

Obiective

Inspectarea documentelor asociate cerințelor, arhitecturii și codului sursă.

Enunț

Să se realizeze următoarele task-uri pentru proiectul primit.

Task, puncte	Descriere task
[Maven] 1 punct	<ul style="list-style-type: none"> Fiecare echipă va lucra pe un proiect pentru rezolvarea task-urilor primite pe parcursul semestrului. Componenta echipei și proiectul se stabilesc în timpul primului laborator. Proiectul conține: un document cu cerințele aplicației, o diagramă de clase și un proiect Maven al unei aplicații Java. Se descarcă din Teams proiectul atribuit și se dezarchivează. Se rulează proiectul Maven. Dacă este necesar, se poate folosi un tutorial dedicat pentru crearea și configurarea unui proiect Maven (vezi Tutorial Maven). Se recomandă folosirea: <ul style="list-style-type: none"> JDK 21+ (https://www.oracle.com/java/technologies/javase-downloads.html); IntelliJ IDEA Ultimate (cu licență gratuită pentru studenții înregistrați cu e-mail @scs.ubbcluj.ro pe platforma JetBrains, https://www.jetbrains.com/idea/download/#section=windows).
[Repository Git] 1 punct	<ol style="list-style-type: none"> Fiecare echipă va crea un repository Git (vezi Tutorial Git) în care se va încărca proiectul Maven cu sursele inițiale, urmând a fi actualizat la fiecare temă de laborator. Repository-ul Git va avea un folder Docs în care se vor depune pentru fiecare temă de laborator predată documentele elaborate. De exemplu, pentru Lab01 se va crea folder-ul Docs/Lab01 care va conține următoarele documente: Lab01_ReviewReport.xlsx, Requirements_v2.0.pdf și Diagram_v2.0.pdf.
[Inspectare și adaptare/corectare] 6 puncte	<ol style="list-style-type: none"> Pentru proiectul primit se vor identifica ambiguitățile/defectele existente în documentele Requirements_v1.0.pdf, Diagram_v1.0.pdf și codul sursă (proiectul Maven). Se vor folosi <i>checklists</i> (vezi CheckLists.zip) asociate fiecărui tip de document inspectat (vezi Tutorial Inspectare): <ol style="list-style-type: none"> 1.1. pentru inspectarea cerințelor se va folosi fișierul Lab01_RequirementsPhaseDefectsChecklist.pdf; 1.2. pentru inspectarea arhitecturii se va folosi fișierul Lab01_ArchitecturalDesignPhaseDefectsChecklist.pdf; 1.3. pentru inspectarea codului sursă se va folosi fișierul Lab01_ProgramCodingPhaseDefectsChecklist.pdf. Pentru documentele inspectate se va completa un <i>raport de inspectare</i> (Lab01_ReviewReport.xlsx). Se vor completa minimum 3 elemente/observații pentru fiecare document inspectat. Pe baza raportului realizat, se vor efectua îmbunătățirile/corecturile corespunzătoare documentelor inspectate (cerințe, arhitectură, cod sursă). Se vor obține <i>3 tipuri de documente noi</i>: <ol style="list-style-type: none"> 2.1. cerințele corectate/actualizate/clarificate; 2.2. arhitectura aplicației, i.e., diagrama de clase corectată/actualizată; 2.3. codul sursă modificat conform cerințelor și reflectând arhitectura actualizată. Se va avea în vedere alocarea a maxim 30 minute pentru fiecare tip de document inspectat.
[SonarQube] 2 puncte	<ul style="list-style-type: none"> După inspectarea codului, se va instala SonarQube (vezi Tutorial SonarQube) pentru a realiza o evaluare statică a calității codului. Se vor analiza/discuta aspectele semnalate de către SonarQube pentru minim două clase. În <i>raportul de inspectare</i> (Lab01_ReviewReport.xlsx) se vor completa în spreadsheet-ul Tool-basedCodeAnalysis minim 5 elemente care au fost indicate de SonarQube și, fie: <ul style="list-style-type: none"> – au fost corectate conform sugestiilor (<i>Before</i> și <i>After</i>); – nu au fost corectate, precizând motivul (<i>Argument</i>).
[Chestionar] 2 puncte	<ul style="list-style-type: none"> Înainte de predarea temei de laborator, studenții sunt invitați să completeze un chestionar cu întrebări referitoare la instrumentele AI utilizate pentru rezolvarea task-urilor incluse în tema de laborator. Link-ul la chestionar va fi furnizat la începutul săptămânii 3. Studenții care completează chestionarul și predau tema de laborator la timp primesc nota 12 pentru Lab01. Tema de laborator predată cu întârziere este depunctată cu minim 2 puncte, dar se acordă cele 2 puncte aferente chestionarului, dacă studentul l-a completat anterior predării temei de lab.

Observații**[Inspectare și adaptare/corectare]**

- I. Inspectarea presupune studierea unor documente și elaborarea unui raport de inspectare.
- II. Inspectarea poate evidenția necesitatea efectuării unor modificări asupra documentelor analizate, determinate de:
 - ambiguități și omisiuni care pot apărea în enunțului problemei;
 - deficiențe existente la nivelul proiectării și / sau a codului sursă.
- III. Modificările aplicate vor permite obținerea unei aplicații cu următoarele caracteristici:
 - codul sursă este corectat conform sugestiilor furnizate SonarQube;
 - **nu se scriu teste pentru funcționalități.**

TimP de lucru recomandat pentru rezolvarea temei de laborator ~ 4 ore/echipă.

Predarea temei de laborator

Se vor elabora următoarele documente:

- [1]. raportul **Lab01_ReviewReport.xlsx** completat cu:
 - i. datele de identificare ale echipei (numele și prenumele, grupa);
 - ii. titlul temei de laborator și data realizării;
 - iii. observații, comentarii și îmbunătățiri sugerate referitor la documentele inspectate;
- [2]. documente create sau modificate pe baza sugestiilor de la punctul [1].iii:
 - i. cerințele modificate (fișier cu numele **Requirements_v2.0.pdf**);
 - ii. o diagramă de clase (fișier cu numele **Diagram_v2.0.pdf**);
 - iii. proiectul Maven cu **codul sursă** modificat după utilizarea SonarQube, pe git.
 - iv. fișierele **Lab01_ReviewReport.xlsx**, **Requirements_v2.0.pdf** și **Diagram_v2.0.pdf** se vor încărca în folderul **Docs/Lab01**, în repository-ul Git propriu, deja creat.

Termene de predare

Săptămâna	Tema de laborator		Primul termen de predare	Ultimul termen de predare
S01	L01.	Inspectare.	S03	S05*
S02		Git, Maven, Sonarlint	S04	S06*

*) Temele restante se vor putea preda în limita timpului disponibil.