Jon Fernandez de Gorostiza

08/12/2024

Prueba 1 **2**
Función de recompensa 2
Información de entrenamiento sobre los modelos 4
Modelo 1 4
Modelo 2 5
Modelo 3 6
Modelo 4 7
Evaluaciones 8
Modelo 1 8
Modelo 2 10
Modelo 3 11
Modelo 4 13
Conclusiones 14
Carrera AWS Virtual Circuit **15**
Prueba 2 **16**
Información de entrenamiento sobre los modelos 17
Modelo 5 17
Modelo 6 18
Modelo 7 19
Evaluaciones 20
Modelo 5 20
Modelo 6 22
Modelo 7 23
Conclusiones 25
Evaluaciones generales: 25
Conclusiones sobre la eficacia de las modificaciones: 25
Prueba 3 **28**
Función de recompensa 28
Información de entrenamiento sobre los modelos 30
Modelo 8 30
Modelo 9 31
Modelo 10 32
Modelo 11 33
Evaluaciones 34
Modelo 8 35
Modelo 9 36
Modelo 10 38
Modelo 11 39
Conclusiones 41
Comparación entre los dos mejores modelos **41**
Actualización Carrera AWS Virtual Circuit **42**

Prueba 1

He desarrollado cuatro modelos utilizando la misma función de recompensa, pero diferenciados por el algoritmo de entrenamiento empleado y el entorno en el que fueron entrenados. Dos de los modelos se entrenaron en la pista ForeverRaceway CounterClockwise, uno utilizando el algoritmo SAC y el otro con PPO. Los otros dos modelos se entrenaron en la pista de Barcelona. Todos los modelos fueron entrenados durante una hora, sin realizar ajustes en los hiperparametros, y operan a velocidades entre 1 y 3 m/s.

Función de recompensa

```
from math import atan2, degrees, sqrt

def reward_function(params):
    # Parámetros proporcionados por DeepRacer
    track_width = params['track_width']
    distance_from_center = params['distance_from_center']
    is_offtrack = params['is_offtrack']
    speed = params['speed']
    steering = abs(params['steering_angle'])
    progress = params['progress']
    waypoints = params['waypoints']
    closest_waypoints = params['closest_waypoints']
    car_position = (params['x'], params['y'])
    heading = params['heading']

    # Recompensa inicial
    reward = 1.0

    # 1. Penalización por salirse de la pista
    if is_offtrack:
        return 1e-3  # recompensa mínima si el coche está fuera de la pista

    # 2. Prioridad: Mantenerse en el centro de la pista
    marker_1 = 0.1 * track_width  # Muy cerca del centro
    marker_2 = 0.25 * track_width  # Cerca del centro
    if distance_from_center <= marker_1:
        reward += 10.0  # Recompensa máxima por estar muy cerca del centro
    elif distance_from_center <= marker_2:
        reward += 5.0  # Recompensa moderada por estar cerca del centro
    else:
        reward += 1.0  # Recompensa baja si está más lejos del centro
```

```
    # Penalización adicional si está muy cerca del borde
    if distance_from_center > 0.5 * track_width:
        reward *= 0.5  # Reducción de la recompensa si está cerca del borde

    # 3. Evitar el zigzag (recompensa por mantener bajo el ángulo de giro)
    if steering < 5:
        reward += 5.0  # Muy estable
    elif steering < 15:
        reward += 2.0  # Ligeramente estable
    else:
        reward += 1.0  # Algo inestable (zigzag)

    # 4. Maximizar velocidad con ajuste en curvas y rectas
    SPEED_THRESHOLD = 2.0  # Velocidad objetivo para rectas
    # Calcular la dirección de la pista usando los waypoints más cercanos
    next_point = waypoints[closest_waypoints[1]]
    prev_point = waypoints[closest_waypoints[0]]

    # Calcular la dirección de la pista entre los puntos más cercanos
    track_direction = degrees(atan2(next_point[1] - prev_point[1], next_point[0] -
prev_point[0]))
    # Calcular la diferencia de ángulo entre la dirección de la pista y la orientación del coche
    direction_diff = abs(track_direction - heading)
    if direction_diff > 180:
        direction_diff = 360 - direction_diff  # Asegura que el ángulo está entre 0 y 180

    in_curve = direction_diff > 10  # Detecta si está en una curva
    if in_curve:
        if speed < SPEED_THRESHOLD:
            reward += 3.0  # Recompensa por reducir velocidad en curva
        else:
            reward *= 0.5  # Penalización por ir demasiado rápido en curva
    else:
        if speed >= SPEED_THRESHOLD:
            reward += 5.0  # Recompensa por mantener alta velocidad en recta

    # 5. Mantenerse en la línea ideal
    ideal_line_dist = sqrt((next_point[0] - car_position[0])**2 + (next_point[1] -
car_position[1])**2)
    if ideal_line_dist < 0.1 * track_width:
        reward += 5.0  # Muy cerca de la línea ideal
    elif ideal_line_dist < 0.2 * track_width:
        reward += 2.0  # A una distancia aceptable de la línea ideal
    else:
        reward *= 0.8  # Penalización por estar lejos de la línea ideal

    # 6. Progreso en la pista
    reward += progress * 0.1  # Recompensa basada en progreso total
```

```
# 7. Recompensa por completar la vuelta (cuando el progreso es alto)
if progress > 95:
    reward += 10.0  # Recompensa adicional por casi completar la vuelta

return float(reward)
```

Información de entrenamiento sobre los modelos

Modelo 1

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

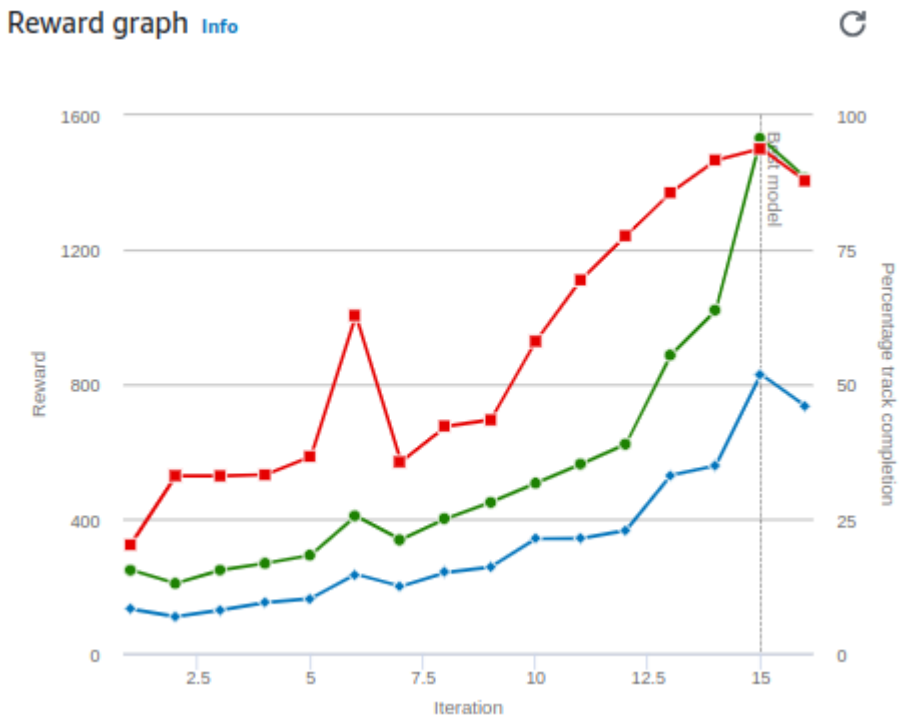
Action space type
Continuous

Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10



Modelo 2

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

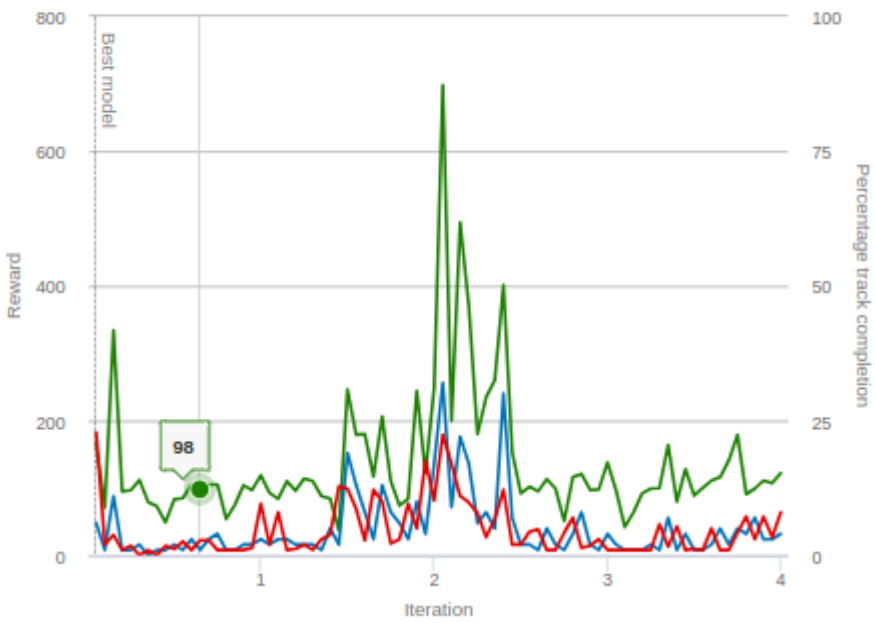
Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
SAC

Hyperparameter	Value
Gradient descent batch size	64
Discount factor	0.99
Loss type	Mean squared error
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	1
SAC alpha	0.2

Reward graph Info



Modelo 3

Training configuration

Race type
Time trial

Environment simulation
Circuit de Barcelona-Catalunya - Clockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

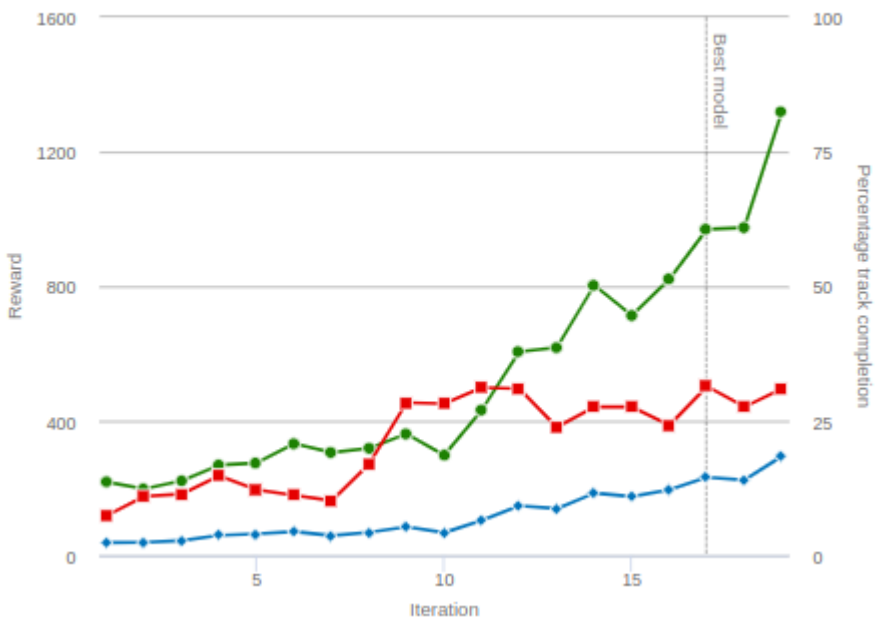
Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10

Reward graph Info



Modelo 4

Training configuration

Race type
Time trial

Environment simulation
Circuit de Barcelona-Catalunya - Clockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

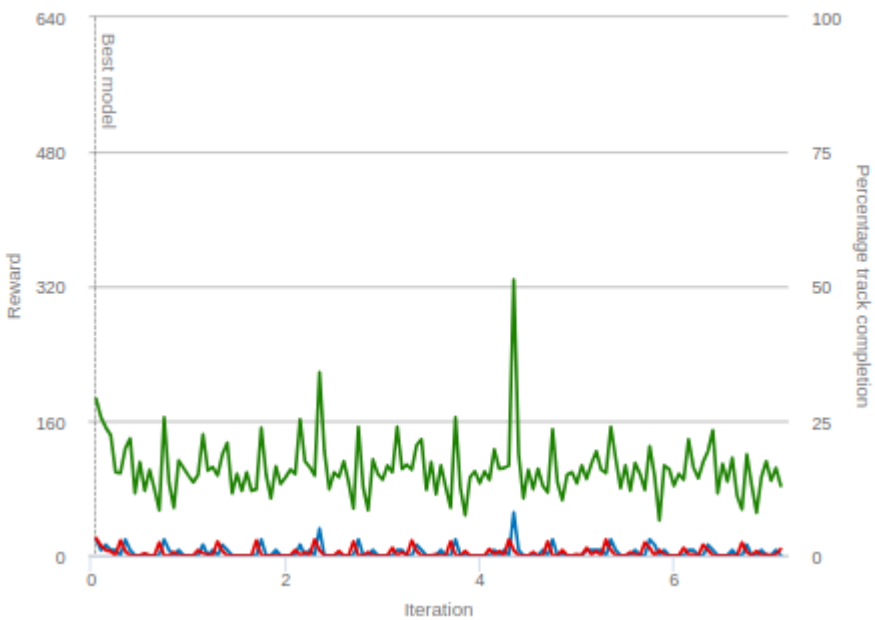
Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
SAC

Hyperparameter	Value
Gradient descent batch size	64
Discount factor	0.99
Loss type	Mean squared error
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	1
SAC alpha	0.2

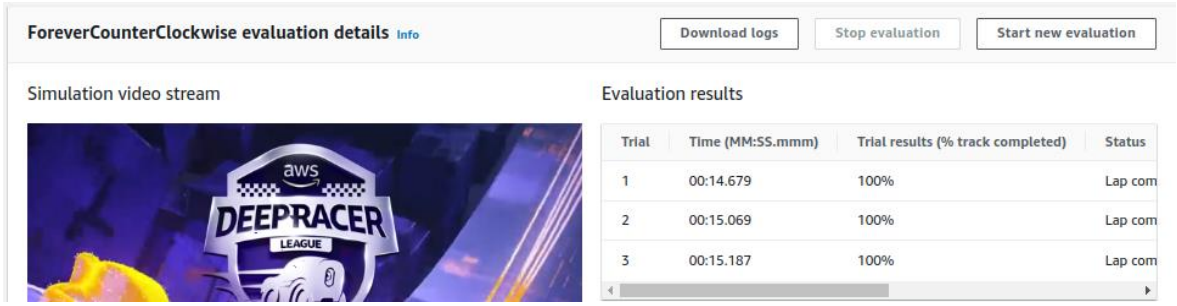
Reward graph Info



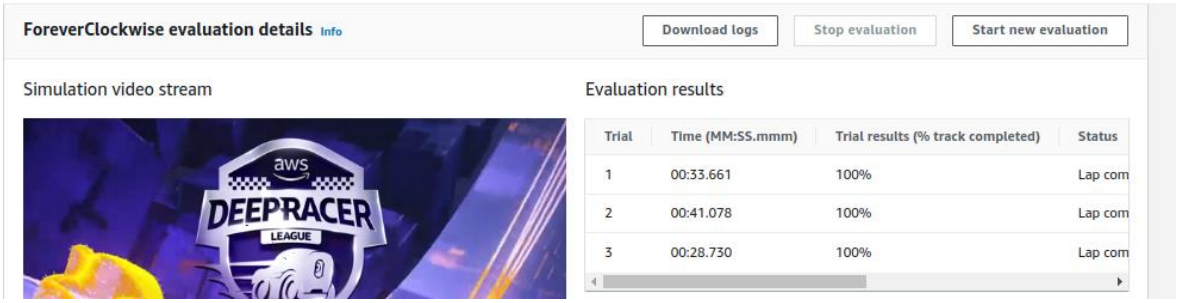
Evaluaciones

Modelo 1

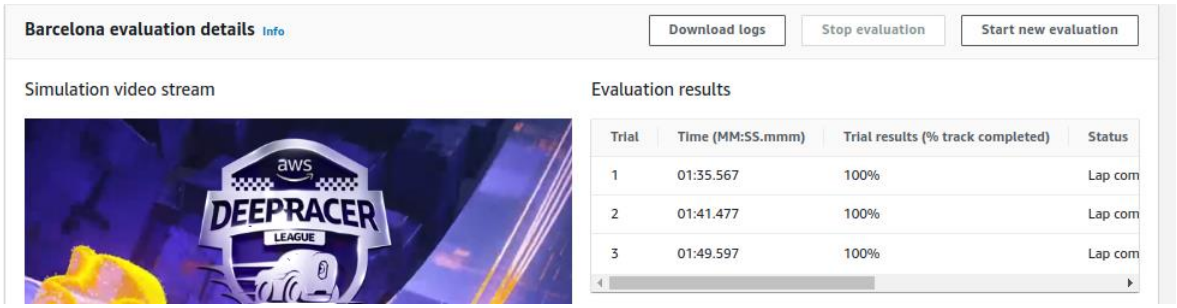
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	0	00:14.679
			0	00:15.069
			0	00:15.187



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	Si	7	00:33.661
			10	00:41.078
			5	00:28.730

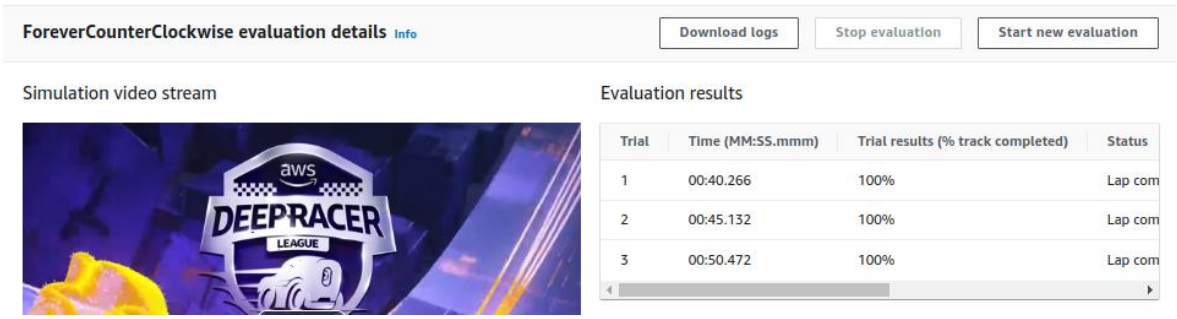


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	23	01:35.567
			25	01:41.477
			29	01:49.597

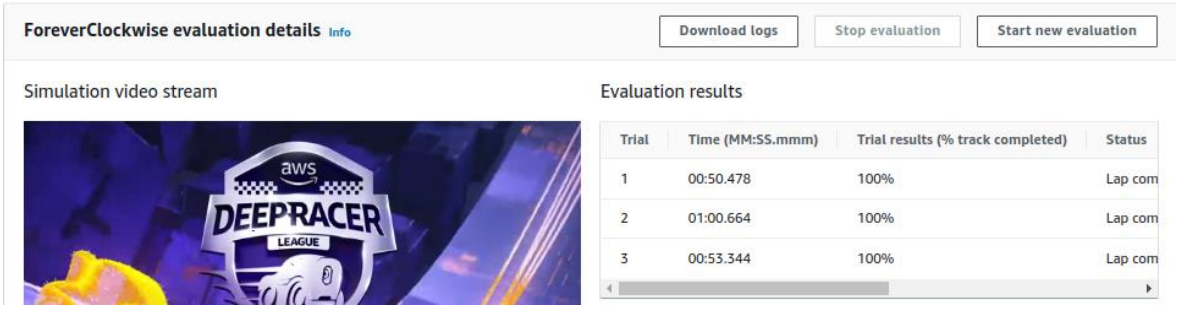


Modelo 2

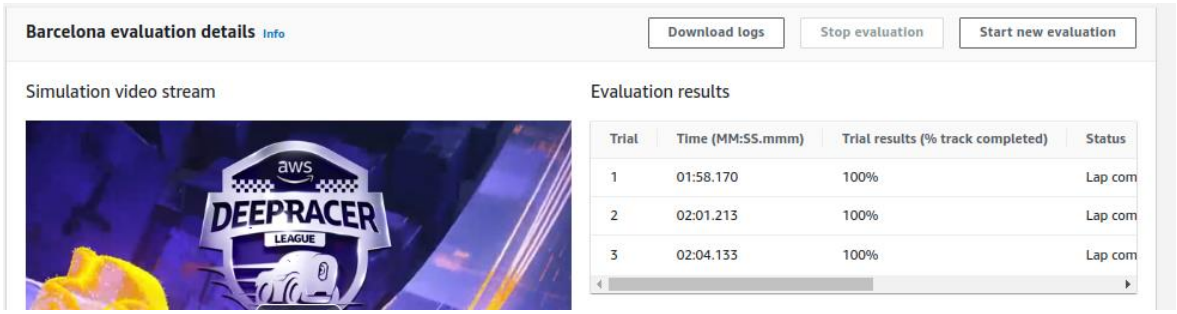
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	11	00:40.266
			13	00:45.132
			15	00:50.472



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	Si	15	00:50.478
			19	01:00.664
			16	00:53.344

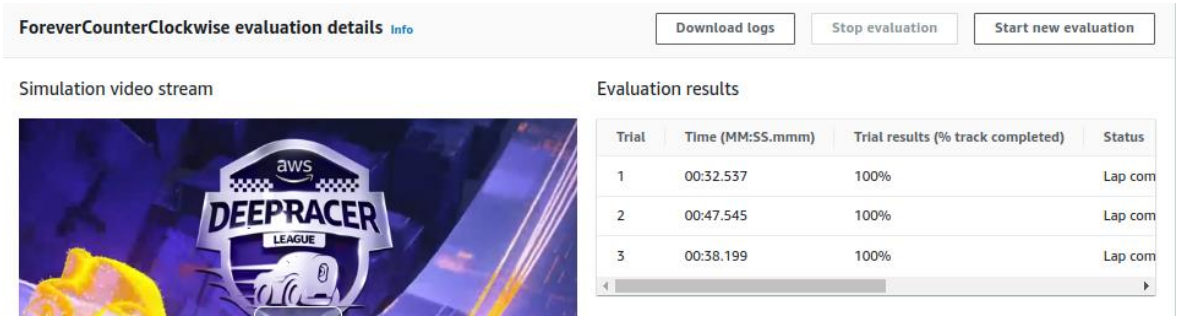


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	35	01:58.170
			36	02:01.213
			37	02:04.133

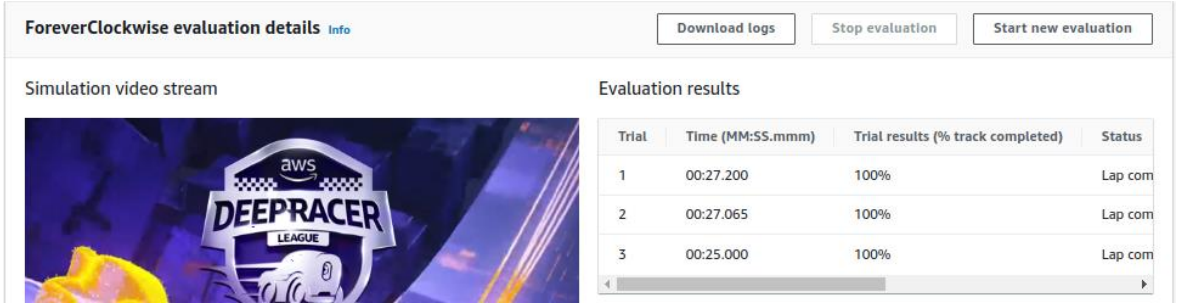


Modelo 3

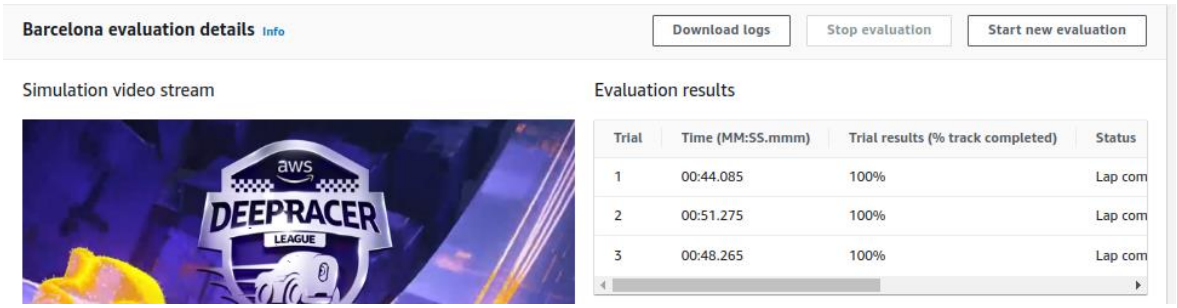
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	8	00:32.537
			14	00:47.545
			10	00:38.19



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	6	00:27.200
			6	00:27.065
			5	00:25.000

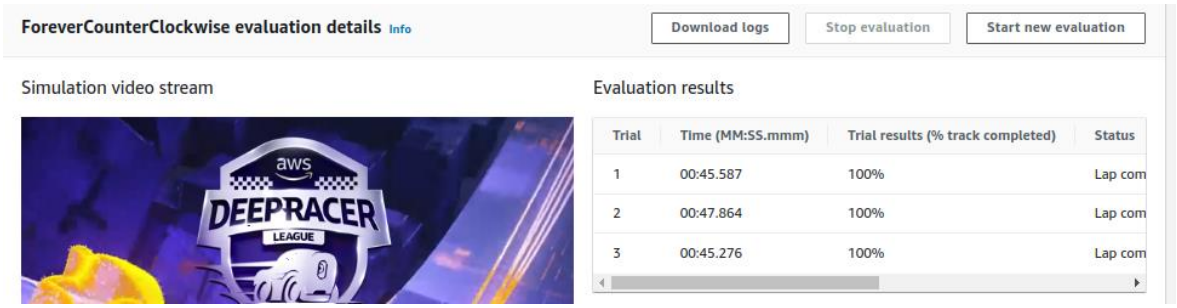


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	4	00:27.200
			7	00:27.065
			5	00:25.000

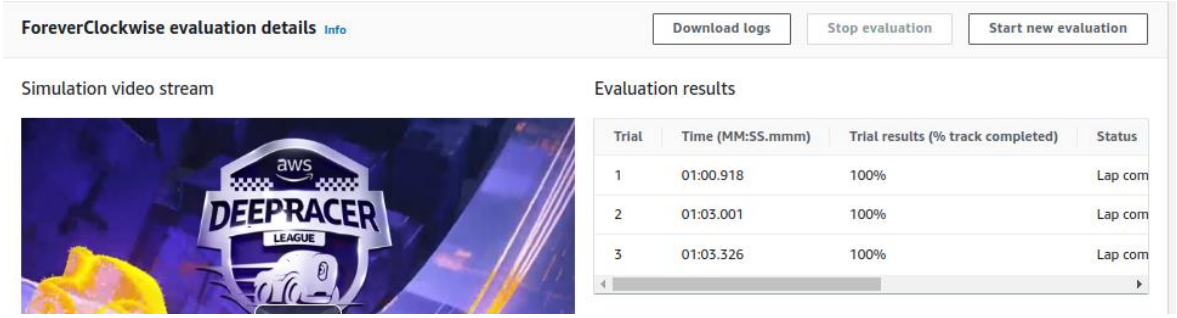


Modelo 4

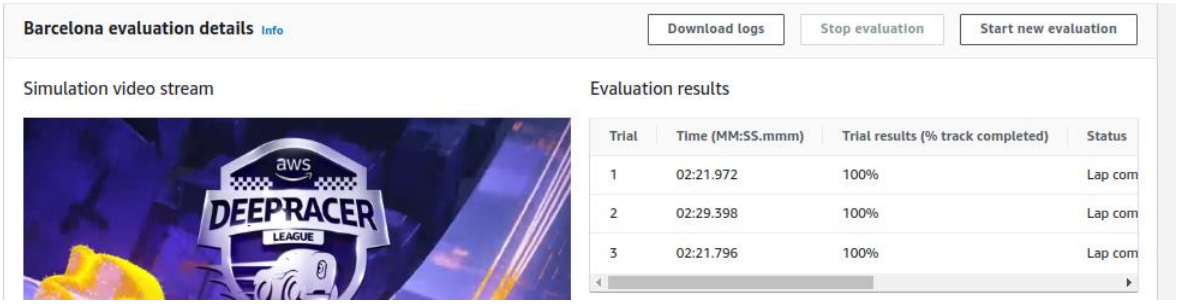
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	13	00:45.587
			14	00:47.864
			13	00:45.276



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	19	01:00.918
			20	01:03.001
			20	01:03.326

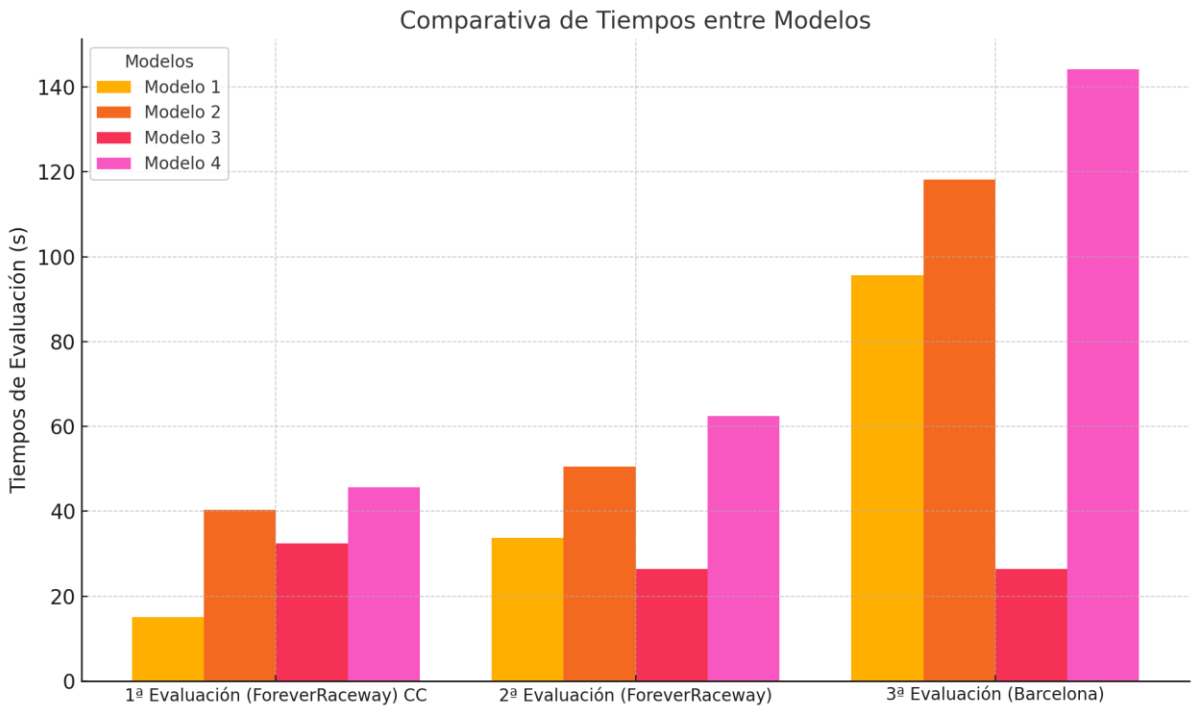


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	44	2:21.972
			47	02:29.398
			44	02:21.796

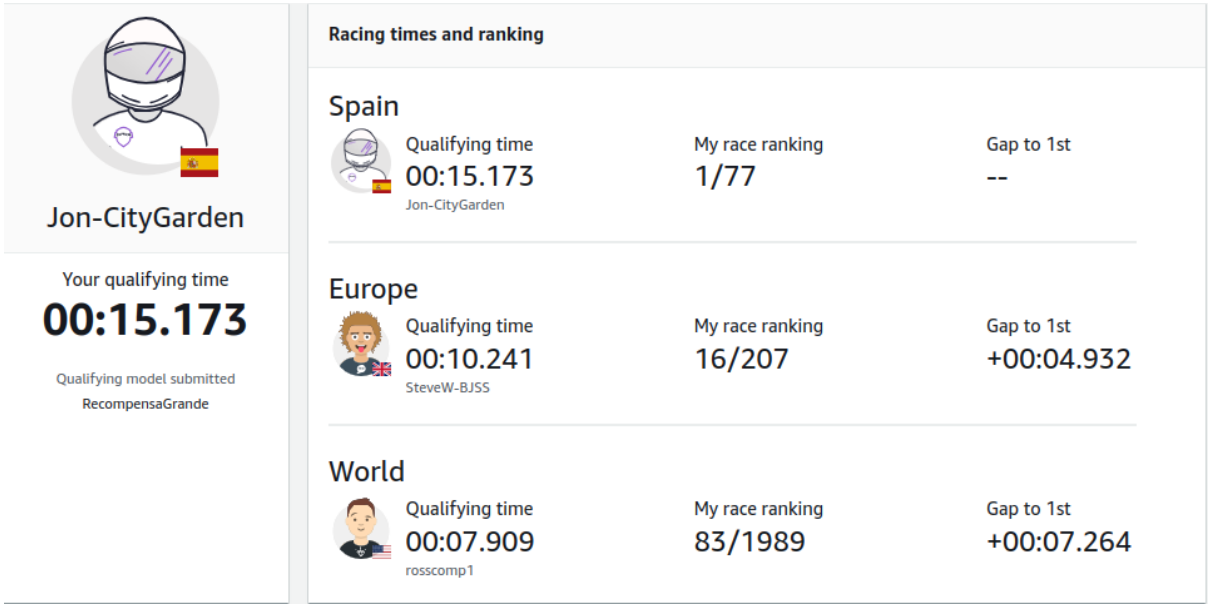


Conclusiones

1. **Mejor Modelo:**
   - El Modelo 3 se destaca como el más estable y adaptable, logrando buenos tiempos y un bajo número de salidas de pista en todas las evaluaciones.
2. **Peor Modelo:**
   - El Modelo 4 es el menos eficiente, con los tiempos más altos y un número excesivo de salidas de pista en ambas pistas.
3. **Impacto del Algoritmo:**
   - Los modelos entrenados con **SAC** (como el Modelo 3) parecen ser más consistentes y efectivos que los entrenados con **PPO**, especialmente en adaptarse a pistas nuevas.



Carrera AWS Virtual Circuit



RecompensaGrande es lo que en este documento se refiere al Modelo 1. Podemos ver los distintos resultados que nos puede dar un mismo modelo en diferentes pruebas en el mismo circuito, en este caso la carrera es en la pista ForeverRaceway counterclocwise.

RecompensaGrande	00:18.532	00:15.004	00:18.532	Submitted	11/21/2024, 5:02 PM	View
RecompensaGrande	00:15.173	00:14.594	00:15.173	Submitted	11/21/2024, 4:49 PM	View
RecompensaGrande	00:15.685	00:15.195	00:15.685	Submitted	11/21/2024, 3:11 PM	View
RecompensaGrande	00:17.353	00:15.598	00:17.353	Submitted	11/20/2024, 6:40 PM	View
RecompensaGrande	00:21.138	00:16.529	00:21.138	Submitted	11/20/2024, 6:15 PM	View
RecompensaGrande	00:16.597	00:14.535	00:16.597	Submitted	11/20/2024, 5:36 PM	View
RecompensaGrande	00:18.978	00:15.461	00:18.978	Submitted	11/20/2024, 5:24 PM	View
RecompensaGrande	00:18.979	00:15.466	00:18.979	Submitted	11/20/2024, 5:00 PM	View

*En este momento ya no voy primero, Gabriel me ha adelantado en la general de España.

Prueba 2

Para comprobar la eficacia de la primera función de recompensa, he clonado el modelo 1 para ver si añadiendo velocidad esta saca mejores resultados para mejorar el tiempo en la carrera. (Modelo 5)
Como segunda prueba, hemos clonado el modelo 1 y hemos añadido velocidad además de cambiar algunos hiperparámetros. (Modelo 6)
Como tercera prueba, he cogido el segundo modelo lo he clonado y lo he entrenado 1 hora más para ver si mejora, ya que PPO ha dado unos resultados muy malos para ver si ha sido por falta de entrenamiento. (Modelo 7)
Todas las pruebas han sido entrenadas 1 hora

Información de entrenamiento sobre los modelos

Modelo 5

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

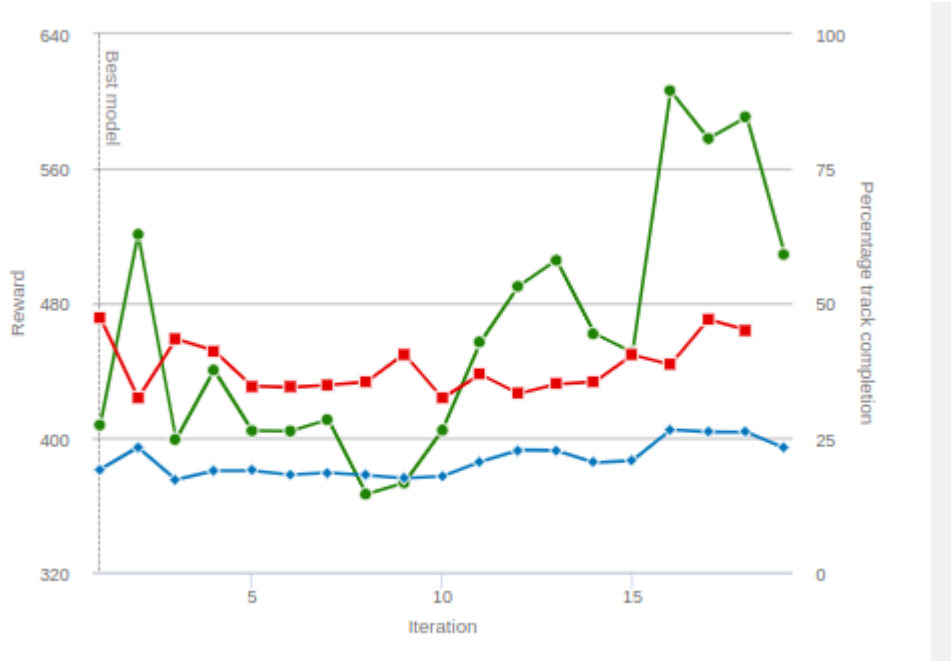
Action space type
Continuous

Action space
Speed: [1.6 : 3.5] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10



Modelo 6

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

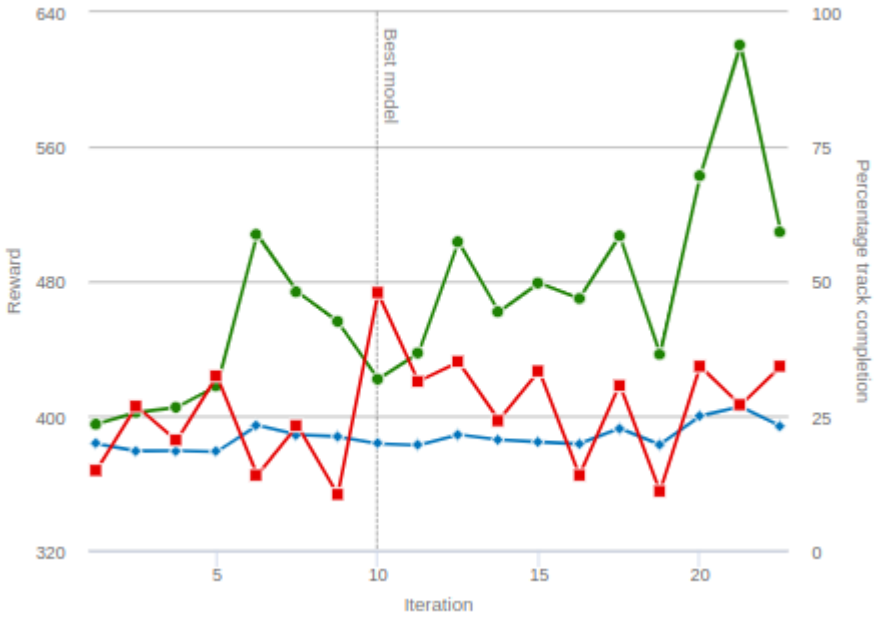
Action space type
Continuous

Action space
Speed: [1.7 : 3.5] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	128
Entropy	0.1
Discount factor	0.99
Loss type	Mean squared error
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	25
Number of epochs	10



Modelo 7

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

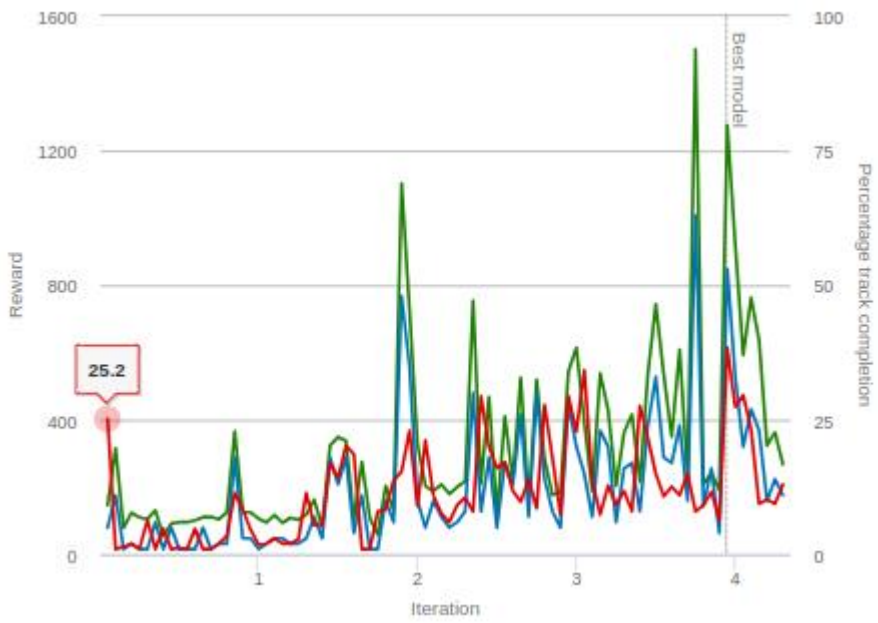
Action space type
Continuous

Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
SAC

Hyperparameter	Value
Gradient descent batch size	64
Discount factor	0.99
Loss type	Mean squared error
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	1
SAC alpha	0.2



Evaluaciones

Modelo 5

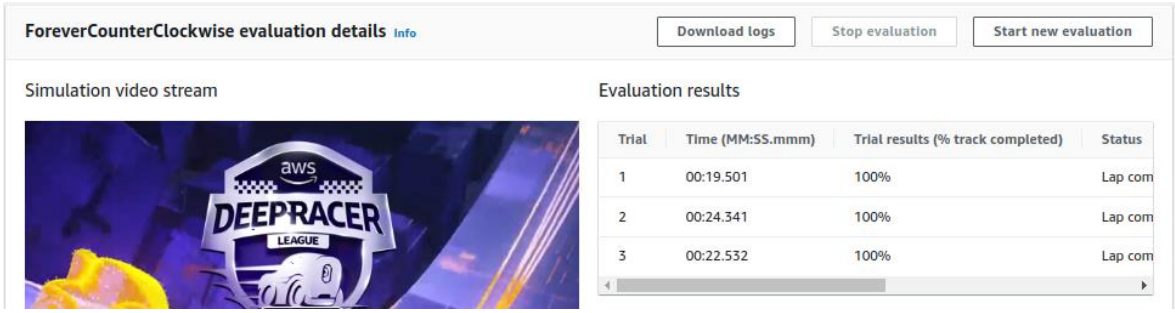
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	4	00:21.882
			4	00:22.140
			4	00:22.342
Ya que el laboratorio no tiene dinero no puedo volver a sacarla				

2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	Si	7	00:30.383
			8	00:33.005
			9	00:34.804
Ya que el laboratorio no tiene dinero no puedo volver a sacarla				

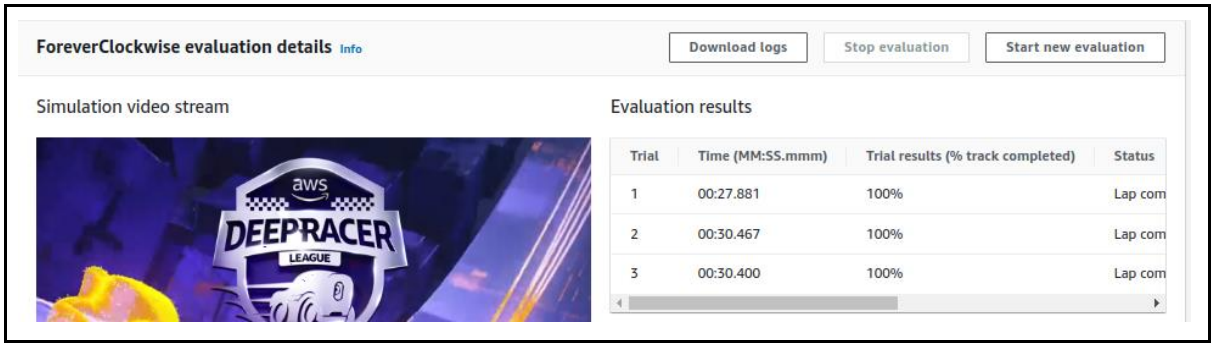
3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	33	01:54.122
			29	01:41.998
			30	01:45.340
Ya que el laboratorio no tiene dinero no puedo volver a sacarla				

Modelo 6

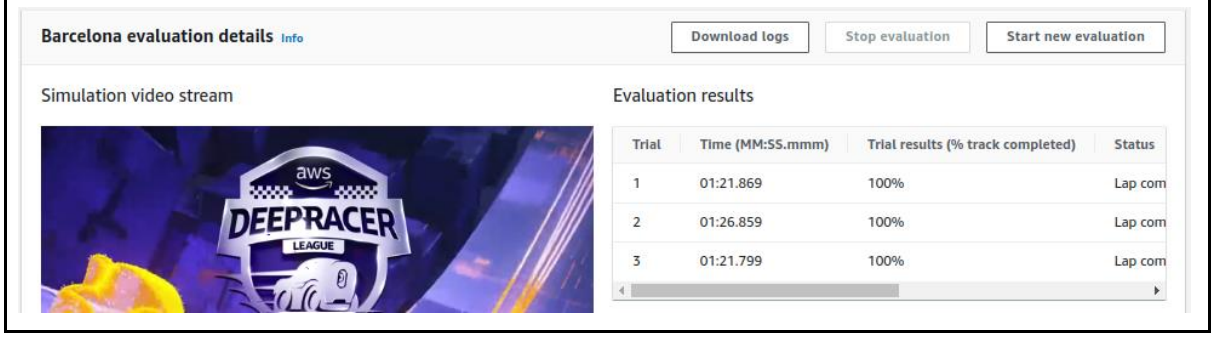
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	3	00:19.501
			5	00:24.341
			4	00:22.532



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	6	00:27.881
			7	00:30.467
			7	00:30.400

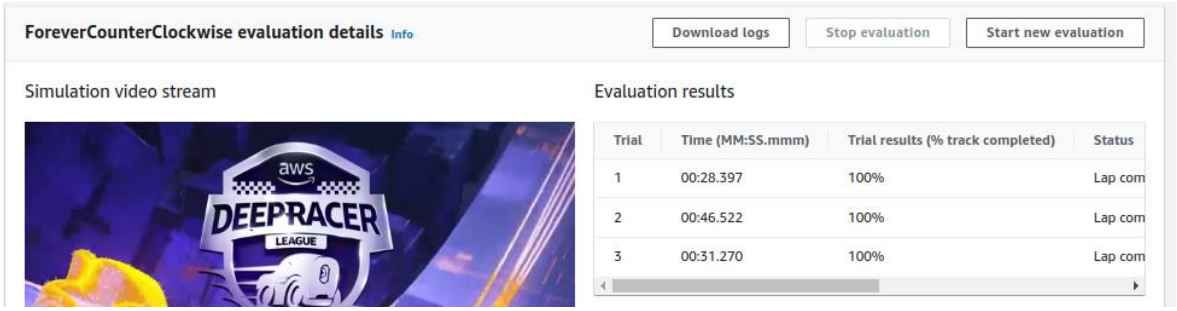


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	20	01:21.869
			22	01:26.859
			20	01:21.799

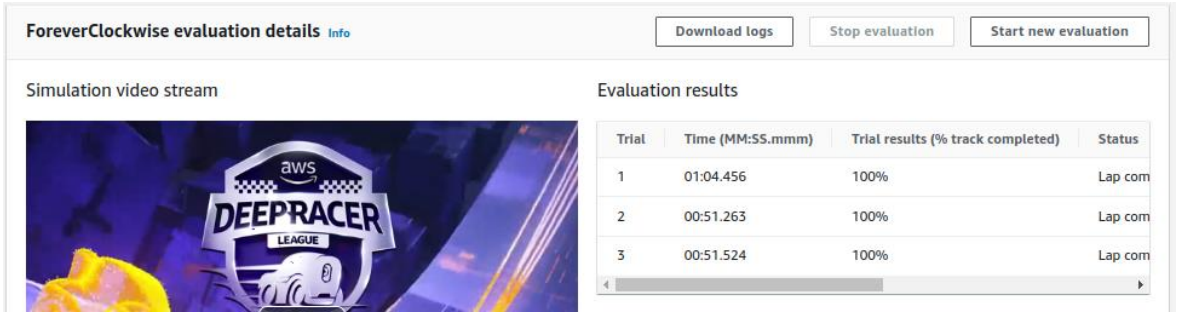


Modelo 7

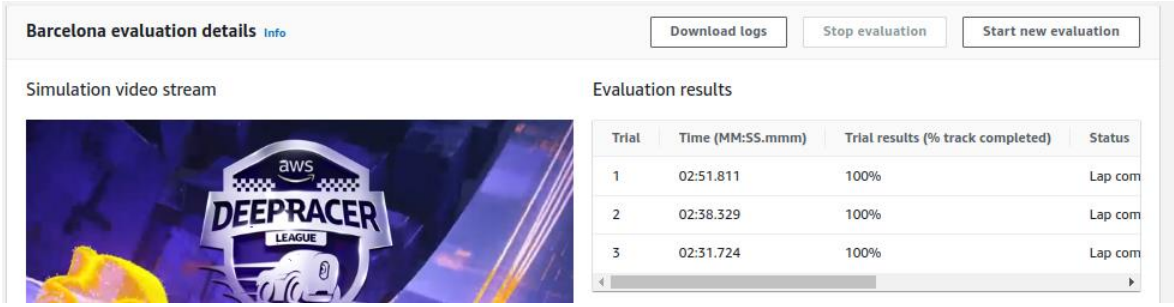
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	6	00:28.397
			13	00:46.522
			7	00:31.270



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	20	01:04.456
			15	00:51.263
			15	00:51.524



3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	57	02:51.811
			51	02:38.329
			48	02:31.724



Conclusiones

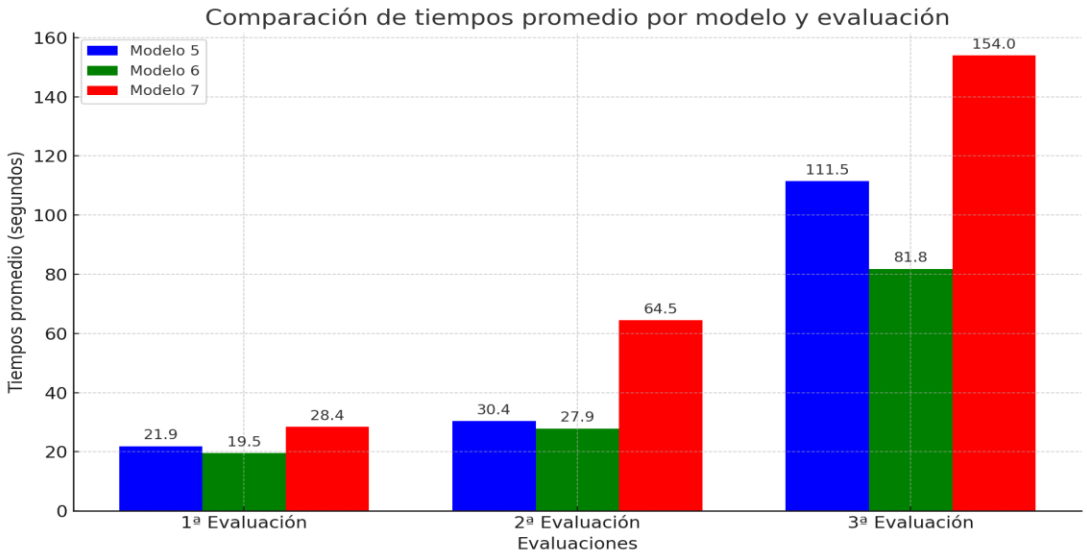
Evaluaciones generales

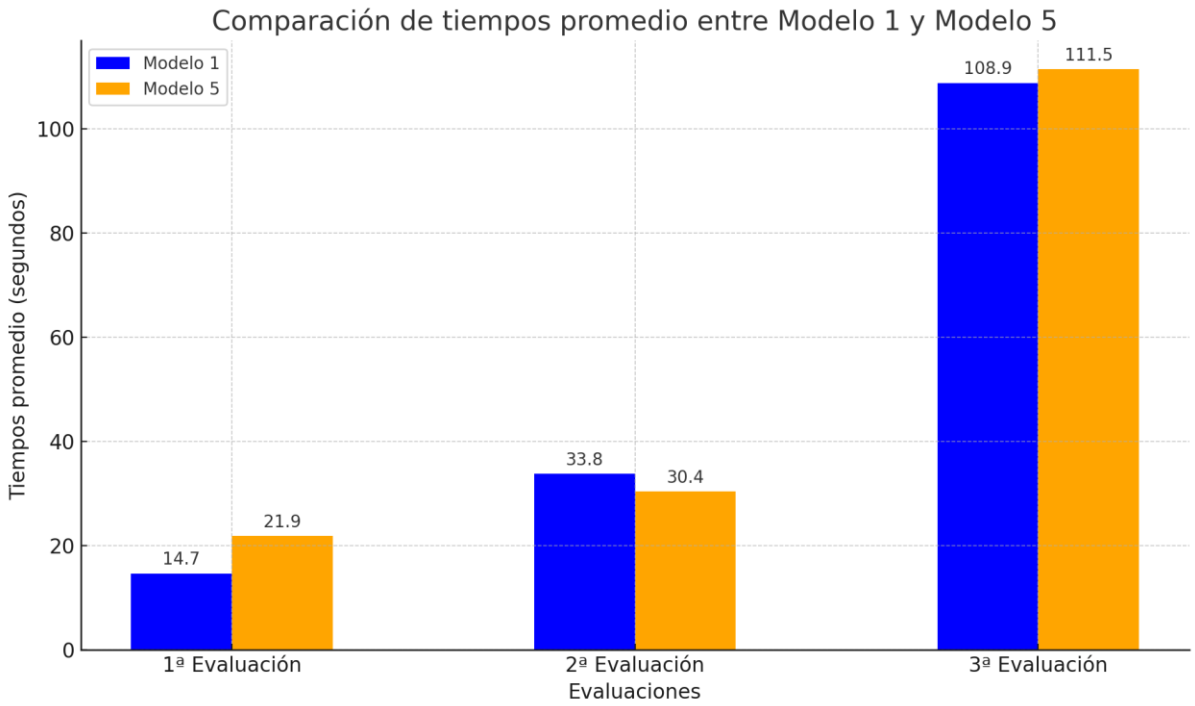
1. **Primera evaluación (ForeverRaceway, sin salidas de pista):**
   - Modelo 1 logra tiempos significativamente más bajos (14.679s promedio) en comparación con Modelo 5, Modelo 6, y Modelo 7, lo que sugiere que agregar velocidad no fue suficiente para superar al modelo base.
   - Modelo 5 mejora ligeramente respecto a los tiempos de Modelo 6 y Modelo 7 en esta pista, mostrando que aumentar la velocidad puede tener un impacto, aunque limitado.
2. **Segunda evaluación (ForeverRaceway, con salidas de pista):**
   - Modelo 1 obtiene mejores resultados en promedio (33.823s) que el resto, con menos salidas de pista.
   - Modelo 6 mejora significativamente respecto a Modelo 5, mostrando que ajustar hiperparámetros junto con la velocidad puede ser más efectivo para manejar la pista con salidas de pista.
   - Modelo 7, aunque entrenado una hora más, no supera a Modelo 6 en esta evaluación. Sus tiempos y salidas de pista aumentan, sugiriendo que el entrenamiento adicional puede haber exacerbado problemas existentes o saturado el modelo.

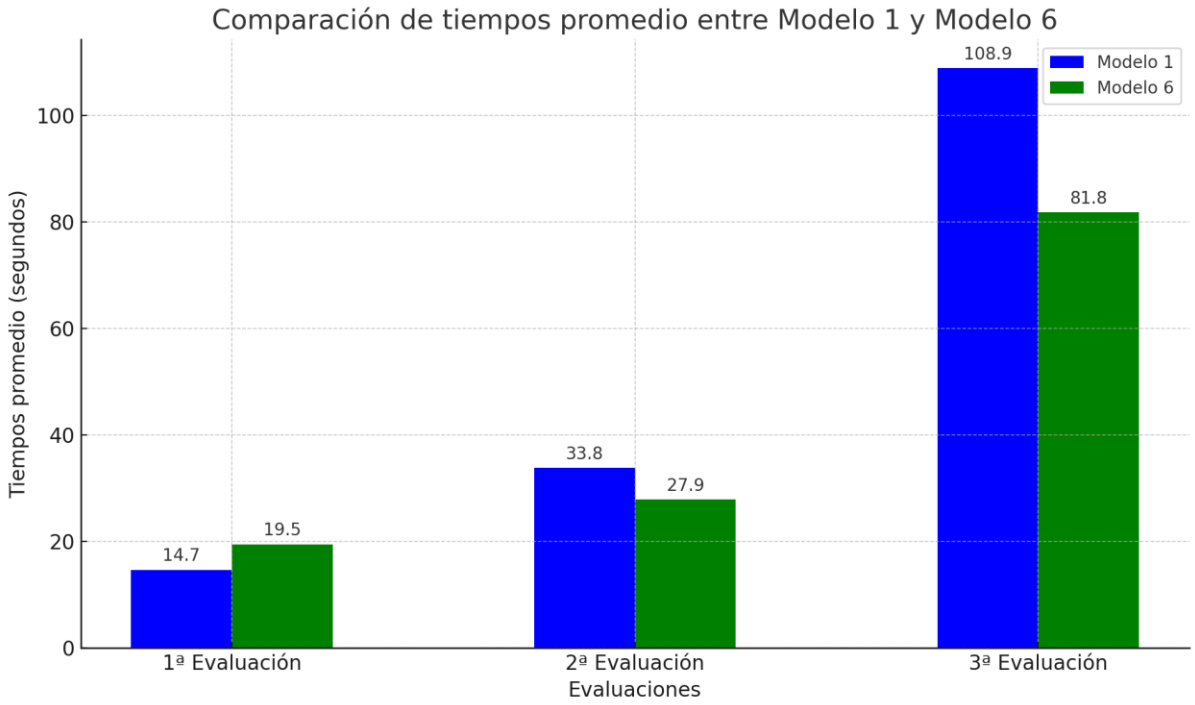
3. **Tercera evaluación (Barcelona, sin salidas de pista):**
    - En esta pista técnica y más larga, Modelo 6 se destaca con el mejor promedio de tiempos (81.842s), seguido de Modelo 5 (111.487s).
    - Modelo 7 tiene tiempos significativamente más altos (153.955s) y más salidas de pista, lo que sugiere que el entrenamiento adicional no mejoró su capacidad para manejar la complejidad del circuito.

Conclusiones sobre la eficacia de las modificaciones:

1. **Impacto de la velocidad (Modelo 5):**
    - Incrementar la velocidad mejoró marginalmente los tiempos respecto al modelo base, pero el resultado no fue suficiente para superar consistentemente al Modelo 1. Esto indica que añadir velocidad sin otros ajustes no garantiza mejoras sustanciales, especialmente en pistas complejas como Barcelona.
2. **Impacto de los hiperparámetros (Modelo 6):**
    - Los ajustes en los hiperparámetros fueron efectivos en las evaluaciones más difíciles, como ForeverRaceway con salidas de pista y Barcelona. **Modelo 6** demuestra ser el más equilibrado y robusto en distintas condiciones, con una clara ventaja en Barcelona.
3. **Impacto del tiempo adicional de entrenamiento (Modelo 7):**
    - Aunque parece razonable esperar que un modelo entrenado más tiempo obtenga mejores resultados, **Modelo 7** tuvo peores tiempos y más salidas de pista. Esto podría deberse a:
        - **Sobreentrenamiento**: El modelo podría haberse ajustado en exceso a ciertos patrones de datos de entrenamiento, perdiendo generalización.
        - **Eficiencia del algoritmo PPO**: Es posible que el algoritmo no haya sido lo suficientemente eficaz incluso con más tiempo de entrenamiento.




Comparación de tiempos promedio entre Modelo 1 y Modelo 5
Modelo 1
Modelo 5
Tiempos promedio (segundos)
0
20
40
60
80
100
14.7
21.9
33.8
30.4
108.9
111.5
1ª Evaluación
2ª Evaluación
3ª Evaluación
Evaluaciones


Comparación de tiempos promedio entre Modelo 1 y Modelo 6
Modelo 1
Modelo 6
Tiempos promedio (segundos)
0
20
40
60
80
100
14.7
19.5
33.8
27.9
108.9
81.8
1ª Evaluación
2ª Evaluación
3ª Evaluación
Evaluaciones

Prueba 3

He creado cuatro modelos distintos, basados en una versión “mejorada” de la función de recompensa anterior, y los he entrenado en la pista ForeverRaceway. Las diferencias entre los modelos son las siguientes: dos modelos tienen un rango de velocidad de 1-3 m/s, mientras que los otros dos tienen un rango de 1,5-4 m/s. Además, de los cuatro modelos, dos fueron entrenados durante 1 hora y los otros dos durante 2 horas.

Función de recompensa

```
from math import atan2, degrees, sqrt

def reward_function(params):
    # Parámetros proporcionados por DeepRacer
    track_width = params['track_width']
    distance_from_center = params['distance_from_center']
    is_offtrack = params['is_offtrack']
    speed = params['speed']
    steering = abs(params['steering_angle'])
    progress = params['progress']
    waypoints = params['waypoints']
    closest_waypoints = params['closest_waypoints']
    car_position = (params['x'], params['y'])
    heading = params['heading']

    # Recompensa inicial
    reward = 1.0

    # 1. Penalización por salirse de la pista
    if is_offtrack:
        return 1e-3  # recompensa mínima si el coche está fuera de la pista

    # 2. Detectar si está en una curva o recta
    # Calcular la dirección de la pista entre los puntos más cercanos
    next_point = waypoints[closest_waypoints[1]]
    prev_point = waypoints[closest_waypoints[0]]
    track_direction = degrees(atan2(next_point[1] - prev_point[1], next_point[0] -
prev_point[0]))

    # Calcular la diferencia entre la dirección de la pista y la orientación del coche
    direction_diff = abs(track_direction - heading)
    if direction_diff > 180:
```

```
        direction_diff = 360 - direction_diff  # Asegura que el ángulo está entre 0 y 180

    # Determinar si es una curva o una recta
    is_curve = direction_diff > 10  # Umbral para considerar una curva

    # Detectar curvas amplias o curvas en "S"
    FUTURE_POINTS = 3  # Número de waypoints a considerar para detectar curvas
    future_directions = []
    for i in range(1, FUTURE_POINTS + 1):
        if closest_waypoints[1] + i < len(waypoints):
            future_next = waypoints[closest_waypoints[1] + i]
            future_prev = waypoints[closest_waypoints[1] + i - 1]
            direction = degrees(atan2(future_next[1] - future_prev[1], future_next[0] -
future_prev[0]))
            future_directions.append(direction)

    # Calcular diferencias consecutivas de direcciones para identificar curvas
    direction_changes = [abs(future_directions[i + 1] - future_directions[i]) for i in
range(len(future_directions) - 1)]
    total_direction_change = sum(direction_changes)

    # Identificar tipo de curva
    is_s_curve = total_direction_change > 30  # Alta variación: curva "S"
    is_wide_curve = 10 < total_direction_change <= 30  # Baja a moderada variación: curva
amplia

    # 3. Recompensa en rectas
    if not is_curve:
        # Priorizar mantenerse en el centro en las rectas
        marker_1 = 0.1 * track_width  # Muy cerca del centro
        marker_2 = 0.25 * track_width  # Cerca del centro
        if distance_from_center <= marker_1:
            reward += 10.0  # Recompensa máxima por estar muy cerca del centro
        elif distance_from_center <= marker_2:
            reward += 5.0  # Recompensa moderada por estar cerca del centro
        else:
            reward += 1.0  # Recompensa baja si está más lejos del centro

        # Penalización adicional si está muy cerca del borde
        if distance_from_center > 0.5 * track_width:
            reward *= 0.5  # Reducción de la recompensa si está cerca del borde

    # 4. Recompensa en curvas amplias
    elif is_wide_curve:
        # Incentivar estar ligeramente hacia el interior de la curva
        if distance_from_center < 0.25 * track_width:
            reward += 5.0  # Recompensa por recortar suavemente
        elif distance_from_center < 0.5 * track_width:
```

```
        reward += 2.0  # Recompensa moderada
    else:
        reward *= 0.8  # Penalización por estar demasiado lejos del interior

# 5. Recompensa en curvas en "S"
elif is_s_curve:
    # Incentivar mantener una línea recta cercana al centro
    if distance_from_center <= 0.1 * track_width:
        reward += 10.0  # Recompensa por estar cerca del centro
    elif distance_from_center <= 0.25 * track_width:
        reward += 5.0  # Recompensa moderada
    else:
        reward *= 0.5  # Penalización por desviarse mucho

# 6. Velocidad (aplica tanto a curvas como a rectas)
SPEED_THRESHOLD = 2.0  # Velocidad objetivo para rectas
if is_curve:
    if speed < SPEED_THRESHOLD:
        reward += 3.0  # Recompensa por reducir velocidad en curva
    else:
        reward *= 0.5  # Penalización por ir demasiado rápido en curva
else:
    if speed >= SPEED_THRESHOLD:
        reward += 5.0  # Recompensa por mantener alta velocidad en recta

# 7. Progreso en la pista
reward += progress * 0.1  # Recompensa basada en progreso total

# 8. Recompensa por completar la vuelta (cuando el progreso es alto)
if progress > 95:
    reward += 10.0  # Recompensa adicional por casi completar la vuelta

return float(reward)
```

Información de entrenamiento sobre los modelos

Modelo 8

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

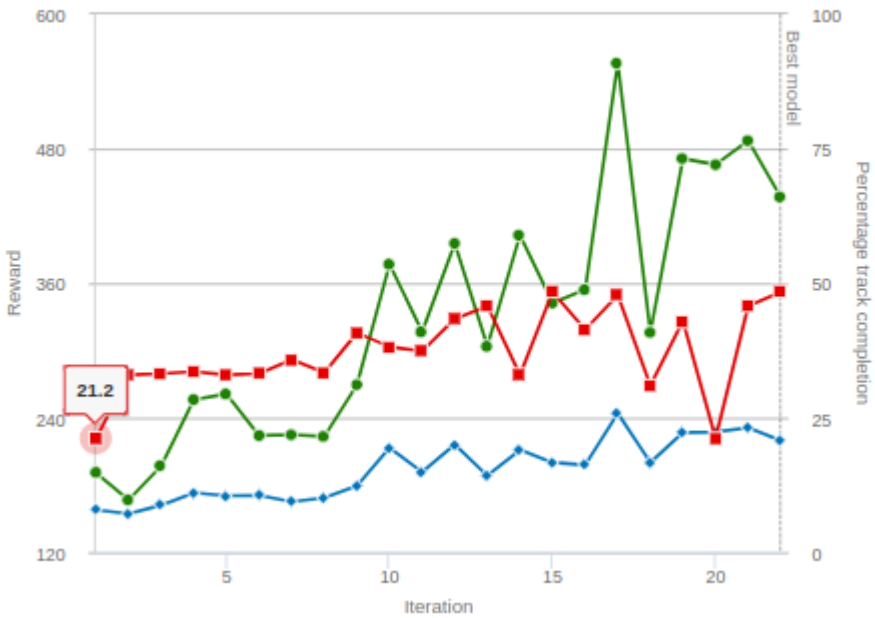
Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10

Reward graph Info



Modelo 9

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

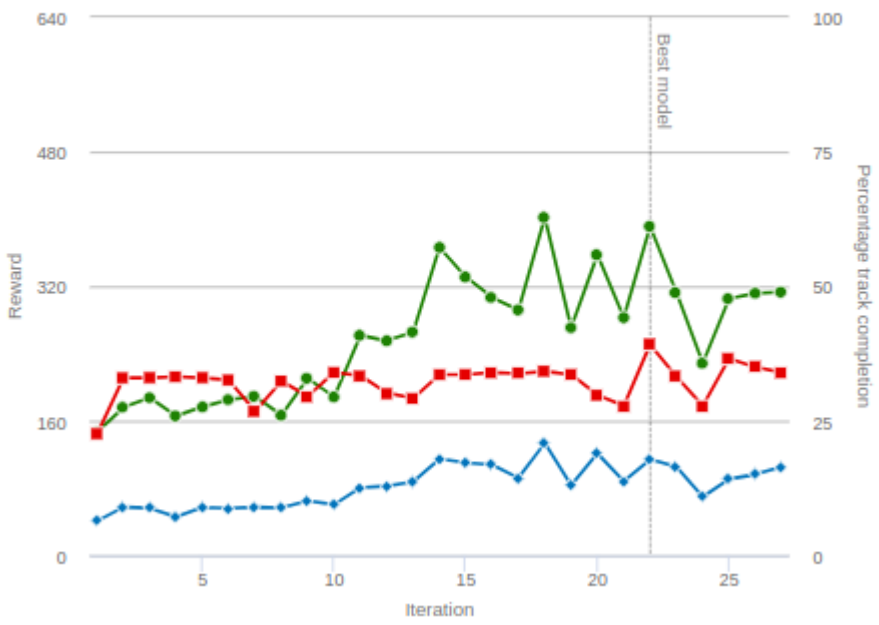
Action space
Speed: [1.5 : 4] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10

Reward graph Info



Modelo 10

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

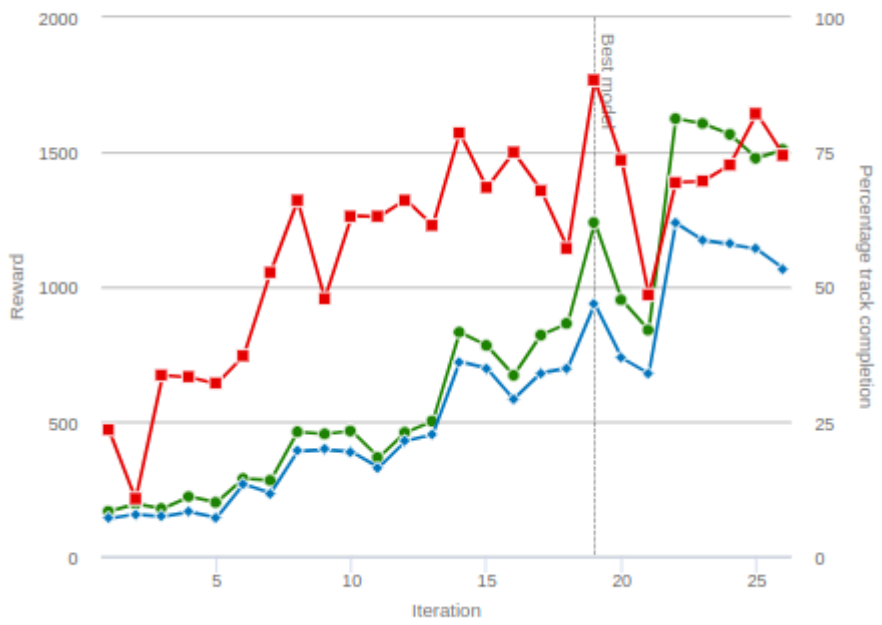
Action space
Speed: [1 : 3] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10

Reward graph Info



Stop condition

Maximum time

02:00:00 / 02:00:00

Modelo 11

Training configuration

Race type
Time trial

Environment simulation
Forever Raceway - Counterclockwise

Reward function
Show

Sensor(s)
Camera

Action space type
Continuous

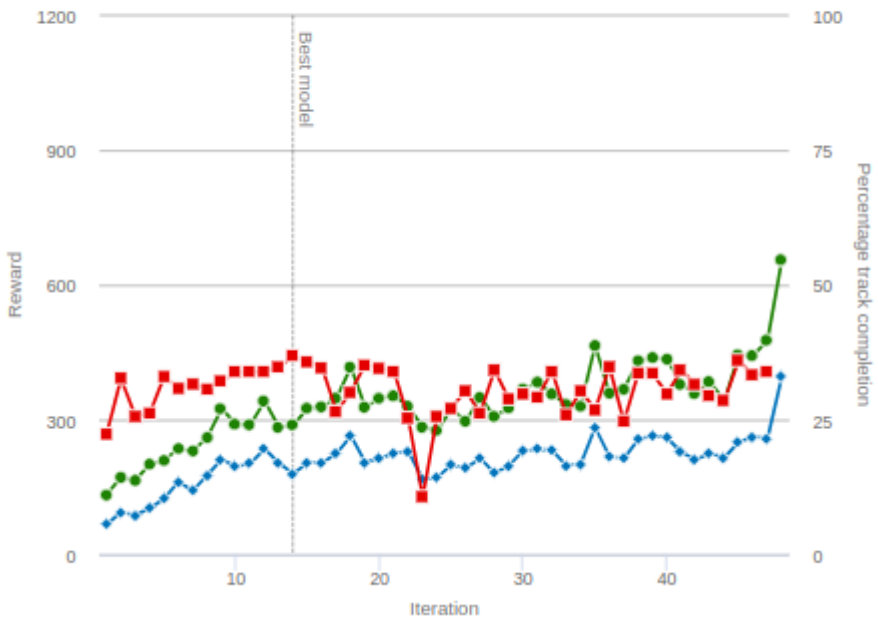
Action space
Speed: [1.5 : 4] m/s
Steering angle: [-30 : 30] °

Framework
Tensorflow

Reinforcement learning algorithm
PPO

Hyperparameter	Value
Gradient descent batch size	64
Entropy	0.01
Discount factor	0.99
Loss type	Huber
Learning rate	0.0003
Number of experience episodes between each policy-updating iteration	20
Number of epochs	10

Reward graph Info



Stop condition

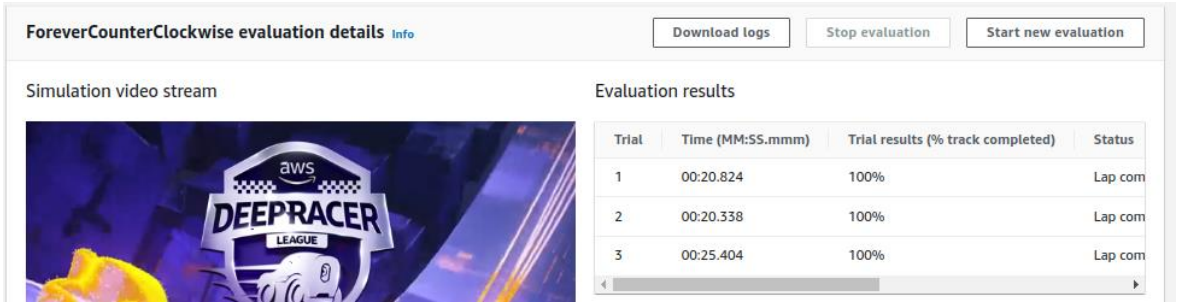
Maximum time

02:00:00 / 02:00:00

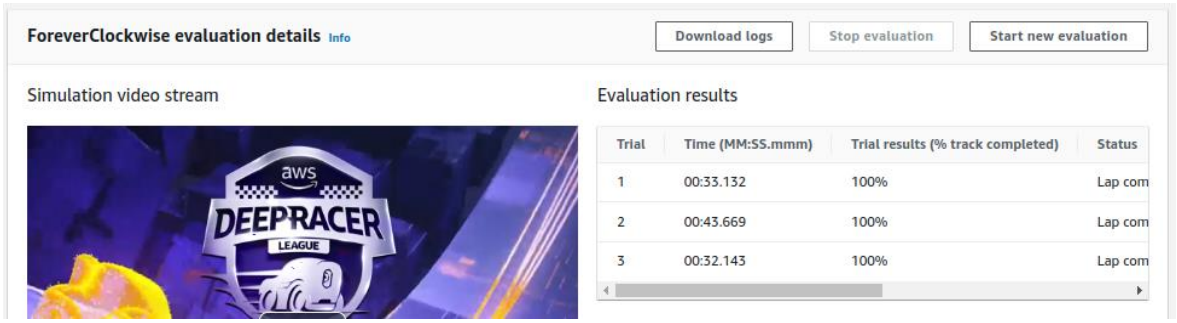
Evaluaciones

Modelo 8

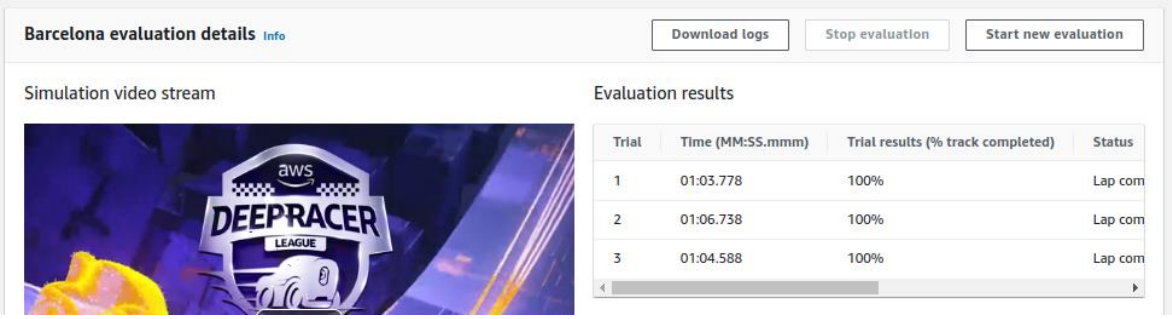
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	3	00:20.824
			3	00:20.338
			5	00:25.404



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	7	00:33.132
			11	00:43.669
			7	00:32.143

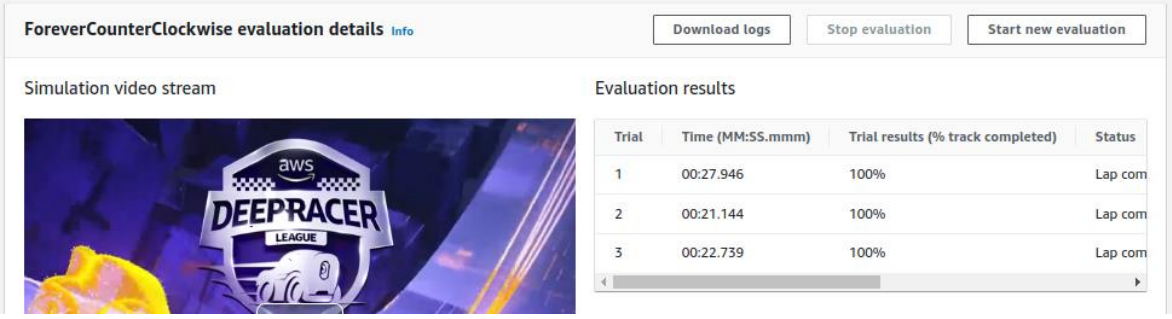


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	10	01:03.778
			12	01:06.738
			11	01:04.588

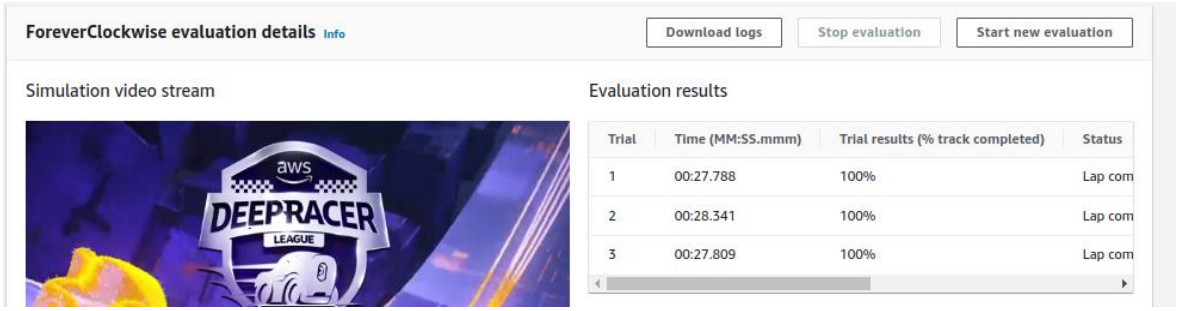


Modelo 9

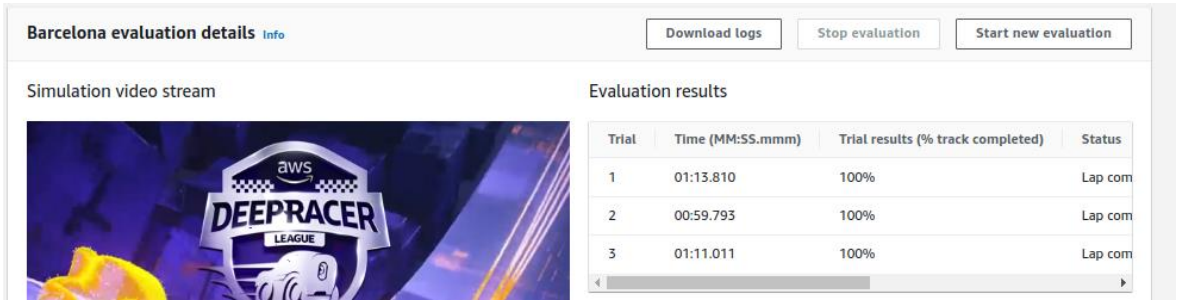
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	6	00:27.946
			3	00:21.144
			4	00:22.739



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	6	00:27.788
			6	00:28.341
			6	00:27.809

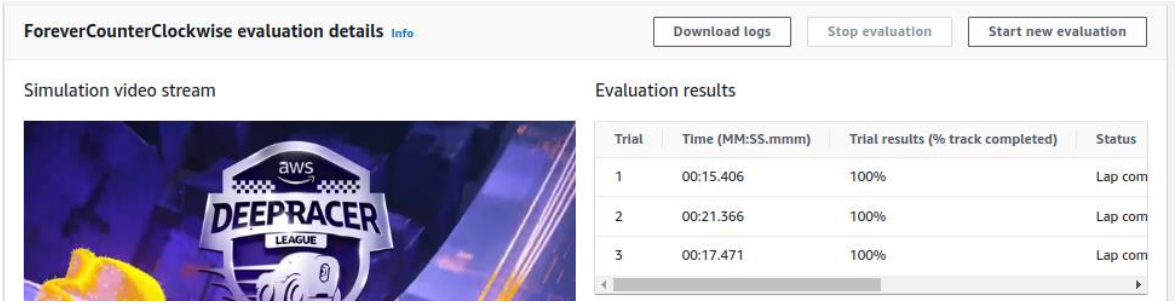


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	15	01:13.810
			10	00:59.793
			14	01:11.011

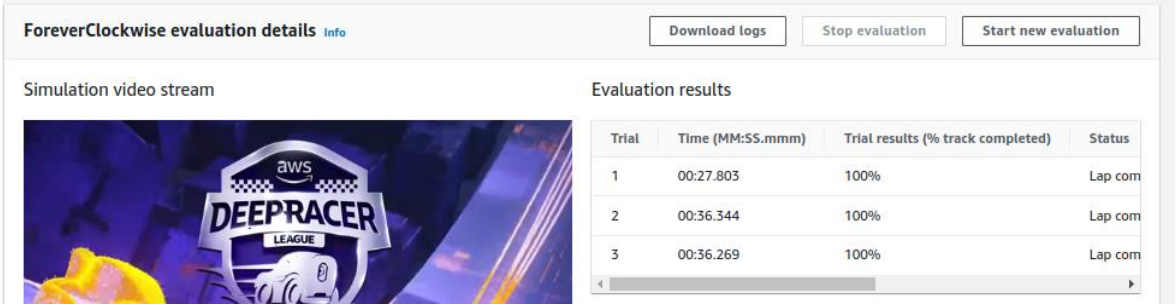


Modelo 10

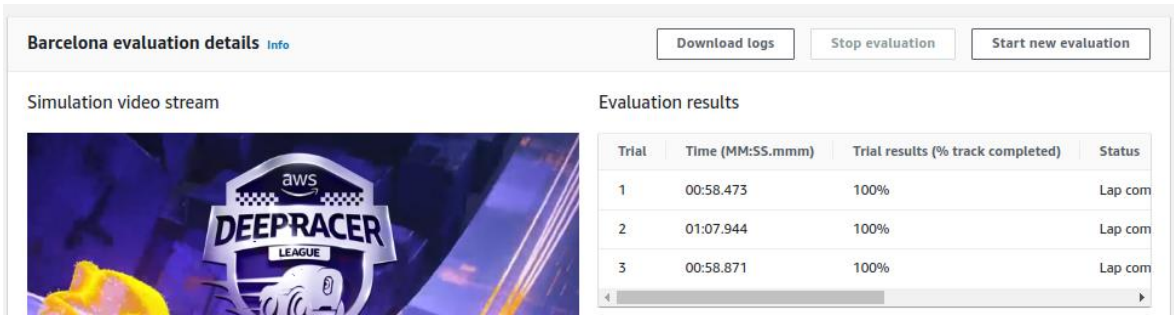
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	0	00:15.406
			2	00:21.366
			1	00:17.471



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	5	00:27.803
			8	00:36.344
			8	00:36.269

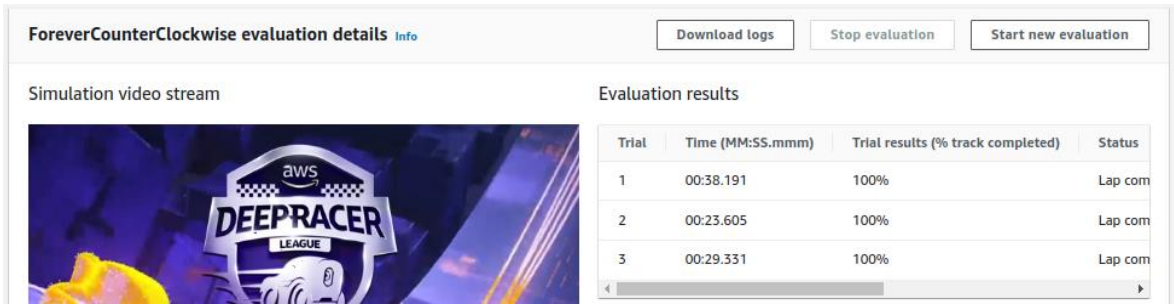


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	9	00:58.473
			12	01:07.944
			9	00:58.871

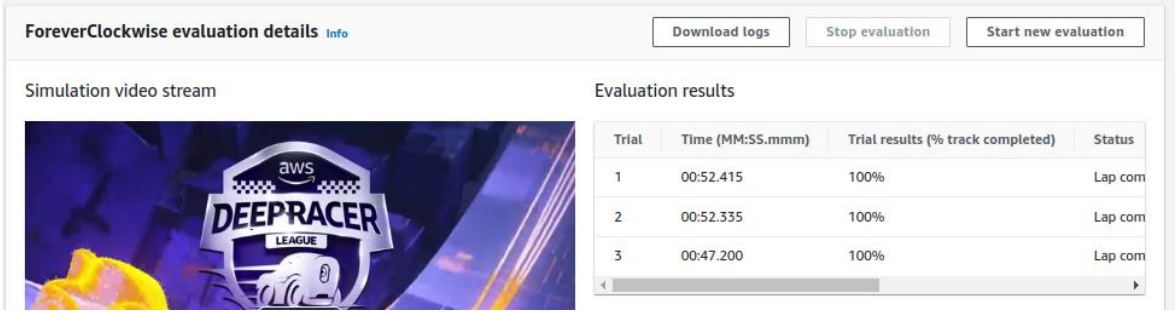


Modelo 11

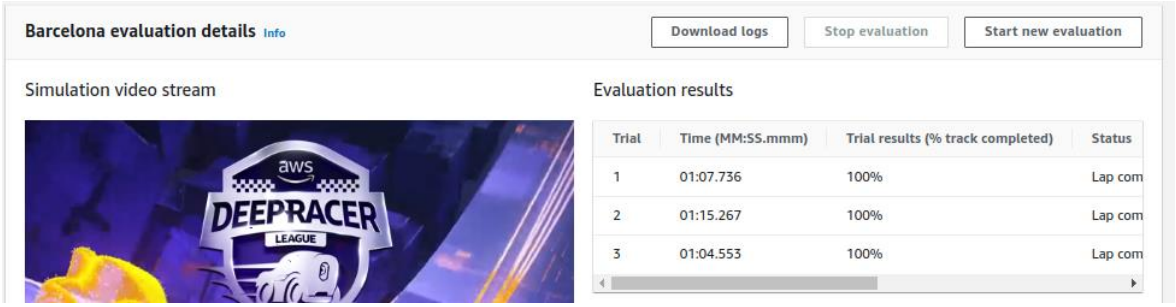
1º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	No	10	00:38.191
			5	00:23.605
			7	00:29.331



2º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	ForeverRaceway	SI	16	00:52.415
			16	00:52.335
			14	00:47.200

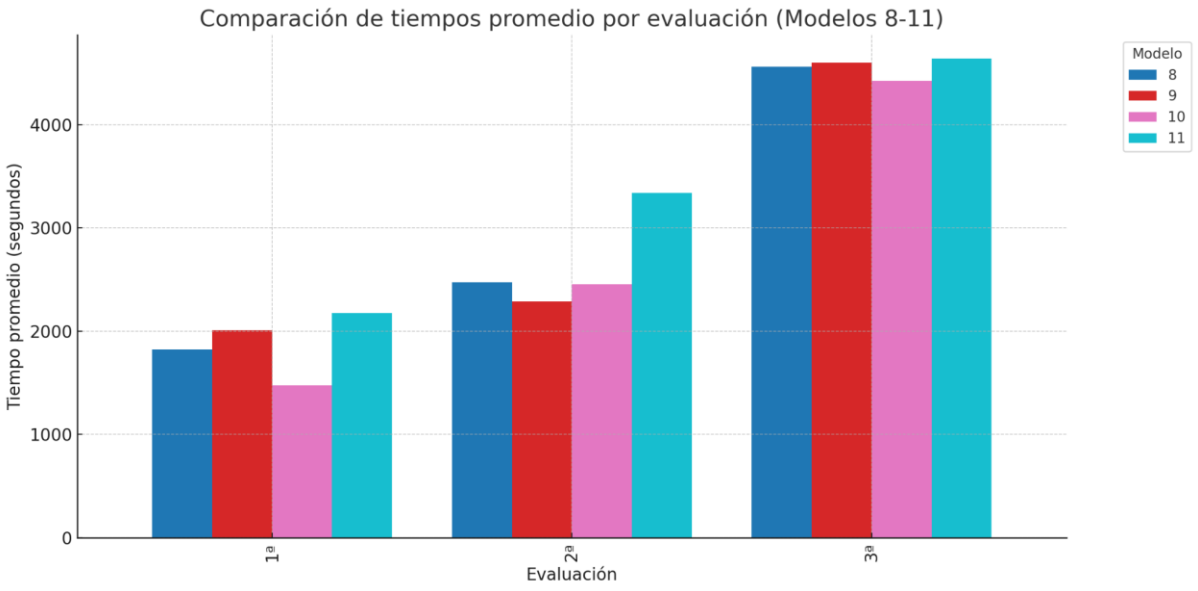


3º evaluación				
Tipo	**Pista evaluación**	**Clockwise**	**Salidas de pista**	**Tiempo**
Contrarreloj	Barcelona	-	17	01:07.736
			20	01:15.267
			16	01:04.553



Conclusiones

- **Mejor modelo:** Modelo 10, debido a su rapidez y estabilidad en distintas condiciones.
- **Rendimiento medio:** Modelo 9, con tiempos aceptables y estabilidad razonable.
- **Peor modelo:** Modelo 11, afectado significativamente por las salidas de pista y los tiempos prolongados.
- **Modelo con mayor margen de mejora:** Modelo 8, que tiene potencial pero pierde rendimiento en condiciones adversas.



Comparación entre los dos mejores modelos

1. Primera evaluación (ForeverRaceway, sin salidas de pista):
   - Modelo 1:
     - Tiempos extremadamente consistentes y rápidos (14.6 a 15.1 segundos) sin salidas de pista.
   - Modelo 10:
     - También mostró buenos tiempos, pero ligeramente más lentos (15.4 a 21.3 segundos), con una leve variación entre pruebas.

Conclusión: El Modelo 1 es más eficiente en condiciones ideales (sin salidas de pista), alcanzando tiempos más bajos que el Modelo 10.

2. Segunda evaluación (ForeverRaceway, con salidas de pista):
   - Modelo 1:

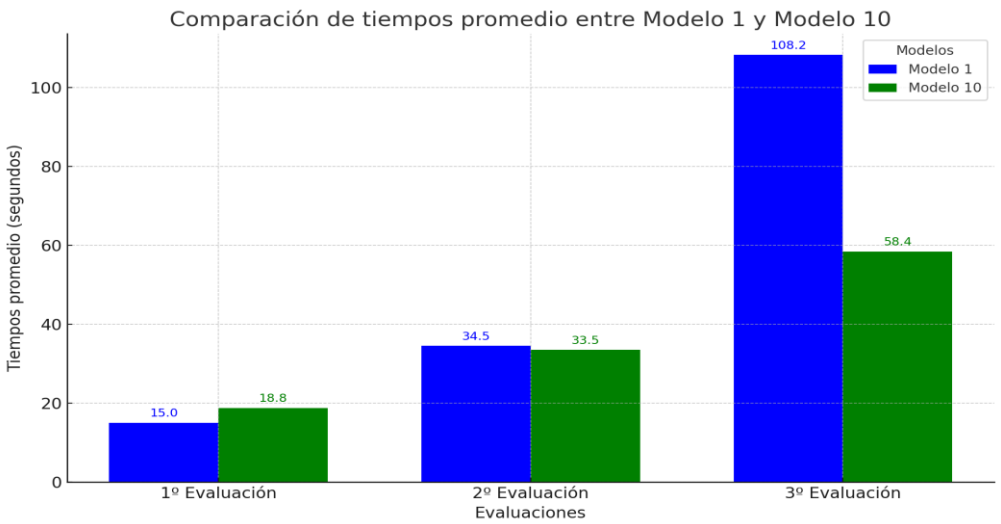
    - Sufrió de un mayor número de salidas de pista (5 a 10) y tiempos más variables (28.7 a 41.0 segundos).
- Modelo 10:
    - Mostró mejor estabilidad, con menos salidas de pista (máximo de 8) y tiempos más consistentes (27.8 a 36.3 segundos).

Conclusión: El Modelo 10 supera al Modelo 1 en esta evaluación, mostrando mayor estabilidad y menor impacto en su desempeño pese a las salidas de pista.

3. Tercera evaluación (Barcelona, pista más compleja):
- Modelo 1:
    - Mostró un notable incremento en tiempos (95.5 a 109.5 segundos) y un alto número de salidas de pista (23 a 29).
- Modelo 10:
    - Fue significativamente más eficiente, con tiempos menores (58.4 a 58.9 segundos) y un número de salidas de pista notablemente reducido (máximo de 12).

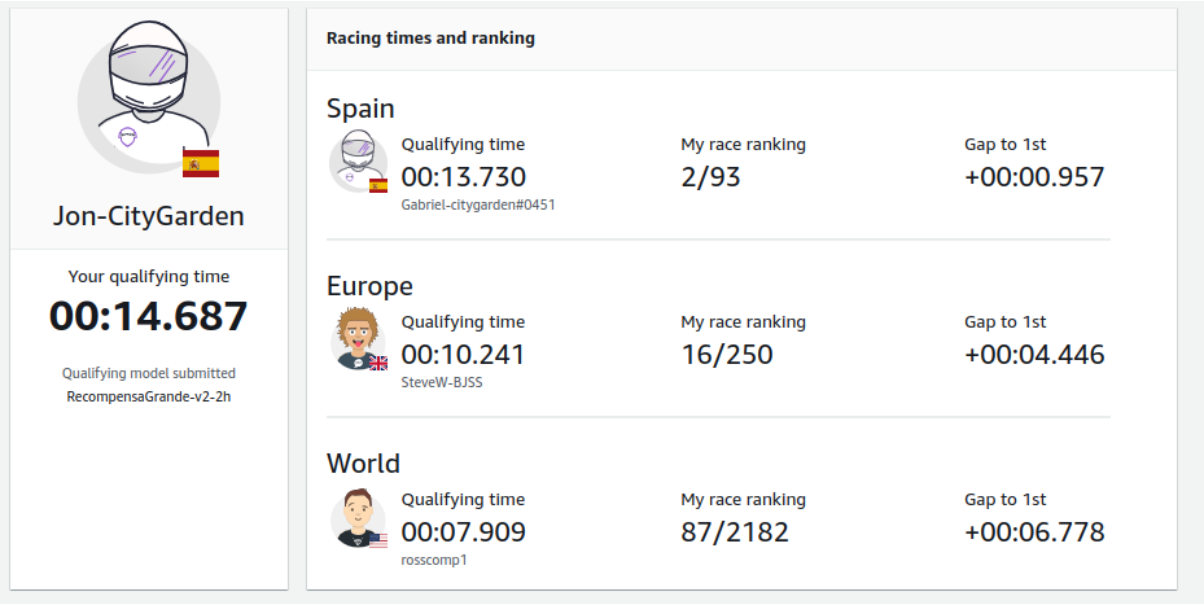
Conclusión: El Modelo 10 supera ampliamente al Modelo 1 en pistas más complejas, manteniendo tiempos más bajos y un mejor control en condiciones desafiantes.

Conclusiones generales:
1. Ventaja del Modelo 1: En pistas simples y sin salidas de pista, el Modelo 1 es más rápido y eficiente que el Modelo 10, mostrando tiempos más cortos.
2. Ventaja del Modelo 10: En condiciones adversas, como pistas más complejas (Barcelona) o cuando se producen salidas de pista, el Modelo 10 es más robusto y consistente, superando ampliamente al Modelo 1.
3. Diferencias clave:
    - El Modelo 1 parece estar optimizado para condiciones ideales, pero sufre un impacto significativo en pistas más técnicas o con mayores obstáculos.
    - El Modelo 10 se comporta como una solución más equilibrada y confiable, adaptándose mejor a diversas condiciones de evaluación.



Actualización Carrera AWS Virtual Circuit

He probado con el nuevo modelo y he mejorado un poco el tiempo anterior, aunque este sigue siendo peor que el de Gabriel.



Model name	Qualifying time	Best lap time	Average lap time	Status	Date submitted to race	Video
RecompensaGrande-v2-2h	00:18.175	00:14.200	00:18.175	Submitted	12/4/2024, 7:54 PM	View
RecompensaGrande-v2-2h	00:16.756	00:14.340	00:16.756	Submitted	12/4/2024, 7:40 PM	View
RecompensaGrande-v2-2h	00:16.561	00:14.676	00:16.561	Submitted	12/4/2024, 7:26 PM	View
RecompensaGrande-v2-2h	00:16.420	00:14.663	00:16.420	Submitted	12/4/2024, 7:05 PM	View
RecompensaGrande-v2-2h-Faster	00:43.952	00:43.858	00:43.952	Submitted	12/4/2024, 6:40 PM	View
RecompensaGrande-v2-2h	00:14.687	00:14.396	00:14.687	Submitted	12/4/2024, 6:30 PM	View
RecompensaGrande-v2	00:31.641	00:29.863	00:31.641	Submitted	12/4/2024, 4:45 PM	View
RecompensaGrande-2	00:28.332	00:22.932	00:28.332	Submitted	11/27/2024, 4:26 PM	View
RecompensaGrande	00:18.532	00:15.004	00:18.532	Submitted	11/21/2024, 5:02 PM	View
RecompensaGrande	00:15.173	00:14.594	00:15.173	Submitted	11/21/2024, 4:49 PM	View
RecompensaGrande	00:15.685	00:15.195	00:15.685	Submitted	11/21/2024, 3:11 PM	View
RecompensaGrande	00:17.353	00:15.598	00:17.353	Submitted	11/20/2024, 6:40 PM	View
RecompensaGrande	00:21.138	00:16.529	00:21.138	Submitted	11/20/2024, 6:15 PM	View
RecompensaGrande	00:16.597	00:14.535	00:16.597	Submitted	11/20/2024, 5:36 PM	View
RecompensaGrande	00:18.978	00:15.461	00:18.978	Submitted	11/20/2024, 5:24 PM	View
RecompensaGrande	00:18.979	00:15.466	00:18.979	Submitted	11/20/2024, 5:00 PM	View