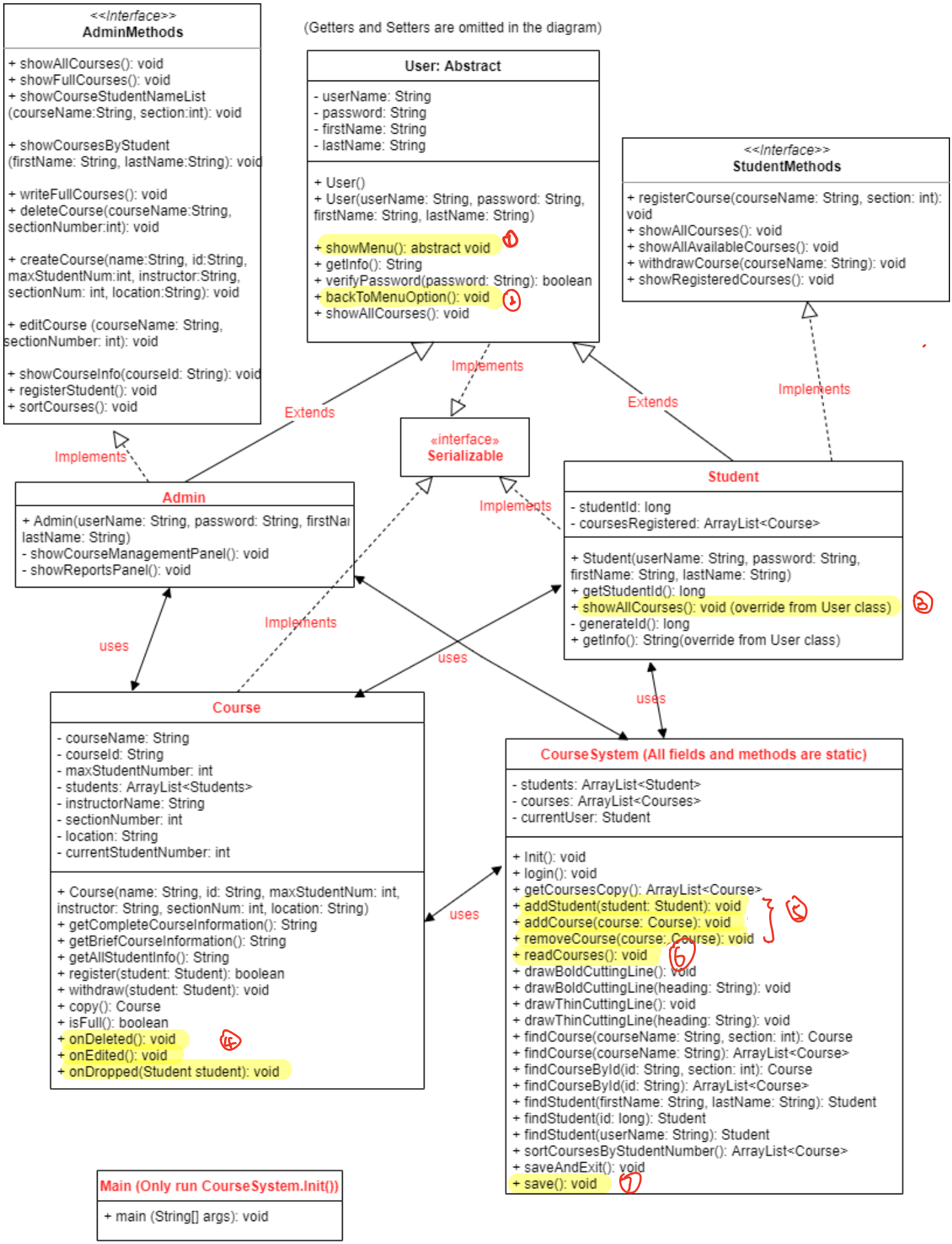
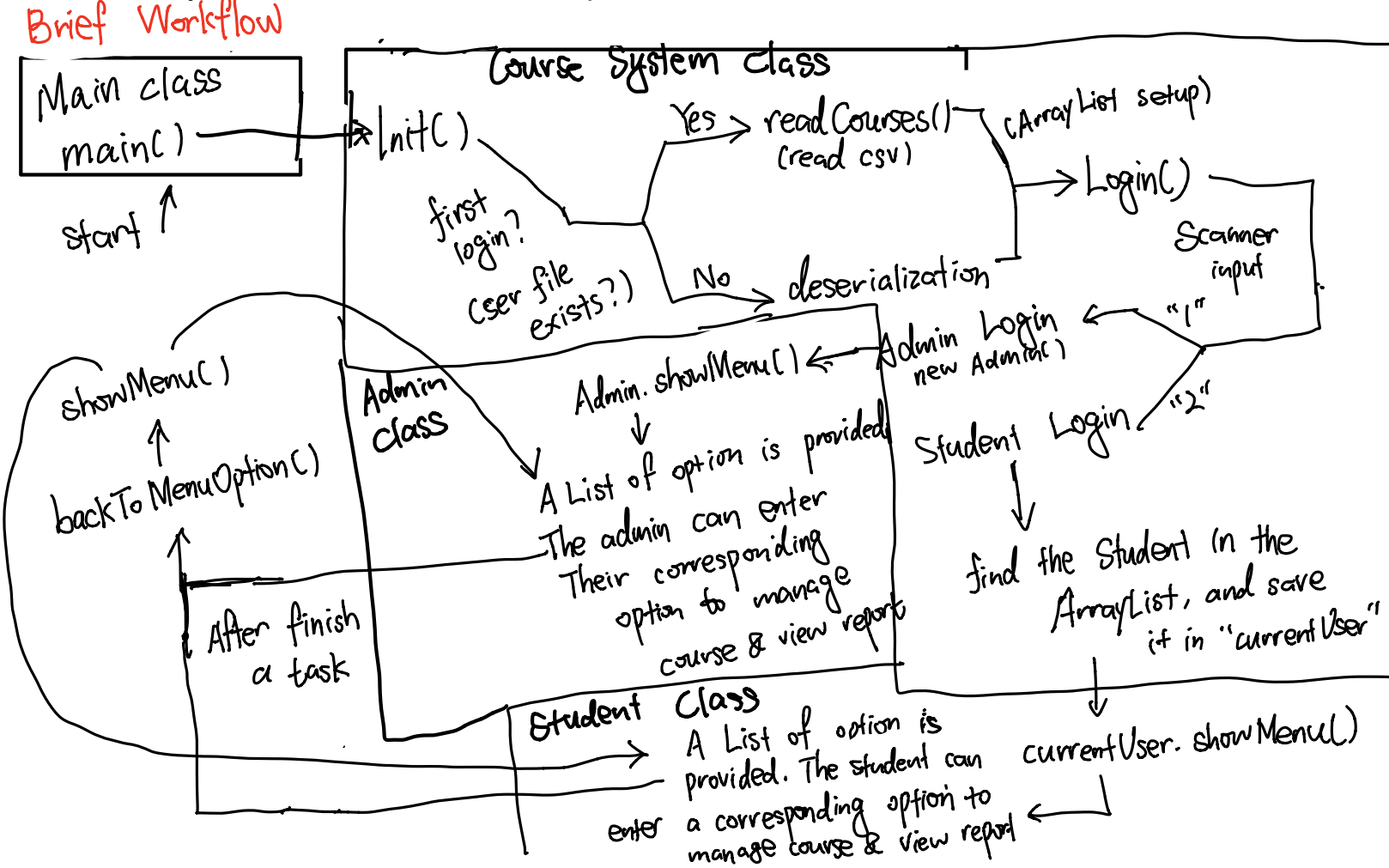
UML Graph Of Course Registeration System

Important fields & methods are marked in the diagram, and are explained in the next page along with the workflow of the program



Clarification: The **CourseSystem** class is a class with only static methods and fields. It acts like a "central system" of the program. It is responsible for storing the ArrayList of all students and courses, as well as serialization/deserialization, login, and searching objects.

An explanation of marked methods in the above diagram:

1. **showMenu():** An abstract method of User class, overridden by Student and Admin classes. Since students & admins have different menus, when it is called, the method shows the specific menu for students and admin.
2. **backToMenuOption():** A method of User class. It is called each time when a user finishes a task (like register a class, show all classes...). When called, it print "Input any key to return to the menu". When the user enters any key, the method then calls **showMenu()** method that shows specific menu for students & admin
3. **showAllCourses():** A method of User class. It shows all classes and their detailed information (student number, list of students' names...). It is overridden by Student class. For students, they can only see **brief** information of all classes (so they are not able to see list of students' names). Since Admin class does not override it, the admin can see all information
4. **onDeleted(), onEdited(), onDropped():** executed each time when **deleteCourse(), editCourse(), and withdrawCourse()** is called by admin or students. It is called to update the corresponding Course object in the ArrayList of CourseSystem class. For **deleteCourse(),** when a course is deleted, all students who registered this course before are automatically dropped from it.
5. **addStudent(), addCourse(), removeCourse():** Executed each time when **registerStudent(), createCourse(), and deleteCourse()** is called by the admin. They are called to update the Course ArrayList and Student ArrayList of CourseSystem class.
6. **readCourses():** Called by Init(). When the program is first run (no .ser file exists), the program reads the .csv file, and update the ArrayList of courses
7. **save():** Called before the program existed; also called when the user finishes each task. It saves the ArrayList of Course and Student objects into their .ser file.