# 🏎️ BucKart Integration Reference Guide

This guide outlines the steps needed to integrate and configure BucKart based on the previous Pong implementation.

## 1. Rename Namespaces

- Replace all instances of 'Pong' with 'BucKart'.  
- Adjust capitalization where necessary (e.g., 'pong' → 'buckart' if lowercase is required).

## 2. BucKart Microservice Setup

### Properties

Open launchSettings.json:

- Update all namespaces to 'BucKart'

- Under 'applicationUrl', set the port to the new empty port → 1950 for BucKart.

### wwwroot

- In /images, create and replace the thumbnail with BucKart’s image.  
- Replace the existing /js folder with BucKart’s version (these files control game functionality).  
- In /lib, ensure 'bucky' and all related GameInfo files are included (see GameInfo.cs section).

### Controllers

- For each game controller (e.g., \*Game\*Controller):

- Change all namespaces from the previous game to 'BucKart'.

- Rename the file accordingly.

- Update all GameInfo fields to describe BucKart.

- Increment the 'Id' value → use X + 1, where X = number of existing games.

### appsettings.json

- Update all namespace references from the previous game to 'BucKart'.  
- Replace thumbnail and related paths as needed.  
- Update:  
 - Port number: 1950 for BucKart.  
 - Docker port: 8085 for BucKart.

### DockerFile

- Add a new line for BucKart following the same template used for Pong, e.g.:  
 BucKart.xxx

### GameInfo.cs

- Update the namespace to 'BucKart'.

## 3. Add BucKart to API

### Controllers

- In GatewayController.cs, add 'BucKart' to the gameKeys list.

### appsettings.json

- In all .json files:

- Add 'BucKart' under 'MicroserviceURLs'

- Add 'BucKart' under 'PublicURLs' for both:

- .containers

- .containersLocal

## 4. BucStop Web App Integration

### wwwroot

- Add BucKart.jpg to the /images folder.

### appsettings.json

- In all .json files:

- Add 'BucKart' under 'MicroserviceURLs'.

- Add 'BucKart' under 'PublicURLs' for .containers and .containersLocal.

**Controllers**

- In GameFeatureManager.cs

- In game feature manager controller in the dictionary add a new line for the game and to make it appear make the bool true.