|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CSC325 Capstone Team project Rubric and Evaluation | | | | | |
|  | Fail | Approaching | Meet | Exceed | **As a team write a self-evaluation for each criteria below** |
| Project submission  25% | Missing more than two of the required elements (updated SRS document, project source files as a zip file, project GitHub link, UML class diagram and sequence diagrams, demo video of 4-8 minutes showing execution of the project) | Missing one or two of the required elements. Or poorly prepared. Or miss match to the camera-ready proposal | Includes all required elements but with minor errors or omissions. | All required elements are included and well-organized.  Delivered all/most of the content of the camera-ready proposal | We delivered all of the requirements and almost all of the content is ready for the camera-ready proposal. |
| GitHub Submission and Project Management  15% | Little to no evidence of branch creation, commits, pull/merge requests, or project management tools (such as Kanban board). | Basic use of GitHub (few branches, minimal commits), and basic project management. | Regular commits, multiple branches, and good project management practices. | Consistent and meaningful commits, comprehensive branch management, and exemplary project management. | While it was a little slow in the beginning of the semester, we started picking up great progress and making meaningful commits to the GitHub with great project management. |
| UI and Style  15% | UI is non-functional, poorly designed, or significantly lacks user-friendliness. | UI functions but with flaws in design or user experience. | UI is functional, visually appealing, and user-friendly. | Exceptional UI design that is both innovative and user centric. | The UI is visually appealing and user centric. |
| Functionality and Backend Integration  20% | Non-functional backend, severe integration issues, or major bugs. | Backend functions with some bugs or integration issues. | Well-functioning backend with minor bugs or issues. | Flawless backend functionality and seamless integration. | We have an extremely well-functioning backend and we rarely run into bugs or issues. |
| Testing and Software Engineering techniques.  10% | None | Basic use of JUnit and some of the design patterns | Good use of testing, Javadoc, and design patterns. | Utilizes the knowledge discuss in this course such as design patterns, SOLID principles, agile development, and so on | We did a great job utilizing the things we were taught in this course while we were making our project. |
| Self and Peer evaluation  15% | Negative feedback or no feedback received from self and peers. | Mixed feedback, with some concerns noted by peers. | Positive feedback from peer with minor suggestions for improvement. | Exceptional feedback, peers highly commend performance and contribution. | Skip this evaluation- each team member submits an individual evaluation form separately |
| **Extra credit-**Exceeding requirements  10% | The project meets only the basic requirements, with no additional features or technologies beyond what was specified. | Slightly go beyond the requirements. The project includes minor enhancements beyond the basic requirements, such as simple additional features, basic themes, or a single method of additional authentication. | The project surpasses the basic requirements, incorporating advanced features like multiple themes, multiple methods of authentication, or other notable enhancements. | Use technologies not discussed in class while effectively delivering the requirements as such the project not only meets the basic requirements but also integrates technologies or methodologies that were not covered in the course, demonstrating a high level of initiative and technical capability. | We did utilize some technologies and techniques learned outside of this class, our project definitely surpasses the basic requirements, and we have implanted a few notable enhancements. |