



C'era una volta... Waterfall



Commessa



Stima (dell'intero progetto)



User Requirements Specifications



Functional Specifications



Altri 57 documenti firmati col sangue



Sviluppo



Ritardi, problemi, fallimenti...



La
risposta?

L'agile

Niente più analisi infinite

Niente più stime infinite

Niente più firme col sangue

Costruiamo il software gradualmente,
insieme al cliente, con feedback
continui



||
risultato?

Niente
più
fallimenti
epici, ma

«Ops... Manca questo pezzo»

«Ma io volevo una ciabatta, cos'è questa scarpa?»

700 wireframe (magari interattivi)

«Hai fatto i 734 smoke test manuali pre-rilascio?»

Nuovo programming language: DT++
(Debito tecnico ++)

Backlog di 4+ anni di sola risoluzione bug
(e DT)...

«Senza requisiti come faccio a sviluppare
XYZ??»



Cosa manca



Behaviour Driven Development

Molto più che testing



Frullato di nozioni

BDD è un framework agile

Definisce quindi un modo di lavorare che coinvolge tutti (e non solo i dev!), con particolare focus sulla definizione degli Acceptance Criteria.

È *complementare* (ad esempio) a *Scrum*, che invece si focalizza di più sul processo iterativo.

Eredita alcuni concetti di Extreme Programming e di Scrum.



Frullato di nozioni

Si focalizza sui requisiti

Un requisito è un Acceptance Criteria.

Ogni requisito, che prende il nome di Behaviour, sarà composto da «step».



Frullato di nozioni

Si basa su tre principi



Frullato di nozioni

Si basa su tre principi

- Business Value



Frullato di nozioni

Si basa su tre principi

- Business Value
- Everything is a behaviour



Frullato di nozioni

Si basa su tre principi

- Business Value
- Everything is a behaviour
- Enough is enough



Frullato di nozioni

Per evitare di ricadere in ancestrali analisi infinite è fondamentale imparare a capire dove fermarsi.

Una volta che si è definito lo stretto indispensabile per poter partire: si parte.

Si avrà tutto il tempo per modificare e/o aggiungere behaviors in iteration successive.

Richiede quindi che si lavori per iteration di prodotto ed MVP



Frullato di nozioni

I più diffusi

- Spec
- Gherkin

Ha diversi formalismi



Frullato di nozioni

BDD è un **framework** agile

Si focalizza sui requisiti

Si basa su tre principi

- Business Value
- Everything is a behaviour
- Enough is enough

Richiede quindi che si lavori per iteration di prodotto ed MVP

Ha diversi formalismi

Requisito «classico»

Come «persona a dieta»

Voglio una pagina di inserimento del nuovo pasto

Per poter inserire un nuovo pasto al diario alimentare

La pagina dovrà contenere

- Un pulsante di aggiunta
- La lista dei pasti
- Ogni pasto
 - sarà composto da uno o più alimenti
 - avrà un tipo (colazione, pranzo, cena)

Formalismi

Spec

- Describe: Lista dei pasti
 - It should *contenere il pulsante «nuovo pasto»*
- Describe: *Pulsante nuovo pasto*
 - It should *mostrare il form «nuovo pasto»*
 - Describe: Form «nuovo pasto»
 - It should *contenere dati del pasto*
 - Describe: *Pasto*
 - It should *contenere la lista degli alimenti*
 - It should *contenere il tipo di pasto*
 - Describe: *Tipo di pasto*
 - It could be *colazione, pranzo o cena*

Gherkin

Funzionalità: *Inserimento nuovo pasto*

Come *«persona a dieta»*

Voglio una pagina di inserimento del nuovo pasto

Per poter inserire un nuovo pasto al diario alimentare

Scenario: *Apertura pagina inserimento dati*

Data la pagina contenente la lista dei pasti

Dato il pulsante «nuovo pasto»

Quando il pulsante «nuovo pasto» viene cliccato

Allora la modale di inserimento pasto viene mostrata

Scenario: *Definizione di un pasto*

Data la modale di inserimento pasto

Allora conterrà la lista degli alimenti

E conterrà il tipo di pasto

Scenario: *Tipo di pasto*

Allora potrà essere colazione, pranzo o cena

Formalismi

Spec

- Describe: Lista dei pasti
 - It should *contenere il pulsante «nuovo pasto»*
- Describe: *Pulsante nuovo pasto*
 - It should *mostrare il form «nuovo pasto»*
 - Describe: Form «nuovo pasto»
 - It should *contenere dati del pasto*
 - Describe: *Pasto*
 - It should *contenere la lista degli alimenti*
 - It should *contenere il tipo di pasto*
 - Describe: *Tipo di pasto*
 - It could be *colazione, pranzo o cena*

Gherkin

Funzionalità: *Inserimento nuovo pasto*

Come *«persona a dieta»*

Voglio una pagina di inserimento del nuovo pasto

Per poter inserire un nuovo pasto al diario alimentare

Scenario: *Apertura pagina inserimento dati*

Data la pagina contenente la lista dei pasti

Dato il pulsante «nuovo pasto»

Quando il pulsante «nuovo pasto» viene cliccato

Allora la modale di inserimento pasto viene mostrata

Scenario: *Definizione di un pasto*

Data la modale di inserimento pasto

Allora conterrà la lista degli alimenti

E conterrà il tipo di pasto

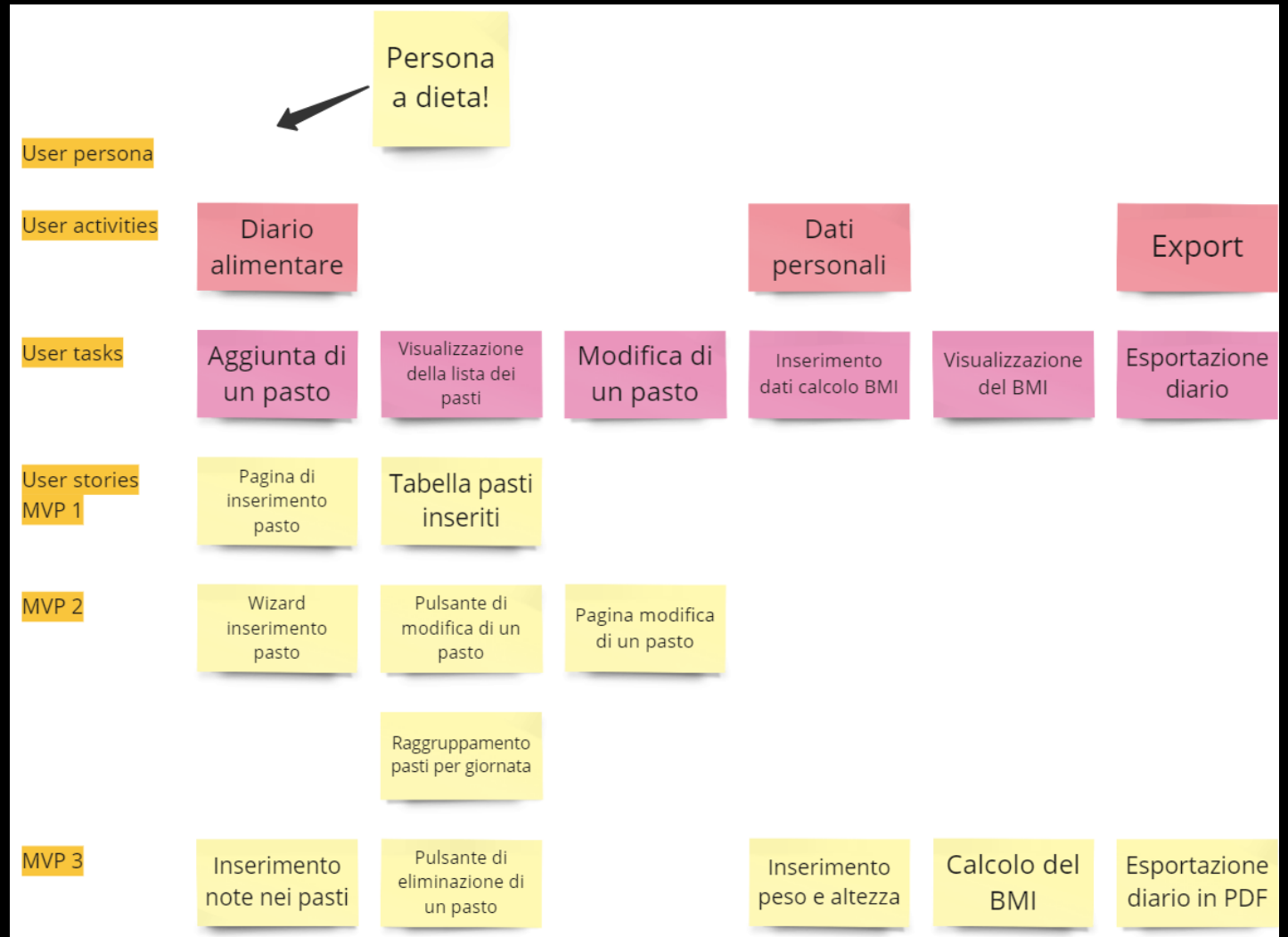
Scenario: *Tipo di pasto*

Allora potrà essere colazione, pranzo o cena

E ora...?

Diario alimentare!

Diario alimentare



Bibliografia

- Extreme Programming
- Clean Code: A Handbook of Agile Software Craftsmanship
di Robert C. Martin
ISBN-13 978-0132350884
- The Clean Coder: A Code of Conduct for Professional Programmers
di Robert C. Martin
ISBN-13 978-0137081073
- Domain-Driven Design: Tackling Complexity in the Heart of Software
di Eric Evans
ISBN-13 978-0321125217

Prossimo step?



workingsoftware
conference

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~

BDD: dai behavior al TDD
Jun/30/2023 — Milan, Italy



Fabio Cocchi

Traduciamoli in Domain Specific Behaviors e Service/Component behaviors, su cui fare TDD!

Ci vediamo alla workingsoftware conference, il 30 giugno!

TDD e BDD: cosa cambia?



[YouTube](#)