easyJet

Roster Engine Ideas

Front-End:

* ReactJS or AngularJS
* Dashboard Design (similar frame to BBC or BA)
* Drag and Drop Interface
* Organise weekly rosters (ability to scroll back and forth between weeks/dates)
* Assumption of laptop/desktop for main usage. Subject to verification.
* Need to consider option to upload a spreadsheet with current data (worthwhile?) or have a basic user portal for employees to enter in their availability/preferences.
* Real-time interaction (drag and drop direct interaction with server) or generic submission after all changes are made. - Dependent on current systems/browser

Back-end:

* Node/Express or Ruby/Sinatra. Perhaps too small (focus wise) to be a good fit for a Rails app.
* Constraint Algorithm to provide automated roster generation/recommendation. Can be done in any language, but there seems to be quite a lot of open source work in Python, so likely to go with this.
* Other logic built on top of constraint algorithm to allow easy manual changes.
* Ties together front-end submissions with user and roster data stored in database.

Database:

* Basic employee information (ID, name, role, availability, preferences)
* MongoDB makes sense. Fairly standard information for each employee. Little in the way of relational data (interactions of employee data is handles by application logic).
* Roster information that is generated by Back-end application logic. Perhaps this can persist a set time length of of rosters (e.g. 2 year history).