Bilkent University

Department of Computer Engineering

Senior Design Project

Project short-name: project Facera

Specifications Report

Group Members: Taha Khurram, Umer Shamaan, Zeynep Berfin Gökalp, Emil Alizada, Verdiyev Zulfugar

Supervisor: Dr. Ayşegül Dündar

Jury Members: Dr. Can Alkan and Dr. Cigdem Gunduz-Demir

Progress/Final Report
October 12, 2020

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2

Table of Contents

1. Introduction

AR (short for augmented reality) is a technique for broadening interactive experience in a natural environment. It gives endless opportunities for manipulating the perception of reality. With today's increasing technological capabilities, AR is becoming more useful and popular. Its potential to give intuitive user experience provides developers with many possibilities to create effective and useful platforms especially in the field of entertainment.

We decided to create a video streaming platform that can enhance the user experience by utilizing aspects of the AR. The main purpose of our platform is to break through the conventional methods of video streaming and harness the power of AR, to make video streaming much more entertaining. This will also open in a new genre of video calling which will feature 3D models in an interactive environment through AR instead of the conventional video call.

1.1. Description

Facera is an AR-based video conferencing app which aims to give a fresh breath to already outdated video calls utilizing cutting edge technologies in order to make video calls more memorable and feeling to people. The app provides users with a unique ability to place a 3D frame that streams the video call anywhere.

The app will reframe old-style video conferencing by using AR. Through this app, the user will call another user by using his front camera to record his face. On the other end, the callee will be able to project a 3D frame with an ongoing video stream onto any surface therefore making it appear in the room of a callee from the back camera. Such an approach will be more pleasing for the users as it makes the caller appear close to the callee and adds interactive features to the call.

The main goal of this project is to set the direction for new video call apps by utilizing the AR and perhaps combining it with AI in the future in order to save bandwidth.

1.2. Constraints

The application development will be subjected to the following constraints:

1.2.1. Implementation Constraints

- Facera will be targeted towards Android and iOS devices.
- JavaScript programming language will primarily be used during development [1].
- React Native development platform will be utilized for developing the mobile application [2].
- ARCore SDK will be used for modeling the environments and placing the 3D objects in the environment [3].
- Sceneform SDK library will be used along with ARCore for the rapid creation and integration of AR experiences in our app [4].
- Firebase will be used for user authentication, for signup, login, and user data security [5]. Firebase will also be used as a real-time cloud database system.
- React-video call API will be used to support live video calling [6].
- HeartBeat AI API will be used along with three.js Library for facial recognition and mapping onto 3D objects [7], [8].
- Github will be used for code sharing among group members and version control [9].
- Flask will be used for backend development [10].

1.2.2. Feasibility Constraints

- We require SDK support for implementing stable AR 3D models in the environment.
- We require video call/messaging SDK support at affordable rates.
- Pose and Face detection API with reasonable accuracy will be required.

1.2.3. Economical Constraints

- Firebase is free to use for low-tier requirements however, additional fees will apply if the usage capacity to be increased [11]
- React Native is free to use.
- ARCore is a software development kit developed by Google for developing AR-based applications and is free to use.

- Sceneform SDK allows you to import 3D models and then render 3D scenes for ARCore apps and is free to use.
- We intend to use Fritz.AI which is free to use.
- Three.js is also free to use.
- We have GitHub student accounts and are therefore able to use it for free.
- We require no additional purchases for development and testing tools as the hardware can be simulated.

1.2.4. TimeLine Constraints

- We will be using the Agile scrum methodology for the development of our application [12].
- The analysis phase is expected to end by November 2020, we expect to learn and be able to implement the required technologies by then.
- The High-Level Design phase is expected to end by the end of December 2020, we expect a working prototype supporting the basic functionalities of our application such as video call feature and a working user interface with progress being made into placing 3D objects using AR.

1.2.6. Hardware Constraints

- Devices with a decent camera will be required on both ends.
- Devices with support for AR is required.

1.3. Professional and Ethical Issues

There are several professional and ethical concerns that we will face throughout the project:

- To determine the facial expression, the application will request permission in order to use the camera. However, the application will not store redundant user data.
- The data will also not be shared with any third-party companies.
- The application will not track users' moves and location for unethical purposes
- The data and information will be secured and will be private.

- Another important issue about the application is its accuracy. Facial recognition systems can have near-perfect accuracy; thus, we will try to predict the face as accurately as possible.
- The user's password will only be known to the user themselves; no third party will have access to their passwords.
- Children friendly avatars will be used (Non-offensive avatars).

1.4. Competitors and similar technologies

Although there are no direct competitors to our project, there are some similar technologies on the market.

WhatsApp video calls:

- Very popular nowadays for personal use.
- Does not provide any entertainment like AR masks and filters.
- Implemented only for mobile phone users.
- Can be used for group calls.

Skype:

- Very popular for business communications.
- Provide a lot of technical features like live translation and screen sharing.
- No implementation of an AR.
- Mainly used on desktop computers for business purposes.
- Can be used for big conferences, including lots of people.

Facebook Messenger Video calls:

- Not popular for personal use.
- Can be used for group calls.

- AR masks are implemented.
- Can be used only on mobile phones.

Google Duo:

- Mobile application.
- AR masks are implemented.
- Maximum people in a conversation: 2.
- Not popular.

2. Requirements

2.1. Functional Requirements

The application system consists of two main components, the mobile applications and the cloud server for backend computations. In mobile devices, we will utilize various SDKs ARCore for placing the 3D objects in AR and React-video call API for implementing live video calls between the devices. For database and authentication, we will use Firebase and for other backend processing such as facial expression recognition, we will use the google cloud platform [13].

2.1.1. Mobile Application requirements

- The application should ask for the user's permission before using the camera
- The user should be able to select from an array of 3D avatars to be displayed on the camera
- The user should be able to resize and move the 3D avatar by dragging it across the screen
- The avatar mimics the user's facial expressions
- The 3D avatar should be able to perform predefined 3D animations by clicking various animation icons
- Send the live video to the backend servers
- Allow for the user to login and signup for the application

- Allow the user to add friends to call/message using the application
- Also, allow for chat messaging in addition to video messaging

2.1.2. Firebase (Backend) requirements

- The Firebase should be able to authenticate the users
- Allow the users to update their account information
- Safely store all user data
- Provide fast and safe access to user data without compromising the data
- Allow the user to delete their accounts

2.1.3. Google cloud platform (backend) requirements

- Securely and quickly process the video for facial expression recognition
- The output should be accurate enough to be mimicked by the 3D avatar

2.2. Non-Functional Requirements

2.2.1. Availability

The application should be available 24 hours and for a large number of users.

2.2.2. Reliability

The system should be reliable, i.e. the system should not crash during use.

2.2.3. Security

The interaction between users should be secure.

2.2.4. Scalability

Allow for at least 500 users to be able to use the application during the initial stages.

2.2.5. Extensibility

The application should be developed in a manner that allows for new features to be easily integrated into the system.

2.2.6. Portability

The system should be usable across different platforms e.g. Windows, Android, iOS, and Linux.

2.2.7. Accuracy

The 3D model should be at least 80% accurate in its mimicking of the user's facial expression.

2.2.8. Maintainability

The code structure of the software should be designed to allow for easy debugging and components should not heavily rely on each other in order to decrease the chances of unwanted side effects after debugging.

2.2.9. User-Friendly environment

The UI of the application will be designed to make it easy to use and understand allowing for a short learning curve. Moreover, the 3D object will be easy to place in the environment and the user will not need to have any prior knowledge in order to use the application easily.

The video call will have a limited number of buttons e.g. an end call button and a button to place the 3D avatar. This will allow the user to be able to properly enjoy the video call without making the screen densely populated, furthermore, the 3D avatar will also be of a feasible size to avoid taking much space on the screen.

3. References

[1] "Free JavaScript training, resources and examples for the community," *JavaScript.com*. [Online]. Available: https://www.javascript.com/. [Accessed: 12-Oct-2020].

[2] "React Native · A framework for building native apps using React," *React Native*. [Online]. Available: https://reactnative.dev/. [Accessed: 12-Oct-2020].

[3] "ARCore," *Google AR & VR | ARCore*. [Online]. Available: https://arvr.google.com/arcore/. [Accessed: 12-Oct-2020].

[4] "SDK Downloads | Sceneform (1.15.0) | Google Developers," *Google*. [Online]. Available: https://developers.google.com/sceneform/develop/downloads. [Accessed: 12-Oct-2020].

[5] *Google*. [Online]. Available: https://firebase.google.com/. [Accessed: 12-Oct-2020].

[6] nguymin4, "nguymin4/react-videocall," *GitHub*. [Online]. Available: https://github.com/nguymin4/react-videocall. [Accessed: 12-Oct-2020].

[7] F. Vázquez, "3D Face Reconstruction with Position Map Regression Networks," *Medium*, 05-Feb-2020. [Online]. Available: https://heartbeat.fritz.ai/3d-face-reconstruction-with-position-map-regression-networks-36f0ac2d3ef1. [Accessed: 12-Oct-2020].

[8] "three.jsr121," *three.js – JavaScript 3D library*. [Online]. Available: https://threejs.org/. [Accessed: 12-Oct-2020].

[9] "Where the world builds software," *GitHub*. [Online]. Available: https://github.com/. [Accessed: 12-Oct-2020].

[10] "Welcome to Flask¶," *Welcome to Flask - Flask Documentation (1.1.x)*. [Online]. Available: https://flask.palletsprojects.com/en/1.1.x/. [Accessed: 12-Oct-2020].

[11] “Firebase Pricing,” *Google*. [Online]. Available: https://firebase.google.com/pricing/?gclid=CjwKCAjw_Y_8BRBiEiwA5MCBJoiWev0gSHUnKDm-sf3AziBAYVMSAqAGUjypm4MsxTvjq2v0qfgUIBoCJ0UQAvD_BwE. [Accessed: 12-Oct-2020].

[12] “What is Scrum?,” *Scrum.org*. [Online]. Available: https://www.scrum.org/resources/what-is-scrum. [Accessed: 12-Oct-2020].

[13] “Google Cloud,” *Google*. [Online]. Available: https://cloud.google.com/. [Accessed: 12-Oct-2020].