

The Better Unplanned NPC Generator

Developer Guide

Overview

The Better Unplanned NPC Generator (or BUNG) is a tool for dungeon masters and game masters alike to create interesting and unique NPCs that fit in to your game setting for your players to interact with on the fly. It also includes the ability to easily save the details of those random NPCs that your players decide to adopt as their new best friends, so you'll never forget who that random person on the street was that your players are asking about 3 weeks later!

Please review the User Guide for full details on how to install and run the application, as well as the NPC generation behaviors. The user guide also contains important details on how the CSV files containing race details, lists of possible names, and others are expected to be formatted.

Features

Implemented:

Choose Preset

On launch, the application will load the Default preset profile, which includes all possible races with an equal chance to be chosen from by the generator. The user is allowed to make changes to the default profile as it suits them, though.

All other files in the presets folder will be populated as options to choose from. Choosing a preset will reset the current Race selection and repopulate the list with the new list.

Generate NPC

By default, all options for NPC generation are set to "Any". "Any" prompts the generator to make a random selection on that characteristic for the user.

The user may also choose to manually set any or all of the options here. The generator will almost always use manually set values and not generate anything for that characteristic.

Clicking Generate NPC will take the currently chosen options and create an NPC using the criteria provided.

See the NPC Generation Flow on page XXXX for additional details on how an NPC is created.

Save / Copy NPC

Users can export the current NPC to a raw text file or copy the text to the clipboard.

NPC History

When an NPC is generated, the corresponding NPC object is added to a list and displayed in a listbox below the NPC panel.

The current NPC will always be at the top of the list.

When the list is at the configured maximum (currently set to 10), the oldest NPC on the list will be dropped.

Clicking on any NPC in the list will load it into the NPC panel and put it on the top of the list.

Not Implemented:

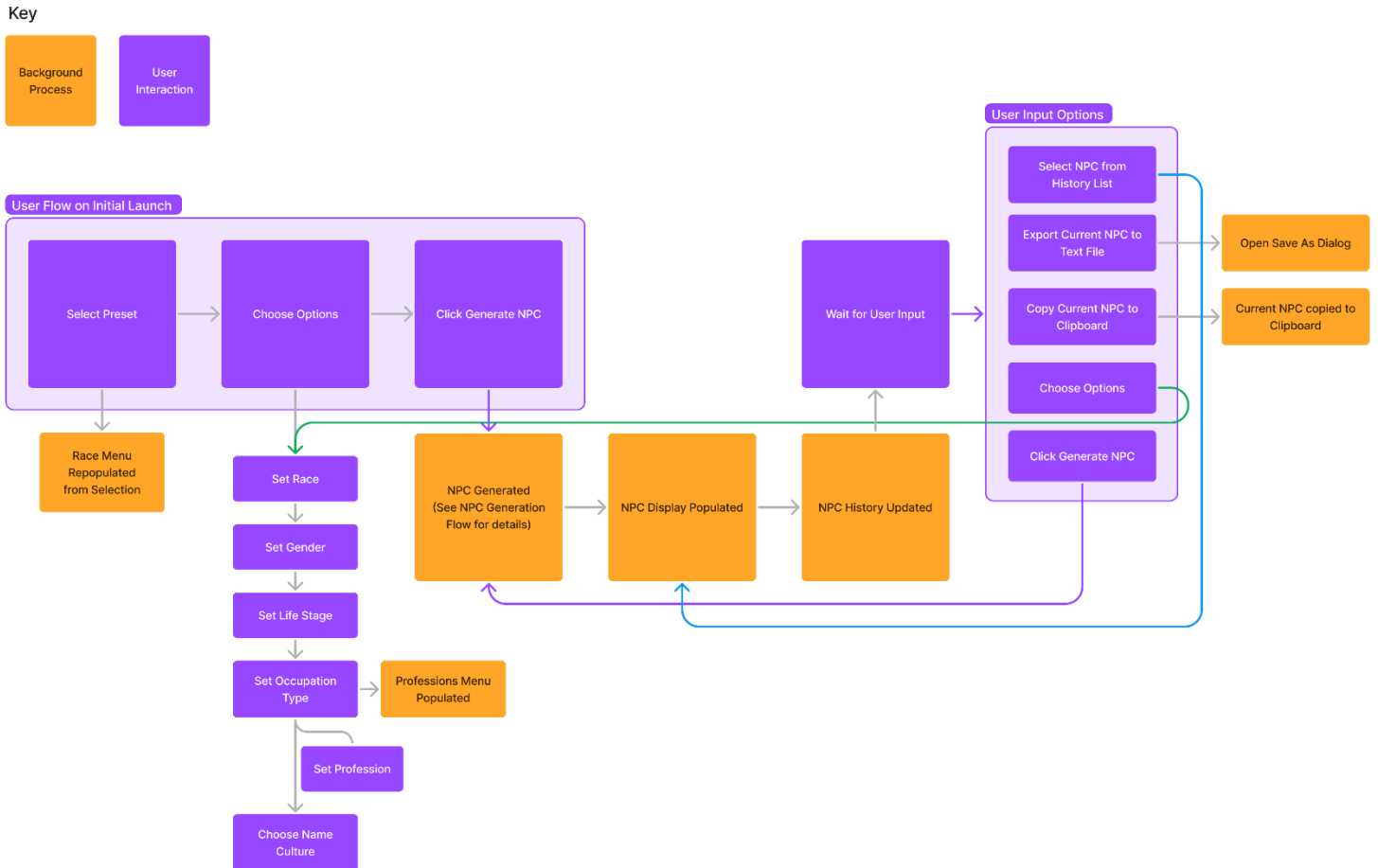
Create New Generation Preset

There is currently no in-UI method to save or modify the presets. Instructions on creating presets in CSV format are included in the user guide in the meantime.

Toggling Characteristics On/Off

The ability to toggle which characteristics get generated and shown in the NPC panel is planned for the future.

User Interface Flow



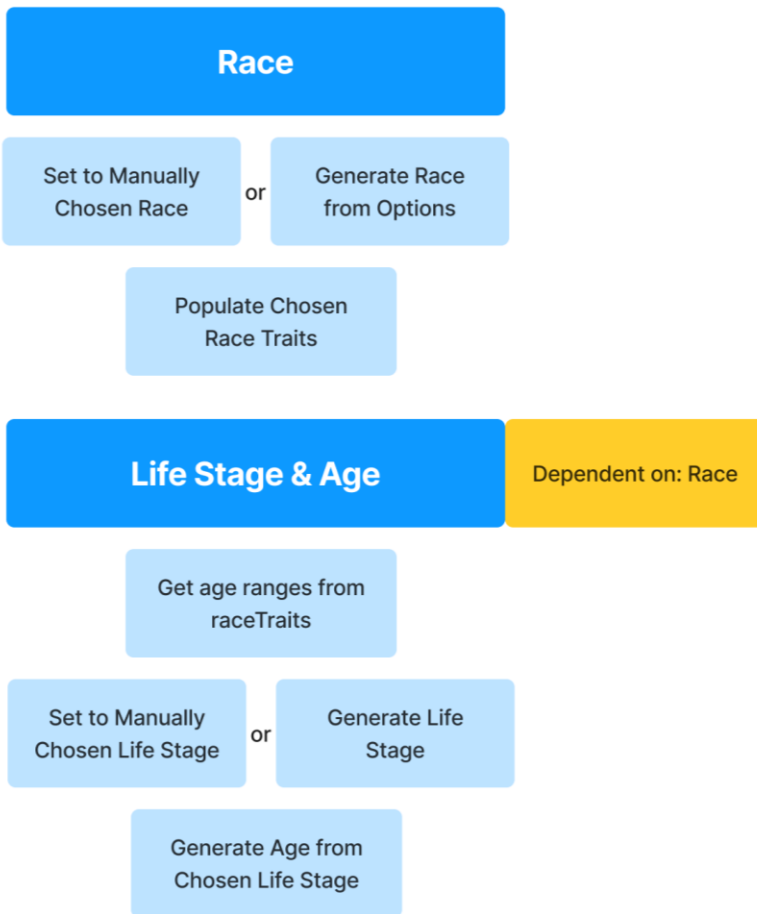
On initial launch of the application, the user will need to select a preset (or keep the default selection), configure options (or keep the default choices), and then click the “Generate NPC” button.

See the “NPC Generation Flow” for details on the NPC generation logic.

When at least one NPC has been generated, the user can save the NPC to a text file or copy the text to the clipboard. When at least 2 and up to 10 NPCs are in the NPC History box, the user can recall previously generated NPCs by choosing one from the list that’s not currently selected.

NPC Generation Flow

Race & Life Stage



NPC generation begins with determining the race. If a value was manually selected by the user, it will use that race. Else, it will randomly choose a race from the possibilities and use weighting, if configured.

Once a race is determined, it will fetch an array of raceTraits for the race.

All age ranges for the given race are loaded, then the program determines the life stage. If a value was manually selected by the user, it will use that life stage. Else, it will randomly choose a life stage from the given possibilities: "Child, Adolescent, Young Adult, Adult, Elder".

Once the life stage is determined, the exact age of the NPC can be generated. The value that corresponds with the current life stage represents the lower bound for the age range. The upper bound is fetched from the next life stage up, or in the case of Elder, from the Max Age value.

Occupation



Occupation represents the combined values of the Industry (called Occupation Type in the UI) and the Profession. Special logic is in place to accommodate determining the occupation of a Child, Adolescent, or Elder.

When Industry and Profession are both specified, but life stage was not, to prevent the generator from assigning an occupation to a Child without the user explicitly asking it to, it checks if Child was randomly assigned before setting the occupation to the provided industry and profession.

Any other condition resulting in Child as the life stage will give it an occupation of "None".

When Profession is set to Any, Adolescents are given a 50% chance to be assigned an occupation.

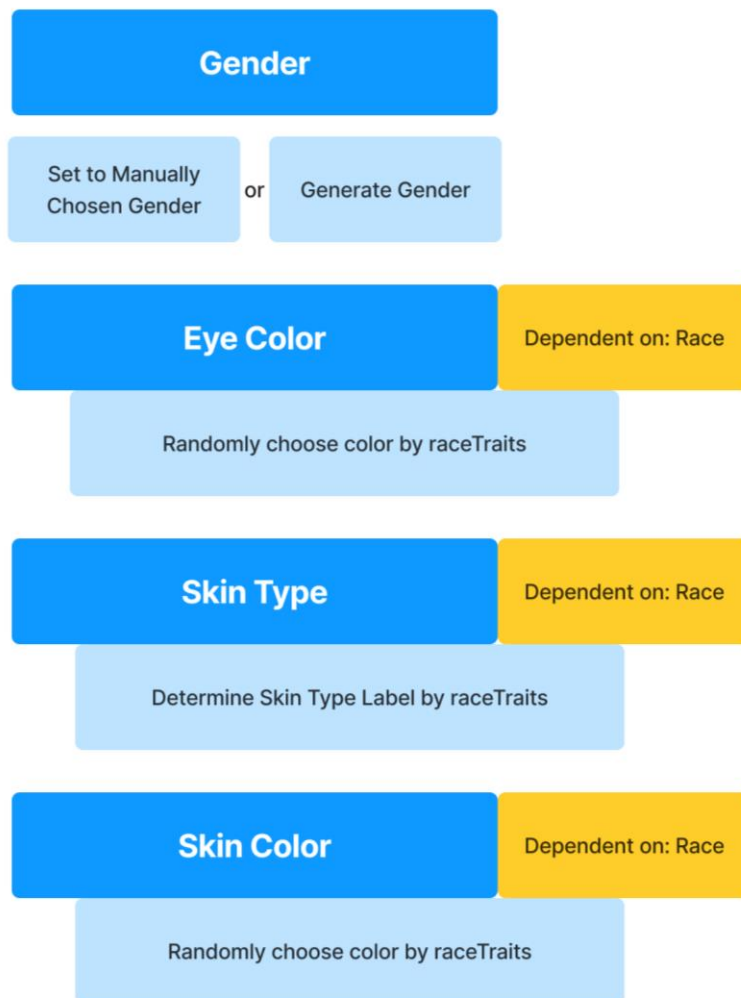
If Industry was not specified, it will generate an industry from the available options in the occupations folder.

Profession is then generated from the chosen industry.

After the occupation has been determined, if the life stage is Elder, regardless of whether it was manually chosen or generated, there is a chance that the "Retired" qualifier will be added to the profession.

Only the Profession is displayed in the NPC Display Panel to save on screen space.

Gender, Eye Color, Skin Type, and Skin Color



Gender is determined either with the user specified value or by randomly selecting one of the possibilities if not specified. A gender code is determined from the first letter of the selected gender, and this code is used to determine name gendering and height deviation.

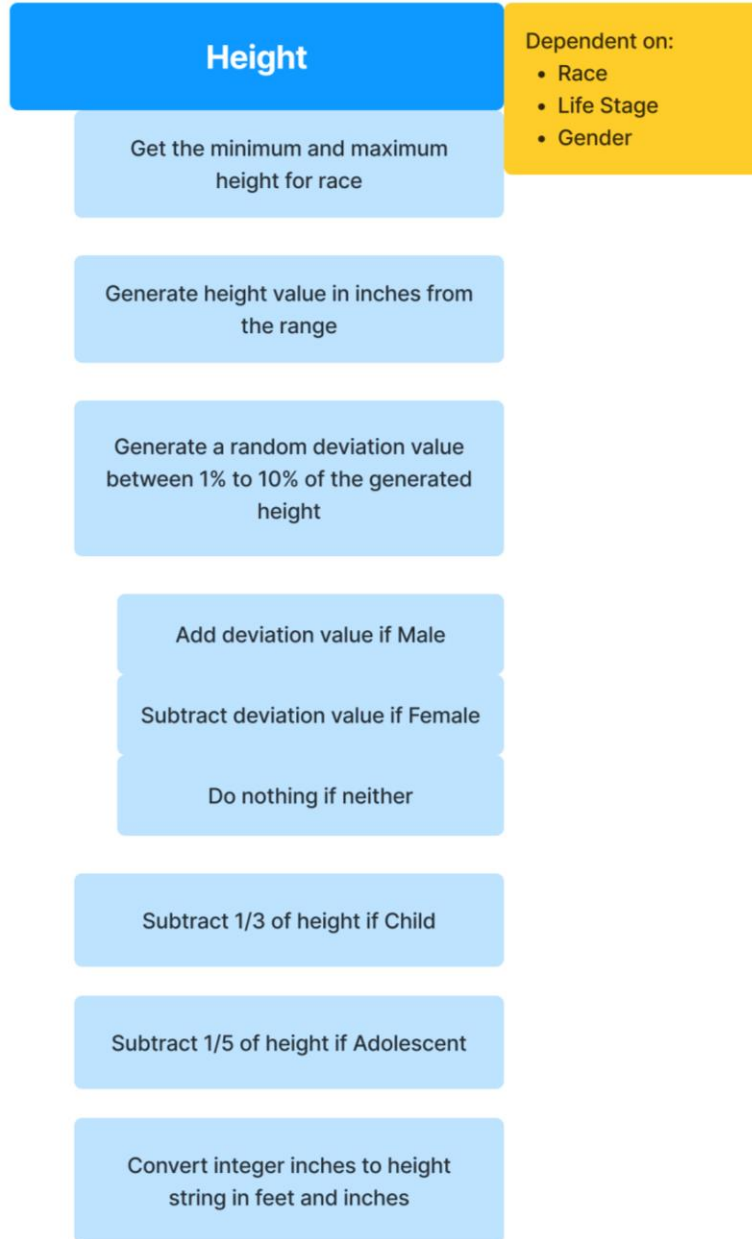
Note: in a future update, would like to add the code to the Genders file so custom genders can be specified as Masculine or Feminine without having to start with the letters M or F.

Eye color is randomly selected from the possibilities configured in the Races file.

The value of Skin Type informs how the Skin Color is labeled in the NPC Display Panel. This is not randomly determined.

Skin color is randomly selected from the possibilities configured in the Races file.

Height



NPC height considers the race, the life stage, and the gender of the NPC.

The minimum and maximum values are retrieved from the raceTraits of the chosen race, and are represented in inches. The function immediately gets a random number between the minimum and maximum height to get a base height.

A random deviation value based on the generated height is then generated. This is set to be between 1% to 10% of the generated height. This deviation value is added to the base height if the gender of the NPC is male, and it is subtracted if the NPC is female. Nothing happens if neither male or female are the chosen gender.

Note: In a future release, the user will be able to override this by specifying that the NPC they're generated is taller/shorter and disregard default gendered behavior.

After gender deviated is calculated, it then considers the life stage. If Child, it will remove 1/3 of the base height value. If Adolescent, it will remove 1/5 of the base height value. This is to simulate younger characters not being fully grown yet.

Body Type, Accents, and Name



The body type is currently selected from a hard coded set of options. This may later change to come from a standalone CSV list or become another racially specified trait.

Accents are racial specific qualities that don't apply to all races. Up to 3 can be configured per race at this time. The Accent Type column determines the label, and the Accent Trait determines the list of possible values for that Accent.

Once all physical traits have been determined, the generator can select a name for the NPC. What name lists the generator pulls from is determined by the Name Culture option chosen by the user.

Any Name Culture will load all names from all name lists in the names folder.

Common Name Culture will load only the Common name lists.

Traditional Name Culture will load the name lists for the race specified Name Traditions.

Once the lists of possible first and last names are populated, the first and last name are randomly selected.

Known Issues

Minor:

There is no way to restore default settings after making changes to the default preset without reinstalling the application.

The application doesn't scale properly on all screens.

Major:

Not all aspects of the program which are dependent on CSV files are protected against missing data/files, making it easy for the user to create a situation that puts the application in an inoperable state.

The way race traits are extracted from the Races CSV is prone to index errors.

Inefficiencies:

As the program expands from test data to full data sets for names and races, the time it takes for the generator to parse through the data will increase and could cause more noticeable wait times, especially when the user is using the default profile without modification and leaving the Race option set to "Any". Also leaving the Name Culture choice set to "Any" when more and bigger name lists are configured.

Future Work

Some areas targeted for future improvement include:

Redesigning the UI to make it scale better

Add ability to create and edit presets within the UI

Add ability to edit generated NPC characteristic within the UI

Add ability to toggle which characteristics the user wants to see in the UI

Improve gender customizability and allow user to situationally remove gendered generation logic.

Move hard coded strings into external settings file(s).

Convert Races.csv to a JSON file instead to simplify referencing fields in code.
May possibly do the same to most of the CSV files.

Make sure all instances of file loading are protected from null pointers and that users are properly informed when there are file related problems.

Make the "Any" name culture option not the default and offer a brief performance warning when using the option.