**OOP Project Checklist**

**General requirements:**

* OOP style
* Properly Serialization
* Intuitive usage
* Consistency with UML
* Comparable, Cloneable, Serializable, equals(), hashcode(), toString()…
* Proper and logically consistent usage of Collections
* Usage of java api (standard classes). Do not reinvent the bicycle.

**Approximate Checklist:**

* **Admin**
  + Manage Users (Add, remove, update)
* **Teacher**
  + View Courses
  + Manage Course files
  + View Students, info about students
  + Put marks
  + Send order to Executors (actually, any employee can send the order)
* **Student**
  + View Courses, Register for Courses
  + View Course files
  + View info about teacher of a specific course
  + View Marks
  + View Transcript
  + Rate teachers
  + Get Trasncript
* **Executor**
  + Accept/reject order
  + View orders (after that accept some order, for example)
  + View accepted orders
  + getSalary() must be overridden to account for hardworking executors (they must get higher salary).
* **Manager**
  + Approve students registration
  + Assign courses to teachers
  + Manager types – OR, Deparments, etc. (use enumeration).
  + Create statistical reports on academic performance
  + Manage news

**Important note:**

**Account for details! For example:**

* Students can’t have more than 21 credits
* Students can’t fail more than 3 times
* FIT Students can’t have FOGI courses (only as electives)
* Mark consists of 1st, 2nd attestation, and final.
* Many more, really … ☺

**Bonus:**

Something extra and valuable, e.g. Schedule generation (take into account room load, room type, etc.), Attendance, etc.