# Ruby on Rails (Coding style in Solium)

* Keep controller methods to a minimum (responsible for controlling the request) Achieved in the following ways:
  + Abstract logic to the model/service layer
  + Abstract UI specific needs into Decorators in order to enhance the models without leaking UI specific needs into the model
  + Abstract life-cycle/cross-cutting concerns into observers which can be very useful when you want to perform before and/or after processing on certain entities (Aware that this is being deprecated)
* Testing ideas
  + Use factories in order to decrease the boiler plate of generating objects (factoryGirl)
  + Use populate to fill your db with all required setup data e.g. account type entries
  + Use rspec (similar to Spock in Groovy)
* How do I create a clean UI
  + Use asset pipeline to ensure that your assets are getting compile, minified in a standardized format
  + Use layouts to store the sites layout in a modular manner
  + Use helpers to store UI specific utility methods promote reuse and testability
  + Ensure to use views and partials appropriately
* How do I deploy/maintain my application
  + Use migrations to handle db migration and rollback strategy
  + Use rake tasks to run once off tasks (not limited to)
  + Use monitoring tool to track errors and analyze performance issues