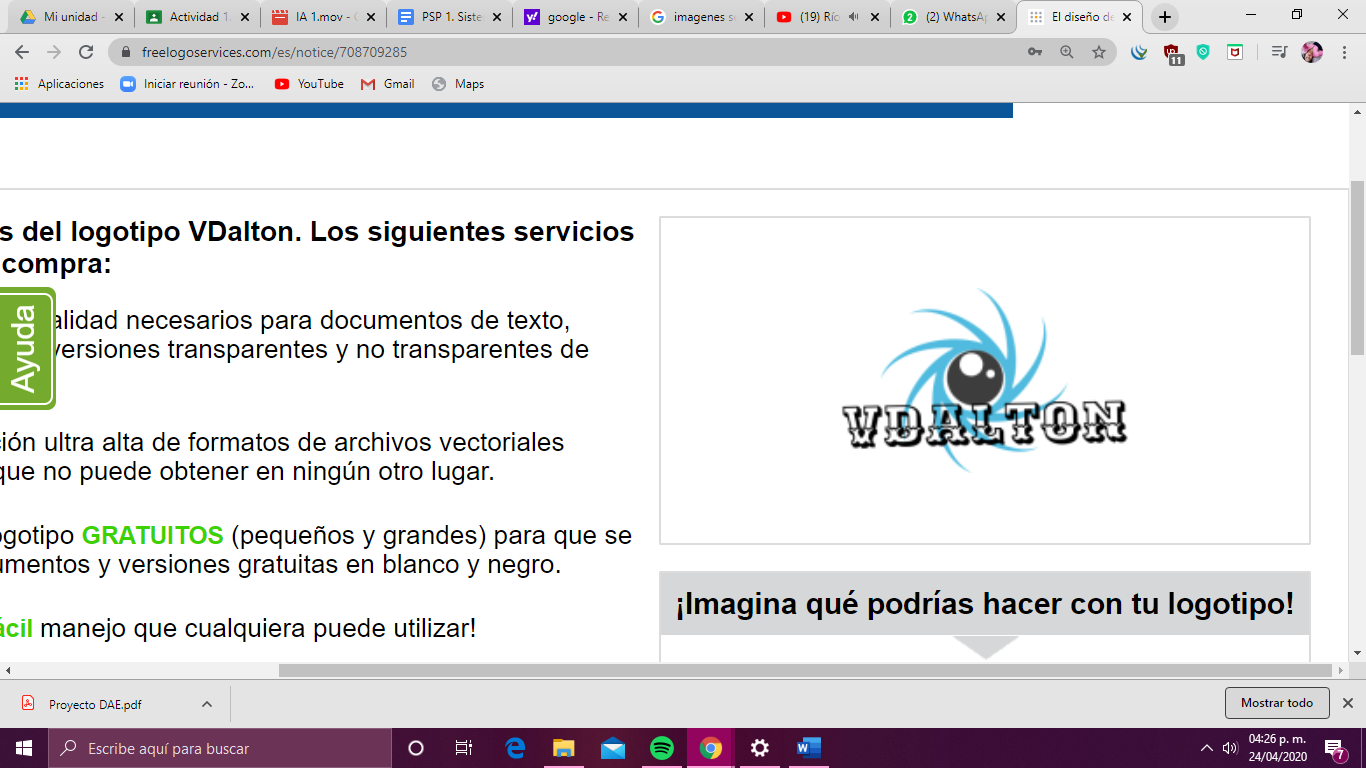
BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

Facultad de Ciencias de la Computación

Ingeniería en Tecnologías de la Información



# Proyecto VDalton

Asignatura: Control de calidad

Docente: Dr. Juan Manuel González Calleros.

Integrantes:

* Luis Alberto amador flores.
* Luis Adrián Pérez san Martín.

índice

**1. Introducción 2**

1.1 Propósito del documento. 2

**2. Descripción general de la metodología 2**

**3. Roles 3**

3.1 Roles de equipo 3

3.1.1 Scrum máster 3

3.1.2 Product Owner 3

3.1.3 Equipo de desarrollo 3

3.2 Roles de usuario 3

3.2.1 Usuario 3

**4. Especificación de requisitos 4**

4.1 Product Roadmap 4

4.2 Historias de usuario 6

4.3 Estimación 11

4.3.1 Ponderación: Planning Poker 11

4.4 Product Backlog 13

4.5 Plan de entregas 14

**5. Diagrama de clases 16**

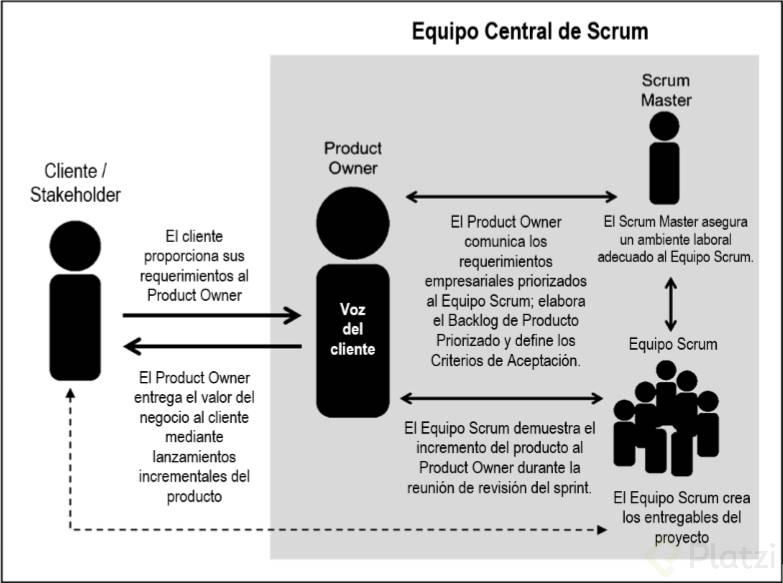
**6. Maquetado de interfaces 17**

6.1Interfaces Finales **20**

1. **Introducción** 
   1. **Propósito del documento**

Este documento tiene la finalidad de llevar un control de forma de guía de usuario para que en un futuro sea reutilizable y fácil de entender el propósito, así como la implementación.

## 2. **Descripción general de la metodología**

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software. Con esta metodología el cliente se compromete con el proyecto puesto que se va a desarrollar iteración a iteración. Así mismo se permite en cualquier momento modificar el software, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración sin ningún problema. Este método de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto. 

**Beneficios:**

* Predicción de tiempos.
* Reducción de riesgos.
* Flexibilidad a cambios.
* Mayor calidad de software.

## 3. **Roles**

### 3.1. Roles de equipo

#### 3.1.1. Scrum master

Aquella persona que se encarga de liderar al equipo, guiándolo para que cumpla las reglas y los procesos de la metodología.

Responsabilidades:

* Gestionar y asegurar que el proceso Scrum se lleve a cabo correctamente.
* Facilitar la ejecución del proceso.
* Resolver dudas acerca de la metodología.
* Eliminar impedimentos.

#### 3.1.2. Product Owner

Aquella persona responsable del éxito del producto desde el punto de vista del cliente.

Responsabilidades:

* Recolectar requerimientos.
* Comunicación frecuente con el cliente.
* Determinar las prioridades de cada una de las funcionalidades del producto.
* Determinar la visión del producto.

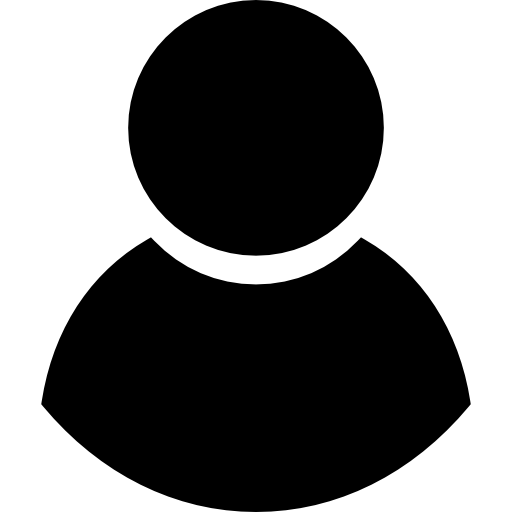
#### 3.1.3. Equipo de desarrollo

Todas y cada una de las personas requeridas para la construcción del producto en cuestión.

Responsabilidades:

* Desarrollar el producto.
* Auto - organizarse.
* Auto - gestionarse.

#### 3.2. Roles de usuario

3.3. Usuario

4. **Especificación de requisitos**

#### 4.1. Product Roadmap

**USUARIO**



Elegir tipo de test a realizar.

Administrar la configuración.

Realizar test elegido por el usuario.

El programa permanezca en segundo plano.

Cancelar configuración.

#### 4.2. Historias de usuario

Especificaciones funcionales que invitan a la conversación para que el detalle de los requisitos del sistema sea más eficaz.

Redactadas bajo el formato “Yo como (rol) necesito (funcionalidad) para (beneficio)”.

* Con las características INVEST:
* Independent (Independientes).
* Negotiable (Negociable).
* Valorable (Valorable).
* Estimable (Estimable).
* Small, scalable, sized (Pequeña).
* Testable (Verificable).

ID # U-1

Título: Inicio de sistema – usuario

yo como: Usuario

Necesito: iniciar sistema

Para: poder elegir test.

ID # U-2

Título: Elegir test – usuario

yo como: Usuario

Necesito: elegir un test

Para: obtener una configuración personalizada

ID # U-3

Título: Realizar test – usuario

yo como: Usuario

Necesito: realizar un test

Para: poder obtener la configuración personalizada obtenida mediante mis respuestas.

ID # U-4

Título: Administrar configuración – usuario

yo como: Usuario

Necesito: administrar la configuración

Para: poder realizar cambios y cancelación a la configuración.

#### 4.3. Estimación

Es necesario realizar la estimación de nuestras historias de usuario como una manera de dimensionar y relacionar la complejidad de las historias de usuario con respecto a otras.

4.3.1. Ponderación: Planning Poker

Como equipo se decidió realizar esta dinámica ágil con el fin de estimar las historias de usuario. La técnica consiste en que cada integrante del equipo tiene una baraja de cartas en donde los números se acomodan (con el previo acuerdo del equipo) para que así correspondan a la sucesión de Fibonacci, entonces se siguen los siguientes pasos:

* El product owner presenta una historia de usuario para ser estimada.
* Todos los participantes realizan su estimación, sin influenciar al resto de sus compañeros ponen su carta elegida boca abajo.
* Una vez que todos hayan colocado sus cartas, se da la vuelta de ellas y se observa la ponderación que cada uno de ellos otorgó, poniendo especial atención y discutiendo los extremos.
* Después de la discusión se vuelve a estimar, esta vez con mayor información a la que se tenía anteriormente.
* El proceso termina hasta que se logra el consenso del equipo y luego se continúa con una nueva historia de usuario.

|  |  |
| --- | --- |
| **Historias de usuario** | |
| **Item** | **Valor (Fibonacci)** |
| Inicio de sistema (usuario) | 5 |
| Elegir test (usuario) | 2 |
| Realizar test (usuario) | 5 |
| Administrar configuración (usuario) | 8 |

### 4.4. Product Backlog

El Backlog es una lista ordenada de todo el trabajo pendiente, contiene descripciones genéricas de todos los requerimientos, funcionalidades deseables, etc. priorizadas según su valor para el negocio.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Item** | **Tipo** | **Estado** | **Estimación** |
| #U1 | Inicio de sistema (usuario) | Historia de usuario | Iniciada | 5 |
| #U2 | Elegir test (usuario) | Historia de usuario | No iniciada | 2 |
| #U3 | Realizar test (usuario) | Historia de usuario | No iniciada | 5 |
| #U4 | Administrar configuración  (usuario) | Historia de usuario | No iniciada | 8 |
|  | **Total** | | | 20 |

### 4.5. Plan de entregas

Una vez creado el product backlog es necesario definir y planificar los sprints o iteraciones. Se debe de tomar en cuenta que cada iteración debe de proporcionar un resultado, un incremento que pueda mostrarse cuando el cliente así lo solicite.

Para planificar los sprints es necesario hablar con el equipo de desarrollo para indicar según su experiencia y trabajo que y cuantas historias de usuario es recomendable realizar en cada iteración.

Un método de apoyo para realizar la actividad anterior es conocer la velocidad de trabajo, esto es sencillo si las personas que conforman el equipo han trabajado juntas antes o simplemente se llevan bien, sin embargo, se vuelve difícil para los demás ya que no tienen ninguna experiencia juntos y por lo tanto planear la velocidad es más complicado.

Considerando 6 semanas de trabajo para 2 sprints se pondera un promedio de 10 puntos de historia por sprint.

|  |  |  |
| --- | --- | --- |
|  | **Sprint 1** |  |
| **ID** | **Item** | **Estimación** |
| #U1 | Inicio de sistema (usuario) | 5 |
| #U3 | Realizar test (usuario) | 5 |
|  | Total de puntos | 10 |

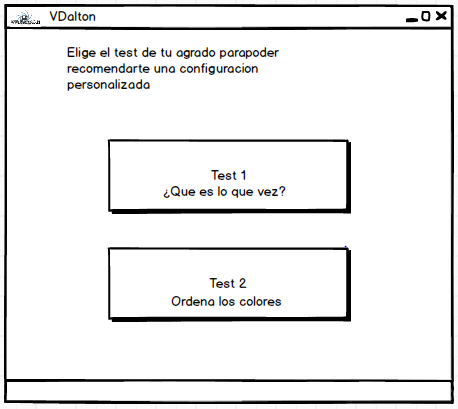
|  |  |  |
| --- | --- | --- |
|  | **Sprint 2** |  |
| **ID** | **Item** | **Estimación** |
| #U2 | Elegir test (usuario) | 2 |
| #U4 | Administrar configuración  (usuario) | 8 |
|  | Total de puntos | 10 |

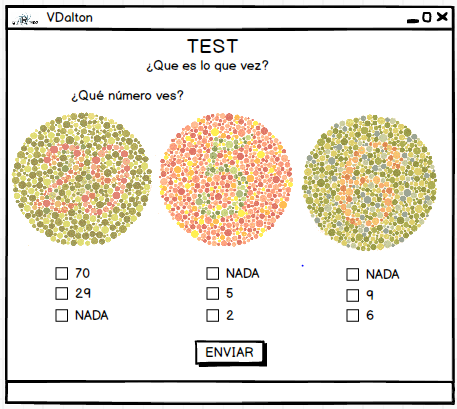
## 5**. Diagrama de clases**

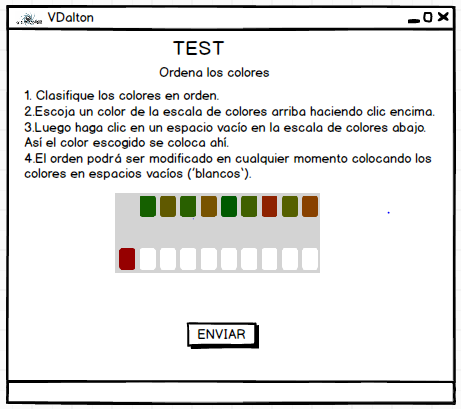
A continuación, se presenta el diagrama de clases.

## **7. Maquetado de interfaces**









## **8. Interfaces Finales**