



Quispe Barriga, Sebastian Andres  
Rodriguez Moscoso, Mauricio Antonio  
Carlos Acosta , Rodrigo Dion

# Índice general

<b>1. Introducción</b>	<b>1</b>
<b>2. Objetivo del proyecto</b>	<b>2</b>
<b>3. Product Backlog</b>	<b>3</b>
3.1. Creacion de los debe: . . . . .	3
3.2. Historias . . . . .	3
3.2.1. H01 - Interfaz de usuario . . . . .	3
3.2.2. H02 - Diseño de Arquitectura . . . . .	3
3.2.3. H03 - Registro de Usuarios . . . . .	4
3.2.4. H04 - Conexión blockchain . . . . .	4
3.3. Planning Poker Vote . . . . .	4
<b>4. Sprints</b>	<b>5</b>
4.1. Sprint 1 . . . . .	5
4.1.1. Sprint Planning . . . . .	5
4.1.2. Sprint Retrospective . . . . .	5
4.1.3. Sprint Review . . . . .	7
4.1.4. Sprint MVP . . . . .	7
4.2. Sprint 2 . . . . .	8
4.2.1. Sprint Planning . . . . .	8
4.2.2. Sprint Retrospective . . . . .	8
4.2.3. Sprint Review . . . . .	10
4.2.4. Sprint MVP . . . . .	11
<b>5. Conclusiones y recomendaciones</b>	<b>12</b>
<b>A. Aplicación Web</b>	<b>13</b>
<b>B. Evidencia de las presentaciones</b>	<b>15</b>

# 1. Introducción

El Perú, al igual que todos los países sufre gravemente de uno de los mayores males que afectan a las sociedades hoy en día, la pobreza y la desigualdad social. Sabemos que el PBI aumenta todos los años, y el índice de pobreza disminuye (bajo medidas sumamente polémicas) pero siempre existe un gran porcentaje de la población que no es capaz de acceder a los "beneficios implícitos" de los incrementos porcentuales en la economía peruana. Por más que lo parezca, el Perú no es un país ajeno a esta realidad, ni ignora que existe. De hecho, en los últimos años la cantidad de iniciativas que buscan mejorar el contexto social y económico de las poblaciones en riesgo no ha hecho más que aumentar. Sin embargo, ¿son realmente efectivas estas iniciativas? Es cierto que existen múltiples casos de éxito, pero también hay muchos otros que muestran la falta de trazabilidad sobre estas poblaciones y otros más que representan la viveza y el criollismo en su más bajo nivel por la corrupción que los envuelve.

Semejantes problemas no pueden ser dejados de lado, un claro ejemplo del efecto que generan se pudo apreciar en el reciente apoyo económico que el ejecutivo planteó para todos los afectados por las medidas efectuadas durante la cuarentena. Este último fue duramente criticado por utilizar datos no actualizados como base para medir el estado financiero de las personas, así como por la baja fiabilidad del sistema. Lo primero causó que muchas personas que necesitaban de este apoyo de forma urgente no fueran seleccionadas, mientras que otras que no, si lo recibieron. Lo segundo generó que muchas personas recibieran el bono en varias ocasiones, falsificando documentos o simplemente por errores del sistema.

Ante este contexto, nos propusimos encontrar un método en el que los problemas de trazabilidad y de fiabilidad pudieran ser solucionados sin comprometer la lógica ya existente. Es así como a lo largo del informe se les explicará el proceso de desarrollo de nuestro producto, el cual hemos nombrado 'Yupay'. Durante este proceso se verán reflejadas las características que convierten a Yupay en una herramienta de gran utilidad al momento de evitar los problemas ya mencionados.

## 2. Objetivo del proyecto

Una de las principales metas de Yupay es brindarle visibilidad económica y social a poblaciones emergentes, que por necesidad trabajan en la informalidad con la finalidad de ganarse el pan de cada día. Estas poblaciones, se ven forzadas a entrar al mercado laboral, por lo que el gobierno pierde el trazo y registro de toda su actividad económica. Debido a esto, el acceso a programas de ayuda social y/o económico brindados por el ejecutivo se ven tremadamente afectados ya que, se tiene que registrar manualmente el acceso a los programas de ayuda, creando solo más barreras que impiden que la ayuda sea instantánea. A grandes rasgos, podríamos decir que tenemos 3 grandes objetivos.

- Trazar a todas las personas que se encuentren en riesgo, con mayor énfasis en quienes se encuentran en la informalidad.
- Centralizar la información que se registre pero descentralizar su proveniencia, a fin de brindarle facilidades al estado para analizar el registro de poblaciones en riesgo.
- Asegurar la inmutabilidad de los datos para evitar casos de corrupción.
- Probar de forma fehaciente el estado de una persona que se encuentra en zona de riesgo.

## 3. Product Backlog

### 3.1. Creacion de los debe:

Los debes se utilizan como base para la creación de las historias, las cuales tienen su fundamento y razón en estas. Nuestro proyecto presenta los siguientes debes:

- Debe representar transparencia (H1,H3)
- Debe segmentar a los usuarios (H1,H3,H4)
- Debe organizar a los usuarios registrados (H3)
- Debe emitir estados de usuarios en tiempo real(H4)
- Debe ser funcional y de fácil acceso (H1,H2)
- Debe administrar datos (H2)
- Debe recolectar información de manera constante (H3,H4)

### 3.2. Historias

#### 3.2.1. H01 - Interfaz de usuario

Cuadro 3.1: H01 - Interfaz de usuario

<b>Como:</b> Usuario	<b>Quiero:</b> Permitir que el usuario pueda utilizar de forma segura y efectiva el estampado correcto de la blockchain.
<b>Para:</b> Lograr el uso efectivo y seguro de la aplicación	<b>Condiciones:</b> - La interfaz debe ser interactiva y de fácil acceso - Ofrecer certificados a las personas que ingresen o se graduen de los programas de ayuda.

#### 3.2.2. H02 - Diseño de Arquitectura

Cuadro 3.2: H02 - Diseño de arquitectura

<b>Como:</b> Administrador de sistema	<b>Quiero:</b> Que encuentre y utilice los datos ingresados en el sistema con facilidad y haga manejo de los mismos.
<b>Para:</b> La correcta realización y ejecución de el diseño planteado para el proyecto.	<b>Condiciones:</b> - La arquitectura debe cumplir con la especificaciones planteadas en el desarrollo de la aplicación. - La arquitectura debe mantener la información inmutable y estar alineada a los objetivos del proyecto.

### 3.2.3. H03 - Registro de Usuarios

Cuadro 3.3: H03-Registro de usuarios

<b>Como:</b> Administrador de los nodos de recolección	<b>Quiero:</b> Que pueda ingresar la información a la red sin complicaciones.
<b>Para:</b> Poder obtener la mayor cantidad de información acerca de las personas en situaciones de riesgo	<b>Condiciones:</b> - El registro de usuario debe cumplir con las condiciones establecidas en un inicio por la aplicación. - Cada registro de usuario debe estar mencionado dentro de la web y posteriormente establecer su estado actual.

### 3.2.4. H04 - Conexión blockchain

Cuadro 3.4: H04 - Conexión blockchain

<b>Como:</b> Administrador de Yupay	<b>Quiero:</b> Que la información pueda ser representada en la arquitectura de la forma esperada
<b>Para:</b> Poder mantener un registro constante e inmutable de todas las personas presentes en el sistema	<b>Condiciones:</b> - El registro debe de ser preciso para su manejo en el sistema. - Debe facilitar el manejo de la información encontrada en la blockchain para los propósitos del proyecto.

## 3.3. Planning Poker Vote

Cuadro 3.5: Planning Poker Vote

Historias	Planning Poker
Interfaz de usuario	8
Diseño de arquitectura	20
Registro de usuarios	13
Conexion blockchain	20
<b>Total:</b>	<b>61</b>

## 4. Sprints

Para nuestro proyecto con el tiempo dado para el avance del proyecto, se dividió el Scrum en dos sprints. Estos serán explicados en detalle en las secciones inferiores.

### 4.1. Sprint 1

Para el primer sprint contamos con el periodo de trabajo de dos semanas, que fueron dados desde la semana diez hasta el inicio de la semana doce del periodo académico 2020-1.

#### 4.1.1. Sprint Planning

Como objetivo para este Sprint queríamos retro alimentar y recolectar la información de las personas mediante varios nodos, a fin de tener un registro actualizado con información real y precisa acerca de su situación económico y/o social. Para ello, decidimos hacer uso de la Blockchain en su forma pública. Cómo lograríamos esto, en un principio decidimos plantearnos los siguientes retos:

- Crear una red de nodos
- Definir una base de datos basada en la población peruana
- Determinar los nodos de recolección, seccionados en principales y secundarios
- Delimitar factores para sectorizar a la población

#### 4.1.2. Sprint Retrospective

##### ■ Semana 10:

En esta primera semana de trabajo, comenzamos buscando información para Definir una base de datos de nuestra población objetiva, dónde encontramos las variables consideradas por RENIEC para poder definir los valores de identificación en la población objetiva. Además, esta información se vería retroalimentada y confirmada por los recolectado por los nodos. Por otra parte, nuestro reto de creación de una red de nodos fue modificado a utilizar la red de Stamping.io como intermediario para nuestras operaciones. En esta, logramos estampar en la blockchain y definimos un sistema que va a ser utilizado para nuestros requerimientos.

■ **Semana 11:**

En esta semana diseñamos un primer esquema para nuestro sistema blockchain.

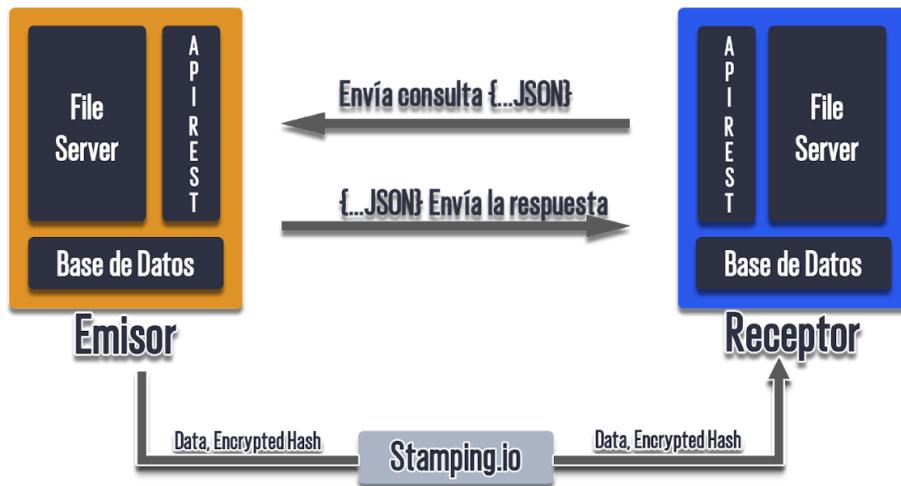


Figura 4.1: Esquema inicial de sistema blockchain

También, determinamos lo que para nosotros, en cuanto a la aplicación a corto plazo del proyecto, consideraremos los nodos principales y secundarios de recolección de data. Véanse a continuación:

- Nodos principales:
  - Empresas proveedoras de agua
  - Empresas proveedoras de servicio eléctrico
  - Empresas de telecomunicación hogar
  - Empresas de telecomunicación móvil
  - Empresas de abastecimiento de primera necesidad
  - Servicios de salud y/o seguro privado
  - Deudas e hipotecas bancarias
  - Propiedades y/o inmuebles
  - RENIEC
- Nodos secundarios:
  - Centros comerciales
  - Restaurantes y servicios de fast food
  - Compras por internet
  - Sistema de pensiones privado o público

Por último, realizamos una investigación acerca de los factores aplicados en la actualidad para clasificar a la población dónde se aplica la *Línea de Pobreza* (LP) como factor principal. Si bien este no es el mejor método, será utilizado como base para el desarrollo de un algoritmo, que clasifique a la población en conjunto de la información recolectada o estampada por los nodos mencionados anteriormente.

#### 4.1.3. Sprint Review

En la siguiente imagen se puede apreciar el review de nuestro avance en conjunto de la previsión de las próximas actividades para nuestro próximo sprint.

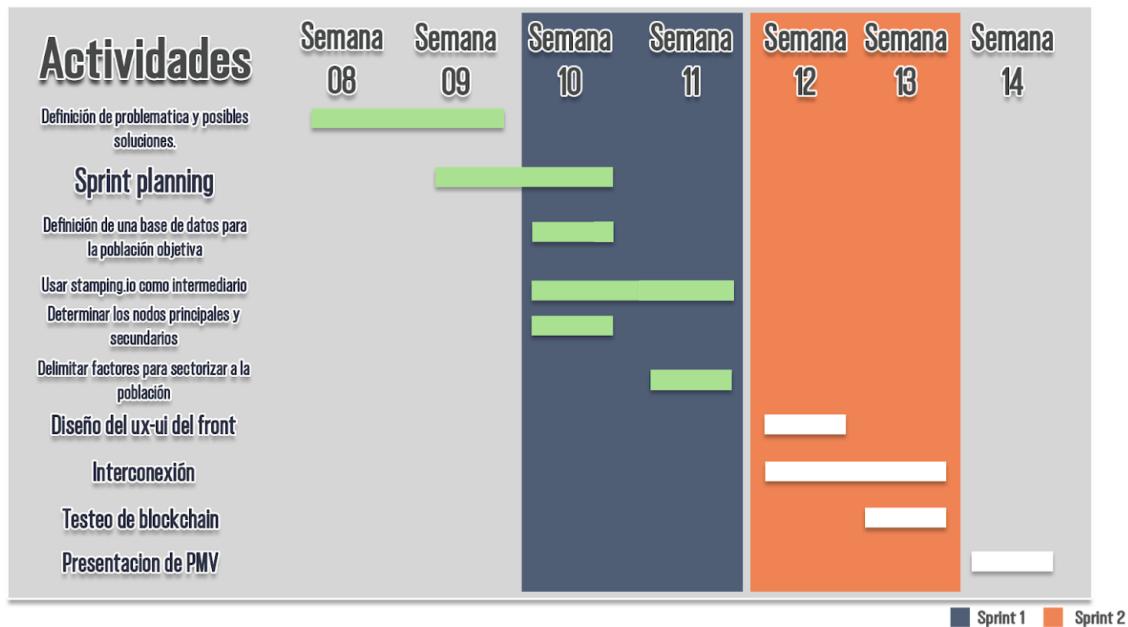


Figura 4.2: Sprint Review

#### 4.1.4. Sprint MVP

Como producto final de esta primera etapa de trabajo, definimos la arquitectura final de nuestro proyecto donde implementaremos todo lo investigado y avanzado en este sprint.

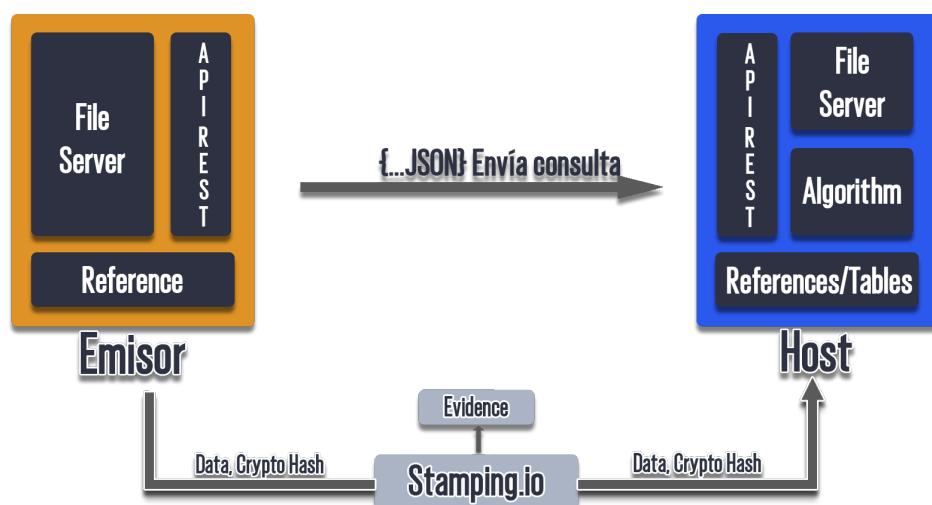


Figura 4.3: Sprint MVP

## 4.2. Sprint 2

Para este segundo y ultimo sprint desarrollado utilizamos un trabajo de tres semanas hasta la presentación del trabajo final,el tiempo fue desde la semana doce hasta la quince del periodo académico 2020-1.

### 4.2.1. Sprint Planning

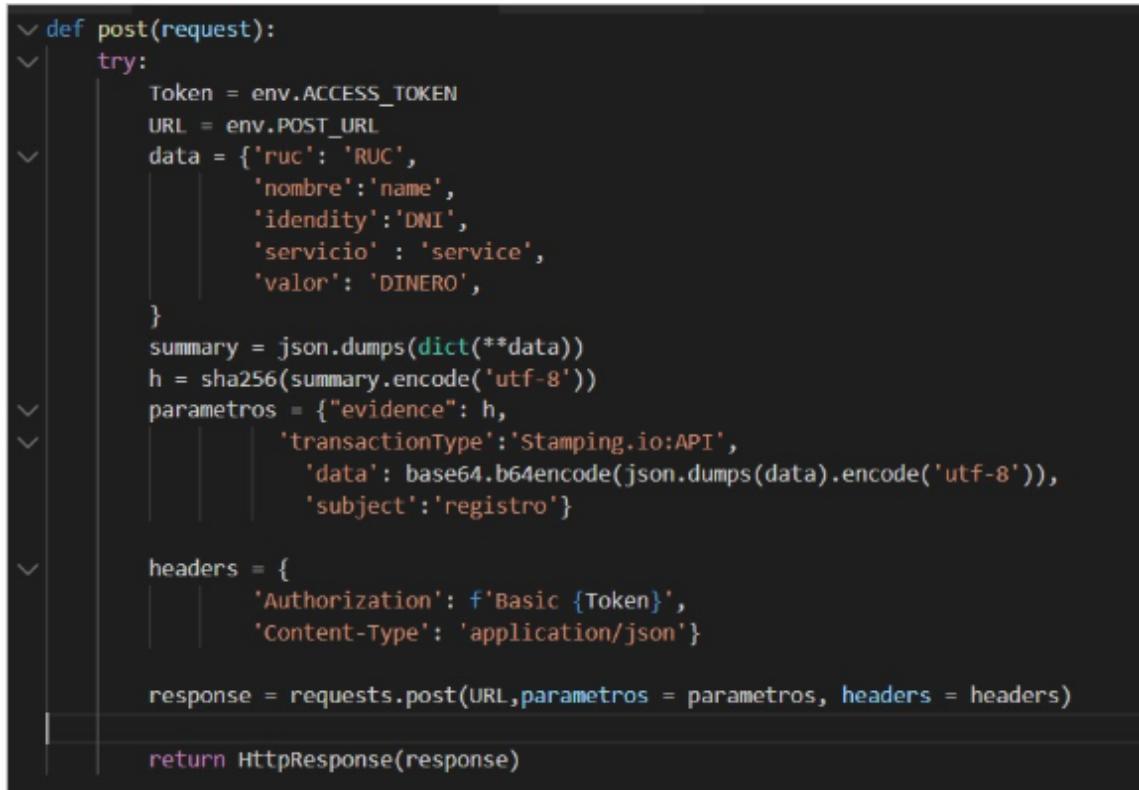
En este sprint nos dedicamos a aplicar los conocimientos aprendidos en el primer sprint, con lo comenzamos el desarrollo de nuestra aplicación basada en blockchain. Para ello, repartimos los logros en tres puntos importantes a desarrollar a lo largo de las tres semanas de trabajo:

- Development de la arquitectura planteada
- Front end two-sided
- Testeo de producto final

### 4.2.2. Sprint Retrospective

#### ■ Semana 12:

En esta semana de trabajo, procedimos a darle una estructura inicial a los *POST requests*, basado en los valores que considerábamos claves para la información que fuese a ser representada en el sistema. Todo esto en base a lo aprendido acerca del uso del API de *Stamping.io*. Véase en la imagen a continuación:



```
def post(request):
    try:
        Token = env.ACCESS_TOKEN
        URL = env.POST_URL
        data = {'ruc': 'RUC',
                'nombre': 'name',
                'idendity': 'DNI',
                'servicio': 'service',
                'valor': 'DINERO'}
        summary = json.dumps(dict(**data))
        h = sha256(summary.encode('utf-8'))
        parametros = {"evidence": h,
                      'transactionType': 'Stamping.io:API',
                      'data': base64.b64encode(json.dumps(data).encode('utf-8')),
                      'subject': 'registro'}
        headers = {
            'Authorization': f'Basic {Token}',
            'Content-Type': 'application/json'}
        response = requests.post(URL, parametros=parametros, headers=headers)
    return HttpResponse(response)
```

Figura 4.4: Posts - Requests

■ **Semana 13:**

Para esta semana de trabajo, terminamos de diseñar e implementar lo básico para que nuestro MVP del primer sprint (Figura 4.3) pueda recibir información sobre los nodos con la data requerida. Para facilitarle a los administradores de nodos, se diseño un front-end que contenía la información, objetivos y participantes de Yupay, dónde fue agregado un formulario para agilizar el proceso de registro de personas, en este ira la misma información que fue diseñada para los post requests. Véase a continuación:



Figura 4.5: Pagina principal

A screenshot of the Yupay registration form. The title "YUPAY" is centered at the top in a bold, black, uppercase sans-serif font. The form consists of several input fields arranged in a grid-like structure. The first row contains a field labeled "RUC DE LA EMPRESA \*". Below it, the letters "RUC" are entered. The second row contains two fields: "INGRESE NOMBRE DE LA EMPRESA\*" on the left and "DNI\*" on the right. Below these, the names "NOMBRE" and "INGRESA EL NUMERO DE DNI" are listed. The third row contains a field labeled "SERVICIO OFRECIDO \*". Below it, the text "Empresa de telefonía" is entered. To the right of this field is a small square checkbox. The fourth row contains a field labeled "DINERO EN SOLES \*". Below it is a horizontal progress bar with a blue slider, which is positioned near the beginning.

Figura 4.6: Formulario de registro de data

- **Semana 14:** En esta última semana de trabajo, se implemento el módulo de referencia de personas registradas, dónde si una persona al ser registrada (en base a su DNI) no se encuentra en el sistema, se crea la primera transacción dónde se almacena el root del arbol de posteos. En caso contrario, si una persona ya se encuentra registrada en la blockchain todos los requests siguientes se referenciaran al root, en báse al DNI linkeado al primer **trx\_id** generado.

```
"code":200,"message":"ok","txid": "fa14517de38770ba5e2bf909d448448+06","result":  
{"data": "eyJ1VScmVzIiwibG9JU2NjQ3lWnvLkHBlhuPfZS16ICJUZXN0InVgZxZQWfQoIAmtAxMDewMTA1CaiC2VymdlZS16ICJvdHlVlwglMfr3VudC16IClMyCJ9","url": "", "userId": "kNNNDuPcrVdfqKxbHD1Hcs  
"data": "eyJ1VScmVzIiwibG9JU2NjQ3lWnvLkHBlhuPfZS16ICJUZXN0InVgZxZQWfQoIAmtAxMDewMTA1CaiC2VymdlZS16ICJvdHlVlwglMfr3VudC16IClMyCJ9","ethTxId": "0x1dab8a28683d96e6907412317ac  
"result": "hash": "d982f46529268b41one15b0923b2fa1e798c9b5251a891ab509479","id": "479890","receipts":  
{"treeNumber": "", "hashTree": "", "blockNumber": "", "hashBlock": "", "hashLeft": "", "hashRight": "", "idFrom": "", "idTo": "", "timestamp_block": "", "base58": "", "ripenedAt": "160", "sha_1": "", "sha_2": "", "evidenChain": "", "btc": "", "eth": "", "tsa": "", "references": []}
```

Figura 4.7: Root de usuario recien registrado

```
{"code": "200", "message": "ok", "txid": "9e1c8ab55abb9d4259f8a71ba3a6e7", "result": "{'data': {'evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfNbHhVkiwglnv1': 'ZKXfjWQoIAiNzYyNz1Mz1LCAiC2VydmljZS16ICfJFxByZXnhIgrUJHnbhBhvKliwgImfb3VudC16IClxMjMwIn0="}, 'url': 'https://evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfNbHhVkiwglnv1'}, 'data': {'evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfNbHhVkiwglnv1': 'ZKXfjWQoIAiNzYyNz1Mz1LCAiC2VydmljZS16ICfJFxByZXnhIgrUJHnbhBhvKliwgImfb3VudC16IClxMjMwIn0="}, 'url': 'https://evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfNbHhVkiwglnv1'}, 'result': {'hash': '6e61f7610e8cb2512f12d96d9dd3fe4469e2b10abbh3b470196546', 'id': '470971', 'receipt': 'receipt'}, 'treeNumber': '1057232', 'hasTree': '751eb3edc0d5aa20ada48b6d50561eed4f242d20ad4251350d0', 'blockNumber': '19535', 'hashBlock': '7917364289d848218b7d314b4d0f305850785201cae07edb247eb6b8f86714', 'version': '17', 'size': '23.00:02', 'baseSize': 'mnWaQWSRySrtAksZWw2j8j9futdywUYHa', 'ripened160': '6F4CC498BEDCA399BB16ADE7E4D4307909A0D07B87', 'sha_1': '', 'sha_2': '64F9AD718B0D5EB4520A1E84D1DC09651CE1FC18F5BA3CE', 'references': '[{"subject": "Added", "data": "evySVMUOmOiaNDEwMjE2Nz1MC1Cjd2b1wYW5X52bhuWtUiOaU2kvYw4d1YSBTQUMlCAiAxXnleP9zC16IC1Nj3NjUzMiljCaJzX2jWaNljogJlkVtcHje2EgZGuc2VydmljWa9zGhe2j3mLCAtYw1", "subject": "Added", "data": "evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfJwvXNzbmdvIwglnvzZKXfjWQoIAiNzYyNz1Mz1LCAiC2VydmljZS16ICfJFxByZXnhIgrUJHnbhBhvKliwgImfb3VudC16IClxMjC9", "subject": "Added", "data": "evySVMUOmOiaNzQsNzE3MtLCAiQ2q7rcGfUeV9uYw1ljogJnfhEcNz28LICaxDnleP9zC16IC1Nj3NjUzMiljCaJzX2jWaNljogJm9cm5LCAiYw1vdW50jogJlwIn0=", "txid": "56e6ab7e19df8d7d067a902", "subject": "Added", "data": "evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfNbHhVkiwglnv1"}, {"subject": "Added", "data": "evySVMUOmOiaMjAyMDIwIwgIkNvbXhbhnlfbzS16ICfNbHhVkiwglnv1"}]
```

Figura 4.8: Root de usuario ya registrado

### 4.2.3. Sprint Review

En resumen, logramos inmutar y relacionar toda la información registrada a través de nuestro sistema que utilizando el *API* de Stamping.io realiza los requests y posts requeridos para que la información de los usuarios recolectada por los nodos sea trazable.



Figura 4.9: Sprint Review

#### 4.2.4. Sprint MVP

Como producto mínimo viable de nuestro proyecto, presentamos la aplicación Web trabajada dónde se puede realizar todas las transacciones para el registro de información de personas existentes o por agregar en función de los nodos seleccionados. En nuestro caso, ha sido hosteado local y temporalmente con nuestros data wallets de la *API* de Stamping.io simulando ser un nodo de recolección de data. En esta, podremos realizar el registro de data de las personas en conjunto de la búsqueda de sus roots en función de su DNI. Véase a continuación:

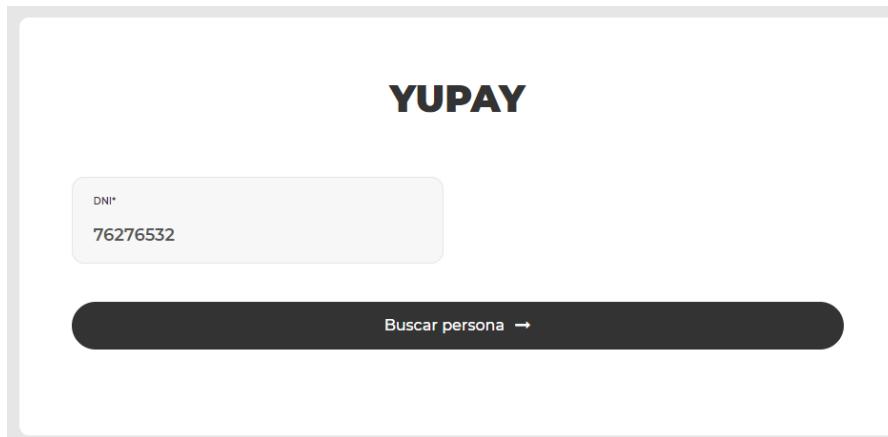


Figura 4.10: Busqueda de roots

Figura 4.11: Root del DNI

## 5. Conclusiones y recomendaciones

- En primer lugar, llegamos a comprobar que el programa realiza de manera efectiva y con transparencia la transferencia de datos de el usuario, brindando mayor seguridad y eficiencia al momento de que se de una transacción y/o lectura de pagos, representando una mejor organización socio-económica para nuestros usuarios. Esto no solo puede ser beneficioso para empresas que quieran contar con este servicio, si no para el mismo estado peruano como forma de regulación frente a los programas sociales, los cuales con la coyuntura actual, demostraron ser ineficientes y que de forma dudosa segmentaron a la población sin antes haber un estudio previo o factores los cuales nos permitiera saber si esa persona se encontraba en estado de pobreza o no, siendo merecedor de el algún bono o programa social de ayuda.
- Por otra parte, los nodos tanto principales como secundarios podrían publicar o modificar información con la intención de corromper el sistema. Sin embargo una vez registrados sería imposible perder el rastro de esta información corrupta en conjunto del responsable, lo que permitiría que el usuario registrado mantenga un seguimiento constante de la data que es manejada y la opción de revalidación o revisiones cuando sea necesario.
- En contraste, la aplicación si bien puede registrar y rastrear la información de las personas aún no se logró implementar un sistema básico para que la información recolectada sea utilizada de forma que apoye a las poblaciones necesitadas, generando certificados de ingreso o graduación de programas sociales. Sin embargo, esto no quita el hecho que en un futuro el sistema de búsqueda pueda ser mejorado para manejar los *timestamps* para organizar mejor la información de los nodos con la finalidad de tener un registro detallado sobre la situación periódico y/o mensual de las personas en el sistema.
- Podemos concluir, que si bien los objetivos iniciales de Yupay no fueron cumplidos en su totalidad, las bases del proyecto para que la información de las personas registradas sea trazable se logró en su totalidad. Además, en conjunto del diseño de nuestra arquitectura planteada permite que el proyecto, en caso se desee, pueda seguir en desarrollo implementando las funcionalidades faltantes y mejorando las ya existentes.

## A. Aplicación Web

Todo la aplicación podrá ser encontrada en el siguiente repositorio de GitHub. Para correr la aplicación y probarla tenemos que realizar lo siguiente. Véase a continuación:

```
PS C:\Users\anqui\Google Drive\PI2\PI2_PF> cd Yupay
PS C:\Users\anqui\Google Drive\PI2\PI2_PF\Yupay> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
July 22, 2020 - 15:27:24
Django version 3.0.7, using settings 'Yupay.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Figura A.1: Running local server



Figura A.2: Main local page

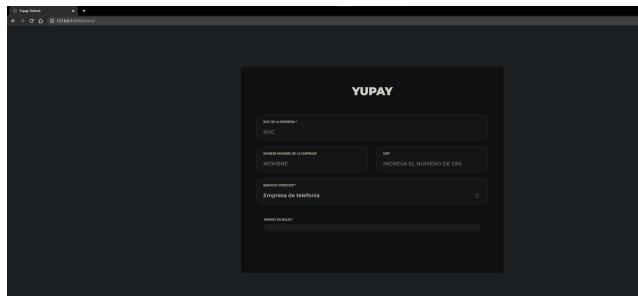


Figura A.3: Inscription local page

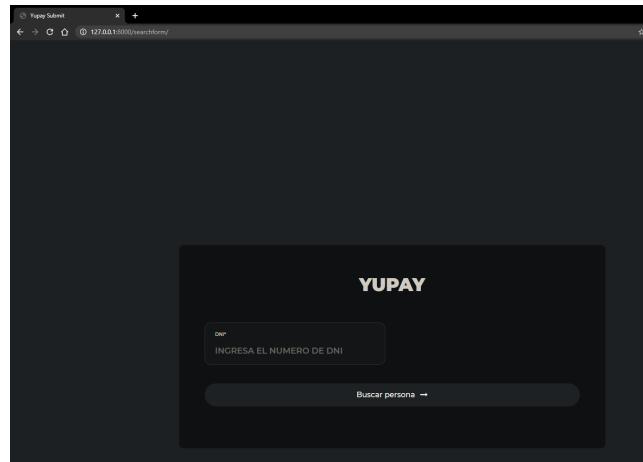


Figura A.4: Search local page

## B. Evidencia de las presentaciones



Yupay es una herramienta que utiliza una blockchain para registrar y trazar ciertos datos de las personas, con la finalidad de sectorizar a la población para facilitar el acceso a programas de ayuda.

2

Figura B.1: Introducción

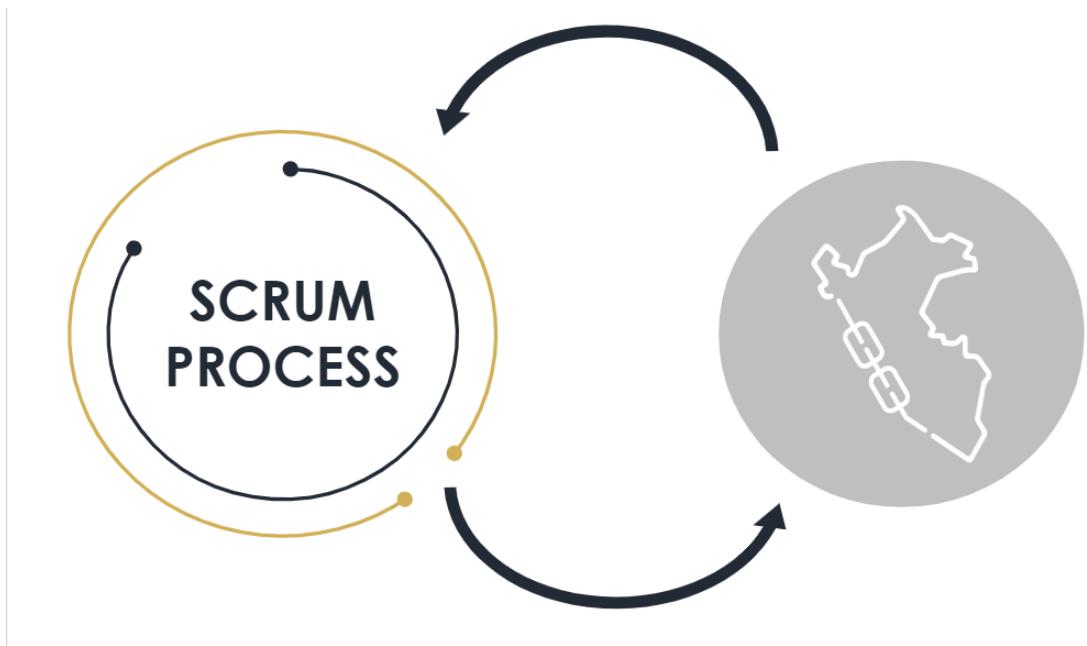


Figura B.2: Revelación de el proceso a llevar para la aplicación

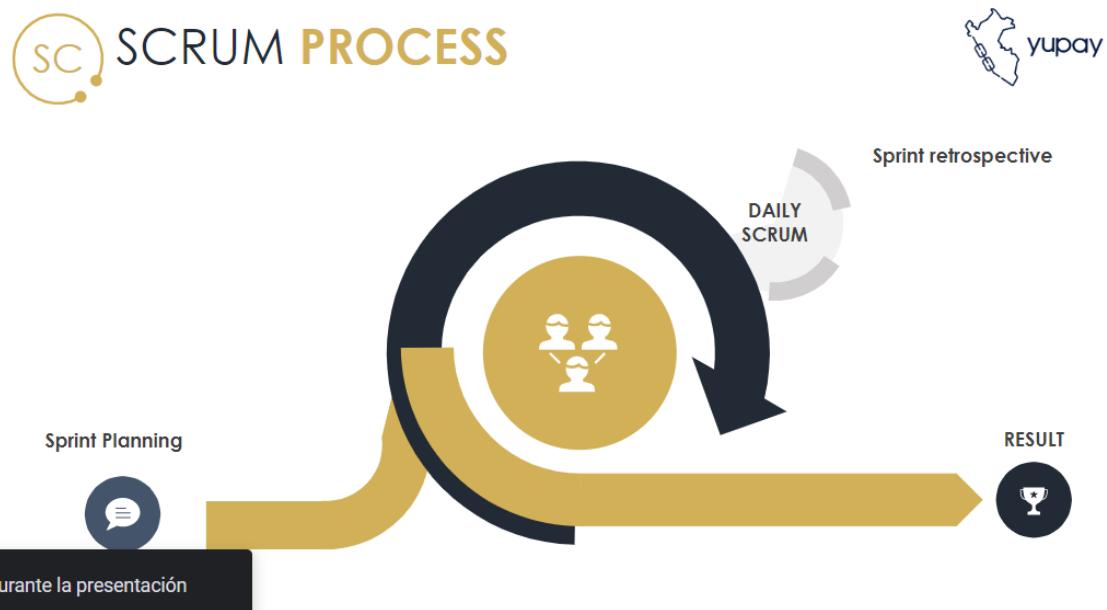


Figura B.3: Desarrollo de nuestro Scrum process



Figura B.4: Sprint planning a detalle

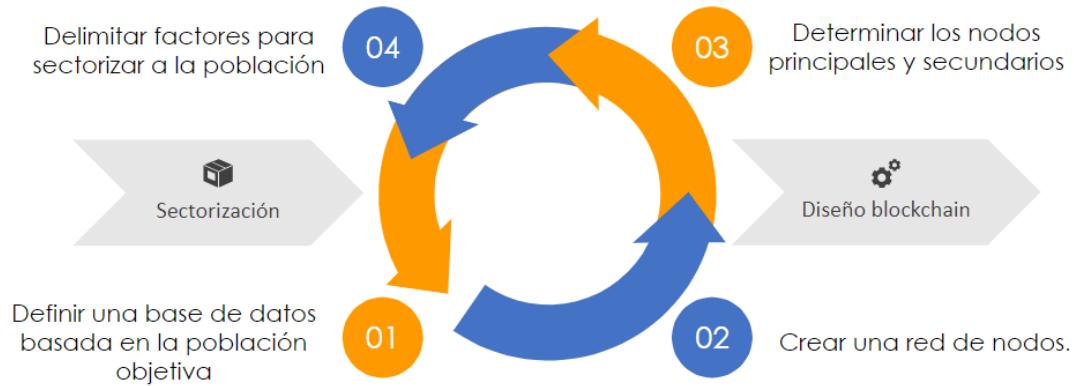


Figura B.5: Sprint1



Figura B.6: Sprint 1 retrospective (semana10)



# SPRINT 1 RETROSPECTIVE



## Semana 11

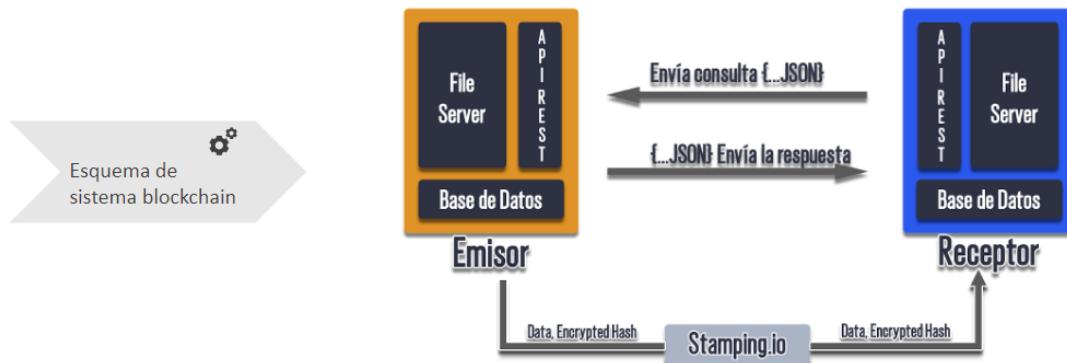


Figura B.7: Sprint 1 retrospective (semana11)



# SPRINT 1 RETROSPECTIVE



## Semana 11

03

Determinar los nodos principales y secundarios

**Nodos principales:**

- Empresas proveedoras de agua potable
- Empresas proveedoras de luz
- Empresas de telecomunicación hogar
- Empresas de telecomunicación móvil
- Empresas de abastecimiento de primera necesidad
- Servicios de salud y/o seguro
- Deudas e hipotecas bancarias
- Propiedades y/o inmuebles
- RENIEC

**Nodos secundarios:**

- Centros comerciales
- Compras online
- AFP y ONP

04

Delimitar factores para sectorizar a la población

Realizamos una investigación acerca de los factores aplicados en la actualidad para clasificar a la población dónde se aplican LP como factor principal. Se utilizará el mismo como base para el desarrollo de un algoritmo, que clasifique a la población en conjunto de la información recolectada o estampada por los nodos.



Figura B.8: Sprint 1 retrospective (semana11)

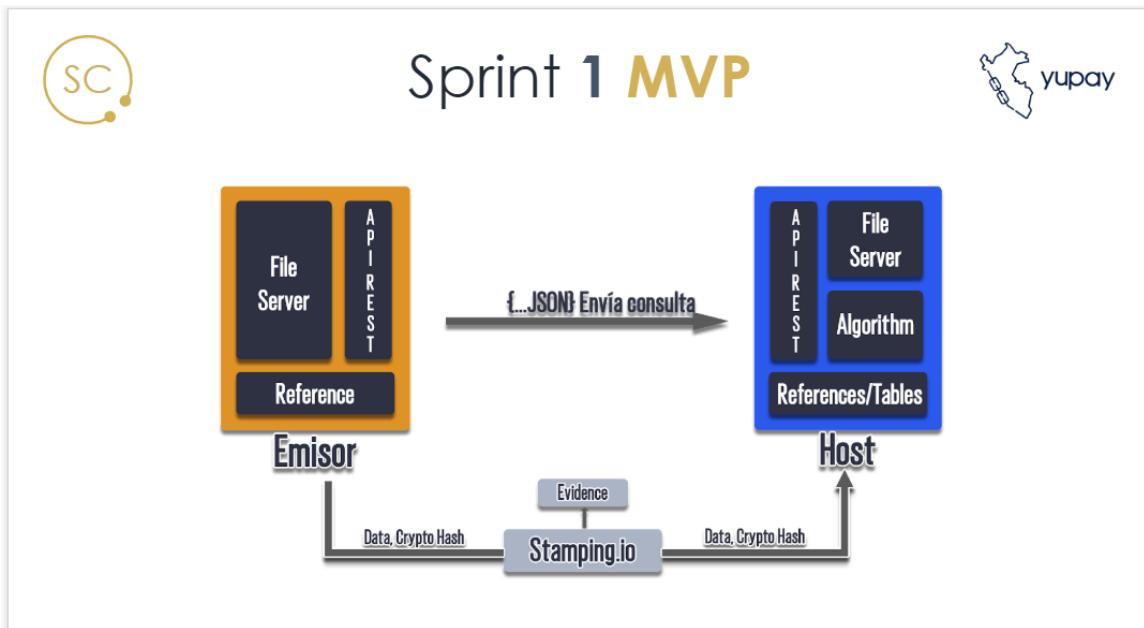


Figura B.9: Primer MVP

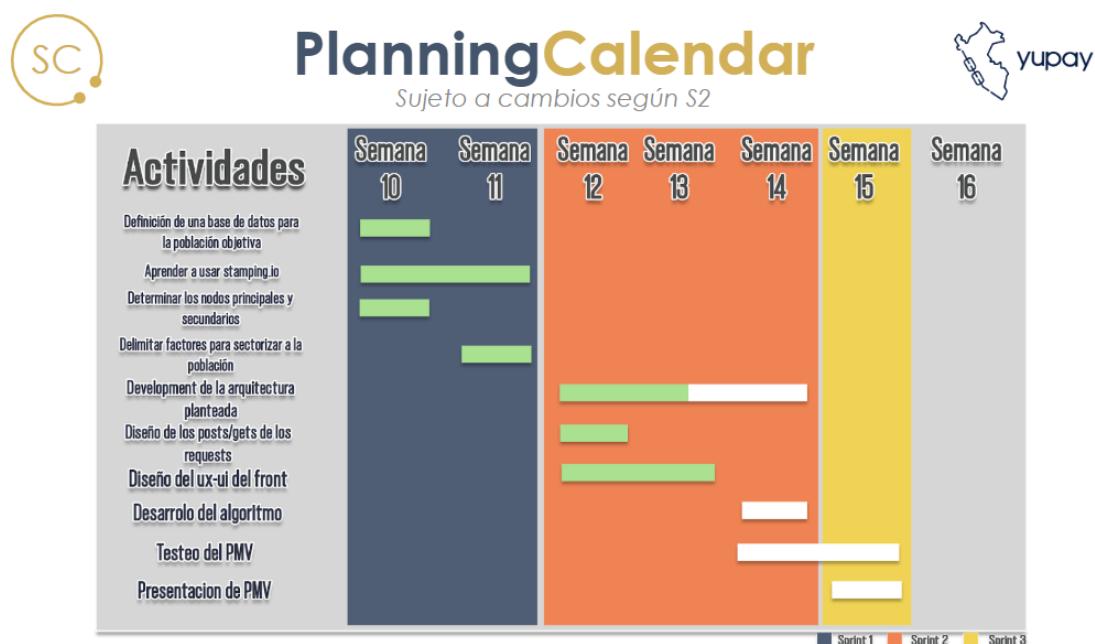


Figura B.10: Calendario de actividades

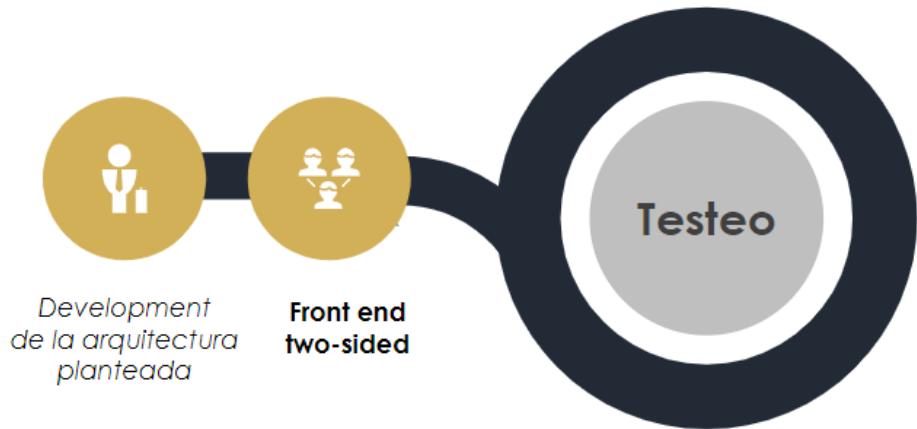


Figura B.11: Sprint2

## SPRINT 2 RETROSPECTIVE

*Semana 12*

03 Diseño de los post/get del request

```

def post(request):
    try:
        token = env.ACCESS_TOKEN
        URL = env.POST_URL
        data = {'true': 'OK',
                'false': 'KO',
                'identity': '001',
                'servicio': 'service',
                'valor': '10000'}
    }
    summary = json.dumps(data)
    h = hashlib.sha256(summary.encode('utf-8'))
    parametros = ('evidence', h,
                  'transactionType', 'Stamping.localPPI',
                  'signature', base64.b64encode(json.dumps(data).encode('utf-8')),
                  'subject': 'register')
    headers = {
        'Authorization': f'Basic {token}',
        'Content-Type': 'application/json'}
    response = requests.post(URL, parametros=parametros, headers=headers)
    return HttpResponse(response)

```

04 Diseño del ux-ui

The image shows two screenshots of the Yupay application. The left screenshot is a mobile phone displaying a dark-themed screen with a map of Ecuador and the word 'yupay' in white. The right screenshot is a desktop application window titled 'YUPAY' with various input fields and a progress bar at the bottom.

Figura B.12: Sprint 2 retrospective (semana12)