Criterion B: Record of Tasks

Task No.	Planned Action	Planned Outcome	Time Estimated	Target Completion Date	Criterion
1	Brainstorm Client and Advisor	Ask my mom to be my advisor again and contact my client Monika	0.25	1/25	A
2	Interaction One with Client	Converse with client about problem and potential solutions (See Appendix B for transcript)	0.5	1/25	A
3	Interaction One with Advisor	Converse with advisor about potential solution and platform to use for coding (See Appendix B for transcript)	0.25	1/26	A
4	Interaction Two with Client	Talk with client about the implementation of the app on Android and privacy (See Appendix B for transcript)	0.25	1/26	A
5	Interaction Three with Client	Email back-and-forth with client to decide success criteria (See Appendix B for transcript)	0.25	1/27	A
6	Complete Section A writeup	Finish Criterion A sections, determining solution, scope, and criteria	0.5	1/27	A
7	Create log-in flow	Finish rough draft of log-in flowchart	0.5	1/28	B
8	Create user flow	Draft intern and overseer user flowcharts	1.5	1/29	B
9	Set up Flutter	Watch introductory Flutter videos and set up Android Studio on App Brewery	2	1/30	C
10	Learn about Flutter UI	Watch Flutter User Interface videos on App Brewery and follow tutorials to increase understanding	3	2/1	C
11	Draft UML Diagram	Diagram initial understanding of relationships between Intern and Overseer classrooms with UML	0.75	2/2	B
12	Draft Wireframes	Diagram initial wireframes for all screens in app	2	2/5	B

13	Implement log-in and create account functionality	Follow Backslash Flutter videos and create log-in and create-account pages	2	2/7	C
14	Add ingenuity to authentication	Create intern and admin screens;add admin authentication; add own design; change InternModel for my own needs; add my own error checks	1	2/8	C
15	Figure out Reading Data	Look through tutorials and YouTube and experiment to find best method of reading data from Firestore	2	2/10	C
16	Implement Reading Data	Display and read student data from Firebase	1	2/11	C
17	Draft data flow diagram	Draft initial diagram of interaction of users with Firestore	1	2/12	B
18	Implement Search Functionality	Implement search functionality so interns can search students database and display data	1.5	2/14	C
19	Debug and Refactor Code	Fix bugs for searching functionality and refactor existing code	1	2/16	C
20	Research creating document	Learn more before implementing add functionality	1	2/18	C
21	Implement Adding Student Functionality	Create Add Student Page and implement creating document in student collection	1	2/19	C
22	Research Editing Documents	Watch YouTube videos about editing documents in functionality	1	2/26	C
23	Implement Editing Student	Create Edit Student page and ensuing functionality	0.5	2/27	C
24	Implement Deleting Student	Research and implement deleting a document/student	0.5	2/27	C
25	Add Admin Search Functionality	Implement functionality so admins can search for interns	0.5	2/28	C
26	Add Admin Edit Functionality	Allow admins to search for and then edit interns on new page	0.5	3/1	C

27	Implement Comment View	Create screen for interns so they can see their comments, reading from database which reflects overseer edits	1	3/1	C
28	Clean up code and refactor	Refactored code, fixed bugs with page navigation throughout code	0.5	3/2	C
29	Design Binary Trees	Design Initial Binary Trees for final suggestions to overseer for comments	1	3/3	C
30	Hard-Code Binary Trees in Java	Using Java, code Binary Tree with createTree() technique and RootNode	1.5	3/4	C
31	Implement path creation in java	Create path to searched text	1.5	3/5	C
32	Implement inserting and removing in java	Use recursion with insert and remove methods and implement in CommentTree	1.5	3/5	C
33	Utilize inheritance and polymorphism	Create SecondCommentTree which inherits CommentTree and implement createTree() method with polymorphism	1	3/7	C
34	Transfer Binary Tree code to dart	Work with new dart code and modulate code so it works, create suggestion pages based on intern years of experience to iterate through. Do a lot of debugging as many of the transfers from Java don't work.	4	3/9	C
35	Type up test plan	Create rigorous test plan to use to make sure code works	1	3/11	B
36	Appendix Drafts of Diagrams	Transfer existing drafts of diagrams to appendix and take screenshots of UI for appendix	1	3/12	B
37	Interaction Four with Client	Show Client User Interface of program and then note down proposed edits	0.5	3/13	B
38	Finalize User Interface	Upgrade design of all screens and then post screenshots in Criterion B	1	3/14	B
39	Finalize Code	Fix all bugs and refactor code	1	3/14	B
40	Finalize User Flow diagrams	Fix User Flow diagrams to reflect current code	1	3/15	B

41	Recreate UML Diagram	Recreate UML Diagram with new structure instead of interns	1	3/15	B
42	Finalize Data Flow Diagram	Edit Data Flow Diagram to reflect the use of collections and other items	0.5	3/15	B
43	Finalize algorithms and data structures	Write algorithms and data structures sections, implementing edited binary trees	2	3/16	B
44	Finalize Section B	Make sure Section B is cohesive and cut down on words to increase precision.	0.5	3/16	B
44	Binary Tree Technique Write-up	Explain Binary Trees fully in Criterion C	1.5	3/17	C
45	Searching and Displaying Data Technique Write-up	Explain searching and displaying data for interns and students fully.	1	3/18	C
46	Finish Criterion C	Cite sources and finish data manipulation and authentication write-ups	2	3/19	C
47	Interaction Five with Client	Evaluate project together and discuss future extensions	0.5	3/19	C
48	Finish Appendix	Finish Works Cited fully and tidy up any loose ends in the Appendix	0.75	3/20	C
49	Finalize Section A	Do final edits to Section A	0.5	3/21	C
50	Finish Section E	Write evaluation and future extensions based on interaction five with client	1	3/21	E
51	Record video	Record video by using test plan as script	1	3/22	D
52	Zip files and upload	Finish and submit IA	0.25	3/22	E

Total Hours: 56.25