Design Proposal

Project Proposal

- **Project Description**

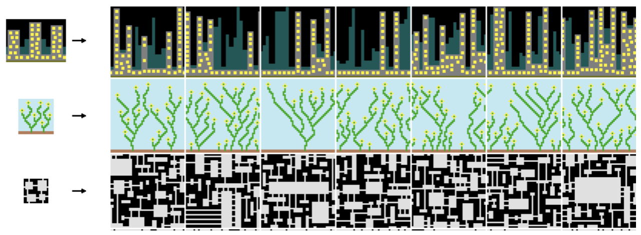
  Cube Spacer

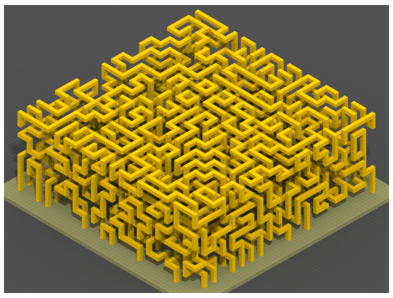
  This project is a constraint based 2.5D world generation. It involves a set of tiles in the form of cubical 2.5D isometric geometry and the Wave Function Collapse algorithm (WFC) (Gumin,2016) to place tiles according to the tile set's connectivity rules.

- **Competitive Analysis**

  Existing applications are mostly in 2d and the very few that are 3d are implemented in game engines such as Unity or Blender. Most of the implementations are in C# and with higher dimensions performance is known to be an issue.

  This project seeks to implement a higher dimension world generation in python within the cmu graphics framework. This would involve resolving performance issues by scaling the setting and constraints. Also it will provide a novel 3d tile set that can serve as a maze or generate interesting 3d patterns.



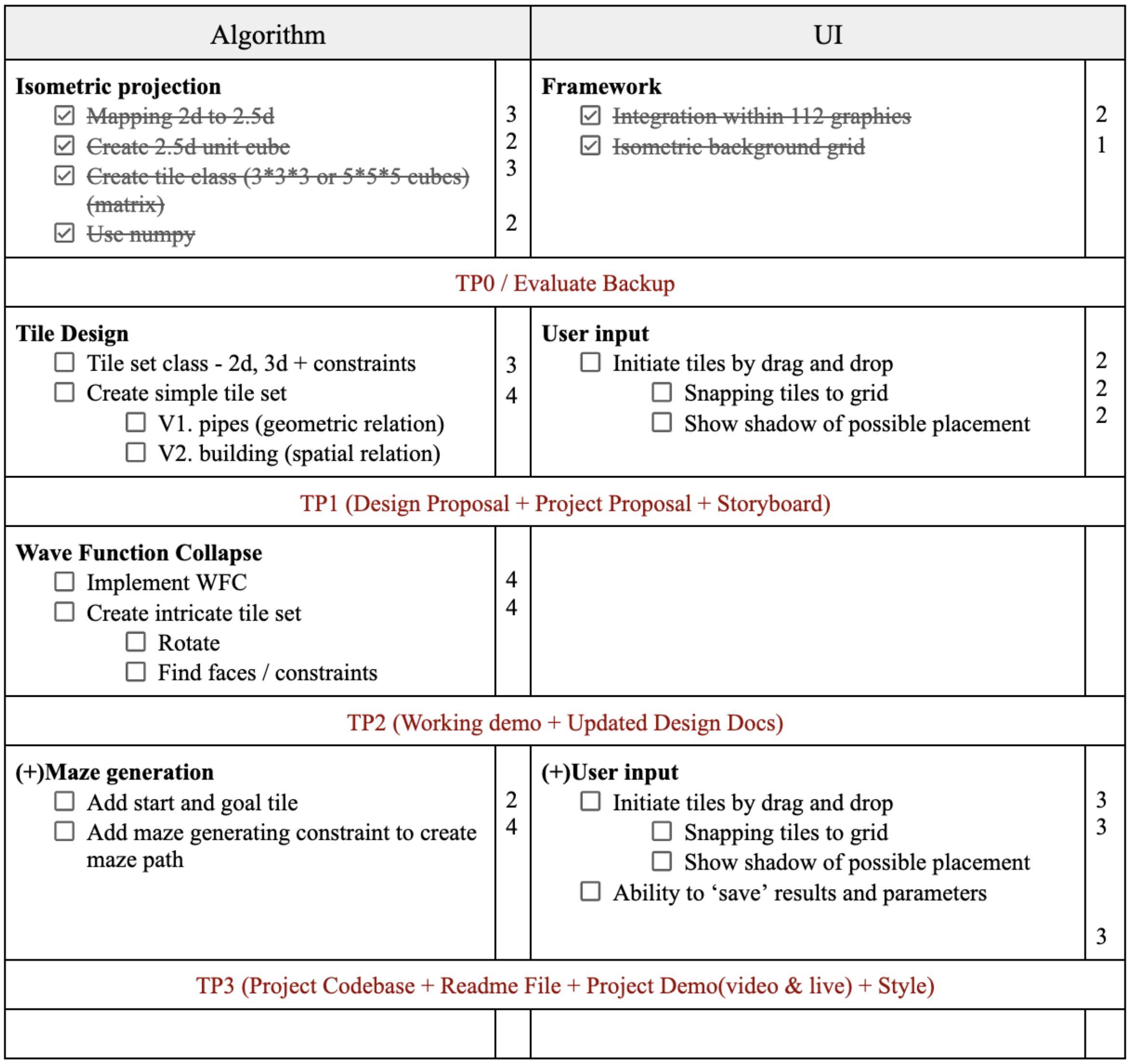


- **Structural Plan**

main.py	onAppStart	bring in settings
		store user action states
	onKeyPress	functions to rotate tile / board
	onMousePress	functions to select tile from tile set and place tiles on board
		isTileSet, isTileLegal
	onMouseMove	functions for play to hold the tile and rotate
	redrawAll	drawBoard, drawTileSet, drawMovingTile
	Isometric Functions	cartToIso, isoToCart, getCornerPointsIsoRect
		drawIsoRect, drawIsoGrid, drawCartGrid
	Tile Functions	drawTileOnCanvas, drawTileSet, drawTileBound
		placeTileOnBoard

settings.py	setting used in main and tile class	returns dictionary where key: margin, rows, cols, cubeSize, cubeDim, tileDim, levels value : input values of setting
tile.py	TileSet and Tile class	includes tile map, adjacency constraints and board/pixel location.

- **Algorithmic Plan / Timeline Plan**



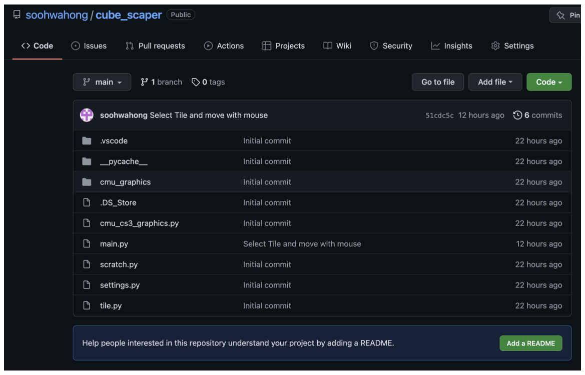
Algorithm		UI	
Isometric projection ☑ ~~Mapping 2d to 2.5d~~ ☑ ~~Create 2.5d unit cube~~ ☑ ~~Create tile class (3*3*3 or 5*5*5 cubes) (matrix)~~ ☑ ~~Use numpy~~	3 2 3 2	**Framework** ☑ ~~Integration within 112 graphics~~ ☑ ~~Isometric background grid~~	2 1
TP0 / Evaluate Backup			
Tile Design ☐ Tile set class - 2d, 3d + constraints ☐ Create simple tile set ☐ V1. pipes (geometric relation) ☐ V2. building (spatial relation)	3 4	**User input** ☐ Initiate tiles by drag and drop ☐ Snapping tiles to grid ☐ Show shadow of possible placement	2 2 2
TP1 (Design Proposal + Project Proposal + Storyboard)			
Wave Function Collapse ☐ Implement WFC ☐ Create intricate tile set ☐ Rotate ☐ Find faces / constraints	4 4		
TP2 (Working demo + Updated Design Docs)			
(+)Maze generation ☐ Add start and goal tile ☐ Add maze generating constraint to create maze path	2 4	**(+)User input** ☐ Initiate tiles by drag and drop ☐ Snapping tiles to grid ☐ Show shadow of possible placement ☐ Ability to 'save' results and parameters	3 3 3
TP3 (Project Codebase + Readme File + Project Demo(video & live) + Style)			

Wave Function Collapse algorithm (WFC) (Gumin,2016) : constraint satisfaction algorithm acting on a set of tiles with associated connectivity rules

```
Iteratively samples a grid location
Samples a tile to be assigned to that location
Updates each remaining grid locations' probability distribution over what tiles can be sampled Given the constraints defined by til
This process repeats, and ends when all grid locations have an assigned tile.
```

- **Version Control Plan**

  Using Github repository for version control.



soohwahong Select Tile and move with mouse 51cdc5c 12 hours ago 6 commits

.vscode	Initial commit	22 hours ago
__pycache__	Initial commit	22 hours ago
cmu_graphics	Initial commit	22 hours ago
.DS_Store	Initial commit	22 hours ago
cmu_cs3_graphics.py	Initial commit	22 hours ago
main.py	Select Tile and move with mouse	12 hours ago
scratch.py	Initial commit	22 hours ago
settings.py	Initial commit	22 hours ago
tile.py	Initial commit	22 hours ago

Help people interested in this repository understand your project by adding a README.

- **Module List :** numpy

Resources

https://robertheaton.com/2018/12/17/wavefunction-collapse-algorithm/

https://boristhebrave.github.io/DeBroglie/articles/index.html

https://bitbucket.org/mxgmn/basic3dwfc/src/master/

TP2 Update

- Path generation mode
    - Created tileset with connectivity constraints based on start and end location
    - Path Finding Game : given the start and end goal, user needs to use tile set & rotations to create path
        - board lights green & red to tell user if tile is legal at location in relation to neighboring tiles
    - Autosolver: Designate start(home) & end(destination) tile and location
    - DFS to find connecting path
- Potential MVP features
    - UI elements (instructions, switch from manual to autosolving mode...)
    - Transparent path + ball moving through path view
    - Rotating view

TP2 Update (4/23 added)

(Video is showing ...)

- Pattern generation mode
    - Created new tileset with connectivity constraints
    - Pattern Creating Game: player places tiles according to their connectivity constraints to create pattern
        - board lights green & red to tell user if tile is legal at location in relation to neighboring tiles
    - Autogeneration
        - Implement Wave Function Collapse Algorithm

- Constraint satisfying algorithm that involves Forward checking with propogation
    - When tile is placed, set of possible tiles for each cell of board changes
    - With each tile placement, the effect propogates through entire board
- Different patterns are generated with different subsets of tile set!