**CSC252 Programming II     Final Project Specification**

You now have a good foundation of Java programming skills that you can use to develop a larger project. You will design and implement one of the following systems, or, **with my prior approval,** some other system. The classes listed here are suggestions – you do not have to implement every one of them, as long as you meet the criteria below.

1.  **Design the program:**  (30 pts)Think about how you might write a program to implement the system.  Include a comprehensive UML diagram for each class except the client. What are the classes? How do they relate? Describe the inheritance, composition, instance variables, static variable, static class. How would the application start? What does the system do? What inputs & outputs would there be?  What might you need that you haven't learned yet?  What assumptions will your program make? **Design and write up an** **algorithm** to describe how the program will run and include a paragraph summarizing the answers to the questions above. Describe how/where your program will used each of the parts in #2 below. You must **upload this design document** when you upload your project. (This part will be due next week.)  
2.  **Implement the program:** (100 pts) The program must have **ALL** of the following components:

1. *(10 pts)* Implement a **client class** and at least **2 other classes, plus 1 subclass. The client will instantiate each of the other classes at least once.  
   Instantiate every class object.**

**2.** *(10 pts)* Declare and use at least 1 **constructor** and **3 instance variables per class, plus 1 public static variable.  
Make sure every class at least 3 variables. 1 public static variable per class too?**

**3.** *(5 pts)* Use **composition** in one of the classes.  
Composition: Player has a Persona that is a Path

OR

Player has a character that is a Type(warrior/rouge/mage)

**4.** *(5 pts)* D**efine get and set methods for each instance variable in each class. Define appropriate toString methods for each class (except the client class.)  
Are toStrings necessary?**

**5**. *(10 pts)* In addition, define and appropriately use at least **one static method and one other regular method.  
did I already do this?**

**6.** *(5 pts)* Define a method that **overrides** a method of its superclass .  
**Does Path perform this?**

**7.** *(5 pts)* Client keeps running in a **loop** until the user decides to end the program  
**If player wins or loses during battle ask to retry?**

8.  *(5 pts)* Client asks the user for **input**, and prints out **output** based on object data (not based on data stored in the client’s variables)

**9.**  *(10 pts)* Generate and use some data necessary to the system using a **random number generator**

**10.** *(10 pts)* Declare and appropriately use at least **one array or ArrayList.**

11. *(5 pts)* **ALL input/output** occurs in the client class. **NO** input/output occurs in the other classes

12. *(20 pts)* **Program runs correctly. System adequately performs its expected tasks.**

13. meet the **code header and documentation** requirements, including specifying *predicted test* input and output   
 (***0 credit*** for the project if this is missing)

3.  **Present your code** to the class. (20 pts) You will introduce the program with a summary, including **what areas presented challenges** and **how they were resolved**.  We will go through a **"code review"** where I and the members of the class **will ask questions** about the code and **give feedback** and constructive suggestions about your code.

**This is an individual assignment.  You are NOT to work with others on this!  
Here's a program you can show off to your friends!  Good luck and have fun!**