Quickstart

Create your first ajax function ¶

Create a file named `ajax.py` inside any of your apps. For example `example/ajax.py`.

Inside this file create a simple function that return json.:

```
from django.utils import simplejson

def sayhello(request):
    return simplejson.dumps({'message':'Hello World'})
```

Now you'll need to register this function as a dajaxice function using the `dajaxice_register` decorator:

```
from django.utils import simplejson
from dajaxice.decorators import dajaxice_register

@dajaxice_register
def sayhello(request):
    return simplejson.dumps({'message':'Hello World'})
```

Invoque it from your JS

You can invoque your ajax fuctions from javascript using:

```
onclick="Dajaxice.example.sayhello(my_js_callback);"
```

The function `my_js_callback` is your JS function that will use your example return data. For example alert the message:

```
function my_js_callback(data){
    alert(data.message);
}
```

That callback will alert the message `Hello World`.

How can I do a GET request instead of a POST one?

When you register your functions as ajax functions, you can choose th method using:

```
from django.utils import simplejson
from dajaxice.decorators import dajaxice_register

@dajaxice_register(method='GET')
def saybye(request):
    return simplejson.dumps({'message':'Bye!'})
```

This function will be executed doing a GET request and not a POST one.

Can I combine both?

Yes! You can register a function as many times as you want, for example:

```
from django.utils import simplejson
from dajaxice.decorators import dajaxice_register

@dajaxice_register(method='POST', name='user.update')
@dajaxice_register(method='GET', name='user.info')
def list_user(request):
    if request.method == 'POST':
        ...
    else:
        ...
```

In this case you’ll be able to call this two JS functions:

```
Dajaxice.user.info( callback );
Dajaxice.user.update( callback );
```

The first one will be a GET call and the second one a POST one.