GitHub (must be three different people)

Task	Group Member
Create repository and add members	Samuel
Make first commit	Hamza
Make project scope commit	Kelly

Report

Task	Group Member
Task delegation	Everyone
Address feedback	Ashton
Software process model selection	Aryan
Functional requirements	Mathew
Non-functional requirements	Aryan
Use case diagram	Hamza
Sequence diagram	Kelly
Class diagram	Brandon
Architectural design	Samuel

Project Deliverable 2

Task	Group Member
Task delegation	Everyone
Project scheduling	Kelly
Cost/effort/pricing estimation	Aryan
Test plan	Mathew
Comparison with similar designs	Aston
Conclusion	Brandon
Presentation slides	Everyone

Software Process Model selections

Spiral model
- We may want to modify our requirements, and the spiral model will allow us to make changes with each circuit
- We want to build the application in increments so that we have the fundamental features working before adding details
- We want to reduce risk throughout the implementation phase

Architectual model

MVC model

Functional requirements minimum 5 , max 7

Functional:
1.The system shall randomize the order of the flashcards
2. A user shall be able to use their flashcards to play matching games
3. The system shall push reminders to study to users
4. The system shall save user progress after the user studies a card
5. A user shall be able to add and remove flashcards from a deck.
6. A user shall be able to view a deck of flashcards
7. A user shall be able to tag decks with languages

Non-Functional requirements

1.Cards sets can be reandomized in .2 seconds
2. Flashcards load for users game in .1 seconds.
3.Game GUI is loaded in .2 seconds
4.remiiders can contact email server in .05 seconds
5.reminds can push text reminder in 20 seconds
6. User progress is saved within .3 seconds of completion of the game
7.adding and removing decks from flashcards takes 15 seconds.

1. Application boots in 10 seconds
2. table is imported when loaded within 5 seconds
3.
4. Many different types of decks to study
5.
6.
7.

Nonfunctional:
Performance: The system shall randomize card sets in 0.2 seconds
Space: The application shall take up no more than 1 GB of device storage
Usability: A user shall be able to navigate to a flashcard deck within 5 seconds of opening the app
Dependability: The system shall be available 24/7. Downtime within a given day shall not be more than 1 minute.
Security: Users shall authenticate themselves using a username and password.
Environmental: The app should be able to work in iOS and Android operating systems
Operational: The screen refresh time shall not exceed 2 seconds
Development: The system shall remain functional during update deployments
Regulatory: Falls within FCC requirement, Andriod/Apple Appstore requirments
Ethical: user data is not sold, only progress , and account id/ password under hashes
Accounting: payment plan completes in 2 minutes
Safety/Security: Passwords shall be encrypted so that user accounts are secure.

Class Diagram

Classes:
flashCard
flashCardDeck
Decks
Text (stories, essays, etc.)