|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PROJECT / GROUP NAME** | **Group 14** | | | |
| **Start Date** | 2024.04.01 | **Finish Date** | 2024.07.01 | |
| **Aim /**  **Objective** | Implementation of front-end information acquisition function: Carry out code development in parallel and implement front-end functions and interfaces according to project design documents and architectural design. This work package will realize the functions of community service inquiry, educational resource integration, and sustainable development guide. Combine architectural design and back-end development to ensure interface consistency. | | | |
| **Work package**  **Manager** | Wang Shizheng | | | |
| **Contributors to this package** | Sun Ruotong  Liu Xinran  Qi Te  Liu Huiyang  Zhang Juncheng | | | |
| **Description /**  **Activities** | Task 3.1 Implement interface   * 3.1.1 Design request interception rules and add request header information appropriately * 3.1.2 Complete the front-end request api according to the interface document * 3.1.3 Find relevant articles to display as demos   Task 3.2 Implement the sustainable development guide function   * 3.2.1 Draw page elements based on design draft * 3.2.2 Write logic for functions * 3.2.3 Properly display the obtained back-end data * 3.2.4 Find relevant articles to display as demos   Task 3.3 Implement community service query function   * 3.3.1 Draw page elements based on design draft * 3.3.2 Handle query requests appropriately * 3.3.3 Find relevant articles to display as demos   Task 3.4 Realize the integration function of community education resources   * 3.4.1 Draw page elements based on design draft * 3.4.2 Write logic for functions * 3.4.3 Find relevant articles to display as demos   Task 3.5 Optimize completed code   * 3.5.1 Some code may need to be refactored   Task 3.6 Joint debugging of front-end and back-end   * 3.6.1 Accept errors reported by testers and analyze whether they are front-end code problems * 3.6.2 Assist with backend staff to fix bugs | | | |
| **Milestones** |  | | | Week |
| M 3.1 Normally around week 4/5  M 3.2 Normally around week 6/7  M 3.3 Normally for Interim report (week 9): What can the overall application do now? How has this work package contributed so far?  M 3.4 Normally around week 11  M 3.5 Normally around week 13  M 3.6 Final Release | | | **x**  **x**  **x**  **x**  **x**  **x** |
| **Deliverables** |  | | | Week |
| D 3.1 These are deliverables – deliverables are normally pieces of code (complete) that are integrated into the overall project  D 3.2 Normally deliverables coincide (happen at the same time) with milestones, but not every milestone has to have a deliverable. For instance, if a milestone is testing a work package code, there may be no deliverable. However if a milestone involves integrating a work package code with the overall project, that would be a deliverable. | | | **x**  **x** |