

Capstone Milestone #1 Report

March 2nd, 2021

Summary

Project SCRAPBook

Team Members

- Adam Hemeon - W0434257
- Brady Getson - W0430313
- Craysyn Trottier - W0226088
- Jayden Morehouse - W0427457
- Kieran London - W0429281

Faculty Mentor David Russell



Things That Went Well

Business issues:

All the coding time for the project works well as opposed to having in class lectures.

The scope of the project seems very appropriate for this class.

No major disagreement in software or practices.

Requirements:

The level of deliverables required for the Capstone course were enough to motivate us and keep us on track to develop quality project management documentation and still provide us ample time to code.

Met all requirements on time during milestone #1, and usually ahead of schedule.

Process:

Our setup, scheduling, and coordination were all products of reliability.

The two SCRUM meetings a week seemed to work well. Frequent check-ins (twice a week) seem to be appropriate for this project.

Everyone seemed to be on the same page for the beginning project vision and stuck to it.

All members seem capable of doing their assigned work and willing to be team players.

Project Management:

The level of detail put into the project planning documents gave us a very good structure to base development on. They seem to have already helped the project and if the project were longer would have helped even more.

Milestones were well laid out in a work breakdown structure and the Agile template allowed us to easily divide the work between team members.

SCRUM meetings work well, though sometimes run longer than anticipated. Everyone can easily find the SCRUM minutes using Discord as our project management software and always have a clear idea on what to do until the next meeting.

Documentation saw great feedback from our instructor, which helped boost morale.

Overall, most activities are on track and work assignments were well made with some bugs.

Technology:

Discord for meetings and as a project management software seems the most convenient option for us and has worked well so far.

Android Studio doesn't seem to be a crowd favourite, but is the default for coding Android projects.



Things That Could Have Gone Better

Business issues:

When starting the research stage for the project during Milestone #0 all team members committed to a learning Git tutorial, but it fell to the backburner with other class work and developing the project management documentation. This caused a snag at the end of milestone #1 when trying to merge the separate branches back together. If we brushed up a bit more on version control as a group it could have saved us some time.

Requirements:

A lot of developer time was spent creating the project planning documentation for Capstone Deliverable #3 and created a "too many cooks" situation with everyone trying to work out of a shared google doc at the same time. Looking ahead and assigning the work earlier in the semester would have allowed more dev time for beginning the coding.

Process:

During this milestone we assigned members work based on large pillars of the project. (User Interface, Access to Photo Library, Storing Stock Photos, RGB Composition, Rock Paper Scissors Game Code) Having only 1 person assigned to large 'pillars' of the project is great when it's working, but when there's issues it's a lot harder to bring someone else up to speed to help. Bugs in the program were left to the person who had originally coded it to fix, and other members had a hard time providing help.

More communicative check-ins outside of twice a week SCRUM meetings was cited as a possible solution.

Project Management:

Team members are split on whether Milestone #1 seemed to be a little front loaded compared to Milestone 2, especially considering the time we had to work on Milestone 1 after finishing project planning.

Milestone 1 seems to include almost all the coding which took longer than anticipated due to bugs. Further breaking down the goals in Milestone #1 into even smaller, easier to manage chunks could have allowed us to better split the work.

Tasks often got pushed a bit back on the work queue (which includes work for other classes) when they should have priority.

Work breakdown as it relates to UI. Preferably, the program should be in their own object functions, but many tasks require access to UI elements. Further developing the working UI before starting some of the backend code might have sped up the process.

Technology:

Having everyone be familiarized with GitHub version control instead of only 1 would probably have allowed for a smoother process.



Things That Surprised Us

Business issues:

N/A

Requirements:

How much effort had to go into the beginning stages of the project and making sure the project management documentation was ready for Milestone #1.

Process:

How long it takes to fix one bug while still maintaining the framework laid out in first drafts / how hard it is to get someone up to speed on a pillar of the project when onboarding somebody to help bug fix somebody else's code.

Project Management:

How much our documentation helped with keeping the project on track.

The difficulty of assigning tasks accordingly and evenly / what happens when either member can't finish their task for unforeseeable reasons.

Technology:

The difficulty of Git and GitHub version control and how the branching/merging took more time and resources than anticipated.

**Lessons Learned**

Keep learning Git and GitHub.

Clear and open communication with the team is required, whether that be you can't get something done on time / you feel the workload is too large / or you've started experiencing bugs and the work you agreed to has multiplied.

Take initiative; be self driven, especially on a team-based dynamic where others are counting on you.

You must be careful not to burn bridges before team members have crossed it.

Time cannot be made; it must be preserved or allocated.



Final Thoughts

Things to keep: Workload is still good, just need to put time into what really needs it (and determine what deserves that time before-hand).

Things to change: Work reflections or checkups may prove to be beneficial to both work ethics and questionnaire. (Perhaps a channel on discord with an alarm, and every twenty-four hours the alarm goes off and we post what we have completed or concerns we have, if any. This may be a little too much policing though).

Overall we think the project is on a good path and the issues we have run into at this point we believe in most cases would not have been apparent to us without experiencing it firsthand.

Milestones Updates

Please include a list of your milestones, and an update to reflect the current state of each of them.

AGILE PROJECT PLAN TEMPLATE (source: Smartsheet)

PROJECT NAME	Project SCRAPBook			PROJECT MANAGER	David Russell
PROJECT DELIVERABLE	Android game built with Android Studio that takes the gameplay of a classic Rock, Paper, Scissors game which incorporates users' photos and color analysis to win. It will be deployed and available to download for free on the Google Play Store.				
SCOPE STATEMENT	The final deliverable for this project will be to have a viable working application published to the Google Play Store.				
START DATE	Feb 3 / 21	END DATE	Mar 31 / 21	OVERALL PROGRESS	30% (16/53) [As of Feb 24/21]

TASK NAME	ASSIGNED TO	START DATE	END DATE	DURATION in days	STATUS
MILESTONE 0	Everyone	Feb 3/21	Feb 17/21	14	In Progress
Researching Git Branching Workflow & Github (1.1.1)	Adam	Feb 3	Feb 10	7	Complete
Researching SCRUM as Agile Methodology (1.1.2)	Craysyn	Feb 3	Feb 10	7	Complete
Researching Using the camera / photo library / stock photos as part of an Android project (1.1.3)	Jayden / Kieran	Feb 10	Feb 17	7	Complete
Researching Color theory and how RGBA works with Rock Paper Scissors Logic (1.1.4)	Craysyn / Brady	Feb 10	Feb 17	7	Complete
Researching Deploying to Google Play Store (1.1.5)	Jayden	Feb 3	Feb 10	7	Complete

UI 1.0 Wireframe (2.1.1)	Kieran / Jayden	Feb 3	Feb 10	7	Complete
Define Coding Style Guide / Conventions(1.2.2)	Cravsyn	Feb 10	Feb 17	7	Overdue
Define Communication Plan (1.2.3)	Jayden	Feb 10	Feb 17	7	Complete
Develop GitHub Code of Conduct (1.2.1)	Adam	Feb 17	Feb 19	2	Complete
MILESTONE 1	Everyone	Feb 17/21	Mar 3/21	14	In Progress
UI Least ViableProduct to Code Against (2.1.2)	Adam	Feb 17	Feb 24	7	Complete
UI Intents / Bundles Navigation (2.1.3)	Adam	Feb 17	Mar 1	12	Complete
UI 1.0 Testing (2.1.4)	Adam	Feb 17	Mar 1	12	Complete
Selecting Up To 9 Photos At Once (3.1.1)	Jayden	Feb 17	Feb 24	7	Complete

Storing And Passing Photos The User Selected (3.1.2)	Jayden	Feb 17	Mar 1	12	In Progress
Accessing Photo Testing (3.1.3)	Jayden	Feb 17	Mar 1	12	In Progress
Find Stock Photos Online Of Varying Colors (3.2.1)	Kieran	Feb 17	Feb 19	2	Complete
Use Resources Folder (3.2.2)	Kieran	Feb 17	Mar 1	12	Complete
Keeping Photos Out Of Memory Until Needed (3.2.3)	Kieran	Feb 17	Mar 1	12	Complete
Storage Testing (3.2.4)	Kieran	Feb 17	Mar 1	12	In Progress
Determine The Color Of Given Photo By The RGB Value (3.3.1)	Brady	Feb 17	Mar 1	12	Complete
Assign Color Value To A Photo (3.3.2)	Brady	Feb 17	Mar 1	12	Complete
Color Algorithm Testing (3.3.3)	Brady	Feb 17	Mar 1	12	In Progress

Compare The User's and Computer's Photos (3.4.1)	Brady	Feb 17	Mar 1	12	In Progress
MILESTONE 2	Everyone	Mar 3/21	Mar 31/21	28	In Progress
Project Management Deliverables (1.2.4)	Everyone	Ongoing	End of Term	Ongoing	In Progress
UI Color Mockup (2.2.1)	TBD	Mar 3	Mar 8	5	
UI Implement Color Mockup (2.2.2)	TBD	Mar 3	Mar 10	7	
UI Prettifying & Color Fixing (2.2.3)	TBD	Mar 10	Mar 15	6	
UI XML Animation (2.2.4)	TBD	Mar 10	Mar 17	7	
UI QoL Changes (2.2.5)	TBD	Mar 10	Mar 17	7	
UI 2.0 Testing (2.2.6)	TBD	Mar 17	Mar 24	7	

Implement Rock Paper Scissors Rules To Determine Winner (3.4.2)	Brady	Mar 3	Mar 10	7	In Progress
Monitor Score Of The Game (3.4.3)	TBD	Mar 3	Mar 10	7	
Rock Paper Scissors Testing (3.4.4)	TBD	Mar 10	Mar 17	7	
Computer Player Randomly Use Stock Photos From Preselected Group (3.5.1)	Kieran	Mar 17	Mar 24	7	In Progress
Computer Player Add Strategic Rules To React To the Player (3.5.2)	TBD	Mar 17	Mar 24	7	
Computer Player Balancing (3.5.3)	TBD	Mar 17	Mar 24	7	
Computer Player Testing (3.5.4)	TBD	Mar 17	Mar 24	7	
LAN Host A Game (3.6.1)	TBD	Mar 17	Mar 24	7	

LAN Generate Room Code (3.6.2)	TBD	Mar 17	Mar 24	7	
Have Client Connect To Game Via WIFI / LAN (3.6.3)	TBD	Mar 17	Mar 24	7	
LAN Error Handling (3.6.4)	TBD	Mar 17	Mar 24	7	
LAN Testing (3.6.5)	TBD	Mar 17	Mar 24	7	
Google Play Create Dev Account (4.1.1)	TBD	Mar 10	Mar 24	14	
Google Play Language / Localization Settings (4.1.2)	TBD	Mar 17	Mar 29	12	
Google Play Licensing (4.1.3)	TBD	Mar 17	Mar 29	12	
Google Play Store Listing (4.1.4)	TBD	Mar 17	Mar 29	12	

Google Play Choose Distribution Options (4.1.5)	TBD	Mar 17	Mar 29	12	
Google Play Finalize Launch Checklist (4.1.6)	TBD	Mar 17	Mar 29	12	
Google Play Publishing (4.2.1)	TBD	Mar 17	Mar 29	12	
Google Play Testing (4.2.2)	TBD	Mar 17	Mar 29	12	
Github Update README (4.3.1)	Adam	Mar 24	Mar 29	5	
Github Delete Unused Branches (4.3.2)	Adam	Mar 17	Mar 24	7	
Github Set Up For Pull Requests (4.3.3)	Adam	Mar 17	Mar 24	7	