Renderer # static constexpr int CELL SIZE + virtual ~Renderer() + GamePixmapItem * renderGame Objects(QList< QMap< DataRole, QVariant >> objectData) + virtual GamePixmapItem * renderGameObject(QMap < DataRole, QVariant > objectData) + virtual void renderGameObject (QMap< DataRole, QVariant objectData, GamePixmapItem *item) + QPixmap rotatePixmap (const QPixmap &originalPixmap, int direction) + Qlmage rotatelmage (const Qlmage &image, int direction) # QImage animateHealthPack (int health, GamePixmapItem *item) # QPropertyAnimation * animateTint(QColor final, QColor initial $=\{0, 0, 0, 0\}$ # QPropertyAnimation * animateAttack(int dir, bool attacking) # QPropertyAnimation * animateBounce() # QPropertyAnimation * animateHealth(Direction dir) # QPropertyAnimation * animateHide() ColorRenderer + ColorRenderer() + void renderGameObject (QMap < DataRole, QVariant > objectData, GamePixmapItem *item) override QPixmap renderTile (QMap < DataRole, QVariant > object) QPixmap renderDoorway (QMap < DataRole, QVariant > object) QPixmap renderHealthPack (QMap < DataRole, QVariant > object) QPixmap renderProtagonist (QMap < DataRole, QVariant > object) QPixmap renderEnemy (QMap < DataRole, QVariant > object) QPixmap renderPEnemy (QMap < DataRole, QVariant > object)

 - QPixmap renderMovingEnemy (QMap< DataRole, QVariant

> object)