**User Story: Feedback with Emotional Context**

*“As a user of this system (student, professor, administrator, etc.) when I encounter a problem, I want to communicate not only the technical details but also how it makes me feel, so that the impact of the problem is fully understood.”*

**Story Tasks**

1. **Research & Requirement Gathering**

* Conduct surveys/interviews with the primary user groups (students, professors, administrators) to understand their needs regarding feedback with emotional context.
* Analyze current feedback mechanisms in the system to identify limitations.

1. **Design**

* Design a feedback form UI that has fields for technical details and emotional context.
* Include options for emotional descriptors (e.g., frustrated, confused, satisfied) or provide a space for users to describe in their own words.
* Ensure the design is intuitive and easy to navigate for all user groups.
* Incorporate visual aids, like emoticons or sentiment sliders, to help convey emotions.
* Review and iterate the design based on user and stakeholder feedback.

1. **Backend Development**

* Create a new database table or modify an existing one to store the emotional context data.
* Implement API endpoints to handle the submission and retrieval of feedback with emotional context.
* Ensure data validation and sanitation for security purposes.

1. **Frontend Development**

* Implement the new feedback form UI based on the approved design.
* Integrate the form with backend API endpoints.
* Handle form validation, error messages, and success confirmation.

1. **Testing**

* Conduct unit tests for the new backend functionalities.
* Perform integration tests to ensure the frontend and backend work seamlessly.
* Arrange user acceptance testing sessions with representative users from each group.
* Collect feedback and iterate on any issues or improvements identified.

1. **Documentation**

* Update user manuals or guides to include instructions on how to provide feedback with emotional context.
* Document the backend changes for future development reference.

1. **Training & Communication**

* Organize training sessions or workshops for users to understand the new feedback system.
* Communicate the changes via emails, system notifications, or other relevant channels.

1. **Deployment**

* Plan and schedule the deployment of the new feature.
* Monitor system performance and user feedback post-deployment to ensure everything operates as expected.

1. **Monitoring & Feedback Loop:**

* Set up a mechanism to regularly review the emotional context feedback to understand the impact of system issues.
* Use the feedback to inform future system improvements or user support needs.

**Functional Requirements**

1. **Feedback Form**

* The system shall provide a feedback form accessible to all user groups: students, professors, administrators
* The feedback form shall include fields for capturing screenshots of specific information.
* The feedback form shall include options for capturing emotional context, they can choose to answer through predefined emotional descriptors (e.g., frustrated, confused, satisfied) or a free-text field.
* The feedback form shall show visual aids like emoticons or sentiment sliders to help convey emotions.
* The feedback form should provide an auto-correct feature to help the user write grammatically correct feedback.
* The system shall provide a way to provide feedback anonymously and only allow users with a UMaine domain send messages.
* The system should offer a "Feedback" feature accessible from the user's public profile.
* The feedback feature should allow users to provide comments either directly or through a structured format provided by a feedback form.
* The system should give the user a way to thread different types of feedback
* The system should return feedback back to the sender in under 3 seconds once completed by user to confirm.
* The system should provide a way to use previous feedback to help answer similar feedback

1. **Backend Integration**

* The system shall store feedback submissions, including both technical details and emotional context
* The system shall use REST API for feedback submission and retrieval.

1. **Error Handling**

* The system shall display a red notification bar and won’t let the user submit if certain ports feedback form are not filled out or if there's the submission does not get sent to the server.
* Upon successful submission of the feedback form, the system shall provide a confirmation message to the user.

1. **Documentation & Guidance**

* The system shall offer an accessible user manual or guide detailing how to provide feedback with emotional context.

1. **Manual Feedback**

* The system shall provide users a way to report accounts which don’t consistently meet the general guidelines listed via the user manual
* The system shall provide a way to moderate users who aren’t providing constructive feedback or are harassing students/professors/administrators.

**Non-functional Requirements**

1. **Usability**

* The feedback form design should be intuitive and easy to navigate for all user groups, ensuring that users can express both technical and emotional feedback.
* Visual aids used for capturing emotional context should use common cultural symbols that 90% of users interpret correctly measured by a survey.
* The system must validate and sanitize submitted data to prevent malicious input.

1. **Performance**

* Feedback form submissions should be processed within at least 5 seconds to ensure users aren't kept waiting.
* The system should be able to handle 300 concurrent feedback submissions suffering data loss.

1. **Security**

* Feedback data, especially the emotional context, should be stored using AES-256 encryption to protect user privacy.
* The system should implement measures SQL injection and Cross-site Scripting (XSS)

1. **Scalability**

* As the user base grows, the system should be capable of scaling to accommodate an increasing number of feedback submissions.

1. **Maintainability**

* The codebase, especially areas handling feedback, every function should have comments explaining what it does to ensure ease of future development and maintenance.
* Changes made to the feedback mechanism should be modular to minimize impact on other parts of the system.

1. **Accessibility**

* The feedback form should be accessible to users with disabilities, complying with relevant accessibility standards like WCAG.

1. **Training & Communication**

* Users should messaged that informs them of the new feedback mechanism, ensuring broad adoption.

1. **Feedback Loop Efficiency**

* The system administrators or relevant stakeholders should be able to efficiently review and act upon the feedback, especially the emotional context, to improve the user experience.