CMT313

Requirements and Design

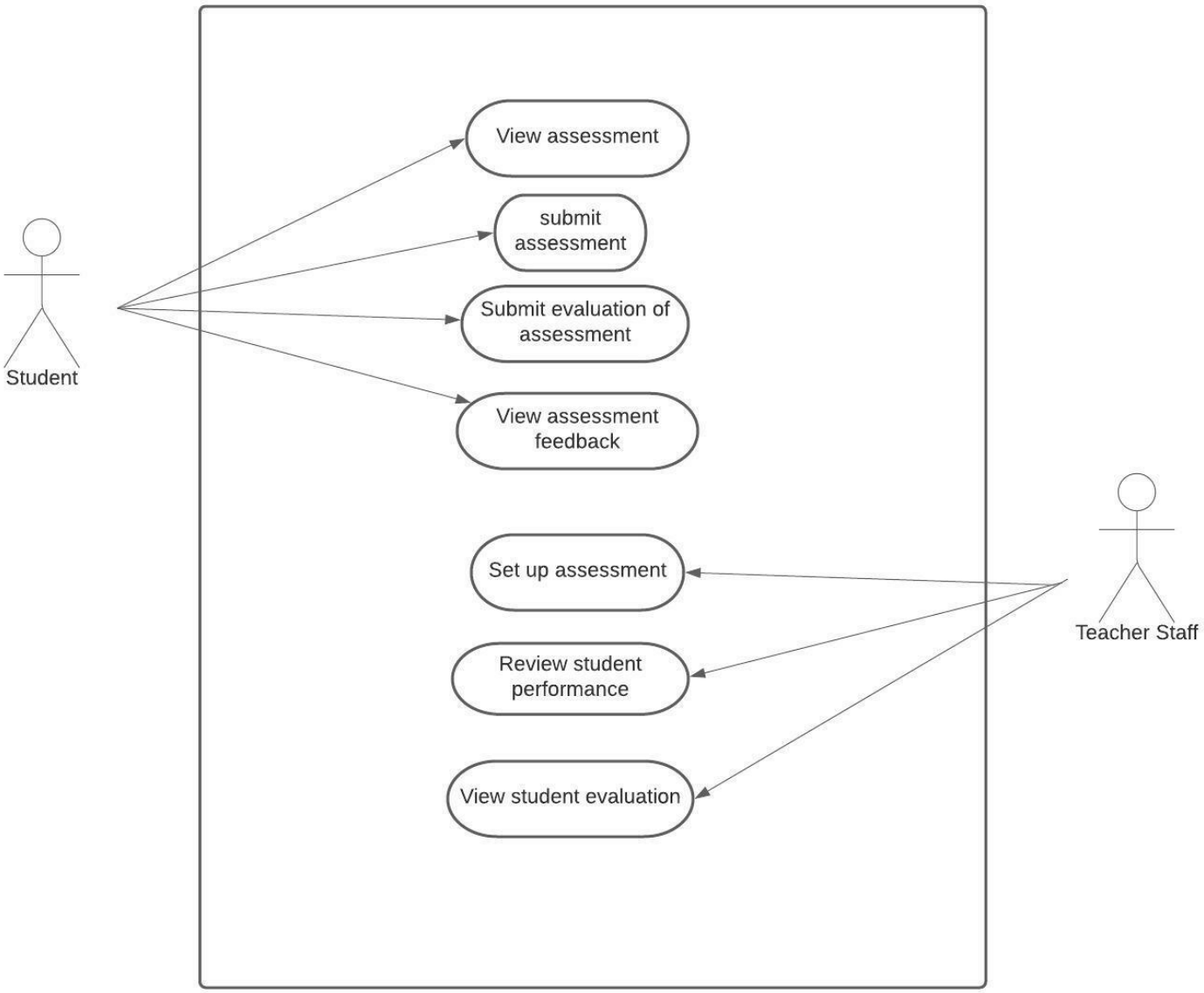
Team R

Team member:

1737516
1887893
21070233
21107384
21110718
21120328

Team members contributed:
1737516 1887893 21070233 21107384 21110718 21120328

Top-level Use Case Diagram



Team members contributed:
1737516 1887893 21070233 21107384 21110718 21120328

User Stories and Acceptance Criteria

Use Case Name	Student ID	Link to GitLab Issue
Student: Submit assessment	21120328	https://git.cardiff.ac.uk/c21070233/teamrautomatedassessmenttool/-/issues/11
Student: Submit evaluation	1737516	https://git.cardiff.ac.uk/c21070233/teamrautomatedassessmenttool/-/issues/9
Student: Access feedback report timely	21107384	https://git.cardiff.ac.uk/c21070233/teamrautomatedassessmenttool/-/issues/8
Teacher: Set up assessments and review when students receive feedback	21110718	https://git.cardiff.ac.uk/c21070233/teamrautomatedassessmenttool/-/issues/10
Teacher: Access student's evaluation	1887893	https://git.cardiff.ac.uk/c21070233/teamrautomatedassessmenttool/-/issues/7
Teacher: Review student performance	21070233	https://git.cardiff.ac.uk/c21070233/teamrautomatedassessmenttool/-/issues/6

Team members contributed:
1737516 1887893 21070233 21107384 21110718 21120328

Non-Functional Requirements

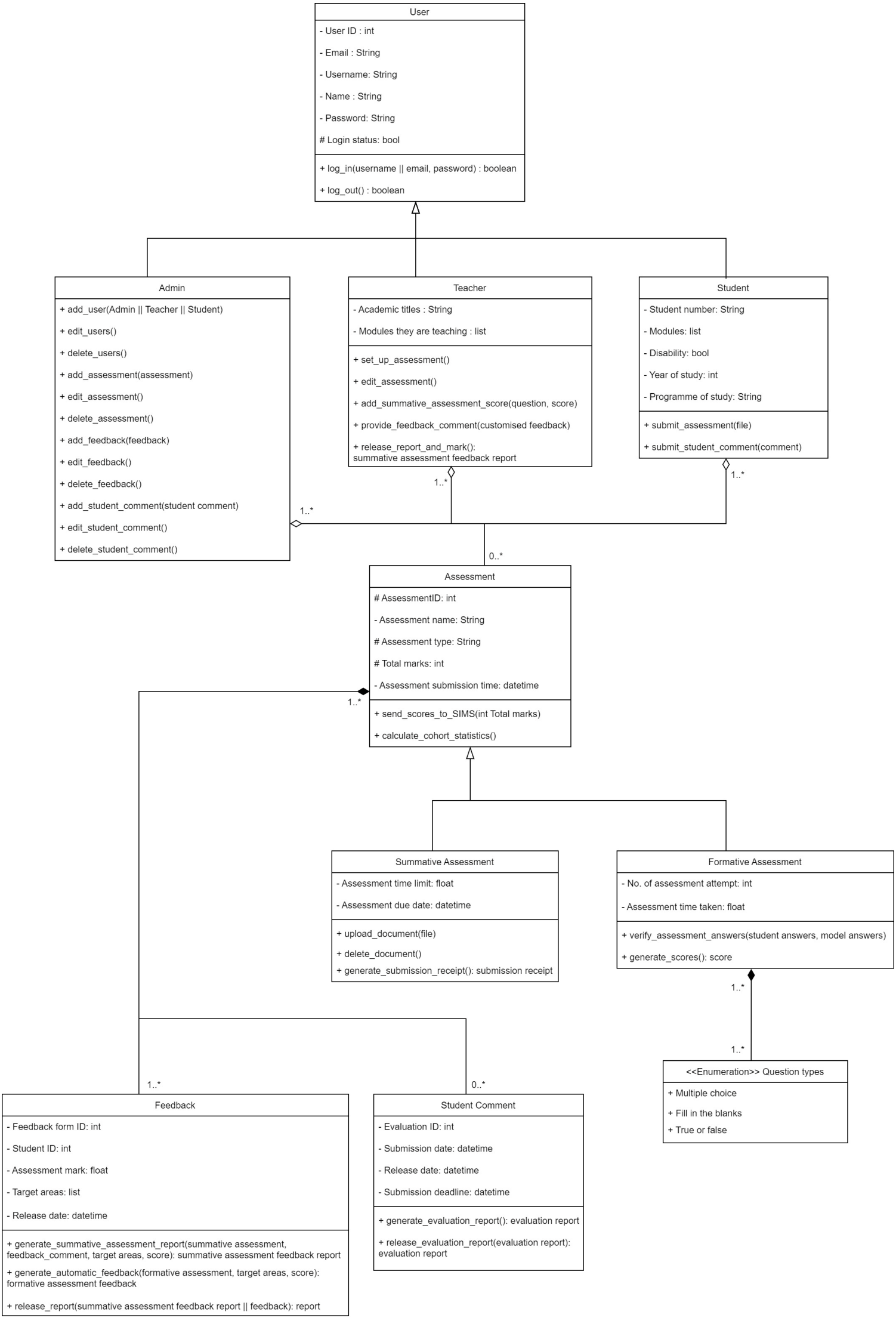
NFR Category	Non-Functional Requirements
Security	1. Database security must satisfy HIPPA requirements (Rome 2020). 2. After three failed login attempts, the account will be locked until user identification is verified. 3. User passwords must contain at least six characters and be a mix of upper and lower case characters along with at least one number and one symbol. 4. User password must not match with vital user information or previously used passwords and not viewable. 5. Student's assessments and feedback can only be viewed by the authorized individual. 6. Access permission to data and roles can be changed by admin only.
Usability	1. Website navigation is intuitive, and users can reach their destination within three steps. 2. The website should load in 3 seconds when there are more than 1000 users are online at the same time. 3. The result of the automatic assessment should be returned to students in 30 seconds.
Availability	1. The system is available for teachers or students 24/7. 2. The system utilises CI/CD so that it is always available to its users (Red Hat 2019).
Maintainability	1. System wide stability testing conducted on a weekly basis to ensure software functionality. 2. Monthly system updates should be provided to fix bugs and improve user experience.
Capacity	1. The system should be able to support at least 300 users[1] submitting or doing their assessments at the same time. 2. The system capacity should scale with cohort sizes. 3. Students will be directed to a virtual waiting room if the capacity of the system is exceeded. 4. File uploads are limited to under 20MB (Cardiff School of Computer Science & Informatics [no date]) and certain file types.

Accessibility	1. System complies with guidance set out in WCAG 2.1 (Hughes 2021).
Recoverability	1. System has a recovery point objective of five minutes (Hughes 2021). 2. System has a recovery time objective of 2 hours (Hughes 2021). 3. System design is compatible with backup tools to back up the whole system e.g., NinjaOne, Veeam (Hughes 2021).
Portability	1. The system is compatible with different web browsers like Google Chrome, Internet Explorer, Firefox, Safari, etc. 2. User interface and features remains the same across different web browsers. 3. Timestamps are recorded in UTC when uploading assessments (George et al. 2022).

[1] The number was estimated from the number of students attending the online lectures on teams/zoom.

Team members contributed:
1737516 1887893 21070233 21107384 21110718 21120328

UML Class Diagram


User
- User ID : int
- Email : String
- Username: String
- Name : String
- Password: String
Login status: bool
+ log_in(username || email, password) : boolean
+ log_out() : boolean
Admin
+ add_user(Admin || Teacher || Student)
+ edit_users()
+ delete_users()
+ add_assessment(assessment)
+ edit_assessment()
+ delete_assessment()
+ add_feedback(feedback)
+ edit_feedback()
+ delete_feedback()
+ add_student_comment(student comment)
+ edit_student_comment()
+ delete_student_comment()
Teacher
- Academic titles : String
- Modules they are teaching : list
+ set_up_assessment()
+ edit_assessment()
+ add_summative_assessment_score(question, score)
+ provide_feedback_comment(customised feedback)
+ release_report_and_mark():
summative assessment feedback report
Student
- Student number: String
- Modules: list
- Disability: bool
- Year of study: int
- Programme of study: String
+ submit_assessment(file)
+ submit_student_comment(comment)
1..*
1..*
1..*
1..*
0..*
Assessment
AssessmentID: int
- Assessment name: String
Assessment type: String
Total marks: int
- Assessment submission time: datetime
+ send_scores_to_SIMS(int Total marks)
+ calculate_cohort_statistics()
1..*
Summative Assessment
- Assessment time limit: float
- Assessment due date: datetime
+ upload_document(file)
+ delete_document()
+ generate_submission_receipt(): submission receipt
Formative Assessment
- No. of assessment attempt: int
- Assessment time taken: float
+ verify_assessment_answers(student answers, model answers)
+ generate_scores(): score
1..*
1..*
<<Enumeration>> Question types
+ Multiple choice
+ Fill in the blanks
+ True or false
1..*
0..*
Feedback
- Feedback form ID: int
- Student ID: int
- Assessment mark: float
- Target areas: list
- Release date: datetime
+ generate_summative_assessment_report(summative assessment, feedback_comment, target areas, score): summative assessment feedback report
+ generate_automatic_feedback(formative assessment, target areas, score): formative assessment feedback
+ release_report(summative assessment feedback report || feedback): report
Student Comment
- Evaluation ID: int
- Submission date: datetime
- Release date: datetime
- Submission deadline: datetime
+ generate_evaluation_report(): evaluation report
+ release_evaluation_report(evaluation report): evaluation report

References

Cardiff School of Computer Science & Informatics. [no date]. *Submitting Assignments Electronically Through Learning Central*. Available at: https://docs.cs.cf.ac.uk/notes/submitting-assignments-electronically-learning-central/. [Accessed: 16 February 2022].

George, A.D. et al. 2022. *Converting times between time zones*. Microsoft Documentation. Available at: https://docs.microsoft.com/en-us/dotnet/standard/datetime/converting-between-time-zones [Accessed: 22 February 2022].

Hughes, D., 2021. *Non-Functional Requirement Examples*. Linkedin.com. Available at: https://www.linkedin.com/pulse/non-functional-requirement-examples-dan-hughes/?trk=articles_directory [Accessed: 22 February 2022].

Red Hat. 2019. *What is a CI/CD pipeline?*. Available at: https://www.redhat.com/en/topics/devops/what-cicd-pipeline [Accessed: 23 February 2022].

Rome, P., 2020. *What are Non Functional Requirements — With Examples | Perforce Software*. Perforce Software. Available at: https://www.perforce.com/blog/alm/what-are-non-functional-requirements-examples [Accessed: 9 February 2022].