COMP2140/7240 – Web/Mobile Programming

# Lecture Week 6:
# Introduction to A2 & Getting Started

**Dr Aneesha Bakharia (Senior Lecturer, EECS)**
**Email: a.bakharia1@uq.edu.au**

# React Web App

| | |
|---|---|
| **Mode** | Product/ Artefact/ Multimedia |
| **Category** | Computer Code, Project |
| **Weight** | 20% |
| **Due date** | 26/09/2025 5:00 pm |
| **Learning outcomes** | L01, L02, L04, L05, L06, L07, L08 |

## Task description

In the *React Web App* assessment, students will build an interactive web app with a cross-platform JavaScript framework (React), testing across web browsers & device sizes. The web app will communicate with web & device APIs. Throughout, students will evaluate the design, implementation & architecture for delivering a complex web app.

The React Web App will be graded in the associated Project Demonstration and Code Review session. The React Web App must be submitted prior to the Project Demonstration and Code Review session.

Artificial Intelligence (AI) and Machine Translation (MT) are emerging tools that may support students in completing this assessment task. Students may appropriately use AI and/or MT in completing this assessment task. Students must clearly reference any use of AI or MT in each instance.

A failure to reference generative AI or MT use may constitute student misconduct under the Student Code of Conduct.

https://course-profiles.uq.edu.au/course-profiles/COMP2140-60496-7560#assessment-detail-2

# React Web App

| | |
|---|---|
| **Mode** | Product/ Artefact/ Multimedia |
| **Category** | Computer Code, Project |
| **Weight** | 20% |
| **Due date** | 26/09/2025 5:00 pm |
| **Learning outcomes** | L01, L02, L04, L05, L06, L07, L08 |

## Task description

In the *React Web App* assessment, students will build an interactive web app with a cross-platform JavaScript framework (React), testing across web browsers & device sizes. The web app will communicate with web & device APIs. Throughout, students will evaluate the design, implementation & architecture for delivering a complex web app.

Compared to COMP2140, COMP7240 includes an additional research component in the assessment, requiring students to critically investigate current practices and challenges in accessibility and responsive design.

The React Web App will be graded in the associated Project Demonstration and Code Review session. The React Web App must be submitted prior to the Project Demonstration and Code Review session.

Artificial Intelligence (AI) and Machine Translation (MT) are emerging tools that may support students in completing this assessment task. Students may appropriately use AI and/or MT in completing this assessment task. Students must clearly reference any use of AI or MT in each instance.

A failure to reference generative AI or MT use may constitute student misconduct under the Student Code of Conduct.

COMP7240 has a different weighting and an additional task

https://course-profiles.uq.edu.au/course-profiles/COMP7240-60085-7560#assessment-detail-2

# What do you need to build?

**Table 1: Available Project Options**

| Project Name & Description | Main Features |
|---|---|
| **ReadySetHire - AI-Powered Interview Platform**<br><br>ReadySetHire is a modern web application designed to streamline the hiring process. It provides recruiters with an easy way to set up interviews, add interview questions, invite applicants, and automatically capture their spoken responses through speech-to-text transcription. By using the latest AI technology, ReadySetHire allows applicants to answer naturally in spoken language, making the experience more authentic and efficient.<br><br>This tool can be used for screening applicants at scale and includes opportunities to extend functionality with Generative AI, such as providing feedback or auto-suggesting interview questions. In the future, ReadySetHire could also serve as a practice platform for job seekers, helping them prepare for interviews with realistic AI-driven simulations. | Allows users to:<br><br>- Create, edit, and delete Interviews<br>- Add interview questions<br>- Invite applicants through unique interview links<br>- Applicants answer using an audio recorder, with AI transcription converting speech to text<br>- One question per page interview flow for clarity and focus<br>- Track interview status (Not Started and Completed)<br>- Review Applicant question answers<br>- Creative Integration of GenAI |

**Custom Project Guidelines**

- You must submit your project idea proposal to **COMP2140@eecs.uq.edu.au** by Mon 8 Sept 2024.

- You will need to provide your own API to store and retrieve your data. You can use external services or deploy an API to your own UQ Cloud Zone. If you are already familiar with React, you might like to explore Next.js and build your front-end and back-end with it. The Teaching Team have used PostgREST (https://docs.postgrest.org/en/v12/) for the API and a self-paced tutorial is provided, if you also wish to take this approach. With PostgREST you only need to setup a database and then enable the RESTFul API.

- Custom projects can also be a Chrome or VS Code extension but must use React.

From the Brief

# Example Screens

# Some Guidelines to for viewing Sample Screens

1. They are sample screens.

2. You need to design your own layout and look and feel using a CSS Framework.

3. The sample screens are not designed for usability and don't include too many user instructions. You will need to include these in your app.

# The Interviews Screen

**ReadySetHire – AI-Powered Interview Platform**

Interviews

## Interviews Management

**+ Add Interview**

| Title | Job Role | Status | Questions | Applicants | Actions |
|---|---|---|---|---|---|
| Frontend Developer Interview | Frontend Developer | Published | ⊘ 3 Questions | 👥 2 Applicants  <br> 1 Completed  1 Pending | ✏️ 🗑️ |
| Backend Engineer Assessment | Backend Engineer | Draft | ⊘ 0 Questions | 👥 0 Applicants | ✏️ 🗑️ |

© 2025 ReadySetHire – Streamlining the hiring process with AI

7

# The Add Interview Form

**ReadySetHire - AI-Powered Interview Platform**

Interviews

← Back to Interviews

## Add New Interview

**Title ***

Interview Title

**Job Role ***

Job Role

**Description ***

Interview Description

**Status ***

Draft ▾

Add Interview    Cancel

# The Questions Screen

**ReadySetHire - AI-Powered Interview Platform**

Interviews

← Back to Interviews

## Questions Management
Interview: Frontend Developer Interview

+ Add Question

| Question | Interview | Difficulty | Actions |
|----------|-----------|------------|---------|
| Tell me about your experience with React | Frontend Developer Interview | Easy | ✏️ 🗑️ |
| How do you handle state management? | Frontend Developer Interview | Intermediate | ✏️ 🗑️ |
| Explain the virtual DOM | Frontend Developer Interview | Advanced | ✏️ 🗑️ |

© 2025 ReadySetHire – Streamlining the hiring process with AI

# The Applicants Screen

**ReadySetHire - AI-Powered Interview Platform**

Interviews

← Back to Interviews

## Applicants Management
Interview: Frontend Developer Interview

+ Add Applicant

| Name | Email | Phone | Interview | Status | Actions |
|------|-------|-------|-----------|--------|---------|
| Mr John Smith | john@email.com | 555-0123 | Frontend Developer Interview | Completed | 🔗 Copy Link  Take Interview ✎ 👥 |
| Ms Sarah Johnson | sarah@email.com | 555-0456 | Frontend Developer Interview | Not Started | 🔗 Copy Link  Take Interview ✎ 👥 |

© 2025 ReadySetHire – Streamlining the hiring process with AI

# The Take Interview UI

**ReadySetHire - AI-Powered Interview Platform**

Interviews

## Welcome to Your Interview

**Applicant Details:**
Ms Sarah Johnson
sarah@email.com

**Interview:**
Frontend Developer Interview
Interview for React/JavaScript frontend position

Start Interview

# The Take Interview UI - Question

**ReadySetHire - AI-Powered Interview Platform**

Interviews

Question 1 of 3

**Question 1:**

Tell me about your experience with React and how you've used it in previous projects.

**Audio Recorder:**

▷ Start Recording

⬤ Click Start Recording to begin

▷| Next Question

© 2025 ReadySetHire - Streamlining the hiring process with AI

# The Take Interview UI – Thank You

**ReadySetHire - AI-Powered Interview Platform**

Interviews

✓

## Thank You for Completing the Interview!

Your responses have been successfully recorded.

Our team will review your interview and get back to you soon.

You may now close this window.

Interview completed for: **Sarah Johnson**
Interview: **Frontend Developer Interview**

© 2025 ReadySetHire - Streamlining the hiring process with AI

# You have design flexibility... (Alternate Design)
## Marks Allocated in the Rubric

# How do I get started?

# How do I get started? (part 1)

1.  Read the Assessment Brief & Rubric

2.  Use Vite to create the starting React Project
    e.g.
    npm create vite@latest interview-app -- --template react
    cd interview-app
    npm install
    npm run dev

3.  Review and understand the file structure of the generated app.

4.  Design your UI using a CSS Framework. Bootstrap, Tailwind CSS, Daisy UI, Mantine, Shdcn UI are all allowed. It is a good idea to create HTML mockups of each screen or prototype using React components

5.  Install and setup React Router (Lecture 5 slides & Week 6 Applied Class)
    npm install react-router-dom

6.  Split up the HTML and make components. Make the layout.
    Move your HTML into the components. Setup routes for each page/screen.

# How do I get started? (part 2)

7. Learn to use a RESTful API (overview in next slide)
   This can be challenging so we provide some sample code to get you started! Get your token from BB My Grades

8. Build the Interview, Question and Applicant List Screens

9. Build the Add/Edit Forms for each

10. Build the Take Interview UI (with the Audio to Text transcription)
    The Week 7 Lecture will cover Audio to Text transcription!

11. Build your Advanced Feature – most importantly have fun!

| 4 | Advanced GenAI Feature | 10 marks | 10 marks |
|---|---|---|---|
| 4.1 | Creative integration of GenAI into the Interview Application | 10 | 10 |

12. COMP7240 – Extra Research Task

| 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 |

From the Rubric

# RESTFul API's

# What is a RESTful API?

- Representational State Transfer

- Architectural style for distributed systems

- **Stateless Communication**
  - Each request from a client contains all the information needed for processing

- **Resource-Based**
  - Resources are identified by URLs
  - Once you know the resource name the process of listing, creating, editing and deleting is the same

- **HTTP Methods**
  - GET, POST, PATCH/PUT, DELETE

# Interview App RESTful API

https://comp2140a2.uqcloud.net/api

## React Web Project Assessment – RESTFul API

This document provides a summary of the available RESTful API endpoints .

### API Overview

- **URL**: https://comp2140a2.uqcloud.net/api

### Authentication

Anonymous access is disabled. Authentication via an JSON Web Token (JWT) is required for all endpoints. The JWT is provided via a column in Blackboard My Grades. Example API calling code is provided to get you started.

### Interview Endpoint

The `/interview` endpoint allows management of interview resources.

### Interview Fields

| Field Name | Type | Size | Required | Description | Example Value |
|------------|------|------|----------|-------------|---------------|
| id | Integer | - | Yes (PK) | Auto-generated identifier | 1 |
| title | String | 255 | Yes | Title of the interview | "Front-end Developer Interview" |
| job_role | String | 255 | Yes | Role being interviewed for | "Senior Front-end Developer" |
| description | Text | - | No | Detailed description | "Interview for candidates with React experience" |
| status | String | 50 | Yes | Publication status | "Published", "Draft", or "Archived" |
| username | String | 255 | Yes | Owner of the interview | "s10000" (This is your student id) |

- Implemented using PostgREST which maps a Database to an API.

- https://docs.postgrest.org/en/v12/

# Sample – Create, Read, Update Delete (CRUD)

- You can perform CRUD (Create, Read, Update, Delete) operations on a resource (eg Project) using various HTTP methods.

- Getting all Interviews (GET)

  https://comp2140a2.uqcloud.net/api/interview

- Reading an Interview by id (GET)
  https://comp2140a2.uqcloud.net/api/interview?id=eq.1
  Note this is slightly different from usual RESTFul but provides more querying capability.
  See Operators section: https://docs.postgrest.org/en/v12/references/api/tables_views.html

- Create a new Interview (POST)

- Edit an Interview (PUT/PATCH)
  Note PostgREST handles this differently. Read the docs.

# Example API Code

**Don't forget to include/change to your JWT from Course GradeCentre**

```javascript
1   import fetch from 'node-fetch';
2
3   // Base URL for the Interview App RESTful API
4   const API_BASE_URL = 'https://comp2140a2.uqcloud.net/api';
5
6   // JWT token for authorization, replace with your actual token from My Grades in Blackboard
7   // From the A2 JSON Web Token column, view Feedback to show your JWT
8   const JWT_TOKEN = 'YOUR TOKEN HERE';
9
10  // Your UQ student username, used for row-level security to retrieve your records
11  const USERNAME = 's1000000';
12
13  /**
14   * Helper function to handle API requests.
15   * It sets the Authorization token and optionally includes the request body.
16   *
17   * @param {string} endpoint - The API endpoint to call.
18   * @param {string} [method='GET'] - The HTTP method to use (GET, POST, PATCH).
19   * @param {object} [body=null] - The request body to send, typically for POST or PATCH.
20   * @returns {Promise<object>} - The JSON response from the API.
21   * @throws Will throw an error if the HTTP response is not OK.
22   */
23  async function apiRequest(endpoint, method = 'GET', body = null) {
24    const options = {
25      method, // Set the HTTP method (GET, POST, PATCH)
26      headers: {
27        'Content-Type': 'application/json', // Indicate that we are sending JSON data
28        'Authorization': `Bearer ${JWT_TOKEN}` // Include the JWT token for authentication
29      },
30    };
31
32    // If the method is POST or PATCH, we want the response to include the full representation
33    if (method === 'POST' || method === 'PATCH') {
34      options.headers['Prefer'] = 'return=representation';
35    }
36
37    // If a body is provided, add it to the request and include the username
38    if (body) {
39      options.body = JSON.stringify({ ...body, username: USERNAME });
40    }
41
```

**Sample code in interviewapp_sample_api_access.zip**

# Tips

- How can I approach building the all the screen in an efficient manner?

  Centralise your API handling code.

  Build both List screens and then both Add/Edit forms. Re-use the form for both adding and editing.

- What components can I use?

  The Rubric lists example open source React components. You can use other components as long as the components are not high level form building or API calling so that they don't allow you to learn the fundamentals of React. It is up to you to review the documentation.

- You'll learn about Forms and RESTful API's in the Week 7 Lecture.

# A2 Q&A

# Thank you

THE UNIVERSITY
OF QUEENSLAND
AUSTRALIA

CREATE CHANGE