

React Web Project Assessment Rubric

The assessment will be marked out of 100 and then scaled to be out of 20. “Custom” is used in the rubric to allow the rubric to also apply to custom projects.

Note: COMP7240 students have an extra task as per the Course Profile.

Table 3: Grading Rubric.

ID	Feature Description	COMP2140 Max Grade = 100	COMP7240 Max Grade = 100
1	Core Functionality	60 marks	60 marks
1.1	Package.json includes all dependencies, React code runs without errors, follows required folder naming convention and includes an example Interview that can be used in the Project Demonstration and Code Review.	5	5
1.2	Includes the ability to list, add, edit and delete Interviews Custom. No paging is required. Add/Edit forms should include all mandatory fields provided by the Interview API endpoint. Includes summary details for Number of Questions and Applicant Status. Use best practice for loading data and having shared components for add/edit forms.	15	15
1.3	Includes the ability to list, add, edit and delete Interview Questions Custom. No paging or re-ordering is required. Add/Edit forms should include all mandatory provided by the Question API endpoint. Interview Questions are added to an Interview. Note: Only text questions are required to be supported. Use best practice for loading data and having shared components for add/edit forms.	10	10
1.5	Add/Edit Applicants to an Interview and generate a unique link that allows the Applicant to take the interview. Add/Edit forms should include all mandatory provided by the Applicant API endpoint.	5	5

1.6	Implement an audio recorder with audio transcription using either Transformers.js or your own server side api (using Express.js to run a Whisper model) and convert speech to text. Applicants must use the audio recorder to answer a question.	10	10
1.7	The Interview UI should only display 1 question per page, only have a Next button, and allow the Applicant to answer the question (using speech) before continuing. Only the text of the answer should be saved back via the Answer API endpoint.	10	10
1.8	View status of the Interview (i.e., Not Started or Completed) for each Applicant and display the questions and answers	5	5
2.	User Interface and User Experience Design	15 marks	10 marks
2.1	Uses a CSS Frameworks and Custom CSS where appropriate. Includes a consistent header, footer and navigation bar.	5	5
2.2	Navigation and workflow are intuitive, guiding users through each step in the process of setting up an interview, with features that are easy to use and clear instructions provided where needed.	10	5
3	Code Style and Quality (Note: Max Grade will be capped at 50% grade points if only 50% of the grade for Core Functionality is achieved)	15 marks	10 marks
3.1	JSX and Javascript/Typescript adheres to Functional paradigm (i.e. avoids embedded imperative logic).	5	4
3.2	Breaks down functionality into well-defined, re-usable components. Centralised Fetch API access, reducing duplicate code in each component that uses the API.	5	4
3.3	Implements appropriate error handling for each component, ensuring robust error detection and management while displaying clear, user-friendly error messages that enhances user experience and facilitates troubleshooting.	2.5	1
3.4	Well formatted and documented code with inline comments and function doc strings included.	2.5	1
4	Advanced GenAI Feature	10 marks	10 marks
4.1	Creative integration of GenAI into the Interview Application	10	10



5	Responsible AI (For COMP7240 Students – Research Question)		10 marks
5.1	Write a short bullet-point outline highlighting the responsible AI and accessibility considerations involved in using speech-to-text translation for hiring interviews at an organisation.		10