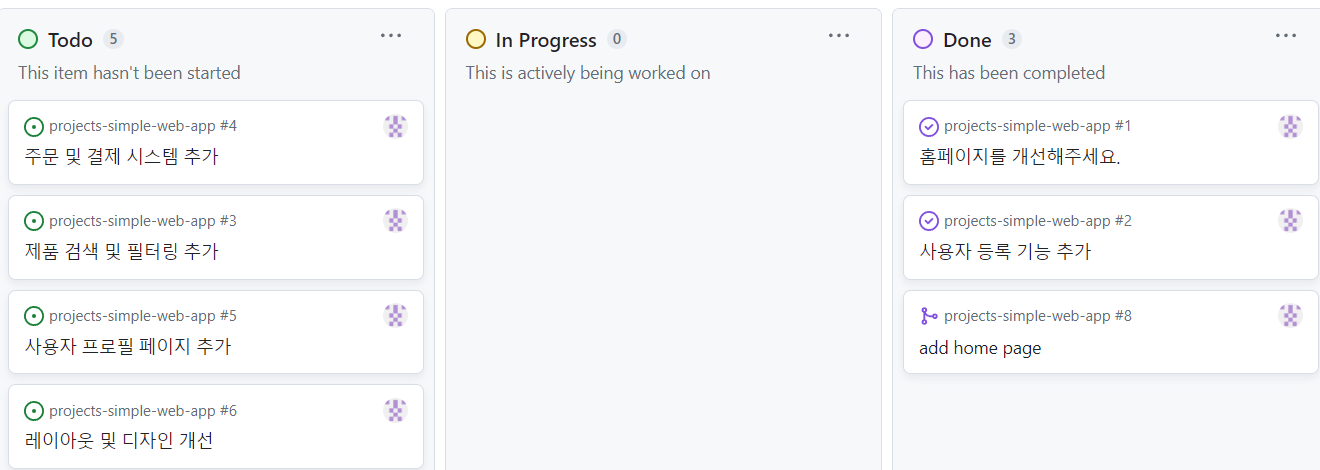
**GitHub 주요 구성 요소**

* **Issue**:
  + 버그, 기능 요청, 개선 사항 등
  + 제목, 설명, 댓글, 레이블 및 담당자 포함
  + Status: Open, Closed
* **Pull Request**:
  + 코드 변경 사항 검토 및 병합 요청
  + 새로운 기능 또는 버그 수정을 원본 코드베이스에 병합 요청
  + 변경 내용 설명, 파일 비교, 코드 리뷰 및 토의 포함
* **GitHub Project**:
  + 프로젝트 관리 도구.
  + Issue와 Pull Request를 조직화하고 추적
  + 프로젝트 진행 상황을 사용자 정의 가능한 작업 보드로 모니터링(예: 진행 예정, 진행 중, 완료)

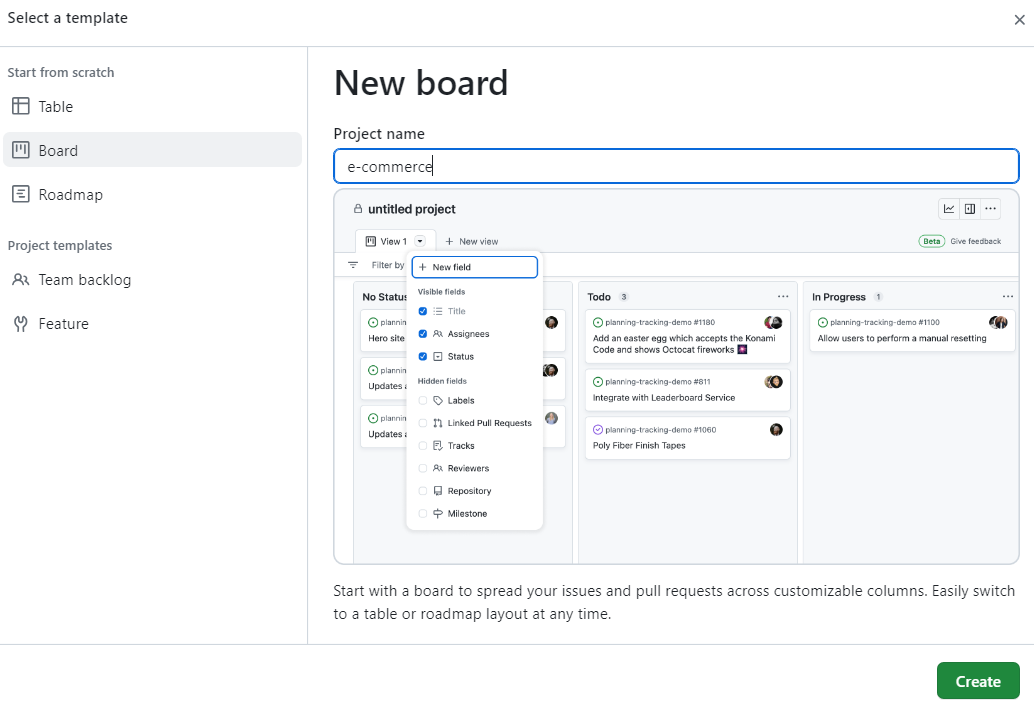


1. 프로젝트 저장소 생성

https://github.com/stelladream/github-projects-demo

1. 쇼핑몰 Project

프로젝트는 GitHub의 issues 및 pull requests과 통합한 적응형 스프레드시트, 작업 보드 및 로드맵입니다



1. 프로젝트 Milestone 추가

Milestone(마일스톤)은 프로젝트 관리에서 중요한 단계 또는 목표를 정의하고 추적하기 위한 도구

Milestone 1: 기본 웹 애플리케이션 개발

Description: 기본 쇼핑몰 웹 애플리케이션을 개발하는 것이 목표입니다. 이 단계에서는 웹 페이지 디자인, 사용자 등록, 상품 표시 및 검색, 장바구니 관리, 주문 및 결제 시스템을 구축합니다.

- 사용자 등록 페이지 및 로그인 기능 개발

- 홈페이지 레이아웃 및 디자인 작업

- 상품 목록 표시 및 필터링 기능 구현

- 장바구니 시스템 개발

- 주문 및 결제 시스템 추가

Milestone 2: 기능 추가 및 성능 향상

Description: 쇼핑몰 애플리케이션에 추가 기능을 구현하고 성능을 향상시키는 것이 목표입니다.

- 상품 리뷰 및 평점 추가

- 상품 카테고리 및 태그 기능 구현

- 사용자 프로필 페이지 추가

- 데이터베이스 쿼리 최적화

- 보안 취약점 해결

Milestone 3: 테스트, 문서화, 배포, 및 추후 개선 계획

Description: 프로젝트를 완료하기 위해 테스트, 문서화, 배포, 그리고 추후 업데이트 및 개선 계획을 설정합니다.

- 유닛 테스트 및 품질 테스트 작성 및 실행

- API 문서 작성

- 다국어 지원 추가

- 프로젝트 배포 및 운영 환경 설정

- 추후 업데이트 및 개선 사항을 문서화하고 계획 수립

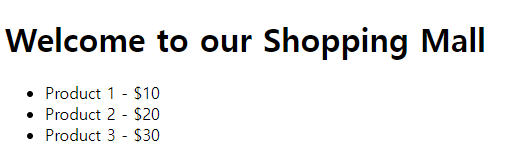
1. 프로젝트 Issues 추가

* **기능 추가(Feature Request)**

1. 이슈 #1: 간단한 홈페이지를 보여주는 초기 프로젝트 생성

설명: Express.js를 사용하여 웹 애플리케이션에서 특정 경로(루트 경로, 즉 "/")로 들어오는 HTTP GET 요청을 처리하는 핸들러 함수를 이용하여 구현하세요.

UI는 아래와 같습니다.



1. 이슈 #2: 사용자 등록 기능 추가

설명: 쇼핑몰 사용자가 새로운 계정을 등록할 수 있도록 회원가입 페이지를 개발합니다.

1. 이슈 #3: 제품 검색 및 필터링 추가

설명: 사용자가 제품을 검색하고 필터링할 수 있는 기능을 추가합니다.

1. 이슈 #4: 주문 및 결제 시스템 추가

설명: 사용자가 주문을 생성하고 결제를 완료할 수 있는 주문 및 결제 시스템을 개발합니다.

1. 이슈 #5: 사용자 프로필 페이지 추가

설명: 사용자가 자신의 프로필을 관리할 수 있는 프로필 페이지를 생성합니다.

* **작업 목록, 문서화**

① 이슈 #6: 레이아웃 및 디자인 개선

## 쇼핑몰 웹 페이지의 디자인 및 레이아웃을 개선하기 위한 작업 항목입니다

- [ ] \*\*레이아웃 리뷰 및 개선\*\*

현재 레이아웃을 검토하고 개선 사항을 식별합니다.

- [ ] \*\*색상 스킴 업데이트\*\*

웹 페이지의 색상 스킴을 업데이트하여 더 현대적인 느낌을 만듭니다.

- [ ] \*\*폰트 선택 및 텍스트 스타일링\*\*

웹 페이지에 사용할 폰트 및 텍스트 스타일링을 선택합니다.

- [ ] \*\*이미지 리소스 개선\*\*

웹 페이지에서 사용할 이미지 리소스를 개선하고 최적화합니다.

② 문서화 (Documentation):

이슈 #7: API 문서 작성

설명: 프로젝트의 API에 대한 문서를 작성하고 업데이트하는 작업을 추적하는 이슈입니다.

* **버그 수정 이슈(buf fix)**

1. 이슈 #8: 장바구니에서 상품이 잘못 표시되는 버그 수정

설명: 장바구니에서 상품이 올바르게 표시되지 않는 문제를 수정합니다.

1. 이슈 #9: 결제 프로세스 중 오류 발생하는 버그 수정

설명: 결제 프로세스 중 오류가 발생하는 문제를 해결합니다.

1. 이슈 #10: 사용자 로그인 문제 수정

설명: 사용자가 로그인할 때 발생하는 문제를 수정하여 로그인 프로세스를 개선해주세요.

Github Project에서 issue 확인하기

5. 쇼핑몰 응용 프로그램 작성 (이슈 1번 수행)

Node.js가 설치되어 있는지 확인 후 존재하지 않을 경우, node.js windows installer 설치한다.

설치 후 node --version으로 정상적인 설치 확인

새로운 feature 추가하기 위해서 feature 브랜치를 생성한다.

* git checkout –b nykim-feature

1. Express.js 설치

(Node.js를 위한 웹 애플리케이션 프레임워크로서 웹 및 API 서버 개발을 용이하게 만드는 도구)

|  |
| --- |
| npm install express |

1. app.js

|  |
| --- |
| const express = require('express');  const app = express();  const http = require('http').Server(app);  const port = process.env.PORT || 3000;  // 루트(/) 경로로 들어오는 HTTP GET 요청을 처리한다. 보라색으로 표현된 코드가 핸들러  app.get('/', **(req, res) => {**  **// 루트 경로에서 쇼핑몰 홈페이지를 직접 렌더링**  **res.send(`**  **<!DOCTYPE html>**  **<html>**  **<head>**  **<title>쇼핑몰 홈</title>**  **</head>**  **<body>**  **<h1>Welcome to our Shopping Mall</h1>**  **<ul>**  **<li>Product 1 - $10</li>**  **<li>Product 2 - $20</li>**  **<li>Product 3 - $30</li>**  **</ul>**  **</body>**  **</html>**  **`);**  **}** );  http.listen(port, () => {  console.log(`Server is running on port ${port}`);  }); |

3) 애플리케이션을 실행

|  |
| --- |
| node app.js |

1. 웹 브라우저 접속

[**http://localhost:3000**](http://localhost:3000)

**PR 생성 후 병합**

* git add .
* git commit –m “add home page”
* git push origin nykim-feature // 원격에 feature 브랜치에 올라간다.



* Open a pull request

PR 메시지 작성을 위해 수행 결과에 대한 스크린샷을 첨부한다. 그리고 Pull Request가 merge될 때 이슈를 자동으로 닫기 위해 *closes #1* 추가한다.

마지막으로 Reviewer를 등록한후 Create Pull Request

* Reviewer는 선택하여 Pull request를 검토한다. 검토후 

Issue가 close가 되고 , Projects에서 done 컬럼으로 이동하는지 확인

* *nykim-feature 브랜치를 삭제한다.*

**커밋 후 푸시**

*git pull origin master*

*git commit -m "add new product closes #2”*

*git push origin master*

*Note:* 커밋 메시지에도 아래 키워드를 사용하면 issue를 close할 수 있다.

close, closes, closed, fix, fixes, fixed, resolve, resolves, resolved