## Introduction

This is mainly a house-keeping release. I’ve added menu-bar functionality, as promised a few posts ago. This gives a context-sensitive menu that shows different options depending on the logged-in user. I’ve also carried out some refactoring, mainly of views which were starting to proliferate!

## Menu Bar

I’ve created a MenuBar class (lib/menubar.php) that implements a CSS-friendly menu. Essentially, it’s just an html unordered list with appropriate CSS classes for selection. I’ve included some standard logic for this in the base view class. There’s still a bit of a code “smell” in doing this because it doesn’t really belong there. (A discussion point – Why not?). Nevertheless, it’s enough to let me get rid of the html links on the home page and generally clean up the UI. At the moment, I’ve just two sets of menus: one public, and one for a logged-in administrator. This will grow!

## Refactoring Views

Up to this point, I’ve been more or less creating one view per controller. That’s setting a bad example because views should be as generic as possible and reusable across controllers. What I’ve done in this post is to replace most of them with a generic View class (lib/view.php) and done the corresponding tidying in the controllers. There are now just three custom views (people, static and tests) and these too are destined for the scrap heap in the not too distant future. Before I do that, though, I’ll add some more generic views.

One of the key ideas in views is the use of templates for the html. I’ve now added two levels of templates. At the overall level, there is a page master template. The view class uses this as the basis for each page and adds content to it. At the second (sub-view) level, another html template is used for the variable content of the page. That way, we keep the html and CSS cleanly separated from the PHP code and the UI designers can work with the CSS and HTML code, while the application designers can focus on creating a robust application.

## Summary

All-in-all, there are no major new concepts in this version, just a general tidy-up and cleaner UI. Regular refactoring is a normal part of agile development, so a lot of this is just *business as usual*. With this release, I hope you’ll get a sense of how the whole system will come together.