
COTATO
https://www.cotato.com
CS EDUCATION
소프트웨어 개발방법론: 애자일
23.09.15
8기 교육팀장 신유승

COTATO

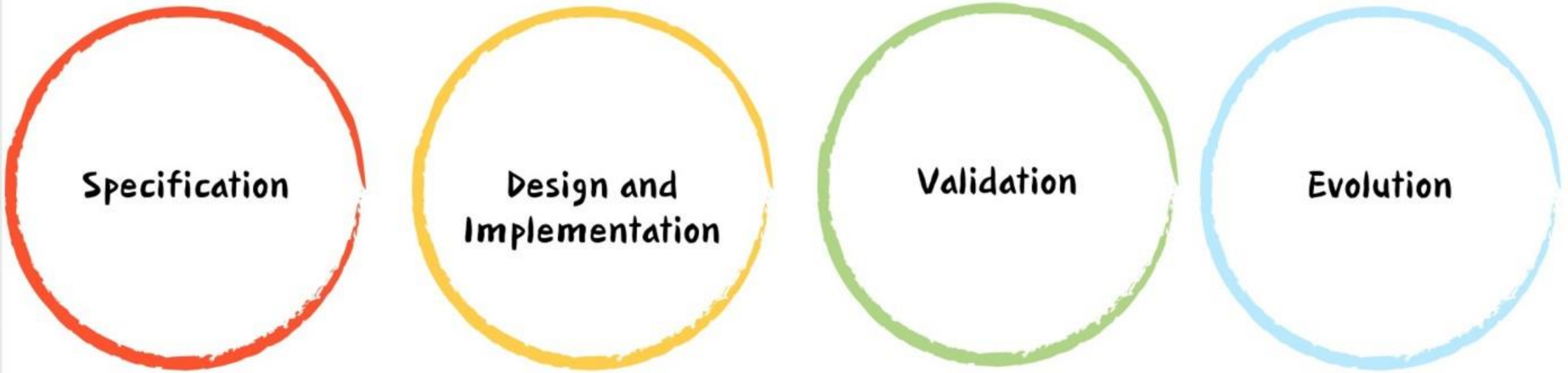
https://www.cotato.com

이제 시작될 프로젝트, 어떤 과정으로 진행되지?

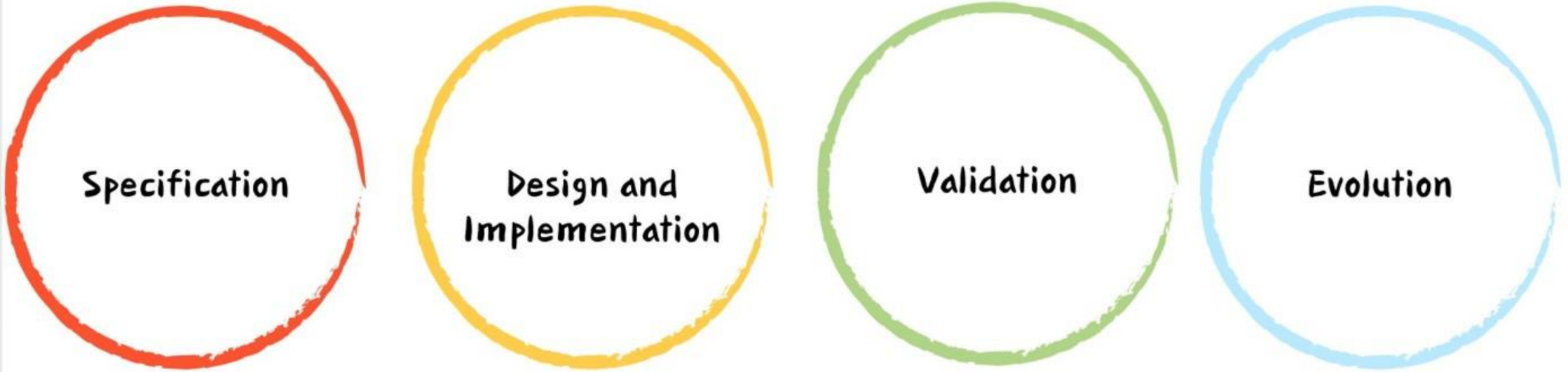
교육 목표

- 애자일이 무엇인지 나도 알 수 있다!
- 프로젝트 일정관리에 도움이 되길..!

소프트웨어 프로세스



소프트웨어 프로세스



요구사항 정의 및 기능 명세

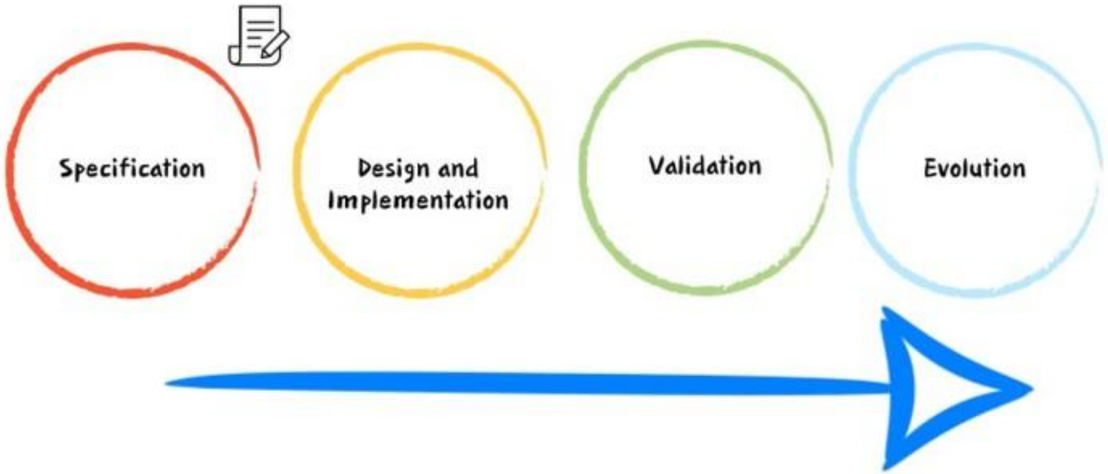
설계 및 실제 개발

테스트

유지보수

소프트웨어 프로세스 모델

Plan - driven

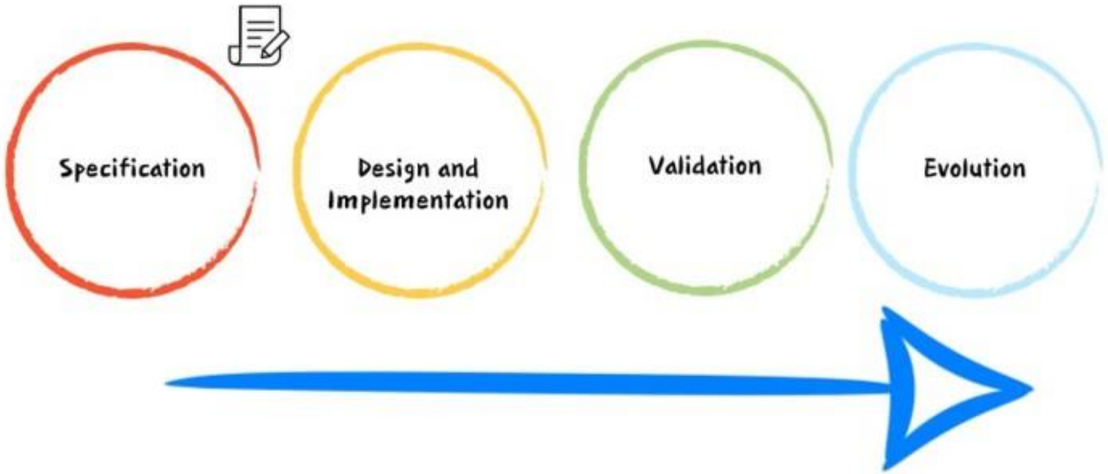


Plan-driven process

- 모든 과정을 사전에 계획해놓고 진행
- Specification 단계에서 요구사항에 대한 문서를 남긴 후 다음 단계를 진행함.

소프트웨어 프로세스 모델

Plan - driven



Plan-driven process

요구사항 변화에 대응하기 어려움

- 모든 과정을 사전에 계획해놓고 진행
- Specification 단계에서 요구사항에 대한 문서를 남긴 후 다음 단계를 진행함.

문서화 과정에서 필요 이상의
오버헤드 발생

COTATO

https://www.cotato.com

Agile Process

Agile이란?

Agile Process

Agile이란?

영어사전

agile

미국식[ˈædʒɪl] 영국식[ˈædʒaɪl]

1 날렵한, 민첩한 (=nimble)

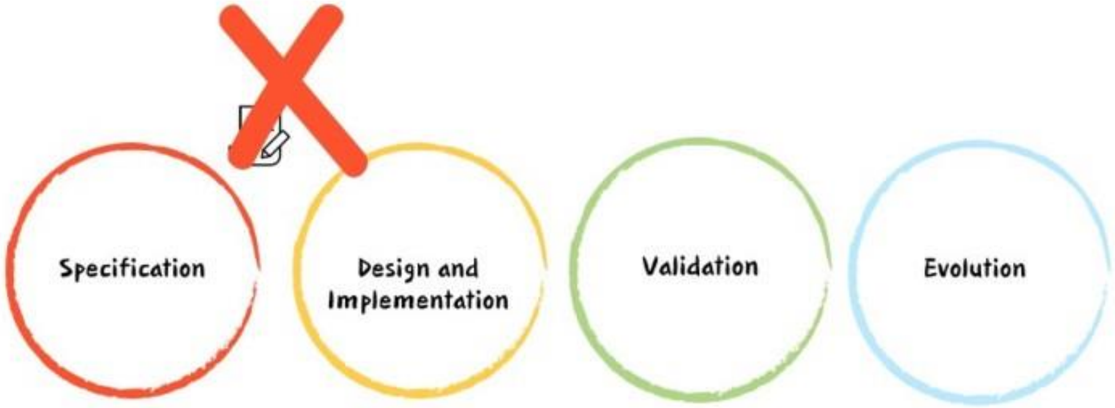
2 (생각이) 재빠른, 기민한

영어사전 결과 더보기

요구사항 변화에 민첩하게 반응하며

기능 개발, 배포 시간을 단축하는 프로세스 모델

Agile Process의 특징



문서화 단계를 없앰
문서 작성에 쓰는 비용을 축소함.

문서화가 필요한 이유

```
public interface Queue<E> extends Collection<E> {
```

이렇게만 보면 Queue가 무엇을
하는지 이해가 안됨!

문서화가 필요한 이유

> Queues typically, but do not necessarily, order elements in a FIFO (first-in-first-out) manner. Among the exceptions are priority queues, which order elements according to a supplied comparator, or the elements' natural ordering, and LIFO queues (or stacks) which order the elements LIFO (last-in-first-out). Whatever the ordering used, the *head* of the queue is that element which would be removed by a call to `remove()` or `poll()`. In a FIFO queue, all new elements are inserted at the *tail* of the queue. Other kinds of queues may use different placement rules. Every Queue implementation must specify its ordering properties.

(queue는 일반적으로 FIFO 구조이고 ...)

문서화가 필요한 이유

> Queues typically, but do not necessarily, order elements in a FIFO (first-in-first-out) manner. Among the exceptions are priority queues, which order elements according to a supplied comparator, or the elements' natural ordering, and LIFO queues (or stacks) which order the elements LIFO (last-in-first-out). Whatever the ordering used, the *head* of the queue is that element which would be removed by a call to `remove()` or `poll()`. In a FIFO queue, all new elements are inserted at the *tail* of the queue. Other kinds of queues may use different placement rules. Every Queue implementation must specify its ordering properties.

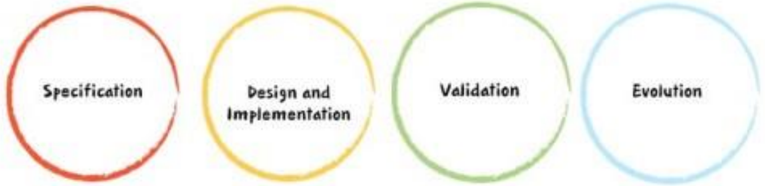
(queue는 일반적으로 FIFO 구조이고 ...)

문서화를 통해 이해도를 높일 수 있음!

Agile Process: 문서화 비용 축소

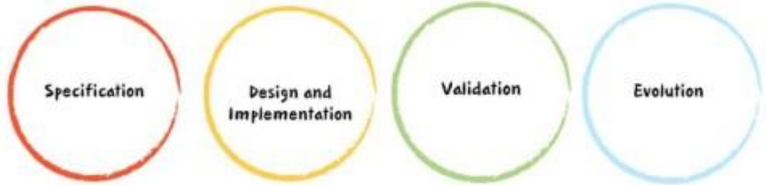
1. 리팩토링
2. 테스트 주도 개발
3. Pair Programming

Agile Process: increment 단위 개발

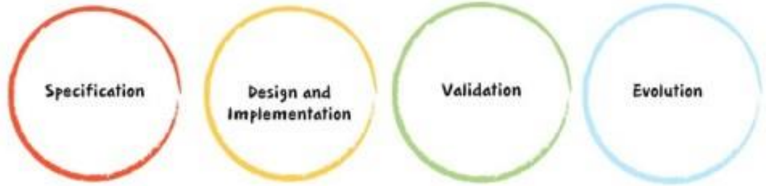


increment1

increment2

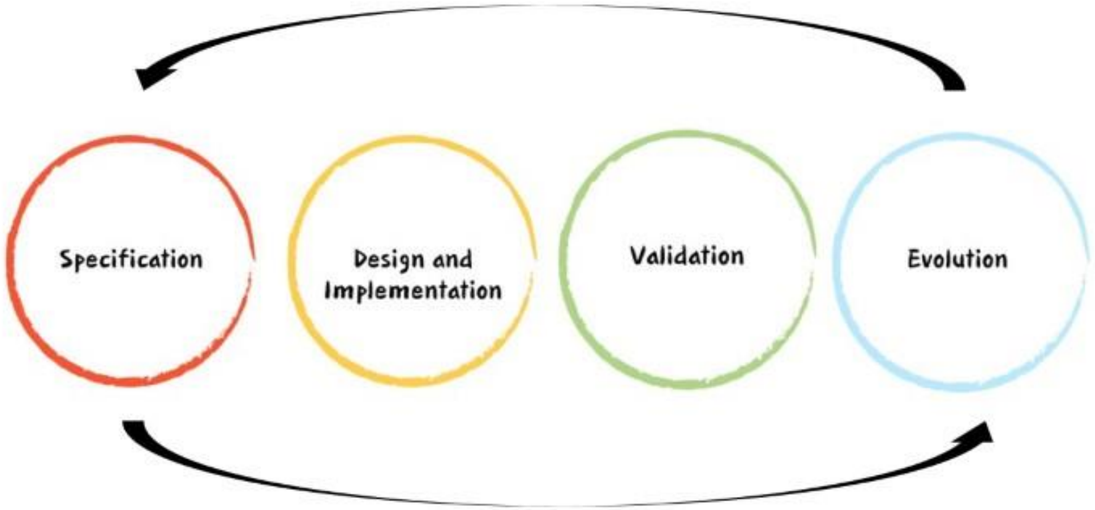


increment3



increment4

Agile Process: increment 단위 개발



increment 단위를 반복함!

Agile Process: Customer가 팀에 참여

- 이름: 김고객
- 역할: 고객 (개발자 아님)
- 하는일: Requirement 제시, 테스트

요구사항을 쉽게 찾고
즉각적으로 반영하기 위함!

Agile Process: 어디에 적합할까?

1. 작은 단위의 소프트웨어 개발에 유리함
2. 범용적인 소프트웨어보단, Client에 특화된 소프트웨어 개발 시 적합함.

Agile Process vs Plan Driven

	Agile Process	Plan driven
요구사항	지속적인 요구사항 수립 및 변경	초기 요구사항 수립 및 엄격한 변경 관리
프로젝트 규모	소규모 프로젝트	대규모 프로젝트
설계	즉시 설계	사전 설계
문서화	경량화된 문서화	상세한 문서화 강조
역할	전체적인 팀워크를 중시	엄격한 역할 분리
모델	스크럼, XP, 린 소프트웨어 개발	Waterfall 모델

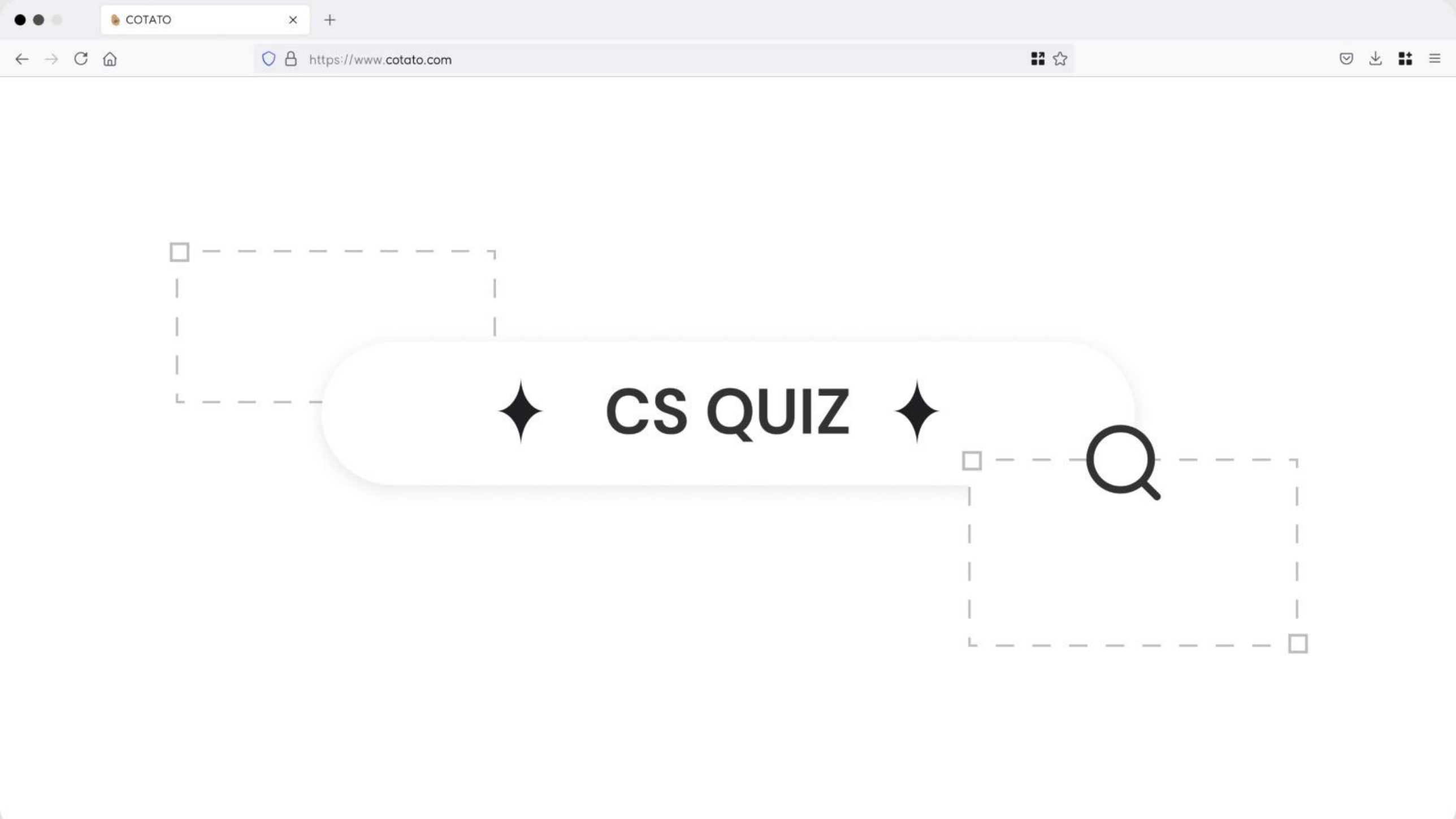
실무에선 어떻게 쓰일까?

plan - driven

자율 주행차

Agile process

UI/UX 디자인


COTATO
https://www.cotato.com
CS QUIZ

Q1. 애자일(Aglie)은 영어로 `OO한`이란 뜻을 가졌다. 애자일 프로세스는 ㅁㅁㅁㅁ변화에 OO한 소프트웨어 개발 방법론이다.

COTATO

https://www.cotato.com

A1. 애자일(Aglie)은 영어로 `OO한`이란 뜻을 가졌다. 애자일 프로세스는 ㅁㅁㅁㅁ변화에 OO한 소프트웨어 개발 방법론이다.

민첩, 요구사항

14 356

Q2.

애자일 방법은 개발초기부터 정해진 설계 내용을 유지하기 위해 정확한 Specification을 만들고 Increment과정을 진행한다. 초기 설계과정을 유지하기위해 Specification단계로 회기하는 것은 지양한다. (O/X)

COTATO

https://www.cotato.com

A2.

애자일 방법은 개발초기부터 정해진 설계 내용을 유지하기 위해 정확한 Specification을 만들고 Increment과정을 진행한다. 초기 설계과정을 유지하기위해 Specification단계로 회기하는 것은 지양한다. (O/X)

X

애자일은 요구사항 변화에 대응하기 위해 Specification으로 잦은 반복을 한다.

14 356

Q3.

애자일 방법론은 이 방법을 사용하여 plan-driven방법론에서 사용하는 문서화 과정을 간소화 했다. 다음중 이 방법에 해당하지 않는 것을 모두 고르시오.

1. 리팩토링(Refactoring)
2. 테스트 주도 개발(TDD)
3. 싱글톤 디자인 패턴(Singleton)
4. 페어프로그래밍(Pair Programming)
5. 프로토콜 지향 프로그래밍(POP)

COTATO

https://www.cotato.com

A3.

애자일 방법론은 이 방법을 사용하여 plan-driven방법론에서 사용하는 문서화 과정을 간소화 했다. 다음중 이 방법에 해당하지 않는 것을 모두 고르시오.

1. 리팩토링(Refactoring)
2. 테스트 주도 개발(TDD)
3. 싱글톤 디자인 패턴(Singleton)
4. 페어프로그래밍(Pair Programming)
5. 프로토콜 지향 프로그래밍(POP)

3,5

14 356

Q4.

최감자는 실무 프로젝트에서 적합한 개발방법론을 선택해야한다. 다음 프로젝트는 어떤 개발 방법론이 적합할지 고르시오. (애자일 , Plan-driven 중)

ㄱ. 자율 주행차 개발 프로젝트

ㄴ. UI,UX 디자인

A4.

최감자는 실무 프로젝트에서 적합한 개발방법론을 선택해야한다. 다음 프로젝트는 어떤 개발 방법론이 적합할지 고르시오. (애자일 , Plan-driven 중)

ㄱ. 자율 주행차 개발 프로젝트 - Plan - Driven

ㄴ. UI,UX 디자인 - 애자일

Q5. 애자일 방법론은 변화하는 요구사항에 민감하게 반응하기 위해서 최대한 많은 사용자를 대상으로한 프로젝트에 유용하게 사용될 수 있다. (O / X)

COTATO

https://www.cotato.com

A5. 애자일 방법론은 변화하는 요구사항에 민감하게 반응하기 위해서 최대한 많은 사용자를 대상으로한 프로젝트에 유용하게 사용될 수 있다. (O / X)

X

14 356

Q6. 애자일 프로세스에서 문서화 과정을 줄이고, 요구사항 변화에 빠르게 대응하기 위한 방법에 대한 설명으로 옳지 않은 것은?

1. 혼자서 코드를 짜지 않고 둘 이상이 같이 개발을 진행한다.
2. 개발을 작은 기능 단위인 Increment 단위로 나누어 개발을 진행했다.
3. 반복적이고 점진적인 방법으로 개발을 진행해 배포가 잦아졌다.
4. 개발자가 고객의 입장에서 요구사항을 제시하고 테스트를 진행한다.

A6. 애자일 프로세스에서 문서화 과정을 줄이고, 요구사항 변화에 빠르게 대응하기 위한 방법에 대한 설명으로 옳지 않은 것은?

1. 혼자서 코드를 짜지 않고 둘 이상이 같이 개발을 진행한다.
2. 개발을 작은 기능 단위인 Increment 단위로 나누어 개발을 진행했다.
3. 반복적이고 점진적인 방법으로 개발을 진행해 배포가 잦아졌다.
4. 개발자가 고객의 입장에서 요구사항을 제시하고 테스트를 진행한다.

개발자가 팀원이 고객으로 팀에 참여함

Q7. 다음은 애자일과 plan-driven 방식을 비교한 표이다 표에서 옳지 않은 부분을 고르시오.

	Agile Process	Plan driven
1. 요구사항	지속적인 요구사항 수립 및 변경	초기 요구사항 수립 및 엄격한 변경 관리
2. 프로젝트 규모	범용적인 소규모 프로젝트	Client가 있는 대규모 프로젝트
3. 설계	즉시 설계	사전 설계
4. 문서화	경량화된 문서화	상세한 문서화 강조
5. 역할	전체적인 팀워크를 중시	엄격한 역할 분리

A7. 다음은 애자일과 plan-driven 방식을 비교한 표이다 표에서 옳지 않은 부분을 고르시오.

	Agile Process	Plan driven
1. 요구사항	지속적인 요구사항 수립 및 변경	초기 요구사항 수립 및 엄격한 변경 관리
2. 프로젝트 규모	범용적인 소규모 프로젝트	Client가 있는 대규모 프로젝트
3. 설계	즉시 설계	사전 설계
4. 문서화	경량화된 문서화	상세한 문서화 강조
5. 역할	전체적인 팀워크를 중시	엄격한 역할 분리

Q8. 애자일 프로세스 관련 모델이 아닌 것은?

1. 스크럼 방식
2. XP 방식
3. Waterfall 방식
4. 린 소프트웨어 개발

COTATO

https://www.cotato.com

A8. 애자일 프로세스 관련 모델이 아닌 것은?

1. 스크럼 방식
2. XP 방식
3. Waterfall 방식
4. 린 소프트웨어 개발

14 356

Q9.

애자일 프로세스에선 요구사항 변화에 대응하기 위해 기능별로 작은 _____단위로 나누어 개발을 진행한다.

A9. 애자일 프로세스에선 요구사항 변화에 대응하기 위해 기능별로 작은 _____단위로 나누어 개발을 진행한다.

increment

Q10. 애자일 프로세스의 특징으로 적합하지 않은 것은?

1. 작은 increment 단위로 개발을 해 잦은 배포, 버전이 존재한다.
2. 소프트웨서 프로세스를 반복한다.
3. 문서를 아예 작성하지 않아 문서화에 대한 오버헤드가 없다.
4. 리팩토링을 통해 코드에 대한 이해도를 높인다.
5. 페어 프로그래밍을 통해 즉각적인 코드리뷰로 코드 품질을 개선한다.

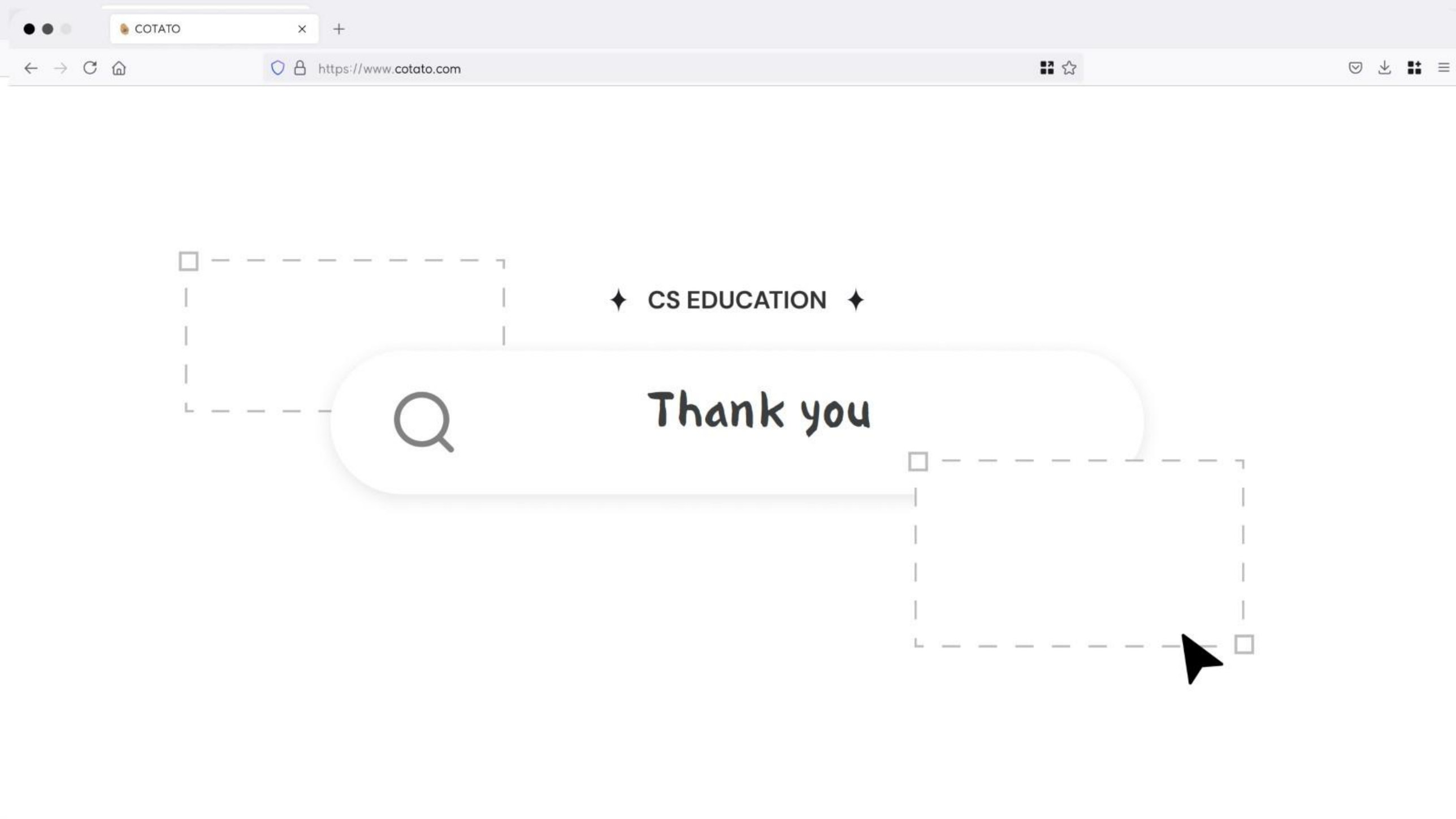
COTATO

https://www.cotato.com

A10. 애자일 프로세스의 특징으로 적합하지 않은 것은?

1. 작은 increment 단위로 개발을 해 잦은 배포, 버전이 존재한다.
2. 소프트웨서 프로세스를 반복한다.
3. 문서를 아예 작성하지 않아 문서화에 대한 오버헤드가 없다.
4. 리팩토링을 통해 코드에 대한 이해도를 높인다.
5. 페어 프로그래밍을 통해 즉각적인 코드리뷰로 코드 품질을 개선한다.

14 356


COTATO
https://www.cotato.com
CS EDUCATION
Thank you