

A real-time collaborative (code) editor

with centralized server infrastructure

The project we propose is a real-time collaborative (code) editor based a centralized self-hosted server infrastructure. The two main components will be a client-side application used to interact with the project files and a server-side application which will be responsible for hosting the files and coordinating the real-time collaboration between the various connected clients.

Since this editor is meant for collaborative projects, we want to focus on implementing specific features supporting a collaborative coding effort. We believe that, given the numerous discrete parts that can be added to our minimum viable product, our proposed project is suitable for a group of our size. We have specified multiple stretch-goals as well in case the minimal product is working properly and we have some time left for implementing more sophisticated features.

The planned components of the application are laid out in the following table:

Precedence	Local	Remote
<i>Minimum viable product</i>	<ul style="list-style-type: none">• A basic terminal application capable of connecting to the proposed server infrastructure and capable of editing remote files.	<ul style="list-style-type: none">• A basic server-side application allowing multiple clients to be connected, presenting the project files to remote clients and allowing real-time editing for all connected clients.
<i>Important additional features</i>	<ul style="list-style-type: none">• Support for non-in-file comments/suggestions• Hexedit mode, handle large files (>2GB)• Show code in different codestyles (e.g. indentation, placement of curly braces)• Communication with other collaborators (chat)• Code completion• Emacs/VI keybindings• Working locally (git-integration)	<ul style="list-style-type: none">• Remote execution of compiled project• Read from and write to file script triggers (e.g. automated build)• Local synchronisation of files on client-side file system• Client login infrastructure
<i>Extra additional features</i>	<ul style="list-style-type: none">• Version history diffs• The option for peer-to-peer real-time collaboration• Caller/callee graphs, definition/implementation for expression under cursor	<ul style="list-style-type: none">• P2P connection vs. centralized system• Git-integration such that multiple collaborators can work on a different or the same branch• Plan board integration

In order to successfully engineer the collaborative editor, the following hardware and software equipment is required for a minimal viable product:

Proposed hardware requirements:

Component	Client	Server
Operating System	Linux, Windows (WSL only in case of shell application), (Apple)	Linux, (Apple?)

Proposed software requirements:

Component	Client	Server
Text-based	VT100 compliant, colors	VT100 compliant
Graphical-based	Target must be supported by Simple Directmedia Library. At least Windows and Linux are supported	N/A

Proposed software libraries:

Component	Library	Version
Graphical-based event and window handling	Simple Directmedia Library	2.0.9
Image subsystem (text)	Ncurses	libncurses5
Image subsystem (graphical)	Simple Directmedia Image Library, libpng, libjpeg, OpenGL	SDL: 2.0 OpenGL: 1.2 and better
Audio subsystem	SDL, OpenAL	2.0, 1.2
Filesystem event handling	inotifywait	3.14

This proposal is presented to you by the following students:

Mark Bebawy
 Martijn Besamusca
 Maxim van den Berg
 Chris Bras
 Sam van Kampen
 Frederick Kreuk
 Sverre Kroesen
 Jorik Smorenburg
 Folkert van Verseveld
 Mund Vetter
 Robin Wacanno
 Jim Wagemans

Antwoordmail over voorstel

Hi Robin,

Thank you for the proposal, it looks very interesting!

It seems like the project has two components: (1) creating a server which allows multiple agents to connect to it to provide real-time collaborative editing (and more), (2) an editor which is able to connect to this server. Even though we are confident that you are very motivated and have a lot of quality in your proposed team, we are a bit scared that you might be overestimating the amount of time you have. You have about 3 weeks of full time work.

Even one part, such as creating an editor capable of all listed functionalities, would be difficult project when we look at the history of the course. This is especially the case as it seems you want to work with more lower-level tools (which we think is really cool!). Even simple editors, like nano, took a long time to get to the place where they are now.

The same can be said about the server, or the protocol of real-time collaborative editing. I don't quite know the experience your proposed team has with real-time network programming and consensus algorithms. However, these problems are quite hard, and might take more time to solve than anticipated. Other things, like "Remote execution of compiled project", are also quite hard in my experience.

So in conclusion, we are very intrigued by the project, however we think that the workload might be a bit high. We think that focusing either on collaborative editing (and making it possible to do real-time collaborative editing with different editors), or on creating a new editor (which might have an architecture which is extra suitable for real-time collaborative editing) might have a more realistic workload.

*Kind regards,
Thomas Schaper*

Indelingsschets

List of non-programming tasks:

Task	People
Meeting planning	Frederick (everybody can suggest meetings)
Meeting chairman	Robin - Backup: Mund, Frederick
Meeting secretary (notulist)	Sverre - Backup: Maxim, Mark, Robin
Presenting	Jim, Frederick, Jorik, Robin
Presentation creation	Mund, Frederick, Robin, Mark, Maxim
Git co-ordinator	Jorik
Git experts (for answering questions by colleagues)	Folkert, Robin (more is appreciated)
C experts (for answering questions by colleagues) (possibly: checking code style and consistency)	Folkert, Sam, Chris (ish), Martijn (ish)
C++ experts (for answering questions by colleagues) (possibly: checking code style and consistency)	Folkert, Sam, Maxim (ish), Chris (ish)
Overall project leader (Announcing deadlines and features removals. Although hard decisions can be discussed and decided within the meetings) (Mostly communication with subleaders and meeting chairman/planning)	Frederick
Subleaders (of groups size 3-8 or smaller) (exact components are to be determined)	Server: Robin Editor: Jorik
Communication coordinator (Making sure every team member stays in contact) (Resolving conflicts within the team)	Jim

List of programming tasks:

Main Components	Subtask	Interested People (write down your name)
Low-end server infrastructure (3 people)	Message handling	Folkert, Maxim, Robin, Chris, Martijn, Sam
	Multithreading	Folkert, Maxim, Chris, Martijn, Robin
	File I/O	Folkert, Maxim, Chris, Martijn, Robin, Sam
<i>Intermediary (2 people)</i>	Network protocol	Folkert, Maxim, Robin, Chris, Sam
Editor + GUI (7 people)	Editor	Folkert, Robin, Maxim, Mund, Sverre, Frederick, Sam, Jim, Martijn
	<i>Intermediary (2 people)</i>	Martijn
	GUI Editor	Frederick, Jorik, Mark, Martijn

Date	Time	Person not present	Reason
4	from 3:00 till 16:00	Mund, Robin	Meeting onderwijscommissie
	After 12:00 (<i>As early as possible</i>)	Frederick	Tentamen Theoretische Aspecten van Programmatuur
	After 13:00	Jim, Mark, Maxim, Martijn	Tentamen Functietheorie
5	Entire day	Mund	Trip to the Efteling
	Entire Day	Chris	ongelukkig geplande rijles, wel bereikbaar op discord
	till +-12:00	Martijn	Piano lesson
	11:00-16:00	Folkert	Assisteren multimedia
6	11:00-14:00	Folkert	Assisteren multimedia
	After 13:00	Maxim	Tentamen
	From 12:45 till 17:15	Frederick	Assistentie bij Matching Informatica
	14:45 - 15:00	Mund	Mentorgesprek
	Na 14:30	Sam	Pakketje (maar werk thuis verder)
7	Entire day	Mark	Buitenland
	After 12:00 (<i>As early as possible</i>)	Frederick	Reünie middelbare school
	After 16:00	Folkert	Mentorgesprek
11	From 10:45 till 13:15	Frederick	Surveilleren bij tentamen Matching Informatica
	From 15:15 till 16:30	Frederick	Bespreking Academische Vaardigheden Informatica 1
	After 15:00	Folkert, Sverre	Training BAPC wedstrijd
	After 13:00	Jim	Oefenen SKKO
11 & 12	Hele dag	Chris	Werk
12	12:00-14:00	Folkert	Assisteren multimedia
13	From 14:45 till 17:15	Frederick	Academische Vaardigheden Informatica 1 Tutoraat
	14:45-15:15	Mund	Maandgesprek onderwijscommissie

	Hele dag	Jorik	Opa verhuizen
	10:30 - 14:00	Maxim	Sollicitatie + wiskundevoorlichting
	11:00 - 13:30	Jim, Mark en Martijn	wiskundevoorlichting
	na 12:30	Folkert	Assisteren
14	Na 13:00	Jim	Studiekeuze Kick-off
17	Hele dag	Maxim	Eindpresentaties project wiskunde
18	From 15:15 till 16:30	Frederick	Bespreking Academische Vaardigheden Informatica 1
19	From 13:15	Mund	Hockeytraining geven
20	From 14:45 till 17:15	Frederick	Academische Vaardigheden Informatica 1 Tutoraat
28	From 11:00 till 15:00 limited availability	Frederick, Folkert	Beoordeling Posterpresentaties Multimedia