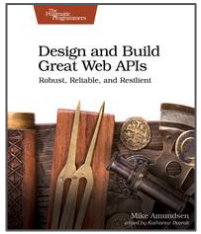
Design and Build Great APIs

@mamund
Mike Amundsen
youtube.com/mamund



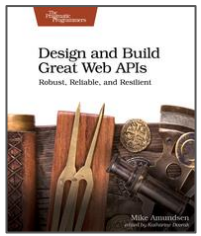
copyright © 2020 by amundsen.com, inc. -- all rights reserved

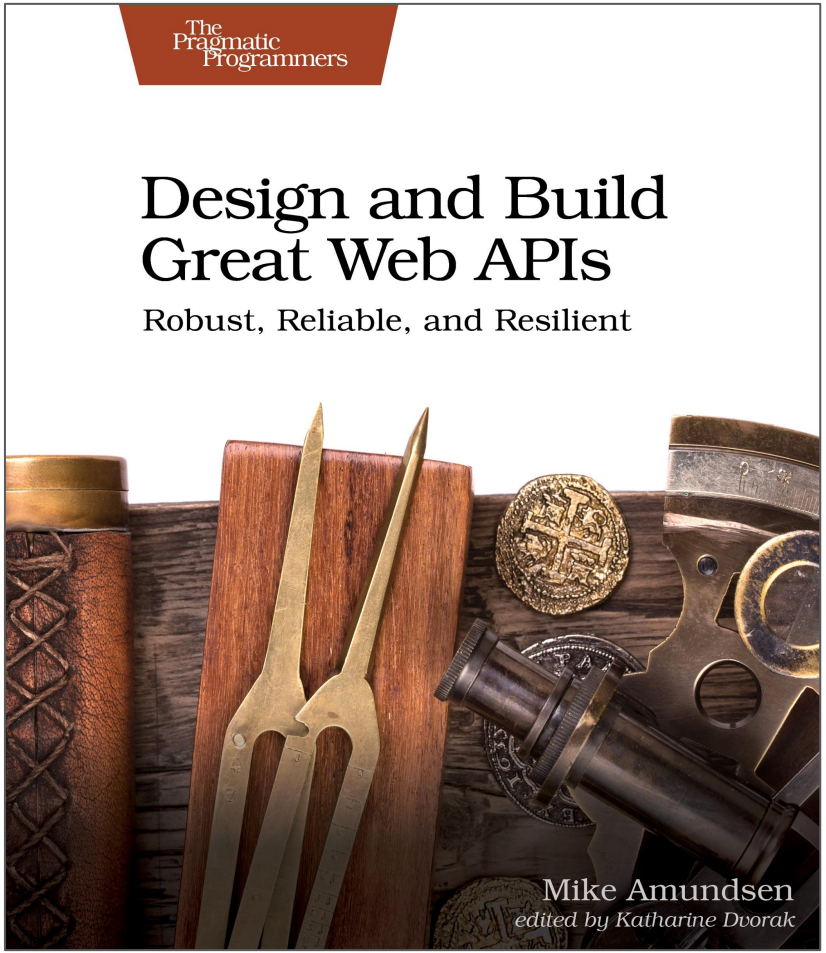
Mike Amundsen
@mamund

g.mamund.com/GreatWebAPIs

"From design to code to test to deployment, unlock hidden business value and release stable and scalable web APIs that meet customer needs and solve important business problems in a consistent and reliable manner."

-- Pragmatic Publishers

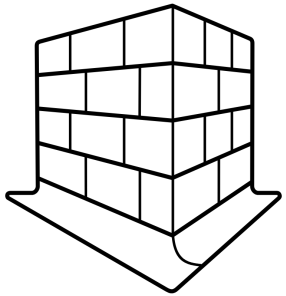


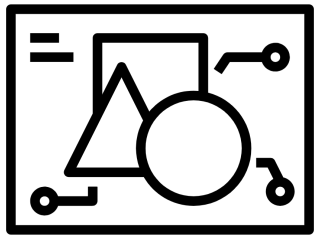


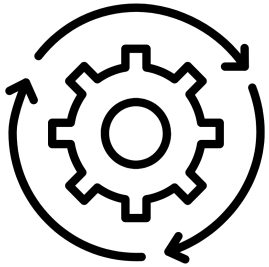
copyright © 2020 by amundsen.com, inc. -- all rights reserved

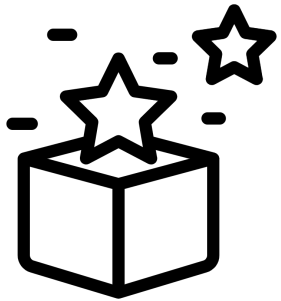
The Big Picture

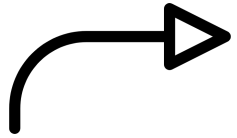
- Foundations
- Design
- Build
- Release
- And Then...

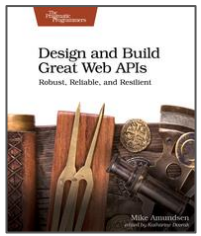




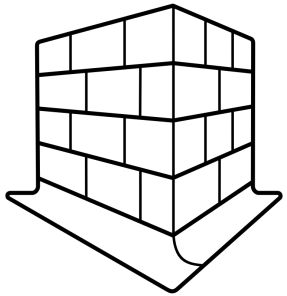








copyright © 2020 by amundsen.com, inc. -- all rights reserved

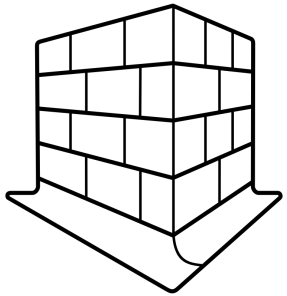


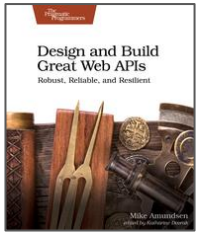
Foundations

copyright © 2020 by amundsen.com, inc. -- all rights reserved

Foundations

- API First
- HTTP, Web, & REST



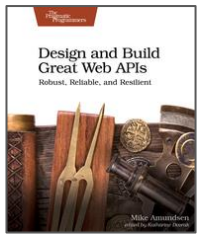


copyright © 2020 by amundsen.com, inc. -- all rights reserved

What is API-First?

"API-first design means identifying and/or defining key actors and personas, determining what those actors and personas expect to be able to do with APIs"

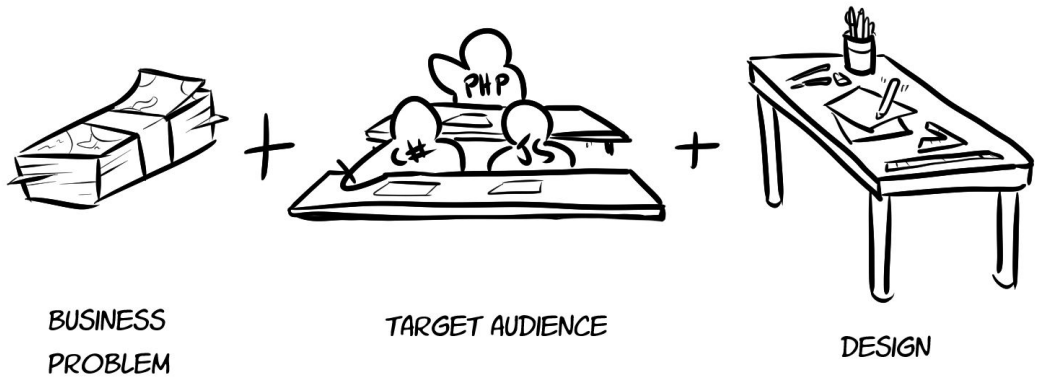
-- Kas Thomas, 2009

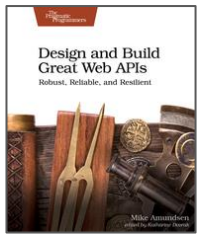


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Foundations

- API First
  - APIs Solve Business Problems
  - Designing APIs for People
  - Design First
- HTTP, Web, & REST

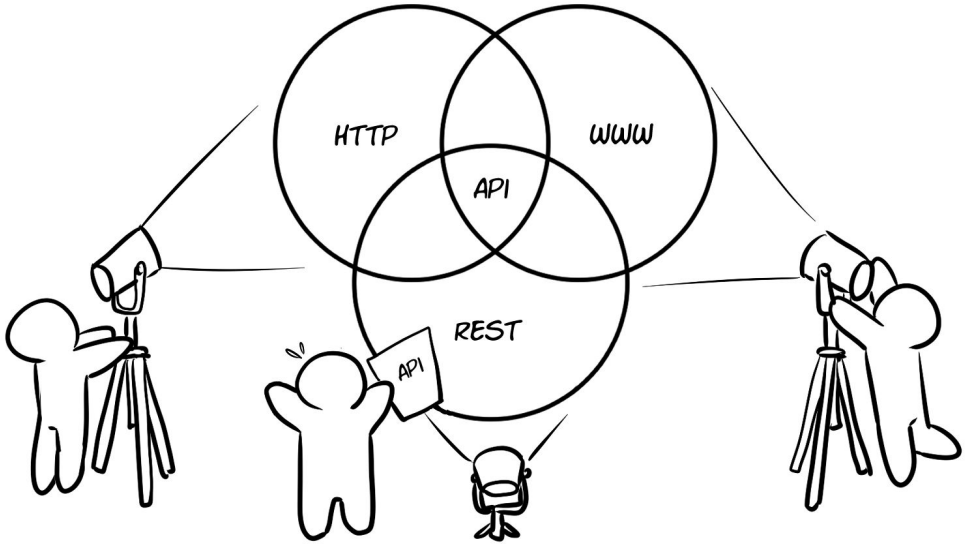


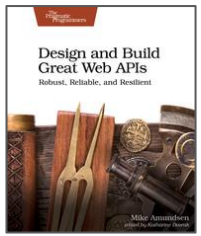


copyright © 202

Foundations

- API First
- HTTP, Web, & REST
  - HTTP is a protocol
    - URLs, Methods, Messages
  - Web is a common practice
    - SoC, links & forms
  - REST is a style
    - properties, requirements, constraints

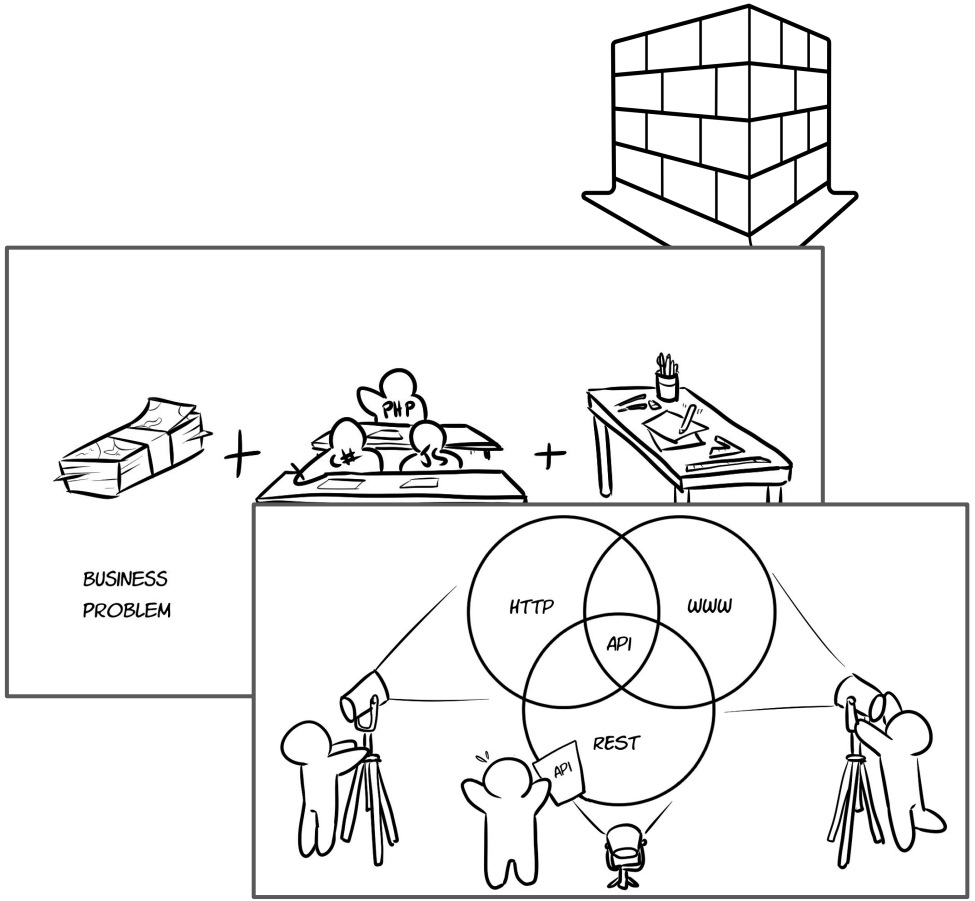


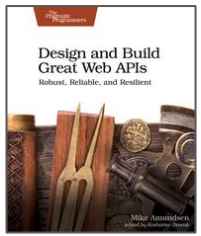


copyright © 2020 by amundsen.com, inc. -- all rights reserved

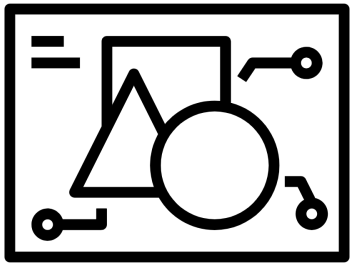
Foundations

- API First
  - Solving business problems for people
- HTTP, Web, & REST
  - Protocol, practice, style





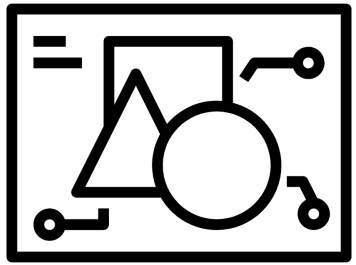
copyright © 2020 by amundsen.com, inc. -- all rights reserved



Designing APIs

copyright © 2020 by amundsen.com, inc. -- all rights reserved

Designing APIs

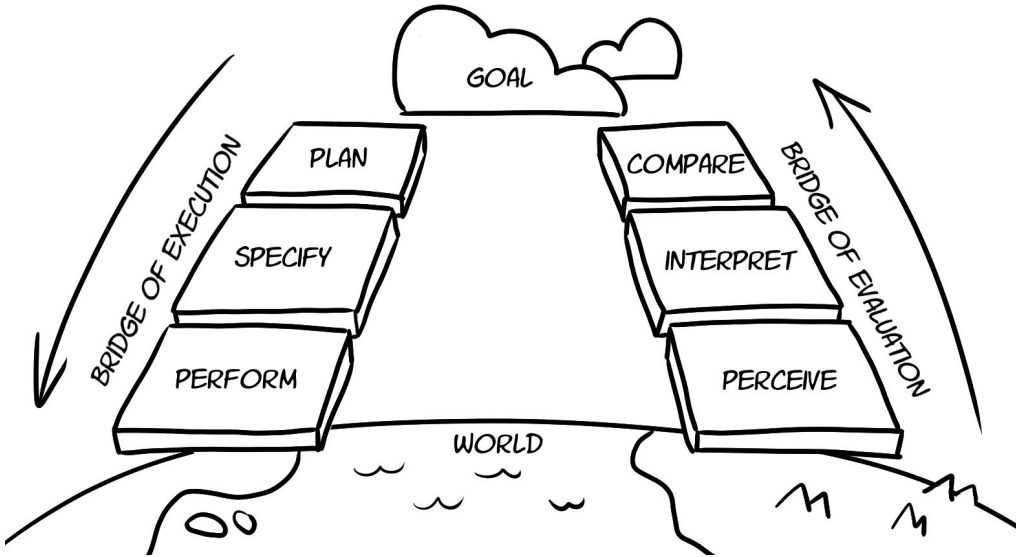


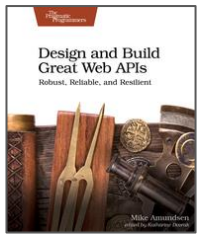
- Model
- Design
- Describe

copyright © 2020 by amundsen.com, inc. -- all rights reserved

API Design

- Model
  - Norman's Lifecycle
  - There are no straight lines
  - The API Story
- Design
- Describe





copyright © 2020 by amundsen.com, inc. -- all rights reserved

13 lines (9 sloc) | 1.09 KB

Raw Blame History

Company Story at BigCo, Inc.

Purpose

We keep track of companies for BigCo, Inc.

Data

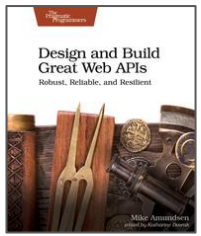
Data we include in a company record includes company name, street address, city, state/province, postal code, country, telephone, and email. Each record has a status value (pending, suspended, active, closed). We also track the date/time the record was created and the last date/time it was updated. We keep copies of the records, even after they have been deleted.

Actions

Typical work on the company records include getting the list of company records, reading a single record, creating, updating, and deleting records. You can also update the status of a single record. Finally, you can get a filtered list of all redords (support for filtering by status, by country, by state, and by company name).

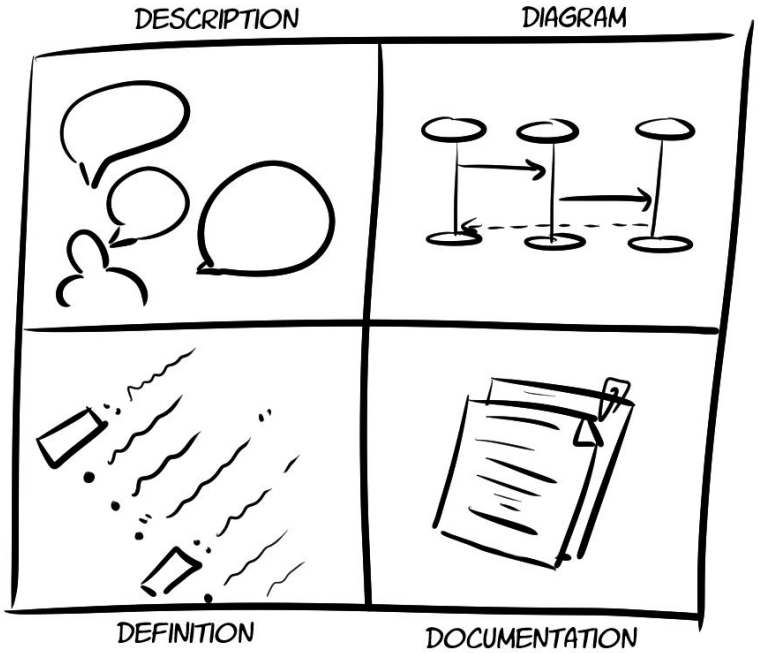
Processing

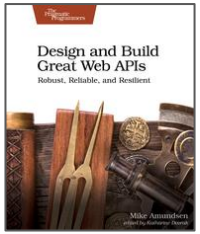
Each company has a unique identifier in the system. Right now that is a combination of the first four letters of their company name and date the company was added to the system (in the format YYYYMMDD. We add another digit if adding that does not result in a unique identifier in the system.



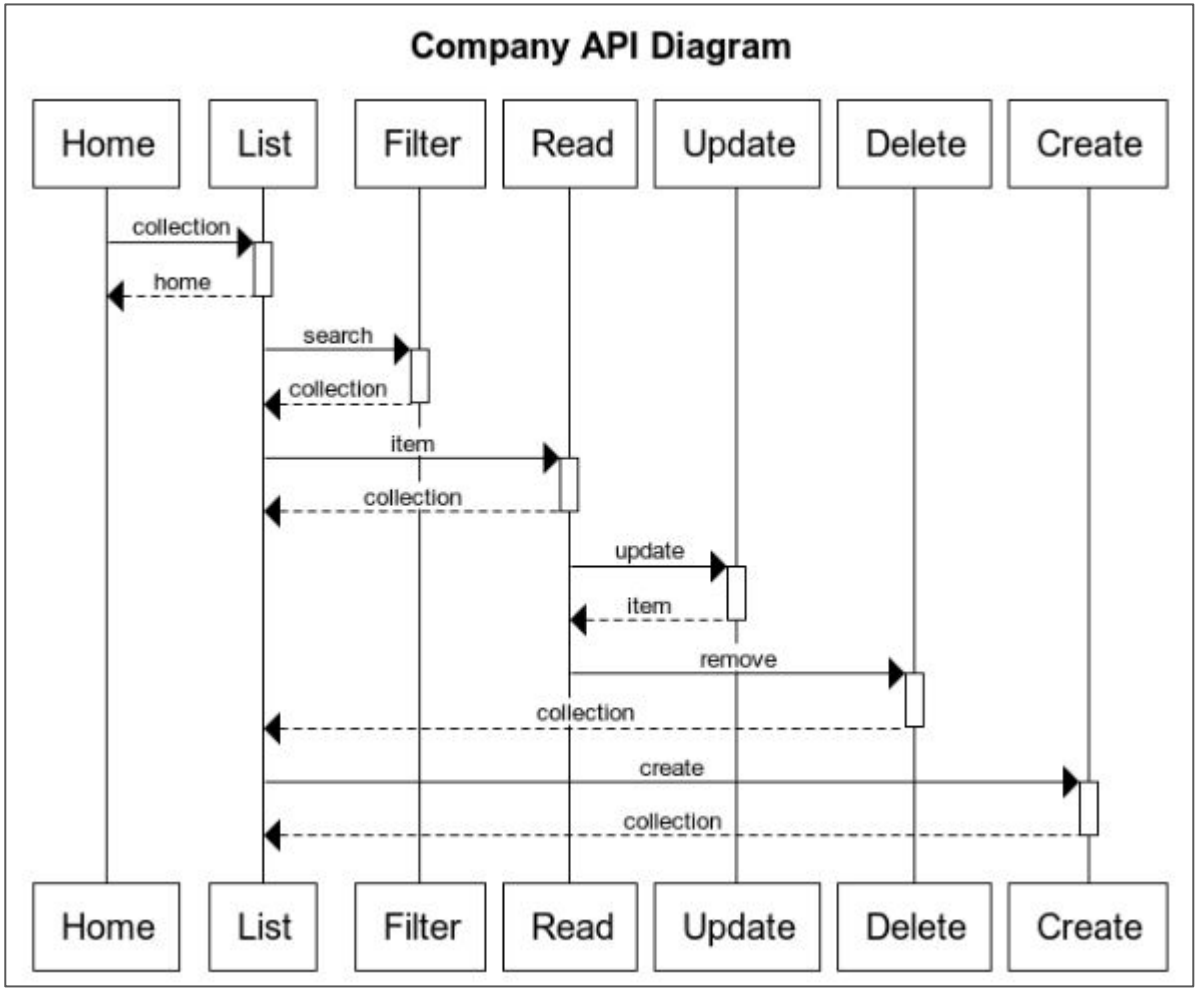
API Design

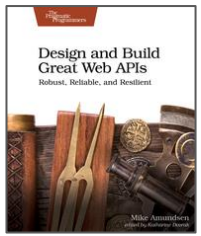
- Model
- Design
  - Design Thinking
  - Jobs-to-be-Done
  - The API Diagram
- Describe





copyright © 2020 by amundsen.com, inc. -- all rights reserved

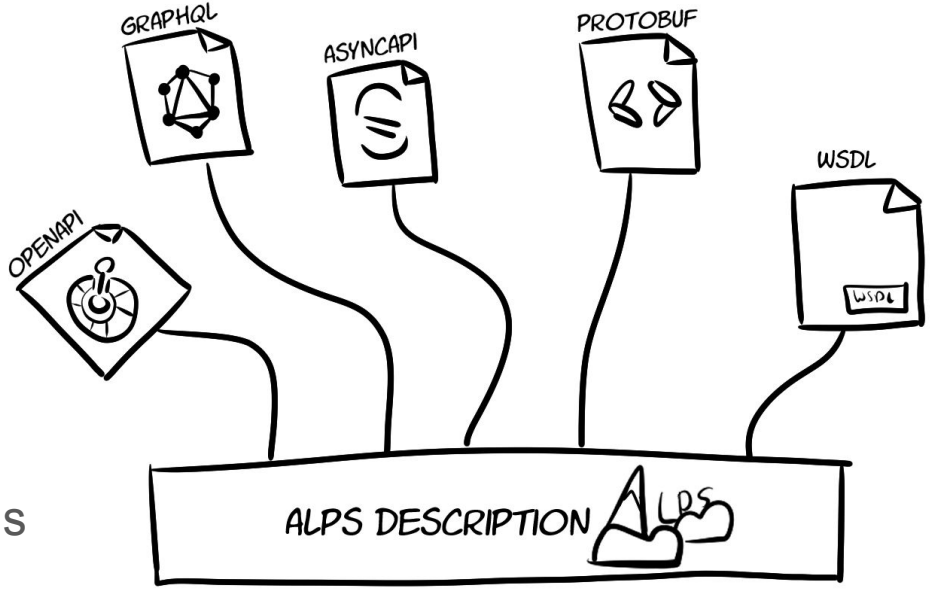


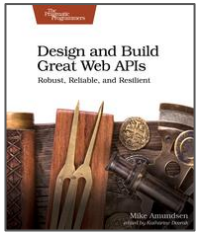


copyright © 2020 by amundsen.com, inc. -- all rights reserved

API Design

- Model
- Design
- Describe
  - Technology agnostic
  - Details on properties & actions
  - ALPS (2014)
  - `https://github.com/mamund/2020-04-unified-api-design`





copyright © 2020 by amundsen.com, inc. -- all rights reserved

91 lines (82 sloc) | 2 KB

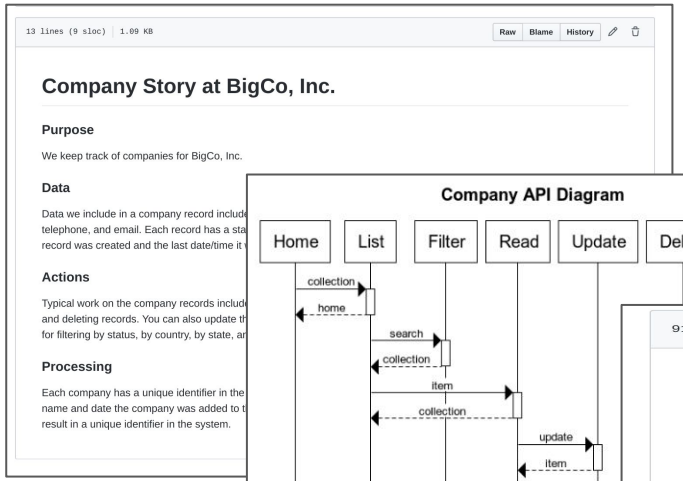
Raw Blame History

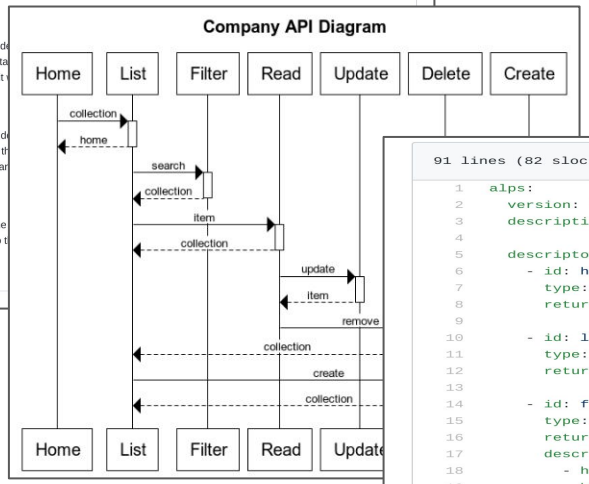
```yaml
alps:
  version: '1.0'
  description: ALPS document for BigCo Company API

  descriptors:
    - id: home
      type: safe
      returns: '#company'

    - id: listCompanies
      type: safe
      returns: '#company'

    - id: filterCompanies
      type: safe
      returns: '#company'
      descriptors:
        - href: '#companyName'
        - href: '#country'
        - href: '#status'
        - href: '#stateProvince'

    - id: readCompany
      type: safe
      returns: '#company'
      descriptors:
        - href: '#id'
```

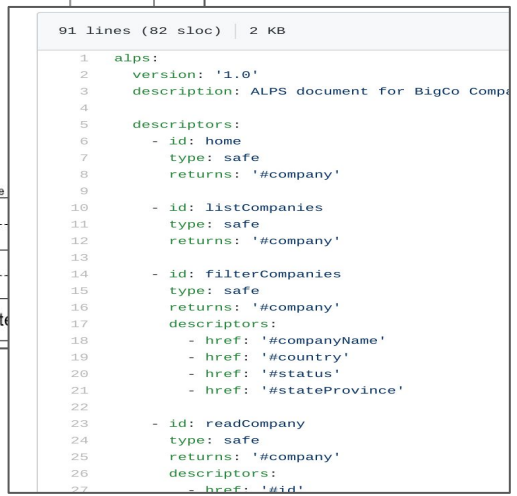
```yaml
    - id: company
      type: semantic
      descriptors:
        - id: id
          type: semantic
        - id: status
          type: semantic
          text: 'suspended, pending, active, closed'
        - id: companyName
          type: semantic
        - id: streetAddress
          type: semantic
        - id: city
          type: semantic
        - id: stateProvince
          type: semantic
        - id: postalCode
          type: semantic
        - id: country
          type: semantic
        - id: telephone
          type: semantic
        - id: email
          type: semantic
```

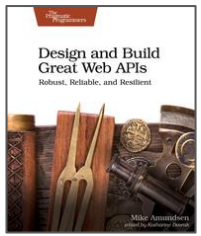
Designing APIs

- Model
  - API Story
- Design
  - API Diagram
- Describe
  - API Description

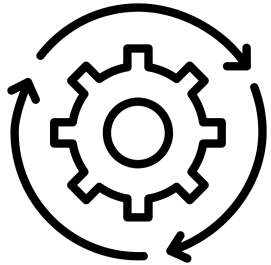








copyright © 2020 by amundsen.com, inc. -- all rights reserved

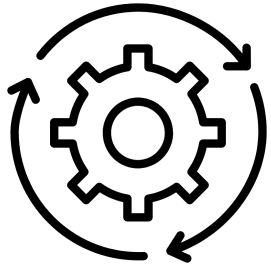


Building APIs

copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs

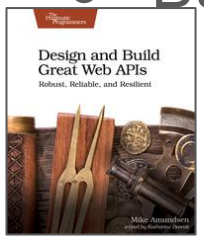
- Sketching
- Prototyping
- Building



copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs

- Sketching
  - Frank Gehry (via Ronnie Mitra)
  - Experiment w/ possibilities
  - Apiary Blueprint (APIB)
  - Sketches are meant to be thrown away
- Prototyping
- Building



copyright © 2020 by amundsen.com, inc. -- all rights reserved

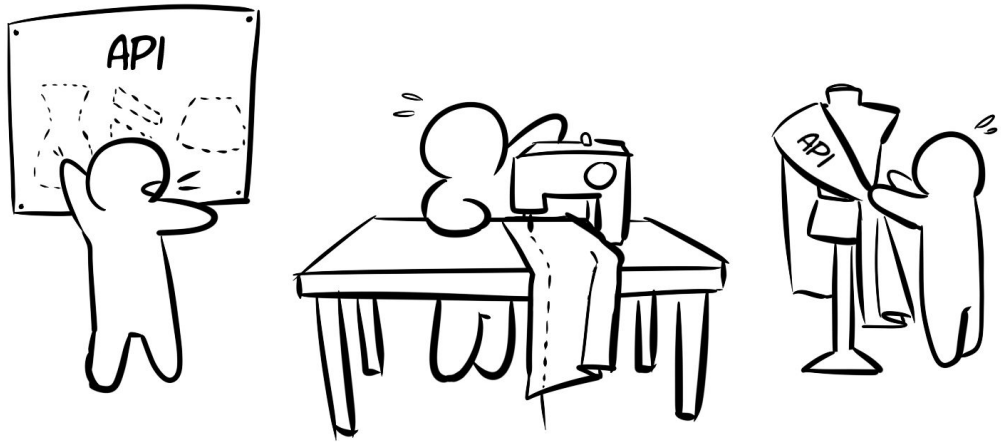
63 lines (50 sloc) | 1.51 KB

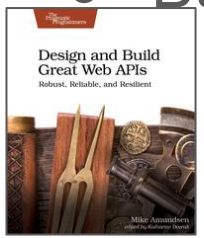
Raw Blame History

```
FORMAT: 1A
HOST: http://polls.apiblueprint.org/

# Credit Check Responses

Public API profile for BigCo's Credit Check service.

## Credit Check History [/list]

Returns a list of past credit rating records.

### List Past Credit Checks [GET]

+ Response 200 (application/json)

        {
          "creditCheck" : [
            {
              "id" : "123",
              "companyName" : "Vandelay Industries",
              "dateRequested" : "2009-06-15T13:45:30",
              "ratingValue" : "5"
            },
            {
              "id" : "123",
              "companyName" : "Vandelay Industries",
              "id" : "789",
              "companyName" : "Vandelay Industries",
              "dateRequested" : "2009-06-21T08:35:15",
              "ratingValue" : "6"
            }
          ]
        }

## Credit Check Item [/list/123]

Returns a list of past credit rating records.

### Single Credit Checks [GET]

+ Response 200 (application/json)

        {
          "creditCheck" : [
            {
              "id" : "123",
              "companyName" : "Vandelay Industries",
              "dateRequested" : "2009-06-15T13:45:30",
              "ratingValue" : "5"
            }
          ]
```

Building APIs

- Sketching
- Prototyping
  - Garment industry toile
  - Explore the details
  - OpenAPI (OAS)
  - Prototypes are made to be tested
- Building





copyright © 2020 by amundsen.com, inc. -- all rights reserved

284 lines (248 sloc) | 6.63 KB

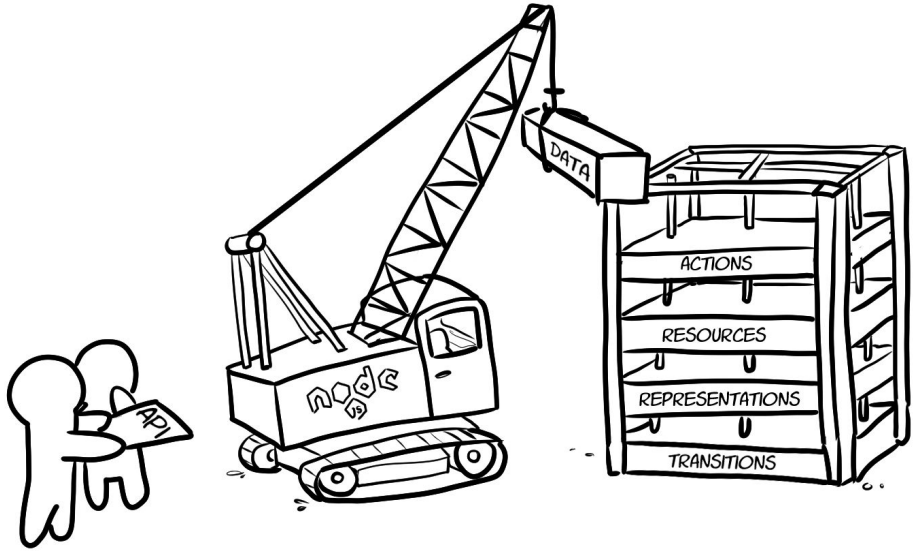
Raw Blame History

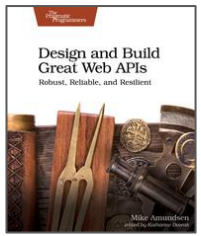
```yaml
openapi: 3.0.0

########################################
# Credit Check API
# 2019-12-22
# mamund
########################################

### info section ###
info:
  version: '1.0.0'
  title: 'Credit Check API'
  description: 'Credit Check service API for BigCo, Inc.'

# Added by API Auto Mocking Plugin
servers:
  - description: SwaggerHub API Auto Mocking
    url: https://virtserver.swaggerhub.com/amundsen/credit-check-api/1

### tags ###
tags:
  - name: credit
    description: "Credit Check API"

### paths section ###
paths:
```

```yaml

  ### property models ###
  schemas:

    ### default error ###
    error:
      type: object
      properties:
        type:
          type: string
          example: "Invalid Record"
        title:
          type: string
          example: "One or more missing properties"
        detail:
          type: string
          example: "Review the submitted record for

    ### 200 OK reply (prag+json) ###
    reply:
      type: object
      properties:
        metadata:
          type: array
```

Building APIs

- Sketching
- Prototyping
- Building
  - Repeatable process
  - Convert prototype/design into code
  - DARRT
    - Data, Actions, Resources, Representations, Transitions



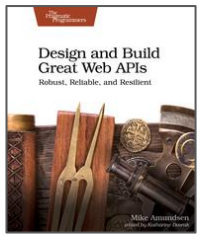


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs : DARRT

A simple, repeatable process for publishing API interfaces

- Data
- Actions
- Resources
- Representations
- Transitions

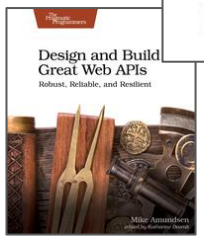


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs : DARRT : Data

- The state properties to pass in messages
  - `properties, requireds, enums, defaults`

```
// this service's message properties
exports.props = [
  'id','status','dateCreated','dateUpdated',

  'companyId','companyName','streetAddress','city','stateProvince',
  'postalCode','country','telephone','email',

  'accountId','division','spendingLimit','discountPercentage',

  'activityId','activityType','dateScheduled','notes'
];
```



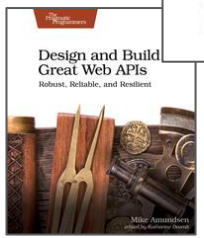
copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs : DARRT : Data

- The state properties to pass in messages
  - `properties, requireds, enums, defaults`

```
// this service's message properties

// required properties
exports.reqd = ['id','status'];
```

```
reated','dateUpdated',
    'companyId','companyName','streetAddress','city','stateProvince',
    'postalCode','country','telephone','email',
    'accountId','division','spendingLimit','discountPercentage',
    'activityId','activityType','dateScheduled','notes'
];
```



copyright © 2020 by amundsen.com, inc. -- all rights reserved

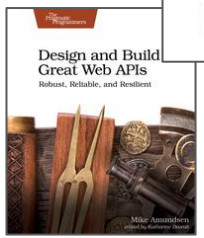
Building APIs : DARRT : Data

- The state properties to pass in messages
  - `properties, requireds, enums, defaults`

```
// this service's message properties
                                    reated','dateUpdated',
    'companyId',
    'postalCode'
    'accountId',
    'activityId'
];
```

```
// required properties
exports.reqd = ['id','status'];
```

```
// enumerated properties
exports.enums = [
  {status:
    ['pending','active','suspended','closed']
  },
  {division:
    ['DryGoods','Hardware','Software','Grocery','Pharmacy','Military']
  },
  {activityType:
    ['email','inperson','phone','letter']
  }
];
```



copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs : DARRT : Data

- The state properties to pass in messages
  - `properties, requireds, enums, defs`

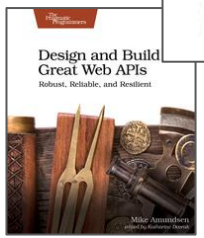
```
// this service's message properties
```

```
// required properties
exports.reqd = ['id','status'];
```

```
'companyId',
'postalCode'
'accountId',
'activityId'
];
```

```
// enumerated properties
exports.enums = [
  {status:
    ['pending','active','suspended',
  },
  {division:
    ['DryGoods','Hardware','Software','Grocery','Pharmacy','Military']
  },
  {activityType:
    ['email','inperson','phone','letter']
  }
];
```

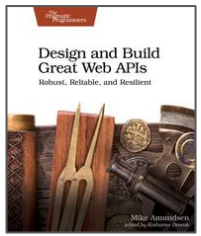
```
{name:"spendingLimit", value:"10000"},
{name:"discountPercentage", value:"10"},
{name:"activityType", value:"email"},
{name:"status",value:"pending"}
];
```



copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs : DARRT : Actions

- The actual internal operations for the interface
  - `approvePayroll, updateCustomer, setStatus`



building/action-readStatus.js

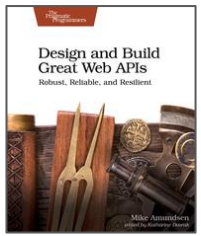
```
module.exports.readStatus = function(req,res) {
  return new Promise(function(resolve,reject){
    if(req.params.id && req.params.id!==null) {
      var id = req.params.id;
      var fields="id, status, dateCreated, dateUpdated"
      resolve(
        component(
          {name:'onboarding',action:'item',id:id, fields:fields}
        )
      );
    }
    else {
      reject({error:"missing id"});
    }
  });
}
```

Building APIs : DARRT : Resources

- The public HTTP resources to access the operations

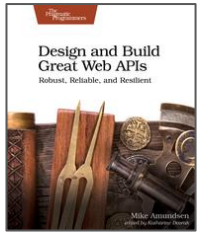
building/resource-list.js

```
// ************************************************************
// public resources for the onboarding service
// ************************************************************

router.get('/',function(req,res){ });
router.post('/wip/', function(req,res){ });
router.get('/wip/',function(req,res){ });
router.get('/wip/filter/', function(req,res){ });
router.get('/wip/:id', function(req,res){ });
router.get('/wip/:id/company', function(req,res){ });
router.put('/wip/:id/company', function(req,res){ });
router.get('/wip/:id/account', function(req,res){ });
router.put('/wip/:id/account', function(req,res){ });
router.get('/wip/:id/activity', function(req,res){ });
router.put('/wip/:id/activity', function(req,res){ });
router.get('/wip/:id/status', function(req,res){ });
router.put('/wip/:id/status', function(req,res){ });
```



Building APIs : DARRT : Representations

- Produce the requested format for resource responses



building/app-json-template.js

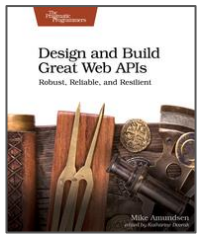
```
// plain JSON rerpresentor template
exports.template =
{
  format:"application/json",
  view:
  `
  {
    "<%=type%>":
    [
     <%var x=0;%>
     <%rtn.forEach(function(item){%>
      <%if(x!==0){%>,<%}%>
      {
        <%var y=0;%>
        <%for(var p in item){%>
         <%if(y!==0){%>,<%}%>
          "<%=p%>":"<%=helpers.stateValue(item[p],item,request,item[p])%>"
         <%y=1;%>
        <%}%>
      }
      <%x=1;%>
    <%});%>
```

Building APIs : DARRT : Transitions

- The list of public actions as expressed in HTTP

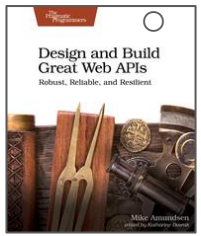
building/add-account-transition.js

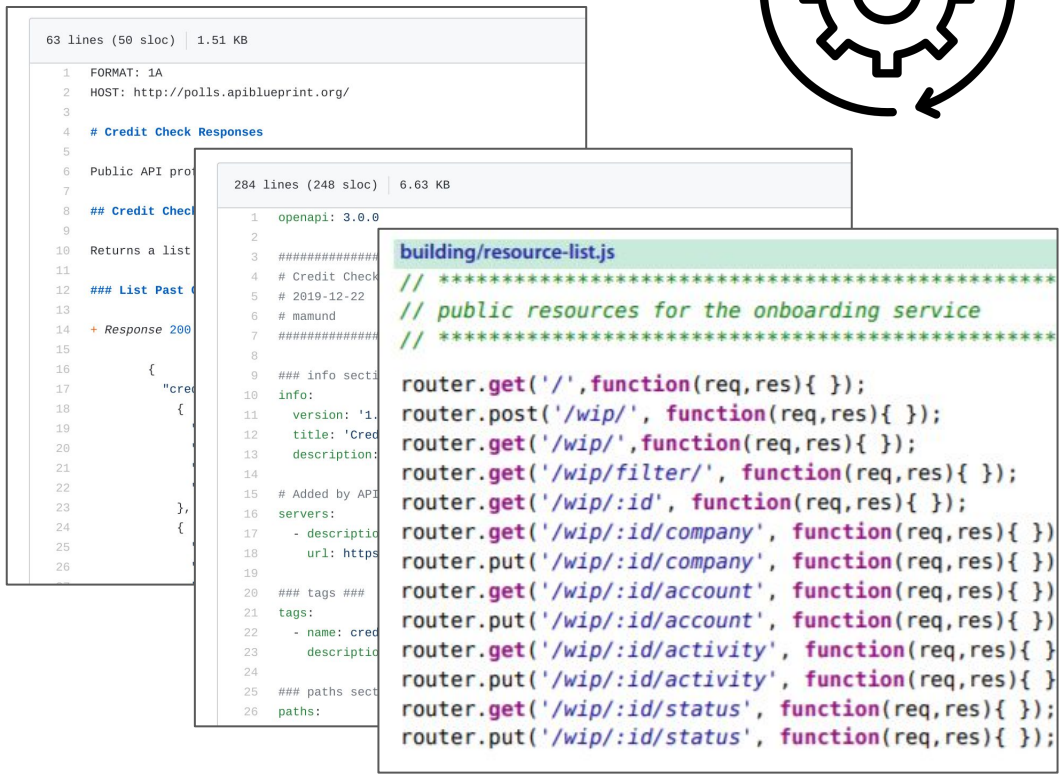
```
{
  id:"addAccount_{id}",
  name:"addAccount",
  href:"{fullhost}/wip/{id}/account",
  rel: "item edit-form onboarding",
  tags: "onboarding list item",
  title: "Add Account",
  method: "PUT",
  properties: [
    {name:"accountId",value:"{accountId}"},
    {name:"division",value:"{division}"},
    {name:"spendingLimit",value:"{spendingLimit}"},
    {name:"discountPercentage",value:"{discountPercentage}"}
  ]
}
```



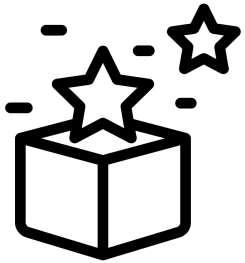
Building APIs

- Sketching
  - Experiment, throw away
  - APIB
- Prototyping
  - Explore, test
  - OpenAPI
- Building
  - Translate, repeat
  - NodeJS/DARRT





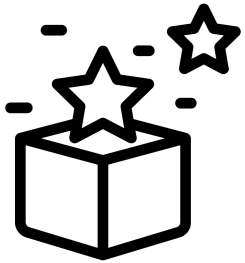
copyright © 2020 by amundsen.com, inc. -- all rights reserved



Releasing APIs

copyright © 2020 by amundsen.com, inc. -- all rights reserved

Releasing APIs

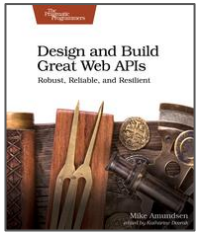


- Testing
- Security
- Deployment

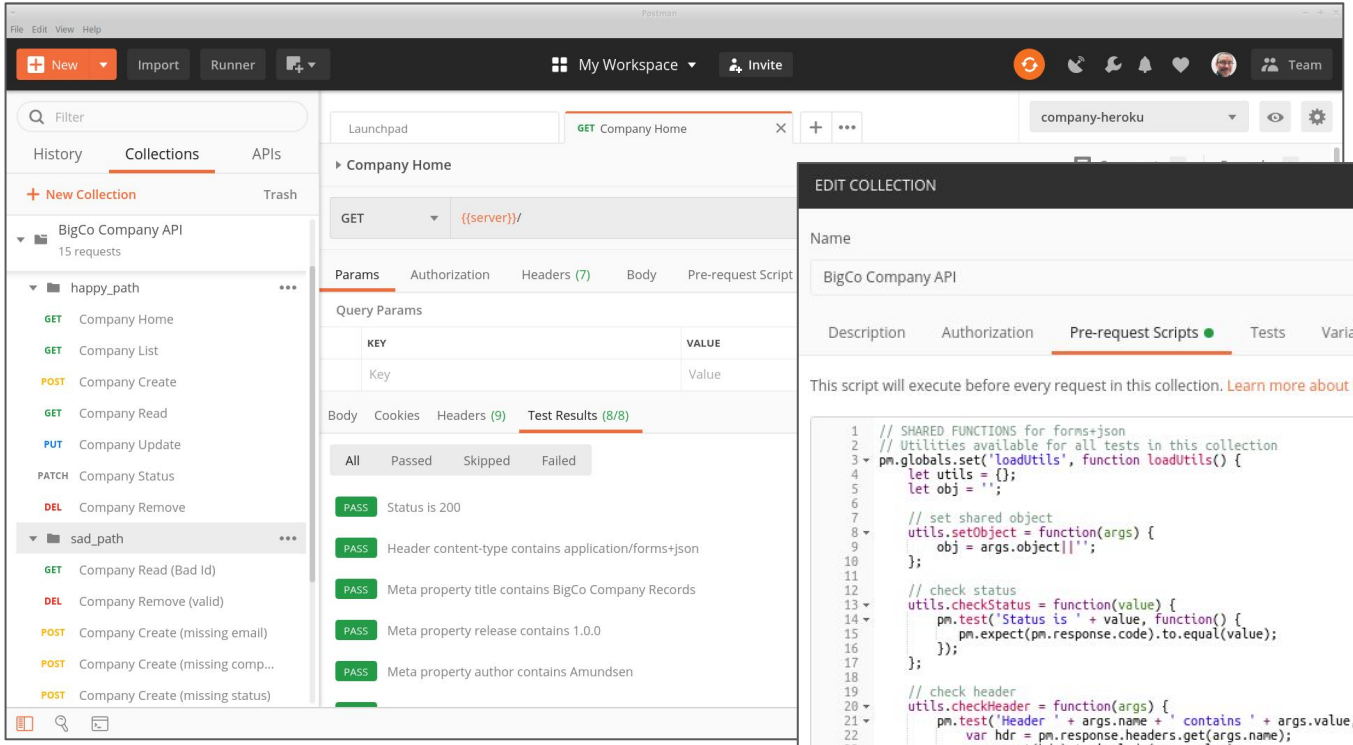
copyright © 2020 by amundsen.com, inc. -- all rights reserved

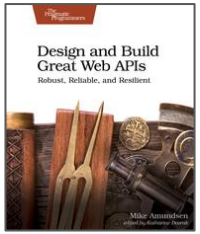
Releasing APIs

- Testing
  - Testing Interface, not Code
  - Happy-path (200) & sad-path (400)
  - Simple Request Tests (SRTs)
  - Postman/Newman for BDD
- Security
- Deployment



copyright © 2020 by amundsen.com, inc. -- all rights reserved





EDIT COLLECTION

Name

BigCo Company API

Description | Authorization | Pre-request Scripts | Tests | Variables

This script will execute before every request in this collection. Learn more about Postman's execution order.

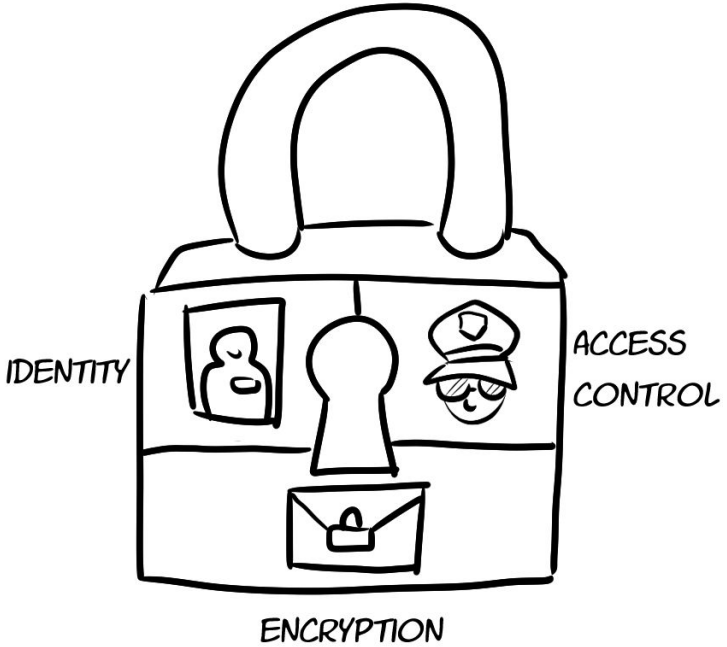
```
// SHARED FUNCTIONS for forms+json
// Utilities available for all tests in this collection
pm.globals.set('loadUtils', function loadUtils() {
    let utils = {};
    let obj = '';

    // set shared object
    utils.setObject = function(args) {
        obj = args.object||'';
    };

    // check status
    utils.checkStatus = function(value) {
        pm.test('Status is ' + value, function() {
            pm.expect(pm.response.code).to.equal(value);
        });
    };

    // check header
    utils.checkHeader = function(args) {
        pm.test('Header ' + args.name + ' contains ' + args.value, function() {
            var hdr = pm.response.headers.get(args.name);
            pm.expect(hdr).to.include(args.value);
        });
    };

    // check metadata
    utils.checkMetaProperty = function(args) {
        pm.test('Meta property ' + args.name + ' contains ' + args.value, function() {
            var meta = body[obj].metadata.find( x => x.name === args.name);
            pm.expect(meta.value).to.include(args.value);
        });
    };
```

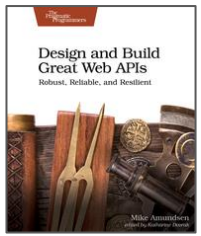
Cancel | Update

copyright © 2020 by amu

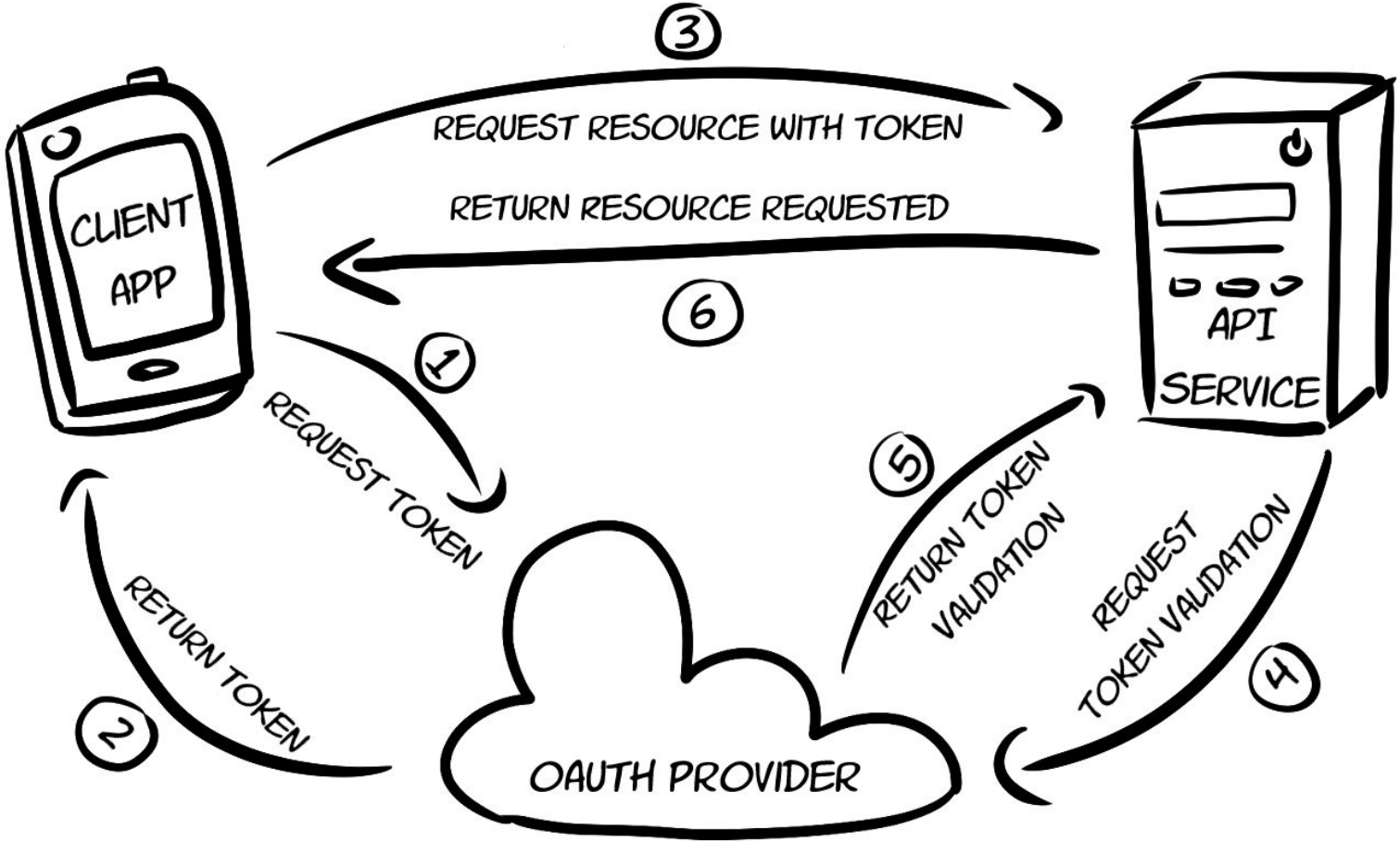
Releasing APIs

- Testing
- Security
  - Encryption, Identity, Access Control
  - M2M Authorization w/ JWTs
  - Auth0 as provider
- Deployment



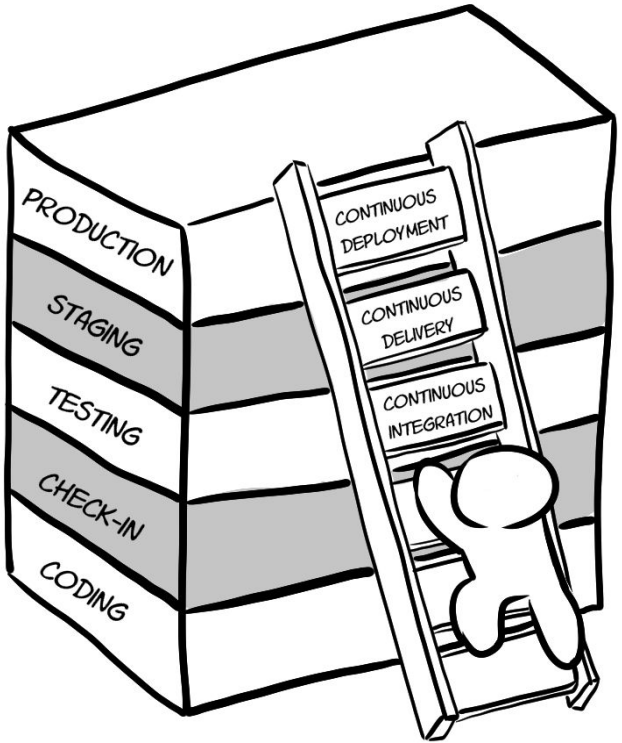


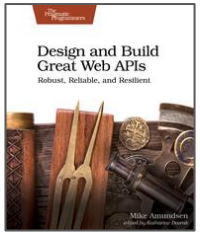
copyright © 2020 by amundsen.com, inc. -- all rights reserved


3
REQUEST RESOURCE WITH TOKEN
RETURN RESOURCE REQUESTED
6
CLIENT APP
API SERVICE
1
REQUEST TOKEN
2
RETURN TOKEN
5
RETURN TOKEN VALIDATION
REQUEST TOKEN VALIDATION
4
OAUTH PROVIDER

Releasing APIs

- Testing
- Security
- Deployment
  - Integration, delivery, deployment
  - Automation improves safety
  - Heroku via git push



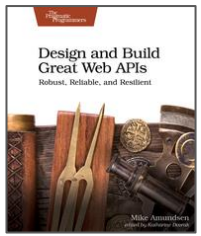


copyright © 2020 by amundsen.com, inc. -- all rights reserved

```
remote: -----> Build succeeded!
remote:        ├── ejs@2.3.3
remote:        └── express@4.13.3
remote:
remote:
remote: -----> Discovering process types
remote:        Procfile declares types -> w
remote:
remote: -----> Compressing...
remote:        Done: 10.2M
remote: -----> Launching...
remote:        Released v4
remote:        https://tranquil-hollows-728
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/tranquil-hollows-
   763c36e..f555ce0  master -> master
~/Code/tranquil-hollows-72848
```

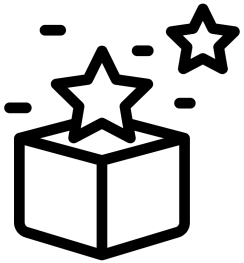
```
Running pre-deployment tests...

BigCo Company Tests
==================================================
Wed May 27 19:55:24 EDT 2020
Running local environment...
Initializing...
rm: cannot remove 'newman/*': No such file or dire
Pulling postman data...

> company@1.0.0 dev /home/mca/projects/api-tool-ki
> nodemon index

[nodemon] 2.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching dir(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index index.js`
Running tests...
listening on port 8484!
Time:  1590623727242 : localhost:8484/ : GET : {}
Time:  1590623727583 : localhost:8484/list/ : GET
Time:  1590623727961 : localhost:8484/ : POST : {"
om","telephone":"123-456-7890","status":"pending"}
```

```
Deploying to production...
Everything up-to-date
Running post-deployment tests...

BigCo Company Tests
==================================================
Wed May 27 19:55:33 EDT 2020
Running remote environment...
Initializing...
Pulling postman data...
[nodemon] restarting due to changes...
[nodemon] starting `node index index.js`
listening on port 8484!
[nodemon] restarting due to changes...
Running tests...
[nodemon] starting `node index index.js`
listening on port 8484!
One or more tests failed!

Test run completed and saved to test-output.txt.

*** POST-DEPLOY TESTS FAILED - job cancelled. ***

mca@penguin:~/projects/api-tool-kit/company$
```

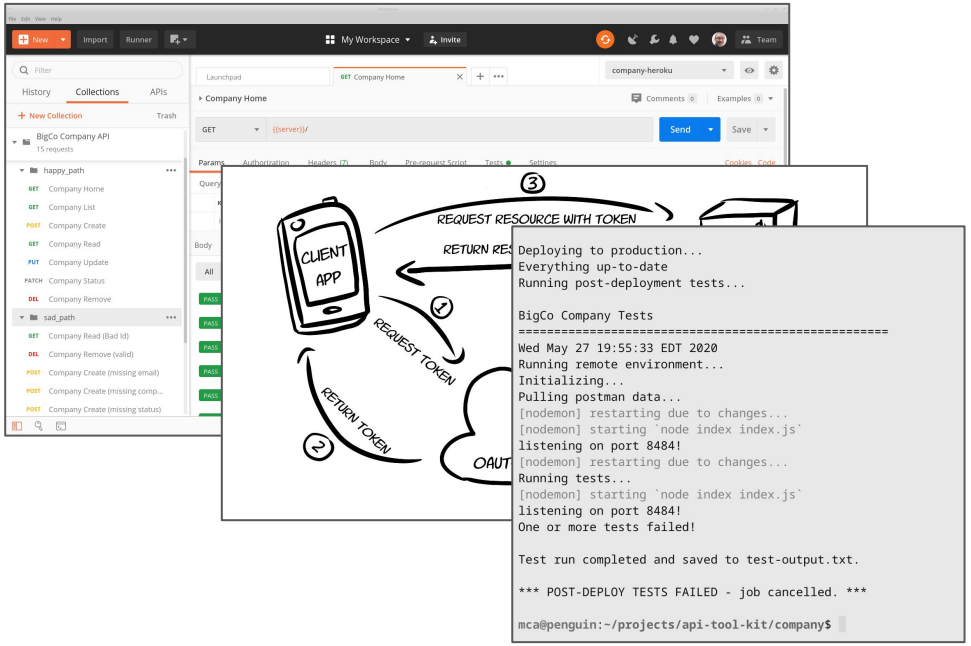


copyright © 2020 by amundsen.com, inc. -- all rights reserved

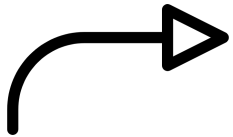
Releasing APIs



- Testing
  - Test interface and behavior
  - Postman/Newman
- Security
  - APIs use client_credentials w/ JWTs
  - Auth0 as a provider
- Deployment
  - Automate for safety
  - Heroku git push



copyright © 2020 by amundsen.com, inc. -- all rights reserved

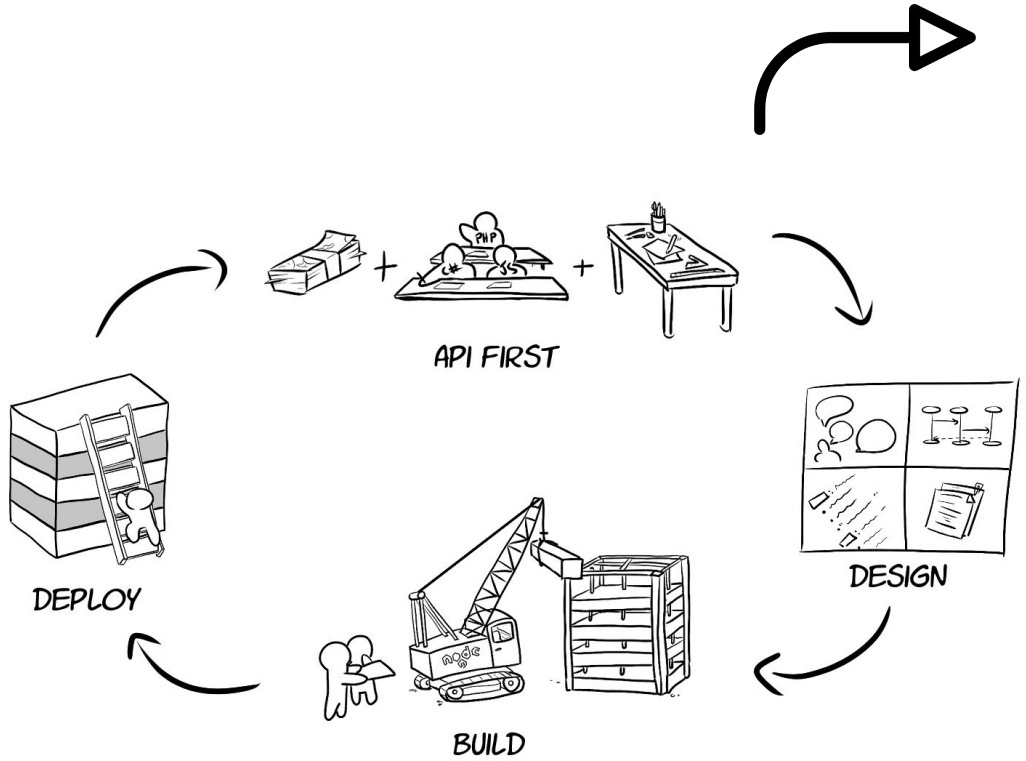


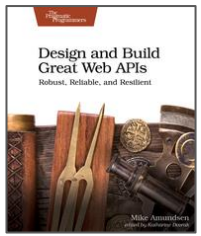
And Then...

copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

- Update Principles
- Design Updates
- Test Updates
- Release Updates
- API Shutdown

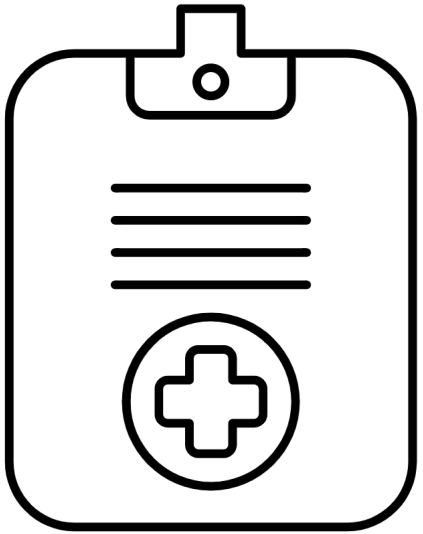


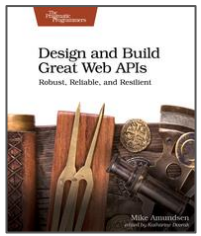


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

- Update Principles
  - First, do no harm
  - Fork your API
  - Know when to say "No"
- Design Updates
- Test Updates
- Release Updates
- API Shutdown

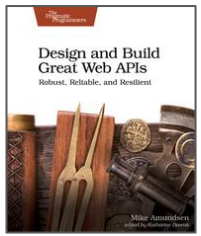




copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

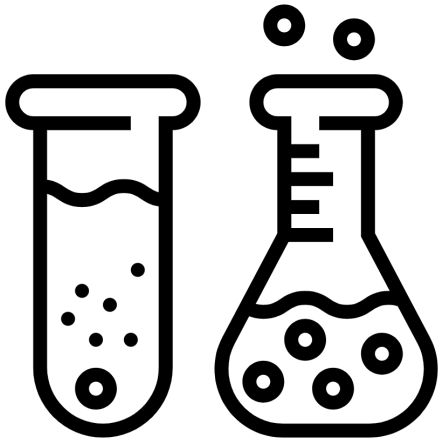
- Update Principles
- Design Updates
  - Take nothing away
  - Don't redefine
  - Additions are optional
- Test Updates
- Release Updates
- API Shutdown

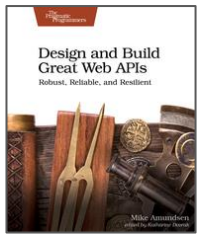


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

- Update Principles
- Design Updates
- Test Updates
  - Use all existing tests
  - Add new tests for each release
- Release Updates
- API Shutdown

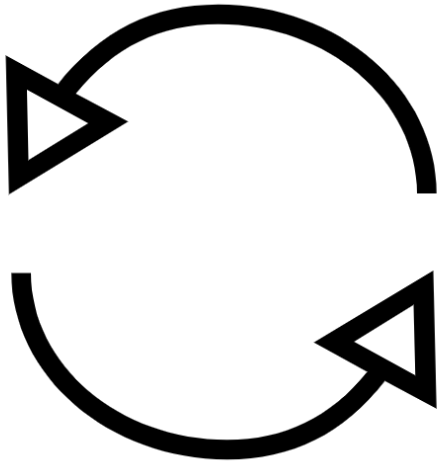


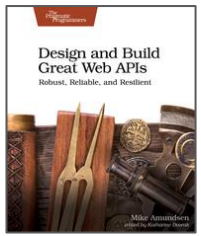


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

- Update Principles
- Design Updates
- Test Updates
- Release Updates
  - Reversibility/Re-entry First
  - Side-by-Side Releases
  - Overwriting Releases
- API Shutdown

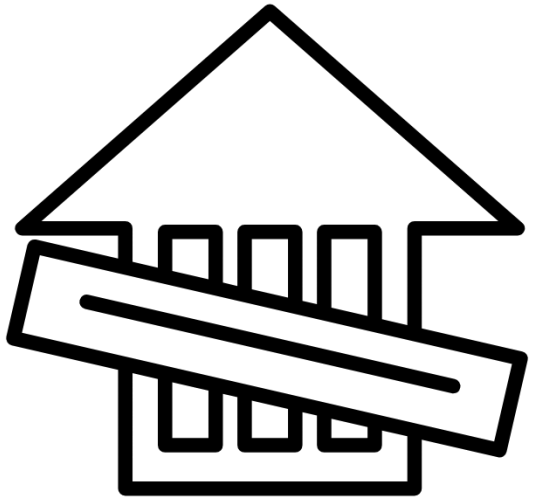


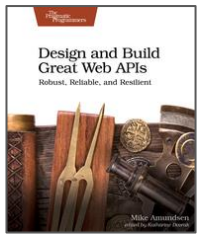


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

- Update Principles
- Design Updates
- Test Updates
- Release Updates
- API Shutdown
  - Place Code in Public Domain
  - Open Source the Interface
  - Recoverable Data
  - Mark the API 410 Gone (w/ a pointer)

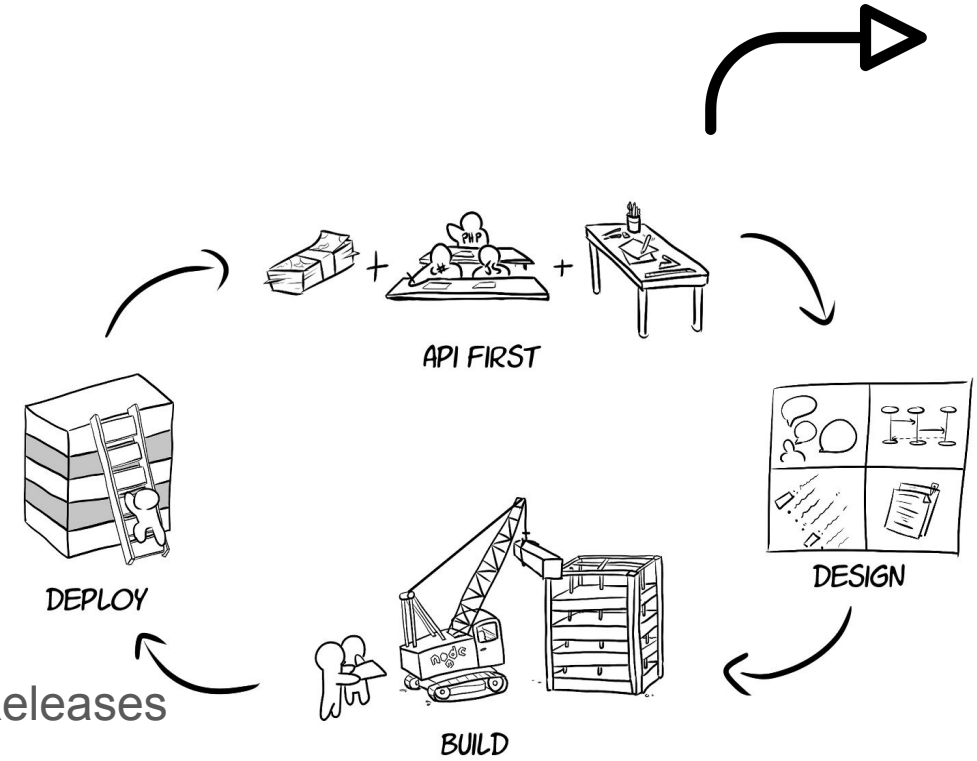


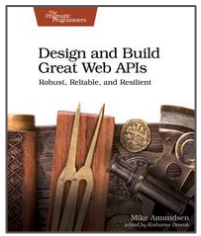


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

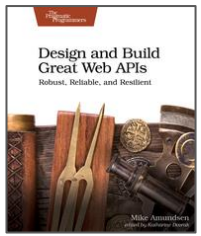
- Update Principles
  - First, Do no harm
- Design Updates
  - No breaking changes
- Test Updates
  - Use all the old tests
- Release Updates
  - Favor Side-by-Side Releases
- API Shutdown
  - Responsible shutdowns





copyright © 2020 by amundsen.com, inc. -- all rights reserved

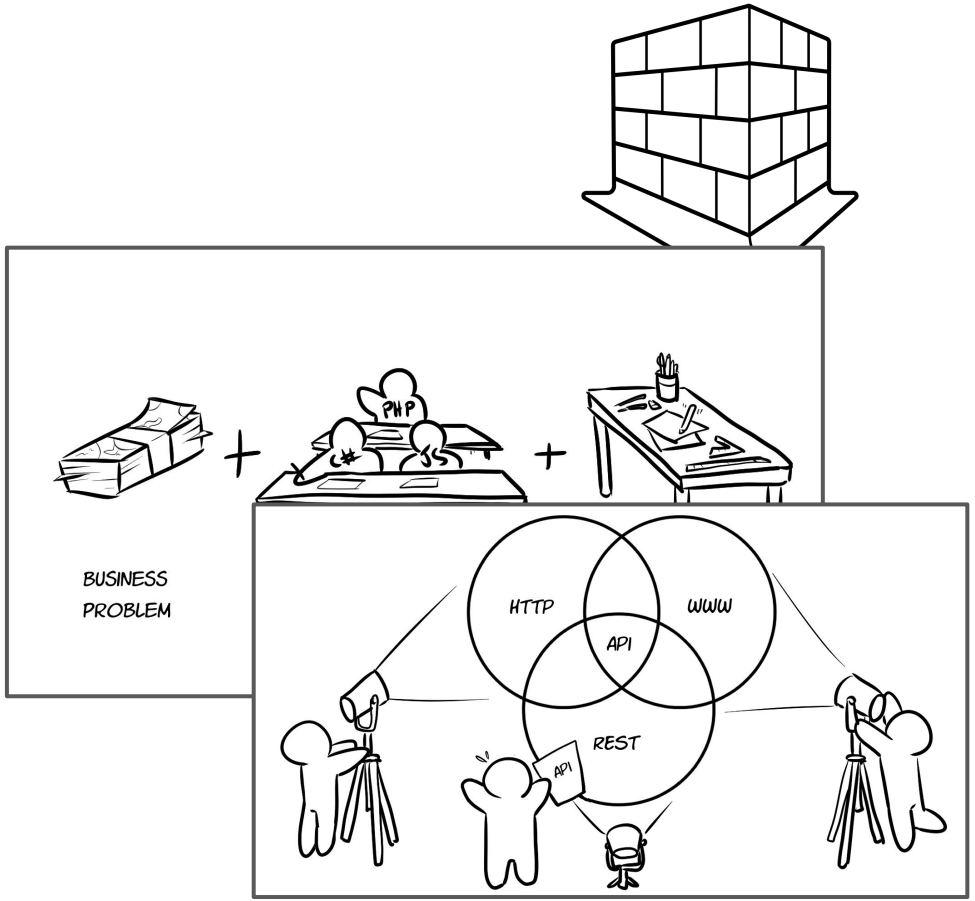
So....

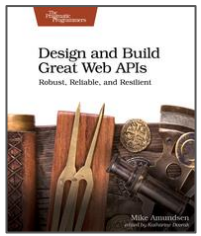


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Foundations

- API First
  - Solving business problems for people
- HTTP, Web, & REST
  - Protocol, practice, style

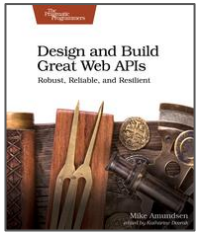


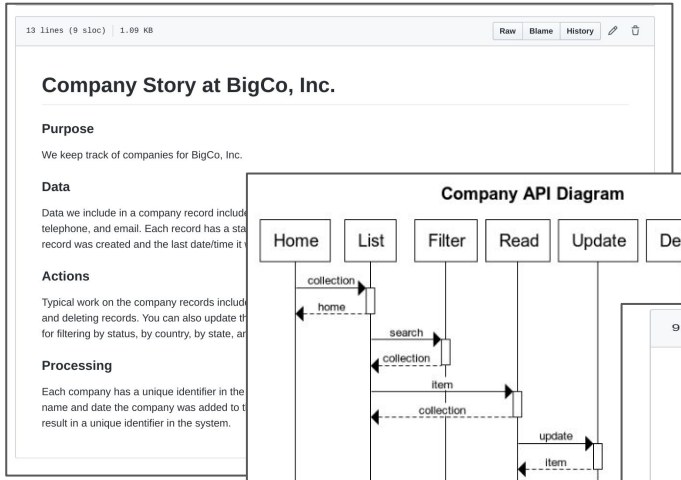


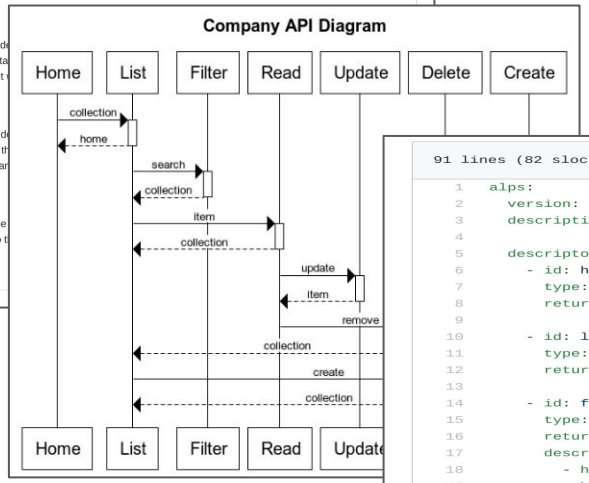
copyright © 2020 by amundsen.com, inc. -- all rights reserved

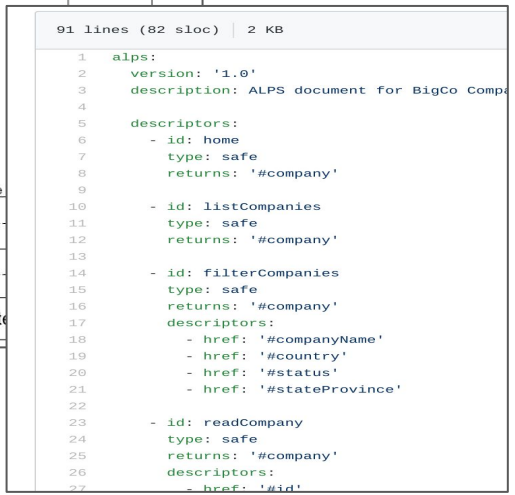
Designing APIs

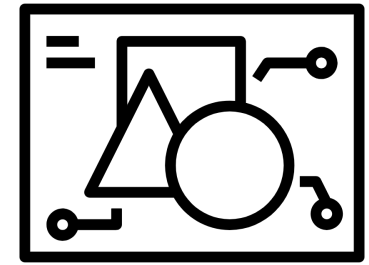
- Model
  - API Story
- Design
  - API Diagram
- Describe
  - API Description







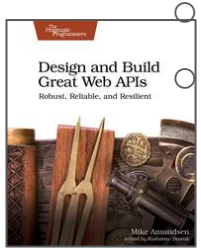


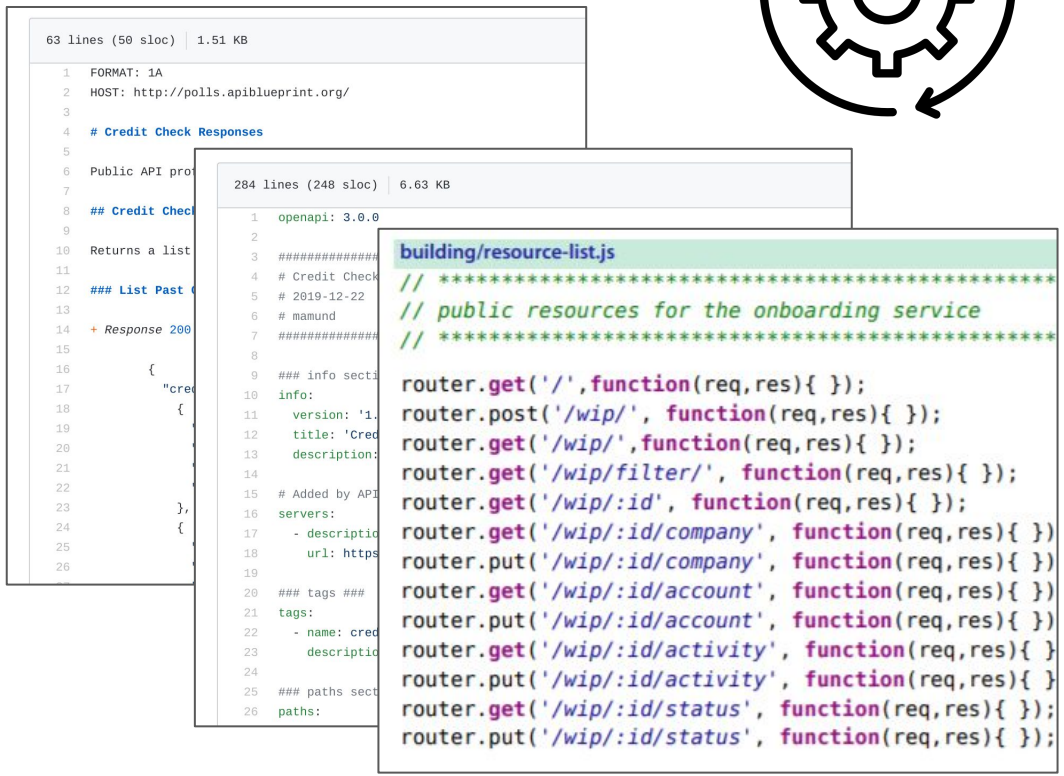


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Building APIs

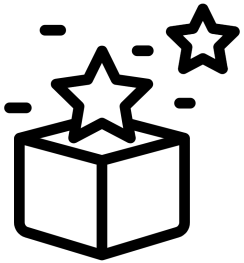
- Sketching
  - Explore, throw away
  - APIB
- Prototyping
  - Details, test
  - OpenAPI
- Building
  - Translate, repeat
  - NodeJS/DARRT



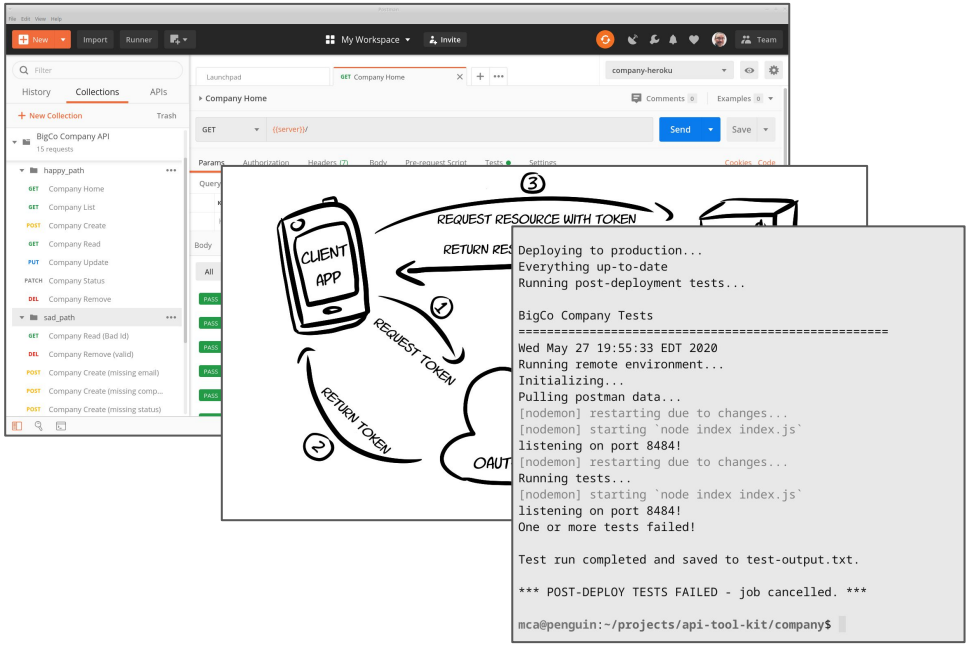


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Releasing APIs



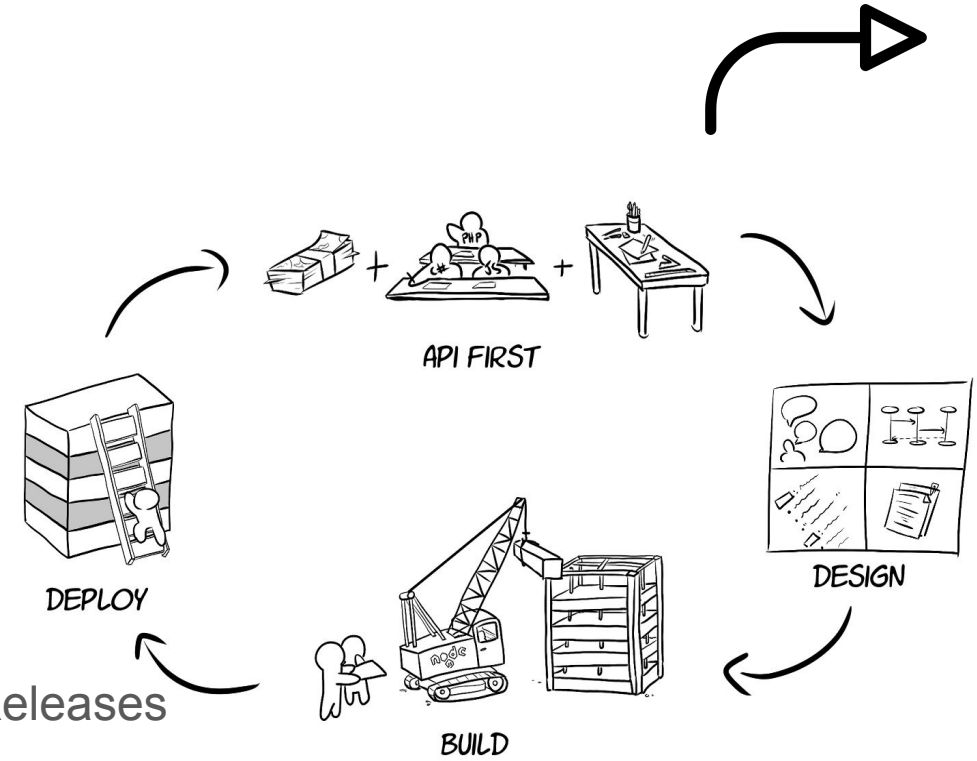
- Testing
  - Test interface and behavior
  - Postman/Newman
- Security
  - APIs use client_credentials w/ JWTs
  - Auth0 as a provider
- Deployment
  - Automate for safety
  - Heroku git push

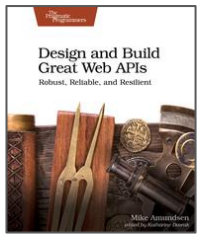


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Modifying APIs

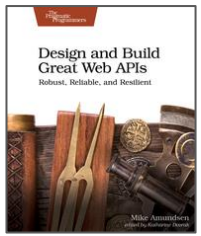
- Update Principles
  - First, Do no harm
- Design Updates
  - No breaking changes
- Test Updates
  - Use all the old tests
- Release Updates
  - Favor Side-by-Side Releases
- API Shutdown
  - Responsible shutdowns





copyright © 2020 by amundsen.com, inc. -- all rights reserved

And, don't forget ...

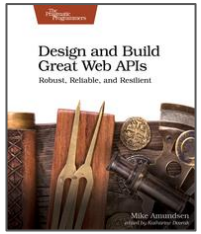


copyright © 2020 by amundsen.com, inc. -- all rights reserved

What is API-First?

*"API-first design means **identifying** and/or defining key actors and personas, **determining** what those actors and personas expect to be able **to do** with **APIs**"*

-- Kas Thomas, 2009

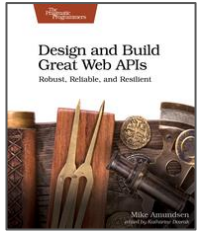


copyright © 2020 by amundsen.com, inc. -- all rights reserved

Design and Build Great APIs

http://g.mamund.com/2020-10-gids

@mamund
Mike Amundsen
youtube.com/mamund



copyright © 2020 by amundsen.com, inc. -- all rights reserved