

Key Performance Indexes

Grupp: jMAPS

Burn Down Chart

Vad:

Hur mycket arbete som återstår av projektet.

Varför:

För att ge gruppen en känsla av hur mycket som är gjort respektive hur mycket som återstår. Visualiserar tydligt vilket är lätt att ta till sig som gruppmedlem. Om mycket har åstadkommits antas det få gruppmedlemmarna att känna sig stolta och glada vilket ökar arbetsmotivationen. Om mycket återstår antas det göra gruppmedlemmarna stressade och skapa en angelägenhet om att arbeta framåt. Stress anses både kunna verka positivt och negativt; ökad produktivitet eller apati, vilket beror av mängden stress och personliga egenskaper. Denna KPI hjälper även till att uppmärksamma om estimeringar av user stories är för höga eller låga så att gruppen kan agera.

Hur:

Arbetsbördan per User Story skall estimeras av gruppen. Summan av arbetsbördan av alla User Stories blir läget i grafen vid tid 0 och därefter sjunker läget med varje avklarad User Story. Gruppen skapar en Definition of Done för att korrekt kunna deklarera en User Story som avklarad. Avklarade User stories redovisas på Weekly Scrum Meetings. Nya tillägg av User Stories uppskattas och höjer läget i Burn Down Chart.

Happiness People Measurement

Vad:

Graden av samarbete och arbetsmotivation

Varför:

För att hålla koll på hur gruppen mår och därefter kunna ta itu med eventuella problem. Exempel på dessa är lycka, stressnivå, nöjdhet i roll, nöjdhet med arbetsbelastningen och nöjdhet med medarbetare.

Hur:

Minst en gång i veckan ska alla gruppmedlemmar fylla i ett formulär. Formuläret frågar hur lycklig personen är med arbetet, hur stressad hen känner sig över arbetet och hur nöjd hen är med samarbetet. Alla svar anges i skala 1-10 där 5 är neutralt. Formuläret genererar antal grafer där man kan se hur de olika medlemmarna upplevde tiden under projektets gång. Formuläret besvaras anonymt för att ge så ärliga svar som möjligt. Nackdelen är problem inte lika lätt kan åtgärdas när det inte framgår vem som upplever dem och varför.

Defect ratio

Vad:

Kvalitet på kod.

Varför:

För att få en bra förståelse för produktens kvalitet utan att skapa mörkertal. Nackdelen med Defect Count är att det skapar incitament att inte registrera/anmäla defekter eftersom att det försämrar KPI:n. Därför finns här ett tillägg där KPI:n förbättras vid varje åtgärdad defekt, något som skapar incitament att förbättra koden.

Hur:

$\text{Defect Count} / \text{Defects resolved} = \text{Defect Ratio}$. En defekt registreras i ett lämpligt system (ex. Git eller Trello) där det även går att registrera om felet åtgärdats. Det viktiga med ett sådant system är att det visar historik så att KPI:n kan räknas ut.