

Western Washington University

Alumni Video Donations (AVD)

AVD Senior Capstone Documentation

Simone Davison, Andrew Lanum, Austin Morin
5-27-2022

WWU Alumni Video Donations

Computer Science Dept., Western Washington University, morina2@wwu.edu,
lanuma@wwu.edu, davisos2@wwu.edu, jagodzf@wwu.edu

1. PROJECT FILE STRUCTURE

1.1 Branches

Master – All the most recent commits and pushes to this repository are located here. Please refer to this branch for the most up-to-date source code and documentation files. The root folder also contains a file called random_numbers.txt which contains predicted runtime statistics of our project (not reflective of our final product).

MS-2 – All work done between the beginning of the project and Milestone 2 January 27th 2022. This branch serves exclusively as an archive for past work.

MS-3 – All work done between Milestone 2 and Milestone 3 February 11th 2022. This branch serves exclusively as an archive for past work.

MS-4 – All work done between Milestone 3 and Milestone 4 February 25th 2022. This branch serves exclusively as an archive for past work.

MS-5 – All work done between Milestone 4 and Milestone 5 March 11th 2022. This branch serves exclusively as an archive for past work.

MS-6 – All work done between Milestone 5 and Milestone 6 April 6th 2022. This branch serves exclusively as an archive for past work.

MS-7 – All work done between Milestone 6 and Milestone 7 April 20th 2022. This branch serves exclusively as an archive for past work.

MS-8 – All work done between Milestone 7 and Milestone 8 May 4th 2022. This branch serves exclusively as an archive for past work.

MS-9 – All work done between Milestone 8 and Milestone 9 May 18th 2022. This branch serves exclusively as an archive for past work.

1.2 Folder Structure

Code – All of our current Python source code is housed here for the stitching and weaving of our video content.

a. **donor-files** – Collection of input and output .csv and .xlsx files that were used to test our weaving program. Includes input and output files specifically used for WWU Give Day. Many of these files contain sensitive information. DO NOT POST, EDIT, REDISTRIBUTE, ETC.

b. **res** – Contains the music, imported fonts, and QR codes for the video content.

c. **storage** – .json files which house authentication keys that allow weave.py and store-access.py to communicate with our WWUCSGiving Firebase account (separate documentation is provided within this account).

d. **webpage** – Contains the source code for the official WWU Alumni Video Donations website that donors receive. The website utilizes Firebase and Webpack JS and is deployed using Firebase.

e. **.gitignore** – Gitignore file. Edit this to remove unwanted files when pushing to the Gitlab repository.

f. **cleanWorksheets.py** – A Python program that takes in a donor .xlsx or .csv file and automatically removes entries without an email, first name, or businesses.

g. **storage_access.py** – Helper Python file for weave.py that downloads and uploads videos from Firebase, removes temp directories, assigns access tokens to the videos, and allows the user to view or clear videos from Firebase.

h. **weave.py** – The backbone of the Alumni Video Donations weaving program. Storage_access.py and weave.py make up the entirety of the project's functionality.

i. **requirements.txt** – Required Python dependencies to run the project (see Page 5 for more details).

Docs – This folder is where all our SRS, Quarter Plans, time logs, vision and scope, and milestone documents are located. The most recent milestone documents will always be stored in the master branch.

a. **planning** – Stores all our week-by-week schedules that are refined once per milestone. All contains our testing schedule that highlights when we will be testing various parts of our python programs and website.

b. **software_requirements** – All official SRS documents finished prior to Milestone 1. These lay out what our program hopes to accomplish and includes images of our software and hardware models as well as outlining our methodologies that we will be following for the remainder of the capstone project.

c. **time_logs** – Stores Andrew Lanum, Austin Morin. And Simone Davison's time logs which are written to each time they start and stop working on the project.

d. **vision_and_scope** – Directory that houses vision and scope documents that lay out initial design decisions, deliverables, and scope limitations. This is where the "mission statement" is stored. The vision_and_scope.pdf file is where you can see all the vision and scope info in one easy-to-reference place.

2. PYTHON PROGRAM INSTRUCTIONS

clean_worksheets.py

- **Usage** – `python3 clean_worksheets.py <pathTo/donorFile OR donorFile>`
`<pathTo/indexes.txt OR indexes.txt>` If file contains spaces, use single quotes e.g.
`path/'file name has spaces.xlsx(or .csv) '`
- **Input** – A .csv or .xlsx file that is a valid WWU Computer Science Donors file (i.e. **must** have at least a first name and email column).
- **Output** – A .csv or .xlsx file that is removes rows without a first name or email, and the time it took in ms to execute the program.
- **Common Errors and Troubleshooting** –
 - `ERROR: The donor file provided in the first argument cannot be found`

Ensure that your donor file is placed in the correct location in your repository.
 - `ERROR: The file type passed in is not supported by this program`

This program works with .xlsx and .csv files ONLY. Ensure that your donor file is a .xlsx or .csv file type.
 - `The file <filename> already exists. Is it okay to overwrite this file? (y/n):`

The program has found a .csv or .xlsx with the same name and location you specified on the command line. Make sure that you are not overwriting an important file before continuing.
 - `ERROR: program was unable to delete the existing file.`

Delete or rename the file you specified then try running the program again.
 - `...Terminating the program...`

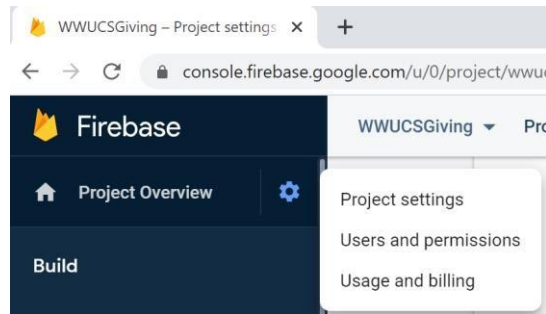
Delete or rename the file you specified then try running the program again.

storage_access.py

- **Dependency compilation** – To ensure that `storage_access.py`'s dependencies work, run `pip3 install -r requirements.txt` before running `weave.py`. `Storage_access.py` is NOT an executable program and serves as a helper file for `weave.py`.
- **Important variables** –
 - `videos` – the name of the folder on Firebase where video content is stored
 - `photos` – the name of the folder on Firebase where photo content is stored
 - `thanks` – the name of the folder on Firebase where video clips of alumni closing

content with “thank you” are stored.

- **config** – the WWUCSGiving firebase configuration. *Ensure that the credentials listed in storage_access match those found on Firebase. To see you Firebase credentials, click on the “gear icon -> Project settings” under Project Overview and look under “SDK setup and configuration”*



weave.py

- **Dependency compilation** – Before attempting to run weave.py enter `pip3 install -r requirements.txt` into the command prompt to download the required Python dependencies.
- **Important variables** –
 - **qrPath** – the directory for the video QR code image
 - **musicPath** – the directory for the video music file
 - **background** – the directory for the background of the 1280x720 ImageClips
 - **fontName** – the directory for the imported font used for the ImageClips
 - **maxWorkers** – the number of weaving processes that are utilized when running the weaving program. Setting this value to 4 is recommended for Ubuntu/Linux lab machines in WWU’s labs.
- **Usage** – `python3 weave.py <pathTo/donorFile OR donorFile>`
`<pathTo/indexes.txt OR indexes.txt>\n` If file contains spaces, use single quotes e.g. `path/'file name has spaces.xlsx(or .csv)'`

Below is a sample invocation of the weave program:

```
morina2@cf420-26:~/Documents/alumni-money/code$ python3 weave.py ./donor-files/test/1Donors2022_CLEAN.xlsx
Column containing first names (A,B,C,...,AA,BB,CC,...): G
Column containing last names (A,B,C,...,AA,BB,CC,...): H
Column containing graduation years (A,B,C,...,AA,BB,CC,...): F
Do all values for the first Donor look correct?
First Name: Kaylee
Last Name: Aalbers
Graduation Year: 2016
6 7 5
(y/n): y
The file: donor-files/test/1Donors2022_CLEAN_OUTPUT.xlsx
already exists.
Is it okay to overwrite this file? (y/n): y
```

When prompted for the first names, last names, and graduation years, open the .xlsx or .csv file and input the corresponding column name containing these three credentials as shown below:

code > donor-files > test > 1Donors2022_CLEAN_OUTPUT.xlsx

	E	F	G	H	I	J	K	L	M	N	O	P
1	4	5	6	7	8	9	10	11	12	13	14	
2		2016	Kaylee	Aalbers	Kaylee N. Aalbers	17 SW Birdsdales			Gresham	OR	97080-6747	United States
3												
4												
5												
6												
7												
8												
9												
10												

Sheet1

For this .xlsx file we inputted G, H, F, where G is the first name, H is the last name, and F is the graduation year. If you are overwriting a previously created donor file, you will be prompted whether you want to overwrite it. Below is a successful invocation of the program:

```
Making video for: Kaylee Aalbers
Adding first clip...
Adding second clip...
About to go through contentList (size of:5)
Adding video: 0.MOV
Adding video: 1.MOV
Adding video: 2.mp4
Adding video: 3.MOV
Adding video: 4.MOV
Finished content list enumeration. New size: 5
Adding final clips to video...
Final clip added. Formatting...
Adding final QR code clip...
MoviePy - Building video /home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2/Kaylee_Aalbers_24536.mp4.
MoviePy - Writing audio in Kaylee_Aalbers_24536TEMP_MPY_wvf_snd.mp4
MoviePy - Done.
MoviePy - Writing video /home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2/Kaylee_Aalbers_24536.mp4
MoviePy - Done !
MoviePy - video ready /home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2/Kaylee_Aalbers_24536.mp4

MoviePy - Done !
MoviePy - video ready /home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2/Kaylee_Aalbers_24536.mp4
/home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2/Kaylee_Aalbers_24536.mp4
<<class 'pathlib.PosixPath'>
[[438960, 'W01122911', 'YES', ' ', nan, 2016, 'Kaylee', 'Aalbers', 'Kaylee N. Aalbers', '597 SW Birdsdales Dr', nan, nan, 'Gresham', 'OR', '97080-6747', 'United States', 'kaylee.aalbers@gmail.com', 'SVBUNT 1-2', 'https://www.givingweb.org/?name=Kaylee_Aalbers_24536']]
[Errno 16] Device or resource busy: '/home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2/.nfs000000000029df00000002e'
[Errno 39] Directory not empty: '/home/morina2/Documents/alumni-money/code/weaveTEMP/5054161f-383b-4632-b0e3-6b64095b2fe2'
A PROCESS HAS FINISHED
Finished in (hh:mm:ss.ms): 0:01:44.310864
morina2@cf420-26:~/Documents/alumni-money/code$
```

NOTE: You may get Errno 16 and 39 messages in the output, but these lint errors can be safely ignored.

- **Input** – A .csv or .xlsx file that is a valid WWU Computer Science Donors file (i.e. **must** have at least a first name and email column).
- **Output** – There are three outputs to this program:
 - 1) A .csv or .xlsx file that contains a new column containing all of the hyperlinks to the personalized website with the weaved video.
 - 2) A .mp4 video file that is placed under the finished_videos folder on the WWUCSGiving storage.
- **Common Errors and Troubleshooting** –
 - ERROR: Likely multiprocessing ended abruptly. Error Listed below.

Unfortunately, there might be no immediate easy fix for this issue. Carefully read the error output log generated by the Python compiler.

- ERROR: The donor file provided in the first argument cannot be found

Make sure that you provided the correct file path for your .csv or .xlsx donor file ie. ('./donor-files/donorFolder/donorFileName')

- ERROR: The file type passed in is not supported by this program

This program works with .xlsx and .csv files ONLY. Ensure that your donor file is a .xlsx or .csv file type.

- The file <filename> already exists. Is it okay to overwrite this file? (y/n):

The program has found a .csv or .xlsx with the same name and location you specified on the command line. Make sure that you are not overwriting an important file before continuing.

- ERROR: program was unable to delete the existing file.

Delete or rename the file you specified then try running the program again.

- ...Terminating the program...

Delete or rename the file you specified then try running the program again.

- ERROR: unable to convert the column letter to an integer index.

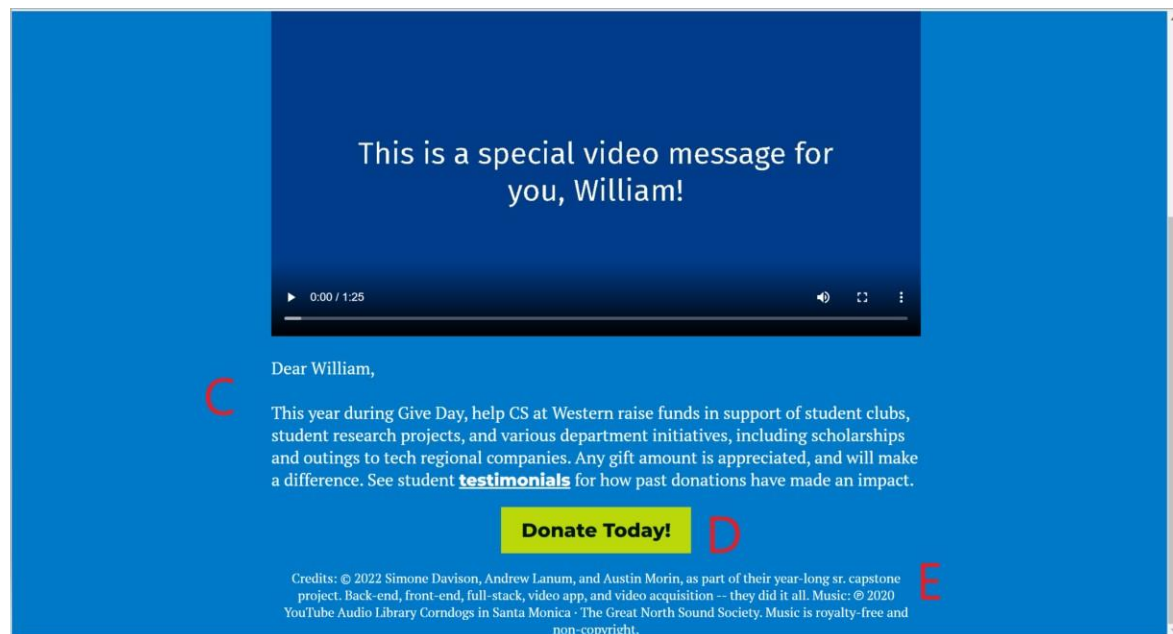
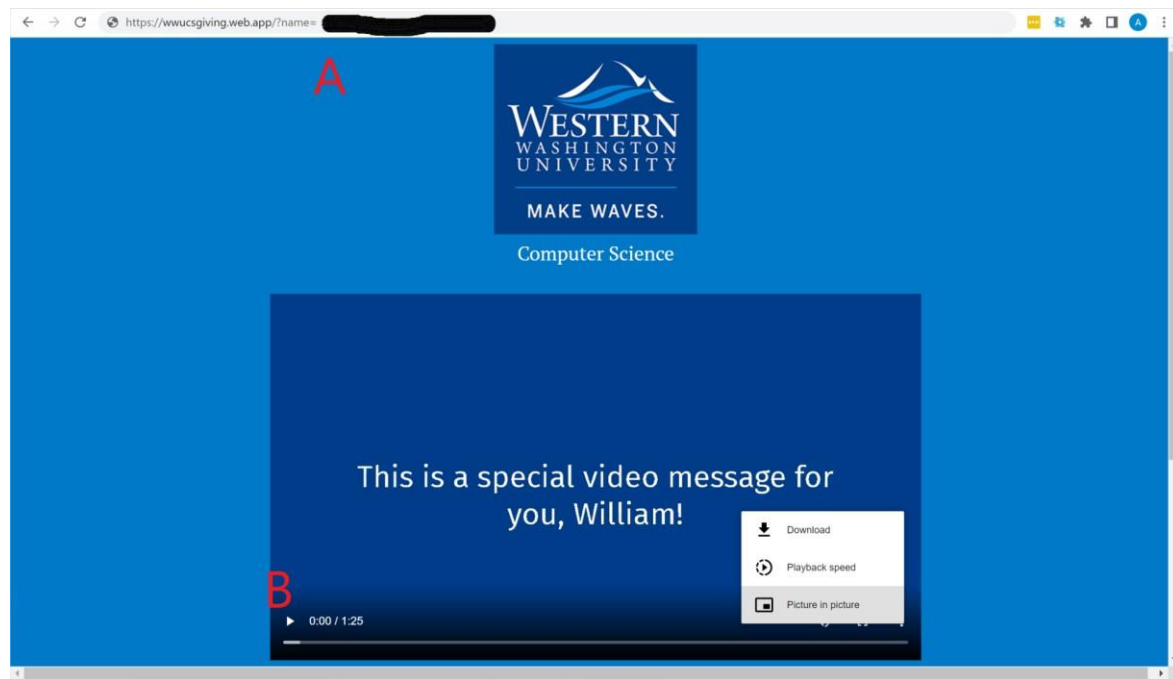
Be sure that you are entering one capital character when inputting the names of the donor file columns. The program will not recognize columns with more than one character, a lowercase character, or symbols such as @ or #.

- ERROR: invalid input expected 'y' for yes or 'n' for no.

...Terminating the program...

Program will only accept “yes” or “no” for an answer.

3. WEBPAGE INSTRUCTIONS



- A. URL Link** – The HTML source code of this page utilizes a special function that takes the substring from the URL link and appends the name to a text element in the message on the page to make the site more personable. The URL usage is as follows:

`https://wwucsgiving.web.app/?name=<DonorName>`

Where <DonorName> is replaced by the name of the video found in the finished_videos directory on Firebase (without the .mp4 extension). Sample: Austin_Morin_80123

B. Video Controls

- a. Play button** – Click this to start and stop the video. NOTE: Included in the HTML source code is a commented-out script and div element that created a green play button overlay. Ultimately this design was scrapped in the final stages of

development, the source code for it is still there and available for further testing.

- b. **Sound** – Mute or unmute the audio and control the volume with a slider.
- c. **Fullscreen toggle** – Toggle between windowed and full-screen.
- d. **Video options**
 - i. **Download** – Download the embedded video as a .mp4 file.
 - ii. **Playback speed** – Control how fast the video is playing.
 - iii. **Picture-in-Picture** – Open a smaller window of the video within the webpage and move it around and resize it.

C. Personal Message – weave.py contains a `formatURL()` function that will remove apostrophes and whitespaces from the URL and those changes will be reflected here without printing those characters in the “Dear <donorName>” message. Included in the message is a hyperlink to the generic CS Donation page: <https://cs.wvu.edu/support-computer-science>.

D. Donation Button – During WWU Give Day, this button will re-direct users to WWU’s Computer Science Give Day page. During any other part of the year, users will be directed to the generic CS Donation page.

404 Message – This video will appear if the URL is mis-typed, the video doesn’t exist in Firebase, the internet connection is lost, or there is an issue with the HTML source code. Be sure to always build the site using `npm run build` before deploying to Firebase.



QR Code – The end of the video will have a QR code that users can scan to be redirected to WWU’s Computer Science Give Day page. The QR code and message below is subject to change depending on the CS event or occasion.

