Hazard Analysis
Software Engineering

Team 21, Alkalytics
Sumanya Gulati
Kate Min
Jennifer Ye
Jason Tran

Table 1: Revision History

Date	**Developer(s)**	**Change**
Date1	Name(s)	Description of changes
Date2	Name(s)	Description of changes
...	...	...

Contents

[You are free to modify this template. —SS]

1 Introduction

[You can include your definition of what a hazard is here. —SS]

2 Scope and Purpose of Hazard Analysis

The purpose of the hazard analysis is to identify potential hazards and its causes, assess the effects, and set mitigation strategies to eliminate or lessen the risk of the hazard. It is important to consider all possible hazards associated with the Alkalytics project and its components, as the following losses could be incurred from an insufficient hazard assessment:

- Data loss: The data gets unintentionally lost, deleted, or corrupted.
- Data integrity: The data is not complete, accurate, nor correct which can lead to computational errors.
- System availability: Users are unable to access the system due to factors such as unstable internet connection or server crashes.
- Security: If there are no proper measures for authentication, unauthorized access to the data or user information may occur.

3 System Boundaries and Components

[Dividing the system into components will help you brainstorm the hazards. You shouldn't do a full design of the components, just get a feel for the major ones. For projects that involve hardware, the components will typically include each individual piece of hardware. If your software will have a database, or an important library, these are also potential components. —SS]

4 Critical Assumptions

[These assumptions that are made about the software or system. You should minimize the number of assumptions that remove potential hazards. For instance, you could assume a part will never fail, but it is generally better to include this potential failure mode. —SS]

5 Failure Mode and Effect Analysis

[Include your FMEA table here. This is the most important part of this document. —SS] [The safety requirements in the table do not have to have the prefix

SR. The most important thing is to show traceability to your SRS. You might trace to requirements you have already written, or you might need to add new requirements. —SS] [If no safety requirement can be devised, other mitigation strategies can be entered in the table, including strategies involving providing additional documentation, and/or test cases. —SS]

6 Safety and Security Requirements

[Newly discovered requirements. These should also be added to the SRS. (A rationale design process how and why to fake it.) —SS]

7 Roadmap

The following table outlines a proposed roadmap of when each safety requirement will be implemented within the capstone timeline and justifications.

Stage	Req. Category	Req. ID(s)	Rationale
PoC Demo (Nov 11)	Access	N/A	The PoC plan will not consider user access features at this time.
	Integrity	SR-5, SR-6, SR-7, SR-8, SR-9, SR-10	The database must adhere to these requirements for a successful PoC.
	Privacy Audit Immunity	N/A N/A N/A	Since the PoC plan will only have the database these requirements are not applicable.
Rev0 Demo (Feb 3)	Access	SR-1	User authentication should be implemented during front-end development.
	Integrity	N/A	The crucial integrity requirements will have already been implemented by the PoC demo.
	Privacy	SR-12, SR-13	At this point there will be user access, thus these requirements must be implemented.
	Audit Immunity	N/A N/A	These requirements are not high-priority.

Stage	Req. Category	Req. ID(s)	Rationale
Rev1 Final Demo (Mar 24)	Access	SR-2, SR-3	User access must be extended to permissions and capabilities prior to release.
	Integrity	N/A	The crucial integrity requirements will have already been implemented by the PoC demo.
	Privacy	SR-14	This requirement is necessary for system availability and robustness to extend past capstone.
	Audit	SR-17	Client and users must be informed about system issues and be able to troubleshoot even without the original development team.
	Immunity	SR-19	This requirement is necessary for system availability and robustness to extend past capstone.
Future considerations	Access	SR-4	This requirement is out of scope for the project timeline.
	Integrity	SR-11	This requirement is out of scope for the project timeline.
	Privacy	N/A	All privacy requirements have been covered.
	Audit	SR-15, SR-16	These requirements would be nice-to-haves but are not essential for the project.
	Immunity	SR-18	This requirement is out of scope for the project timeline.

Table 2: Roadmap of the implementation of the safety and security requirements.

Appendix — Reflection

[Not required for CAS 741 —SS]

1. What went well while writing this deliverable?

2. What pain points did you experience during this deliverable, and how did you resolve them?

3. Which of your listed risks had your team thought of before this deliverable, and which did you think of while doing this deliverable? For the latter ones (ones you thought of while doing the Hazard Analysis), how did they come about?

4. Other than the risk of physical harm (some projects may not have any appreciable risks of this form), list at least 2 other types of risk in software products. Why are they important to consider?