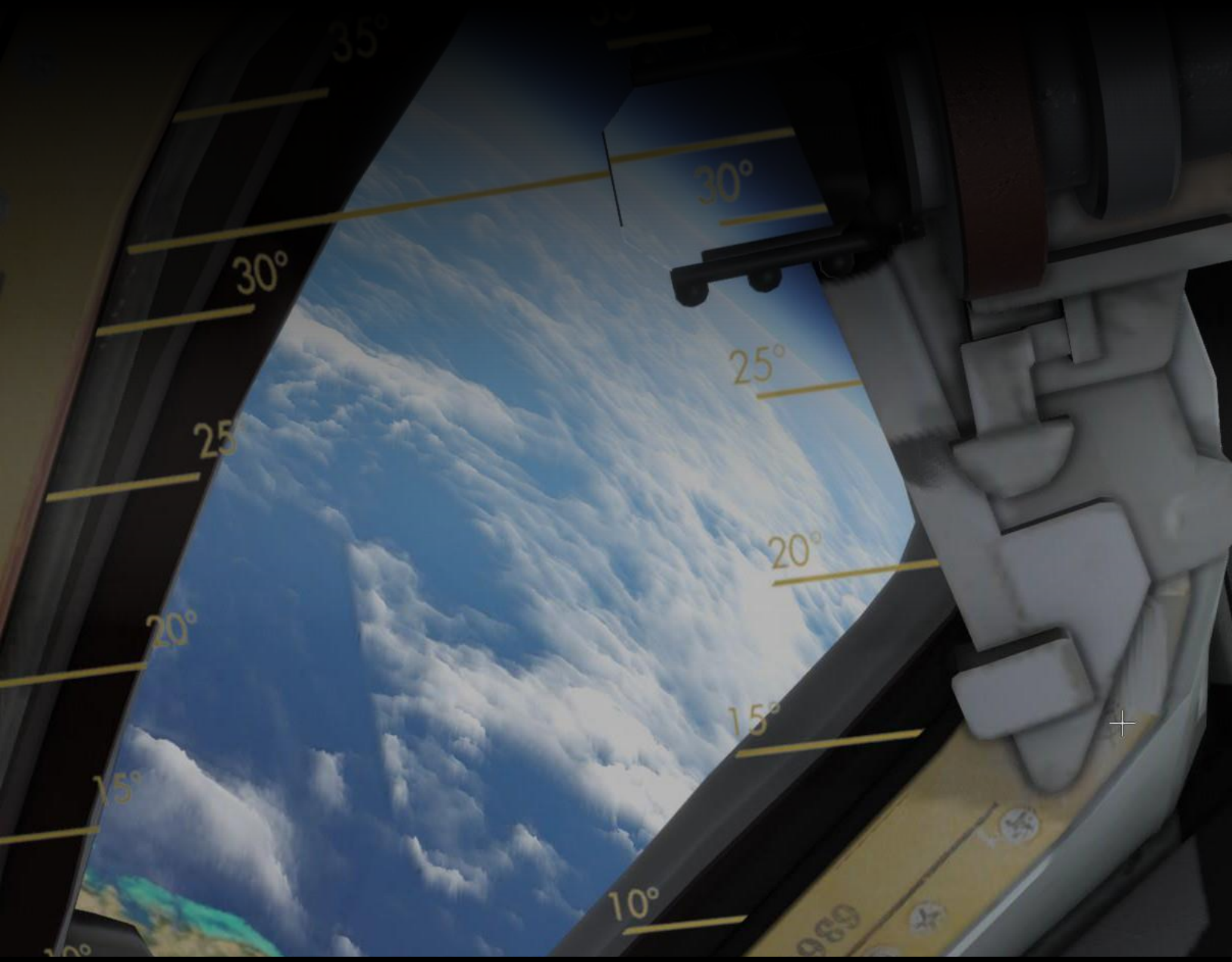


RENTRY

A SPACE FLIGHT SIMULATOR

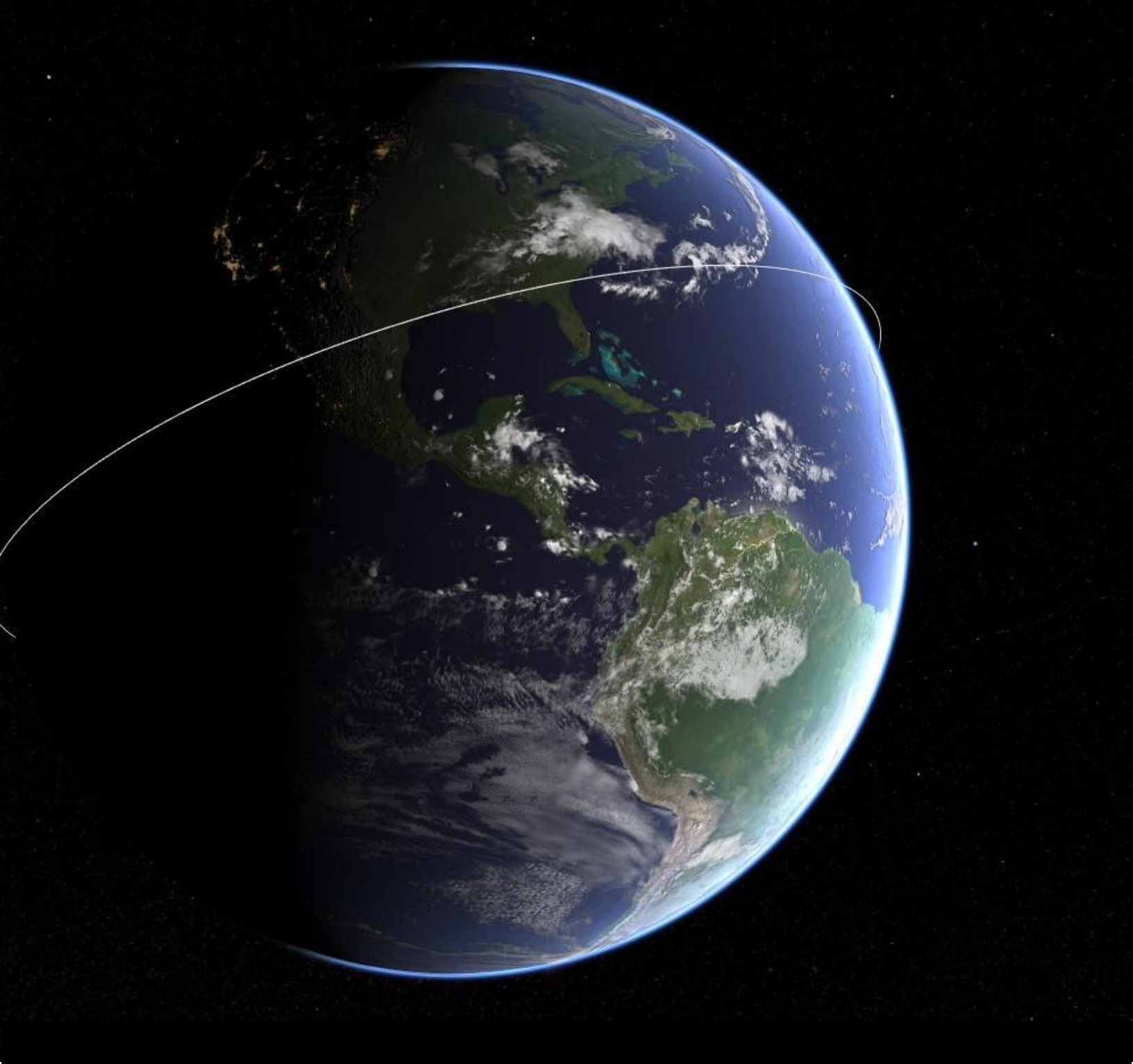
USER MANUAL

REENTRY

A SPACE FLIGHT SIMULATOR
USER MANUAL

TABLE OF CONTENTS

*



1\Introduction

I. INTRODUCTION

1. ABOUT

Thank you for buying Reentry – A Space Flight Simulator!

Reentry – A Space Flight Simulator is a game that allows you to fly and operate spacecrafts based on real spacecrafts used in manned missions.

The available spacecrafts have been programmed using the real manuals made public by NASA, with some modifications and simplifications made to allow this to be an enjoyable and gamified experience.

Please note that the game is in its second phase of development, and might crash or malfunction at any time. I'm working hard to fix bugs, and if you encounter an error, I highly suggest you send me the player log file located here:
C:\Users\<your user>\AppData\LocalLow\Wilhelmsen Studios\ReEntry\player.log

DOWNLOAD

The game can be purchased by following the links on the web site: http://reentrygame.com/ - the game package comes with the Mercury, Gemini and Apollo spacecrafts.

JOIN THE COMMUNITY – CONTRIBUTING

Use the Community Hub to discuss the game, as well as talk with the community, get help and give feedback to the developer.
The community hub for REENTRY can be found here:

Official Discord server:
https://discord.gg/reentrygame

Steam Hub:
https://steamcommunity.com/app/882140/

Reddit:
https://www.reddit.com/r/reentrygame/

GitHub:
https://github.com/ReentryGame

YouTube:
https://www.youtube.com/ReentryAnOrbitalSimulator

WHAT IS THIS MANUAL?

This manual contains most of the information you need to understand how REENTRY – An Orbital Simulator works. This manual covers the game itself, such as installation, setup and how to get you started. The Academy and the Flight Manuals are used to teach you the REENTRY spacecrafts.

2\Installation

II. INSTALLATION

1. DOWNLOADING

Reentry is distributed through Steam on the following link:
https://store.steampowered.com/app/882140/Reentry__A_Space_Flight_Simulator/

You will need to purchase the game to start the download, and to play it. Once downloaded, start the game through the Steam client.

2. SYSTEM REQUIREMENTS

The system requirements can be seen on the Steam page for REENTRY, and will have the latest known system requirements. It is not guaranteed that the game will run on your system, even if you meet the requirements.

MINIMUM	**RECOMMENDED**
Requires a 64-bit processor and operating system.	Requires a 64-bit processor and operating system
OS 64-bit Windows 7, Windows 8.1, Windows 10, Windows 11	**OS** 64-bit Windows 7, Windows 8.1, Windows 10, Windows 11
Processor Intel Core i5-4430 / AMD FX-6300	**Processor** Intel Core i5-6600K / AMD Ryzen 5 1600
Memory 16 GB RAM	**Memory** 32 GB RAM
Graphics NVIDIA GeForce GTX 960 2GB / AMD Radeon R7 370 2GB	**Graphics** NVIDIA GeForce GTX 1070 / Radeon RX Vega 56
DirectX Version 11	**DirectX** Version 11
Storage 20 GB available space	**Storage** 20 GB available space

3\Setup

III. SETUP

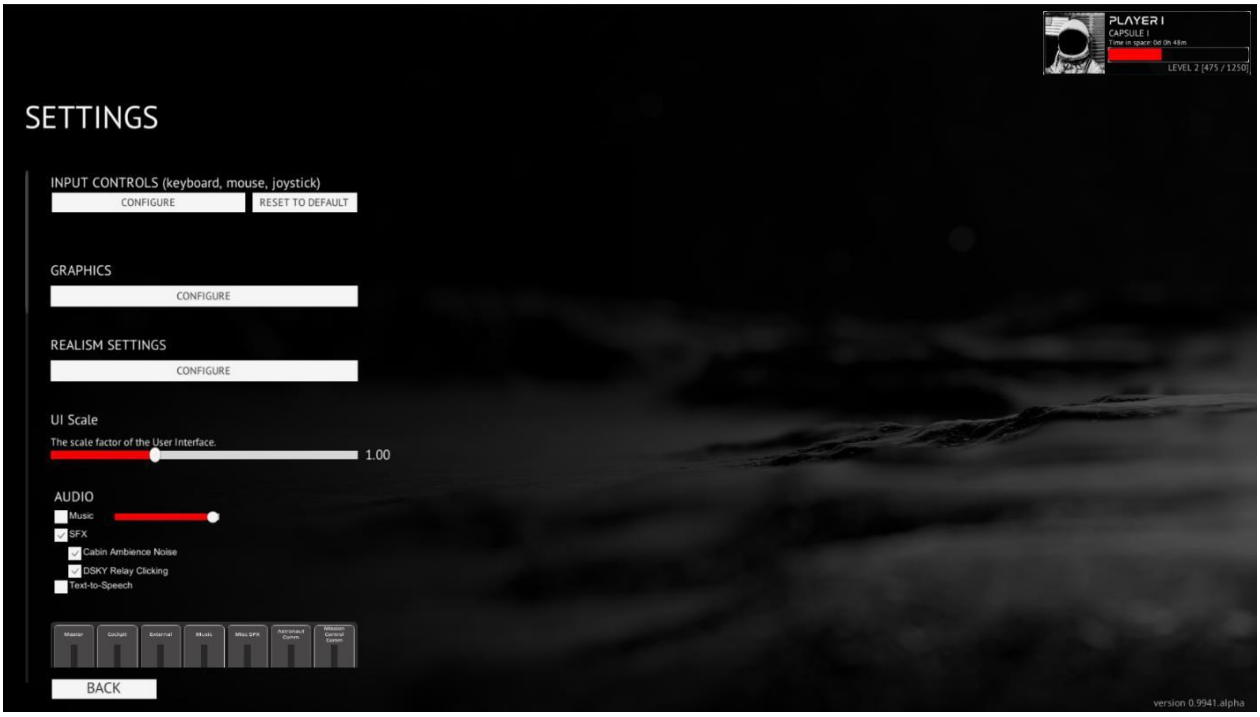
1. GENERAL

Once the game is installed you should be able to start the game using the Steam launcher.

The game can be configured in the SETTINGS menu accessible from the main menu. Given the complexity of the simulator, and the huge differences in for example rendering the Apollo Command Module vs. the Mercury Capsule, it is important to tune the graphics settings to ensure the game runs at the best frame rates possible. In some cases, you might need to even change the settings depending on what space craft you wish to focus on.

2. REENTRY SETTINGS

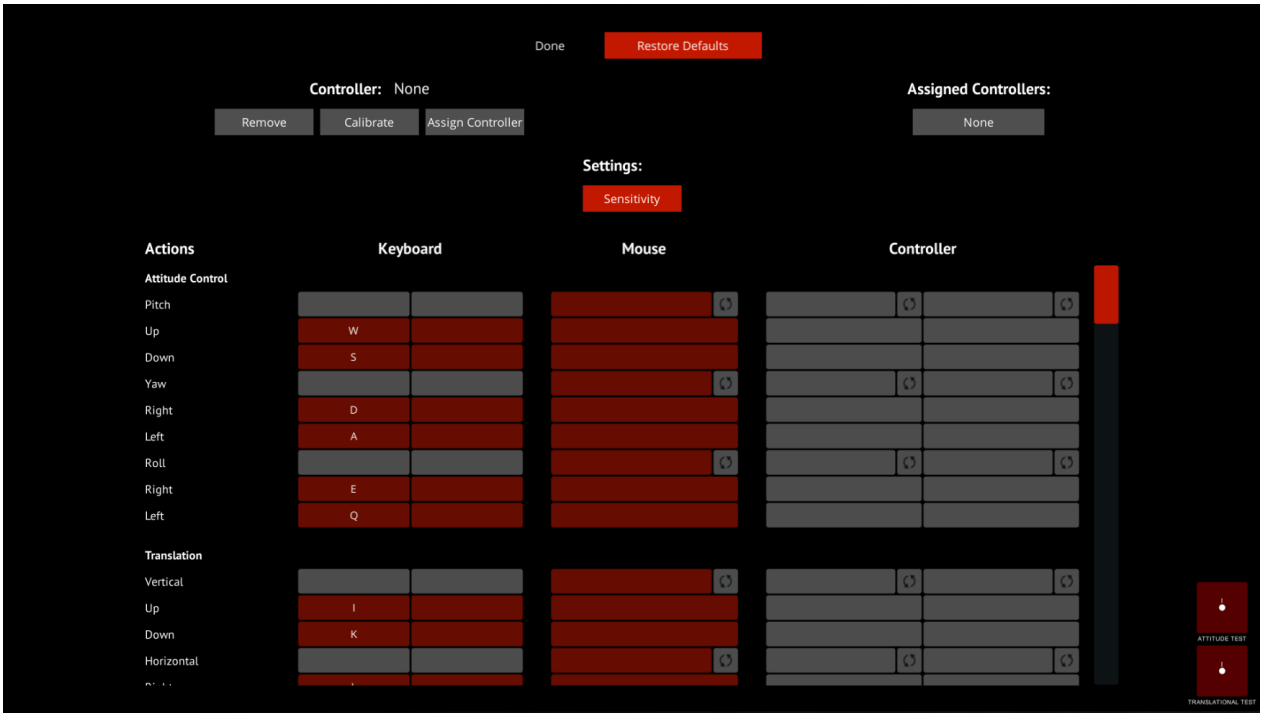
The SETTINGS menu can be accessed from the main menu of REENTRY. The SETTINGS menu is divided into multiple sections and contains all the main settings of the game. The core settings are available directly in the main settings page, while subsections such as INPUT, GRAPHICS and REALISM settings will open in a dedicated sub-page.



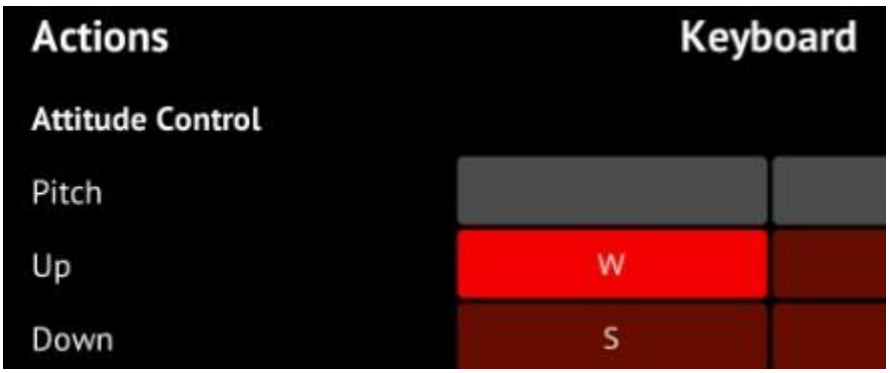
2.1. INPUT CONTROLS

The INPUT section is probably the most important section to familiarize yourself with. Most of the important controls will be covered in the games Academy. The CONFIGURE button will open the input settings, while the RESET TO DEFAULT will reset all the mappings to the factory settings.

Pressing CONFIGURE will open the input configurations where you can assign joysticks, set up the keyboard, mouse and joystick mappings.

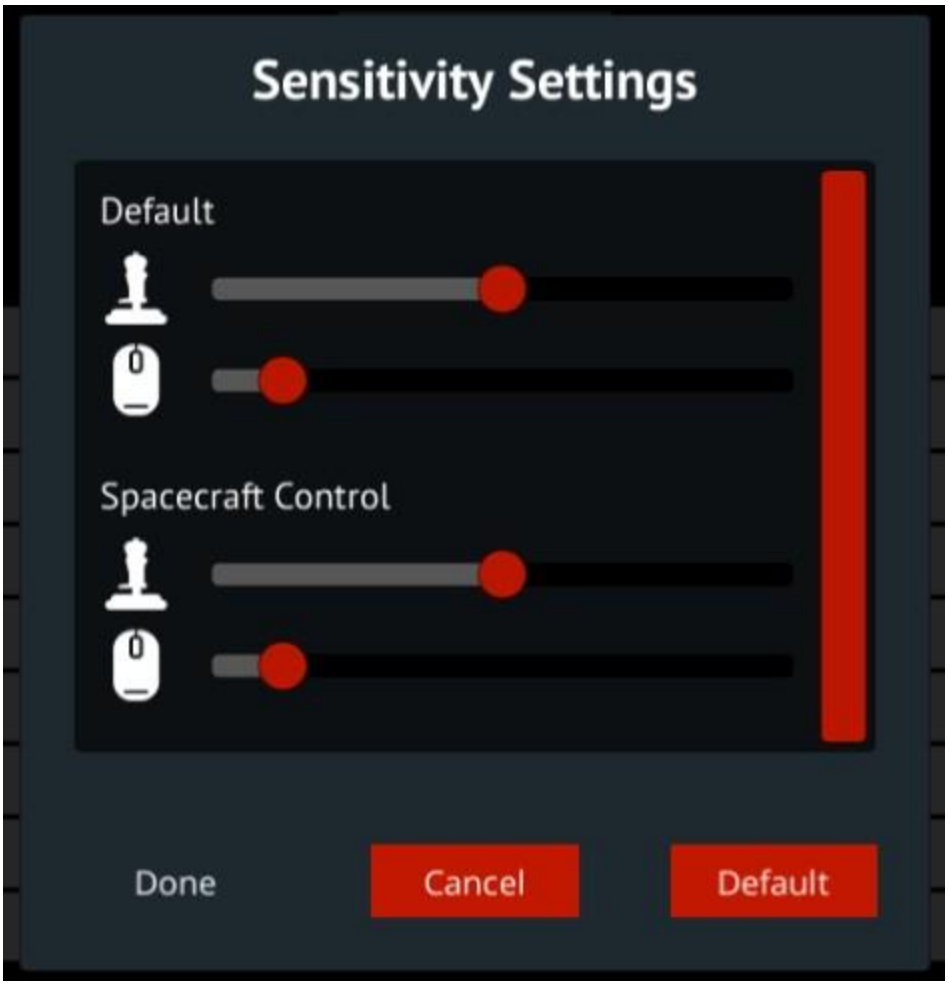


To assign a key, press the red input field and follow the instructions.



In most cases, you can assign a primary key and an alternative key for each action.

Sensitivity can be tuned using the sensitivity tool.

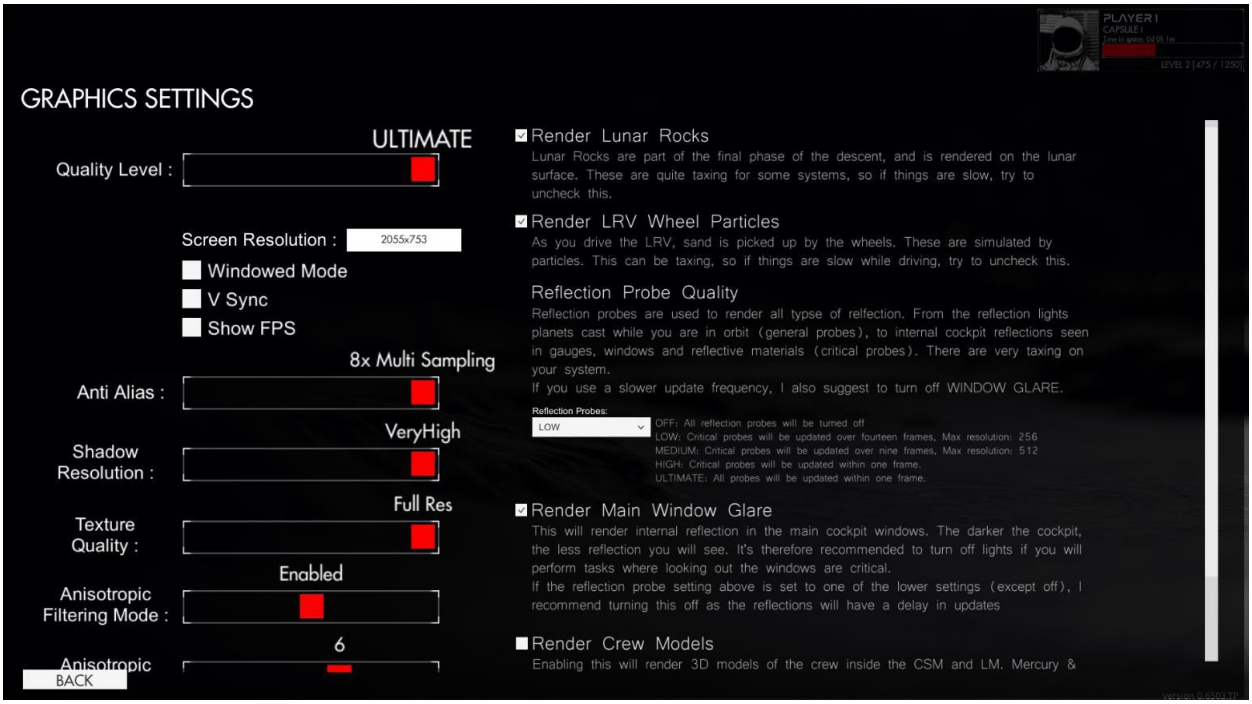


To assign a joystick, press ASSIGN and press a button on you joystick, The game will then recognize the joystick and you can assign the buttons and axes for it.

2.2. GRAPHICS

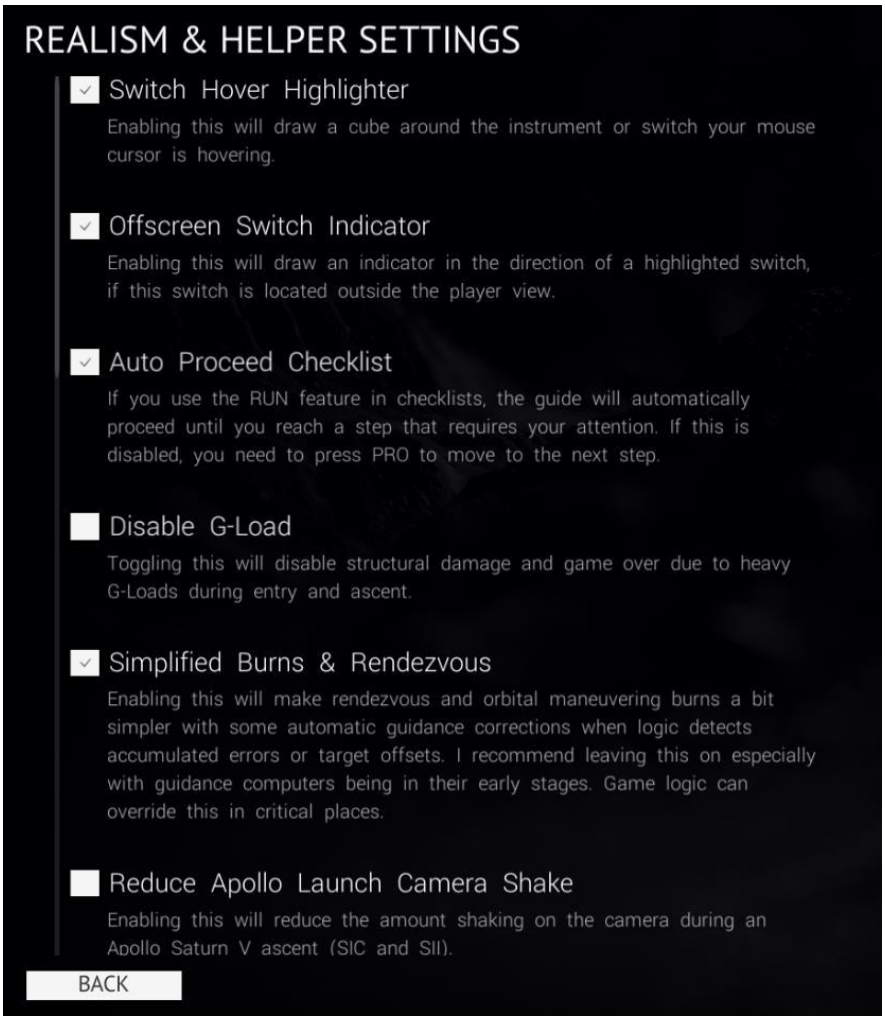
The graphics quality setting allows you to set the general level of the graphics quality setting of the game. The lowest setting will set the level to a mode that performs well, but reduces the quality of the graphics. The highest setting lowers the performance of the game, but increases the quality of the graphics.

The CONFIGURE button will allow you to further optimize your graphics settings, and toggle in-game effects and features.



2.3. REALISM SETTINGS

The Realism settings contain various options that will affect the realism level of the game. Each of the realism settings available contains a checkmark to enable/disable the feature along with a description of each.



2.4. UI SCALE

The UI Scale section can be used to scale the UI. This is good of you are required to run the game with a resolution that is smaller than 1080, or if you are running on very high resolutions. Drag the slider to see the effect of the scale immediately.

2.5. AUDIO

The Audio section allows you to enable or disable in-game music and SFX, and toggle the cockpit ambience sound effects. This is a cockpit sound effect generator that creates a realistic cockpit ambience. You can tune the audio mixers to your preference.

2.6. LANGUAGE

Language support is a community driven effort and will let you switch the primary language of the game. As language support evolved, more and more of the game will support multiple languages. English is the default language. If you select a different language, the game will use the available translated language files for the selected language. However, if a specific element or mission is yet to be translated, it will automatically use the English variant for the particular element.

2.7. CHECKLISTS

Hundreds of built-in checklists (default checklists) are available for use in-game and are a key element of the game. Every time you start the game, the game will check for updates to the currently installed set of default checklists. If a checklist has been updated, a popup will let you know and you will get the option of installing the latest checklists. This will overwrite the old checklist with the latest version.

You can create your own checklists in-game. Your custom checklists will be left alone.

You can also modify the default checklists. However, it is generally not recommended to modify the default checklists and is reserved for advanced users.

If you modify a default checklist, the game will think it has received an update as it differs from the standard definition. If you are making changes to the default checklists, you can disable the auto-updater and instead update these manually. If you update the checklists, all the modifications to the default checklists will be overwritten.

You can access the default checklists from the games GitHub repository.

2.8. PANEL STATES

Panel states are used to load the state of a panel at any time, either through a mission definition or through the console. You can save a panel state and load it another time. The game will ship with multiple default states such as “ingress” and “cold and dark”.

2.9. AUTOSAVE

Autosave is used to automatically create a new save state for each key action in-game, such as when planning a new burn or when the missions enters a new critical phase. Having this enabled will let you revert back to an earlier mission state if you make a critical mistake or if the game crashes. You should clean up the Save State directory in the games AppData folder at times if you keep this enabled.

2.10. COCKPIT CURSOR

The cockpit cursor is an in-game aid that renders a cursor on top of interactable switches. It will show you the name of the switch you are hovering, and its current position. A large dictionary is used to look up the current ID of the switch you are hovering, and is used to show you its related data.

The cockpit dictionary can be translated as well, and can be used to generate checklists.

2.11. EARTH TEXTURES

The Earth Textures tab allows you to select the detail level of Earth. High will increase loading time and the performance of the game, and requires a lot of memory. Low requires less.

Low Earth: 16k textures used

High Earth: 64k textures used

2.12. TRACK IR

The Track IR option will toggle the use of Track IR. Enable this option if you have Track IR connected and will be using this in the current game session. Remember to toggle this off is you disconnect Track IR, or else the cameras in-game will not know where to receive data from.

2.13. VIRTUAL REALITY

Virtual Reality is not yet a public feature of the game and is currently in a research phase. Follow the instructions to set up VR, and check the VR Setup video available on the games YouTube channel.

2.14. DATA

The Data tab allows you to delete all the progress you have made, level data, settings and so on. Don't use this unless you know what you are doing.

2.15. COMPUTER OUTPUT FEED LIVE

This setting will output the current computer state to a file so that you can create external devices that can render their displays. Players have been using this to create external OBC/AGC/LGC devices or programs.

4\Quickstart

IV. QUICKSTART

1. LEARNING THE BASICS OF THE GAME

There is a lot to learn when you first start playing REENTRY. You will first need to understand the game mechanics and the various systems available, and at the same time learn how to operate and fly the spacecrafts.

The recommended path is to first start with the Mercury Academy. This will walk you through the basics of Reentry and how the game mechanics work, while also learning about the space crafts.

This includes how to move around inside the cockpit, interacting with the panels and using the checklists.

To get started, load up the game and from the MAIN MENU, press ACADEMY. You will now see a list of each space craft. Select the PROJECT MERCURY academy. You should now see a list of various lessons in sequential order. Click on LESSON 1 to select the mission, and press LAUNCH to load into the simulator.

Follow the instructions to complete the lessons.

5\Getting Started

V. GETTING STARTED

Reentry comes packed with various missions designed to help you learn how to operate the crafts in Reentry. The career path mode will help you navigate all of this but before covering that, I want to briefly introduce you to the key areas of the game.

1. THE ACADEMY

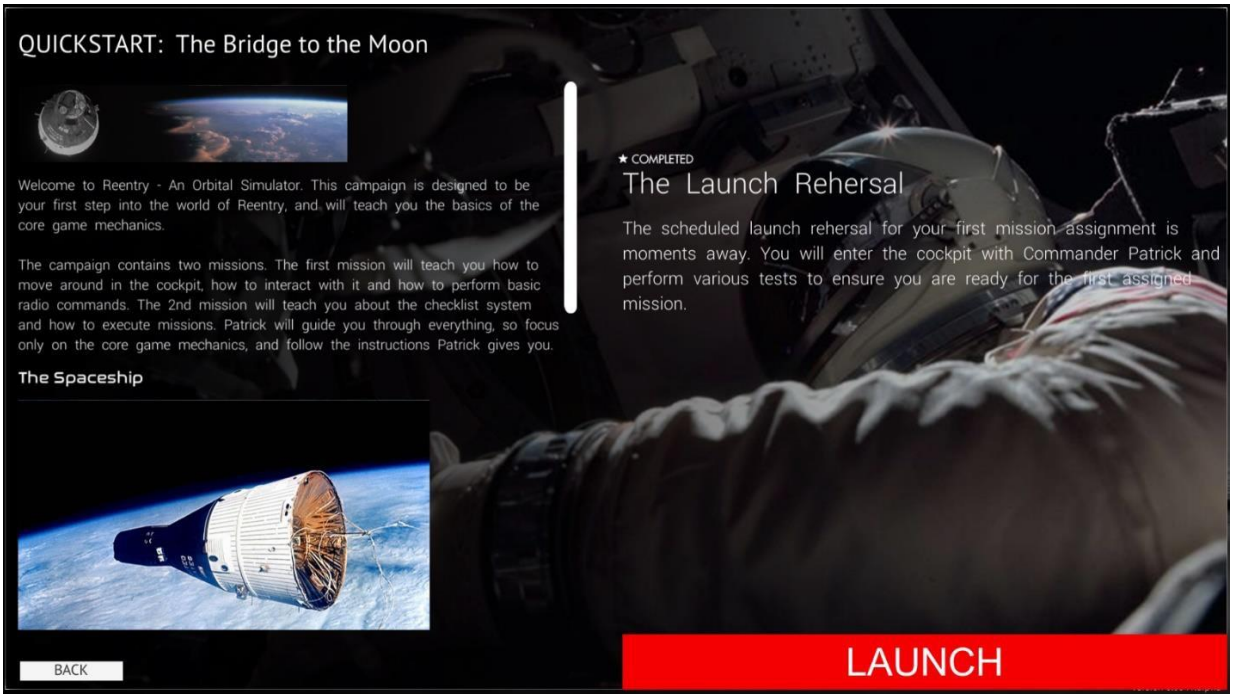
The academy is accessed from the main menu, and is where you can read and learn about the supported spacecrafts.

The main source of information for each spacecraft is its flight manual, however, the academy will teach you the basics of each, and take you through hands-on lessons.

The academy has a link to the latest version of the Flight Manual, and multiple lessons for each of the supported spacecrafts. I recommend skimming through the manual and reading the introduction chapters for the spacecraft you wish to learn. A basic knowledge about the main internal components and major systems will help you learn the craft faster.

2. CAMPAIGNS

A campaign is a collection of missions set in sequence. The missions are not limited to only be within one space program, so a campaign can host missions from all of the supported space programs.



The campaigns contain 4 main campaigns that put you in a fictional journey based on real events.

3. CAREER MODE

Reentry comes with a huge set of missions in both the Academy and the Campaigns. Navigating between them can be hard, and for the first-time player, knowing where to start can be a challenge.

The career mode will help you with this. Once you press CAREER from the Main Menu, the first thing you will need to do is select what craft you wish to learn. The easiest and logical craft to learn first is Project Mercury.

Once you select a craft you will see a board with a lot of nodes, each representing a goal to progress on the tree.

The node board has two paths on it, the top path is the astronaut-path that contains all the nodes you should complete to progress on your journey to learning the craft and becoming a commander.

The 2nd path is designed for Mission Controllers and will guide you to learning more about the various roles you can have in a multiplayer mission control session.

The board will automatically be updated when you complete them. You can either navigate to the requested mission or goal manually by navigating to the correct mission using the main menu, or you can click on a node, and it will try to prepare the correct mission for you.

The career board will guide you along the following sequence of missions:
1) Complete the two missions in the Quickstart campaign first
2) Complete the Mercury Academy
3) Complete Campaign I: Orbital Survival.
4) Complete the Gemini Academy.
5) Complete Campaign 2: Mastering Orbital Flight
6) Complete the Apollo Command Module Academy
7) Complete Campaign 3: The Dark Side of the Moon
8) Complete the Lunar Module Academy
9) Complete Campaign 4: The End of the Decade

By following this order, you will both get a gentle introduction to the core mechanics of Reentry, and learning the spacecrafts in the correct order. The real evolution of space crafts can be seen as you progress, and typically a lot of the systems in Apollo are based on what you learned in Gemini, and a lot of the systems in Gemini are based on what you learned in Mercury.

4. SCENARIOS

In many cases you want to simply enter a spacecraft to fly. The Free Play scenarios and the missions in the dedicated Scenarios sections will let you get a basic starting point with no assigned mission. It can be to insert you in lunar orbit with the Lunar Module configured for landing or ascent, or make Gemini start close to Agena for docking practice etc.

5. CUSTOM MISSIONS

You can create custom missions where you decide the objectives and communication flow. Separate Mission Editors for each of the supported space programs are available in-game, and the mission is stored as a .json file.

The custom missions are stored in the following directories:
C:\Users**<Your Windows User>**\AppData\LocalLow\Wilhelmsen Studios\ReEntry\CustomMissions

4\Virtual Cockpit

VI. VIRTUAL COCKPIT

1. GENERAL

The virtual cockpit is where you control and operate the various systems in each of the supported spacecrafts, and is your home when executing the mission.

Hundreds of switches, selectors and circuit breakers are used to configure hundreds of systems that work together to operate the spacecraft. A spacecraft consists of both primary and backup systems to increase the chance of survival if a system fails.

The panels are designed to group switches together with their function. Attitude controls are usually on one panel, environmental control systems are on another, etc.

2. INTERACTION

Almost every switch in the cockpit can be interacted with. Using the mouse cursor, you can position a switch into the desired position.

Left mouse button is used to insert a circuit breaker or move a switch to the left, and the right mouse button is used to pull a circuit breaker out, or move a switch to the right/up.

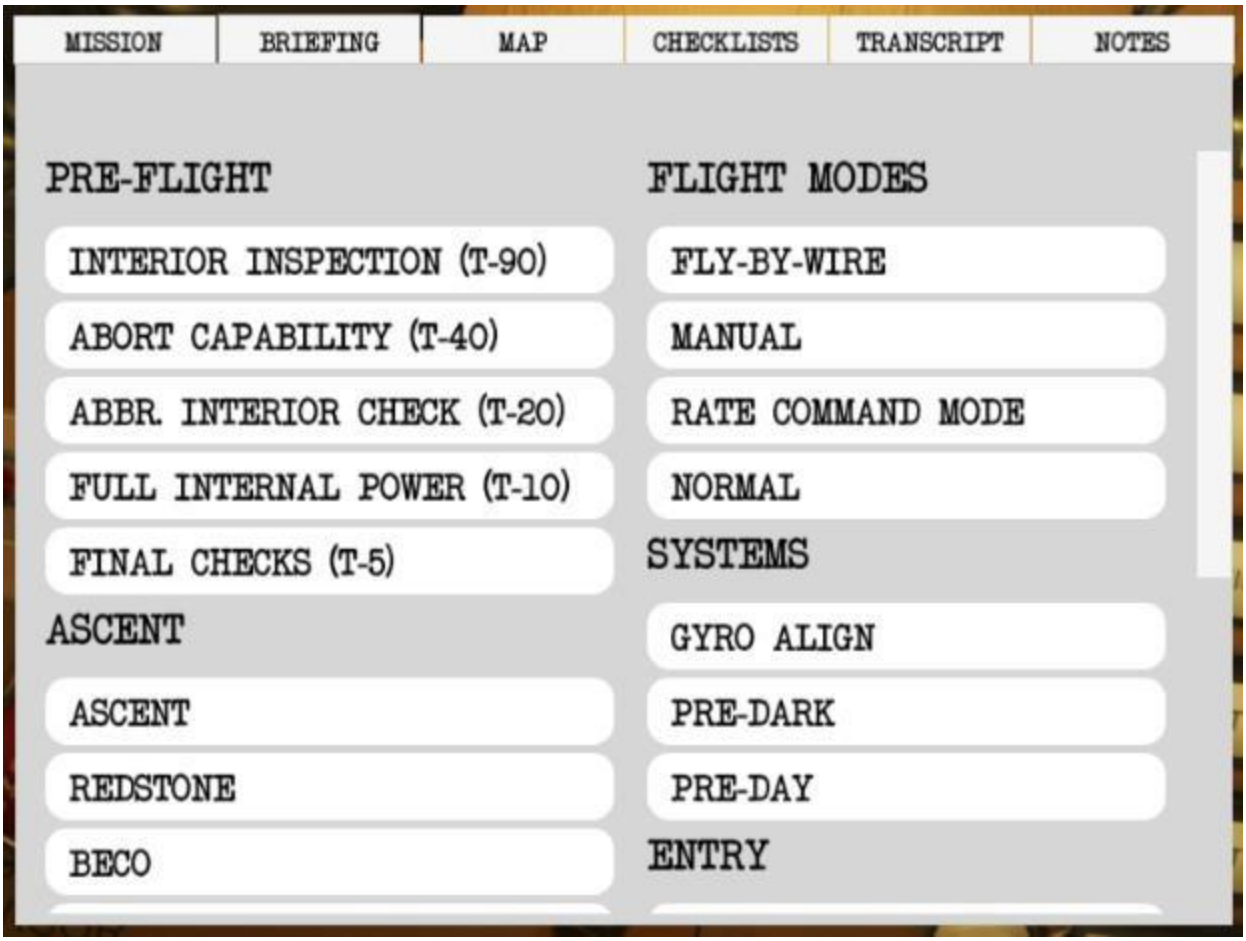
Please see the dedicated flight manual for a specific spacecraft to learn more.

3. HIGHLIGHTING

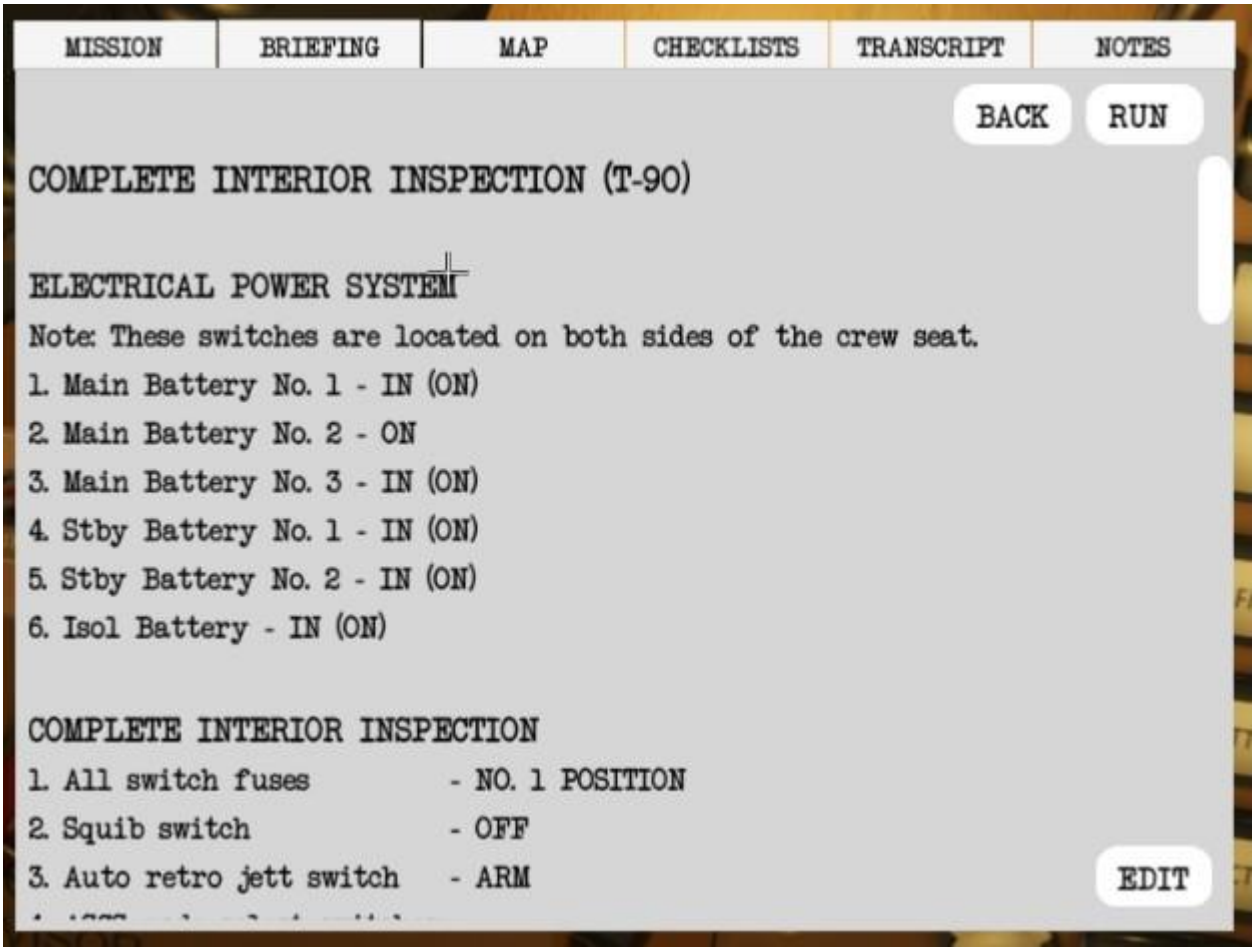
The highlighter system can be used to highlight the switch you are interacting with, and the cockpit cursor can show the switch label and what position it is set to. The cockpit cursor and highlighter system can be disabled from the SETTINGS menu.

4. CHECKLISTS

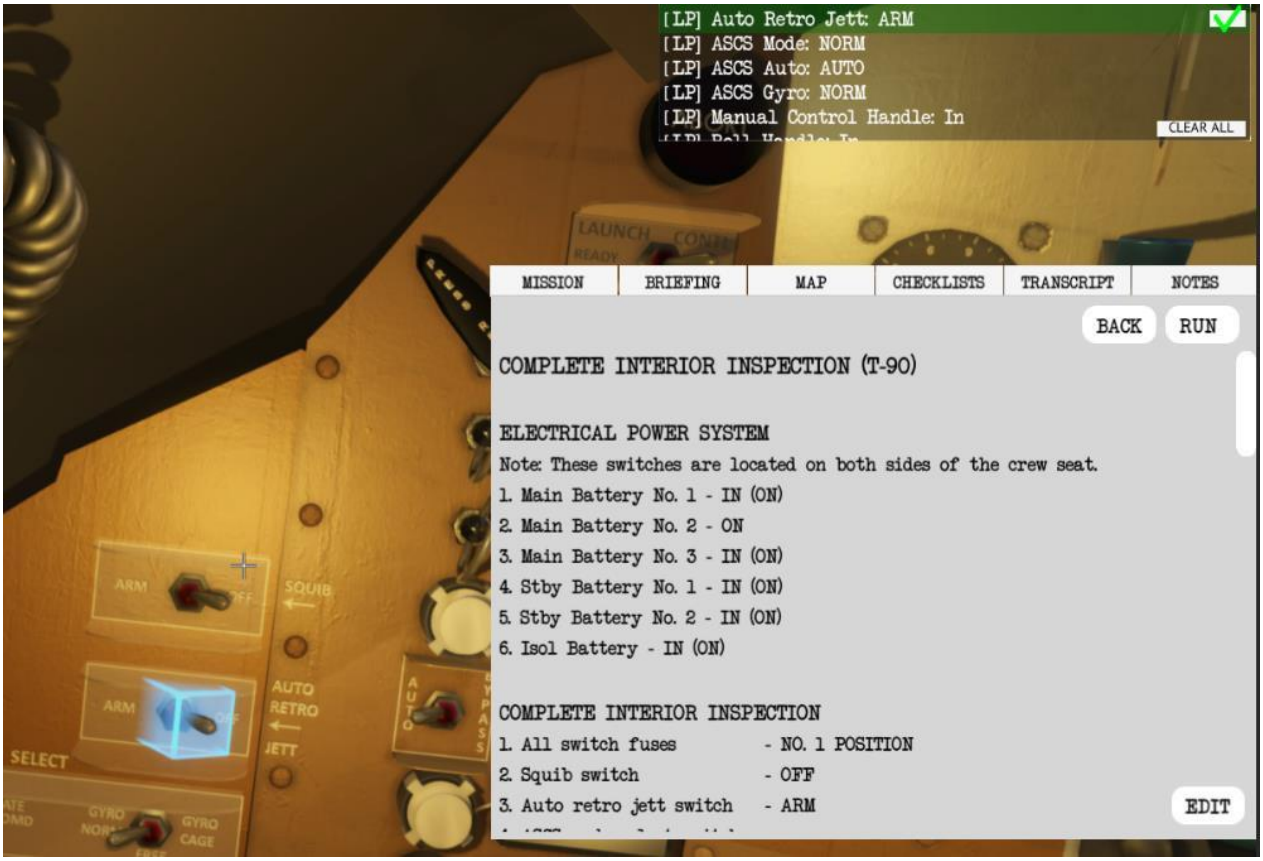
Checklists are used to help you configure the spacecraft. Hundreds of checklists are available and can be followed to configure the spacecraft into a specific configuration. Press M to open the Mission Pad where all the checklists exist.



To open a checklist, press the checklist name, for example INTERIOR INSPECTION in the screenshot above.



You can follow the checklist to complete it manually, or use the RUN feature to move the checklist into a smaller UI that highlights each step in the virtual cockpit one by one. It is very important to refer to the steps in the checklist along with the RUN feature to get the full picture.



Some missions will require that you use the RUN feature, while other missions will let you execute the checklist however you want.

Tip: If you wish to execute a checklist fully manual even if the mission required you to use the RUN system, press the selected checklist, then press RUN, then press CLEAR to remove the steps assigned to the checklist guidance system. This will make the mission proceed to the next step. Remember to also execute the checklist itself manually before actually proceeding with the mission.

Learning to use the checklists is key to flying a spacecraft and you will spend a lot of time in checklists.

5. MOVING AROUND

To move around you can use the arrow keys on the keyboard and the mouse. By holding the middle mouse button in (scroll wheel), you can move the mouse to change the direction you are looking.

A predefined list of camera positions exists and can be accessed using the function keys:
F5 – F11

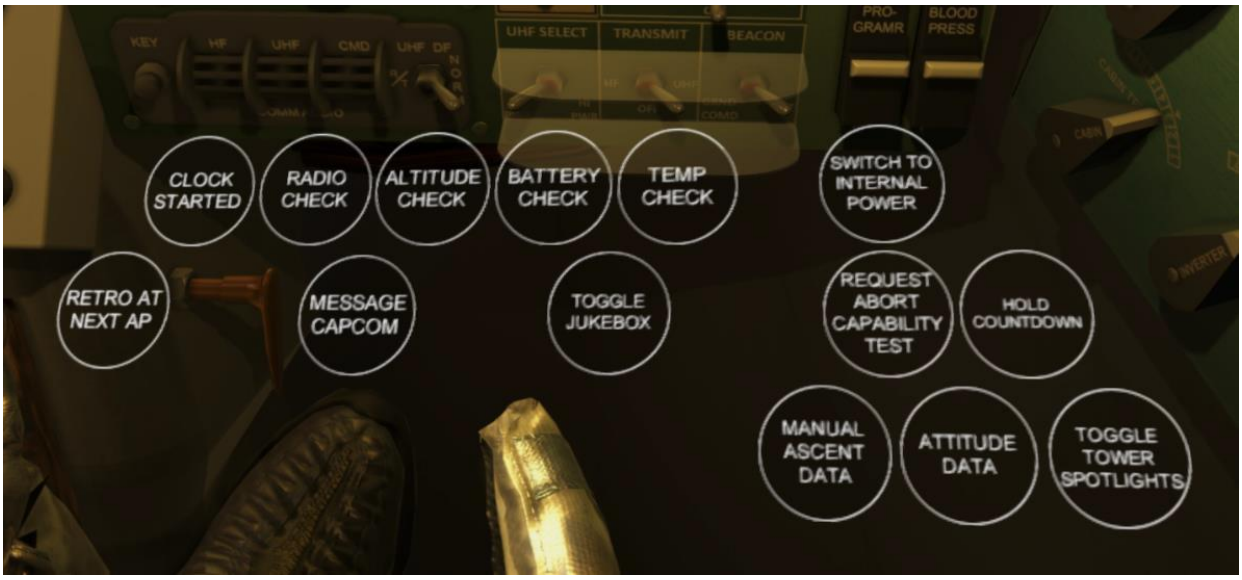
This will quickly move the camera to the pre-defined position. The same pre-defined positions can be accessed from the View menu. Press V to toggle this, and then press the location you wish to move the camera to.

The view menu can also let you toggle the Mission Pad instead of having to press M on the keyboard.

6. BUBBLE FUNCTIONS

Circular buttons, also called Function Bubbles, are used to trigger actions in the virtual cockpit such as sending some radio commands, opening a hatch, installing the COAS in the Apollo Command Module etc.

In Mercury and Gemini, most radio commands can be found as a function bubble, while in Apollo a dedicated Radio Command menu exists. Please see the dedicated flight manuals for more information about this and what each of them do.


CLOCK STARTED
RADIO CHECK
ALTITUDE CHECK
BATTERY CHECK
TEMP CHECK
SWITCH TO INTERNAL POWER
RETRO AT NEXT AP
MESSAGE CAPCOM
TOGGLE JUKEBOX
REQUEST ABORT CAPABILITY TEST
HOLD COUNTDOWN
MANUAL ASCENT DATA
ATTITUDE DATA
TOGGLE TOWER SPOTLIGHTS

5\Mission Editor

VI. MISSION EDITOR

1. GENERAL

The mission editors are used to create your own missions and tutorials for Reentry, or even your own story mode/campaign! All of the missions and campaigns available in Reentry has been made using this tool.

Each of the different supported spacecrafts in Reentry has their own mission editor. There are minor differences between them as each spacecraft has different types of missions. To open the mission editor, either select "Create Mission" from the mission menu for a specific space program such as Project Mercury and launch the mission, or press ESC from an in-game session and select Mission Editor.

A dedicated Mission Editor tutorial exists on the official Reentry YouTube channel. You can use this to create your first mission.

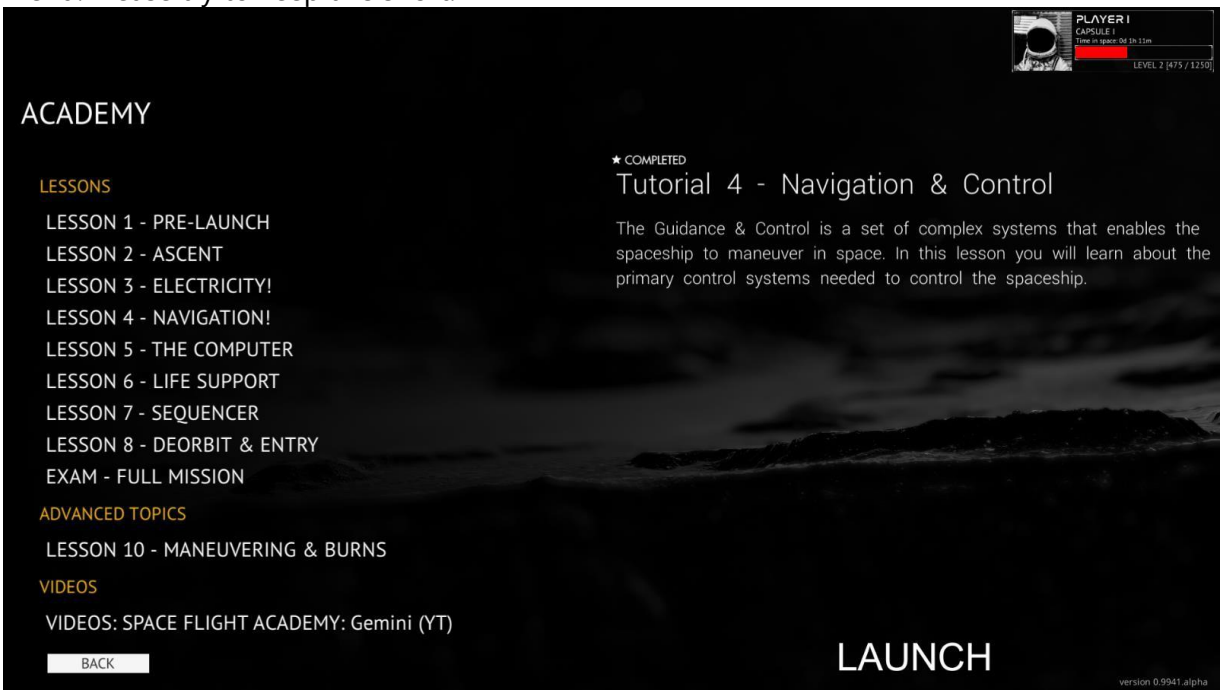
2. MISSION TAB

The Mission Tab contains the main details of the mission such as launch time and date, what panel to load when the mission starts or if it should start from a dedicated mission state (typically a save state used as the start of the mission).

This is also where you set the mission title and description, and details such as the names of the crew.

The Mission ID is used to store the rating and completion of the mission, and thus won't really be used for custom missions.

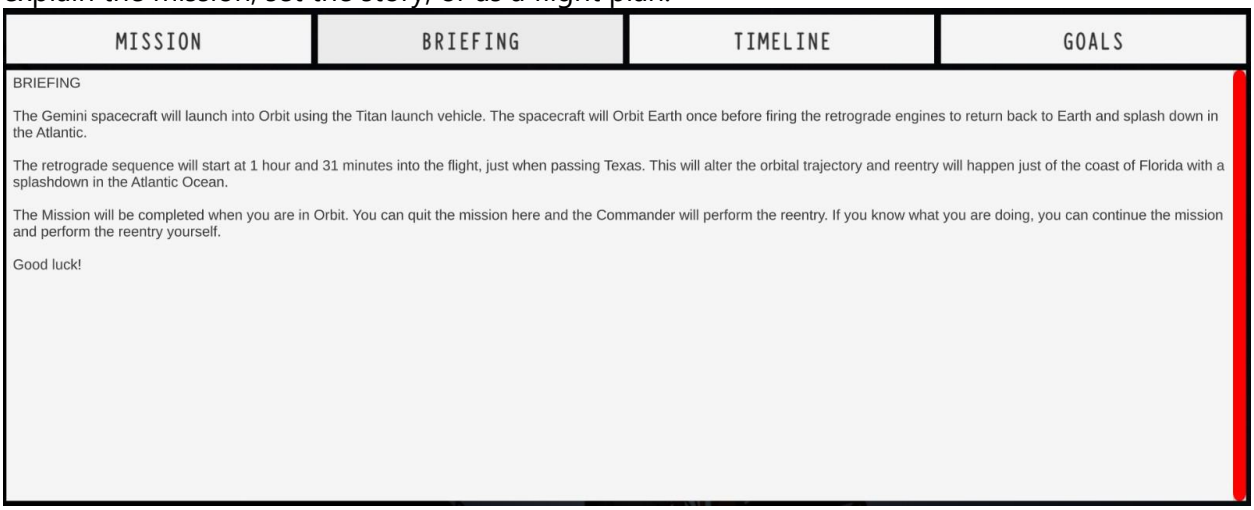
The Mission Title and the Short Description is used when you select the mission from the main menu. Please try to keep this short.



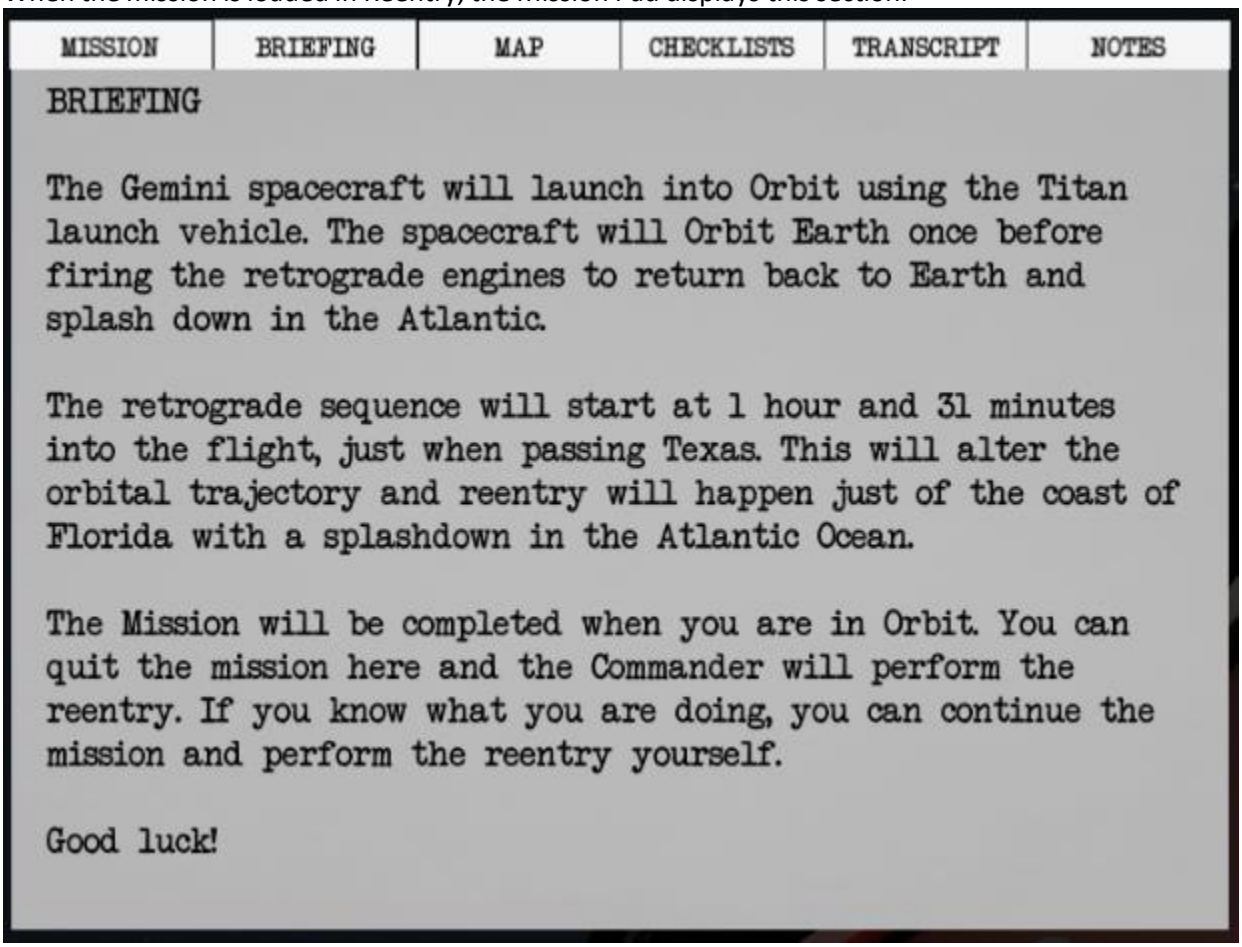
These details are also accessible in the Mission Pad:

3. BRIEFING TAB

The briefing tab is used to set the briefing page of the in-game mission tab. This can be used to explain the mission, set the story, or as a flight plan.

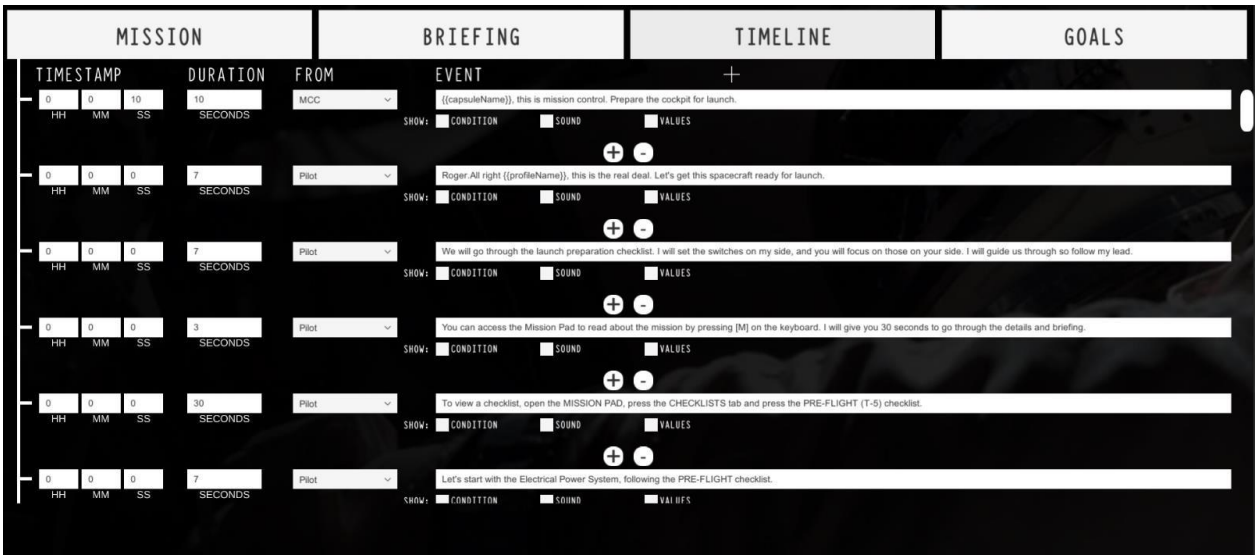


When the mission is loaded in Reentry, the Mission Pad displays this section:



4. TIMELINE

The Timeline tab is used to program the mission flow and sequence. It can display radio messages from MCC, or messages from the flight crew. It can be used to check the state of the mission, for example that the timeline flow won't move forward before the Tower has been jettisoned, or until a switch has been placed into a given position.



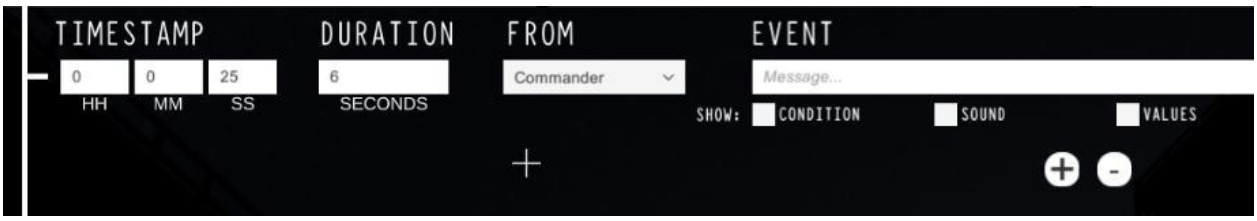
When you first start creating a new mission, the TIMELINE will be empty. Press the + button to add a new command. This will let you write the message that will be displayed, from who the message is from, and what action that requires to be completed before moving to the next instruction.

Timing is the hardest thing to get right. Even if you have access to the direct timestamp of when you want a message to come, it's typically not used that way (only in rare cases). What you typically want to do is to design the sequence of mission based on events, and use the DURIATION to set how long it will take from when the player first sees the message to when it is able to proceed (if the action is of type message).

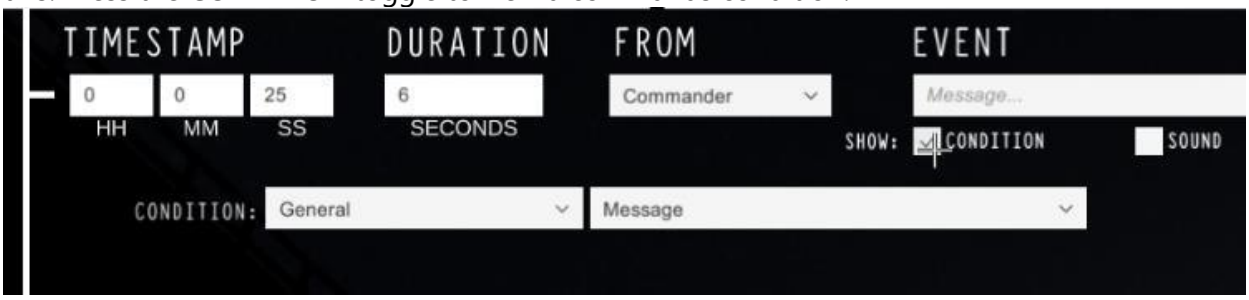
Actions will decide when a message is completed outside of the DURATION field. For example, you can ask the command to proceed once a switch is set to its UP position, or if an engine is ignited, or if a checklist is completed.

Let's try to create a simple mission. In a new Mission, press the + button to add a new command:

A new blank command is added:



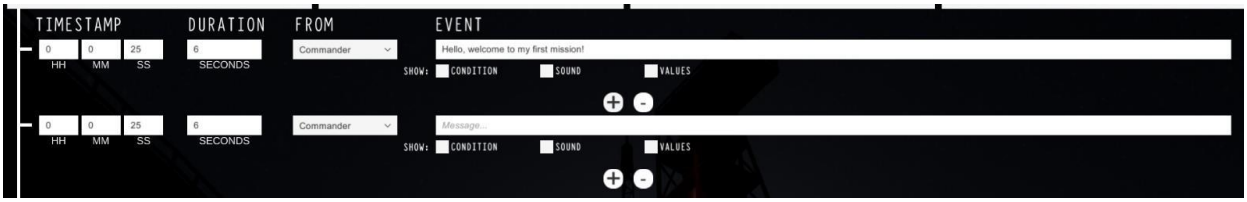
Most commands are messages, so the default condition is set to Message. Let us take a look at this. Press the CONDITION toggle to view a commands condition.



As you can see, the condition is set to General->Message.

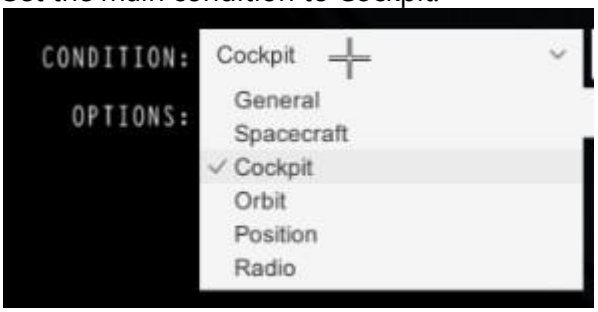
On the EVENT columns, the message can be filled in. Press it and type “Hello, welcome to my first mission!”.

Toggle off the CONDITION to hide the condition for this message, and then press the + symbol below the above instruction to add a new instruction.
Your mission should look like this:

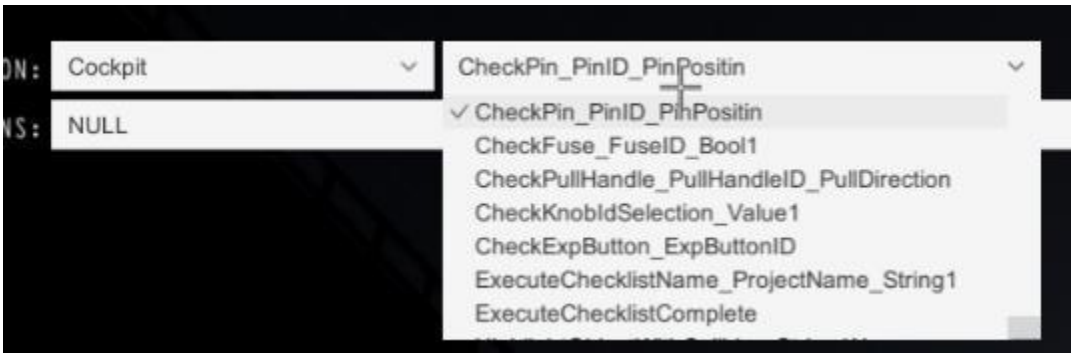


In the EVENT column, type in “Please arm the squibs”, and toggle the CONDITION.

Set the main condition to Cockpit:



Then make sure the condition action is set to this:

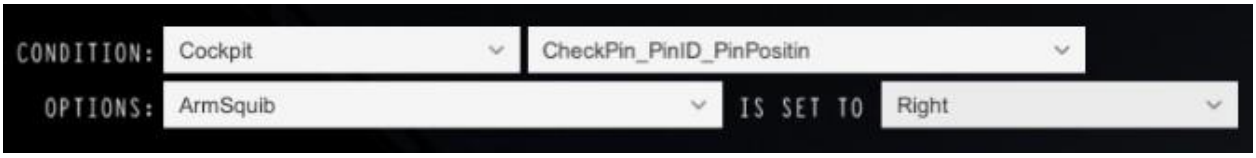


The CheckPin_PinID_PinPosition will be used to decide if this command is completed or not, and will bypass the DURATION setting. The mission will automatically proceed to the next command once a pin switch is set to the required position.

The OPTIONS section will now show two dropdowns. Set the first one to ArmSquib:

And set the 2nd option to RIGHT:

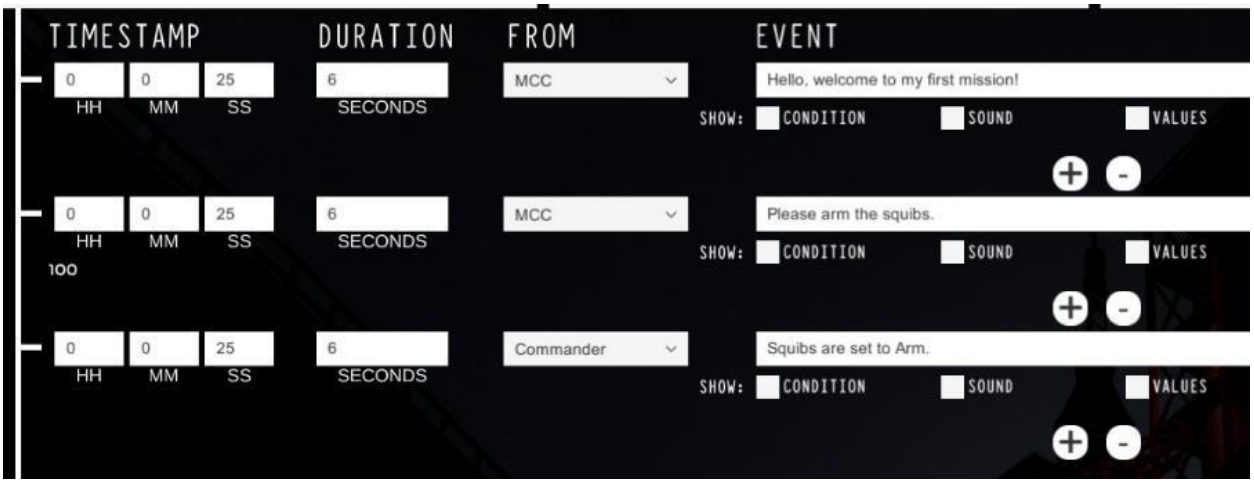
The condition should now look like this:



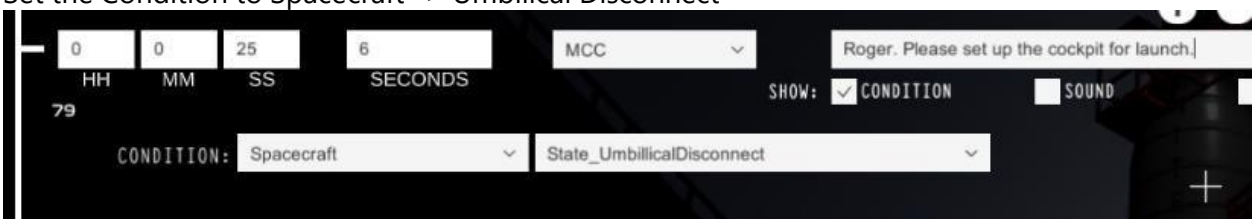
This means that the ArmSquib will need to be set to RIGHT for the mission to proceed to the next command.
Toggle off the Condition again for the command and add a new command below it.
Set the first commands FROM column to MCC, and the 2nd commands FROM column to MCC.
Your mission should now look like the following:

The new command should come from the Commander and the message should be: "Squibs are set to Arm."

The mission timeline should now look like this:



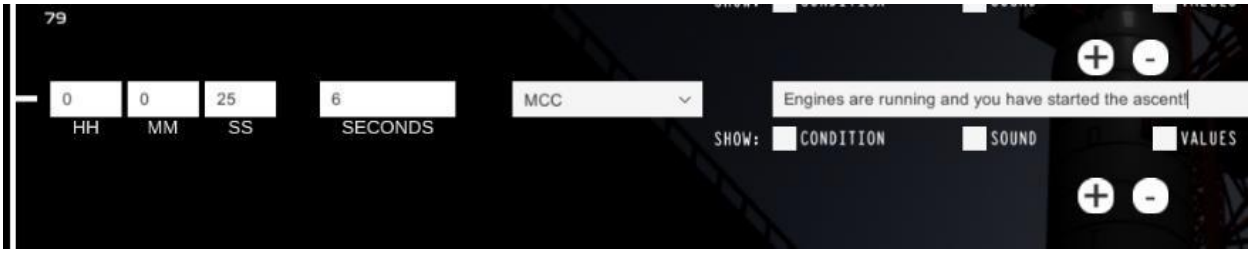
Add a new command and set the message to come from MCC, and the message set to: "Roger. Please set up the cockpit for launch."
Set the Condition to Spacecraft -> Umbilical Disconnect



This will show the message after the previous message, but it will not proceed to the next command before the umbilical has been disconnected (happens when the rocket is released from the tower).

It is important to know that the message on this command will be displayed before the umbilical has been disconnected, not the command will not be completed until the action has happened. You design the mission with this mindset. Show a message but do not proceed until the instruction is completed.

Add a new instruction:



This message will show immediately after you have started the ascent, since the umbilical disconnect has happened. You are now ascending towards orbit, so add the message to "Engines are running, and you have started the ascent".

KEYWORDS

Some reserved keyword surrounded by {{x}} can be used to insert variables into the messages, such as names, mission time, ETAs and so on. When writing a communication message, you can for example write: {{capsuleName}}, this is X, please do Y. The {{capsuleName}} keyword will be converted into what the player has configured in the Profile menu.

A complete list can be seen below:

Mercury

```
{{Value1}}
{{Value2}}
{{Value3}}
{{String1}}
{{String2}}
{{String3}}
{{profileName}}
{{capsuleName}}

{{ap}}
{{pe}}
{{inc}}
{{period}}
{{endOfOrbitTime}}
{{timeToAp}}
{{timeToPe}}
{{timeToRetro}}
{{retroTime}}
{{currentRadioStation}}
{{g}}
{{cabinPSI}}
{{voltMainBus}}
```

```
{{voltIsolBat}}
{{ampsMainBus}}
{{fuelAB}}
{{oxygenAB}}
{{pitch}}
{{roll}}
{{yaw}}
{{bloodStatus}}
```

Gemini

```
{{Value1}}
{{Value2}}
{{Value3}}
{{String1}}
{{String2}}
{{String3}}
{{profileName}}
{{capsuleName}}
{{coPilotName}}

{{ap}}
{{pe}}
{{inc}}
{{period}}
{{endOfOrbitTime}}
{{timeToAp}}
{{timeToPe}}
{{timeToRetro}}
{{timeToTransferBurn}}
{{retroTime}}
{{currentRadioStation}}
{{g}}
{{cabinPSI}}
{{voltMainBus}}
{{ampsMainBus}}
{{pitch}}
{{roll}}
{{yaw}}
{{valueInCore_Value3}}

{{IVIs}}
```

Apollo

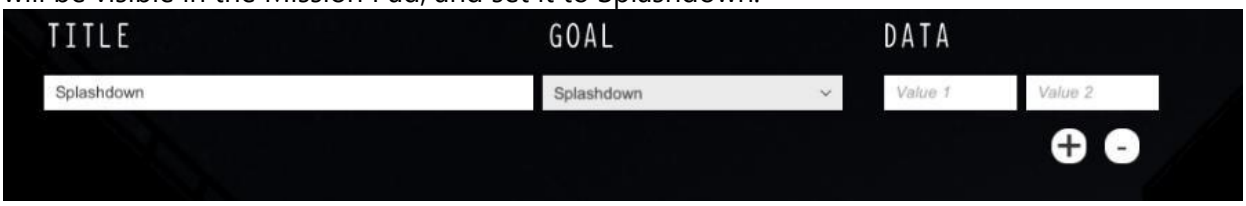
```
{{Value1}}
{{Value2}}
{{Value3}}
{{String1}}
{{String2}}
{{String3}}
{{profileName}}
{{capsuleName}}

{{ap}}
{{pe}}
{{inc}}
{{period}}
```

```
{{endOfOrbitTime}}
{{timeToAp}}
{{timeToPe}}
{{timeToRetro}}
{{retroTime}}
{{currentRadioStation}}
{{g}}
```

5. GOALS

The Goals tab is used to set the goals of the mission. A mission can be completed in two ways: On Splashdown or then a command has the action General->Complete Mission.
The complete mission action is usually used during tutorials or if the mission will end before splashing down.

To add a new goal, press the GOALS tab and press the + button. Enter the goal description that will be visible in the Mission Pad, and set it to Splashdown.



This is a required goal and should be part of all missions no matter what. If you design the mission to complete using the command action, keep this goal in there and set the title to "Complete the mission" or "Complete the tutorial" etc, something that fits the purpose of the mission.

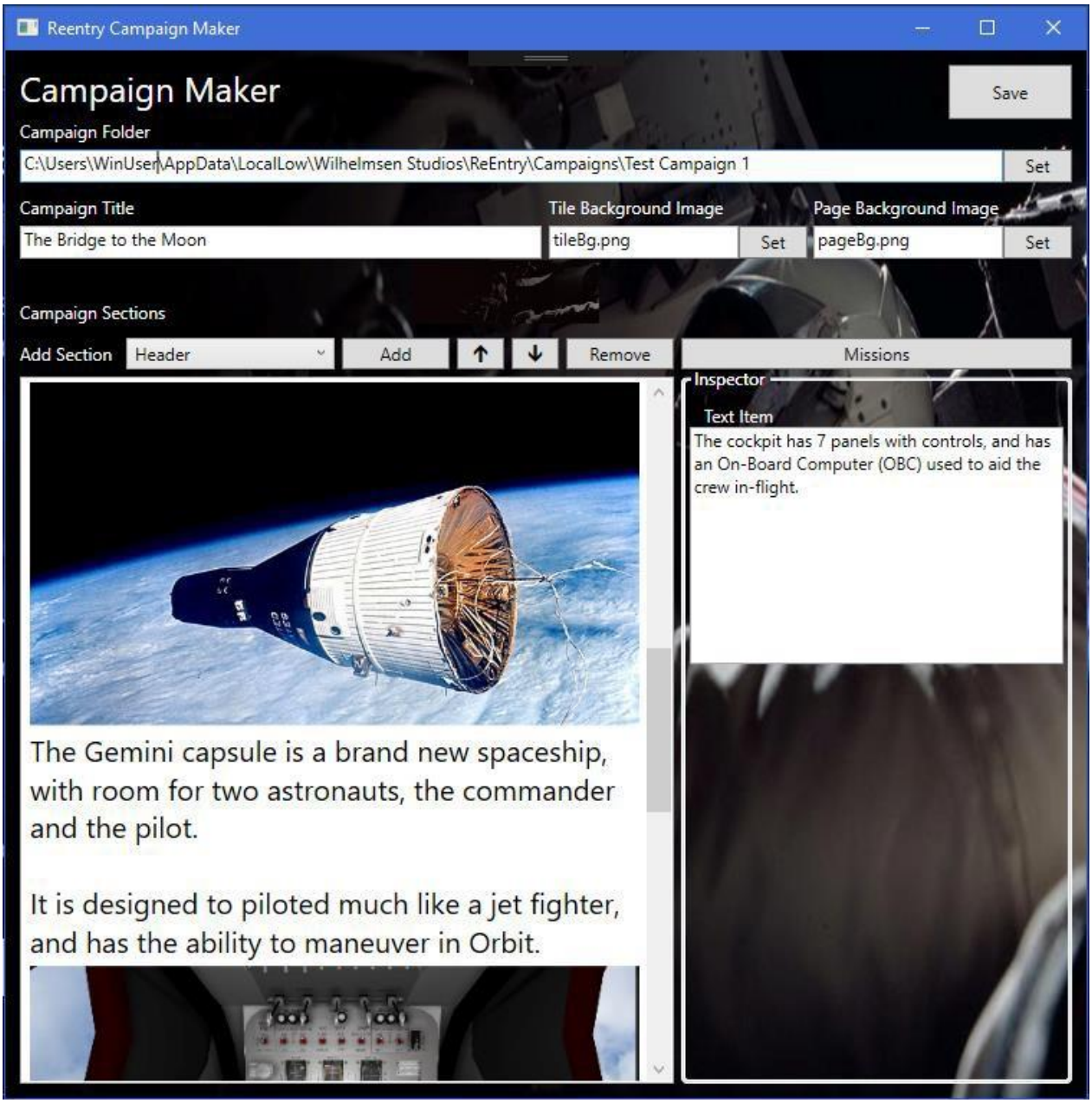
Other goals can be added as well, so feel free to be creative with this. By completing goals, you can get a better score from completing the mission.

6\Campaing Maker

IX. CAMPAIGN MAKER

1. GENERAL

The campaign maker allows you to make custom campaigns for Reentry and is an external tool found on the GitHub page for Reentry.



A campaign is built up of the following items:

- Campaign Title
- Tile
- Campaign page background

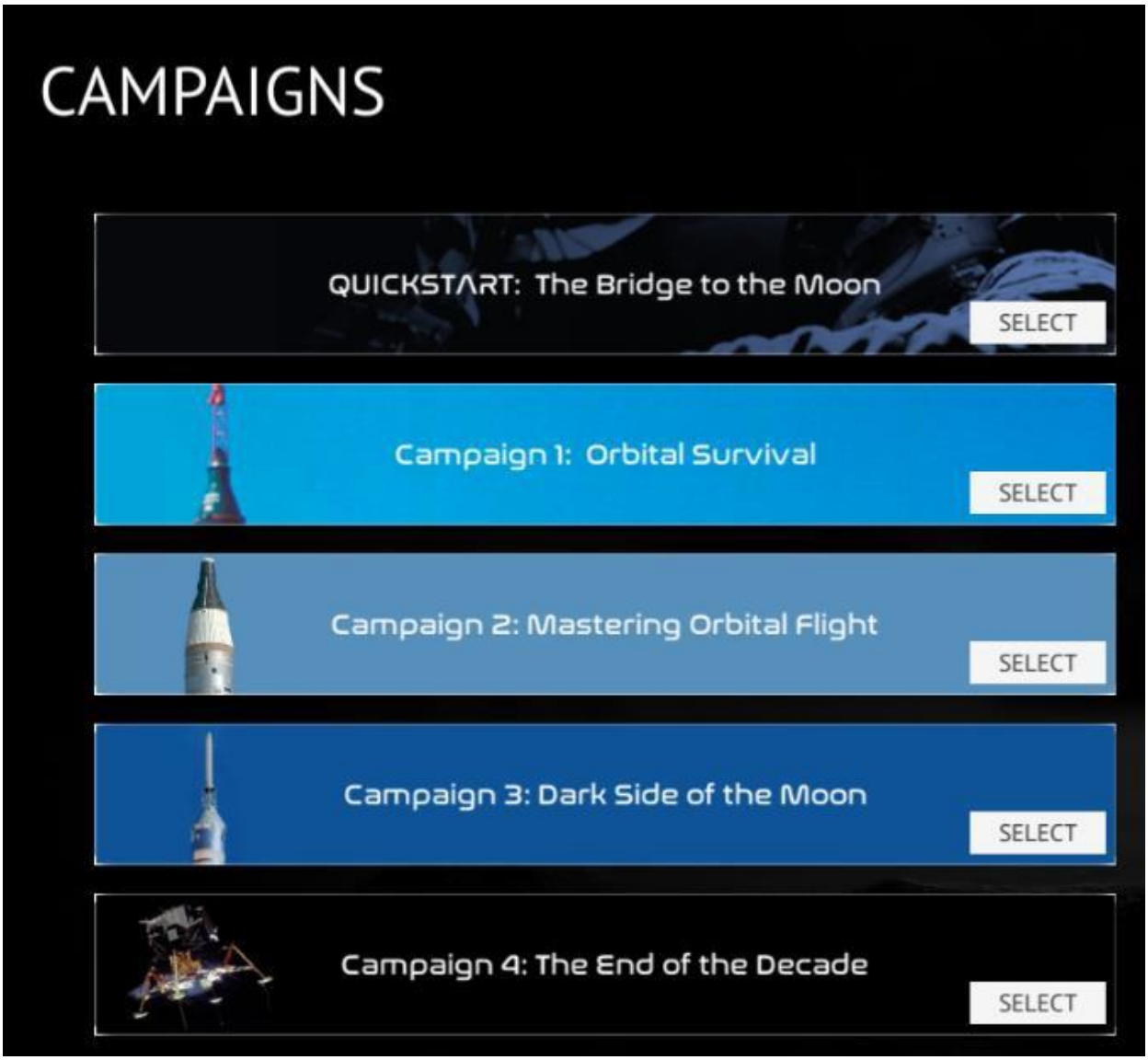
- Campaign page sections, describing the campaign using Headers, longer text or images (PNG!)
- A set of Missions

The Title and the Tile Background Image builds up the tile that makes it selectable in the campaign menu if your campaign is located in the Campaigns folder:
Steam:
C:\Users**<Your Windows User>**\AppData\LocalLow\Wilhelmsen Studios\ReEntry\Campaigns

Missions will need to be created outside of the campaign maker and stored in the campaign folder. The Missions window will let you set some additional data for the mission, and the relative path to the mission itself (typically in a folder named Missions).

A campaign is really just a simple system that lets you put together images and text to create a setting, and show a message between each mission, and a campaign completed text.

The selection screen looks like this, with each of the campaign you have installed listed:


CAMPAIGNS
QUICKSTART: The Bridge to the Moon
SELECT
Campaign 1: Orbital Survival
SELECT
Campaign 2: Mastering Orbital Flight
SELECT
Campaign 3: Dark Side of the Moon
SELECT
Campaign 4: The End of the Decade
SELECT

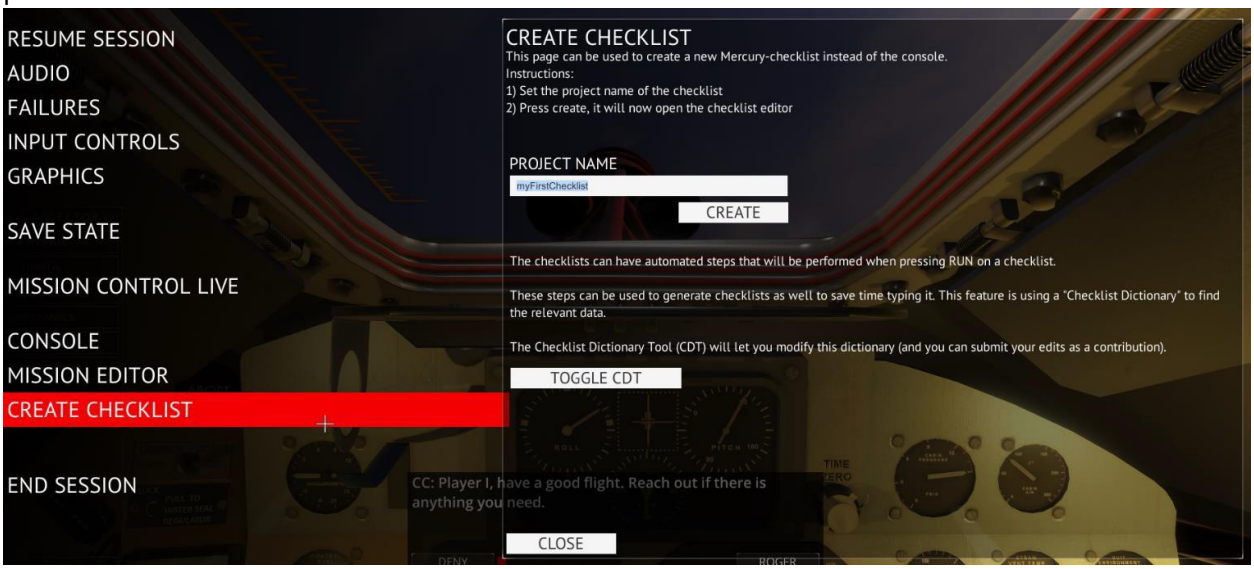
7\Checklist Editor

X. CHECKLIST EDITOR

1. GENERAL

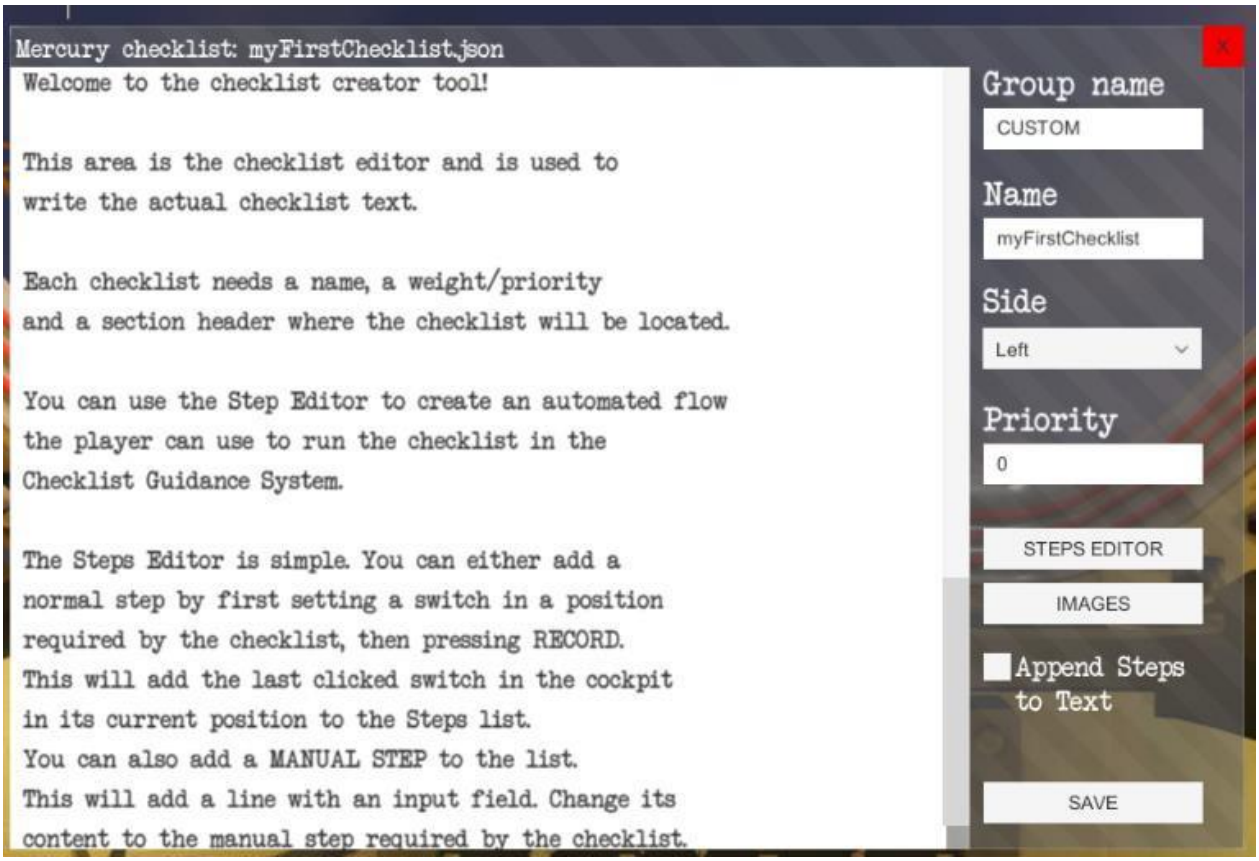
The checklist editor can be opened for any in-game checklist by pressing the EDIT button in the lower-right corner of the checklist you have opened, or by pressing ESC->Create Checklist.

Let's create a new checklist. Start the Mercury -> In-Orbit scenario, open the in-game menu and press CREATE CHECKLIST:



Name the checklist "myFirstChecklist" and press CREATE. This will not yet create any files on your system, but instead create a space in-memory.

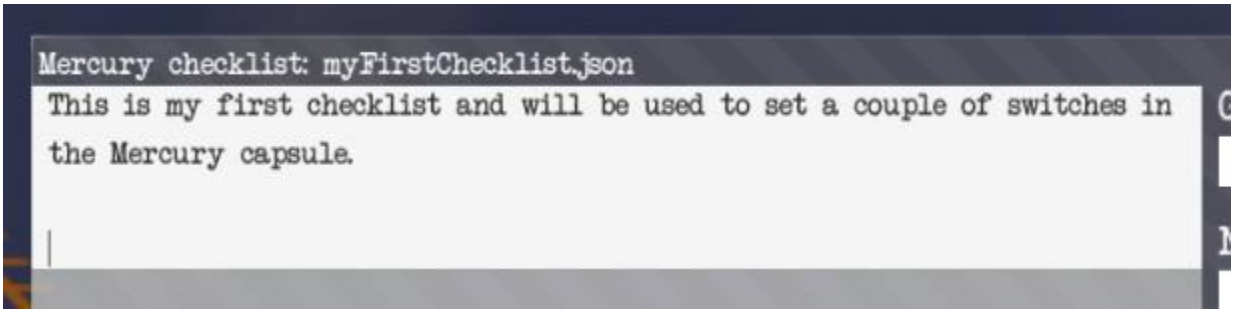
After pressing CREATE, hit ESC to hide the main menu and gain access to the checklist editor:



A default template will describe how the checklist editor works.

Press this text and delete it.

Type the following message into the Text area:



Then set the Group Name to MY CHECKLISTS. This will be the header of the section your checklist will exist in.

Set the name to My First Checklist. This will be the name of the checklist in the mission pad, under the MY CHECKLISTS section.

The Priority is a little bit hard to understand but it will be the order of the checklists in your section. However, this is a global priority and will be used to match with other checklists as well.

You can use the console to see the priority of all loaded checklists, and can help you pick a number. This is typically used to make your section in-between the built-in/default checklists and so on.

For now, we will set this to 90000 to move it in the bottom of the left side checklists.

If you wish to support the RUN feature of the checklist, the STEPS EDITOR will be used to create the flow of steps.

The Images section can be used to render images inside the checklist, and the Append Steps to Text can be used to automatically generate the checklist text based on the STEPS EDITOR.

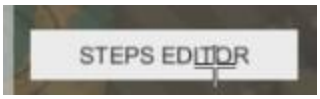
Note: A checklist can be fully complete without any RUN feature. Simply type the checklist into the text editor. This will be the simplest form of a checklist, and will simply just show the text when the player opens it.

Adding your first step and making support for the RUN feature

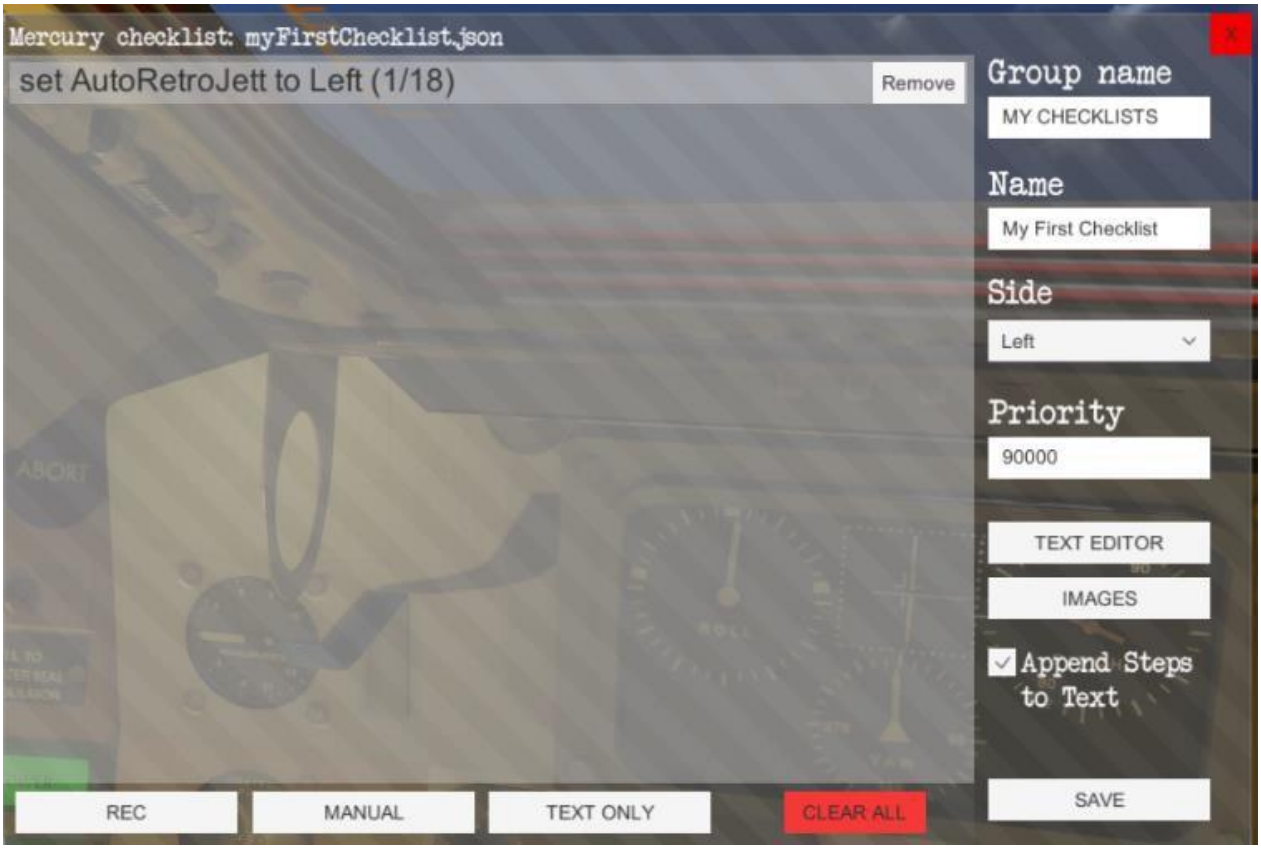
Move the checklist editor out of the view, and move the AUTO RETRO switch to the left.

This will store your last action in-memory, and you can use this to add the step to the checklist steps.

Then click on the STEPS EDITOR to open the steps view of the checklist:



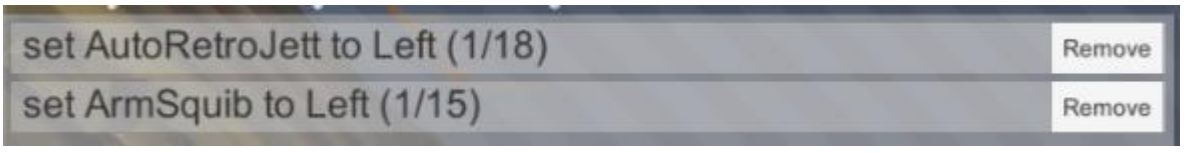
This is currently blank. However, the switch we just interacted with is stored in-memory, and you can use the RECORD button to add it to the list. Press RECORD now.



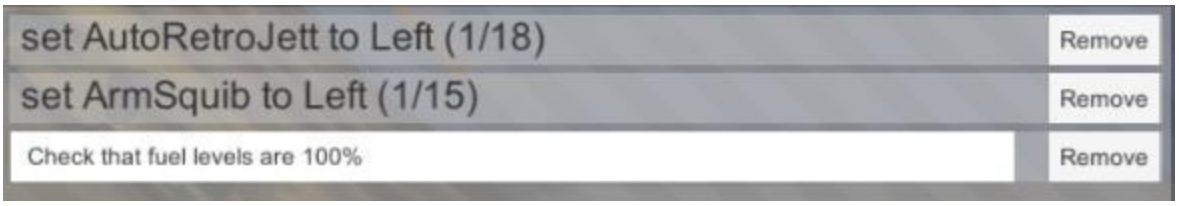
As you can see, the switch has been added to the checklist.

Now, in the cockpit, move the SQUIB switch to the left (even if its already in this position – we want to store its position into memory) and press REC in the editor.

You should now see two switches in the editor:



The next step we want is a MANUAL step. This is the type of step that the player will need to complete manually using the checkmark button. Press MANUAL and type "Check that fuel levels are 100%"

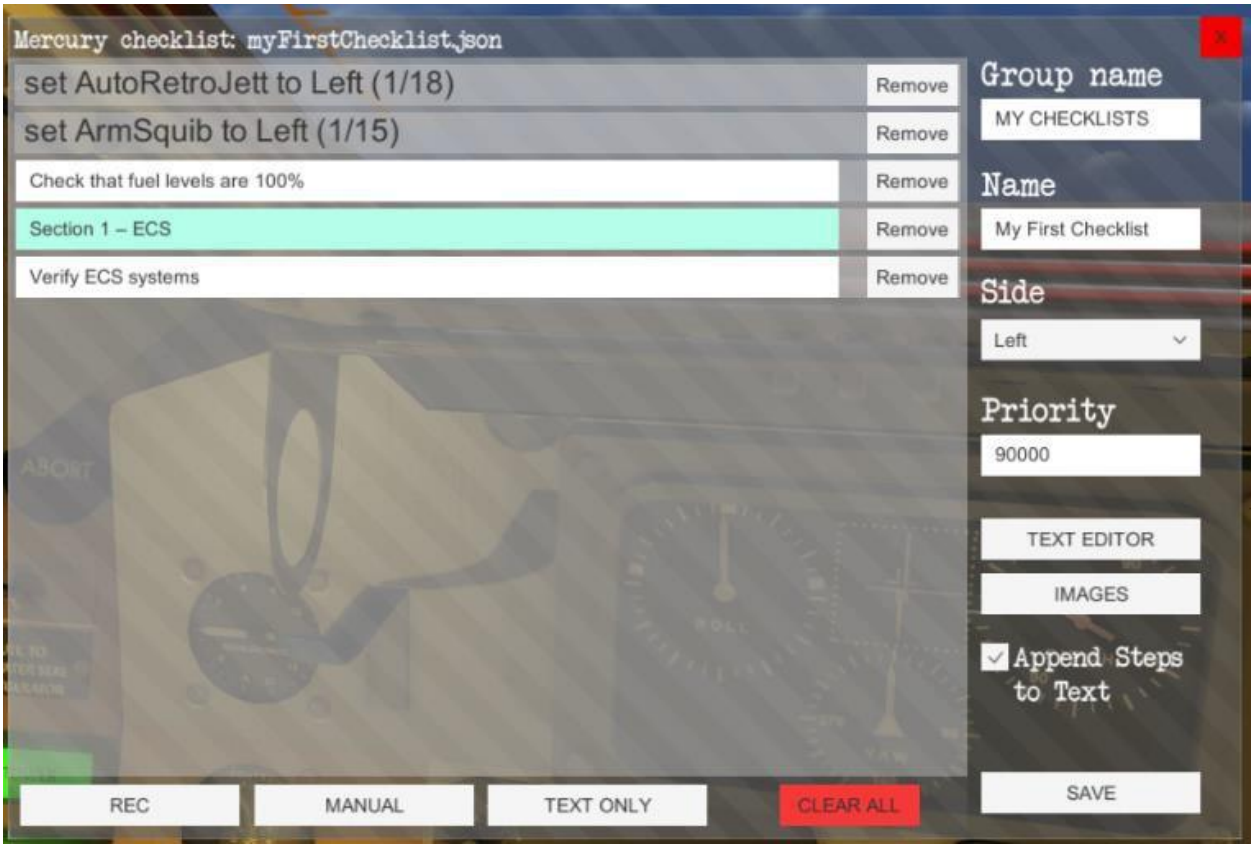


On the right side, tick the Append Steps to Text. We want the STEPS to be used to generate the checklist instead of having to manually type everything in the checklist body.

The last step we will add is a TEXT ONLY element. This element will not be visible in the UI while running the checklist, and is only used if you generate the checklist text using the Append Steps to Text. This can be used to add section headers or notes that are visible in the checklist only.

Add a TEXT ONLY element and type "ECS".

Then add a new manual step saying "Verify ECS systems". Your checklist should look like this:

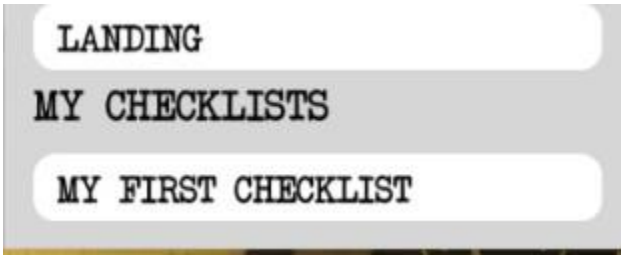


Let us take a look at our new checklist. Press SAVE to store the checklist on your file system.

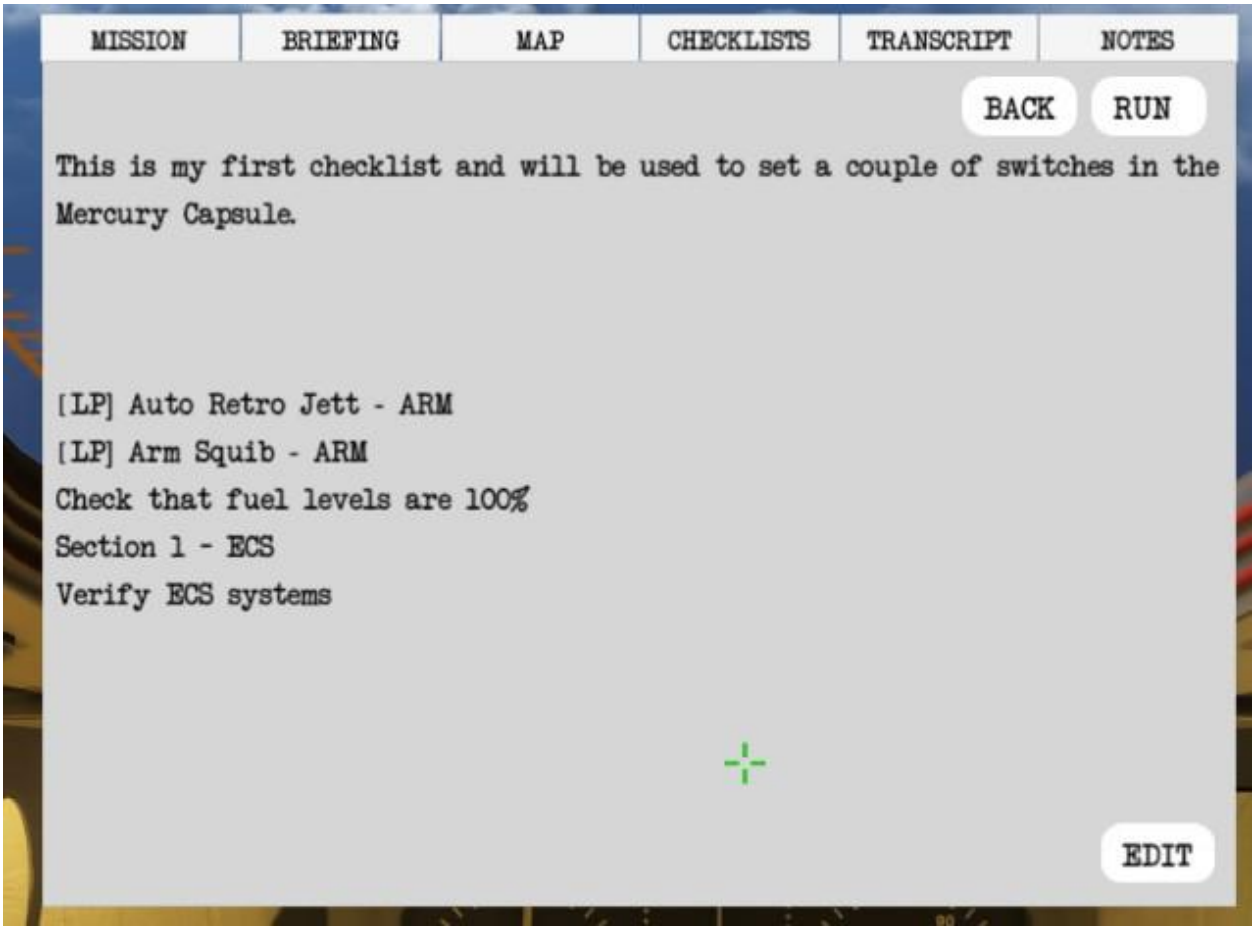
This will now automatically load when you open the Mission Pad.

Note: Please note that loading checklists happens when you open the checklist tab for the first time. So if you create a new checklists, and its not visible in the list, re-load the mission to see it again. This applies only to new checklists, so if you are only making changes to a checklist, those changes should be visible immediately once you re-open the checklist in the Mission Pad.

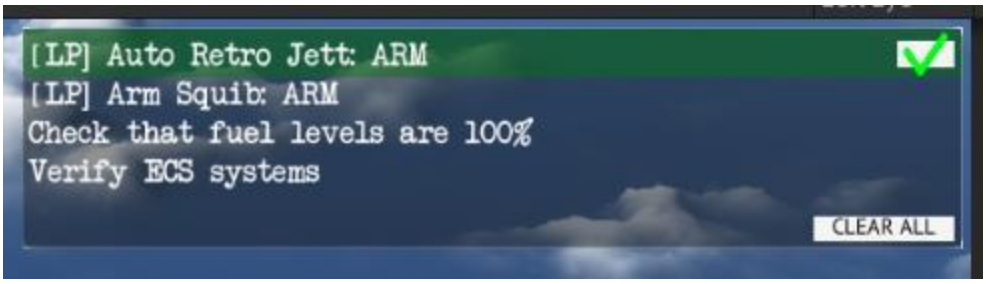
When you now open the checklist, you can find it in the Mission Pad:



You can see the contents of it by opening it:



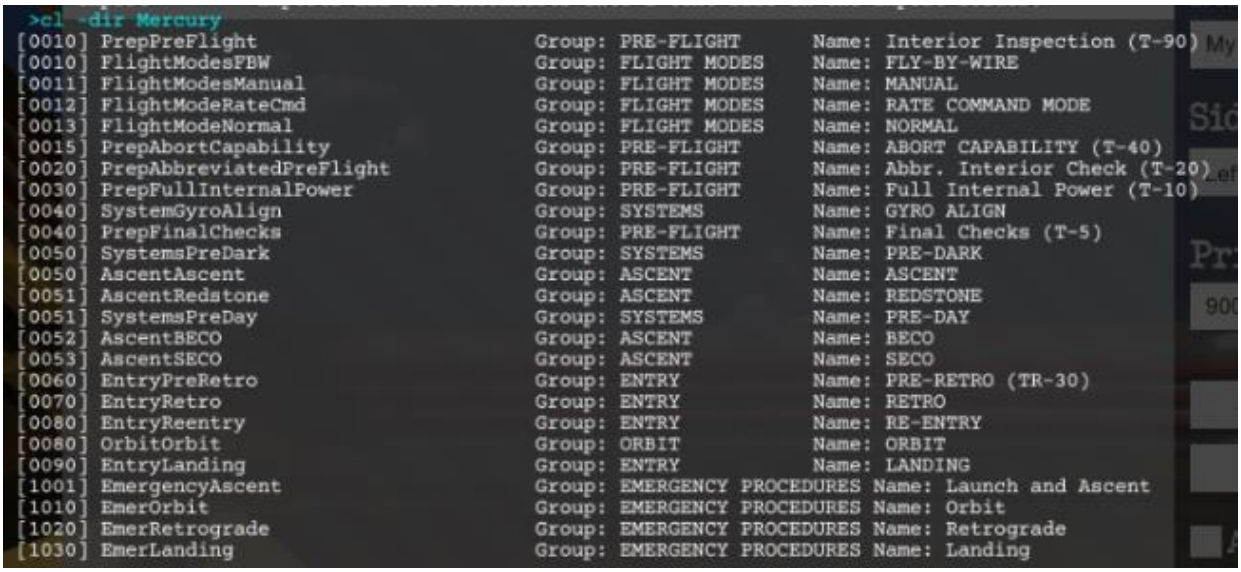
Pressing run will execute the steps. Notice that the TEXT ONLY is visible in the editor, but not in the steps UI.



PRIORITY

In the console (you will learn about this later), type cl to get access to similar tools when creating checklists.

Type cl -dir Mercury to list all the checklists in the Mercury program and their priority.


>cl -dir Mercury
[0010] PrepPreFlight Group: PRE-FLIGHT Name: Interior Inspection (T-90)
[0010] FlightModesFBW Group: FLIGHT MODES Name: FLY-BY-WIRE
[0011] FlightModesManual Group: FLIGHT MODES Name: MANUAL
[0012] FlightModeRateCmd Group: FLIGHT MODES Name: RATE COMMAND MODE
[0013] FlightModeNormal Group: FLIGHT MODES Name: NORMAL
[0015] PrepAbortCapability Group: PRE-FLIGHT Name: ABORT CAPABILITY (T-40)
[0020] PrepAbbreviatedPreFlight Group: PRE-FLIGHT Name: Abbr. Interior Check (T-20)
[0030] PrepFullInternalPower Group: PRE-FLIGHT Name: Full Internal Power (T-10)
[0040] SystemGyroAlign Group: SYSTEMS Name: GYRO ALIGN
[0040] PrepFinalChecks Group: PRE-FLIGHT Name: Final Checks (T-5)
[0050] SystemsPreDark Group: SYSTEMS Name: PRE-DARK
[0050] AscentAscent Group: ASCENT Name: ASCENT
[0051] AscentRedstone Group: ASCENT Name: REDSTONE
[0051] SystemsPreDay Group: SYSTEMS Name: PRE-DAY
[0052] AscentBECO Group: ASCENT Name: BECO
[0053] AscentSECO Group: ASCENT Name: SECO
[0060] EntryPreRetro Group: ENTRY Name: PRE-RETRO (TR-30)
[0070] EntryRetro Group: ENTRY Name: RETRO
[0080] EntryReentry Group: ENTRY Name: RE-ENTRY
[0080] OrbitOrbit Group: ORBIT Name: ORBIT
[0090] EntryLanding Group: ENTRY Name: LANDING
[1001] EmergencyAscent Group: EMERGENCY PROCEDURES Name: Launch and Ascent
[1010] EmerOrbit Group: EMERGENCY PROCEDURES Name: Orbit
[1020] EmerRetrograde Group: EMERGENCY PROCEDURES Name: Retrograde
[1030] EmerLanding Group: EMERGENCY PROCEDURES Name: Landing

8\Console

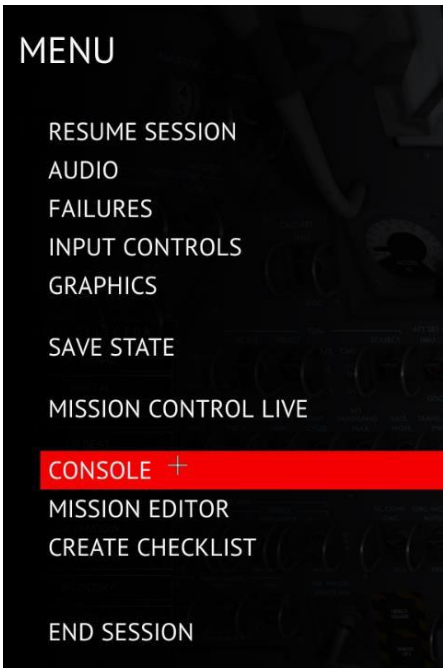
VII. CONSOLE

1. GENERAL

The console can be used to accomplish various tasks or launch tools such as the checklist editor (instead of using the UI).

The main console commands will contain some common commands and some commands that vary between the modules (Project Mercury, Project Gemini, Project Apollo).

The console can be toggled using the in-game menu -> console:



It can also be closed with the same option, or if you press the console input field and press ESC.

The console can also be toggled with a hot key bindable in the input settings:

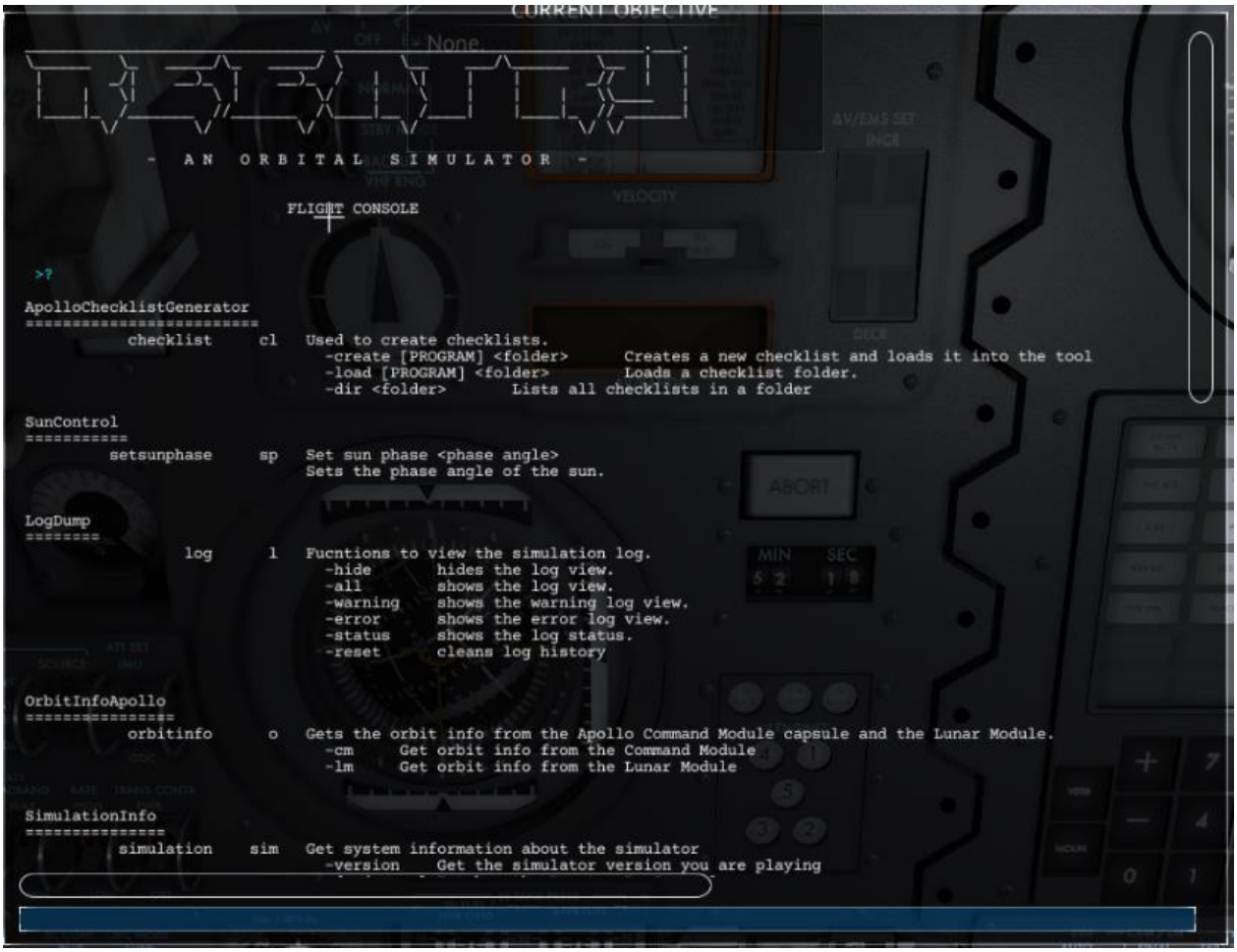
By default, the back quote button is assigned to the console. Press this key to toggle the console (while the console input field is not in focus).

2. COMMANDS

The available commands in the console can be listed by executing the ? command.

Type ? in the field and press ENTER to execute the command.

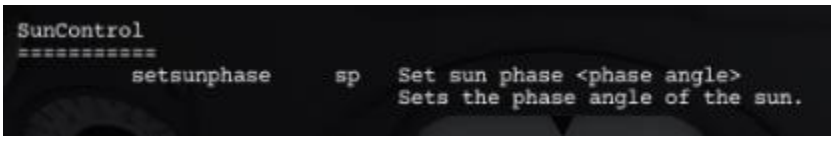
A command can be executed by typing in the command you wish to run in full, or by using the short format.



Set Sun Phase

Command: setsunphase

Short: sp



Sets the sun phase angle. This is useful if you need to change the position of the sun. It requires an angle as its input.

For example:

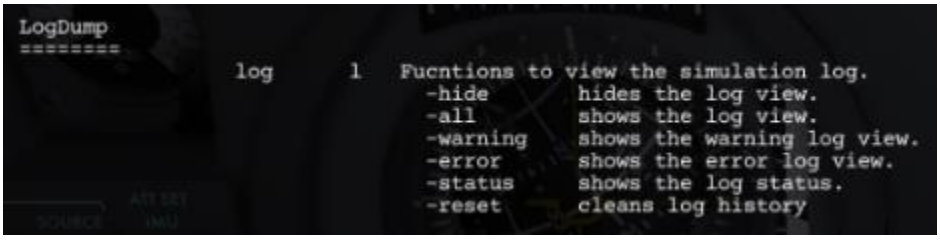
setsunphase 90

This command will set the phase of the sun to 90. This will override the current sun position relative to Earth to 90 degrees.

LogDump

Command: log
Short: l

Lists the log in the console.



For example:

Log -status

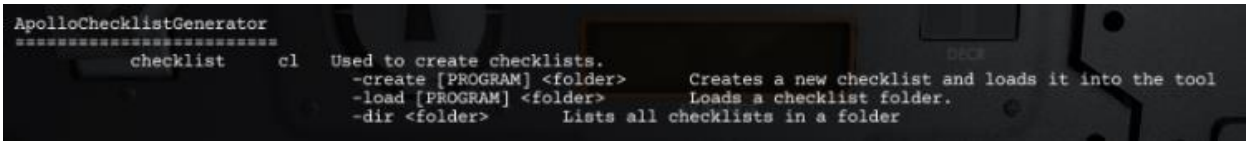
This will show you the log status of the current session.

Log -error

This will list all the 9 errors shown in status:

Checklist

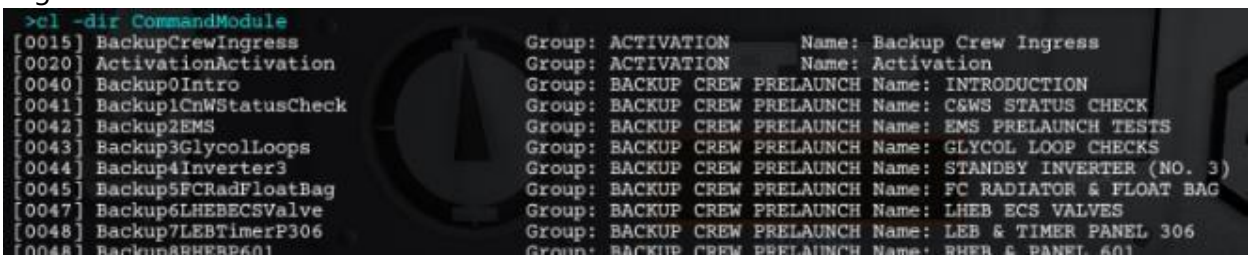
Command: checklist
Short: cl



Used to list checklists, create checklists and load checklists. This is similar to using the UI (Edit or Create).

For example:

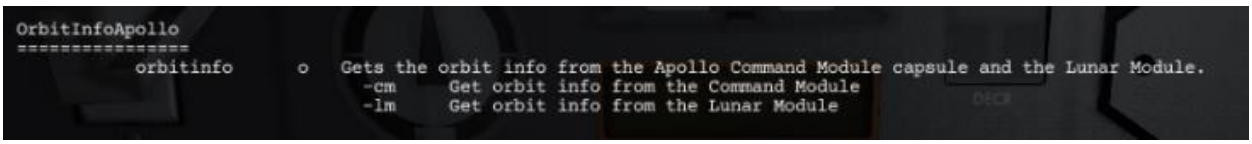
Log -dir CommandModule



Orbit Info

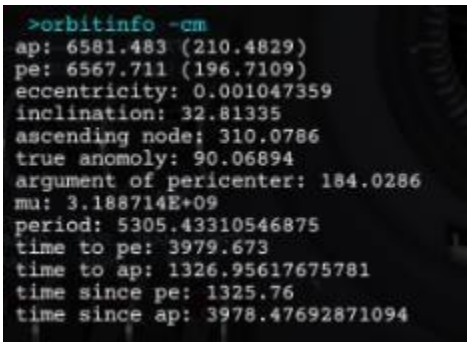
Command: orbitinfo
Short: o

Renders the current orbital parameters.



For example:

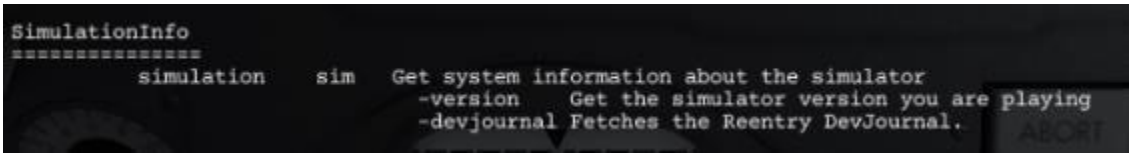
Orbitinfo -cm



Simulation Info

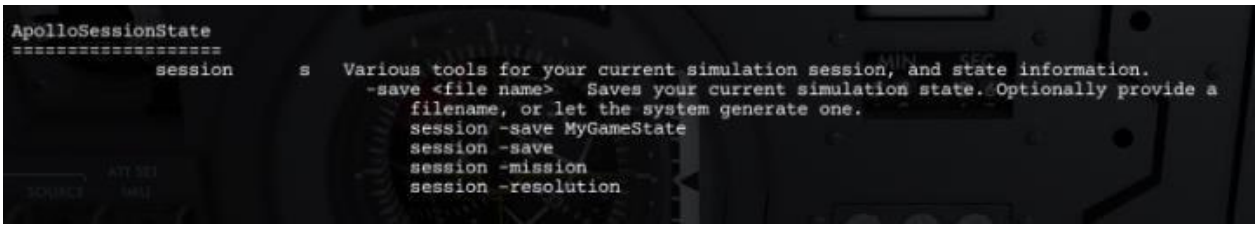
Command: simulationinfo
Short: sim

Used to show information about the version of the game.

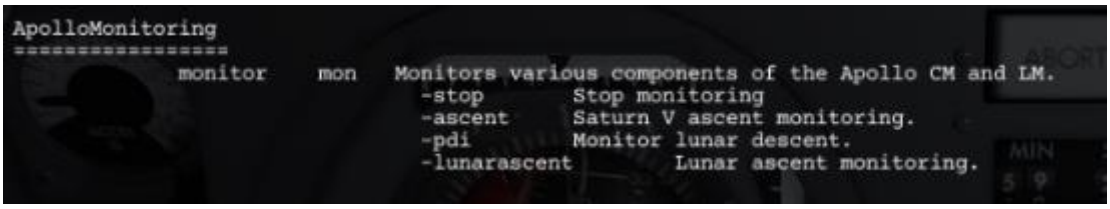


Session State

Command: session
Short: s

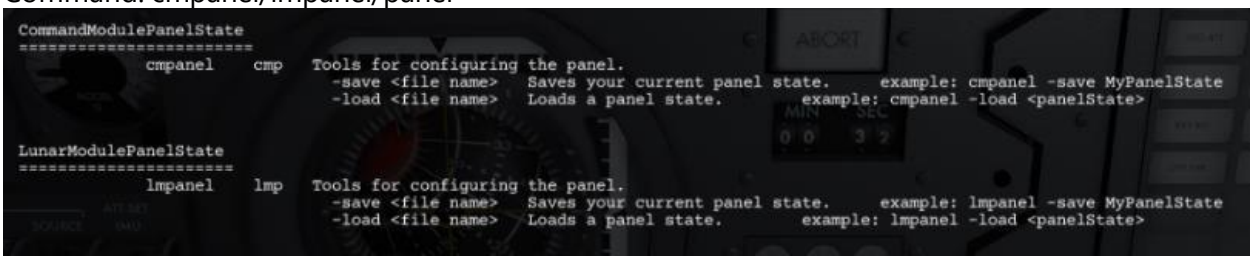


Monitoring
Command: monitor
Short: mon

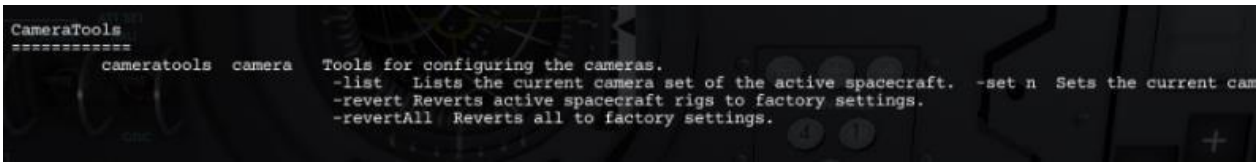


Opens a monitoring window for various phases of the mission.

Panel States
Command: cmpanel/lmpanel/panel



Camera Tools
Command: cameratools
Short: camera



Used to override and modify the default camera positions accessible with the function keys.

9\Alpha Features

IX. ALPHA-FEATURES

1. GENERAL

Alpha-Features are new work-in-progress features that have started its development, but not yet reached a state where it is a main priority (they are future features). The features are only being worked on during code freeze and testing, so development is slow.

However, these are still features that I wish to see in the game, and I have made them available through the alpha-features screen so that you can test them, provide feedback and as a way for me to include you and the games community early in the development. Your voice and suggestions are important.