



# COM5940: NEW MEDIA BUSINESS MODEL & INNOVATION ACHIEVE PRODUCT-MARKET FIT IN AGILE DEVELOPMENT

Bernard Suen  
Center for Entrepreneurship  
Chinese University of Hong Kong



Center for  
Entrepreneurship

# **Today's Agenda**

1. **Setup Python Flask and MySQL on PythonAnywhere.**
2. **Perform SQL CRUD operations using the Python Pymysql module and display results using the Python Pandas module.**
3. **From Business Model Canvas to Lean Canvas.**
4. **The importance of “early adopters” in Lean Method.**
5. **Introducing Google Design Sprint in the context of Agile Development.**
6. **9 business models for hypothesis testing in customer discovery.**

**Python Flask  
&  
SQL**

**Lean & Agile  
Development**

**9 Types of  
Business Model**

# **Uploading and Running Flask Codes on PythonAnywhere**



# Host, run, and code Python in the cloud!

Get started for free. Our basic plan gives you access to machines with a full Python environment already installed. You can develop and host your website or any other code directly from your browser without having to install software or manage your own server.

Need more power? Upgraded plans start at \$5/month.

[Start running Python online in less than a minute! »](#)

[Watch our one-minute video »](#)

Not convinced? Read what our users are saying!

The screenshot shows the PythonAnywhere dashboard with a green header bar indicating 'All done! Your web app is now set up. Details below.' Below this, there's a list of web apps: 'example.pythonanywhere.com' (disabled), 'www.mydomain.com' (disabled), and 'www.myotherdomain.com' (selected). A button 'Add a new web app' is visible. To the right, there's a section for 'Configuration for www.myotherdomain.com' with a 'Reload' button and a note about DNS setup: 'How to point your domain at your website.' It shows a CNAME entry: 'CNAME: webapp-4.pythonanywhere.com'. Below this is a 'Traffic' section with two line graphs: 'Hits per month' and 'Hits per day', both showing low activity levels.

## Start hosting quickly

Just write your application. No need to configure or maintain a web server — everything is set up and ready to go.

[More »](#)

## Develop anywhere

Take your development environment with you! If you have a browser and an Internet connection, you've got everything you need.

[More »](#)

## Teach and learn

PythonAnywhere is a fully-fledged Python environment, ready to go, for students and teachers — concentrate on teaching, not on installation hassles.

## Amazing support

Need help with PythonAnywhere? If you get in touch, you can talk directly with the development team. Help for developers, from developers.

[More »](#)



## Plans and pricing

### Beginner: Free!

A limited account with one web app at `your-username.pythonanywhere.com`, restricted outbound Internet access from your apps, low CPU/bandwidth, no IPython/Jupyter notebook support.  
It works and it's a great way to get started!

[Create a Beginner account](#)

### Education accounts

Are you a teacher looking for a place your students can code Python? You're not alone. Click through to find out more about our [Education beta](#).

All of our paid plans come with a no-quibble 30-day money-back guarantee – you're billed monthly and you can cancel at any time. The minimum contract length is just one month. You get unrestricted Internet access from your applications, unlimited in-browser Python, Bash and database consoles, and full SSH access to your account. All accounts (including free ones) have screen-sharing with other PythonAnywhere accounts, and free SSL support (though you'll need to get a certificate for your own domains).

Hacker	\$5/month	Web dev	\$12/month	Startup	\$99/month	Custom	\$5 to \$500/month
Run your Python code in the cloud from one web app and the console		If you want to host small Python-based websites for you or for your clients		Start a business and don't worry about having to scale to handle traffic spikes		Want a combination that's not on the list? Create your own! All custom plans have:	
A Python IDE in your browser with unlimited Python/bash consoles		A Python IDE in your browser with unlimited Python/bash consoles		A Python IDE in your browser with unlimited Python/bash consoles		A Python IDE in your browser with unlimited Python/bash consoles	
One web app on a custom domain or <code>your-username.pythonanywhere.com</code>		Up to 2 web apps on custom domains or <code>your-username.pythonanywhere.com</code>		Up to 3 web apps on custom domains or <code>your-username.pythonanywhere.com</code>		Up to 20 web apps, on custom domains or <code>your-username.pythonanywhere.com</code>	
Enough power to run a typical 100,000 hit/day website. <a href="#">(more info)</a>		Enough power to run a typical 150,000 hit/day website on each web app. <a href="#">(more info)</a>		Enough power to run a typical 1,000,000 hit/day website on each web app. <a href="#">(more info)</a>		As many web workers as you need to scale your site's capacity. <a href="#">(more info)</a>	
2,000 CPU-seconds per day for							



## Create your account

Username:

This username is already taken. Please choose another.

Email:

Password:

Password (again):

I agree to the [Terms and Conditions](#) and the [Privacy and Cookies Policy](#),  
and confirm that I am at least 13 years old.

[Register](#)

We promise not to spam or pass your details on to anyone else.

[Dashboard](#) [Consoles](#) [Files](#) [Web](#) [Tasks](#) [Databases](#)

**Warning** You have not confirmed your email address yet. This means that you will not be able to reset your password if you lose it. If you cannot find your confirmation email anymore, send yourself a new one [here](#).

## Dashboard

Welcome, [bssuen](#)

**CPU Usage:** 0% used – 0.00s of 100s. Resets in 23 hours, 59 minutes [More Info](#)

[Upgrade Account](#)

**File storage:** 0% full – 48.0 KB of your 512.0 MB quota

Recent  
Consoles

[+ 5 -]

Recent  
Files

[+ 5 -]

You have no recent consoles.

You have no recently edited files.

New console:

\$ Bash

>>> Python ▾

More...

Welcome to PythonAnywhere!

This tour will give you a quick overview of how the site works. It only lasts 7 steps. Promise!

- Prev

Next +

End tour

Browse files

does not support this browser. Upgrade your account to get access!

All  
Web apps

You don't have any web apps.

Open Web tab



## Dashboard

Welcome, [bssuen](#)**CPU Usage:** 0% used – 0.00s of 100s. Resets in 23 hours, 59 minutes [More Info](#)**File storage:** 0% full – 48.0 KB of your 512.0 MB quota[Upgrade Account](#)

### Recent Consoles

[+ 5 -](#)*You have no recent consoles.*

### New console:

[\\$ Bash](#)[>>> Python ▾](#)[More...](#)

### Recent Files

[+ 5 -](#)*You have no recently edited files.*[+ Open another file](#)[Browse files](#)

### Recent Notebooks

[+ 5 -](#)

Your account does not support Jupyter Notebooks. [Upgrade your account](#) to get access!

### All Web apps

*You don't have any web apps.*[Open Web tab](#)

[Dashboard](#) [Consoles](#) [Files](#) **Web** [Tasks](#) [Databases](#) [Add a new web app](#)

You have no web apps

To create a PythonAnywhere-hosted web app,  
click the "Add a new web app" button to the left.

[Dashboard](#) [Consoles](#) [Files](#) **Web** [Tasks](#) [Databases](#)[+ Add a new web app](#)

### Create new web app

x

#### Your web app's domain name

Your account doesn't support custom domain names, so your PythonAnywhere web app will live at `bsaen.pythonanywhere.com`.

Want to change that? [Upgrade now!](#)

Otherwise, just click "Next" to continue.

[Cancel](#)[« Back](#)[Next »](#)

[Dashboard](#) [Consoles](#) [Files](#) **Web** [Tasks](#) [Databases](#)[+ Add a new web app](#)

## Create new web app



## Select a Python Web framework

...or select "Manual configuration" if you want detailed control.

- » [Django](#)
- » [web2py](#)
- » [Flask](#)
- » [Bottle](#)
- » [Manual configuration \(including virtualenvs\)](#)

What other frameworks should we have here? Send us some feedback using the link at the top of the page!

[Cancel](#)[« Back](#)[Next »](#)

[+ Add a new web app](#)

## Create new web app

x

## Select a Python version

- » Python 2.7 (Flask 1.0.2)
- » Python 3.4 (Flask 1.0.2)
- » Python 3.5 (Flask 1.0.2)
- » Python 3.6 (Flask 1.0.2)
- » Python 3.7 (Flask 1.0.2)

**Note:** If you'd like to use a different version of Flask to the default version, you can use a virtualenv for your web app. There are [instructions here](#).

[Cancel](#)[« Back](#)[Next »](#)

[Add a new web app](#)

### Create new web app



#### Quickstart new Flask project

Enter a path for a Python file you wish to use to hold your Flask app. If this file already exists, its contents will be overwritten with the new app.

**Path**[Cancel](#)[« Back](#)[Next »](#)

[Dashboard](#) [Consoles](#) [Files](#) **Web** [Tasks](#) [Databases](#)

All done! Your web app is now set up. Details below.

X

**bssuen.pythonanywhere.com**[+ Add a new web app](#)Configuration for [bssuen.pythonanywhere.com](#)

Reload:

[↻ Reload bssuen.pythonanywhere.com](#)

## Best before date:

We're happy to host your free website – and keep it free – for as long as you want to keep it running, but you'll need to log in at least once every three months and click the "Run until 3 months from today" button below. We'll send you an email a week before the site is disabled so that you don't forget to do that. [See here for more details](#).

This site will be disabled on **Tuesday 17 December 2019**[Run until 3 months from today](#)[Paying users' sites stay up forever without any need to log in to keep them running.](#)

## Traffic:

How busy is your site?

This month (previous month)	1 (0)
Today (yesterday)	1 (0)
Hour (previous hour)	1 (0)

Want some more data? [Paying accounts](#) get pretty charts ;)

[/home/](#) bssuen

## Directories

[New directory](#)

.local/   
.virtualenvs/   
mysite/

## Files

[New file](#)

	.bashrc		2019-09-17 09:03	559 bytes
	.gitconfig		2019-09-17 09:03	266 bytes
	.profile		2019-09-17 09:03	79 bytes
	.pythonstartup.py		2019-09-17 09:03	77 bytes
	.vimrc		2019-09-17 09:03	4.6 KB
	README.txt		2019-09-17 09:03	232 bytes

[Upload a file](#)

100MB maximum size



/home/ bssuen

### Directories

[New directory](#)

.local/   
.virtualenvs/   
mysite/

[Dashboard](#) [Consoles](#) **Files** [Web](#) [Tasks](#) [Databases](#)[Open Bash console here](#)

0% full – 132.0 KB of your 512.0 MB quota

### Files

[New file](#)

.bashrc	2019-09-17 09:03 559 bytes
.gitconfig	2019-09-17 09:03 266 bytes
.profile	2019-09-17 09:03 79 bytes
.pythonstartup.py	2019-09-17 09:03 77 bytes
.vimrc	2019-09-17 09:03 4.6 KB
README.txt	2019-09-17 09:03 232 bytes

[Upload a file](#)

100MB maximum size



/home/bssuen/ mysite

### Directories

static

[New directory](#)

\_\_pycache\_\_

### Files

Enter new file name, eg hello.py

[New file](#)

flask\_app.py 2019-09-17 09:27 186 bytes

[Upload a file](#)

100MiB maximum size



/home/bssuen/ mysite

### Directories

templates|

New directory

[\\_\\_pycache\\_\\_](#)/  
[static/](#)



### Files

Enter new file name, eg hello.py

New file

flask\_app.py 2019-09-17 09:27 186 bytes

Upload a file

100MiB maximum size



/home/bssuen/ mysite

### Directories

[New directory](#)

\_\_pycache\_\_/  
static/  
templates/



### Files

[New file](#)

flask\_app.py 2019-09-17 09:27 186 bytes

[Upload a file](#)

100MiB maximum size



/home/bssuen/ mysite

### Directories

[New directory](#)

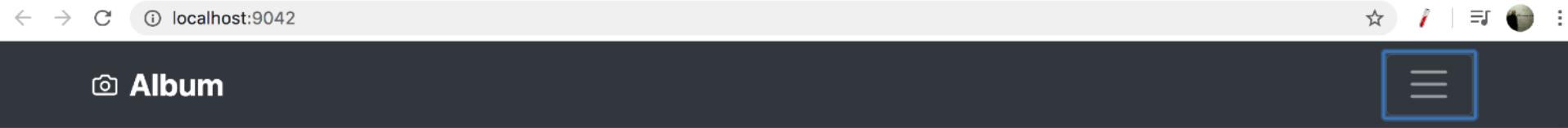
\_\_pycache\_\_  
static/  
templates/



### Files

[New file](#)

flask\_app.py 2019-09-17 09:27 186 bytes  
[Upload a file](#)  
100MiB maximum size



# Bernard's Album

Something short and leading about the collection below—its contents, the creator, etc. Make it short and sweet, but not too short so folks don't simply skip over it entirely.

Main call to action

Secondary action

Thumbnail

This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

[View](#) [Edit](#)

9 mins

Thumbnail

This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

[View](#) [Edit](#)

9 mins

Thumbnail

This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

[View](#) [Edit](#)

9 mins

# **Integrate Airtable REST API Data into Flask Template**



## Bernard's Album

Something short and leading about the collection below—its contents, the creator, etc. Make it short and sweet, but not too short so folks don't simply skip over it entirely.

Main call to action

Secondary action

Pencil HB

Col#:1  
Product Code:PEC  
Quantity:0  
Price:99999.99

[View](#) [Edit](#)

9 mins

Pencil 3B

Col#:2  
Product Code:PEC  
Quantity:500  
Price:0.52

[View](#) [Edit](#)

9 mins

Pencil 6B

Col#:3  
Product Code:PEC  
Quantity:500  
Price:0.47

[View](#) [Edit](#)

9 mins

```
1  from flask import Flask, render_template, jsonify
2  import pymysql.cursors
3  from flask_cors import CORS
4  import requests
5
6  app = Flask(__name__)
7  connection = pymysql.connect(host="localhost",
8      user="root",
9      passwd="root",
10     db="southwind",
11     charset='utf8mb4',
12     cursorclass=pymysql.cursors.DictCursor)
13
14 @app.route("/")
15 def home():
16     user = {"name": "Bernard"}
17     headers = {
18         'Authorization': 'Bearer keyc
19     }
20
21     params = (
22         ('view', 'Main'),
23     )
24
25     r = requests.get('https://api.airtable.com/v0/appVFIVS0NkWT41fM/products?api_key=          &sortField=_c
26     dict = r.json()
27     dataset = []
28     productID = []
29     productCode = []
30     name = []
31     for i in dict['records']:
32         dict = i['fields']
33         dataset.append(dict)
34     # {% for row in dataset %}{{ row["name"] }}{% endfor %}
35     return render_template('index3.html' album_user=user, dataset=dataset)
36
```

```

82 <section class="jumbotron text-center">
83   <div class="container">
84     <h1>{{album_user.name}}'s Album</h1>
85     <p class="lead text-muted">Something short and leading about the collection below—its contents, the creator, etc. Make it
short and sweet, but not too short so folks don't simply skip over it entirely.</p>
86     <p>
87       <a href="#" class="btn btn-primary my-2">Main call to action</a>
88       <a href="#" class="btn btn-secondary my-2">Secondary action</a>
89     </p>
90   </div>
91 </section>
92
93 <div class="album py-5 bg-light">
94   <div class="container">
95     {% for item in dataset %}
96       {% if loop.index == 1 %}{% set counter = ( counter | default(1) ) %}{% endif %}
97       {% if counter == 1 %}<div class="row">{% endif %}
98       <div class="col-md-4">
99         <div class="card mb-4 shadow-sm">
100           <img alt="Placeholder image for the album" class="bd-placeholder-img card-img-top" width="100%" height="225" xmlns="http://www.w3.org/2000/svg"
preserveAspectRatio="xMidYMid slice" focusable="false" role="img" aria-label="Placeholder: Thumbnail">
<title>Placeholder</title><rect width="100%" height="100%" fill="#55595c"/><text x="50%" y="50%" fill="#eceeef" dy=".3em">{{ item['name'] }}</text></img>
101         <div class="card-body">
102           <p class="card-text">Col#:{{ loop.index }}<br>Product Code:{{ item['productCode'] }}<br>Quantity:{{ item['quantity'] }}<br>Price:{{ item['price'] }}</p>
103         <div class="d-flex justify-content-between align-items-center">
104           <div class="btn-group">
105             <button type="button" class="btn btn-sm btn-outline-secondary">View</button>
106             <button type="button" class="btn btn-sm btn-outline-secondary">Edit</button>
107           </div>
108           <small class="text-muted">9 mins</small>
109         </div>
110       </div>
111     </div>
112   </div>
113   {% if counter == 3 %}</div><!--end row-->{% set counter = ( counter | default(1) ) %}{% else %}{% set counter = (
counter | default(0) ) + 1 %}{% endif %}
114   {% endfor %}
115 </div>
116 </div>

```

**Can you see the Airtable REST API when viewing  
the source codes (including the API key) in your  
browser?**

**Let's compare that with what we did last semester  
with our front-end `jQuery getJSON` codes.**

# Airtable GetJSON

## Airtable API Demo

Using jQuery getJSON and  
Airtable API for fun!

Airtable is a wonderful spreadsheet-like database with many useful features for data modeling and basic analytics. When used with other data visualization libraries, it can be a great tool for rapid prototyping

[Show Table](#)[Show Total](#)

[https://ybsuen.github.io/com5961-2019/lesson9/test\\_getjson.html](https://ybsuen.github.io/com5961-2019/lesson9/test_getjson.html)

Show Total

Show 10 entries

Search:

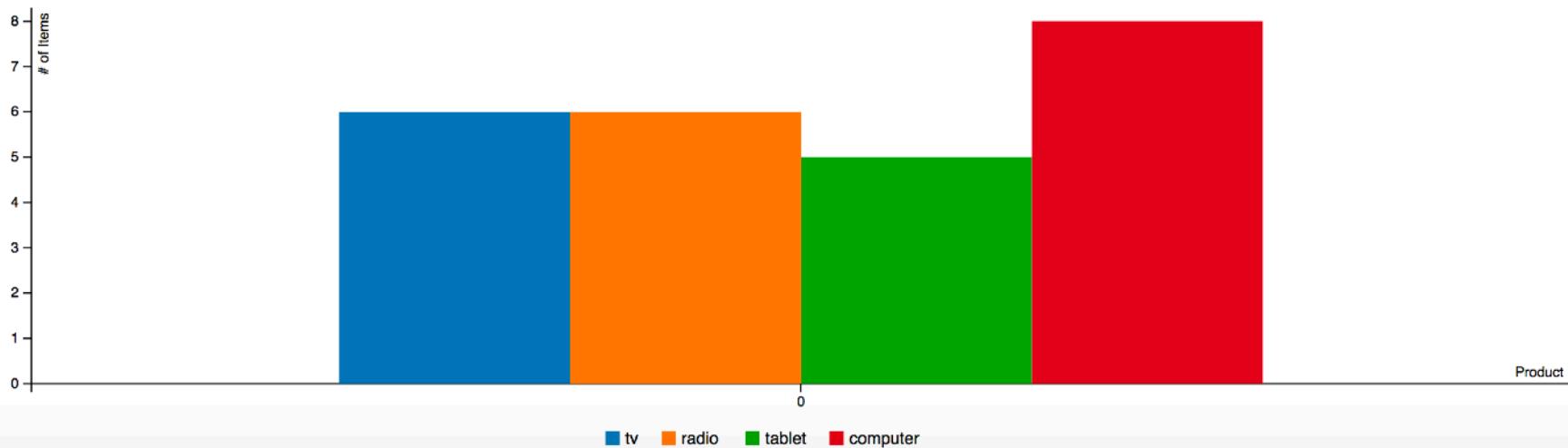
Product	Total Amount
computer	8
radio	6
tablet	5
tv	6

Showing 1 to 4 of 4 entries

Previous

1

Next



**Using JS Array & Object and GetJSON to obtain  
Airtable API data.**

# Airtable GetJSON

Airtable API Demo

Using jQuery getJSON and  
Airtable API for fun!

Airtable is a wonderful spreadsheet-like database with many useful features for data modeling and basic analytics. When used with other data visualization libraries, it can be a great tool for rapid prototyping

Show Table

Show 10 entries

Search:

Company ↴	Product ↴	Number ↴	Geo Code ↴	JSON	↪	Gender ↴	Name
Akzo	tv	12	Leeuwardenweg 179, arnhem, the netherlands	<a href="http://maps.googleapis.com/maps/api/geocode/json?sensor=false&amp;address=Leeuwardenweg+179%2C+arnhem%2C+the+netherlands">http://maps.googleapis.com/maps/api/geocode/json? sensor=false&amp;address=Leeuwardenweg+179%2C+arnhem%2C+the+netherlands</a>	↪	m	p. vanser
Akzo	radio	34	Leeuwardenweg 181, arnhem, the netherlands	<a href="http://maps.googleapis.com/maps/api/geocode/json?sensor=false&amp;address=Leeuwardenweg+181%2C+arnhem%2C+the+netherlands">http://maps.googleapis.com/maps/api/geocode/json? sensor=false&amp;address=Leeuwardenweg+181%2C+arnhem%2C+the+netherlands</a>	↪	m	p. jansse
Akzo	tv	43	Leeuwardenweg 178, arnhem, the netherlands	<a href="http://maps.googleapis.com/maps/api/geocode/json?sensor=false&amp;address=Leeuwardenweg+178%2C+arnhem%2C+the+netherlands">http://maps.googleapis.com/maps/api/geocode/json? sensor=false&amp;address=Leeuwardenweg+178%2C+arnhem%2C+the+netherlands</a>	↪	m	p. bansei
Akzo	tablet	9	Leeuwardenweg 183, arnhem, the netherlands	<a href="http://maps.googleapis.com/maps/api/geocode/json?sensor=false&amp;address=Leeuwardenweg+183%2C+arnhem%2C+the+netherlands">http://maps.googleapis.com/maps/api/geocode/json? sensor=false&amp;address=Leeuwardenweg+183%2C+arnhem%2C+the+netherlands</a>	↪	f	l. lokken

# **How does it work - getting Airtable data into JQuery DataTable?**

## Javascript sourced data

Using DataTables

### Examples

[Basic initialisation](#)[Advanced initialisation](#)[Styling](#)

### Data sources

[API](#)[Ajax](#)[Server-side](#)[Plug-ins](#)

### Manual

### Reference

### Extensions

### Plug-ins

[More Help](#)

### Blog

### Forums

### Support

### FAQs

At times you will wish to be able to create a table from dynamic information passed directly to DataTables, rather than having it read from the document. This is achieved using the `data Option` in the initialisation object, passing in an array of data to be used (like all other DataTables handled data, this can be arrays or objects using the `columns.data Option`).

A `table Tag` must be available on the page for DataTables to use. This examples shows an empty `table Tag` element being initialised as a DataTable with a set of data from a Javascript array. The columns in the table are dynamically created based on the `columns.title Option` configuration option.

Show `10` entries

Search:

Name	Position	Office	Extn.	Start date	Salary
Airi Satou	Accountant	Tokyo	5407	2008/11/28	\$162,700
Angelica Ramos	Chief Executive Officer (CEO)	London	5797	2009/10/09	\$1,200,000
Ashton Cox	Junior Technical Author	San Francisco	1562	2009/01/12	\$86,000
Bradley Greer	Software Engineer	London	2558	2012/10/13	\$132,000
Brenden Wagner	Software Engineer	San Francisco	1314	2011/06/07	\$206,850
Brielle Williamson	Integration Specialist	New York	4804	2012/12/02	\$372,000

[https://datatables.net/examples/data\\_sources/js\\_array.html](https://datatables.net/examples/data_sources/js_array.html)

Get DataTables

[Download](#)

[Purchase](#)

Caesar Vance	Pre-Sales Support	New York	8330	2011/12/12	\$106,450
Cedric Kelly	Senior Javascript Developer	Edinburgh	6224	2012/03/29	\$433,060
Charde Marshall	Regional Director	San Francisco	6741	2008/10/16	\$470,600
Colleen Hurst	Javascript Developer	San Francisco	2360	2009/09/15	\$205,500

Showing 1 to 10 of 36 entries

Previous 1 2 3 4 Next

[Javascript](#) [HTML](#) [CSS](#) [Comments \(0\)](#)

The Javascript shown below is used to initialise the table shown in this example:

```

1 var dataSet = [
2   [ "Tiger Nixon", "System Architect", "Edinburgh", "5421", "2011/04/25", "$320,800" ],
3   [ "Garrett Winters", "Accountant", "Tokyo", "8422", "2011/07/25", "$170,750" ],
4   [ "Ashton Cox", "Junior Technical Author", "San Francisco", "1562", "2009/01/12", "$86,000" ],
5   [ "Cedric Kelly", "Senior Javascript Developer", "Edinburgh", "6224", "2012/03/29", "$433,000" ],
6   [ "Airi Satou", "Accountant", "Tokyo", "5407", "2008/11/28", "$162,700" ],
7   [ "Brielle Williamson", "Integration Specialist", "New York", "4804", "2012/12/02", "$372,000" ],
8   [ "Herrod Chandler", "Sales Assistant", "San Francisco", "9608", "2012/08/06", "$137,500" ],
9   [ "Rhona Davidson", "Integration Specialist", "Tokyo", "6200", "2010/10/14", "$327,900" ],
10  [ "Colleen Hurst", "Javascript Developer", "San Francisco", "2360", "2009/09/15", "$205,500" ],
11  [ "Sonya Frost", "Software Engineer", "Edinburgh", "1667", "2008/12/13", "$103,600" ],
12  [ "Jena Gaines", "Office Manager", "London", "3814", "2008/12/19", "$90,560" ],
13  [ "Quinn Flynn", "Support Lead", "Edinburgh", "9497", "2013/03/03", "$342,000" ],
14  [ "Charde Marshall", "Regional Director", "San Francisco", "6741", "2008/10/16", "$470,600" ],
15  [ "Haley Kennedy", "Senior Marketing Designer", "London", "3597", "2012/12/18", "$313,500" ],
16  [ "Tatyana Fitzpatrick", "Regional Director", "London", "1965", "2010/03/17", "$385,750" ],
17  [ "Michael Silva", "Marketing Designer", "London", "1581", "2012/11/27", "$198,500" ],
18  [ "Paul Byrd", "Chief Financial Officer (CFO)", "New York", "3059", "2010/06/09", "$725,000" ],
19  [ "Gloria Little", "Systems Administrator", "New York", "1721", "2009/04/10", "$237,500" ],
20  [ "Bradley Greer", "Software Engineer", "London", "2558", "2012/10/13", "$132,000" ],
21  [ "Dai Rios", "Personnel Lead", "Edinburgh", "2298", "2012/09/26", "$217,500" ],
22  [ "Jenette Caldwell", "Development Lead", "New York", "1937", "2011/09/03", "$345,000" ],
23  [ "Yuri Berry", "Chief Marketing Officer (CMO)", "New York", "6154", "2009/06/25", "$675,000" ],
24  [ "Caesar Vance", "Pre-Sales Support", "New York", "8330", "2011/12/12", "$106,450" ],
25  [ "Doris Wilder", "Sales Assistant", "Sydney", "3023", "2010/09/20", "$85,600" ],
26  [ "Angelica Ramos", "Chief Executive Officer (CEO)", "London", "5797", "2009/10/09", "$1,200,000" ],
27  [ "Gavin Joyce", "Developer", "Edinburgh", "8822", "2010/12/22", "$92,575" ],

```

[https://datatables.net/examples/data\\_sources/js\\_array.html](https://datatables.net/examples/data_sources/js_array.html)

```

8      [ "Herrod Chandler", "Sales Assistant", "San Francisco", "9608", "2012/08/06", "$137,500" ],
9      [ "Rhona Davidson", "Integration Specialist", "Tokyo", "6200", "2010/10/14", "$327,900" ],
10     [ "Colleen Hurst", "Javascript Developer", "San Francisco", "2360", "2009/09/15", "$205,500" ],
11     [ "Sonya Frost", "Software Engineer", "Edinburgh", "1667", "2008/12/13", "$103,600" ],
12     [ "Jena Gaines", "Office Manager", "London", "3814", "2008/12/19", "$90,560" ],
13     [ "Quinn Flynn", "Support Lead", "Edinburgh", "9497", "2013/03/03", "$342,000" ],
14     [ "Charde Marshall", "Regional Director", "San Francisco", "6741", "2008/10/16", "$470,600" ],
15     [ "Haley Kennedy", "Senior Marketing Designer", "London", "3597", "2012/12/18", "$313,500" ],
16     [ "Tatyana Fitzpatrick", "Regional Director", "London", "1965", "2010/03/17", "$385,750" ],
17     [ "Michael Silva", "Marketing Designer", "London", "1581", "2012/11/27", "$198,500" ],
18     [ "Paul Byrd", "Chief Financial Officer (CFO)", "New York", "3059", "2010/06/09", "$725,000" ],
19     [ "Gloria Little", "Systems Administrator", "New York", "1721", "2009/04/10", "$237,500" ],
20     [ "Bradley Greer", "Software Engineer", "London", "2558", "2012/10/13", "$132,000" ],
21     [ "Dai Rios", "Personnel Lead", "Edinburgh", "2290", "2012/09/26", "$217,500" ],
22     [ "Janette Caldwell", "Development Lead", "New York", "1937", "2011/09/03", "$345,000" ],
23     [ "Yuri Berry", "Chief Marketing Officer (CMO)", "New York", "6154", "2009/06/25", "$675,00" ],
24     [ "Caesar Vance", "Pre-Sales Support", "New York", "8330", "2011/12/12", "$106,450" ],
25     [ "Doris Wilder", "Sales Assistant", "Sydney", "3023", "2010/09/20", "$85,600" ],
26     [ "Angelica Ramos", "Chief Executive Officer (CEO)", "London", "5797", "2009/10/09", "$1,20" ],
27     [ "Gavin Joyce", "Developer", "Edinburgh", "8822", "2010/12/22", "$92,575" ],
28     [ "Jennifer Chang", "Regional Director", "Singapore", "9239", "2010/11/14", "$357,650" ],
29     [ "Brenden Wagner", "Software Engineer", "San Francisco", "1314", "2011/06/07", "$206,850" ],
30     [ "Fiona Green", "Chief Operating Officer (COO)", "San Francisco", "2947", "2010/03/11", "$" ],
31     [ "Shou Itou", "Regional Marketing", "Tokyo", "8899", "2011/08/14", "$163,000" ],
32     [ "Michelle House", "Integration Specialist", "Sydney", "2769", "2011/06/02", "$95,400" ],
33     [ "Suki Burks", "Developer", "London", "6832", "2009/10/22", "$114,500" ],
34     [ "Prescott Bartlett", "Technical Author", "London", "3606", "2011/05/07", "$145,000" ],
35     [ "Gavin Cortez", "Team Leader", "San Francisco", "2860", "2008/10/26", "$235,500" ],
36     [ "Martena Mccray", "Post-Sales support", "Edinburgh", "8240", "2011/03/09", "$324,050" ],
37     [ "Unity Butler", "Marketing Designer", "San Francisco", "5384", "2009/12/09", "$85,675" ],
38   ];
39
40 $(document).ready(function() {
41   $('#example').DataTable( {
42     data: dataSet,
43     columns: [
44       { title: "Name" },
45       { title: "Position" },
46       { title: "Office" },
47       { title: "Extn." },
48       { title: "Start date" },
49       { title: "Salary" }
50     ]
51   } );
52 });

```

[https://datatables.net/examples/data\\_sources/js\\_array.html](https://datatables.net/examples/data_sources/js_array.html)

```
17
18     $("button#get_data").click(function() {
19         var items = [];
20         var i = 0;
21         var airtable_read_endpoint = "https://api.airtable.com/v0/appKIU0zkHt3";
22         var dataSet = [];
23         $.getJSON(airtable_read_endpoint, function(result) {
24             $.each(result.records, function(key,value) {
25                 items = [];
26                 items.push(value.fields.company);
27                 items.push(value.fields.product_code);
28                 items.push(value.fields.product_no);
29                 items.push(value.fields.geocode2);
30                 items.push(value.fields.json);
31                 items.push(value.fields.gender);
32                 items.push(value.fields.name);
33                 dataSet.push(items);
34                 console.log(items);
35             }); // end .each
36             console.log(dataSet);
37
```

JS

```
$('#table1').DataTable( {  
    data: dataSet,  
    retrieve: true,  
    columns: [  
        { title: "Company",  
            defaultContent:""},  
        { title: "Product",  
            defaultContent:""},  
        { title: "Number",  
            defaultContent:""},  
        { title: "Geo Code",  
            defaultContent:""},  
        { title: "JSON",  
            defaultContent:""},  
        { title: "Gender",  
            defaultContent:""},  
        { title: "Name",  
            defaultContent:""},  
    ]  
} );  
}); // end .getJSON  
}); // end button
```

```
<button type="button" class="btn btn-warning" id='get_data'>Show Table</button><br>  
<div class="table-responsive">  
| | | <table id="table1" width="100%" class="table table-bordered"></table>  
</div><!-- table responsive -->  
<button type="button" class="btn btn-warning" id='get_data2'>Show Total</button><br>  
<div class="table-responsive">  
| | | <table id="table2" width="100%" class="table table-bordered"></table>  
</div><!-- table responsive -->
```

HTML

**The same logic can be applied to C3D3.**

```


><p id="demo">Airtable is a wonderful spreadsheet-like
  <br/>!-- Sign and date the page, it's only polite! -->
  <button type="button" class="btn btn-warning" id='get_data'>Show Table</
  <div class="table-responsive">
    <table id="table1" width="100%" class="table table-bordered"></tab
  /div><!-- table responsive -->
  <button type="button" class="btn btn-warning" id='get_data2'>Show Total<
  <div class="table-responsive">
    <table id="table2" width="100%" class="table table-bordered"></tab
  /div><!-- table responsive -->
  <div id="chart"></div>


```

HTML

JS

```

$.getJSON(airtable_read_endpoint, function(result) {
  $.each(result.records, function(key,value) {
    items = [];
    items.push(value.fields.Name);
    items.push(value.fields.total_items_by_catego
  }); // end .each
  console.log(dataSet);

  $('#table2').DataTable( {
    data: dataSet,
    retrieve: true,
    columns: [
      { title: "Product",
        defaultContent:""},,
      { title: "Total Amount",
        defaultContent:"" },
    ]
  });
}

var chart = c3.generate({
  data: {
    columns: dataSet,
    type : 'bar'
  },
  axis: {
    x: {label: 'Product'},
    y: {label: '# of Items'}
  },
  bar: {
    title: "# of Items by Product Category:",
  }
});
```

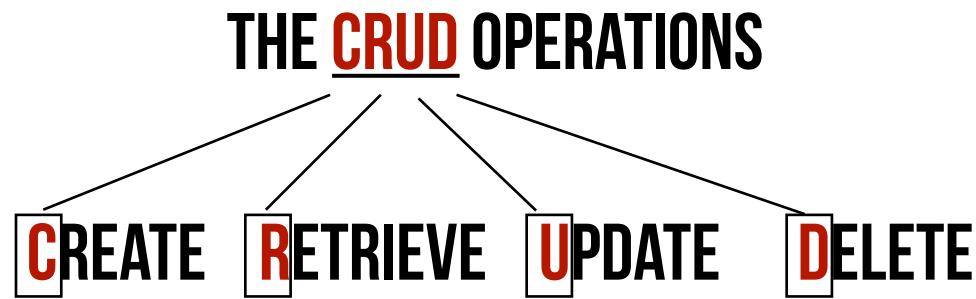
## The lesson learned:

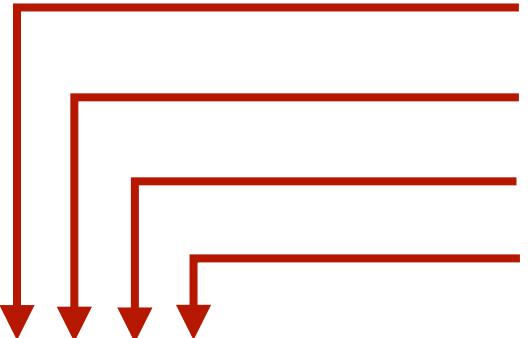
- Transition from **Front-end JS codes** to **Back-end Python codes**
- Almost the **same logic** but **different programming language syntax.**
- From **JS array** and **object** to Python **list & dictionary**
- Looping in **JS** to looping in **Python.**
- Back-end coding allows hiding of **API key.**

**Going beyond the “R” in CRUD.**

# **Perform CRUD Operations in Jupyter Notebook Using PyMysql**

# Data Manipulation Language (DML)

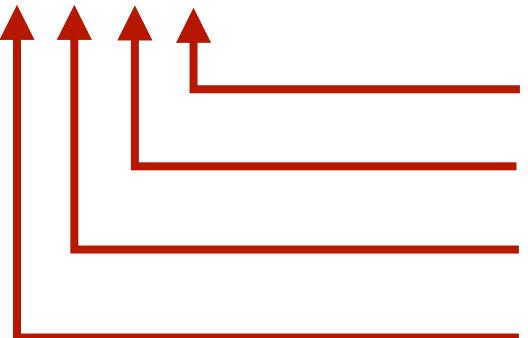




Create  
Retrieve  
Update  
Delete

**“where”** filter can be appended to CRUD operations.

## CRUD Operations in MySQL equivalent codes



delete from <table>  
update <table> set <column> = ...  
select \* from <table>  
insert into <table> values  
(<columns>...)

# Dropping and Creating a Table

```
1 import pymysql.cursors #initialise PyMysql module
2 import pandas as pd # initialise pandas module
3 connection = pymysql.connect(host="localhost",
4                               user="root",
5                               passwd="root",
6                               db="abc",
7                               charset='utf8mb4',
8                               cursorclass=pymysql.cursors.DictCursor)
9 cursor = connection.cursor()
10 sql1 = "drop table if exists test"
11 sql2 = "CREATE TABLE test (productID int(10) unsigned NOT NULL,productCode char(3) NOT NULL DEFAULT '' ,name varchar(255) NOT NULL,description text NOT NULL,price decimal(10,2) NOT NULL,stockLevel int(10) unsigned NOT NULL,PRIMARY KEY(productID)) ENGINE=InnoDB"
12 try:
13     cursor.execute(sql1)
14     cursor.execute(sql2)
15     print("Table successfully created.")
16 except Exception as e:
17     print("Exception occurred.",e)
18 connection.close()
```

Table successfully created.

In [77]:

```
1 import pymysql.cursors #initialise PyMysql module
2 import pandas as pd # initialise pandas module
3 connection = pymysql.connect(host="localhost",
4                               user="root",
5                               passwd="root",
6                               db="southwind",
7                               charset='utf8mb4',
8                               cursorclass=pymysql.cursors.DictCursor)
9 with connection.cursor() as cursor:
10     # Execute SQL Select command in Python using PyMysql built-in methods
11     sql = "SELECT * FROM products"
12     cursor.execute(sql)
13     # display results
14     result = cursor.fetchall()
15     dataset = []
16     temp = {}
17     i = 0
18     for row in result:
19         temp[i] = row
20         dataset.append(temp[i])
21         i = i + 1
22     print('\n')
23     print(pd.DataFrame(dataset)) # additional command
24     print('\n')
25 connection.close()
```

## CRUD - Retrieve data with SELECT

Using Pandas module for framing dataset in a table format

	productID	productCode	name	quantity	price	supplierID
0	1001	PEN	Pen Red	5000	1.23	501
1	1002	PEN	Pen Blue	8000	1.25	501
2	1003	PEN	Pen Black	2000	1.25	501
3	1004	PEC	Pencil 2B	10000	0.48	501
4	1005	PEC	Pencil 2H	8000	0.40	502

```

1 import pandas as pd # initialise pandas module
2 connection = pymysql.connect(host="localhost",
3                               user="root",
4                               passwd="root",
5                               db="abc",
6                               charset='utf8mb4',
7                               cursorclass=pymysql.cursors.DictCursor)
8 cursor = connection.cursor()
9 # Execute SQL Insert command in Python using PyMysql built-in methods
10 sql = "INSERT INTO test VALUES('2020', 'PEN', 'Pen Purple', 600,15.5, 503),('2020', 'PEN', 'Pen Black', 400,25.5, 501);"
11 try:
12     cursor.execute(sql)
13     connection.commit()
14     print("Entry successfully inserted.")
15 except Exception as e:
16     print("Exception occurred.",e)
17 # Display results
18 sql = "SELECT * FROM test"
19 cursor.execute(sql)
20 result = cursor.fetchall()
21 connection.close()
22 dataset = []
23 temp = {}
24 i = 0
25 for row in result:
26     temp[i] = row
27     dataset.append(temp[i])
28     i = i + 1
29 print('\n')
30 print(pd.DataFrame(dataset)) # additional command
31 print('\n')

```

## CRUD - Create data with INSERT

Entry successfully inserted.

```

1 import pandas as pd # initialise pandas module
2 connection = pymysql.connect(host="localhost",
3                               user="root",
4                               passwd="root",
5                               db="abc",
6                               charset='utf8mb4',
7                               cursorclass=pymysql.cursors.DictCursor)
8 cursor = connection.cursor()
9 # Execute SQL update command in Python using PyMysql built-in methods
10 sql = "UPDATE test SET quantity = quantity + 50, price = 1.99 WHERE name = 'Pen Purple';"
11 try:
12     cursor.execute(sql)
13     connection.commit()
14     print("Entry successfully updated.")
15 except Exception as e:
16     print("Exception occurred.",e)
17 # Display results
18 sql = "SELECT * FROM test"
19 cursor.execute(sql)
20 result = cursor.fetchall()
21 connection.close()
22 dataset = []
23 temp = {}
24 i = 0
25 for row in result:
26     temp[i] = row
27     dataset.append(temp[i])
28     i = i + 1
29 print('\n')
30 print(pd.DataFrame(dataset)) # additional command
31 print('\n')

```

## CRUD - Update data with UPDATE

Entry successfully updated.

```

1 import pandas as pd # initialise pandas module
2 connection = pymysql.connect(host="localhost",
3                               user="root",
4                               passwd="root",
5                               db="abc",
6                               charset='utf8mb4',
7                               cursorclass=pymysql.cursors.DictCursor)
8 cursor = connection.cursor()
9 # Execute SQL delete command in Python using PyMysql built-in methods
10 sql = "DELETE FROM test WHERE name LIKE 'Pen Black';"
11 try:
12     cursor.execute(sql)
13     connection.commit()
14     print("Entry successfully deleted.")
15 except Exception as e:
16     print("Exception occurred.",e)
17 # Display results
18 sql = "SELECT * FROM test"
19 cursor.execute(sql)
20 result = cursor.fetchall()
21 connection.close()
22 dataset = []
23 temp = {}
24 i = 0
25 for row in result:
26     temp[i] = row
27     dataset.append(temp[i])
28     i = i + 1
29 print('\n')
30 print(pd.DataFrame(dataset)) # additional command
31 print('\n')

```

## CRUD - Delete data with DELETE

Entry successfully deleted.

**Python Flask  
&  
SQL**

**Lean & Agile  
Development**

**9 Types of  
Business Model**

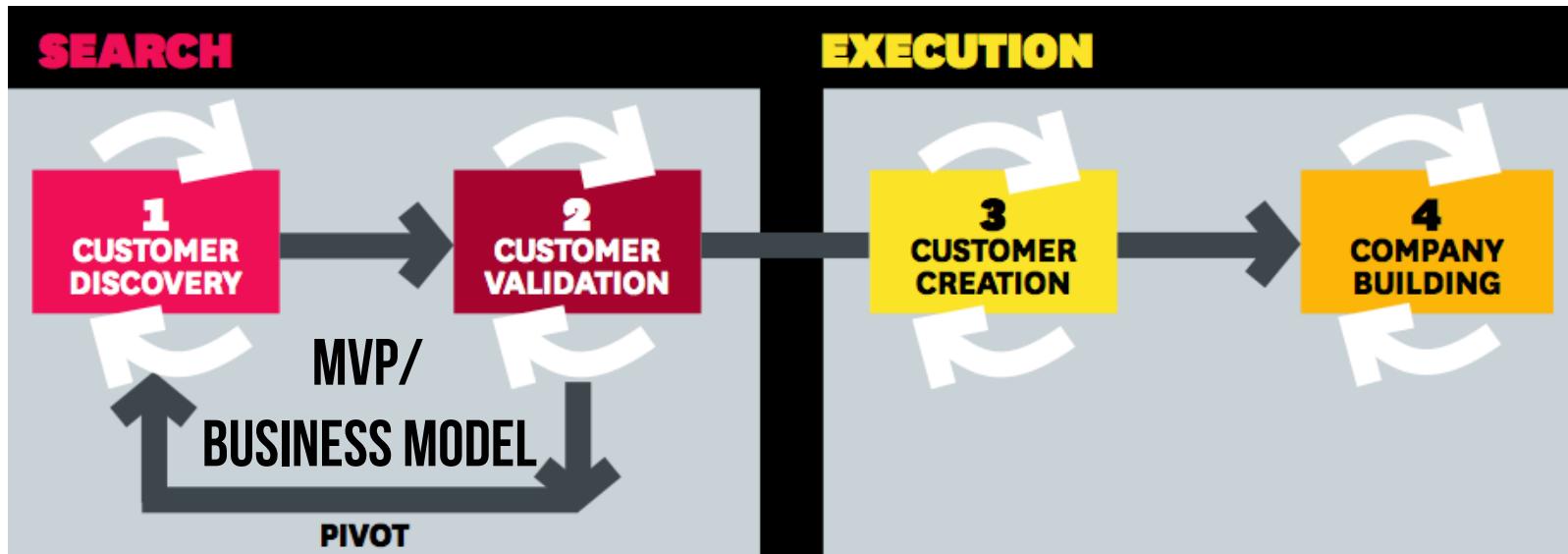
# **The Lean Startup Approach**

# Lean StartUp 精益创业



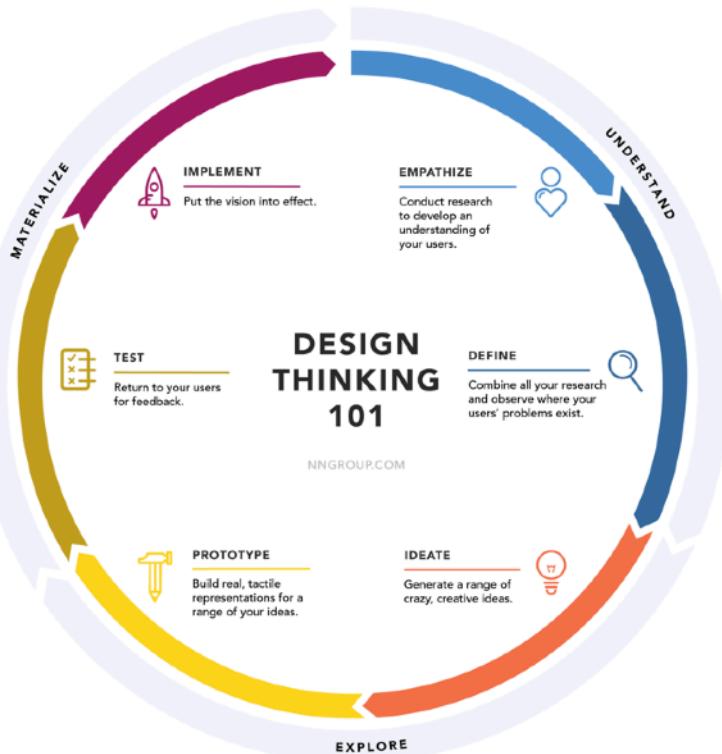
# **What Distinguishes a Startup from a Corporation?**

# BUILD-MEASURE-LEARN CYCLE IN SEARCH OF PRODUCT-MARKET FIT

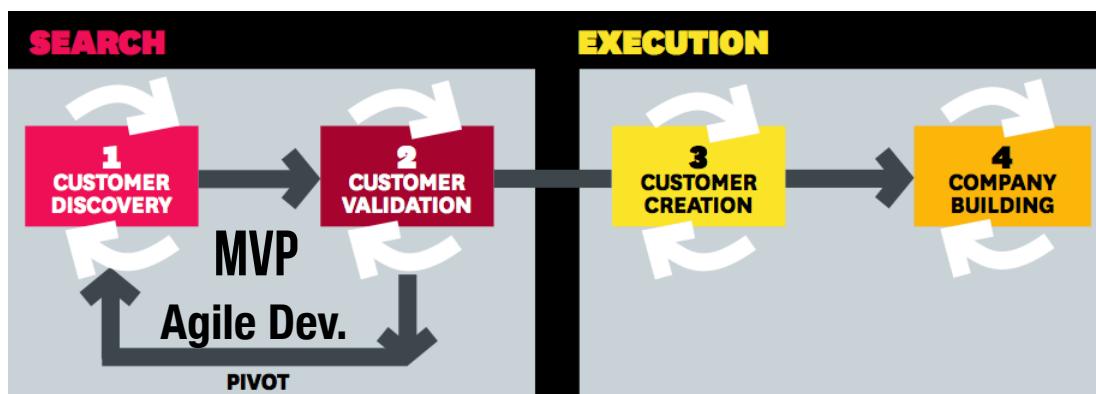


Source: "Why Lean Start-up Changes Everything" by Steve Blank, Harvard Business Review May 2013

# **Can Design Thinking and the Lean Method Work Together?**



Source: Sarah Gibbons,  
Design Thinking 101  
Nielsen Norman Group

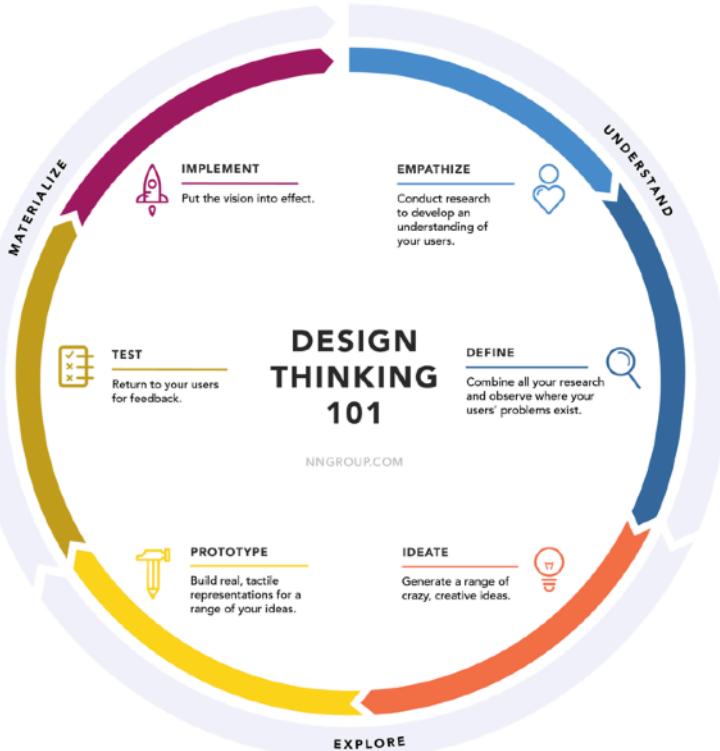


Source: “Why Lean Start-up Changes Everything” by Steve Blank, Harvard Business Review May 2013



# **The Google Design Sprint: Combining Lean and Design Thinking with Agile Development**





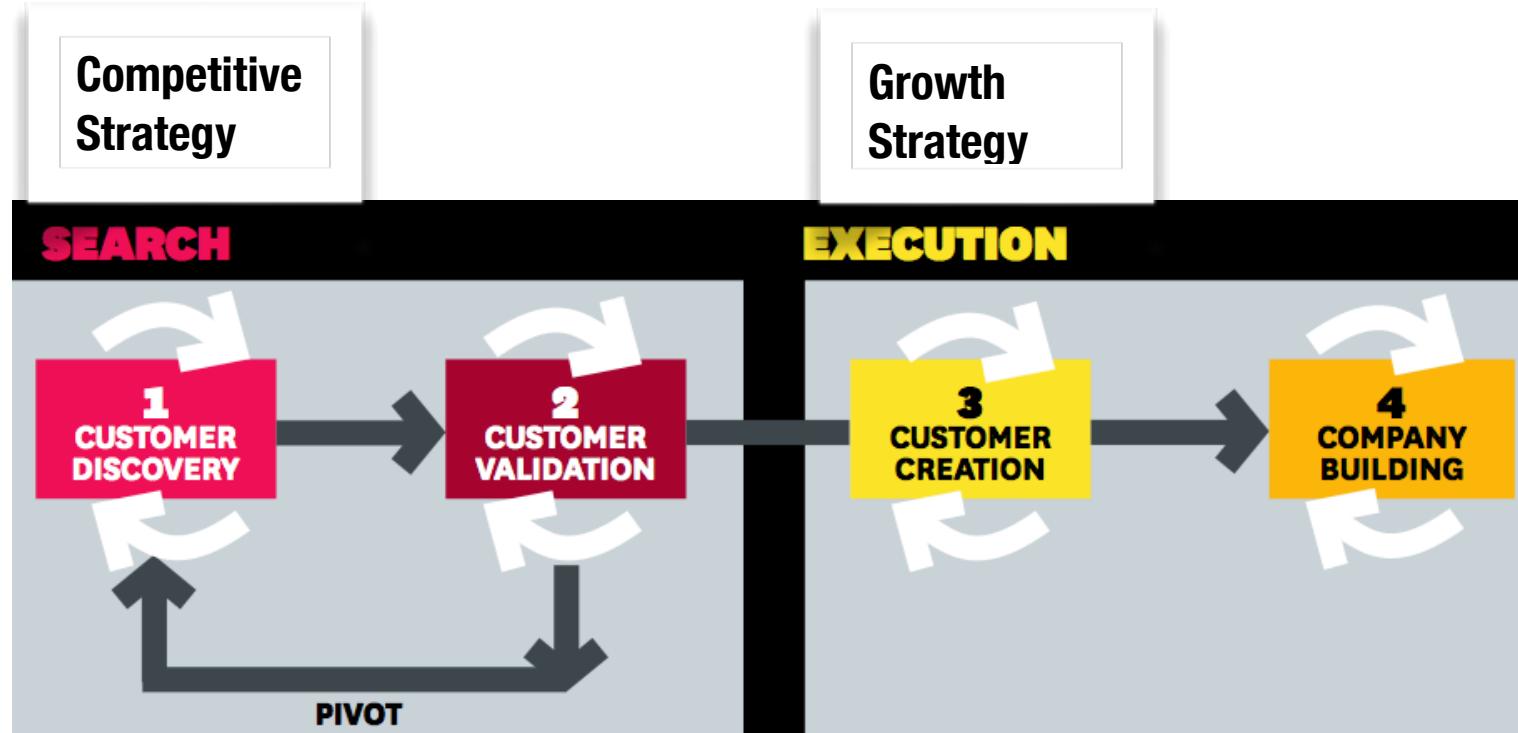
## USE DESIGN SPRINT IN AGILE TO SUPPLEMENT MVP ITERATION



Source: “Why Lean Start-up Changes Everything” by Steve Blank, Harvard Business Review May 2013

Source: Sarah Gibbons,  
Design Thinking 101  
Nielsen Norman Group

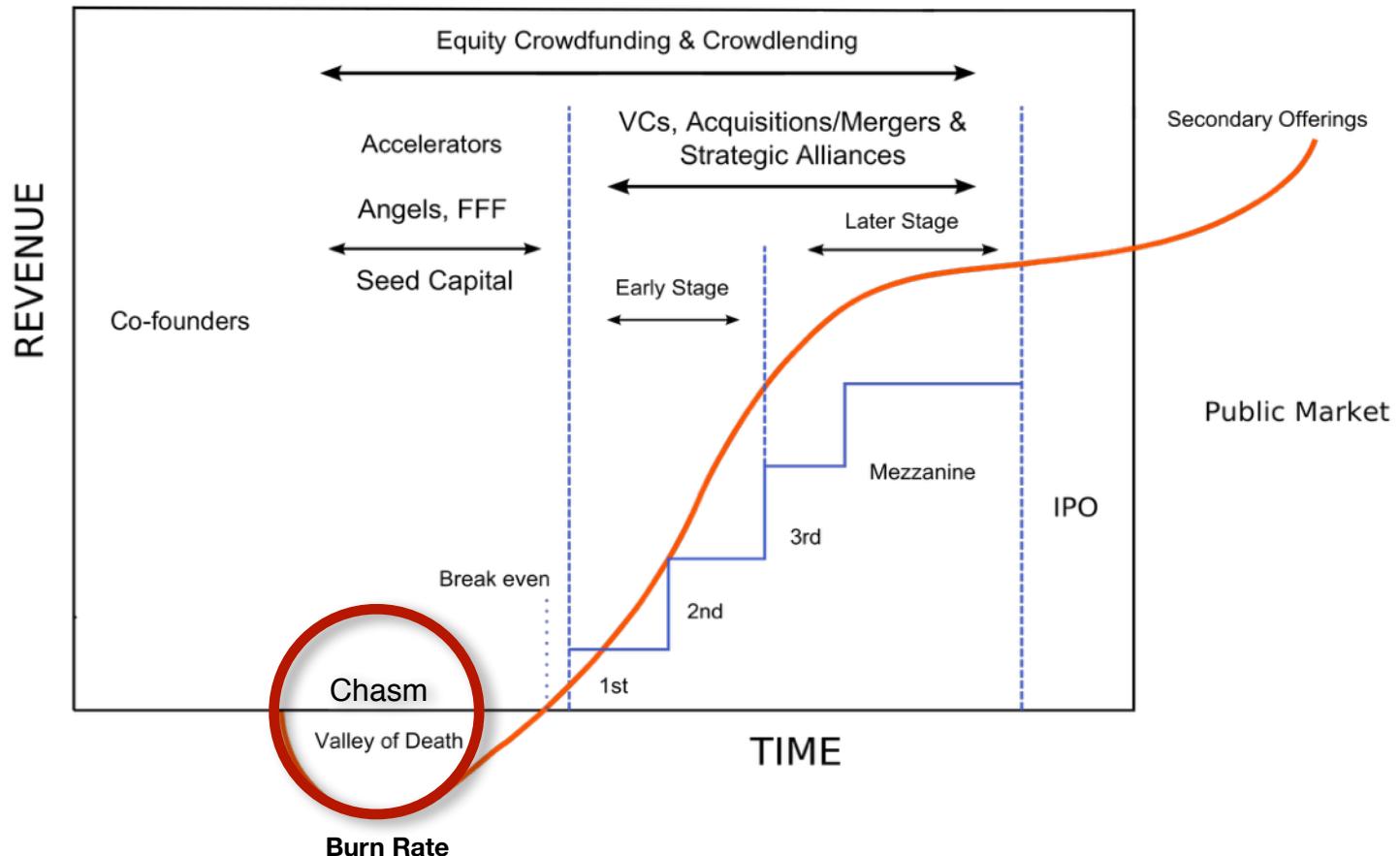
**From converting user to customer and transforming prototype into minimum Viable Product, startups aim to control unnecessary wastage, create sustainable value and scale up the business rapidly.**

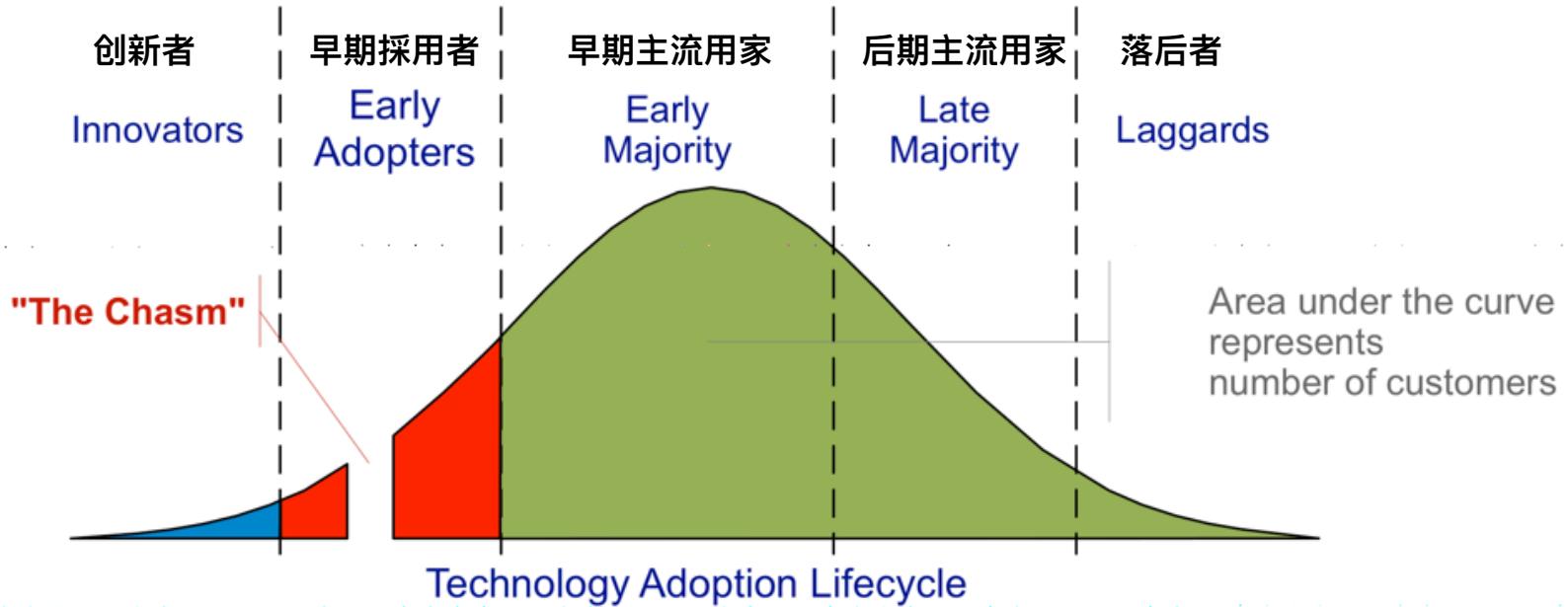


Source: "Why Lean Start-up Changes Everything" by Steve Blank, Harvard Business Review May 2013



## Startup Financing Cycle



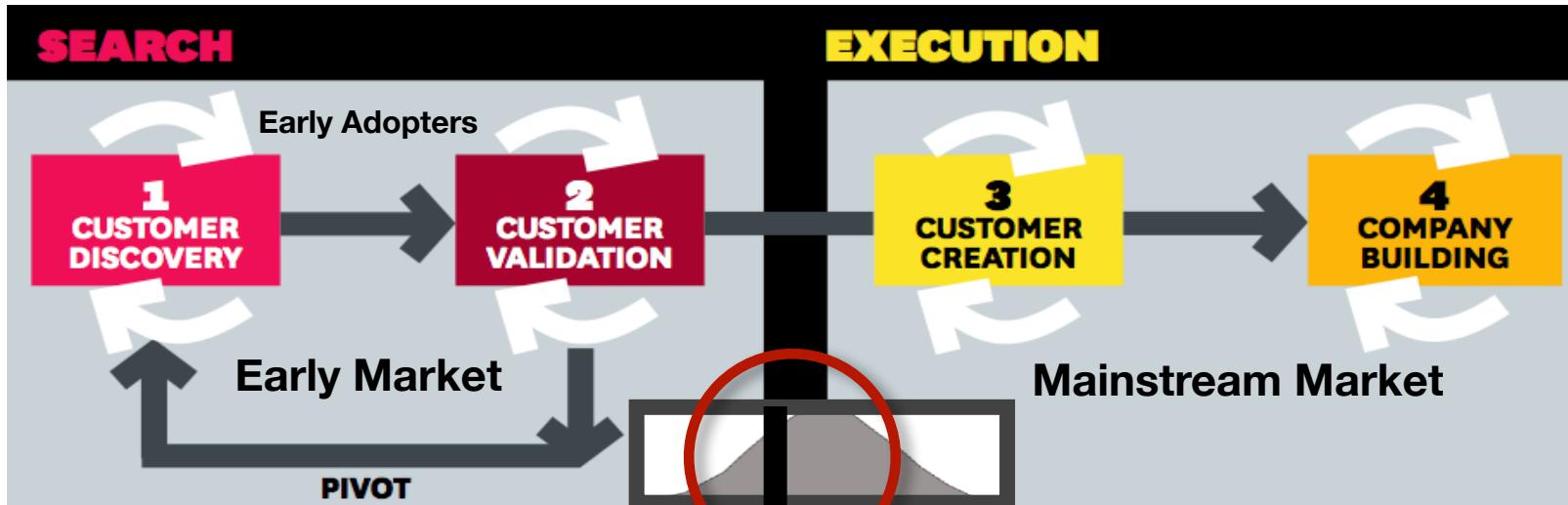


Source: From Wikimedia Commons, the free media repository  
“Crossing the Chasm” is the book title by the author **Geoffrey Moore**



## **Who are Facebook's early adopters?**

Source:



PROBLEM	SOLUTION	UNIQUE VALUE PROPOSITION	UNFAIR ADVANTAGE	CUSTOMER SEGMENTS
KEY METRICS			CHANNELS	
<b>COST STRUCTURE</b>		<b>REVENUE STREAMS</b>		

How much

现金流的计算

# **LEAN CANVAS: How does it work?**

**The LEAN method is used to test out the hypothesis hidden in the business model.**



## Which Business Model and How to Test It ?

Adapted from Alex Osterwalter

# 最小化可行产品 MVP(Minimum Viable Product)



验证甚么？



这个有人买吗？



# HYPOTHESIS TESTING

## 验证假设

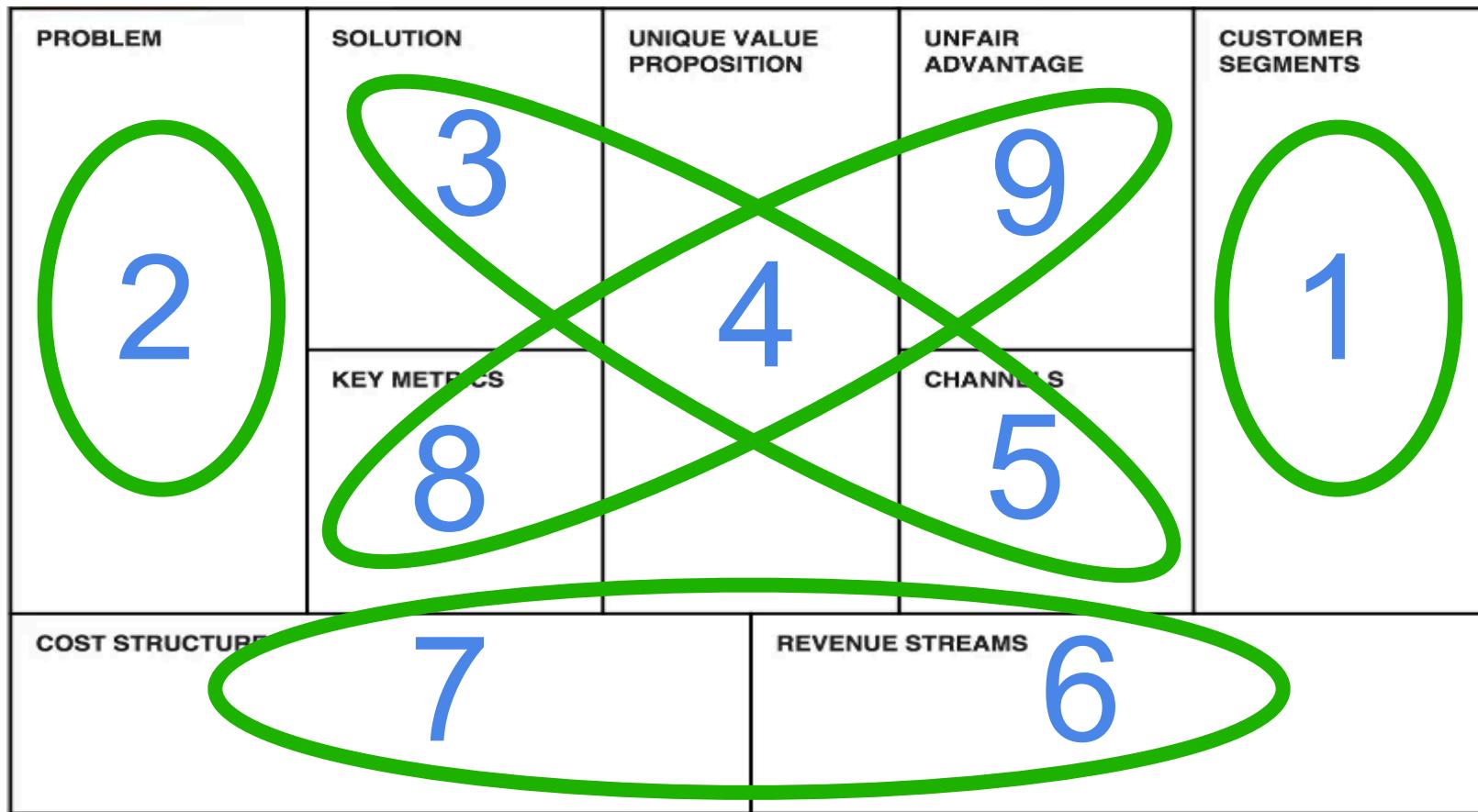
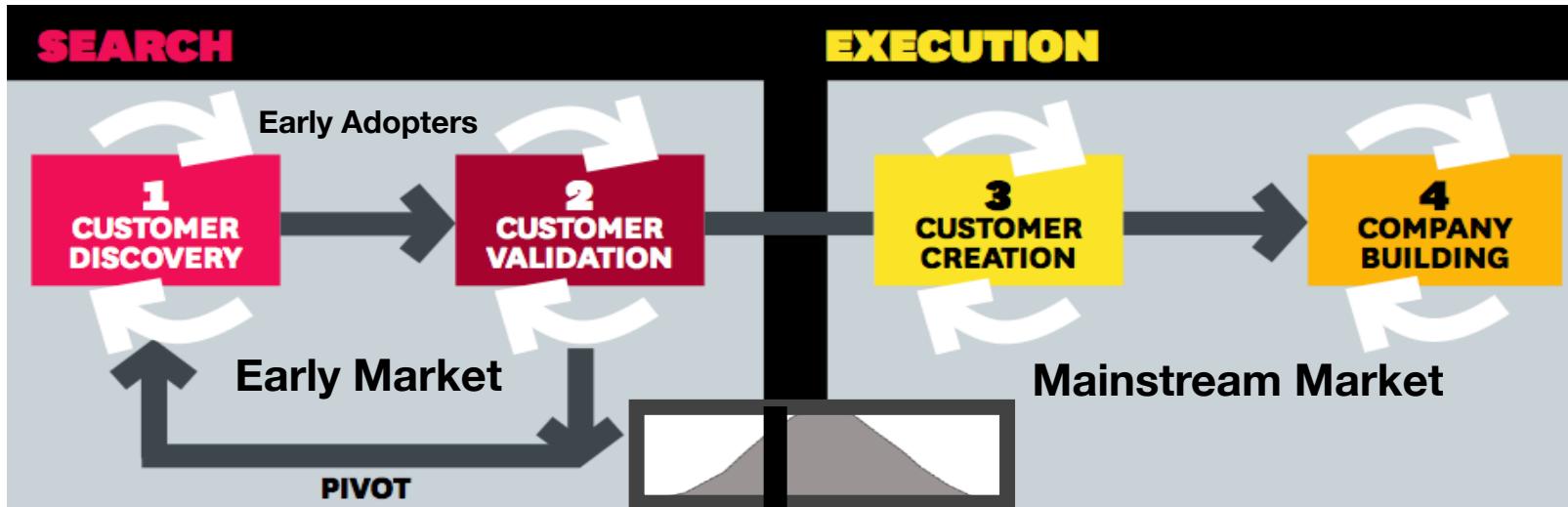


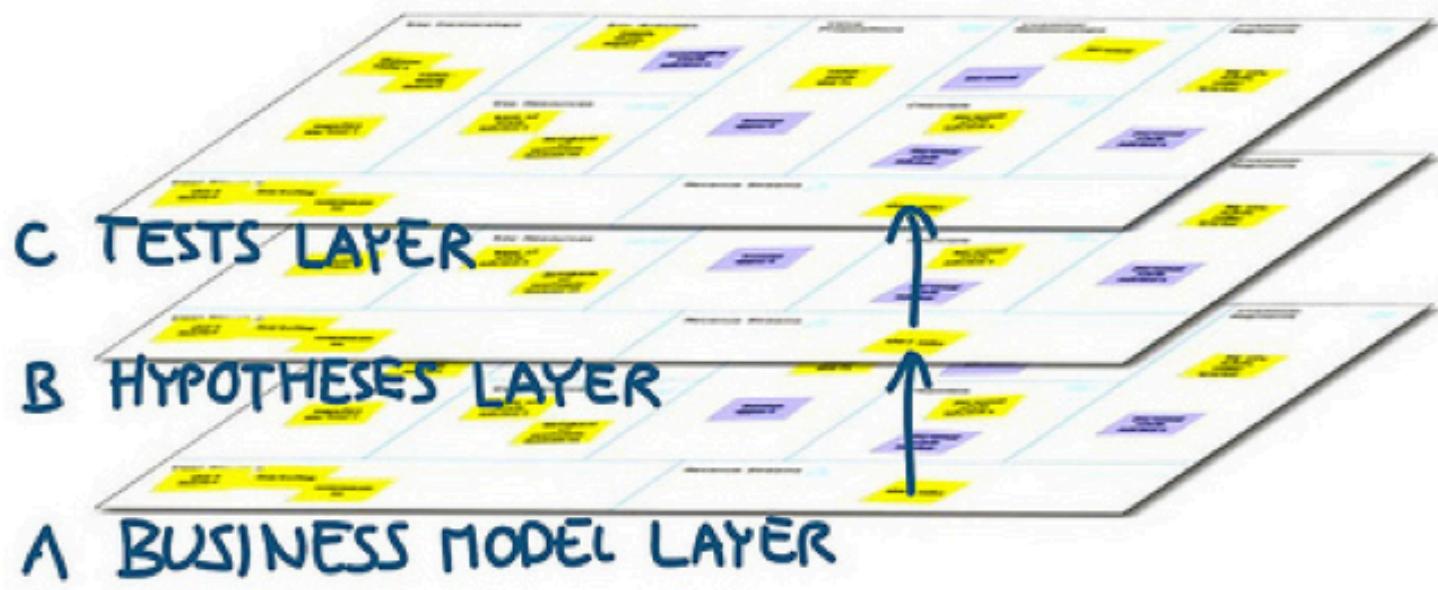
## Which Business Model and How to Test It ?

Adapted from Alex Osterwalter

心法：  
2 H 1 T

心法：  
一右一左，過三關  
右收左出，度門檻

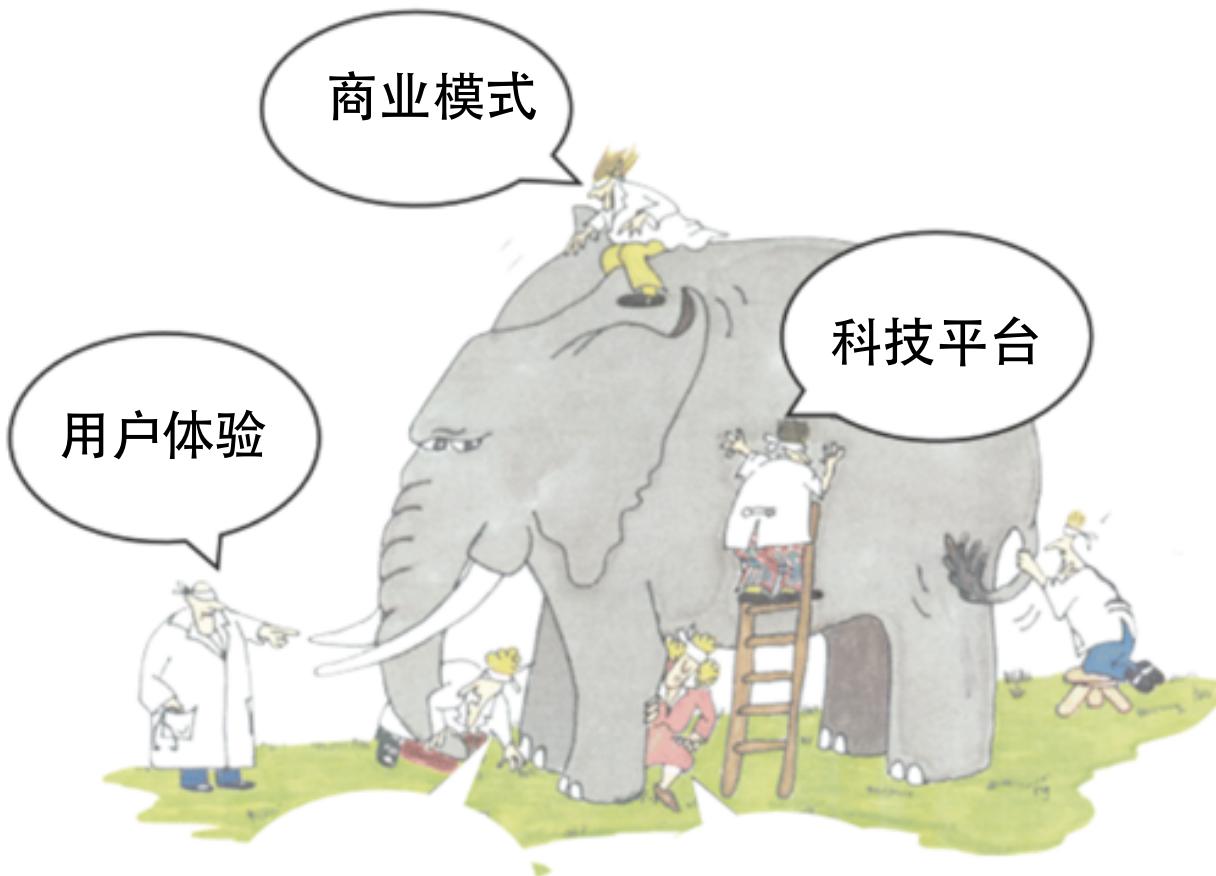


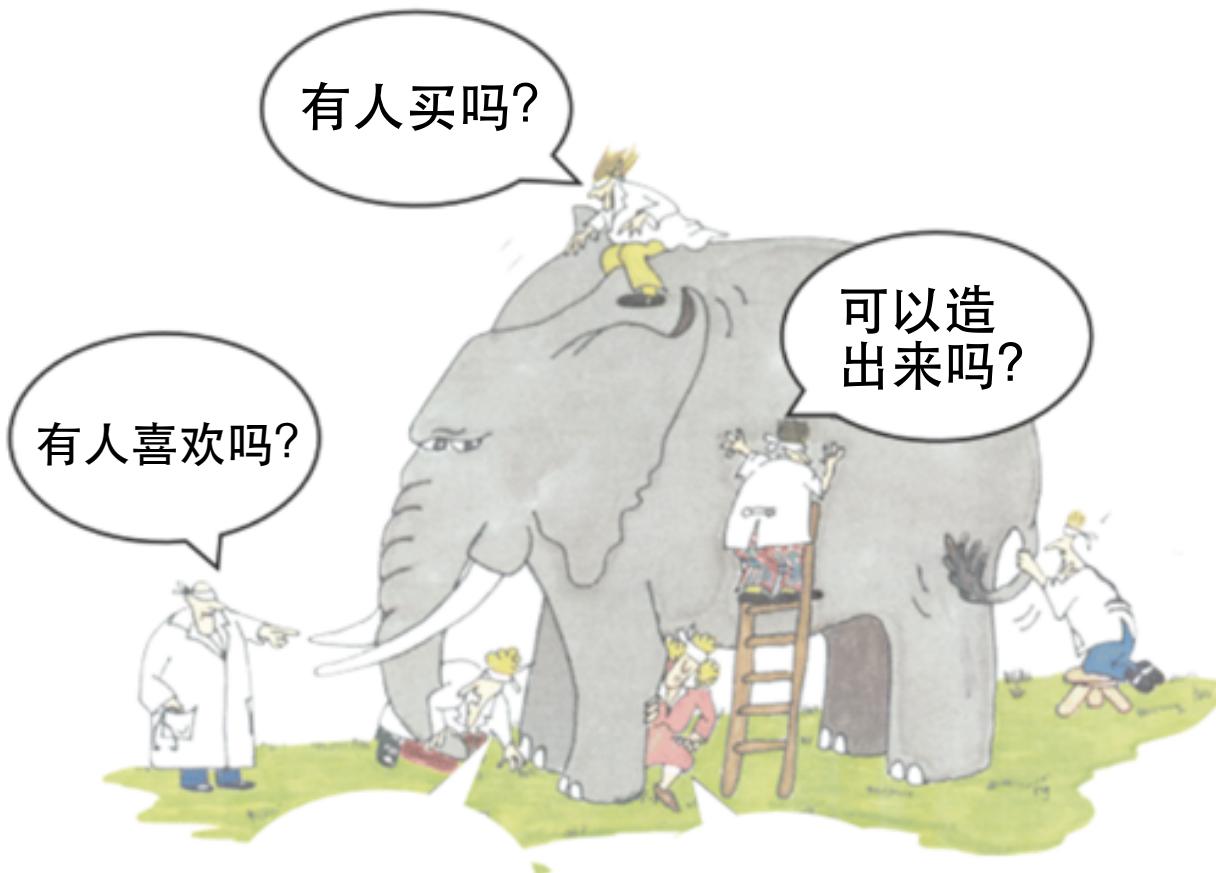


## Which Business Model and How to Test It ?

Adapted from Alex Osterwalter

**“No problem solution will be workable without the technical and organisational know-how to implement it. At the same time, mere know-how and solution without a customer willing to pay for it will also be unsustainable...”**





# 最小化可行产品 **MVP**(Minimum Viable Product)



# 最小化可行架构 MVA (Minimum Viable Architecture)



**甚麼是MVP?**

**“Minimum Viable Product”**

**最小化可行产品**

**那么MVA?**

**“Minimum Viable Architecture”**

**最小化可行架构**

# 为什么要從最小化可行产品开始?

用來验证用户体验

# 为何要有最小化可行架构?

以最低成本支援验证

# **How?**

# **Agile Development**

PROCESS

# AGILE DEVELOPMENT



EXTREME  
PROGRAMMING

ITERATION

PROCESS

# AGILE DEVELOPMENT



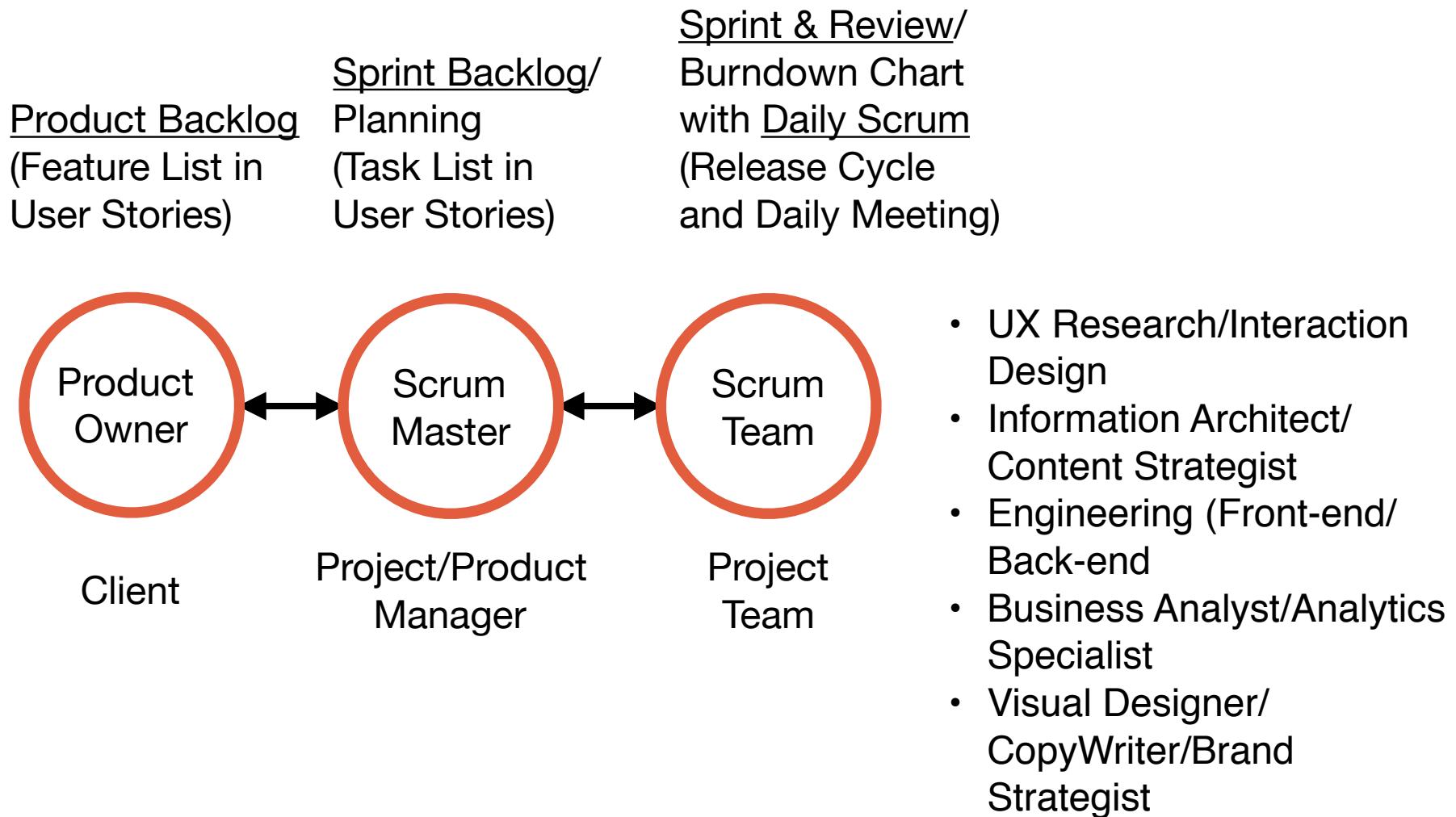
**“How does this work?”**

EXTREME  
PROGRAMMING

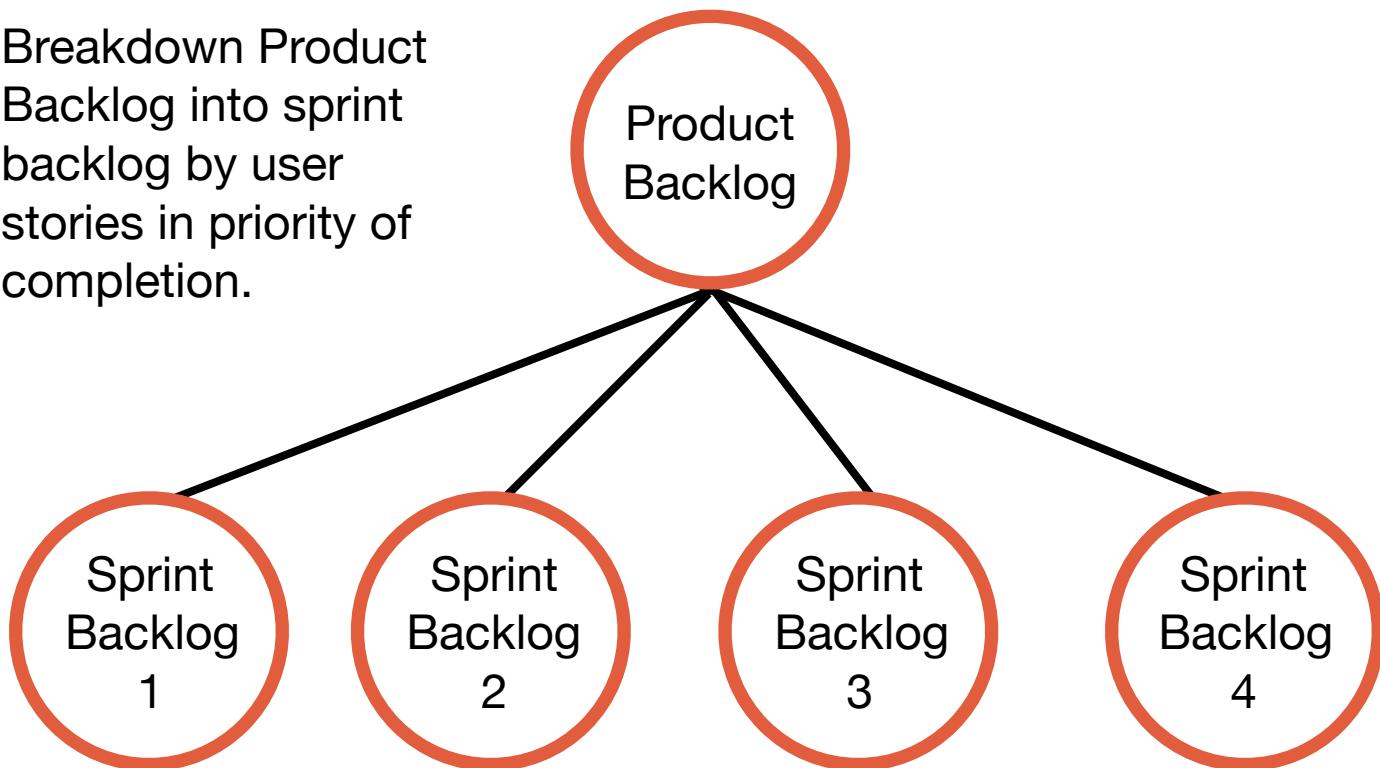


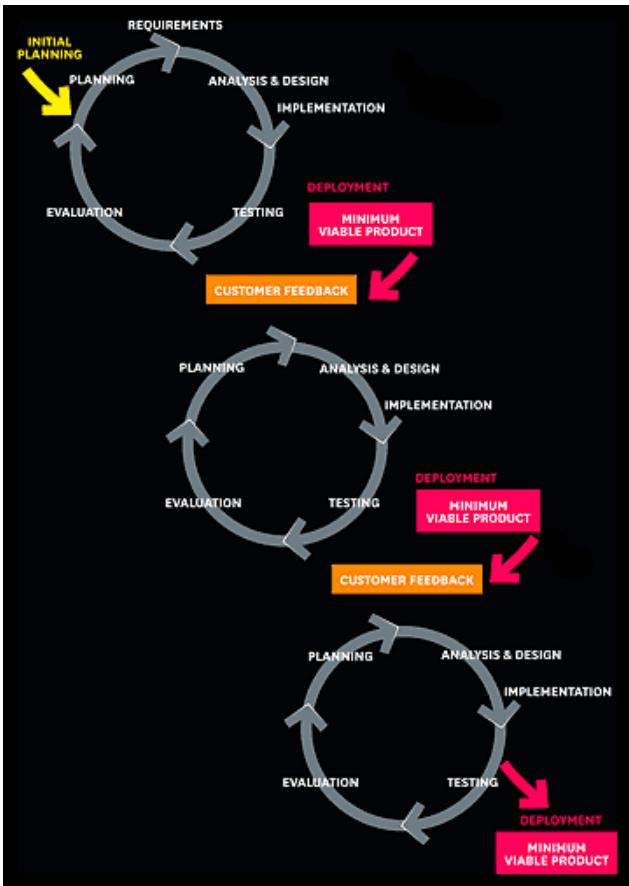
# **Practice Using SCRUM Methodology for Agile Software Development**





Breakdown Product Backlog into sprint backlog by user stories in priority of completion.





## Sprint Planning (Release Planning)



## Sprint Review (Release Review)

## Daily Scrum (Daily Meeting)

Each build-measure-learn cycle (ends with a review) is called a “sprint” usually measured in 1 to 3 weeks.

Source: “Why Lean Start-up Changes Everything”  
by Steve Blank



## 敏捷开发中用户故事地图(User Story Mapping)学习

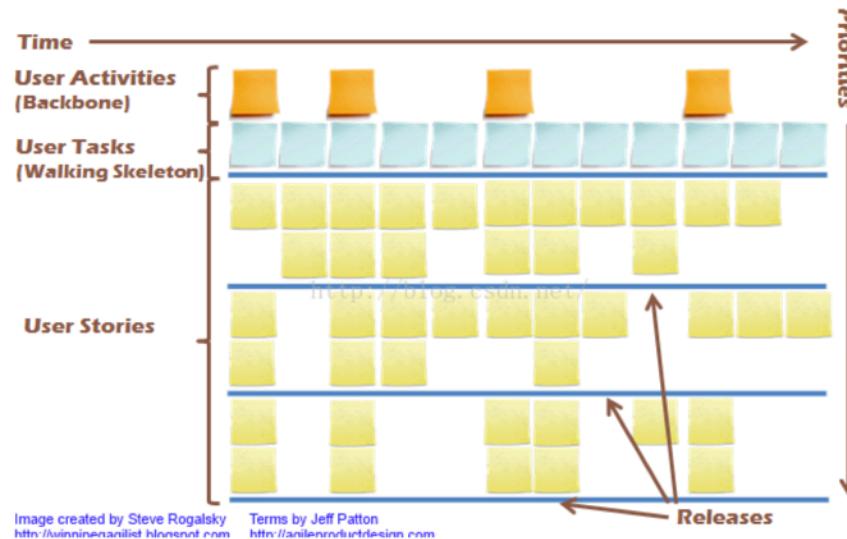
2016年09月25日 20:07:11 [佚名-2018\\_](#) 阅读数: 3485

### 1. 用户故事地图

User Story是敏捷开发和管理的核心，User Story是从用户的角度对系统的某个功能模块所作的简短描述。一个User Story描述了项目中的一个小功能，以及这个功能完成之后将会产生什么效果，或者说能为客户创造什么价值。

用户故事地图，英文名为User Story Mapping。用户故事地图已经成为敏捷需求规划中的一个流行方法。用户故事地图可以将你的 backlog（待办事项）变成一张二维地图，而不是传统的简单列表。

### 2. 用户故事地图编写模板







## Templates

Use these starter bases to get a jump start on your project. For inspiration from community-published bases, check out Universe.



Search templates

### CATEGORIES

- Featured
- Content production
- Creative
- Event Planning
- Everyday Life
- Groups, Clubs & Hobbies
- HR & Recruiting
- Legal
- Local Business
- Marketing
- Nonprofit
- Personal
- PR & Communications
- Product, design, and UX
- Project Management
- Publishing
- Real Estate

# Agile Workflow

[Use template](#)

#PRODUCT, DESIGN, AND UX

Even if you don't want to adopt the entire Agile manifesto, shifting your workflow to incorporate aspects of the Agile methodology—a small-a "agile" workflow—is a good idea for virtually any software development team.

By taking an agile approach to the development process, you let customer needs direct the features and improvements you're working on. By tailoring your product development to focus on the user, you waste less effort on irrelevant or unnecessary features than you do with traditional approaches.

Our blog post here walks you through how to use this template to apply agile processes to suit your team's needs. Hand it off to the scrum master to build your next sprint, schedule standups for your team, and make a note of stories for future sprints in one database that is easily shareable between your entire team. By tagging every potential feature or improvement in a project with the function it serves, the utility it provides to the customer, and its prioritization in the product, this base helps you gather steam and keep rolling into the next sprint after each completed project.

Backlog	Sprints	Standups	People
Main Story View	Hide fields	Filter	Group Sort ...
Storylines	Priority	Duration	Sprint Status

Agile Workflow

HELP ? 🔍 🔔 🚫

Backlog Sprints Standups People +

Main Story View Hide fields Filter Group Sort Color ...

	Story Name	Priority	Function	Sprint	Status	Use Case	Notes	story_id	Sprint Release Date
1	S1 : Remote login	★★★	Remote login	Mobile	In Progress	I can login to the app from where...		1	2/19/2018
2	S2 : SSO	★★★★★	SSO	Mobile	Done (Needs Review)	I can use my Google account to lo...		2	2/19/2018
3	S3 : More color options	★★★★★	More color options	UI	Not Started	I can customize the colors to my li...		3	3/14/2018
4	S4 : Customizable referral link	★★★	Customizable referral link		Not Started	I can send more effective referral ...		4	
5	S5 : Zapier integration	★★★★★	Zapier integration	Integrations	Finished	I can build 3rd-party utilities usin...		5	2/1/2018
6	S6 : Bitcoin support for transactio...	★★★	Bitcoin support for transactions	Integrations	Finished	I can spend my BTC with your app		6	2/1/2018
7	S7 : Night reading mode	★★★★★	Night reading mode	UI	Not Started	I don't hurt my eyes using the app...		7	3/14/2018
8	S16 : Multiple timezone support	★★	Multiple timezone support		Not Started	I can use the app from around the...		16	
9	S17 : Cleaned up mobile icon	★★★	Cleaned up mobile icon	UI	Not Started	I can enjoy looking at the app on ...		17	3/14/2018
10	S18 : Export as PDF option	★★★	Export as PDF option		Not Started	I can send content to my boss thr...		18	
11	S19 : Import feature	★★	Import feature		Not Started	I can add my own content to the a...		19	
12	S20 : Search feature	★★★★★	Search feature			I can more easily find the pages I ...		20	
13	S21 : Search bar on homepage	★★★★★	Search bar on homepage			I can more easily find the pages I ...		21	
+									

SHARE 🔍 🔔 🚫 BLOCKS 🔍 🔔 🚫

Story View

UI

UX

Design

Code

Test

Deployment

Analytics

Feedback

Help

## Agile Workflow

HELP ? 🔍 🔔 🚫



Backlog Sprints Standups People +

SHARE 🔍 🔔 BLOCKS

	Name	Status	Goal	Target Release Date	Stories	Notes	Standups	Start
1	Mobile	In Progress	Enhance functionality of app on mobile	2018-02-19	S2 : SSO   S1 : Remote login		Mobile Weekly 2-9-2018	N 2018
2	Integrations	Completed	Build out more 3rd-party integrations	2018-02-01	S6 : Bitcoin support for transactions   S5 : Zapier integration		Integrations Planning 1-24-	2018
3	UI	Planning	Improve look and feel of app	2018-03-14	S17 : Cleaned up mobile icon   S7 : Night reading mode   S3 : Mor			





## Agile Workflow

HELP ? 🔍 🔔

Backlog Sprints Standups People +

SHARE 🔍 BLOCKS



	Name	Type	Sprint	Date	Sprint Start	Sprint End	Owner	Participants	Meeting Notes	+
1	Mobile Retro 2-20-2018	Retro	Mobile	21/2/2... 7:00...	2/5/2018	2/19/2018	Putera Gunnar	Basit Nizar Priapos Paula		
2	Mobile Weekly 2-15-2018	Weekly	Mobile	15/2/2... 8:00...	2/5/2018	2/19/2018	Putera Gunnar	Basit Nizar Priapos Paula		
3	Mobile Weekly 2-9-2018	Weekly	Mobile	9/2/2... 10:0...	2/5/2018	2/19/2018	Putera Gunnar	Basit Nizar Priapos Paula		
4	Mobile Planning 2-5-2018	Planning	Mobile	6/2/2... 2:00...	2/5/2018	2/19/2018	Putera Gunnar	Basit Nizar Priapos Paula		
5	Integrations Retro 2-1-2018	Retro	Integrations	1/2/20... 10:0...	1/24/2018	2/1/2018	Priapos Paula	Anton Anatoli Putera Gunnar		
6	Integrations Weekly 1-29-2018	Weekly	Integrations	30/1/2... 2:00...	1/24/2018	2/1/2018	Priapos Paula	Anton Anatoli Putera Gunnar		
7	Integrations Planning 1-24-2018	Planning	Integrations	25/1/2... 4:30...	1/24/2018	2/1/2018	Priapos Paula	Anton Anatoli Putera Gunnar		
+										





## Agile Workflow

HELP ? 🔍 🔔



Backlog Sprints Standups People +

SHARE 🔍 BLOCKS



Grid view Hide fields Filter Group Sort Color ...

	Name	Stories	Sprints	Standups	+ +
1	Putera Gunnar	S1 : Remote login S2 : SSO	Mobile	Integrations Planning 1-24-	
2	Basit Nizar	S7 : Night reading mode S17 : Cleaned up mobile icon S3 : More color options		Mobile Planning 2-5-2018	
3	Priapos Paula	S4 : Customizable referral link S5 : Zapier integration	Integrations	Mobile Planning 2-5-2018	
4	Hamid Febe	S18 : Export as PDF option S19 : Import feature			
5	Anton Anatoli	S6 : Bitcoin support for transactions S16 : Multiple timezone support		Integrations Planning 1-24-	
+					



## Agile Workflow

HELP ? 🔍

SHARE ⏪ BLOCKS



Backlog Sprints Standups People

Kanban Stacked by Status Customize cards Filter Sort Color ...



Uncategorized

S20 : Search feature

★ PRIORITY



2 records



S21 : Search bar on home...

★ PRIORITY



1 record



Not Started

S3 : More color options

★ PRIORITY



S7 : Night reading mode

★ PRIORITY



S4 : Customizable refer...

★ PRIORITY



S17 : Cleaned up mobile...

★ PRIORITY



S18 : Export as PDF opti...

★ PRIORITY



S16 : Multiple timezone ...

★ PRIORITY



7 records



In Progress

S1 : Remote login

★ PRIORITY



1 record



Done (Needs Review)

S2 : SSO

★ PRIORITY



1 record



Finished

S5 : Zapier integration

★ PRIORITY



S6 : Bitcoin support for tr...

★ PRIORITY



2 records



+ New stack

# **Examples from Previous Students' Projects**

**Python Flask  
&  
SQL**

**Lean & Agile  
Development**

**9 Types of  
Business Model**

# **9 BUSINESS MODEL PATTERNS**

**Enterprise**

**SAAS**

**Usage-based**

**Subscription**

**Transactional**

**Marketplace**

**E-Commerce**

**Advertising**

**Hardware**

Anu Hariharan - Y Combinator

**Enterprise  
(e.g. IBM)**

- ▶ **Bookings (signed contract)**
- ▶ **Total Customers**
- ▶ **Revenue**

- SaaS (e.g.  
Zoom)**
- ▶ **MRR (Monthly Recurring Revenue)**
  - ▶ **ARR (Annual Recurring Revenue)**
  - ▶ **Gross MRR Churn**
  - ▶ **Paid CAC (Customer Acquisition  
Cost Per Customer )**

\*\*\* SaaS - Software as a Service

- Subscription (e.g. Netflix)**
  - ▶ **MRR (Monthly Recurring Revenue)**
  - ▶ **MRR CMGR (MRR Compound Monthly Growth Rate)**
  - ▶ **Gross User Churn**
  - ▶ **Paid CAC (Cost per customer)**

- Transactional** ▶ **Gross Transaction Volume (Total payment dollar volume)**
- ▶ **Net Revenue (Portion of GTV e.g. transaction fee)**
- ▶ **User Retention (Percent of customers who go on make at least one purchase in month 2)**
- ▶ **Paid CAC**

**Usage-Based**  
**(e.g. AWS Lambda )**

- ▶ **Monthly Revenue**
- ▶ **Revenue CMGR**
- ▶ **Revenue Retention**
- ▶ **Gross Margin**

- MarketPlace  
(e.g. Airbnb)** ▶ **GMV (Gross Merchandise Value)**
- ▶ **Net Revenue (Portion of GMV)**
- ▶ **Net Revenue CMGR**
- ▶ **User Retention (e.g. Track searching behaviour)**
- ▶ **Paid CAC (e.g. Acquire hosts)**

- E-Commerce (e.g. Taobao)**
  - ▶ **Monthly Revenue**
  - ▶ **Revenue CMGR**
  - ▶ **Gross Margin (\* Beware hidden costs)**
  - ▶ **Paid CAC (Cost per customer)**

## **Advertising (e.g. YouTube)**

- ▶ **Daily Active Users (\* Definition of being active is key)**
- ▶ **Monthly Active Users**
- ▶ **Percent Logged-In**

**Hardware**  
(e.g.  
**Xiaomi**)

- ▶ **Monthly Revenue**
- ▶ **Revenue CMGR**
- ▶ **Gross Margin**
- ▶ **Paid CAC**

# **ASSIGNMENT #2**

1. Retrieve selected data of your choice from your com5961 project's Airtable database through the Airtable REST API generated in your Flask app.
2. Build the needed data structure (e.g. list/dictionary) to be embedded (i.e. return render\_template(...) ) into the HTML template stored in the templates sub-folder.
3. Build up the template with the Flask template codes (e.g. {%, %}, or {{}}) and place the data structure into the HTML template with looping logic to produce the resulting HTML page for display in the browser.
4. You can make reference to the sample codes I shared.  
Please makes sure you change the app code, API key, table name, and any codes pertaining to your own Airtable setup.

- 
- 
- 
- 
- 
- 
5. Also make sure you remove the Python codes for running your program on Jupyter Notebook to make it work on PythonAnywhere.
6. Upload your Python codes and Flask HTML templates to PythonAnywhere, making sure the codes will go into the flask\_app.py file stored inside the mysite folder. Also, make sure the required HTML templates are uploaded into the templates sub-folder underneath the mysite folder.
7. First test your codes on Jupyter Notebook before uploading to PythonAnywhere. Remember to upload your Jupyter notebook codes and templates to Github (placed in your assignment #2 folder) so I can trace your logic and match that against the result shown on PythonAnywhere's website.

**THANK YOU FOR YOUR TIME!**