**Single Player Game Architecture**

* **Space Invaders- “Scum Invaders”**
  + **button** module
    - Button class
      * init: declare class variables
      * checkClicked: returns mouse position
      * checkHover: checks if mouse position is over a button and calls flip\_image function if it does
      * flip\_image: calls click function to play a sound and replaces current image with another image
      * click: plays a sound
      * draw: draws image at current position
  + **enemy** module
    - Enemy class
      * init: declare class variables
      * fire: calls the missile class while tracking missile usage
      * moveLeft: moves the position of the sprite to the left while staying on screen
      * moveRight: moves the position of the sprite to the right while staying on screen
      * moveDown: shifts the sprite down 32 pixels while staying on screen
      * getPos: returns the current position
  + **game** module
    - game class
      * init: declare class variables
      * reset: resets the game by putting the game back in its initial state
      * setGrid: creates a grid for enemy sprites to aligned in
      * draw: draws the images at their designated positions
      * update: updates changes every frame
  + **main\_menu** module
    - Main\_Menu class
      * init: declare class variables
      * draw: draws the images at their designated positions
      * mouseUpdate: checks if mouse clicks on a button and return button click if so
      * update: updates changes every frame
  + **missile** module
    - Missile class
      * init: declare class variables
      * update: updates changes every frame
      * getEnemyPos: returns enemy’s position
      * getPos: returns the current position
  + **player** module
    - Player class
      * init: declare class variables
      * moveLeft: moves the position of the sprite to the left while staying on screen
      * moveRight: moves the position of the sprite to the right while staying on screen
      * getPos: returns the current position
      * fire: calls the missile class while tracking missile usage
  + **ScumInvaders** module
    - ScumInvaders class
      * init: declare class variables
      * game\_loop: manages running game events (menu and game)
  + **soundManager** module
    - soundManager class
      * init: declare class variables
      * loadAll: load all sounds to be ready for play
      * playCurrentMusic: continues to play the current music
      * playSound: plays a sound file
  + **Sprite\_Manager** module
    - sprites class
      * init: declare class variables
      * load: loads the sprite sheet
      * loadAll: load all images to be ready for display
      * getSprite: returns a sprite
      * getAll: returns a set of all sprites
    - Animate class
      * init: declare class variables
      * update: updates changes every frame
      * draw: draws the images at their designated positions