**모바일/웹서비스 프로젝트 공통평가 02**

**수행 결과 보고서**

**학번 : 2018100980**

**이름 : 정남준**

[적/부판정 범례] ○:적합, ×:부적합, △:점검제외, N/A:해당없음

첨부 그림은 모두 문서에 포함하는 형식으로 저장 할 것

| **수업 요구사항** | | | | **이행현황** | | |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **구분** | **요구사항명** | **요구사항 내용** | **완료**  **여부** | **적/부 판정** | **관련증빙** |
| 1 | 제출문 | 폴더 구조 | - ROOT  - PhotoBlogServer[Dir]  : Django Server  - PhotoViewer[Dir]  : Android Client  - YOLOv5[Dir]  : Edge Client  -모바일/웹서비스 프로젝트 공통평가 02\_수행 결과 보고서.docx[File]  제출 : 탐색기, 파인더 또는 Git레포지토리 폴더 구조 캡처 화면  **주의 : “공동평가 01”과 다른 레포지토리로 제출** | –○– |  |  |
| 2 | 제출문 | Github 레파지토리 push | 아래 평가 항목에 대하여 구현 혹은 테스트를 완료하고, 캡처 이미지를 제공된 MS-Word 파일로 정리하여 Github URL을 제출 한다.  제출 : URL | ○ |  | https://github.com/omija-tea/MWSP-mobile-photo-blog-yolo |
| 3 | 강의 자료 | YOLOv5 설치 및 실행 | 강의 자료 [6페이지]  제출 : 1. 변경 없는 기본 상태의 “detect.py”실행  2. 화면 객체 인식 완료된 화면 캡처 | ○ |  |  |
| 4 | 강의 자료 | 검출기 코드 수정 | 강의 자료 [8,9,10페이지]  제출 : VSCODE 추가된 소스코드 화면 캡처 | ○ |  |  |
| 5 | 강의 자료 | 블로그 등록 결과 | 강의 자료 [12페이지]  제출 : Server 브라우저 출력화면 (localhost) (api\_root) | ○ |  |  |
| 6 | 강의 자료 | 블로그 등록 결과 | 강의 자료 [12페이지]  제출 : Server 브라우저 출력화면 (localhost) (목록 및 이미지) | ○ |  |  |
| 7 | 제출문 | 모바일 화면 확인 | 제출 : YOLO에서 검출되고 서버로 전송된 이미지를 모바일에서 확인하는 화면 캡처 | ○ |  |  |
| 8 | 제출문 | Pythonanywhere 클라우드상 동작 | 강의 자료 [12페이지]  제출 : Server 브라우저 출력화면 (Pythonanywhere) (api\_root) | ○ |  |  |
| 9 | 제출문 | Pythonanywhere 클라우드상 동작 | 강의 자료 [12페이지]  제출 : Server 브라우저 출력화면 (Pythonanywhere) (목록 및 이미지) | ○ |  |  |
| 10 | 제출문 | 기타 기획 되지 않은 편의 기능 [최대 2개] | 제출 : 해당 기능 정의, 설명 및 동작 증빙 화면 캡처  (서버/클라이언트 각각 구현이 필요한 경우 별도 요구사항으로 등록 함) | ○ |  | 1. 환경변수 분리(yolo)   서버로 요청할때 사용하는 token은 github 에 공개되면 안됩니다. .env 파일ㅇ르 분리한 후 python dotenv 를 이용하여 load 하는 과정을 추가하였습니다.   1. 2. False Positive 저항성 추가   Yolo 는 지속적으로 객체를 탐지합니다.yolo는 근본적으로 오류가 발생할 수 밖에 없습니다. 실제로는 화면에 존재하지 않았던 객체이지만 내부적으로 잘못된 판단을 하여 객체가 존재하였다고 판단하는 경우가 자주 일어납니다.  예를 들면, 실제로 화면에는 아무것도 없으나 순간적으로 객체가 나타났다고 판단하는 경우가 있습니다.  이는 –conf-thres 옵션을 사용하여 threshold를 올림으로써 어느정도 해결할 수 있지만, 결국 확률의 문제입니다.  이를 근본적으로 해결하기 위해 FP\_tolerance 라는 변수를 추가하였습니다.  탐지율은 0,1 로 토글되는 것이 아닌 0 -> 1 로 FP\_tolerance 값 만큼 증가되며, 1보다 커지는 순간 실제로 객체가 증가되었다고 판단하여 서버로 결과를 전송하게 됩니다.  즉 단순히 객체가 프레임에 한번 잡혔다고 서버로 전송되는 것이 아니라, 연속적으로 N 번 이상 잡혀야 실제 객체가 존재한다고 판단하게 되는 것입니다. |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |