```
resources/ ·····················································Asset files used in program
data/
├──── database/
│         ├───────── migrations/ ················Database migrations that track changes to table
│         │                                       schema
│         └────────database.db
├──── helpers/
│         ├────────bitboard_helpers.py
│         └──── ...
├──── shaders/
│         ├────────classes/ ····················Python classes for each fragment shader
│         ├────────fragments/ ···············Fragment shader code
│         └────────vertex/ ···················Vertex shader code
├──── widgets/
│         ├────────bases/ ·····················Abstract classes for widgets
│         └──── ...
├──── states/
│         ├────────game/
│         │          ├────────components/ ····Collection of classes involved in running of game screen
│         │          ├────────cpu/ ··········Contains CPU engine classes and ancilliary helper classes
│         │          ├────────mvc/
│         │          ├────────game.py
│         │          └────────widget_dict.py ·Initalises a dictionary containing widgets for the game screen GUI
│         └──── ...
├──── utils/
│         ├────────assets.py ···················File containing dictionaries of external loaded assets
│         ├────────constants.py ·············File containing constants used by multiple files
│         ├────────enums.py ··················File containing Python enum classes
│         └────────event_types.py ···········File containing screen-specific EventType enum classes
├──── app_data/ ·····················JSON files used to store settings and configurations
├──── components/ ·····················Collection of useful classes used in multiple areas
├──── managers/ ·····················Collection of classes that manage game systems
├──── main.py
├──── setup.py
├──── control.py
├──── constants.py
├──── loading_screen.py
└──── windows_setup.py
```