

ReactJS

Lab Assignment

Document Code	25e-BM/HR/HDCV/FSOFT
Version	1.0
Effective Date	19/10/2021

RECORD OF CHANGES

Full	Effective Date	Change Description	Reason	Reviewer	Approver
1	26/Oct/2021	Create new assignment	Create	ThuanNH3	VinhNV

Contents

I. Goals	4
II. Application overview	4
1. Features	4
2. Interface	4
III. Details.....	5
1. “Start game” screen.....	5
2. “In Game” screen	5
3. “End Game” screen.....	7
4. “Review Game” screen.....	7
5. Color codes	8
6. Note	8



CODE:	ASM2- ReactJS-quiz_app
TYPE:	N/A
LOC:	N/A
DURATION:	N/A

I. Goals

- Able to use function components.
- Able to use the two basic hooks, useState and useEffect.
- Capable of analyzing and dividing common components.
- Control the data communication between components.
- Understand conditional rendering.

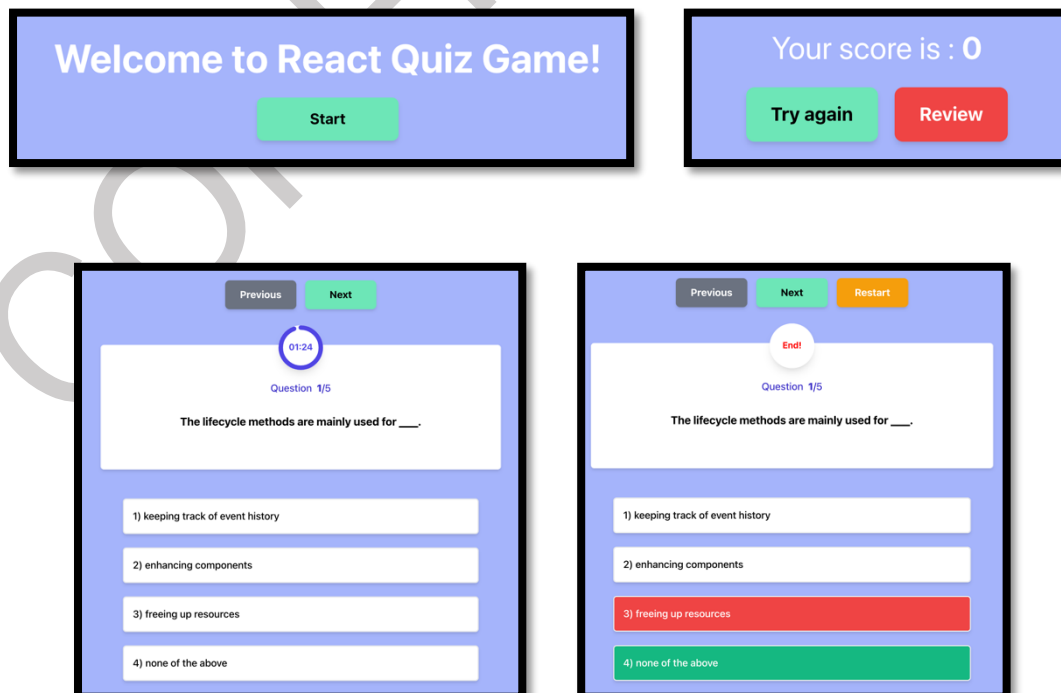
II. Application overview

1. Features:

- Quiz application.
- Features: Next, Previous and Submit.
- Allows users to review the answer after submitted.

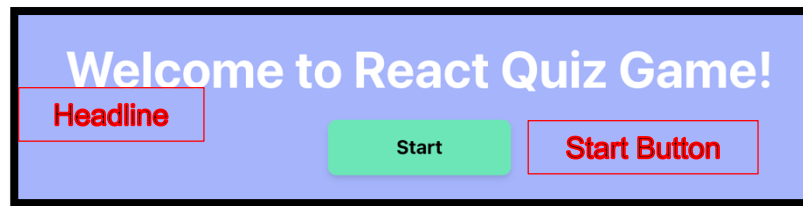
2. Interface:

- The app's interface can be found at the following link:
<https://fa-react-quiz-app.vercel.app>
- The app has 4 screens: Start game, In Game, End game and Review.



III. Details

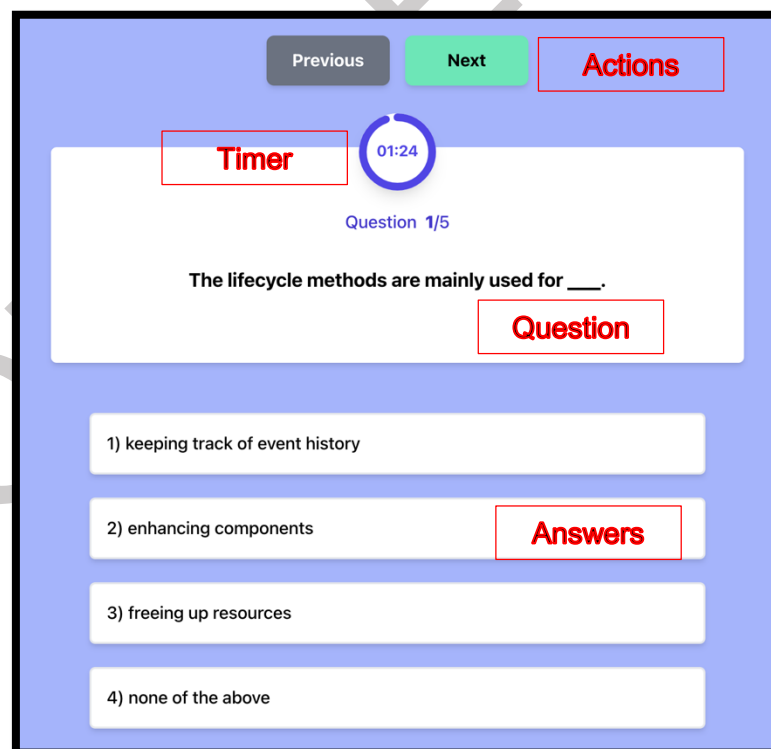
1. Start game screen:



Requirements:

- a) Interface:
 - A title.
 - A start button.
- b) Function:
 - Click on the Start button, go to the In-game screen.

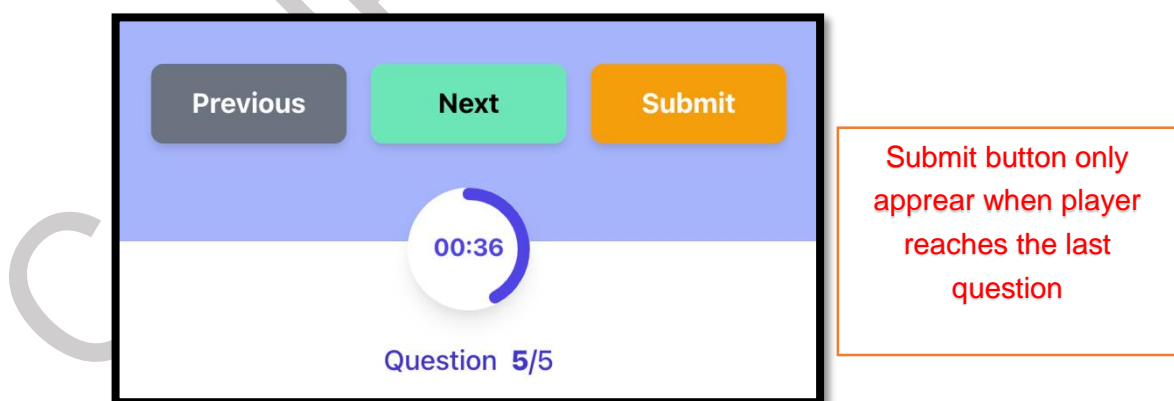
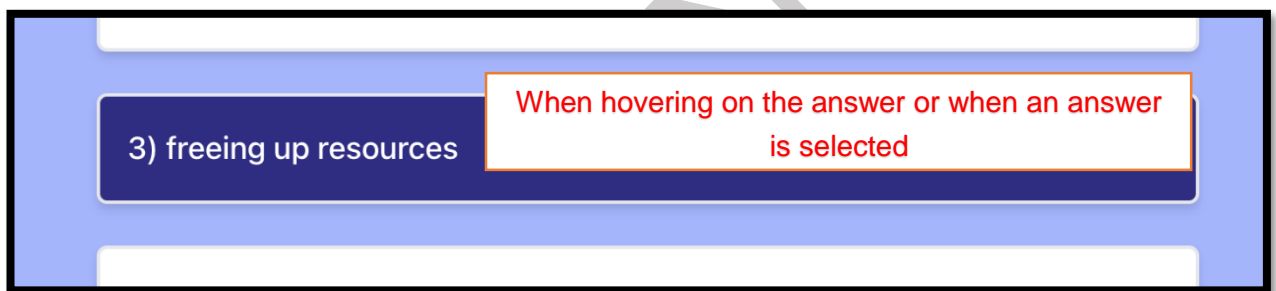
2. In Game Screen:



Requirements:

a) Interface:

- The interface is as design as possible.
- Actions: Previous, Next and Submit.
- When the player is at the last question, the submit button will appear.
- The question will consist of two parts:
 - The current question/total number of questions.
 - The content of the question.
- The answer will be displayed accordingly. Each question has a different number of answers.
- When hovering any answer, highlight it.
- When user select an answer, the selected answer will be highlighted and as same as hovering.
- **Bonus:** Display a Timer - the countdown clock.

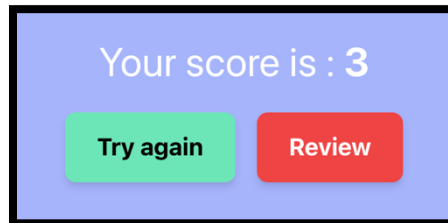


b) Function:

- When the player click on "Previous" or "Next" button, go to the previous or next question.
- The Submit button only appears at the last question. After submitted, go to the "End game" screen.
- When the player changes the question, the current Question/total questions will be updated accordingly.

- When player selects one answer, highlight it.
- **Bonus:** count down (default 1 minute 30seconds). When timeout, automatically go to the end game screen.
- **Bonus:** Disable Previous/Next button when the player is at the first or last question.

3. End Game screen:



Request:

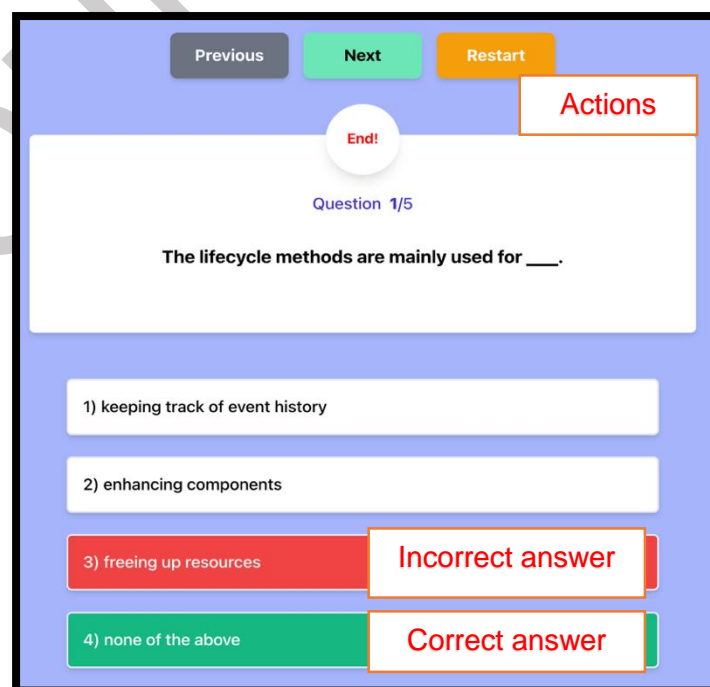
a) Interface:

- Score.
- Try again button.
- Review button.

b) Function:

- Display player's score based on the number of correct answers.
- Clicking the Try again button, go to the Start game screen.
- Clicking the Review button, go to the Review screen.

4. Review Screen Game:



Request:

a) Interface:

- Hide the Submit button then show the Restart button.
- The incorrect answer will be marked as red.
- The correct answer is marked as green.
- **Bonus: Timer display "End!"**

b) Function:

- Clicking Restart button, reset the game and go to the Start screen.
- Display incorrect answers as red, correct answers as green.

5. Color code:

- App background color: #A5B4FC
- Start/Next/Try Again Buttons: #6ED5B7
- Previous Button: #6B7280
- Submit/Restart Buttons: #F59E0B
- Answers hover background, Timer, Current Questions: #312E81
- Incorrect answer: #EF4444
- Correct answer: #10B981

6. Note:

- Only use Function Component.
- Do not use any frameworks, libraries such as Bootstrap, NPM packages....
- You are provided a JSON input question file (questions.json)

```
{
  "id": "1",
  "question_content": "React is mainly used for building ____",
  "answers": [
    {
      "answer_content": "Database",
      "correct": false
    },
    {
      "answer_content": "User Interface",
      "correct": true
    },
    {
      "answer_content": "Design Platform",
      "correct": false
    }
  ]
},
{
  "id": "2",
  "question_content": "The lifecycle methods are mainly used for ____",
  "answers": [
    {
      "answer_content": "keeping track of event history",
      "correct": false
    },
    {
      "answer_content": "enhancing components",
      "correct": false
    },
    {
      "answer_content": "freeing up resources",
      "correct": false
    },
    {
      "answer_content": "none of the above",
      "correct": true
    }
  ]
},
{
  "id": "3",
  "question_content": "____ can be done while multiple elements need to be returned from a component.",
  "answers": [
    {
      "answer_content": "using the <div> tag",
      "correct": true
    },
    {
      "answer_content": "using the <div> tag",
      "correct": false
    },
    {
      "answer_content": "using the <div> tag",
      "correct": false
    },
    {
      "answer_content": "using the <div> tag",
      "correct": false
    }
  ]
}
```

- Usage: Import directly to js or ts files.

------(END & GOODLUCK)-----