

Introduzione



Questo libro non è un'opera di ricerca. Non ho svolto un diligente lavoro di raccolta di informazioni. Quelle che state per leggere sono le mie personali considerazioni, osservazioni e opinioni, derivanti dal mio coinvolgimento ormai ventennale con il mondo Agile; niente di più, niente di meno.

Lo stile di scrittura che ho scelto è colloquiale. A volte ho scelto parole un po' crude. E anche se non sono solito imprecare, una [piccolo intervento di edulcorazione] parolaccia in queste pagine ci stava proprio bene, per esprimere il significato che intendevo trasmettervi.

Oh, ma non preoccupatevi: questo libro non è un impeto di rabbia, uno sfogo. Quando l'ho ritenuto necessario, ho citato alcune fonti di riferimento, che vi esorto a seguire. Ho cercato di discutere le mie opinioni con alcuni dei personaggi che sono nella comunità Agile almeno da quanto vi sono io. Ho anche chiesto a qualcuno di loro di fornire elementi e argomenti di precisazione o disaccordo, dedicando loro alcuni capitoli e paragrafi. In ogni caso, non considerate questo libro come un testo di studio. È molto meglio considerarlo come una memoria: i mugugni di un brontolone che dice ai novelli e sedicenti fan della programmazione Agile di andarsene via dal suo prato.

Questo libro è per programmatori e per non-programmatori. Non è un libro tecnico. Non vi troverete codice. Ha il solo scopo di fornirvi una panoramica di quello che era l'intento originale dello sviluppo software Agile, senza entrare nei dettagli tecnici della programmazione, del testing e della gestione.

Questo è solo un piccolo libro. Piccolo in quanto l'argomento non è poi così ampio. Quella Agile è solo una piccola idea riguardante il piccolo problema dei piccoli team di programmazione mentre preparano piccole cose. Quella Agile *non* è una grande idea riguardante un grande problema di un grande team di programmazione che realizza grandi cose. È perfino ironico che questa piccola soluzione di un piccolo problema abbia un nome. Dopotutto, il piccolo problema in questione è stato risolto negli anni Cinquanta e Sessanta, praticamente subito dopo che è stato inventato il software. Tornando a quei tempi, piccoli team software impararono piuttosto bene a fare piccole cose. Il problema esplose negli anni Settanta, quando i piccoli team software che facevano piccole cose rimasero invischiati in un'ideologia in base alla quale con grandi team sarebbe stato possibile fare grandi cose.

Non è del tutto logico che per fare grandi cose ci vogliano grandi team? Ma per l'amor di Dio, no! Per fare grandi cose non ci vogliono grandi team; le grandi cose arrivano grazie alla collaborazione di molti piccoli team che fanno molte piccole cose. Questo è ciò che i programmatori degli anni Cinquanta e Sessanta sapevano per istinto. Ed è proprio questo ciò che si è dimenticato negli anni Settanta.

Perché è stato dimenticato? Temo che la colpa sia della discontinuità. Negli anni Settanta il numero di programmatori nel mondo iniziò a esplodere. In precedenza, nel mondo vi era solo qualche migliaio di programmatori. Poi diventarono centinaia di migliaia. Oggi tale numero si avvicina ai cento milioni.

I primi programmatori, negli anni Cinquanta e Sessanta, non erano affatto degli sprovveduti. Avevano iniziato a programmare a trenta, quaranta e cinquant'anni. Negli anni Settanta, appena prima che la popolazione di programmatori iniziasse a esplodere, questi "vecchietti" iniziarono ad andare in pensione. Per questo motivo, la generazione successiva non ha mai avuto la formazione necessaria. Così si mise all'opera tutto un esercito di ventenni, proprio mentre gli esperti programmatori se ne stavano andando e così la loro esperienza non è mai stata trasferita come avrebbe dovuto essere.

Qualcuno direbbe addirittura che questo abbia dato origine a quelli che potremmo chiamare gli “anni oscuri” della programmazione. Per trent’anni, abbiamo lottato con l’idea che avremmo dovuto fare grandi cose con grandi team, avendo ormai dimenticato che il segreto consisteva invece nel fare tante piccole cose con tanti piccoli team.

Poi a metà degli anni Novanta, iniziammo a renderci conto di che cosa avevamo perduto. L’idea dei piccoli team iniziò a germinare e a crescere. L’idea si diffuse in tutta la comunità degli sviluppatori software, acquisendo forza. Nel 2000, ci rendemmo conto che avevamo bisogno di un vero e proprio reboot del settore. Dovevamo tornare a imparare quello che i nostri predecessori conoscevano istintivamente. Dovevamo riscoprire, ancora una volta, che le grandi cose vengono realizzate da tanti piccoli team che collaborano facendo tante piccole cose.

Per rendere popolare questo concetto, dovevamo dargli un nome. Lo chiamammo “Agile”. Ho scritto questa introduzione nei primi giorni del 2019. Sono trascorsi due decenni dal reboot avvenuto nel 2000 e io credo che ci sia bisogno di un nuovo reboot. Perché? Perché il piccolo e semplice messaggio Agile è stato distorto nel corso degli anni. Si è mescolato con i concetti di Lean, Kanban, LeSS, SAFe, Modern, Skilled e molti, molti altri. Queste altre idee non sono sbagliate, ma non fanno parte del messaggio originale di Agile.

Pertanto, è tempo, ancora una volta, di tornare a ricordare quello che i nostri predecessori sapevano negli anni Cinquanta e Sessanta e che abbiamo ri-appreso nel 2000. È tempo di ricordare che cosa significa davvero Agile.

In questo libro, non troverete nulla di particolarmente nuovo, nulla di eccezionale o di dirimente, nulla di rivoluzionario e illuminante. Quello che vi troverete è la riaffermazione del principio Agile così come venne enunciato nel 2000. Certamente, il tutto visto da una nuova prospettiva, derivante da ciò che abbiamo appreso negli ultimi 20 anni. Ma in generale, il messaggio di questo libro è il messaggio del 2001 e il messaggio del 1950. È un vecchio messaggio. Ma è un messaggio vero. È un messaggio che ci dà la piccola soluzione al piccolo problema del fatto che un piccolo team software possa fare piccole cose.

Ringraziamenti

Il mio primo ringraziamento va a una coppia di intrepidi programmatori che hanno gioiosamente scoperto (o riscoperto) le pratiche contenute in questo libro: Ward Cunningham e Kent Beck.

Subito dopo viene Martin Fowler, senza il cui supporto, agli esordi, la rivoluzione Agile non avrebbe mai visto la luce.

Ken Schwaber merita una speciale menzione per l’indomabile energia che ha profuso nella promozione e adozione di Agile.

Anche Mary Poppendieck merita una speciale menzione per la disinteressata e inesauribile energia che ha dedicato al movimento Agile e alla cura che ha rivolto alla Agile Alliance. Sono convinto che Ron Jeffries, tramite i suoi discorsi, articoli, blog e il persistente calore del suo carattere, abbia rappresentato la coscienza del primo movimento Agile.

Mike Beedle ha combattuto la lotta del giusto per Agile, ma è stato assassinato senza motivo da un vagabondo nelle vie di Chicago.

Gli altri autori originali dell'*Agile Manifesto* meritano una citazione speciale. Arie van Bennekum, Alistair Cockburn, James Grenning, Jim Highsmith, Andrew Hunt, Jon Kern, Brian Marick, Steve Mellor, Jeff Sutherland e Dave Thomas. Jim Newkirk, mio amico e socio all'epoca, ha lavorato instancabilmente per sostenere Agile, affrontando nel contempo gravi problemi personali che la maggior parte di noi (e certamente io) non possiamo neanche immaginare. Poi devo menzionare il personale di Object Mentor. Tutti loro si sono assunti il rischio di aver iniziato ad adottare e promuovere Agile. Molti di loro sono presenti nella seguente fotografia, scattata all'inizio del primo XP Immersion Course.



Figura I.1 *Dietro:* Ron Jeffries, autore, Brian Button, Lowell Lindstrom, Kent Beck, Micah Martin, Angelique Martin, Susan Rosso, James Grenning.
Davanti: David Farber, Eric Meade, Mike Hill, Chris Biegay, Alan Francis, Jennifer Kohnke, Talisha Jefferson, Pascal Roy.
Non presenti nella fotografia: Tim Ottinger, Jeff Langr, Bob Koss, Jim Newkirk, Michael Feathers, Dean Wampler e David Chelimsky.

Vorrei inoltre ringraziare tutti coloro che si sono uniti a formare la Agile Alliance. Alcuni di loro sono rappresentati nella fotografia che segue, scattata alla riunione iniziale di quella ora augusta alleanza.



Figura I.2 *Da sinistra:* Mary Poppendieck, Ken Schwaber, autore, Mike Beedle, Jim Highsmith. Non presente nella fotografia: Ron Crocker.

Infine, un grazie a tutto il personale di Pearson e specialmente alla mia editor Julie Phifer.