

# CS Capstone Design

## Alpha Prototype Demo Grading Sheet (100 pts)

### TEAM: LangLens

**Overview:** The purpose of the Alpha Prototype Demo is to clearly demonstrate the extent to which all core user flows envisioned for the product are supported by the current implementation. The flow of the demo is very natural: you simply introduce each of the major usage scenarios, and then follow through each of them, just as an end-user would in using the product. Grading is based on how completely the current product supports all key functional aspects within a coherent, realistic user flow. Interface refinement, clunkiness, and aesthetics should be ignored for now; the focus is simply on functional ability to complete the user flow.

This template is fleshed out by the team, approved by the team mentor, and brought to demo as a grading sheet.

### Overview of major product use cases

Based on the Requirements document and subsequent development discussions with your client and mentor, briefly describe each of the key use cases for your product:

**UC1: Short title for Use Case.** <Concise 1-5 sentence description of this use case: nature of the user (admin, normal, guest) and the what they are seeking to do with the product in this use case. >

**UC2: short title.** Description. Repeat for other challenges.

**Etc. Most products will have between 2 and 5 core use cases.**

### User Flows: Detailed walk-through for each use case:

In this section, we outline the demonstrations of each use case that we have prepared, giving a step-by-step outline of the user flow that would be followed by a real user for that use case.

---

### **All use cases will be presented via a mobile phone**

**Use case 1: Homepage** - On the homepage, the user (a normal guest accessing the webapp) is seeking to choose between two detection modes: Object Detection and Text Recognition. They are also seeking to choose a native language for the base of the website to be displayed in and by default this is set to English.

User Flow: Step by step overview of user interactions with product

1. First the user will access the homepage of the webapp.
2. Then the user will be shown two buttons for scanning modes. These are labeled Object Detection Mode and Text Recognition Mode.
3. Then the user will select their preferred native language using the Google Translate Widget.
4. The Google Translate widget will translate the **homepage** of the webapp to whichever language the user selects. *By default, English is shown on the initial page load.*

5. After choosing the native language, the user will press on either of the detection mode buttons and be redirected to the corresponding detection mode URL.
  - a. If the user presses Text Recognition Mode, see Use Case 2.
  - b. If the user presses Object Detection Mode, see Use Case 3.

Evaluation and Comments:

- ✓ Convincingly demo'd each of listed challenges?
  
- ✓ Other evaluative comments:

---

**Use Case 2: User presses Text Recognition Mode button** - If the user (a normal guest on the webapp) presses the Text Recognition Mode button on the homepage, they will be redirected to the ocr.html page which contains a brief description of how to use the detection mode, a target language selection toggle featuring Spanish, French, or Korean, and a start button that will change to a stop button after the user begins the scanning process. After scanning, the user will be seeking to click on the detected word to be redirected to a learning page containing the detected words definition, sentence usage, and image.

User Flow: Step by step overview of user interactions with product

1. The user presses the text recognition button on the homepage.
2. The server redirects the user to the ocr.html page and displays the Text Recognition Mode GUI.
3. The user selects a target language from the dropdown list.
4. The server displays a message to the user confirming their target language choice.
5. Then the user presses the start button to begin the scanning process.
6. The server then displays a live video feed using the user's mobile camera of their current environment.
7. The user directs their camera at the word they want to scan within the designated scanning box.
8. The scanning window displays a live translation of the user's desired word to their selected target language.
9. The user presses the detect button to end the scanning process.
10. The user is given a learning page link related to their detected word.

Evaluation and Comments:

- ✓ Convincingly demo'd each of listed challenges?
  
  - ✓ Other evaluative comments:
-

**Use case 3: User presses Object Detection Mode button** - If the user presses the Object Detection mode button, they will be redirected to the yolov5.html page which contains description of how to use the object detection mode. The user will also have an option to select the desired foreign language, with a start button to begin the object detection process. Once the user points their camera at the desired object(s), they will click on the “scan” button to take the photo of the object, and apply the yolov5 algorithms in order to detect it. Finally the user will be clicking on the popup links attached to the scanned objects, taking them to the learning page.

User Flow: Step by step overview of user interactions with product

1. The user will press the object detection button on the homepage.
2. The user is sent to the yolov5.html page and displays the Object Detection Mode GUI.
3. The user selects a target language from the dropdown list.
4. The server displays a message to the user confirming their target language choice.
5. The user presses a button to start the scanning process.
6. The user will see their surroundings through a live camera feed.
7. The user will press a button for taking a photo of their camera.
8. The objects in the photo will be detected with a popup link presented for all of the objects.
9. The user will choose to click on any of the links and be taken to that object’s learning page.

Evaluation and Comments:

- ✓ Convincingly demo’d each of listed challenges?
  
- ✓ Other evaluative comments:

**Known short-comings: Functionality still deficient/missing:**

If there were challenges you listed earlier that were *not* covered by a demo, list here. This will hopefully be a short list...but better to be clear about where you are. If you have items here, you could list (if applicable) any pending plans/schedule to get this implemented.