

Group symbol: **3**

Team: **3**

Project title: **A fast, lightweight and intuitive text-editor**

**Team members** (*filled by PM, Team Leader*):

<b>N o</b>	<b>Name</b>	<b>Surname</b>	<b>Student ID</b>	<b>Role</b>
1	Gabriele	Simoni	293981	<i>PM, Team Leader</i>
2	Hüdalfa Bera	Dalgın	293988	<i>Team member</i>
3	Nozomi Malke	Shirasaki	288599	<i>Team member</i>
4	Erik	Parra Mejido	293864	<i>Team member</i>
5	Cédric Minh	Prétet	293891	<i>Team member</i>

## 1. Elaboration of application concept (S1)

### 1.1. Project (business) goals

#### **Main Goal**

The main goal of the project is to deliver a user-friendly, open-source text editor that empowers users to create, modify, and manage text files with efficiency and reliability. The core mission is to provide a free, accessible, and unrestricted tool that meets the needs of everyday users, developers, and writers alike, combining simplicity, performance, and openness in a single, cohesive platform.

#### 1. Functionality

The editor will include all essential features for text manipulation, creating, opening, editing, and saving files with ease. In addition, future iterations and extensions will provide a suite of convenient utilities such as find and replacement, undo and redo, word wrapping, and basic text formatting as well as limited coding capabilities.

Future enhancements will aim to extend functionality through customization options, and keyboard shortcuts, ensuring the editor adapts to a wide range of workflows.

#### 2. Performance

Performance will be a central focus. The editor will be engineered for rapid startup, low memory footprint, and fast response to user input. It should maintain smooth performance. Efficient algorithms and lightweight design principles will ensure a consistently fast and reliable editing experience.

#### 3. Cross-Platform Compatibility

The application will be programmed in C, allowing it to be natively supported on Windows, macOS, and Linux, ensuring consistent functionality and appearance across platforms. The user experience will be unified, allowing seamless transition between devices without the need for reconfiguration.

## 4. Security and Data Integrity

The software will emphasize data safety and reliability at every stage of file handling. Robust error management mechanisms will protect against crashes, data corruption, and accidental loss during editing or saving. Features such as auto-save, backup creation, and version recovery will further safeguard user content in future versions.

## 5. Transparency and Open Source

As an open-source initiative, the project will foster collaboration, peer review, and community driven improvement. The source code will be freely accessible, allowing developers and users to inspect, modify, and redistribute it under a permissive license. This openness will ensure long-term sustainability, encourage innovation, and build trust through shared ownership and accountability.

### 1.2. Identification of project's internal and external Stakeholders

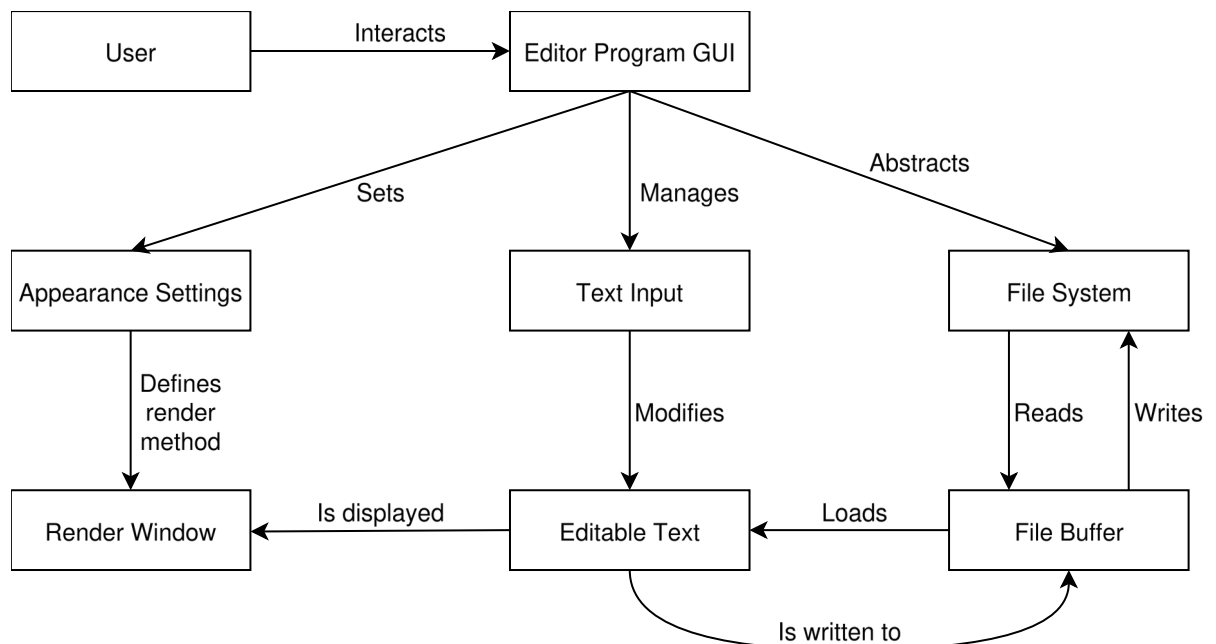
#### • Internal Stakeholders

Symbol	Name	Role	Description
1	Gabriele Simoni	Project Manager	Plans tasks, coordinates the team, monitors progress and contributes to development
2	Hüdalfa Bera Dalgın	Team Member	Works on tasks assigned by PM, contributes to development, testing, or documentation as needed.
3	Nozomi Malke Shirasaki	Team Member	Works on tasks assigned by PM, contributes to development, testing, or documentation as needed.
4	Erik Parra Mejido	Team Member	Works on tasks assigned by PM, contributes to development, testing, or documentation as needed.
5	Cédric Minh Prétet	Team Member	Works on tasks assigned by PM, contributes to development, testing, or documentation as needed.

## • External Stakeholders

Symbol	Name	Role	Description
1	Computer users	End users	Generic computer users, ranging from casual users to programmers who may use the program to write text files, edit configuration files or write code.
2	Open source community	Community	The program may be adopted by open source projects as part of toolchains. The program may help expand the set of open source tools available.
3	Future contributors	Expand the project	Individuals who may contribute to the project's future development to fix bugs and add features to gain experience.

### 1.3. Domain description



- **Entities**

Entities	Description
User	The one who controls the application. Interaction is done only through the Editor Program GUI.
Editor Program GUI	The primary controller for the application. It processes GUI interactions and keyboard inputs from the user.
File System	File manipulation is done through this system. It opens files to read and edit. Several files can be open at the same time and this system keeps track of them.
File Buffer	The content of each open file is loaded in a file buffer. When saving to a file, the data stored in the file buffer will be written to the corresponding file.
Text Input	Processed from the Editor Program GUI, text input is taken to use to edit an editable text.
Editable Text	This is where the content of each file is stored and edited in the application. Each file has a corresponding Editable Text to use.
Appearance Settings	Controls the way that the application renders the text editor window based on saved settings.
Render Window	This is the section of the application which displays the text to be edited. It takes into account the appearance settings and the syntax highlighting when displaying known file types.

- **Relationships**

<b>Relationships</b>	<b>Name</b>	<b>Description</b>
User -> Editor Program GUI	Interacts	All user input is processed through the Editor Program GUI.
Editor Program GUI -> File System	Abstracts	The file system isolates the file related tasks from the Editor Program GUI to create a more organized set up.
File System -> File Buffer	Reads	Data from opened files are read into the File Buffer.
File Buffer -> File System	Writes	Data from the File Buffer is written into a file opened in the file system.
Editor Program GUI -> Text Input	Manages	Input is processed then only the text input is passed to be used to edit the focused text.
Text Input -> Editable Text	Modifies	The text input is used to edit the Editable Text that has focus.
Editor Program GUI -> Appearance Settings	Sets	The application sets the settings for the appearance of the application window.
Appearance Settings -> Render Window	Defines render method	The appearance settings stored in the application are used to initialize the appearance of the window.
File Buffer -> Editable Text	Loads	The content of an Editable Text is loaded from the File Buffer.
Editable Text -> File Buffer	Is written to	When saving a file, the content of an Editable Text is written to the File Buffer to prepare to write to the corresponding file.
Editable Text -> Render Window	Is displayed in	The text content from an Editable Text is displayed in the Render Window.

- **Events**

Events	Description
File Opened	Triggered when the user selects a file to open through the Editor Program GUI. The File System will then read the contents of the file to the buffer and assign an Editable Text to hold the contents.
File Saved	Triggered when the SaveButton is pressed. Overwrites the contents of the TextBox onto the selected file.
Text Edited	Change the contents of the focused Editable Text based on the user's processed text input.
Document Closed	Remove the corresponding Editable Text and any associated data.
Settings Changed	Save the new settings to a file and render the window again with the new settings.

#### 1.4. Project schedule (Gantt chart)

The project schedule is regularly updated in a shared Google Drive spreadsheet.

You can access it using the following link:

[https://docs.google.com/spreadsheets/d/1ZAp8YltL8X0dDxpcvXPFH82-Os02tLYX/edit?](https://docs.google.com/spreadsheets/d/1ZAp8YltL8X0dDxpcvXPFH82-Os02tLYX/edit?usp=drive_link&oid=107940987460351811396&rtpof=true&sd=true)

[usp=drive link&oid=107940987460351811396&rtpof=true&sd=true](https://docs.google.com/spreadsheets/d/1ZAp8YltL8X0dDxpcvXPFH82-Os02tLYX/edit?usp=drive_link&oid=107940987460351811396&rtpof=true&sd=true)

The following is a current snapshot of the project schedule:

Tasks Tracker									
No	State	Name	Artifact	Duration	Start	Finish	Predecessors	Assignment	
								No	Name
1	Done	Make Gantt graph	<i>schedule.pdf</i>	5 days	08/10/25 15:15	13/10/25 15:15		1	Simoni
2	Done	Compile team list	<i>team_list.pdf</i>	5 days	08/10/25 15:15	13/10/25 15:15		1	Simoni
3	Done	Choose title	<i>title.pdf</i>	5 days	08/10/25 15:15	13/10/25 15:15		1	Simoni
4	Done	Define business goals	<i>business_goals.pdf</i>	5 days	08/10/25 15:15	13/10/25 15:15		4	Parra Mejido
5	Done	Write domain description	<i>domain.pdf</i>	5 days	08/10/25 15:15	13/10/25 15:15		3	Shirasaki
6	Done	Identifying external stakeholders	<i>stakeholders.pdf</i>	5 days	08/10/25 15:15	13/10/25 15:15		2	Dalgin
7	Done	Identifying alternative solutions	<i>solutions.pdf</i>	5 days	15/10/25 15:15	20/10/25 15:15		5	Prézet
8	Done	Identify application context	<i>app_context.pdf</i>	5 days	15/10/25 15:15	20/10/25 15:15		3	Shirasaki
9	Done	Identify technological context		5 days	15/10/25 15:15	20/10/25 15:15		3	Shirasaki
10	Done	Identify organisational context		5 days	15/10/25 15:15	20/10/25 15:15		1	Simoni
11	Done	Identify legal context		5 days	15/10/25 15:15	20/10/25 15:15		2	Dalgin
12	Done	Define technologies used		5 days	15/10/25 15:15	20/10/25 15:15		1	Simoni
13	To-do	Choose documentation tool						0	unassigned

## 1.5. Identification of existing or alternative solutions

Alternatives	Main features	Advantages	Disadvantages
<i>Visual Studio Code</i>	<ul style="list-style-type: none"> <li>• Syntax Highlighting</li> <li>• Extensive plugin ecosystem</li> <li>• Cross-platform compatibility</li> <li>• Integrated terminal and Git support</li> <li>• Customizable interface</li> </ul>	<ul style="list-style-type: none"> <li>• Highly extensible through extensions</li> <li>• Active open-source community</li> <li>• Frequent updates and feature improvements</li> <li>• Integrated debugging and development tools</li> </ul>	<ul style="list-style-type: none"> <li>• Built on Electron, leading to higher memory usage</li> <li>• Slower performance compared to lightweight editors</li> </ul>
<i>Sublime Text</i>	<ul style="list-style-type: none"> <li>• Syntax Highlighting</li> <li>• Cross-platform support</li> <li>• Extensive plugin system via Package Control</li> <li>• Fast startup and performance</li> </ul>	<ul style="list-style-type: none"> <li>• Extremely fast and responsive</li> <li>• Rich ecosystem of plugins</li> <li>• Highly customizable</li> </ul>	<ul style="list-style-type: none"> <li>• Closed source, limiting community involvement</li> <li>• Paid license after trial period</li> </ul>
<i>Notepad++</i>	<ul style="list-style-type: none"> <li>• Syntax Highlighting</li> <li>• Open source</li> <li>• Plugin support</li> <li>• Lightweight and fast</li> <li>• Windows-focused</li> </ul>	<ul style="list-style-type: none"> <li>• Lightweight and efficient</li> <li>• Active open-source community</li> <li>• Supports many programming languages</li> </ul>	<ul style="list-style-type: none"> <li>• Limited native support on non-Windows platforms</li> <li>• Outdated UI compared to modern editors</li> </ul>
<i>Atom</i>	<ul style="list-style-type: none"> <li>• Syntax Highlighting</li> <li>• Cross-platform</li> <li>• Plugin and theme support</li> <li>• GitHub integration</li> </ul>	<ul style="list-style-type: none"> <li>• Open-source and hackable</li> <li>• Good for customization and prototyping</li> <li>• Rich plugin ecosystem</li> </ul>	<ul style="list-style-type: none"> <li>• Discontinued since 2022</li> <li>• Slower performance (Electron-based)</li> <li>• No longer actively maintained</li> </ul>
<i>Windows Notepad</i>	<ul style="list-style-type: none"> <li>• Basic text editing</li> <li>• Included with Windows</li> <li>• Fast startup</li> </ul>	<ul style="list-style-type: none"> <li>• Very lightweight and simple to use</li> <li>• Instant startup time</li> </ul>	<ul style="list-style-type: none"> <li>• Closed source, limiting community engagement</li> <li>• No syntax highlighting or advanced features</li> <li>• Windows-only</li> </ul>



## SE 2025 Winter - Project Report S1

<i>Gedit</i>	<ul style="list-style-type: none"><li>• Syntax Highlighting</li><li>• Plugin support</li><li>• Open source</li><li>• Cross-platform (Linux, Windows, macOS)</li></ul>	<ul style="list-style-type: none"><li>• Simple and user-friendly</li><li>• Open-source and community-driven</li><li>• Integrates well with GNOME desktop</li></ul>	<ul style="list-style-type: none"><li>• Lacks advanced developer tools</li><li>• Not as customizable as other editors</li></ul>
<i>Kate</i>	<ul style="list-style-type: none"><li>• Syntax Highlighting</li><li>• Multiple document interface</li><li>• Plugin support</li><li>• Open source</li><li>• Cross-platform</li></ul>	<ul style="list-style-type: none"><li>• Feature-rich for an open-source editor</li><li>• Strong community support</li><li>• Good performance</li></ul>	<ul style="list-style-type: none"><li>• Interface may be overwhelming for beginners</li><li>• Less popular on Windows and macOS</li></ul>
<i>Obsidian</i>	<ul style="list-style-type: none"><li>• Markdown-based text editing</li><li>• Cross-platform (Windows, macOS, Linux, mobile)</li><li>• Plugin and theme support</li><li>• Graph view for linking notes</li><li>• Offline support</li></ul>	<ul style="list-style-type: none"><li>• Powerful organization and linking features</li><li>• Large and active user community</li><li>• Plugin system allows customization</li><li>• Excellent for notetaking and knowledge management</li></ul>	<ul style="list-style-type: none"><li>• Closed source (limits community engagement)</li><li>• Heavier and slower startups than lightweight editors</li><li>• Focused more on note management rather than pure text editing</li></ul>
<i>Notion</i>	<ul style="list-style-type: none"><li>• Online collaborative editor</li><li>• Cross-platform (web, desktop, mobile)</li><li>• Rich-text editing with databases, tables, and media embedding</li><li>• Real-time synchronization and collaboration</li></ul>	<ul style="list-style-type: none"><li>• Excellent for structured content and team collaboration</li><li>• Clean and modern interface</li><li>• Integrates with many external services</li></ul>	<ul style="list-style-type: none"><li>• Closed source requires an internet connection for most use cases</li><li>• Slower and heavier than traditional text editors</li><li>• Not optimized for code or plain-text editing</li></ul>

<i>VIM</i>	<ul style="list-style-type: none"> <li>• Modal text editing (normal, insert, visual, command modes)</li> <li>• Syntax Highlighting</li> <li>• Cross-platform (Linux, macOS, Windows)</li> <li>• Highly configurable via scripts and plugins</li> <li>• Open-source and terminalbased</li> </ul>	<ul style="list-style-type: none"> <li>• Extremely lightweight and fast</li> <li>• Fully open source with a large, active community</li> <li>• Runs efficiently even on low-end systems</li> <li>• Highly customizable and extensible</li> </ul>	<ul style="list-style-type: none"> <li>• Steep learning curve for new users</li> <li>• Minimal GUI (primarily terminal-based)</li> <li>• Configuration and plugin management can be complex</li> </ul>
------------	---	--	--

## 1.6. Project context

### Application Context

This project aims to create a text editor made for developers. It will be lightweight, fast, and open source. As a text editor as its base, users will be able to open, edit, and save files. As an open source project, additional functionalities will be added and shared. These extra features can include helpful tools for developers such as undo and redo commands, syntax highlighting for programming languages, and search commands. The base will be minimal in comparison to other programming text editors as it needs to be lightweight and high performance. It will also need to be robust and able to run on many platforms as it will act as the basis for all future extensions.

### Technological Context

To accomplish the functionalities mentioned above, the technologies utilized will need to be selected accordingly. The following technologies will be used in the production and deployment of this project.

- The project will be written in C to allow for a minimal and high-speed application. This will also allow the application to run on all the most common operating systems
- Gtk will be used as an open-source and cross-platform option to implement the GUI.
- Make will be used to automate the build process and allow easy usage across different systems.

- Git will enable version tracking and promote transparency for the project which will in turn aid the expansion of this project after release.
- The editor is planned to be available cross-platform across Linux, Windows, and macOS

## **Organizational Context**

The text-editor project is undertaken by a group of five students as part of a university course. The team follows flexible roles with a basic hierarchy consisting of one project manager and four members who receive tasks to complete. The members are assigned to different tasks such as compiling documents, reviewing artifacts, and writing code.

Communication between members of the group takes place on a social platform (Discord) where the Project Manager regularly posts updates on the project. A spreadsheet containing all the scheduled tasks is maintained by the Project Manager and shared with the rest of the team. This provides transparency and offers clarity about the tasks left to be completed. The spreadsheet also functions as a tool to track which member was assigned to which task.

## **Legal Context**

### **1. Licensing**

The Text Editor project will be distributed under the GNU General Public License (GPL v3). This license allows users to freely use, study, modify, and redistribute the software, ensuring that all versions remain open source. By adopting the GPL, we aim to support transparency, collaboration, and innovation within the open-source community.

### **2. Intellectual Property**

All members of the project team are joint owners of the original codebase and resources created for this project. Once the software is released under the GPL license, it becomes open to public contribution. Any modifications or extensions made by external contributors must remain under the same GPL terms and include proper credit to the original authors.

### **3. Compliance and Ethical Use**

The team commits to adhering to both legal and academic standards throughout the development process. No copyrighted or proprietary materials will be used without authorization, and any external dependencies included will be compatible with the GPL license. We also commit to ethical software engineering practices, fostering transparency and respect for intellectual property rights.

### **4. Liability Disclaimer**

In compliance with GPL v3, the software will be provided 'as is', without any warranty. The development team is not responsible for any potential damage, data loss, or issues that may arise from its use. Users assume full responsibility when utilizing the application.

## 1.7. Technologies used in the project

### 1.7 Technologies used

Name	Description	Justification	Responsibilities	Website
<i>C Programming Language</i>	C is a general-purpose programming language. It provides low-level memory access, structured programming constructs, and is widely used for system software, embedded systems, and performance-critical applications.	C is widely adopted for open-source projects for its portability. Its efficiency aligns with our goal to deliver a text-editor with optimal performance.	<ul style="list-style-type: none"> <li>Defining software logic</li> <li>Managing memory</li> <li>Facilitating interaction between hardware and other software components</li> </ul>	<a href="https://en.wikipedia.org/wiki/C_(programming_language)">https://en.wikipedia.org/wiki/C_(programming_language)</a>
<i>GNU Toolchain</i>	The GNU Toolchain is a collection of open-source programming tools used to develop software.	It provides a reliable set of tools. The ones concerning our project are the C compiler (GCC), the debugger (GDB) and the build system (Make). It makes our project more accessible to open-source contributors, who are already familiar with the toolkit.	<ul style="list-style-type: none"> <li>Compiling source code into executables</li> <li>Debugging applications</li> <li>Automating the build and installation process</li> </ul>	<a href="https://en.wikipedia.org/wiki/GNU_toolchain">https://en.wikipedia.org/wiki/GNU_toolchain</a>
<i>GTK</i>	GTK is an open-source library used for creating applications that have graphical user interfaces. It supports multiple platforms and languages.	It allows for cross-platform compatibility, good performance and is open-source.	<ul style="list-style-type: none"> <li>Handling user interaction with the application</li> <li>Integrating GUI with the application logic</li> </ul>	<a href="https://www.gtk.org/">https://www.gtk.org/</a>
<i>GitHub</i>	GitHub is a web-based platform for version control and collaborative software development using Git. It allows developers to host repositories, track changes and collaborate in teams.	GitHub allows for collaborative development. Because Github is widely recognized in the software development community, hosting a project there increases its visibility.	<ul style="list-style-type: none"> <li>Hosting the project code</li> <li>Tracking version history</li> <li>Facilitating team collaboration</li> <li>Tracking team members contributions to the code</li> </ul>	<a href="https://github.com/about">https://github.com/about</a>
<i>Draw.io</i>	Draw.io is a diagramming tool for creating flowcharts and UML diagrams.	It provides a simple and free way to create diagrams. It is open-source.	<ul style="list-style-type: none"> <li>Offering a simple and effective way to design diagrams</li> </ul>	<a href="https://www.drawio.com/about">https://www.drawio.com/about</a>
<i>LibreOffice Writer</i>	LibreOffice Writer is a free and open-source word processor.	It's free and open-source. It has all the features of a modern word processor.	<ul style="list-style-type: none"> <li>Allowing the team to create, edit, and format text documents, including reports, artifacts and technical documentation.</li> </ul>	<a href="https://www.libreoffice.org/discover/writer/">https://www.libreoffice.org/discover/writer/</a>