

# Database Functions API

## NetworkSIM/Database/dbModels

- Contains a separate file for each schema/model pair.
- models.js registers each schema for use in a node session

### **To create a new object representing a schema simply with**

```
var mongoose = require('mongoose');  
var User = require("../Database/dbModels/userModel.js");
```

In this example. User can be used to create a new object with User attributes.

```
ex: var aUser = new User();  
you can pass in another User object  
var aUser = new User(deviceJSON);  
or add each thing manually  
aUser.token = "14141441441"
```

.save() at the end of a model saves it in the database.

```
ex aUser.save();
```

*Currently if you add an attribute that does not belong in a User object, it simply will not save in the database but will not break the application.*

### **To use methods declared in a schema simply with**

```
var mongoose = require('mongoose');  
var Userm = mongoose.model("User");
```

In this example we will use the .modifyUser(token, user) function

```
Userm.modifyUser(123123, aUser);
```

finds user with a token and replaces them with a different users attributes

## **Current Functions**

### **User**

- .modifyUser(token, user) //finds user with token, replaces them with new user object
- addUser(aUser) //saves a user, same as .save() right now
- .save() saves the object

### **Network**

- getNetworkByName(name, callback) //returns network object with that name
- modifyNetworkByName(name, network) //finds a network with name and replaces it with new network object
- .save() saves the object

### **Simulation**

- modifySimByName(name, aSim) // same
- findallSimulations (callback) looks for sim and returns them all
- getSimByName(name , callback)

## **An Application Object**

application:

‘name’ : {name}

‘app\_spec’ : {spec object}

spec object is:

```
{  
  "name": "CounterApp",  
  "version": "0.1.0",  
  "description": "A small counter app for testing incrementing  
data type",  
  "main": "index.html",
```

```
    "rdt_list" : [ String]
}
```

## A RDT Object

rdt:

‘name’ : {name}

‘rdt\_spec’ : {spec object}

spec object is:

```
{
  "name": "incRDT",
  "version": "0.1.0",
  "description": "A replicated integer counter data type", "main":
  "rdtInterface.js"
}
```