Task Graph Scheduling as an Educational Game

Lucia Jiang

Master of Science
Computer Science
School of Informatics
University of Edinburgh
2024

Abstract

This dissertation presents the development and evaluation of a gamified educational application designed to teach task graph scheduling algorithms. The application addresses the limitations of traditional teaching methods by offering an interactive, hands-on approach to learning, with features such as customisable graphs, immediate algorithmic feedback, and gamified challenges.

The application was developed with a focus on user-centred design and usability. User surveys, including the System Usability Scale (SUS) and User Experience Questionnaire (UEQ), indicated that the tool effectively enhances engagement and understanding of scheduling concepts, achieving a SUS score of 77.9. However, feedback highlighted areas for improvement, such as expanding algorithm options, and improving accessibility across devices.

Research Ethics Approval

This project obtained approval from the Informatics Research Ethics committee.
Ethics application number: 335342
Date when approval was obtained: 2023-29-08
The participants' information sheet and a consent form are included in the appendix.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Lucia Jiang*)

Acknowledgements

I would like to thank my supervisor, Murray Cole, for his guidance, wisdom, and feedback throughout the entire process. His expertise and encouragement have been invaluable. I am deeply grateful for his patience and dedication.

I also want to thank all my friends, both in the UK and at home, who have stood by my side throughout this dissertation and my master's journey. Your constant support have been a great boost in overcoming challenges and keeping me motivated. Thank you for listening when I needed to vent and for celebrating my achievements with me.

I want to express my deep gratitude to my family for your unwavering support and understanding throughout my dissertation and the entire master's programme. Your encouragement and unconditional love have been a cornerstone in this academic journey. Thank you for believing in me and for being an endless source of motivation and support at every stage.

Finally, I want to express my appreciation to everyone who contributed to this project by participating in the surveys and questionnaires. Your input was crucial to the success of this study, and it would not have been possible without your valuable feedback.

Table of Contents

1 Introduction **1**

2 Background **3**

2.1 Context 3

2.2 Previous work 5

2.3 Literature review 5

2.3.1 Key definitions 5

2.3.2 Graph Properties 6

2.3.3 Algorithms 7

2.3.4 Gamification 9

3 Conceptual Design **12**

3.1 Design overview 12

3.2 User Interface Design and Educational Features 14

3.3 Justification of Algorithm Choices 16

3.4 Design Considerations and Alternative Approaches 17

4 Implementation **19**

4.1 Technologies, tools and frameworks used 19

4.2 Challenges and solutions 21

5 Application Overview and Functionality **23**

5.1 Home page 23

5.2 Algorithms page 24

5.3 Create customised graph 25

5.4 Users solve problem 26

5.5 Games 26

6 Analysis and Evaluation **28**
6.1 User surveys . . . 28
6.1.1 Demographic questions . . . 29
6.1.2 System Usability Scale (SUS) . . . 29
6.1.3 User Experience Questionnaire (UEQ) . . . 30
6.1.4 Open-Ended Questions . . . 31
6.2 Surveys result and analysis . . . 31
6.2.1 Demographic results . . . 31
6.2.2 System Usability Scale (SUS) Analysis . . . 32
6.2.3 User Experience Questionnaire (UEQ) Analysis . . . 35
6.2.4 Open-Ended Questions Results . . . 36
6.3 Critical analysis . . . 37
6.3.1 Achievement of Original Goals . . . 37
6.3.2 Unmet Objectives . . . 38

7 Conclusions **39**
7.1 Future work . . . 39

Bibliography **41**

A Designs **45**
A.1 Initial sketches . . . 45
A.2 Sitemap . . . 49
A.3 Templates of graphs . . . 50

B Survey questions and answers **53**
B.1 Demographic questions . . . 53
B.2 System Usability Scale (SUS) . . . 57
B.3 User Experience Questionnaire (UEQ) . . . 61
B.4 Open-Ended Questions . . . 63

C Participants' information sheet **68**

D Participants' consent form **72**

Chapter 1

Introduction

Task graph scheduling consists of determining the optimal timing and allocation of tasks to maximise efficiency and resource utilisation. The primary goal of this dissertation is to design and develop an interactive platform where students can engage with the task scheduling problem, gaining hands-on experience with various scheduling algorithms.

The **motivation** for this project stems from the limitations in traditional teaching methods for algorithms, particularly in the domain of Data Structures and Algorithms (DSA). Conventional approaches often lack the visual and interactive components that can significantly aid in understanding complex concepts such as task graph scheduling. We will try to address this gap by incorporating gamification elements into the learning process. This dissertation seeks to achieve the following objectives:

1. **Identify Effective Gamification Strategies:** Explore existing gamification techniques used in teaching DSA concepts and determine which strategies are most suitable for teaching task graph scheduling.

2. **Develop a Gamified Learning Tool:** Create an application that employs these gamification strategies to teach task graph scheduling algorithms.

3. **Evaluate the Effectiveness of the Tool:** Conduct user studies to assess the impact of the gamified application on learners' understanding and engagement with task graph scheduling.

The main **contribution** of this project lies in the creation of an innovative educational tool that combines gamification with comprehensive learning resources tailored to the study of task graph scheduling. The application offers a range of features, including:

- **Educational Tools:** Tutorials and explanations of different scheduling algorithms, allowing users to learn theoretical concepts and apply them in practice.

- **Interactive Learning Experience:** Users can engage with task graph scheduling problems through an interactive platform that allows them to explore various algorithms, revisit steps, and solve the problems themselves.

- **Customisation:** A drag-and-drop interface that enables users to create and customise graph problems, adjusting several parameters, such as the number of processors, node weights, and edge costs.

- **Algorithmic Feedback:** Immediate feedback on user-generated solutions by comparing them to algorithm-generated solutions, helping users identify errors and improve their approaches.

- **Gamification Challenges:** A range of challenges designed to test users' skills, such as solving problems within a time limit or predicting the next steps in a given algorithm.

The result of the dissertation can be accessed by visiting `https://lucia-jiang.github.io/taskGraphScheduling-front/`. **For a walkthrough of the application, see the video without sound and with timestamps of each section of the website, available at** `https://youtu.be/HGe8qkXnfRA`.

The application has obtained good results, with users studies indicating improved engagement and better understanding of task graph scheduling concepts.

The dissertation is structured as follows. It begins with a **Background** chapter, establishing the theoretical foundation, and reviewing relevant literature on the topic. Following this, the **Conceptual Design** chapter outlines the key design principles and decisions that determined the creation of the application, focusing on user-centred design, minimalism and usability. The **Implementation** chapter then delves into the technical execution, detailing the selection and integration of the technologies, tools, and frameworks used to build the application, along with the challenges encountered and solutions implemented. The **Application Overview and Functionality** chapter offers a detailed exploration of the application's features and functionalities. The **Analysis and Evaluation** chapter presents an assessment of the application based on user feedback, including both quantitative data from surveys and qualitative insights from open-ended questions. Finally, the dissertation concludes with a **Conclusion** chapter, which discusses the findings, reflects on the work completed, and offers suggestions for future developments.

Chapter 2

Background

Task graph scheduling is a critical problem in Computer Science, centred on the efficient execution of tasks represented as nodes within a directed acyclic graph (DAG) [18]. Each node in the graph corresponds to a specific task, while the directed edges represent dependencies between these tasks. The goal is to schedule the tasks on a set of processors such that the overall completion time, or makespan [13], is minimised while respecting the task dependencies.

This chapter is structured as follows. We will first establish the context of the dissertation, explaining the basic principles and relevance of task graph scheduling. Next, we will review previous work related to this topic. In the literature review, we will define important terms, study graph properties, and examine the three algorithms selected for our application. Finally, we will analyse common gamification elements used in educational contexts and determine which ones will be used in our application.

2.1 Context

In task graph scheduling, each node in the DAG represents an individual task, assigned a specific weight that denotes the computational load or time required to complete the task. The directed edges between nodes signify dependencies, indicating that a task cannot begin until all its predecessor tasks are completed. The costs associated with these edges reflect the communication delays or data transfer times needed when dependent tasks are assigned to different processors [18]. Notably, if a predecessor task and its dependent task are assigned to the same processor, the communication cost is avoided [13]. The primary objective of task graph scheduling is to allocate tasks to a set of processors or machines in a manner that ensures all tasks are completed in the shortest possible time, while adhering to the dependencies dictated by the edges [23].

It is important to note that in the version of the problem discussed here, **machines cannot be "timesliced"** in the conventional manner used with processors, whether single or multicore. This means that once a processor begins executing a task, it runs to completion without any interruptions. Scheduling algorithms that allow timeslicing, where tasks can be preemptively switched in and out, represent a different class of problems and are not covered in this dissertation.

The principles of task graph scheduling are not confined to theoretical research but extend to numerous practical, real-world scenarios. In parallel and distributed computing environments, efficient task scheduling is paramount for the performance and scalability of applications [26, 28]. The effective execution of parallel tasks is crucial in high-performance computing clusters, multi-core processors, and distributed systems, making task graph scheduling a cornerstone of computational efficiency and productivity across various domains. These include scientific computing, big data processing, and real-time systems, where the timely and coordinated execution of tasks is essential.

Task graph scheduling is classified as an **NP-hard problem** [11], implying that finding the exact optimal solution is computationally infeasible for large instances [31]. This complexity arises from the exponential number of possible ways to assign tasks to processors while considering dependencies and communication costs. Consequently, researchers have developed numerous heuristic and approximation algorithms to provide feasible solutions within a reasonable timeframe. Some of these algorithms will be explored in subsequent sections of this dissertation, and they employ various strategies to balance the trade-offs between computational efficiency and solution optimality.

Among the most prominent strategies are **list scheduling algorithms**, which prioritise tasks based on specific criteria (such as earliest start time or static level) and assign them to processors accordingly. In this dissertation, we focus exclusively on list scheduling algorithms due to their simplicity, ease of implementation, and effectiveness in a wide range of scenarios. Other techniques, such as clustering algorithms, which group tasks to minimise inter-processor communication, and genetic algorithms, which apply evolutionary principles to iteratively refine task assignments, are outside the scope of this work [21].

2.2 Previous work

Despite the extensive research and application of task graph scheduling across various fields, there is a notable gap in the educational tools available to effectively teach this concept. While many Data Structures and Algorithms (DSA) problems have been explored in educational contexts using visualisations and metaphors to enhance understanding [29], task graph scheduling has not received the same level of attention.

In the realm of education, visualisations and interactive tools have proven to be highly effective in helping students grasp complex concepts. For instance, visual tools for Linked Lists and Hash Maps enable learners to see how elements are connected and how operations like insertion, deletion, and searching are performed.

However, task graph scheduling, a topic equally critical in computer science, lacks such dedicated educational tools. The complexity and abstract nature of task graph scheduling present challenges in creating intuitive learning aids. This dissertation aims to bridge this gap by investigating and developing an application that incorporates gamification elements to teach task graph scheduling algorithms.

2.3 Literature review

2.3.1 Key definitions

Before delving into the algorithms for solving the task graph scheduling problem and examining the common gamification elements used in teaching Computer Science algorithms, it is crucial to clarify some foundational definitions [13].

We define a directed acyclic graph as $G(V,E,C,W)$, where:

- V is the set of nodes, where each node $v_i \in V$ represents a task.
- W is the set of computation costs, where $w_i \in W$ indicates the execution time required for task v_i.
- E is the set of communication edges. The directed edge $e_{i,j}$ links nodes v_i and v_j, with v_i as the predecessor and v_j as the successor. In task graph scheduling, v_j cannot begin until v_i is completed and the necessary data is transferred.
- C is the set of communication costs, where each edge $e_{i,j} \in E$ has an associated communication cost $c_{i,j} \in C$.

Additionally, a task without any parent is called an **entry task**, and a task without any child is called an **exit task**. In the context of processors, we assume:

- Each processor is capable of executing a task while simultaneously communicating with other processors.
- Once a processor starts a task, it runs to completion without any interruptions. Upon finishing, it immediately transmits the output data to all children tasks concurrently.

2.3.2 Graph Properties

In this section, we examine the attributes commonly used for assigning priority in list-scheduling algorithms. List-scheduling is an approach where tasks are given a priority, either statically or dynamically, and processors are assigned to tasks based on this priority.

One key attribute is the **Static Level (SL)**, which represents the longest path from the node to the exit node [13]. It is computed recursively by traversing the DAG starting from the exit node as follows:

$$SL(v_i) = w_i + \max_{v_m \in succ(v_i)} \{SL(v_m)\},$$

where $succ(v_i)$ denotes the set of immediate successors of v_i, and $SL(v_{exit}) = w(v_{exit})$. The time complexity of calculating SL is $O(n+e)$, where n is the number of nodes and e the number of edges.

Another important attribute is the **Latest Start Time (LST)**, which measures how long the start time of a node can be delayed without increasing the schedule timing [22]. It is defined as:

$$LST(v_i) = \min_{v_m \in succ(v_i)} \{LST(v_m) - c_{i,m}\} - w_i,$$

where $succ(v_i)$ again represents the immediate successors of v_i, and $LST(v_{exit}) = EST(v_{exit})$. The **Earliest Start Time (EST)** is calculated by traversing the graph downward from the entry node, v_{entry}, as follows:

$$EST(v_i) = \max_{v_m \in pred(v_i)} \{EST(v_m) + w_m + c_{m,i}\},$$

where $pred(v_i)$ represents the set of immediate predecessors of v_i, and $EST(v_{entry}) = 0$.

2.3.3 Algorithms

The primary focus of this dissertation is not to develop new task graph scheduling algorithms but to design an application that helps students understand these algorithms more effectively. The selected algorithms address the fundamental problem of task graph scheduling, where the graph structure and computational costs are arbitrary, communication is considered, duplication is not allowed, a limited number of processors is used, and all processors are connected [18].

Figure 2.1 presents a classification of various versions of the task graph scheduling problem along with the corresponding algorithms. Highlighted within the figure are the three algorithms chosen for our application: Highest Level First with Estimated Time (HLFET), Minimum Communication Time (MCP), and Earliest Task First (ETF). In this section, we will describe these algorithms in detail.

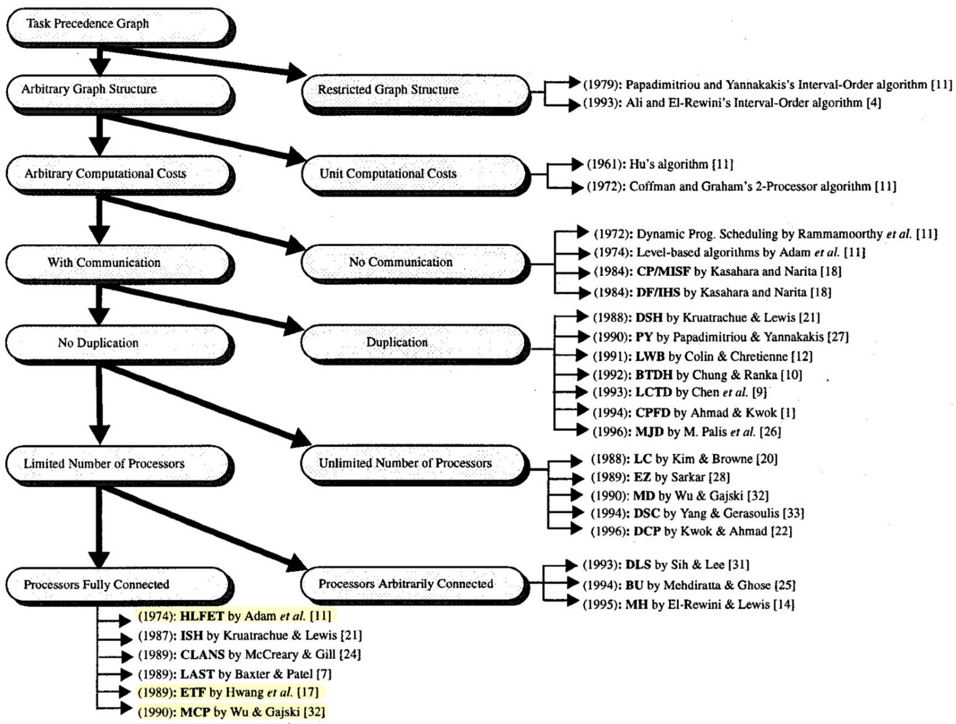

Figure 2.1: Classification of task graph scheduling algorithms [18]

2.3.3.1 Highest Level First with Estimated Time (HLFET)

HLFET assigns task priorities statically, meaning priorities are determined before scheduling [13]. Its complexity is $O(p \cdot n^2)$, where p is the number of processors and n is the number of nodes [22]. The pseudocode for this algorithm is as follows [13, 22]:

1. Calculate the Static Level (SL) for all tasks.
2. Create a list L of all nodes sorted in descending order of SL.
3. Sort the list L consisting of all nodes v_i in descending order of SL.
4. While sorted list L is not empty:
    (a) Dequeue v_i from the sorted list L.
    (b) Compute the earliest execution start time for v_i across all processors.
    (c) Assign the task to the processor that allows the earliest start.

2.3.3.2 Minimum Communication Time (MCP)

MCP assigns priorities statically, by computing the Latest Start Time (LST) for each task. Its complexity is also $O(p \cdot n^2)$, where p is the number of processors and n is the number of nodes [22]. The algorithm operates similarly to HLFET but uses LST to assign priorities [13, 22]:

1. Calculate the Latest Start Time (LST) for all tasks.
2. Create a list L of all nodes sorted in ascending order of LST. Resolve ties by taking the lowest LST value of children nodes.
3. While sorted list L is not empty:
    (a) Dequeue v_i from the sorted list L.
    (b) Compute the earliest execution start time for v_i across all processors.
    (c) Assign the task to the processor that allows the earliest start.

2.3.3.3 Earliest Task First (ETF)

In contrast to the previous two algorithms, Earliest Task First (ETF) computes priorities dynamically, meaning that the priorities are assigned during the scheduling process [13, 15]. The complexity of the algorithm is $O(p \cdot n^3)$, where p is the number of processors and n is the number of nodes. The pseudocode for ETF is as follows [13, 15]:

1. Calculate Static Level (SL) for each task.
2. Initiate a ready nodes list L with only the entry node.
3. While list L's length is smaller than the number of tasks:
   (a) Calculate the earliest possible start time for each node in the ready list on all processors.
   (b) Choose the node-processor pair that allows the earliest start time. If tied, prefer the node with a higher SL.
   (c) Schedule the selected node on the chosen processor.
   (d) Add any newly ready nodes (nodes whose dependencies are satisfied) to the ready list L.

While the original papers explaining these algorithms, such as [15], are very mathematical and symbolic, we have tried to simplify the terminology and instructions here for clarity. Some errors in other papers, like [13], where they confuse between v_{node} and v_{entry} in their pseudocodes, have been addressed. The pseudocodes presented in this dissertation aim to simplify each algorithm without losing its essence, ensuring they remain comprehensible. Additionally, we have minimised mathematical notation to make the algorithms accessible to users who may not have an extensive mathematical background.

2.3.4 Gamification

Gamification involves incorporating game-like features into services to create enjoyable experiences and encourage specific behaviours [14]. While there is limited research specifically focused on gamifying the learning experience of task graph scheduling, numerous studies have explored gamification in other fundamental data structures and algorithms, such as linked lists and hashmaps [10].

Research indicates that a lack of motivation and confidence are significant factors contributing to dropout rates in Computer Science degrees [10]. Integrating game elements into educational platforms can create engaging experiences and improve students' learning outcomes. This section examines common gamification elements [10] and evaluates their suitability for teaching task graph scheduling problems:

- **Points:** Rewards assigned to students for completing specific tasks, providing a sense of accomplishment and progression.
- **Badges:** Recognition given when students achieve particular goals or milestones.
- **Leaderboards:** Rankings that display students' points, fostering a competitive environment that can spur increased effort and engagement.
- **Storylines:** Narratives that add context and motivation, encouraging users to become invested in the plot and strive to progress.
- **Levels:** Dividing the content into stages of increasing difficulty, challenging users as they advance and ensuring a gradual learning curve.
- **Challenges:** Specific tasks that users are required to complete, adding elements of difficulty and accomplishment.
- **Feedback:** Providing guidance and information to prevent users from becoming confused or lost, ensuring they stay on track and understand each step.
- **Metaphors:** Using familiar concepts and game elements to represent abstract ideas, making complex information more relatable and easier to understand [29].

Our goal is to develop an entertaining application that encourages students to learn, without making the game the primary focus. While we should incorporate gamification elements to promote active learning, we must ensure they do not distract from the main focus, which is the learning path.

Thus, we decided **against including storylines or a point system**. Excluding these features allows users to interact more easily with the application, as they do not need to sign up or log in. Studies have shown that elements such as badges and points have minimal impact on students' motivation [27, 6]. Furthermore, excluding these elements simplifies the implementation, as we do not need to integrate a database to track each user's high scores, points, and milestones.

Regarding visualisation, several papers use metaphors such as cars in parking spots to represent linked lists [29]. However, since the task graph scheduling problem is already a graph, we do not need additional metaphors.

Our main inspiration for gamification elements comes from Beckwith et al. [4]. They developed a website where students learned about binary search data structures through simple interactions, such as clicking, as they progressed through the learning material. Students first learned how the tree data structure works, including connections and search functions, and then they could create and modify trees by inserting and deleting values while visualising the tree [27, 4]. The results were very positive, with over 70% of participants stating they enjoyed the game, and nearly half finding the gaming experience more effective than traditional learning methods.

Our game-based website is designed to be a **minimalist**, **fast-reacting application** that introduces users to the task graph scheduling problem and the common algorithms used to find solutions. Further details and a comprehensive evaluation of the user experience will be presented in the following sections.

Chapter 3

Conceptual Design

In this chapter, we delve into the conceptual design of the task graph scheduling application, exploring the key principles and decisions that shaped its development. The design process was informed by user-centred principles, minimalism, and adherence to established usability standards, ensuring that the application is both functional and accessible. We will discuss the overall design choices, the user interface design, the specific features and algorithms implemented, and the gamification elements that enhance the learning experience. Each section will provide detailed insights into the rationale behind the design choices, highlighting how these decisions align with the educational objectives of the application.

We remind the reader that the application can be accessed by visiting `https://lucia-jiang.github.io/taskGraphScheduling-front/`. For a walkthrough of the application, see the video without sound and with timestamps of each section of the website, available at `https://youtu.be/HGe8qkXnfRA`.

3.1 Design overview

The design of our website is crafted with a focus on minimalism, user-centred principles, and adherence to usability standards, as outlined in ISO 9241-11, which will be discussed later. This approach ensures that users can engage with task graph scheduling problems effectively and enjoyably. The application was designed and developed with the following aspects in mind:

Minimalistic Design. Our website embraces a minimalist design philosophy to reduce distractions and enhance user focus. By utilising a limited colour palette, primarily based on Bootstrap's primary blue, we maintain visual clarity and simplicity.

This approach helps users concentrate on their tasks without unnecessary visual clutter.

Consistent Structure. All pages of the website follow a similar structure, as showcased through initial design sketches detailed in Appendix A.1. This consistent structure helps users navigate the website with ease and reduces the learning curve associated with using the platform.

We also aimed to organise each page in the most logical way possible. Figure 3.1 shows our **UX sitemap**, a visual representation that outlines all the key pages of the website. These pages are organised in a hierarchical structure, allowing users to easily navigate and locate the web content [34].

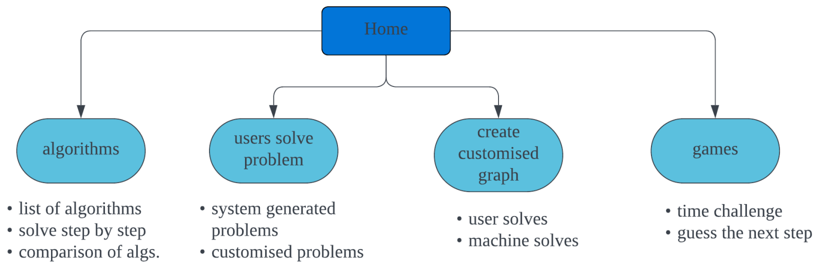

Figure 3.1: Sitemap of the website

User-Centred Design. Our design process is rooted in user-centred design principles, ensuring that the platform is tailored to meet user needs. By adopting the perspective of the user, we make design decisions that simplify interactions and enhance usability [3].

Usability. The design prioritises usability by focusing on ease of learning, effectiveness in task completion, and a pleasant user experience [24]. Features such as interactive learning experiences and algorithmic feedback are designed to support users in achieving their goals accurately and efficiently.

ISO 9241-11 Principles: We adhered to the ISO 9241-11 principles throughout the design process, ensuring that the platform is effective, efficient, and satisfying for users. These principles include:

- **Effectiveness** refers to the accuracy and completeness with which users achieve their goals [16]. We ensure effectiveness by providing interactive learning experiences where users can create and personalise task graphs, solve problems using various algorithms, and receive immediate feedback on their solutions. This supports users in achieving their goals with high accuracy and completeness [16].

- **Efficiency** relates to the resources expended in relation to the accuracy and completeness with which users achieve their goals [16]. The design promotes efficiency by offering a minimalist and fast-reacting platform.

- **Satisfaction** pertains to the comfort and acceptability of the work system to its users and others affected by its use [16]. To enhance satisfaction, we incorporate gamification elements that make learning engaging and enjoyable. Challenges such as time-limited problem-solving and predicting algorithm steps add an element of fun. Additionally, educational tools like tutorials and algorithm explanations ensure a comfortable and supportive learning environment [10].

3.2 User Interface Design and Educational Features

As outlined in Section 2.3.4, the decision was made to omit traditional gamification features like points, badges, leaderboards, storylines, and levels, which are commonly found in other educational applications, such as those discussed by Gari et al. [10]. The primary objective of our platform is to minimise distractions and competition, focusing instead on deepening users' understanding of task graph scheduling algorithms.

To achieve this goal, we concentrated on incorporating elements that directly support learning through engagement and interaction. Our approach includes integrating challenges, providing immediate feedback, and breaking down algorithms into clear, visual steps. These features are designed to enhance user interaction with the application and foster a thorough comprehension of each algorithm.

On the other hand, the user interface design is crafted to enhance usability, accommodate user needs, and provide a seamless interaction experience. These considerations are reflected in the following subsections, where we delve into the key features and design principles that underpin the platform's educational and user experience strategies.

Customisation and Flexibility. The platform offers users the ability to create and personalise task graphs through a drag-and-drop interface. This allows for easy modification of nodes, processors, node weights, and edge costs. Users can also download graphs as images or JSON files, facilitating the saving and reloading of their work for continued progress. The JSON format supports reloading graphs in the customisation section, enabling users to resume their work seamlessly.

Intuitive Interactions and Visual Feedback. The UI is designed to be intuitive, featuring clear labels and tooltips that guide users through each step, including creating and customising task graphs. This design reduces cognitive load, making the platform more accessible and user-friendly. Immediate feedback mechanisms are in place to inform users about the outcomes of their actions, such as algorithmic feedback on problem-solving solutions, helping them learn and adjust their strategies in real-time.

Error Handling and Reversibility. Understanding that mistakes and slips are common in user interactions [8], the platform includes features to make errors easily reversible. Slips, typically more frequent, are accidental errors that the platform mitigates by allowing users to change processor assignments with a refresh button or remove nodes and edges via keyboard commands.

To further reduce the likelihood of errors, we have imposed restrictions on how users can input data such as the number of processors, edge costs, and node weights. For instance, since edge costs and node weights can be floating-point numbers, the application only permits the entry of digits and a single decimal point. Any attempt to enter additional decimal points or non-numeric characters is blocked. Similarly, the number of processors must be a positive integer, so we have implemented a number spinner, where users can select the appropriate amount using - and + buttons.

Algorithm Prediction and Step-by-Step Visualisation. To reinforce users' understanding of the algorithmic processes, the platform includes an algorithm prediction feature. Users are prompted to predict the next step in a given algorithm, engaging them actively in the learning process. Alongside this, the platform provides a step-by-step visualisation of algorithms with detailed explanations, allowing users to comprehend each algorithm's operation in a clear and visual manner.

Engaging Visuals and Educational Challenges. The platform employs engaging visuals and interactive elements to maintain user interest. Animations and visual cues guide users through complex concepts, making them more accessible. Additionally, the platform includes challenge games, such as time-limited problem-solving and algorithm step prediction, to make learning more dynamic and engaging. These challenges encourage users to apply their knowledge quickly and accurately, reinforcing their learning through practice.

Performance Comparison and Customisable Problem Settings. Users have the ability to compare the performance of all algorithms side by side, gaining insights into the efficiency and trade-offs of each algorithm. This comparative analysis is complemented by the platform's customisable problem settings, where users can modify graph structures, node weights, communication costs, and the number of processors before selecting an algorithm to solve the problem.

Immediate Feedback for Enhanced Learning. Immediate feedback is provided throughout the platform, particularly during task assignments. If a user makes an incorrect step, the platform suggests the correct option, aiding in real-time learning and correction.

3.3 Justification of Algorithm Choices

To teach students about task graph scheduling, we selected three fundamental algorithms: HLFET (Highest Level First with Estimated Times), MCP (Modified Critical Path), and ETF (Earliest Time First). Detailed explanations and pseudocode for these algorithms can be found in Section 2.3.3 of the Literature Review.

These algorithms were chosen for their relevance and educational value, particularly because they all fall under the category of list-scheduling algorithms. In list scheduling, tasks are prioritised before being assigned to processors, making it a straightforward and intuitive approach for beginners to understand. Furthermore, these algorithms address the same variant of the task graph scheduling problem, as characterised by Kwok and Ahmad's benchmarking study [18]. This variant includes the following features:

- Arbitrary graph structure and computational costs.
- Consideration of communication costs.
- No allowance for task duplication.
- Use of a limited number of processors.
- Full interconnection of all processors.

Additionally, each algorithm was chosen for its unique contribution to the learning experience:

Highest Level First with Estimated Times (HLFET) prioritises tasks based on their static levels, which reflects their overall impact on the schedule. This approach is particularly effective in scenarios where task priorities can be determined in advance, helping users understand the significance of task hierarchy in scheduling [13].

Modified Critical Path (MCP) focuses on minimising delays by assigning tasks based on their earliest possible start times. This algorithm emphasises the importance of the critical path, making it ideal for understanding how critical tasks influence the overall schedule [13].

Earliest Time First (ETF) dynamically updates task priorities during the scheduling process based on the earliest feasible start times. This adaptability makes ETF particularly useful in scenarios that require real-time adjustments, providing insights into dynamic scheduling environments [13].

3.4 Design Considerations and Alternative Approaches

During the development of the task graph scheduling application, various design options and alternative approaches were considered but ultimately not implemented. This section reflects on these alternatives, providing insight into the decision-making process and the rationale behind the final design choices.

Exclusion of Leaderboards. One of the early ideas was to include leaderboards as part of the application, as discussed in the Literature Review (Section 2.3.4). Leaderboards are often used in gamified applications to enhance user engagement by fostering competition among users. However, after careful consideration, we decided against their inclusion. The primary focus of our application is on education, specifically helping users understand task graph scheduling algorithms. Introducing a competitive element like leaderboards could shift the emphasis away from learning and towards competition, potentially detracting from the educational value of the tool. Our decision reflects a commitment to maintaining the integrity of the learning experience, prioritising comprehension over competition.

Simplification of Gamification Elements. In addition to the selected gamification strategies, we explored the possibility of incorporating more advanced elements such as points, badges, and storylines, which were also reviewed in the Literature Review section. These features are common in gamified systems and serve to incentivise user participation and motivation. However, we chose to simplify the gamification aspects, excluding these elements to avoid distractions from the core educational objectives. Instead, we focused on providing interactive challenges, immediate feedback, and clear visualisations. These elements were considered more aligned with our goal of enhancing the user's understanding of the algorithms, ensuring that the educational content remains the central focus.

Limiting the Number of Algorithms. Initially, we considered including a broader selection of task scheduling algorithms within the application. Offering a wider variety of algorithms could have provided users with more options for exploration and comparison, potentially enriching the educational experience. However, we ultimately decided to limit the number of algorithms included in the final application. This decision was made to maintain clarity and depth in the educational content, allowing us to provide a more detailed exploration of each selected algorithm. By focusing on a smaller set, we ensured that users could gain a deeper understanding of the key concepts without feeling overwhelmed by too many choices.

Avoiding Complex Visualisations. Advanced visualisation techniques, such as interactive 3D graphs or real-time animations of algorithm execution, were also considered during the design phase. These sophisticated visualisations could have offered deeper insights and a more engaging user experience. However, they were ultimately deemed too complex for the scope of this project. The potential for these visualisations to overwhelm users and complicate the development process outweighed their benefits. Instead, we opted for simpler, more functional visual elements, such as 2D graph visualisations and step-by-step animations. This approach ensured that users could easily understand and interact with the material, aligning with our goal of making the application accessible and educational without unnecessary complexity.

Focus on Desktop Optimisation. Another consideration was expanding the application to support a broader range of platforms, such as mobile or tablet devices. While the application is designed to be responsive and functional across various screen sizes, it performs best on desktop computers. After evaluating the potential benefits and challenges, we decided to prioritise a responsive design optimised for desktop use. Supporting additional platforms could have increased the application's accessibility, but this was not deemed as critical as delivering a robust and well-functioning desktop experience. The decision to focus on desktop optimisation reflects our understanding of the target audience's needs and the importance of ensuring a high-quality user experience. This consideration could be revisited in future work as part of ongoing efforts to enhance the application's accessibility and reach.

Chapter 4

Implementation

In this chapter, we delve into the technical aspects of the dissertation, detailing the selection and integration of various technologies, tools, and frameworks that underpin the development of the task graph scheduling application. We explore the rationale behind choosing specific programming languages, libraries, and platforms, and how these choices have contributed to building a robust, interactive, and educational tool. This chapter also addresses the challenges encountered during the implementation process and the solutions devised to overcome them.

4.1 Technologies, tools and frameworks used

To develop a robust and interactive task graph scheduling application, we utilised a combination of Python for backend processing and React.js for frontend development. This choice of technologies allowed us to leverage Python's powerful computational libraries for algorithm processing [7] and React.js's dynamic interface capabilities to create an engaging user experience [19].

Python played a critical role in implementing the task graph scheduling algorithms and calculating essential graph attributes, such as static level (SL), earliest start time (EST), and latest start time (LST). We chose Python due to its extensive libraries, which facilitated the computational tasks and data manipulations necessary for the project. The algorithm steps were computed and stored as arrays of JSON objects, each containing detailed descriptions and explanations. These JSON objects are then accessed via APIs, allowing for seamless interaction between the backend and frontend systems.

FastAPI was selected because it offers automatic interactive API documentation, high-speed execution, and support for asynchronous programming [32], all of which contribute to the responsiveness and efficiency of our application. By sending a PUT request with the graph JSON as a parameter, the API returns an array of JSON objects with detailed step information. The input JSON must include parameters such as `num_processors`, an integer indicating the number of processors, and lists of `nodes` and `edges`, representing the graph's structure.

For graph operations, we employed the **NetworkX** library, a powerful tool for creating, manipulating, and studying complex networks [12]. NetworkX facilitated the computation of graph-related attributes and the visualisation of task dependencies within the graph structure, making it an essential component of our backend.

On the frontend, **React.js** was used to develop the user interface. React is a widely-used JavaScript library known for its component-based architecture [19]. A component in React is a reusable piece of the user interface that can manage its state and props independently [9]. This enables efficient updates and re-rendering by only updating the components affected by user actions, rather than the entire page. This results in a highly responsive and dynamic user experience, aligning with our goal of creating an engaging and interactive educational tool, as discussed in the Gamification section in Background 2.3.4.

Several libraries were integrated into the React ecosystem to enhance functionality. **Bootstrap** was used for styling, providing pre-designed components and responsive layouts that made the interface visually appealing and user-friendly. **ReactFlow** was another key library, enabling the creation of drag-and-drop features for graph visualisations, which allowed users to interact directly with the graphs [33]. ReactFlow also facilitated the transformation of visual graphs into JSON objects, which were then passed to the backend for calculations. **Axios** was used for API calls [25], ensuring efficient communication between the frontend and backend, thereby contributing to a smooth user experience.

Version control for both the frontend and backend was managed using **GitHub**, ensuring an organised development process. The source code is divided into front-end: `https://github.com/lucia-jiang/taskGraphScheduling-front` and back-end: `https://github.com/lucia-jiang/taskGraphScheduling-back`.

For deployment, we utilised GitHub Pages to host the frontend. **GitHub Pages** offers a simple method for publishing web pages directly from a GitHub repository [1], making our application accessible to users worldwide without requiring them to install any additional software or packages. However, since GitHub Pages does not support API services [1], we hosted our APIs on the Koyeb platform. **Koyeb** was chosen for its flexibility, scalability, and seamless integration with other services, providing an ideal solution for deploying serverless applications and APIs. Koyeb also connects with GitHub, simplifying the deployment process and ensuring that updates to the code are automatically reflected in the live application [17].

The complete list of APIs, including the expected parameters and resulting values, can be found in the documentation automatically generated by Koyeb: `https://task-graph-scheduling-lucia-jiang-2e58e4e5.koyeb.app/docs`.

4.2 Challenges and solutions

The development of the task graph scheduling application presented a range of challenges, each requiring thoughtful solutions to ensure the final website was both functional and user-friendly. This section details the key obstacles encountered during the implementation phase and the strategies employed to overcome them, aligning with the overall objectives of the project.

One of the most significant challenges was **managing the display of multiple components** on the algorithm step-by-step page. The initial design aimed to include five distinct elements: pseudocode, graph visualisation, processor assignment, graph properties, and step-by-step explanations. However, fitting all this information onto a single page resulted in an overwhelming and cluttered interface, complicating user navigation. To address this, we implemented dropdown menus and collapsible sections. These features allowed less critical information to remain hidden until needed, thereby reducing visual clutter and enhancing the overall user experience. This approach aligns with our project's focus on creating an intuitive and educational tool, as outlined in the Gamification section.

Another challenge involved the **generation and visualisation of randomised graphs**. Initially, we intended to create fully random graphs to offer diverse user experiences. However, achieving aesthetically pleasing and clear visual representations proved difficult. To mitigate this issue, we limited the number of processors to a range of 2 to 5 and predefined specific x and y coordinates for node positions to maintain a structured layout. Instead of generating completely random graphs, we adopted a "pseudo-random" approach. This involved developing 25 graph templates with varying node counts and structures. When a graph is generated, one of these templates is randomly selected, with node weights, edge costs, and the number of processors adjusted accordingly. Examples of these templates are provided in Appendix A.3, reinforcing the project's emphasis on user-centred design.

Ensuring the application was **responsive across different screen sizes** also posed a significant challenge. While the app was primarily designed for desktop use, we recognised the importance of making it accessible on various devices. Although it performs optimally on computers, we incorporated responsive design principles to ensure the application remains functional on smaller screens as well. This is crucial for broadening the accessibility of the tool, reflecting the project's aim to cater to a wide audience. Figures 4.1 and 4.2 demonstrate the app's appearance on different screen sizes, illustrating our commitment to maintaining usability across platforms.

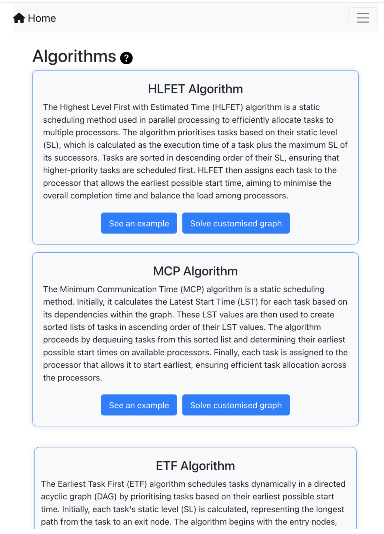

Figure 4.1: App in a smaller screen

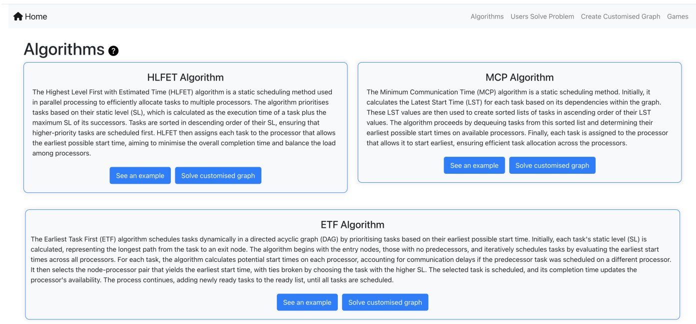

Figure 4.2: App in full screen

Finally, integrating the API on the cloud presented its own set of challenges. Initially, we considered using Heroku, a popular platform for hosting applications. However, **Heroku's recent transition to a paid model** made it less viable for our needs, particularly given budget constraints. After evaluating several alternatives, we selected Koyeb due to its cost-effectiveness and suitability for our project requirements. Koyeb provides a flexible and scalable solution for deploying serverless applications and APIs, and its seamless integration with GitHub further streamlined our deployment process [17].

Chapter 5

Application Overview and Functionality

This chapter provides a comprehensive examination of the task graph scheduling application, detailing its features and functionalities. We will systematically explore each section of the application, explaining the design choices and operational aspects that supports its structure. The accompanying sitemap, illustrated in Figure 3.1, serves as a guide to the application's layout and navigational elements.

We remind the reader that the application can be accessed by visiting `https://lucia-jiang.github.io/taskGraphScheduling-front/`. For a walkthrough of the application, see the video without sound and with timestamps of each section of the website, available at `https://youtu.be/HGe8qkXnfRA`.

5.1 Home page

The home page introduces users to the task graph scheduling problem by outlining its fundamental principles and objectives.

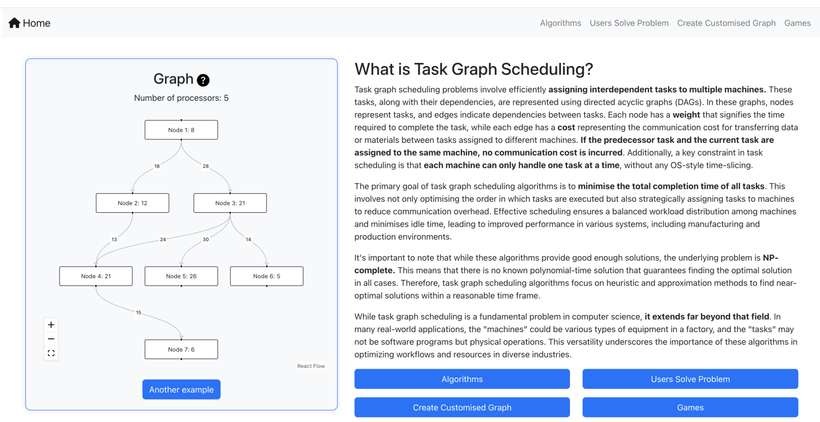

Figure 5.1: Home page

It serves as the primary entry point for navigating the application. From this page, users can access the four main sections of the application, as depicted in Figure 3.1, or utilise the navigation bar present on all pages. The “Another example” button, situated at the left, allows users to view a different task graph scheduling problem, providing varied examples.

5.2 Algorithms page

The Algorithms section offers detailed descriptions of three distinct scheduling algorithms. Users have the option to either independently solve a problem or view the resolution of a randomly generated problem through a step-by-step visualisation. If users select the step-by-step view, the webpage will generate a random graph and present it across five sections.

Pseudocode displays the pseudocode for the selected algorithm, with collapsible sections for users familiar with the algorithm. **Graph Display** visualises the task graph with interactive controls for zooming and panning. Users can also download the graph as an image or as a JSON file. **Graph Attributes Display** provides a table of necessary attributes for the algorithm (e.g., SL for HLFET, EST, and LST for MCP), with steps for calculating these values, initially collapsed. **Steps List** presents a sequential list of the algorithm’s steps, allowing users to navigate through them. Finally, **Processor Assignment Table** updates with each step to show the final assignments of tasks to processors, including start and end times.

Additionally, users can create or upload their own graphs and select an algorithm to solve it step by step. Details on customising a graph are provided in Section 5.3.

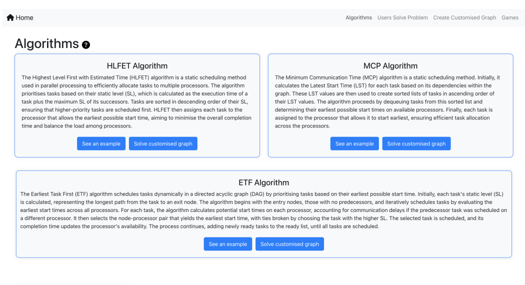

Figure 5.2: Algorithm selection

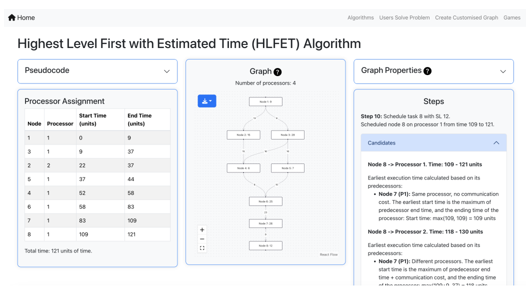

Figure 5.3: Show steps

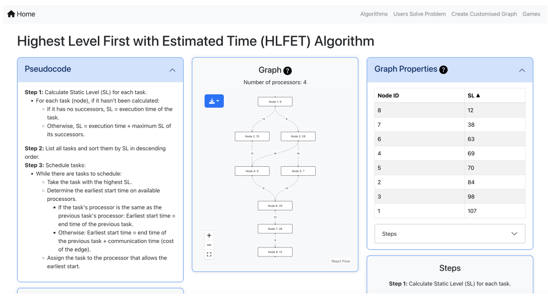

Figure 5.4: Pseudocode and graph attributes display

5.3 Create customised graph

The application allows users to create customised graphs for scheduling problems. Users can input node weights and edge costs through numeric fields and use an input spinner to select the number of processors, ensuring that only integer values are entered. Nodes can be dragged from the right panel into the central workspace, and edges are created by clicking and dragging between nodes. Nodes and edges can also be deleted by pressing the delete key on the keyboard.

After creating a custom graph, users must select the algorithm they wish to apply to solve the problem. It's important to note that the file upload component will display in the language of the user's browser. For example, in the provided screenshot, this component appears in Spanish because the browser is set to Spanish. However, for users with their systems set to English, the component will display in English.

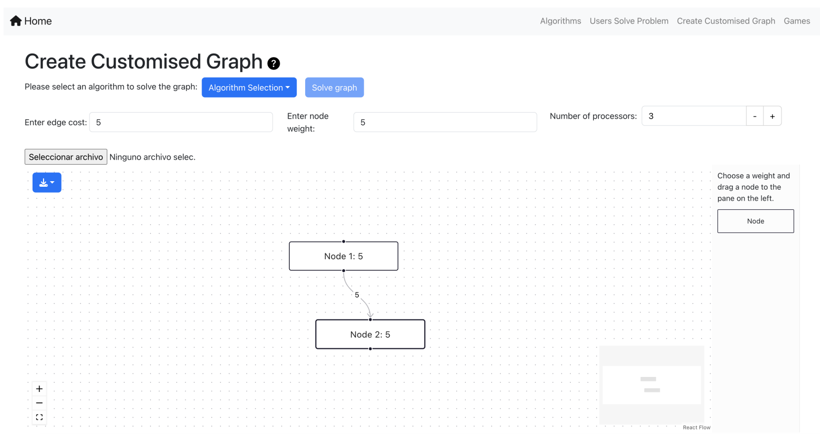

Figure 5.5: Customise graph page

5.4 Users solve problem

This section of the application enables users to solve a randomly generated scheduling problem. Users can refresh the page to generate a new graph. To assign tasks to processors, users select a node, which displays a list of available processors. The "Assignment Details" table records each assignment, including start and end times.

Nodes are colour-coded to reflect their status: green for assigned nodes, blue for the selected node, white for nodes ready to be assigned, and grey for unavailable nodes. Importantly, a node can only be selected if all its predecessor nodes have already been assigned to a processor, adhering to the principle that a task cannot be completed until all dependent tasks are finished. Users can restart the assignment process by clicking "Refresh choices", ensuring that the task scheduling adheres to the correct precedence constraints.

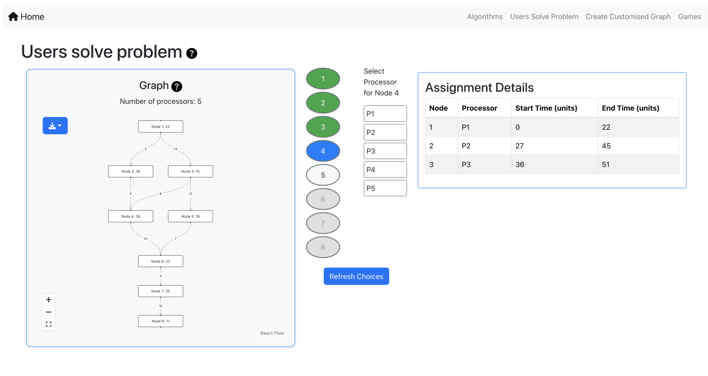

Figure 5.6: Users assigns processors to tasks interface

5.5 Games

The final section of the application includes two interactive challenges designed to enhance users' mastery of task graph scheduling:

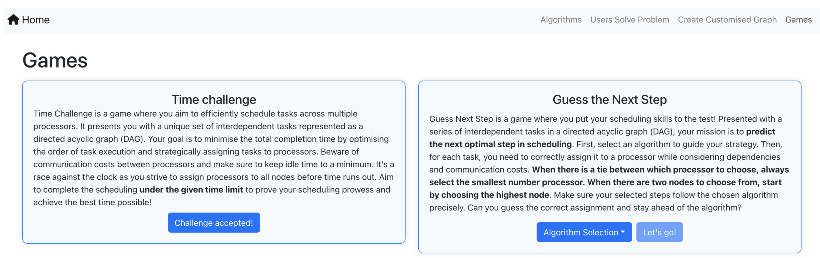

Figure 5.7: Challenge selection

Time Challenge presents a random graph that has been pre-solved using all three algorithms. Users must solve the graph by assigning tasks to processors within an optimal time plus a 10-second threshold, and within a 5-minute real-time limit. After completing the assignment or when time expires, users receive feedback on their performance and can choose to replay the problem, generate a new one, or review the completion times of each algorithm. Additionally, users can view specific assignments and the step-by-step process for each algorithm.

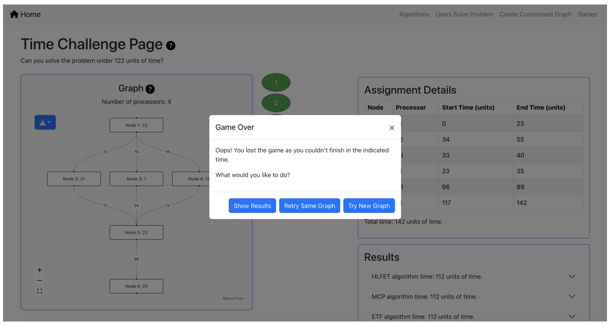

Figure 5.8: Time challenge interface

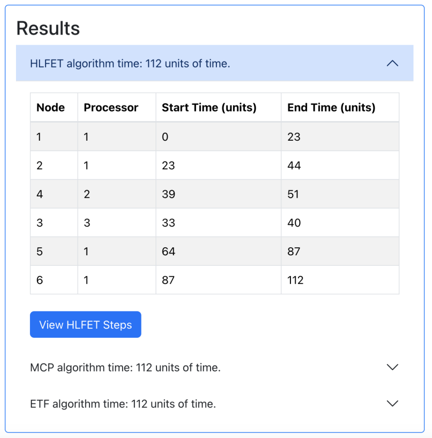

Figure 5.9: Results

In the second challenge, **Guess the Next Step**, users select an algorithm and assign tasks to processors in the order and pairs specified by the selected algorithm. As mentioned in Section 3.1, feedback is provided at each step to indicate correctness, with incorrect assignments corrected and explained.

Both challenges are designed to maintain consistency with the "Problem Solving" page, ensuring a cohesive and minimalistic user experience throughout the application.

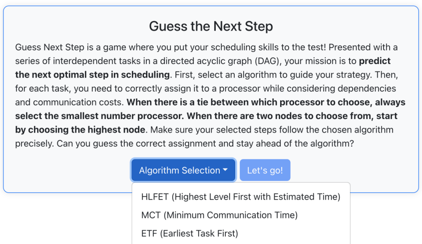

Figure 5.10: Algorithm Selection

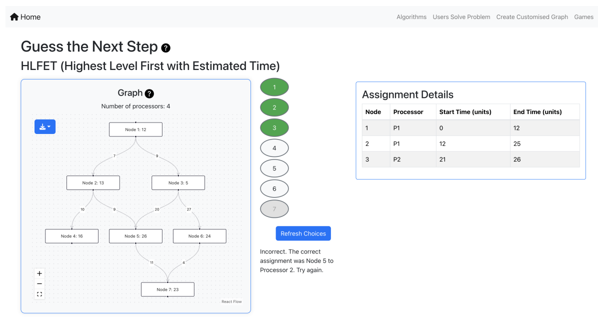

Figure 5.11: Guess the next step interface

Chapter 6

Analysis and Evaluation

In this chapter, we conduct a detailed analysis and evaluation of the application based on user feedback. We begin by examining the results from a structured user survey, which includes demographic questions, System Usability Scale (SUS) assessments, and User Experience Questionnaire (UEQ) evaluations. We will also explore insights gathered from open-ended questions to provide a comprehensive understanding of user experiences and opinions. Finally, we offer a critical analysis of the application to assess whether the initial objectives have been successfully met and identify any areas for further development.

6.1 User surveys

To evaluate the effectiveness of our application, we designed an user survey that aimed to gather both qualitative and quantitative data. The survey was structured into four distinct categories: demographic questions, System Usability Scale (SUS) questions, User Experience Questionnaire (UEQ) items, and optional open-ended questions for additional comments and suggestions. In the following subsections, we will present and justify each set of questions, as well as explain the methodologies behind the SUS and UEQ assessments. Ethical considerations have been addressed and are discussed at the beginning of this dissertation.

The survey was distributed to 25 participants via Google Forms, and each section of the survey was introduced with a brief explanation to guide the respondents. The complete survey, including all questions and response options, is provided in the Appendix B.

6.1.1 Demographic questions

The demographic questions were included to gather background information about the participants, which would enable us to contextualise the survey results. Understanding the demographic profile of respondents allows us to identify patterns or trends in the data, such as whether certain groups find the application more useful or user-friendly than others.

All questions in this section were multiple-choice, designed to elicit clear, categorical responses. For example, one question offered five levels of familiarity with Data Structures and Algorithms, while other questions provided a binary Yes/No response. The diversity in question types and response options was intended to capture a nuanced understanding of the participants' backgrounds and experiences, which could then be correlated with their feedback on the application.

Questions 1-4 help us gauge the participants' existing knowledge and experience with core concepts relevant to the application. Understanding the baseline familiarity allows us to assess whether the application is accessible to users with varying levels of expertise. For instance, users with a strong background in Computer Science might engage differently with the application compared to those with less experience.

Questions 5-7 aimed to assess the participants' general technological competence and their experience with similar tools. These questions provided insight into the broader context of the participants' technological and educational backgrounds. Higher educational attainment and computer proficiency are likely to correlate with more nuanced feedback regarding the application's usability. Additionally, previous experience with educational games or simulations might affect how intuitively users interact with the application, potentially influencing their overall user experience.

Last two questions sought to uncover participants' preferred learning styles and methods for acquiring new technical knowledge. For example, users who prefer hands-on learning might find interactive elements particularly beneficial, while those who favour traditional methods like reading might have different expectations.

6.1.2 System Usability Scale (SUS)

The System Usability Scale (SUS) is a widely recognised tool for assessing the usability of a system or application. It consists of 10 questions, each framed as a statement to which users respond on a 5-point Likert scale [20], ranging from "Strongly Disagree" to "Strongly Agree". The SUS is designed to provide a quick, reliable measure of usability, with questions alternating between positive and negative phrasing to prevent response bias [5].

The SUS score is determined by converting responses into a single score on a scale from 0 to 100. For each participant's answer, we process the scores differently depending on the type of question: for positive statements (odd-numbered questions), we subtract 1 from the response, while for negative statements (even-numbered questions), we subtract the response from 5. The sum of these adjusted scores is then multiplied by 2.5 to convert it to a 0-100 scale [5]. The overall SUS score is the average of all participants' scores. Typically, a score above 68 is considered to indicate good usability [30], though individual items should also be analysed to identify specific strengths and weaknesses of the application.

Our application received a **SUS score of 77.9**, indicating that it has above-average usability. This score suggests that users found the application relatively easy to use and navigate, with most of them having a positive experience. However, while the overall score is promising, it is important to examine the individual question scores to identify specific areas where the application excels and where there may be room for improvement. In the following section, we will delve deeper into these individual question scores to provide a more detailed analysis of user feedback.

6.1.3 User Experience Questionnaire (UEQ)

The User Experience Questionnaire (UEQ) is another established method for evaluating a user's interaction with an application, focusing on both classical usability aspects (efficiency, perspicuity, dependability) and user experience aspects (stimulation, novelty) [2]. The UEQ consists of 26 predefined items that respondents rate on a 6-point semantic differential scale, where each item is presented as a pair of contrasting adjectives (e.g., "annoying/enjoyable", "creative/dull") [2]. The complete set of questions can be found in Appendix B.

Firstly, the **Attractiveness** dimension evaluates the overall appeal of the application, reflecting how visually and aesthetically pleasing the users find it [2]. **Perspicuity** dimension measures how easy it is for users to understand and learn to use the application, highlighting its clarity and user-friendliness [2]. The **Efficiency** dimension assesses how quickly users can complete tasks within the application [2]. **Dependability** examines the reliability and consistency of the application, ensuring that it performs as expected without errors [2]. **Stimulation** dimension gauges the degree to which the application is engaging and motivating, indicating how well it maintains user interest and enthusiasm [2]. Lastly, the **Novelty** dimension looks at the innovation and creativity of the application's design, assessing how it introduces new and unique features [2].

These dimensions provide a comprehensive overview of both the usability and overall experience of using the application. The results from the UEQ will be analysed in the subsequent section, offering insights into how well the application meets user expectations across these varied aspects.

6.1.4 Open-Ended Questions

In addition to the structured questions, we included a series of optional open-ended questions to gather more detailed feedback. These questions invited participants to share their thoughts on what they liked most about the application, what they disliked, and any suggestions for improvement.

The open-ended nature of these questions was intended to capture insights that might not be fully addressed by the structured questions. This feedback is invaluable for understanding user needs and preferences, which can guide future iterations of the application. In the next sections, we will discuss the results from these open-ended responses in greater detail.

6.2 Surveys result and analysis

6.2.1 Demographic results

The majority of participants (over 85%) fell within the 18-24 age range, with a smaller subset (approximately 15%) in the 25-34 age range. Regarding educational background, the majority of participants (approximately 55%) were currently studying Computer Science or had graduated from a related degree programme. However, a significant portion of respondents (45%) did not have a background in Computer Science. This intentional selection of participants with varying levels of technical expertise was aimed at exploring differences in user experience between those with and without a Computer Science background, providing a broader perspective on the application's usability across different user profiles.

In terms of familiarity with key concepts, such as Data Structures and Algorithms, and graphs with nodes and weights, there was a wide range of responses. About 35% of participants reported being moderately or very familiar with these concepts, while the remaining 65% were either slightly familiar or completely unfamiliar. Educational backgrounds were diverse, with most participants holding undergraduate degrees and a smaller group having postgraduate qualifications.

Computer proficiency was generally high, with most users identifying as advanced or expert, though a few were beginners, indicating that the application should remain user-friendly across different skill levels. Engagement with educational games was low, with most users engaging only occasionally or rarely. This suggests the application might introduce some users to educational games, potentially influencing their learning preferences. In terms of learning styles, the group was diverse, with visual and kinesthetic being the most common preferences.

This results will help us understand later the results of SUS, UEQ and open-ended questions.

6.2.2 System Usability Scale (SUS) Analysis

In this section, we analyse the results of the System Usability Scale (SUS) survey, which was designed to assess various aspects of the application's usability. The survey consisted of ten questions, each targeting a specific usability factor. Figure 6.1 presents the results, where each row corresponds to a question, and the cells indicate the number of participants who rated that question on a scale of 1 to 5. The shading in each cell highlights the frequency of each rating, with greener cells representing higher response rates. Based on the SUS scoring methodology outlined in Section 6.1.2, the overall SUS score of our application is 77.9.

	1	2	3	4	5
1	0	0	6	6	13
2	11	10	4	0	0
3	0	0	1	8	16
4	10	6	3	2	4
5	0	0	1	8	16
6	17	4	2	1	1
7	0	1	5	11	8
8	9	7	4	2	3
9	0	3	4	7	11
10	11	3	3	6	2

Figure 6.1: SUS questionnaire results

The first question, **"I think that I would like to use this application regularly if I needed to learn about task graph scheduling algorithms"**, was designed to assess the **likelihood of regular use**. The results were mainly positive, with over 75% of participants indicating that they would use the application regularly. This suggests that the application meets the needs of its target audience, providing a resource that users find valuable enough to return to repeatedly. However, the few neutral responses might be linked to the participants' demographic diversity. Non-CS participants, who may not foresee a regular need for the application, might find the subject matter less relevant.

The second question, **"I found the application unnecessarily complex for understanding task graph scheduling concepts"**, aimed to assess the perceived **complexity** of the application. Encouragingly, more than 80% of participants disagreed with the notion that the application was overly complex. This result is significant because it indicates that the design and functionality effectively simplified complex ideas. Nonetheless, some participants, particularly those with less technical background or lower familiarity with concepts like graphs, found certain aspects challenging. This aligns with open-ended feedback highlighting difficulties related to technical jargon and complex functionalities.

The third question, **"I thought the application was easy to navigate and use"**, focused on **ease of navigation**. The responses were overwhelmingly positive, with nearly all participants agreeing that the application was easy to navigate and use. This indicates that the interface design, layout, and interactive elements are well-executed, enabling users to focus on learning without being hindered by usability issues. The strong result here reflects the success of the design choices made to ensure the application was accessible and intuitive.

The fourth question, **"I think that I would need the support of a technical person to use this application effectively"**, examined the perceived need for **external assistance**. The responses were mixed: approximately 60% of participants felt they could use the application without technical support, while nearly half expressed concerns about needing assistance. This result is particularly relevant given the demographic diversity, with non-CS participants expressing more concerns about needing support. To address this issue, incorporating more detailed tutorials, onboarding processes, or even a guided walkthrough could help reduce these concerns and improve user confidence.

The fifth question, **"I found that the various features and functions of the application were well integrated"**, assessed the **integration of features** within the application. Integration is crucial for providing a cohesive user experience, where different functionalities work together seamlessly to support learning objectives. The feedback for this question was universally positive, with all participants acknowledging that the features, such as customising graphs, visualising steps, and engaging in games, were well integrated. Furthermore, over 60% of participants strongly agreed with this statement. This positive response was also reflected in the open-ended questions, where users complimented the seamless inclusion of features.

The sixth question, **"I thought there was too much inconsistency in the application's design and functionality"**, sought to identify any **inconsistencies** in the application's design. More than 80% of participants disagreed with the statement, indicating that the majority did not perceive significant inconsistencies. However, open-ended feedback mentioned minor inconsistencies or areas for improvement, such as some confusing button placements.

The seventh question, **"I would imagine that most people would learn to use this application very quickly"**, addressed the **ease of learning**, which is crucial for user adoption, especially for educational tools. The results were very positive, with more than 75% of participants stating that they considered the application easy to learn. This suggests that the design of the application effectively supports quick onboarding.

The eighth question, **"I found the application very cumbersome to use for learning task graph scheduling algorithms"**, and the ninth question, **"I felt very confident using the application to understand task graph scheduling"**, explored the **complexity of use** and **user confidence**. These questions received the most mixed responses. While a majority found the application manageable and felt confident in using it, a notable portion of users expressed some difficulty. Open-ended feedback indicated that users with less prior knowledge found the application more challenging, which may contribute to the mixed responses.

Finally, the tenth question, **"I needed to learn a lot of things before I could get going with the application"**, focused on the amount of **prior knowledge required** to use the application effectively. Around 40% of participants felt that they needed more background knowledge to fully understand the application. This group primarily consisted of non-computer science (CS) participants who lacked familiarity with graphs and task graph scheduling problems. This result was expected, as the application deals with specialised content that requires some foundational knowledge. Conversely, participants with a CS background and prior knowledge of graphs found the application sufficiently comprehensive for learning about the algorithms. This split in responses highlights the importance of considering the target audience's prior knowledge when designing educational tools.

Overall, the SUS results were largely positive, indicating that the application is user-friendly and effective for its intended audience. However, the mixed responses, particularly from non-CS participants, suggest that further refinements could enhance the application's accessibility and usability for a broader range of users.

6.2.3 User Experience Questionnaire (UEQ) Analysis

Figure 6.2 presents the results of a User Experience Questionnaire (UEQ) that evaluates six key dimensions of the system: Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, and Novelty. Overall, the results indicate a generally positive user experience, with most dimensions receiving high average scores. However, there are variations in user perceptions across these dimensions, with some areas showing room for improvement.

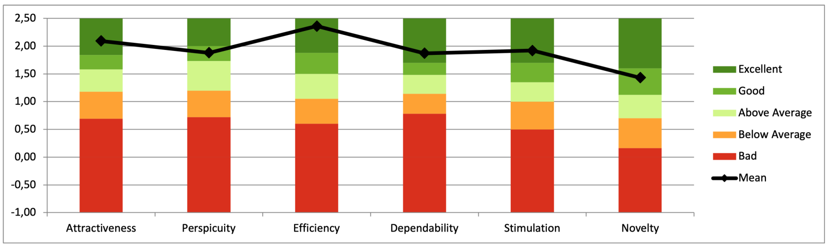

Figure 6.2: UEQ results

The dimension of **"Efficiency"** received the highest ratings, with most users finding the system effective and responsive. This aligns with the positive feedback from SUS, particularly the question regarding the ease of navigation, where nearly all participants agreed that the application was easy to use. The high efficiency score also reflects the open-ended feedback, where users praised the application's speed and responsiveness. This effectiveness is particularly appreciated by users with a Computer Science background, who likely value the system's ability to streamline complex tasks and calculations.

The **"Attractiveness"** and **"Perspicuity"** dimensions both received high ratings, indicating that users found the system visually appealing and easy to understand. This is consistent with SUS results, especially the question about the integration of features, which received strong positive feedback. The demographic data supports these findings, as participants in the 18-24 age range, who are typically more accustomed to engaging digital interfaces, likely responded positively to the application's design and usability.

"Stimulation" and **"Dependability"** received close ratings. While users find the application engaging and motivating, there are occasional concerns about its reliability.

This mixed feedback aligns with SUS results, particularly in questions related to the perceived need for technical support. The varied responses, especially from non-CS participants, indicate that while the application is stimulating, some users may find it challenging or inconsistent in certain areas.

Finally, **"Novelty"** dimension received the lowest ratings, although still positive. This indicates that while the system includes some innovative features, it may not be perceived as particularly groundbreaking. The feedback suggests that while the application is functional, it may not offer a significantly novel approach to teach algorithms compared to other tools.

6.2.4 Open-Ended Questions Results

Throughout the analysis of the System Usability Scale (SUS) and User Experience Questionnaire (UEQ), insights from the open-ended questions have been highlighted. This section further explores some of the most significant feedback provided by participants. The complete set of responses is available in the Appendix B.

6.2.4.1 Positive Feedback

Participants praised the application's **minimalistic design** and the **clear visual representation** of task graphs, which made complex concepts more accessible, even for those new to Task Graph Scheduling. The **interactive features**, including the ability to **customise and download graphs** and the **game-like tasks**, were highly valued for their engaging and educational impact. Users also appreciated the smooth graph creation process and the **zoom functionality**, which enhanced their overall experience.

The inclusion of **tooltips** and a **consistent design** across all pages helped users understand the application's functionalities. The balance between theory and practice was noted as a strong point, enabling users to learn and apply concepts effectively. Even those without a background in Computer Science found the application **user-friendly** and **easy to navigate**. Overall, the application was commended for being both engaging and educational.

6.2.4.2 Areas for Improvement and Suggested Changes

Some users felt that the structural layout of the application, particularly in the Algorithms section, could be better organised. They suggested adding **short descriptions** for each section to help users understand their purpose before navigating through them. Another common concern was the presence of **too much text**, particularly in the algorithm explanations and pseudocode sections. This made the application feel overwhelming, especially for those less familiar with the subject matter.

On the other hand, in the timer game, participants expressed a desire for a **more visual representation of the countdown**, such as a progress bar or circular timer, which would make it easier to track time. Suggestions were also made to enhance interactivity in the problem-solving sections. For example, users proposed **enabling direct drag-and-drop functionality for assigning processors to nodes**, rather than relying on selection lists.

Finally, the addition of a **glossary of key terms** was frequently recommended, particularly for users with less background in computer science. This would make the application more accessible and help users navigate the technical content more easily. Participants also suggested including background information about the application's creator, references for the provided content, and links to further learning materials to enhance the application's credibility and educational value.

6.3 Critical analysis

The primary goal of this dissertation was to develop a game-like application that would allow students to explore task graph scheduling problems and algorithms in an engaging and interactive manner. With a strong emphasis on gamification, the objective was to create a tool that not only educates but also motivates learners, making scheduling algorithms more accessible through hands-on experience. In this analysis, we reflect on how well our application has met these original objectives and where it may have fallen short.

6.3.1 Achievement of Original Goals

Educational Tool with Gamification. The application has successfully achieved its primary goal of integrating gamification into the learning of task graph scheduling. All SUS, UEQ and open-ended feedback results highlighted positive feedback on the engaging, game-like tasks and interactive features.

Interactive Learning Experience. The interactive nature of the application, including features for graph customisation and immediate algorithmic feedback, was well-received. High ratings in the UEQ for efficiency and perspicuity reflect the effectiveness of these interactive elements in helping users grasp complex concepts and apply them practically.

Customisation and Feedback. The ability for users to create and manipulate graphs, along with the provision of real-time feedback, successfully supported the goal of providing a practical learning environment. This feature was particularly praised in the SUS results for its usability and in the UEQ for enhancing user engagement.

6.3.2 Unmet Objectives

Despite the application's success in several areas, one significant unmet objective was the **management of the text complexity and length**. Feedback from users and SUS results revealed that the extensive text and detailed explanations were overwhelming, particularly for those less familiar with the subject matter. This issue suggests that while the application achieved its goals in terms of engagement and interactivity, it did not fully address the need for clear, concise explanations that could make complex concepts more accessible. The text-heavy content proved difficult for users to follow, ultimately detracting from the overall educational value. For future development, we should reconsider the optimal scenarios for the application's use. It may be more effective as a supplementary tool alongside classroom instruction, where the text's length can be reduced without compromising the learning experience.

In conclusion, the application met most of its primary goals, including providing an interactive and gamified learning experience. However, the issue of overwhelming text remains a significant area for improvement.

Chapter 7

Conclusions

For this dissertation we designed, implemented and evaluated an educational application aimed at teaching task graph scheduling algorithms. Through a combination of technical development, user-centred design, and different evaluation techniques, we have gained valuable insights into both the strengths and areas for improvement in our application.

During the development phase, we prioritised creating an intuitive interface that supports learning through interactive features. By incorporating visual representations, customisable graphs and game-like tasks, our goal was to make complex concepts more accessible to a broad audience, including those without a background in Computer Science.

The user evaluation, conducted through several types of surveys and open-ended feedback, provided comprehensive insights into the application's usability and user experience. The demographic analysis revealed a diverse participant group, allowing us to assess the application's effectiveness across different levels of technical expertise. The System Usability Scale results indicated a generally positive reception, with an overall score of 77.9, reflecting the website's user-friendliness and functional design. Similarly, the User Experience Questionnaire highlighted strengths in areas such as attractiveness and perspicuity. Feedback from open-ended questions underscored the application's minimalistic design, clear visual representations, and effective integration of theory and practice as key strengths.

7.1 Future work

One possible direction for future development is the integration of this application into a larger educational platform, where users can explore not only task graph scheduling but also to a broader spectrum of computer science topics.

Another important direction for future development is a consideration of the scenarios in which the application is most effectively used. As previously mentioned, the application's text-heavy content was identified as a barrier for some users. To address this, future work should explore the optimal contexts in which the application can be deployed, such as using it alongside classroom explanations. In such scenarios, the application could serve as a supplementary tool that reinforces learning without overwhelming users with extensive text. This approach would allow educators to deliver complex content more effectively by reducing the on-screen text and relying on in-person guidance to clarify intricate concepts.

Introducing more advanced functionalities is another key area for improvement. For instance, incorporating an "Undo" button during the task-to-processor assignment process would allow users to experiment and learn from their mistakes without the need to reset their entire progress. Additionally, expanding the selection of algorithms and incorporating brute-force solutions for smaller graphs could offer users more diverse approaches to problem-solving, especially considering that the algorithms explored are not guaranteed to yield optimal solutions due to the NP-hard nature of the problem.

Automated testing of the web interface and APIs would also be a significant step forward, ensuring the application's reliability and accuracy as it evolves. Revisiting the idea of contrasting different algorithms by highlighting varying task assignments is another feature worth reconsidering, despite its complexity, as it could greatly enhance users' understanding of algorithmic differences.

Another crucial aspect for future development is adapting the application for all screen sizes and devices, ensuring that it is fully responsive and accessible across various platforms.

User feedback has highlighted specific areas for refinement that should be prioritised in future iterations. In particular, adding a visual progress bar or circular timer would improve the application's usability, making it more intuitive and user-friendly.

In conclusion, while the current iteration of this application has successfully met its initial goals, there remains significant potential for growth and enhancement. By pursuing these future developments, the application can become a more versatile, engaging, and comprehensive educational tool, capable of serving a diverse range of users with varying levels of expertise.

Bibliography

[1] Creating a github pages site. URL: https://docs.github.com/en/pages/getting-started-with-github-pages/creating-a-github-pages-site.

[2] User experience questionnaire (ueq). URL: https://www.ueq-online.org/?lang=es.

[3] Chadia Abras, Diane Maloney-Krichmar, Jenny Preece, et al. User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications*, 37(4):445–456, 2004.

[4] Brandon Beckwith and Dewan Ahmed. Gamification of undergraduate computer science data structures. In *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 648–650. IEEE, 2018.

[5] John Brooke. Sus: a retrospective. *Journal of usability studies*, 8(2), 2013.

[6] Darina Dicheva, Keith Irwin, and Christo Dichev. Oneup: Engaging students in a gamified data structures course. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, pages 386–392, 2019.

[7] Python Software Foundation. *Python Documentation*, 2023. URL: https://docs.python.org/3/.

[8] Sara V Fuller. Understanding the cognitive aspects of human error will increase the usability of user interfaces. 2005.

[9] Cory Gackenheimer. *Introduction to React*. Apress, 2015.

[10] Mourya Reddy Narasareddy Gari, Gursimran Singh Walia, and Alex David Radermacher. Gamification in computer science education: A systematic literature review. In *2018 ASEE Annual Conference & Exposition*, 2018.

[11] Michael R Gary and David S Johnson. Computers and intractability: A guide to the theory of np-completeness, 1979.

[12] Aric Hagberg, Pieter J Swart, and Daniel A Schult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

[13] Tarek Hagras and J Janeček. Static vs. dynamic list-scheduling performance comparison. *Acta Polytechnica*, 43(6), 2003.

[14] Juho Hamari, Jonna Koivisto, and Harri Sarsa. Does gamification work?–a literature review of empirical studies on gamification. In *2014 47th Hawaii international conference on system sciences*, pages 3025–3034. Ieee, 2014.

[15] Jing-Jang Hwang, Yuan-Chieh Chow, Frank D Anger, and Chung-Yee Lee. Scheduling precedence graphs in systems with interprocessor communication times. *siam journal on computing*, 18(2):244–257, 1989.

[16] International Organization for Standardization. ISO 9241-11: Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts, March 2018. International Standard. URL: https://www.iso.org/standard/63500.html.

[17] Koyeb. *Koyeb Documentation*, 2023. URL: https://www.koyeb.com/.

[18] Yu-Kwong Kwok and Ishfaq Ahmad. Benchmarking and comparison of the task graph scheduling algorithms. *Journal of Parallel and Distributed Computing*, 59(3):381–422, 1999.

[19] Inc. Meta Platforms. *React Documentation*, 2023. URL: https://react.dev/.

[20] Tomoko Nemoto and David Beglar. Likert-scale questionnaires. In *JALT 2013 conference proceedings*, volume 108, pages 1–6, 2014.

[21] Michael A. Palis, Jing-Chiou Liou, and David S. L. Wei. Task clustering and scheduling for distributed memory parallel architectures. *IEEE Transactions on Parallel and Distributed Systems*, 7(1):46–55, 1996.

[22] Emilia Popa, Mauro Iacono, and Florin Pop. Adapting mcp and hlfet algorithms to multiple simultaneous scheduling. *International Journal of Parallel Programming*, 46:607–629, 2018.

[23] Yves Robert et al. Task graph scheduling., 2011.

[24] Cristian Rusu, Virginica Rusu, Silvana Roncagliolo, and Carina González. Usability and user experience: What should we care about? *International Journal of Information Technologies and Systems Approach (IJITSA)*, 8(2):1–12, 2015.

[25] Ujjawal Shrestha. Implementing api in reactjs. 2020.

[26] Oliver Sinnen. *Task scheduling for parallel systems*, volume 60. John Wiley & Sons, 2007.

[27] Adam Spanier, Sherri Weitl Harms, and John Hastings. A classification scheme for gamification in computer science education: Discovery of foundational gamification genres in data structures courses. In *2021 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE, 2021.

[28] Srishti Srivastava and Ioana Banicescu. *Scheduling in Parallel and Distributed Computing Systems*, pages 313–337. Springer International Publishing, Cham, 2018. doi:10.1007/978-3-319-93109-8_11.

[29] Simon Su, Edward Zhang, Paul Denny, and Nasser Giacaman. A game-based approach for teaching algorithms and data structures using visualizations. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, pages 1128–1134, 2021.

[30] Will T. Measuring and interpreting system usability scale (sus), Feb 2021. URL: https://uiuxtrend.com/measuring-system-usability-scale-sus/.

[31] Jeffrey D. Ullman. Np-complete scheduling problems. *Journal of Computer and System sciences*, 10(3):384–393, 1975.

[32] François Voron. *Building Data Science Applications with FastAPI: Develop, manage, and deploy efficient machine learning applications with Python*. Packt Publishing Ltd, 2023.

[33] webkid GmbH. *React Flow Documentation*, 2023. URL: https://reactflow.dev/.

[34] Yudho Yudhanto, Wahyu Mangantasno Pryhatyanto, and Winita Sulandari. Designing and making ui/ux designs on the official website with the design

thinking method. In *2022 1st International Conference on Smart Technology, Applied Informatics, and Engineering (APICS)*, pages 165–170, 2022. doi:10.1109/APICS56469.2022.9918684.

Appendix A

Designs

A.1 Initial sketches

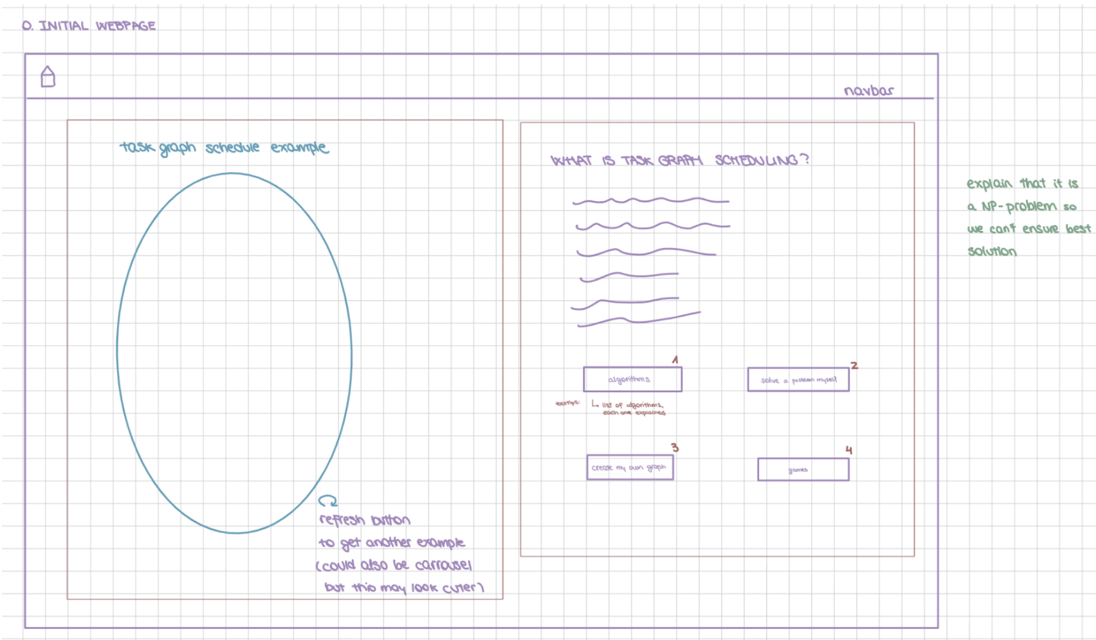

Figure A.1: Homepage

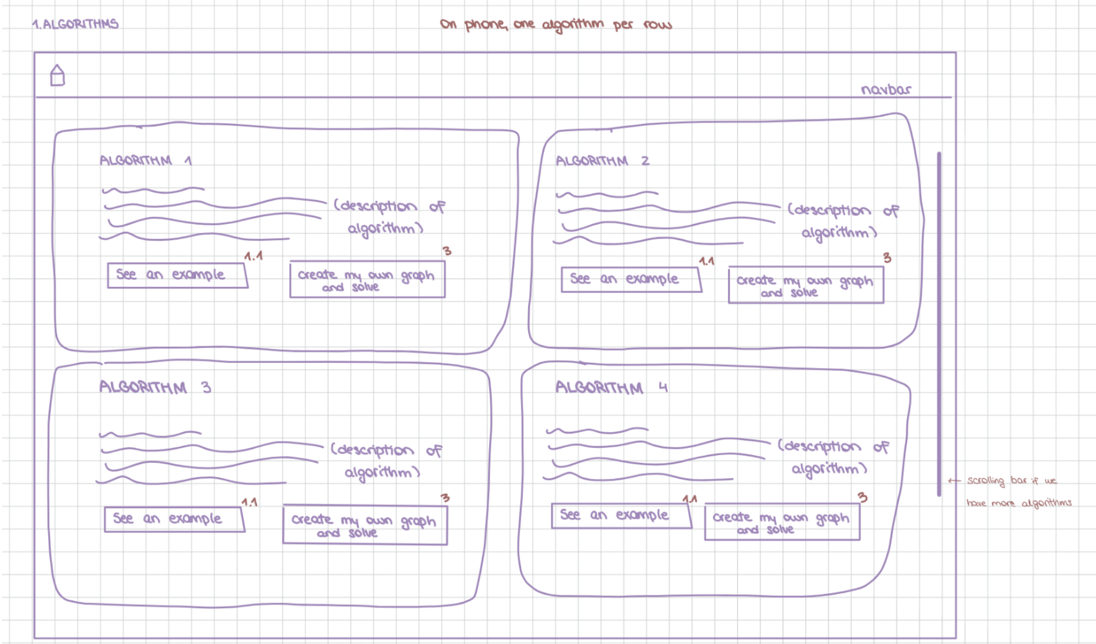

Figure A.2: Algorithms page

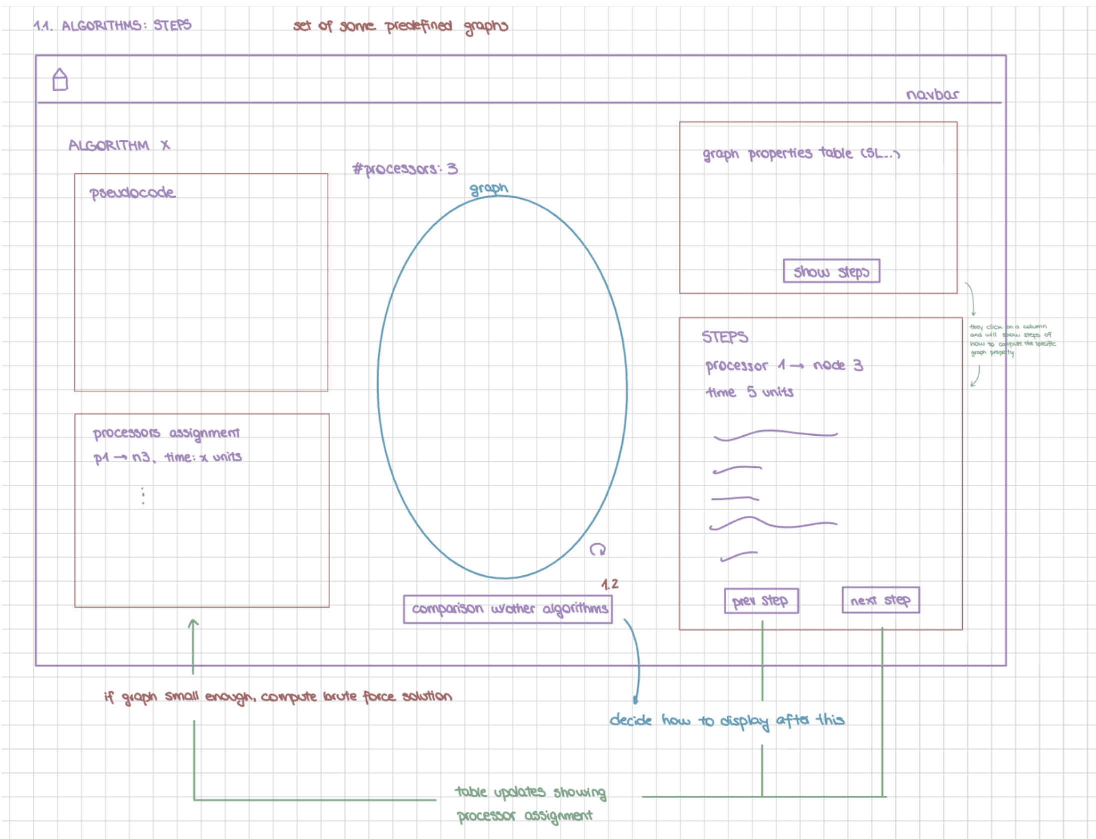

Figure A.3: Algorithms - steps

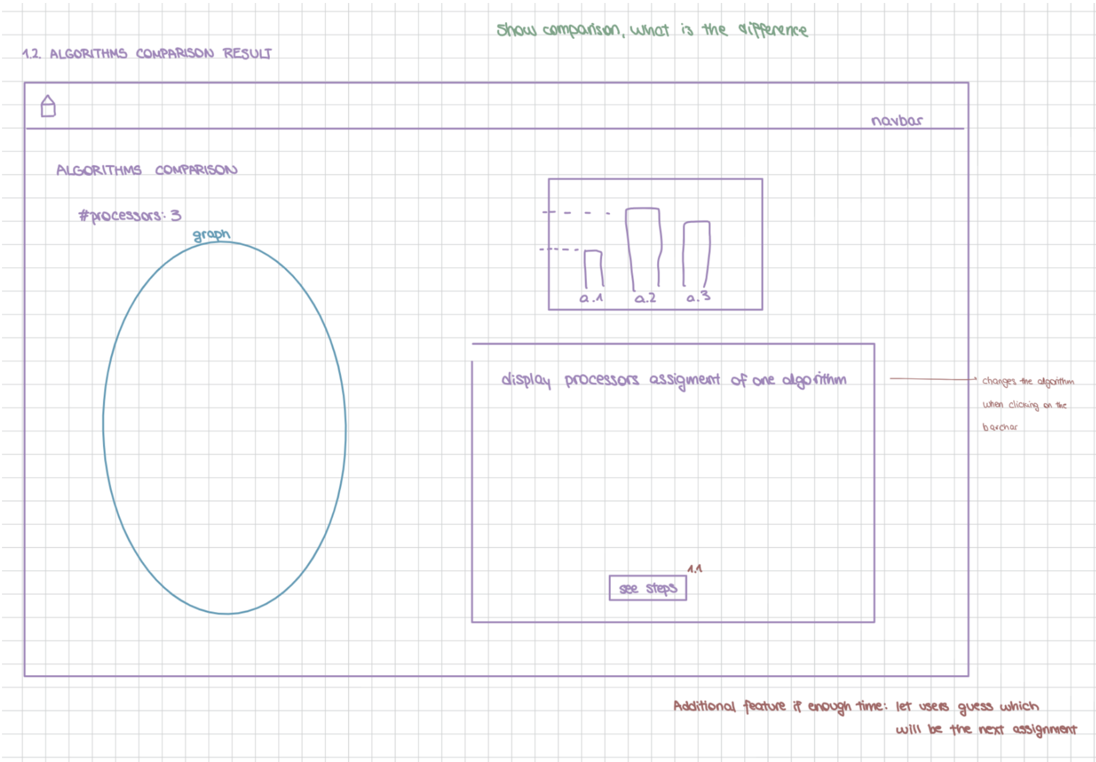

Figure A.4: Algorithms - comparison

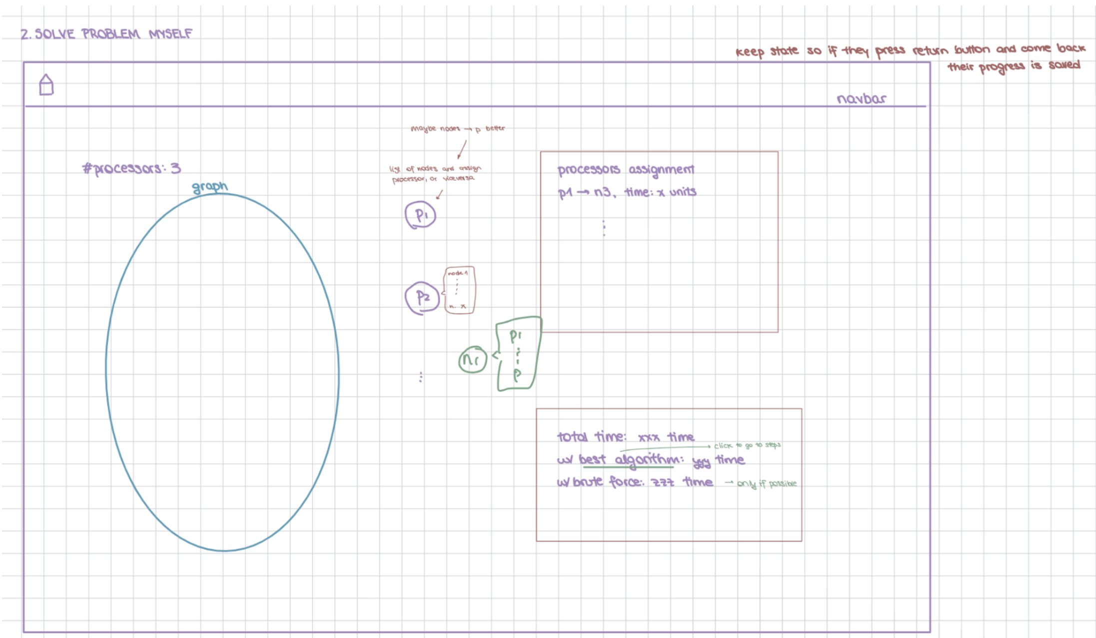

Figure A.5: Users solve problem page

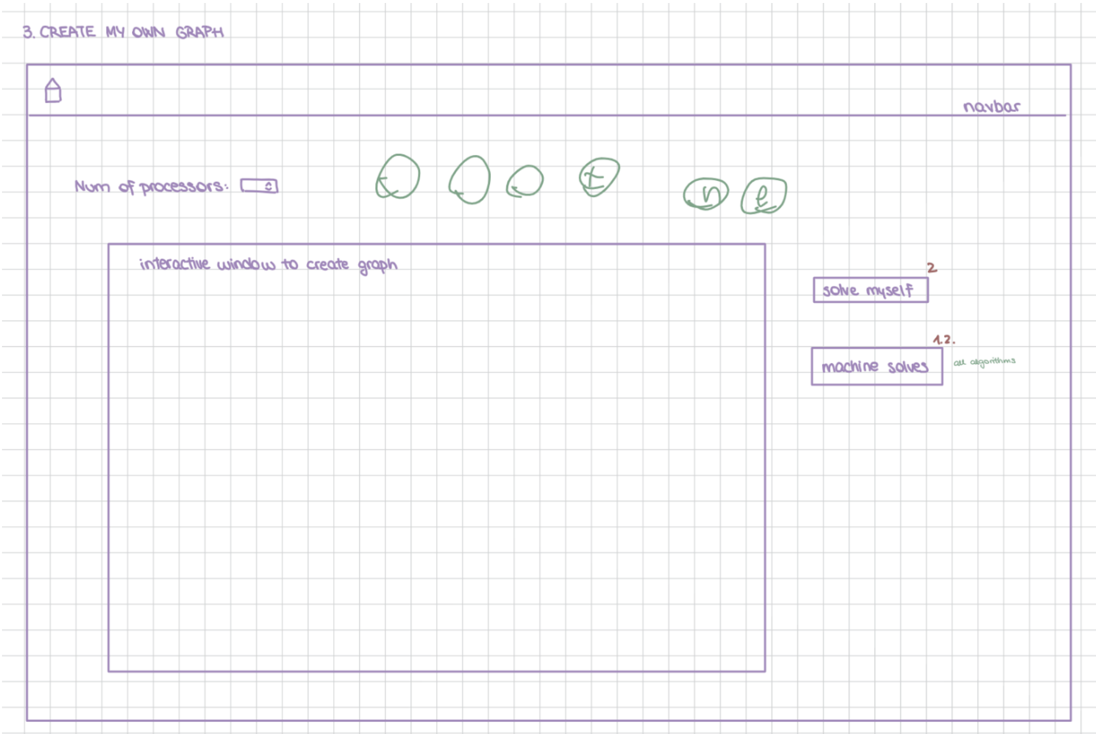

Figure A.6: Customise graph page

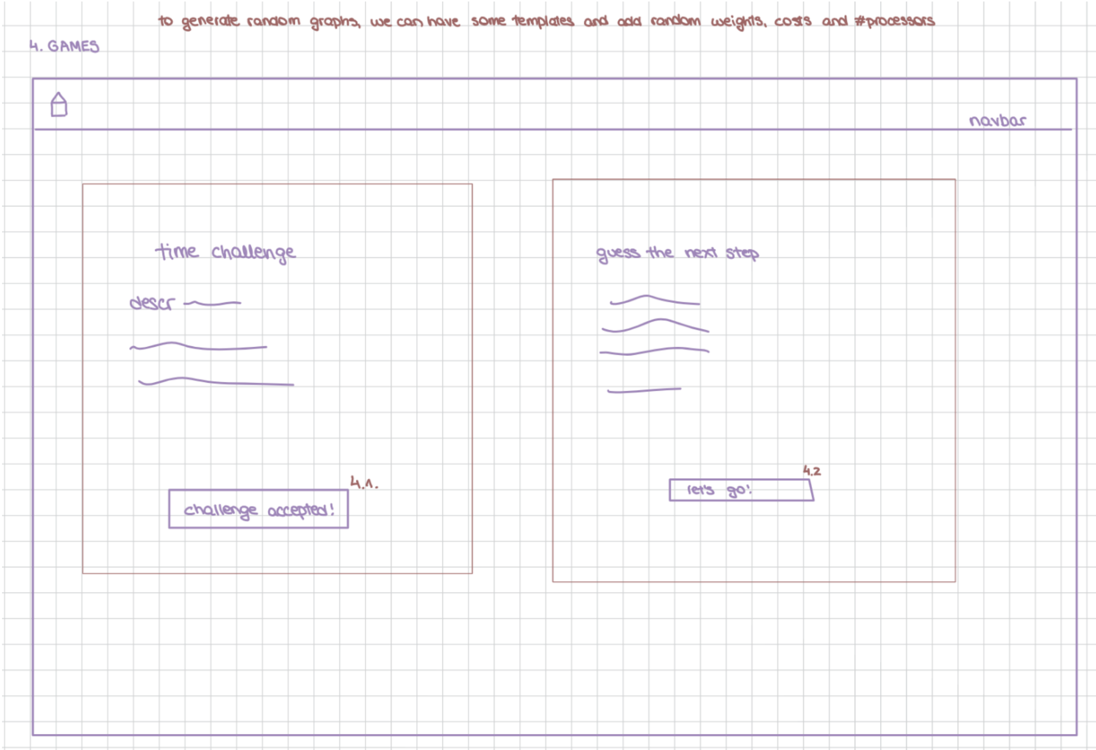

Figure A.7: Game selection page

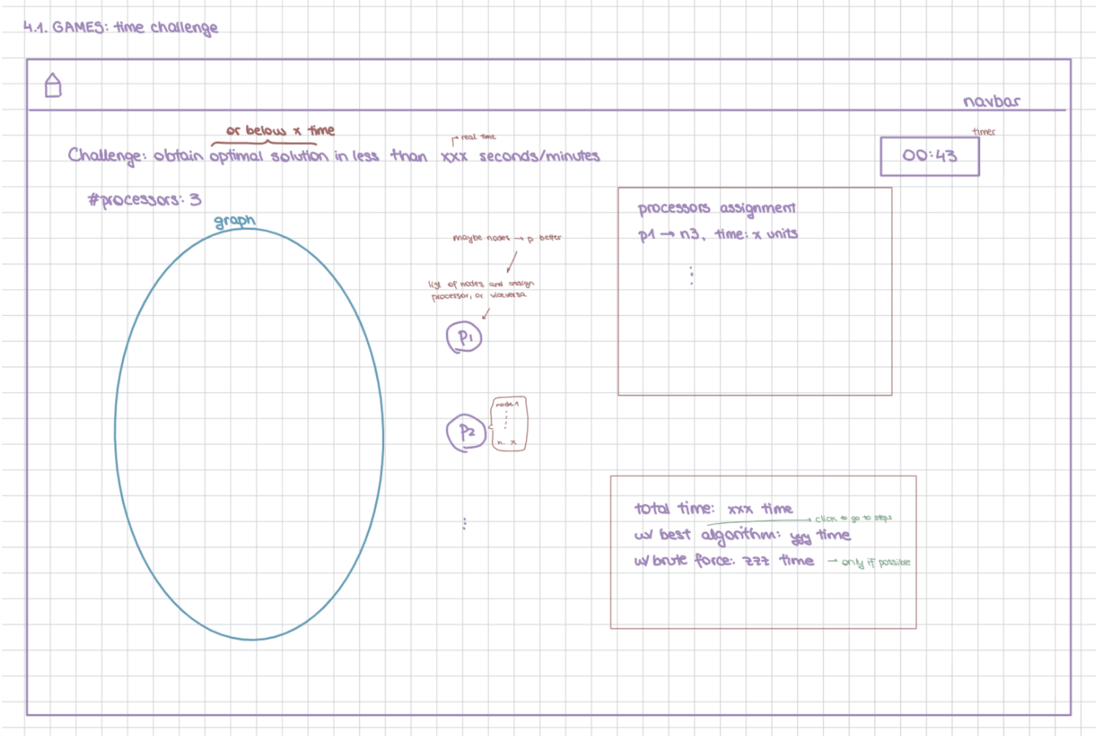

Figure A.8: Time challenge game

A.2 Sitemap

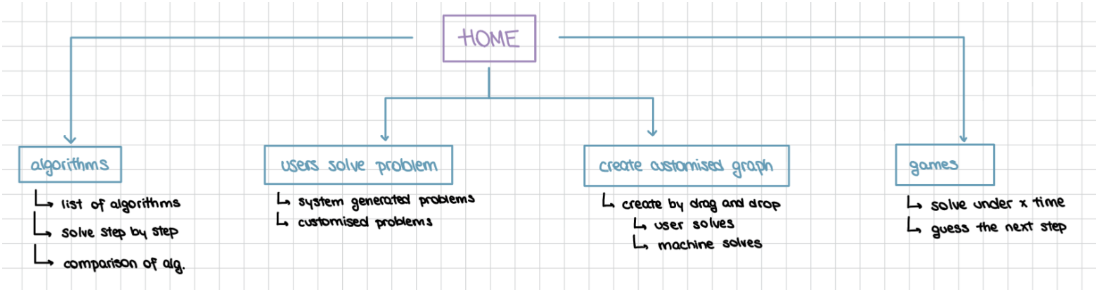

Figure A.9: Sitemap

A.3 Templates of graphs

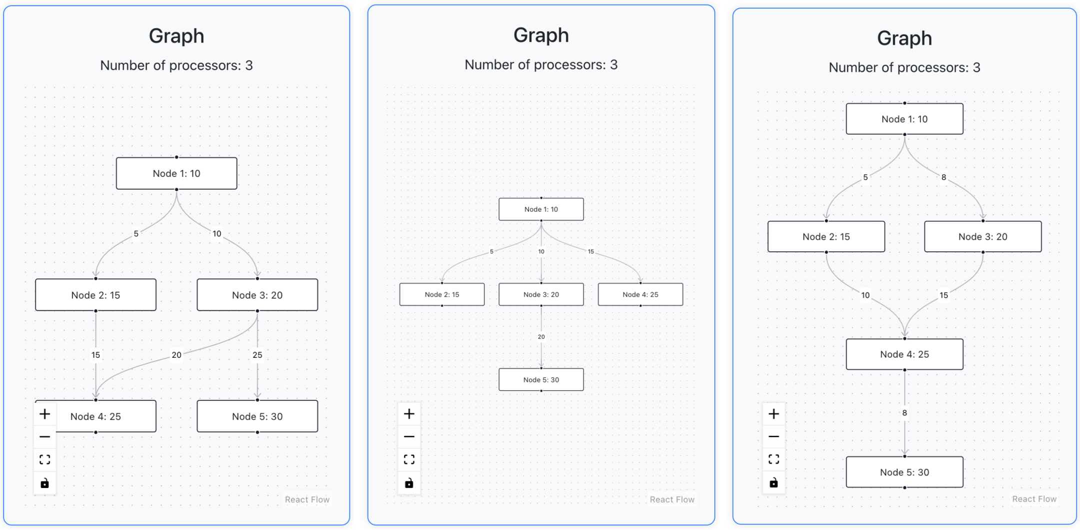

Figure A.10: 5 nodes templates

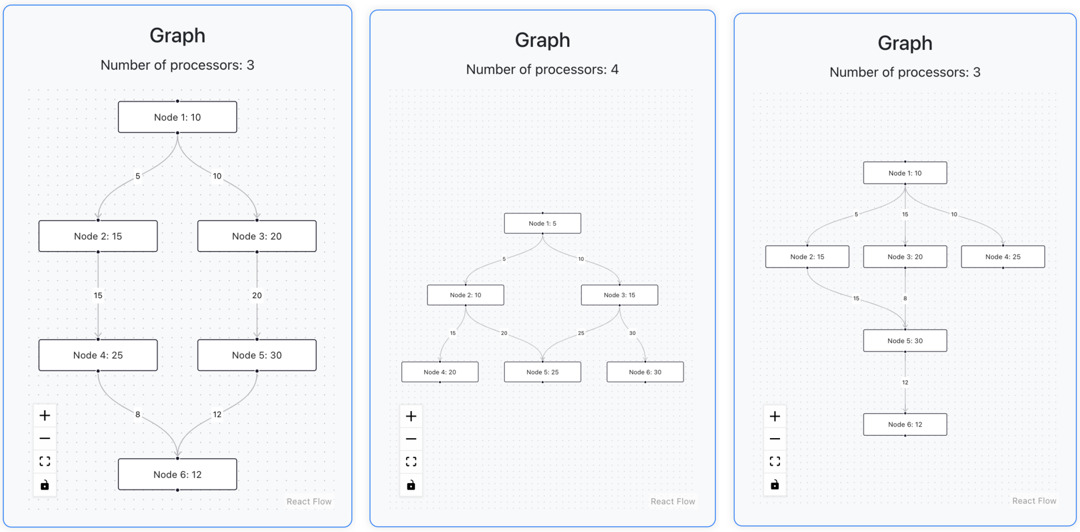

Figure A.11: 6 nodes templates

Figure A.12: 7 nodes templates

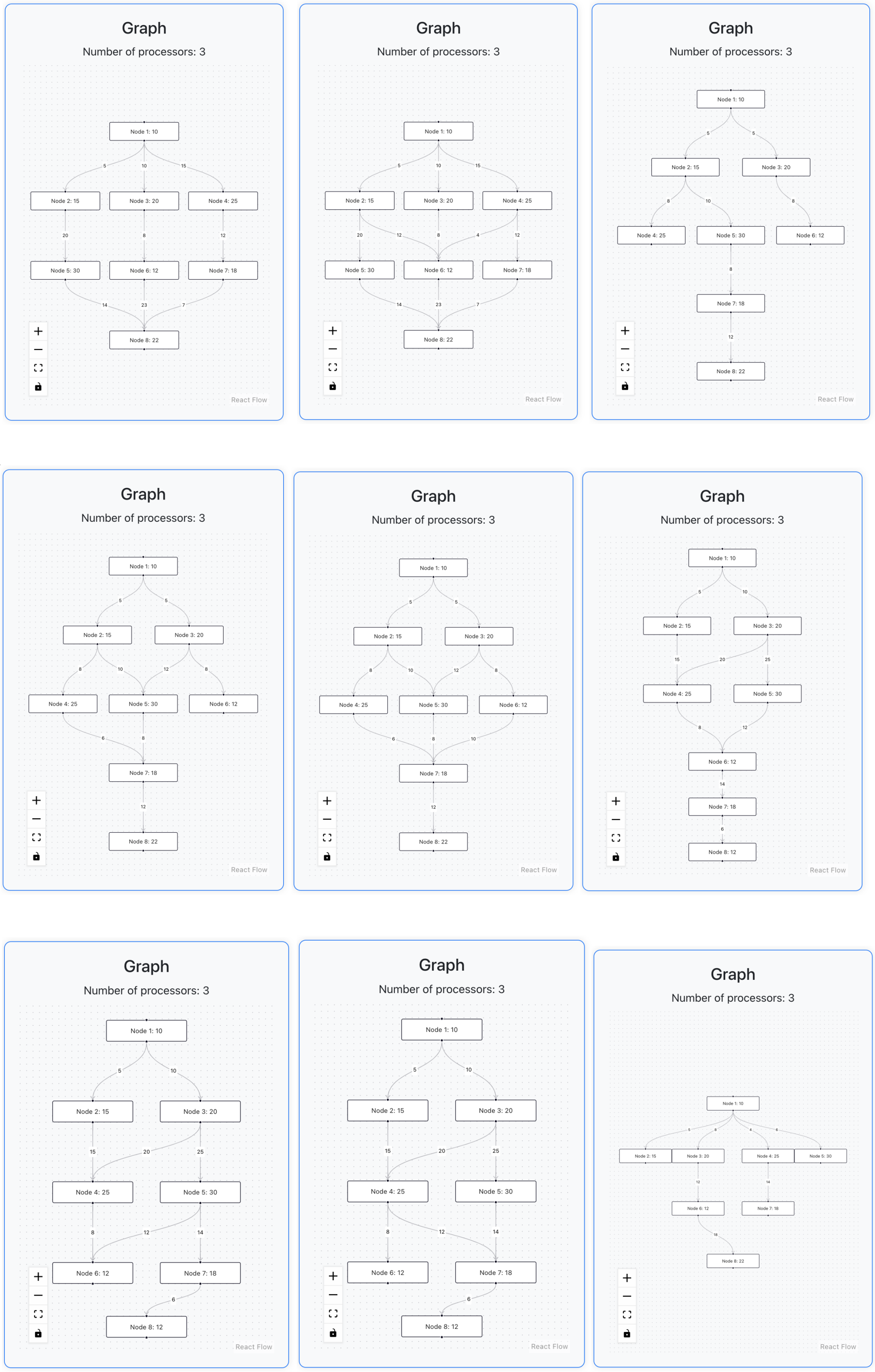

Figure A.13: 8 nodes templates

Appendix B

Survey questions and answers

Description of survey for users:

Hello! Thank you for taking the time to participate in this survey. I am working on my dissertation for the MSc in Computer Science at the University of Edinburgh, and your feedback will be invaluable. The survey is divided into four parts: general demographic questions, System Usability Scale (SUS) questions, User Experience Questionnaire (UEQ) items, and some optional open-ended questions for your suggestions and other comments. After exploring the website, completing this survey should take approximately 10-20 minutes.

Please use a computer to browse the website, as it is designed for optimal experience on a larger screen. Additionally, some features require a mouse or trackpad to function properly, such as drag-and-drop actions.

You can find the website here: `https://lucia-jiang.github.io/taskGraphScheduling-fro`

Thank you once again for your assistance!

B.1 Demographic questions

Description of section for users: To help us better understand the diversity and background of our participants, please answer the following demographic questions. This information will allow us to analyse the results more effectively and ensure our findings are representative of different groups. Your responses will be kept confidential and will only be used for research purposes.

Question 1: What is your range age?

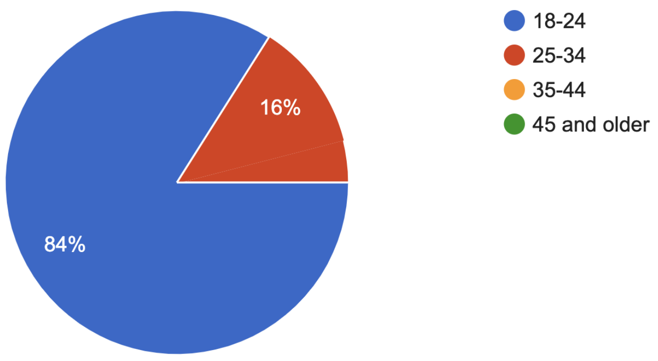

Figure B.1: Answers to demographic question 1

Question 2: Are you currently studying Computer Science or have you graduated from a related degree programme?

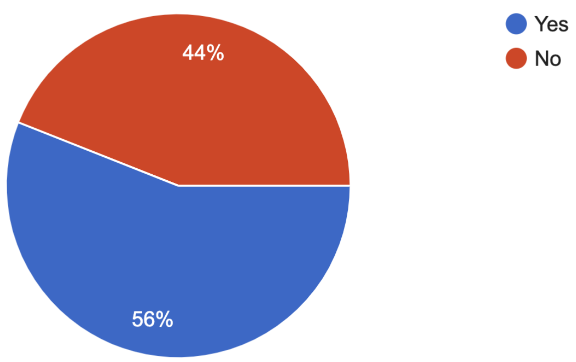

Figure B.2: Answers to demographic question 2

Question 3: How familiar are you with Data Structures and Algorithms?

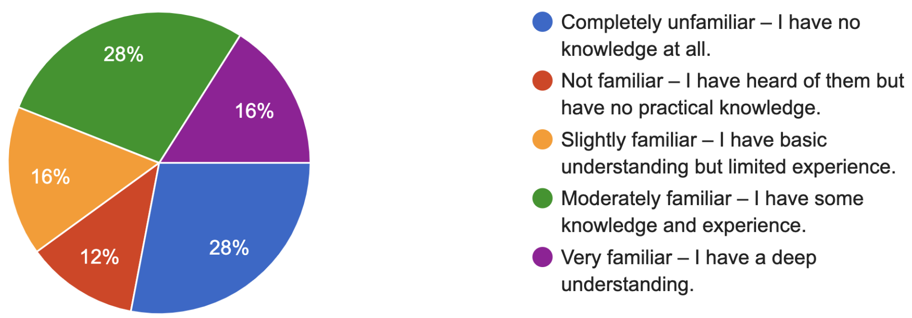

Figure B.3: Answers to demographic question 3

Question 4: Do you know what a graph is, or do you understand the concept of graphs with nodes and weights?

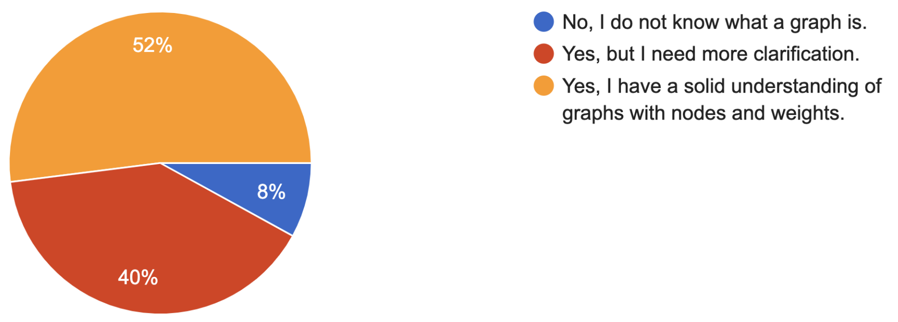

Figure B.4: Answers to demographic question 4

Question 5: Are you familiar with task graph scheduling problems and algorithms?

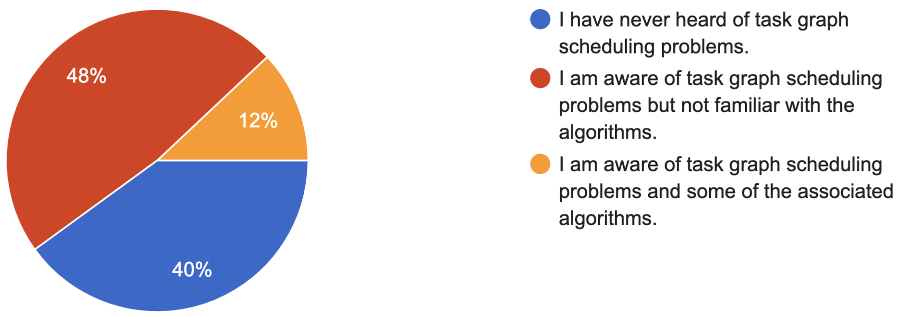

Figure B.5: Answers to demographic question 5

Question 6: What is your highest level of education?

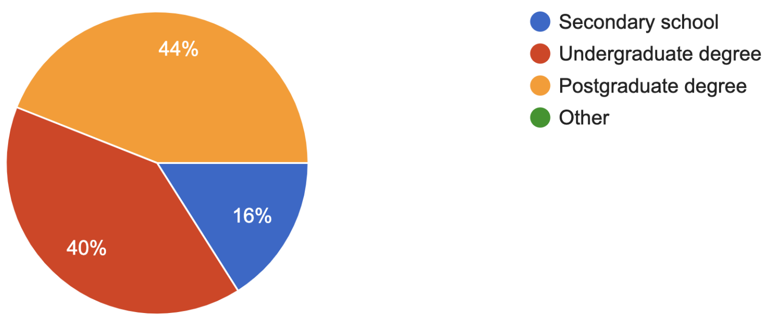

Figure B.6: Answers to demographic question 6

Question 7: How would you rate your proficiency with using a computer?

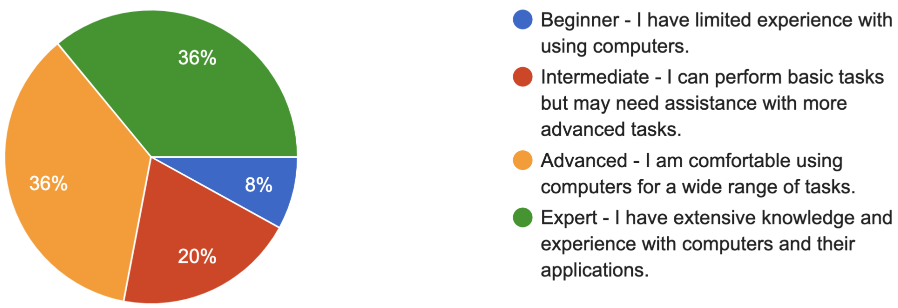

Figure B.7: Answers to demographic question 7

Question 8: How often do you engage with educational games or simulations?

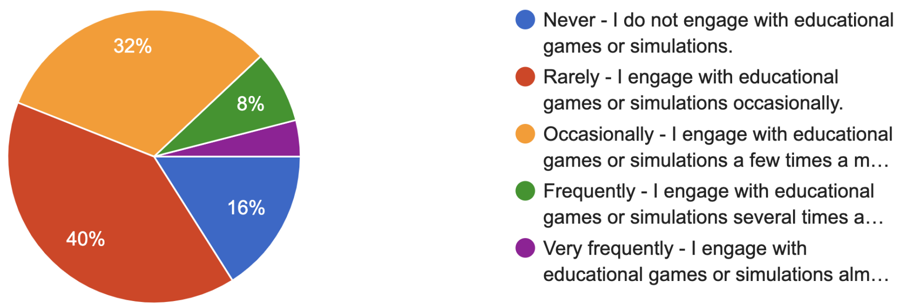

Figure B.8: Answers to demographic question 8

Question 9: What is your preferred learning style?

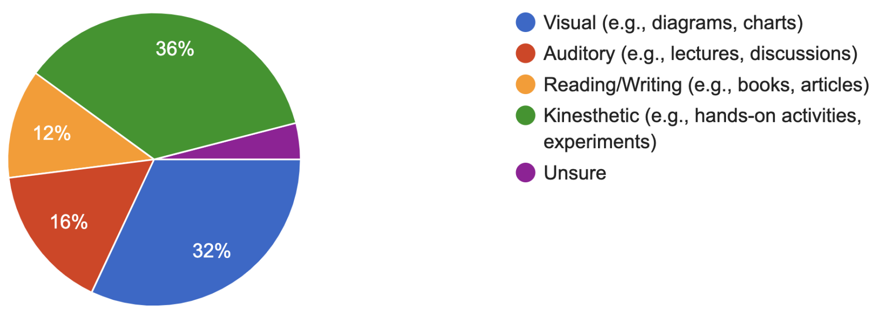

Figure B.9: Answers to demographic question 9

Question 10: How do you usually learn new technical concepts?

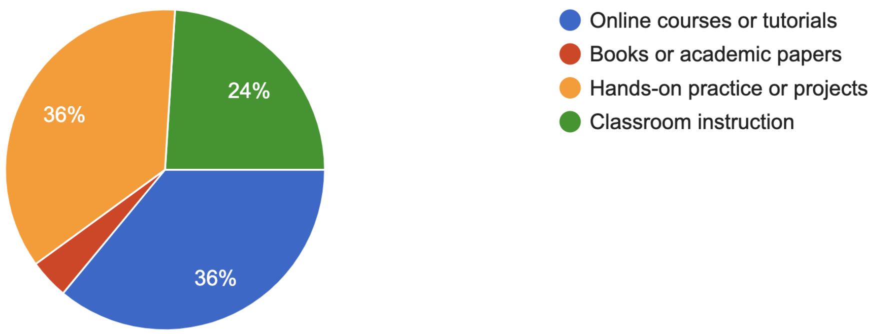

Figure B.10: Answers to demographic question 10

B.2 System Usability Scale (SUS)

Description of section for users: In this section, you will be asked to rate your agreement with a series of statements about the website. Your responses will help us understand how user-friendly and efficient the application is. Please rate each statement on a scale from 1 (Strongly Disagree) to 5 (Strongly Agree).

Question 1: I think that I would like to use this application regularly if I needed to learn about task graph scheduling algorithms or other algorithms.

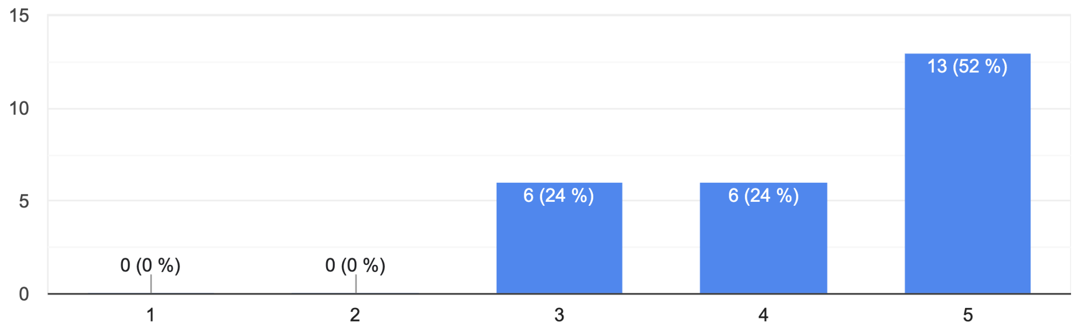

Figure B.11: Answers to SUS question 1

Question 2: I found the application unnecessarily complex for understanding task graph scheduling concepts.

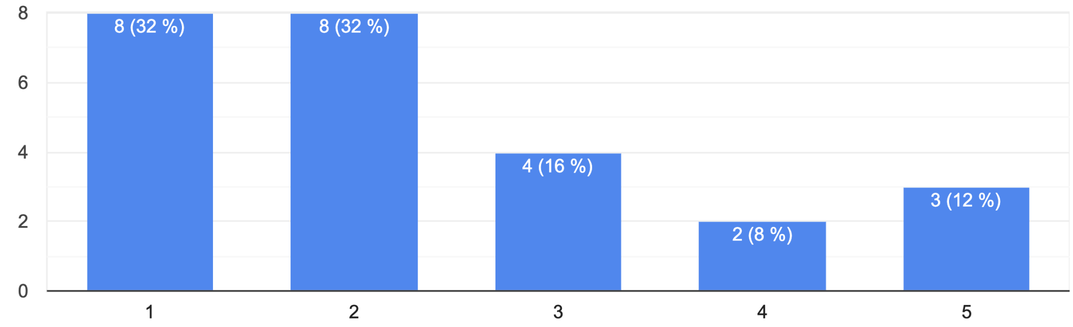

Figure B.12: Answers to SUS question 2

Question 3: I thought the application was easy to navigate and use.

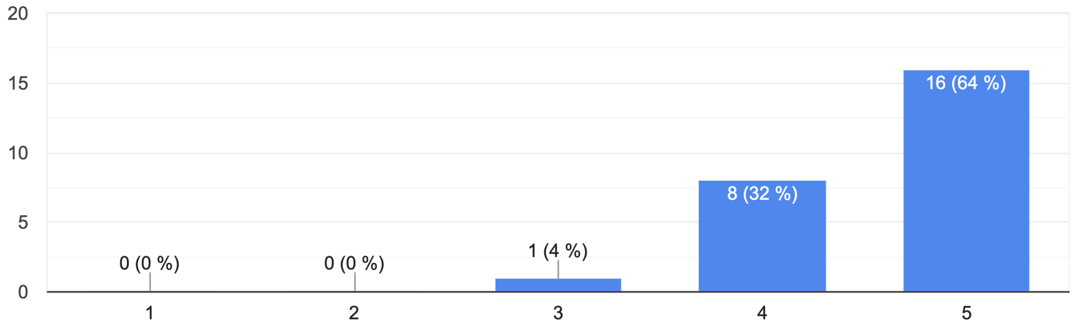

Figure B.13: Answers to SUS question 3

Question 4: I think that I would need the support of a technical person to use this application effectively.

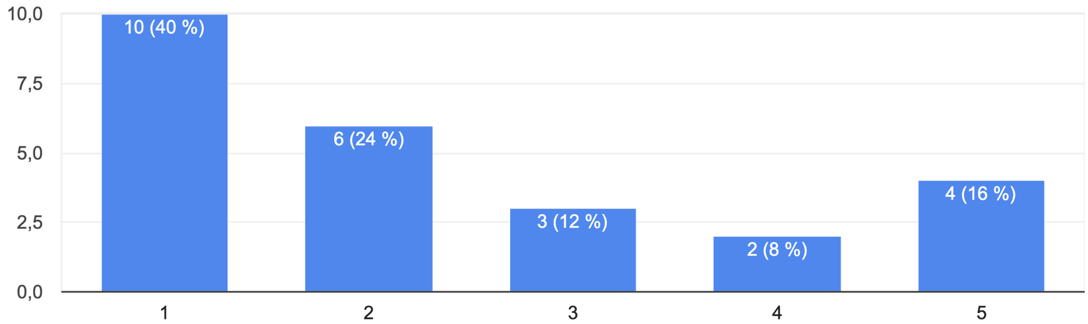

Figure B.14: Answers to SUS question 4

Question 5: I found that the various features and functions of the application were well integrated.

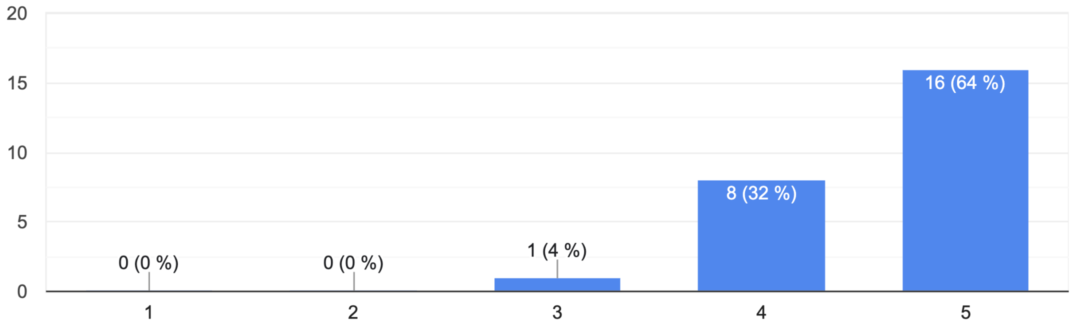

Figure B.15: Answers to SUS question 5

Question 6: I thought there was too much inconsistency in the application's design and functionality.

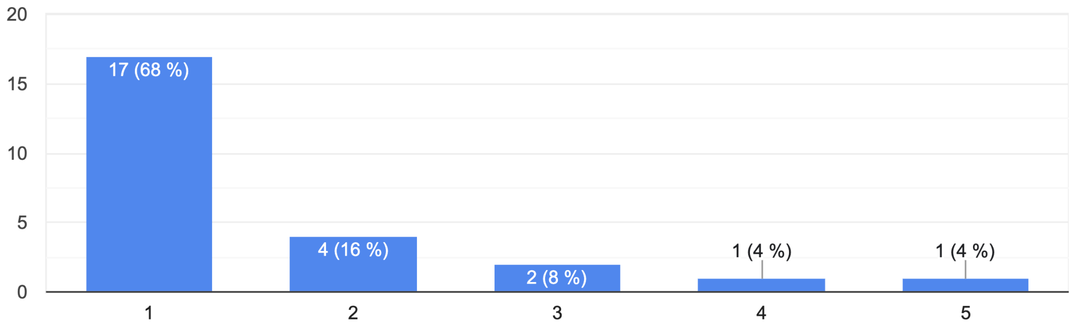

Figure B.16: Answers to SUS question 6

Question 7: I would imagine that most people would learn to use this application very quickly.

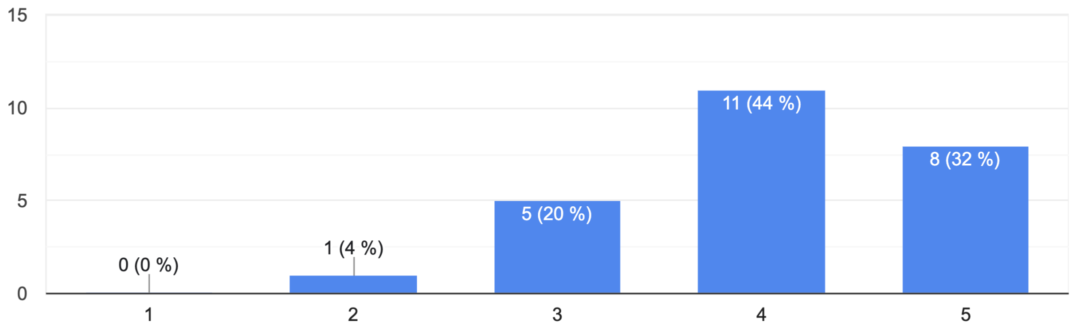

Figure B.17: Answers to SUS question 7

Question 8: I found the application very cumbersome to use for learning task graph scheduling algorithms.

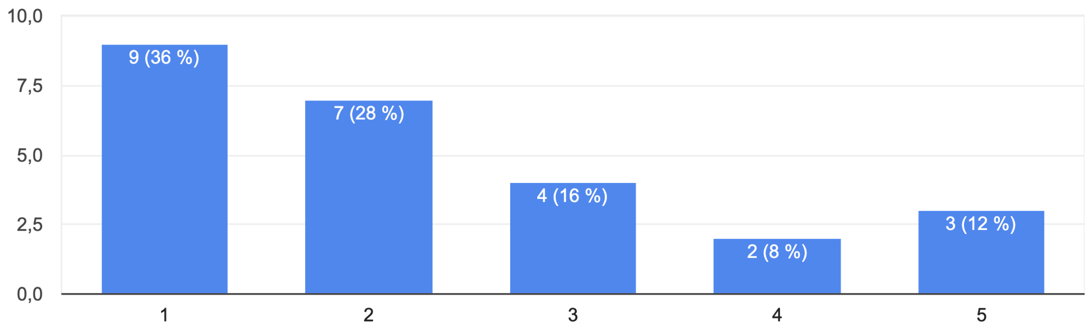

Figure B.18: Answers to SUS question 8

Question 9: I felt very confident using the application to understand task graph scheduling.

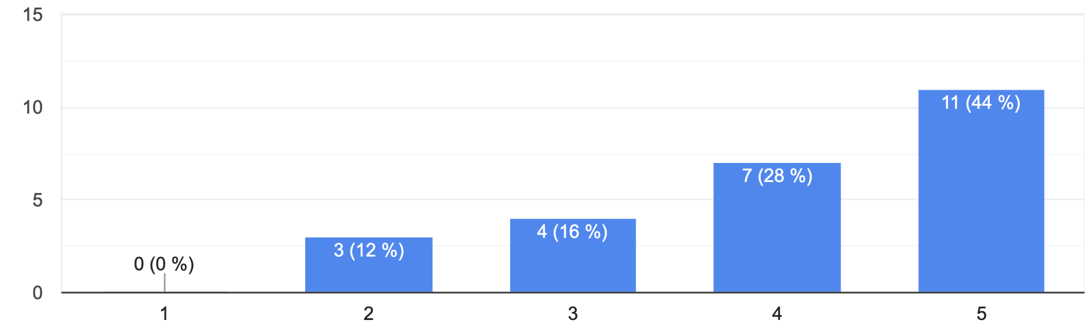

Figure B.19: Answers to SUS question 9

Question 10: I needed to learn a lot of things before I could get going with the application.

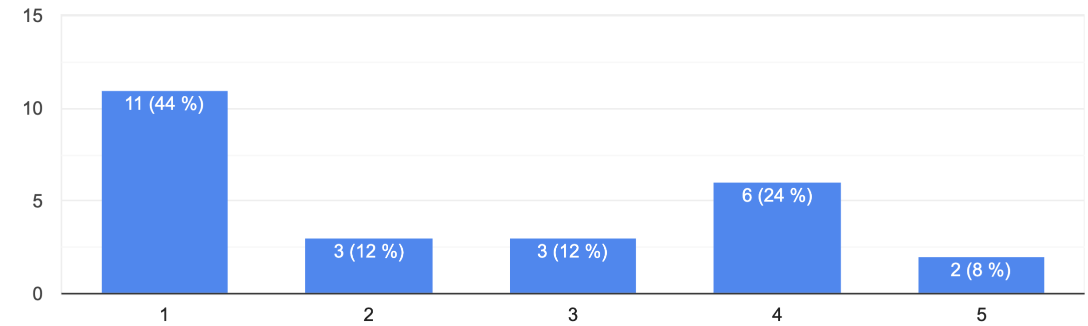

Figure B.20: Answers to SUS question 10

B.3 User Experience Questionnaire (UEQ)

Description of section for users: This section consists of pairs of contrasting attributes that may apply to the application. The circles between the attributes represent gradations between the opposites. You can express your agreement with the attributes by ticking the circle that most closely reflects your impression.

Please decide spontaneously. Don't think too long about your decision to make sure that you convey your original impression. Sometimes you may not be completely sure about your agreement with a particular attribute or you may find that the attribute does not apply completely to the particular product. Nevertheless, please tick a circle in every line. It is your personal opinion that counts. Please remember: there is no wrong or right answer!

	1	2	3	4	5	6	7		
annoying	○	○	○	○	○	○	○	enjoyable	1
not understandable	○	○	○	○	○	○	○	understandable	2
creative	○	○	○	○	○	○	○	dull	3
easy to learn	○	○	○	○	○	○	○	difficult to learn	4
valuable	○	○	○	○	○	○	○	inferior	5
boring	○	○	○	○	○	○	○	exciting	6
not interesting	○	○	○	○	○	○	○	interesting	7
unpredictable	○	○	○	○	○	○	○	predictable	8
fast	○	○	○	○	○	○	○	slow	9
inventive	○	○	○	○	○	○	○	conventional	10
obstructive	○	○	○	○	○	○	○	supportive	11
good	○	○	○	○	○	○	○	bad	12
complicated	○	○	○	○	○	○	○	easy	13
unlikable	○	○	○	○	○	○	○	pleasing	14
usual	○	○	○	○	○	○	○	leading edge	15
unpleasant	○	○	○	○	○	○	○	pleasant	16
secure	○	○	○	○	○	○	○	not secure	17
motivating	○	○	○	○	○	○	○	demotivating	18
meets expectations	○	○	○	○	○	○	○	does not meet expectations	19
inefficient	○	○	○	○	○	○	○	efficient	20
clear	○	○	○	○	○	○	○	confusing	21
impractical	○	○	○	○	○	○	○	practical	22
organized	○	○	○	○	○	○	○	cluttered	23
attractive	○	○	○	○	○	○	○	unattractive	24
friendly	○	○	○	○	○	○	○	unfriendly	25
conservative	○	○	○	○	○	○	○	innovative	26

Figure B.21: UEQ questionnaire

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	Q23	Q24	Q25	Q26
P1	7	7	1	1	1	7	7	4	1	2	7	2	7	6	7	7	1	1	1	7	1	7	1	1	1	7
P2	4	6	1	2	4	4	5	6	1	2	6	2	5	5	4	5	1	4	4	5	2	6	2	2	2	4
P3	7	5	3	2	1	4	6	4	1	1	7	1	7	7	3	7	1	3	1	7	1	7	1	2	1	7
P4	5	5	3	4	3	4	5	4	3	3	5	3	5	4	4	5	3	3	4	5	3	5	3	4	3	4
P5	7	7	1	1	1	7	7	4	1	1	7	1	7	7	7	7	1	1	1	7	1	7	1	1	1	7
P6	5	5	1	3	2	4	3	4	4	2	7	3	2	4	4	6	4	4	2	6	5	7	1	1	3	5
P7	6	7	2	1	1	6	7	5	1	2	7	1	6	7	7	7	1	1	1	7	2	7	1	2	2	5
P8	6	7	5	1	1	3	7	7	1	5	7	2	7	6	1	6	1	2	2	7	1	7	1	3	1	2
P9	6	7	2	1	1	6	7	6	1	7	7	1	7	6	1	7	1	2	1	7	1	7	1	1	1	7
P10	4	5	4	3	5	5	5	6	3	3	4	2	4	4	5	5	4	3	3	5	2	5	2	4	3	5
P11	6	6	3	1	1	5	6	7	1	4	7	1	6	7	5	7	1	2	1	7	1	7	1	1	1	4
P12	7	7	1	2	1	7	7	7	2	2	7	1	5	7	6	7	1	2	2	7	1	7	2	1	1	6
P13	5	4	1	3	2	5	7	6	1	2	5	3	4	5	4	6	1	4	1	6	3	7	1	3	3	5
P14	7	5	2	3	2	7	6	4	3	2	5	1	5	6	6	6	1	1	1	6	5	6	1	2	3	6
P15	6	3	2	5	2	4	4	4	1	2	4	2	3	6	2	5	2	3	3	6	2	6	2	3	3	4
P16	6	6	2	3	2	5	6	5	2	4	5	1	5	6	4	5	2	4	4	5	2	4	3	3	4	4
P17	5	5	2	3	1	5	6	2	1	2	6	1	5	5	6	5	1	3	3	7	3	6	1	3	2	5
P18	7	7	3	2	1	6	7	4	2	2	7	1	7	7	5	7	1	1	1	7	1	7	7	6	1	7
P19	5	5	3	4	2	5	5	4	2	3	6	3	5	5	4	5	3	3	3	5	2	6	3	2	3	4
P20	7	6	1	2	1	6	6	5	2	2	7	1	6	6	6	6	1	1	1	6	2	6	1	2	1	6
P21	6	7	2	3	1	6	6	4	3	2	6	1	5	6	5	6	2	2	2	5	2	6	2	2	2	6
P22	7	7	1	1	1	7	7	3	1	1	7	1	7	7	6	7	1	1	1	7	1	7	1	1	1	7
P23	7	7	1	1	1	6	6	2	1	2	7	1	6	7	5	7	1	1	1	7	1	7	1	1	1	7
P24	7	7	2	1	2	6	7	2	1	1	7	1	6	5	5	6	1	2	1	7	1	7	1	1	1	7
P25	6	7	2	1	1	7	7	6	1	1	6	1	7	7	6	6	1	2	1	7	1	7	1	2	1	5

Table B.1: Results of UEQ

B.4 Open-Ended Questions

What did you like most about the application? Please describe any features, design elements, or aspects of the application that you found particularly useful or enjoyable.

- very intuitive
- Honestly, even though i don't really understand much about this topic, i still thought it was easy to use and pretty much clear and simple. Plus, I thought the navigation map was pretty cool.
- The application's clear visual representation of task graphs stands out as a particularly strong feature. The intuitive design makes it easy to understand task dependencies and communication costs, and the interactive elements, such as creating customized graphs and solving problems, add an engaging and educational dimension. Additionally, the detailed explanations provided offer valuable context and insights into the significance and complexity of task graph scheduling, enhancing the overall learning experience.
- I found very useful adding the games feature, as it will provide a more enjoyable way to learn Task Graph Scheduling. Moreover, the interface is really easy to understand and use.
- I really like how the information is distributed on the page, it is really nice to have a mix between theory and practice to acquire better all the knowledge
- The organization, the graphs parts and the games
- I liked that It is easy to use
- I liked the part of "Create Customised Graph", because it was super easy to choose a weight and drag a node to the pane.
- All the pages have the same style which helps the user to understand all the functionalities. All the comments and theories are useful and are easy to understand.
- to give you examples of the use of the algorithms and that you can check if you understand them in the games part

- Minimalistic Design and descriptions make it easy to learn the algorithms. The game-like tasks are great for almost all types of learners
- I found the design to be quite sleek and compact in the information it delivered. For someone who is relatively new to the idea of Task Graph Scheduling, and someone not studying Computer Science, I was still able to pick on the concepts. For Informatics students, I can definately see its uses, especially with the hands-on examples added. The visual aids for explaining the algorithms was also helpful.
- I really liked that you could move through learning about the algorithm, seeing an example and then being able to solve the problem yourself. The design for using the app is very intuitive.
- I found the application very minimalistic and I liked the color combinations
- Tooltips on the pages were particularly helpful to understand how each section worked
- I liked the fact that we could download the graphs and upload and create our own. Creation of graphs work very smoothly
- I liked the fact that the graphs were interactive. I appreciate that I can also zoom in and out. Also the feature of uploading a json helps a lot because I can return back to the same problem easily without needing to create the graph again, specially if it is a big one.

What did you dislike about the application? Please describe any features, design elements, or aspects of the application that you found confusing, frustrating, or unhelpful.

- I am guessing this is because I don't fully understand the topic and I am not the target audience, so I would say that the biggest issue for me was that I didnt really understand what was happening, especially since I am pretty bad at numbers and I didn't really understand some of the terms.
- The user interface, while functional, could benefit from a more polished and modern design to increase visual appeal and user engagement.
- Maybe because i dont know much about these topics, but i found the information a little bit overwhelming. Not because it was too much information, rather it had too many technicalities.

- Creating my customised graph is a bit confusing at the beginning, but as soon as you draw the first node, it is really simple and useful
- the large amount of text in the explanations that makes me feel like I'm being bombarded with a lot of information
- Somethings are a little bit confusing
- I didn't dislike anything honestly.
- I don't think there is any unpleasant element.
- that sometimes they don't give the ordered list in the examples in the algorithms section
- Some features were confusing. When you hold your cursor over the learning algorithms graphs, it changes to a cursor leading to the possibility of making actions, changing the graphs, though they are inmutable.
- Maybe a bit too much text, but it isn't a major issue in my opinion but it can be hard to follow especially when it comes to algorithms.
- Perhaps some rearranging/additions to the structural layout of the application? While it is perhaps intuitive to first click on the top left "Algorithms" section, which provides the basic concepts, it could be useful to provide a short description so users know what each section is for before clicking through.
- Sometimes, even on the large screen the images for the graph displayed a bit too small and with not enough contrast between the background and lines. I may choose to put the graph in its own row on the algorithm pages.
- Sometimes I felt like there was too much text. I feel like it would be a better application if it was a support from class, rather than a complete self-learning tool. For example the explanation of the problem on the main page, and all the pseudocodes are too long to read. I found it a little bit tiring to have to read all of that.

Do you have any suggestions for improving the application? Please provide any ideas or recommendations for enhancing the functionality, usability, or overall experience of the application.

- I think it is already pretty efficient to be honest
- Updating the user interface with modern design elements and smoother animations could improve user experience. Additionally, implementing real-time feedback and interactive tutorials would provide valuable guidance, helping users to learn more effectively.
- I think adding a ”glossary” section might be good for people who have a beginner or intermidiate level since a beginner wouldn’t be that much knowledgeable about these type of terms.
- I would like to have a better explanation on the section ”Create customised graph”
- Summarize the explanations and put an option to show the full explanation to not overload the page with text.
- No I do not
- I would add information about the person who created it and her. background, the references behind the information provided, and maybe a page with link to further material (videos, articles, books, ...)
- Although there solved exercises, I think that the explanation could use some visual explanation and not rely only on text.
- put in the examples of the algorithms the ordered lists in the steps that order or create a list
- Try to make clear where you are able to manipulate the graphs and were not
- if you have more time, you could have made a tab that could switch the language, or even design the website based on the knowledge of computer science fundamentals you possess. However, that is a rather hard task.
- Perhaps a section containing a glossary/collection of key terms used throughout the application.
- Creating a customised graph was difficult to figure out how this page was supposed to work.

- In addition for the user solve problems, it would be nice to have a function that lets you know at each step whether you have chosen correctly instead of waiting until I submit my full assignments."
- In the creation of graph section, I found it confusing that for the number of processors the + button was at the left of -
- When we can assign nodes to processors, it would be cool if we could interact with the graph dragging the processors directly to the nodes, instead of selecting the node from the list, and then selecting the processor
- For the timer game, it would be great if there would be something more visual like a progress bar or a circled timer rather than just the number in seconds

Any additional comments or feedback? Please share any other thoughts or observations you have about the application.

- In terms of UX and such, i would say it is a pretty clean and clear website, not hard to figure out if you know about this field.
- Ensuring mobile compatibility could broaden the application's accessibility, allowing users to engage with it on various devices. However, from an overall perspective, the application is functional and educational.
- Because I don't know much about these type of topics, i don't think i am the indicated to talk about wether if this was easy to understand or not. However, i can say for certain the interface and features it has makes it a very practical and enjoyable way to learn about Task Graph Scheduling.
- I would use this app if i had to solve this kind of problems on my job, it is really quick to use and learn, congrats!
- The page is amazing :DDDDDD
- I think the application is super practical and easy to use.
- Well done on the website!
- Overall the application looks very nice and would be very useful as a teaching aide.

Appendix C

Participants' information sheet

Participant Information Sheet

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator:	Professor Murray Cole
Researcher collecting data:	Lucia Jiang
Funder (if applicable):	N/A

This study was certified according to the Informatics Research Ethics Process, RT number **335342**. Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

Lucia Jiang is an MSc student in the School of Informatics, undertaking this study as part of their MSc project. The project is supervised by Professor Murray Cole.

What is the purpose of the study?

The project involves the design and implementation of one or more educational games intended for use by junior Informatics students. Each game aims to help the participant gain a better understanding of a classic Computer Science algorithm, by engaging with an interactive game.

Why have I been asked to take part?

As an Informatics student, we seek your views on various aspects of the designed games, and how they might be perceived by learners.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time without giving a reason, until the project has been submitted. After this point, personal data will be deleted and anonymised data will be combined such that it is impossible to remove individual information from the analysis. Your rights will not be affected. If you wish to withdraw, contact the PI. We will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

You will complete an electronic (or paper) questionnaire, asking for your opinions on properties of an algorithm based game or games (for example, attractiveness, difficulty, intuitiveness) as they might be perceived by junior informatics students. You will complete this in your own time, without supervision. You are not being asked to write the systems, merely to assess them.

Are there any risks associated with taking part?

There are no significant risks associated with participation.

Are there any benefits associated with taking part?

No, though we hope you will find the programs interesting and may learn something!

What will happen to the results of this study?

The results of this study will be summarised in the project report and presentation. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. Your data may be archived for a maximum of 1 year (in practice, only until the MSc dissertation, that will be informed by the data collected, has been submitted). All potentially identifiable data will be deleted within this timeframe if it has not already been deleted as part of anonymization.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team: Lucia Jiang and supervisor Professor Murray Cole.

All electronic data will be stored on any or all of a password-protected encrypted hard drive, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (eg Sharepoint) and any paper records will be stored in a locked filing cabinet in the PI's office. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the lead researcher, Professor Murray Cole (mic@inf.ed.ac.uk).

If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be made available on http://web.inf.ed.ac.uk/infweb/research/study-updates.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Professor Murray Cole (mic@inf.ed.ac.uk).

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

Appendix D

Participants' consent form

Participant number: 1

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date	Signature
Lucía Solera	19/07/2024	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH informatics

Participant number: 2

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Miao Ying Zhang Chen	19/07/2024	[signature]
Name of person taking consent	**Date**	**Signature**
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH informatics

Participant number: 3

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	✓	

Name of person giving consent	Date	Signature
Haoran CHeng	19/07/2024	

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	

THE UNIVERSITY of EDINBURGH
informatics

Participant number: 4

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1. I agree to take part in this study.

Yes	No
X	

Name of person giving consent	Date dd/mm/yy	Signature
Laura Rojo	19/07/2024	[signature]
Name of person taking consent	**Date**	**Signature**
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH
informatics

Participant number: 5

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1. I agree to take part in this study.

Yes	No
√	

Name of person giving consent	Date dd/mm/yy	Signature
Jimena Martín Reina	20/07/2024	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH
informatics

Participant number: 6

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

	Yes	No
1. I agree to take part in this study.	✓	

Name of person giving consent	Date	Signature
Yi Feng Zhang Chen	20/07/2024	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

Participant number: 7

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	√	

Name of person giving consent	Date dd/mm/yy	Signature
Jimena Martín Reina	20/07/2024	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH informatics

Participant number: 8

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Alicia Jiang	20/07/2024	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

Participant number: 9

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date	Signature
Xinru Huang	20/07/2024	

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	

Participant number: 10

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	✓	

Name of person giving consent	Date (dd/mm/yy)	Signature
Rujia Lin	20/07/2024	

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	

THE UNIVERSITY of EDINBURGH
informatics

Participant number: 11

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1. I agree to take part in this study.

Yes	No
✓	

Name of person giving consent	Date (dd/mm/yy)	Signature
Yuxuan Han Ma	21/07/24	韩雨萱
Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY of EDINBURGH informatics

Participant number: 12

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	Y	

Name of person giving consent	Date dd/mm/yy	Signature
Qi Nohr Chen	22/07/24	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH informatics

Participant number: 13

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Adrián Alonso Ledesma	22/07/2024	Adrian

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY of EDINBURGH informatics

Participant number: 14

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Jorge Torrejón López	24/07/2024	JT
Name of person taking consent Lucia Jiang	Date 18/07/2024	Signature [signature]

THE UNIVERSITY of EDINBURGH informatics

Participant number: 15

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Raúl Fernández	24/07/2024	Raúl F.

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY *of* EDINBURGH informatics

Participant number: 16

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Felipe Guo	**26/07/24**	[signature]
Name of person taking consent Lucia Jiang	Date 18/07/2024	Signature [signature]

THE UNIVERSITY *of* EDINBURGH informatics

Participant number: 17

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Daniel Henríquez	26/07/24	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

Participant number: 18

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	✓	

Name of person giving consent	Date dd/mm/yy	Signature
Richard Bai	28/07/2024	Richard

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY of EDINBURGH
informatics

Participant number: 19

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1. I agree to take part in this study.

Yes	No
X	

Name of person giving consent	Date dd/mm/yy	Signature
ROCIO BENITEZ	28/07/24	[signature]

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

Participant number: 20

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

	Yes	No
1. I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Paula Hidalgo	28/07/2024	

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	

THE UNIVERSITY *of* EDINBURGH
informatics

Participant number: 21

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Maria Lopez	29/07/24	ML

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

THE UNIVERSITY of EDINBURGH informatics

Participant number: 22

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

	Yes	No
1. I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Kira Smith	29/07/24	Kira

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	

Participant number: 23

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date (dd/mm/yy)	Signature
Tian Hu	29/07/24	TH

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[signature]

Participant number: 24

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	No
1.	I agree to take part in this study.	X	

Name of person giving consent	Date dd/mm/yy	Signature
Mellena Villanueva	29/07/24	

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	

Participant number: 25

Participant Consent Form

Project title:	Task Graph Scheduling as an Educational Game
Principal investigator (PI):	Professor Murray Cole
Researcher:	Lucia Jiang
PI contact details:	mic@ed.ac.uk

By participating in the study you agree that:

I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.

- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

		Yes	**No**
1.	I agree to take part in this study.	X	

Name of person giving consent	Date	Signature
Megan Morris	dd/mm/yy 06/08/2024	Megan Morris

Name of person taking consent	Date	Signature
Lucia Jiang	18/07/2024	[illegible]

THE UNIVERSITY *of* EDINBURGH informatics