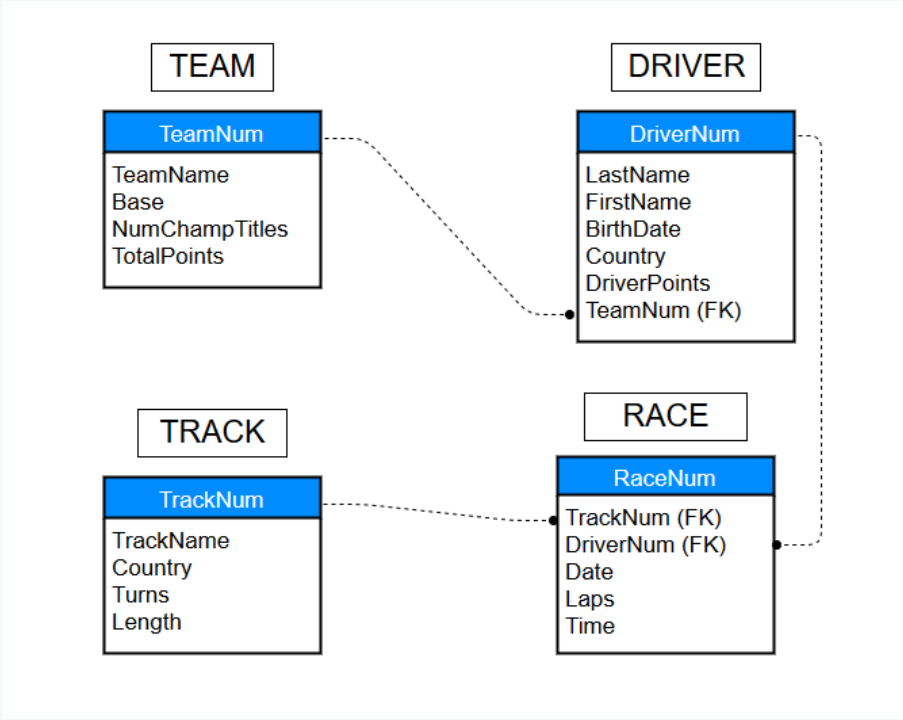
F1 Live Bot: Real-Time Formula 1 Race Updates on Telegram

F1 Live Bot:

- **Real-Time Updates** Receive live race standings and lap times.
- **Telegram Integration** Updates delivered directly to your Telegram chat.
- **Race Simulation** Test mode to simulate historical races for analysis.
- **Automated Updates** Messages update in place for a clean chat experience.

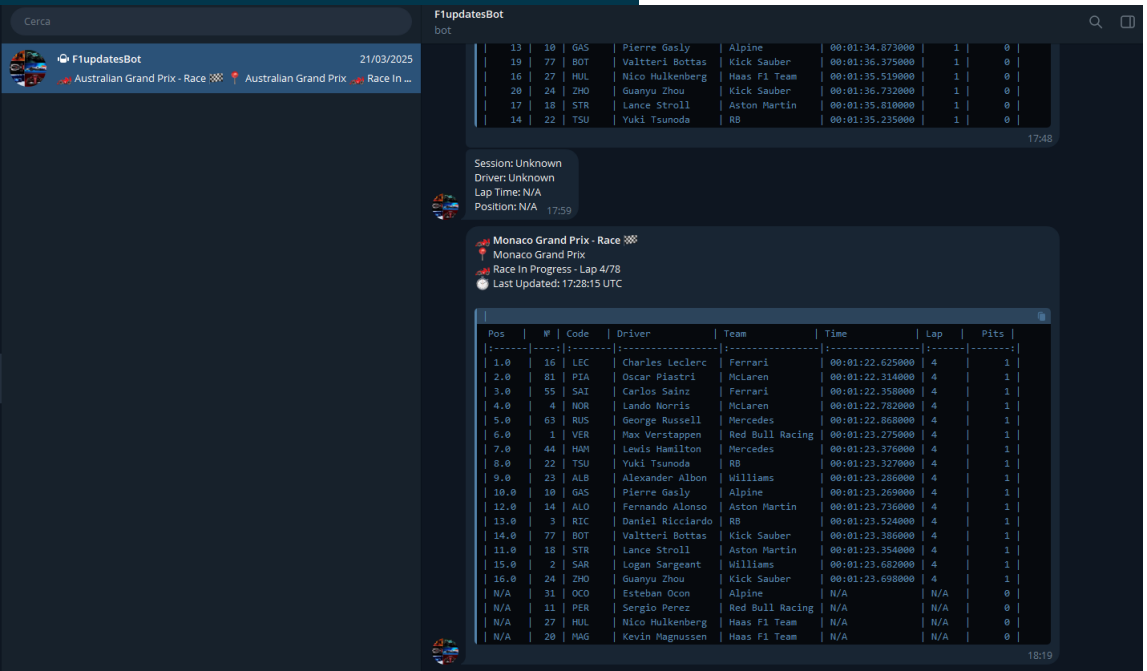
How It Works

1. **Start the System** Deploy the Docker containers with your configuration.
2. **Connect Telegram** Link your Telegram bot to receive updates.
3. **Track Live Races** Automatically fetches data during race weekends.
4. **Simulate Past Races** Test mode allows replay of historical races.



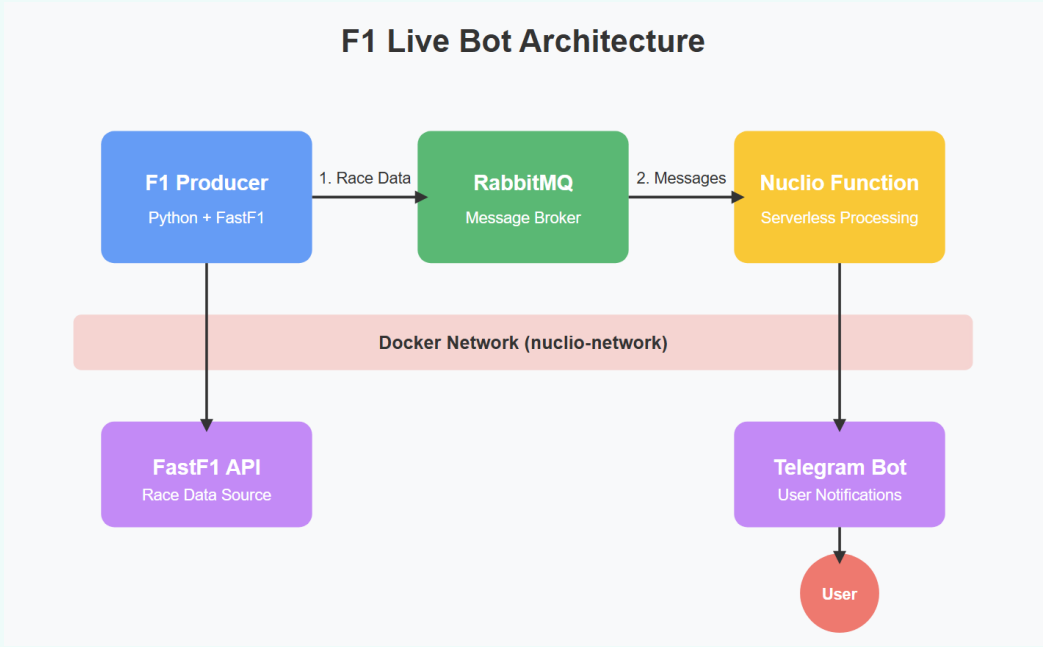
Live Race Updates Features

- Position tracking
- Lap times
- Driver information
- Team details
- Pit stop counts
- Current lap vs. total laps
- Real-time standings



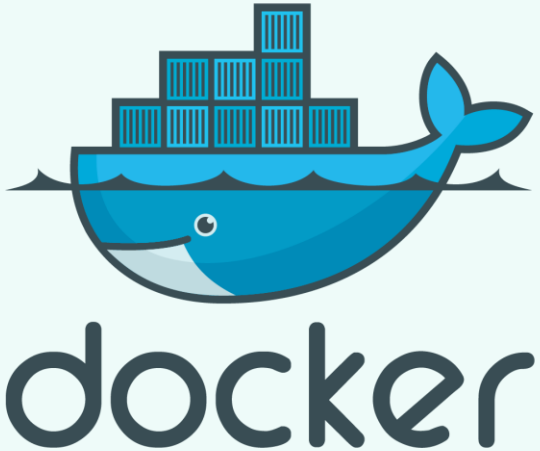
System Architecture

- **F1 Data Producer** Fetches race data from FastF1 API
- **RabbitMQ Message Broker** Handles message queuing between components
- **Nuclio Serverless Function** Processes messages and sends Telegram notifications
- **Telegram Bot API** Delivers formatted race updates to users



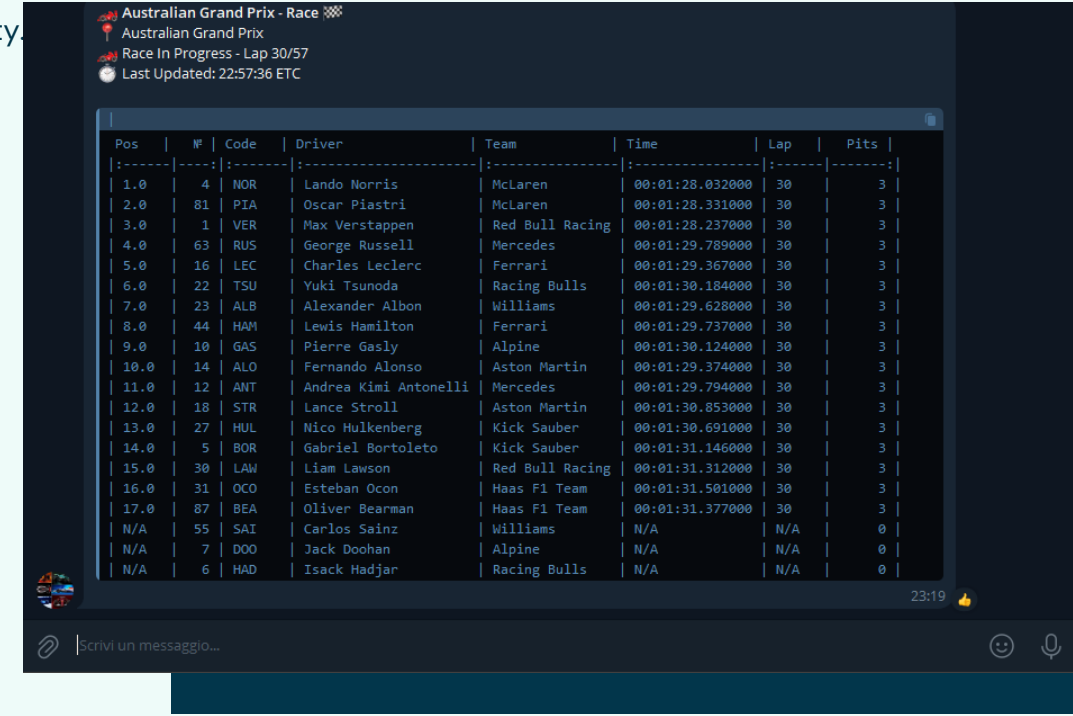
Technical Components

1. **Docker Containerization** Easily deployable on any system with Docker support.
2. **FastF1 API Integration** Professional-grade F1 timing data source.
3. **Message Queue Architecture** Reliable message delivery with RabbitMQ.
4. **Serverless Processing** Efficient message handling with Nuclio functions.



Advantages of F1 Live Bot

- **Free** No subscription costs for basic functionality.
- **Available 24/7** Accessible whenever races are happening.
- **Familiar Interface** Uses the Telegram platform you already know.
- **Customizable** Configure update frequency and test modes.



Pos	№	Code	Driver	Team	Time	Lap	Pits
1.0	4	NOR	Lando Norris	McLaren	00:01:28.032000	30	3
2.0	81	PIA	Oscar Piastri	McLaren	00:01:28.331000	30	3
3.0	1	VER	Max Verstappen	Red Bull Racing	00:01:28.237000	30	3
4.0	63	RUS	George Russell	Mercedes	00:01:29.789000	30	3
5.0	16	LEC	Charles Leclerc	Ferrari	00:01:29.367000	30	3
6.0	22	TSU	Yuki Tsunoda	Racing Bulls	00:01:30.184000	30	3
7.0	23	ALB	Alexander Albon	Williams	00:01:29.628000	30	3
8.0	44	HAM	Lewis Hamilton	Ferrari	00:01:29.737000	30	3
9.0	10	GAS	Pierre Gasly	Alpine	00:01:30.124000	30	3
10.0	14	ALO	Fernando Alonso	Aston Martin	00:01:29.374000	30	3
11.0	12	ANT	Andrea Kimi Antonelli	Mercedes	00:01:29.794000	30	3
12.0	18	STR	Lance Stroll	Aston Martin	00:01:30.853000	30	3
13.0	27	HUL	Nico Hulkenberg	Kick Sauber	00:01:30.691000	30	3
14.0	5	BOR	Gabriel Bortoleto	Kick Sauber	00:01:31.146000	30	3
15.0	30	LAW	Liam Lawson	Red Bull Racing	00:01:31.312000	30	3
16.0	31	OCO	Esteban Ocon	Haas F1 Team	00:01:31.501000	30	3
17.0	87	BEA	Oliver Bearman	Haas F1 Team	00:01:31.377000	30	3
N/A	55	SAI	Carlos Sainz	Williams	N/A	N/A	0
N/A	7	DOO	Jack Doohan	Alpine	N/A	N/A	0
N/A	6	HAD	Isack Hadjar	Racing Bulls	N/A	N/A	0

Configuration Options

- Race update frequency
- Test mode with historical races
- Race selection by year and event
- Live race simulation speed
- Starting lap configuration

.env

```
# Telegram Bot Configuration
TELEGRAM_BOT_TOKEN=
TELEGRAM_CHAT_ID=

# RabbitMQ Configuration
RABBITMQ_USER=guest
RABBITMQ_PASS=guest

# Application Configuration
CHECK_INTERVAL=60
TEST_MODE=True
TEST_YEAR=2025
TEST_RACE=Australia #https://en.wikipedia.org/wiki/List_of_Formula_One_Grands_Prix use this for names of gps
SIMULATE_LIVE=True
```

Future Development

Short Term:

- Driver-specific alerts
- Race prediction features
- Customizable update formatting

Long Term:

- Multi-language support
- Weather integration
- Strategy predictions
- Web dashboard interface

Start Now with F1 Live Bot

1. **Set Up Docker** Install Docker and Docker Compose.
2. **Create Telegram Bot** Get a token from @BotFather.
3. **Deploy the System** Follow the quick start guide.
4. **Enjoy Real-Time Updates** Experience F1 races like never before.

GitHub - emanuelemusto/F1LiveBotIot

Quick Start

1. Clone this repository:

```
git clone https://github.com/emanuelemusto/F1LiveBotIot.git
cd F1LiveBotIot
```

2. Create a `.env` file in the project root with your configuration:

```
# Telegram Configuration
TELEGRAM_BOT_TOKEN=your_telegram_bot_token_here #use token generated before
TELEGRAM_CHAT_ID=your_chat_id_here #chat id of your telegram bot

# Application Configuration
CHECK_INTERVAL=90
TEST_MODE=True
TEST_YEAR=2024
TEST_RACE=Italian
SIMULATE_LIVE=True
CURRENT_LAP=1
```

3. Start the entire system with Docker Compose:

```
docker-compose up -d
```

4. Monitor the logs:

```
docker-compose logs -f
```

5. Connect to localhost:8070 and, after creating a new project, create a new function importing the function.yaml file present in nuclio folder Remember to update my_token and my_telegram_ID and rabbitmq url, you can check rabbitmq url by running this line:

```
docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' f1livebot1-rabbitmq-1
```

6. Check host in connection settings of function and click deploy