Codeigniter project

We would like you to look at PHP, MySQL and optionally CodeIgniter framework and develop a simple contact database.

It needs a functionality to add, edit, deactivate and list records.

Sample fields for the database: Salutation, first name, middle name, last name, date of birth, address, city, postcode, tel, email.

It would be nice if all data entry/update points had a simple validation. For example, email has to be in a correct format, DOB in a date format etc.

You can do this in a standard PHP. If you would like to look at the framework we use, you can download Codeigniter 3.1.6 from here: [https://codeigniter.com](https://codeigniter.com/). There are some tutorials on youtube if you are interested.

For the front end, we use Foundation 6 framework and jQuery. However, you can code yours in plain CSS and HTML, although there will be extra points for user Foundation 6.

Security - go back to page 1 of installation and implement other security measures (including changing folder structure?)

Have put the basic Foundation stuff in a folder in the web route (Zurb). Probably not the best way but works for now – created a working link to the local foundation.min.css, so that gets me en route. (Later would be nice maybe to work out how you use SASS to combine app.css and foundation.css into the minified css file).

Some quick prettifying without too much reading and figuring out how their grid system etc works – copy and paste from <https://github.com/zurb/foundation-sites-template/index.html>

Why am I not having to register helpers etc in autoload.php? For e.g. base\_url() and form\_open are working regardless of not being registered. Similarly with models, everything ok when I ‘load’ model in a controller but just adding it to autoload.php doesn’t make it available…

<https://github.com/avenirer/CodeIgniter-MY_Model>

<https://github.com/avenirer/codeigniter-matches-cli> - create and run migrations from command line. No seeder though (**incorrect – you can do seeding in the migration**), so that limits the usefulness of migrations… NB don’t forget to address the security implication of using this: <http://avenir.ro/codeigniter-tutorials/step-2-set-environments/>. In fact this is not necessary in the case you are using this via CLI only. It is only an issue if you create a Migrate controller in order to run migrations via browser.

Pagination bug in CI? Error if number of rows modulo rows per page is zero? (Using the avenirer model might be a way of avoiding this problem). Basically, the problem boils down to the number of pages being calculated wrong in the above circumstance. For example, if there are 4 per page and 12 records, currently the pagination shows four pages. I *think* this was caused by record\_count returning wrong number of rows. Certainly so after I introduced soft deltes, but possibly before there was a malformed row?

Slug – I’ve abandoned use of slug (which was in some tutorial I found). The problem is it is awkward to generate a unique slug when you create a new record – you have to use the generated id not just the name or whatever (because these might not be unique). Possibly worth doing in a live system, but a bit of a faff for this.

**Accidentally overwriting all the records – FFS!** Because of that stupid thing of specifying ‘where’ before doing update, then I moved things around so there was no ‘where’ and it overwrote every record. Jesus!

To dos:

* JS modal for delete/deactivate confirmations?
* Filter/search?
* Order by/sort on table header (can potentially do this and the above with js datatables now I’m using a table and as long as the data doesn’t get too big – remember last time I used this there was that issue re the ajax version failing in some way. Actually that was wrt use of Backpack CRUD in Laravel iirc. Anyway, main point here is to let them know I’m aware of pros and cons. Ideally it should be done with ajax rather than a page refresh for ever re-order – that is v old school)
* Use before/after action methods in controller <https://github.com/avenirer/CodeIgniter-MY_Model>
* Check for existing contact with same name?
* Folder restructure for security (move application out of root)?
* Foundation asset compiling?
* Make form responsive – why isn’t it?
* Try MY\_Controller (avenirer) – not with Contacts but cities maybe
* Tests?

### Ordering index page

I can pass and retrieve a url param for ordering of results, but bit of a head scratcher as to how to construct the correct url in the index page, i.e. taking account of any pagination param or existing ordering param… (and how to do the routing).

Ah but the thing with this particular example (ordering) is *it doesn’t actually matter* because by definition if you change the order you always want to go back to the first page of pagination.

Still a problem though, because how do I capture the order\_by when it might be segment 1 or segment 2 of the URI? Answer, use a conditional and test first segment is numeric.

Next problem, the pagination links do not retain the order\_by param…

### Using Foundation SASS

For example, to implement motion-ui so I can make the confirmation messages closable…

### Postcode validation function

Shouldn’t be in the contacts controller, but how then to access it? Either make it into a library and load it in constructor, or inject it into constructor…

If I make it into a helper (put in application/helpers/ and nb must be named ‘…\_helper’) there is a problem with accessing the error message. I could return the message, but that isn’t right – a validation function should return t/f.

This is really tricky in fact – the current code signature is what is specified in the CI manual for user-defined validation. A lot is going on in the background. In the manual it describes how you can use a function from another model for validation (and how to then set a custom error message). So rather than a helper, I would make a postcode model… This isn’t right though because my validation rule is ‘valid uk postcode or area code’.

In short then, how *do* you move a validation function outside of you controller? I suppose in the current case it breaks down like this: (1) we could reasonably have a postcode model that validates a postcode and returns true/false (2) if the validation rule is ‘valid postcode or area code’ then that has become a specialised validation rule so it isn’t unreasonable for it to go in the Contacts model (and the postcode model could also include a method for validating area code)

### Edit/update validation

The edit route directs to controller/edit/id. This method just displays the form – and should run the validation but for some reason doesn’t. The form action is update, so that directs to contacts/update/id. Now the weird thing is that the update method doesn’t take id param. It just runs update\_contact which then gets the id from the post vars.

The delete method of the model (for e.g.) has to take an id because there is no intermediary equivalent to the edit method i.e. the edit/update process has two stages – show the form, do the update, whereas with delete (and deactivate) it’s just one action. The edit method needs to take an id in order to be able to populate the edit form. The update method doesn’t need an id because the id comes from the form via a hidden input.

The difference between edit/update and create/store is that I don’t have a store method in the controller. The store\_contact method is called direct from the controller’s create method, but I don’t see why this would make any difference.

For some reason it must be that form\_validation->run is returning true, but why why why??

Well I finally fixed this by getting rid of separate update route and method. NB currently have the form action as /edit/id but still require hidden field with id – why?

### Pagination woe

I *finally* figured this out. Why was it so hard? (1) lack of decent up-to-date examples available online (2) inadequate documentation (3) the way it is done is obscure and outdated – for e.g. you need a ‘get items’ method that takes start index and items per page as params i.e. has to take a particular form if you are using pagination compared to not using pagination. Also, your index view must use array if using a ‘normal’ get items method, but object if using pagination – **incorrect** – this was just because the original get\_contacts method used $query->result\_array(). And, the ‘links’ object returned to the view is just a string, so any styling has to be done in the controller ffs.

Presentation problem – pagination being displayed vertically, because one of the links is not getting surrounded by li tag. But why??? In the end it was because I had ‘num\_tag\_open’ twice, and no ‘num\_tag\_close’. This took me well over an hour to work out.

In the documentation it says you can set uri\_segment as a pagination config, but given last thing above (that your get items method requires start index) this can’t work.

I wonder, is the get items method I’ve been using outdated? No don’t think so – got it from the CI docs.

Working this out was a horrible experience and very reminiscent of my early Zend days.

## Things I don’t like about Codeigniter:

* No migrations – actually, it appears there *is* a migrations class…
* No equivalent of Artisan
* No built in ORM
* Weird way that query clauses are defined using the db object before any query is executed
* In the tutorial at least, using get\_objects method to return all or objects or just one, depending on presence of ‘slug’ param (and why did they choose to use ‘news’ as an example which doesn’t give you any clues about conventions for singular/plural naming of controllers, models etc?)
* The routing!
* Lack of a ‘public’ folder – just unfamiliar
* Problem with htaccess file in root
* Have to manually create created\_at etc. (?)
* Lack of useful material on the internet
* Lack of good third party packages
* Documentation – for e.g. they use some weird char for single quotes so you can’t directly copy and paste
* The pagination! What a performance.
* Having to use parent::\_\_construct (apparently)
* Inadequate error trace – impossible to know where errors are coming from. Seems like when I put a try catch in controller it generates an error that comes from the CI index.php where it tries to load the bootstrap file???