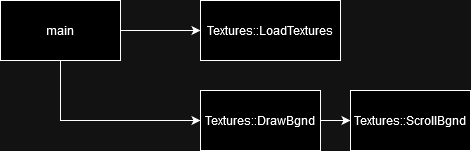
# Side Scrolling Shmup Design Document

### Understanding the template

In order to better understand the codebase i created a flowchart and procedure dependency for diagram for the template codebase. They are very simple for this template

#### Procedure Dependency Diagram



#### Flowchart



## Upgrading the template

In order to help me plan the changes i wanna make i made a flowchart for the program after the planned upgrades

I dont think the current system of storing the sprites in the template is easily extendable, and as such i will be building my own system, storing the sprites within an entity object.

##### Entity object

At the core of the planned upgrades is the entity object, storing all the infomation and methords required to move the ship and any future objects required in the future. Building this as an entity object will allow the code to be neatly contained and minimise repeat/redundant code. For future proofing and as good practice i will then extend a ship class from that entity class in minimise the overhead and clutter for non ship objects that may be added in the future.

Here is some psudo code for what i predict will be contained within it

|  |
| --- |
| Entity object{  Float xPos, yPos  Sprite entSpr  initEnt{  setTexture()  setPos()  }  RenderIn{}  setPos{}  accellEnt{}  mvEnt{  clampToScreen()  }  updateEntPos{}  }  Ship::entity object{  handleUserInputs{  Vector2 inputtedDirect  getUsrInForInputtedDirect  accellEnt(inputtedDirect\*MoveSpeed\*deltaSinceLastFrame)  }  } |

### Upgrade flowchart

