

## 주요 일정

- **Final Presentation : 12월 10일 화요일 10시 강의실**
- **All Artifacts Due: 12월 15일 일요일**
- **Final Exam: 12월 17일 화요일 10시 강의실**
- **Peer Review: 12월 17일 화요일**

# Final Presentation: 12/10/2024

- **Duration: 10 min**
- **Contents:**
  - **Project Introduction**
  - **Key Features**
  - **Final Architecture**
  - **Highlight Design**
  - **Live Demo**
    - **Record demo video (for back-up, 기능 데모)**
  - **Test Results**
  - **Postmoterm (Lessons Learned)**
- **Presenters: all team members**

## 발표 팁

- 되도록 보고 읽지 말고, 청중과 아이 컨택트
- 스토리 텔링
  - 중요한 것에 중점을 두고 발표
  - 우리 프로젝트만의 고유의 가치/USP
- 팀원과 드라이 런 해 볼 것

# Project Submission Guideline

# Github Structure

- Main에 모든 문서 + 코드 최종 버전이 포함되어 있어야 함.
  - (project team member가 ) github에서 zip 으로 다운로드 받은 파일에 branch 변경 없이 accessible

## /README

- final report를 대신하여 사용하는 목적 (참고: slide#4, 5)

## /proposal/SOW.md

- 프로젝트 proposal시 사용한 README file을 copy후 업데이트 (참고: slide#3)

## /artifacts

- requirements, high-level architecture, uml design, ui design, coding standard, test case + test results 포함 (**모든 버전** 포함할 것)

## /reports

- midterm presentation slides, final presentation slides, demo video

**/source code** 및 **unit testing** 와 관련된 directory는 project team에서 결정한 대로 자유롭게

- 단 **main** 에 최종 버전 코드 포함할 것

**/classroom** 등은 그대로 둘 것

# /proposal/SOW.md

- Project Title:
  - Project Scope: 서울 시립대학교 컴퓨터과학부 2024년 소프트웨어공학 프로젝트로, Software Development Life-Cycle 을 기반으로 객체지향 소프트웨어공학 방법론을 적용하여 <<→>> 개발
  - Project Values/Motivations: 프로젝트가 주는 가치, 필요성 등
  - Project Duration: 2024년 2학기
  - Project Description (Key Features)
  - Expected Deliverables:
    - 요구사항 분석 명세서
    - 아키텍처 및 설계 (UML, UI Desig)
    - Coding standard
    - Source code
    - 테스트 케이스 및 결과
  - Project Constraints
  - Project Team Members
- 예) 효율적인 수강신청을 가능케하는 웹 기반  
어플리케이션  
교내 카페 모바일 주문을 도와주는 안드로이드  
앱  
...

# /README.md (1/2) 다음을 포함하여 자유롭게 작성

- Project Title:
- Project Scope: 서울 시립대학교 컴퓨터과학부 2024년 소프트웨어공학 프로젝트로, Software Development Life-Cycle 을 기반으로 객체지향 소프트웨어공학 방법론을 적용하여 << >> 개발
- Project Duration: 2024년 2학기
- Highlighted Features
  - 구현한 features를 리스트 합니다
  - Demo Video (Playable)
- Project Constraints
- High level Architecture 설명 및 그림
- Technology stacks (including version if needed)
- Installation guideline

## /README.md (2/2) 다음을 포함하여 자유롭게 작성

- Project Deliverables:
  - 요구사항 분석 명세서 final version: <<링크>>
  - Architecture 및 Design Documents
    - Software architecture: <<관련 문서 링크>>
    - Software Design: <<관련 문서 링크>>
    - UI Design: <<<<관련 문서 링크>>
  - Coding Standard: <<관련 문서 링크>>
  - Code: branch description 과 코드에 관한 documentation 설명
  - 테스트 케이스 및 결과
- Repository Structure
  - 각각의 디렉토리에 대한 설명
- Project Team Members (w/ roles)



# Documentation Check list

- Cover 가 잘 작성되었는가?
  - Team 명
  - Project 명
  - 날짜
  - 팀원 이름
  - Version info
- TOC (Table of Contents) 가 존재하고 링크가 제대로 되었는가?
  - 단, Test Case template은 원래대로 사용
- Revision history가 잘 작성 되었는가?
- Formal 한 문서로 형식을 잘 갖추고 있는가?

# Project Artifacts (TENTATIVE) Due: 12/15/2024

- **Requirements Doc – final version**
- **Architecture and Design Doc – final version**
  - UML diagrams
- **UI Design Doc – final version**
  - Ui design diagram
- **Coding + Review Guideline Doc – final version**
- **Test Case Doc – final version**
- **Code (zipped file) + bugs (in github) + API documents**
- **Final Presentation Slides**
- **Demo video**

## Where to Submit?

- Github 에 모든 내용을 업데이트 되었으면,
  - main branch 에 모든 문서 + 코드 최종 버전이 포함되어 있어야 함.
  - (project team member가 ) github에서 zip 으로 다운로드 받은 파일에 branch 변경 없이 accessible
- **<project\_team\_name>.zip file로 다운로드 받아서, 온라인 강의실에 제출**
- 제출 완료 후, Github에 변경된 내용은 참조하지 않습니다.
- Github 은 2025년 1학기 시작되기 전까지 accessible 할 것