

# Spesifikasi Tugas Besar 2

## IF2210 - Pemrograman Berorientasi Objek

### Semester 2 2017/2018

#### ArkavQuarium

#### 1. Outcomes

Pada tugas besar kali ini, Anda diharapkan:

1. Mampu menerjemahkan sebuah aplikasi C++ menjadi aplikasi yang fungsinya sama dalam bahasa JAVA.
2. Mampu membuat dokumentasi *source code* dengan menggunakan JavaDoc.
3. Mampu melakukan *static analysis* sebuah *source code* dengan menggunakan Checkstyle.
4. Mampu mengimplementasikan *Test Unit* menggunakan JUnit.
5. ( Bonus ) Mampu melakukan pengukuran nilai “metric” *source code* Java dengan menggunakan tools JDepend.

#### 2. Deskripsi Umum Tugas

Terjemahkan dua versi terakhir dari Tugas Besar 1 dari bahasa C++ ke bahasa Java. Jika sebelumnya Anda memakai STL yang disediakan C++, kali ini pakailah STL yang disediakan Java. Pada tugas besar ini, Anda harus membagi pekerjaan/tugas dengan **ADIL** di antara anggota kelompok. Setiap mahasiswa harus bertanggung jawab terhadap minimal 1 Unit/Class.

Untuk GUI yang sebelumnya menggunakan *library* C++ dari kami, gantilah penggunaan *library* tersebut dengan Java Swing, silahkan lakukan eksplorasi secara mandiri. Setelah bersama-sama menentukan rencana perubahan, Anda juga harus membagi pekerjaan untuk *test* aplikasi dan pemakaian *tools* dengan rincian sebagai berikut:

1. Setiap kelas harus memiliki *unit testing* menggunakan JUnit. Mahasiswa yang bertanggung jawab atas suatu kelas harus membuat unit test untuk kelas tersebut.
2. Satu orang mahasiswa sebagai penanggungjawab kualitas *source code* menggunakan CheckStyle. Mahasiswa yang bertanggung jawab hanya bertugas untuk memastikan kualitas *source code*. Perbaikan *source code* dari suatu kelas agar mencapai standar *CheckStyle* merupakan **Bonus**.
3. Satu orang mahasiswa sebagai penanggungjawab dokumentasi *source code* dengan JavaDoc. Mahasiswa yang bertanggung jawab hanya bertugas memastikan dokumentasi *source code* lengkap. Perbaikan dokumentasi untuk *source code* dari suatu kelas merupakan **Bonus**.

4. ( **Bonus** ) Satu orang mahasiswa sebagai penanggungjawab pengukuran metrik perangkat lunak menggunakan JDepend.

Selain itu berikut merupakan komponen OOP yang harus ada:

1. Inheritance (“Is-a” relationship)
2. Method override pada inheritance
3. Aggregation (“Has-a” relationship)
4. Polymorphism
5. Generic
6. Abstract class
7. Function/method overloading
8. Interface
9. Multiple inheritance (dalam bentuk satu class dan satu atau lebih interface)

**Tambahan:**

**Spesifikasi Bonus pada Tugas Besar 1 akan menjadi Bonus juga pada Tugas Besar 2**

### 3. Tugas Kelompok

Tugas dikerjakan secara berkelompok dengan rincian sebagai berikut :

1. Menganalisis kembali aplikasi yang akan diterjemahkan dan membagi pekerjaan ke **SEMUA** anggota tim
2. Membuat rancangan kelas/interface JAVA (hasilnya seperti dijelaskan dalam dokumen teknis)
3. Menerjemahkan kode program dalam bahasa Java yang dikerjakan **oleh setiap individu** (setiap anggota kelompok harus jelas mengimplementasikan kelas tertentu), dan melakukan *Unit Testing*.
4. Melakukan integrasi dan test aplikasi menggunakan skenario yang sama dengan yang pernah dilakukan pada program C++

### 4. Jadwal Pengerjaan Tugas

Week	Tanggal/Periode	Deskripsi Aktivitas
W13	12 April s.d. 24 April 2018	Klarifikasi tugas, Pembagian kerja, dan analisis perubahan, koding
W15	Rabu, 25 April 2017	Submisi <i>source code</i> , hasil pengukuran dengan <i>tools</i> dan pengumpulan <i>softcopy</i> ke Olympia.

	Kamis, 26 April 2017	Demo penilaian eksekusi dan pemeriksaan hasil tugas lainnya
--	----------------------	---

**Catatan :**

1. Anda **wajib** melakukan asistensi **minimal 1x** di luar jam kelas dengan asisten. Daftar asisten dapat dilihat di Olympia.
2. Salah satu asistensi harus dilakukan **sebelum tanggal 19 April 2018**. Namun Anda boleh melakukan asistensi lagi setelah tanggal tersebut.
3. Ingat bahwa yang harus Anda wujudkan adalah aplikasi yang berjalan dengan baik (jika belum lengkap), dan bonus hanya dikerjakan jika Anda punya waktu lebih.
4. Anda **disarankan** menggunakan **IDE** yang mendukung bahasa Java untuk membantu pembuatan tugas besar kali ini.

## 5. Kerangka Deliverable

1. Folder *src* yang berisi:
  - a. *Source code* yang telah dibuat
  - b. *Makefile* untuk kompilasi program
2. Folder *doc* yang berisi:
  - a. Laporan **dalam format .pdf** sesuai template laporan. Dinamakan **Laporan.pdf**.
  - b. Hasil generation JavaDoc.
  - c. *Readme.txt* yang berisi cara menjalankan program dan hal-hal yang harus diperhatikan

Semua deliverable tersebut di-zip dengan format nama:

Tubes2OOP_<NIM1>_<NIM2>_<NIM3>_<NIM4>.zip
---

*Deliverable* dikumpulkan selambat-lambatnya pada hari **Rabu, 26 April 2018 pukul 23.59** melalui *uploader* di Olympia.

Perhatikan bahwa **Anda akan mendemokan yang Anda submit terakhir saat deadline**. Sebagai gambaran, saat demo, peserta akan diberikan berkas hasil *submission* oleh asisten dan harus di-*build* ulang di komputer peserta. Jadi, pastikan minimal yang Anda

*submit* adalah **program yang dapat dieksekusi**. Jika yang Anda *submit* terakhir tidak dapat dieksekusi, Anda dipastikan mendapat nilai 0.

## 6. Bahan Bacaan

<http://junit.org/junit4/>

<http://www.tutorialspoint.com/junit/>

[https://en.wikipedia.org/wiki/Static\\_program\\_analysis](https://en.wikipedia.org/wiki/Static_program_analysis)

<http://www.sw-engineering-candies.com/blog-1/comparison-of-findbugs-pmd-and-checkstyle>

[https://en.wikipedia.org/wiki/List\\_of\\_tools\\_for\\_static\\_code\\_analysis#Java](https://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis#Java)

<https://en.wikipedia.org/wiki/Checkstyle>

<http://clarkware.com/software/JDepend.html>

<http://www.oracle.com/technetwork/java/javase/tech/index-137868.html>

IntelliJ IDEA Tutorial (**Tidak Harus Diimplementasikan Melalui Metode Ini**)

<https://www.jetbrains.com/help/idea/configuring-testing-libraries.html>

<https://www.jetbrains.com/help/idea/generating-javadoc-reference-for-a-project.html>

<http://www.practicesofmastery.com/post/intellij-checkstyle-google-java-style-guide/>