

## JAPANESE WIJSHEDEN VOOR DE TESTPROFESSIONAL

Door Berry Kersten • [berry.kersten@improveqs.nl](mailto:berry.kersten@improveqs.nl) [@berrykersten](https://twitter.com/berrykersten)



*In de loop van de tijd doe je veel nieuwe kennis en vaardigheden op. Maar op een gegeven moment kun je het gevoel hebben dat je leercurve niet meer zo snel stijgt als je zou willen. Dat is jammer, want blijven leren is een belangrijke katalysator voor jezelf, zeker op een Agile project.*

*Maar om in jezelf te blijven investeren, moet je wel weten hoe! Een Japanse visie kan hierbij nieuwe inzichten geven.*

*Vanuit mijn interesse voor het Japanse gedachtegoed over kennismanagement geef ik in dit artikel op een niet-alledaagse manier aandacht aan alledaagse zaken. Door middel van een aantal Japanse termen wordt inzicht gegeven in het belang van kennis, leren, verbeteren en herkennen van je kwaliteiten.*

### Agile Manifesto

Agile kan worden beschouwd als een interactiemodel om wendbaar te worden. De waarden en principes hieromtrent zijn verwoord in het Agile Manifesto (ref 1). De eerste zin van het manifest is: **'we are uncovering better ways of developing software by doing it and helping others do it'**.

Zelf vind ik 'we are uncovering better ways' treffend. Want dit impliceert een proces van leren en verbeteren. Alvoorens dit proces te doorlopen is het goed eerst eens een stap terug te doen en jezelf af te vragen waarom je dit allemaal zou moeten doen. Dit kan onder andere door te kijken naar het Japanse begrip Ikigai.

### Ikigai

Voor de Japanners is Ikigai een onderdeel van het dagelijks leven. Het betekent: 'de reden waarom je elke dag opstaat'. Ikigai is een combinatie van spirituele elementen, zoals missie en roeping, en praktische en sociale elementen als wat de wereld nodig heeft en waar je voor betaald wordt. Dit wordt weergegeven in de afbeelding.



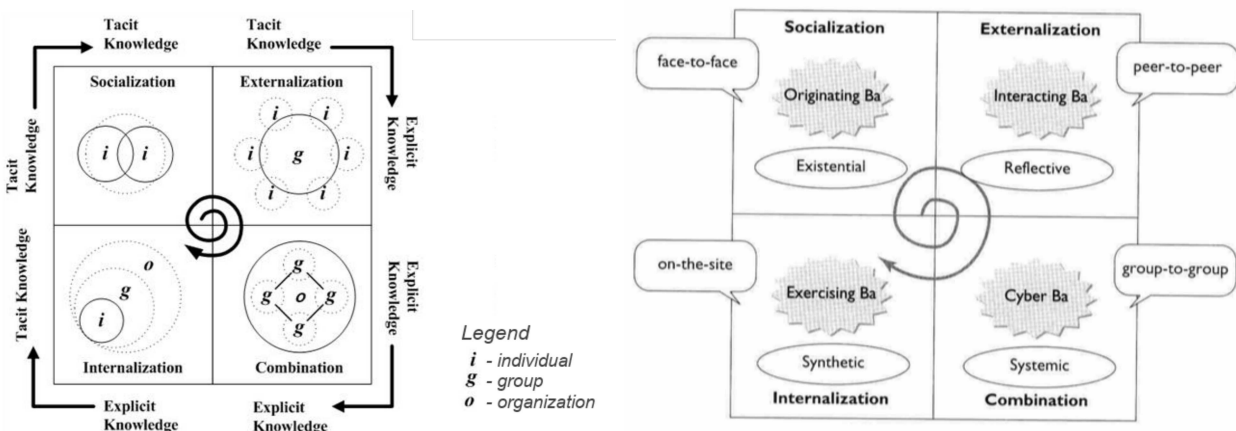
Iedereen heeft Ikigai, maar het is een zaak van bewustwording en keuzes maken. Sommigen kunnen dit ervaren als zweverig, maar je kunt het ook beschouwen als een alternatief voor bestaande theorieën over behoeften en motivatie. Denk aan Daniel Pink's Drive (ref 2) of de behoeftepiramide van Maslow (ref 3).

Lijkt het je de moeite om te onderzoeken hoe Ikigai bij jou in elkaar steekt? Vul je eigen Ikigai-schema in en kijk hoe je er nu voor staat en vul het nog eens in voor hoe je wilt dat het er over een jaar voor staat.

Uiteindelijk kun je als kenniswerker met Ikigai doelgerichter met je carrièrepad omgaan, want kennis is van belang om jezelf wendbaar te maken in een snel veranderende wereld. Volgens de Japanse professoren Ikujiro Nonaka en Hirotaka Takeuchi is menselijke interactie een randvoorwaarde om nieuwe kennis te creëren (ref 4).

## Ba

Op basis van onderzoek in succesvolle Japanse bedrijven, realiseerden Nonaka en Takeuchi een theorie voor het ontwikkelen van kennis in organisaties, te weten het SECI-model: Socialisatie, Externalisatie, Combinatie en Internalisatie. Of, de stroom van kenniscreatie waarin spiraalsgewijs impliciete kennis in expliciete kennis wordt omgezet en vice versa. De ruimte en de context waarin deze kruisbestuiving van kennis plaatsvindt wordt Ba genoemd. Deze ruimte is niet alleen fysiek, maar ook mentaal en virtueel. Ba kan dan een platform zijn voor het bevorderen van individuele en/of collectieve kennis op de werkvloer. Nonaka en Takeuchi visualiseren SECI en Ba als volgt.



Een mond vol, maar wat betekent dit in de praktijk? In termen van Scrum reflecteert Ba de processen om kennis te delen, te ontwikkelen en om te leren. Volgens Fægri (ref 5) zijn deze processen zeer voornaam bij de hele cyclus van softwareontwikkeling. De concepten van Ba en SECI zijn complex en het gaat in dit artikel te ver om deze theorie diep uit te werken, maar mijn stelling is dat het nuttig is om je bewust te zijn van deze concepten. Verandering van kennis en vaardigheden begint tenslotte bij bewustwording.

Om verder in te zoomen naar de praktijk geef ik wat voorbeelden (ref 7 & 8) vanuit Scrum. Want het huidige Scrum raamwerk is gebaseerd op de theorie van het werk van Nonaka en Takeuchi (ref 9).

### Socialisatie

Het delen van impliciete kennis tussen teamleden tijdens de stand-up. In deze Originating Ba worden ervaringen, gevoelens en emoties uitgewisseld.

### *Externalisatie*

Het expliciet maken van impliciete kennis via dialoog. Het team bepaalt wat en in hoeverre iets gedocumenteerd wordt in user stories via de Interacting Ba.

### *Combinatie*

Expliciete kennis uit verschillende bronnen worden samengevoegd en geordend. Dit leidt tot nieuwe expliciete kennis. Denk aan de BDD-aanpak 'Specification by Example' (ref 10) in combinatie met Cucumber (ref 11) of SpecFlow (ref 12). Dit is een virtuele (Cyber) Ba waarin gebruik wordt gemaakt van tooling.

### *Internalisatie*

Expliciete kennis wordt impliciet gemaakt. In feite leren door te doen waar de opgedane kennis eigen wordt gemaakt en men bekwaam wordt. Ik zie dit als een 'kennis increment' nadat een sprint afgelopen is in de Exercising Ba. Tijdens mijn opgedane Agile ervaring de afgelopen jaren heb ik mijn eigen kennis increment opgebouwd en heb dit vertaald naar nieuwe vaardigheden op bijvoorbeeld het vlak van testautomatisering en requirements engineering.

SECI en Ba hebben mij geholpen om een betere T-shaped testprofessional te worden. De context (Scrum) heeft dit gefaciliteerd. Andersom heeft een Scrum team professionals nodig die daarin weten te acteren. Om deze interactie succesvol te laten zijn, zijn gecommitteerde mensen nodig die willen leren en verbeteren. Dit laatste wordt ook wel Kaizen genoemd.

## **Kaizen**

Beter worden door veranderen via kleine stappen; dat is in een notendop wat de Japanse filosofie Kaizen inhoudt. Het Japanse woord 'kai' betekent veranderen en 'zen' betekent 'goed'. Kaizen is oorspronkelijk gericht op het continu verbeteren van organisatieprocessen, maar de filosofie is prima toepasbaar op het individu. Veelal komen deze inzichten vanuit de mindfulness hoek. Persoonlijk interpreteer ik Kaizen als meer rendement uit jezelf halen door ononderbroken verandering en verbetering. Dit klinkt als een utopie, maar vaak is het een kwestie van gewoon doen en focus blijven houden.

In een blog van Blaz Koz (ref 13) staan enkele tips die kunnen helpen:

**Onderzoek je talenten.** Je hebt vaardigheden die je kunt oppoetsen, maar het is zaak je ware talent(en) en passie(s) te vinden waarin je kunt excelleren. Zelf heb ik via een cursus 'Personal Branding' ontdekt waar mijn eigen talenten liggen.

**De omgeving is belangrijk.** We zijn deels een 'product' van de omgeving. Zoek de omgeving (werkgever, opdracht, netwerk) waarin je je kunt profileren.

**Er zijn altijd kansen.** Je hebt altijd een keuze om jezelf beter te maken. Neem een proactieve houding aan en ga uitdagingen aan.

**Maak kleine stappen.** Soms lijkt een verbeterdoel groot, maar je kunt (net als op de werkvloer) uitdagingen opbreken in kleinere brokken.

**Let op je valkuilen.** Tijdens het proces van verbeteren kun je problemen tegenkomen, teleurgesteld worden of fouten maken. Leer hiervan en bewandel een ander pad zodat je niet in dezelfde valkuilen stapt als voorheen.

Een andere manier om jezelf te verbeteren is vaak oefenen, want oefening baart kunst. In Japan noemen ze dit Kata.

### Kata

In de Japanse industrie is Kata 'de dingen op de juiste manier blijven doen'. Kata is tevens een term bij karate: een patroon van bewegingen waarin series van technieken zitten. Deze bewegingen worden heel vaak herhaald om stap-voor-stap te verbeteren. Bij eXtreme Programming is het begrip Code Kata (ref 14) niet onbekend. Het doel hiervan is simpelweg een betere softwareontwikkelaar te worden door vaak te oefenen. Wil je als tester soortgelijke kata's uitvoeren? Neem dan eens een kijkje in de Testing Dojo (ref 15).

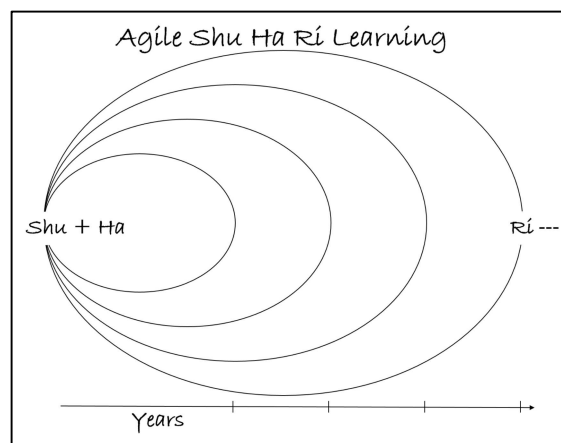
Vanuit de Japanse vechtsport Aikido is er nog een soortgelijke term die ik interessant vind, namelijk Shuhari.

### ShuHaRi

Shuhari is een manier van denken die gaat over het proces van leren en het zich meester maken van een vaardigheid. Hierin worden drie fasen onderkend op weg naar het meesterschap:

- **Shu:** leren en volgen van de regels
- **Ha:** regels weten te breken/ombuigen
- **Ri:** eigen stijl ontwikkelen

Een van de grondleggers van het Agile Manifesto, Alistair Cockburn, heeft dit concept vertaald naar het vakgebied Agile softwareontwikkeling (ref 16). Cockburn maakt de vergelijking met Agile volwassenheid. Dave Moran (ref 17) visualiseerde dit als volgt:



Shu kan beschouwd worden als het dogmatisch toepassen van de Agile practices, terwijl bij Ha en Ri de focus verschuift naar de Agile principes respectievelijk de Agile waarden. Met andere woorden, de verschuiving vindt plaats van Agile doen naar Agile denken, en daarmee in staat zijn het verschil te kunnen maken. Voor jezelf kun je besluiten of je vasthoudt aan oude/bestaande werkwijzen of dat je voor het meesterschap gaat en dus je eigen stijl ontwikkelt. Niels Talens schetste dit via een voorbeeld (ref 18):

<b>Shu</b>	<i>Volg de Scrumguide. Vertrouw dat dit raamwerk goed werkt</i>
<b>Ha</b>	<i>Naarmate de Agile volwassenheid toeneemt, voegen bepaalde Scrum meetings minder waarde toe (omdat ze op een andere manier ingevuld worden). De Scrum regels worden dus aangepast</i>