Requirements Document

Mobile Apps to Reduce Cross-Race Recognition Deficit

University of Colorado **Boulder**

Department of Psychology and Neuroscience

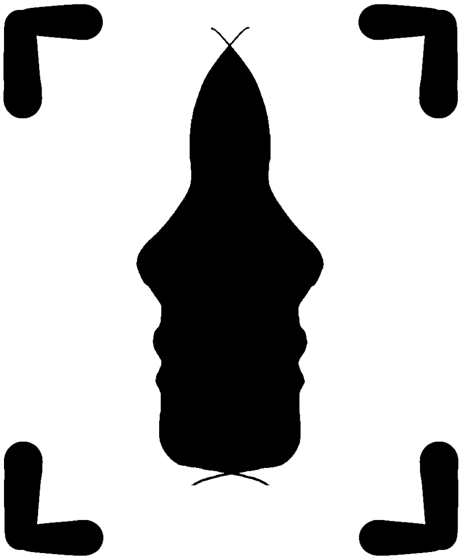

Table of Contents

Requirements

1. Introduction

1.1 Purpose of Document

This is a Requirements Specification document for existing web and mobile based apps for the CU Psychology Department. These apps measure racial bias in human faces. This document describes the scope, objectives and goal of the upgraded application. In addition to describing non-functional requirements, this document models the functional requirements with use cases, and interaction diagrams. This document is intended to direct the design and implementation of the target system in an object-oriented language.

1.2 Project Summary

This project aims to build on last year's work to help publish two mobile apps that help reduce and study CRD. We know that training can work to reduce this deficit. This project aims to be another helpful training assist for users in reducing their own CRD. This app would use various exercises to train users in individuating faces of other races and measure progress over time.

1.3 Background

In 2017, the New York State Court of Appeals in the United States issued a decision that jurors be educated about the unreliable nature of eyewitness testimony in all cases involving cross-race eyewitnesses. Research surrounding cross-race facial recognition suggests that although we are accurate at identifying faces within our own racial group, it can be much more difficult to identify the faces of those in different racial groups. This deficit between the accuracy of identifying faces within our own racial group in comparison to those in other racial groups is referred to as the cross-race deficit (or CRD).

1.4 Project Scope

The scope of this project is a mobile application that provides users the ability to improve on their cross-racial facial recognition. The current web-based application will be captured in a new mobile app interface. This new upgraded application will extend the existing set of facial recognition trainings.

Note: any Changes to the scope must be carefully managed and then signed off on by all members involved.

1.5 Purpose

The goal of the apps is to help reduce the implicit bias between different racial/gender groups. The apps will also help train people to recognize an individual other than merely belong to a group, (i.e. a label). The new system will upgrade the current website and apps to provide users the ability to take daily tests and experiments. These results are used for research purposes by the CU Psychology department.

1.6 Audience

This application has two audience groups. The first group is the participants. The participants can download the app for free from either the app store or the google play store. After they download the app they can use the app to test their cross-race deficit (or CRD). The user has the option to select what race set they want to use and the tests increase in difficulty to keep them engaged. The second participant group is the researchers. The researchers can use the app to track user progress, and collect results. Researchers here can also add faces to increase the size of their data set.

2. Functional Objectives

”Any Requirement Which Specifies What The System Should Do.”

User Story ID	As a <type of user>	I want to <perform some task>	So that I can <achieve some goal>
1	Researcher	Collect and manage user performance data.	Use it for research purposes.
2	Participant	Be able to download the app.	So that they can participate in the training process.
3	Participant	Be able to train using different face sets.	To help participants recognize individuals from different face sets.

4	Participant	Do an 8-day face training.	Improve the ability to distinguish individual faces.
5	Participant	Check the history of participating training.	Track how well I did for the last training and see the improvements.
6	Participant	Be able to create an account.	So that my progress can be tracked.
7	Participant	Do a forced choice assessment before and after the training process.	The progress can be measured (by researcher).
8	Researcher	Increase the data set by adding new faces	Keep the app fresh and produce better results for my research.
9	Researcher	Know how my participants are doing.	Potentially change the difficulty or collect results.
10	Researcher	Have documents that help me use the app.	Fix and understand how the app works.
11	Researcher	Have documents that new developers can use.	Find a new team to continue to improve the apps.
12	Researcher	Have access to lesson plans.	Know what plans are live or change live lesson plans if I wish.
13	Participant	Be able to recover my account.	I lose access to it and need to recover it.

2.1 High Priority

1. Expand the apps to allow for multiple genders and racial groups.
2. Implement/Improve the “Forced choice” task daily assessment. (The user sees a face and then has to pick from 4 images).
    2.1. “Forced choice” should be the final, ultimate, in-the-end-of-day test.
    2.2. Dynamically sample different sets of faces during the training.
3. Increase task difficulty dynamically based on performance.

4. Prepare and publish an iOS app to the App Store (iPhone).
5. Prepare and publish an Android app to the Google Store (Android).

2.2 Low Priority

1. Update the user UI so that the applications are more pleasant to use.
2. Dynamically generate facial sets.
3. Increase task difficulty dynamically via machine learning algorithms.

3. Non-Functional Objectives

"Any Requirement That Specifies How The System Performs A Certain Function."

User Story ID	As a <type of user>	I want to <perform some task>	So that I can <achieve some goal>
4	Developer	Develop an algorithm to pick up faces for 8 sets in one training session.	Modify the training process difficulty.
2	Developer	Get app approval from the App Store and Google Play	Make the app available in the app store for users.
3	Developer	Update the current database to include more face sets	Help users recognize individuals from different face sets.
3	Developer	Update user interface to include more face sets	Help users recognize individuals from different face sets.
5	Developer	Update the current database.	So that the app can handle new app functionality.
6	Developer	Update the training algorithm	Help users to memorize and distinguish individual faces better
7	Developer	Require users to take a "Forced Choice" task	So that user progress can be measured

		when they finish their training.	
8	Developer	Design the database. Design the functions to retrieve information	Help research manage the collected data.
9	Developer	Get access to CU boulders app store and google play store accounts	So that we can publish the app.

3.1 Performance

- By making our code flexible. It should be able to integrate new API's and datasets to expand upon features in the future.

3.2 Usability

- Team members will create lessons bundled into short lessons of progressing difficulties. These lessons will contain hand-curated faces, as opposed to randomly selected ones.
- Each lesson a team member makes will be short enough for the user to complete in a sitting (5-10 minutes). This will be verified by simple testing.
- Develop algorithms so that users can be presented with different facial sets based on performance.
- A newly redesigned and easy to use UI should aim to maximize user engagement.
- A newly redesigned UI should be easy enough to use so that unfamiliar users are not frustrated by the app.
- Our team will make entertaining lessons to maximize user engagement.

3.3 Reliability

- Our team will test and make sure the user apps are stable.
- Our team will test and make sure the researcher portal is stable.
- Our team will test and make sure the database can handle a potentially large user base.
- Our team will test and make sure any third party software and API`s are reliable, stable and have long term support.

3.3 Documentation

- We will create documentation so that future developers can build on the app easily.
- We will improve the existing readme documents.
- We will create documentation as needed.

4. Use Case Model

Login User

Use Case Name:	Login User
Summary:	In order to participate in any of the trainings, a user must login so that the system can track progress and provide appropriate trainings.
Basic Flow:	1. The use case starts when a user selects the login button. 2. The system requests the email and password. 3. The user enters their email and password. 4. The system verifies the email and password against all registered users. 5. The user selects the login button. 6. The system starts a login session and presents the dashboard to the user.
Alternative Flows:	Step 4: If the email or password is invalid, the use case returns to step 2.
Preconditions:	The user is registered.
Postconditions:	The user can now participate in trainings.

Register User

Use Case Name:	Register User
Summary:	In order to participate in any of the trainings, a user must have an account to track their progress and trainings.
Basic Flow:	1. The use case starts when a user selects the register button. 2. The system requests email, password, and password confirmation. 3. The user enters email, password, and confirmation of password. 4. The system requests demographic info (i.e. age, ethnicity, nationality, gender). 5. The user enters demographic info.

	6. The system requests acceptance of terms and conditions. 7. The user accepts the terms and conditions of application. 8. The user selects the register button.
Alternative Flows:	Step 3: If user enters an invalid email, the system prompts the user for a valid email and the use case returns to step 2. Step 3: If user enters different inputs for password and confirm password fields, the system prompts the user for a matching password and the use case returns to step 2.
Preconditions:	*none*
Postconditions:	The user now has a valid account and can login.

Name and Face Training

Use Case Name:	Name and Face Training
Summary:	This is one daily training process, where the user matchs names with corresponding faces.
Basic Flow:	1. The use case starts when the user selects the “Name and Face Training” face training button. 2. The user can choose which face sets to train with. 3. The user will enter the training screen with three sets of name and face showing at the center for 5 seconds, one by one. 4. The screen will show one face and the user will choose the correct name for that face among three names provided. 5. The UI will alert whether or not the choice by the user is correct. 6. The user can then proceed into another training session by swiping left. 7. Repeat step 2-6 until the user completes the series of training sessions for each level.
Alternative Flows:	Steps 3-5: The user may exit the training at any point in the process by selecting the back arrow in the navigation bar and selecting “Quit”. Steps 7: The user may retry the training lesson.
Preconditions:	The user is logged in.
Postconditions:	The user has completed the Name and Face Training.

Who’s New Training

Use Case Name:	Who’s New Face Training
Summary:	This face training works by presenting the user a list of faces at the start of the day and prompting the user to select which face they did not see at the start of the day.
Basic Flow:	1. The use case starts when the user selects the “Who’s New” face training button. 2. The user can choose which face sets to train with. 3. The user is then prompted to select which face out of a given set that was not included in the initial set they were shown at the start of the day. 4. The user will select which face they think was new. 5. The UI will alert the user on whether their selection was correct. 6. The user can then proceed to the next level in the training by swiping left. 7. Repeat steps 2-6 until the user completes each level of the training until all levels are complete. 8. The user swipes left to view their results. 9. The user selects the “Finish” button to end the training. 10. The user will be taken back to the dashboard.
Alternative Flows:	Steps 3-5: The user may exit the training at any point by selecting the back arrow in the navigation bar and selecting “Quit”. Step 9: The user may select “retry” to restart the training.
Preconditions:	The user is logged in.
Postconditions:	The user has completed the Who’s New Face Training.

Memory Match Training

Use Case Name:	Memory Match Training
Summary:	This training works by presenting the user with a set of 16 faces arranged on a board. The user will then be prompted to match the face placement on the board.
Basic Flow:	1. The use case starts when the user selects the “Memory Match” face training button. 2. The user can choose which face sets to train with. 3. The UI will present a 4x4 arrangement of faces for the user to memorize the placement.

	4. The user will have 10 seconds to memorize the face placement on the board. 5. The UI will flip over the faces so that the user cannot see any of the faces. 6. The user will then select their guess for two matching faces. 7. The user's guesses will be flipped over to show which faces they selected. 8. a. If the user selected two matching faces, their score would increase by 4 points. The user's guesses will remain face-up on the board. b. If the user selected two incorrect faces, their score would be penalized by 1 point. The UI will flip the user's guesses back over so the faces are not showing. 9. Repeat steps 5-7 until all the faces have been matched. 10. The user swipes left to view their results. 11. The user selects the "Finish" button to end the training. 12. The user will be taken back to the dashboard.
Alternative Flows:	Steps 5-7: The user may exit the training at any point by selecting the back arrow in the navigation bar and selecting "Quit". Step 11: The user may select "retry" to restart the training.
Preconditions:	The user is logged in.
Postconditions:	The user has completed the Memory Match Training.

Shuffle Training

Use Case Name:	Shuffle Training
Summary:	This training works by presenting the user a set of 4 faces for 10 seconds, and then the user will be prompted to rearrange the shuffled set of faces so that they are in their initial placement.
Basic Flow:	1. The use case starts when the user selects the "Memory Match" face training button. 2. The user can choose which face sets to train with. 3. The UI will present a 2x2 arrangement of faces for the user to memorize the placement. 4. The user will have 10 seconds to memorize the arrangement of the faces. 5. The UI will shuffle the arrangement of the faces. 6. The user will select two cards to swap in order to change the arrangement of the faces.

	7. The user will repeat step 5 until they have completed their guess for the initial arrangement of faces. 8. The user will select "Done". 9. The UI will show the user their result for their guess. 10. The user will swipe left to continue. 11. Repeat steps 3-10 until they have completed all four levels of the training. 12. The user swipes left to view their results for the entire training. 13. The user selects the "Finish" button to end the training.
Alternative Flows:	Steps 5-7: The user may exit the training at any point by selecting the back arrow in the navigation bar and selecting "Quit". Step 10: The user may toggle between "Correct" and "Selected" to show either the correct placement of the faces, or their guess for the placement. Step 11: The user may select "retry" to restart the training.
Preconditions:	The user is logged in.
Postconditions:	The user has completed the Shuffle Training.

Forced Choice Training

Use Case Name:	Forced Choice Training
Summary:	This is the training before and after any training process, in order to provide data for researchers to measure the user progress.
Basic Flow:	1. The use case starts after the user selects any face training button and face sets. 2. The user will enter the training screen and do the forced choice training according to the type of training they chose. 3. The UI will alert the user on whether their selection was correct. 4. The UI will alert the user the forced choice training finished and proceed to the formal part of the training by swiping left. 5. After the user finishes the whole training period, the user needs to do the forced choice training again before the end of training process. 6. The user swipes left to view their results. 7. The user selects the "Finish" button to end the training. 8. The user will be taken back to the dashboard.
Alternative Flows:	Steps 2-5: The user may exit the training at any point by selecting the back

	arrow in the navigation bar and selecting “Quit”.
Preconditions:	The user is logged in and selects any training and face sets.
Postconditions:	The user has completed the Forced Choice Training.

5. Appendix

Requirement Change Control Process

Potential changes will be discussed in a Sponsor meeting. All viewpoints will be heard and respected. Changes to scope/requirements may be considered, but they will require careful, formal, disciplined control. A democratic vote will then be held to decide on the new changes. After which the new results will be documented in an updated Requirements document. Finally, this new document will be signed by all the members of the project.

Contact Information

Any serious questions regarding project requirements should be relayed to our instructor Alan Paradise.

Name	Role	Email
Alan Paradise	Instructor	Alan.Paradise@Colorado.edu
Joshua Correll	Project Owner	Joshua.Correll@colorado.edu
Chelsea Chandler	Teaching Assistant	Chelsea.Chandler@colorado.edu
Madison Rivas	Front-End Developer	Madison.Rivas@Colorado.edu
Liyang Ru	Back-End Lead Developer	Liyang.Ru@Colorado.edu
Alvaro Santillan	Mobile-App Lead Developer	Alvaro.Santillan@Colorado.edu

Guangshi Xu	Machine Learning Lead Developer	Guxu4949@Colorado.edu
Siyu Yao	Front-End Lead Developer	Siya7259@Colorado.edu