

## “Año de la lucha contra la corrupción e impunidad”, nomenclatura que será utilizada como membrete en documentos oficiales del Estado”

PROYECTO

Trabajo Final

CURSO: Programación Concurrente y Distribuida (CC65)

Profesor: Canaval, Luis

Ciclo: 2019-2

Alumno:

Alosilla Lajo Juan Diego

Monterrico, octubre 2019

**Sistema de transacciones.**

**Análisis:**

Un registro público que permita ser monitoreado por cada una de las partes involucradas, e incluso en el caso de fondos del Estado peruano pudiese ser accesible para todos los ciudadanos de la nación, permitiría que cada una de las transacciones realizadas sea reconocida y se pueda verificar su procedencia, su fin y destino; reduciendo así las posibilidades de actos de corrupción en gran magnitud.

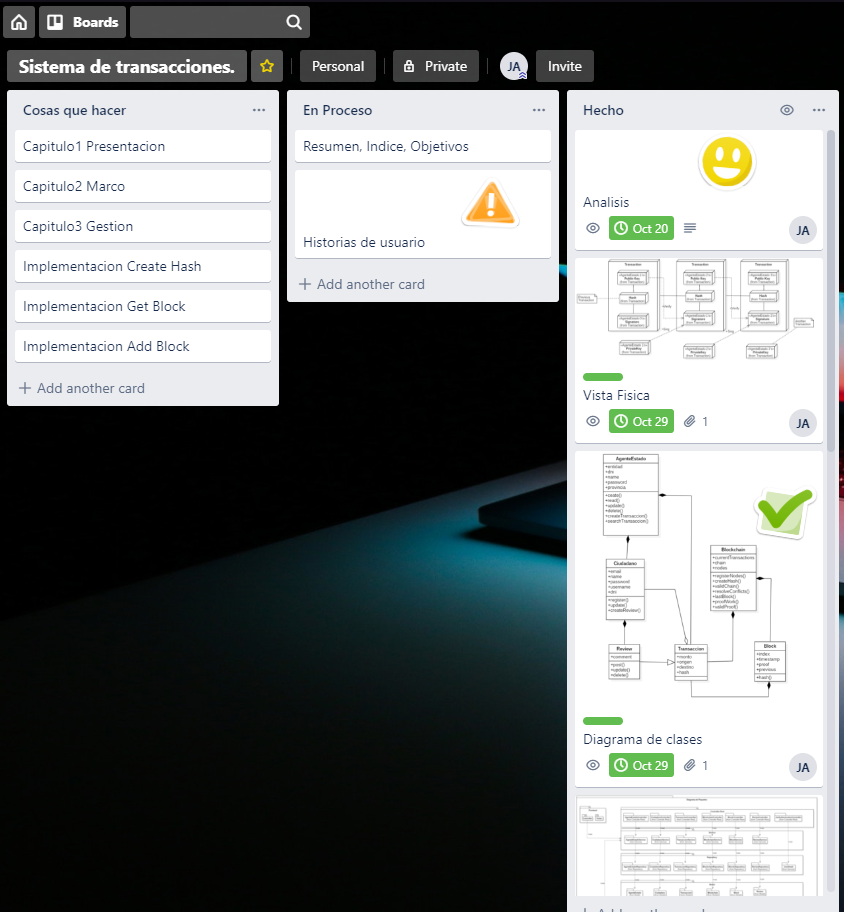
Blockchain puede darle un duro golpe a la corrupción, ya que las características de su tecnología representan una nueva forma más eficaz de preservar la transparencia en cualquier transacción.



**Organización de Tareas**

Para la organización de tareas se utilizó la herramienta trello.

https://trello.com/b/epJ8IqoR



**Hito 2**

1. **Elabora**

Artefactos de diseño para la solución.

* 1. **Historias de Usuario**
  2. **Requerimientos**

**Módulo de transacciones**

Modulo encargado de todos los servicios informativos de las transacciones con respecto a dos tipos de usuario, ciudadano y estado. Estos servicios permitirán listar todas las transacciones o asignaciones de presupuestos hechas por el estado, asimismo podrán ver a detalle cada una de estas transacciones. Este módulo se comunica con el módulo de comentarios.

**Módulo de seguridad**

Modulo esencial para cualquier sistema, encargado de proteger información que pueda ser sensible al uso mal intencionado; como también, delimitar los servicios y funciones para los dos tipos de usuarios según el rol que estos manejen. Este módulo no se relacionará directamente con otros módulos.

**Módulo de asignación de presupuestos**

Modulo encargado de manejar todos los servicios administrativos, los que permitirán realizar transacciones o asignación de presupuesto a distintas jurisdicciones dentro del estado o entes publicos. Este módulo no se relacionará directamente con otros módulos.

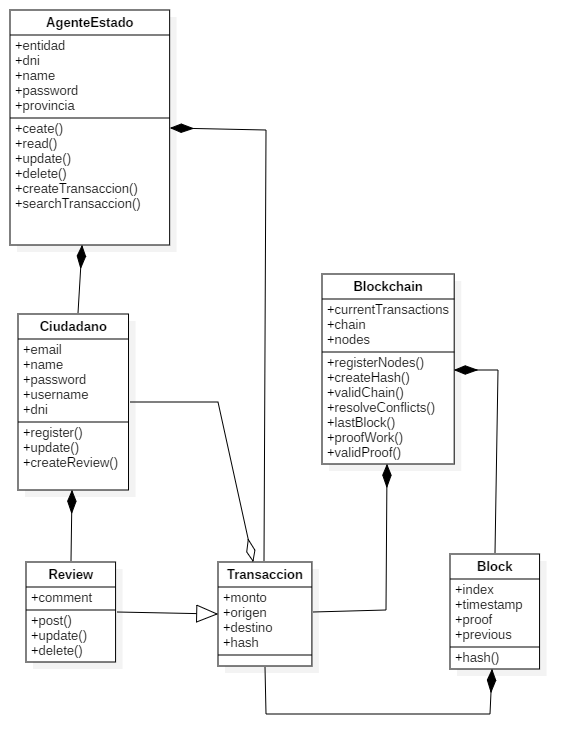
**Módulo de comentarios**

Modulo responsable tanto de publicar como listar comentarios en cada detalle de transacciones, por parte de los usuarios. Este módulo está relacionado con el módulo de transacciones.

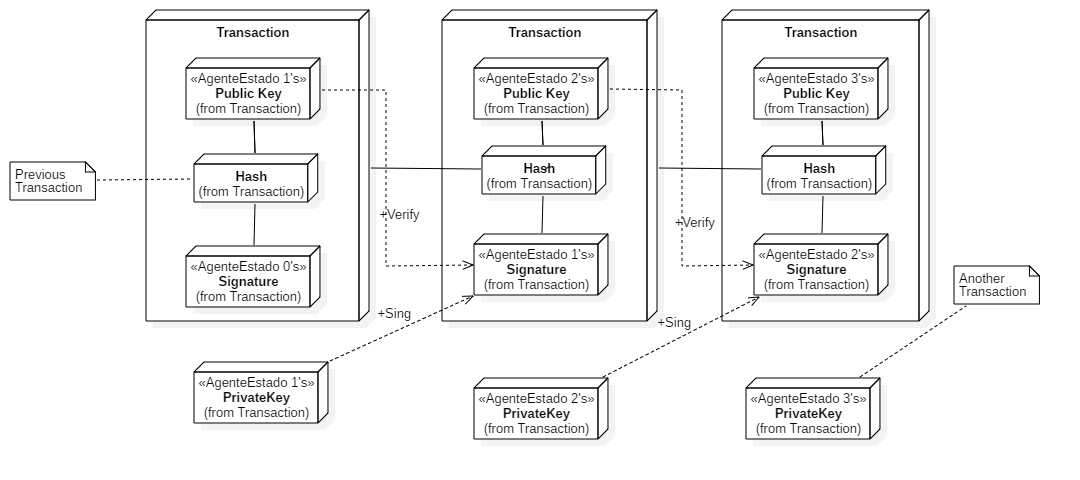
**Requerimientos Funcionales y No Funcionales**

|  |  |
| --- | --- |
| **REQUERIMIENTOS FUNCIONALES** | |
| **REQUERIMIENTOS** | |
| **CÓDIGO** | **DESCRIPCIÓN** |
| RF01 | * La aplicación deberá contar con un función que permita iniciar sesión ya sea como ciudadano o agente del estado. |
| RF02 | * La aplicación debe permitir que el ciudadano pueda agregar un comentario a una transacción. |
| RF03 | * La aplicación debe permitir que el ciudadano pueda ver un listado de las transacciones. |
| RF04 | * La aplicación debe permitir al agente del estado poder realizar una transacción o asignación de presupuesto a otros entes públicos o privados. |
| RF05 | * La aplicación debe permitir al agente del estado poder ver los registros de transacciones o asignaciones realizadas. |
| RF06 | * La aplicación debe permitir al agente del estado poder ver el detalle de una transacción o asignación realizada. |
| RF07 | * La aplicación debe permitir al usuario en general su registro en el aplicativo para su uso. |
| **REQUERIMIENTOS NO FUNCIONALES** | |
| **REQUERIMIENTOS** | |
| **CÓDIGO** | **DESCRIPCIÓN** |
| RNF01 | * La aplicación debe contar con un diseño de interfaz igual en todas sus pantallas; es decir, mismo tipo de letra y color. |
| RNF02 | * La aplicación debe permitir mostrar una o más fotos de referencia sobre el detalle de la transacción para que el ciudadano tenga una idea del propósito de esa transacción. |

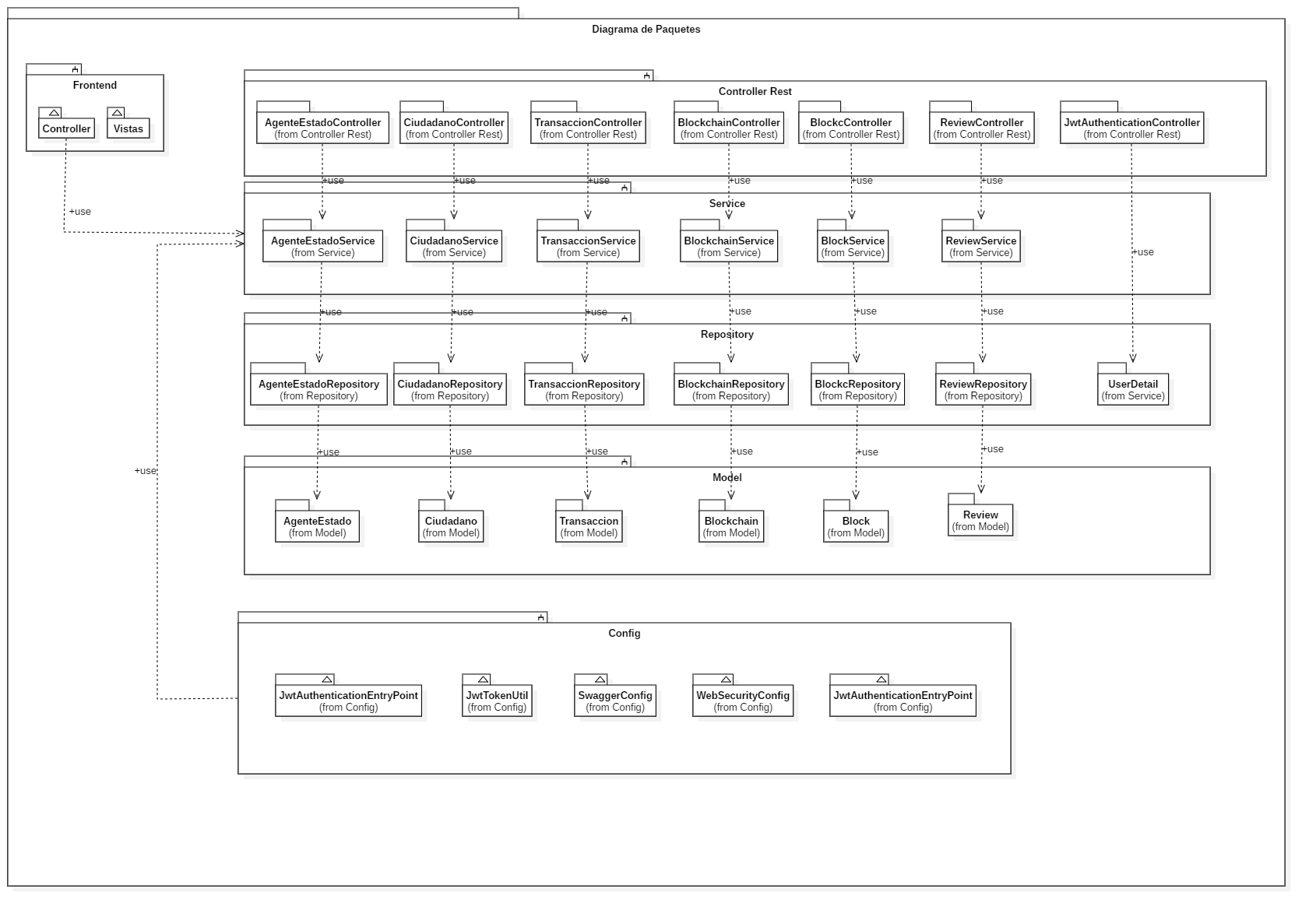
* 1. **Diagrama de clases**

****

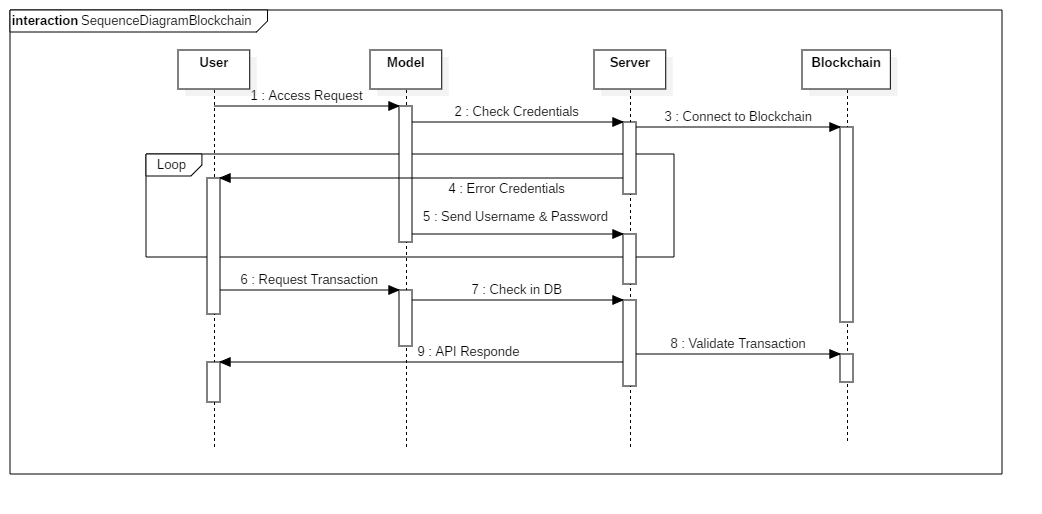
* 1. **Vista Física**

****

* 1. **Diagrama de paquetes**

****

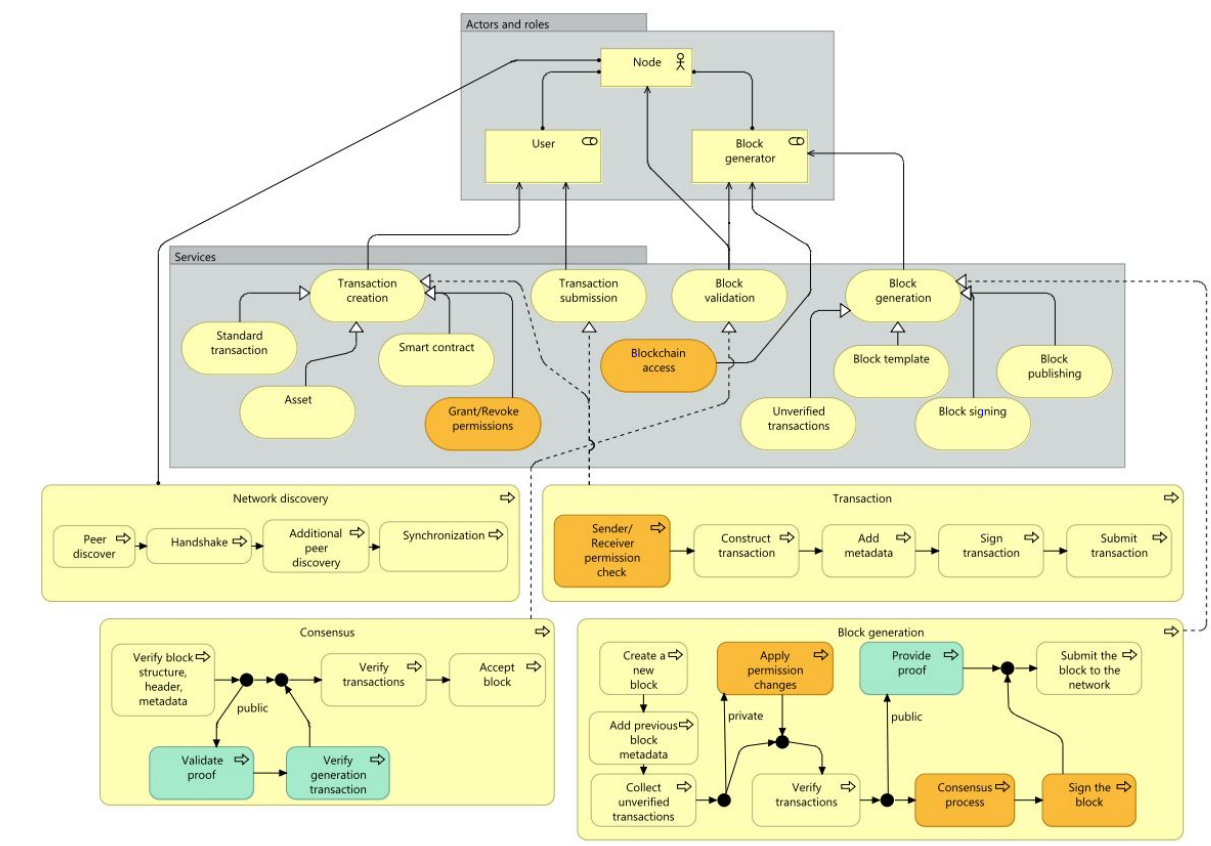
* 1. **Diagrama de secuencias**

****

* 1. **Casos de Uso**

|  |  |
| --- | --- |
| **DIAGRAMA DE CASOS DE USO** | |
| **C:\Users\USUARIO\Documents\GitHub\Blockchain\src\UseCaseDiagramBlockchain.png**  Ilustración 3 Diagrama de Caso de Uso | |
| **DESCRIPCION DEL DIAGRAMA DE CASOS DE USO** | |
| **ACTORES** | Ciudadano |
| Agente de Estado |
| Blockchain |
| **CASOS DE USO** | Obtener Transacciones |
| Obtener historial de Transacciones |
| Agregar comentarios |
| Obtener Hash |
| Obtener key |
| Generar transacción |
| Validad Transacción |

* 1. **Diagrama de componentes**



1. **Define**
2. **Configura**

Git, hoy en día es uno de los sistemas de control de versiones más populares, por la flexibilidad de poder trabajar con ramas (branches) y agregar nuevas funcionalidades, sin poner en “peligro” de inconsistencias. Esa por eso que lo aplicaremos al proyecto.

