

```

/* ~~~~~
    user.schema.js
~~~~~ */

const {Schema, model} = require('mongoose');
const uniqueValidator = require('mongoose-unique-validator');

const STATUS = ['ACTIVE', 'INACTIVE'];
const UserSchema = new Schema({
  username: {
    type: Number,
    required: true,
    unique: true,
  },
  hash: {
    type: String,
    required: true,
  },
  admin: {
    type: Boolean,
    required: true,
  },
  cageId: Number,
  boxId: {
    type: Number,
    trim: true,
    sparse: true,
  },
  status: {
    type: String,
    enum: STATUS,
    default: STATUS[1],
  },
});

UserSchema.plugin(
  uniqueValidator,
  {messages: 'Error: expected {PATH} to be unique.'},
);
module.exports = model('User', UserSchema);

/* ~~~~~
    cage.schema.js
~~~~~ */

const mongoose = require('mongoose');

const uniqueValidator = require('mongoose-unique-validator');

const CageSchema = new mongoose.Schema({
  id: {
    type: Number,
    required: true,
    unique: true,
  },
  status: {
    type: String,
    enum: ['ACTIVE', 'INACTIVE', 'ERROR'],
    default: 'INACTIVE',
  },
  boxes: [{
    assignedUser: {
      type: mongoose.Types.ObjectId,
      ref: 'User',
      unique: true,
      default: null,
    },
    box: {
      type: Number,
      required: true,
      sparse: true,
    },
    _id: false,
  }],
});

```

```
    lastUsed: {  
      type: String,  
      default: 'N/A',  
    },  
    __v: false,  
  });  
  
CageSchema.plugin(uniqueValidator);  
module.exports = mongoose.model('Cage', CageSchema);
```