

Index

A

Amazon Web Services (AWS), 291

Auth0

- application, 243
- create authorization, 248, 250, 252
- create user, 253–256
- setting up, 243–246, 248

Authentication endpoints

- Auth0, 243
- authorization *vs.* authentication, 240
- login across lines, 239
- OAuth 2, 240, 241
- OAuthZ/AuthN, applying, 241, 242
- tokens
 - access token, 268–270
 - authorization check, 271–273
 - ID, 264, 265
 - programmatically parse, 266, 267
 - refresh, 274, 275
- world Wide Web's mainstream, 239

Authorization, 240

B

Backend design, 37

Borrowing and ownership

- memory, 23–25
- passing method and returning, 23
- passing string, 22
- parsing u32, 22

reference/dereferencing, 25

setting variable, 21

C

Camera

cross compilation

Docker, 490, 491

OpenCV libraries, 490

Pi, 490

rust-embedded project, 491–493,
495, 497, 499

rust-embedded project, 496

deploying Pi, OpenCV install, 520–523

facial recognition, 485

goals, 484

installation, 485, 486, 488, 489

video content, type, 483, 484

Cap'n Proto

advantages, 206, 207

definition, 204

installation, 210

vs. JSON, 209

microservices, 209

Protobuf, 205, 206

RPC, 207, 208

rust

client, creating, 216, 218, 219

implementing application, 208

message, define, 211–215

MQTT layer, 219–225

INDEX

Cap'n Proto (*cont.*)

- retrieval_svc, 225
- RPC server, creating, 226–234

Cargo

- commands, 29, 30
- default package manager, 29
- feature flags, 30
- release process, 30, 31

Command Query Responsibility and

Segregation (CQRS), 183

- aggregator, 186–189
- asynchronous layer, 178
- command and event, 184–186
- commanded endpoint, 190–196
- definition, 175
- dispatcher, 189, 190
- environment, setting up, 182
- eventsourcing, 180
- event stream
 - eventsourcing package, 197
 - executor creation, 197, 198
 - stream monitor, 199–204
- eventual consistency, 177
- Gotchas, 182
- implementing
 - application, 181, 182
 - solution, 180
 - steps, 178, 179
 - typical model, 175, 176

Commands, sending

- application
 - client side, 546, 547
 - IPC client, 547–550
 - IPC server side, 544, 545
 - message queue,
 - subscribe, 542–544
 - video loop, integration, 550, 552
- video file, uploading, IPC, 541

Configurations, Kubernetes

- controllers (*see* Controllers)
- ingress controller, 339, 340
- namespace, 322, 323
- service (*see* Services)
- yaml files, 322

connection_method, 286

Containerization

- abstraction layer, 294
- application deployments, 295
- docker layers, 300–303
- full machines, 294
- virtual machine, 295, 296

Containers

- build execution, 305
- cargo build--release, 304
- docker, 297
- image tag, 306
- kernel, 298
- multiple VMs, 299
- rust application, 304
- self-contained images, 298
- server-side application, 299
- VMs, 298

Controllers

- accessMode, 331
- basic_deploy.yaml, 325
- deployment, 331
- flow from pod to filesystem, 328
- get pods, 333
- get pvc, 334
- iot namespace, 323, 324
- ki get pods, 325
- kubectl command, 330
- local/upload_svc image, 323
- logging, 335
- PersistentVolumeClaim, 329, 330
- pod configuration, 326

- port-forward, 335
- re-query, 326
- RESTful endpoints, 331
- StatefulSet, 323, 332
- upload-dep-basic deployment, 327
- volumeClaimTemplate, 334
- Cryptographic operations, 416
- Custom buildroot
 - Dockerfile, 578–581
 - Linux, 578
 - makefile build options, 582
 - makefiles and configurations, 578
 - package copying, 582
 - provisioning application, 584
 - sdcard.img, 583
 - software, 578
 - wireless interfaces, 581
- D**
- Database design
 - architecture, 39
 - enumerations, 39
 - media_datas table, 38
 - one-to-many relationship, 38
- Data services
 - cloud provider, 290
 - data store interactions, 290
 - EventStore, 290
 - MQTT, 291
- Degrees, minutes, seconds
 - (DMS), 101
- Deployment
 - data services, 290
 - Docker (*see* Docker)
 - gitlab-ci, 289
 - Kubernetes, 389
 - microservices, 290, 389
 - Postgres, 291
 - traditional methodology, 291
- Deployments, helm
 - built-in systems, 347
 - deletes/redeploys, 348
 - dependencies, 348
 - directory structure, 350, 351
 - EMQX chart, 349
 - eventstore, 348
 - helper template, 352
 - ingress, 365, 366
 - MQTT (*see* MQTT service)
 - sealed secrets, 350
 - secret service, 353, 354
 - template files, 351
- Device authentication flow
 - authentication library, 470, 471
 - Auth0 FlowDelegate, 476–478
 - entry point struct, 475, 476
 - method, 471–473
 - rasp-auth library package, 469
 - Raspberry Pi app integration, 479, 481
 - VisualDisplay trait, 474, 475
 - Yup OAuth 2, 468, 469
- device_code, 259
- device_id, 535
- Diesel
 - get_connection, 76
- Diesel
 - command-line tool, 57
 - configuration app, 57, 58
 - database tables, 58, 59
 - deleting, 68
 - derive enum crate, 71–73
 - enumerations, 69 (*see* Enumerations)
 - inserting, 66, 67
 - middleware (*see* Middleware)
 - ORM, 65

INDEX

Diesel (*cont.*)

- querying, 68, 69
- schema, 56
- standard relationships, 73
- struct, 65
- UUID, 74, 75
- well-documented full-featured
 - extensible, 56

Digital Ocean

- adding domain, 378
- contexts/switch, 383
- creating cluster, 376
- depolying, helm chart, 386–388
- dns entry, 379, 380
- docker images, 385, 386
- Gitlab CI/CD pipeline, 384
- kubeconfig, 381
- kubectrl command, 382
- kube-system, 381
- minimal configuration, 372
- Postgres instance, 372
- selecting data center and pools, 374
- setting name, cluster, 375
- setup page, 373

Docker

- artifacts, 292
- containerization (*see* Containerization)
- database
 - GIS, 59, 60
 - name 60
 - p 61
 - Postgres, 59
 - Redis+Mongo+Spring app, 59
 - running containers, 61
 - running generation scripts, 62–64
 - schema file, 64, 65
 - starting, 62
 - stopping, 61

PaaS, 292

- virtual machines, 292

Dockerfiles

- MQTT service, 311, 312
- retrieval service, 312, 313
- upload service, 310

E

Enumerations

- media_datas, 71
- MYSQL, 70
- one-to-many relationship, 69
- types, creation, 70

Eventual consistency (EC), 9

Exchangeable Image File Format (EXIF), 94

F

Facial recognition, 485

File uploads

- FileMetaData enum, 117
- HTTP requests, 110, 111
- middleware crate, 113
- multipart, 110, 111
- retrieval service, 110
- router, 112
- SaveData enumerations, 116, 117
- saving the file, 118, 119
- send_to_retrieval_svc, 115
- unique id, 111
- frames_per_file variable, 513

G

Geographic information system (GIS), 59, 101

Google Cloud Platform (GCP), 291

grant_type, 260

GraphQL

- Facebook, 135
- Juniper, 141
- mutations, 138, 139
- playground, 140, 141
- problems, 134
- REST, 134
- RESTful query, 135–137
- subscriptions, 139

H

handle_command, 187, 188

handle_video function, 512

HapKit Accessory Protocol
(hap-rs), 554

Helm

- chart data, 346
- chart file structure, 344, 345
- deploy directory, 366
- deployment (*see*
Deployments, helm)
- deployment package manager, 341
- docker-for-desktop, 369
- dry run/debug, 367, 368
- Golang style syntax, 341
- image tags, 342
- installation, 342, 343
- manual process, 340
- memory/CPU applications, 342
- microservices, 341
- release attributes, 346
- replicas, 342
- retrieval service (*see* Retrieval
service, helm)
- templates, 345
- values.yaml, 346, 347
- yamls, 342

HomeKit

- add accessory, 569, 570
- add available devices, 571
- Apple, 553
- definition, 553
- generic temperature sensor,
implementation, 557–559
- hap-rs
 - accessories, 554
 - definition, 554
 - motion sensor,
characteristics, 556
 - Pi configuration, 556, 557
 - thermostat, characteristics, 555
- manager module, 566
- module, 567, 568
- motion sensor,
implementation, 563–565
- Rasp Pi
 - display message, add, 572
 - motion sensor, add, 574, 575
 - thermostat sensor, add, 573
- temperature sensor,
implementation, 560, 562
- video module, 565

I

i2c (pronounced eye-squared-see)
protocol, 435

Interactions, creating

- channels, 456
- commands, 456
- daily module, 459, 461, 462
- joystick, 463, 464
- modules, 456
- rx.recv().await, 465–467
- tokio async run, 457, 459

INDEX

Internet of Things (IoT)

- beneficial application, 3
- board application, 12, 13
- Cloud, 9
- consumer, 1
- customizable solution, 2
- deploying, 40
- devices, 1, 2
- docker, 41
- hardware, 5
- hardware device, 1
- HTTPS path, 5
- microservice architecture, 6
- ML, 6
- mobile devices, 5
- MOSFET, 1
- MQs, 5
- open source community, 2
- OTA, 3
- Raspberry Pi, 9
- reference, 2
- reuse certificates, 577
- software libraries, 577
- source code, 10
- techniques, 9
- TLS, 6
- web application, 10–12
- Wi-Fi connected, 1

Inter-process communication

(IPC), 540

ipc-channel, 541

J

- Java Web Token (JWT), 264
- Joystick control, sensehat-stick-rs, 455
- JSON Web Token (JWT), 241
- Juniper

- context object, 142
- creating mutations, 146
- creating queries, 144, 145
- integrating iron
 - route, 147–149
- setting up, 142
- setting up schema, 143

K

Kubernetes

- configurations (*see* Configurations, Kubernetes)
- container orchestration, 314
- Docker control plane, 320
- helm charts, 371
- Kubeadm, 371
- kubectl, 321
- master, 318, 319
- Minikube, 319
- node (*see* Nodes)
- pod, 314, 315

L

- Learning rust, 15, 16
- Lightweight Directory Access Protocol (LDAP) system, 240

M

- Machine learning (ML), 6
- Message queues (MQs), 5
- Message Queuing Telemetry Transport (MQTT), 149
 - certificates
 - certificate authority (CA) *vs.* self-signed, 277

- client, 284, 286
 - creating server, 278
 - container portion, 362, 363
 - HTTP port, 364
 - message queue
 - service, 286–288
 - TCP, 275, 276
 - Messaging
 - calling messaging service
 - Eclipse Mosquitto, 152
 - JSON example, 153
 - microservice, 156, 157
 - multi-level wildcard (+), 154
 - publish, 155
 - single-level wildcard (+), 153
 - subscribe, 154
 - design, 150
 - health topic, subscribe,
 - 160–163, 165, 166
 - IoT devices, 149
 - local messaging service, 151, 152
 - message queue
 - middleware, 167–170
 - publish function, 166
 - publish-subscribe protocol, 149
 - recording module, 170–172
 - Metadata
 - creating struct and sending HTTP
 - request, 122
 - FileUpload, 121
 - parsers, 120
 - parsing incoming data, 129
 - request Client, 124
 - retrieval service, 121
 - saving data, 130
 - storing, 124
 - updating database, 125, 127
 - updating structs, 127, 128
 - Metal-oxide-semiconductor field-effect
 - transmitter (MOSFET), 1
 - Microservices
 - application by service, 34
 - cloud high availability, 35
 - cross-service
 - communication, 34
 - external application, 35
 - large *vs.* small instances, 36
 - small instances, 35
 - Middleware
 - AfterMiddleware, 52
 - AroundMiddleware, 52
 - BeforeMiddleware, 79, 80
 - BeforeMiddleware, 52
 - connection pool, 76, 78, 79
 - database connection, 77
 - deleting, 81, 82
 - DieselMiddlewareConnPool, 78
 - DieselReqExt, 80
 - handlers, 52
 - instantiating, 77
 - iron_diesel_middleware, 76
 - logging, 52–54
 - macros, 51
 - performing operations, 78
 - timer, 54–56
 - monitor_controls function, 548
 - monitor_notifications/subscribe
 - function, 161, 542
 - mqtt.fullname function, 352
- ## N
- Nodes
 - architecture, 316
 - cAdvisor/Prometheus, 317
 - container runtime, 317

INDEX

Nodes (*cont.*)

- Kubelet, 317
- proxy manager, 317
- virtual machine hardware, 316

O

Object-relational mapping (ORM), 56

on_read and on_update methods, 556

Open Computer Vision (OpenCV), 485

- definition, 499
- installation, 500
- running application
 - displaying camera, 504
 - facial recognition,
 - capturing, 505–507
 - image content, save, 516–518
 - running cascades, 508, 509
 - text to video, apply, 514, 515
 - trained cascades, 508
 - video capturing, 501–504
 - video content, saving, 510–514
 - video processing, calling, 519

Over-The-Air (OTA), 3

P, Q

Parameters

- JSON parsing, 47, 49, 50
- params parsing, 46, 47
- retrieve request, 45
- URL REST parsing, 47

Parsing image data

- data structure, 95, 97
- exif reader, 97, 98
- EXIF, 94

GIS, 101–104

- Kamadak EXIF, 94, 95
- OSX, 92
- types, 98–101
- uploading pictures, 92, 93

persist method, 231

Platform-as-a-service (PaaS), 292

R

Raspberry Pi, 33, 392

- assemble, 394
- board labeled, 395
- cross compile, 414, 415
- debugging cable, 393
- developing
 - arguments, 421–423
 - command-line
 - arguments, 427
 - device certificates, 421
 - heartbeat
 - creation, 423, 424, 426
 - main method, 419
 - requirements, 419
 - timer, 426
 - UUID creation, 420
- GPIO pins, 396, 398
- heat sinks, 394
- hello world application, 414
- image creation, 392
- kit, 393
- operating system, 400
 - final step, 408
 - initialize the disk, 402, 404
 - install image, 401, 405, 406
 - Linux box, 404, 405
 - screen capture, 409

- SSH access, 412
- unpack the file, 402
- Wi-Fi setup, 410–412
- SD card, 393
- Recording module, 542
- recording_monitor
 - variable, 547
- Remote Procedure Call (RPC), 204
- Representational State Transfer (REST), 134
- request/response function, 43
- RESTful endpoints, 40
- Retrieval service, helm
 - diesel migrations, 359, 360
 - env database reference, 359
 - HTTP command, 358
 - init containers, 361
 - RPC service, 359
- run_face_detect function, 502
- Rust
 - basic application, 17
 - borrowing and ownership (*see* Borrowing and ownership)
 - class-based inheritance-type
 - language, 14
 - command-line tools, 14
 - components to purchase, 7, 8
 - embedded, 15
 - Golang applications, 33
 - learning, 15, 16
 - memory management, 14
 - microservice, 14
 - modules, 14
 - multi-paradigm
 - programming, 6
 - mutation, 19
 - pre-requisites, 7
 - types, 20

- variable name, 18, 19
- web-blown features, 33

S

- Scratch containers
 - C libraries, 307
 - dockerfile, 307, 308
 - .dockerignore, 308, 309
 - musl, 308
- Security
 - authentication endpoints, 239
 - device flow, 257–263
 - goals, 238
 - techniques, 238
- send_temperature method, 562
- send_to_server, 533
- Sense HAT
 - all-in-one board, 430
 - board features, 431
 - capabilities, 429
 - chipsets, 430
 - config.txt, 434
 - GPIO extender, 433
 - hardware, 430
 - interaction (*see* Creating interactions)
 - joystick control, 431
 - login authorization flow, 429
 - multiple background processes, 431
 - Raspberry Pis, 430
 - sensors (*see* Sensors)
 - space limitations, 432
 - textual displays, 429
 - unboxing, 432
- Sensors
 - chipsets, 435
 - debugging tools, 437
 - i2c tools, installation, 436

INDEX

Sensors (*cont.*)

- LED display, 438, 439
 - controls, 444–449
 - frames, 440–443
 - adding sensehat-screen, 439
- running i2cdetect, 437
- SCL, 435
- SDA, 435

Serial clock (SCL), 435

Serial data (SDA), 435

Server certificates

- CA root, 278, 279
- file extensions, 278
- message queue
 - server, 280, 281
- MQTT server, 282–284

Service-level agreement (SLA), 526

Services

- cluster ip, 336, 337
- headless, 338
- kube-proxy, 336
- node port, 338, 339
- port-forward, 336

Single-page web applications

- (SPAs), 134, 250

SQLite

- adding entry, database, 530, 531
- database, creating
 - interactions, 529, 530
- definition, 526
- design, 527, 528
- SQL-92 features, 527
- video file, uploading, 534
 - failures, 532
 - marking entry, 538
 - marking entry, not successful, 539
 - sending to server, 535, 537
 - src/camera/db.rs, 533–535

T

Tagged Image File Format (TIFF), 94

Temperature display

- atmospheric interactions, 452–454
- Cargo.toml, 450
- feature flags, 450
- gyroscope and accelerometer, 449
- humidity sensor, 451
- thermometer's reading, 451

Traits

- abstraction, 25
- instantiating, 27, 28
- optional implementation, 26
- person to student, 27
- Typemap, 29

Transport Layer Security (TLS), 6

U

Upload service

- container section, 356
- metadata label, 355
- resources section, 358
- selector area, 355
- statefulset, 355
- values.yaml references, 357
- upload_ss.yaml file, 355

V

Video data

- audio codec, 109, 110
- data structure, 105, 106
- mp4 parser, 105
- reading file, 106
- rust, 104
- static image, 104
- VideoMetaData model, 107

Video, upload, *see* SQLite
Viola-Jones algorithm, 507

W, X, Y, Z

Web framework

- command line parameters, 82, 84, 86
- creation, Hello World, 42, 43
- handling errors, 86, 87
- Hyper, 41

- Iron, 42
- loggers, 88, 89
- mobile application, 41
- parameters (*see* Parameters)
- primary services, 41
- redirects, 44
- response codes, 44
- rocket, 42
- routing, 50, 51
- writer.release(), 530