MINIGENIUS

SOFTWARE ENGINEERING

Design Specification Document
Dr. Sherif Aly
Spring 2023

Done by:

Nada Badawi   nada_badawi@aucegypt.edu
Salma Aly   itissalma@aucegypt.edu
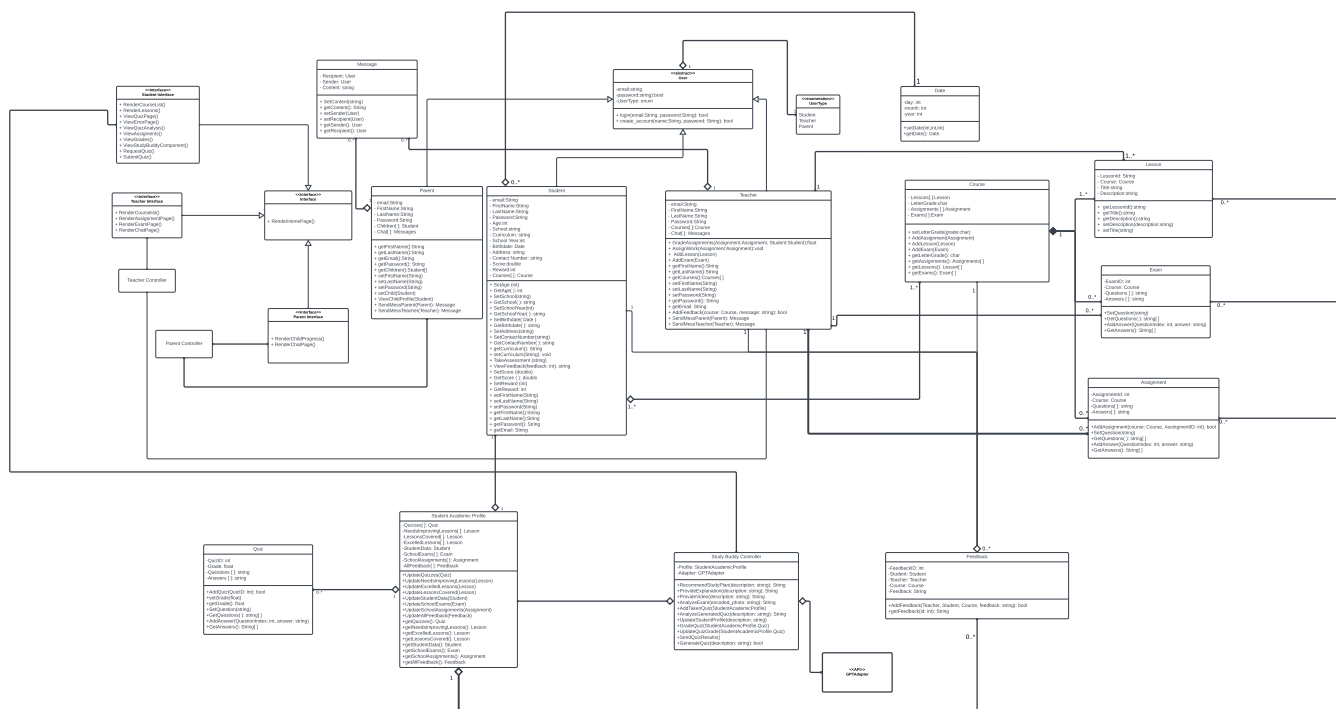Nermien Elassy   nelagamy@aucegypt.edu
Muhammad El Mahdi
muhammadahmedelmahdy@aucegypt.edu

Table of Contents

Section 1

Class Diagram

UML Class Diagram

<<interface>> Student Interface
+ RenderCoursela(id)
+ RenderLessons()
+ ViewQuizPage()
+ ViewErrorPage()
+ ViewQuizAnalysis()
+ ViewAssignments()
+ ViewGrades()
+ ViewStudyBuddyComponent()
+ RequestQuiz()
+ SubmitQuiz()

<<interface>> Teacher Interface
+ RenderCourselda()
+ RenderAssignmentPage()
+ RenderExamPage()
+ RenderChatPage()

Teacher Controller

Parent Controller

<<interface>> Interface
+ RenderHomePage()

<<interface>> Parent Interface
+ RenderChildProgress()
+ RenderChatPage()

Message
- Recipient: User
- Sender: User
- Content: string
+ SetContent(string)
+ getContent(): String
+ setSender(User)
+ setRecipient(User)
+ getSender(): User
+ getRecipient(): User

Parent
- email:String
- FirstName:String
- LastName:String
- Password:String
- Children[ ]: Student
- Child[ ]: Messages
+ getFirstName(): String
+ getLastName() String
+ getEmail() String
+ getPassword(): String
+ getChildren(): Student
+ setFirstName(String)
+ setLastName(String)
+ setPassword(String)
+ setChild(Student)
+ ViewChildProfile(Student)
+ SendMessParent(Parent): Message
+ SendMessTeacher(Teacher): Message

Student
- email:String
- FirstName:String
- LastName:String
- Password:String
- Age:int
- School:string
- Curriculum: string
- School Year:int
- Birthdate: Date
- Address: string
- Contact Number: string
- Score:double
- Reward:int
- Courses[ ]: Course
+ SetAge (int)
+ GetAge(): int
+ SetSchool(string)
+ GetSchool(): string
+ SetSchoolYear(int)
+ GetSchoolYear( ): int
+ SetBirthdate(Date)
+ GetBirthdate( ): string
+ SetAddress(string)
+ SetContactNumber(string)
+ GetContactNumber(): string
+ getCurriculum(): String
+ setCurriculum(String): void
+ TakeAssessment (string)
+ ViewFeedback(teacher, int): string
+ SetScore (double)
+ GetScore ( ): double
+ SetReward (int)
+ GetReward: int
+ setFirstName(String)
+ setLastName(String)
+ setPassword(String)
+ getFirstName(String)
+ getLastName(String)
+ getPassword(): String
+ getEmail: String

Teacher
- email:String
- FirstName:String
- LastName:String
- Password:String
- Courses[ ]:Course
- Exams[ ]:Exam
+ GradeAssignments(Assignment: Assignment, Student: Student):float
+ AssignWork(Assignment:Assignment):void
+ AddLesson(Lesson)
+ AddExam(Exam)
+ getFirstName() String
+ getLastName() String
+ getCourses() Courses[ ]
+ setFirstName(String)
+ setLastName(String)
+ setPassword(String)
+ getEmail: String
+ AddFeedback(course: Course, message: string): bool
+ SendMessParent(Parent): Message
+ SendMessTeacher(Teacher): Message

<> User
- email:string
- password:string;bool
- UserType: enum
+ login(email:String, password:String): bool
+ create_account(name:String, password: String): bool

<<enumeration>> UserType
Student
Teacher
Parent

Date
-day: int
-month: int
-year: int
+setDate(m,int,int)
+getDate(): Date

Course
- Lessons[ ]:Lesson
- Course: Course
- Assignments [ ]:Assignment
- Exams[ ]:Exam
+ setLetterGrade(grade:char)
+ AddAssignment(Assignment)
+ AddLesson(Lesson)
+ AddExam(Exam)
+ getLetterGrade():: char
+ getAssignments(): Assignments[ ]
+ getLessons(): Lesson[ ]
+ getExams(): Exam[ ]

Lesson
- LessonId: String
- Course: Course
- Title:string
- Description:string
+ getLessonId():string
+ getTitle():string
+ getDescription():string
+ setDescription(description:string)
+ setTitle(String)

Exam
- ExamId: int
- Course: Course
- Questions [ ]: string
- Answers [ ]: string
+SetQuestions(string)
+GetQuestions( ): string [ ]
+AddAnswer(QuestionIndex: int, answer: string)
+GetAnswer(): String[ ]

Assignment
-AssignmentId: int
-Course: Course
-Questions[ ]: string
-Answers [ ]: string
+AddAssignment(course: Course, AssignmentID: int): bool
+SetQuestion(string)
+GetQuestions(): string [ ]
+AddAnswer(QuestionIndex: int, answer: string)
+GetAnswers( ): String[ ]

Quiz
-QuizID: int
-Grade: float
-Questions [ ]: string
-Answers [ ]: string
+AddQuiz(QuizID: int): bool
+setGrade(float)
+getGrade(): float
+GetQuestions(string)
+GetQuestions( ): string [ ]
+AddAnswer(QuestionIndex: int, answer: string)
+GetAnswers( ): String[ ]

Student Academic Profile
-Quizzes[ ]: Quiz
-RecentImprovingLessons[ ]: Lesson
-LessonsCovered [ ]: Lesson
-ExcelledLessons[ ]: Lesson
-StudentData: Student
-SchoolExams [ ]: Exam
-SchoolAssignments [ ]: Assignment
-AllFeedback[ ]: Feedback
+UpdateOutcome(Quiz)
+UpdateRecentImprovingLessons(Lesson)
+UpdateExcelledLessons(Lesson)
+UpdateLessonsCovered(Lesson)
+UpdateStudentData(Student)
+UpdateSchoolExams(Exam)
+UpdateSchoolAssignments(Assignment)
+UpdateAllFeedback(Feedback)
+getQuizzes(): Quiz
+getRecentImprovingLessons(): Lesson
+getExcelledLessons(): Lesson
+getLessonsCovered(): Lesson
+getStudentData(): Student
+getSchoolExams(): Exam
+getSchoolAssignments(): Assignment
+getAllFeedback(): Feedback

Study Buddy Controller
-Profile: StudentAcademicProfile
-Adapter: GPTAdapter
+RecommendStudyPlan(description: string): String
+ProvideExplanation(description: string): String
+ProvideVideos(description: string): string
+AnalyzeExam(student, photo: string): String
+AddTutorQuiz(StudentAcademicProfile)
+AnalyzeGenerate(Quiz)(description: string): String
+UpdateStudentProfile(description: string)
+CreateQuiz(StudentAcademicProfile:Quiz)
+UpdateQuizGrade(StudentAcademicProfile:Quiz)
+SendQuizResults()
+GenerateQuiz(description: string): bool

Feedback
-FeedbackID: int
-Student: Student
-Teacher: Teacher
-Course: Course
-Feedback: String
+AddFeedback(Teacher, Student, Course, feedback: string): bool
+getFeedback(id: int): String

<<API>> GPTAdapter
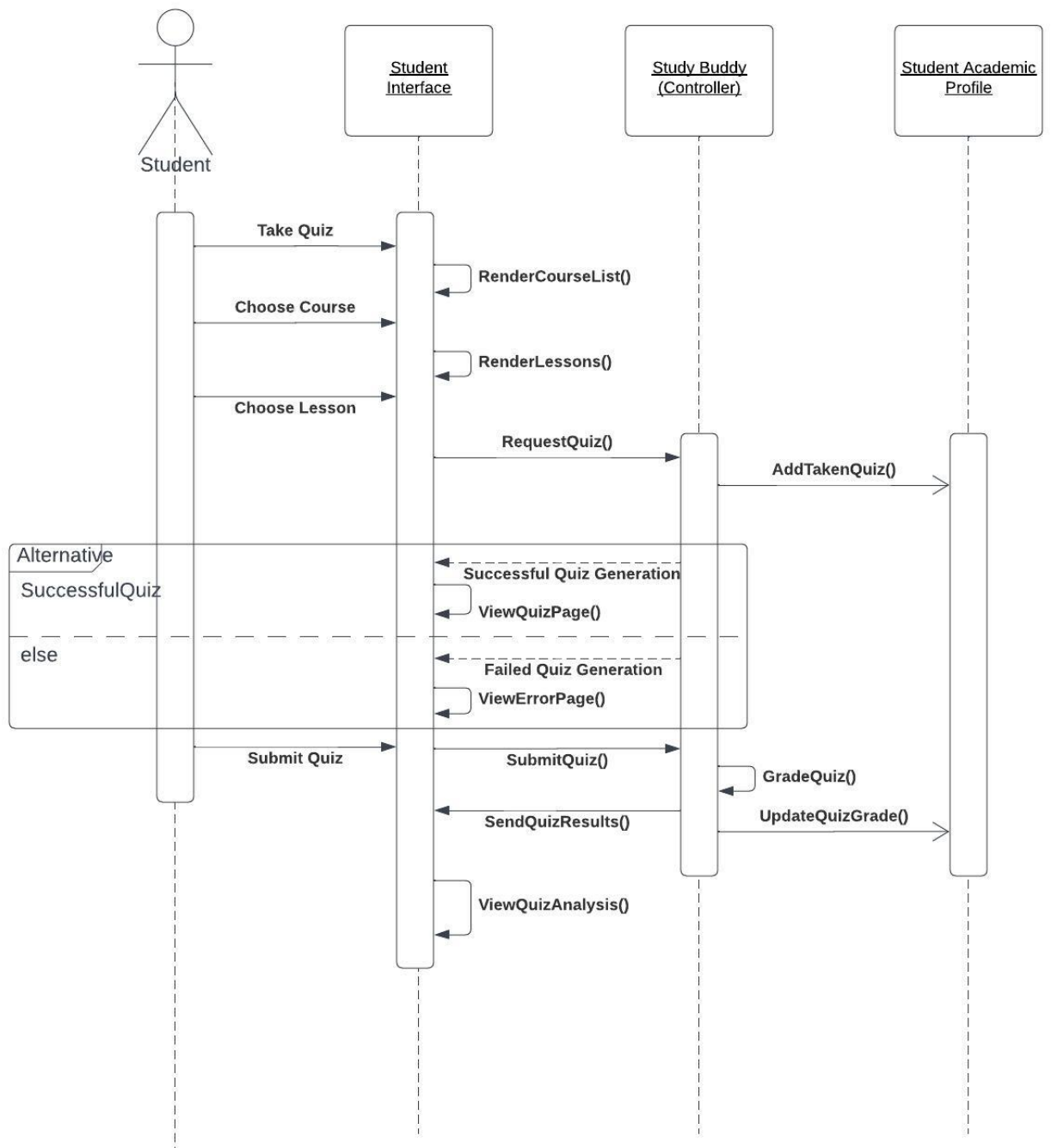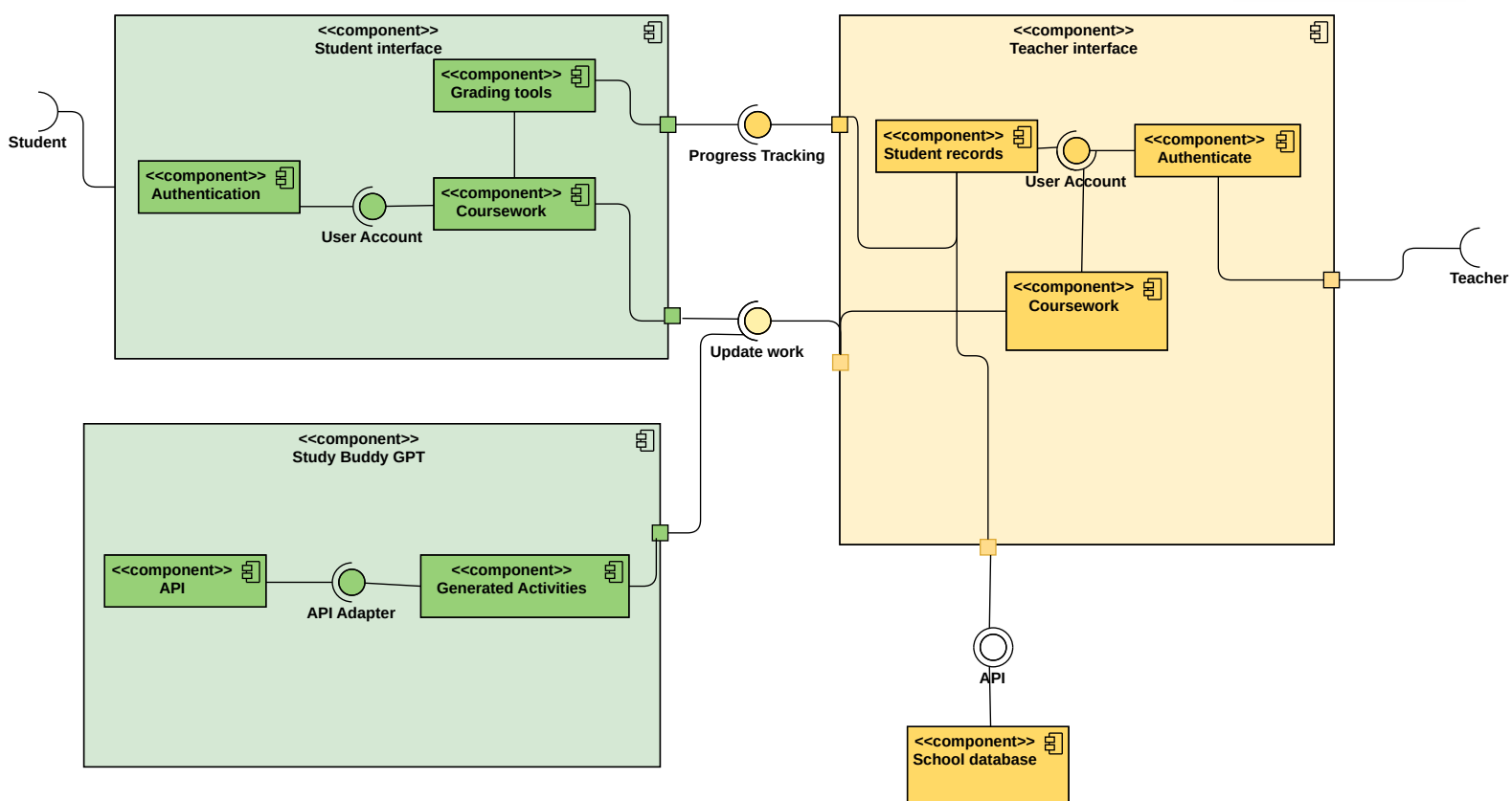
Section 2

Sequence Diagram

The following is a sequence diagram representing a single use-case, which describes the process of taking a quiz. This process involves the student requesting the quiz and is displayed after the GPT component generates it.

Section 3

Component Diagram

Section 4

Useful Links

1. Click-Up Sprint Backlog

2. Class Diagram

3. Sequence Diagram

4. Figma Prototypes

5. Github Repository