Use Cases and Logical Architecture

- XID: X00162027
- Name: Milan Murray
- Project Title: TomaNota

Section 1: For Each Use Case:

Title (goal)	Attempt a note learning session
Primary Actor	Student
Story	1. The student navigates to the website 2. The student clicks on the topic they want to study 3. The student attempts to answer each of the prompts on the quiz until completion 4. The student reviews the results from the quiz 1a. The student navigates to the website and signs in 4a. The system stores the results of the session alongside the document

Title (goal)	Create a new note document
Primary Actor	Student
Story	1. The student navigates to the website 2. The student signs in to their account 3. The student clicks an option to add a new note 4. The student types in the contents of the notes 5. The new note is processed correctly and is added to the user's session 5a. The new note is incorrectly formatted and a warning is displayed 5b. The student corrects the errors in formatting

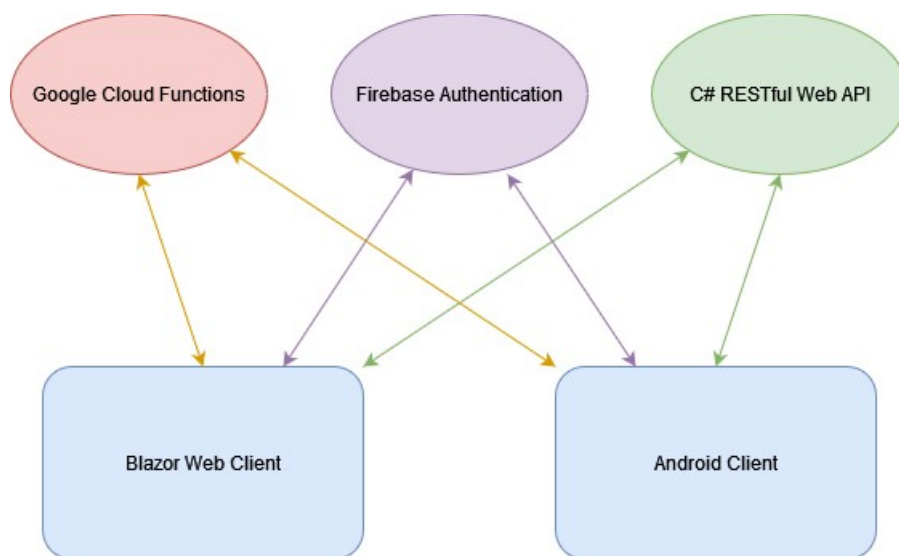| Title (goal)	Review progress and results
Primary Actor	Student
Story	1. The student navigates to the website 2. The student signs in to their account 3. The student navigates to their account page 4. The user is shown a chart containing a history of their progress of note learning

Title (goal)	Create an account
Primary Actor	Student
Story	1. The student navigates to the website 2. The student clicks a button to create a new account 3. The student enters an username, email and password 4. The student clicks a button to create the account 5. The system is able to create the account 3a. The student enters an email and a password 5a. The system is unable to create the account 5b. The student is shown any errors with the text they entered and the student tries again

Title (goal)	Edit an existing document
Primary Actor	Student
Story	1. The student navigates to the website 2. The student signs in to their account 3. The student selects a topic 4. The student attempts a quiz on the selected topic 5. Upon completion the user selects an option to review incorrect answers 6. The student enters an updated answer 7. The student completes the review 8. The student enters a new unique name for the notes with these new changes 9. The student selects an option to save the notes to their account 8a. The student enters an existing title of a note document 9a. The student selects the save option and is asked to confirm if they wish to overwrite an existing note stored on their account

Title (goal)	Remove existing note documents

Primary Actor	Student
Story	1. The student navigates to the website 2. The student signs in to their account 3. The student clicks a topic that is stored on their account 4. The student clicks an option to remove the note from their account 5. The student accepts that this action cannot be undone 6. The system deletes the file

Section 2: Logical Architecture

Logical Architecture Discussion

The student accesses the client via a web browser from a desktop or mobile device, or using the android app. The student is shown available notes available from the google cloud functions. The student can sign in via Firebase Authentication and is shown the notes stored on their account from the C# RESTful Web API. The C# RESTful API connects to an SQL database to access and modify the user's stored data.