

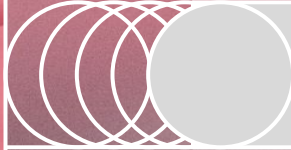


Cos-Aiko

A mobile app for the
cosplay community

Marc Bittle, Aka'sh Carver, Garrett Dittrich, Rohan George
02/04/2022 - UNC-Greensboro - CSC490-02





INTRODUCTION

The previous attempts at services with similar goals have ended up unmanaged, unliked, and unused. We aim to provide con-going artists, designers, speakers, business owners, and more, with a new, fluid, fun, secure, cutting edge mobile application.



Cos-Aiko Features

1.

Plan

Set your goals; Plan your deadlines.

2.

Chat

Speak; Communicate;
Make plans.

3.

Meet

Special meetups; Special Activities.

4.

Try

Try different looks. This is cosplay after all.

5.

Photos

Take photos, that's why we are here.

6.

Sell

Artists; Designers; Dealers.



Project Requirements

- (1) COCKED PISTOL. Maximum readiness. Immediate response.
- (2) FAST PACE. Engineers ready to code a week after next stage.
- (3) ROUND HOUSE. A planned state of readiness.
- (4) DOUBLE TAKE. Lowest state of readiness.
- (5) FADE OUT. This is not our requirement focus/we are not ready.
 - (a) Pre-Login
 - (b) Profile
 - (c) Plan
 - (d) Meet
 - (e) Chat
 - (f) Image Mod+





Functional Requirements 1

Registration **A1** Prompt form for email and password. Accessible in default view. Persistence.

Login **A1** Prompt form for email and password. Accessible in default view.

Account Recovery **A3** Prompt form for email and password for access to an existing account. Accessible from default view.

User Name, Picture, Bio Field **B1** Device storage linkage. Name handling. 25 Char. Alphanumeric.

Distinction **B4** Handling of official accounts. Marker on profile, ability to apply/contact.

Profile View **B1** View personal information and data.

Con-History **B2** Ability to add under profile. Next stage may be necessary.

Favourites **B4** Specialized profile data, linkage to certain APIs.

Settings **B3** View and manage technical aspects of the application. Change data. Feedback.

Bio Tokenization **B3** Recognize key terms in bio for linkage with Meet Page features.

Calendar Page **C1** Ability to plan and mark dates. Possible linkage with Con API.

Push Notifications **C4** Ability to set alarms, display prompts to the Android system.





Functional Requirements 2

Auto Scheduler **C5** Feature to make a auto-schedule to complete a task by deadline.

User Profile Browse **D1** Queryable via con attendances and favorites, a stream of accounts.

Meet Page **D1** Connect to account stream for Front End controls to flow from user to user, linkage to their profile view.

Display Calendar **D1** A different calendar view for public past and future con attendance.

Chat Shortcut **D1** Ability jump to chat page to contact based on user preferences.

Like//Dislike **D2** User can apply likes/dislikes such that future stream will be more attuned to them.

Queryable Stream by User Data on Meet Page **D3** Query account stream by user data from bio entry.

Chat Page **E1** Front End implementation. Chat options in view.

Variant Text Data **E1** Send text, photo, video, audio, and pdf doc data to another account.

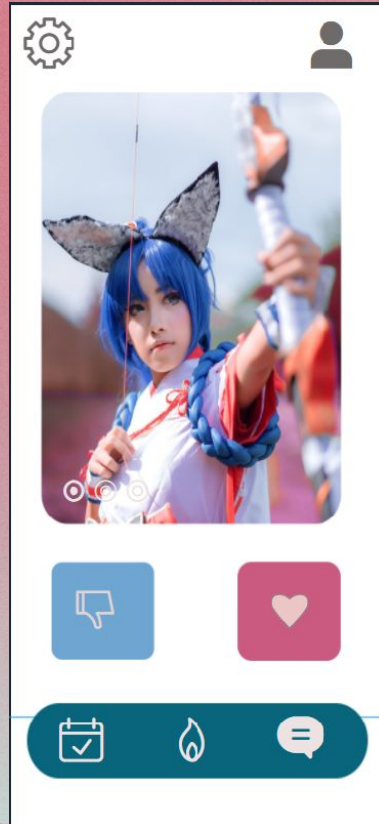
E2EE **E3** No decryption in transit.

Photo Sharing **E2** Linkage to Android camera app.

Image Editor **F1** Filters on camera feed. Resizing, cropping.



User interface Wireframes



Usability Requirements

User Interface:

- Attractive; Fun; Easy; Consistent.
- Proper Error Handling.

Performance:

- Smooth; Quick; Casual.
- Multi-threading.

Accessibility:

- Accommodating settings.



System Requirements

Hardware:

- No hardware will be needed for the app other than servers hosted in azure which will hold backend software and database instances

Database:

- Our app will use Mysql for its database which will be hosted on azure



System Requirements 2

Software:

- Android studio/ sdk for development of the app, libraries needed include Retrofit for REST calls and glide for image loading
- Spring Boot will be used for the backend application to fetch database results and return them to the Android app via REST calls.



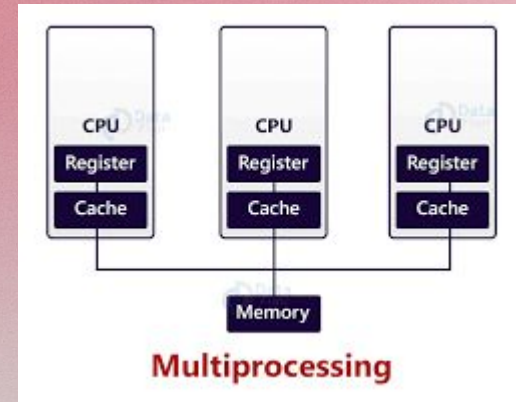
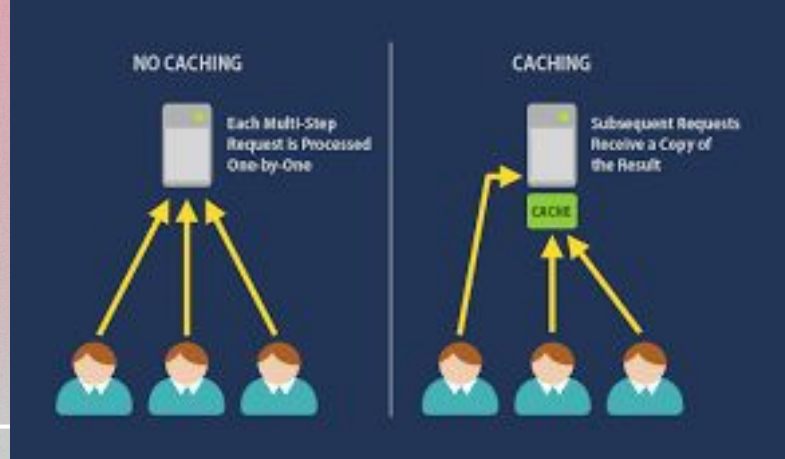
Security

- Cos-Aiko will implement SHA-256 hashing for usernames and passwords which will then be validated in Mysql for all logins.
- Sensitive user data like addresses and messages will never be stored in plain text.
- Spring Boot on the backend will use non-deprecated versions of all needed libraries to avoid vulnerabilities like Log4j
- E2EE for message passing



Performance

- Where possible Cos-Aiko will use multi threading when making api calls or completing cpu intensive tasks.
- The app will also cache database and api requests such that expensive io tasks are kept to a minimum.





Development Environment

- Android Studio IDE, Bumblebee v2021.1.1
- Remote MySQL server (SQL)
- Java 17
- Gradle Build Tools v7.2

Front end: Garrett Dittrich, Rohan George

Back end: Marc Bittle, Aka'sh Carver



Focus/Domain

- The target domain for Cos-Aiko are cosplayers, con-going artists, designers, etc.
- Encouraging social interaction, networking, collaboration, and planning events such as conventions and art shows



Platform/Genre

- This application will be designed to run on all android mobile devices
- Cos-Aiko is a social media application targeting the cosplay niche
- This app will provided a nice ways for cosplay designers to also find potential customers for their services.





A Look Ahead

- Stage III: Plan:

We will be working on models and designs.

Getting everything at least the basic functions up and running.



