

## 9. CodeScope

We ask you to build an application analyzing packages, classes and methods and producing reports using Seaside. Use the same image as you used for the previous Seaside exercises.

Think about a suitable user interface. It should be possible to select a package from all existing packages, select a class from all existing classes of the selected package, and finally select a method from the class. For all the selected items, we want to gather information that should be displayed.

### Exercise 9.1

Implement the User Interface using Seaside. Try to make it fast and fun to use. Feel free to reuse code from examples of the seaside distribution!

### Exercise 9.2

For the selected package, display the number of classes.

### Exercise 8.3

For the selected class, display the number of methods and class methods.

### Exercise 9.4

For the selected method, display the number of lines of code.

### Hints

We suggest to use the `PackageInfo` packages (not simple class categories) for this exercise. The central class is `PackageInfo`. Packages are described using instances of this class.

**Q:** How do I get a list of all packages?

**A:** use `PackageInfo allPackages`

**Q:** How do I get a list of all classes of a package?

**A:** have a look at `#classes`

**Q:** How do I get a list of all methods of a class?

**A:** you can get the method dictionary of a class by sending `methodDict`

**Q:** How do I calculate the number of lines of code?

**A:** Have a look at class `CompiledMethod`, category *source code management*

**Please save the Monticello package `CodeScope` and send it by mail to [st-staff@iam.unibe.ch](mailto:st-staff@iam.unibe.ch). Attach your written solutions that are not part of the source-code to the mail or hand them in as hardcopy at the beginning of the next exercise session. Your mail and solutions should be clearly marked with names and matrikel numbers of the solution authors.**