Population dynamics Group 07

Exam INF200 - June 2020
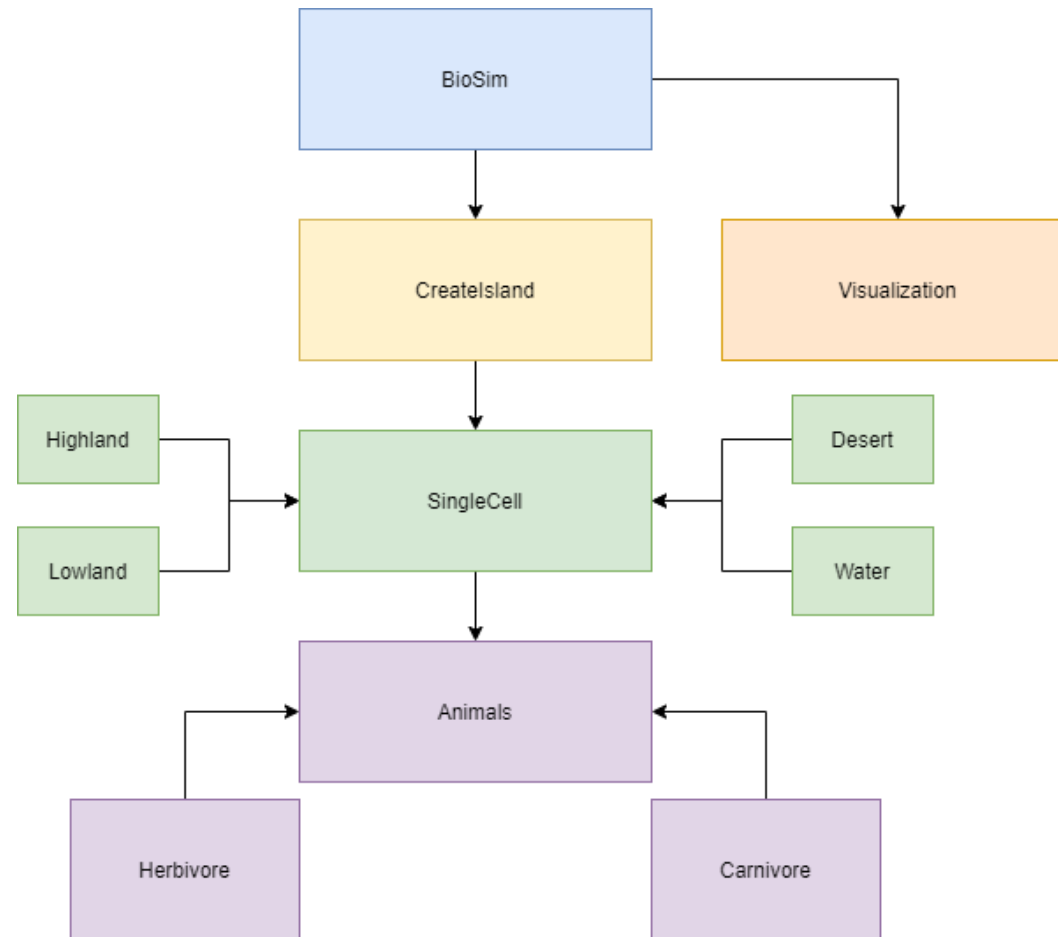
Astrid Hæve Sedal

Mikal Breiteig
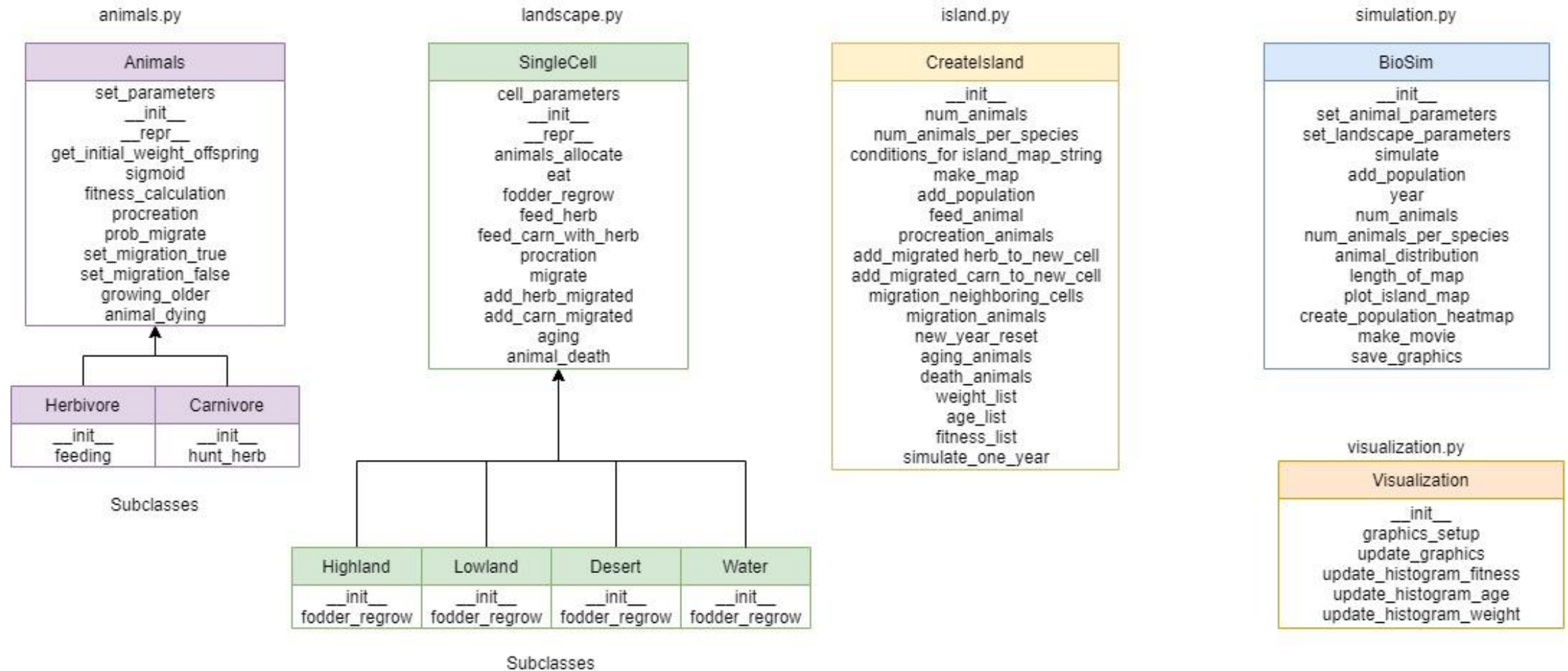
Our approach

- Planning prosess and time management

  – Flowchart and overall structure of project

- Created the framework

  – Files, classes and methods

  – Simple tests based on requirements

- Code and further tests

- Debugging, documentation and prioritizing

Overview of classes

Detailed overview of file and structure

The code quality

- Tests based on task requirements

- PEP8

- Readable code using clearifying methods and variable names

- Simplified code

- Documentation
    - NumPy

- Unittests

- Mocker

- Statistical test

Simplify and increase of usability

```python
      def add_migrated_herb_to_new_cell(self, new_loc, herbivore):
          """
          Adds migrated herbivore to new cell by position.

          Parameters
          ----------
          new_loc : tuple
          herbivore : object
          """

          self.map[new_loc].add_herb_migrated(herbivore)

      def add_migrated_carn_to_new_cell(self, new_loc, carnivore):
          """
          Adds migrated herbivore to new cell by position.

          Parameters
          ----------
          new_loc : tuple
          carnivore : object
          """

          self.map[new_loc].add_carn_migrated(carnivore)
```

```python
  def migration_animals(self):
      """

      Checks each cell in the map if it is accessible for the animals, and if the condition is
      met it finds the cells' neighbors. It then calls migrate method from SingleCell and
      returns the herbivores and carnivores, if any, that has migrated to a new cell.
      In migrate method in SingleCell the migrated animal gets deleted from that cell.
      The migrated animal gets added to the new cell that was selected randomly.


      Methods
      -------
      SingleCell.migrate()
      """


      for loc, cell in self.map.items():
          if cell.accessibility is True:
              neighboring_cells = self.migration_neighboring_cells(loc)
              migrated_herb, migrated_carn = cell.migrate(neighboring_cells)

              for new_loc, herb in migrated_herb:
                  self.add_migrated_herb_to_new_cell(new_loc, herb)

              for new_loc, carn in migrated_carn:
                  self.add_migrated_carn_to_new_cell(new_loc, carn)
```

Simulation over 390 years
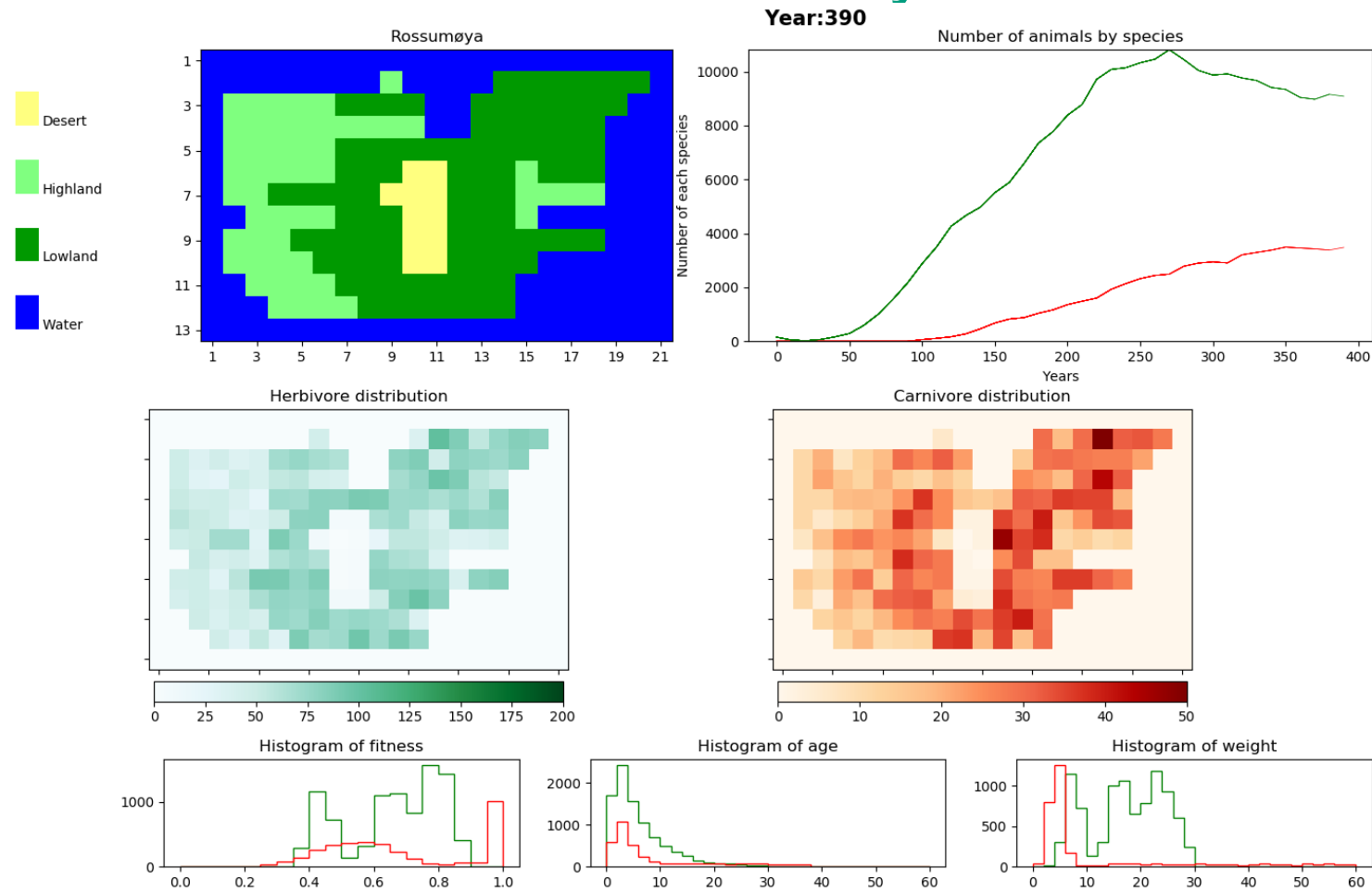
Improvements of the project

- Locate the bug and achieve more accurate graphs.

- Increase test coverage.

- Simplify code and decrease the runtime.

- Overview of costly computation and decrease it.

- Retrieve and store data to gather more detailed information about the ecosystem on Rossumøya.

Play movie

Thank you for your attention!