



Building a Start-up with Lean Python

Toni Abdala

#PyConAr2018 - 24/11/18

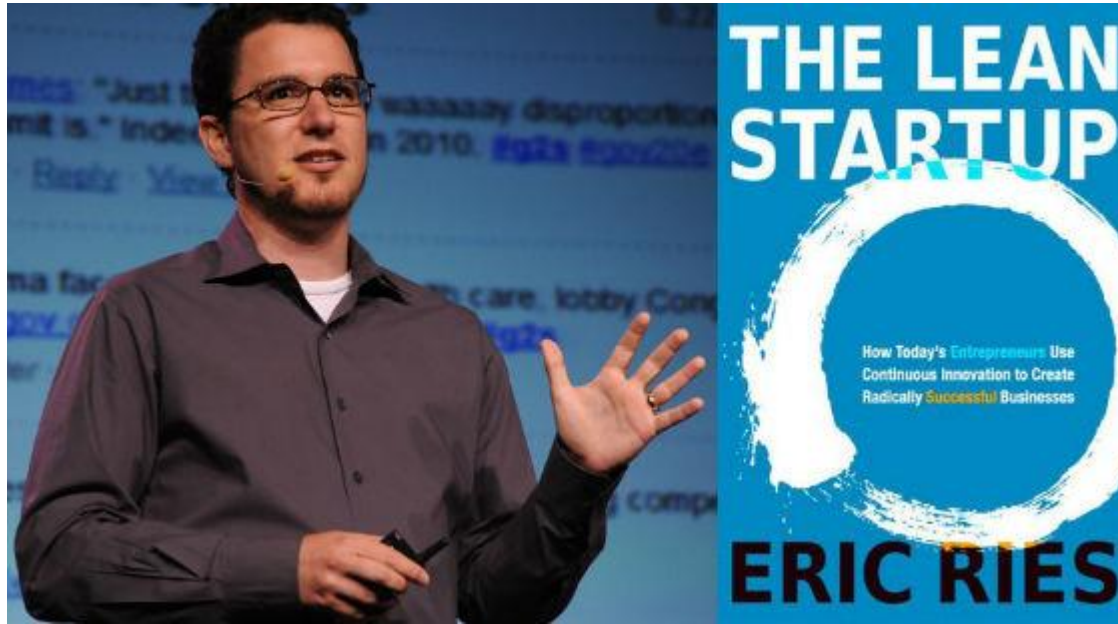
Presentación

- 8 años, emprendiendo. 6 años Full Life
- 2 Emprendimientos Agrotecnológicos.
- Formación inicial en Telecomunicaciones y Sistemas Embebidos.
- Padre de 2 chiques.

Disclaimer

- Formados por **equipos pequeños** de capacidades heterogéneas
- Mercado de **nicho**, pocos clientes, alto revenue.
- **Ramen Profitable.**
- Enfocados en **Data Science**, pero no hago foco en eso.
- No tocaremos en profundidad ninguna tecnología.
- **No voy a hablar de mi start-up** en detalle sino del proceso de desarrollo que llevamos.
- **No me alcanza el tiempo.**

¿Qué es Lean Start-Up?



Running Lean

RUNNING LEAN ASH MAURYA

ISBN 978-1-949-50517-8
 Sketched by @sachac Dec 28, 2012

Document your Plan A → Show this to people & ask

Riskiest aspect of plan?
 How have they overcome similar risks?
 How would they test these risks?
 Are there other people I should speak with?

Problem ①	Solution ④ Key metrics	Unique value Proposition ③	Under Advantage ⑤	Customer Segments ②
Cost structure ⑦	Revenue streams ⑥	Channels ⑧		

① ~15 minutes (p.20)
 ? OK to leave blanks
 Be concise
 Think in the present
 Be customer-centric

Draw several of these quickly so that you can rank them.

- Customer pain level
- Ease of reach
- Profit/gross margin
- Market size
- Technical feasibility

The 3 stages of a startup

Problem/solution fit → Product/market fit → Scale

Running Lean

Understand the problem → Define the solution → Validate qualitatively → Verify quantitatively

⚡? ⚡ ⚡ ⚡

p.77 has an email template for asking for interviews

Charge from day 1, collect on day 30

80% existing features, 20% new features

Work hacks → p.177

SaaS pricing (p.180) single plan, free trial, cents

Features → customer pull

Interview Guides

Problem Interview

Understand cust. worldview

- How do customers rank the top 3 problems?
- How do they solve them today?
- Is this a viable customer segment?

minutes

- Welcome
- Demographics
- Problem story
- Problem ranking
- Explore worldview
- Wrap up
- Document results

Solution Interview

Test 'demo' before building

- How do you identify early adopters?
- What is the minimum feature set?
- Will people pay? How much?

minutes

- Welcome
- Demographics
- Problem story
- Demo (what is created? what is needed? where?)
- Test pricing
- Wrap up
- Document results

MVP Interview

- landing page?
- activation flow?
- usability hot spots?
- MVP delivers unique value?
- more customers?
- do customers pay?

minutes

- Welcome
- Landing Page
- Pricing Page
- Signups/Activation
- Wrapping Up
- Document results

Cohort analysis

Acquisition → Activation → Revenue (by join date or other attributes)

Kanban tips

- Backlog
- In-progress
- Done
- Code
- Feature roll out
- Validate qualitatively
- Full rollout
- Verify quantitatively

Sales letter

LARGE PROMISE (your unique value proposition)

Connect w/ cust. problem

Generate interest/desire (solution)

Sketched by Sacha Choudhury
ExperiVis

Foco



MVP

Not like this....



1



2



3



4

Like this!



1



2



3



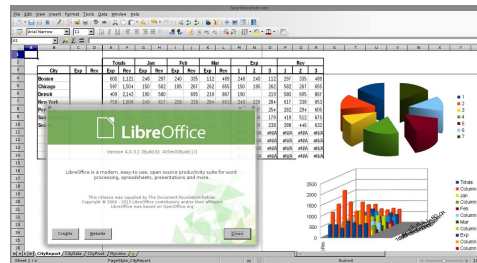
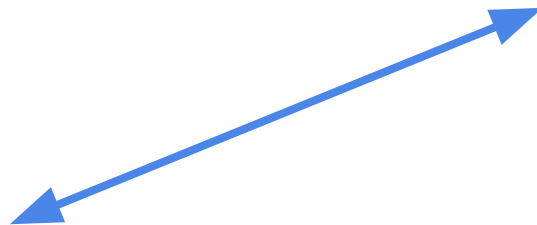
4



5

Henrik Kniberg

Mi primer MVP



Mi primer MVP

- UX ->UI
 - Comprar un template listo es simple y barato, toma tiempo elegir el correcto.
- Servers Económicos:
 - Linode o Digitalocean te permiten tener una VPS por 5.
- Métricas de Cliente:
 - django-tracking2
 - google-analitycs

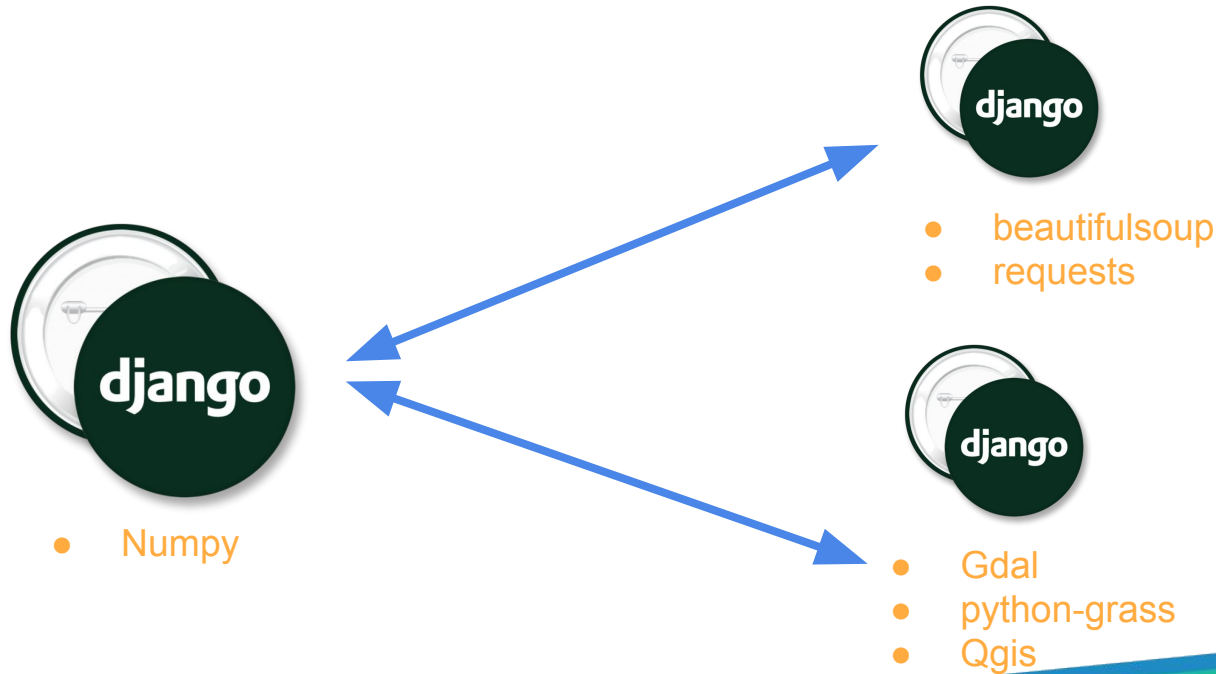
Evolucionando



Evolucionando

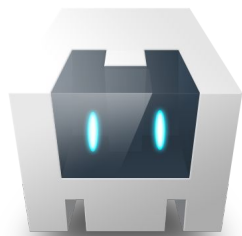
- Levantamos varios servers económicos.
- **Se contrataron expertos freelancers**, para realizar tareas específicas en el cálculo de datos especializados (Ej. Imágenes satelitales).
- Se comenzó a usar **Scrapers para relevar datos Climáticos** libres o con autorización explícita.
- Se armó un app de android que era solo **un webapp**.

Evolucionando



Una app super rápido

- Solo trabajando sobre su front responsive, para que se vea muy bien mobile.
- Creando una simple app con Cordova que solo levante esa web.
- Listo tenemos un app para subir al Store super rapido y validar que los clientes la usan.



APACHE
CORDOVA™

Python Sat. Ver 1.0



GRASSGIS

Bringing advanced geospatial technologies to the world

QGIS

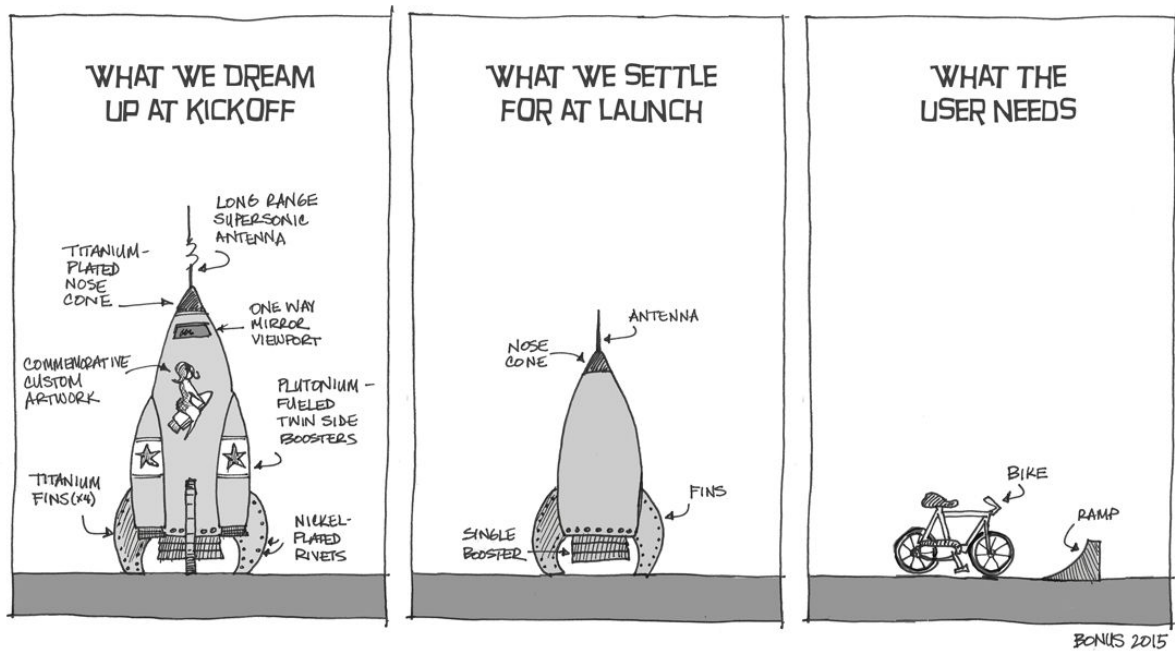


Validando Cuantitativamente

- Expansión de usuarios.
- Reuniones de Relevamiento: descontracturadas y re-preguntando de distintas maneras la UX

Por ser Techie NO tienes la razón

THE UX DESIGNER PARADOX



Y donde esta mi Ing. de Software



Y donde esta mi Ing. de Software

- Agile en equipos super pequeños no funciona completamente.
- Solo podemos poner énfasis en **testear** las partes más importantes, en nuestro caso Datos, **el modelo!**
- El código tiene que ser **simple sobre todo.**
- **Funcionalidades vs estabilidad:** Cada cosa que se nos ocurra poner puede ser usada en nuestra contra.

Codigo Simple?

```
from django.shortcuts import get_object_or_404
from books.models import Book, Publisher

def publisher_books_list(request, publisher_name):
    publisher = get_object_or_404(Publisher, name__iexact=publisher_name)
    book_list = Book.objects.filter(publisher=publisher)

    return render_to_response('books/books_by_publisher.html', {
        "book_list": book_list,
        "publisher": publisher,
    }, context_instance=RequestContext(request))
```

```
from django.shortcuts import get_object_or_404
from django.views.generic import ListView
from books.models import Book, Publisher
```

```
class PublisherBookListView(ListView):
```

```
    context_object_name = "book_list"
    template_name = "books/books_by_publisher.html",
```

```
    def get_queryset(self):
        self.publisher = get_object_or_404(Publisher, name__iexact=self.args[0])
        return Book.objects.filter(publisher=self.publisher)
```

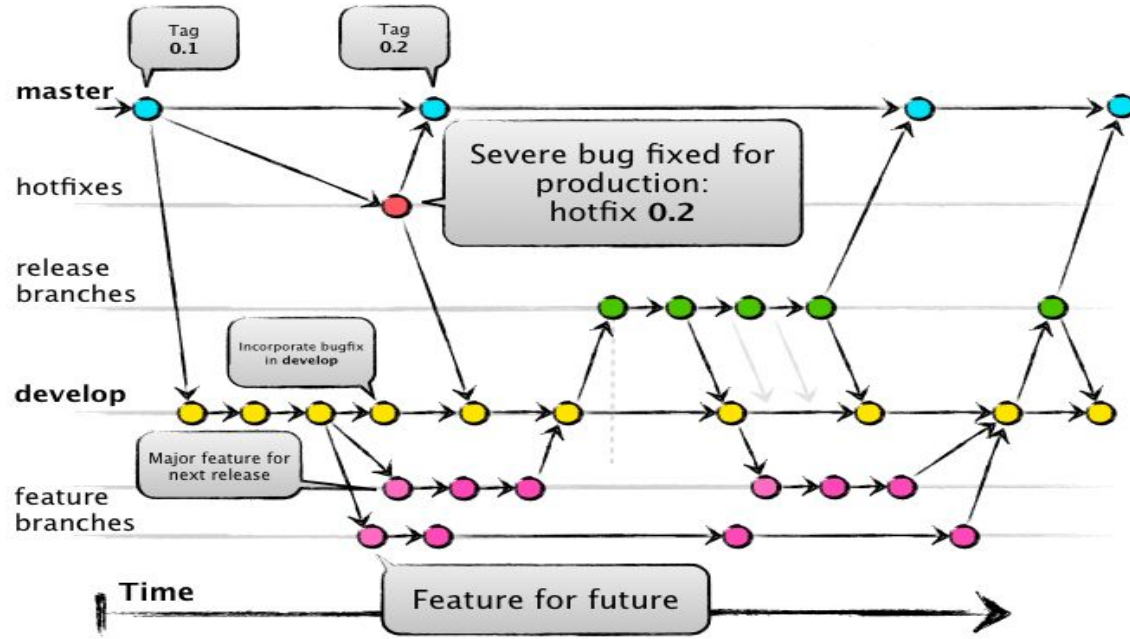
```
    def get_context_data(self, **kwargs):
        # Call the base implementation first to get a context
        context = super(PublisherBookListView, self).get_context_data(**kwargs)
        # Add in the publisher
        context['publisher'] = self.publisher
        return context
```

Y si Agile No, entonces Que?

The image shows a Kanban board with three columns: TO DO, In Progress, and Done. Each column contains several user stories with varying levels of completion.

- TO DO:**
 - As a client, I want to receive litigation rains loaded on the system so I can use my inputs correctly.
 - as an admin, I want to upload the sample when a case is created in order to ensure consistency.
 - Yo como usuario me gustaria añadir funciones de imagenes satelitales al Sistema.
 - As an installer, I want to have a quick installation guide in order to install the IoT device correctly.
- In Progress:**
 - As a user, I want to have better ndvi data from the base of historical values in order to have a more accurate forecast in the region.
 - As a technician, I want to have a verification plan in order to determine if the board is working correctly.
 - As a Kilimo member, I want to see information about clients use web site application (analytics) in order to take better decisions based on the client use cases.
- Done:**
 - Small Talk about NDVI.
 - As a Kilimo member, I want to see what is the performance of the forecast prediction in order to have data for improvement.
 - As a Kilimo member, I want to know what is the performance of the recommendation algorithm in order to have data to improve it.
 - Solve a problem with turn off feature.

Git Flow

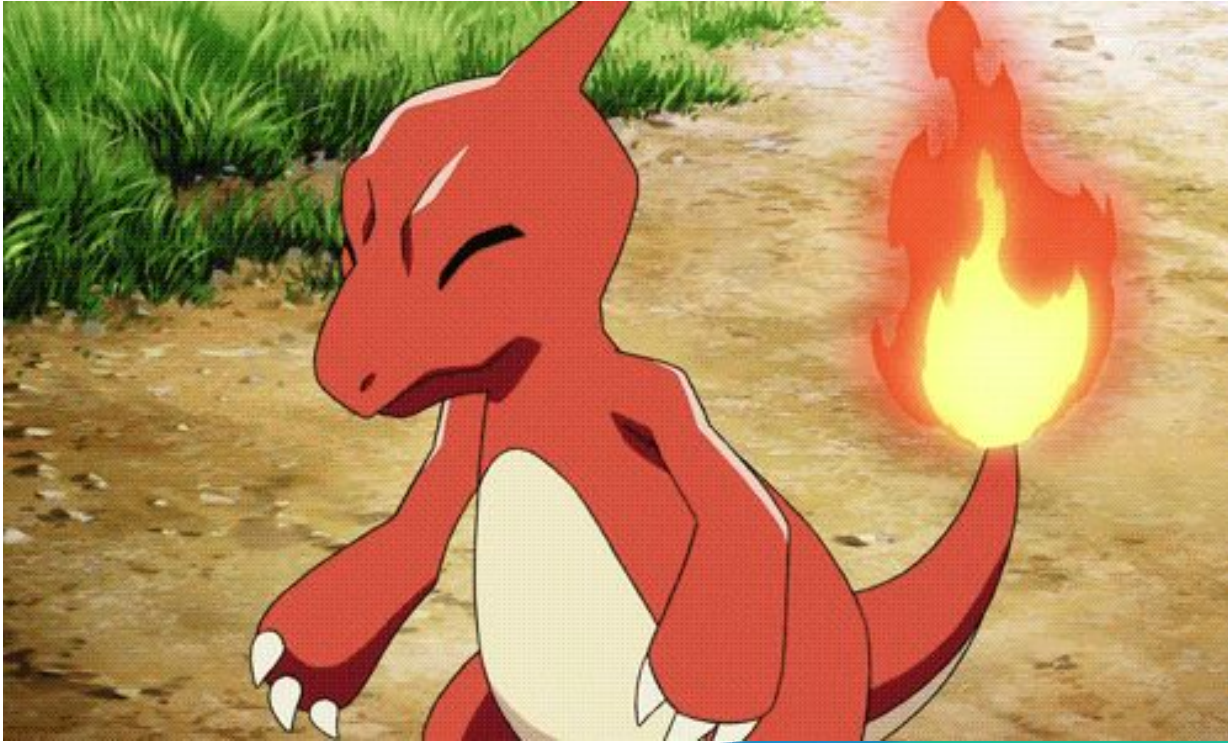


Trello Flow

The image shows a Trello board with five columns representing different stages of a workflow:

- TO DO:** Contains three task cards. The first card is about a client wanting to improve a system. The second is about an admin wanting to load sample data. The third is about a user wanting to add satellite images. Each card has a progress indicator (0/1) and a '+ Añada otra tarjeta' button.
- In Progress:** Contains two task cards. The first is about a user wanting to improve a system. The second is about a technician wanting to verify a plan. Each card has a progress indicator (1/3) and a '+ Añada otra tarjeta' button.
- Done:** Contains three task cards. The first is about a user wanting to improve a system. The second is about a Kilimo member wanting to see what is the performance of the forecast program. The third is about a Kilimo member wanting to know what is the performance of the recommendation algorithm. Each card has a progress indicator (3/3, 4/4) and a '+ Añada otra tarjeta' button.
- Staging:** A grey column with a '+ Añada una tarjeta' button.
- Production:** Contains three task cards. The first is about a user wanting to take samples. The second is about an admin wanting to get weather information. The third is about revising a merge request. Each card has a progress indicator (1/1) and a '+ Añada otra tarjeta' button.

Volver a Evolucionar



Outsourcing?

- Tienes que decidir **cuál va a ser el core de tu equipo dev.**
- Buscar Freelancers o Consultoras, expertas en cada tema.
 - Infra
 - Front - App

Volver a Evolucionar

- Infra en Cloud, Amazon, Azure o Heroku. Consejo hay crédito gratuito pero hay que saber cuándo tomarlo.
- Todo con **API REST** y Oauth
- Todos los datos en CLOUD S3 o Blobs. Hay plugins que permite integrarlos con los modelos de Django.
 - django-storages
- Front deployado en Angular y APP en Ionic con Base de datos Local.

APIS y mas APIS con DRF

django

REST

framework

Python Sat. Ver 2.0

- Rasterio
- Shapely
- Fiona
- Pillow

APPS y Front mas rapidas



Y ahora mi Ing. en Software

- Ahora si Agile Scrum, usamos **Taiga** para llevarlo.
- Ningún pull request sin sus Tests.
- **Métricas de Funcionamiento:** Dashboards y más dashboards, KlipFolio, Grafana.
- Integración continua.

Espartanos de las Features



¿Preguntas?



¡MUCHAS GRACIAS!



www.kilimoagtech.com.ar



[@toniabdala](https://twitter.com/toniabdala)

toni@kilimoagtech.com