More Sophisticated Behaviour

Technical Support System **V2.0**

Produced by:

Ms. Mairéad Meagher
Dr. Siobhán Drohan
Ms. Siobhán Roche

SE TU
Ollscoil Teicneolaíochta an Oirdheiscirt
South East Technological University

Department of Computing and Mathematics
http://www.setu.ie/

Topic List

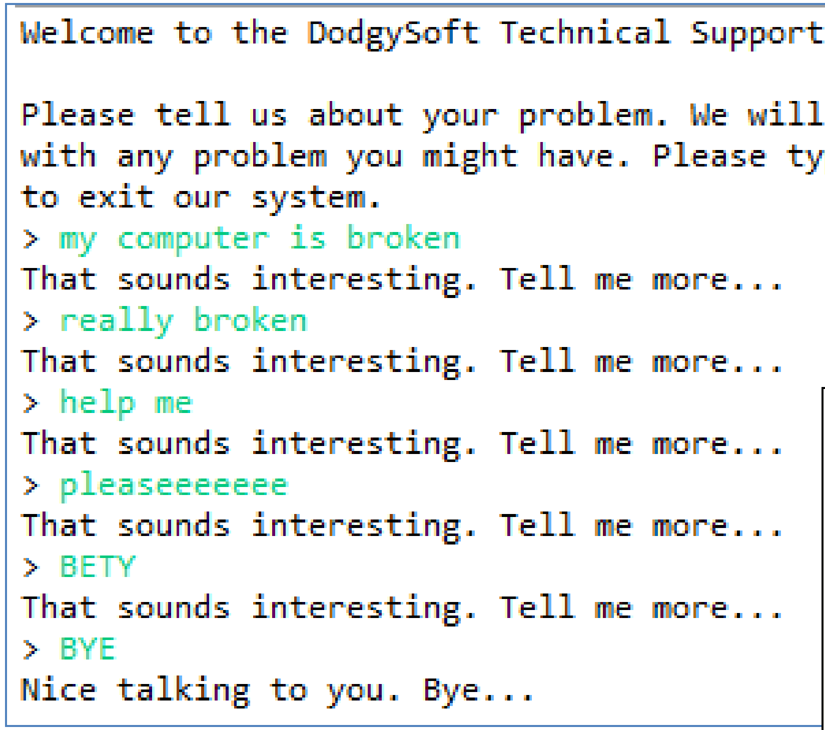
1. Recap: Technical Support System V1

2. Technical Support System V2
   - Overview of the System
   - Responder class
     - Random

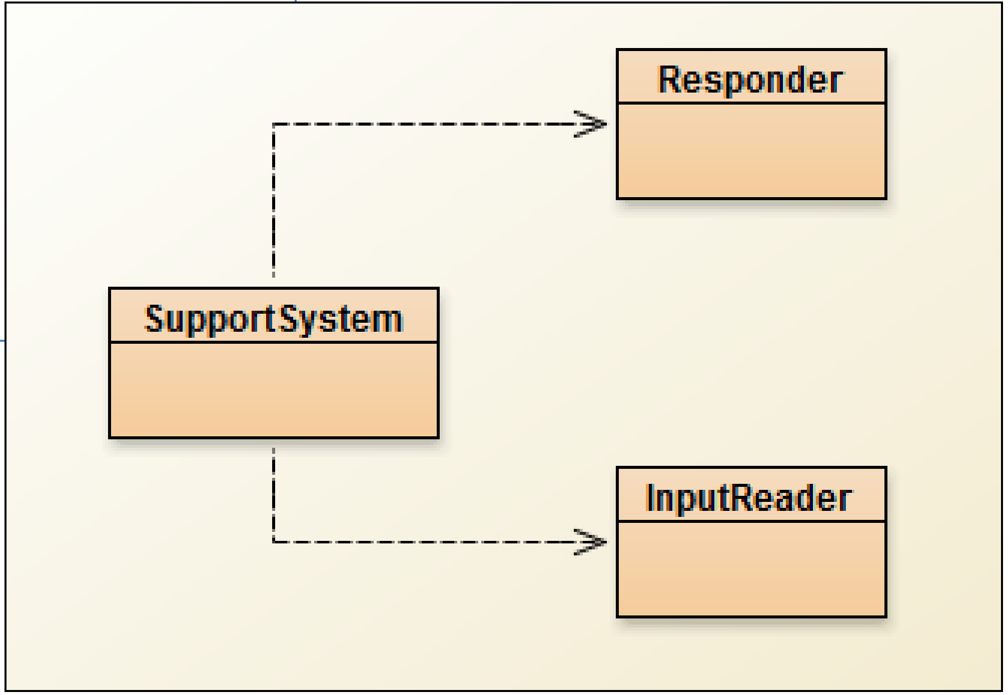
Technical Support System V1

- A console based system.
- A textual dialog system i.e. you enter text on the console and the system will provide a response.
- In this version, the system responds with the same String always:
  - “That sounds interesting. Tell me more…”

Technical Support System V1



```
Welcome to the DodgySoft Technical Support System.

Please tell us about your problem. We will assist you
with any problem you might have. Please type 'bye'
to exit our system.
> my computer is broken
That sounds interesting. Tell me more...
> really broken
That sounds interesting. Tell me more...
> help me
That sounds interesting. Tell me more...
> pleaseeeeeee
That sounds interesting. Tell me more...
> BETY
That sounds interesting. Tell me more...
> BYE
Nice talking to you. Bye...
```



Topic List

1. Recap: Technical Support System V1

2. Technical Support System V2
   - Overview of the System
   - Responder class
     - Random

Technical Support System **V2**

- A console based, textual dialog system.

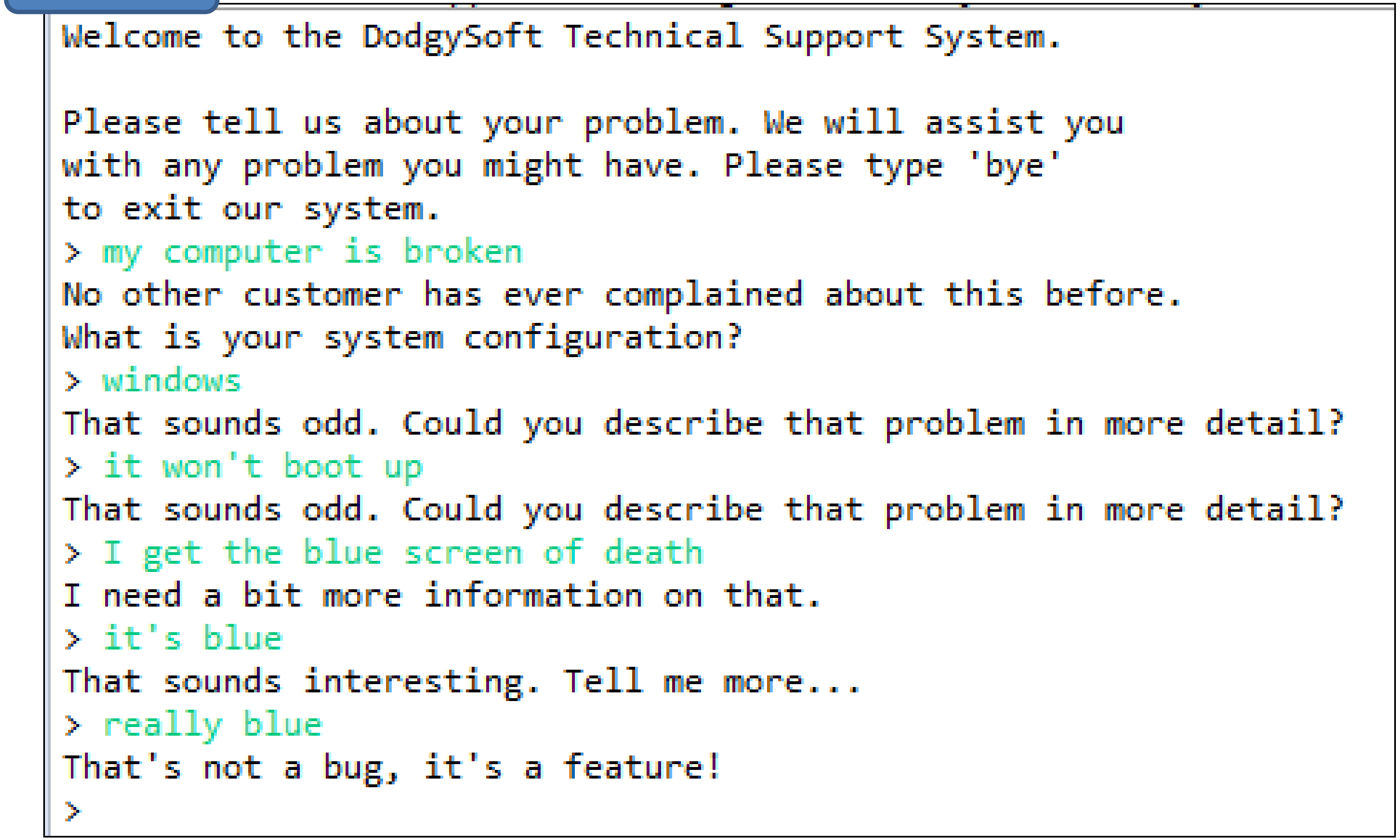
- The system provides a **random** response from a list of pre-defined responses

  - "That sounds interesting. Tell me more..."
  - "I need a bit more information on that."
  - "Have you checked that you do not have a dll conflict?"
  - "That is explained in the manual. Have you read the manual?“
  - "That's not a bug, it's a feature!"
  - "Could you elaborate on that?“
  - etc.

repository

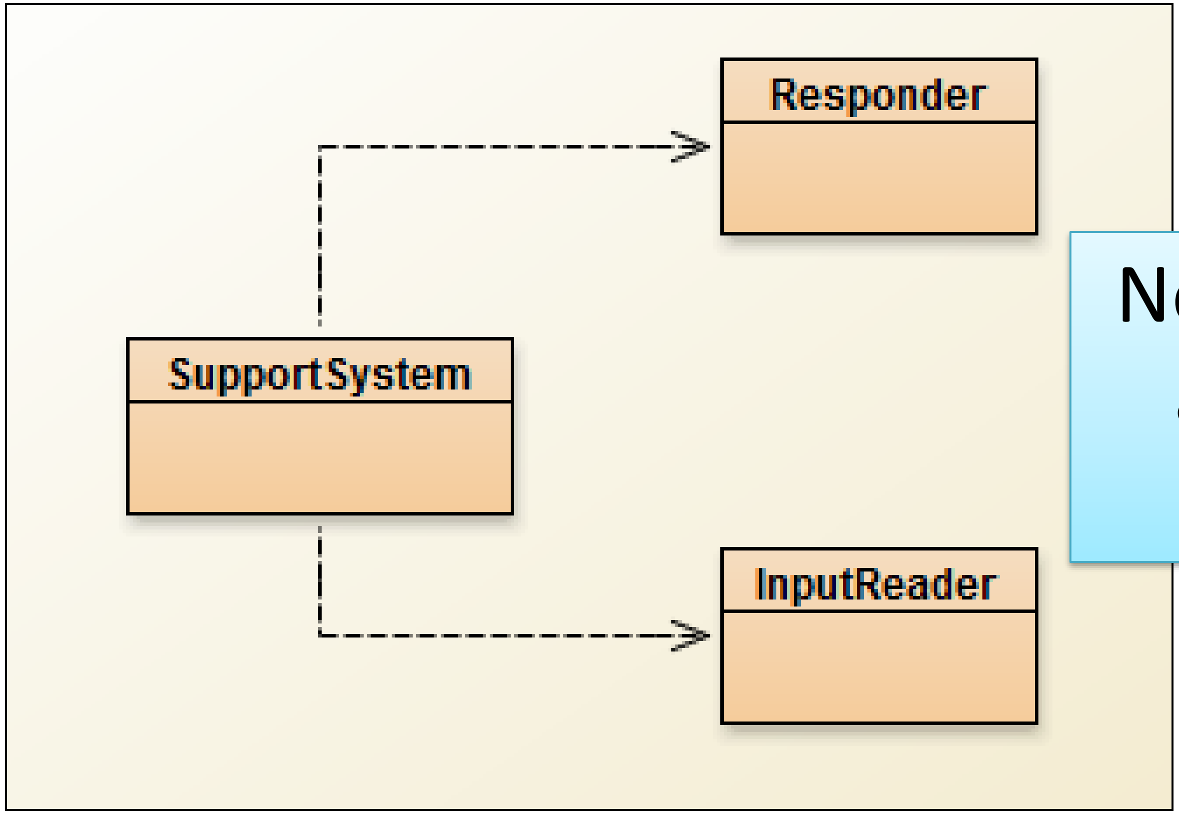
Technical Support System **V2**

Sample

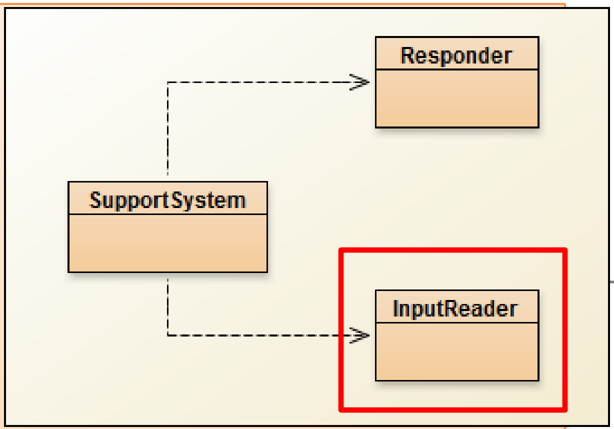


```
Welcome to the DodgySoft Technical Support System.

Please tell us about your problem. We will assist you
with any problem you might have. Please type 'bye'
to exit our system.
> my computer is broken
No other customer has ever complained about this before.
What is your system configuration?
> windows
That sounds odd. Could you describe that problem in more detail?
> it won't boot up
That sounds odd. Could you describe that problem in more detail?
> I get the blue screen of death
I need a bit more information on that.
> it's blue
That sounds interesting. Tell me more...
> really blue
That's not a bug, it's a feature!
>
```

Class Diagram **V2**



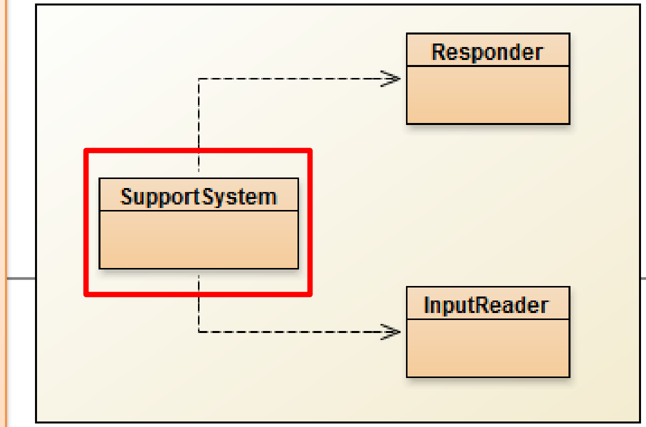
No change at class level



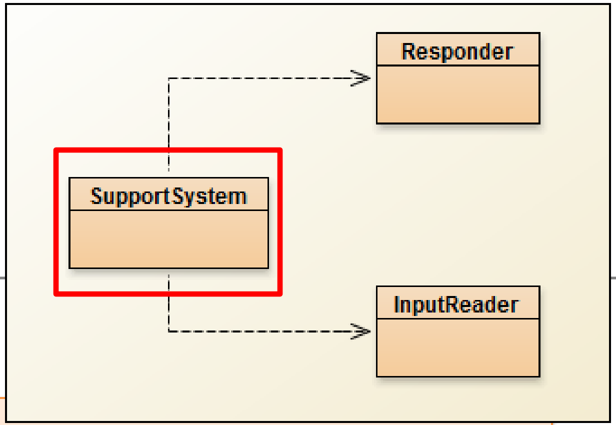
```
import java.util.Scanner;

public class InputReader{

    Scanner input;

    public InputReader(){
        input = new Scanner(System.in);
    }

    /**
     * Read a line of text from the console and return it as a String.
     *
     * @return  A String typed by the user.
     */
    public String getInput() {
          System.out.print("> ");               // print prompt
          String inputLine = input.nextLine().trim().toLowerCase();
          return inputLine;
    }
}
```

No change in this class

```
public class SupportSystem{
    private InputReader reader;
    private Responder responder;

    public SupportSystem() {
        reader = new InputReader();
        responder = new Responder();
    }

    public static void main(String[] argvs){
        SupportSystem app = new SupportSystem();
        app.start();
    }

    public void start(){
        printWelcome();
        String input = reader.getInput();
        while(! input.startsWith("bye")) {
            String response = responder.generateResponse();
            System.out.println(response);
            input = reader.getInput();
        }
        printGoodbye();
    }
```



No change in this class



```
private void printWelcome(){
      System.out.println("Welcome to the DodgySoft Technical Support System.");
      System.out.println();
      System.out.println("Please tell us about your problem. We will assist you");
      System.out.println("with any problem you might have. Please type 'bye'");
      System.out.println("to exit our system.");
  }

  private void printGoodbye(){
      System.out.println("Nice talking to you. Bye...");
  }

}
```

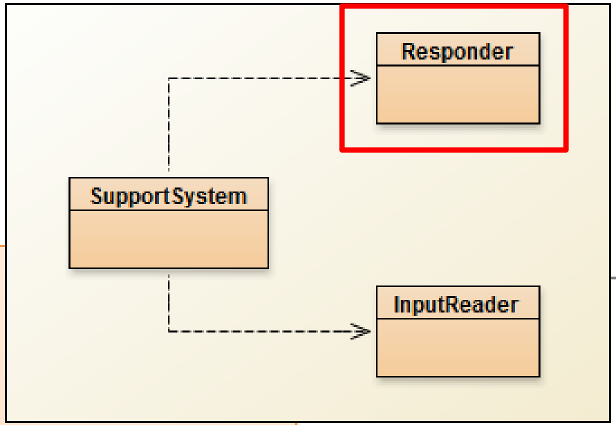
No change in this class

Topic List

1. Recap: Technical Support System V1

2. Technical Support System V2
   - Overview of the System
   - **Responder** class
     - Random

This class <u>will</u> change - to generate a random response



```
//V1 code
public class Responder{

  /**
   * Construct a Responder - nothing to do
   */
   public Responder(){
   }

   /**
   * Generate a response.
   * @return   A string that should be displayed as the response
   */
   public String generateResponse(){
       return "That sounds interesting. Tell me more...";
   }
}
```

Repository of Responses

- Instead of responding with:
  - "That sounds interesting. Tell me more...";

- We would like to respond with a **<u>random</u>** response from a repository of responses

  - "That sounds interesting. Tell me more..."
  - "I need a bit more information on that."
  - "Have you checked that you do not have a dll conflict?"
  - "That is explained in the manual. Have you read the manual?“
  - "That's not a bug, it's a feature!"
  - "Could you elaborate on that?“
  - etc.

  repository

- But how do we **randomise** the selection of a response?

Topic List

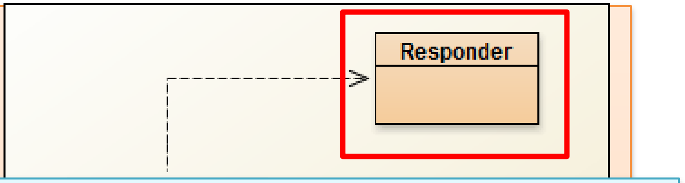
1. Recap: Technical Support System V1

2. Technical Support System V2
   - Overview of the System
   - **Responder** class
     - Random

Using **Random**

- The library class **Random**
  can be used to generate random numbers

```
import java.util.Random;  // import the library class
...
Random randomGenerator = new Random();  // declare & init variable
...
//random int number (no upper or lower bound)
int index1 = randomGenerator.nextInt();

//random number between 0 (inclusive) and 100 (exclusive)
int index2 = randomGenerator.nextInt(100);
```

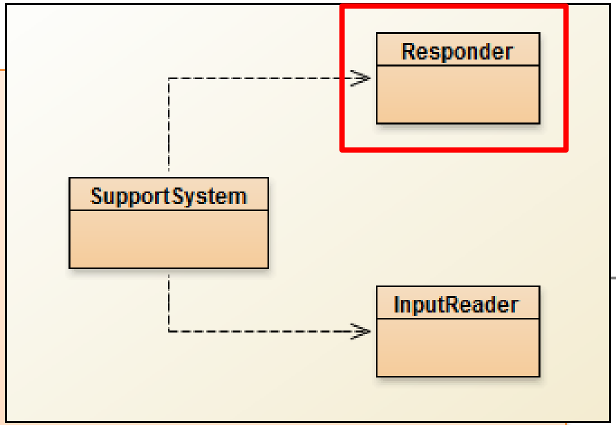


```
import java.util.ArrayList;

public class Responder{
  private ArrayList<String> responses;

  public Responder()   {
    responses = new ArrayList<String>();
    fillResponses();
  }

  private void fillResponses() {
    responses.add("That sounds odd. Coul
    responses.add("No other customer has ever complained about this before. \n" +
                     "What is your system configuration?");
    responses.add("That sounds interesting. Tell me more...");
    responses.add("I need a bit more information on that.");
    responses.add("Have you checked that you do not have a dll conflict?");
    responses.add("That is explained in the manual. Have you read the manual?");
    responses.add("Your description is a bit wishy-washy. Have you got an expert\n" +
                     "there with you who could describe this more precisely?");
    responses.add("That's not a bug, it's a feature!");
    responses.add("Could you elaborate on that?");
  }
}
```

V2.0 Responder…changes

The Responder class declares a private ArrayList of Strings called **responses**, which is initialised in the constructor.

This is used by fillResponses()
to create the repository of responses.



```java
import java.util.ArrayList;
import java.util.Random;

public class Responder{
    private Random randomGenerator;    // declare
    private ArrayList<String> responses;

    public Responder()    {
            randomGenerator = new Random(); // init
            responses = new ArrayList<String>();
            fillResponses();
    }

    public String generateResponse() {
            // Pick a random number between 0 (inclusive) and the size
            // of the ArrayList (exclusive).
            int index = randomGenerator.nextInt(responses.size());  // repo size
            return responses.get(index);
    }
```

V2.0 Responder changes…

to return a **random** response from a repository of responses.

Any Questions?