APRIL 30, 2024

DEVELOPMENT PLAN

DISHYDISHES

Revisions

Date	Revision No.	Authors	Comments
Oct 25, 2023	Revision 0	Jie Zhang Zirui He Shuren Xu Kaicheng Xu	Plan initialized
Dec 1, 2023	Revision 1	Zirui He	Updates based on feedback received from POC
March 30, 2024	Revision 2	Shuren Xu Zirui He	Final update based on actual progress

1. **Team Meeting and Communication Plan**
   The team is going to hold a 1–2-hour weekly meeting to delegate tasks among members and share everyone's progress on Wednesday or Thursday. This meeting is also where we ask questions and help each other.
   We heavily rely on Microsoft suite to communicate; documents are mostly shared via MS Teams.
   We will also use the Issue Board in GitLab to keep track. Gitlab issue board will be utilized as a Kanban to visualize the software development workflow as a series of stages, such as open, in-progress, testing, and closed.

2. **Team Member Roles**
   a. Responsible people for each of the functional and non-functional requirements
      i. Backend features

Priority	Features		Notes	
			Responsible people	Expected Date
P0	User Registration/Login (Users database creation)		Shuren Xu	Nov 17, 2023
P0	Uploading recipes (Recipes database creation)		Kaicheng Xu	Nov 17, 2023
P0	Search functionality	By food name	Kaicheng Xu Shuren Xu	Nov 17, 2023
		By user alias		
P0	Getting recipe content by ID		Kaicheng Xu	Nov 17, 2023
P1	Commenting on a recipe (Comments database creation)		Shuren Xu	Mar 21, 2024
~~P1~~	~~Rating a recipe~~		Rating related features are deprecated as they are redundant to bookmark features to some extent.	
P1	Bookmarking a recipe as favorite		Kaicheng Xu Shuren Xu	Mar 21, 2024
P1	Recipes recommendation	6 most liked posts last week	Shuren Xu	Mar 21, 2024
		6 random posts in past 24 hours		
		6 most recent posts from user's followees		
P1	Tagging recipes by types/categories when creating		This became a part of uploading recipes, and we leave the flexibility to users to tag their	

				recipes.
P1	Getting recipes by tag		Kaicheng Xu	Mar 21, 2024
P1	Search functionality	By ingredients	Kaicheng Xu	Mar 21, 2024
P1	Getting the user's	personal info	Shuren Xu	Mar 21, 2024, except getting comments from a user as we doubt if it's necessary.
		recipe posts		
		comments		
		~~rates~~		
		bookmarked recipes		
		preference		
P2	Advanced Search functionality	Filtering out recipe posts based on religion and/or allergies	Sorry we don't have time to finish these features.	
P2	Updating a recipe			
P2	Deleting a recipe		Kaicheng Xu	Mar 30, 2024
P2	Updating personal preference (e.g., religion and/or allergies)		Shuren Xu	Mar 30, 2024
P2	Updating personal info (e.g., alias, profile picture, …)		Shuren Xu	Mar 30, 2024
P2	Edit recipe visibility (By site admin/reviewer)		Sorry we don't have time to finish these features.	
P2	Edit comment visibility			
P2	Notify users about their recipe/comment visibility change			
P2	Deleting a comment		Shuren Xu	Mar 30, 2024
~~P2~~	~~Deleting a rating~~		Rating related features are deprecated as they are redundant to bookmark features to some extent.	
~~P2~~	~~Changing a rating~~			
P2	Remove a recipe from favorites		Shuren Xu	Mar 30, 2024
P3	Commenting on a comment		Sorry we don't have time to finish this feature.	
P3	Thumb-up to a comment		Shuren Xu	Mar 30, 2024

P3	Notify user when their recipes get favorited or commented, or comments get commented or thumbed-up	Sorry we don't have time to finish these features.	
P3	Attaching pictures to a comment		
P3	Social media sharing of recipes		
P3	Following another user	Shuren Xu	Mar 30, 2024
P3	Getting who a user is following/followed by	Shuren Xu	Mar 30, 2024
P3	Calories Estimation	Sorry we don't have time to finish these features.	
P4	Recommendations based on user behavior		
P4	Auto tagging		

ii.   Frontend features

Priority	Features		Responsible People	Expected Date
P0	User Registration/Login Page		Jie Zhang	Static page by Nov 17, 2023, integration with backend by Mar 21, 2024
P0	Nav bar	On the left is the logo, by clicking on which user will be redirected to the main page	Jie Zhang	Static page by Nov 17, 2023, integration with backend by Mar 21, 2024
		Search field to query recipes with name, ingredients, or user alias		
		On the right is the user profile picture, by clicking on which user will be redirected to 1. the user registration/ login page if not logged in, 2. or to the user portal after logged in		
P0	Main Page	A List of types/categories	Jie Zhang	Static page by Nov 17, 2023, integration with backend by Mar 21,
		A carousel to list 6 most liked posts last week		
		A dashboard to show 6 random new posts in past 24 hours		

		A + sign button to upload a recipe		2024
P0	Recipe creation UI	The form to upload a recipe's name, description, images, ingredients, and steps	Zirui He	Static page by Nov 17, 2023, integration with backend by Mar 21, 2024
P0	Recipe page	Starting with the recipe name as title and a background photo	Zirui He	Static page by Nov 17, 2023, integration with backend by Mar 21, 2024
		Description section		
		A heart-shape button to favorite this recipe, ~~and a 5-star rating section~~.		
		A block on the right side stating current info of this recipe, including number of likes, review scores, uploading time…		
		A List showing every ingredient and their usage amount		
		Presentation of each step with picture		
		Corresponding tags section		
		Comment section		
P1	Search View Page	Display search results (recipes by names, tags, ingredients; users by alias)	Jie Zhang	Static page by Nov 17, 2023, integration with backend by Mar 21, 2024
P1	User Portal Page	personal info	Shuren Xu Zirui He Jie Zhang	Static page by Nov 17, 2023, integration with backend by Mar 21, 2024
		recipe posts		
		comments		
		~~rates~~		
		bookmarked recipes		
		preference		
P2	Main Page	2 rows for common searched tags	Jie Zhang	Mar 21, 2024
		6 most recent posts from user's followees		
P2	All the changes accommodating to P2 backend		Zirui He	Mar 30, 2024,

	features	Jie Zhang	except updating preference
P3	All the changes accommodating to P3 backend features	Zirui He	Mar 30, 2024

   b. Coordinator/program manager
   Our team is a compact one of only 4 people. We all engage closely and actively so no manager is needed.

3. **Workflow Plan**
   a. GitLab Usage
      i. Branch usage:
         - The main branch remains stable and clean. Only well-tested code is allowed to merge into it.
         - Every time we'd like to develop a new feature, we ought to create a new branch where we can work on. Once the code gets verified and merged, this branch can get deleted. Naming convention feature/xxx is recommended.
         - A bug-fix branch to fix bugs may also be created if necessary. Like feature branches, we can delete it after the fix is done. Naming convention hotfix/xxx is suggested in this case.
      ii. Pull request:
         - When the code in secondary branch is ready, the coder should create a pull request to merge the code to main branch.
         - To safeguard integrity and reliability, direct commits skipping pull requests is not permitted.
         - Team members are responsible for reviewing each other's code. A merge shall happen only if receiving more than 1 approval.
      iii. Issue management:
         We will keep each other posted to make sure no conflicts happen when we commit changes to GitLab. But if conflicts do appear, there will be emergency meetings among the coders who commit conflicting files to work this out and not interfere with other coders' progress.
   b. Agile
   Though we don't intend to follow scrum/sprint methodology strictly, we are going to convey weekly meetings to monitor our project, bring in tasks from backlog to the plate, and integrate modifications. We shall also regularly review and reflect on completed work. In this fashion, a team is capable of learning from feedback and then strategizing upcoming tasks.
   c. Storage of users' data

We will use a google cloud services to store users' data and other data if applicable.

d. What tool/method is used to achieve each of the requirements and achieve the performance metrics that were proposed in your SRS?

   i. For functional requirements, we assign the priority tasks to each member, and test them using vitest and JUnit.

   ii. For non-functional requirements, we check our content regularly to ensure it meets our cultural, security and privacy, and look and feel requirements.

   iii. For speed and performance requirements, we test it manually after each stage of development. We expect our platform to respond to user interactions within 100ms and load time shall be within 3s.

4. **Proof of Concept Demonstration Plan**

We will try to implement P0 priority features for both frontend and backend, and integrate them together so that a demo will be there for presentation. If time permits, we will attempt to establish mock web pages for front-end features with P1 priority.

5. **Technology**

a. Specific programming language (front end and backend), coding environment, unit testing framework.

   We are planning to use Vue.js 3, an open source MVVM JavaScript framework for our front end. The performance and usability of Vue.js has been testified by its 40.9k stars on GitHub, and two of our group members have accumulated profound knowledge during COOP experiences. For the back end, we plan to take advantage of the simplicity of Spring-boot, which is a Java based framework and increases the development efficiency by minimizing the amount of configuration compared to Spring. As for unit testing framework, we would like to use vitest and JUnit, as they are the most popular unit testing framework for our front end and backend with faster iterations providing more powerful functionalities and up-to-date document for us to reference.

b. ML libraries

   We do not intend to use machine learning libraries of any kind, since our purpose for this project lies in delivering seamless well-refined user experience in an area where it lacks. Including machine learning libraries could be a future goal for our project after its core element has been implemented and polished. At this time, we would focus on items documented in our current design.

c. The usage of GPU or any other relevant technology aspects.

   We will not use GPU or require any additional technology configurations since this project is web-based and should be compatible in most scenarios. Moreover, while there are graphic libraries like WebGL, our project does not require heavy duty graphic rendering, simple HTML+CSS (perhaps some CSS libraries like ant design vue or ElementUI too) would suffice.

6. Project Scheduling

   Refer to Gantt Chart shared by our team.