1. Introduction

1.1Purpose

The document is an official statement of what the system developers should implement. It includes both the user requirements for a system and a detailed specification of the system requirements. The document is essential for outside contractor who developing the software system. It is also useful to write a short supporting document that defines the business and dependability requirement for the system. The document is for system customers, managers, system engineers, system test engineers, system maintenance engineers.

1.2 Scope

Gitgroup design for people who has less development experience such as project manager, project members and people who are related to the project. It is an easy access online application that served for product management team based on Github API.

Our online application sets up by several software development tools. In langrage part, we use both TypeScript and JavaScript. TypeScript is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [programming language](https://en.wikipedia.org/wiki/Programming_language).It is a strict syntactical [superset](https://en.wikipedia.org/wiki/Superset) of [JavaScript](https://en.wikipedia.org/wiki/JavaScript), and adds optional static typing to the language.(具体怎么用的用在什么功能上)Javascript is a language which is also characterized as [dynamic](https://en.wikipedia.org/wiki/Dynamic_programming_language), [weakly typed](https://en.wikipedia.org/wiki/Weak_typing), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) and [multi-paradigm](https://en.wikipedia.org/wiki/Multi-paradigm_programming_language).（同上）For frontend and backend, we used ReadctJS(Framework), Bulma(CSS Framework), Frontawesome(Icon Set and Toolkit) and NodeJS(JS Engine), TS-Node(TypeScript execution and REPL for node.js), ExpressJS(Framework), Mongoose(ODM) in different part. In the test process, our tool is Jest. It is a Javascript testing software.(选这个test工具的原因，和其他的test工具相比较有什么优势) Also we regard MongoDB as our host DBMS, which is one of the most popular database management tool. Others like Git, Heroku, VSCode, Postman, are also included in developing process.

1.3 Definitions, acronyms, and abbreviations

|  |  |
| --- | --- |
| Owner | Users who set up the project or mainly charge of the project |
| Project | Users who intended to plan and undertake |
| Collaborator | Users who assists owner or project participants |
| Repository（Github） |  |
| Kanban(GitGroup) | Users often set up tasks on |
| KanbanColumn(GitGroup) | A set of Kanban |
| Issue |  |
| Card |  |
| Chart | Tool which helps managers track process |
| Chat | Function which make users barrier free to communicate |
| Notification | A reminder function for users |
| Access token(in sequence diagram) |  |

1.4 Reference

*IEEE Recommended Practice for Software Requirements Specifications IEEE std 830-1998(revision of IEEE stf 830-1993)*

1.5 Overview

The rest of the document will include overall description of Gitgroup in following perspectives. Product perspective, product functions, user characteristics, operating environment, design and implementation constraints, assumptions and dependencies. For interface requirement, we put efforts on user interfaces, hardware interfaces, software interfaces and communications interfaces. we will also include requirements like user requirements, system requirements, functional requirements, non-functional requirements and system models like use case diagram, activity diagram, sequence diagram, class diagram, system architecture, web design.

2.overall description

This section of the SRS will include general factors that affect the product and its requirements. This section does not state specific requirements. Instead, it provides a background for those requirements, which are defined in detail in Section 3 of the SRS, and makes them easier to understand.

2.1 Product perspective

The background of the Gitgroup coming from project managers are often busy to managing multiple projects at the same time. Sometimes, they are not familiar with details with products especially in technology perspective. However, Git and Github have a high entrance standard which project manager hard to manipulate it. For example, when you want to create an issue to suggest a new idea or track a bug, you need to learn markdown syntax, emoji code. And it is even more complex when you do pull request and merge the project. Comparing with Git and Github, GitGroup is an easier access tool . People who has little development experience, such as product manager. They also can handle GitGroup easily. The only thing they need to do is sign up an account and access to the GitHub api. The user can create online group chat, schedual and resord conference. After finding project idea, our online app also provide the task management service. In each project, every user will have their own grade. Furthermore, the online web can have a recommendation system which help the product manager and programmer match each other. In conclusion, it is a online service based on the GitHub API, and improve the user experience.

* 1. Product functions

Our product focus on easy access, clearly, convenience, efficiency. Functions are surrounded our goals. Mainly function lists in the following.

* Online meeting: Online meeting group chat. Record the conference. Make conference scheduling form and alarming the coming meeting.
* Task management: The function is Like GitHub task management, but Gitgroup has better UI design. Make TODO list also included in task management, users create an issue to suggest a new idea or track a bug by means of setting up a TODO list. Then user can organize and assign tasks to their team members.
* Developer grading system: Record the working of any team member. According the performance of users in all team which he joined, get users a grade;
* Developer finding system: If you are a team leader, you can find a perfect developer who has related skill to your project. If you are a software developer, you can post your information to find suitable project for you.
  1. User characteristics

Gitgroup consists of 2 user flows, one for owner, the other is collaborator. Features for owners and collaborators lists in the following. As we mentioned before, GitGroup is an easy access online application. As long as they are familiar with project. They can set up tasks on Kanban, trace project progress on Chart, chat with team members on Chat.

For owner, they can

* Invite collaborator
* Remove repository
* Set up tasks on Kanban board
* Adjust task status
* Delete Kanban board
* View chart
* Import issue card from Github
* In-team chat

For collaborator, they can

* Set up tasks on Kanban board
* Adjust task status
* View chart
* Import issue card from Github
* Tracing residual tasks
* Tracing finished tasks
* Tracing throughput volume
* Tracing work velocity
* Chat with owner

A high-level diagrammatic representation of the user activity is depicted below

* 1. Constraints

This subsection of the SRS should provide a general description of any other items that will limit the devel- oper’s options. These include

1. a)  Regulatory policies;
2. b)  Hardware limitations (e.g., signal timing requirements);
3. c)  Interfaces to other applications;
4. d)  Parallel operation;
5. e)  Audit functions;
6. f)  Control functions;
7. g)  Higher-order language requirements;
8. h)  Signal handshake protocols (e.g., XON-XOFF, ACK-NACK);
9. i)  Reliability requirements;
10. j)  Criticality of the application;
11. k)  Safety and security considerations.
    1. Assumptions and dependencies

Assumptions

Since the system will be deployed using a cloud based hosting service, an assumption about its continuous, uninterrupted availability is made.

Dependencies

Since GitGroup need to be authorized by Github, we assumed Github will authorize it.

3.interface requirements