# Introduction

This API documentation provides a comprehensive guide for the interaction between the Factory in Roblox and the Backend. This document includes the endpoints that can be used to seamlessly integrate and synchronize game state, workspace elements, transporters, and workstations.

# Base URL

The base URL for all endpoints is determined by the server hosting the backend script:

* Localhost: **http://localhost:3000**
* Paul: **https://131.188.76.66:3001**
* Glitch: **https://arrow-troubled-clover.glitch.me**

# Endpoints

1. Game State
   1. **Update Game State**
      1. **Endpoint:** /updateGameState
      2. **Method:** POST
      3. **Description:** Updates the game state with the data provided from the gameState.json file.
   2. **Get Game State**
      1. **Endpoint:** /getGameState
      2. **Method:** GET
      3. **Description:** Retrieves the current game state.
2. Workspace Elements
   1. **Update Workspace Elements**
      1. **Endpoint:** /Workspace
      2. **Method:** POST
      3. **Description:** Updates the workspace elements with the provided data from Roblox.
   2. **Update Construction Zone Elements**
      1. **Endpoint:** /Workspace/ConstructionZone
      2. **Method:** POST
      3. **Description:** Updates the construction zone elements with the provided data from Roblox.
   3. **Update FactoryFloor Elements**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor
      2. **Method:** POST
      3. **Description:** Updates the factory floor elements with the provided data from Roblox.
   4. **Get Workspace Elements**
      1. **Endpoint:** /Workspace
      2. **Method:** GET
      3. **Description:** Retrieves the current workspace elements.
   5. **Get ConstructionZone Elements**
      1. **Endpoint:** /Workspace/ConstructionZone
      2. **Method:** GET
      3. **Description:** Retrieves the current construction zone elements.
   6. **Get FactoryFloor Elements**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor
      2. **Method:** GET
      3. **Description:** Retrieves the current factory floor elements.
3. Transporters
   1. **Get Transporter Names**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor/Transporters
      2. **Method:** GET
      3. **Description:** Retrieves the current transporter names and locations.
   2. **Get Specific Transporter Data**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor/Transporters/:id
      2. **Method:** GET
      3. **Description:** Retrieves the name and location for a specific transporter by ID.
   3. **Update Transporter**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor/Transporters/:id
      2. **Method:** PUT
      3. **Description:** Updates the location for a specific transporter by ID.
      4. **Request Body Example**: {“position”:”x = 110,y = 13,z = 120”}
   4. **Add Transporter**
      1. Yet to be completed.
   5. **Delete Transporter**
      1. Yet to be completed
4. Workstations
   1. **Get Workstation Names**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor/Workstations
      2. **Method:** GET
      3. **Description:** Retrieves the current workstation names and locations.
   2. **Get Specific Workstation Data**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor/Workstations/:id
      2. **Method:** GET
      3. **Description:** Retrieves the name and location for a specific workstation by ID.
   3. **Update Workstations**
      1. **Endpoint:** /Workspace/ConstructionZone/FactoryFloor/Workstations/:id
      2. **Method:** PUT
      3. **Description:** Updates the location for a specific workstation by ID.
      4. **Request Body Example:** {“position”:”x = 100,y = 13,z = 110”}
   4. **Add Workstation**
      1. Yet to be completed.
   5. **Delete Workstation**
      1. Yet to be completed
5. Error Handling

In case of an error, the response will include a relevant error message with an HTTP status code indicating the type of error.