|  |  |
| --- | --- |
| **A01** | **Create a New Hero** |
| **Goal** | Create a new Hero and save it. All new Heroes are young adults of Peasant Klass and actor-selected Race. After displaying the resulting Hero, the actor Saves the Hero in the dormitory, or Cancels. |
| **Actor** | Player |
| **UXs** | [UX00\_PortalOpeningView.docx](../../Adv_UserExperience/UX00_PortalOpeningView.docx)  [UX01\_NewHeroInputPanel.docx](../../Adv_UserExperience/UX01_NewHeroInputPanel.docx)  [UX02\_HeroDisplay.docx](../../Adv_UserExperience/UX02_HeroDisplay.docx) |
| **Pre-Conditions** | User is on Main Action page with Create Hero button enabled. |
| **Post-Conditions** | * Hero is created and attributes, inventory, skills, and magic-related data are displayed. * Hero is saved into the Dormitory with other Heroes. |
| **Invariants** | * Registries for Heroes, Race, Occupations, associated Skills, and Inventory Items are available. * List of hair color options are available. * Dormitory exists for saving the new Hero and ensuring uniqueness. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Detailed Description** | | | | |
|  | **Actor Actions** | | **System Response** | |
| **1** | Requests to create a new Hero. | | Retrieves the data for selections and displays the new hero input screen (UX01) of user input options: name of Hero, gender, Race, and hair colors. | |
| **2a** | Enters the new Hero’s desired name (required). | | Validates the Hero’s name (required), which must be unique in the Dormitory; details are in Input section. | |
| **2b** | Optionally, changes gender from Male [default] to Female. | | Gender affects the Hero’s traits. | |
| **2c** | Optionally, changes hair color from bald [default] to other color. | | Different hair colors affect the Hero’s physical description. | |
| **2d** | Optionally, changes Race from Human [default] to one of the non-Human races | | Each Race has abilities and supporting values specific to that Race. | |
| **3** | SUBMITs the input data to create the new Hero. | | If Hero’s name is non-null and within length limits, creates the Hero from the given data. The algorithm uses the 4d6-1 algorithm for the 6 prime traits of the Hero.  The actual rules for generating the character is given in a supplemental document *Generating the Character*, available in the use case folder. | |
| **Inputs:** | | | NAME (no default; max limit = 45 characters, including white space), GENDER [male], HAIR\_COLOR [bald], RACE [Human], KLASS [Fighter] | |
| **Outputs:** | | | Hero Nameplate: NAME, GENDER, RACE, KLASS  Attributes: XP (0), LEVEL (1), HP (Peasant = 10), OCCUPATION (randomly selected), HUNGER STATE (FULL); AC (10 before mods and inventory), SPEED (AP and height adjusted), HEIGHT (feet and inches; race and gender adjusted), WEIGHT (lb; race and gender adjusted); GP (Peasant default), GOLD\_BANKED (0.0), WEIGHT\_CARRIED (calculated from inventory); MAX\_LANGS Knowable (calc from INT), LANGUAGES KNOWN (Common and race language); PHYSICAL DESCRIPTION (inferred from physical traits and Charisma)  Inventory(Item List): Category, Name, Quantity, Weight, [Occupation Kit], and total inventory weight  Skills: Literacy (based on INT), Occupational Skill (based on Occupation), [Racial Skills], [Klass skills] | |
| **Notes:** | | | 1. The current 7 race options and associated racial skills are: Human, Dwarf, Elf, Gnome, Half-Elf, Half-Orc, and Hobbit. 2. Hero’s height and weight are used to find a body type then CHR is used to determine if the description should be positive or negative (e.g. skinny or lithe; squat or stocky). Description also includes a note about racial appearance (e.g. pointed ears for elf; snout-like face for Half-Orc.) 3. There is never a need for the Player to edit their Character, so Edit Hero does not exist as a Use Case. *Adventurer* updates the Hero as the Player performs actions, e.g., buying inventory items, or gaining HP. The player can Delete his Hero though. 4. Each Hero is an object characterized by personal attributes, Race, Klass, and Inventory component, and is associated with other supporting objects, e.g., Item. Each of these components initializes their own aspects of the Hero. | |