POLYTECHNIQUE MONTREAL LOG8430E

FALL 2025 - TP1

Design Principles and Software Quality: Object-Oriented Architectures

This assignment focuses on object-oriented architectures and design, specifically design principles and quality. You will also be identifying anomalies and correcting them. All following questions will pertain to the open-source project **AntennaPod**, an open-source podcast manager for Android (https://github.com/AntennaPod/AntennaPod).

To answer the questions, you may use any of the tools recommended on Moodle, the introductory slides, or any other relevant tools. You must explicitly cite any tools and sources you use to complete this assignment.

The report must be prepared using the provided LaTeX format. You may add subsections as needed.

Submit a single document per team with the answers to the questions. Include your team's name and all team members' names in the report. Additionally, include all external references, such as articles, links, documentation, and tools.

Please name your report as "TP1_[team_name].pdf".

Deadline: Saturday, October 18, 2025, before 23h55

Question 1: Software architecture analysis (24 points)

You are asked to analyze the architecture of the application with the goal of understanding its organization. Focus primarily on the **app** module, excluding test code and non-code files (such as images) from your analysis. Specifically, you are asked to:

1) Create an architecture (package) UML diagram to illustrate the architecture of the system. **(10 points)**
   a) Make sure your figures are clear (not too small or zoomed out).
   b) Use vector images (e.g., EPS or PDF) to ensure the quality of the image.
   c) Remove any details that are not necessary for your architecture discussion (e.g., methods, attributes or event classes) to avoid overloaded figures.
   d) Feel free to use multiple diagrams (e.g., focusing on specific parts) to facilitate the rest of the discussion.
   e) Make sure that all elements referenced in the discussion are present in the diagram.

2) By referring to the styles of architecture presented in the course, identify the architectural style(s) for the implementation of the system. **(4 points)**

3) By referring to the standard organization pattern as per the employed style(s), identify the role of each module (package) in the architecture. **(10 points)**
   a) For example, if a client-server architecture is identified, what modules are part of the client side and which ones of the server side?
   b) If you identify more than one style, a module can have a role in more than one style. Make sure to clearly describe the roles for each of the identified styles.

Format of the answer:

- Write a section about the architecture of the application. In this section, briefly present the architecture of the system and the styles that are present in it.

- Present your diagram(s) and comment how the diagrams show the styles you have identified.

- Write one subsection for each one of the styles you have identified and discuss the roles of the different modules. You can add more diagrams here to facilitate the discussion for each style.

Question 2: SOLID Design Principles (24 points)

You are asked to:

1) Find and name **one** instance of **two** different SOLID principles present in the application. **(6 points)**

2) Present a UML diagram for each SOLID principle found in the application separately. **(6 points)**

3) Find and name **one** instance of **two** different SOLID violations in the application. **(6 points)**

4) Present a UML diagram for each SOLID violation found in the application separately. **(6 points)**

Format of the answer:

- Write a section about the found SOLID principle/violation in the application. Write a small paragraph summarizing your findings (what principle you found, what they are used for, what is wrong, etc.)

- Write one subsection for each principle including the UML diagram or the code snippet.

Question 3: Measuring the design quality of the application (32pts)

1) Present a report with all metrics for the application (cohesion, complexity, coupling, size) including averages, maximums, standard deviations etc. Present the metrics which we have already discussed (NOM, LOC, LCOM, WMC, CC, CBO, etc.). Present the metrics on the appropriate level (project, package, class). **(20 points)**

   Q: Which metrics should we present? How do we know we have included enough?

   A: Simple rules :
   - Present all the metrics that you refer to in the report.
   - Present all metrics that you use in calculating other metrics.

2) Comment on the overall quality of the system and its impact on the development and the maintenance of the system (e.g., "the system is cohesive which guarantees the SRP (Single Responsibility Principle) and facilitates maintenance.") **(12 points)**

   Q: How do we comment? What is the expected level of the comments?

   A: Comment based on what we have seen so far, concerning principles, design quality properties. The level should be rather high and abstract. Do not concentrate on specific cases (yet).

Question 4: Find two (2) different anomalies and correct them by refactoring (20 points)

1) Present the anomalies and justify their presence using metrics. It is possible that the tools may not consider certain cases as anomalies. In such cases, disregard the tools. **(5 points)**

2) Name the anomalies in terms of code smells (for example, low cohesion + high coupling = Feature Envy). **(5 points)**

3) Refactor and present the code before and after the refactoring. If the code is too long, provide UML diagrams or a shortened version of the code (e.g., with only the method signatures). **(5 points)**

4) Explain why each anomaly is corrected by the refactoring you have chosen. **(5 points)**

Recommended tools

For metrics:

1. MetricsReloaded - https://plugins.jetbrains.com/plugin/93-metricsreloaded
2. MetricsTree - https://plugins.jetbrains.com/plugin/13959-metricstree
3. Understand - https://licensing.scitools.com/login

For refactoring/code smells:

1. Android Studio - https://developer.android.com/studio
2. SonarCloud - https://sonarcloud.io/

Note of evaluation

The submitted report will be evaluated on the accuracy and detail of the responses and the quality of its writing. Treat this as an official and professional report to management and colleagues. Please make sure to provide the reference to any material, tools or texts that you will use in your assignment.

Note on completing the assignment

Tools to automatically identify design patterns or design principles are mainly products of research and they are not known for their stability or for their usability. You are always free to experiment with these tools, but your main tools are the UML generation tools, which implies studying diagrams. This is an expected challenge for this assignment.