University of Puerto Rico
Mayagüez Campus
Department of Computer Science and Engineering
Department of Software Engineering

Homework 2

Luis F Caro Monroig
Carlos J Figueroa Vázquez
Antonio G Lugo Alicea
John A Robles Pérez
Alejandro Ruiz Almodóvar

INSO4101
Introduction to Software Engineerings
2020/03/06
Prof. M. Schütz Schmuck

Refinement

Domain Entities:

- Atomic Entities
    - Water quality report: report of the water quality for the user's specified beach.

- Composite Entities
    - Beaches: Beaches who receive regular water quality reports from the Department of Natural Resources and the Blue Water Task Force.
        - Name: Full name of the beach
        - Image: Image of the beach that most accurately highlights the beach’s most defining qualities.
        - Location: GPS Location of the beach on a map (Satellite view).
        - Water Quality: Description of the beach’s most recent water quality parameters.
        - Date of report: Date of most recent publication of water quality report.

Domain Functions:

- Search for a beach
    - search: List x Position on the Screen = Beach name x List
- Look at Image of the Beach
    - look: Image x Position on the Screen = Image x Beach name
- Look at Location of the Beach
    - look: GPS location x Position on the Screen = GPS location x Beach name
- Read "About" section
    - read: Information origin x Position on the Screen = Information origin x Beach Name

Function Definition:

- The *search* function:

  - When applied to:
    - Beaches
  - Yields:
    - The specified beach
- The *look* function:
  - When applied to:
    - Image
    - GPS location
  - Yields:
    - Image of the beach
    - GPS map of Puerto Rico detailing the location of the specified beach.
- The *read* function:
  - When applied to:
    - About section
  - Yields:
    - Information regarding the sources of water quality reports used in this application, as well as the names of the developers who worked on the application.

Domain Events:

- Internal Events:
  - User finds beach of interest: event that marks the completion of searching for a beach of interest.
  - User finds the beach's location: event that marks the end of the user's beach search activity and review of the water quality report. The user is then shown the beach's location by using the GPS map.
  - User inspects sources for water quality reports: event that marks the completion of the user verifying the "about" section.
- External Events:
  - New water quality report is published by organizations for any given beach: event that marks the publishing of a new water quality report for any given beach.

Domain Behaviours:

- End users will glance over beaches and will search for the water quality of the beach that will be visited.
- End users can find the location of the beach by using the GPS map.
- Typing-in-search: user decides what to type.
- Search: user verifies the various suggestions are relevant items presented and chooses the one that best fits their criteria or needs.
- Prompt-location: User gets a notification about location accessibility, user clicks yes and gets a map showing their current location in regards to the location of the beach of interest where they'll head to.

Algebra

- Given the shallow nature of this project, we use an abstract algebra to attempt to define a "Search Bar" functionality.
- We have a set Beach = $\{b_1, b_2, \ldots, b_n\}$, for $n \subset$ natural numbers, which represents our beaches. $\forall\ b_n \subset$ **Beach**, b_n will be an object with properties *location* and *isContaminated*. Location will be a link to a pin of the beach location, if available. The property *isContaminated* will have values of "clean" or "contaminated", making it a set of boolean values. If a string is searched and yields no results, that string will not be a part of the set **Beach**.
- We can apply some operations to a **Beach** object, primarily *relevantToSearch*, *getCleanBeaches*, and *searchBeach*.
    - The operation *relevantToSearch* will compare the input string as it is updated against a list of strings that contains popular or often searched beaches and return a list of suggested strings to complete the search.
    - The operation *getCleanBeaches* checks the information sources for all beaches that are identified as clean and returns them in a list.
    - The operation *searchBeach* compares a string obtained from user input or the suggested list and searches the information sources for a string that matches it, taken to be the beach searched. Then it returns quality report data and shows it to the user.
- The operations and properties of the **Beach** set will be obtained from **DataGatherer**. This function will check public and permitted private records for beach quality data and return it to our project for further classification and use. **DataGatherer** will extract information as elements, specifically tagging location and origin page.
- Types:
    - Search

  - Suggestions
  - Location
  - Beach
  - Data
  - Contaminated
  - Page
  - Elements

- Function signatures
  - Search()
  - Predict()
  - FetchData()
  - ShowData()

Behaviors

- Typing-in-search: user decides what to type.
- Search: user verifies the various suggestions are relevant items presented and chooses the one that best fits their criteria or needs.
- Prompt-location: User gets a notification about location accessibility, user clicks yes and gets a map showing their current location in regards to the location of the beach of interest where they'll head to.

Function Signatures

- searchBeach() -> As a parameter we have a string type of the beach name, which returns the name of the desired beach if it exists, otherwise, return a message to search for another beach.
- predictBeach() -> This function is applied whenever the user modifies their search string. It takes the entire search string as a parameter and returns a list of possible input completions.

- fetchData() -> As a parameter we have a string type of the beach name, this string is then put into the query for the web scraping scheme, and returns the data of the water quality report retrieved from the DRNA or the Blue Water Task Force's web service.
- showData() -> As a parameter we have a string type of the beach name, which returns the water quality report of the selected beach.

Predicates

```
Scheme Report
    Class
        Type Search, Suggestions, Location, Beach, Data, Contaminated
        Value
            allBeaches: Beach □ Object { },
            relevantToSearch: Suggestions □ set-string,
            getCleanBeaches: Data x Beach → Object { }
            getLocation: Location -> list-latitude-longitude
            searchBeach: Beach x Suggestions → set-string
            isContaminated: Contaminated → Bool
        axiom
            ∀ beaches: Beach, data: Data
                Let availableBeach = searchBeach(
                    relevantToSearch(
                        allBeaches( ),
                        getLocation( ),
                            ) ∃ beach { c | c: Contaminated ·
                        isContaminated(c), getCleanBeaches }
                    )
                end
            end
```

For this scheme, we have the following types: Search, Suggestions, Location, Beach, Data, and Contaminated. The values serve to provide information about a beach that a user can act on or inform their decisions with.

```
Scheme DataGatherer
    Class
        Type Search, Location, Page, Elements
        Value
            getPage: Page x Search →Object,
            getLocation: →String,
            getElements: →Object,
            Extract: Elements →set-objects
        axiom
            ∃ beaches: Beach, data: Data
                Let availableBeach = searchBeach(
                    getPage(
                        getLocation( ),
                        getElements( ),
                            ) ∃ element { e | e: id · Extract(e) }
                    )
                end
            end
```

For this scheme, we have the following types: Search, Location, Page, Elements. The values of this scheme provide information regarding specific web pages, their elements, and the location of the specified beach. This function will scrape public and permitted private records for beach quality data, extract the wanted data, and return it to our project for further classification and use.