# 1. Proposal:

*NOTE: Complete this section only if you are intending to create a project that does not use the templates provides in class. If you are not sure, please ask!*

## Overview:

Provide a description of what product(s) you intend to create in this project:

|  |
| --- |
| A video game that requires the player to navigate a spaceship using directional thrust to rescue floating astronauts and avoid hazards such as asteroids. |

## Objective:

Provide a description of what skills and understandings you intend to gain in completing this project:

|  |
| --- |
| Test of the AnimationLoop engine |

## Connections:

Describe how this project connects with two other CTS courses that you have taken (these can be specific one-credit courses within Computer Science):

|  |
| --- |
| OO1 (use of classes and instances to model the game world) and Data Structures (use of arraylists to maintain all sprites) |

## Resources:

Provide a list of resources, outside of what is available in class, that you will require to complete this project:

|  |
| --- |
| Google Images, some free audio clips |

## Approval

At this point, submit this document to your teacher to obtain feedback and approval on your proposal

# **2. Planning**

Time Appreciation

We’ve often discussed different phases. Then, provide estimates for how long each phase will take. You have already completed the first phase, which is to create a proposal. You are currently on the second phase, which is planning. Break your implementation up into smaller steps (e.g. interface, logic layer, fart, etc).



# **3. Design**

## Input

*Note: If applicable – e.g. for graphical games*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Action | Mouse | Action | | Other | Action |
| Left arrow | Turn left |  |  |  | |  |
| Right arrow | Turn right |  |  |  | |  |
| Up arrow | Apply thrust (move forward) |  |  |  | |  |
| Space | Shoot bullets |  |  |  | |  |
|  |  |  |  |  | |  |
|  |  |  |  |  | |  |
|  |  |  |  |  | |  |

## Processing

Provide the required algorithms and/or general computational strategy

*Note: If applicable – e.g. for programs that process data*

|  |
| --- |
| Not applicable |

## Storyboard

Provide a storyboard for your program. Your *user* should be able to understand how the program would *execute*. For graphical projects, consider what your project would *look* like. For console-based projects, provide a “script” for how the program would accept input and provide output.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# 4. Process

## Journal

Provide a journal for the time that you worked on this project:

* You do not need to provide an entry for each period or session that you worked on the project, but your entries should cover all time spent.
* This is not meant to be a simple accounting of your time. Rather, this is a living document that records the process of building your projects and demonstrates your learning. Entries can contain your thoughts on
  + Work completed
  + Victories achieved
    - getting something to work
    - understanding a concept
    - Challenges encountered
  + Resources used
  + Meta-cognition
    - What have you learned about you as a learner?
    - What is working well?
    - What is not?

*Note 1: It is strongly recommended that you use a template for each entry: For an example, use the template provided on the next page. You can copy and paste new entries as you go.*

*Note 2: If this is a team project,* ***each member is expected to provide a separate journal.***

| Date |  | Hours: |  |
| --- | --- | --- | --- |
| **Work Completed:** |  | | |
| **Victories Achieved:** |  | | |
| **Resources used:** |  | | |
| **Working well:** |  | | |
| **Not working well:** |  | | |
| **NEXT STEPS:** |  | | |

| Date |  | Hours: |  |
| --- | --- | --- | --- |
| **Work Completed:** |  | | |
| **Victories Achieved:** |  | | |
| **Resources used:** |  | | |
| **Working well:** |  | | |
| **Not working well:** |  | | |
| **NEXT STEPS:** |  | | |

## 5. Documentation

### Deployment / Execution

Provide instructions on how your game can be deployed (installed) and executed (run) on another computer.

* If you have used the templates provides in class, then this should be straight forward:
  + Import into Eclipse
  + run <main class>
* If you are using another platform (language, development environment, etc):
  + Note any dependencies and platform requirements
  + Be precise on how to run the project. Use screen captures if necessary
  + If you cannot explain this adequately, then your project likely does not meet the intended outcome, which is to create a stand-alone product.

|  |
| --- |
|  |

## 6. Reflection

*Note 1: Don’t be constrained by the size of the answer box!*

*Note 2: If this is a team project,* ***each member is expected to provide a separate reflection.***

1. Provide an updated time tracking chart, reflecting the hours actually spent on each category.



1. Refer to your original specification. How does the planned duration of each task compare with the actual time it took to complete?

|  |
| --- |
|  |

1. Did you need to deviate from your original specifications? If so, describe which and why.

|  |
| --- |
|  |

4. What would you change about your process (i.e. how you approached the creation of your product)?

|  |
| --- |
|  |

5. What worked well in your process?

|  |
| --- |
|  |

6. What part of the finished product are you most proud of, and why?

|  |
| --- |
|  |

7. If you were to create a version 2.0 of your product, what part of the product would develop further?

|  |
| --- |
|  |

8. Are there any specific bugs or functional shortcomings in your product? Be honest – being aware of these and recording them is a good thing!

|  |
| --- |
|  |

9. Which specific skills and understandings did you gain while working on this project?

|  |
| --- |
|  |