Project Requirement Document

Team members: Shuiji Zhan, Bo Luan, Chunqing Liu, Zixia Weng, Zehao Li
Team Name: Microsophons

1. Define project specifics

This project is a web application supplying a real-time code editor that could be accessed by multiple users and a real-time text communication tool that could be used while coding together. Users will be involved in a web environment which would solve the collaboration problems faced by programmers for decades, such as conflicts of combining codes. With this web application, programmers could communicate with each other online through chatting window in texting window, audio window and even video window. Also, coding together could also be a way of communicating form, since users could show what they think about a specific line or word and different ideas could be reflected on the change of the code. Therefore, this product could be the right tool for programmers to code remotely.

2. Team goals and objectives (Major function and features)

- Create a real-time collaborative IDE in a browser
- Build an online code management system
- Establish real-time chat communications
- Allow users to code and communicate with each other efficiently

3. Background and strategic fit

Programming face by face in real time is somewhat difficult in our study and work. Our project provides a remote online coding IDE for different people programming together at the same time. Suppose you are a computer scientist, and you want to finish a project with someone who are thousands of miles away. It would be too much cost to meet each other. Although it is a good idea to split the project into halves and code separately, putting the code together would be very painful. Merging issues are the biggest problems, and besides that, you two need extra time to go through each other's code and try to understand the other person's opinions, which could also take a bunch of time. To save you a bunch of time and energy, an online coding application allowing you two working together could be the best solution.

4. Assumptions

There are a lot of demand to program on the same file at the same time. For example, when people are doing pair programming, instead of one of them can only watch, it is much better if they can edit the same file on two clients, while being able to see other's cursor and modification on the file. Thus, each user needs to have access to the internet where the entire web application is based on. Also, each user needs to have the correct browser such as Chrome, which would not result in any blocking or delaying issue while using the application. Meanwhile, we don't need to put the product on https, but on http instead. Moreover, infrastructures are

provided: the code for Monaco Editor and Socket.io needs not be implemented from scratch, so we can build additional functions onto them.

5. Requirements
1) User Stories or Use Cases (Functional)

Precondition: Running our program on the server. Log in/user identification.

Open an chrome tab and enter http://ruby-on-rails-shunjizhan.c9users.io/

Use Cases:

1. As a user, I can edit code in browsers, with codes highlighting.

Acceptance Criteria: On the website, I can code in the monaco editor window. The character I type on keyboard will present in the screen.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/32de6162a47978e09eafc8ffb05a5fbc450922fe

2. As a user, I can set my name in the name window.

Acceptance Criteria: On the website, I have a default name in the name window. I could also type my own name and click the set name button to save it.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/cff9104966afed8226df493a3b7f2792f8a37913

3. As a user, I can see other people's names in the name window.

Acceptance Criteria: On the website, I could see others' names in the name window.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/e3e7143f61cf70b15df99abbd4b5257a2157e745

4. As a user, I can see my own cursor and code change in the monaco editor window.

Acceptance Criteria: On the website, I could see the cursor and code change in the screen when I type.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/32de6162a47978e09eafc8ffb05a5fbc450922fe

5. As a user, I can see cursor position change as it moves.

Acceptance Criteria: On the website, I can see the coordinate of the cursor in the showevent window.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/32de6162a47978e09eafc8ffb05a5fbc450922fe

6. As a user, I can see other's chat words in the chat window.

Acceptance Criteria: On the website, I can see others' chat contents in the chat window.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/32de6162a47978e09eafc8ffb05a5fbc450922fe

7, As a user, I can send chat by type in the send window and click the send button to send it.

Acceptance Criteria: On the website, I could type in the send window and click the send button. Then, the words I send will present in the chat window.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/32de6162a47978e09eafc8ffb05a5fbc450922fe

8. As a user, I can see other people's work, so that I can give advice and do my job accordingly.

Acceptance Criteria: After other users log in, I can see their cursors with their name tags in the monaco editor window.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/db8a8fcb1756212325f55980cc58779bf6f1c896

9. As a user, I can upload a file to the editor.

Acceptance Criteria: the editor should show the content of the file after the upload file button is clicked.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/49d3e2c2ec9b1e40d9db49222f91a1df178239b8

10. As a user, I can download the content in the editor to local file system.

Acceptance Criteria: when click "save as" button, a new tab should show the raw text of the content.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/052f41777c224701df2701c8db217c4a397a46da

11. As a user, I can open multiple files and show them as tabs

Acceptance Criteria: A new tab with sample content will show up when new tab button is clicked, and when a file is uploaded, a new tab should be created to contain the content of the uploaded file.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/6eba46d542a2769a477df537587431f8cb01121d

12. As a user, I want the content to be synchronized between users even if we are working on different files that the same time

Acceptance Criteria: the content and cursor synchronization should stay working in spite of different files in different tabs are being edited.

Github commits:

https://github.com/shunjizhan/Microsoft-Capstone/commit/6eba46d542a2769a477df537587431f8cb01121d

13. As a user, I can save the code on the cloud so that my work will not be lost.

Acceptance Criteria: After I finish coding, I can click the save button on the website and save the project I am working on in the server.

14. As a user, I can retrieve back to where I left so that I can continue to code.

Acceptance Criteria: When I need to access it, I could easily click the project and restore the status of the last time. The cursor will present on the position last time.

15. As a user, I can directly see the results of my web code, thus I can save the time of running it on other IDEs.

Acceptance Criteria: When the project I am working on is a web program, I could see the result on the right of the monaco editor section after clicking "run" button, which is presented as a web tab.

16. As a user, I can set permissions for the code to people who can view/edit it so that only the authorized users can access the data.

Acceptance Criteria: Users who create the project could set permission for other users. This holder could give other participants permit to view/edit this project.

17. As a user, I can share my code to social networks, such as Twitter, so that my friends can see my code as well.

Acceptance Criteria: I could click the share button to send the project page with Twitter or friends' email links

18. As a user, I can chat instantly with other co-workers so that the coding process could be very efficient.()

Acceptance Criteria: When I am editing a project, I could see a chat window on the right side of the website. I could communicate with other people in this project.

19. As a user, I can voice chat with other people, so that they can communicate even they are remote.

Acceptance Criteria: When I am editing a project, I could click the voice chat button and create a voice group between invited users of this project.

20. As a user, I can invite other people to collaborate on my project so that it could be contributed by different people.

Acceptance Criteria: Users who create the project could click invite button and choose the people to view/edit the code. The users who are invited could choose the project, access the code and edit it.

2) User Stories or Use Cases (Nonfunctional)

1. Stability. As an user, the code synchronization will be stable over time and will be less likely to have any loss of data.

2. Reliability. In most conditions, our project will complete the content sharing and instant communication functions successfully.

3. Security. Hostile attacks will be reported and private data will be protected.

4. Recovery. If any exception, including program crashes and data losses, happens, the programs and files will be recovered.

5. Performance. Our project allows extremely fast content sharing, low utilization of resources and short response time of a given piece of work.

6. User Interaction and Design

Online IDE in a browser with chat
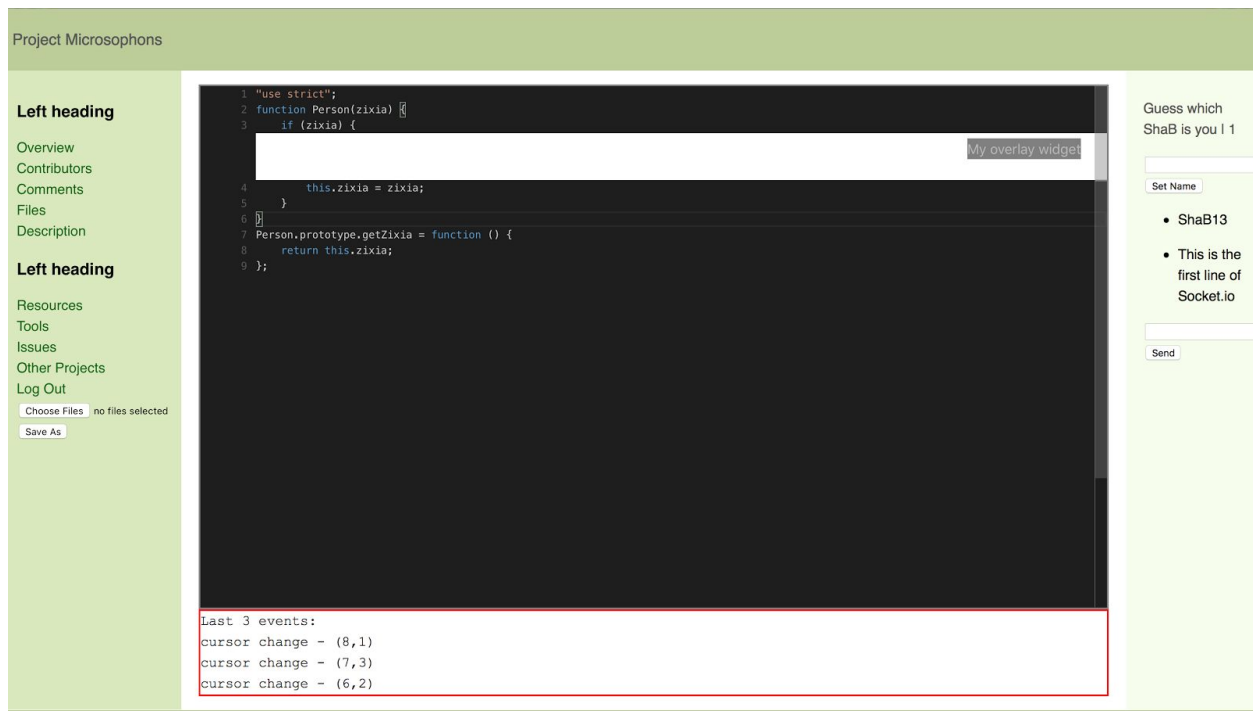
7. Major Problems to solve

1. How to achieve instant synchronization while coding in the monaco editor.
2. How to create database for users and files.
3. How to put website on the server.
4. The design of user interface in the browser.

8. Tools and References:

- Monaco
- Socket.io
- Node.js
- AngularJS

UML Diagram

editor

editors[];
editorID[];
filenames[];
content[];
lang[];
current_ID;

---

editor_function() // initial setup;
 $('#new-tab').on('click');
 $('.tab').on('click');
 $("#file-upload").on('change');
 $("#save-as").on('click');
 socket.on('cursor');
 socket.on('content');
 socket.on('new-user');
 socket.on('user-exit');
 socket.on('request-content');
 socket.on('new-tab');
 showEvent(str);
 setup_editor(div, content, language)
   editor.onDidChangeCursorPosition(function(e));
   editor.onDidChangeModelContent(function(e));
 create_cursor(user_id, y, x, color);
 new_tab(tab_name, content, language, foreground, new_ID);
 switch_tab(new_id);
 get_type(extension);

Server

Users[];
UserID[];
color[];

---

socket.on('new-user');
 socket.on('reply-content');
 socket.on('chat message');
socket.on('cursor');
socket.on('content');
socket.on('new-file');
socket.on('new-tab');
socket.on('user-name');
socket.on('disconnect');

socket communication: user->server

socket communication: server->user

socket comm: server->user

socket comm: user->server

Socket

socket;

$('form#send').submit();
 socket.on('chat message');
 $('form#user_form').submit;
 socket.on('update_user');
 socket.on('reply-content');