

# Web Development

---

Produced  
by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics  
Waterford Institute of Technology

<http://www.wit.ie>

<http://elearning.wit.ie>



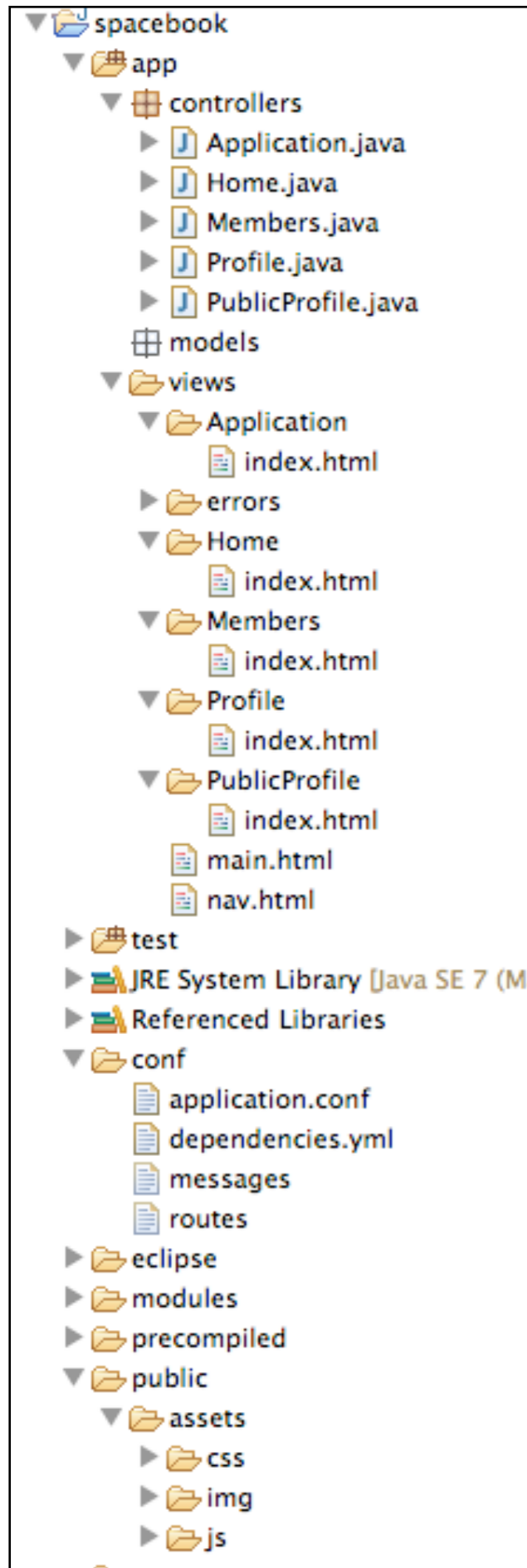
Waterford Institute of Technology  
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRCE



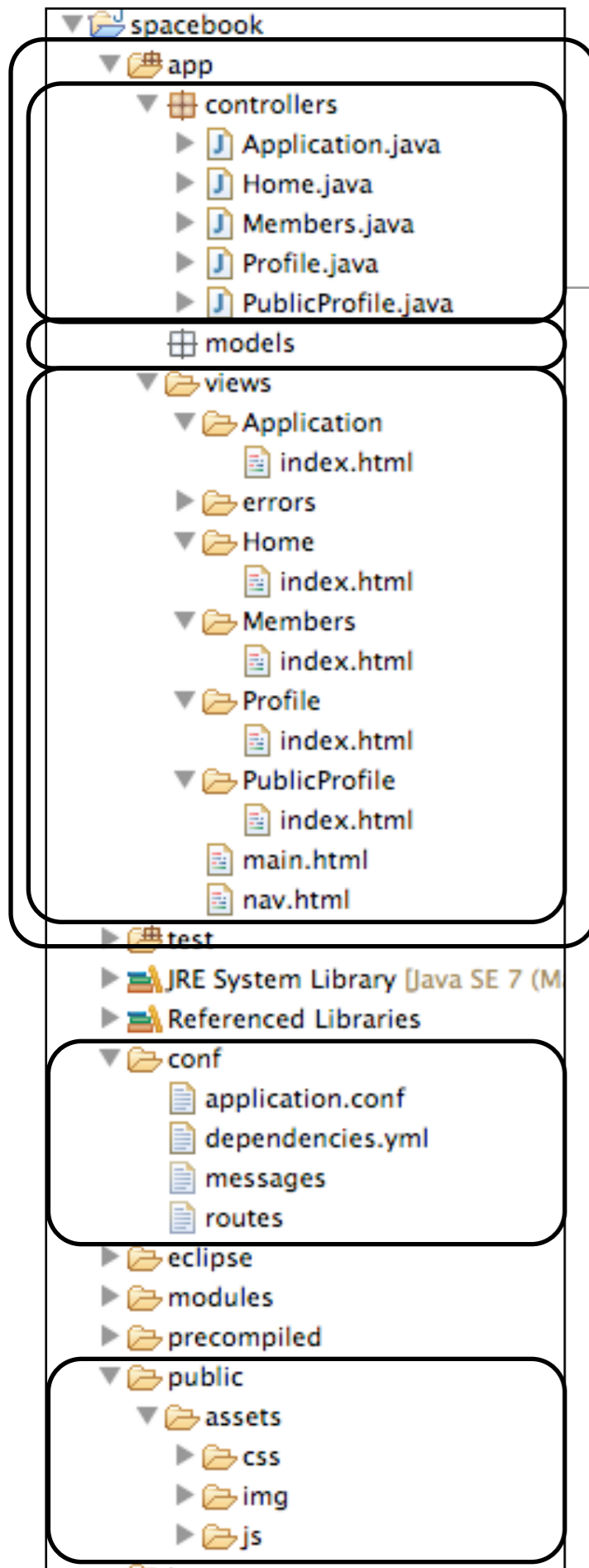
# Play Project Structure

---

# Play Project Structure



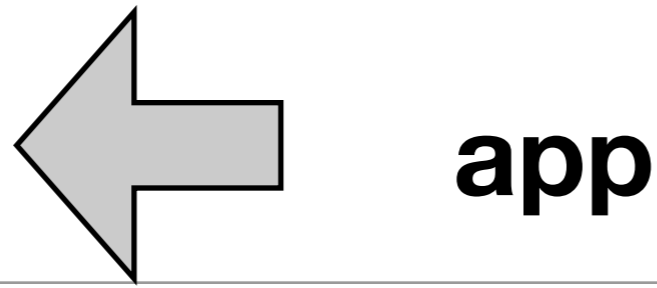
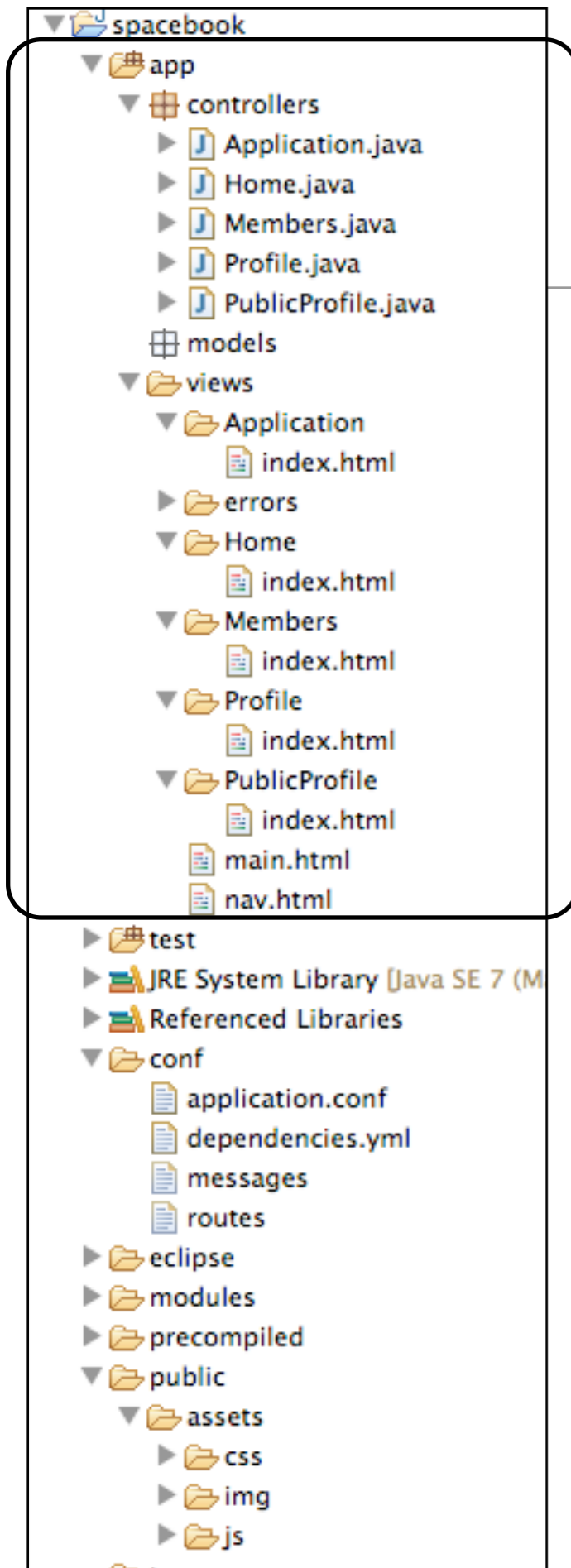
- All Play projects have a very specific structure.
- This structure ensures that there is a consistent *separation of concerns*
- This ensures a clear distinction in the project structure between views, models and controllers.
- Views are expressed in HTML + a simple 'template' language
- Controllers and Models are in standard Java
- Additionally, there are small fixed number of configuration and 'yaml' files



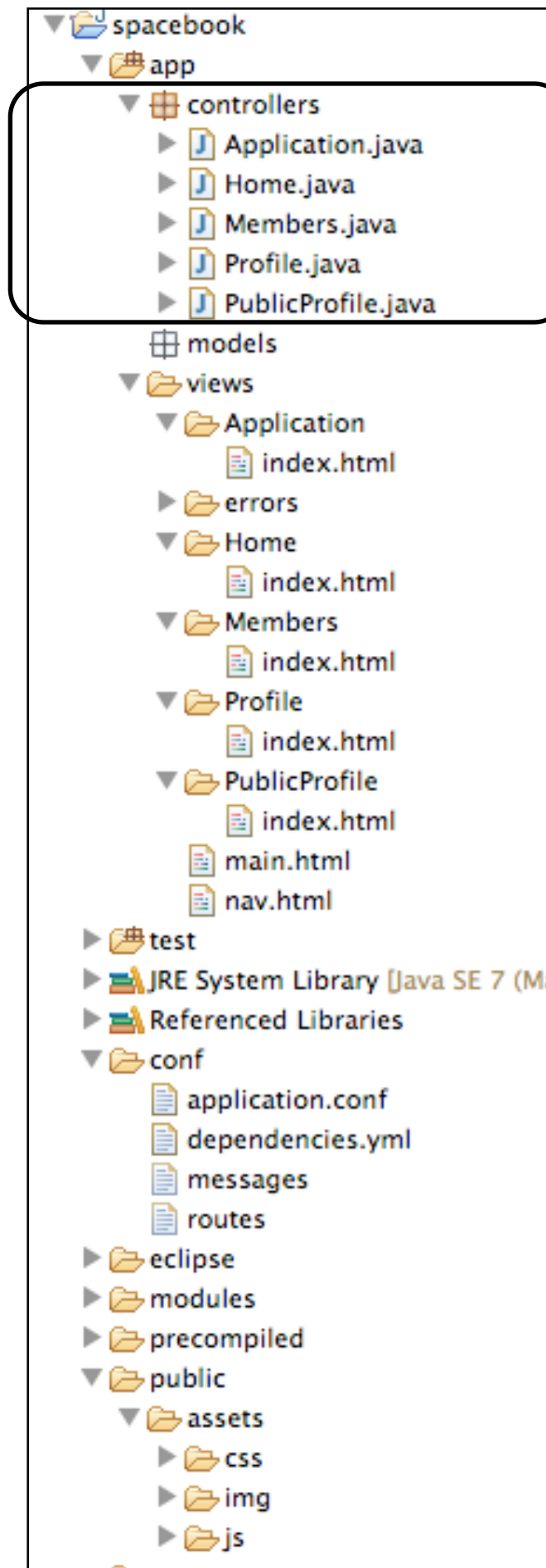
- app
  - controllers
  - models
  - views

## 5 Key Sections

- conf
- public

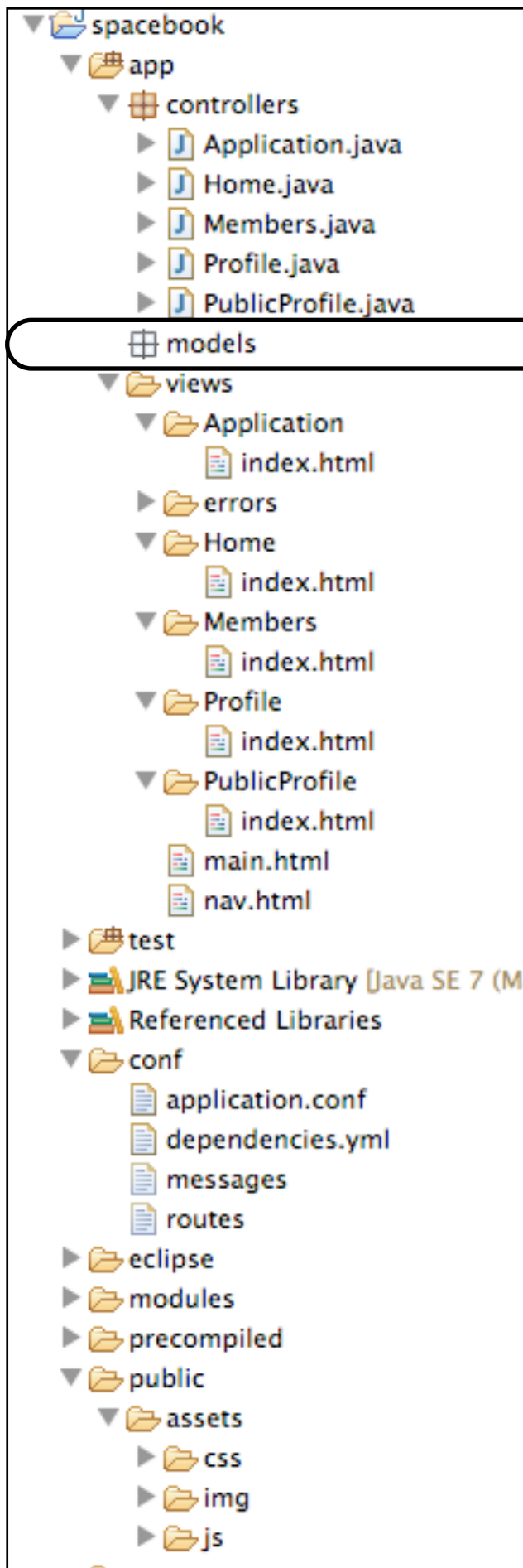


- Application ‘sources’, where most of the development effort takes place
- In a folder called ‘app’
- In turn divided into three folders:
  - controllers
  - models
  - views



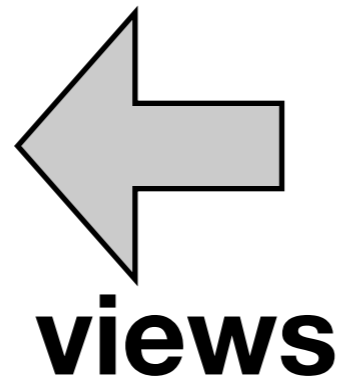
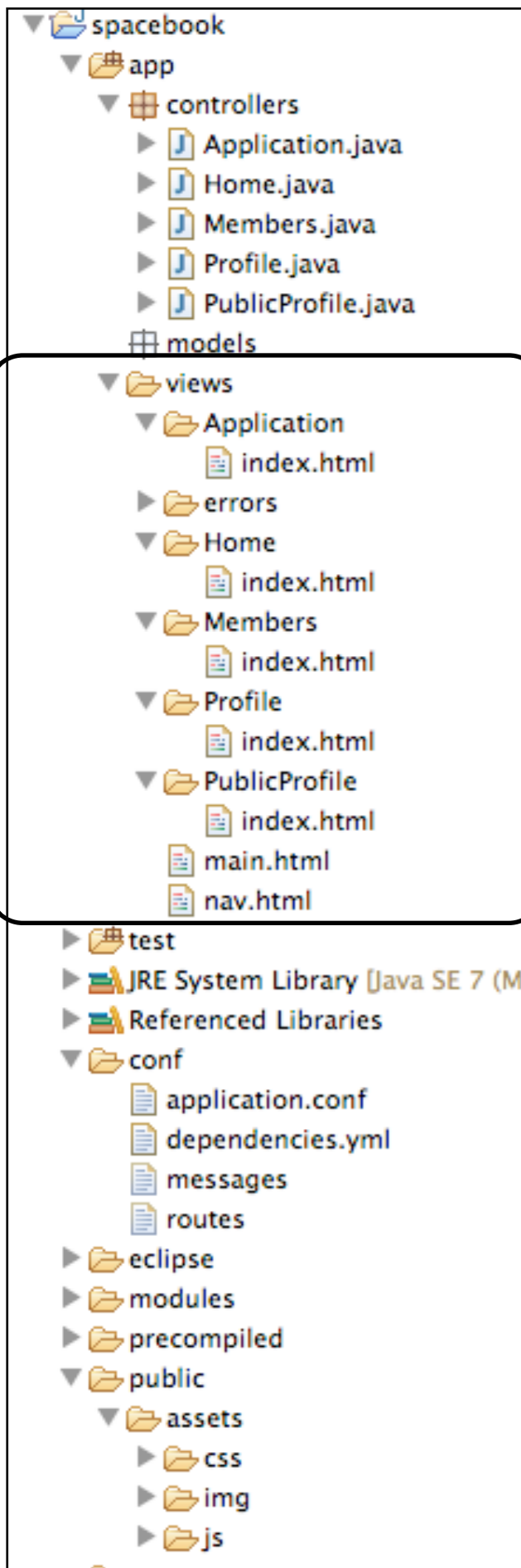
# controllers

- Controllers accept all HTTP ‘requests’
- Requests are initiated by user:
  - placing url in browser address bar
  - clicking on a link
  - submitting a form
- Requests result in methods in controllers being called
- These methods are called “Actions”



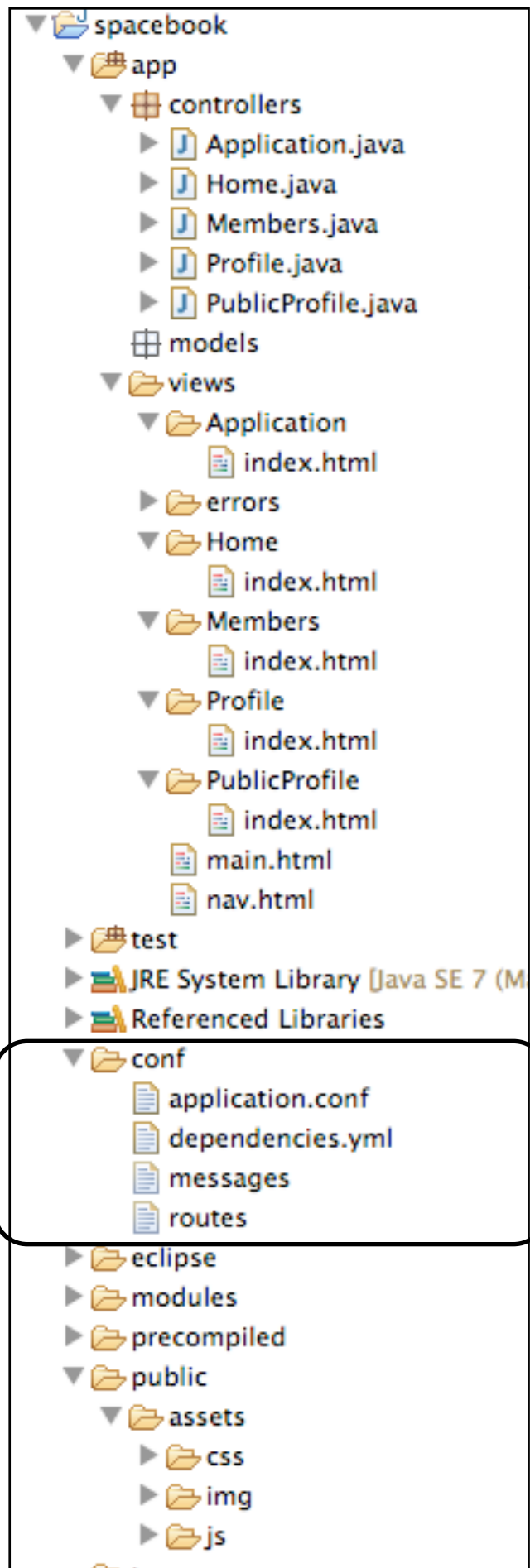
## models

- Any information that must be 'remembered' will be part of the model
- For spacebook, this might include user details, names, passwords, messages, profile status, images etc...
- Currently empty - but will be introduced shortly as Model classes.
- Instances of these classes will be saved to a MySQL database



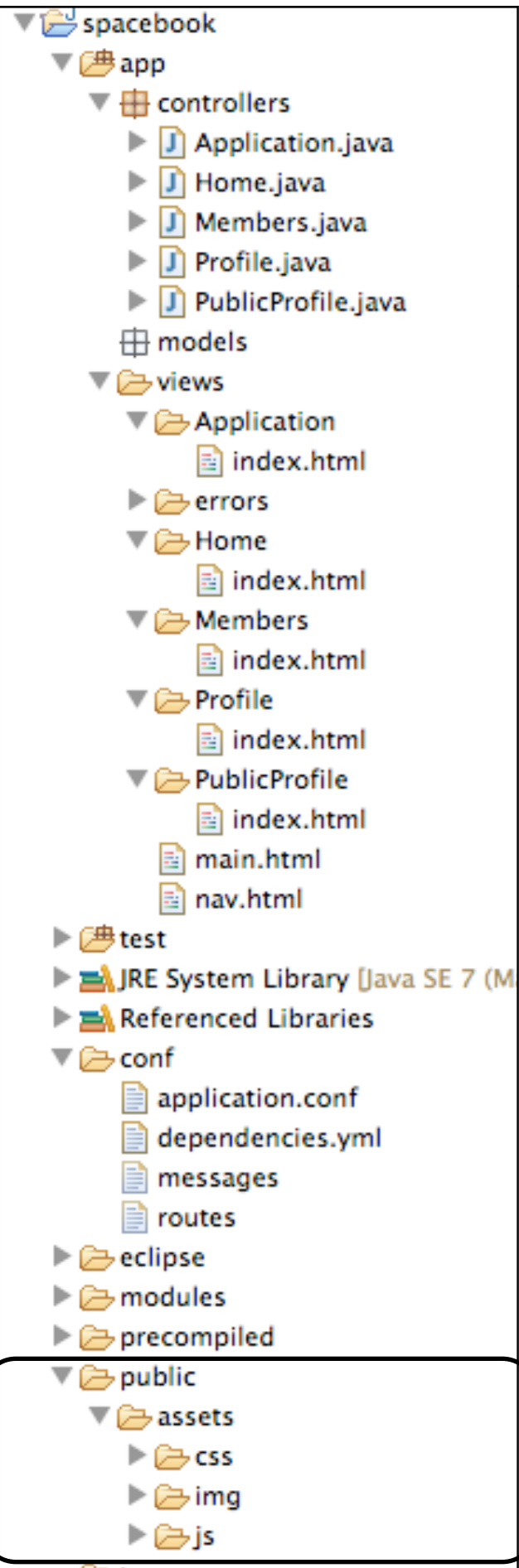
- Fragments of HTML pages
- Usually modeled on a base template - main.html in this project
- main.html defines a top level structure for a complete page
- Individual aspects of the main.html template are replaced as needed to deliver a complete 'view'
- Page fragments contain HTML + a simple templating language



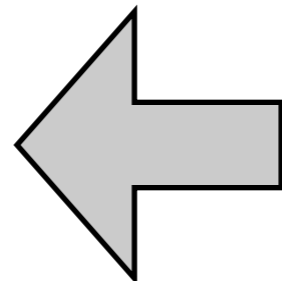


**conf**

- Configuration of the application
- Simple files (usually)
- *routes* - application routing table
- *application.conf* - global settings
- *messages* - string for internationalization
- *data.yml* - initial test data



**public**



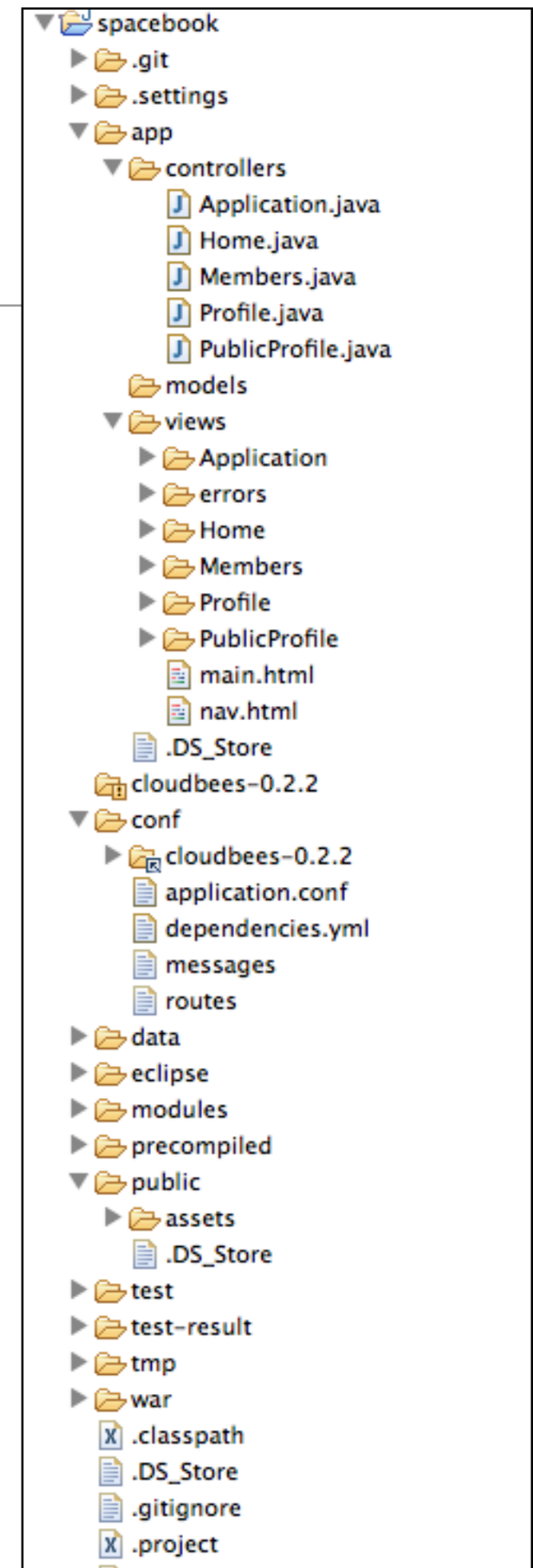
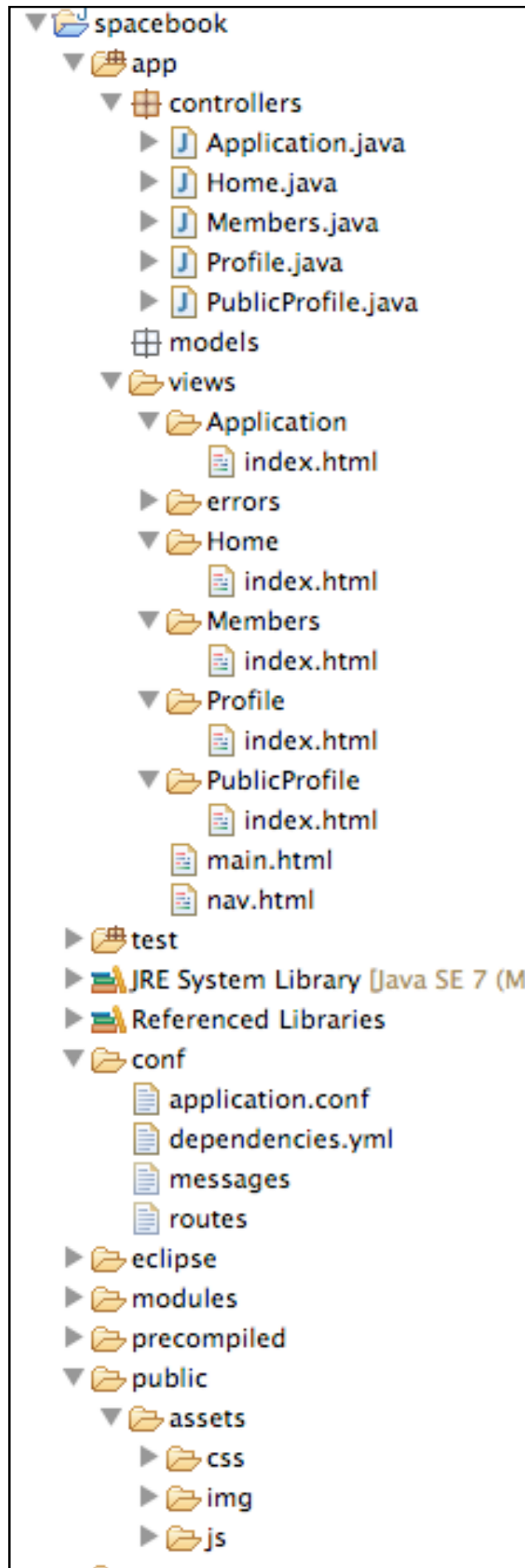
- ‘Static’ files that are never expected to change
- Usually css + javascript
- May include static images - favicons and other stock (unchanging) images
- Not involved in ‘rendering’ process - no templates here

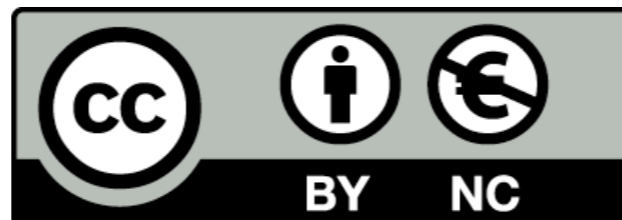
# Logical - Package Explorer

- *Programmer centric view.  
Emphasizes context  
aware editors suited to  
different languages  
syntaxes*

# Physical - Navigator

- *Complete list of all files in  
the project folder.  
Majority of these artifacts  
are 'generated' by build  
process and are never  
managed by the  
programmer*





Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>

