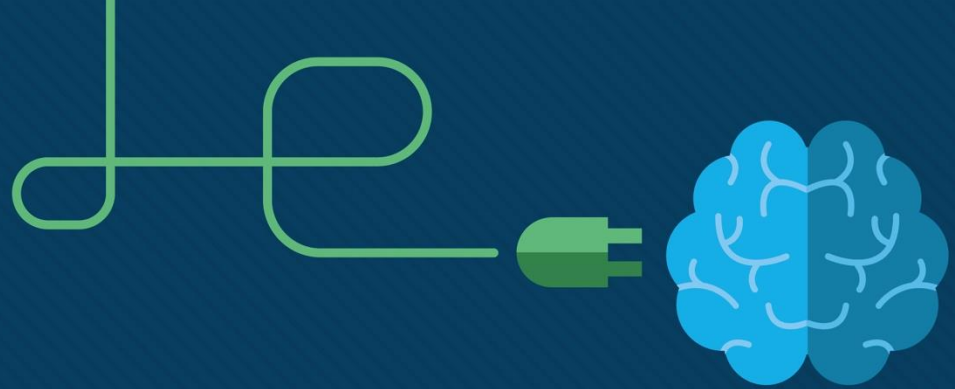


GFO TFE Team of Technical Managers





From Sensor to Sense

PL-App Demo

Eugene Morozov

Technical Manager CEE-RCIS, N&B

20 April 2018
Fulda

#NetAcadIPD





San

5



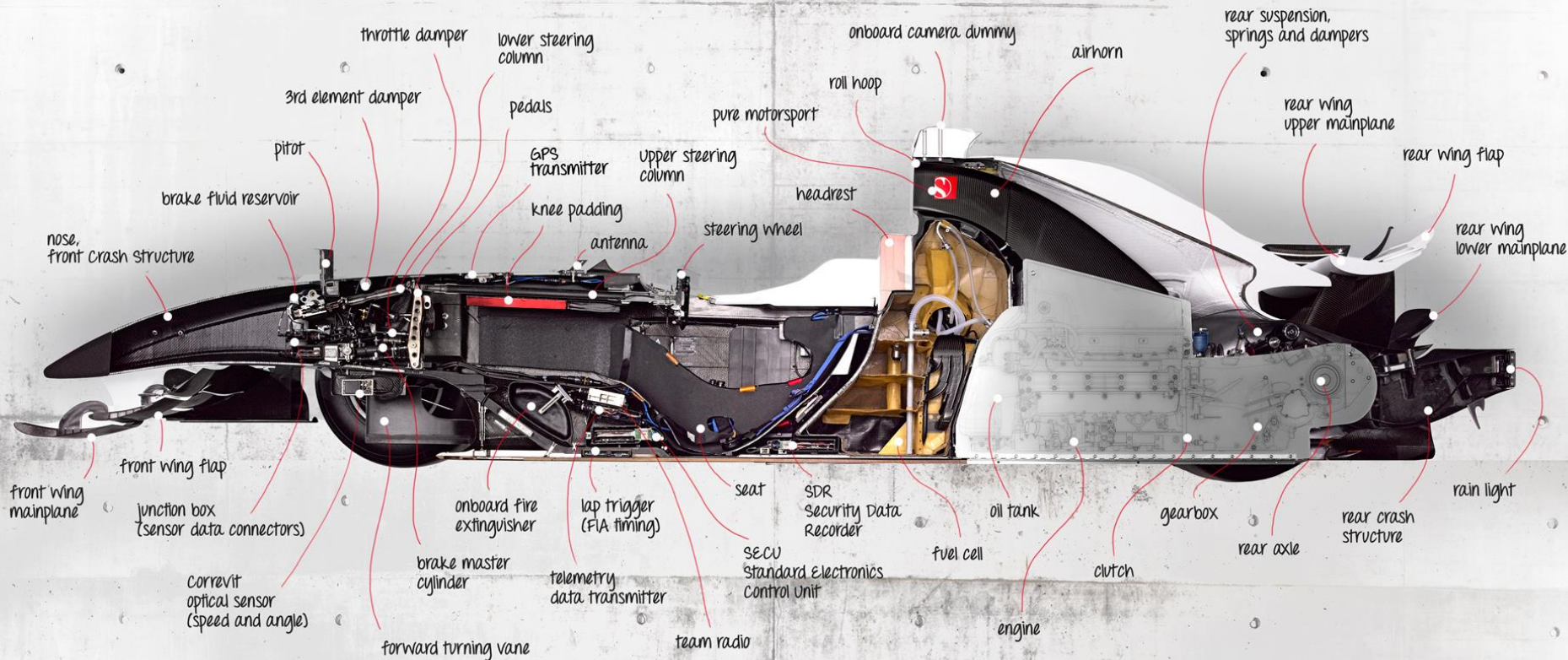
KASPERSKY'S



MAHLE



Sauber F1 Team



Collecting the data

IoT Fundamentals: Connecting Things

Course Overview

Students learn how to securely interconnect sensors, actuators, microcontrollers, single-board computers, and cloud services over IP networks to create an end-to-end IoT system.

Benefits

Students will develop multi-disciplinary skillsets required to prototype an IoT solution for a specific business case with a strong focus on the security considerations for emerging technologies.

Learning Components

- Understand and explain the concepts, opportunities and challenges of digital transformation using IoT.
- Interconnect sensors/actuators, microcontrollers (Arduino), Single Board Computers (Raspberry Pi) and cloud services (Cisco Spark restful API) to create an end-to-end IoT system.
- Understand the relevant aspects of cybersecurity and privacy for an IoT solution.
- Understand how digitalization is changing vertical markets such as manufacturing, energy, and smart cars.
- Use simulation tools (Packet Tracer) to create end-to-end IoT system.



Features

Target Audience: Secondary, Vocational, 2-year and 4-year College, 4-Year University students

Prerequisites: Basic programming, networking and electronics

Languages: English

Course Delivery: Instructor-led

Estimated Time to Complete: 40-50 hours

Recommended Next Course: IoT Fundamentals: Big Data & Analytics or Hackathon Playbook

Instructor Training: Required

Cisco Prototyping Lab

Tool Overview

The Cisco Prototyping Lab is a comprehensive learning environment created by Cisco for Networking Academy students to learn and practice key aspects of the foundational IoT technologies. Using an engaging, hands-on approach, it supports both the learning and creative phases of the Networking Fundamentals curriculum.

Career Prep

Provides an easy to use, comprehensive learning environment using real devices, code, coding tools and data that students use to create the physical interconnection of an end-to-end IoT and the logical data pipeline to acquire, analyze and present data.

Learning Components

- Prototyping Lab App
- Prototyping Lab Kit
 - Raspberry Pi 3 CanaKit Ultimate Starter Kit (or equivalent)
 - SparkFun Inventor's Kit for Arduino v3.2 (or equivalent)
 - Cables, sensors & actuators

Features

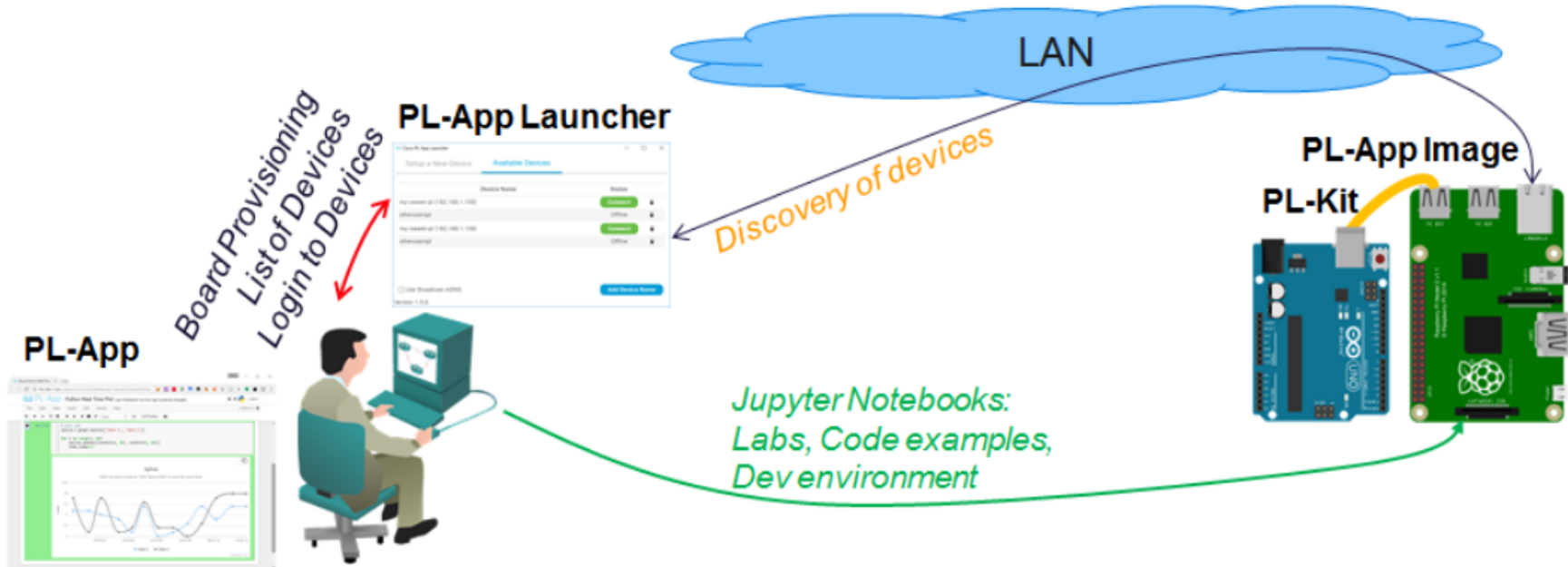
As an integral part of the Networking Academy learning experience, Cisco Prototyping Lab provides

- Interactive labs using Jupyter Notebook
- Visual programming with Blockly
- Device programming with Python
- Data visualization & analytics
- Connected applications via APIs
- Rapid Prototyping



Prototyping Lab App

- Standard tool used in IoT Fundamentals courses



PL-App Launcher

Prototyping Lab App (PL-App) Launcher Application

Cisco PL-App Launcher

Setup a New Device Available Devices

- 1 Insert the SD card reader into the USB port and select it from the dropdown menu:

 Refresh
- 2 Select the PL-App image file that you have downloaded from NetAcad.com:

 Find Image:
- 3 Create a unique Device Name and Password for the PL-App device:

 Device Name: *Include your name or initials (e.g. jj-pi1984)*

 Device Password: *Password to access PL-App on your device*
- 4 Optional settings (connecting the device to an existing Wireless LAN):

 WiFi SSID:

 WiFi Password:
- 5

Version 1.5.8



Prototyping Lab App (PL-App) Launcher Application

Cisco PL-App Launcher

Setup a New Device Available Devices

Device Name	Status
my-sweet-pi (192.168.1.109)	<input type="button" value="Connect"/>
otheruserspi	Offline <input type="button" value=""/>

Use Broadcast mDNS

Version 1.5.8

PL-App Notebooks

Notebook

Markdown

▶ Code

Markdown

▶ Code

▶ Code

Markdown

Combines:

- Explanatory text, graphics, media
- Programming code in Python, Bash or Blockly
- Data visualization

The screenshot shows the PL-App interface for a 'Real Time Plot in Blockly'. The workspace contains the following Blockly blocks:

- Graph with chart type Line** (green block)
- create list with** (purple block) containing `Label 1` and `Label 2`
- repeat** (green block) set to **10** times
- do** (green block) containing:
 - Graph update values** (green block)
 - create list with** (purple block) containing `random integer from` and `random integer from`
 - sleep** (green block) set to `0.5` seconds

The right-hand pane shows the following Python code:

```

1 report chartstr: graph as graph
2 report random
3 from time import *
4
5
6 chart = graph.Line(['Label 1', '
7 for count in range(10):
8     chart.update([random.randint(1,
9     sleep(0.5)
10

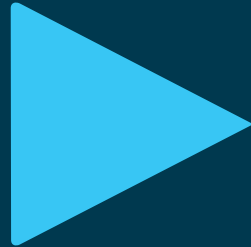
```

The bottom section displays a line graph titled 'Line' with the following data series:

Time	Label 1 (Value)	Label 2 (Value)
01:52:36.000	85	40
01:52:36.500	50	35
01:52:37.000	20	15
01:52:37.500	10	55
01:52:38.000	45	35
01:52:38.500	10	5
01:52:39.000	70	50

Code

- Infinite loop
- Variables:
 - time
 - vibration
 - light
- Time = seconds from start
- Vibration = read from pin Analog 1
- Light = read from pin Analog 0
- Write to CSV file



[CT 5.1.1.3](#) Connecting and Digitizing Industry

[BDA 1.2.1.3](#) intelligent devices in cars can be used to facilitate day-to-day activities

[BDA 1.3.1.1](#) challenges of Big Data



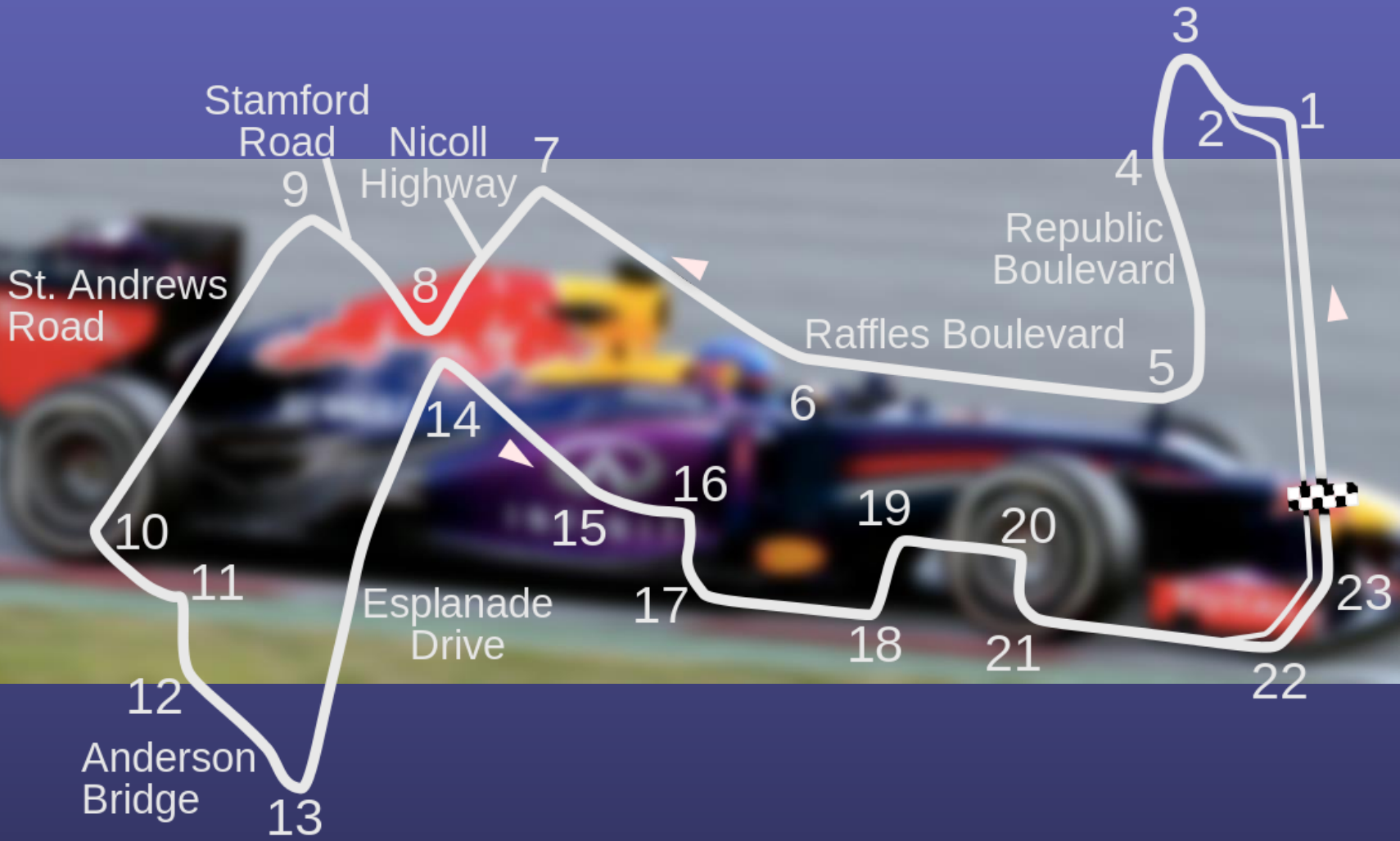
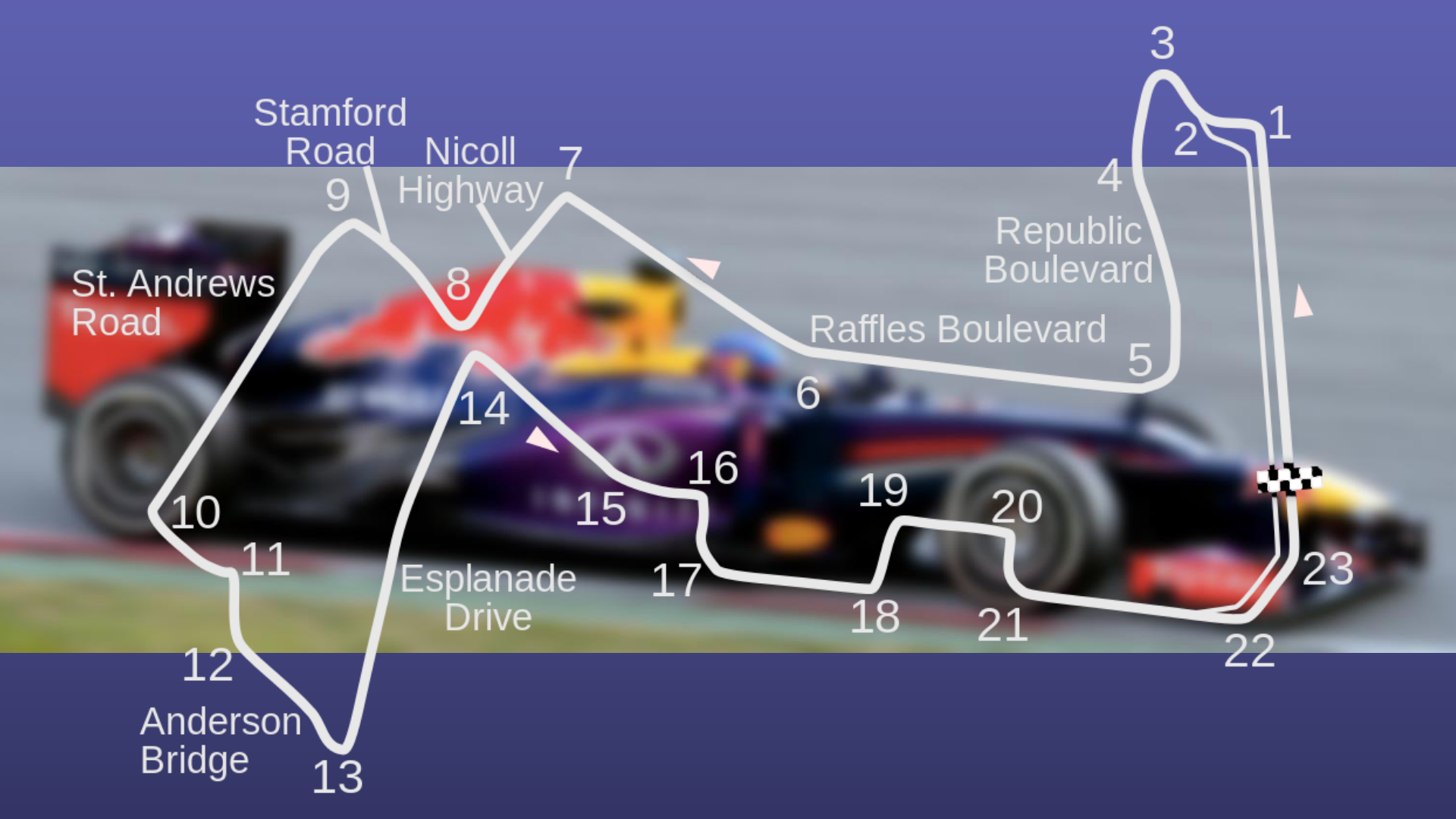
“The network will play a crucial part in how we develop the car; gathering data, learning from it and adapting will ultimately determine our season.”

Christian Horner
Team Principal, Infiniti Red Bull Racing



“The network will play a crucial part in how we develop the car; gathering data, learning from it and adapting will ultimately determine our season.”

Christian Horner
Team Principal, Infiniti Red Bull Racing



Analyzing the data

IoT Fundamentals: Big Data & Analytics

Course Overview

Students will learn how to use Python data libraries to create a pipeline to acquire, transform and visualize data collected from IoT sensors and machines.

Benefits

The transformative element of any IoT system is the data that can be collected from it. Thus the ability to extract data and using data analytics techniques to gain insights increases employability.

Learning Components

- Use Python to read data from sensors and store data in a SQL data base.
- Use Python Data Analysis library to clean, manipulate, integrate data sets.
- Use Python Visualization Libraries to visualize real-time data and explore acquired data sets.
- Explain the fundamental principles of a modern scalable Big Data platforms like Hadoop.
- Use storytelling to present the insights gained from extracted data.



Features

Target Audience: Secondary, Vocational, 2-year and 4-year College, 4-Year University students

Prerequisites: IoT Fundamentals: Connecting Things

Languages: English

Course Delivery: Instructor-led

Estimated Time to Complete: 40-50 hours

Recommended Next Course: IoT Fundamentals: Hackathon Playbook

Instructor Training: Required

PL-App Notebooks

Notebook

Markdown

▶ Code

Markdown

▶ Code

▶ Code

Markdown



Combines:

- Explanatory text, graphics, media
- Programming code in Python, Bash or Blockly
- Data visualization

The screenshot shows the PL-App interface for a 'Real Time Plot in Blockly'. The workspace contains several Blockly blocks: a 'Graph with chart type Line' block, a 'repeat 10 times' loop containing a 'do' block with 'Graph update values' and 'sleep 0.5 seconds' blocks. The Python code editor on the right contains the following code:

```

1 report chestnut.graph as graph
2 report random
3 from time import *
4
5
6 chart = graph.Line(['Label 1', '
7 for count in range(10):
8     chart.update([random.randint(1,
9     sleep(0.5)
10

```

The graph below shows two data series, 'Label 1' (blue line) and 'Label 2' (black line), plotted over time. The x-axis represents time in seconds, and the y-axis represents 'Value' from 0 to 100. The graph shows two lines fluctuating over time, with 'Label 1' generally having higher values than 'Label 2'.

IoT Fundamentals Course Summary



Connecting Things

Students learn how to securely interconnect sensors, actuators, microcontrollers, single-board computers, and cloud services over IP networks to create an end-to-end IoT system.

Students will develop multi-disciplinary skillsets required to prototype an IoT solution for a specific business case with a strong focus on the security considerations for emerging technologies.

Course Delivery: Instructor-led
Estimated Time to Complete: 40-50 hours

Big Data & Analytics

Students will learn how to use Python data libraries to create a pipeline to acquire, transform and visualize data collected from IoT sensors and machines.

The transformative element of any IoT system is the data that can be collected from it. Thus the ability to extract data and using data analytics techniques to gain insights increases employability.

Course Delivery: Instructor-led
Estimated Time to Complete: 40-50 hours

Hackathon Playbook

The Hackathon Playbook is a comprehensive framework of tools and templates to prepare and run a Hackathon as a result of best practices and lessons-learned collected from the global execution of IoT Hackathons within Networking Academy and by other organizers.

Student reinforce and deepen their multidisciplinary IoT and data skills by defining, designing, prototyping and presenting an IoT solution to a panel of industry experts and peers.

Course Delivery: Instructor-led
Estimated Time to Complete: 20-30 hours

A New NetAcad Hands-On Experience

IoT Fundamentals | Lab Experiences



Analyze the Problem
with User Focus



Hands-on Design and
Maker Mindset



Rapid Prototyping, Iterating,
Presenting

