tbrpggepp User Manual

Tutorial and Troubleshooting Guide

JAMES DENOVAN

Kinross Wolarai School

Contents

1   Introduction

tbrpggepp (Text-Based Role-Playing Game Graphical Editor Plus Plus) is a graphical desktop program for creating the narratives of branching text-based computer games. The program works on both Linux and Windows systems, and has two requirements: a working installation of `Python 3.7` and the Python library `PyQt5`, which you can install by entering the following command in a terminal:

```
pip install PyQt5
```

Within tbrpggepp, you create the game-world that the player character navigates throughout their quest. You write the words that the player sees, the enemies and scenarios they encounter, and how they navigate your world. You can write a game about anything: from exploring ancient dungeons, to escaping derelict space stations. The only limit is your imagination.

To run the program, simply double-click the `mainMenuForm.py` file and Python should do the rest.

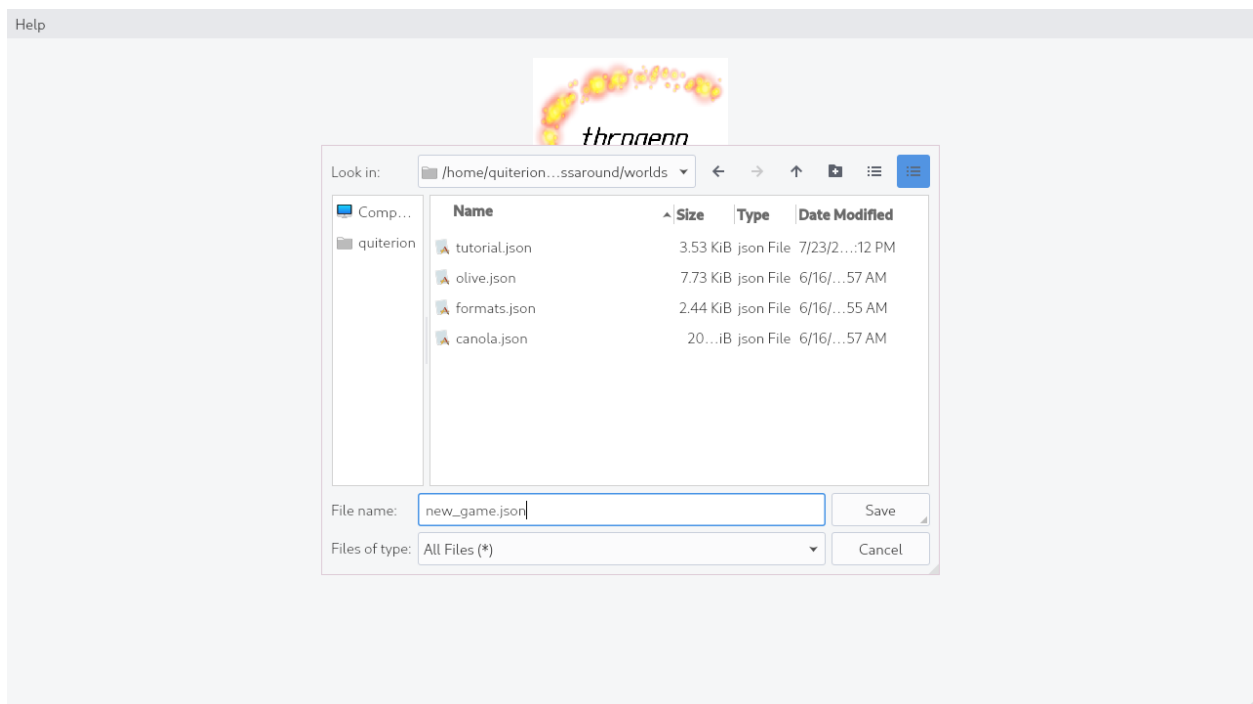# 2 Using the Program

2.1 The Introduction Form
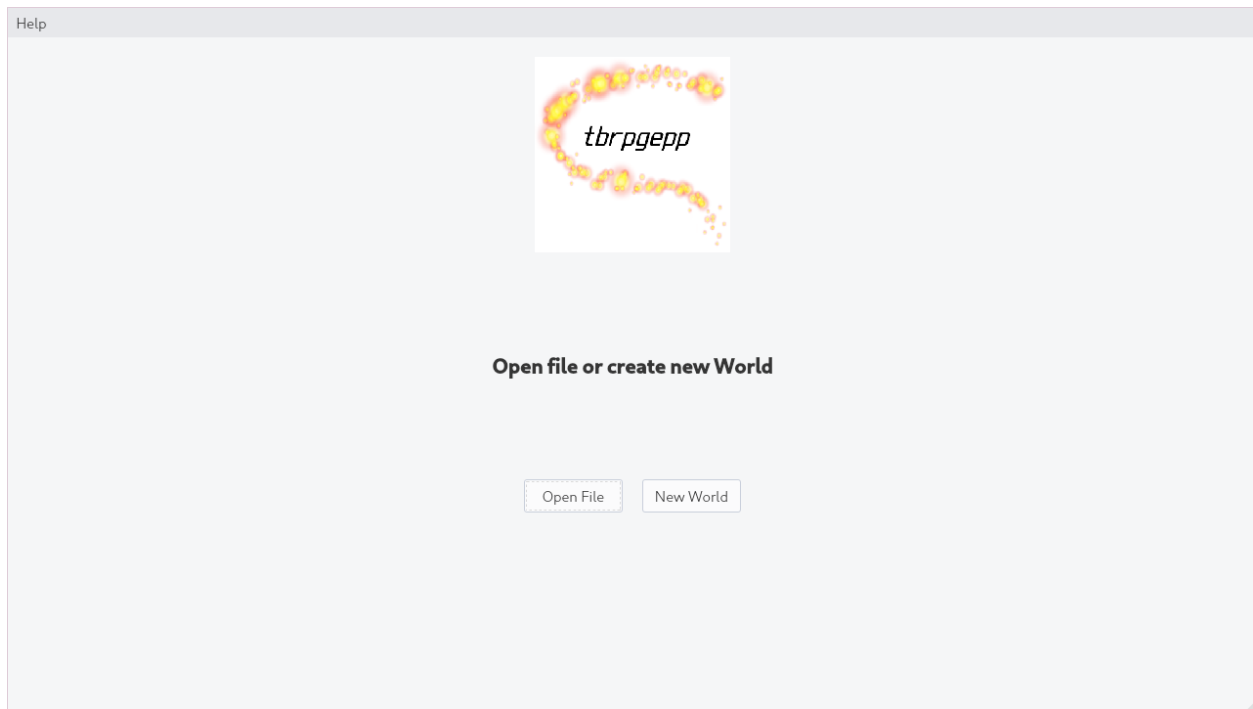
This is the first form you see when you execute the `mainMenuForm.py` file. You are presented with two options: open a preexisting file containing a `tbrpggepp` game world, or create a new one.

2.1.1   Creating a new game-world

Upon clicking "New World", a dialog will appear prompting you to create a new file. You may create this file in any directory you please, but **note that game-world files must have a "`.json`" extension.** After creating a file, the Introduction form will close and take you to the Main Menu form.

2.1.2   Opening a preexisting game-world file
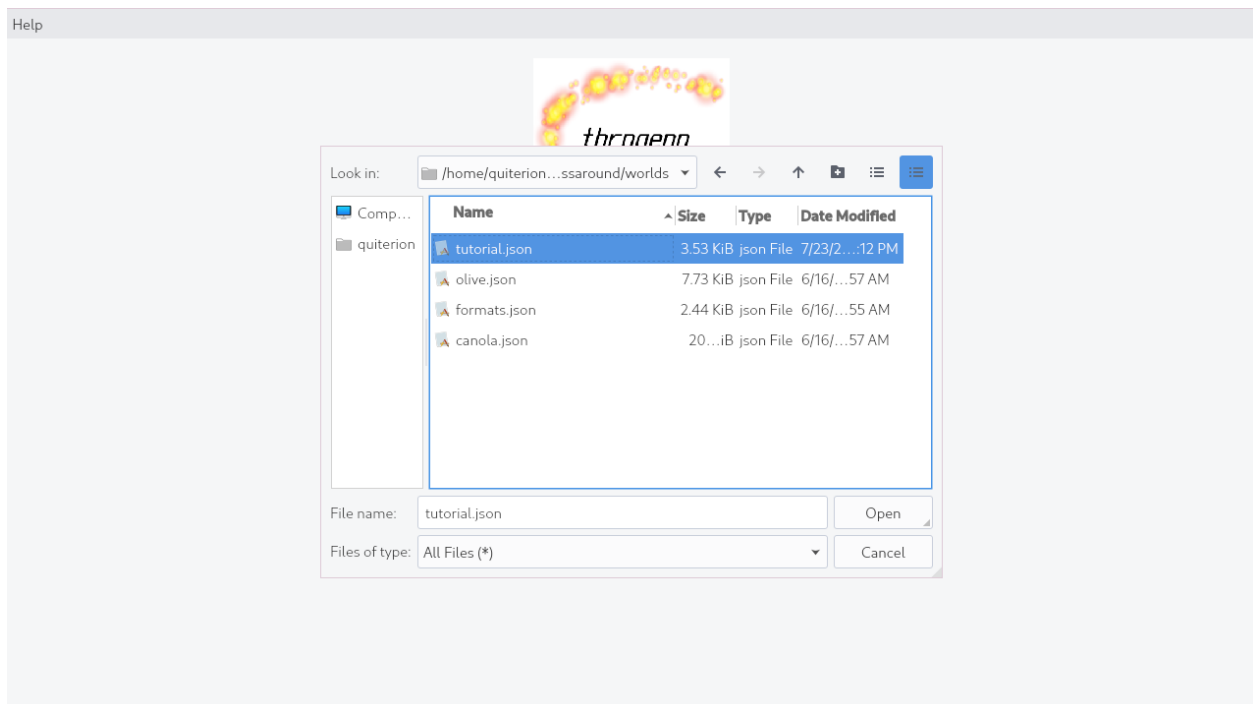
Upon clicking "Open File" an identical dialog will appear, this time prompting you to select a preexisting file. **Note that a valid game-world file must end in a "**`.json`**" extension.** After selecting a file, the Introduction form will close and take you to the Main Menu form.

2.2   The Main Menu Form
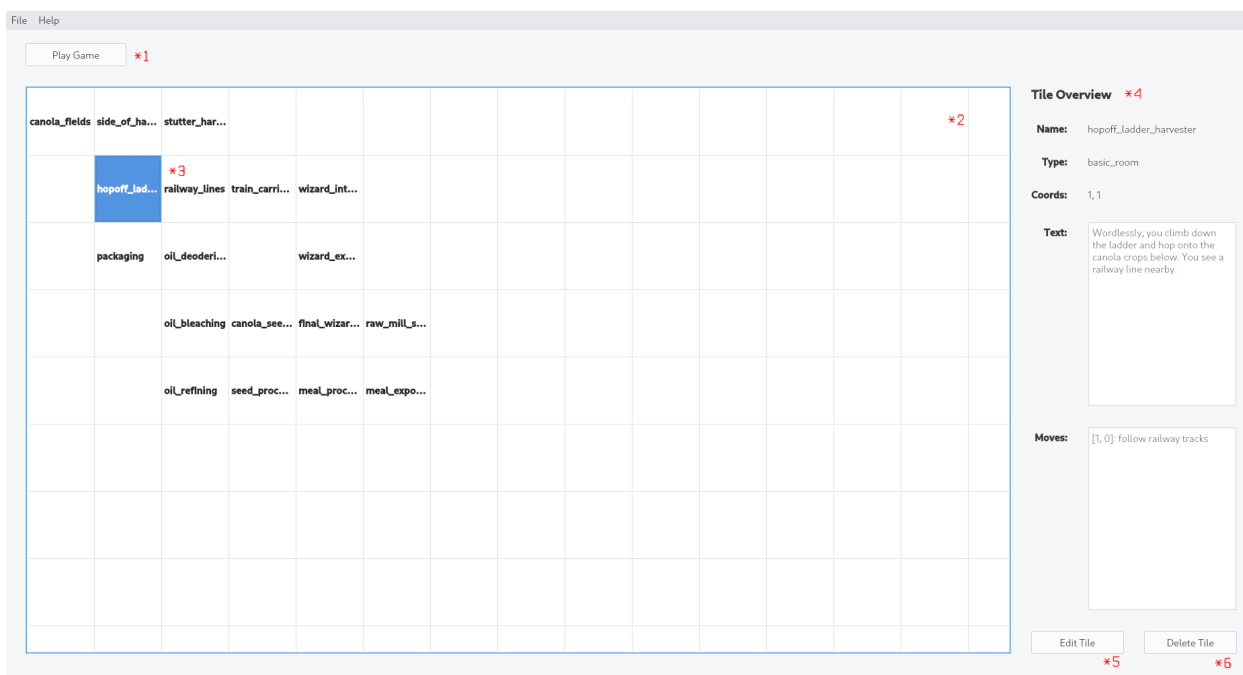
This is the main form of `tbrpggepp`. The elements in this screenshot are labelled as follows:

1. "Play Game" button. This opens a terminal in a new window running the text-based game, allowing you to play out your narrative.

2. The game-world grid. This grid essentially provides a top-down view of the "world" for your text-based game, consisting of hundreds of invisible, empty "tiles" which you may edit and populate with data for the player to interact with. Within this screenshot we see many tiles have already been created and populated with data, thus creating a short text-based adventure. **Note that at the beginning of every game the player always starts at the topmost leftmost tile, which has coordinates of 0,0.**

3. You may click on a populated or empty tile to interact with it or view its contents. In this screenshot we see a populated tile has been selected, displaying its contents in the sidebar and allowing the user to either edit it or delelete it by clicking the appropriate buttons.

4. The sidebar ie. Tile Overview. This presents a read-only view of the contents of the selected tile, including the following elements:

   - Name: the unique identifier for a tile, unseen by the player but cannot be reused.
   - Type: the category of tile, which defines how the user interacts with it. Categories include `basic_room` (simply displays text), `enemy_room` (contains an enemy for the player to fight), `item_room` (contains an item that the player picks up), and `victory_room` (identical to a basic tile, except the game ends after the player enters it).
   - Coords: the X and Y coordinate values of the tile (in that order), representing the tile's location within the game-world.
   - Text: the words that the player sees upon entering a tile of any type for the first time.
   - Moves: an overview of the methods in which the player can navigate out of the selected tile. Displays a "coordinate delta" and "label" for each move. The label is simply what the player enters to conduct the move. The "coordinate delta" mathematically represents the change in coordinates applied to the player after conducting the move. For example, a player residing in coordinates 1,1 with a move of [1,0] applied

to them would end up in 2,1; which is one unit to the right on the game-world grid. For another example, applying the move [0,-1] on a player residing in 2,6 would move them to 2,5; which is one unit upwards on the game-world grid.

5. "Edit Tile" button. Clicking this opens the Edit Tile form, allowing you to edit the contents of the currently selected tile. If an empty tile was initially selected then a new tile will be created at its coordinates.

6. "Delete Tile" button. Clicking this will turn the currently selected tile into an empty tile, handle with care!

7. An additional element is the "file" menu option in the upper left: this allows you to re-open the Introduction form and open/create a different world file. The "help" option simply links to relevant pages of this document.

2.2.1 Playing the Game

As stated above, clicking the "Play Game" button in the Main Menu form will open a terminal in a new window that is running the text-based game. A screenshot of one such terminal is provided below:

When the game program begins, it gives you the decision to choose any of the other game-world files within the directory of your `tbrpggepp` world file. After loading the desired game-world file, we see the player met with the introductory text of the tile at coordinates 0,0; followed by a prompt where the player may either:

- "`look around`": displays the text in the "Description" field shown in the Edit Tile Form section.

- "`check health`": displays the current HP (Health Points) value of the player

- "`check inventory`": displays the current inventory of the player

- "`go east`": evidently one of the moves created for the tile at coordinates 0,0.

Other scenarios display different prompts, such as when one has consumable health items in their inventory or are engaged in a battle with an enemy. This "tutorial" questline is provided alongside `tbrpggepp`, which you may want to attempt in order to gain familiarity with `Adventure Quest`'s style of gameplay.

2.3   Edit Tile Form

Within this form you can edit the properties of individual tiles. The visible UI components for editing properties are as follows:

- Name: the unique identifier for a tile, unseen by the player but cannot be reused.

- Type: the category of tile, which defines how the user interacts with it. Categories include `basic_room` (simply displays text), `enemy_room` (contains an enemy for the player to fight), `item_room` (contains an item that the player picks up), and `victory_room` (identical to a basic tile, except the game ends after the player enters it).

- X and Y Coords: the X and Y coordinate values of the tile , representing the tile's location within the game-world. Both the X and Y coordinate values must be greater than or equal to 0.

- Narrative Text: the words that the player sees upon entering a tile of any type for the first time.

- Description: the text the player sees when they enter the "`look around`" action. Should contain a more detailed description of the environment within the tile.

- Return Text: the text the player sees when they re-enter the tile after having visited it before. For example, if this were an `enemy_room`, the Return Text should contain something like "a defeated enemy lies on the floor".

- "Edit Element": This is greyed out because the tile is currently selected to be a "`basic_room`" type. Otherwise, this button would either open the Edit Item or Edit Enemy form, depending on whether the tile is of type "`item_room`" or "`enemy_room`" respectively. The text label of the button changes correspondingly as the tile type changes as well.

- "Edit Moves" button: this opens the Edit Moves form, allowing you to edit the list of moves for the tile.

- "Cancel" button: this will close the form and all sub-forms without saving any changes in the game-world..

- "Save" button: this will close the form and all sub-forms while saving all changes in the game-world.

2.4   Edit Item Form

This is the form that is opened when you click the "Edit Item" (otherwise "Edit Element") button within the Edit Tile form, when the tile is of type "item_room".

- Name: this is the name of the item within the game, and it will be seen by the player.

- Dmg Type: the damage type of the item; in more complicated narratives this feature can be used to enrich game-play. A weapon of a specific damage type interacting with an enemy of the same damage type deals potentially massive damamge.

- HP Effect: the maximum amount of damage inflicted by the item if it is a weapon, or the amount of player health the item heals if it is a consumable healing item. Note that the maximum player HP is 10.

- Quantity: the number of items contained in the tile overall. Note that an item_room tile may only contain one type of item, but may possess multiple items of that type.

- "Is Healing" checkbox: Determines whether the item is a weapon that inflicts damage, or is a consumable healing item instead.

- Effect text: the text displayed to the player after consuming a the item, only relevant if "Is Healing" is checked.

- Cancel: closes the form without saving changes.

- Save Item: closes the form while saving the item to the parent Edit Tile form.

2.5   Edit Enemy Form

This is the form that is opened when you click the "Edit Enemy" (otherwise "Edit Element") button within the Edit Tile form, when the tile is of type `enemy_room`.

- Name: this is the name of the enemy within the game, and it will be seen by the player.

- Dmg Type: the damage type of the enemy; in more complicated narratives this feature can be used to enrich gameplay. A weapon of a specific damage type interacting with an enemy of the same damage type deals potentially massive damamge.

- Max HP: The maximum number of health points carried by the enemy. Note that the player character starts with maximum health at 10 HP.

- Attack Dmg: The maximum amount of HP the enemy can detract from the player during one turn of a battle.

- Cancel: closes the form without saving changes.

- Save Item: closes the form while saving the enemy to the parent Edit Tile form.

2.6 Edit Moves Form

This is the form that is opened when you click the "Edit Moves" button within the Edit Tile form.

- Move Label: the name of the move as seen by the player. The move is executed after the player enters the move label into the game's text prompt. The label of a move, like the name of a tile, is a unique identifier and reusing a label will result in overwriting the move that previously shared the label.

- Move Direction: the "coordinate delta" of the move which as explained in section 2.2, mathematically represents the change in X,Y coordinates which is applied to the player after they execute the move.

- "Clear" button: erases the current list of moves for the tile being edited. If this is clicked accidentally, simply cancel the Edit Moves form and reopen it from the Edit Tile form.

- "Add Move" button: adds the new individual move (as defined by the label and direction specified in the above input fields) to the tile's overall list of moves.

- Cancel: closes the form without saving changes.

- Save Moves: closes the form while saving the updated moves to the parent Edit Tile form.

- On the right-hand side of the form there is a large read-only list box which displays the list of moves currently assigned to the tile being edited.

With all these forms combined, you're all set to start creating text adventure games!

3 Troubleshooting Guide

3.1 Issue 1: Invalid world map file!

- This issue is caused simply by not selecting a valid json file. When creating/opening a world-map file, please ensure that the filename ends with ".json".

3.2 Issue 2: No world file selected!

- This issue is caused when one attempts to modify the game-world grid when infact no file has been created or loaded for the Main Menu form. This could have been a result of manually closing the Introduction Form before selecting/creating a file. Please resolve this by using the "file" menu in the top-left of the Main Menu Form to reopen the Introduction form and create/open a valid world-map file.

3.3 Issue 3: Tile name already used!

- Despite the player within the text-game never seeing the given names of tiles, they are essential in that they uniquely define the tile both to the game-designer and the game program internally. Therefore no two tiles can share the same name without overwriting another tile and causing loss of data. tbrpggepp has resolved this temporarily by renaming the newly updated tile with a randomly generated name, which you may rename again to a unique, more human-readable alternative at your leisure.

3.4 Issue 4: Tile name left blank!

- For reasons stated immediately above, the presence of a unique name per each tile is essential to the function of tbrpggepp. Therefore you must enter a unique name into the Edit Tile form before you can save your work.

3.5 Issue 5: The program crashed!

- Whoops! Please tell me what you were doing immediately before the crash and how you were doing it.