Phase 2 Sprint Meetings

Meeting Sprint phase 2: 8(10/29/22):
Made 5min video and updated readme

Meeting Sprint phase 2: 7(10/28/22):
Continue to create Camera class

Meeting Sprint phase 2: 6(10/27/22):
Create Camera class

Meeting Sprint phase 2: 5(10/26/22):
Time was spent trying to implement travis. However it was learned that GithubCI must be used. No progress was made, in regard to coverage and build badges.

Meeting Sprint phase 2: 4(10/25/22):
Help Mayank run the program; python3 had to be updated

Meeting Sprint phase 2: 3(10/24/22):
Aniruddh helped Joshua to run the program; Yolov5 had to be installed and the “models” folder had to be moved to “app” folder

pytorch 1.11 had to be downloaded

Meeting Sprint phase 2: 2(10/23/22):

Mayank/Joshua tried to make their program work

Joshua has a travisCI trial. If the program could be built on his machine, then a coverage could be produced

Meeting Sprint phase 2: 1 (10/22/22):

To do:
Try to seek coveralls issues
Learn how build the program (Mayank and Joshua)
Research how to track an object in using a video

Create a results directory with the following: cppcheck output, cpplint output?

Phase 1 Sprint Meetings

Meeting Sprint 1 (10/16/22):

• Setup coverCV
• Setup Github CI?
• Create an iteration development branch before the beginning of each iteration, All work must occur in a development branch
• Select and add a software license as a file named LICENSE
• Add a “UML” directory to the repository and an “initial” subdirectory. Add UML files to the initial directory (PDF). If design iterations occur during implementation, revise UML diagrams and add to a **“revised”** sub-directory.
• Commit often with informative commit messages (1-2 sentences including Task number used in your product backlog)
• Create a docs directory with generated Doxygen files
• URL of a 3 minute (max) video explaining the Phase 1 status of your API
• Unit tests, and test coverage
• Update readme

o Product Backlog example -- see Software Engineering textbook section 12.4.3, Table 12.7, pg 193

Table 12.7 Final Release Backlog Table With Delayed Completion of the Entire Feature Set

Plan After X Hours of Work	*0*	*100*	*285*	*405*	*475*
1: initial	10	10	10	10	10
2: inventory	30	50	50	50	50
3: multiple prices	30	40	40	40	40
4: promo prices	30	30	30	30	30
5: cashier login	20	20	20	20	20
6: multiple cashiers	30	30	55	55	55
7: cashier sessions	35	80	80	80	80
8: detailed sale	30	30	30	30	30
9: multiple line items	35	70	70	70	70
10: payment	20	20	20	20	20
11: credit payment	40	40	40	50	50
12: check payment	20	20	20	20	20
remaining effort	330	340	180	70	0
total effort needed to reach the goal	330	440	465	475	475

o Iteration Backlog example -- see Software Engineering textbook section 12.2, Figure 12.1, pg 182 (also section 13.1, Figure 13.1, pg 198)

Table 12.1 Granularities of Software Processes

Granularity	*Example*
life span	staged, V-model
Stage	evolution, servicing
Process	SIP, AIP, DIP, CIP
task	software change, build
phase, subtask	concept location, actualization
action, step	inspection of a class

o Work/Time log example -- see Software Engineering textbook section 12.3, Table 12.2, Table 12.3, pg 184, 185

Table 12.3 Time Log

Process Enactment						*Interrupt*		
Task		*Comments*	*Date*	*Start*	*End*	*#*	*Time*	*Size*
1	Ini	add cashier session	4/21	8:32	8:39	1	2	
2	CL	`CashierRecord`		8:42	8:52			340
3	IA	4 classes		8:52	9:23	2	12	420
4	Ref	extract class `Session`		9:27	10:46	3	25	36
5	Ex	install new version of Bugzilla		10:51	11:42	2	6	
6	Act	Class `Session`		1:23	2:17			98
7	Bas	add two regression faults to the backlog		2:22	3:12	3	12	12,000
8		. . .						

Product:
Phase 1: Rectangle around 1 person in a single image
- Create classes for program
- Get Postprocessing working
- Properly comment and use cplint
- Update UML diagram

Phase 2: Rectangle around 1 person in a video

Meeting Sprint 2 (10/17/22):
Short Meeting:

Building the program doesn’t work on some people’s devices

Goal: Update UML
Try to implement test cases and use GitubCI with coveralls.io

Meeting Sprint 3 (10/18/22):

Changed CMakelists
Goal:
Get GithubCI working
Create test cases

Meeting Sprint 4 (10/18/22):

Todo:
Put Video in and update readme
Update spreadsheet
Doxygen
github.ci
test cases
license.md

After the merge(with master branch), delete the iteration development branch, and create a tag indicating the iteration name (e.g. PhaseI).

Overview and purpose of the project (what does it do? Main Features? This should be a write-up of several paragraphs like a short report). Include results/performance examples.

Coveralls badge (must show 90+% coverage by end of the project)

API and other developer documentation (e.g. parameters and their definitions and default values)

- § How to build (from command line)
- § How to run the demo (from command line)
- § How to run tests (from command line)
- § How to generate Doxygen documentation (from command line)

Record known issues/bugs (in readme probably)?

Put in readme as a bug, that there is an error with the badge is constructed but shows build:failed

Put in readme as a bug, that 2 team members cannot build the program. Might have to do something with incorrect installation