**D&D project requirements – Walter Stuart Davenport**

User Requirements

* CharacterCreation – Functional Requirement
  + The User should be able to choose from several different races that will edit the character’s ability scores and traits.
  + The User should be able to choose from several different classes that will determine the character’s saving throws, starting equipment, hit points, and traits.
  + The User should be able to choose a level for which their character will be, this will modify the character’s abilities

System Requirements

* CharacterCreation – functional requirement
  + Global Variables
    - int StrMod
      * will be based off of the character’s strength score
    - int StrSave
      * will be based off of the character’s strength score
    - int DexMod
      * will be based off of the character’s dexterity score
    - int DexSave
      * will be based off of the character’s dexterity score
    - int ConMod
      * will be based off of the character’s constitution score
    - int ConSave
      * will be based off of the character’s constitution score
    - int IntMod
      * will be based off of the character’s intelligence score
    - int IntSave
      * will be based off of the character’s intelligence score
    - int WisMod
      * will be based off of the character’s wisdom score
    - int WisSave
      * will be based off of the character’s wisdom score
    - int CharMod
      * will be based off of the character’s charisma score
    - int CharSave
      * will be based off of the character’s charisma score
    - ArrayList<Armor> allArmor
      * The armor the character can equip (including shields)
    - ArrayList<Weapon> allWeapons
      * The weapons the character can equip
    - String name
      * The character’s name
  + Methods
    - Getters and Setters
    - void equiptArmor(Armor armor)
      * to determine the character’s armor class
    - void equiptWeapon(Weapon weapon)
      * to determine the attacks the character can make
    - void turn(Monster enemy)
      * calls action, bonus action, reaction determined by the character’s class
* Abstract Race – non-functional requirement
  + Global Variables
    - String name
    - Size size
  + Methods
    - Getters and Setters
    - void editScores(String score1, String score2)
      * all races will edit two scores
* Abstract CharacterClass – non-functional requirement
  + Global Variables
    - String name
    - int level
    - int prof
      * determined by level
  + Methods
    - Getters and Setters
* CharacterClasses – functional requirement
  + Global Variables
    - Die hitDie
      * Determines the health of the character
  + Methods
    - Getters and Setters
    - if Spells are available
      * have a spell list
      * have spell slots
        + keep track of how many spells they have left and at what levels they can cast it at
    - if has resistances
      * which damage resistances it has
    - define all of its actions, bonus actions, and reactions
      * int rollToHitWith(Weapon weapon, Enemy enemy)
        + for actions or bonus actions
* Race – functional requirement
  + Global Variables
  + Methods
    - Depending on the race, will give attributes to the character