효과적인 테스트 관리 도구

소프트웨어 테스트관리를 효율적이고 효과적으로

DEFECTMATE AI

Generative Ai Software Test management solution

2025. 6

생성형 AI 기반 요구사항 상세 자동화

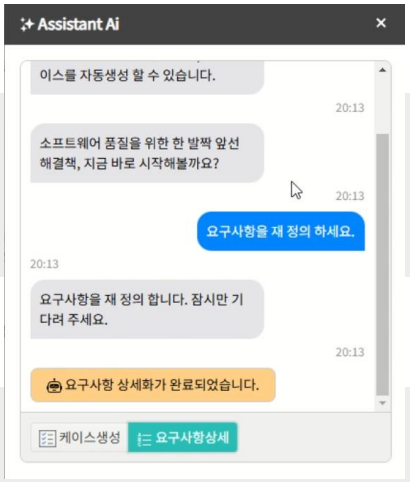
DefectMATE AI 기능은 입력된 단순 요구사항을 기반으로 상세한 요건 정의 및 업무 처리 내용을 자동으로 생성하여 요구사항 작성 효율성과 품질을 높여줍니다.

기본 등록 요구사항

Description

1. 사고 접수 및 보험금 청구를 간소화한 프로세스 제공.

2. 필요한 서류 업로드 및 상태 조회 기능 포함.



AI를 통해 적용된 상세화 된 요구사항

Description

1. 사고 접수 및 보험금 청구를 간소화한 프로세스 제공

- **기능설명:** 사용자는 웹사이트를 통해 복잡한 절차 없이 사고를 접수하고 보험금을 청구할 수 있어야 한다. 이는 사용자의 편의성을 증대시키며, 보험금 청구 과정에서 발생할 수 있는 스트레스를 줄일 수 있다.
- **세부 요구사항:**
  1. 사용자는 간편한 절차를 통해 사고를 접수할 수 있어야 한다.
  2. 사용자는 사고 접수 후 보험금 청구를 할 수 있어야 한다.
  3. 사용자는 가입한 보험 상품에 따른 보험금 청구 방법을 확인할 수 있어야 한다.

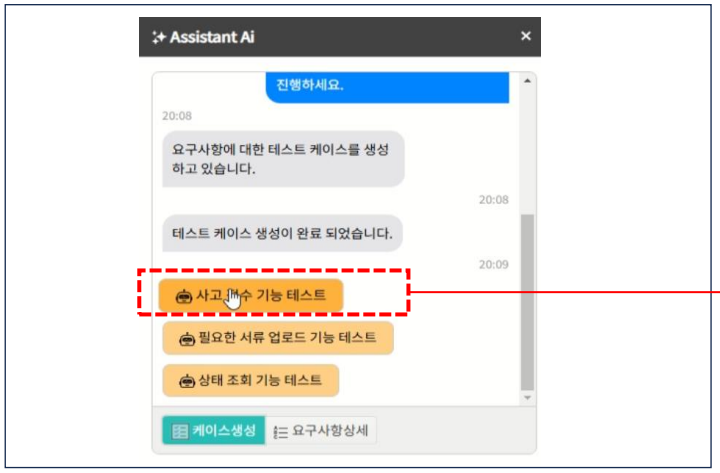
2. 필요한 서류 업로드 및 상태 조회 기능 포함

- **기능설명:** 사용자는 보험금 청구를 위해 필요한 서류를 웹사이트에 직접 업로드할 수 있어야 한다. 또한, 사용자는 청구한 보험금의 상태를 실시간으로 조회할 수 있어야 한다.
- **세부 요구사항:**
  1. 사용자는 웹사이트를 통해 필요한 서류를 업로드할 수 있어야 한다.
  2. 사용자는 업로드한 서류의 상태를 확인할 수 있어야 한다.
  3. 사용자는 보험금 청구 상태를 실시간으로 조회할 수 있어야 한다.

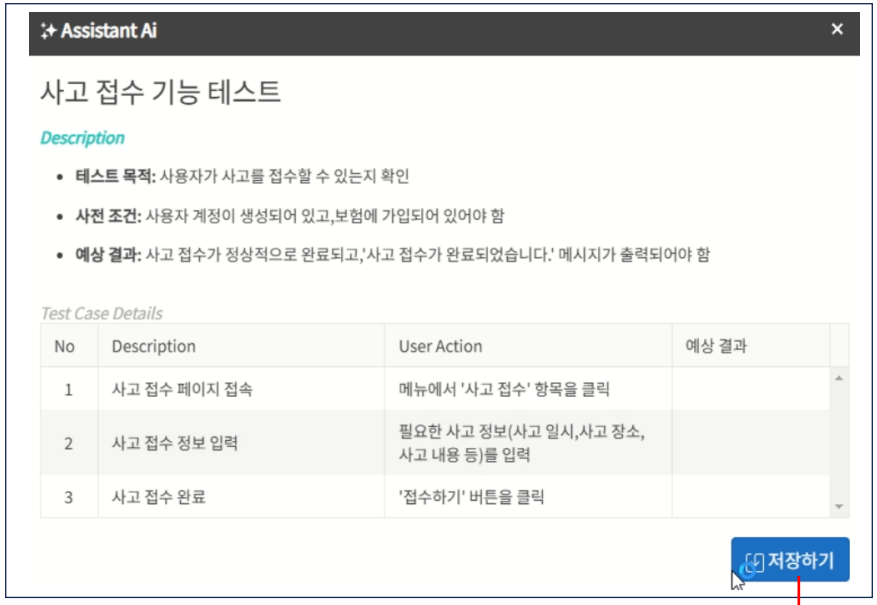
https://youtu.be/O20VDmf_wns?si=J7igmKzgIzG8nNO1

DefectMATE AI는 요구사항 입력만으로 테스트 항목 추천부터 테스트 케이스 자동생성 → 자동 등록까지 지원하여 테스트 설계 효율성을 극대화합니다.

① 기능 테스트 항목 자동 추천.제시



② 테스트 케이스를 자동 생성하여 제시 (사용자 검토)



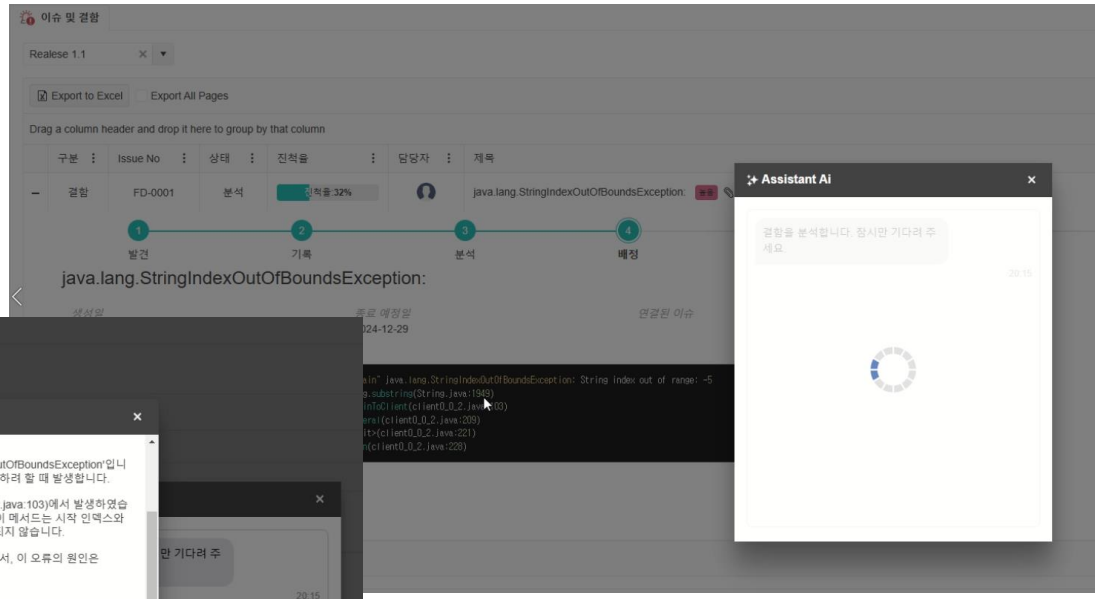
③ 저장만으로 DefectMATE 테스트 케이스 항목에 자동 등록

https://youtu.be/O20VDmf_wns?si=J7igmKzgIzG8nNO1

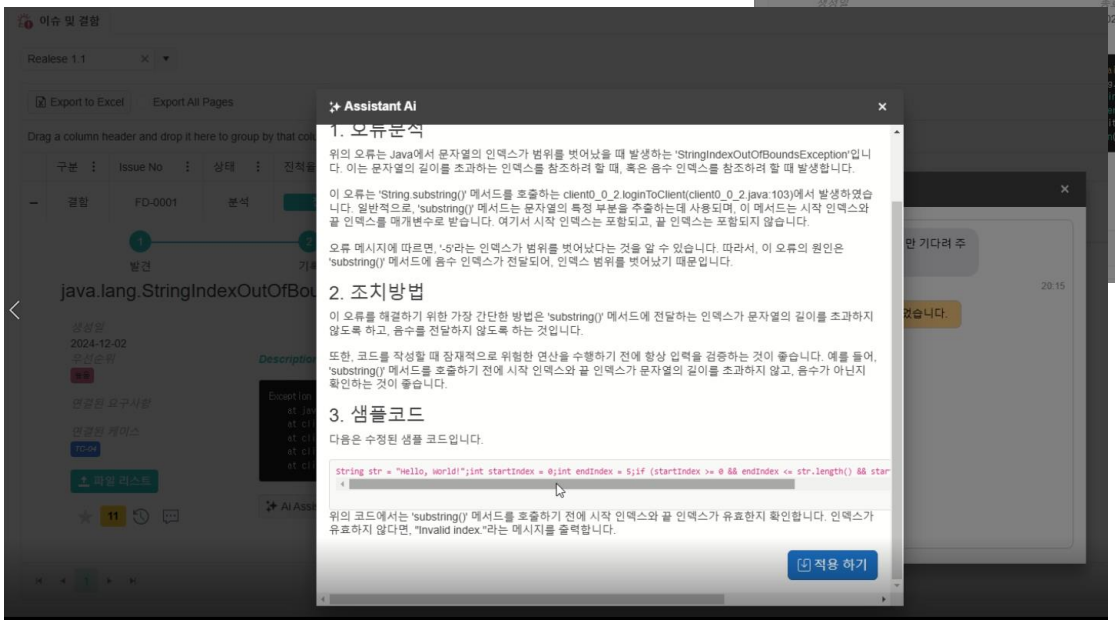
생성형 AI 기반 오류 분석 및 조치 방안 제시

DefectMATE AI는 발견된 결함 로그를 분석하여 오류 분석과 조치방법 그리고 처리가능한 샘플코드까지 제공합니다. 오류 원인이 되는 코드와 함께 분석 시 해당 코드의 오류 부분을 수정해 개발자 개발 향상성을 크게 도와 줍니다.

Assistant Ai를 통한 오류 메시지 분석



AI를 통해 적용된 분석 결과 및 수정 코드



https://youtu.be/O20VDmf_wns?si=J7igmKzgIzG8nNO1

DefectMATE AI는 Code Assistant를 함께 제공하고 있습니다. 가장 기본적인 개발 코드 요청에 더하여, 필요한 구조, 클래스, 환경구성 방안까지 모두 가이드로 제공 됩니다.

Assistant Ai를 통한 개발 지원

로그인됨: test@example.com

Code Assistant 입니다. 무엇을 도와드릴까요?

오후 4:23

java 기반의 post 형태의 api 서비스를 만드려고 해. json형태로 데이터를 받고 응답을 보내는 기본 서비스 코드를 작성해줘

오후 4:24

아래는 Java 기반의 POST 형태의 API 서비스를 만드는데 필요한 기본 코드입니다. 이 예제에서는 Spring Boot 프레임워크를 사용하고 있으며, JSON 데이터를 받고 응답하는 간단한 컨트롤러 클래스를 작성하였습니다.

먼저, 필요한 의존성을 추가하는 `pom.xml` 파일을 확인해주세요:

```
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <!-- JSON 처리를 위한 의존성 -->
    <dependency>
        <groupId>com.fasterxml.jackson.core</groupId>
        <artifactId>jackson-databind</artifactId>
    </dependency>
</dependencies>
```

그 다음, 데이터를 받을 DTO(Domain Transfer Object) 클래스를 작성합니다. 예시에서는 `User`라는 이름의 DTO를 사용하겠습니다:

```
public class User {
    private String name;
    private int age;
```

메시지를 입력하세요...

전송

로그인됨: test@example.com

그 다음, 데이터를 받을 DTO(Domain Transfer Object) 클래스를 작성합니다. 예시에서는 `User`라는 이름의 DTO를 사용하겠습니다:

```
public class User {
    private String name;
    private int age;

    // Getter와 Setter 메서드
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }
}
```

이제, POST 요청을 처리할 컨트롤러 클래스를 작성합니다:

```
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;
```

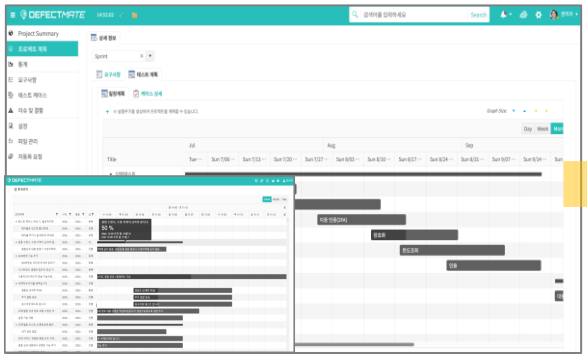
메시지를 입력하세요...

전송

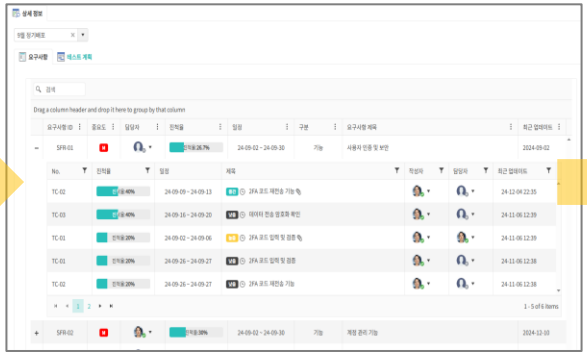
https://youtu.be/O20VDmf_wns?si=J7igmKzgIzG8nNO1

안정적 운영관리를 위한 체계적 테스트관리 시스템이 필요합니다.

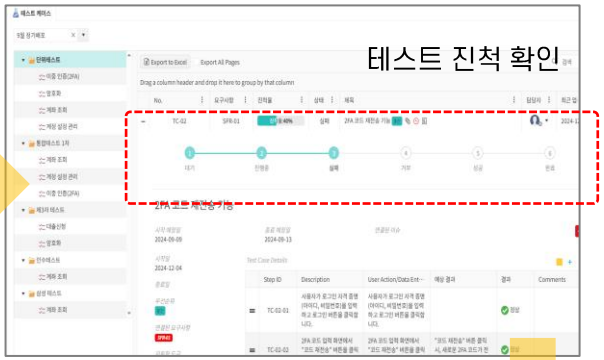
테스트 관리는 소프트웨어 개발 생명주기에서 중요한 단계로, 소프트웨어 제품의 품질을 유지하고 개선하기 위해 계획, 실행, 결과 및 결함 추적 등을 체계적으로 관리하는 프로세스입니다. 테스트 관리는 프로젝트 팀이 목표에 도달하고 소프트웨어 제품이 예상대로 작동하는지 확인하는 핵심 도구 및 기술을 제공합니다.



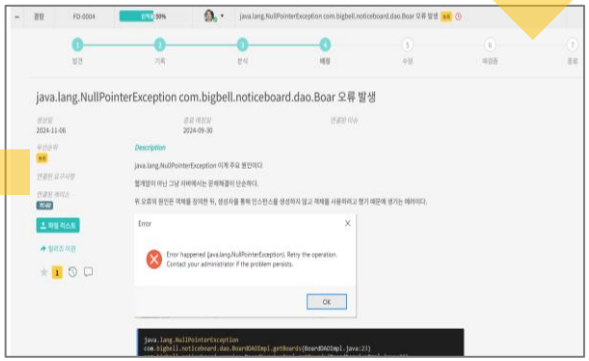
테스트 계획 및 설계



요구사항(SR) 정의



요구사항에 따른 테스트 계획 수립 및 설계



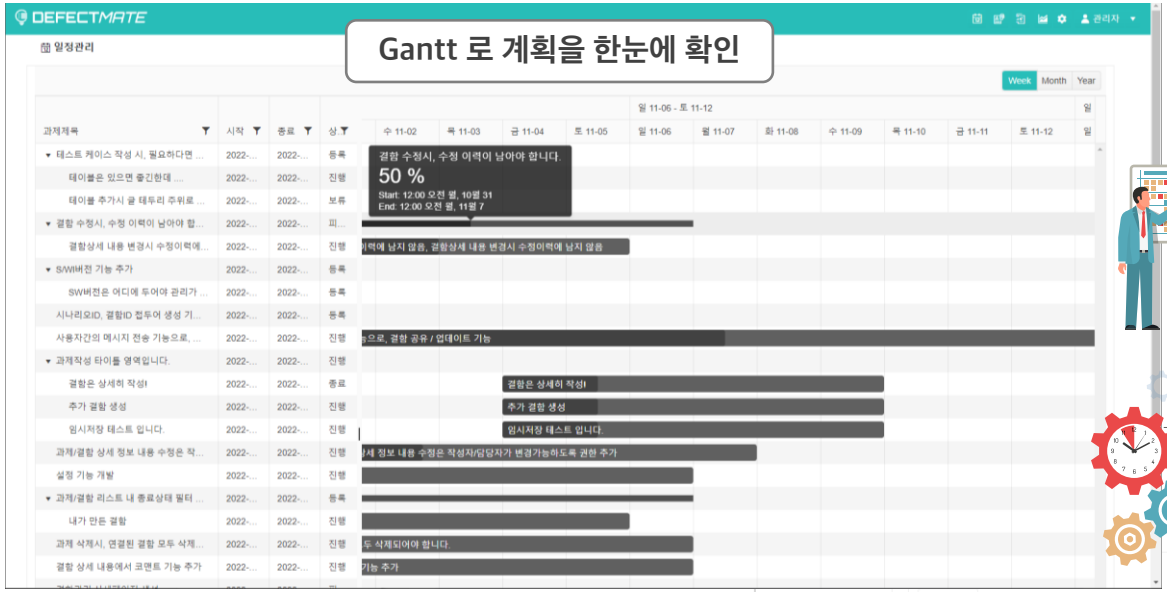
테스트 수행과 결함관리

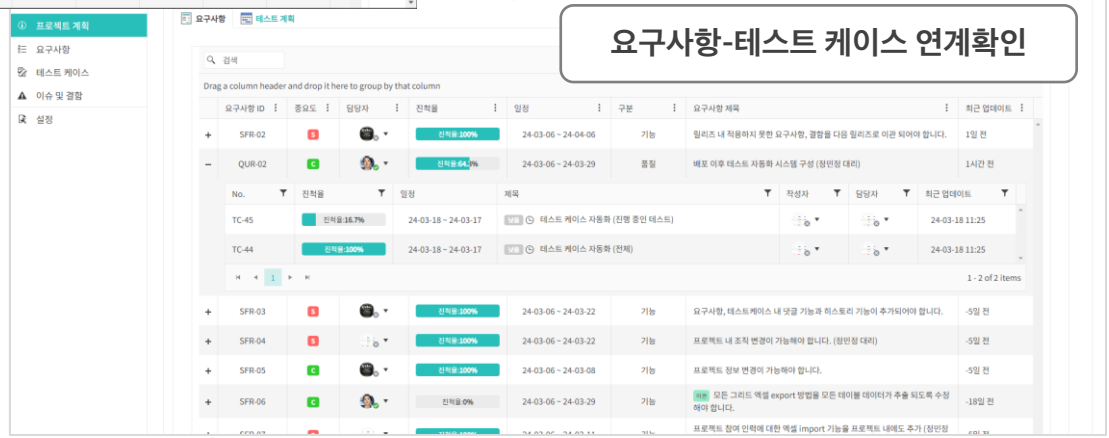
산출물 등록 및 확인

요구사항 진척 확인 및 종료 확인

테스트 계획의 복잡성을 한눈에 확인

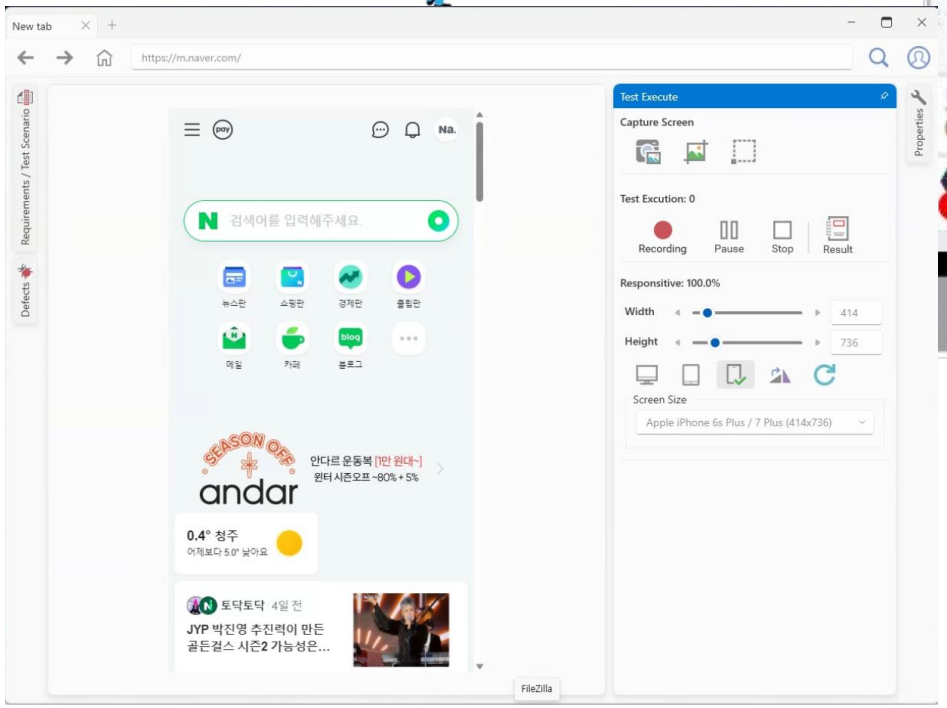
소프트웨어 테스트 관점에서 테스트 계획은 테스트 활동에 대한 접근 방식, 범위, 자원 및 일정을 계획적으로 설계하는 중요한 단계입니다. 이 프로세스는 테스트 노력이 체계적으로 조직되고 프로젝트 목표와 일치하는지를 확인합니다. 테스트 계획에는 테스트 목표, 테스트 산출물, 종합적인 테스트 범위를 달성하기 위한 전략 등이 포함됩니다.





자동화된 도구나 수동 테스트를 통해 테스트 케이스를 실행

테스트 실행은 소프트웨어의 기능이나 시스템이 예상대로 작동하는지 확인하기 위한 핵심적인 프로세스입니다. 이 단계에서는 테스트 케이스를 실행하고 결과를 기록하여 소프트웨어의 실질적인 동작을 확인합니다. 테스트 실행은 자동화된 도구나 수동 테스트를 통해 수행될 수 있으며, 테스트 결과의 신속한 분석이 중요합니다.



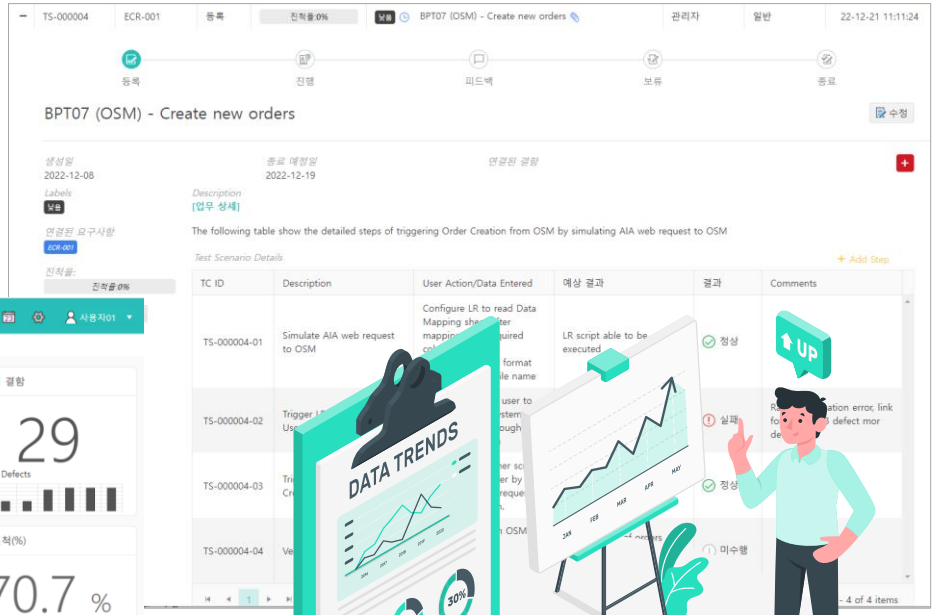
기능 테스트 수행도구

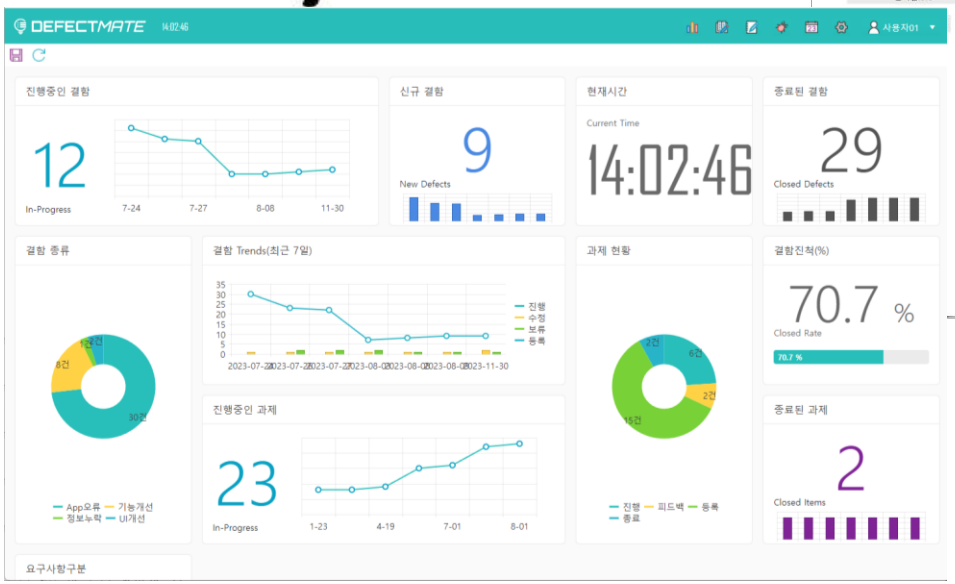
자동화 테스트 수행도구

테스트 실행 중 얻은 결과를 분석하고 문서화

결과 관리는 테스트 과정에서 핵심적인 부분으로, 테스트 실행 중 얻은 결과를 분석하고 문서화하는 작업을 포함합니다. 이 단계는 소프트웨어의 성능을 이해하고 예상대로 동작하지 않는 부분을 식별하는 데 체계적인 접근을 제공하며, 효과적인 결과 관리는 신중한 의사 결정에 기여하며 소프트웨어 품질 향상을 도모할 수 있습니다.

수행결과를 기록하고 분석

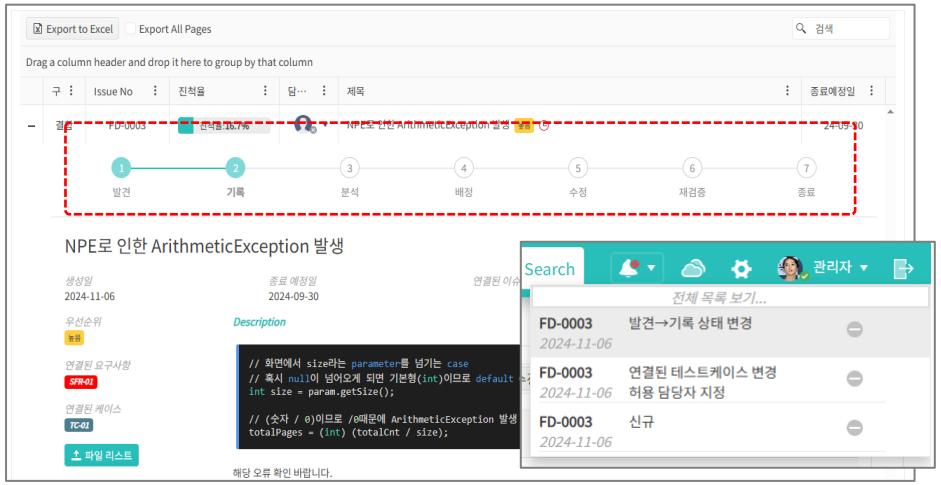




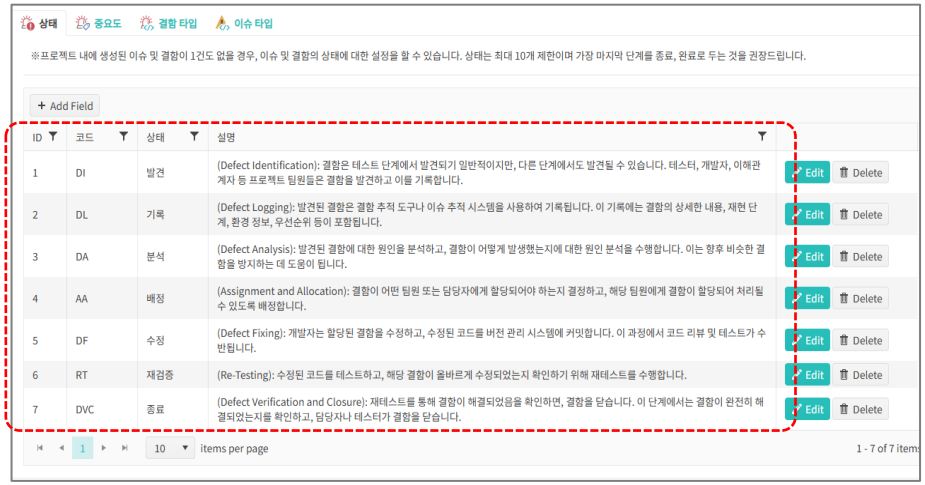
진행현황을 실시간 대시보드로 확인

테스트 중 발견된 결함을 식별하고 기록

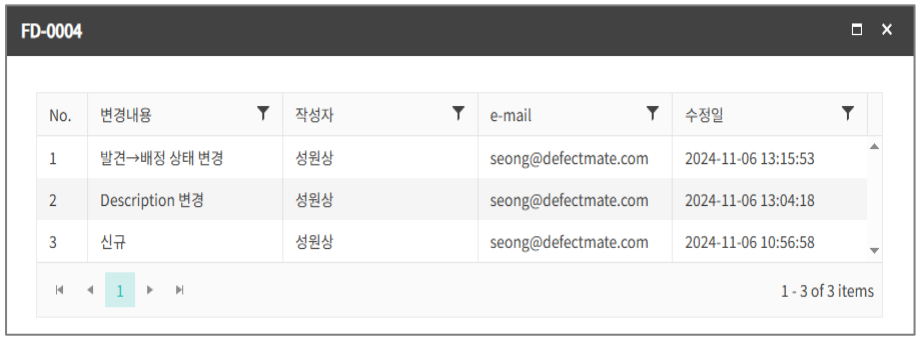
결함 추적은 테스트 중 발견된 결함을 식별하고 기록하여 이후 조치 및 개선을 위한 과정을 의미합니다. 효과적인 결함 추적은 소프트웨어 품질을 향상시키는 데 도움이 되며, 결함이 어떻게 발생했는지 이해하고 해결책을 마련하는 데 중요한 역할을 합니다.



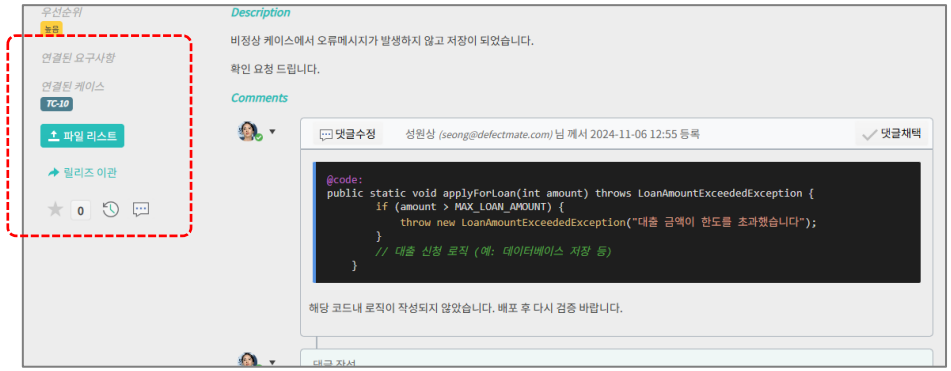
결함 진척 별 담당 및 상태 확인/알람 기능



진척 관리를 위한 커스텀 기능제공



변경 이력 저장



발생된 요구사항, 테스트케이스 확인/첨부 및 댓글 기능 제공

서비스 라이프사이클 전반에 걸친 효율성, 협업, 통제 강화

DEFECTMATE

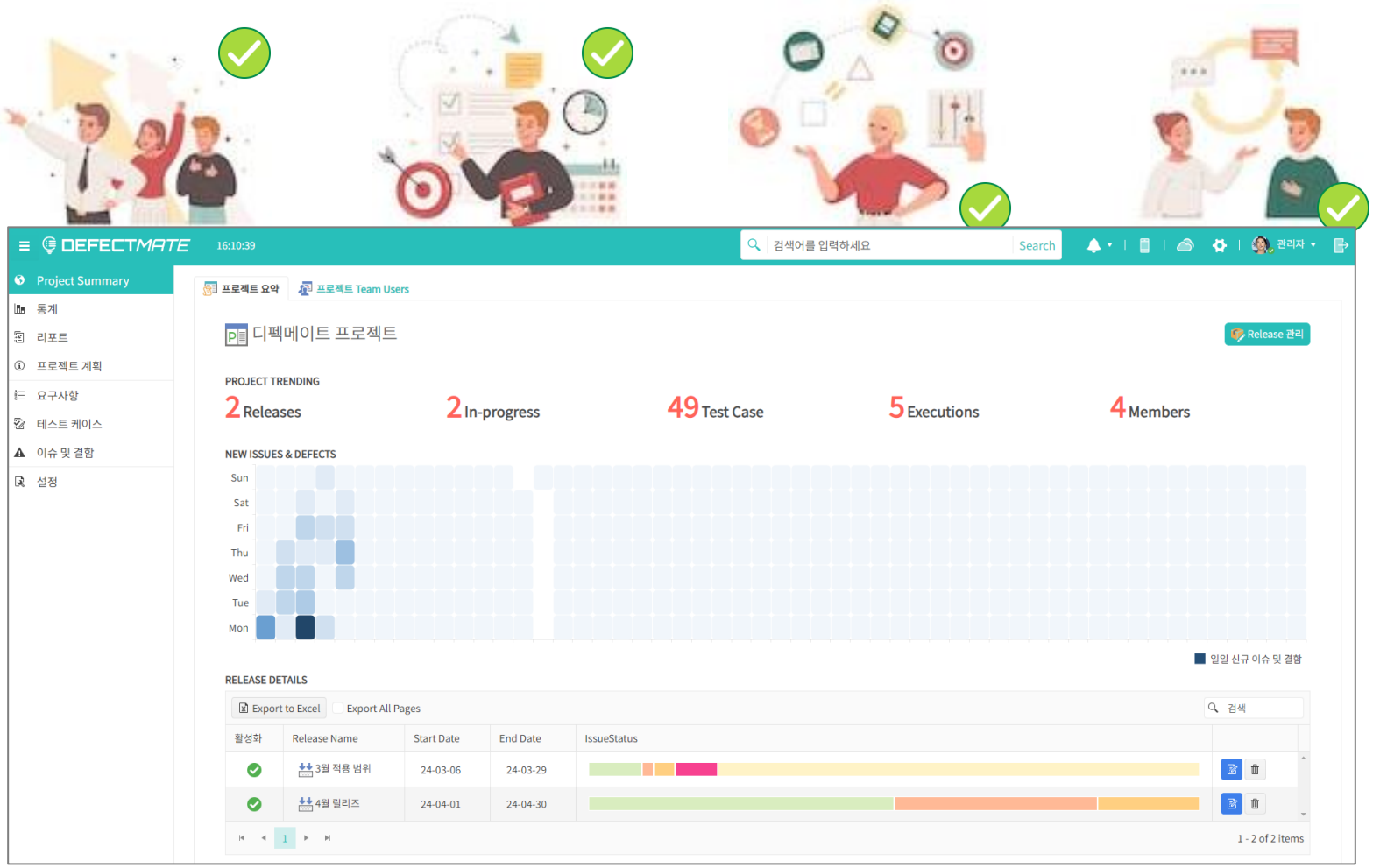
통합된 테스트 관리는 소프트웨어 개발 생명주기 동안 다양한 테스트 활동을 종합적으로 조정하고 통합하는 접근 방식으로, 테스트 계획, 실행, 결과 관리, 결함 추적 등을 효과적으로 통합함으로써 전체적이고 체계적인 테스트 프로세스를 강조합니다. 통합된 테스트 관리 도구는 전체 테스트 라이프사이클 동안 효율성, 협업, 그리고 통제를 강화하는 데 중요한 역할을 합니다.

테스트 프로세스 향상, 빠른 결함 해결, 생산성 향상, 프로젝트 일정 준수

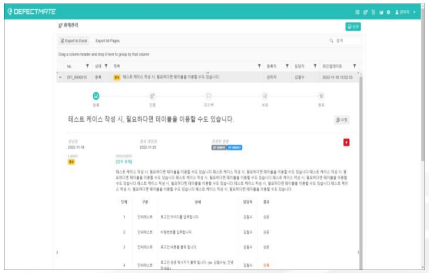
효과적인 테스트 관리는 소프트웨어 품질 향상과 프로젝트 성공에 기여하는 여러 가지 장점을 제공합니다. 실제적인 테스트 프로세스 향상, 빠른 결함 해결, 생산성 향상, 프로젝트 일정 준수 등이 포함되고, 팀 간 협업이 원활하게 이루어지며 품질 향상에 필요한 데이터 기반 의사 결정이 가능해집니다.

효과적인 테스트 관리는 성공적인 소프트웨어 개발 프로젝트의 핵심

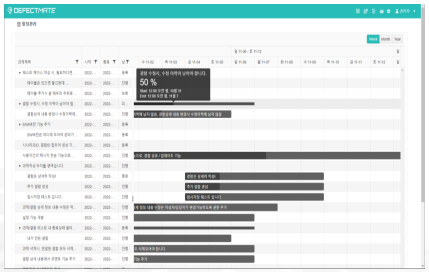
계획, 실행, 결과 관리 및 결함 추적을 체계적으로 수행함으로써 높은 품질의 소프트웨어를 제공합니다. 테스트 관리 도구의 통합된 접근 방식은 협업을 촉진하며 테스트 프로세스에 대한 통제를 강화하고 의사 결정에 기여하고, 지속적인 개선과 테스트 관리의 반복적인 특성이 프로젝트를 성공적으로 이끌어 줍니다.



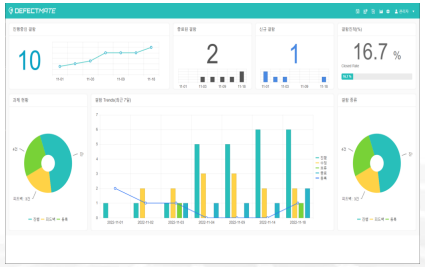
소프트웨어 운영/개발 기간 중 발생하는 결함에 대한 모든 이력관리를 할 수 있습니다. 우선순위를 정해 논의하며 실시간 시각적 데이터를 기반으로 진행중인 결함에 대해 놓치지 않고 최고의 품질을 유지시켜 줍니다.



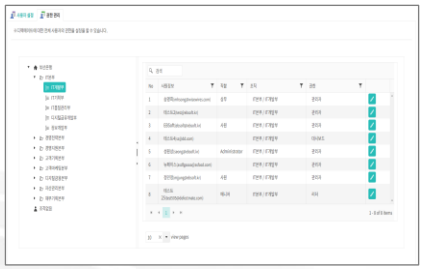
과제 / 테스트 케이스 / 결함 관리



테스트 수행 스케줄 관리



실시간 품질 모니터링



그룹/부서/사용자 관리기능

최상의 품질 관리

1 과제 / 테스트 케이스 관리

과제 변경에 따른 히스토리 관리부터 스케줄관리가 가능하며, 과제 및 테스트케이스 그리고 결함까지 한번에 확인

과제 작성만으로 월, 주, 일별 테스트 케이스를 한눈에 확인하고 테스트 진척관리까지 가능

2 실시간 품질 모니터링

다양한 그래프를 이용하여 실시간 품질 모니터링이 가능하며, 빠른 의사결정에 정확한 데이터를 제공

3 필터기능을 통해 빠른 리포트 확인

과제 및 결함관리에서 필터기능을 통해 즉각적인 리포트 생성이 가능

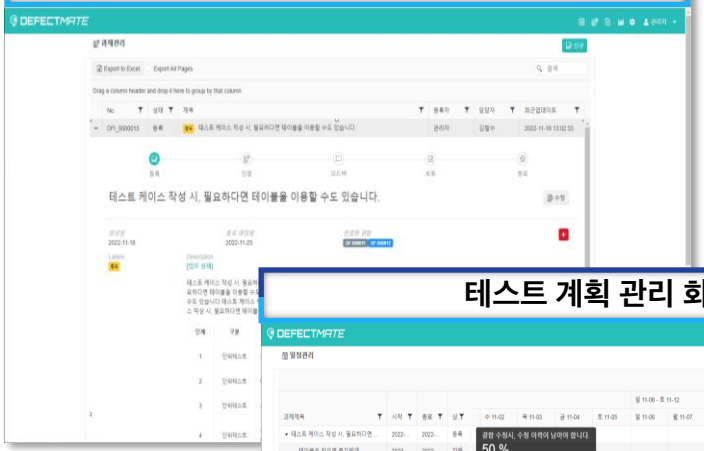
공정에 따른 수행 화면 예시

DEFECTMATE

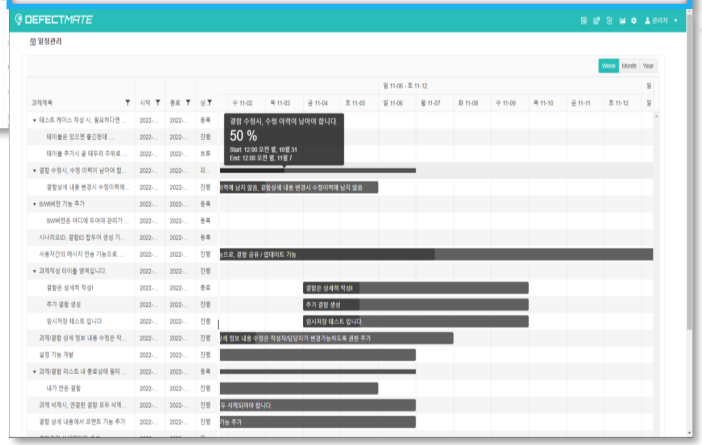
1. 테스트 계획 관리 → **2. 테스트 수행** → **3. 결함 관리**

테스트 시나리오 → 테스트 케이스 → 테스트 시나리오 수행 (자동 테스트 · 수동 테스트) → 테스트 결과 분석 및 결함 관리

과제/테스트케이스 관리 화면

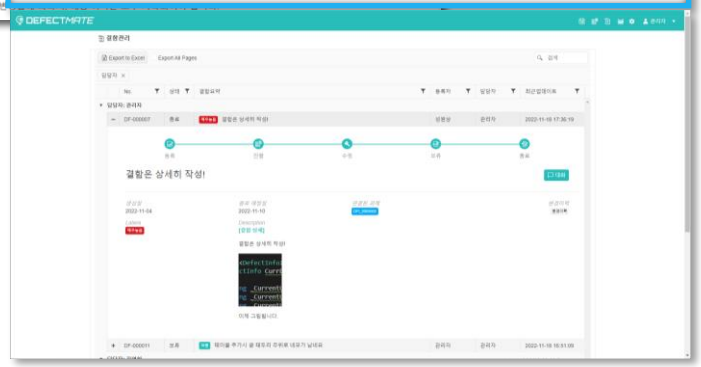


테스트 계획 관리 화면



과제/테스트케이스 관리 화면

결함 관리 화면



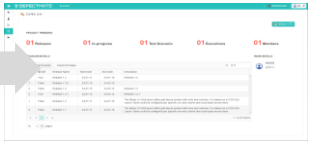
프로젝트 관리부터 테스트 자동화 수행까지 모든 Life Cycle에 대해 모니터링이 가능하도록 구축 합니다. 특히 개발자 테스트 기록을 자동화 솔루션과 연동하여 주요 단위 업무의 회기테스트 수행을 통해 효율적 운영이 가능합니다.

프로젝트 및 릴리즈 관리

릴리즈 단계별 통계정보, 요구사항, 테스트 케이스, 진척, 결함 관리 등 프로젝트 시작부터 종료까지 통합 관리

애자일 / 스프린트 방식의 개발 관리에 적합한 관리

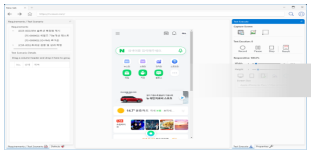
프로젝트 관리


애자일 버전관리

개발자 테스트 수행 결과 등록 및 자동화 연동

개발자 테스트 기록을 저장하여 테스트 자동화 스크립트를 생성하고, 일정 및 리소스관리를 통해 프로젝트를 효율적 진행

개발자 테스트 도구를 통해 수행결과를 실시간 확인 및 테스트 자동화 수행


개발자 테스트 도구

자동화 테스트 수행

테스트 케이스 재사용

타 프로젝트, 릴리즈 버전 별 사용한 테스트 케이스를 그대로 활용하여 수정 사용 가능하며, 테스트 결과의 일관성 유지로 품질향상을 기대

재사용 가능한 테스트 케이스로 시간 및 비용절감 커버리지 향상

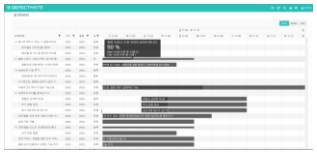
테스트 케이스 확장

테스트 케이스 재사용

실시간 진척 및 커버리지 분석

타 프로젝트, 릴리즈 버전 별 사용한 테스트 케이스를 그대로 활용하여 수정 사용 가능하며, 테스트 결과의 일관성 유지로 품질향상을 기대

수동 및 자동화 테스트 수행결과는 실시간 진척 및 빠른 의사결정


실시간 진척

실시간 대시보드

개발자 업무 생산성 측정

수행이력, 테스트 결과 기록 등을 분석하여 개발자 업무 생산성 측정을 위한 KPI를 개발 가능하고, 생산성 측정을 통해 작업시간 효율화와 프로세스 개선

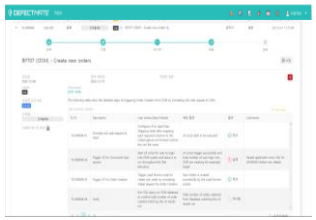
작업시간 분석과 프로세스 개선 기회를 통해 자원 할당 최적화 및 예측 가능성 증가

개발자 업무 생산성 분석

개선기회 및 성과평가

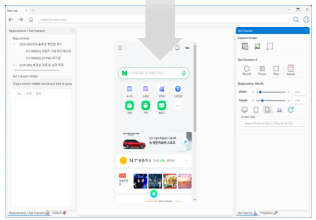
소프트웨어 품질 확보를 위한 개발자 테스트 수행결과를 기록하고 자동으로 리포트 합니다. 수집된 테스트 결과는 자동화 테스트 도구에 기록되어 자동화 수행 담당자가 관리하고 자동화 수행합니다.

1 개발자 테스트 결과를 기록하여 테스트 자동화 수행



테스트 관리도구

스케줄 된 일정에 맞춰 개발자 및 테스터들에게 해당 테스트 케이스가 알람이 됩니다. 이메일, SMS, 사내 메신저 등을 통해 팀원간 업무 상태에 대해 자동 알림이 발생 합니다.



모바일 및 단말 테스트 도구

개발자 PC내 설치된 테스트 수행도구를 통해 할당된 테스트 케이스를 수행하는 것만으로 테스트 결과가 기록되고 처리 됩니다. 기록된 테스트 결과는 리포트 및 통계기록에 저장되어 실시간으로 진척 현황 확인이 가능합니다.

테스트 자동화 도구

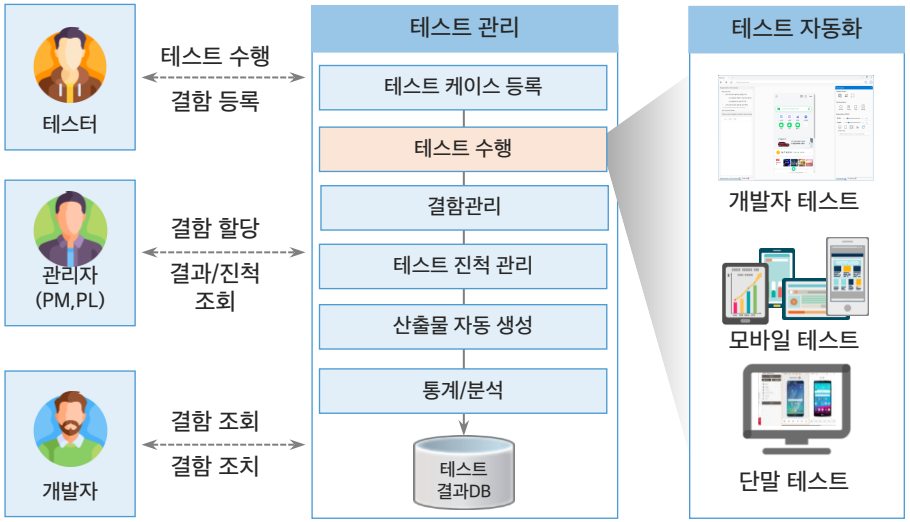
테스트 수행도구를 통해 개발자 테스트 수행기록을 그대로 테스트 자동화 스크립트로 자동 생성 가능합니다. 주요 테스트 케이스의 경우, 스크립트 수정을 통해 통합테스트 및 사용자 테스트로 통합하여 지속적인 테스트 자동화로 사용할 수 있습니다.

2 기능테스트 결과를 테스트 자동화 연동

테스트 수행 후 결과 자동 저장 및 공유

- 테스트 수행도구를 통해 결과가 자동 기록되고, 자동화 테스트 스크립트 작성
- 실시간 결과 리포트 수집

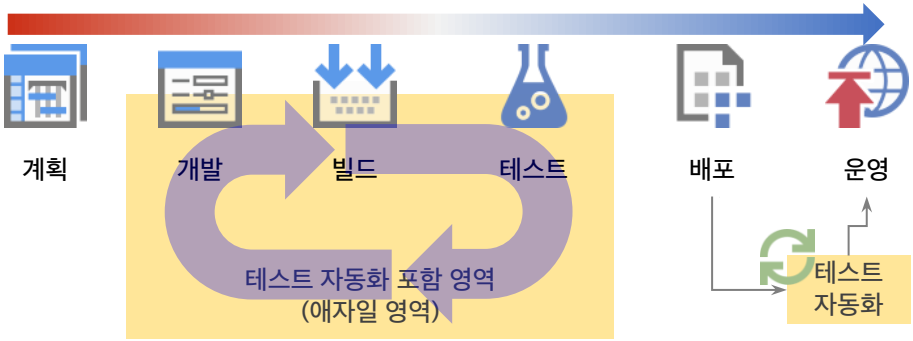
3 팀 협업 강화



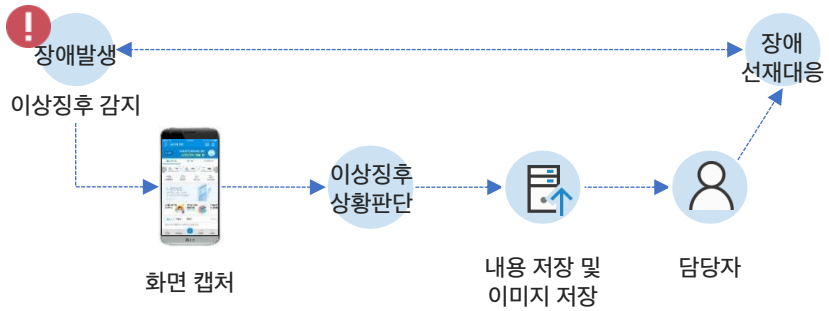
테스트 자동화는 개발 Life Cycle 전반에 걸쳐 적용 가능합니다. 애자일 기간동안(개발-빌드-테스트) 외 배포 도구와 통합하여 배포이후 테스트 자동화 업무를 수행하여 운영환경에서도 안정적으로 동작하는지 확인 가능합니다.

4 CI/CD Pipeline 내 테스트 자동화 통합

CI/CD Pipeline 내 테스트 자동화가 통합됩니다. 특히, Jenkins, Bamboo 및 Jira 등의 솔루션과 함께 통합되어 배포 이후 테스트 자동화 테스트 수행으로 운영환경에서도 안정적으로 동작하는지 확인 가능합니다.



5 금융 업무시스템. 비 대면 서비스 모니터링 시스템 구축



6 개발자 업무 생산성 측정[1)]

Task Mining 기술을 활용하여 개발자 업무 생산성 측정을 수행할 수 있습니다. 이는 소프트웨어 개발 수명주기 전반에 걸쳐 여러 지점을 모니터링 할 수 있으며, 개발된 코드 줄 수가 아니라 프로세스적 측면에서 측정 합니다.

- 솔루션 1) 깃허브, 코파일럿 기술 등을 이용하여 개발자 생산성 향상
- 솔루션 2) 프로세스 관점에서의 워크플로우 병목현상 확인

실시간 진척 확인 | 일정 내 개발자 업무 분석 | 개선 기회 확인

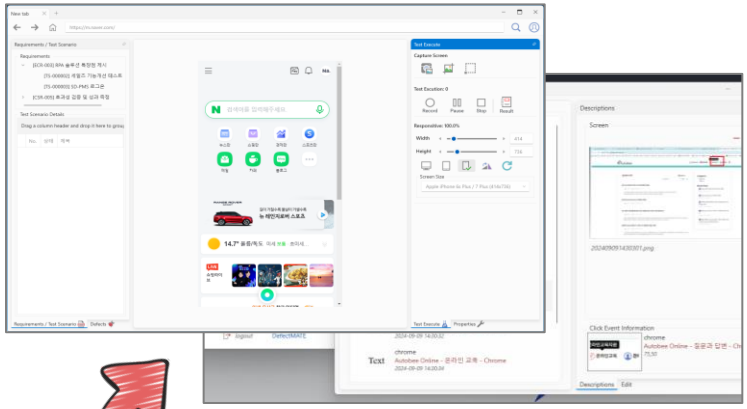
※ 대부분의 소프트웨어는 개별 개발자가 아닌 개발자 팀에 의해 만들어 지므로 생산성 향상의 효과를 제대로 측정하기 위해 장기간에 걸쳐 팀의 생산성(ex. 단위 시간당 가치)을 평가해야 합니다.

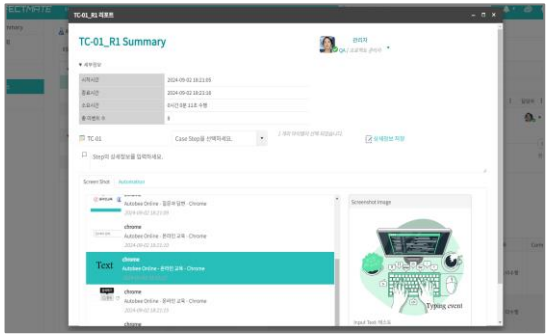
1) 업무생산성 측정방법은 개발자PC에 Agent가 설치되어 프로세스 사용량을 측정합니다.

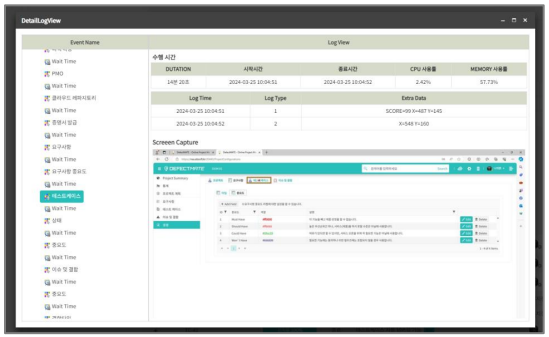
CI/CD 파이프라인의 자동화 테스트 플랫폼의 활용의 가장 가치 있는 기능 중 하나로 개발자의 수동 테스트 결과와 자동화된 테스트를 결합하여, 더 효율적이고 일관된 품질 보증을 제공합니다.

1 테스트 도구를 이용하여 테스트 수행결과를 기록, 저장할 수 있습니다. 기록된 결과는 DefectMATE에 저장되어 테스트 결과를 수행시간, 소요시간 등을 포함하여 상세히 기록 합니다.

2 저장된 테스트 기록을 QA담당자나 PL에서 검증하고 필요에 따라 자동화 시나리오로 전환 할 수 있습니다. 필요에 따라, 테스트 데이터나 테스트 조건 등을 조정하여 수행 가능 합니다.



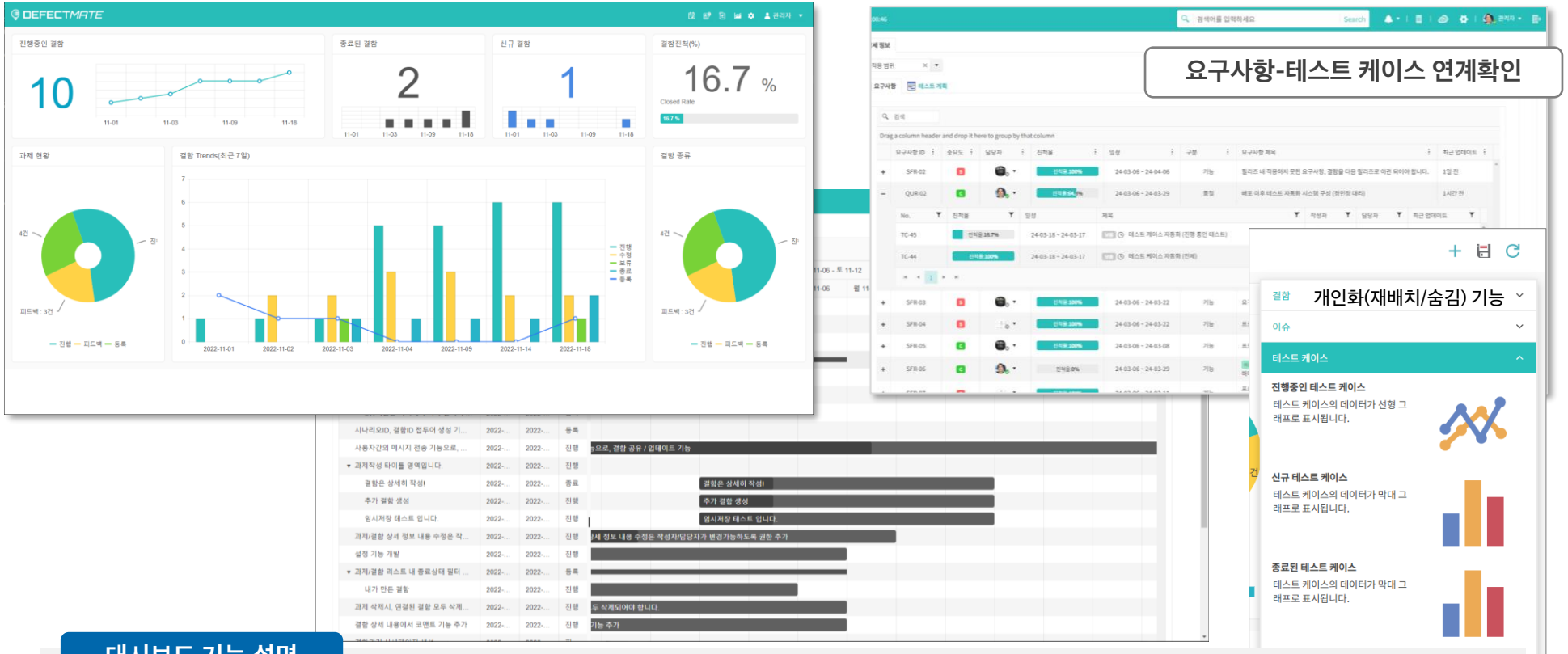




4 자동화 테스트 수행기록을 확인하고 검토/분석 가능합니다. 테스트 수행 결과에 따라 결과가 집계되고 통계가 함께 기록되어 더 효율적이고 일관된 품질 보증을 제공 합니다.

3 스케줄 되거나 즉시(Queue) 자동화 테스트 수행이 가능 합니다. 이 프로세스는 자동화 테스트 도구가 포함되어야 수행 가능합니다.

이슈, 요구사항, 결함과 관련된 전체 진척 내용의 실시간 정보가 각종 차트를 통해 제공 됩니다.



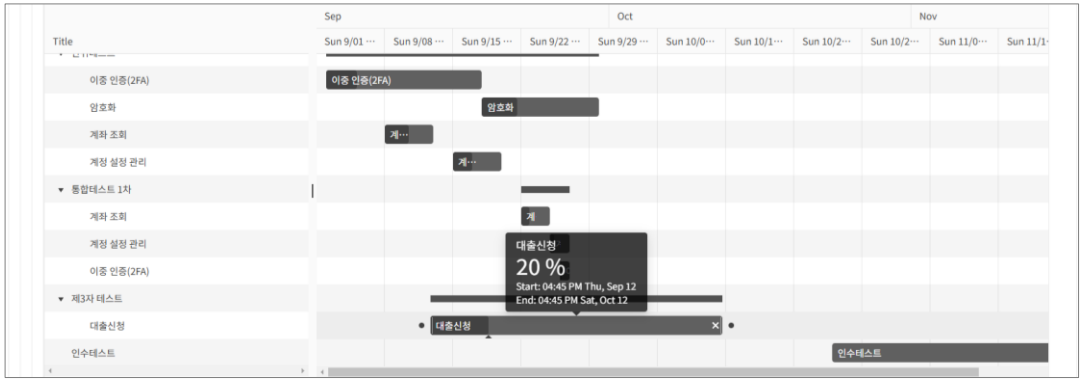
대시보드 기능 설명

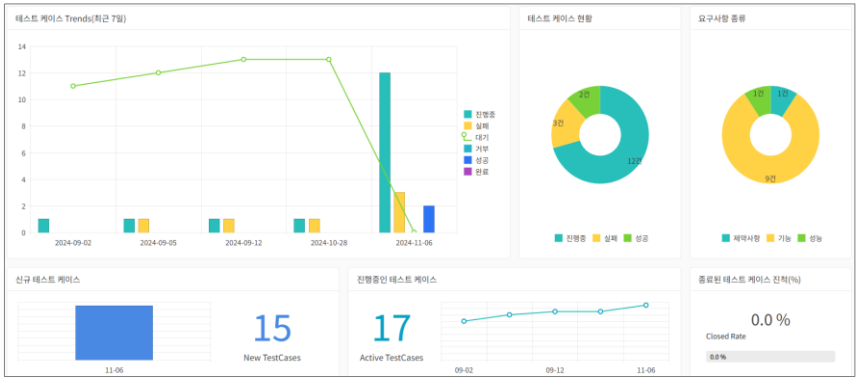
① 직관적으로 확인 가능한 숫자형 통계보드

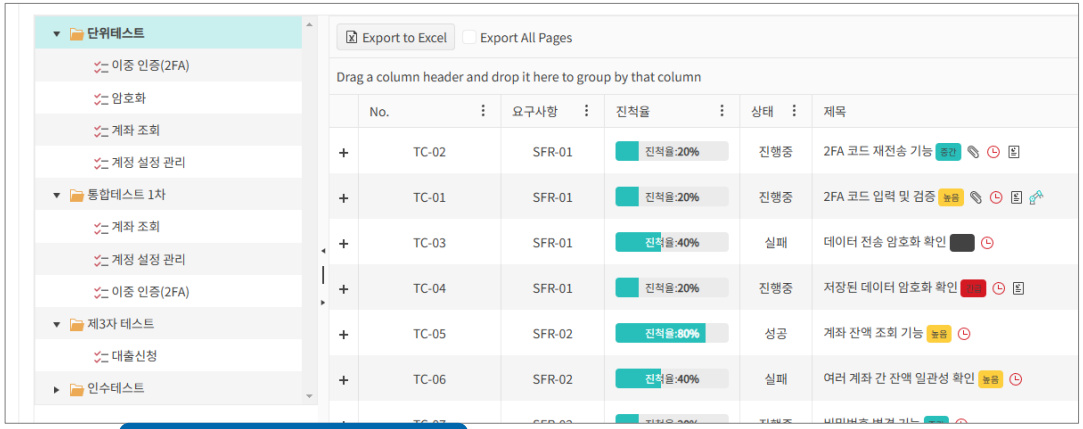
② SR 요구사항, 테스트 케이스, 결함/이슈 에 관련된 다양한 그래프를 제공하며, 해당 그래프는 필요에 따라 개인화(재배치/숨김) 기능을 함께 제공하고 있습니다.

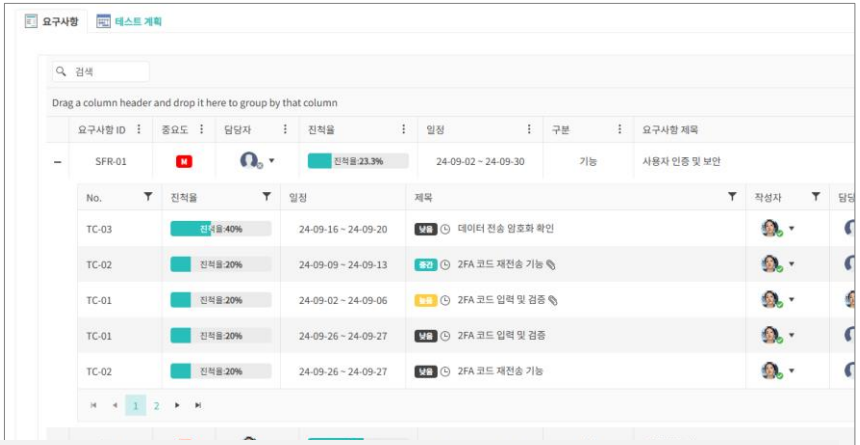
③ 해당 대시보드는 커스터마이징이 가능하며, 실시간으로 대시보드에 반영됩니다.

요구사항, 리스크, 테스트케이스, 테스트 계획, 테스트 실행, 테스트 결과, 결함 통계 와 보고서 생성 이력 관리가 가능합니다.







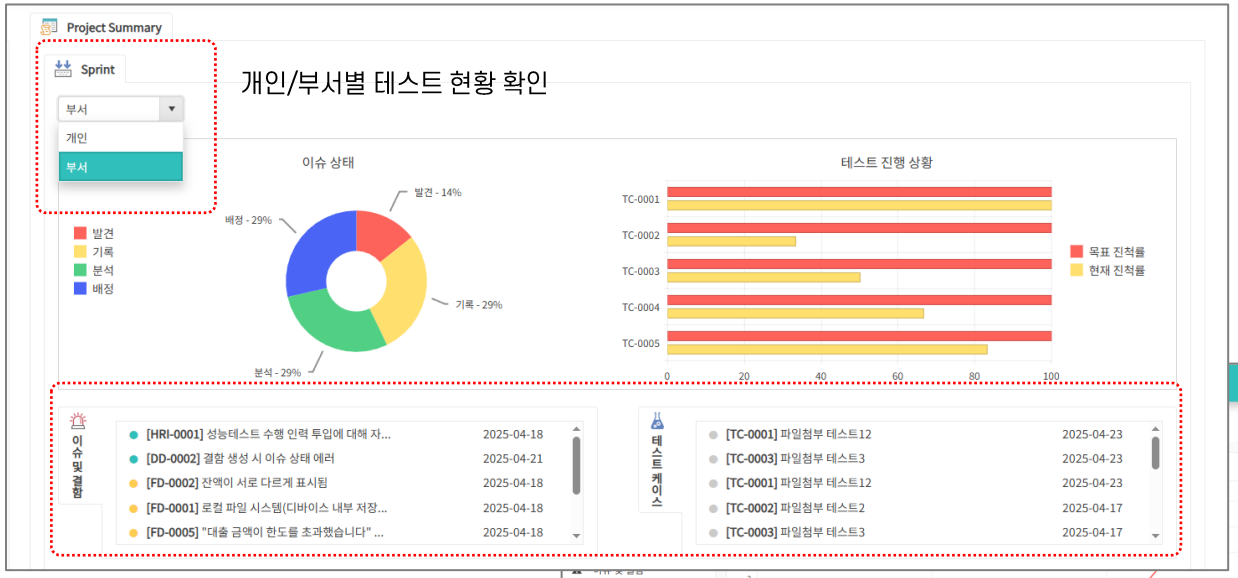


기능 설명

① 대시보드를 통해 기본적인 요약정보가 제공 되고 있습니다.

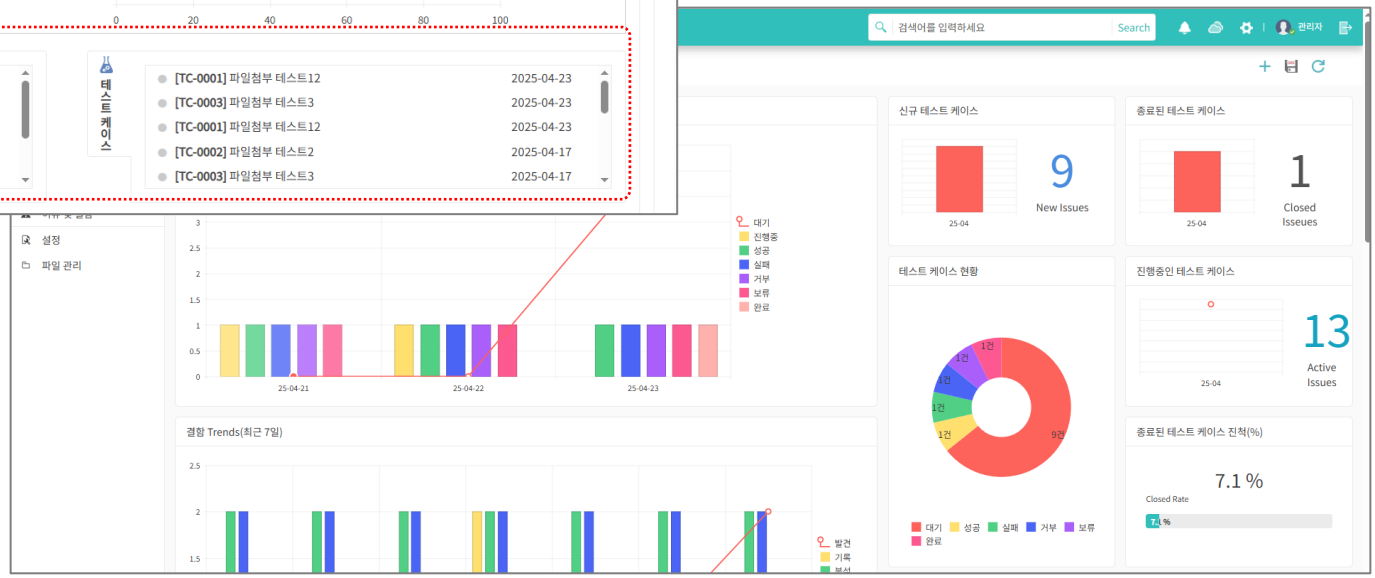
② 통계 및 상세 보고 내용은 고객과 협의하여 보고서 형태로 커스터마이징 하여 제공 될 수 있습니다.

개인/팀(부서)별 테스트 현황 대시보드와 함께, 개인화 페이지를 제공 합니다. 이 기능을 통해 사용자는 할당된 테스트, 결함 등을 확인하며, 지연된 업무나 목표 케이스를 확인할 수 있습니다.

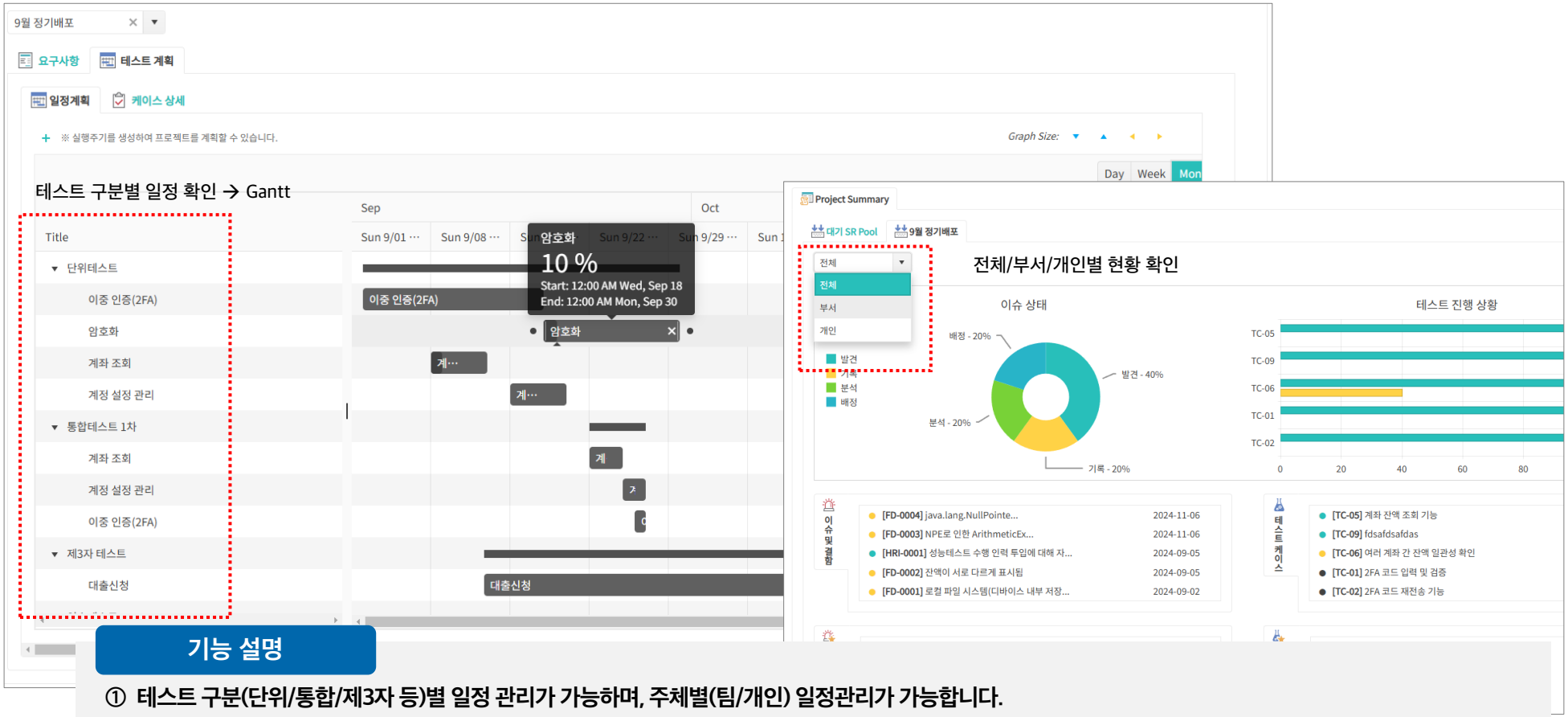


개인/부서별 진행중인 결함 및 테스트 현황 확인이 가능합니다.

부서/개인별 대시보드 확인이 가능합니다.



테스트 구분, 주체별 일정(공정) 관리와 관련 업무 담당자에게 알림 발송기능을 제공 합니다.



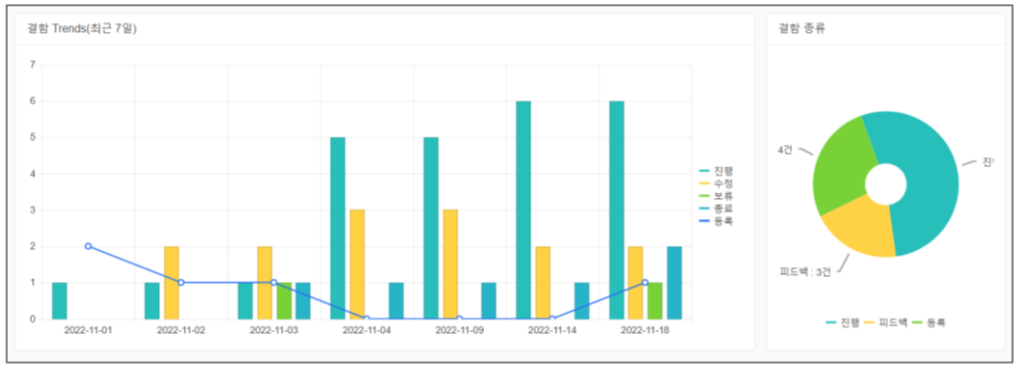
기능 설명

① 테스트 구분(단위/통합/제3자 등)별 일정 관리가 가능하며, 주체별(팀/개인) 일정관리가 가능합니다.

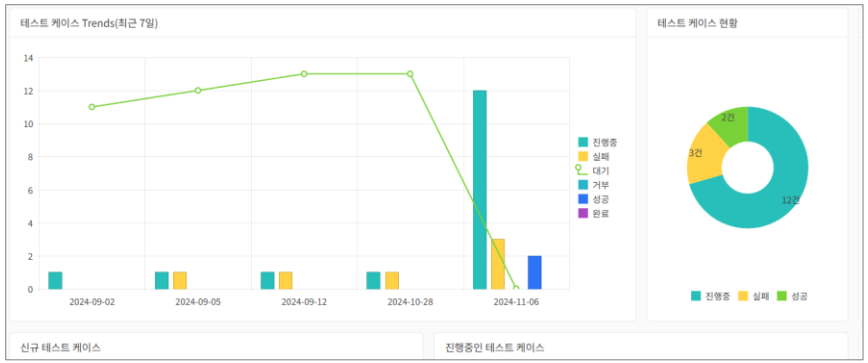
② 개인화된 페이지에서 개인/부서별 일정을 확인할 수 있으며, 배정된 테스트 시나리오/결함 등을 확인 할 수 있습니다.

③ 테스트 진행에 따른 이메일/메신저 연동 알림을 제공하며, 일정내 종료되지 못한 업무들 역시 알림 기능을 제공합니다.

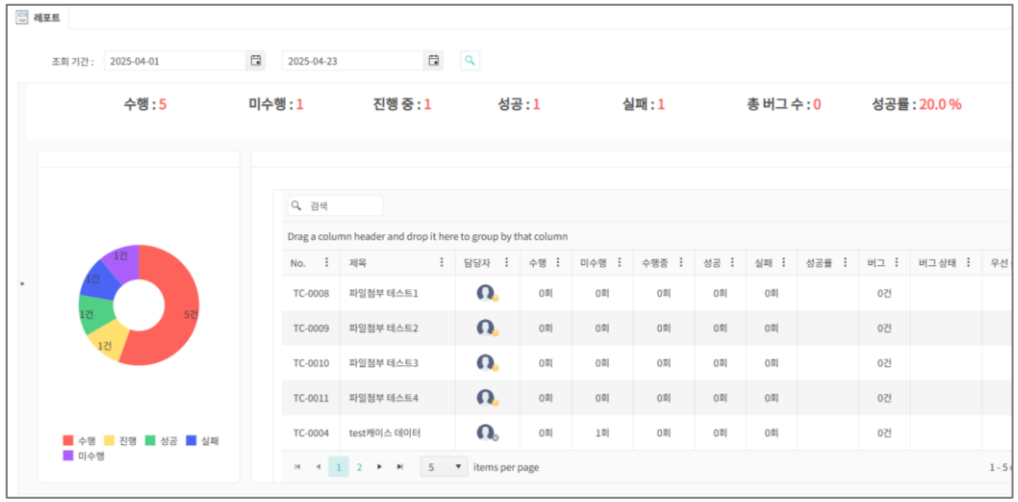
일자별 수행 실적보고서를 조회, 다운로드 가능하며, 많은 종류의 그래프와 차트를 제공 합니다. 해당 그래프는 모두 drag & drop 기능을 통해 위치를 변경 가능하며, 일별 처리량 등을 모두 제공하여 테스트 현황을 즉시 파악할 수 있습니다.



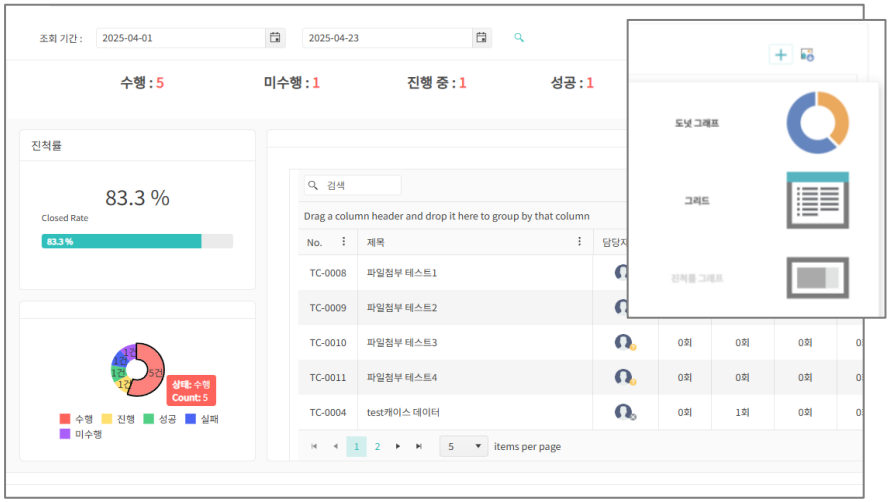
결함 일별 트렌드



테스트 일별 트렌드



기간별 수행 실적 보고서 조회/출력



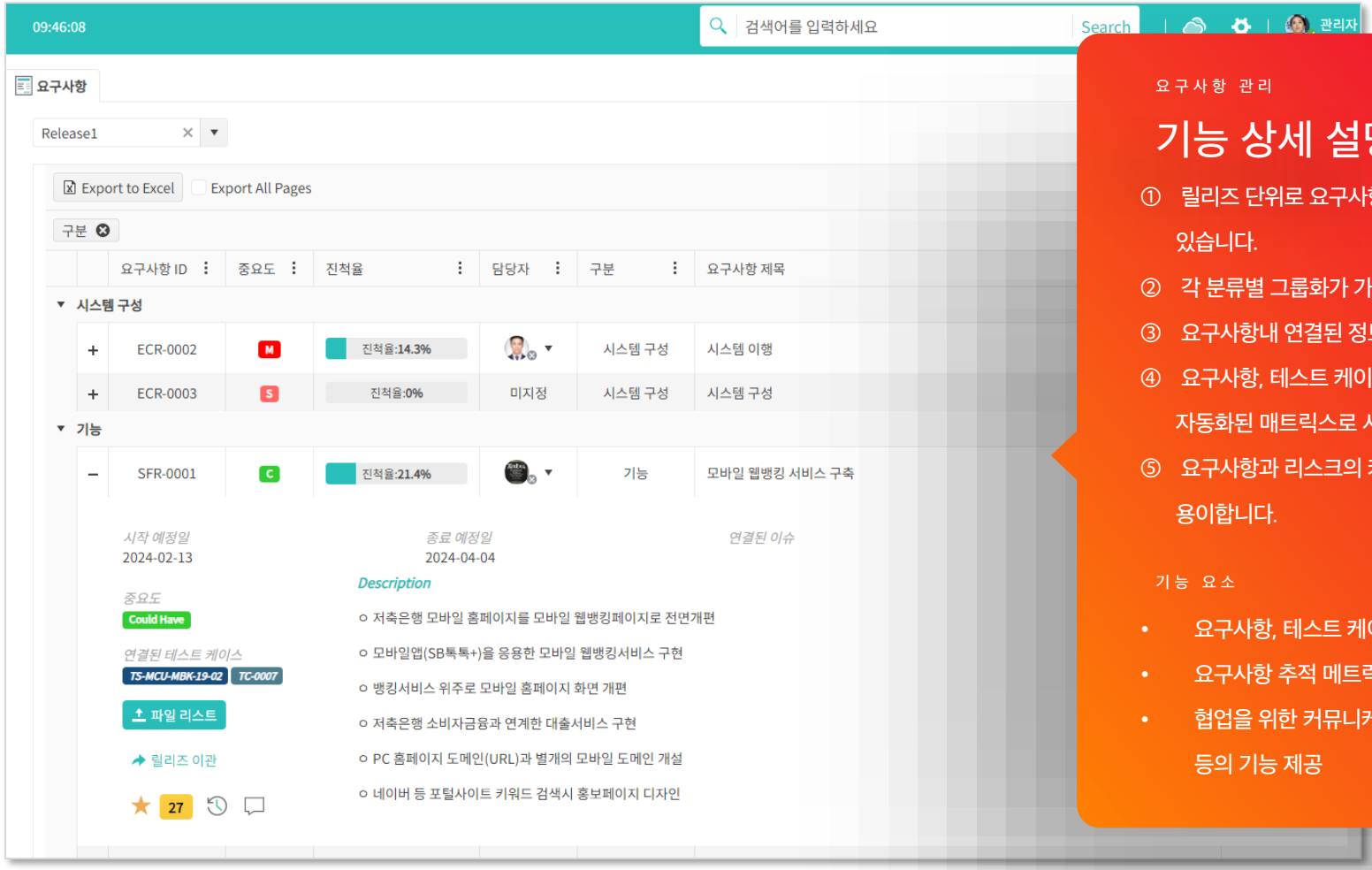
실적 보고서 그래프 추가/숨기기 제공, drag & drop으로 필요에 맞게 수정

요구 사항 및 위험 관리:

요구사항 기반 테스트	• 소프트웨어 기능을 분해하고 이해관계자의 기대를 측정 가능한 요구사항으로 캡처 • 요구사항을 테스트 케이스로 연결하여 견고한 테스트 기반 마련
요구사항 및 리스크 필터링	• 직관적인 필터를 통해 테스트 케이스, 실행, 결과, 이슈를 조직화하고 우선순위 설정 • 테스트 진행 상태 모니터링 및 작업량 예측
요구사항 추적 매트릭스	• 요구사항, 테스트 케이스, 테스트 결과 간의 관계를 자동화된 매트릭스로 시각화 • 요구사항과 리스크의 커버리지 확인 및 진단 용이
SR 프로세스의 지속적인 최적화	• 주기적으로 SR 처리 속도, 해결 시간, 발생 빈도 등의 데이터를 분석하여 병목 현상을 파악하고, 프로세스를 최적화
커버리지 보고서	• 프로젝트의 테스트 진행 상황과 결과를 한눈에 파악할 수 있는 대시보드 제공

DEFECTMATE AI | 2025.ALL RIGHTS

일정, 릴리즈별 요구사항관리가 가능합니다.



요구사항 관리

기능 상세 설명

① 릴리즈 단위로 요구사항이 관리/이관 관리할 수 있습니다.

② 각 분류별 그룹화가 가능하여 분석에 용이합니다.

③ 요구사항내 연결된 정보들을 한눈에 확인 가능합니다.

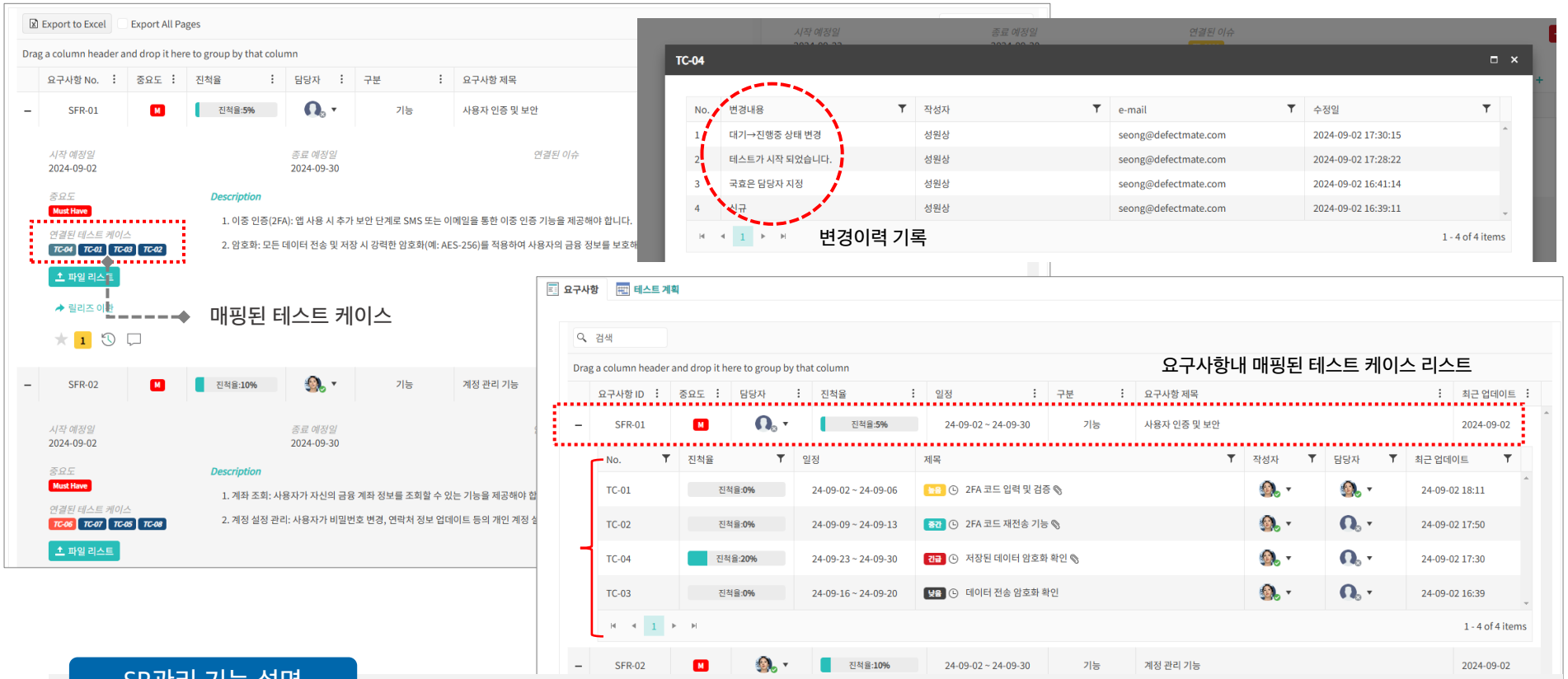
④ 요구사항, 테스트 케이스, 테스트 결과 간의 관계를 자동화된 매트릭스로 시각화 합니다.

⑤ 요구사항과 리스크의 커버리지 확인 및 진단 용이합니다.

기능 요소

- 요구사항, 테스트 케이스, 테스트 진척을 확인
- 요구사항 추적 메트릭, 변경이력 관리
- 협업을 위한 커뮤니케이션과 첨부파일, 엑셀 저장 등의 기능 제공

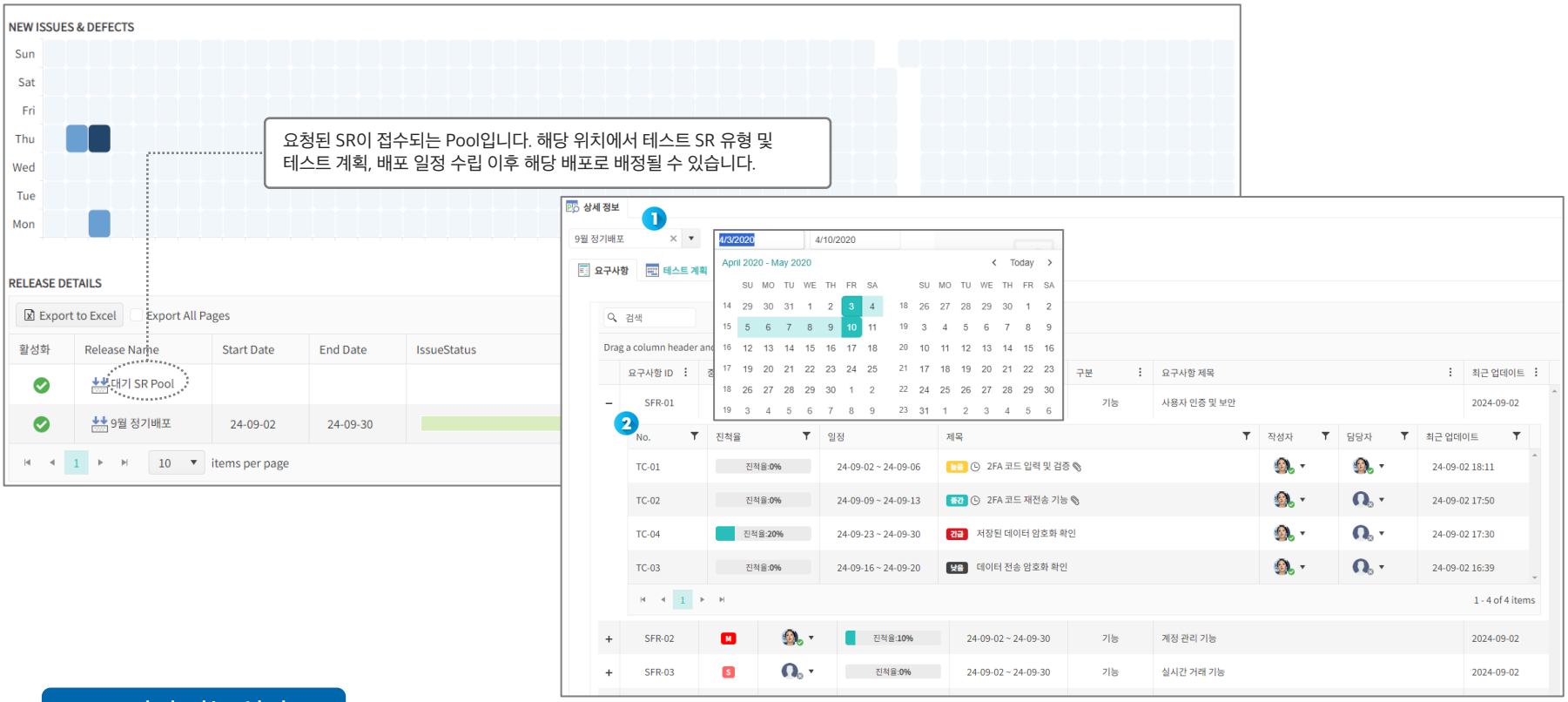
업무개발 요청 기본 정보 정의 및 SR요청에 따른 PM 및 개발자 IT요건정의 생성이 가능합니다.



SR관리 기능 설명

① SM 프로세스와 SI 프로세스를 구분하여 SR 요구사항을 생성할 수 있습니다.
② 필요에 따라, IT 요건정의를 추가/수정할 수 있으며 해당 내용에 대한 변경이력까지도 기록 됩니다.
③ SI 프로세스의 경우, 해당 요구사항에 맞는 테스트 케이스를 매핑할 수 있고, SM 프로세스는 업무 또는 메뉴로 구성된 테스트 케이스에서 매칭하여 설정 가능합니다.

SR의 유형(버그 수정, 기능 개선, 신규 기능 추가 등)에 따라 분류할 수 있으며, 각 SR에 대한 처리를 빠르게 결정할 수 있습니다.
배포(릴리즈)일정에 손쉽게 적용할 수 있으며, 우선순위를 설정하여 처리기준을 확인할 수 있습니다.



SR관리 기능 설명

① 배포대상 및 기간별 조회에 따라 SR 상태 확인 기능

② SR에 할당 된 테스트 케이스를 확인 가능하며, 추가/삭제가 가능합니다.

1개의 테스트 케이스에 차수(여러 담당자)를 두어 테스트 수행이 가능하도록 기능이 확장됩니다.

테스트 케이스 관리

- 클라우드 레파지토리를 통해 테스트케이스의 버전관리 및 배포 할 수 있습니다.
- 케이스별 상세 내역과 진행, 요구사항, 결함과의 연결을 매우 쉽게 사용하실 수 있습니다.

고급 기능

- 연결된 결함상태, 과제 진척현황, 우선순위에 따른 상세 내역 확인
- 테스트 수행결과, 리포트를 관리할 수 있습니다.

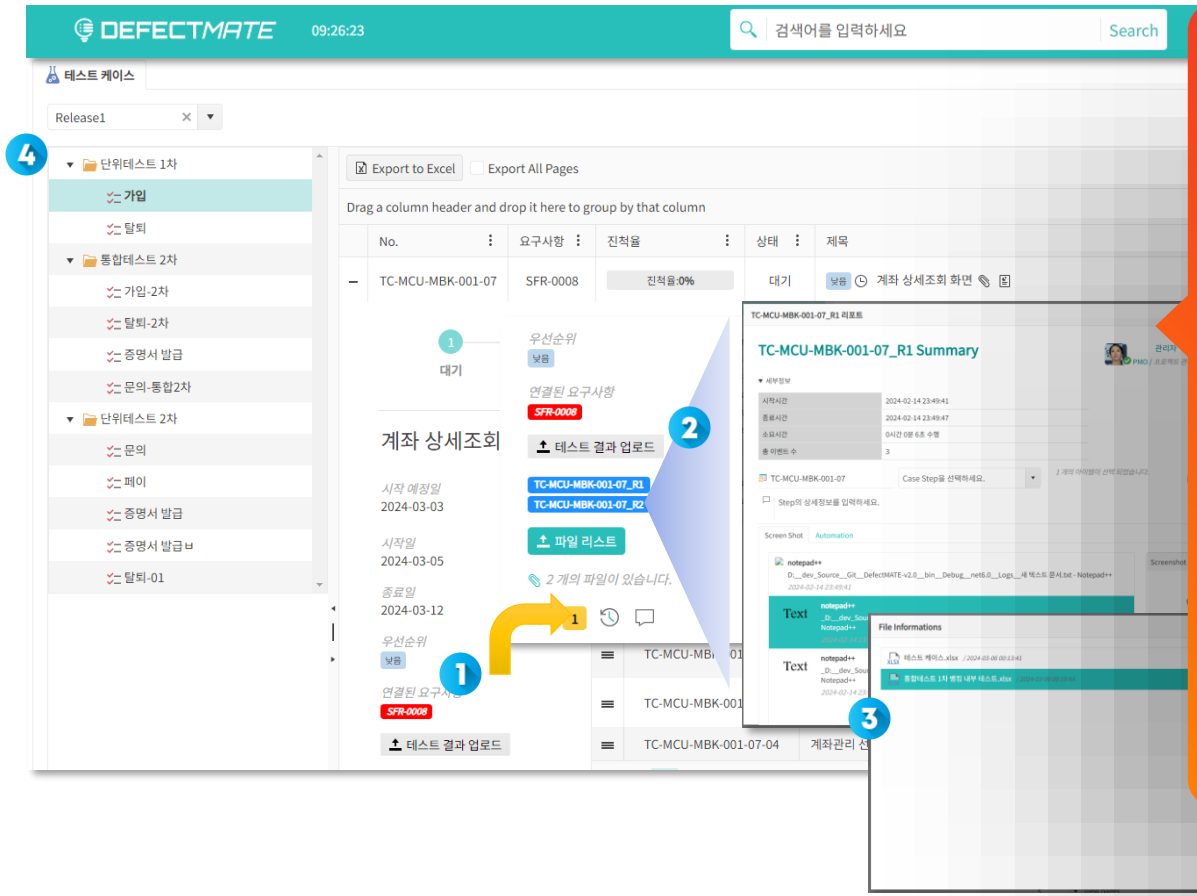
협업 및 추적

- 수행된 리포트를 즉시 확인이 가능하며 소요시간, 수행방법 모두 리포트를 통해 확인할 수 있습니다.
- 테스트 과정, 결과에 대한 모든 커뮤니케이션 기록하고 공유합니다.

가져오기 및 내보내기

- 테스트 차수 그룹별로 분리관리 할 수 있습니다.
- 테스트 계획 및 결과 데이터를 엑셀등의 파일에서 가져오거나 내보낼 수 있습니다.

케이스별 상세 내역과 진행, 요구사항, 결함과의 연결을 매우 쉽게 사용하실 수 있습니다.



테스트 케이스 관리

기능 상세 설명

① 테스트 수행결과, 리포트를 관리할 수 있습니다.

② 수행된 리포트를 즉시 확인이 가능하며 소요시간, 수행방법 모두 리포트를 통해 확인할 수 있습니다.

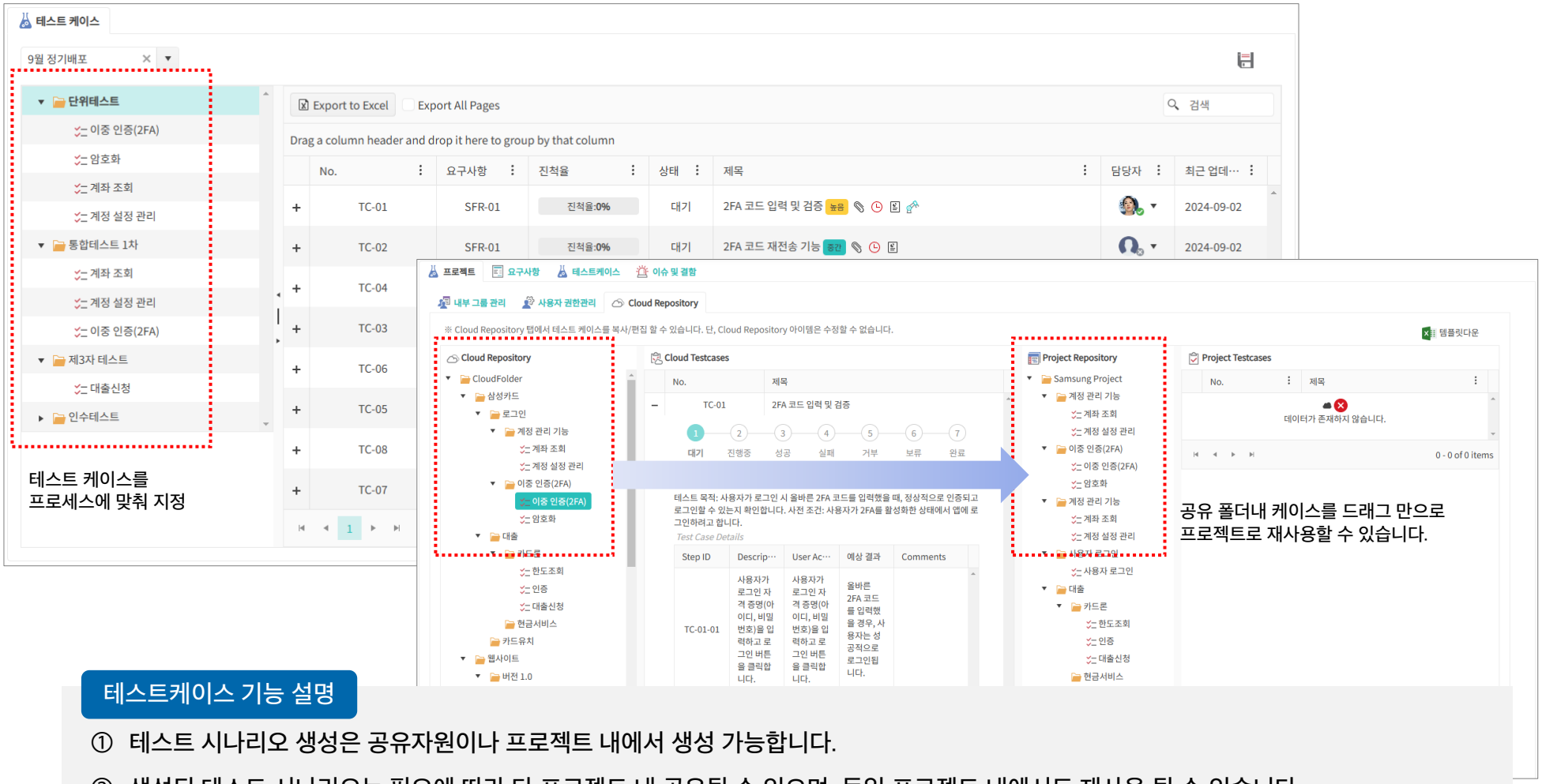
③ 테스트 수행에 필요한 첨부파일들을 관리할 수 있습니다.

④ 테스트 차수 그룹별로 분리관리 할 수 있습니다.

기능 요소

- 테스트 일정 관리
- 연결된 결함 생성 및 조회 가능
- 테스트 진척 및 결과 기록
- 테스트 결과 상세
- 테스트 커뮤니케이션 기록

테스트 시나리오 생성 및 목록관리 및 단위테스트, 통합테스트, 제3자 테스트 등 테스트 케이스 관리가 편리합니다.



테스트케이스 기능 설명

① 테스트 시나리오 생성은 공유자원이나 프로젝트 내에서 생성 가능합니다.

② 생성된 테스트 시나리오는 필요에 따라 타 프로젝트 내 공유될 수 있으며, 동일 프로젝트 내에서도 재사용 될 수 있습니다.

③ 테스트 프로세스에 따라 단위/통합/제3자/UAT등 테스트 케이스를 프로세스에 맞춰 지정 가능합니다.

테스트 수행 후 결함 및 결과 등록 관리 및 회귀 테스트 수행이 가능하도록 테스트 수행관련 정보를 관리할 수 있습니다.

+	TC-03	SFR-01	진척율:0%	대기	데이터 전송 암호화 확인	2024-09-02
−	TC-06	SFR-02	진척율:40%	실패	여러 계좌 간 잔액 일관성 확인 높음	2024-09-05

1 대기 — 2 진행중 — 3 실패 — 4 거부 — 5 성공 — 6

여러 계좌 간 잔액 일관성 확인

연결된 결함

시작 예정일
2024-09-09

종료 예정일
2024-09-13

연결된 이슈
FD-0002 CD-0001

시작일
2024-09-05

종료일

우선순위
높음

연결된 요구사항
SFR-02

자동화 도구

테스트 결과 업로드

파일 리스트

0

Test Case Details

	Step ID	Description	User Action/Data Entered	예상 결과
≡	TC-06-01	사용자가 로그인 후 "계좌 조회" 옵션을 선택합니다.	사용자가 로그인 후 "계좌 조회" 옵션을 선택합니다.	정상
≡	TC-06-02	여러 계좌를 각각 선택하여 잔액을 확인합니다.	여러 계좌를 각각 선택하여 잔액을 확인합니다.	잔액확인됨
≡	TC-06-03	모든 계좌의 잔액이 정확하게 표시되는지 확인합니다.	모든 계좌의 잔액이 정확하게 표시되는지 확인합니다.	각 계좌의 잔액이 일 확하게 표시되어야
≡	TC-06-04	사용자가 화면을 새로고침하거나 다른 메뉴로 이동한 후 다시 계좌 잔액을 조회합니다.	사용자가 화면을 새로고침하거나 다른 메뉴로 이동한 후 다시 계좌 잔액을 조회합니다.	잔액 조회 시 새로고 뉴 이동 후에도 잔액 게 유지되어야 합니

1 5 items per page

Description

- 테스트 목적: 사용자가 여러 계좌를 보유한 경우, 각각의 계좌 잔액이 일관되게 표시되는지 확인합니다.
- 사전 조건: 사용자가 동일한 은행 내에 여러 계좌를 보유하고 있으며, 앱에 로그인한 상태입니다.

댓글형태의 커뮤니케이션 가능

- 사전 조건: 사용자가 동일한 은행 내에 여러 계좌를 보유하고 있으며, 앱에 로그인한 상태입니다.

Comments

댓글수정 성원상 (seong@defectmate.com) 님 께서 2024-09-05 13:45 등록

잔액이 정상적으로 표시되지 않고 있습니다. 결함 생성 후 진행 요청 드립니다.

댓글수정 성원상 (seong@defectmate.com) 님 께서 2024-11-06 10:49 등록

```
@code:
//Please don't change class name 'Main'

abstract class Calculator{
    int left, right;
    public void setOprands(int left, int right){
        this.left = left;
        this.right = right;
    }
    int _sum() {
        return this.left + this.right;
    }
    public abstract void sum();
    public abstract void avg();
    public void run(){
        sum();
        avg();
    }
}
class CalculatorDecoPlus extends Calculator {
    public void sum(){
        System.out.println("+ sum :"+_sum());
    }
    public void avg(){
        System.out.println("+ avg :"+(this.left+this.right)/2);
    }
}
class CalculatorDecoMinus extends Calculator {
```

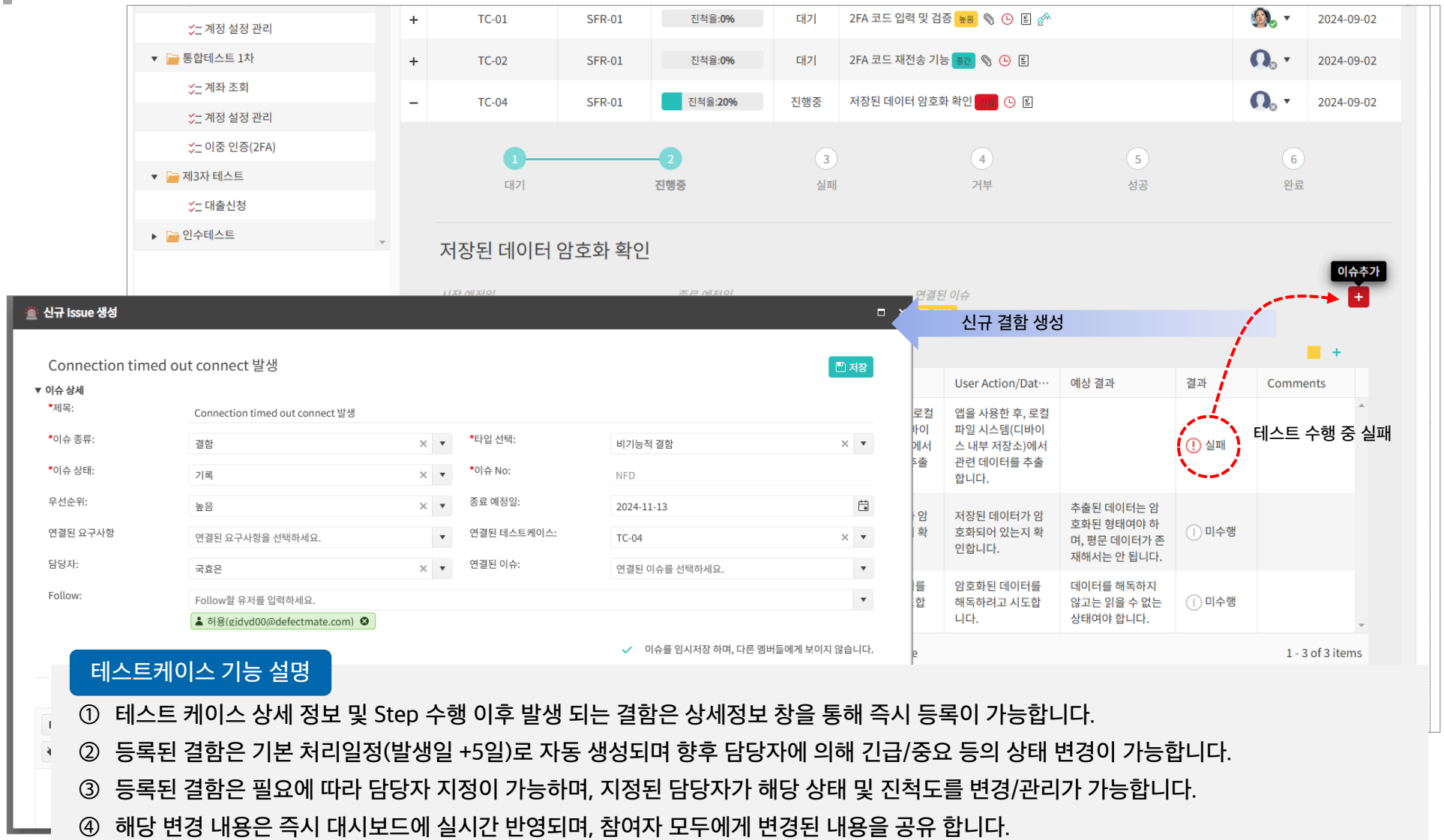
테스트케이스 기능 설명

① 테스트 수행(수동/자동) 후 해당 테스트 케이스에 연결되어 결함 및 결과등록이 가능합니다.

② 해당 테스트 케이스는 필요에 따라 여러 차수(회기) 수행이 가능하며 차수별 수행 정보가 기록됩니다.

③ 테스트 케이스 마다 댓글 형태의 커뮤니케이션을 지원하여, 협업 도구로서의 기능 또한 제공 됩니다.

테스트 수행 후 결함 및 결과 등록 관리 및 회귀 테스트 수행이 가능하도록 테스트 수행관련 정보를 관리할 수 있습니다.



테스트케이스 기능 설명

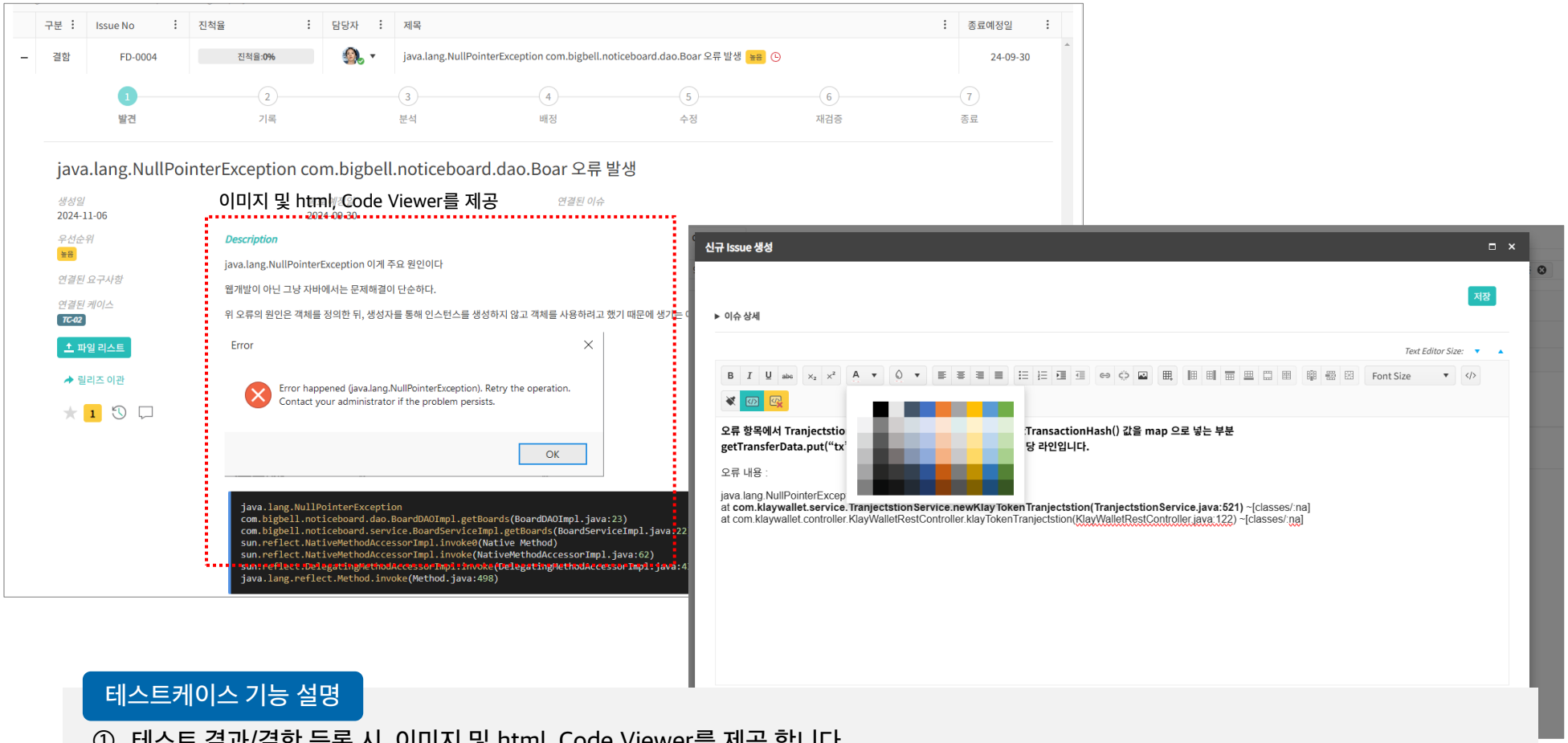
① 테스트 케이스 상세 정보 및 Step 수행 이후 발생 되는 결함은 상세정보 창을 통해 즉시 등록이 가능합니다.

② 등록된 결함은 기본 처리일정(발생일 +5일)로 자동 생성되며 향후 담당자에 의해 긴급/중요 등의 상태 변경이 가능합니다.

③ 등록된 결함은 필요에 따라 담당자 지정이 가능하며, 지정된 담당자가 해당 상태 및 진척도를 변경/관리가 가능합니다.

④ 해당 변경 내용은 즉시 대시보드에 실시간 반영되며, 참여자 모두에게 변경된 내용을 공유 합니다.

테스트 결과 등록 시, 캡처 이미지 등록/첨부파일 등록이 가능하고, 특수 문자입력과 메모(댓글) 기능이 제공 됩니다.



테스트케이스 기능 설명

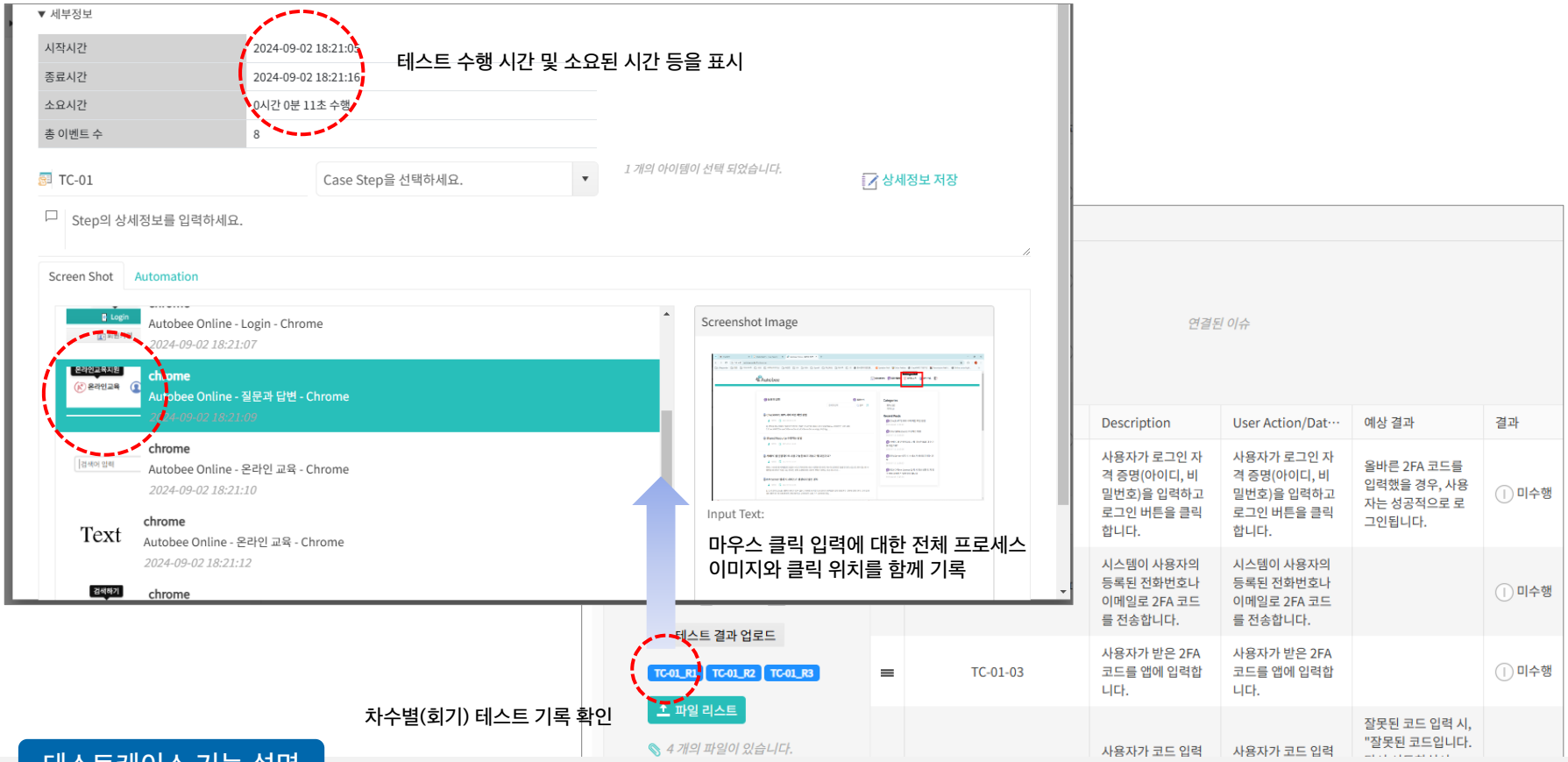
① 테스트 결과/결함 등록 시, 이미지 및 html, Code Viewer를 제공 합니다.

② 각종 특수문자 및 개발 코드를 등록할 수 있어 개발자 간의 커뮤니케이션이 원활히 이루어 집니다.

③ 기본적으로 html 에디터를 제공하여 강조, 이미지, 밑줄등 rich text editor 제공되고 있으며, 첨부파일 및 이미지 등을 함께 등록 할 수 있습니다.

④ 테스트 결과 등록은 상세 Step에서 즉시 해당 결과를 등록 할 수 있으며, 동일 상세페이지에서 확인 가능합니다.

개발자 테스트 도구를 통해 상세 테스트 수행 정보를 저장하고 수행결과까지 관리 할 수 있습니다.



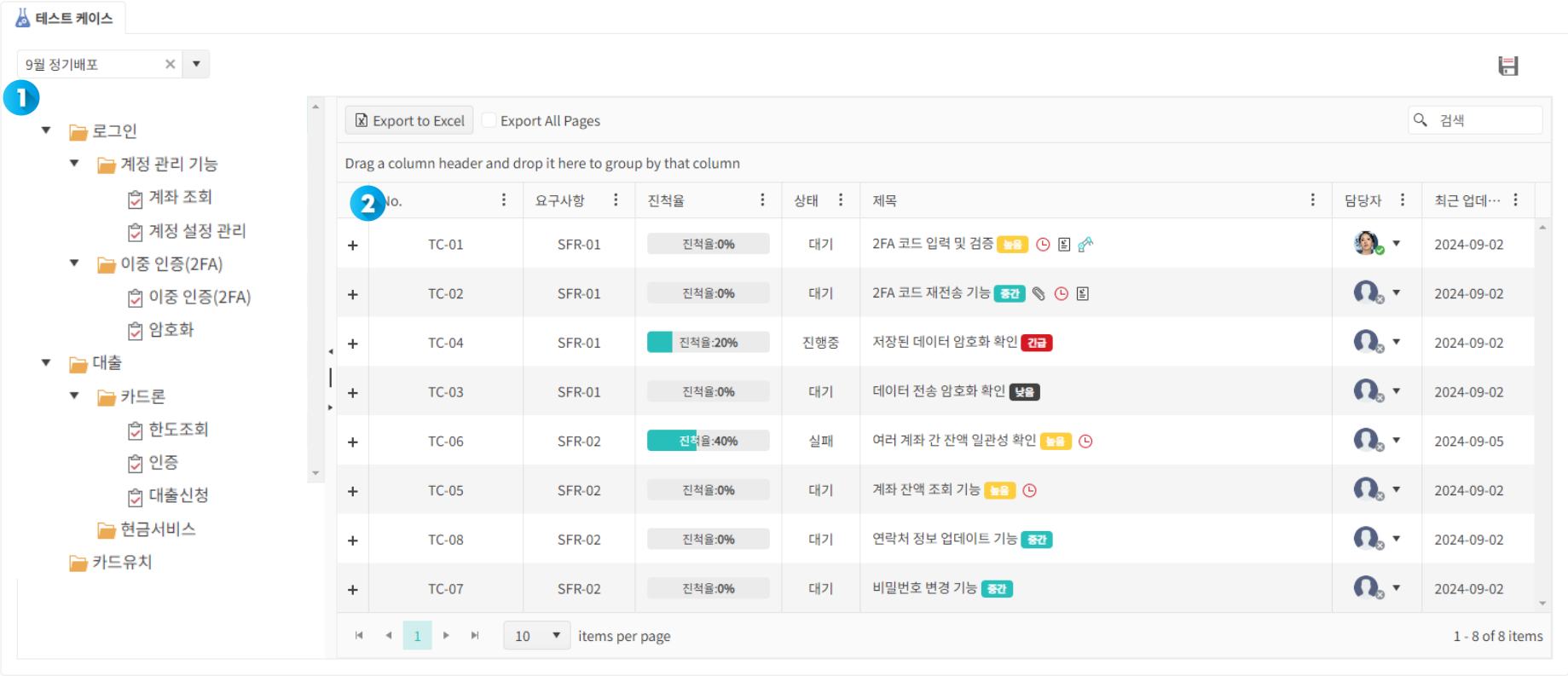
테스트케이스 기능 설명

① 개발자 테스트 도구를 통해 개발 테스트 결과를 사용자 Action에 따라 기록할 수 있습니다.

② 마우스 클릭/키보드 입력에 대해 모두 기록하며, 전체 프로세스 이미지와 클릭한 위치를 함께 기록합니다.

③ 해당 결과를 통해 테스트 수행 시간 및 소요된 시간, 테스트 모든 수행 기록을 이미지로 기록하여 차수별(회기) 테스트 기록을 함께 확인 가능합니다.

과제 / 테스트 케이스 관리



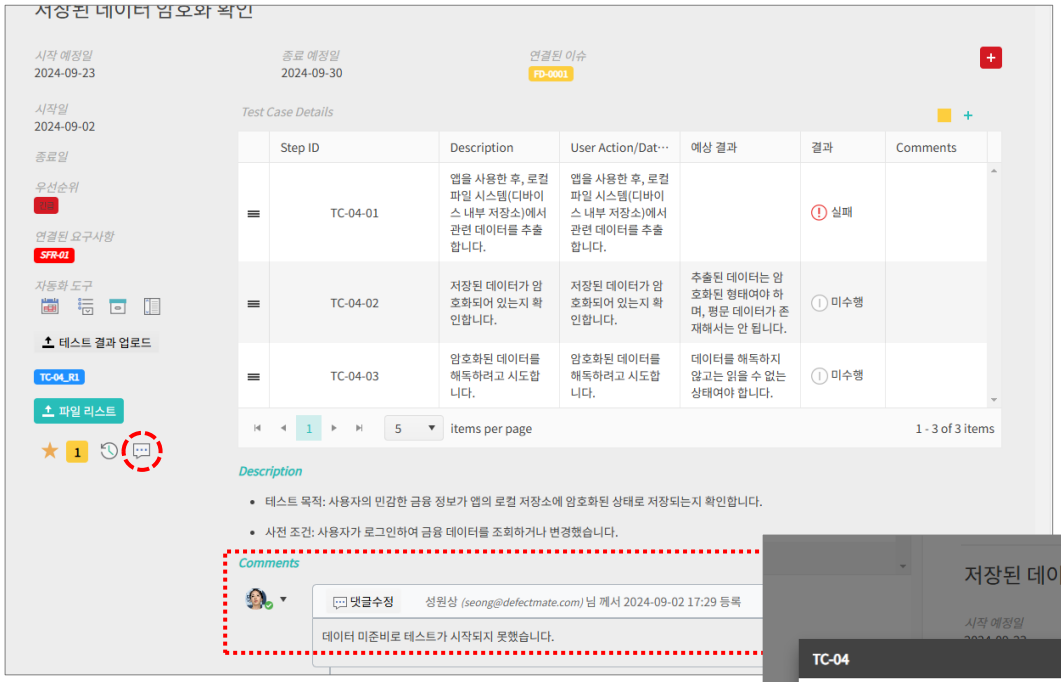
테스트케이스 기능 설명

① 업무 기준으로 케이스를 관리하며 담당자 지정(업무별, 그룹별, 테스트 케이스별) 기능이 추가 됩니다.

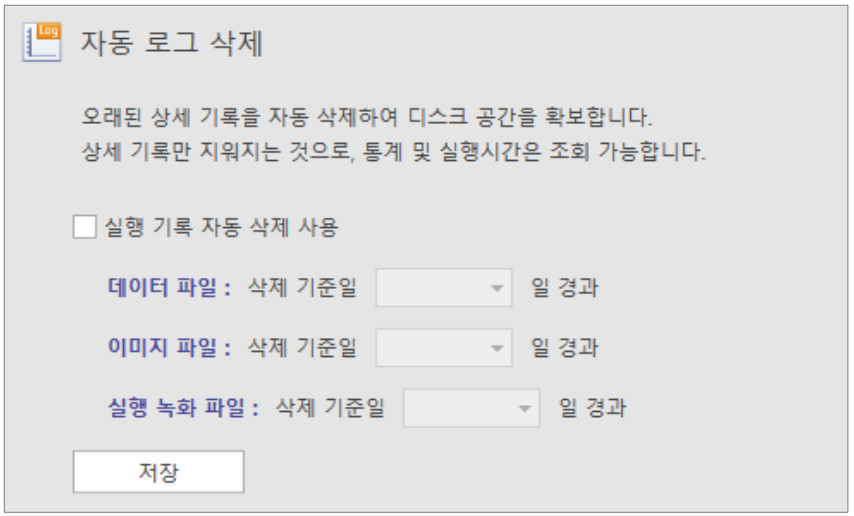
해당 업무는 배포 대상 케이스만 보여집니다.

② 연관 테스트 케이스 기능으로 이전 테스트 케이스가 수행 완료 이후 진행이 가능하도록 구성 예정(상세 프로세스 재 정의 中)

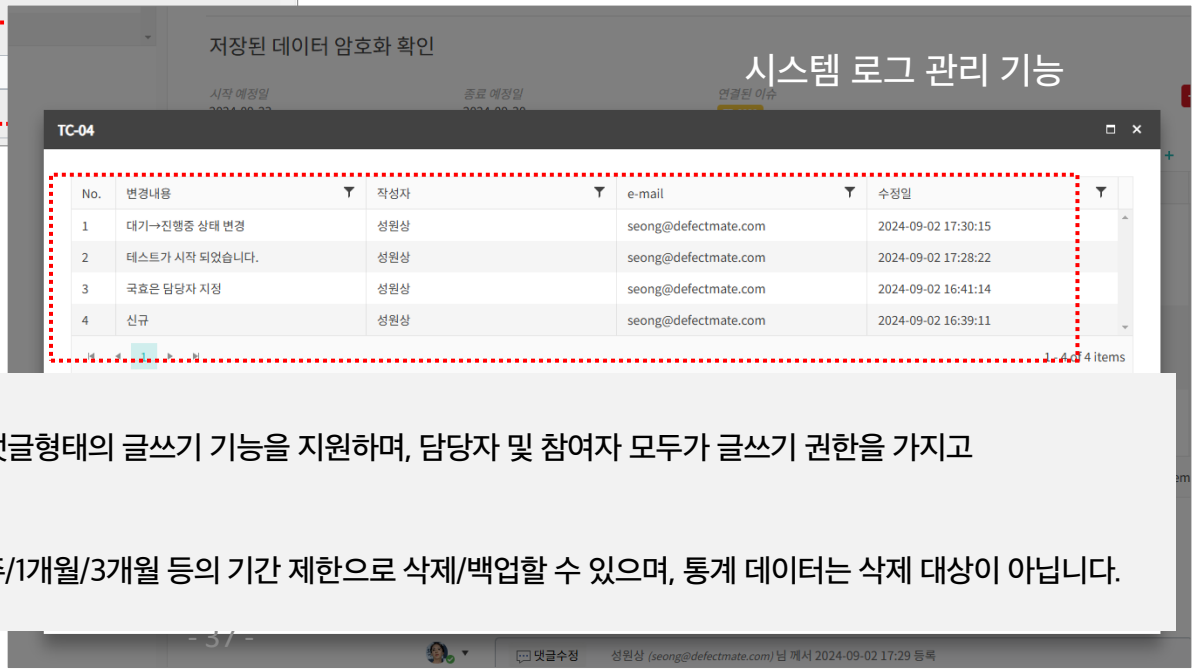
테스트 담당자별 메모 기능



로그 관리 주기설정



시스템 로그 관리 기능

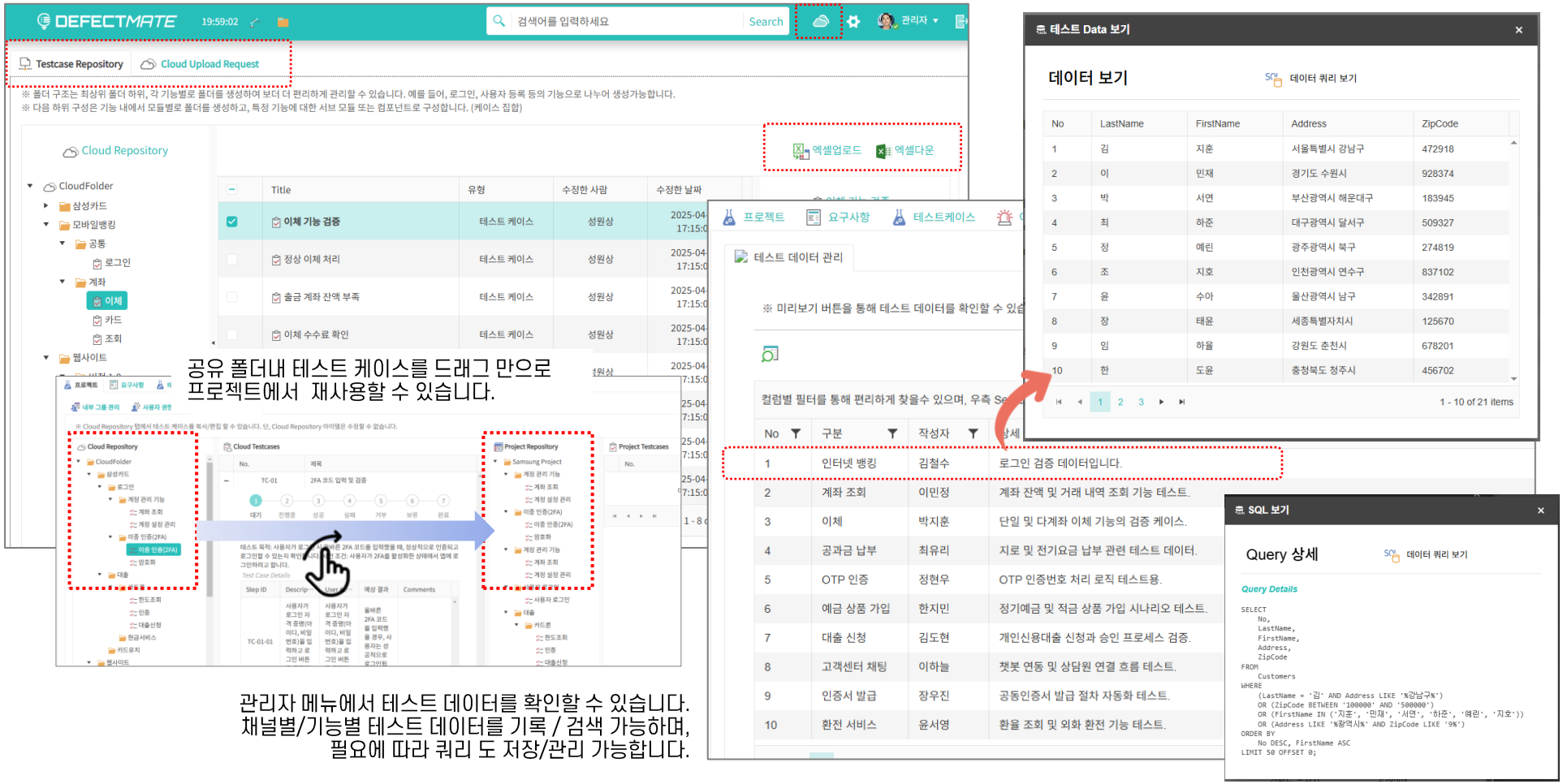


테스트케이스 기능 설명

① SR 요구사항, 테스트 시나리오(케이스), 결함 내 댓글형태의 글쓰기 기능을 지원하며, 담당자 및 참여자 모두가 글쓰기 권한을 가지고 커뮤니케이션 할 수 있습니다.

② 텍스트(접속정보, 변경이력 등), 이미지 로그는 1주/1개월/3개월 등의 기간 제한으로 삭제/백업할 수 있으며, 통계 데이터는 삭제 대상이 아닙니다.

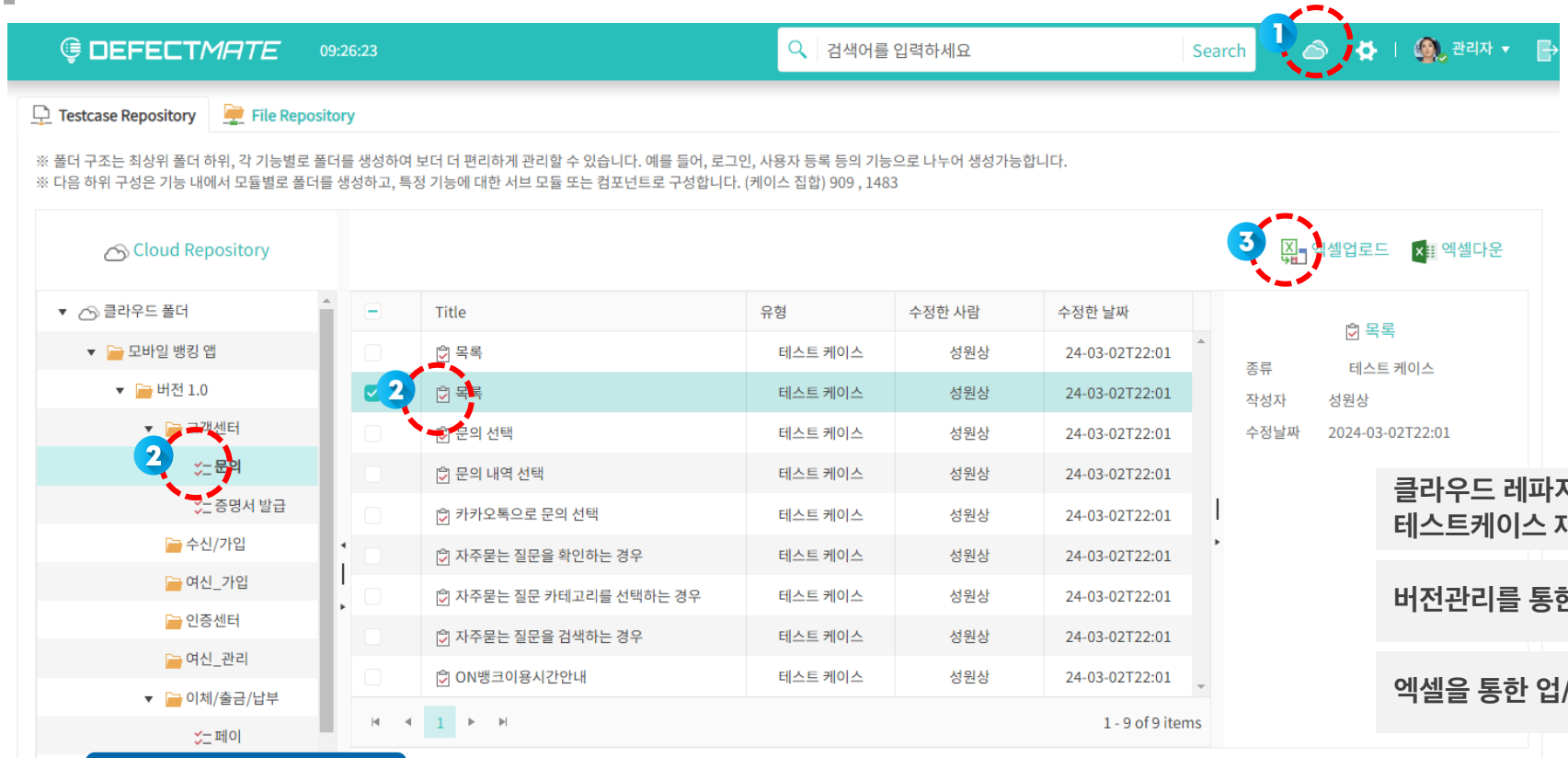
테스트 케이스는 관리자를 통해 클라우드 레파지토리(DB기반)에 저장되고 관리됩니다. 엑셀을 통한 업로드 기능에서 수정/공유 기능을 제공합니다. 테스트 데이터 관리기능을 제공하여 재 테스트 수행에서 정확한 데이터 사용이 가능합니다.



공유 폴더내 테스트 케이스를 드래그 만으로 프로젝트에서 재사용할 수 있습니다.

관리자 메뉴에서 테스트 데이터를 확인할 수 있습니다. 채널별/기능별 테스트 데이터를 기록 / 검색 가능하며, 필요에 따라 쿼리 도 저장/관리 가능합니다.

클라우드 레파지토리를 통해 테스트케이스의 버전관리 및 배포 할 수 있습니다.



클라우드 레파지토리를 통한 테스트케이스 재사용

버전관리를 통한 개선된 케이스

엑셀을 통한 업/다운로드

기능 설명

① 클라우드 레파지토리를 통해 테스트케이스를 관리할 수 있습니다.

② 버전 및 그룹별로 테스트 케이스를 분리/관리가 가능합니다.

③ 테스트케이스를 엑셀을 통해 업로드 / 다운로드가 가능합니다.

** 해당 정보는 보기를 위한 임의 정보 입니다.*

결함 관리:

결함 추적 및 관리

- 결함 발생 시 자동화된 문서화 및 우선순위 설정
- 문제의 원인 분석 및 해결책 기록

프로젝트 협업 도구

- 팀 간 실시간 커뮤니케이션 및 상태 업데이트
- 사용자 지정 알림 및 이슈 할당

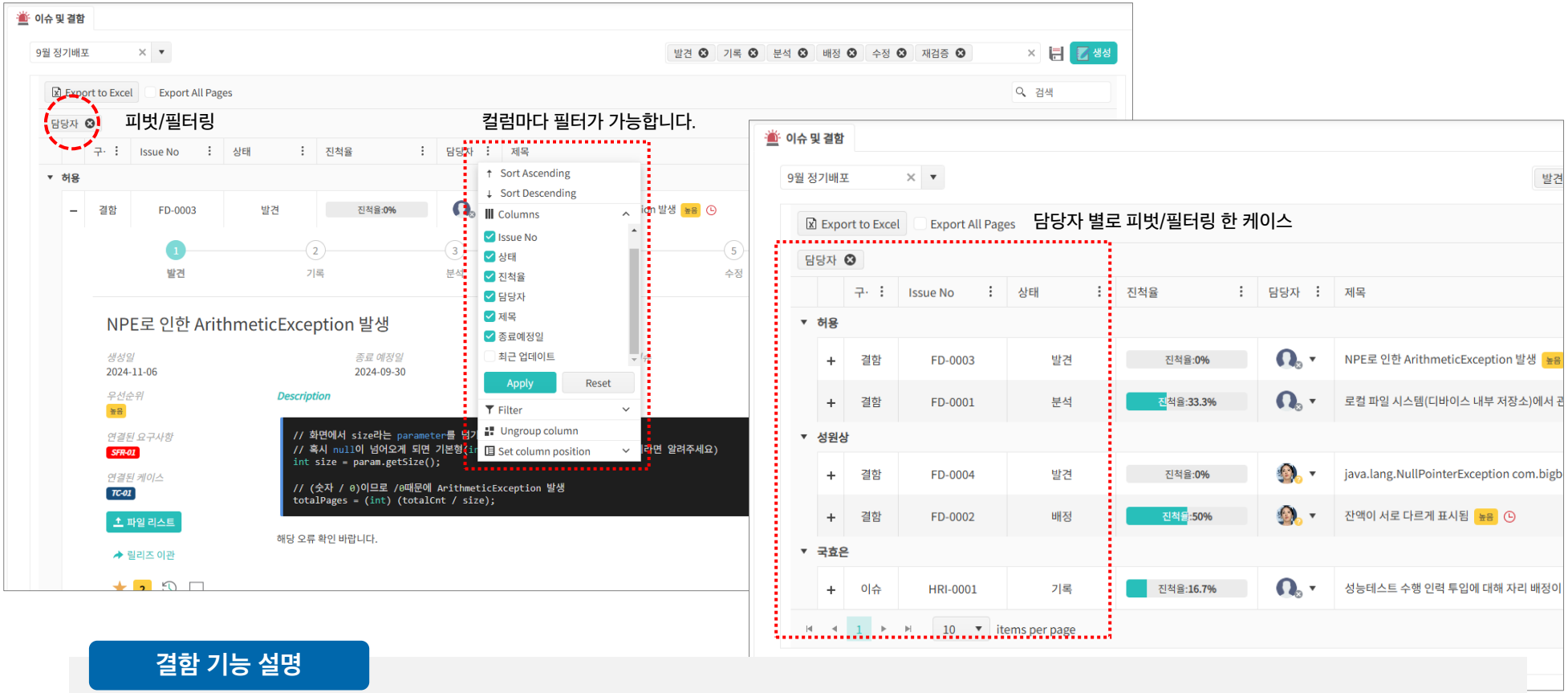
보고 및 통계

- 결함 상태, 발생 빈도, 해결 속도 등 다양한 메트릭 제공
- 시각적 대시보드로 진행 상황 모니터링

자동화 도구와 통합

- Jira, DevOps 등 다양한 개발 도구와의 통합

결함 관련된 정보(SR, 테스트 시나리오, 테스트 케이스,결함내용 등) 및 내역 조회가 가능합니다.



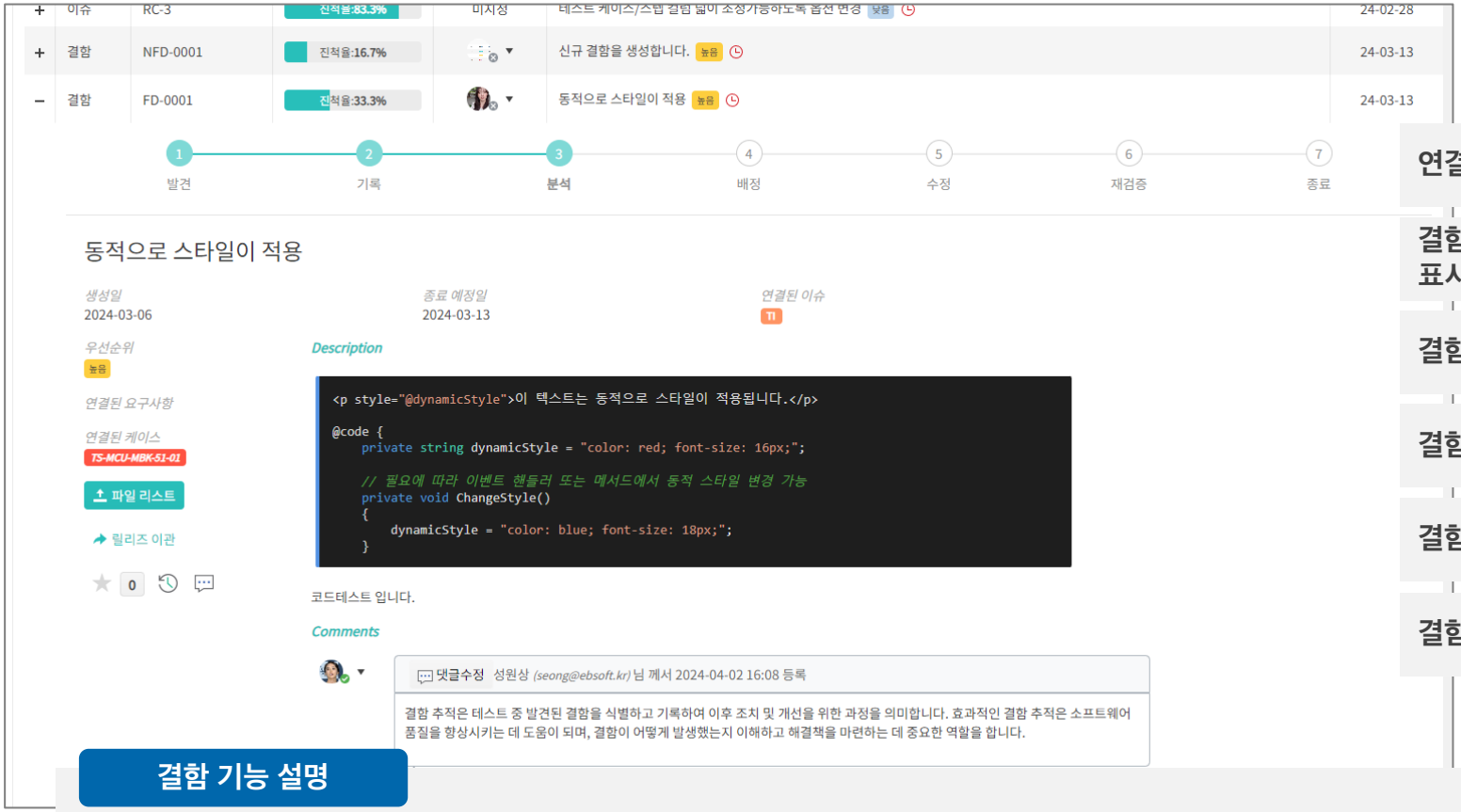
결함 기능 설명

① 등록된 결함을 필터링하여, 중요도/담당자/우선순위/일정지연 등으로 관리 할 수 있습니다.

② 테스트 케이스에 담당자 및 참여자를 두어 등록되는 모든 결함에 대해 알림을 받을 수 있습니다. 담당자의 경우, 해당 결함을 수용/반려 등 관리할 수 있는 권한을 가지고 있습니다.

③ 등록된 결함은 단독적으로 생성 가능하지만, 기본적으로 SR 요구사항, 테스트 시나리오(케이스) 등과 연결되며 검색기능을 제공합니다.

결함의 생성, 수정, 종료 까지 모든 라이프사이클을 관리합니다.



- 연결된 테스트 케이스 정보
- 결함과 연결된 상세 정보들 표시
- 결함 상태 표시
- 결함 상세내용 표시/이미지첨부
- 결함 진척 이력
- 결함 커뮤니케이션 기록

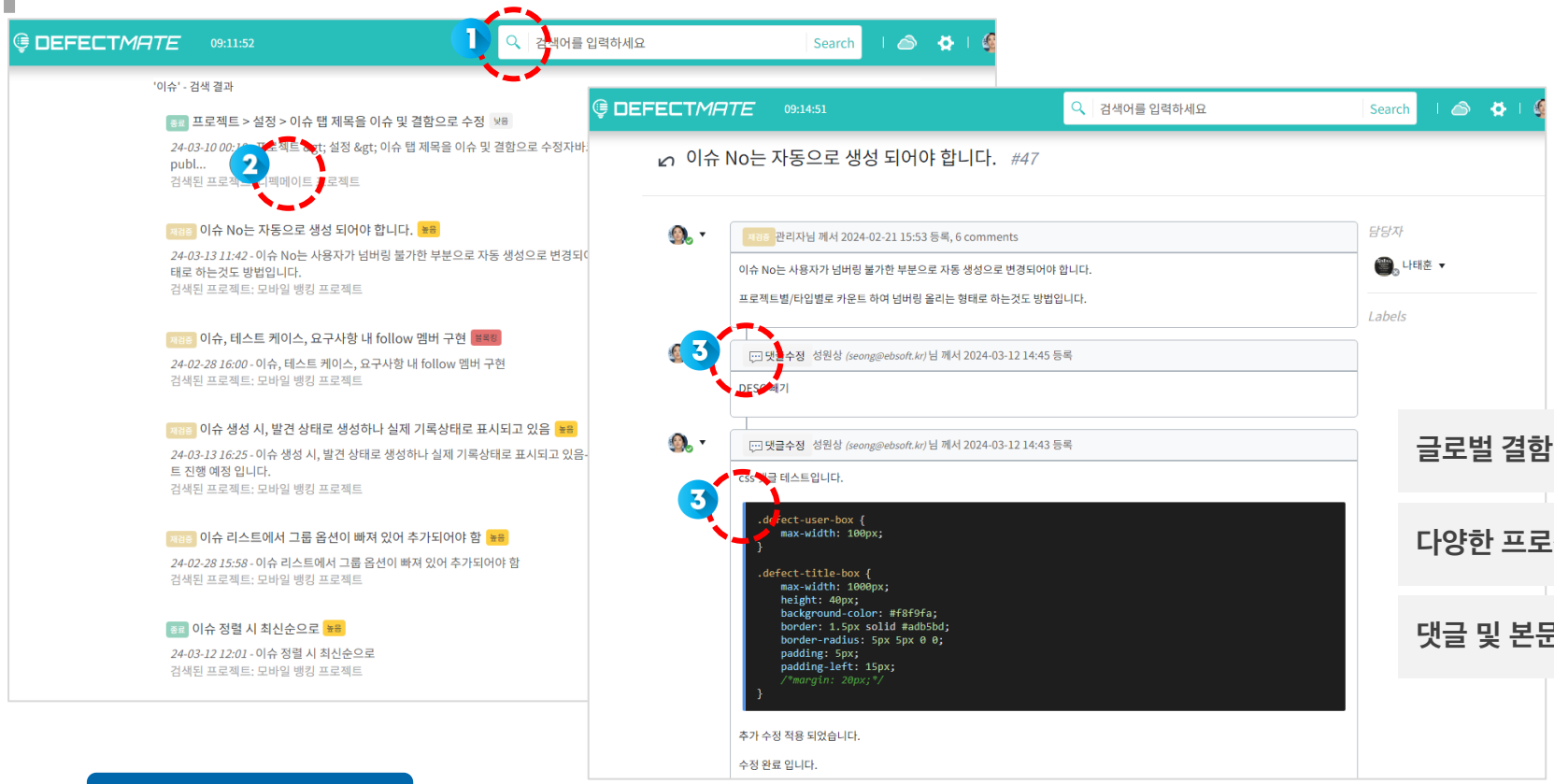
결함 기능 설명

① 아주 간편히 결함을 등록 / 조회할 수 있습니다.

② 등록된 결함의 상태를 확인 할 수 있고, 처리결과를 기록 할 수 있습니다.

③ 코맨트를 이용하여 사용자간 커뮤니티를 더욱 편리하게 하실 수 있습니다.

결함은 글로벌 검색기능이 지원되어 모든 프로젝트의 결함을 검색 가능합니다.



글로벌 결함검색 기능

다양한 프로젝트 지원

댓글 및 본문 내 코드 보기

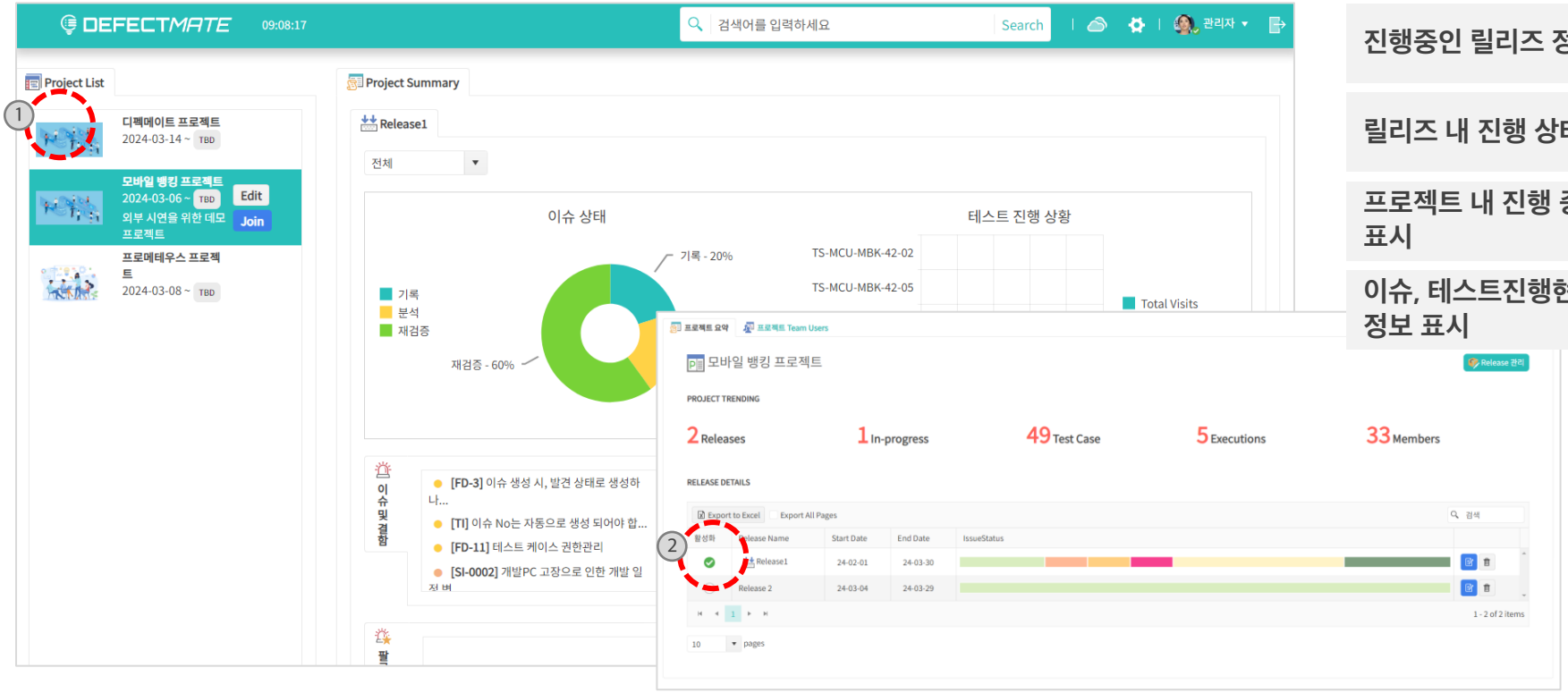
결함 기능 설명

① 프로젝트의 종류에 관계없이 모든 결함을 검색할 수 있습니다.

② 현재 생성된 모든 프로젝트의 결함에서 검색할 수 있습니다.

③ 타임라인 형태의 댓글이 지원되며, 개발자 보기를 위한 코드 보기를 지원하여 편리하게 내용확인이 가능합니다.

다양한 프로젝트, 릴리즈 단위로 관리가 가능합니다.



진행중인 릴리즈 정보 표시

릴리즈 내 진행 상태 정보 표시

프로젝트 내 진행 중인 정보 표시

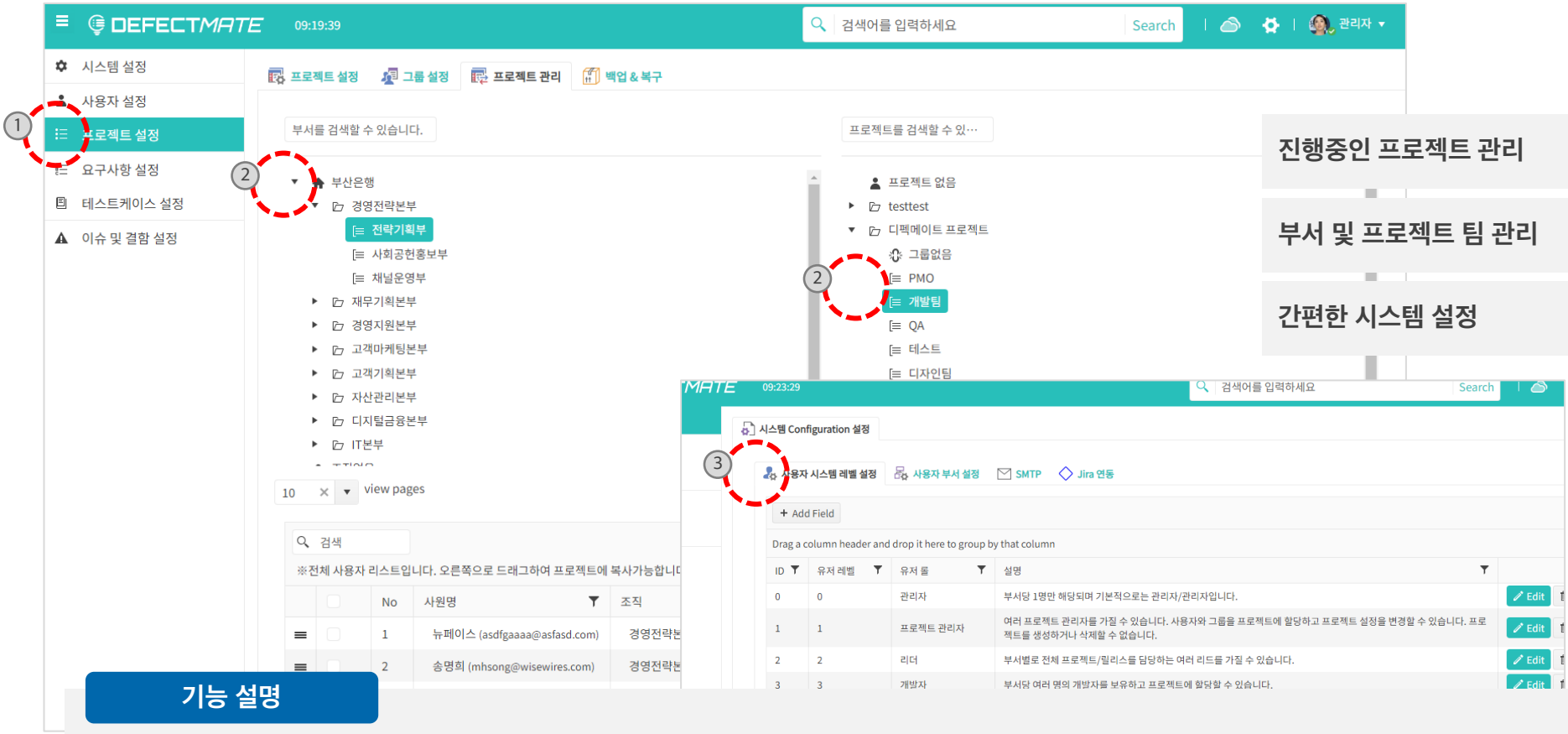
이슈, 테스트진행현황 개인화 정보 표시

기능 설명

① 다수의 프로젝트를 관리할 수 있습니다.

② 프로젝트 내, 릴리즈/스프린트별 공정관리가 가능합니다.

다양한 시스템 설정과 진행중인 프로젝트를 관리하고 팀을 관리할 수 있습니다.



진행중인 프로젝트 관리

부서 및 프로젝트 팀 관리

간편한 시스템 설정

기능 설명

① 관리자 페이지 내에서 프로젝트 및 다양한 시스템 설정이 가능합니다.

② 전사 부석관리에서 프로젝트 내 팀 관리까지 모두 할 수 있습니다.

③ 다양한 시스템 설정과 연동(SMTP, 지라연동)을 손쉽게 설정 가능합니다.

** 해당 정보는 보기를 위한 임의 정보 입니다.*

자동화와 연계:

시간 절약:
- 반복적이고 시간이 많이 소요되는 테스트 작업 자동화
- 테스트 수행 시간 단축

정확성 향상:
- 인간의 실수를 최소화하여 테스트 정확도 향상
- 동일한 테스트 케이스 반복 수행으로 신뢰성 확보

범위 확장:
- 광범위한 테스트 케이스의 자동 실행 가능
- 다양한 환경 및 조건에서 테스트 수행 가능

빠른 피드백:
- 테스트 결과의 신속한 피드백 제공
- 문제 발생 시 즉각적인 수정 조치 가능

비용 효율성:
- 초기 자동화 도구 도입 비용 후 지속적인 테스트 비용 절감
- 인력 자원 최적화

지속적 통합 및 배포(CI/CD):
- 지속적 통합 및 배포 파이프라인에 자동화 테스트 통합
- 코드 변경 시마다 자동 테스트로 품질 보증

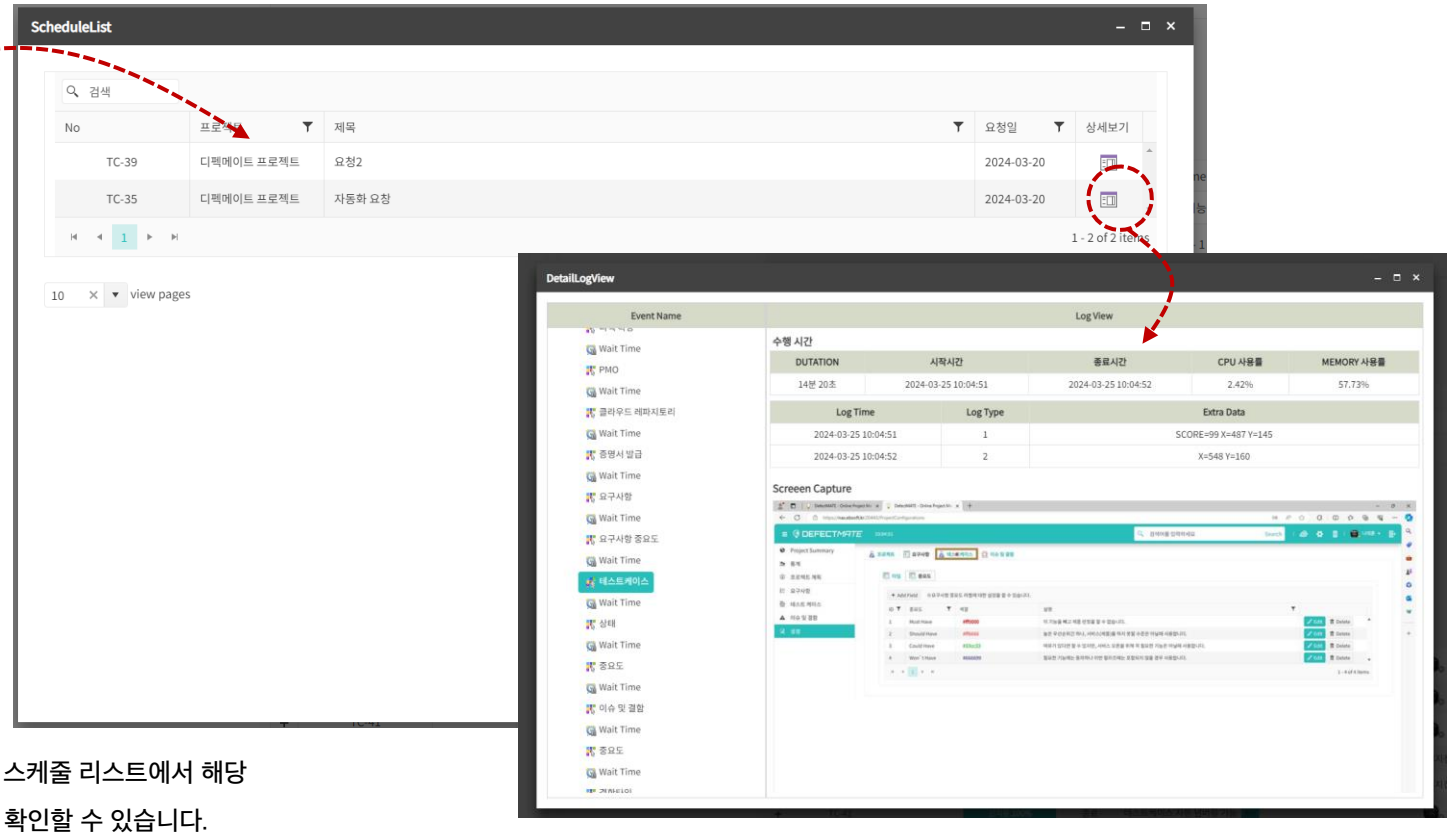
복잡한 시나리오 처리:
- 복잡한 테스트 시나리오도 자동화 가능
- 인간이 놓칠 수 있는 엣지 케이스[1] 검증

리포팅 및 분석:
- 자동화된 리포팅 도구를 통해 상세한 테스트 결과 제공
- 데이터 기반의 테스트 분석 가능

*1) **엣지 케이스:** 일반적이지 않은 극단적인 상황에서 발생할 수 있는 문제*

테스트 자동화 도구 내 등록된 업무의 Schedule 정보를 확인할 수 있습니다. 이 정보를 통해 수행된 테스트 케이스 스케줄 정보와 더불어 수행 결과, 상세 로그를 조회할 수 있습니다.

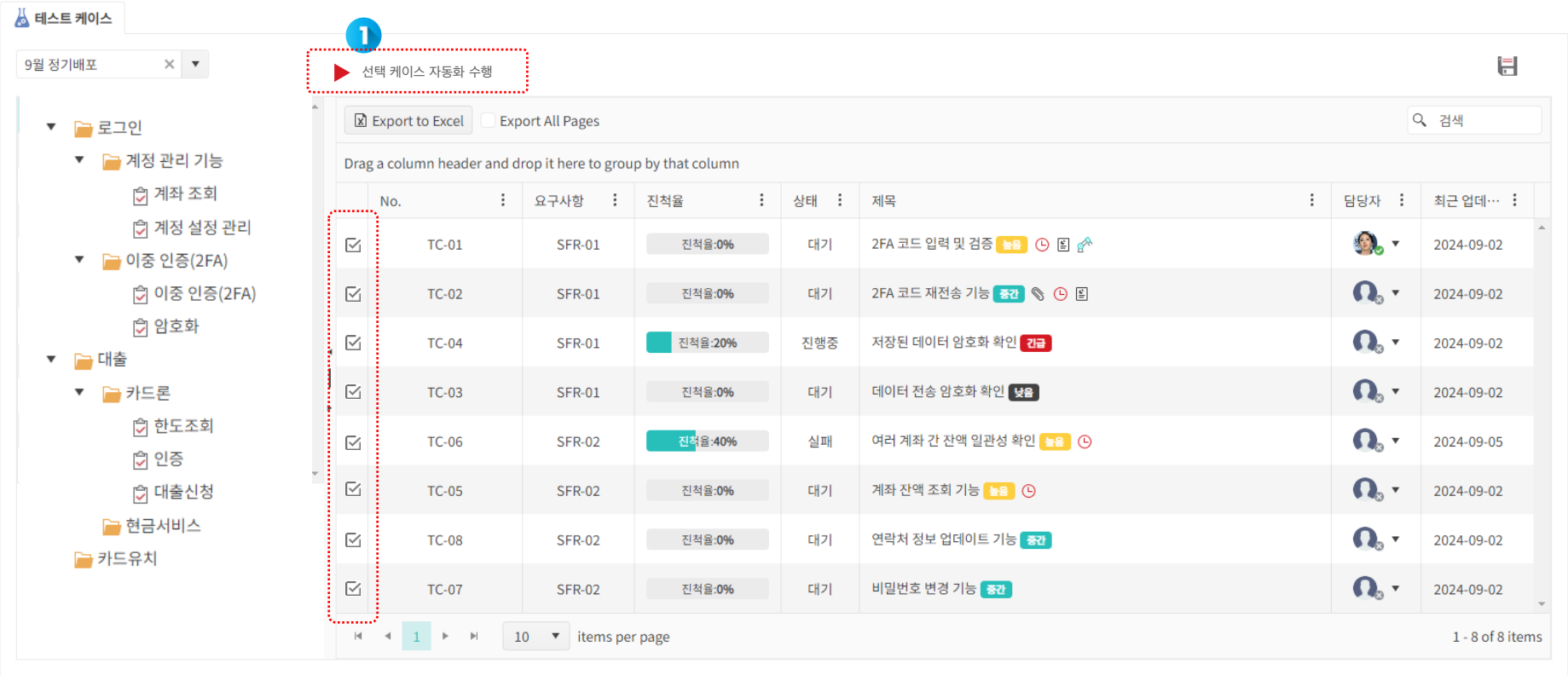
※ 스케줄 정보를 선택하면 지난 1주일 치의 스케줄 상세정보를 확인 가능합니다.



※ 수행이력을 선택하면, 우측 스케줄 리스트에서 해당 봇에 등록된 Schedule 정보를 확인할 수 있습니다.

※ Log View 버튼을 클릭하면 테스트 수행결과 상세 정보가 열립니다.

최종 단계의 테스트 수행은 개발자 테스트 수행이 완료된 단계에서 필요 테스트 케이스를 선택하여 테스트 자동화 수행이 가능합니다. 해당 기능으로 QA팀은 테스트 수행결과에 대해 빠른 대응 및 커버리지를 높일 수 있습니다.



테스트케이스 기능 설명

① 개발자(단위) 테스트 수행 완료된 케이스에 대해 QA 팀 등에서 배포 이후 최종 검증(통합테스트)을 자동화 하여 수행 할 수 있습니다.

시스템 구성도

Mobile 및 Desktop에 모두 최적화된 UI을 제공하고, 심플한 시스템 구성으로 높은 시스템 안정성을 보장 합니다.
Web Server / RDB 구성으로 매우 간편한 시스템 관리를 하게 해줍니다.

Windows + IIS + DBMS

Microsoft IIS7

Microsoft SQL Server

ORACLE

MariaDB

Linux + Apache + DBMS

APACHE

Microsoft SQL Server

ORACLE

MariaDB

최소 사양

DefectMATE

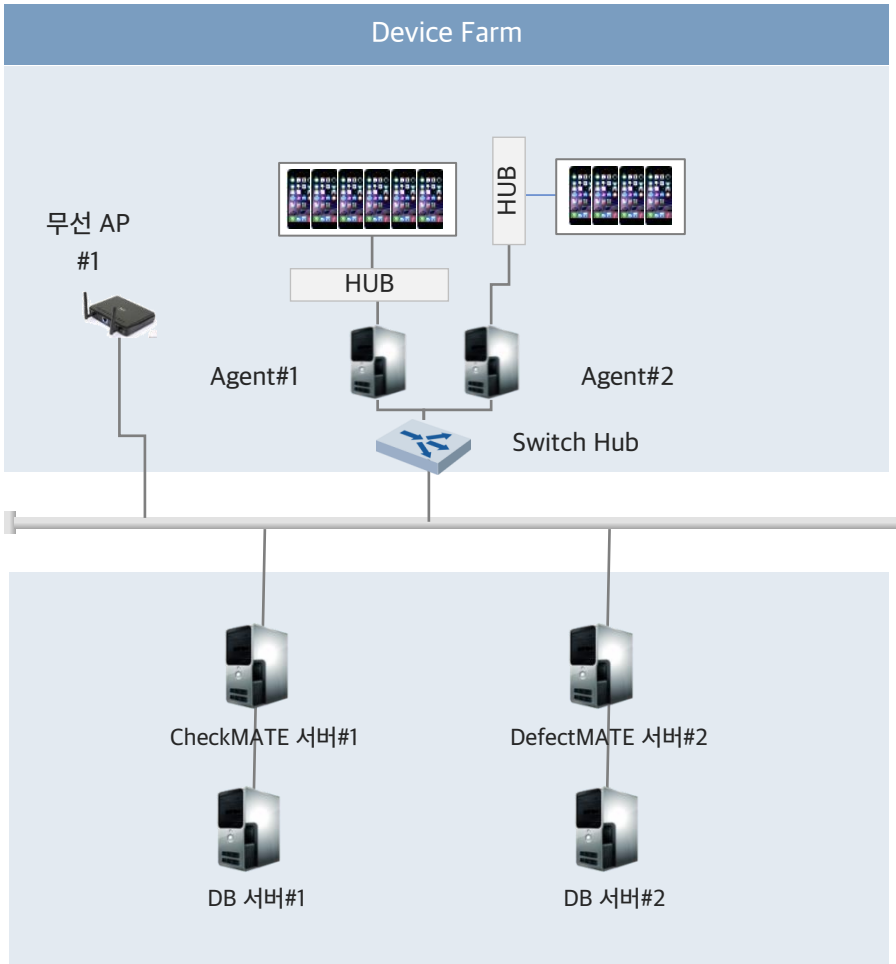
Hardware

- CPU : Intel i5 2.6GHz 이상 (4Core 이상)
- Memory : 16GB 이상
- SSD : 500Gb 이상

Software

- OS : Windows Server 2019 이상, Linux
- Edge, Chrome, Safari (html5 지원)
- IIS + MariaDB(MySQL)
- Apache + MariaDB(MySQL)

H/W 구성도

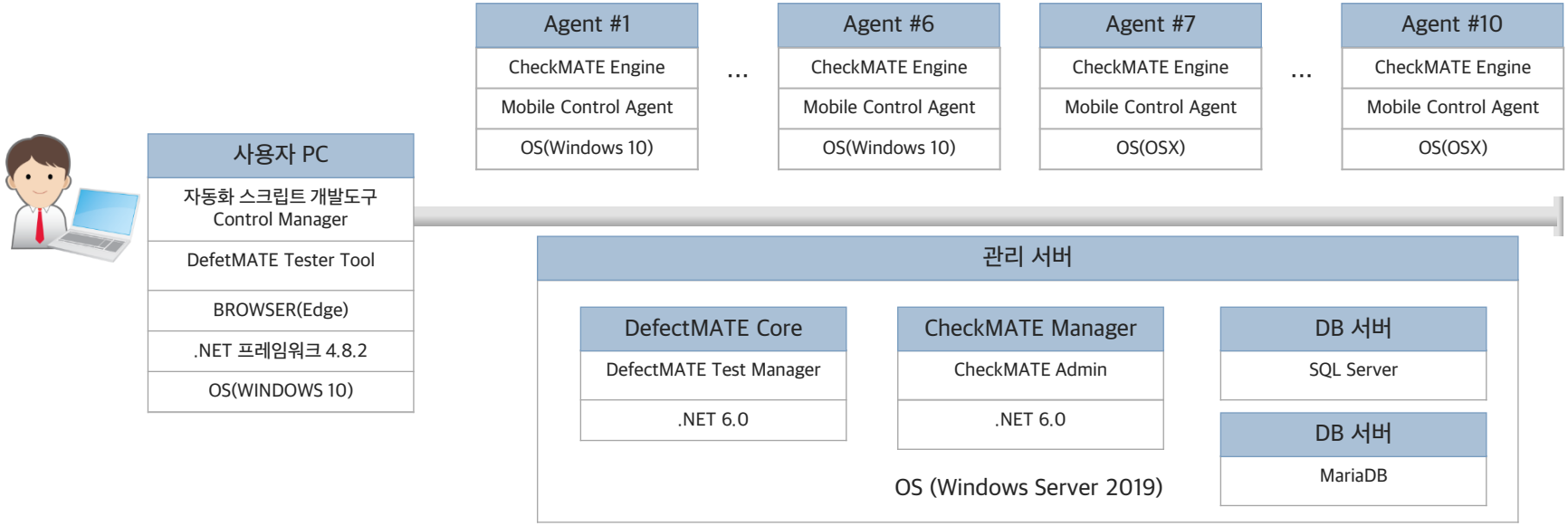


구분	HW	내역	수량
Device Farm	Agent PC	디바이스 Agent	2
	Mac Mini	iOS 디바이스 Agent	1
	USB Hub	USB Hub 7 Port 유전원	2
	진열대	디바이스 거치용 선반	(옵션)
기타 장비	디바이스	신규 디바이스, 중고 디바이스	10
	무선 네트워크	디바이스 무선 접속용	무선AP 1
	네트워크	스위치 Hub	1
	KVM	VPS 화면 전환 Display	(옵션)

※ 최대 10ea 기기(Android 6, iOS 4) 동시운영 H/W spec 입니다.

구분	내역	수량	권장 사양
서버	CheckMATE Server DefectMATE Server	2	• CPU : Intel i5 2.8GHz 이상 (4Core 이상) • Memory : 16GB 이상 • SSD : 500Gb 이상
	CheckMATE DB서버 (SQL-Server) DefectMATE DB서버 (MariaDB)	2	• CPU : Intel i5 2.8GHz 이상 (8Core 이상) • Memory : 32GB 이상 • SSD : 500Gb 이상

S/W 구성도 및 목록



솔루션

NO.	구분	SW	수량
1	CheckMATE	Automation Mobile Agent	10
2		Control Manager(자동화 개발도구)	10
3		CheckMATE Admin	1
4		SDK for Android, iOS	-
5	DefectMATE	Developer Test tools	100
6		테스트 공정관리 Core	1

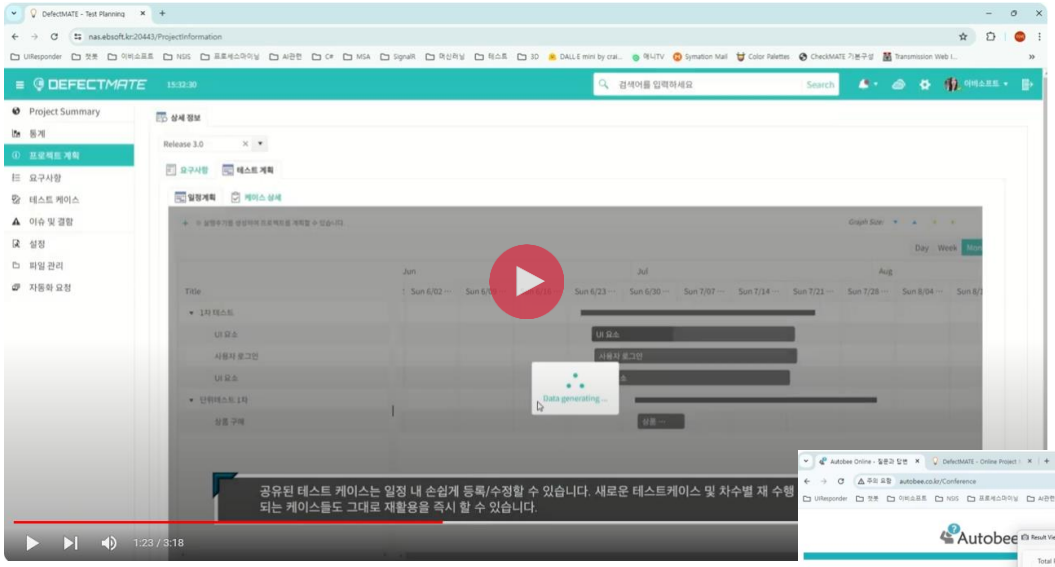
시스템 S/W

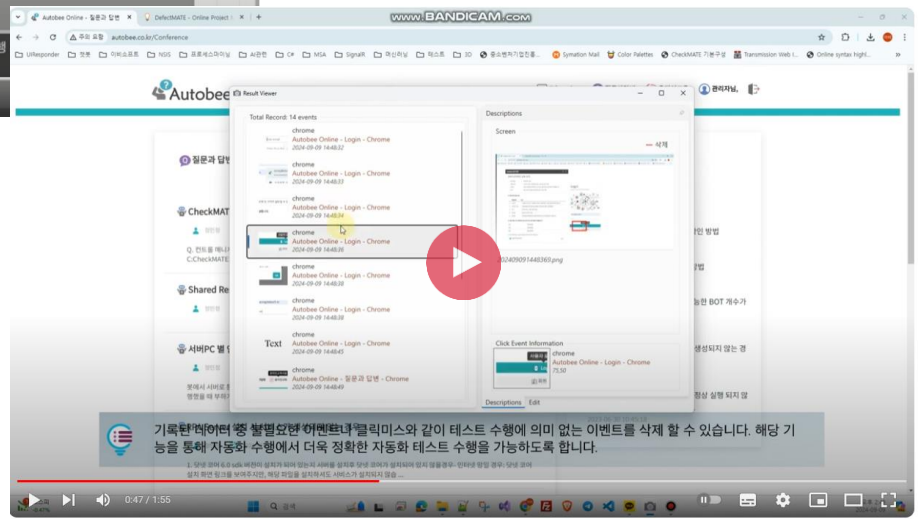
NO.	구분	SW	수량
1	AP Server	IIS, .NET 6.0	2
2	OS	Windows Server 2019	4
3	OS	OSX	1
4	DBMS	SQL Sever	1
5	DBMS	MariaDB	1

	DefectMATE	HPE ALM	마이크로소프트 TFS
서비스 형태	설치	SaaS	SaaS
제조사	㈜ 이비소프트	HPE	마이크로소프트
장점	• 프로젝트 상태, 일정 및 리소스 할당 관리 • 테스트 케이스 수집 및 자동 테스트 수행 • 개인 개발자의 경우 저비용으로 사용하며, 기업의 경우에도 비용이 저렴. • 팀간/모든 개발자와의 의 프로젝트 정보공유	Water fall 방법론에 가깝고, 요구사항, 테스트 케이스를 수집하고 자동으로 테스트 케이스를 수행할 수 있다. 최근에는 DevOps와 Agile을 지향한다.	전체 애플리케이션을 관리할 수 있으며, 모든 팀원이 이를 확인할 수 있다. 소프트웨어 개발 프로세스를 단순화, 단기간에 고품질의 소프트웨어를 시장화 할 수 있다.
단점 / 개선사항	• 테스트 케이스 작성의 편리성 개선 • 테스트 자동화 도구와의 연계/통합 • 해외 사용자에게 맞는 지역화	프로젝트 추적이 다소 복잡. 탐색에 시간이 필요함. 보고서의 어려움. 너무 비싼 가격. IE 브라우저만 지원. ActiveX 설치 필요.	사용자가 편집 내용을 간략히 확인할 수 있는 History view 기능이 미비. 관리가 어려움.

DefectMATE 소개

https://youtu.be/DR0g3TmmWUs?si=RAiKjFNvMI_x_eQz





기능(개발자, 제3자) 테스트 수행 도구에서 자동화 수행 예시 https://youtu.be/7pE42FY9TXk?si=miMoFBYubRDVkKdW

감사합니다

성 원 상 Seong Wonsang
대표 CEO

서울 금천구 가산디지털2로 98, 1-815

Email seong@ebsoft.kr