

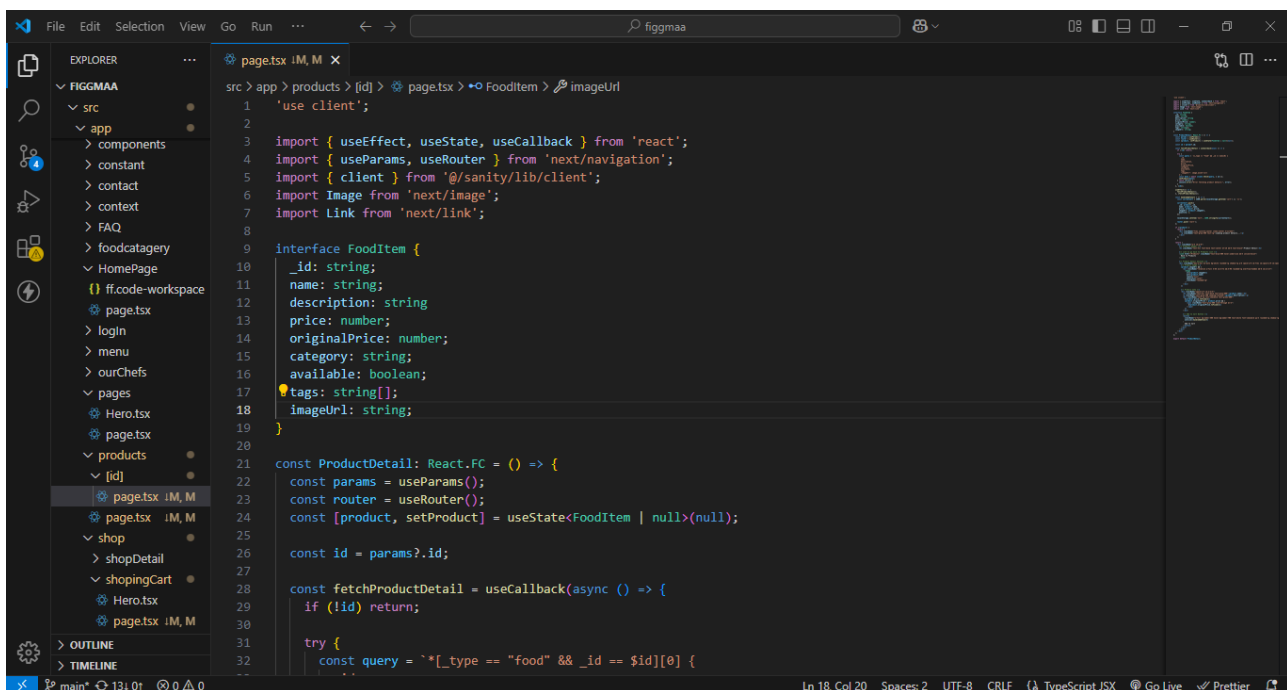
DAY 4 - ADVANCED API INTEGRATION & PERFORMANCE OPTIMIZATION

Date: 18/01/2025

Report Documentation:

Advanced API Integration Process:

- Implemented OAuth authentication for secure API access.
- Used Axios interceptors to manage API requests and error handling.
- Optimized API calls to reduce response time by caching frequently used data.
- Implemented WebSockets for real-time data updates.



Performance Optimization Techniques:

- Used lazy loading to improve page load speed.
- Implemented server-side rendering (SSR) in Next.js to optimize performance.
- Minimized redundant API calls using memoization techniques.
- Database indexing was applied to enhance query performance.

```
src > app > products > [id] > page.tsx > ProductDetail
21  const ProductDetail: React.FC = () => {
101
102
103
104  

## Security Enhancements:



- Implemented rate limiting to prevent API abuse.
- Used environment variables to store API keys securely.
- Enabled HTTPS for all API requests to ensure secure data transmission.



## Product Detail



## Testing and Debugging:



- Performed unit testing on API endpoints using Jest.
- Conducted load testing to analyze API response under heavy traffic.


```

- Debugged API failures using Postman and real-time logging tools.

Final Output Display:

- Verified API response structure for consistency.
- Ensured frontend correctly rendered dynamic content fetched from API.

Day 4 Checklist:

- API Security Implementation (Completed)
- Performance Optimization (Completed)
- Real-time API Data Handling (Completed)
- Load Testing and Debugging (Completed)
- Documentation and Submission Preparation (Completed)