PROJECT / GROUP NAME	**Group 14**		
Start Date	2024.03.04	**Finish Date**	2024.7.10
Aim / Objective	Project design: The purpose of this work package is to think about what problems need to be solved from the customer's perspective and to provide solutions to these problems from the software development team's perspective. It is necessary to provide clear project description documents, complete processes for specific functions, precise UI design, etc. Later, the architecture design will be connected and some details will be explained and corrected.		
Work package Manager	Liu Xinran		
Contributors to this package	Sun Ruotong Wang Shizheng Qi Te Liu Huiyang Zhang Juncheng		
Description / Activities	Task 1.1 Define user requirements • 1.1.1 Gather ideas from team members on possible project topics through brainstorming. • 1.1.2 Discuss the feasibility and relevance of each option to the topic. • 1.1.3 Vote anonymously for the most suitable solution. • 1.1.4 Convert the plan into requirements and specify the functions that may need to be implemented. Task 1.2 Specify functionalities • 1.2.1 Hold a brainstorming to think about all the functional according to the user requirement. • 1.2.2 Evaluate the feasibility and necessity of each function. • 1.2.3 Refined description of all the functions involves, such as combining a function with the corresponding user story. • 1.2.4 Check duplications and conflicts among functions. Task 1.3 Design UI • 1.3.1 Determine the overall website structure and components. • 1.3.2 Determine the theme style and color scheme. • 1.3.3 Design the individual interface for each function. • 1.3.4 Determine the size and proportion of each element. • 1.3.5 The design draft is finally presented using design software such as Figma and public to everyone on the team. Task 1.4 Optimize UI • 1.4.1 Check the correspondence between functions and designs, ensuring that each function has a demonstration in UI design.		
Milestones			Week

	M 1.1 Complete project design and determine tasks	**2**
	M 1.2 Determine the detailed function of each task	**6**
	M 1.3 Complete prototype UI design draft	**9**
	M 1.4 Complete the detailed UI design, including all functions	**11**
	M 1.5 Confirm that the UI design contains all functions	**16**
Deliverables		Week
	D 1.1 Deliver idea of project design and task description documents	**2**
	D 1.2 Deliver detailed project design description, including documentation of all functions	**6**
	D 1.3 Deliver project UI design draft	**11**
	D 1.4 Deliver the completed UI design	**16**

PROJECT / GROUP NAME	**Group 14**		
Start Date	2024.03.18	**Finish Date**	2024.07.10
Aim / Objective	Architecture and interface design: Connect with the project design department, review project design-related documents from the perspective of the development team, and discuss the feasibility of solutions and functions and more details. Design the connections and details of each part of the entire software from the architectural level to ensure high cohesion and low coupling of each part of the software, while maintaining high scalability to cope with changing needs. Provide complete interface documents and database design to connect subsequent code development processes and ensure the feasibility of architectural design.		
Work package Manager	Sun Ruotong		
Contributors to this package	Liu Xinran Wang Shizheng Qi Te Liu Huiyang Zhang Juncheng		
Description / Activities	Task 2.1 Evaluate function feasibility •2.1.1 Re-read the existing feature list. •2.1.2 Evaluate the feasibility and necessity of function from a software engineer's view. •2.1.3 Re-discuss the unclear functions and deal with the controversy. Task 2.2 Conduct functional descriptions •2.2.1 Explain the tasks that have a specific meaning in the project. •2.2.2 Further explain functions as engineering description. •2.2.3 Use tools like flowcharts to describe functions with complex interaction logic. Task 2.3 Design interface documentation •2.3.1 Design interfaces for user registration, login, etc. •2.3.2 Design the interface for displaying articles. •2.3.3 Design interfaces for other functions. o 2.3.3.1 Determined the data involved in the function o 2.3.3.2 Map variables to the front and back ends •2.3.4 Organize interfaces and find identical or similar interfaces to merge and split. Task 2.4 Design database •2.4.1 List the data required for each function. •2.4.2 Combine identical data and assign easy-to-understand and unique variable names to the data. •2.4.3 Abstract data relationships and create entity relationship diagrams. •2.4.4 Generate ER diagrams according to relational database tables. •2.4.5 Test database availability and reliability Task 2.5 Select technology stack		

	•2.5.1 Determine the appropriate technology stack selection based on the knowledge mastered by team members and combined with project functional requirements.	
Milestones		Week
	M 2.1 Confirm that all functions are feasible and necessary	**5**
	M 2.2 Map all functional descriptions into professional terms	**7**
	M 2.3 Draw a flow chart	**9**
	M 2.4 Organize the required interfaces and design all interface	**11**
	M 2.5 Design, create, and implement database	
	M 2.6 Determine technology selection and technology stack	**13**
		16
Deliverables		Week
	D 2.1 Deliver a complete flowchart or ER diagram	**9**
	D 2.2 Deliver complete interface documentation	**11**
	D 2.3 Deliver a complete design database	**13**
	D 2.4 Delivery technology selection and technology stack description	**16**

PROJECT / GROUP NAME	**Group 14**		
Start Date	2024.04.01	**Finish Date**	2024.07.10
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 Interface routing design completed	**5**
	M 3.2 Successfully displayed and obtained backend data	**7**
	M 3.3 Successfully implemented community service inquiry function	**10**
	M 3.4 Realize the integration function of community education resources	**12**
	M 3.5 Optimize and refactor existing code	**14**
	M 3.6 Complete front-end and back-end joint debugging	**16**
Deliverables		Week
	D 3.1 Deliver front-end API part code	**6**
	D 3.2 Deliver and implement part of the code for sustainable development guidance	**8**
	D 3.3 Deliver part of the community service query function code	**10**
	D 3.4 Deliver community education resource integration functions	**12**
	D 3.5 Deliver optimized and refactored code	**14**
	D 3.6 Deliver all back-end code after joint debugging	**16**

PROJECT / GROUP NAME	**Group 14**		
Start Date	2024.04.01	**Finish Date**	2024.07.10
Aim / Objective	Front-end interactive function logic implementation: Carry out code development in parallel and implement front-end functions and interfaces according to project design documents and architectural design. This work will realize the functions of user system, owner resource management, and community employment platform. Combine architectural design and back-end development to ensure interface consistency.		
Work package Manager	Qi Te		
Contributors to this package	Sun Ruotong Liu Xinran Wang Shizheng Liu Huiyang Zhang Juncheng		
Description / Activities	Task 4.1 Implement interface •4.1.1 Design request interception rules and add request header information appropriately •4.1.2 Complete the front-end request api according to the interface document Task 4.2 Implement user system functions •3.2.1 Split user-related functionality into appropriate components •3.2.2 Draw page elements according to design draft •3.2.3 Realize the functions of each component •3.2.4 Test functions such as user registration and login Task 4.3 Realize owner resource management function •4.4.1 Obtain owner-related content data •4.4.2 Use charts to demonstrate •4.4.3 Call the artificial intelligence module to analyze some data and finally give reasonable suggestions. Task 4.4 Realize community employment platform function •4.4.1 Obtain existing recruitment information •4.4.2 Display recruitment information •4.4.3 Collect job application information filled in by users and send it to the backend Task 4.5 Optimize completed code •4.5.1 Some code may need to be refactored Task 4.6 Joint debugging of front-end and back-end •4.6.1 Accept errors reported by testers and analyze whether they are front-end code problems •4.6.2 Assist with backend staff to fix bugs		
Milestones			Week

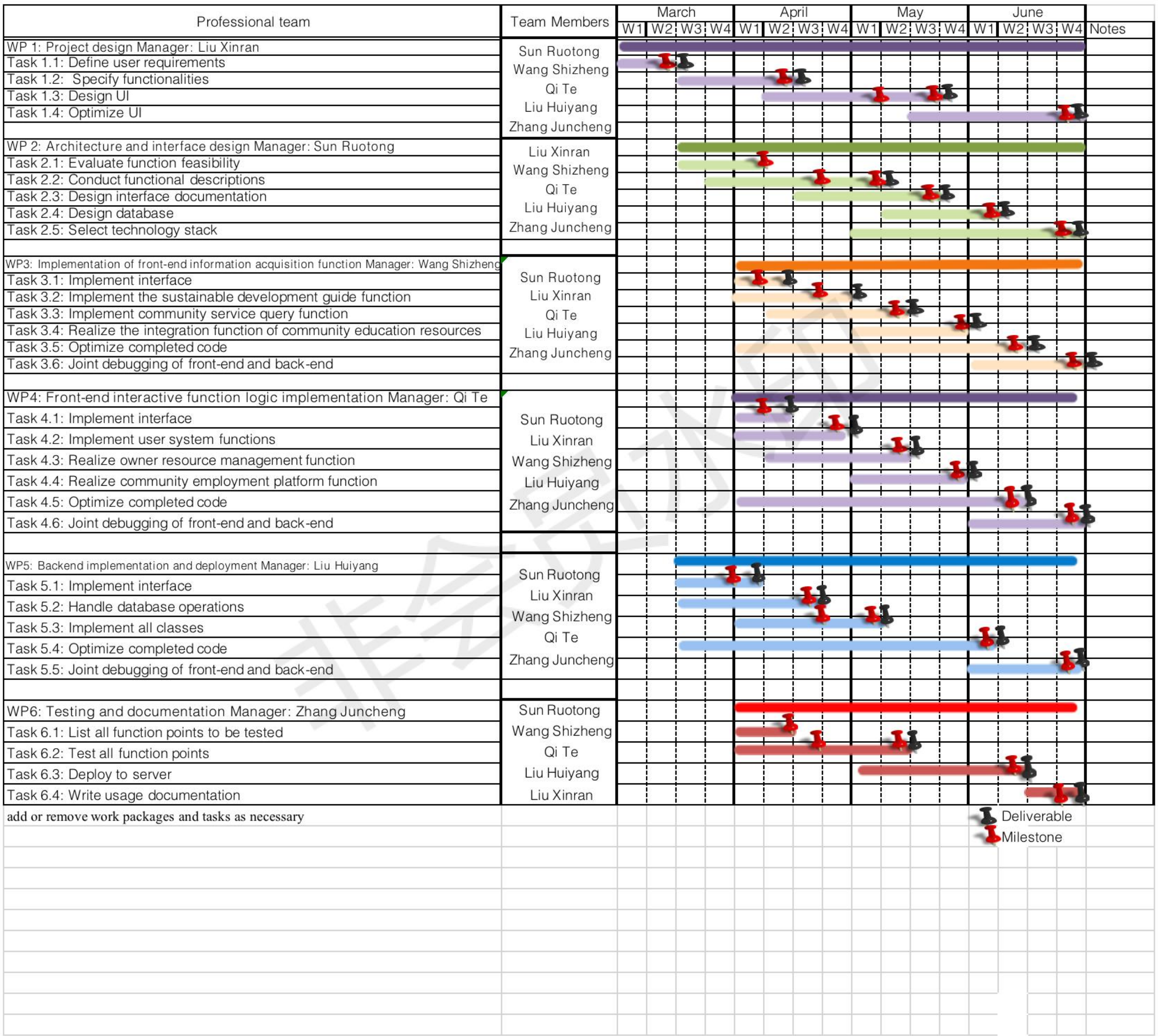
	M 4.1 Interface routing design completed	**5**
	M 4.2 Successfully implemented user system functions	**8**
	M 4.3 Successfully implemented owner resource management function	**10**
	M 4.4 Successfully implemented community employment platform function	**12**
	M 4.5 Optimize and refactor existing code	**14**
	M 4.6 Complete front-end and back-end joint debugging	**16**
Deliverables		Week
	D 4.1 Deliver front-end API part code	**6**
	D 4.2 Deliver part of the system function code to the user	**8**
	D 4.3 Deliver part of the code that implements the owner resource management function	**10**
	D 4.4 Deliver part of the community employment platform function code	**12**
	D 4.5 Deliver optimized and refactored code	**14**
	D 4.6 Deliver all back-end code after joint debugging	**16**

PROJECT / GROUP NAME	**Group 14**		
Start Date	2024.03.25	**Finish Date**	2024.07.10
Aim / Objective	Backend implementation and deployment: Carry out code development in parallel, and implement back-end functions and interfaces based on project design documents and architecture design. Design and implement back-end code structure based on project architecture and entity relationships to ensure code readability and high performance. Connect architecture design and front-end development to ensure interface consistency.		
Work package Manager	Liu Huiyang		
Contributors to this package	Sun Ruotong Liu Xinran Wang Shizheng Qi Te Zhang Juncheng		
Description / Activities	Task 5.1 Implement interface •5.1.1 Design backend routing paths •5.1.2 Complete the back-end request api according to the interface Task 5.2 Handle database operations •5.1.1 Check and test whether relational database tables meet back-end usage requirements •5.1.2 Establish an appropriate connection with the database and encapsulate basic operations such as addition, deletion, modification, and query Task 5.3 Implement all classes •5.3.1 Establish corresponding simple class diagrams based on the ER diagram and functional description given in the architectural design •5.3.2 Implement code according to the designed class diagram •5.3.3 Maintain high cohesion and low coupling of class implementation while maintaining code readability and scalability Task 5.4 Optimize completed code •5.1.1 Some code may need to be refactored Task 5.5 Joint debugging of front-end and back-end •5.5.1 Accept errors reported by testers and analyze whether they are back-end code problems •5.5.2 Assist with frontend staff to fix bugs		
Milestones		Week	
	M 5.1 Interface routing design completed M 5.2 Test database meets requirements M 5.3 Design class diagram based on ER diagram M 5.4 Implement and optimize all classes M 5.5 Optimize and refactor existing code	**4** **7** **7** **9** **13**	

	M 5.6 Complete front-end and back-end joint debugging	**16**
Deliverables		Week
	D 5.1 Deliver the backend api part of the code D 5.2 Deliver a database that establishes a connection to the backend D 5.3 Deliver all backend classes D 5.4 Deliver optimized and refactored code D 5.5 Deliver all back-end code after joint debugging	**5** **7** **9** **13** **16**

PROJECT / GROUP NAME	**Group 14**		
Start Date	2024.04.08	**Finish Date**	2024.07.01
Aim / Objective	Testing and documentation: This work requires comprehensive testing against the function points in the original project design to ensure the usability and fault tolerance of each function. Propose the problematic parts as early as possible and provide a rough modification plan, and notify the team members responsible for this part to repair them. After the software has been tested and no problems exist, a usage document will be written to explain how to use each part of the function. Finally, the software needs to be deployed to the server		
Work package Manager	Zhang Juncheng		
Contributors to this package	Sun Ruotong Liu Xinran Wang Shizheng Qi Te Liu Huiyang		
Description / Activities	Task 6.1 List all function points to be tested •6.1.1 Automate testing of content that can be automated •6.1.2 Design multiple interactive tests for function points with interactive requirements Task 6.2 Test all function points •6.2.1 When a function point that does not meet expectations occurs, record all relevant test conditions to ensure the reproducibility of the exception. •6.2.2 Remind team members who develop corresponding function points to make repairs or improvements Task 6.3 Deploy to server •6.3.1 Ensure that local testing of software no longer performs unexpectedly •6.3.2 Package and deploy the software to the given server •6.3.3 Perform functional testing again and repeat Task 6.2 when problems arise Task 6.4 Write usage documentation •6.4.1 Declare the development environment and deployment environment of the entire software •6.4.2 Compare the listed function points for a detailed introduction •6.4.3 Reread the document, look for content that is difficult to understand or make sense, and rewrite that part of the document		
Milestones			Week
	M 6.1 List all test points and design interactive test plans M 6.2 Test all function points M 6.3 Confirm that no accidents will occur during local testing		6 **7** **10**

	M 6.4 Deploy to server M 6.5 Complete usage documentation	**14** **16**
Deliverables		Week
	D 6.1 Deliver tests and correct improved function points and code D 6.2 Deliver successfully deployed project products D 6.3 Deliver usage documentation including development and deployment environment, etc.	**10** **14** **16**



Professional team	Team Members
WP 1: Project design Manager: Liu Xinran	Sun Ruotong Wang Shizheng Qi Te Liu Huiyang Zhang Juncheng
Task 1.1: Define user requirements	
Task 1.2: Specify functionalities	
Task 1.3: Design UI	
Task 1.4: Optimize UI	
WP 2: Architecture and interface design Manager: Sun Ruotong	Liu Xinran Wang Shizheng Qi Te Liu Huiyang Zhang Juncheng
Task 2.1: Evaluate function feasibility	
Task 2.2: Conduct functional descriptions	
Task 2.3: Design interface documentation	
Task 2.4: Design database	
Task 2.5: Select technology stack	
WP3: Implementation of front-end information acquisition function Manager: Wang Shizheng	Sun Ruotong Liu Xinran Qi Te Liu Huiyang Zhang Juncheng
Task 3.1: Implement interface	
Task 3.2: Implement the sustainable development guide function	
Task 3.3: Implement community service query function	
Task 3.4: Realize the integration function of community education resources	
Task 3.5: Optimize completed code	
Task 3.6: Joint debugging of front-end and back-end	
WP4: Front-end interactive function logic implementation Manager: Qi Te	Sun Ruotong Liu Xinran Wang Shizheng Liu Huiyang Zhang Juncheng
Task 4.1: Implement interface	
Task 4.2: Implement user system functions	
Task 4.3: Realize owner resource management function	
Task 4.4: Realize community employment platform function	
Task 4.5: Optimize completed code	
Task 4.6: Joint debugging of front-end and back-end	
WP5: Backend implementation and deployment Manager: Liu Huiyang	Sun Ruotong Liu Xinran Wang Shizheng Qi Te Zhang Juncheng
Task 5.1: Implement interface	
Task 5.2: Handle database operations	
Task 5.3: Implement all classes	
Task 5.4: Optimize completed code	
Task 5.5: Joint debugging of front-end and back-end	
WP6: Testing and documentation Manager: Zhang Juncheng	Sun Ruotong Wang Shizheng Qi Te Liu Huiyang Liu Xinran
Task 6.1: List all function points to be tested	
Task 6.2: Test all function points	
Task 6.3: Deploy to server	
Task 6.4: Write usage documentation	
add or remove work packages and tasks as necessary	

Deliverable

Milestone