skru av Foreign Key Checking for å få lagt inn data. Dette tyder på en eller annen type feil i logikken min som jeg ikke finner. Jeg har også gjort feilen å gjøre Quest\_names til Primary keys som betyr jeg sliter med å ha flere instanser av en quest med samme navn. Burde heller hatt en Quest\_ID som kan bli referert til fra egen tabell. Et lignende problem oppsto siden jeg gjorde Item\_Identifier i Inventory Unik. Ideen her var faktisk og ha en egen tabell med de forskjellige objektene som kunne bli referert til men ettersom Identifier må være unik funker ikke dette med denne setupen.

Ideelt sett så ville Sever hatt en liste med Stores og Characters i seg, Character ville hatt et instans av Inventory og Credits til seg samt varierende instanser av oppdrag fra en liste, Quests, som den bare ville referert til å lagret statusen i forhold til spiller. I alle disse tilfellene så ville man trengt et mellomledd/tabell for å få dette til å funke