Team G

OopsFix
Software Requirements Specification
For OopsFix

Version 1.0

Revision History

Date	Version	Description	Author
3/17/25	1.0	Updated Purpose	Nur Dogrusoz
3/17/25	1.0	Updated Introduction & Use Case Model Survey & Specific Requirements	Klea Meta
3/17/25	1.0	Updated Overall Description	Sujana Yeasmin
3/18/25	1.0	Updated Assumptions & Dependencies, Use-Cases	Gaurav Gupta
3/18/25	1.0	Updated Purpose	Tony Christopher

Table of Contents

Software Requirements Specification

1. Introduction

1.1 Purpose

OopsFix is an AI-powered cooperative text editor designed to assist users in improving their writing through automated text correction and collaborative editing. The system leverages a Large Language Model (LLM) to identify and suggest corrections for grammar, spelling, and syntax errors, allowing users to refine their writing with precision. Unlike traditional spell-checkers, OopsFix provides users with an interactive correction process, where they can choose to accept or reject AI-generated fixes, ensuring full control over their writing. Additionally, users can opt for self-correction, giving them flexibility in how they improve their texts. To maintain fairness and access, the system operates on a token-based economy, allowing free users limited access, while paid users can purchase tokens for extended use.

What sets OopsFix apart is its commitment to making it more engaging, playful and user-friendly, ensuring a unique and enjoyable editing experience. Instead of a dry, mechanical correction system, OopsFix incorporates interactive elements, such as visualized error tracking, gamification, and AI-powered feedback, to make the writing process feel less like a chore and more like a collaborative learning experience.

1.2 Scope

Our proposed text editor aims to create a collaborative, AI-powered writing assistant that enhances users' text refinement process through interactive corrections and structured access control. This system integrates self-correction and LLM-assisted correction, allowing users to identify, review, and improve grammatical, spelling, and syntax errors with a guided approach. The software will require registration to be used and will allow three types of user roles. The three types of user roles that will be supported by the software are:

1. **Free users**
2. **Paid users**
3. **Super users**

Overall, the software will provide users with the ability to submit and edit text, collaborate on writing projects, manage their token-based access, and interact with AI-powered corrections in a structured and engaging way. Therefore, all OopsFix user roles will be able to engage with the platform's text editing and correction features according to their specific privileges and responsibilities. Free users will have limited access, allowing them to submit short texts for AI-assisted correction, while paid users will have greater flexibility with extended editing, file-saving, and collaboration capabilities. Super users will oversee system moderation, user management, and blacklist approvals, ensuring fairness and compliance within the platform.

1.3 Definitions, Acronyms, and Abbreviations

1. **Super User (SU):** An administrator who manages user accounts, approves blacklisted words, handles complaints, and enforces penalties.
2. **Paid User (PU):** A registered user who purchases tokens for AI-assisted text corrections, file-saving, and collaboration features.
3. **Free User (FU):** A non-paying user who can submit only 20-word text entries per session before facing temporary restrictions.
4. **Tokens:** A virtual currency used by paid users to access text editing, AI corrections, file-saving, and collaboration features.
5. **Blacklist:** A list of restricted words managed by super users, where flagged words are replaced by '*' and charged based on length.
6. **Correction Request:** A user-submitted text that is processed by either self-correction or AI-powered editing.
7. **Self-Correction:** A manual editing process where users revise their text at a reduced token cost (number of tokens equal to half of the words self-corrected).
8. **AI Correction (LLM Correction):** A system where an LLM automatically suggests fixes, charging one token per accepted correction.
9. **Collaboration:** A feature that allows paid users to invite others to co-edit documents with self or AI correction options, which is charged 3 tokens in case of reckless inviting.
10. **Saved Document:** A paid user's corrected text file that can be stored for future reference at a cost of 5 tokens.
11. **Penalty:** A token deduction imposed on users who exceed word limits, misuse AI corrections, or violate system rules.
12. **Bonus Tokens:** A reward system that grants 3 tokens if a 10+ word text has no AI-detected errors.

13. **Complaint:** A formal report filed against another user that super users review to determine penalties or resolutions.
14. **Submission Limit:** A restriction preventing free users from submitting more than 20 words per session.
15. **Correction Log:** A history of all AI-generated and manual corrections that helps paid users track their writing progress.

1.4 References

Laban, Philippe, Marti A. Hearst, and Jeffrey Heer. "Guided Writing with AI: A Mixed-Initiative Interface for Controlling Large Language Models." *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, 2024,

Hughes, Christian. "A Year of Showing Up." *Make Art with Python*, 2024

Allman, Jon. "Tech Primer: What Hardware Do You Need to Run a Local LLM?" *Puget Systems Labs*, 12 Aug. 2024

1.5 Overview

The documentation is organized into two primary sections for clarity and ease of understanding. The first section, "Overall Description," presents a comprehensive use-case model survey, detailing the roles of different users, their permissions, and key system functionalities. This section also includes visual diagrams to illustrate user interactions and feature accessibility, providing a structured overview of the system's architecture.

The second section, "Specific Requirements," outlines detailed use-case reports along with additional system requirements, ensuring a clear definition of feature functionality. This section specifies how users engage with essential components such as text input, AI-powered corrections, token transactions, and collaboration tools. By following this structured approach, the documentation offers a well-defined perspective on OopsFix's operation, ensuring both usability and system compliance.

2. Overall Description

2.1 Use-Case Model survey

Actors & Roles

Free User

A Free User has limited access to the LLM-based editor which allows them to submit up to 20 words per request for text correction. However, they must wait 3 minutes before making another submission. Their interface is simplified, providing only basic functionalities without access to advanced features like token-based corrections or collaboration. While they can view corrections made by the LLM, they do not have access to past correction history, file uploads, or interactive revision features. Also, Free Users can suggest words for the blacklist, but these must be reviewed and approved by a Super User. To access longer text submissions, collaboration tools, and advanced features, they have the option to upgrade to a Paid User.

Paid User

A Paid User has full access to the system, including the ability to purchase and use tokens for submitting texts beyond the Free User limit. They can choose between self-correction or LLM-assisted correction, where the AI highlights errors for review and manual approval. Paid Users can upload text files, track their correction history, and customize how the LLM interacts with their text. Additionally, they have collaboration privileges. They can invite other Paid Users to review or co-edit documents, as well as accept or reject collaboration requests. They also gain access to a statistical dashboard, displaying token usage, total corrections made, and system engagement metrics. This membership provides a more interactive, personalized writing experience, improving collaboration and accuracy.

Super User

A Super User or the system administrator(s) is responsible for managing user accounts, handling complaints, and implementing moderation policies. They oversee user sign-ups, ensuring Paid Users meet platform guidelines and resolving conflicts between collaborators. Also, Super Users manage the blacklist, reviewing suggested words from both Free and Paid Users before approving or rejecting them. They have the authority to suspend, fine, or terminate Paid Users who violate platform rules, ensuring fair usage of the LLM based system. Their dashboard includes complaint logs, user violation reports, and system analytics which allows them to maintain a secure and well-moderated environment.

Use-Cases

Text Submission

This tool enables users to input text by either entering in a text box or uploading a file. Upon submission of text by a paid user (PU), the system computes the word count and deducts one token for each word prior to processing the content. Upon detection of any banned terms, they are substituted with asterisks, and further tokens are deducted in accordance with their length. When the user possesses an inadequate number of tokens, the submission will be unsuccessful, resulting in the loss of half of the remaining tokens as a penalty.

The typical scenario is that the user possesses sufficient tokens, resulting in successful text processing.

Exceptional cases include insufficient tokens, in which case the submission is rejected and a penalty is applied.

Self-Correction

Users who wish to manually edit their writing may utilize the self-correction mode, enabling them to fix errors at a reduced token cost of half a token per word. This option provides customers enhanced choice while continuing to leverage the system's AI-driven support. The typical scenario involves the user modifying their text, resulting in a corresponding deduction of tokens. A unique situation arises when the user lacks sufficient tokens for self-correction, leading to an error message and an inability to advance.

LLM-Based Text Correction

If a paid user selects LLM Correction, the system scans the submitted text for grammar, spelling, and syntax errors and highlights suggested corrections. The user can review each suggestion and decide whether to accept or reject it. For every accepted correction, 1 token is deducted, while rejected suggestions incur no charge. However, if the user rejects an LLM-generated correction, they must provide a reason, which is then reviewed by the super user (SU). If the reason is deemed valid, only 1 token is deducted; if the reason is invalid, the user loses 5 tokens as a penalty. The normal case is that users accept corrections and are charged accordingly. Exceptional cases include rejecting valid LLM corrections with an invalid reason, leading to an increased token penalty.

Saving Edited Text

After completing the correction process, paid users have the option to save their edited text for future reference. Saving a file incurs a flat fee of 5 tokens, encouraging mindful storage to prevent excessive file-saving. The normal case is that the user has enough tokens, and the file is successfully saved. The exceptional case occurs when the user lacks sufficient tokens, in which case the system prevents saving until more tokens are purchased.

Collaboration and Shared Editing

Paid users can invite other paid users to collaborate on a text file, allowing shared editing with self or AI correction options. When an invitation is sent, the recipient can accept or reject the request. If the invitee accepts, both users can edit the document together. If the invitee rejects, the inviter loses 3 tokens as a penalty for reckless inviting. The normal case is that both users agree to collaborate, and the document is shared successfully. Exceptional cases occur when the invitation is rejected, triggering a penalty, or when a collaborator misuses the document, prompting the inviter to file a complaint with the super user.

Performance-Based Rewards

To encourage high-quality writing, OopsFix rewards paid users for submitting well-structured content. If a text of 10 or more words is submitted and the AI detects zero errors, the user receives a bonus of 3 tokens. The normal case is that the user submits an error-free text and is rewarded with bonus tokens. The exceptional case occurs if an AI error falsely detects a mistake, in which case the user does not receive the bonus.

Error Tracking and System Statistics

OopsFix provides detailed logs that allow paid users to track their error history, correction patterns, and token usage statistics. This feature enables users to identify frequent mistakes and monitor their progress over time. The normal case is that users can access their correction logs and view system-generated insights. The exceptional case occurs if the system fails to record a correction, in which case the user may need to request a manual review from a super user.

Complaint and Dispute Handling

If a user feels that a collaborator has misused a shared document or violated system policies, they can file a complaint with the super user. Upon login, the accused user is notified and must respond to the complaint. The super user reviews the complaint and determines a resolution, which may involve penalizing either the complainant or the accused user. The normal case is that disputes are fairly resolved, with appropriate token deductions applied. The exceptional case occurs when a complaint is found to be fraudulent or baseless, in which case the complainant may face penalties instead.

Blacklist Management

The blacklist is a system feature that prevents users from submitting restricted words or phrases, which are either inappropriate, offensive, or flagged by users for review. When a paid or free user submits text, the system scans it against the blacklist and automatically replaces any restricted words with '*' characters. In addition, the system deducts tokens based on the length of blacklisted words used in the submission.

Users can suggest words for inclusion in the blacklist, but only super users (SU) have the authority to approve or reject these suggestions. If a word is approved, it is permanently added to the blacklist and applied to all future text submissions. If a user repeatedly attempts to bypass the blacklist by altering restricted words, they may receive a penalty or face temporary suspension from the system.

Token Purchase

To access premium features, paid users must purchase tokens through the platform's integrated payment system. The user selects the number of tokens to buy, confirms the transaction, and the system adds the purchased tokens to their balance. The normal case is that the transaction is successful, and the tokens are credited to the user's account. Exceptional cases include payment failures due to insufficient funds, expired payment methods, or transaction errors, in which case the system notifies the user and prompts them to retry the purchase.

User Role Upgrades

A free user can upgrade to a paid user by purchasing tokens and registering for premium features. The system prompts them to complete payment and grants full access to paid functionalities once the transaction is successful. The normal case is that the upgrade is processed smoothly, and the user transitions to a paid account. The exceptional case occurs when an error prevents the upgrade, such as technical issues, payment failures, or system glitches, requiring manual intervention from a super user.

Emergency Token Borrowing System

For users who run out of tokens mid-session, OopsFix provides an Emergency Token Borrowing option, allowing paid users to borrow up to 10 tokens at a time. Borrowed tokens must be repaid within 48 hours, either by purchasing new tokens or accepting a system-generated penalty. The normal case is that users repay borrowed tokens on time. The exceptional case occurs when a user fails to repay on time, resulting in an additional penalty or restricted editing access.

Undo Last AI Correction

Users sometimes accidentally accept an LLM correction they didn't intend to. To fix this, OopsFix includes an undo button, allowing users to revert the last correction within 30 seconds of applying it. The normal case is that users undo mistaken changes quickly. The exceptional case occurs when the undo period expires, requiring users to manually re-edit the correction.

Oops Meter

The Oops Meter provides a live visual representation of the error density in the user’s text in the form of a heatmap. As corrections are made (or suggestions denied), the heatmap is automatically adjusted to reflect the new error density. This allows the user to analyze the strength of their writing in a fun and easy-to-see way.

Explanations for Correctness

For each suggestion, the user can request to see the reason for the suggestion. This will prompt the system to reveal the LLM model’s logic for the recommended correction. This feature is closely linked with the UI in that each highlighted suggestion will have a corresponding reason which can be displayed on request.

2.2 Assumptions and Dependencies

Multiple systems must work together smoothly in order for OopsFix to function as intended. The backend system is responsible for handling user text input, corrections, and interactions (accept/reject/self-edit/etc), all of which is needed for the editor to be responsive and effective. The frontend system will be simple and easy to navigate, assuring convenient and seamless user access of the backend functionality (i.e. undo feature, explanations for suggestions, Oops Meter, etc).

The most important part of this software is the text correction engine. This engine is responsible for determining which corrections should be made, the reason they should be made, the impact/priority (quantitative) measure of each correction. In short, this engine will process user input, highlight-errors, and provide AI-generated suggestions. It is assumed that this engine will work in sync with the token management system, blacklist management system, and oops meter, which will visually represent the density of errors in the user-provided input. Whenever the user accepts or denies a suggested correction, the Oops Meter heatmap should be adjusted accordingly.

As users may sometimes accept suggestions by mistake, they are given the ability to undo an accepted suggestion within 30 seconds of accepting it. This requires the system to temporarily store recent changes and provide a smooth rollback mechanism in the event a correction is undone.

Additionally, it is assumed that the system maintains some form of long-term data storage for paid users so that they can analyze their error history.

The user management system is another crucial component. There are three-tiers of users, as defined above (free users, paid users, and super users). The user management system must provide the respective users with appropriate permissions and restrictions for their user-tier. Super-users are responsible for handling sign-ups, disputes, and moderating restricted content.

Another essential element is the token system. The system must accurately track token deductions, refunds, purchases, and penalties, ensuring fairness for all users and maintaining effective functionality of the collaboration and performance-based rewards features.

The blacklist management system helps filter out inappropriate or flagged words. Users can suggest words for the blacklist, but final decisions are made by super users. The system must also manage complaints and disputes, notifying users when they are reported and allowing super users to review cases and issue penalties when necessary.

Since OopsFix is designed as a local application, it does not rely on an internet connection, making it accessible even in offline environments. The system is built to be lightweight, meaning it won’t require a powerful computer to run.

3. Specific Requirements

3.1 Use-Case Reports

Accessing OopsFix

When a free user (FU) or paid user (PU) accesses OopsFix, they are able to:

- Submit text for LLM correction (PU only)
- Perform self-corrections (PU only)
- View correction history
- Invite paid users to collaborate (PU only)
- Suggest blacklisted words (PU only)

Upgrading to a Paid User

When a free user upgrades to a paid user, they gain access to:

- Submitting text beyond 20 words
- Accessing LLM-powered corrections
- Saving corrected text for later use
- Purchasing tokens for continued AI assistance
- Inviting collaborators to edit shared text files

Token Management

When a paid user purchases tokens, they:

- Select the number of tokens to buy
- Complete the transaction and receive the tokens
- Use tokens for text submission, LLM corrections, and file-saving

When a paid user runs out of tokens mid-session, they:

- Use the Emergency Token Borrowing system
- Borrow up to 10 tokens, repayable within 48 hours
- Continue using OopsFix features during the borrowing period

When a paid user attempts to submit text without enough tokens, they:

- See a submission failure notification
- Have half of their remaining tokens deducted as a penalty
- Are prompted to purchase additional tokens before resubmitting

LLM Corrections & Editing

When a user accidentally accepts an LLM correction they didn’t intend to, they:

- Click “Undo” within 30 seconds
- Have the original text restored without token deductions
- Continue editing without applying the unintended change

When a paid user submits text containing blacklisted words, the system:

- Replaces restricted words with ‘*’
- Deducts tokens based on the number of blacklisted words
- Notifies the user about restricted content

When a user reviews their text using the Oops Meter, they:

- See a live heatmap of error density
- Watch the heatmap adjust as corrections are made
- Analyze their writing strength visually

When a user wants to understand why a correction was suggested, they:

- Click “Why?” to reveal LLM’s reasoning
- View grammatical or stylistic explanations
- Improve writing by learning correction logic

Collaboration & User Interactions

When a paid user invites another user to collaborate, the invited user:

- Receives a collaboration request
- Can accept or reject the invitation
- Gains shared editing access if accepted
- Causes the inviter to lose 3 tokens if rejected

When a user wants to save their corrected text, they:

- Click “Save” after finishing corrections
- Are charged 5 tokens for file storage
- Can access the saved document later

Super User Management

When a super user (SU) manages the system, they:

- Approve or deny user registrations
- Manage blacklisted words by reviewing flagged terms
- Resolve user complaints and disputes
- Issue warnings, suspend, or fine users who violate policies

When a super user issues a warning to a user, the warned user:

- Receives a violation notification
- Can view past warnings and reasons
- Must comply to avoid penalties

When a user files a complaint against another user or system decision, the system:

- Notifies the accused user and requires a response
- Escalates the complaint to a super user
- Allows the super user to assign penalties or dismiss the complaint

When a paid user submits a well-written text with no errors, they:

- Receive a bonus of 3 tokens for texts over 10 words with no detected mistakes
- Track accuracy and writing progress over time

3.2 Supplementary Requirements

Dispute a Report

A super user (SU) oversees disputes that arise between two parties in the system. There are two cases in which this could happen:

- Case 1 - Paid User (PU1) vs. Paid User (PU2): If PU1 files a complaint against PU2, the SU determines the outcome of the dispute. If PU1 wins, then PU2 receives a warning related to the reported violation. If PU1 loses, they receive a warning for filing a false dispute, and no further action is taken against PU2.
- Case 2 - Free User (FU) vs. Paid User (PU1): If a free user files a complaint against a paid user, the SU reviews the report and makes a decision. If the free user wins the dispute, the paid user receives a warning related to the issue reported. If the free user loses, then no action is taken against either party.

Manage Blacklisted Words List

A super user (SU) is responsible for maintaining the blacklist, which contains words that cannot be used in text submissions. If a paid user submits text containing blacklisted words, the system automatically replaces them with asterisks (‘*’) and deducts tokens based on the length of the blacklisted words. If a user believes a blacklisted word should be removed, they can file a request for review, and the SU decides whether to keep or remove the word.

View Warnings

Warnings issued by the super user to a paid user will occur in various situations:

- If a paid user submits text containing blacklisted words, they receive a warning in their profile and tokens are deducted.
- If a paid user attempts to submit text without sufficient tokens, they receive a warning and must purchase more tokens to continue.
- If a paid user rejects valid LLM corrections with invalid reasoning, they may receive a warning and additional token deductions.
- If a paid user files a complaint that is deemed false or malicious, they receive a warning for abuse of the complaint system.

These warnings can be viewed in the user’s profile, where they can track past violations and penalties.

Token Expiration and Refund Requests

Paid users who do not use their tokens for an extended period may have a percentage of unused tokens deducted as an inactivity fee. Users can request a refund for expired tokens, but only the super user can approve or deny these requests. The refund request must be filed within a set time frame, and users may receive a partial refund based on system policies.

Automated System Logs for User Actions

To maintain fairness and transparency, OopsFix keeps automated logs of user actions, including:

- Corrections made by the LLM and whether they were accepted or rejected.
- Token transactions, including purchases, deductions, and borrowing history.
- Warnings issued by the super user and their corresponding reasons.
- Complaint and dispute resolutions, showing which user was found at fault.

4. Supporting Information