



Requerimientos Bayesian Poker v0.2

Nombre: Bayesian Poker

Agregar a la aplicación las siguientes características:

- Agregar un diseño estético
- Tipos roles que pueden ser dados de alta **Scrum master** y **miembro del equipo de Scrum (miembro)**.
- Incorporar usuarios al sistema
 - Datos
 - Nombre,
 - apellido,
 - edad,
 - sexo,
 - Rol
 - Scrum master
 - Miembro del equipo de Scrum (miembro)
 - Product owner (temporalmente deshabilitado)
 - Correo (valido)
 - Validar por correo
 - Usuario
 - Contraseña
 - Fecha de alta (oculto a los usuarios)
- Proyectos (Salas)
 - Datos
 - Nombre
 - Descripción
 - Fecha de creación (oculto a los usuarios)
 - Solo el Scrum Master puede generar proyectos (Salas)
 - Un proyecto (sala) solo puede aceptar miembros del equipo
 - Los proyectos tienen asociados Miembros del equipo de Scrum y sprints
- **Scrum Master**
 - Solo el Scrum master puede generar proyectos (modificarlos, deshabilitarlos),
 - Las historias de usuario pueden ser modificadas o deshabilitadas
 - Podrá ver a los miembros y sus estimaciones (históricos).
 - Puede habilitar o deshabilitar miembros
 - Podrá consultar estadísticas en general de los proyectos generados.
 - Puede deshabilitarse, mas no eliminar del sistema, al hacer esto los proyectos asociados se deshabilitan también.
 - El rol no puede ser modificado.





- Controla el estatus de una historia de usuario.
- **Miembro del equipo de SCRUM**
 - Para incorporarse a un proyecto (sala) debe ser aceptado por el Scrum Master.
 - Puede ser través de un código.
 - Puede modificar su nombre, apellido, edad, sexo, contraseña
 - Puede deshabilitarse, mas no eliminarse del sistema.
 - El rol no puede ser modificado.
 - Su función se limita a estimar historias de usuario, no tiene funciones de Scrum master
 - El sistema debe almacenar los estimaciones del usuario en los diversos proyectos (histórico)
 - EL rol no puede ser modificado.
- **Sprint**
 - Un sprint tiene historias de usuarios
 - El Scrum master puede hacer la gestión de los sprint
 - Datos del sprint
 - Nombre
 - Descripción
 - Fecha de creación (oculto a los usuarios)
- **Historias de usuario:**
 - Datos:
 - Nombre de la historia de usuario
 - Descripción
 - Estatus
 - Creada. - Historia recién creada (valor de inicio, solo la primera vez antes de ser deliberada, ya cambiado el estatus no puede regresar a este estatus)
 - Deliberar. - Historia votada pero se tendrá que deliberar (volverse a votar) por no llegar a conceso.
 - Aceptar. - Historia aceptada.
 - La historia puede ser aceptada por varios motivos:
 - Consenso: todos de acuerdo
 - Promedio: Se obtiene un promedio de las estimaciones.
 - Decisión del Scrum master: El Scrum master decide.
 - Pueden ser creadas dentro de proyectos, solo por el Scrum master asociado.
 - Modificar historias de usuario ya cerradas para volver a estimar,
 - Debe de mostrar el histórico
- Evaluar historia de usuario:





- En la aplicación se muestra la historia de usuario y las tarjetas 1,2,3,5,8,13,20
- Cada miembro del equipo de forma independiente estima la complejidad.
- El scrum master decide si la historia se acepta o se procede a otra ronda (toda calificación queda registrada)
- Información de inicio:

Bayesian Poker

Bayesian Póker representa un prototipo de software para determinar la complejidad de historias de usuario de SCRUM basado en la descomposición de la complejidad.

Planning Póker es una técnica de estimación mediante cartas basadas en la secuencia Fibonacci. Esta técnica ofrece muchas ventajas, sin embargo, no siempre es eficiente ya que las estimaciones son realizadas en base a criterio de expertos, lo que resulta confuso en cuanto a los factores que se consideran para realizar las estimaciones.

Este trabajo propone un modelo de conocimiento para determinar la complejidad de las historias de usuarios basadas en el contexto de Planning Póker en Scrum. El objetivo de este trabajo es modelar la naturaleza compleja de la estimación de la historia del usuario para facilitar esta tarea a los desarrolladores principiantes.

Se construyó una red Bayesiana basada en el modelo propuesto que considera la complejidad. Esta propuesta es útil para guiar a los desarrolladores principiantes a evaluar la complejidad de las historias de los usuarios a través de preguntas. Los desarrolladores principiantes podrían usar la propuesta para estimar mejor que al utilizar tradicional Planning Póker.

Citar: López-Martínez, J., Ramírez-Noriega, A., Juárez-Ramírez, R., Licea, G., & Jiménez, S. (2017). User stories complexity estimation using Bayesian networks for inexperienced developers. Cluster Computing, 1–14. <http://doi.org/10.1007/s10586-017-0996-z>

