

## Zeroconf

### Database

<u>services</u>	
PK	hostname (string: 20)
PK	serviceType (string:15)
	protocol (string:5)
	interface (string:10)
	port (int)
	TXTDATA (string:20)
	TTL (int)
	advertised (bool)
	questioned (bool)

<u>myServices</u>	
PK	hostname (string: 20)
PK	serviceType (string:15)
	protocol (string:5)
	interface (string:10)
	port (int)
	TXTDATA (string:20)
	TTL (int)
	advertised (bool)
	questioned (bool)

<u>requestedServices</u>	
PK	id (int)
	serviceType (string:15)

<u>DNSTable</u>	
PK	hostname (string: 20)
PK	interface (string:10)
	ip (string:40)

### Functions:

**service[] getServices()** :  
**service[] getMyServices()**: both return an array of structs service (which contains the fields described on previous tables) from the corresponding table.

**int addService(service)**: accepts a service and adds it to the table Services(it is important to check first if that service(aka hostname and service\_type) is already added. If so, it should do update and not insert).

**int addmyService(service):** accepts a service and adds it to the table myServices(it is important to check first if that service(aka hostname and service\_type) is already added. If so, it should do update and not insert).

**refreshTTL():** updates TTL times from both tables for all services ( $TTL = TTL - t$ , where t is a specific value)

**bool removeService(string hostname, string serviceType):** accepts a service's hostname and service\_type and removes it from the table Services (it should support the string ANY for both hostname and serviceType. More specifically, if a hostname is ANY, it should remove the services of ALL hostnames. The same for the service types.)

**bool removemyService(string serviceType):** accepts a service\_type from the table myServices (it should support the string ANY for serviceType. More specifically, if a service type is ANY, it should remove ALL services.)

**/\* int removeServiceByHost(hostname):** removes all services from the table Services corresponding to the specific hostname. \*/

**char\*\* getRequestedService(int &size):** returns an array of strings of containing the service\_type records. The argument returns the size of the array.

**int getServicesNum(void):** returns an integer equal to the number of records in myServices table  
**\*note\*:** application layer should add a corresponding record in myServices table for each service the device offers

**bool addDnsRecord(string hostname, string interface, string ip):** adds a record in DNSTable

**bool removeDnsRecord(string hostname, string interface):** removes a record from DNSTable

**bool updateDnsRecord(string hostname, string interface, string ip):** updates the ip of the record having hostname and interface given with the ip given