|  | Day | |  | Month | |  | Year | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DATE** | 2 | 5 | **–** | 0 | 7 | **–** | 2 | 0 | 2 | 3 |

| **CodeFlow**  **Empowering Novice Programmers with Intuitive Coding using Flowchart** | |
| --- | --- |
| **Brief Description:**  Novice programmers and students often encounter difficulties in understanding programming concepts due to the abstract nature of code and complexities in creating logical program flows. Students can not understand how things works behind the scene. To bridge this gap, CodeFlow proposes an innovative web-based solution having intuitive drag-and-drop interface, enabling users to create program code by assembling elements of a flowchart, sophisticated algorithm for code-to-flowchart conversion accurately translates textual code into visual flowcharts, user can execute program with a clear representation of program logic. Moreover, memory map visualization, enabling learners to observe data storage and manipulation during program execution.  **Extended Abstract:**  **Problem:**  The field of programming holds immense potential and opportunities, but for novice learners, it presents a formidable challenge. Many aspiring programmers and students encounter difficulties understanding programming concepts when starting our coding journey. The abstract nature of code and the intricacies of creating logical program flows often leave students feeling overwhelmed and discouraged. These experiences reflect the broader issue of limited accessibility to programming education and the lack of intuitive tools to visualize code effectively.  Aspiring programmers face obstacles in comprehending how programming works behind the scene, how memory allocated and struggle to create effective logic. The absence of a user-friendly learning environment compounds the problem, hindering our progress and stifling our passion for programming.  **Related Work:**  **CODE AND FLOW**  Code and flow is a flowchart to program construction project developed in UCP but it has its limitations it was not able construct accurate code and can not handle complex logics. Moreover this project does not support code to flowchart conversion and memory map visualization.  **RAPTOR**  Raptor, a flowchart-based programming software designed to simplify coding for beginners. However, despite its benefits, Raptor has some limitations that may restrict its effectiveness for certain learners. Raptor's flowchart-based approach may not be directly transferable to traditional programming languages, which predominantly use textual code.  **Scratch**  A visual programming language developed by the MIT Media Lab. Scratch offers a graphical interface where users can drag and drop code blocks to create interactive stories, games, and animations. It is widely used in educational settings and provides a stepping stone for learners to transition from block-based programming to text-based languages.  **Proposed Methodology:**  Our proposed methodology for CodeFlow is a web based solution which create an optimal learning experience for novice programmers. The frontend development is centered around utilizing component-based architecture and interactive user interfaces where user can create program by drag-and-drop components of flowchart simplifying logic construction for learners. Additionally, the development process includes the creation of a sophisticated algorithm for code-to-flowchart conversion. This algorithm accurately translates textual code into visual flowcharts, offering learners a clear representation of program logic. Moreover, CodeFlow integrates memory map visualization during program execution. Learners can observe data storage and manipulation, gaining valuable insights into memory management concepts.  **Expected Results:**  By adopting CodeFlow, we anticipate several positive outcomes. Firstly, novice learners will find the platform more approachable, empowering them to overcome initial barriers and gain confidence in programming. The code-to-flowchart conversion feature will provide a visual representation of code, helping learners better comprehend program logic. This increased understanding is expected to lead to improved retention of programming principles and boost students' problem-solving skills.  The dynamic learning environment will foster creativity and encourage learners to explore programming concepts freely. As a result, CodeFlow users are expected to exhibit enhanced engagement and motivation in their learning journey. Furthermore, the memory map visualization will offer invaluable insights into program execution, equipping learners with essential knowledge of computer memory management.  **Conclusion:**  CodeFlow emerges as a transformative solution to the challenges faced by novice learners in programming education. The project's primary objective is to empower aspiring programmers by making programming concepts more accessible and comprehensible. The existing challenges of abstract code and complex logical flows experienced by students are effectively addressed through CodeFlow's innovative features.  Compared to existing solutions like Raptor, CodeFlow distinguishes itself through its web-based approach, an intuitive and interactive interface for code construction by drag-and-drop flowchart components. The platform's code-to-flowchart conversion algorithm accurately translates textual code into visual flowcharts, providing learners with a clear representation of program logic.  Furthermore, CodeFlow's integration memory map visualization, providing learners with valuable insights into data storage and manipulation during program execution. | **BS Program:**  **BSCS ☐**  **BSSE ☒** |
| **Term of Registration:**  **☒Fall \_\_\_2023\_\_\_**  **☐****Spring \_\_\_\_\_** |
| **Tools to be used:**  **VScode**  **GCC**  **GDB**  **C++**  **Reactjs** |
| **Project Type:**  **☐Research based**  **☐Hardware based/Embedded**  **☐Game based**  **☒Software Development**  **☐Artificial Intelligence (AI)**  **☐Mobile Application**  **☒Web Application**  **☐Robotics**  **☒Database**  **☐Other: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

**Particulars of the students:**

| **Sr. #** | **Registration#**  eg.**L1F00BSCS0101** | **Name in Full**  Use Block Letters | **Email**  Only UCP Email | **Contact #** | **CGPA** | **Signatures** |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | L1F20BSSE0191 | MUHAMMAD HASEEB NAWAZ | l1f20bsse0191@ucp.edu.pk | 03219173161 | 2.5 |  |
| **2** | L1F20BSSE0183 | MUHAMMAD MUJEEB | l1f20bsse0183@ucp.edu.pk | 03354140322 | 2.25 |  |
| **3** | L1F19BSSE0073 | ROHAN QAMAR | l1f19bsse0073@ucp.edu.pk | 03101444965 | 2.43 |  |

**Name and Signatures of the advisor**:

**Mohsin Sami**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |

**For Project Office use only**

| Remarks: | | Signatures and Date  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Manager Projects** |
| --- | --- | --- |
| Group No |  |

**UNDERTAKING FOR UNDERGRADUATE FINAL YEAR PROJECTS**

**Acceptance of Project Idea**

Students will be required to defend their idea before the scrutiny committee (SC) which has the authority to accept or reject the project idea. The decision taken by the SC will be final and cannot be challenged.

Similarly, in phase wise evaluations, the marks awarded by the evaluators will be considered final. No excuses on their skill, relevancy, competency or biasedness will be acceptable as an absolve.

**Issuance of hardware**

Due to the prevailing COVID 19 pandemic situation, it may not be possible to acquire any kind of specific hardware. This situation is due to the import/export problems faced by all countries and it can't be predicted when the situation will improve. Therefore, it is highly recommended that only the hardware which is readily available in Lahore (Hall Road) may be requested for the projects. Furthermore, if the hardware is not available, the HOD may recommend change in the project scope.

The Project Office will facilitate the requestor only in initiating the procurement process. However, Project Office bears no liability if the hardware is not procured timely as it beyond its purview. It shall be sole responsibility of the project advisor or the group itself to arrange the required hardware for the project if predominantly essential. Furthermore, the problem(s) of procurement of necessary hardware will not be considered admissible plea to revise the awarded grades.

A group/advisor is bound to collect procured hardware within two days on intimation from the project office and return within a week of grade notification.

**Supervisory Meetings**

A group is required to hold at least two meetings per month and maintain meeting minutes. Missing two consecutive meetings without any notice may lead to withdrawal of the project.

I, Mr. **MOHSIN SAMI**, solemnly declare that I have read and understood the above mentioned instructions and shall abide by these in both letter and spirit.

Project Title: **CodeFlow**

**MOHSIN SAMI**

Advisor‘s Name & Signature

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date:24-07-2023

Student 1: Student 2: Student 3:

Name: Name: Name:

##### MUHAMMAD HASEEB NAWAZ MUHAMMAD MUJEEB ROHAN QAMAR

Registration# Registration# Registration#

L1F20BSSE0191 L1F20BSSE0183 L1F19BSSE0073

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature Signature Signature