	Candidate				
Task #	Planned action	Planned outcome	Time estimate	Target completion date	Criterion
1	Preliminary discussion with client	Client will outline the basic problem	20 minutes	23 June	A
2	Initial discussion with computer science teacher	Ideas approved by advisor	15 minutes	25 June	A
3	Follow up interview with client	Discussion about solutions and design choices	30 minutes	15 July	A
4	Draw UML diagrams to establish class relationships	Clarity on the relationship between classes	15 minutes	8 September	B
5	Draw flowcharts for algorithms	Understand the flow of the application and for each sub program	2 hours	13 September	B
6	Data flow diagram	Create a diagram to show how data flows within the application and between it and the database	30 minutes	19 September	B
7	Create some relevant pseudocode for algorithm	Understand missing logic not shown by flowcharts (updating balance)	1 hour	23 September	B
8	Create test plan based on success criteria	Understand how to test the success criteria to ensure all requirements are met	1 hour	23 September	B
9	Start designing GUI for client	To ensure the client is happy with the design before delivering a final product	8 hours	10 October	B
10	Show prototype to client	For approval that it is easy to understand, aesthetically pleasing, and easy to navigate	30 minutes	15 October	B
11	Make any necessary changes	Make changes to highlighted problem	30 minutes	17 October	B

		areas in GUI			
12	Set up programs — MySQLWorkbench and IntelliJ	Download connector between database to codebase	10 minutes	5 November	C
13	Set up IntelliJ	Create classes: user, lesson, DAO, page controller, lesson controller, student controller, calendar controller, login controller	10 minutes	5 November	C
14	Learn how to set up DAO and check connector works	Connector works and DAO can access database	15 minutes	5 November	C
15	Create JSP pages with very rough GUI	JSP pages for each GUI exists	15 minutes	8 November	C
16	Create CSS files for each JSP	Ensure they exist for coding later and connect them to JSP files	15 minutes	8 November	C
17	Learn how to use a Controller and HTTP libraries	Be able to switch between these JSP pages (no data yet) and request set attributes can be used in JSP	30 minutes	10 November	C
18	Practice java and begin login validation in login controller through DAO	Be able to validate if a user exists and if their password is correct, then navigate to homepage	3 hours	25 November	C
19	Learn how to use Model And View and get request dispatcher	Be able to navigate to other pages easily	15 minutes	27 November	C
20	Learn to use jstl library for html	Be able to utilise c:forEach	20 minutes	27 November	C
21	Create DAO functions and complete page controller for teacher homepage as well as JSP	Be able to display all lessons today and tomorrow in time order (using c:forEach)	3 hours	1 December	C
22	Create dropdown menu with links to	Click on menu to navigate to all other	30 minutes	6 December	C

	other pages (with a common css file)	pages			
23	Design student info page and controller to send student details to JSP through DAO	Be able to see all students, their balances, and be able to edit them or create a new one	5 hours	17 December	C
24	Design bookings page and controller to send lesson details to JSP	Be able to see approved lessons latest to earliest, and unapproved lessons on the other side and also be able to create a new lesson, or edit current lessons, or approve unapproved lessons	4 hours	6 January	C
25	Design calendar page (use pre-made calendar)	Change css and edit Sunday - Saturday format to Monday - Sunday, adapt calendar for program use	1 hour	6 January	C
26	Create calendar week page with controllers finding the date that was clicked, and its week and then send lessons on that week	Be able to see all lessons in the week selected and create a new lesson or block time	5 hours	11 January	C
27	Code logout function	Ends session and returns to login page using Model And View	15 minutes	12 January	C
28	Create student version of homepage (controller, DAO, and JSP)	View all upcoming and unapproved lessons, and view all past lessons, and outstanding balance	3 hours	20 January	C
29	Create student version of calendar (controller, DAO, and JSP) (request lesson function)	Can only request lesson in main calendar and calendar week, and can only view their own lesson details	2 hours	25 January	C
30	Create validation for each step of controller (get if session user is a teacher)	Prevent students from accessing teacher-side	30 minutes	27 January	C

31	Code lesson counter at login to update student balances	Ensure student balances are up to date at every log in	45 minutes	10 February	C
32	Add css to make all pages similar to GUI	Ensure site is easy to navigate and looks nice according to client’s wishes	10 hours	1 March	C
33	Black-box testing to check any problem areas	Find any bugs or errors	1 hour	5 March	C
34	White-box testing to investigate these problem areas and make amends (using debugger tool)	Program functions smoothly and as intended	2 hours	5 March	C
35	Show product to client	Ensure it completes success criteria	20 minutes	7 March	C
36	Give product to client for beta-testing	Client can create feedback	10 minutes	7 March	E
37	Discussion with client about any improvements necessary	Understand client and end-user feedback	20 minutes	17 March	E
37	Make any essential changes	Program works as intended with new features/ changes added for regular use	30 minutes	17 March	E
38	Plan future updates	More usable program	20 minutes	17 March	E