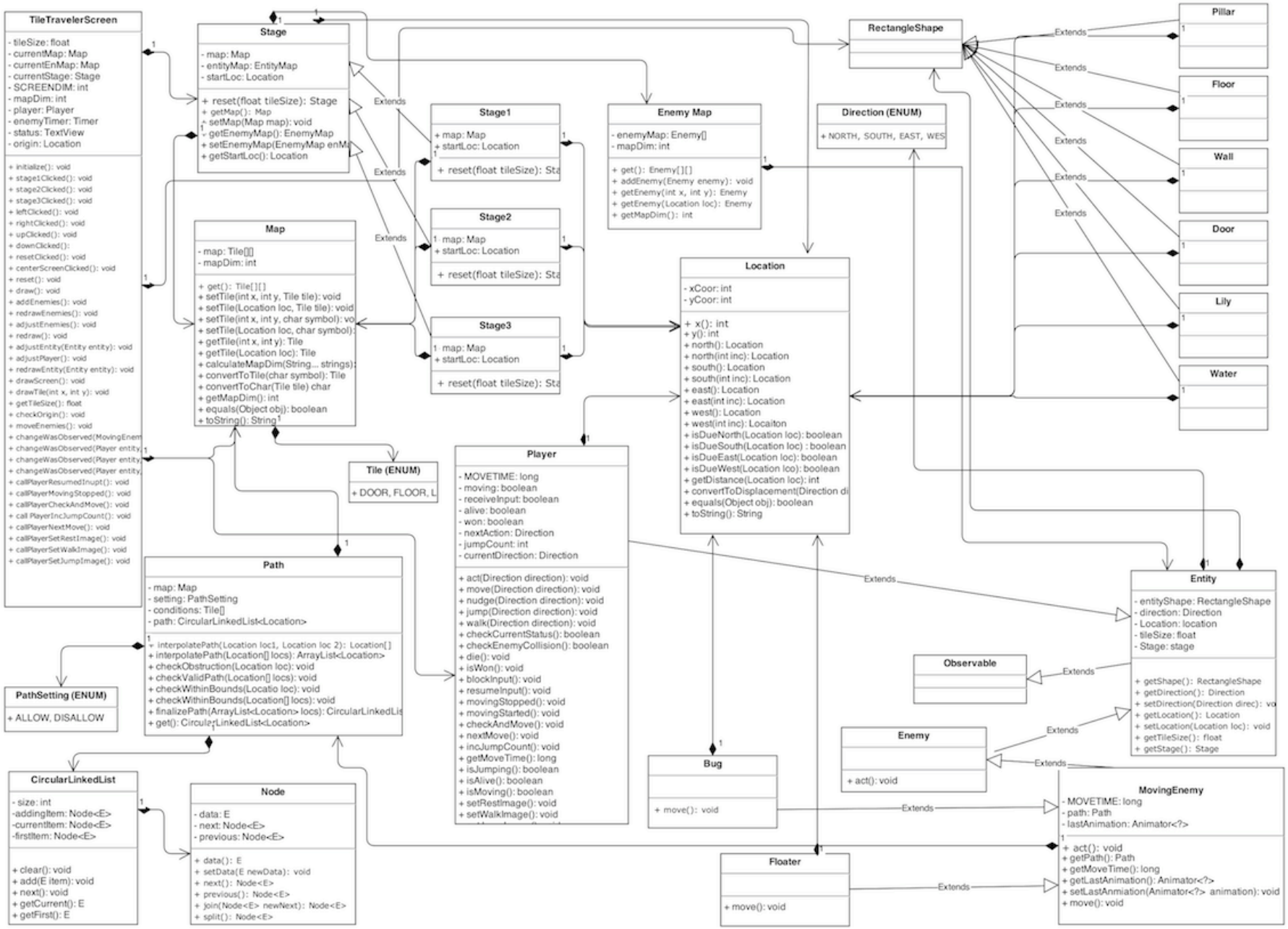
TileTravelerScreen
- tileSize: float
- currentMap: Map
- currentEnMap: Map
- currentStage: Stage
- SCREENDIM: int
- mapDim: int
- player: Player
- enemyTimer: Timer
- status: TextView
- origin: Location
+ initialize(): void
+ stage1Clicked(): void
+ stage2Clicked(): void
+ stage3Clicked(): void
+ leftClicked(): void
+ rightClicked(): void
+ upClicked(): void
+ downClicked():
+ resetClicked(): void
+ centerScreenClicked(): void
+ reset(): void
+ draw(): void
+ addEnemies(): void
+ redrawEnemies(): void
+ adjustEnemies(): void
+ redraw(): void
+ adjustEntity(Entity entity): void
+ adjustPlayer(): void
+ redrawEntity(Entity entity): void
+ drawScreen(): void
+ drawTile(int x, int y): void
+ getTileSize(): float
+ checkOrigin(): void
+ moveEnemies(): void
+ changeWasObserved(MovingEnem
+ changeWasObserved(Player entity)
+ changeWasObserved(Player entity)
+ changeWasObserved(Player entity)
+ callPlayerResumedInupt(): void
+ callPlayerMovingStopped(): void
+ callPlayerCheckAndMove(): void
+ call PlayerIncJumpCount(): void
+ callPlayerNextMove(): void
+ callPlayerSetRestImage(): void
+ callPlayerSetWalkImage(): void
+ callPlayerSetJumpImage(): void
Stage
- map: Map
- entityMap: EntityMap
- startLoc: Location
+ reset(float tileSize): Stage
+ getMap(): Map
+ setMap(Map map): void
- getEnemyMap(): EnemyMap
+ setEnemyMap(EnemyMap enMa
+ getStartLoc(): Location
Stage1
+ map: Map
+ startLoc: Location
+ reset(float tileSize): Sta
Stage2
1 map: Map
+ startLoc: Location
+ reset(float tileSize): Sta
Stage3
1 map: Map
+ startLoc: Location
+ reset(float tileSize): Sta
Enemy Map
- enemyMap: Enemy[]
- mapDim: int
+ get(): Enemy[][]
+ addEnemy(Enemy enemy): void
+ getEnemy(int x, int y): Enemy
+ getEnemy(Location loc): Enemy
+ getMapDim(): int
Direction (ENUM)
+ NORTH, SOUTH, EAST, WES
RectangleShape
Pillar
Floor
Wall
Door
Lily
Water
Extends
Map
- map: Tile[][]
- mapDim: int
+ get(): Tile[][]
+ setTile(int x, int y, Tile tile): void
+ setTile(Location loc, Tile tile): void
+ setTile(int x, int y, char symbol): vo
+ setTile(Location loc, char symbol):
+ getTile(int x, int y): Tile
+ getTile(Location loc): Tile
+ calculateMapDim(String... strings):
+ convertToTile(char symbol): Tile
+ convertToChar(Tile tile) char
+ getMapDim(): int
+ equals(Object obj): boolean
+ toString(): String
Location
- xCoor: int
- yCoor: int
+ x(): int
+ y(): int
+ north(): Location
+ north(int inc): Location
+ south(): Location
+ south(int inc): Location
+ east(): Location
+ east(int inc): Location
+ west(): Location
+ west(int inc): Locaiton
+ isDueNorth(Location loc): boolean
+ isDueSouth(Location loc) : boolean
+ isDueEast(Location loc): boolean
+ isDueWest(Location loc): boolean
+ getDistance(Location loc): int
+ convertToDisplacement(Direction di
+ equals(Object obj): boolean
+ toString(): String
Tile (ENUM)
+ DOOR, FLOOR, L
Player
- MOVETIME: long
- moving: boolean
- receiveInput: boolean
- alive: boolean
- won: boolean
- nextAction: Direction
- jumpCount: int
- currentDirection: Direction
+ act(Direction direction): void
+ move(Direction direction): void
+ nudge(Direction direction): void
+ jump(Direction direction): void
+ walk(Direction direction): void
+ checkCurrentStatus(): boolean
+ checkEnemyCollision(): boolean
+ die(): void
+ isWon(): void
+ blockInput(): void
+ resumeInput(): void
+ movingStopped(): void
+ movingStarted(): void
+ checkAndMove(): void
+ nextMove(): void
+ incJumpCount(): void
+ getMoveTime(): long
+ isJumping(): boolean
+ isAlive(): boolean
+ isMoving(): boolean
+ setRestImage(): void
+ setWalkImage(): void
Path
- map: Map
- setting: PathSetting
- conditions: Tile[]
- path: CircularLinkedList<Location>
+ interpolatePath(Location loc1, Location loc 2): Location[]
+ interpolatePath(Location[] locs): ArrayList<Location>
+ checkObstruction(Location loc): void
+ checkValidPath(Location[] locs): void
+ checkWithinBounds(Locatio loc): void
+ checkWithinBounds(Location[] locs): void
+ finalizePath(ArrayList<Location> locs): CircularLinkedLis
+ get(): CircularLinkedList<Location>
PathSetting (ENUM)
+ ALLOW, DISALLOW
CircularLinkedList
- size: int
-addingItem: Node<E>
-currentItem: Node<E>
-firstItem: Node<E>
+ clear(): void
+ add(E item): void
+ next(): void
+ getCurrent(): E
+ getFirst(): E
Node
- data: E
- next: Node<E>
- previous: Node<E>
+ data(): E
+ setData(E newData): void
+ next(): Node<E>
+ previous(): Node<E>
+ join(Node<E> newNext): Node<E>
+ split(): Node<E>
Bug
+ move(): void
Floater
+ move(): void
Entity
- entityShape: RectangleShape
- direction: Direction
- Location: location
- tileSize: float
- Stage: stage
+ getShape(): RectangleShape
+ getDirection(): Direction
+ setDirection(Direction direc): vo
+ getLocation(): Location
+ setLocation(Location loc): void
+ getTileSize(): float
+ getStage(): Stage
Observable
Enemy
+ act(): void
MovingEnemy
- MOVETIME: long
- path: Path
- lastAnimation: Animator<?>
+ act(): void
+ getPath(): Path
+ getMoveTime(): long
+ getLastAnimation(): Animator<?>
+ setLastAnmiation(Animator<?> animation): void
+ move(): void