Bullet class CIS 29 Group 6 Covid Defense game_map namespace Tower class + SIDE PIX : extern const unsigned - bullet : sf::Sprite Thomas Zhang, Jonathan Ly, Rohan Shrotri, Kuo Fu GameMap class - tower : sf::RectangleShape tvpe : std::strina + SIDE FLT : extern const float - towerType : std::string - towerX : double + SPRITE SCALE : extern const - cells : Cell*** - haveTarget : bool sf::Vector2f - towerY : double - sideLength : unsigned - bulletOffsetX : double - enemyX : double + RESOURCE PATH: extern const - startCoords : sf::Vector2u - bulletOffsetY : double - enemyY : double std::string - exitCoords : sf::Vector2u - placedDown : bool + coordsToPosition(const sf::Vector2u&) - drawOrigin : sf::Vector2f + Bullet(std::string, double, - groundLocation : Cell* sf::Vector2f - groundTexture : sf::Texture double, double, double) shootingGap: int pathTexture : sf::Texture MainMenuScreen class + setPos(sf::Vector2f) : void + Tower(sf::Vector2f, sf::Vector2f, + fire(): void std::string. Cell*) + getLeft() : int + GameMap(const std::string&, const selectedItemInde: int + getGroundLocation(): Cell*& + getTop(): int sf::Vector2f& = sf::Vector2f(0.f, 0.f)) : -font: sf::Font + setPosition(sf::Vector2f) : void + draw(sf::RenderWindow&): void Cell abstract class + ~GameMap() -playButton: sf::Text + setPlacedDown(bool) : void + rotate(): void + setExitCoords(sf::Vector2u) : void -scoreButton: sf::Text + setHaveTarget(bool) : void + getSprite(): sf::Sprite& + setStartCoords(sf::Vector2u) : void # coords : sf::Vector2u -exitButton:sf::Text + incrementShootingGap(): void + getCells() : Cell** # cellSprite : sf::Sprite + getHaveTarget() : bool + MainMenuScreen(float width, float height); + getSideLength() const : unsigned # position : sf::Vector2f + getPlacedDown(): bool HighScoreList + ~MainMenuScreen(); + getStartCoords() const : sf::Vector2u + getOffsetX(): double + draw(sf::RenderWindow &window): void + getExitCoords() const : sf::Vector2u + getOffsetY() : double -struct ListNode + GetPressedItem(): int + getDrawOrigin() const : sf::Vector2f + Cell() = default : + shouldShoot(): bool - *head: ListNode + Cell(unsigned, unsigned): + displayMenu(sf::RenderWindow &, std::string &): int + getGlobalBounds() const : sf::FloatRect + getRect() : sf::RectangleShape& - count: int + draw(sf::RenderWindow&) const : void + Cell(const sf::Vector2u&): + displayWelcome(sf::RenderWindow&): std::string + getX(): int -outFilename: std::string + virtual ~Cell() = default + getY(): int + virtual setCoords(const sf::Vector2u&) +HighScoreList(): + getPosition(): sf::Vector2f final: void + insertNode(PlayerData): void + getType() : std::string + virtual setSpriteTexture(const + draw(sf::RenderWindow&) : void + getCount() :int CellEnum enum class sf::Texture&) final : void + isEmpty(): bool + virtual setSpritePosition(const + update():void + PATH sf::Vector2f&) final: void + writeToFile(std::string): void + GROUND + virtual setSpriteScale(const +~HighScoreList(); sf::Vector2f&) final : void + virtual getCoords() const final : sf::Vector2u + virtual getPosition() const final : sf::Vector2f + virtual getSprite() const final : sf::Sprite GamePlayScreen class + virtual getCellType() const = 0 : Animation class Ground class CellFnum Corona class + namespace draw - imageCount : sf::Vector2u - cellType : const CellEnum + namespace prices - movespeed : float - currentImage : sf::Vector2u - hasTower : bool + namespace messages - drawCovid : bool - totalTime : float + namespace input - switchTime : float covidDirection : sf::Vector2f getLocation(sf::Vector2f, GameMap&) + uvRect : sf::IntRect - covidDestination : sf::Vector2f + Ground(unsigned, unsigned): Path class Cell* - distanceToDestination : sf::Vector2f + Ground(sf::Vector2u): - checkTowerPlacement(const + Animation(): - covidCoords : sf::Vector2u + setHasTower(bool) : void sf::Vector2f&, GameMap*, + Animation(sf::Texture*, - cellType : const CellEnum + getHasTower() const : bool std::vector<Tower>&): sf::Vector2f sf::Vector2u, float): + Corona(): nextPathCoords : sf::Vector2u + virtual getCellType() const override : - player : PlayerData + Animation(const Animation&) : + Corona(sf::Texture*, GameMap sf::Vector2, Path*) nextPathPosition: sf::Vector2f CellEnum + ~Corona() + ~Animation() gameOver : bool + Update(int, float) : void + setMoveSpeed(const float) : void gameVictory : bool + Path(unsigned, unsigned, unsigned, - windowClosed : bool + getSwitchTime() : float + getMoveSpeed(): float + setDrawCovid(bool) : void unsigned) + Path(const sf::Vector2u&, const + getDrawCovid() : bool + GamePlayScreen() = default + setCovidDirection(sf::Vector2f) : void sf::Vector2u&): EnemyBase class + ~GamePlayscreen() = default + getCovidDirection(): sf::Vector2f + setNext(unsigned, unsigned) : void + draw(sf::RenderWindow&): void + setNext(const sf::Vector2u&) : void + setCovidDestination(sf::Vector2f): void - sprite : sf::Sprite + setPlayer(const PlayerData&) : void + getNextCoords() const : sf::Vector2u + getCovidDestination(): sf::Vector2f + getPlayer() : PlayerData - setDistanceToDestination(sf::Vector2f): void + enemyBase + getNextPosition() const : sf::Vector2f + getWindowClosed() : bool + virtual getCellType() const override : + getDistanceToDestination(): sf::Vector2f + ~enemyBase + getGameOver() : bool + setCovidCoords(sf::Vector2u): void + setSprite : void CellEnum + getGameVictory(): bool + getCovidCoords(): sf::Vector2u getSprite : sf::Sprite + friend normalize(const sf::Vector2f&) sf::Vector2f + friend loadTexture(sf::Texture&, std::string&): void GameOverScreen + friend loadSound(sf::SoundBuffer&, HighscoreScreen Collision namespace std::string&): void int selectedItemIndex; - backButton : sf::Text + friend loadFont(sf::Font&, std::string&) + PixelPerfectTest(const sf::Sprite&, const MainMenuScreen class - title : sf::Text hsFont: sf::Font void sf::Sprite&, sf::Uint8 = 0) : bool - font : sf::Font hsName: sf::Text CreateTextureAndBitmask(sf::Texture&. + namespace menu - selectedItemIndex : int exitButton:sf::Text const std::string&) : bool - selecteditemIndex : int scores[scores::NUM SCORES]: sf::Text CircleTest(const sf::Sprite&, const - font : sf::Font sf::Sprite&) : bool + HighscoreScreen(sf::RenderWindow &); GameOverScreen(sf::RenderWindow) playButton : sf::Text + BoundingBoxTest(const sf::Sprite&, ~GameOverScreen() - scoreButton : sff::Text + ~HighscoreScreen(); const sf::Sprite&) : bool draw(sf::RenderWindow): void exitButton : sf::Text + draw(sf::RenderWindow&): int displayEnd(sf::RenderWindow, constPlayerData): void + MainMenuScreen(float, float) + ~MainMenuScreen() + draw(sf::RenderWindow): void MouseDown enum class + GetPresseditem(): int + PRESS + displayWelcome(sf::RenderWindow) + HOLD PlaceTower enum class void + RELEASE + displayMenu(sf::RenderWindow & + NONE window): int + SANITIZER

+ SOAP