# Character Builder Data Structure Documentation

for D&D 5e

## Simplifications

For simplicity, I eliminated fields that are not inherent properties of a character but rather things a character owns or can perform such as spells, features, actions, and equipment. This also means that the character builder does not handle encumbrance. I have also removed the ability to multiclass and choose sub-races. I might add these features later, but they are not necessary for the MVP.

## AT1 | Fields

See UML diagram.

## AT2 | Data Types

See UML diagram.

## AT3 | Restrictions

General:

*all number fields except character.proficiencyBonus*: must be >= 0

Character:

level: 1 <= this.level <= 20

Description:

background: *optional*

appearance: *optional*

Characteristics:

*all fields*: *optional*

age: character.race.age.min <= this.age <= character.race.age.max

Personality:

*all fields*: can contain 0 or many elements

Stats:

*all fields*: each Stat is initialized with a unique value from the set [15, 14, 13, 12, 8]

Health:

current: this.current <= this.max

Age:

min: this.min <= this.max

## AT4 | Dependencies

Character:

proficiencyBonus: this.SomeUniqueClass.proficiencyBonuses[this.level]

armorClass: 10 + this.savingThrows.dexterity.modifier

+ (this.savingThrows.dexterity.proficient ? this.proficiencyBonus : 0)

initiative: this.savingThrows.dexterity.modifier

Stat:

proficient: character.SomeUniqueClass.statProficiencies[this.name]

modifier: Math.floor(this.value – 10)

racialBonus: character.race.statBonuses[this.name]

total: this.value + this.modifier + racialBonus + (this.proficient ?

character.proficiencyBonus : 0)

name: passed down from initialization in Stats

Senses:

perception: 10 + character.stats.wisdom.modifier

investigation: 10 + character.stats.intelligence.modifier

insight: 10 + character.statrs.wisdom.modifier

Skill:

proficient: character.SomeUniqueClass.skillProficiencies[this.name]

modifier: this.relevantStat.modifier + (this.proficient ? character.proficiencyBonus : 0)

name: passed down from initialization in Skills

Health:

max: character.SomeUniqueClass.initialHealth + subsequent health increases defined in

character.SomeUniqueClass.healthIncrease

## AT5 | Conditionals

NA. See simplifications.