# Diagnosing Design Problems: Tasks

* Required

1. **Name:** *

2. **Phase:** *

   *Mark only one oval.*

   - 1 *Skip to question 3.*
   - 2 *Skip to question 4.*

## Basic Training

## Task 1) You will receive a basic training about code anomalies and design problems. The training will be 15 minutes long.

3. **Task 1: before starting this task, indicate here what time it is now:** *

   *Example: 8:30 AM*

## Configuration

4. **Component:** *

   *Mark only one oval.*

   - PushPull
   - Workflow

## Task 2) You will have 20 minutes to understand the component above. For this task, you should read the component's documentation and source code, which were provided to you before you start.

5. **Task 2: before starting this task, indicate here what time it is now:** *

   *Example: 8:30 AM*

6. **Technique:** *

   *Mark only one oval.*

   - Traditional Technique *Skip to question 7.*
   - Synthesis Technique *Skip to question 18.*

## Tasks with Traditional Technique

## Task 3) You will understand how to use traditional technique to diagnose design problems. In this task, you will receive a guide and a basic training about the traditional technique. The training will be 10 minutes long.

7. **Task 3: before starting this task, indicate here what time it is now:** *

   *Example: 8:30 AM*

## Task 4) In this task, you will use traditional technique to diagnose DESIGN PROBLEMS. For each problem found, you have to provide the following information: (i) short description of the problem, (ii) possible consequences caused by the problem, (iii) classes, methods and components realizing the problem in the source code, and (iv) name(s) of code anomaly(ies) that helped you to diagnose the design problem. You will have 40 minutes to finish this task.

8. **Task 4: before starting this task, indicate here what time it is now:** *

   *Example: 8:30 AM*

9. **Design problems:** *
   (i) short description of the problem, (ii) possible consequences caused by the problem, (iii) classes, methods and components realizing the problem in the source code, and (iv) name(s) of code anomaly(ies) that helped you to diagnose the design problem

## Answer the following questions regarding Task 4:

10. **A - Which were the main challenges to diagnose design problems?** *

11. **B - Did you understand all information provided by the technique? Please provide details about this.** *

12. **C - Which types of information were fundamental to diagnose design problems? Please rank these types of information according to their relevance.** *

13. **D – Was there any piece of provided information that was useless to perform Task 4? Why and Which one(s)?** *

14. **E - Do you feel there is any non-provided information that could help to diagnose design problems? Please explain what additional information would be helpful to diagnose the design problems.** *

15. **F - Have you used all types of code anomaly? Please provide details about this.** *

16. **G – Which was the most useful type of code anomaly?** *

17. **H – How the graphical interface provided by the technique affected Task 4?** *

*Stop filling out this form.*

## Tasks with Synthesis Technique

## Task 3) You will understand how to use synthesis technique to diagnose design problems. In this task, you will receive a guide and a basic training about the synthesis technique. The training will be 10 minutes long.

18. **Task 3: before starting this task, indicate here what time it is now:** *

*Example: 8:30 AM*

## Task 4) In this task, you will use synthesis technique to diagnose DESIGN PROBLEMS. For each problem found, you have to provide the following information: (i) short description of the problem, (ii) possible consequences caused by the problem, (iii) classes, methods and components realizing the problem in the source code, and (iv) name(s) of agglomeration(s) that helped you to diagnose the design problem. You will have 40 minutes to finish this task.

19. **Task 4: before starting this task, indicate here what time it is now:** *

*Example: 8:30 AM*

20. **Design problems:** *

    (i) short description of the problem, (ii) possible consequences caused by the problem, (iii) classes, methods and components realizing the problem in the source code, and (iv) name(s) of code anomaly(ies) that helped you to diagnose the design problem

## Answer the following questions regarding Task 4:

21. **A - Which were the main challenges to diagnose design problems?** *

22. **B - Did you understand all information provided by the technique? Please provide details about this.** *

23. **C - Which types of information were fundamental to diagnose design problems? Please rank these types of information according to their relevance.** *

24. **D – Was there any piece of provided information that was useless to perform Task 4? Why and Which one(s)?** *

25. **E - Do you feel there is any non-provided information that could help to diagnose design problems? Please explain what additional information would be helpful to diagnose the design problems.** *

26. **F - Have you used all categories of agglomerations? Please provide details about this.** *

27. **G – Which was the most useful category of agglomeration?** *

28. **H – How the graphical interface provided by the technique affected Task 4?** *