Web Application Proposal: Empowering Users to Learn React.js

Anthony Gudiel, Jashan Gill, Clement Lau, Duc viet Nguyen
Group 16 Moons

CMPT 276 D300 - Fall 2023

GitHub Repository:
https://github.com/anthony-gudiel/reactjs-learning-app

Table of Contents

Project Overview

With our web application, we aim to create a platform where beginners and amateurs can gain a basic understanding of how to utilize React.js, a popular JavaScript library. Through a combination of artificial intelligence, instructional videos, and an in-browser development environment, we hope to deliver an easily digestible and pleasant experience where aspiring developers can learn to get comfortable with React.js and set the foundation for their coding endeavors.

One of our key features that we aim to implement in our web application is the use of AI-powered learning assistance. Through the use of artificial intelligence, we look to help users with any questions that they may have after reading through the content, provide practice problems and quizzes, as well as help them find possible errors in their code.

In addition to our usage of artificial intelligence, we want to create a library of comprehensive video tutorials. Utilizing videos from experienced programmers, we aim to embed important video tutorials for users to follow, allow users to search for their own specific tutorials that we may not have included ourselves, and allow users to create their own playlist of videos that they find useful.

If time permits, the last feature we want to implement is an in-browser development environment for the user. Our goal is for the user to have all the resources they could possibly need in one web application, and including an in-browser development environment brings us one step closer to our vision. With this, users won’t have to seek external resources and install a separate integrated development environment. This may be especially useful for beginners who are not familiar with IDEs and their functionalities.

With AI-powered learning, instructional videos, and in-browser coding environments, we aim to make learning React.js an enjoyable and accessible experience for a broad audience.

Our SDLC Model

As a group, we decided that it would be best for us to utilize the V-Shaped software development lifecycle for this project. The V-Shaped SDLC provides us with a clear, structured, and systematic approach to our project, as the majority of our group does not have much experience with what we’re aiming to create. By having this structured approach, it gives us a well-defined roadmap to follow, providing us with a feeling of clarity and simplicity that some other SDLCs may not offer. On top of that, the V-Shaped model consists of constant testing throughout and after every stage, which will help us catch any errors that may be caused during development.

Alongside the V-Shaped model, we also implemented some elements of Kanban, assigning different objectives to each team member, each with different priorities and respective deadlines. We thought this would be beneficial for us, as communication is crucial during the development of a project like this, especially when our primary mode of communication is not in person.

Ultimately, by using the V-Shaped model and some elements of Kanban, we believe that we'll be able to reach our goal of creating a high-quality web application through the use of structure, testing, and communication.

APIs and Features

The APIs that we have decided to incorporate play a pivotal role in enhancing our application's functionality. We decided on the following APIs, as we thought that they could give us the best chance of bringing our vision to life:

YouTube Data API

- Why Youtube Data API?
  - Popularity: YouTube is currently the biggest video hub in the world, which comes with the benefit of a large video library for React.js.
  - UI/UX: It is likely that everyone is familiar with YouTube UI and how to navigate in it.
  - Cost: The cost of the YouTube Data API is free.
  - Reliability: YouTube's servers are stable and the tutorials on the platform are highly reliable and of high quality.
  - Seamlessness: The YouTube Data API can perform most user actions without the need to visit the YouTube website itself. This provides a seamless experience for users who are looking for visual tutorials.
- Features:
  - Search for specific tutorials: Users can use an embedded search bar to search for relevant videos from the YouTube platform.
  - Create a playlist of useful videos: After watching a video, users will have an option to add the video to an existing or new playlist.
  - Subscribe to a channel: Users can also subscribe to the channel of their current video through a dedicated button.

OpenAI GPT-3.5 API

- Why GPT-3.5 API?
  - Popularity: GPT-3.5, which is the generative AI for the popular ChatGPT application, is highly rated and reliable.
  - Cost: The cost per 1000 tokens is low compared to GPT-4, and OpenAI provides $5 worth of free credits.
  - Real-time response: GPT-3.5 is able to take in a variety of prompts and return customized answers in real-time.
  - Content generation: GPT-3.5 can also generate different practice questions and solutions, which aligns with our goal of AI-based React.js education.
- Features:
  - Interactive Q&A: Beginners often have questions that may not be thoroughly covered in standard tutorials. GPT-3.5 can engage in interactive dialogue, allowing users to ask

    questions in natural language and receive clear, conversational responses. This can make the learning process feel more personal and less intimidating.
    - Question generation: GPT-3.5 can generate questions that correlate with the tutorial subject from a predetermined prompt, providing users the opportunity to test their knowledge and receive feedback.
    - Code debugging and review: GPT-3.5 has the ability to understand and generate code, which means it can assist users in debugging by suggesting corrections and improvements to their code. This immediate feedback is invaluable for learning and can help prevent the frustration that is often accompanied with debugging.

CodeSandbox API

- Why CodeSandboxAPI?
    - Popularity: CodeSandbox is one of the most popular cloud-based IDE APIs. It can run code in-browser without the need to install external IDEs.
    - Cost: The cost of the API is free.
    - Real-time and hands-on experience: By implementing the CodeSandbox API, users can actively participate and engage with the tutorials through its in-browser programming interface.
    - Safe Environment for Experimentation: Users can freely experiment with React.js code without the risk of breaking anything in the main application. This encourages learners to try out new things and learn from their mistakes without any repercussions.
- Features:
    - In-browser programming interface: An interface will be provided in our web application where the user can write code in their browser without needing to install an IDE.
    - Web-based compilation: The code is compiled on the website using API calls. Any errors in the syntax and compilation will be shown.
    - Real-time feedback: A live panel will display the user's React.js application in real-time. This provides immediate feedback that encourages the user to experiment with new changes.

User Stories

Creating user stories for this project — and generally most projects — is one of the key steps we need to undertake before starting development. By creating user stories for each of our APIs, they provide us with guidelines and help us design our application from the user's perspectives and expectations so that we can create a quality product that satisfies the user's needs.

YouTube Data API

As a beginner with only basic programming knowledge, I want to learn UI/UX from scratch. The diversity of different resources helps me understand the material from different perspectives and angles. I wish there was a search bar and subscribe button so that I could easily search for a variety of videos and subscribe to my favourite channels.

As an aspiring React.js developer, learning from only one source is not clear. Although there are numerous videos about React.js, they do not correspond well with the material I'm working on. I wish there were videos that are suggested and listed on the same website as the tutorial so that I don't get distracted by unnecessary information and tab-switching.

GPT-3.5 API

As a student who is busy with work and assignments, I want tools that quickly summarize paragraphs and respond to any questions I have so that I can go through the learning materials as efficiently as possible.

As a new learner, I want a diverse collection of questions and their answers after each section so that I can review my learning progress and understand any mistakes I make. Perhaps an AI could generate the questions so that I am not limited to the ones provided by the creators of the website.

CodeSandbox API

As someone who just wants to try React.js out of curiosity, I want an in-browser development environment without the hassle of installing unnecessary IDEs that I do not plan to use in the future.

As a learner who prefers hands-on experiences, I hope that there is an embedded development environment so that I can code and receive visual feedback as I go through the learning materials.

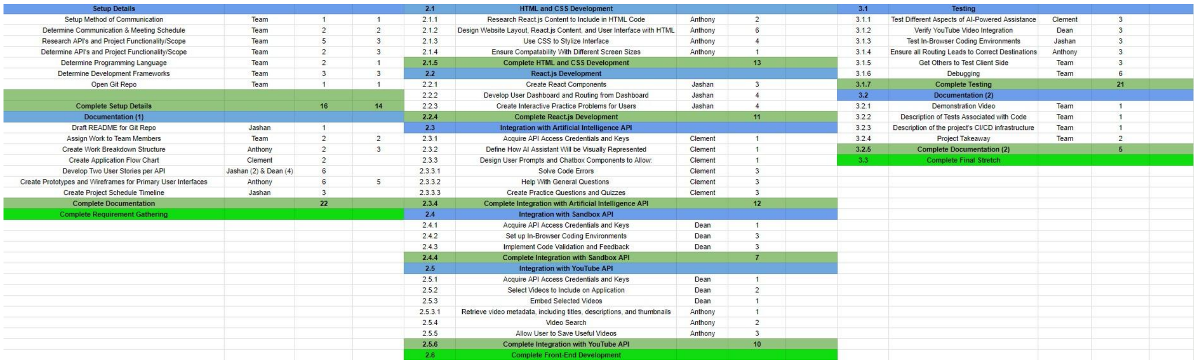
Tech Stack

For our coding languages, APIs, and frameworks, we made sure to choose elements that would not only help us to create a robust application, but also help us to advance our coding knowledge with relevant skills that apply in our future careers. Considering these reasonings, we decided on the following:

- Front-End: React.js, HTML5, CSS3, JavaScript ES6+
- AI Integration: OpenAI GPT-3.5 API
- Browser-Based Editor: CodeSandbox API
- Video Integration: YouTube API
- Deployment & Hosting: Vercel
- Version Control: Git

Work Breakdown Structure

Creating a work breakdown structure helped our team break down big, intimidating tasks into smaller, more manageable steps. Additionally, it allowed us to assign members to these smaller tasks, giving us an even better sense of direction on how to tackle development.



Setup Details			
Setup Method of Communication	Team	1	1
Determine Communication & Meeting Schedule	Team	2	2
Research API's and Project Functionality/Scope	Team	5	3
Determine API's and Project Functionality/Scope	Team	2	3
Determine Programming Language	Team	2	1
Determine Development Frameworks	Team	3	3
Open Git Repo	Team	1	1
Complete Setup Details		**16**	**14**
Documentation (1)			
Draft README for Git Repo	Jashan	1	
Assign Work to Team Members	Team	2	2
Create Work Breakdown Structure	Anthony	2	3
Create Application Flow Chart	Clement	2	
Develop Two User Stories per API	Jashan (2) & Dean (4)	6	
Create Prototypes and Wireframes for Primary User Interfaces	Anthony	6	5
Create Project Schedule Timeline	Jashan	3	
Complete Documentation		**22**	
Complete Requirement Gathering			

2.1	**HTML and CSS Development**			
2.1.1	Research React.js Content to Include in HTML Code	Anthony	2	
2.1.2	Design Website Layout, React.js Content, and User Interface with HTML	Anthony	6	
2.1.3	Use CSS to Stylize Interface	Anthony	4	
2.1.4	Ensure Compatability With Different Screen Sizes	Anthony	1	
2.1.5	**Complete HTML and CSS Development**		**13**	
2.2	**React.js Development**			
2.2.1	Create React Components	Jashan	3	
2.2.2	Develop User Dashboard and Routing from Dashboard	Jashan	4	
2.2.3	Create Interactive Practice Problems for Users	Jashan	4	
2.2.4	**Complete React.js Development**		**11**	
2.3	**Integration with Artificial Intelligence API**			
2.3.1	Acquire API Access Credentials and Keys	Clement	1	
2.3.2	Define How AI Assistant Will be Visually Represented	Clement	1	
2.3.3	Design User Prompts and Chatbox Components to Allow:	Clement	1	
2.3.3.1	Solve Code Errors	Clement	3	
2.3.3.2	Help With General Questions	Clement	3	
2.3.3.3	Create Practice Questions and Quizzes	Clement	3	
2.3.4	**Complete Integration with Artificial Intelligence API**		**12**	
2.4	**Integration with Sandbox API**			
2.4.1	Acquire API Access Credentials and Keys	Dean	1	
2.4.2	Set up In-Browser Coding Environments	Dean	3	
2.4.3	Implement Code Validation and Feedback	Dean	3	
2.4.4	**Complete Integration with Sandbox API**		**7**	
2.5	**Integration with YouTube API**			
2.5.1	Acquire API Access Credentials and Keys	Dean	1	
2.5.2	Select Videos to Include on Application	Dean	2	
2.5.3	Embed Selected Videos	Dean	1	
2.5.3.1	Retrieve video metadata, including titles, descriptions, and thumbnails	Anthony	1	
2.5.4	Video Search	Anthony	2	
2.5.5	Allow User to Save Useful Videos	Anthony	3	
2.5.6	**Complete Integration with YouTube API**		**10**	
2.6	Complete Front-End Development			

3.1	**Testing**			
3.1.1	Test Different Aspects of AI-Powered Assistance	Clement	3	
3.1.2	Verify YouTube Video Integration	Dean	3	
3.1.3	Test In-Browser Coding Environments	Jashan	3	
3.1.4	Ensure all Routing Leads to Correct Destinations	Anthony	3	
3.1.5	Get Others to Test Client Side	Team	3	
3.1.6	Debugging	Team	6	
3.1.7	**Complete Testing**		**21**	
3.2	**Documentation (2)**			
3.2.1	Demonstration Video	Team	1	
3.2.2	Description of Tests Associated with Code	Team	1	
3.2.3	Description of the project's CI/CD infrastructure	Team	1	
3.2.4	Project Takeaway	Team	2	
3.2.5	**Complete Documentation (2)**		**5**	
3.3	Complete Final Stretch			

(Please refer to the git repository for a clearer image, as the full WBS is too large for the report)

Project Schedule

Outlining a project schedule in the planning phase is a crucial step in ensuring the success of a project. By creating a schedule, our team has a roadmap to follow, helping us decide where and when to allocate the most of our time and resources, as well as allowing us to track our progress and determine whether or not we're on pace to meet our deadlines.

<u>November 1 - 2</u>
- Setup Method of Communication
- Determine Communication & Meeting Schedule
- Research API's and Project Functionality/Scope
- Determine API's and Project Functionality/Scope

<u>November 3</u>
- Determine Programming Language
- Determine Development Frameworks
- Open Git Repo
- Project Report and Video Presentation #1 due

<u>November 4 - 7</u>
- Draft README for Git Repo
- Assign Work to Team Members
- Create Work Breakdown Structure
- Create Application Flow Chart
- Develop Two User Stories per API
- Create Prototypes and Wireframes for Primary User Interfaces

<u>November 8 - 14</u>
- Create Project Schedule Timeline
- Begin HTML and CSS Development

- Start React.js Development

November 15 - 17
- Finish React.js Development
- Integration with Artificial Intelligence API
- Project Check-in

November 18 - 20
- Complete Integration with Artificial Intelligence API
- Begin Integration with Sandbox API

November 21 - 22
- Complete Integration with Sandbox API
- Begin Integration with YouTube API

November 23 - 24
- Complete Integration with YouTube API
- Start Testing Different Aspects of AI-Powered Assistance
- Verify YouTube Video Integration

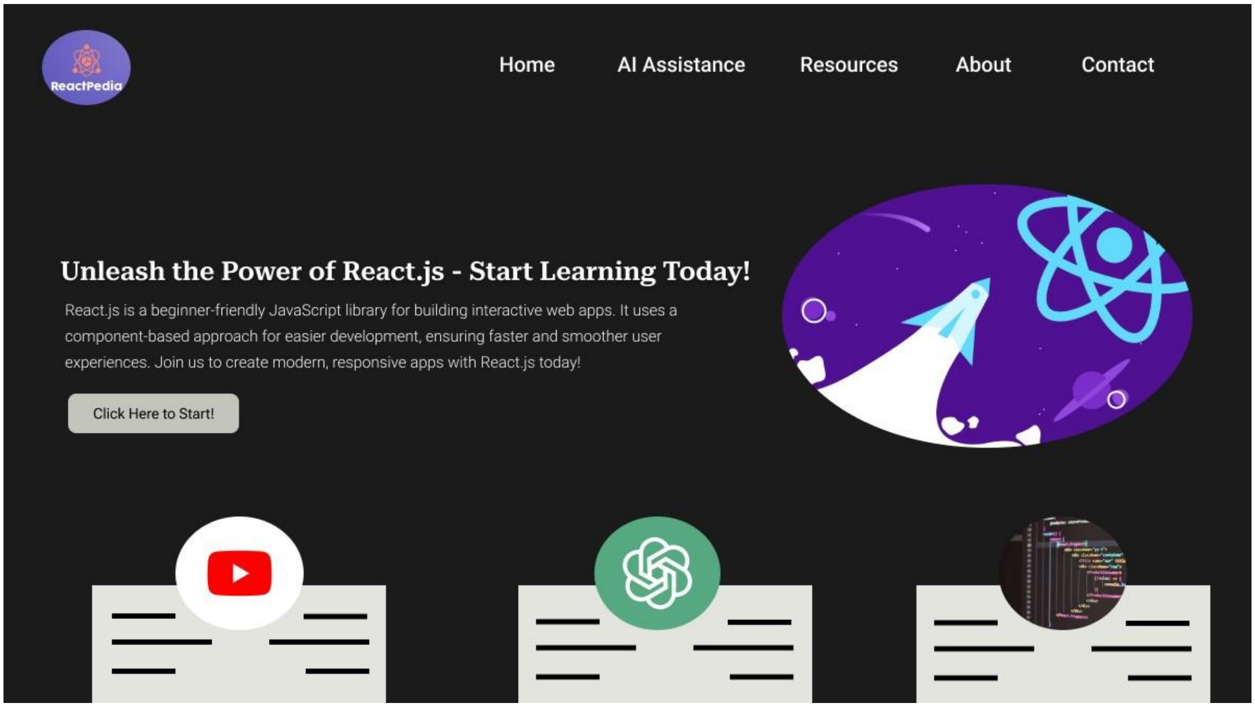
November 25 - 26
- Test In-Browser Coding Environments
- Ensure all Routing Leads to Correct Destinations
- Presentation #2 Slides due

November 27
- Get Others to Test Client Side
- Debugging
- Demonstration Video
- Description of Tests Associated with Code
- Description of the project's CI/CD infrastructure
- Project Takeaway
- In-class Presentation #2

Wireframes and Prototypes

Wireframes and prototypes are instrumental in helping a team bring their idea to life. These diagrams visualize what teams may want to include in their key interfaces and how they might want their web application to appear. This provides clarity prior to the development phase, so when development is actually underway, things run even just a bit more smoothly. Therefore, we created three different wireframes and prototypes for our web application to help us get an idea of how we want to proceed with design and development in the future.

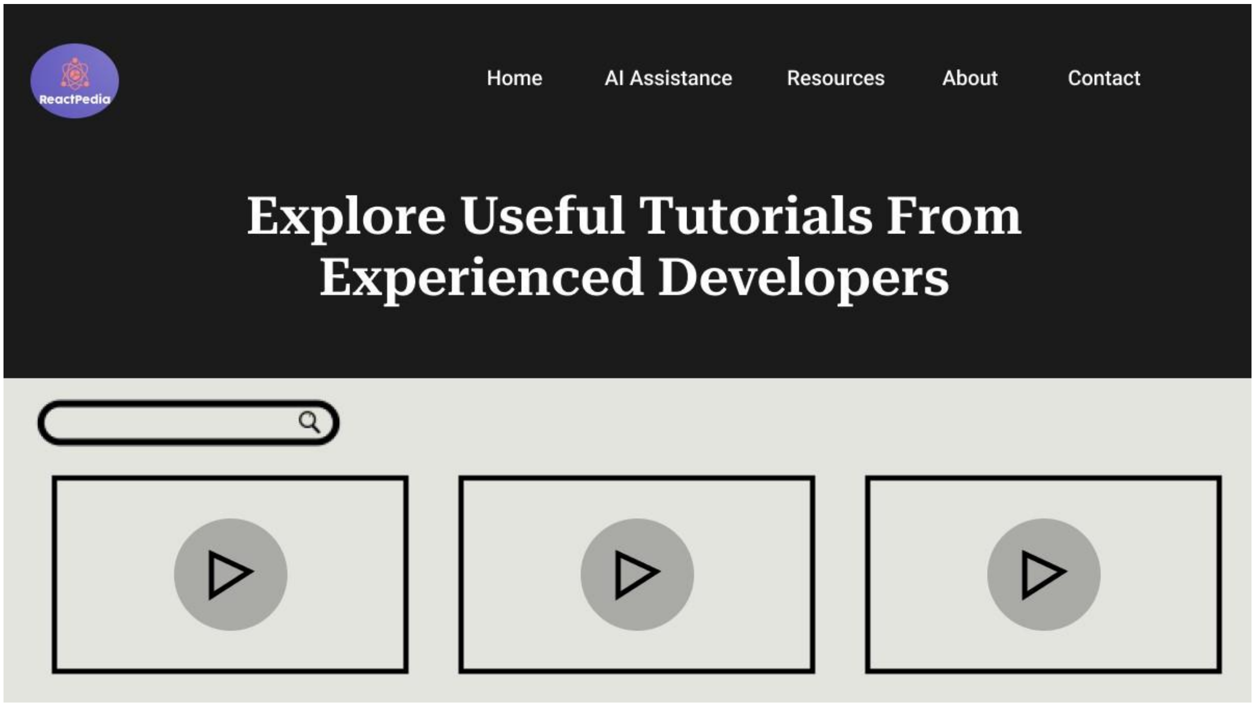
The first image is a prototype for the application’s landing page, where it sets the tone for a minimalistic yet aesthetic appearance for the application. Additionally, it lays out the essential elements, such as the header, which includes important links the users may find useful, highlights a quick summary of the purpose of this web application, and provides users with a call to action on how to get started.



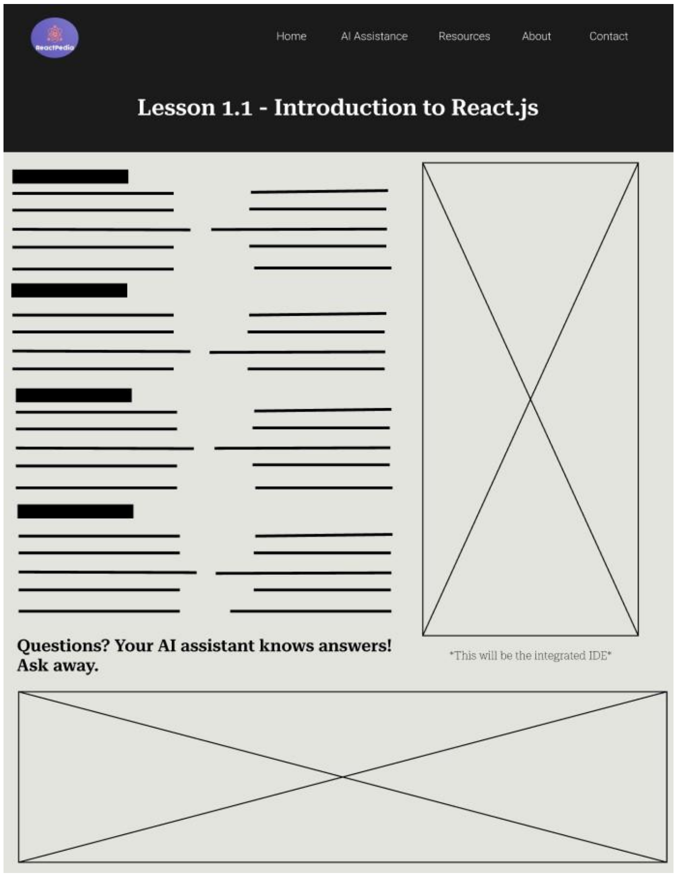
Unleash the Power of React.js - Start Learning Today!

React.js is a beginner-friendly JavaScript library for building interactive web apps. It uses a component-based approach for easier development, ensuring faster and smoother user experiences. Join us to create modern, responsive apps with React.js today!

Next, the wireframe for our video gallery outlines the layout and structure of the gallery. It provides a visual guide to the arrangement of video thumbnails, navigation, and user interactions, ensuring an intuitive user experience. This wireframe enables our team to fine-tune the gallery's design and functionality before beginning the development phase, ultimately contributing to a user-friendly and informative video gallery.



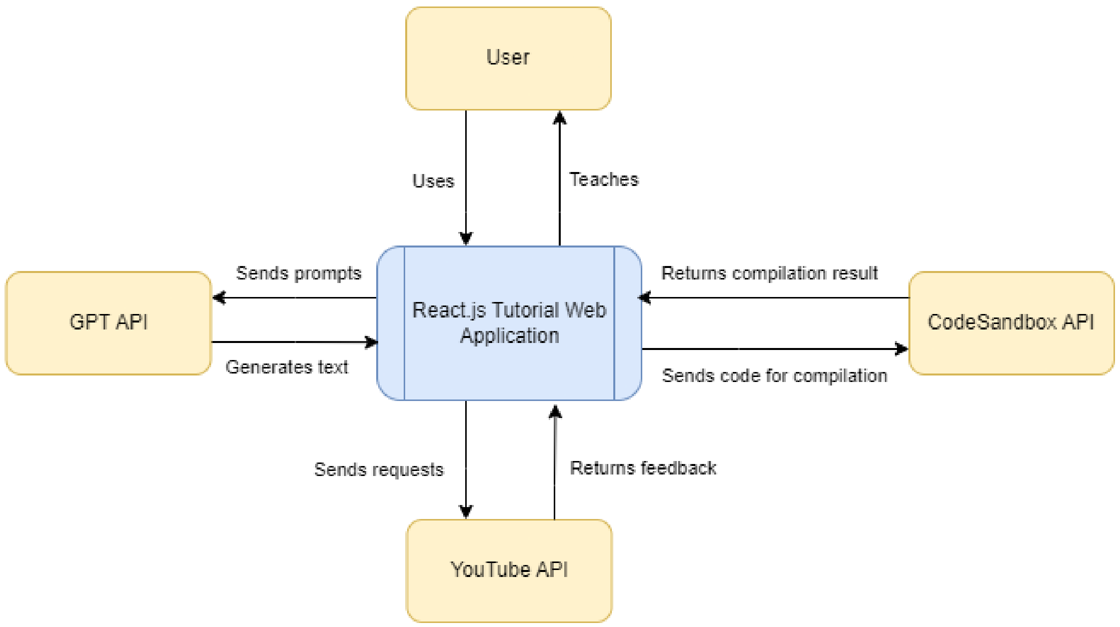
The last wireframe enables us to strategically arrange the positions of content modules, AI-powered support, and the in-browser development environment that leads to a seamless user experience. Through this wireframe, we establish a clear layout that encourages effective learning and development for users.


ReactPedia
Home
AI Assistance
Resources
About
Contact
Lesson 1.1 - Introduction to React.js
Questions? Your AI assistant knows answers! Ask away.
This will be the integrated IDE

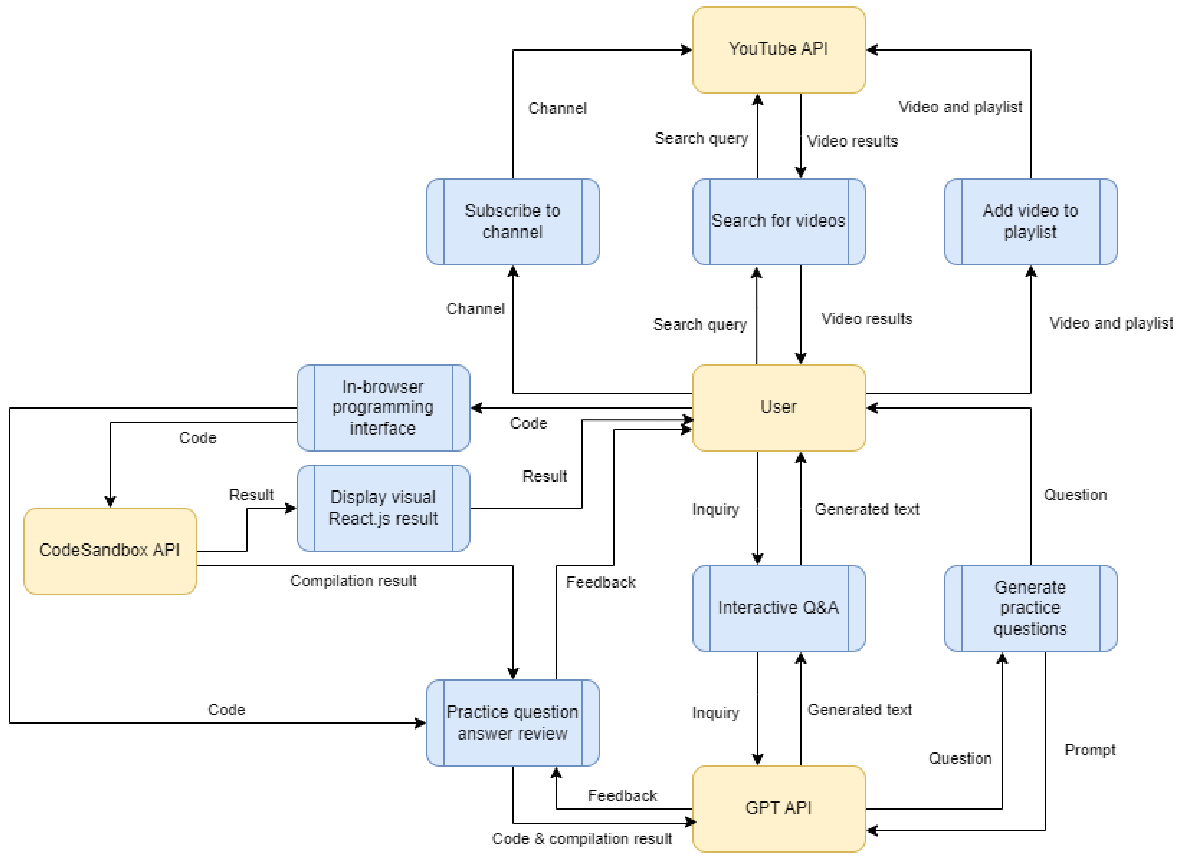
Data Flow Diagrams

Creating data flow diagrams was an important step for the planning of our project, as they provide a visual representation of how data moves within our system. These diagrams helped us gain a clear understanding of data processes, inputs, outputs, and the flow of information throughout the application. In our data flow diagrams, yellow boxes represent external entities, while blue boxes represent processes.

High Level



Detailed


YouTube API
Channel
Video and playlist
Search query
Video results
Subscribe to channel
Search for videos
Add video to playlist
Channel
Search query
Video results
Video and playlist
In-browser programming interface
User
Code
Code
Result
Display visual React.js result
Result
CodeSandbox API
Inquiry
Generated text
Question
Compilation result
Feedback
Interactive Q&A
Generate practice questions
Code
Practice question answer review
Inquiry
Generated text
Question
Prompt
Feedback
GPT API
Code & compilation result