**UC-238 DogEm**

**Undergraduate Capstone**

**Fall 2022**

**Authors:** Chloe Chung, Melissa Iniestra, Deo Intshakal A Nzeng, Zane Atkinson, Joshua Willson

**Advisor:** Professor Sharon Perry

**Project Website:** <https://dogemapp.github.io/>

**Project Source Code Directory:** <https://dogemapp.github.io/sourcecodedir.html>

**Mobile App Site QR Code:**



**Abstract**

The DogEm project's overarching objective was to produce a functional cross-platform mobile app that floods a list of contacts' inboxes by repeatedly calling, texting, and/or emailing. To accomplish this goal, we have compiled preliminary research, produced a series of prototypes, and completed the development process.

Our preliminary research consists of research pertaining to our tech stack, our user base, our app's requirements, possibilities for messaging and calling features, UX/UI research, reading through React Native documentation, and constraints on the messaging and calling features on iOS vs. Android.

Each member produced a DogEm app prototype to familiarize ourselves with the technology that will be employed throughout the project: React Native and Expo CLI. Features implemented throughout the prototypes include, React Native Navigation, expo-sms, expo Linking, Firebase, and various other React Native libraries and APls.

The result of our development process was a cross-platform mobile app capable of reliably sending thousands of calls, texts, and emails to a contacts list. Our development process consisted of a team effort in which members were assigned an important functionality of the app. Members then actively contributed their changes to a GitHub repository holding our final product. GitHub has been an important tool for our development efforts because it allows for code review and effective version control. With regard to version control, we tracked and maintained changes made to code on our GitHub. It served as an integral part of our software development process. In order to facilitate best practices in our source code management, the DogEm team has a GitHub containing all of our source code and our site has a source code directory.

**Table of Contents**

About The Project......................................................................................................p.1 – p.5

1.1. What is DogEm?........................................................................................p.1-3

1.1.1. App Description……………………………..….…………...……..p.1

1.1.2. Criteria for DogEm’s Success……...………...…………..………p.1

1.2. How DogEm Works – High Level Overview..............................................p.1-3

1.2.1. High-Level Architecture Overview…………………..……..…...p.1-2

1.2.2. App Features………………………………………..………..…..p.2-3

1.3. A Three Phase Project Process: Planning Phase, Initial Phase, and

Development Phase…………………………………………..………...……..p.3-5

1.3.1. Activity Breakdown by Phase………………..………...……….p.3-5

1.3.2. Gantt Chart……………………………………….………………p.5

Research……………………………………………………………………….………….p.6 – p.15

2.1. Research’s Role Throughout Our Project……………….………………………..p.6

2.2. Planning Phase Research………………………………….………………………p.6-7

2.2.1. Mobile Application Technology Stacks…………….……………………….p.7

2.2.2. The Mobile Market: Understanding Our User Base...………..…………...p.7

2.2.3. Criteria for Mobile App Success…………………………..……..………….p.7-8

2.3. Initial Phase Research…………………………………………..…………..……..p.8-12

2.3.1. Principles of Quality UX/UI………………..…………………..……………..p.8

2.3.2. Calling and Messaging Constraints on iOS vs. Android…..……….……..p.8-9

2.3.3. Calling Function Research……………………………………..……..……..p.10

2.3.4. Messaging Function Research……………………..……………….………p.10-12

2.4. Development Phase Research…………………………..……………….………p.13-15

2.4.1. Discerning between Development Phase 1 and Development

Phase 2….……………………………………………………..……………...p.13

2.4.1.1 Phase 1: Why Make a Minimum Viable Product?.................p.13

2.4.1.2 Phase 2: Monetization Avenues for DogEm……..……....…..p.13

2.4.2. Deployment Research…...………………………………....………..…p.13-14

2.4.2.1. Deployment Options…………...……………..………..….p.13

2.4.2.2. Why Expo Go…………………………………….……..…p.14

2.4.3. Selecting a Database……………………………………...…………...p.14-15

2.4.3.1 Why Firebase…………………………...…………..………....p.14

2.4.3.2 Firebase Authentication……………………….……………...p.14-15

2.4.3.3 Types of Databases: Relational vs. Hierarchical…….....….p.15

2.4.3.4 Firebase Cloud Firestore...........................................….….p.15

3. Requirements………….……………………………………………………………p.15 – p.18

3.1. Planning and Initial Phase Requirements……………….………………p.15-16

3.2. Development Phase Requirements…………………….......……………p.16-18

3.3.1. Development Phase 1 Requirements……………….……….p.16-17

3.3.2. Development Phase 2 Requirements……………….……….p.17-18

4. Test Plans…..………….……………………………………………………………p.18 – p.21

4.1. Initial Phase Test Plan………………………………….………………….p.18-19

4.2. Development Phase Test Plan……………………………………………p.19-21

4.2.1. Development Phase 1 Test Plan…………………….……….p.19-20

4.2.2. Development Phase 2 Test Plan…………………….……….p.20-21

5. Mockups………….……….…………………………………………………………p.21 – p.23

5.1. Initial Phase Mockup…………………………….…………...…………….p.21-22

5.2. Development Phase Mockup……………………………...………………p.22-23

5.2.1. Development Phase 1 Mockup……………………….……….p.22-23

5.2.2. Development Phase 2 Mockup……………………….……….p.23

6. Prototypes……………………………………………………………………………p.23 – p.29

6.1. Initial Phase: Member Prototypes………..…….…………...…………….p.23-27

6.1.1. Deo Intshakal A Nzeng’s Prototype………………….……….p.23-25

6.1.2. Melissa Iniestra’s Prototype..………………………………….p.25-26

6.1.3. Chloe Chung’s Prototype………..…….…………...………….p.26-27

6.1.4. Zane Atkinson’s Prototype………..……………...…..……….p.27

6.2. Development Phase: The Final Prototype………………..……..………p.28-29

7. Version Control………………………………………………………………………p.29-31

7.1. How the DogEm Team Uses Version Control………………………….p.29-30

7.2. The Source Code Directory………………………………………………p.30-31

8. Results…………………………………………………………………………….….p.31-36

8.1. Discussion and Demo of the Final Product…………………………….p.31-32

8.2. DogEm Calling vs. Conventional Mobile Calling………………………p.33

8.3. DogEm Messaging vs. Conventional Mobile Messaging……………..p.33-34

8.4. The DogEm Website……………………………………………………..p.34-36

9. Conclusion ……………………………………………………………………….….p.36-37

Appendices………………………………………………………………………..……p.38 – 45

Appendix 1 – Weekly Logs...............................................................................…p.38-44

A.1.1. Week 1: 8/16/22 - 8/22/22.........................................................................p.38

A.1.2. Week 2: 8/23/22 - 8/29/22.........................................................................p.38

A.1.3. Week 3: 8/30/22 - 9/05/22.........................................................................p.38-39

A.1.4. Week 4: 9/06/22 - 9/12/22.........................................................................p.39

A.1.5. Week 5: 9/13/22 - 9/19/22.........................................................................p.39

A.1.6. Week 6: 9/20/22 - 9/26/22.........................................................................p.39-40

A.1.7. Week 7: 9/27/22 - 10/03/22.......................................................................p.40

A.1.8. Week 8: 10/04/22 - 10/10/22.....................................................................p.40-41

A.1.9. Week 9: 10/11/22 - 10/17/22.....................................................................p.41

A.1.10. Week 10: 10/18/22 - 10/24/22.................................................................p.41

A.1.11. Week 11: 10/25/22 - 10/31/22.................................................................p.42

A.1.12. Week 12: 11/01/22 - 11/07/22.................................................................p.42-43

A.1.13. Week 13: 11/08/22 - 11/14/22.................................................................p.43

A.1.14. Week 14: 11/15/22 - 11/21/22.................................................................p.43-44

A.1.15. Week 15: 11/22/22 - 11/28/22.................................................................p.44

A.1.16. Week 16: 11/29/22 - 12/01/22.................................................................p.44

Appendix 2 – Suggestions for the Future.........................................................…p.45

References………………………………………………………….………………….p.46

**1. About the Project**

**1.1. What is DogEm?**

**1.1.1. App Description**

**What is DogEm:**

UC-238 DogEm is an undergraduate capstone project developed for CS 4850. Our team spent a semester building a mobile application capable of repeatedly calling or texting and emailing a user-entered contacts list.

**What is DogEm’s Purpose:**

The purpose of DogEm is to send a massive volume of calls or text messages and emails to a series of contacts. The reason for this is to get the attention of said contacts and re-establish communication between DogEm users and their contacts list.

**What DogEm Can Be Used For:**

- **Emergencies**: In the event of an emergency, DogEm can be used to rapidly send contacts hundreds—or even thousands of calls or texts and emails to get their attention.

- **Pestering a List of Contacts**: Users may feel neglected by their contacts, so flooding their inboxes has the potential to re-establish communication.

1.1.2. Criteria for DogEm’s Success

Criteria for success are non-negotiable characteristics the DogEm app has to possess. The criteria for success were devised prior to development in order to set a standard for what we were to accomplish. The criteria as well as the rationale behind why we designated it as highly important are as follows:

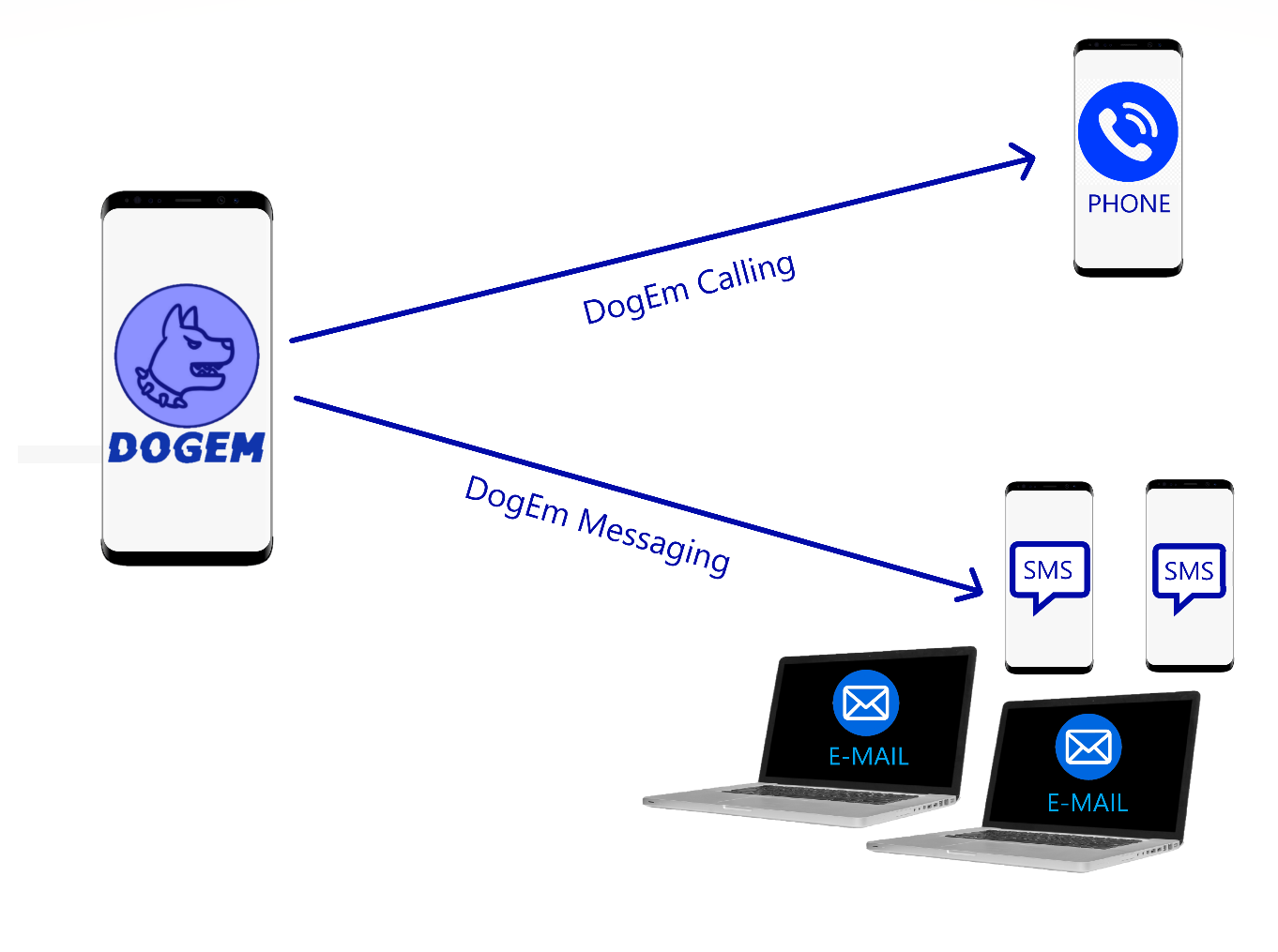
**DogEm must be cross-platform**. This is to ensure that we can access the largest user base possible.

**DogEm must prioritize users’ privacy**. Although our team developed a messaging and calling app, we will not access or save the user’s messages or contacts.

**DogEm must have a generous free-tier**. The team wanted to develop an app that all users can derive a great deal of utility from. As such, our monetization avenue is an ad-free experience rather than one that imposes messaging or calling limitations on our users.

**1.2. How DogEm Works – High Level Overview**

**1.2.1. High-Level Architecture Overview**



**1.2.2. App Features**

- **Easy Onboarding**: DogEm’s registration page makes it easy for users to join the app. A simple registration process is conducive to gaining new users.

- **User Authentication**: Only registered users can use DogEm. This is a security measure to prevent anonymous malicious users from abusing the app.

- **Welcome page**: The welcome page has a welcome message including the user’s email address to show that they are logged into their account. The page also has a video explaining how to use the app.

- **DogEm page – Control variable**: The control variable allows the user to define a number of calls or texts/messages they wish to send.

- **DogEm page – Editable Contacts List**: The user has the ability to both add and delete contacts from their contacts list.

- **DogEm page – Clear Contacts Button**: The user can completely delete their list of contacts if they wish to start another session with a different contacts list, or they wish to not call/message any of the contacts they previously listed.

- **DogEm page – Calling**: The calling feature calls a user’s contact for the number of times they specified in the control variable and iterates every 10 seconds.

- **DogEm page – Messaging**: The messaging functions sends SMS or Emails depending on the entered contact. It will iterate for the number of times that the user entered for the control variable. It automatically fills the contacts list and SMS/Email body of the user’s default messaging app for every iteration.

- **DogEm page – Stop Messaging Button**: This sets the control variable to 0 and will stop both calling and messaging.

**1.**3. A **Three Phase Project Process: Planning Phase, Initial Phase, and Development Phase**

**1.3.1. Activity Breakdown by Phase**

**Rationale Behind Choosing to Divide the Project into Phases**

During the very first meeting, the DogEm team decided that dividing the project into three well-defined phases would give our efforts structure and allow us to better define our objectives and goals with respect to our required timeline. We sub-divided phases into objectives and goals. Objectives are modularized actions that serve to facilitate bigger-picture outcomes, or goals.

Defining the Phases

Each phase had a set of rigid objectives and goals the team was to complete within specific timelines. They are as follows:

**Planning Phase:** The planning phase was the very first phase of the project. It was primarily centered around requirements gathering and setting the foundation for the project.

**Planning Phase Objectives:**

PO1) Establish communication and collaboration avenues including a GroupMe, a Microsoft Teams, and a GitHub (corresponds with PG1).

PO2) Work on planning phase research pertaining to mobile app technology stacks, the mobile market, and criteria for mobile app success (corresponds with PG1).

PO3) Create a preliminary requirements document based on planning phase research (corresponds with PG2).

PO4) Derive an initial test plan from the requirements document (corresponds with PG2).

PO5) Every team member was instructed to set up their development environment (corresponds with PG3).

**Planning Phase Goals:**

PG1) Gain a high level of understanding of the project.

PG2) Have a preliminary roadmap for the various steps required for success.

PG3) Move towards competency with React Native.

**Initial Phase:** The initial phase was the second phase of the project. It is an intermediary between planning and development because this phase was heavily focused on refining requirements from planning and creating prototypes to move towards development.

**Initial Phase Objectives:**

IO1) Conduct initial phase research pertaining to what constitutes quality UX/UI, iOS vs. Android comparison, calling research, and messaging research (corresponds with IG1).

IO2) Refine the requirements document from planning (corresponds with IG2).

IO3) Each member must produce a prototype derived from the most recent requirements document that include important app requirements such as React Navigation, styling, calling, messaging, etc (corresponds with IG2).

IO3) Select our favorite prototype UI (corresponds with IG3).

IO4) Build on our favorite prototype to produce our final prototype (corresponds with IG3).

**Initial Phase Goals:**

IG1) Gain an understanding of how to ensure the calling and messaging functionalities are truly cross-platform.

IG2) Each member must demonstratecompetency with React Native, Expo CLI, APIs, and libraries

IG3) Move towards developing our first version of the app.

**Development Phase:** The development phase is the final phase of the project. It is broken down into two subphases: Development Phase 1, in which we produced our minimum viable product (MVP) and Development Phase 2, in which we produced a version demonstrating potential for app monetization.This phasebuilds onthe initial phase by finalizing requirements with light to limitations identified in initial phase research, and by adding onto the final prototype to produce versions 1 and 2 of the app.

**Development Phase Objectives:**

DO1) Conduct development phase research pertaining to the MVP, monetization, databases, and deployment (corresponds with DG1).

DO2) Refine the requirements document from the initial phase with regard to our app’s constraints (corresponds with DG2).

DO3) Ensure our app is functional and meets our minimum requirements (corresponds with DG3).

DO4) Build on our MVP as a team and incorporate monetization (corresponds with DG4).

**Development Phase Goals:**

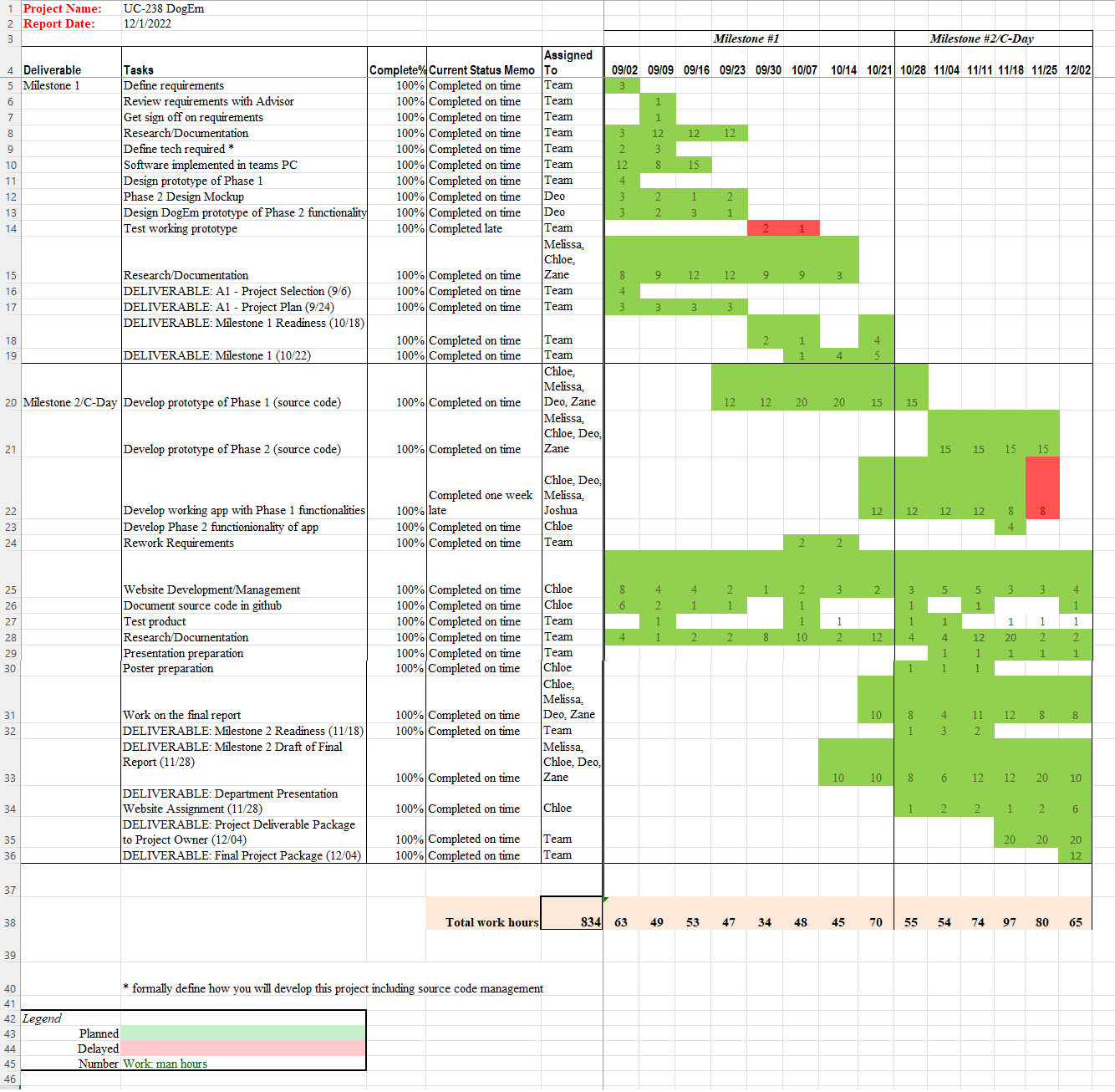
DG1) Further understand what is necessary to implement a full-stack cross-platform mobile app.

DG2) Implement our core functionality in spite of limitations.

DG3) Develop the MVP.

DG4) Demonstrate that DogEm possesses great potential for monetization.

**1.3.2. Gantt Chart**



**2. Research**

**2.1. Research’s Role Throughout Our Project**

Research was a significant part of our project and the team emphasized research at every phase. By conducting research at every phase, the team solidified our knowledge of React Native, Expo, APIs, databases, and other technical areas. To better understand how to implement the core functionalities of our app, we watched tutorial videos, read articles containing examples for implementing various functionalities, and also referred to resources for setting up our development environments. Research was not confined to technical areas. We also looked into our user base, the mobile market, and best practices, which heavily influenced various design and architecture decisions.

**2.2. Planning Phase Research**

Our planning phase research included mobile app technology stacks, the mobile market, and criteria for mobile app success.

**2.2.1. Mobile Application Technology Stacks**

**DogEm’s Tech Stack:**

For developing our app, we used react-native, Babel JavaScript, and Expo CLI to target both iOS and Android. Our development environment consisted of the Node.js command prompt, VS Code, Git and GitHub for version control. To store our data, we used Firebase's Firestore which is their database for mobile development. For authentication, we used Firebase Authentication.

**Comparison of Cross-Platform Mobile Application Tech Stacks That Our Team Researched:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tech Stack** | **Pros** | **Cons** | **Why the DogEm did/did not select this tech stack** |
| **Flutter**  Flutter is a mobile UI framework. | - Strong support.  -Large developer community. | - Flutter is powered by Dart, which is a somewhat obscure programming language. | -The DogEm team chose not to use Flutter because we did not want to work with Dart. |
| **Xamarin**  Xamarin is a platform for building cross-platform mobile apps. | -Xamarin applications are written in C#. | -Limited support. While doing research for tech stacks, the team saw that Xamarin.Forms (used for building cross-platform mobile apps) was going to be replaced with .NET MAUI. | -Given its uncertain future, we chose not to use Xamarin. |
| **Ionic**  Ionic is a SDK for cross-platform mobile development. | -Ionic is based on JavaScript. | -Unstable native plugins.  -Ionic is not as widely used relative to some other options, so support is less readily available. | -We chose not to Ionic because it is more obscure than other options available. |
| **React Native**  React Native is a UI software framework. | -Large developer community.  -React Native uses JavaScript.  -When used with expo, it is easy to test on a device.  -We looked at expo libraries and wanted to apply some to our code. | -React native is slower than writing native code.  - Updating React native can (and for us, has been) a hassle. | -Being able to use JavaScript and expo were two major factors behind why we chose React Native. |

**2.2.2. The Mobile Market: Understanding Our User Base**

Our research pertaining to the mobile market helped shape one of our criteria for success: the app must be cross-platform. In short: Apple holds a greater share of the mobile market in the US and Europe, but Android has a considerably greater global influence. In order to effectively reach as many users as possible, we decided it was imperative that DogEm be cross-platform. The decision to have our app be cross-platform, shaped much of our development efforts.

**2.2.3. Criteria for Mobile App Success**

We divided our criteria for mobile app success into business and user objectives to understand what characteristics we needed to prioritize

**Business Objectives:**

App Growth: Attract more users to DogEm by using our webpage, having a clear and concise landing page, and encouraging user reviews on the app store.

Engagement and User Retention: Promote engagement/retention with the app by ensuring app quality, easy and efficient onboarding, and add engaging features such as analytics and notifications in addition to the base functionality.

Monetization: Offer a paid tier with additional services and have ads on the base tier.

**User Objectives:**

High quality UX: The user must have a positive experience with the app. The primary ways to achieve this are with an easy to use and attractive UI (see below) and through app reliability/quality. The app must have effective and fast mechanisms to execute its core functionality (repeated messaging until a response from the contact is received).

Easy to use and attractive UI: The app must have a robust and attractive UI in order to ensure that users derive the greatest value and most positive experience from the app. Employing modular design principles to each page will assist in making an app that is not only user-friendly, but also visually appealing.

**2.3. Initial Phase Research**

In our initial phase, we began working towards development. This was an intermediary phase in which we built prototypes and started gaining momentum in our endeavor to produce DogEm.

**2.3.1. Principles of Quality UX/UI**

Since this phase was concerned with beginning development, the team decided to research criteria for quality UX/UI in order to make sound development decisions.

Some important principles we chose to adhere to are:

UX: The app must not only be functional, it must be fast and reliable. All messages and calls sent to contacts must quickly and reliably reach those contacts.

UX: The app must be easy to interact with.

UI: The app must be attractive and showcase an intuitive, modular UI.

UI: The app must not be a hassle to join: easy onboarding is critical for gaining new users.

UX: The app must not inundate users with obnoxious ads, popups, sounds, etc. It must be a positive and fun experience for users.

**2.3.2. Calling and Messaging Constraints on iOS vs Android**

**Messaging Constraints on iOS vs. Android**

|  |  |  |
| --- | --- | --- |
| **Constraint** | **iOS** | **Android** |
| Unable to programmatically send SMS/Email/MMS messages on this mobile OS. | Yes, it is a constraint for iOS.  In order to programmatically send messages, you must use a third party API such as Twilio. | No, it is not a constraint for Android.  Android is far less restrictive in this regard and allows for developers to programmatically send messages through various avenues including react-native-sms and Android’s SmsManager API. |
| Requirement for user intervention when sending SMS/Email/MMS messages.  \*An extension of above that proved to be a notable constraint for DogEm | Yes, this extends from the inability to programmatically send messages. User intervention is required whenever sending messages through the mobile user’s default messaging app. | This depends on what API is used. Expo-sms requires user intervention, but it was only chosen for this project because it is compatible with iOS and Android and was the fastest in our testing. Android, in general, allows for programmatically sending messages so user intervention can be extremely limited relative to iOS. |

**Calling Constraints on iOS vs. Android**

|  |  |  |
| --- | --- | --- |
| **Constraint** | **iOS** | **Android** |
| Unable to programmatically make phone calls on this mobile OS. | Yes, it is a constraint for iOS.  In order to programmatically make phone calls, you must use a third party API such as Twilio. | No, it is not a constraint for Android.  Android is far less restrictive in this regard and allows for developers to programmatically make phone calls through various APIs including react-native-immediate-phone-call |
| Requirement for user intervention making a phone call. | Yes, this extends from the inability to programmatically make phone calls. User intervention is required whenever making a phone call through the mobile user’s default messaging app. | This depends on what API is used. Expo-linking requires user intervention, but it was only chosen for this project because it is compatible with iOS and Android and was the fastest in our testing. Android, in general, allows for programmatically making phone calls so user intervention can be extremely limited relative to iOS. In addition, Android allows to modify the calling permission in order to generate automated phone calls. |

**2.3.3. Calling Function Research**

**What is the DogEm Calling Function?**

The DogEm calling function is an important functionality of the Dogem app which allows the user to make repeated phone calls.

**How does it work?**

The user enters a phone number or selects one from his contact list. Upon selecting the “Phone Call” button, DogEm opens their device’s default calling app and automatically displays the dialer with the selected phone number. Then, the user is required to click on the call button (which is usually green on both iOS and Android). Once they hit the “call” button in the dialer menu DogEm opened, the contacts will be called via their phone numbers for a specified interval (currently we allow 2 calls every hour). After making the initial phone call, DogEm will re-open their default calling app again and automatically fill the contact field with the phone number previously entered. More importantly, after entering the contact information then selecting the “Phone Call” button, all the users must do for repeated phone calls is repeatedly hit the call button in the dialer. DogEm will keep track of the phone call status and monitor when the phone call has been declined or answered. Therefore, the repeated messaging function will stop immediately once the ‘dogged’ contact answers to the phone call.

**Semi-Automated vs. Fully Automated:**

The biggest difficulty we faced while implementing the phone call function was related to the inability to make automated phone calls without requiring the user input. To begin with, we were able to implement this functionality using the React Native CLI with the native module called ‘react-native-immediate-phone-call’. This API allows to make automated phone calls without opening the phone’s dialer menu. However, this functionality only works on the Android platform, since iOS restricts the implementation of fully automated phone call. Since our most important requirement is having a fully functioning and cross-platform app, we decided to compensate with a semi-automated approach. The automated component is the app’s ability to repeatedly fill the contact information in the dialer menu before making the phone call, which prevents the user from selecting the desired contact multiple times. The required user interaction is hitting the call button for each phone call in the dialer menu. By choosing this approach, we save the user 1 keystroke per phone call which reduces the time required to send repeated phone calls.

**2.3.4. Messaging Function Research**

**What is the DogEm Messaging Function?**

The DogEm messaging function is a major functionality of the DogEm app which facilitates repeated messaging of a user’s contacts via sms and email.

**How does it work?**

The user enters a series of contacts’ email addresses and phone numbers as well as a message they wish to send their contacts. Upon selecting the “Send Message” button, DogEm opens their device’s default messaging app (iMessage for iOS and Messages for Android) and automatically fills the message body with their selected message and the contact information with the contact emails and phone numbers they entered. Once they hit the “send” button on the new message DogEm opened, the contacts will be messaged via their email addresses (emails) and phone numbers (sms) for a specified interval. After sending the initial message, DogEm will re-open their default messaging app again and automatically fill the contact field and sms body field with the contact information and message they entered. It is important to note that after entering their contacts’ information and message then selecting the “Send Message” button, all the user has to do for repeated messaging is repeatedly hit the send button.

**Semi-Automated vs. Fully Automated:**

A major obstacle we encountered was the inability to programmatically send sms messages via iMessage. To compensate for this, we chose to take a semi-automated approach. The automated component is the app’s ability to repeatedly fill the message body of the sms message, which eliminates the requirement for the user to copy/paste their message for each send operation. The required user interaction is hitting the send button for each message. By choosing this approach, we save the user 2 keystrokes per message which halves the time required to send batches of messages (further discussion of this can be found under “Results”).

**Rationale for Choosing the current Implementation of the DogEm Messenger Function: DogEm Messaging vs. Twilio vs. Other Third Parties**

**DogEm Messaging:**

We wanted to ride on the user’s system rather than using a mail server. This decision means that we can capitalize on the fact that the overwhelming majority of phone carriers utilize an sms to email gateway, meaning we can send both sms and email messages to multiple contacts reliably and simultaneously.

By riding on the user’s system, we can ensure that users are limited by their own system rather than ours.

By using expo-sms we ensure that the messaging function is completely free for free-tier users.

Other mailing alternatives:

|  |  |  |  |
| --- | --- | --- | --- |
| **Service Name** | **What it is** | **Pros** | **Cons** |
| **EmailJS** | -JS library that sends emails using client-side technologies. | -Easier to use than some of the other options. | -VERY limited free tier usage (200 messages per month).  -Can’t be used for calling. It would basically send Gmail emails (you can send sms via gmail though)  -Third-party dependency |
| **expo-sms** | -An npm package that provides access to the mobile phone’s messaging system (messages for android, iMessage for iOS). | -Easier to implement than some other options.  -Completely free, no usage limitations. | -Requires user intervention for messaging (makes them select the “send” button)  -Only for SMS/email  -Not compatible with web. Mobile-only. |
| **expo-linking** | -Code library from Expo’s SDK. It allows the app to interact with other apps via deep links. | -Easier to implement than some other options.  -Completely free, no usage limitations  -Useful URL schemes including: tel (calling), mailto (emailing), and sms (text)  -Cross-platform (web, mobile) | -Requires user intervention for messaging (makes them select the “send” button) |
| **Nodemailer** | -It is a module for Node.js applications to send messages. | -It is free. | -Requires a lot more heavy-lifting from the backend. |
| **Twilio** | -Twilio is a communications API. | -Text, call, and email  - Easy to implement  - Developer support | - Third party dependency  - Can be costly: 0.0079 per sent text and 0.0075 per received text.  SendGrid is cheaper, but given our volume of messaging it will add up.  Calling is 0.0013/min to send, 0.0085/min to receive. |

**2.4. Development Phase Research**

**2.4.1. Discerning between Development Phase 1 and Development Phase 2**

**2.4.1.1. Why Make a** Minimal Viable Product

A minimal viable product is a part of Agile where you develop a product with efficient features to satisfy consumers. In the business world, MVP provides feedback from consumers which could help us develop a better product. It gives everyone an understanding of what can be fixed to make the customers happy and give a clearer path for the company for how to succeed.

The DogEm team chose to make a minimum viable product to ensure that we have a version of the app that fulfills all of our baseline criteria. The MVP is a valuable tool because it sets the foundation for further development efforts and ensures that the team has some form of useful product in the event of catastrophic failure with further versions of the app.

In the end, our MVP checked off all of our requirement boxes and it evolved into v1.0.0 of the app, which is the paid ad-free version. v2.0.0, which accounts for monetization via ads, is our free tier. The minimum viable product certainly provided value for our team because it gave us an efficient and functioning version of our app to both showcase and build off of.

**2.4.1.2. Monetization Avenues for DogEm**

The chosen route for monetization for the DogEm app was through the use of in-app ads. The specific way DogEm has ads displayed is through the use of a banner that will appear on each page of the application. This version of the app is considered DogEm version 2.0.

Other monetization avenues for DogEm were considered. At one point, the team briefly introduced a messaging limit for free-tier users, but it was decided that this decision was contrary to one of our criteria for success: the app must have a generous free-tier. It makes little sense to enforce a messaging limit, calling limit, or restrict the number of contacts the user can message when our app rides on their system. Having the free-tier be an experience with ads was the most optimal solution since it takes how the app works, as well as ensuring a quality user experience into account.

**2.4.2. Deployment Research**

**2.4.2.1. Deployment Options**

For deployment, we planned on making the app capable for iOS and Android using React-Native and Expo CLI along with Firebase. We used user-authentication and database capabilities through Firebase.

**React Native**

React native is a front-end framework that uses JavaScript to develop apps for both iOS and Android. It contains libraries and pre-built components which makes it easier for development.

**Expo CLI**

Expo CLI is a framework built on top of React Native. It provides developers with tools that simplify the creation of React Native apps.

**Node.js**

Node.js is an open-source platform built on Chrome’s JavaScript runtime engine. Essentially, it allows developers to execute a JavaScript program on the local machine and is used to help develop applications. Node.js contributed to the development of the mobile app by allowing us to run JavaScript in the terminal to add modules, or packages, for multiple dependencies of the app.

**Firebase**

Firebase is an app development platform which provides cross-platform SDKs to help developers build apps on Android, iOS, and web. The services provided by Firebase includes real-time database, cloud storage, authentication, crash reporting, machine learning, and more.

**2.4.2.2. Why Expo Go**

Expo Go makes it easier to use react-native. It can use its own servers to create builds. Expo Go is a mobile app that can be downloaded onto your personal device, either an Android or iOS device, that allows developers to run their project. This eliminates the need for any simulators. The project can be shared via the QR code presented at the start of the server. It also provides APIs for both iOS and Android.

**2.4.3. Selecting a Database**

**2.4.3.1. Why Firebase**

We chose Firebase because it provides real-time synchronization, offline persistence, and minimal steps. It is concurrent among clients in real time. Setting up Firebase Firestore and Authentication require minimal steps and helped us to meet our deadlines faster.

**2.4.3.2. Firebase Authentication**

Firebase Authentication has the advantage of having multiple built-in UI libraries to support various methods of authentication for users. It supports authentication using passwords, phone numbers, emails, and authentication from other providers like Google, Facebook and Twitter, and more. Our app implements authentication based on email and password. This form of authentication allows the app to have a limit of 3,000 daily active users (DAUs). Daily active usage is calculated based on how many unique users sign in during a 24-hour period.

Firebase Authentication is used throughout the app. In the registration page, there are many fields that a user is required to enter in order to sign up to our app. However, only the email and password will be stored in the Firebase authentication upon a successful registration of the user. The emails must be unique, meaning that a user cannot register another account with the same email. Once registered, the user will has a designated uid, or unique user identification number. Firebase also addresses the user’s password and provides the appropriate errors to prevent any issues. For instance, the password cannot be less than six characters long for minimal required security of any app. In the login page, the authentication requires only the registered email and password to allow the user access to the app. Upon logging in, a user can only be identified based on the authentication session from logging in which involves the email and uid. Therefore, the only information we have on the user to welcome them to the app is their email or uid. We used the email as the unique identifier rather than the uid.

**2.4.3.3. Hierarchical vs Relational Databases**

In a hierarchical database, the data gets stored in a tree-like structure. They can hold duplicate data unlike relational databases which contain no duplicate data. In relational databases, it is easier to obtain data because they can connect databases with relationships. It takes longer to get data from hierarchical databases. Hierarchical databases are a first-generation model while relational databases are a second-generation model. The tree structures in hierarchical databases contain parent and child nodes and relational databases have tables that consist of tables and rows. Relational databases use SQL which is a query language to retrieve data and they aren’t dependent on applications.

**2.4.3.4. Firebase Cloud Firestore**

Firebase has two different databases available, Cloud Firestore and Realtime Database. We went with the newest one, Firestore, which allows us to construct structured hierarchical data for the users of our app. Cloud Firestore has does not use SQL and instead is considered a document-oriented database. This is because data is stored in documents, which are organized into collections instead of having tables. Firestore also has offline persistence which means we can use it even if we have a bad internet connection. It even provides a free plan which is convenient for us to use along with 1GB of storage.

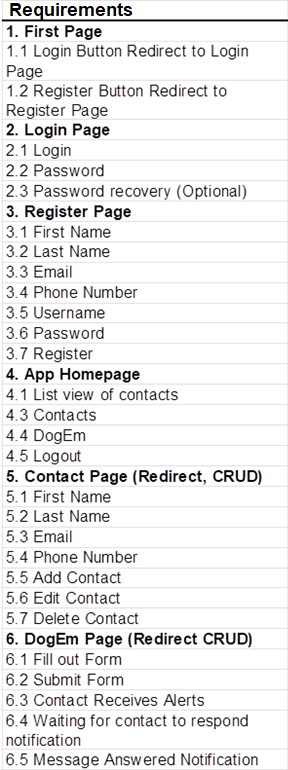
Firestore is only used to store a user's information during the registration period. We have been able to limit the creation of a user to occur only when a registration is successful, meaning there is a unique email entered, in order to avoid duplicate data in the database when a user clicks on the registration button. The database stores user's information under the collection we created, named “users”. Once a user successfully registers, a document is created with a unique identifier and this identifier represents the user created. Within the document are the fields that the user was required to enter in order to sign up to the app.

**3. Requirements**

Requirements are a page-by-page needs assessment document. For each phase, we assessed requirements to determine what components were necessary for our app to function in accordance with expectations.

**3.1. Planning and Initial Phase Requirements**

The requirements developed in the planning phase were used when we were developing our final prototype. They are below:

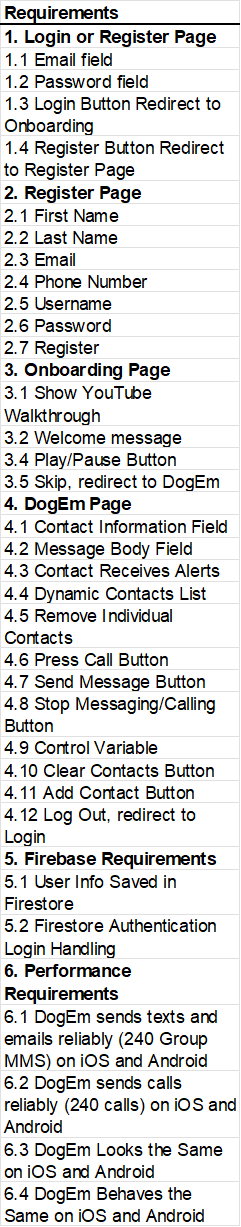


**3.2. Development Phase Requirements**

Versions 1 and 2 of the app were created during the development phase. Our requirements documents were instrumental guides during the development process.

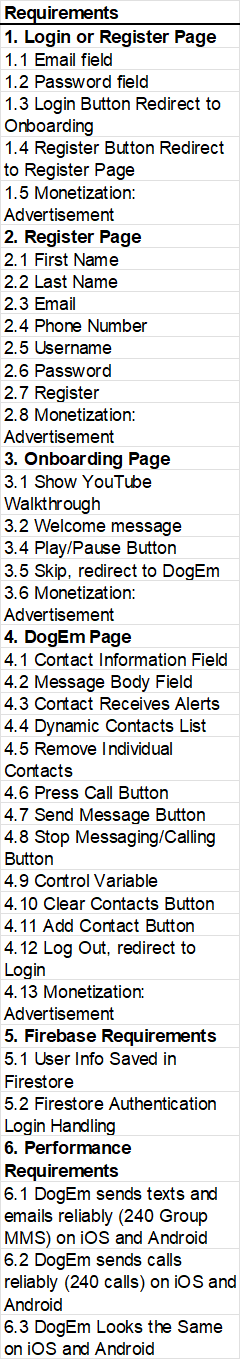
**3.2.1. Development Phase 1 Requirements**

The requirements developed for development phase 1 encapsulated all of the requirements necessary for our app to function as-needed. These requirements also account for performance. They are below:



**3.2.2. Development Phase 2 Requirements**

Development phase 2’s requirements are nearly the same as development phase 1. The difference is that it accounts for monetization through ads on top of each page. The requirements are below:

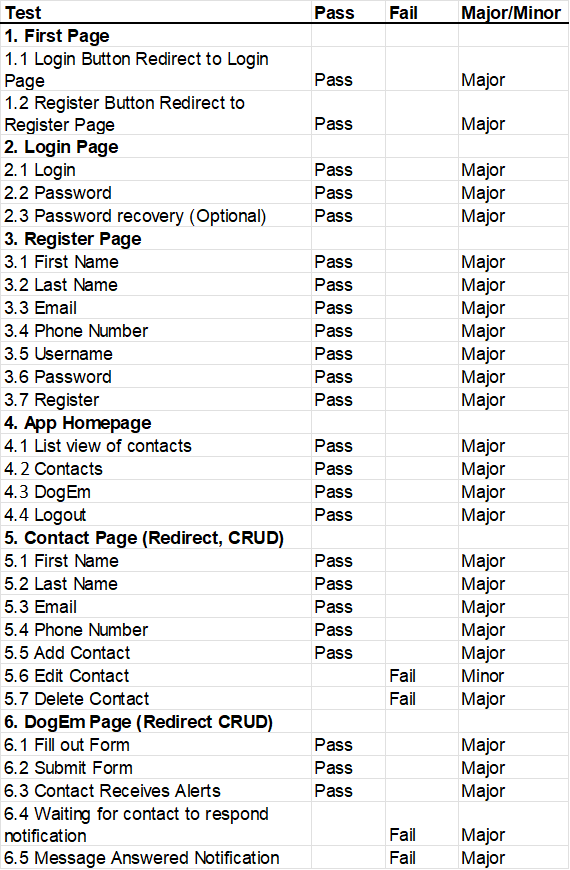


**4. Test Plans**

Test plans are derived from requirements documents. They are used to ensure that the app is functioning correctly. Our team used them to assess whether or not certain components or functionalities were working as-needed. We also paid special attention to tests listed as “Major”, since those are integral to the app.

**4.1. Initial Phase Test Plan**

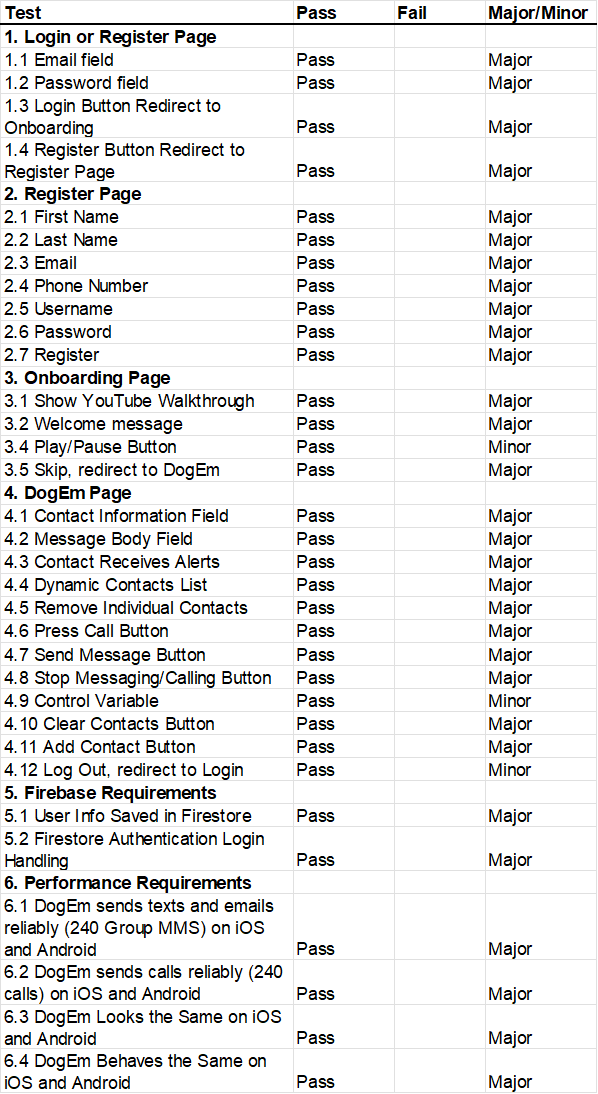
The initial phase test plan was used to test our final prototype.



**4.2. Development Phase Test Plans**

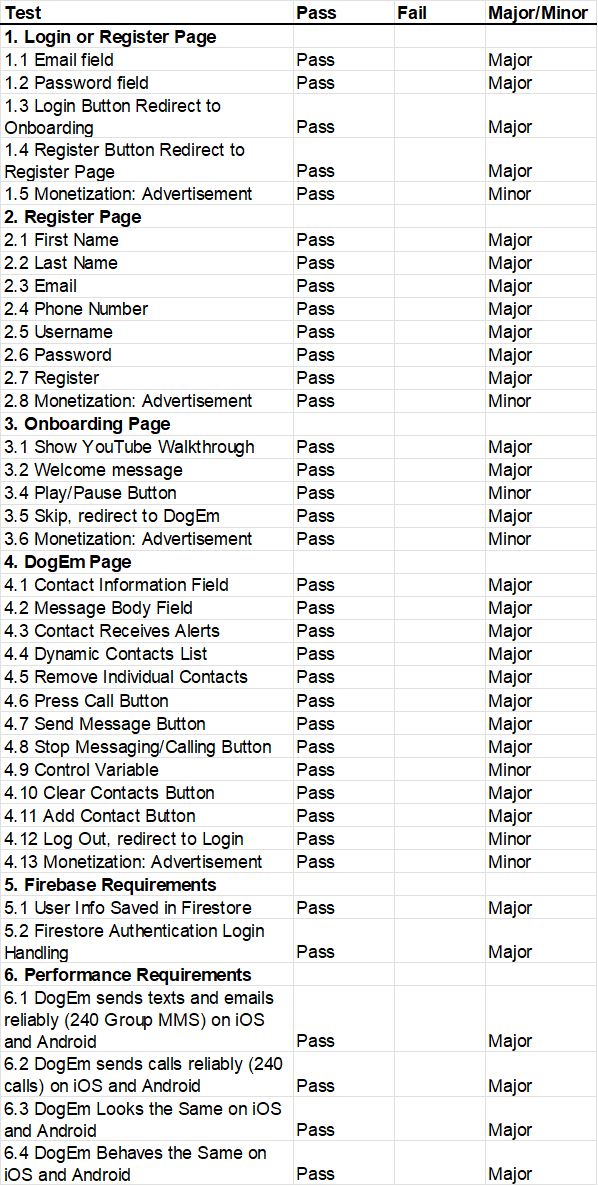
**4.2.1. Development Phase 1 Test Plan**

Our development phase 1 test plan assessed our version 1.



**4.2.2. Development Phase 2 Test Plan**

Our development phase 2 test plan was used to test our version 2.

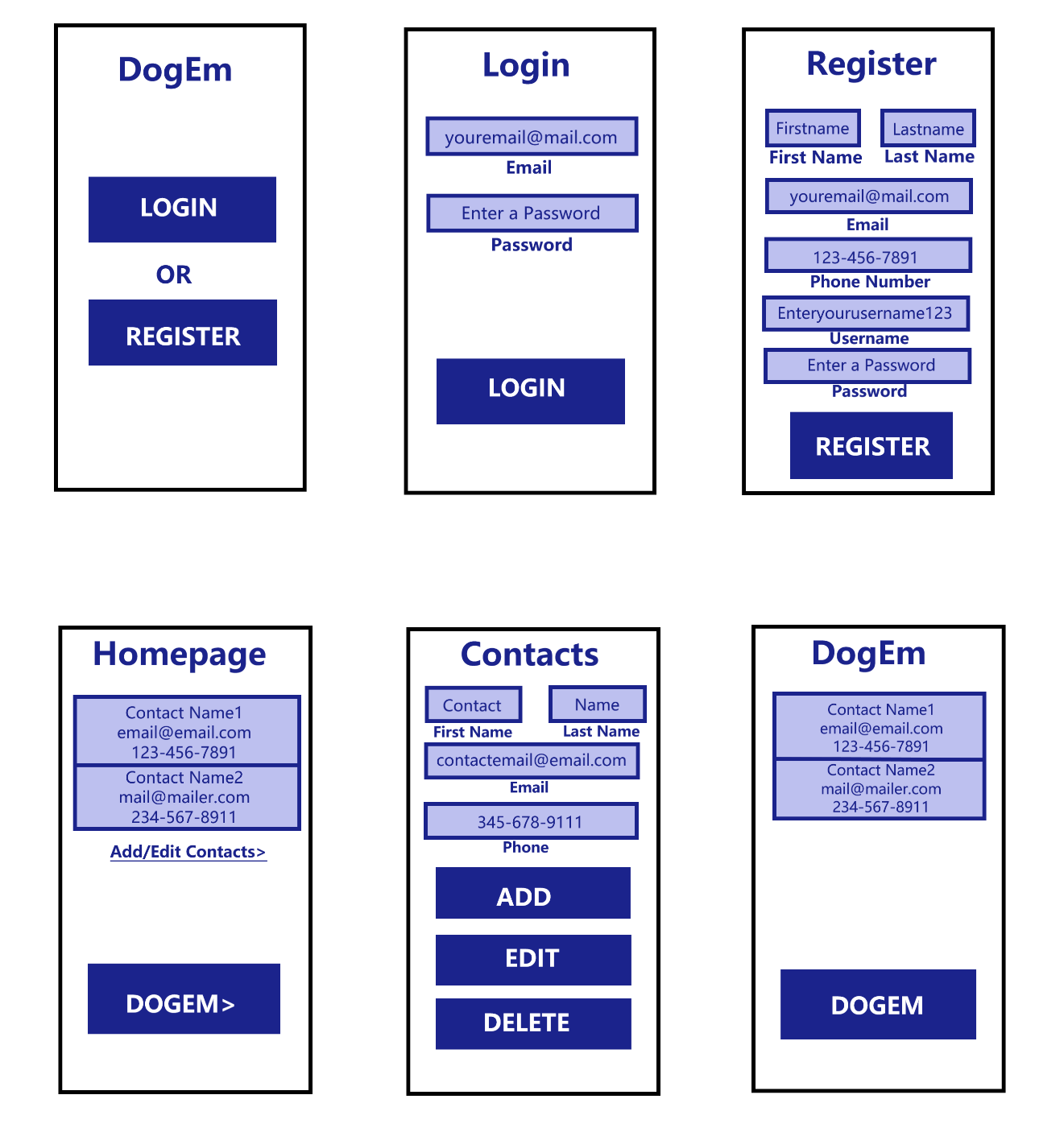


**5. Mockups**

Mockups, or paper prototypes, are a way to get a sense of what the app should look like and how pages interact. The DogEm team derived our mockups from our requirements documents.

**5.1. Initial Phase Mockup**

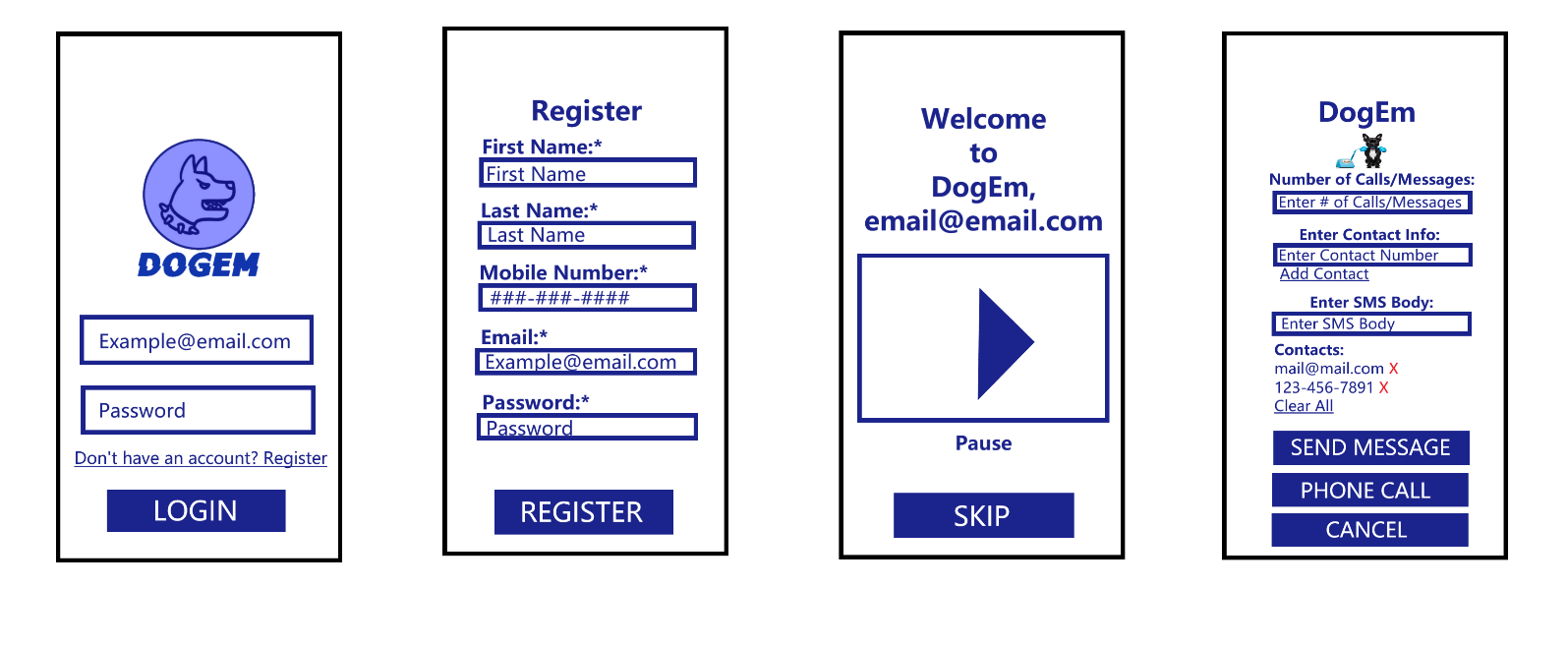
The initial phase mockup was used as a model for what our final prototype was supposed to look like.



**5.2. Development Phase Mockup**

**5.2.1. Development Phase 1 Mockup**

Our development phase 1 mockup was hugely important for developing our version 1. Our version 1 is nearly identical to this mockup.



**5.2.2. Development Phase 2 Mockup**

The development phase 2 mockup was used to make our version 2. It shows sample advertisements.

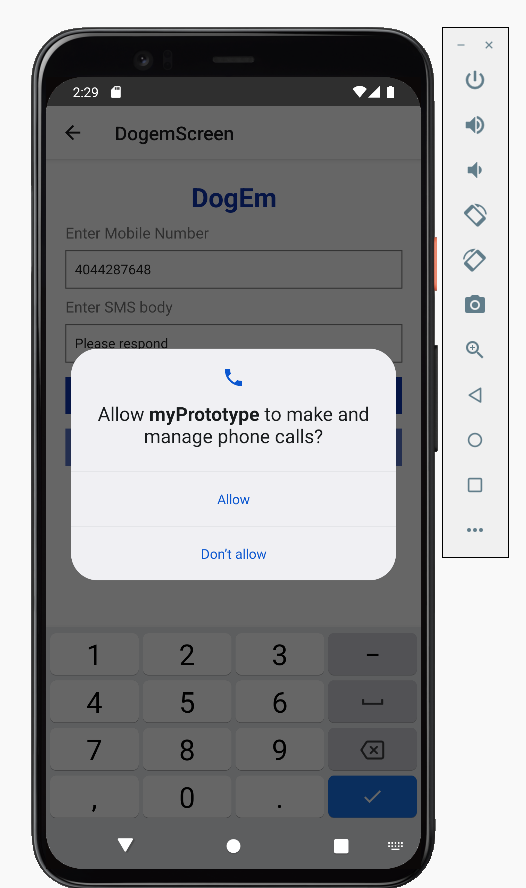
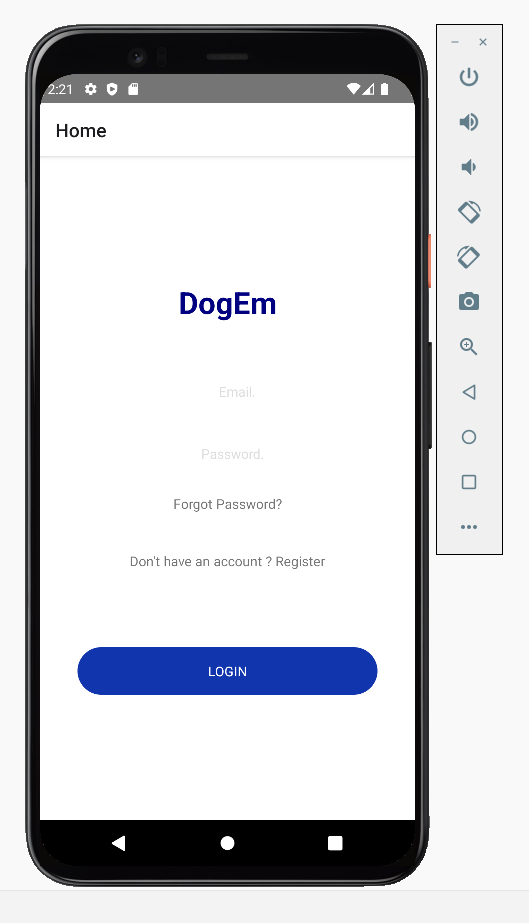
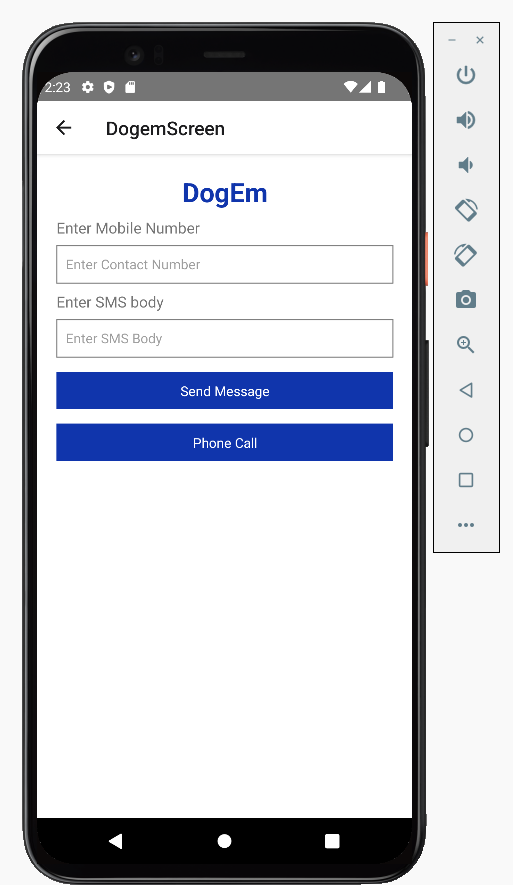
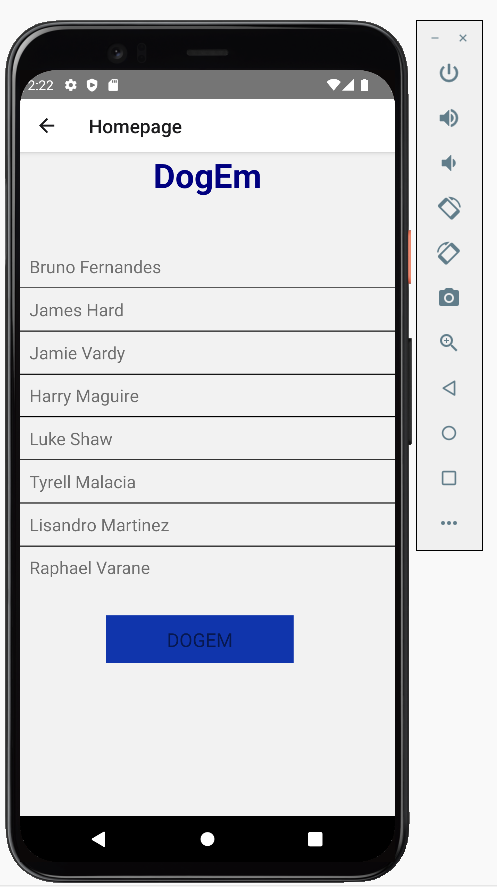
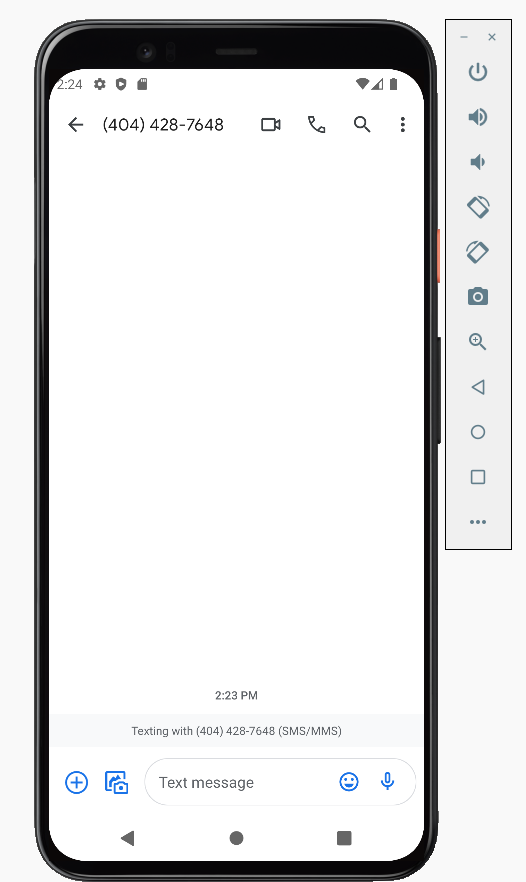
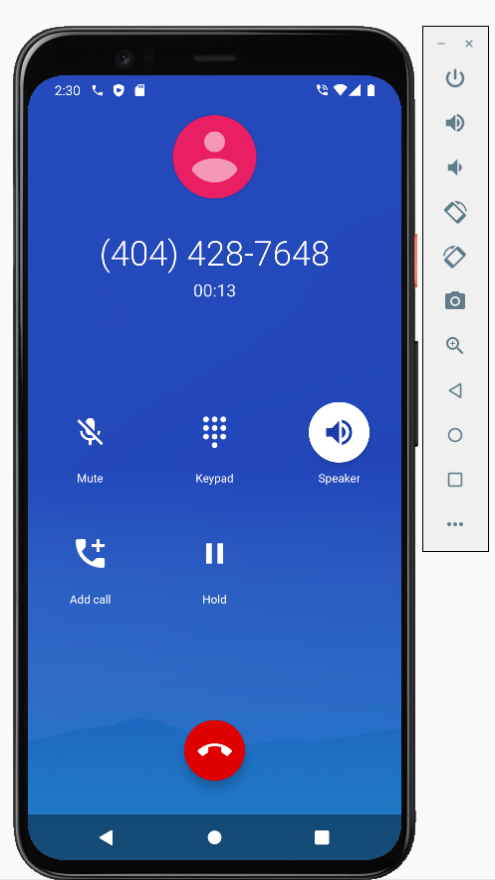


**6. Prototypes**

**6.1. Initial Phase: Member Prototypes**

During our initial phase, each team member was tasked with creating a prototype of their own. The team would then judge each of the member prototypes and select the one whose UI was the group’s favorite.

**6.1.1. Deo Intshakal A Nzeng’s Prototype**

**6.1.2. Melissa Iniestra’s Prototype**

Graphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generated

![Graphical user interface

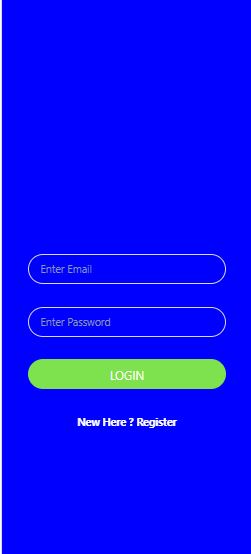
Description automatically generated with medium confidence](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD4RXhpZgAATU0AKgAAAAgABAE7AAIAAAAPAAAISodpAAQAAAABAAAIWpydAAEAAAAeAAAQ0uocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAERhdmlkIEF0a2luc29uAAAABZADAAIAAAAUAAAQqJAEAAIAAAAUAAAQvJKRAAIAAAADNzYAAJKSAAIAAAADNzYAAOocAAcAAAgMAAAInAAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAADIwMjI6MTE6MTQgMjA6NDA6NDgAMjAyMjoxMToxNCAyMDo0MDo0OAAAAEQAYQB2AGkAZAAgAEEAdABrAGkAbgBzAG8AbgAAAP/hCyFodHRwOi8vbnMuYWRvYmUuY29tL3hhcC8xLjAvADw/eHBhY2tldCBiZWdpbj0n77u/JyBpZD0nVzVNME1wQ2VoaUh6cmVTek5UY3prYzlkJz8+DQo8eDp4bXBtZXRhIHhtbG5zOng9ImFkb2JlOm5zOm1ldGEvIj48cmRmOlJERiB4bWxuczpyZGY9Imh0dHA6Ly93d3cudzMub3JnLzE5OTkvMDIvMjItcmRmLXN5bnRheC1ucyMiPjxyZGY6RGVzY3JpcHRpb24gcmRmOmFib3V0PSJ1dWlkOmZhZjViZGQ1LWJhM2QtMTFkYS1hZDMxLWQzM2Q3NTE4MmYxYiIgeG1sbnM6ZGM9Imh0dHA6Ly9wdXJsLm9yZy9kYy9lbGVtZW50cy8xLjEvIi8+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczp4bXA9Imh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8iPjx4bXA6Q3JlYXRlRGF0ZT4yMDIyLTExLTE0VDIwOjQwOjQ4Ljc1ODwveG1wOkNyZWF0ZURhdGU+PC9yZGY6RGVzY3JpcHRpb24+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iPjxkYzpjcmVhdG9yPjxyZGY6U2VxIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpsaT5EYXZpZCBBdGtpbnNvbjwvcmRmOmxpPjwvcmRmOlNlcT4NCgkJCTwvZGM6Y3JlYXRvcj48L3JkZjpEZXNjcmlwdGlvbj48L3JkZjpSREY+PC94OnhtcG1ldGE+DQogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgIDw/eHBhY2tldCBlbmQ9J3cnPz7/2wBDAAcFBQYFBAcGBQYIBwcIChELCgkJChUPEAwRGBUaGRgVGBcbHichGx0lHRcYIi4iJSgpKywrGiAvMy8qMicqKyr/2wBDAQcICAoJChQLCxQqHBgcKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKir/wAARCAGjANoDASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6/9oADAMBAAIRAxEAPwD6RrJk8QQf8JFFo1pE91cbS9w0Z+W2XHBY+pPQde9a1efTfDe1i8az36W7XNjqjl7kG4ZHgkx94YPKnpjtQB6DRVPS9LtdHsVtLBGSFSWAZyxyfc1coAKwZfF+nRaHquqMsvkaXPJBOAvJZMZx+db1eW6j8OLm98P+J2MU39o3l/PNaot0wR0YgqSucevWgD0f+1LFbiK3ku4EuJlDJC8gDsPZc5qN9c0mKVopNTs0kQEsrTqCuOuRntXn2qeENZn1TUIo9Pjma+ubaaDUzKAbVUChlx142nGOuaengO6e7sprjT4XZdemup2YgkwMDtz6844oA9Bk1fTobWO5lv7VIJTiOVplCv8AQ5watghlBUggjII715Bc+AtdSGy/0eSW3hku0NrDImUWSTKMN4I6ceozXqOhWLab4fsbJy5aCFUO99zcDueM0AX6KKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDl/ib/ySjxV/wBgi6/9FNXwJX338Tf+SUeKv+wRdf8Aopq+BKAP0joqPZJ/z0/SjZJ/z0/SgCSio9kn/PT9KNkn/PT9KAJKKj2Sf89P0o2Sf89P0oAkoqPZJ/z0/SjZJ/z0/SgCSio9kn/PT9KNkn/PT9KAJKKj2Sf89P0prkxqWkmCgdScCgCaiqUF9b3LYgvEc+gI5qzsf/nofyoAkoqPY/8Az0P5UbH/AOeh/KgCSio9j/8APQ/lRsf/AJ6H8qAJKKj2P/z0P5UbH/56H8qAJKKj2P8A89D+VGx/+eh/KgCSio9j/wDPQ/lRsf8A56H8qAJKKj2P/wA9D+VGx/8AnofyoA5v4m/8ko8Vf9gi6/8ARTV8CV95fFUyxfCXxOyyddMnU8dihBr4NoA/SOiiigAooooAKKKKACiiigAqOeeO2iMkzBVHc08kKpJOAOtcfqmonULpyzhLWEn5i2Bx3NTKVhpXNOfXZ5X22UQCdnaoZIJLkM9yzPuHrwKraRqWk3kJOnX9vdbOGEcgbH5VrpPE42rx7UR13GzDfTfJb5DtRRkYqpLeakQY47mXH+90revmEduxxXPPdp5LKDtJ+83tVOyEY15r+q2Ep+zXshcepyKn0v4m6hbyhNXtkmj7vFw1ef8AjL4j6Lot59jEm+UfeEY3EfWs19fMkccqr+6kAYMPQ1i2ykkfSOi+IbDXbfzbGXJH3o24ZfwrTr5xsdTvbBorzT5jkfMGU16r4L+Itvr8i2N+BDeYwD2f/wCvVxmnoxOJ3NFFFWSFFFFABRRRQAUUUUAcf8Wv+SReJ/8AsHTf+gmvgyvvP4tf8ki8T/8AYOm/9BNfBlAH6R0UUUAFFFFABRRRQAUUUUAZPiG6aDTtiNhpTt/DvXIX1hFqmiXemyMyJcRNGSp5GR1rU166M2sOmcpEmAPeshZfKjMjHBrmlK8rmyjofMOr6H4q+Fuvm4s55kiVv3dxGTtcejCvXPh18bbbXpYtO1vFrftwr5+SQ/0NZHjnxil7ezWCW6zJHw5YcGuV8O/D+z1W/i1KN5baOOQNsU9x/KtObS7FbXQ+j9S1tfsZ9MV8/wDxF+Kk0c02kaCxVxlZZweh9BXrEKSXtsYdyqF/vHrXmXiLwPpOmarJqEUHmzuxfbuLAn1AqIyb1kNpdDifCfgeXWJv7T1+byoM7gjn55T/AIV3V3bwrCY4APLUYVR2FcTqmp3azYSd1x/yzKbdtamjatJcwqJTk9DTnfcSsdj4anE+kSxbsSwv09RVUXMml67FNASCJAy4+tVdHc2l1KR0cV0Wm6M+ueNtNs4V3KxEsh9FByaz3ZR9E2MrTafBK4wzxqT+VT01EEcaovAUYFOrrMAooooAKKKKACiiigDj/i1/ySLxP/2Dpv8A0E18GV95/Fr/AJJF4n/7B03/AKCa+DKAP0jooooAKKKKACiiigAooooA881IPDrl4svBLZHuDWDrF75cZRT8zDAPpXe+KdLEqC+j4dBtYeorz+eCOSQmZhvDYAPYetefUvGVjrhZq5xb+DYb7UvtJheXecs27ABrUvQujWJESqGC4CDjNdZp0kUPyLznqQKxPFf2cQu23fIwwoIAxXRT95amUnZnkl74tvDNNuklR1PADEA1t+CtUu9VnEd47Ff4WIya5fWdNiF4Xll2kkkqDWj4av8A7DfJHE2E/lW9kkZ3Z0Hi3w3ZSDzoZPLkbnLKMmuVtoFtThPmx3rp7u7W/jZjJJ1PAX+tc3OZFkIVMAHjNck27msEbGnOJpxltuBkmvc/hd4ZktUk128Hz3EYjgB6qnr+NfO1pcOZhFg7mOOO9fX+gxmHw9YRsMFYEBH4VdFXdwqOyNCiiiukwCiiigAooooAKKKKAOP+LX/JIvE//YOm/wDQTXwZX3n8Wv8AkkXif/sHTf8AoJr4MoA/SOiiigAooooAKKKKACis7WtdsNAsxc6nN5UbNtB2k5NYP/CzfDX/AD+t/wB+mrOVWEXaTsdVLB4mtHmp0215I6m7h+0WcsX95SBXhmry3Nvqc8VwjKYnIb3r1fT/AB34e1KcQwaigkbgCQFc/nVrWvDOm6/Di7iAftKnDCspxjWV4MHCrh3y1Ytep4j/AG1LCpdW29hXP63rFxqEqQxycKefevXNQ+EMEwP2PUGUeki5rn5vgtqiyiSC9tmIPcEZqIwqRJcoyPJr3wyLhhJJcbsnOBSN4fSzhWcEqyHOc9q9df4S63s+WW1LDoNxrK1D4SeJrmP97LbRxqMsTJwB+Vac07bEKMbnm1vetH5xGAJDwPSqksrPLz09q6v/AIQiDztkviPT0C8EhXOP0rqfDnhLwLpcgudb1z+0pF5EQjZU/LvXPzRb3R3fU8TbSnL7mZfwx+Ht3r2tQ6lewtHpsDB97DHmEdAK+j1UIoVRgAYAqjotzp93pMMujlPsm3EYQYAHpir9dsIpLQ8+fNzWkrMKKKKsgKKKKACiiigAooooA4/4tf8AJIvE/wD2Dpv/AEE18GV95/Fr/kkXif8A7B03/oJr4MoA/SOiiigAooooAKKKKAMDxtbwz+D9Q8+NX2Qsy5GcHHWvG/h/YW2peMLa2voVmhZWJRxkHivX/Ht7BaeDr8TSqjSxFEBPLH0FeQfD/ULXTfGFtc30ywwqGBdug4rycVy/WYXPt8lVX+ysQ4Xv0t6dC38RvDtt4e1+L+z18uG4TeEH8JB7V3HgjxfHF4De71mbiyfy9xOS47D61xXxK8QWmv69Aumv50UCbN4H3iT2o1Xw/qGl/De0kmjdRNc+bKmPujGFzWMZunWnKlsjvq4dYrA4ejjHacmt9/6sdFcfFq8ZXubLRS1mjbTK7H+Y4FdR4d8fadrmlz3Lg20lsm+WJjnA9R615t4b1SwPgPWtLvpUjkI8yIN/Efaqngm3MZ1C/u4pG0+K1dZio+9kcCrp4mrzR1vdfcc2JyjBulUSp8jg0k7/ABLTv1O6tviPqGs3dymgaSksVuhdmmlwSo9hUGkfEPU/Eut22m2enxQKzf6QXbd8veuH8L3gsp7uW31aLTRIpj2zRl9yn6d67X4fxeF9N1Jjb6sLvUZQQC6FB7gZqqVarUcby9dvwMsbgcHhI1OWldpLl0fzb6HFeO7GCw8Z3cNvGscRIbaowBmunR/Cd9daZotpp9vKbqLbLcpw8b4rm/iDdRXPjW8eJwyrhcg56V1Vhp/he0m0rW4tStbf7NDumhDZZ3x6VhDWrNK1r/geliJNYKhKblzculr720udCZrf4ZeE0jYTXymU7SBjBPr6Vi2fxfEsdy9xppHlrmJY2LE/U9hVTXfF8niTwVqrTQpFCk6pbkdWH+NYngBrdbPXjcFP+PI43fjW868lUUKTtGx5tDLaTw1StjKfNUUtde9v8zptL+L6zzSJf6cUG0mPyWLFj6YpdM+LhutaitrvThFBLIEDq5LLk45FcZ8P2hHjiz88rs+YDd0ziqV08Q8auyFRGL0EHsBvrFYqtyxlzdTvnk+XutVpKltG6d357H0cDkAjoaWmQkNChU5BUYIp9e6fnL3CiiigQUUUUAcf8Wv+SReJ/wDsHTf+gmvgyvvP4tf8ki8T/wDYOm/9BNfBlAH6R0UUUAFFFFABRRRQBl654d03xDBHFqkHnLG25OSCD+FYn/CsfDP/AD4t/wB/W/xrr6KzlShJ3kjqp4zE0o8tOo0vJnO6b4F8P6XOJrbT4/MByGf5sfnW5dWkF5bNb3MSyxOMMjDIIqaiqjCMVZIzqV6tWXPOTb7tnHN8L/DTXBk+yOBnOwSHb+Vaeq6N9l8J3VhoFtFExjISMKME1vUVCowSdla5s8diJyi6k3K2tm9D5/0/StPto5IvEWk6mLlWODAvymr/AIR8FX+o+Io7r7LNa6fFJu3TcMR6V7eY1PVQfwpQAOlckcDBNXex7lXiSvOElCNnLzul6Locm/wz8NSSM7WLFmOT+9b/ABpp+GPhkjH2Jv8Av63+NdfRXV7Cl/KjxFmGMW1WX3s5iT4f6DJpEWnG2YQRuXXEhzk+9R2vw58PWlvPFHauROu1y0hJxXV0U/Y0735UL69ikuX2jtvucnYfDfw9p94lzFbO0kZyu+QkA1G/ww8NvdGY2r8tuKiU4rsKKPY0rW5UP+0MXdy9q7vzGRRJDCkcY2ogAA9BT6KK1OLcKKKKACiiigDj/i1/ySLxP/2Dpv8A0E18GV95/Fr/AJJF4n/7B03/AKCa+DKAP0jooooAKKKKACiiigCtPqFpaybLi5iiYANh2A4JwD+fFPW8t3heVJkaNCQzA5AI6/lXOeK/C8+vXtrJBKkaIjpNuzlhwVx9GArJj8F6wklqTeqVWL96FcgCQkliBjkHPt0rnlUqKTSjoejTw+GnTUnVs+qO7huIriBJoXV45FDKwPBB70/cPX9a4ODwhrENq8PnxFvKVRJ5zjIAUeXjsODz70o8GapIEM1yvykbVErYRd5JUevBxS9rU/kH9Vw9/wCMreh3eRTUmjkz5bq2Dg4OcH0rjtJ8Lavaa7b3V1e+ZBEu3aJTwOflwRyOR3FQTeDtVW9la0mijiedpFIkYFMkHdgd+ox71XtJ2vyk/VaHM17VbHdZo3CuO1fQ9R1DxDItncyww/ZtzMSQpk6D9Kpjwpr8cWIrmJVfIMfnNiIZB4OOelDqyT0iEcJScU3VSbO4nu4LZVaeRYw7BFJPUnoKl3CuTbwpcP4cisZJFeb7SJpXZydwye9Z58Ia78oF6mVj2iTzWyowRtA9Dwc0OpNP4RRw9CS/i23O83A96Nwrgz4K1aONvst6InYEbhKx4Kjj8wfzqtF4X11NQt4RIQIo1fzmmJWM7ySFwBnjtU+2mt4GiwdBptVl9x6LuFMmuIreMyTyLGgxlmOAM8V5xfeHNes7CZpT9p8yXcIIZGOTtIz0455q7aeENXkmaS9uDIkkafIZiOhU7SMHpg96SrTbtyFPA0Yx53WVjvgcjI6UU1F2xqvoMU6uo8oKKKKACiiigDj/AItf8ki8T/8AYOm/9BNfBlfefxa/5JF4n/7B03/oJr4MoA/QT/hNdE/5+/8Axw0f8Jron/P3/wCOGvBZLvU7VTEgaaULkKYyd+VJLbu2G4xVmyv7u81JVClLYAksYiN2AnGT7lvyrwf7Qrb6H2CyTCt2vL+vke4/8Jron/P3/wCOGj/hNdE/5+//ABw14P8AaLiISyA3LXYdw6FWaNU34Bx04XkY60qX2qSOSo/dqyAN5ODKGkK7vbC4NP8AtCt5A8kwq6y/r5Hu/wDwmuif8/f/AI4aP+E10T/n7/8AHDXz/Fquqxx2ieU88hiHmb4iNzYbvjsQB2qdr3UNySRzF18mTrblQXwpAIxnjn8qPr9fyF/YuF7y/r5HvP8Awmmif8/f/jho/wCE00T/AJ+//HDXhv26c6fZXE3mKftGJSIzyvIzgdulQz6jfJdnyFbyWmwv7oksML0z06n0pLMKz6IbyTCpXvL+vke8f8Jpon/P1/44aP8AhNNE/wCfv/xw14zpTzPp6m7dnmDMGLLg/eOP0q5xUPM6qdrI0WQYZq92etf8Jpon/P3/AOOGj/hNNE/5+/8Axw15LxRxS/tSr2Q/9X8N/Mz1r/hNNE/5+/8Axw0f8Jpon/P3/wCOGvJeKOKP7Uq9kH+r+G/mZ61/wmmif8/f/jho/wCE00T/AJ+//HDXkvFHFH9qVeyD/V/DfzM9a/4TTRP+fv8A8cNH/CaaJ/z9/wDjhryXijij+1KvZB/q/hv5mes/8Jpon/P3/wCOGl/4TTRP+fv/AMcNeS8UcUf2pV7IP9X8N/Mz1r/hNNE/5+//ABw0f8Jpon/P3/44a8l4o4o/tSr2Qf6v4b+ZnrX/AAmmif8AP3/44aP+E00T/n7/APHDXkvFHFH9qVeyD/V/DfzM9a/4TTRP+fv/AMcNH/CaaJ/z9/8AjhryXijij+1KvZB/q/hv5mdj8TPFGlah8LfElta3G+V9Om2rtIz8hNfElfS/ibH/AAhut/8AYPn/APRbV80V6mDxEq8HKR4GaYKng6ihBvVdT7nh03SZ4VkSwhwR054/Wqby+G47y4tWggE1sYxKm1sr5hAX8yRWqumzW0rSwozhiB5YYADnrWRqPgtr/UJL1ZzDM1zFNlQPupsyh55B2A+xrHDwc4fvoJP8PVHZOvJP3Zv72WWh8PKjOyWIVW2MxkGAfQ89aUQeHzH5gjstm/Zu8wY3enXr7Vl2vgG4iuHlubtZt2ONnoki55Y8/vM8YHHSkn+HrS+VtuAAiLGUwQrDyhGT8rA5O3NdHsaf8q+4j6xU/mf3s3f7H0z/AJ8IfyP+NH9j6Z/z4Q/kf8avrayKgULwBjrS/Zpf7v6ij2MP5UV7eX87+9mf/Y+mf8+EP5H/ABo/sfTP+fCH8j/jWh9ml/u/qKPs0v8Ad/UUexh/Kg9vL+d/ezOl07RrWymubmzhSOJdxY54/WsOLWfDEsyobAxq5wsjxMFP45rq7mwa5026hkhWQOmAjYINcNYeHdfbUYhf6ZF9k3DcplUgDjt+FY1KaTXLFfcdFCpGSbnN/wDgVuhuXEvhu1vJLWeCBZo4fPZMN9zOMj1qw1toSRiR4bREIzuZ8D+dVtb8GHWJpphMYJWjRI3UAlME578gg4xVZfA1ybqNprtHhifcieXz97dyd3P5Vt7Gn/KvuOV4ipf4n97LU0nhqDyy6WZWRwgZXyAT6nPFWHttBjYh4rNSF3kF8YX169KzJPAcytC1tPEhiGMPCGVuSc4yPWol+HRWRi1yZFIB+fJIPGcDdjHHpR7Gn/KvuD6xUv8AE/vZvLpWlOoZLKBlIyCMkH9aX+yNM/58If1/xrQW1kVQAgAHAGRR9ml/u/qKPYw/lX3Fe3l/O/vZn/2Rpn/PhD+v+NPj0bS2WQtYQ/Kue/8AjV37NL/d/UU9LaUwzKF5ZCByKFRp3+FEzxE+XSb+9nKR6p4TlvBaRrbmZm2BdrdfSrdyuiW19HZvp++Z134jiZsLnGSR0qtF8P7KG8W7SBhOr7w3ncZ+lXNR8M3N5rcGoxmDMcQjKyx7sfMTkEMMHmqdGl0iZU8TWafPJ/exUi8PySCONLFnJwFWQEk/TNQ3L+HbW3854Ld48kZi+bnIGOD/ALQqungRo4YljmVZI0iUSCMZym7nr33VWj+Ht1sYTXwZmbdkIfSMd2P/ADz/AFqfY0/5V9xr9YqfzP72dANJ0wjP2GH8j/jR/ZGmf8+MP5H/ABq+LaUADb+opfs0v939RR7GH8q+4r28v5397M/+yNM/58YfyP8AjR/ZGmf8+MP5H/GtD7NL/d/UUfZpf7v6ij2MP5V9we3l/O/vZxvxA02wh+G/iKSC0jjddNnwy5yPkNfG1fa3xHhkT4Y+JCy8f2bP3/2DXxTW9KKitFY8vGTc5pt3PtI+JpLOQwTeVe+SyrLPE+3O5yi4XnJBHPNV4fGNzNPaBrGKGKUCSQtOThDE0gI46/Kc10Q02xUxEWcAMIxHiMfJ9PSnGwtGCg2sJC42/IOMDA/Qmsz0bM56z8Ztes0cViolRm3iSfYu1VQ5BKjJxIODjoaWfxBqp8OQXttbQG7kvzb+SGJVlEjKRnsSF/OtxdI05EVEsbdVV94AiGA3r9eBU62tui4SCMDzPMwFH3+u76+9ArPqcpbeO0kVHMPmpLdtEMEqyL5ioMjB5y3OSOlTQ+M5Wjjmm03ZA6xOWWfcwWQkLxt65XmugbS7B3V3srdmVi6sYxkMTkn655qQWdsqgLbxAKFAGwcben5UwtLuc9beJrvUbGC7t7b7LG1zCjCVS3mJIR908YPPoa6jn1qvFY2sEflw20Uabt+1UAG71+tWKRSXcSZpl026a3ZVlEeVLrkA/TIrmxrGoyw6W8dzbiW5jVnh8vqP4mJz8oA/WupjGbeYHkY/rVCTS7GWSOSSzgd4hhGaMEqPQelMi2rMew1+dbi9jvi1wkO1keCLOQQTn5SeOOO9ZkfizUruxeeIxoI7lhIqqAyxBQRw+MnnnFdfBaW9ru+zQRxbzltigbj70kthZzjE1rDIN2/5kB+b1+tIfKzA0nxRLe30yXEMoRiRbiNASQMAkgHI696r6jreq2f2k29wtwizLCh8lQ27GWA5AOOOtdTHaW8U7zRQRpK/33VQC31NNksrWaEwy28TxltxRkBBPrimHK7HM2XiHVbjU7WCRUUXKEoBCSuNud+7Pr2rf0W7nvdJhnuiplbIYouAcHHSrKWsEcgkjhjVwu0MFGQPT6VJHGkSBY1CqOgAxSGlbcdz61HdXo0/Tbu6ZS4ijztBxnkD+tSVJGiyRTpIoZWQgqRkEU1uKfwnLyeK5oJgs1jHtXaZXS5DBQzFRjjnp3xVUeOWkEgg08SPGGc5mIBRUDZBK+hrS/tHQbdXiVI1CMECpbnDEE8LgfNg56VAureGbZisMcY3jnyrVjnK5xwvXHagXzK7+NXjmW3bTJGuC3zJG5YBcKc529cOODj61Ym168tE1H7QYmaG4WOIqhAVTEJDkZ5IGfrWulrp97DFOkEEqNiRGMY9OD+WKW8022vYHimjADsGLLwcgYBz9OPpSKszm/8AhMZ7SACe2N1NJJsjCDHI35B25/55nHHepG8Z3G1pF0r92qSud8+GAjClsjb1+bj6VuQaLYQ2f2Y20UkZOW3xr8x9TgYqx9itdpH2eLBBBGwdD1/PApitLuc3ceM8301pBDgo4CShshh5gjbqPU9s1v6Rcy3miWNzMcyTW8cjkDHJUE/zpV0yxWZ5Vs4BI5yziMZY5zyfrVlEWONUjUKigBVAwAPSgaT6nO/ET/kmfiT/ALBs/wD6Aa+LK+0/iJ/yTPxJ/wBg2f8A9ANfFlaQ2POxnxo+/PIX/ntH+Zo8hf8AntH+ZrjP7avLWVIrFpriDK+T9ohZnuAzkNhuwUDOT1FQx63rplsnuvkjIWSRY7Y/NuhZtnXsygfjWZ33fc7nyF/57R/maPIX/ntH+ZrhbHxPqt18syxwIJGCztbOwkwsZVNo6E725/2anln1q48NQCC7ddQOpmMyCLGFErYDD+6QBn2NAXfc7PyF/wCe0f5mjyF/57R/ma8/g8UaxGYvMtJUeW9dXhmiYlV81U2hh2AJP05q1BrniAQxTyQxyhkgcwrbsp/eFgVznjbgH8aBX8ztvIX/AJ7R/maPIX/ntH+ZrirLUtX1PTYLm53QOl5CCkAI3KxG5WBHbJ/rXX0FK76luOECGUeanQevrUXkL/z2j/WoZ0aTTLtI5HiYx8PH94fSuQa7mA0iOW+u4rp4laTJYKAOuRj5mPTBoJV03qdr5C/89o/1o8hf+e0f61yOkXt+s9/9njmvoVKmPzZGBzg5+8BznHHAFZRvtZNi8l691BLHdsfLkDDzBtGEUxngA9M9aB3fc9D8hf8AntH+tHkL/wA9o/1rjdF1PVpdUuI7m3aQuSfLZ2TyFGMDJBBz7VW1U38MN1PZ3F4kRnESiSaQjA6tnkgE+lAXdr3O78hf+e0f60eQv/PaP9a4fT31mTV7SOaa4KzRbpH3EBF291IwGz3610Ogmc6ND9qeSSUZDNL94896Bq76mv5C/wDPaP8AWnrbB4ZlW4VSyY3KeRz9KrYqK9uJrTSbye1j8yVIsquCe45wOuOtCJmny7lJvCdqc7b6ZcSebEBJ/qm5yV49z1zRb+EdOtgBHL0YPy55OzZnp6GsZtc1YMHhMdxDGELMtq4Mm5ypxk8YHNU4fE2uXQmWCFF2K7h2tm5wgYLjPXORTFY7ez06GysobWGZfLhQIuSc4H4VN9nX/ntH+tcPP4h1+3uI7f7NE+ZMGYxMqtlUIXHOPvEZ9qsXd7f6fFqeZp3BulEbN1VTEGKrx6ggfWkPXudh9nX/AJ7J+tH2df8AntH+tcB/wkGr2liTaKbhWl2NLKrkRff5+bnB2r3/AIvcVYbW/ERjeYRwptjmfyhbs3MaqQM553FiOnagLvudv9nX/ntH+tH2df8Ansn61ws3iPVJ9TuLRLdvKWQBWWJlZMShSM55yDn9eldPorSvoOntcFzKbaMuX+9u2jOffNA1d9TN+I8QT4Y+JCJEb/iWz8DP9w18UV9p/ET/AJJn4k/7Bs//AKAa+LK0hsefi78yufe2BRipR9mYZV3IPQgD/GjFv/ek/wC+R/jWZ6XMRYH+RRipsW/96T/vkf40hFsOruPwH+NFg5iLFGKmAtj0d/8Avkf40bbf+/J/3zRYOYhpal22/wDff/vkf40bbf8AvSf98/8A16LBzBF/qZfoP51DgelW4xB5MuGfGBn5R61CTaggGVsnoMDn9aLEqSuyLFGKlX7K2dsjnBwcAf40jNaL96VhzjkD/GixXMR0YqbFt/z0f8h/jTBJZEEic4HXpx+tFg5hlFS/6Lu2+Y+cZxgf40u22PR3/wC+R/jRYOYiqWD7sv8AuUbbf+/J/wB8/wD16esVtLDMjF2VkwwIoSJlLQ5tvEiA4jsZ3EkvkwEFQJWGcjrx0PWqo8YwyMPs2nXMoJC5BQYbZvxyfTPNbb6BobmQtaqTKcsdg5Pr14qSPSNIiAEduFCkEYjHGF2/y4oDmf8ASEsrqO+sYbqEHy5kDruHODUk8EVzC0U8ayRt1VhkGpobezt4Ehg3pGg2qqqMAfnT8W39+T/vkf40FcxVgtoraARQIERegqTFTD7MRkPIf+Aj/GjFv/fk/wC+RQHMQ4oqY/Zh1dx9QP8AGgfZmUFXcg8ghRz+tIOY5b4if8kz8Sf9g2f/ANANfFlfa/xHEI+GPiTy2cn+zZ+o/wBg18UVtDY83Fu8kfZDQaxHOg020uLOAlTDCsi7E+cl94z3XGAM4qGPT/EKTWcs8l3OYgrspmA+cwsGB9t+3867SisrnpcpxViniVgBfLfCLzG2CN0EgO2PbuJY5XPmf4VYl0O+u/DUFhI9yky6mZGlaXLKglZlcHPQDacV1tFFw5ThYIPFVu0WYJEdrx5JTFICjAyrk4yOCmSOvXGM1ZgsfE0UUUv2q5efZAWSSQFNxLCTI9ANv9K7Gii4uU5Cx03WLnToH1fzJbqG7hkUPgFACu8g7jkda6+iikUlYSeBbrTLuGRd6vHgrnrXJNpkwXSbeXTJnaGJPNuEKkptPCDJ4z3I7V2kX+ol+g/nUNMlK7ZzGk2GpW89/JaQJZrJt8tJ0HUA/wB0+p+8eT6VljQtWjsXF1bNLOt200ZjZJVYlQMsHHTPpyK7ujii4+VM5TRLHW7bVLg3G1RKS0sksasp6YCYIbGM9ag1bQLieG5ntrJYnknACRIhPlgddpODknvXZUUByq1jjtP8PX0eq2jTJ+68vM7MPm+7jZu3cj2xxXQ6FaSWOjQ28qbGTI25zgZ4rQopDSsLUN8Lo6TeDT+LjyvkxjPUdM98ZqWpEdI4pmkYKoTkk4AprcUtjjzb66zLLbyXyrGEKRzSpljvO7djr8uKpw2/iq4EyyvdwKFdk/eKG3bBtGcnI3Z9PpV+TxTcK7BRZoDN5Q3s2YRk/NJ7HHt1qCPxTqt0VNvaWqKzBP3m48+Xvzx24xRqRoMntfFMdxHHDcTm38zO8kO/Kp15Hy531NqNnc6da6o8MRWKe6R+vDDyh19AXABro9Nu/t+l212ybDNGHK+mRVqi5fKcGkOunT9+kiSOJpf3mzZuZfnwVCsB/czyOlW20/xK0byNeXXm+XMQFdVUsFXy+O2Tu/rXYgADAGB7Cii4uU4uSHxLcapcK8Uv2UyAgOyleJRgrz/cz2H4nmuo0aCS20KwgnXbLFbRo656MFAIq5RSGo2Oc+In/JM/En/YNn/9ANfFlfafxE/5Jn4k/wCwbP8A+gGviytYbHnYz40fop/ZFt6N+dH9kW3o351foq7I5/bVP5ih/ZFt6N+dH9kW3o351fqKa5ht13TyJGvq7ACjlXYFVqt2TZV/si29G/Oj+yLb0b86T+3dK/6CNr/3+X/Gj+3tK/6CNr/3+X/Gny+Rr/tPZ/cxf7ItvRvzo/si29G/Ok/t7Sv+gja/9/l/xo/t7Sv+gja/9/l/xo5fIP8Aaez+5jxpduqsAGw3Xmk/si29G/76pv8Ab2lf9BG1/wC/y/40f29pX/QRtf8Av8v+NHJ5Ctiez+5jv7ItvRv++qP7ItvRv++qb/b2lf8AQRtf+/y/40f29pX/AEEbX/v8v+NHJ5D/ANp7P7mO/si29G/76o/si29G/wC+qb/b2lf9BG1/7/L/AI0f29pX/QRtf+/y/wCNHJ5B/tPZ/cx39kW3o3/fVH9kW3o3/fVN/t7Sv+gja/8Af5f8aP7e0r/oI2v/AH+X/Gjk8g/2ns/uY7+yLb0b/vqlGlWwVhhsMMHmmf29pX/QRtf+/wAv+NH9vaV/0EbX/v8AL/jRyeQrYns/uYn9iWZzmP73X3pf7GtP7p/Oj+3tK/6CNr/3+X/Gj+3tK/6CNr/3+X/Gjl8h/wC09n9zFGj2oGAGA+tH9kW3o350n9vaV/0EbX/v8v8AjR/b2lf9BG1/7/L/AI0cvkH+09n9zF/si29G/Oj+yLb0b86kg1SxuW2293BK3okgNW80uVdjOVStF2k2ih/ZFt6N+dH9kW3o351fooshe2qfzHCfFPTobf4UeJpIw24abN1P+ya+F6+8/iz/AMki8T/9g6b/ANBNfBlMiUpS+Jn6R0UUUEnP+LvE0fhrSvNwHuZcrDGe59T7CvFdT1e+1e6ae/uHlYngE8L9B2rpvihdSTeKlhYnZBEAo+vJrjK7KcUlc/SsiwFKhho1be9LW/6BRRRWp9CFFFFAworLuNaW3ku0aLLQ42DP+szj/GkfXI1lljVAzJFvA39Wxnb+VTzI5/rFLa5q0Vlf25F9lklCYKKvyM2PmPY0/wDtUPteMR+VhSxd8HJ7CjmQ/b03szSoqha6i00E0zoAkecABs8H1Ix+VH225AiUwRb5uU+c4xjPPFHMivawtcv0Vj/28FlZHg24UYbdwWz0qR9UlghuJJ44sRMFUK5+Yn60cyJWIpvqalFUI9QkumH2SON1CqzFnx19KjbULpIndoIyFlMZKliBjueKOZD9tC1zTopkMnmwJJx8wz8pyPzp9UahRRRQMcjvGwaNmVgcgg9K9F8CeOp2vI9K1mXzFkO2GZjyD2BPfNecU5HaKRZEOGUggjsamUVJWODHYGljKTp1Fr0fZn0rRVbTpzc6ZbTt1kiRz+Kg1ZrgPyOS5W0zj/iz/wAki8T/APYOm/8AQTXwZX3n8Wf+SReJ/wDsHTf+gmvgygR+kdFFFAHlnxW0eRby31WJSYnXy5CB90jp/n2rzuvpC8s4L+0ktruNZYpBhlbvXmesfCq4Wdn0S5R4yciKY4K/j3rqp1FazPuclzqhCisPiHytbPo0ed0V1v8AwrPxJ/zyg/7+ij/hWfiT/njB/wB/RWnPDufRf2ngf+fsfvOSorrf+FZ+JP8AnjB/39FH/Cs/En/PGD/v6KPaQ7h/aeB/5+x+84qSwtpZC8kKsxJOT7jH8qb/AGbabdvkLjOff867f/hWfiT/AJ4wf9/RR/wrPxJ/zxg/7+ilzQ7k/wBoZf8A8/I/ejiv7PtfNMhgQsxBOR1I6cU06baEj9zgegYgHvyO9dv/AMKz8Sf88YP+/oo/4Vn4k/54wf8Af0Uc1MX9oZf/AM/I/ejjIrSGFWWNW2tnKlyRz7Go/wCzrbZt8tuvB3nI+hzkV2//AArPxJ/zxg/7+ij/AIVn4k/54wf9/RRzQ7j/ALRwFre0j95xJ0+1MZQwqVOOD7dKe1pAzZaJSd27nnn1rs/+FZ+JP+eMH/f0Uf8ACs/En/PGD/v6KOaHcP7RwH/PyP3o4kadaqwZYtuDnAYgH6inSWcEilShGWLZViDn1yK7T/hWfiT/AJ4wf9/RR/wrPxJ/zxg/7+ijmh3D+0cBa3tI/echHGsUapGNqqMAelOrrf8AhWfiT/njB/39FH/Cs/En/PGD/v6KfPDuV/aeB/5+x+85Kiut/wCFZ+JP+eMH/f0Uf8Kz8Sf88YP+/oo9pDuH9p4H/n7H7zkqu6Ppk2satb2NupZpXAJH8I7n8BXTWvwu12WQC5a3gTu2/cfyFei+GPCNj4agPkZluXGJJ2HJ9h6CplViloebj89w1Ck1RlzSe1jbgiWC3jiT7qKFH0AxUlFFcZ+bN3d2cf8AFn/kkXif/sHTf+gmvgyvvP4s/wDJIvE//YOm/wDQTXwZQB+kdFFFABRRTWdUXLEADuTigB2KMVX+323/AD3i/wC/go+323/PeL/v4KV13HyS7fgWMUYqv9vtv+e8X/fwUfb7b/nvF/38FF13Dkl2/AsYoxVf7fbf894v+/go+323/PeL/v4KLruHJLt+BYxRiq/2+2/57xf9/BR9vtv+e8X/AH8FF13Dkl2/AsYoxVf7fbf894v+/go+323/AD3i/wC/gouu4cku34FjFGKr/b7b/nvF/wB/BR9vtv8AnvF/38FF13Dkl2/AsYoxVf7fbf8APeL/AL+Cj7fbf894v+/gouu4cku34FjFGKr/AG+2/wCe8X/fwUfb7b/nvF/38FF13Dkl2/AsYoxVf7fbf894v+/go+323/PeL/v4KLruHJLt+BYxRUUdxFN/qpEf/dYGpaYttAooooA4/wCLP/JIvE//AGDpv/QTXwZX3n8Wf+SReJ/+wdN/6Ca+DKAP0jooooA5/wAY+LLXwlorXc48yZztghz99v8ACvn3XfF2teIrlpdQvZChPywodqKPQCup+M2oS3PjCO0Zj5VrCNq+7ck/yrzyvmMfiZzquCeiP1nhvKqFDCRxEo3nLW/ZdEg3Me5oyfWiivMuz6vlXYMn1oyfWirFlZS31wsUCFiTzihysrsUuWKuyvk+tGT61cv9Plsr9rZlbcDgAipb3R57KxhnmRl8zNT7Raa7ke0pe7trsZ2T60ZPrWjY6NcXltLOqNsRC2RWPPc+RMUaInau5jnGPwqovndoi9rSV9VoT5PrRk+tVzewc/Ofb5Tz9PWmi+jaXauNoXcWJwfyxV8k+we1o7XRayfWjJ9aq/2hAVJBY4HQoRmtrTNIn1GJ5Y0by1QtkVM7wV5aCdaio811Yz8n1oyfWrFtZTXV0II0JbOCKk1DTpLC+Nu6tu6DI71POr2uXzU+bk0uU8n1oyfWtG70eez0+K5lRl8wkVnURmpK6Y4ShNXjqGT60ZPrRRTuy+Vdie1vrqymEtpcywSKcho2Kn9K9f8Ahx8TJtRu49H8QSBp3+W3uSMbz/db39DXjNPgme2uI54WKyRsHVh2IORXTh8TOhNNPQ8rMsrw+YUXCcVzdH1TPrmiqumXJvNKtbk9ZoUkP4qD/WrVfYJ3Vz8QknFtPocf8Wf+SReJ/wDsHTf+gmvgyvvP4s/8ki8T/wDYOm/9BNfBlMR+kdFFFAHifxq0SWHWLbV41Jhnj8p2A+6w6fmP5V5fX1fq2k2mtabLY6hEJYJRgj09x6GvGNf+DWr2lw76FIl9b5+VGYLIvtzwa+ex2Bqe0dSmrpn6Xw9n+GWGjhcTLllHRN7NHm9FdT/wrTxd/wBAeT/v4n/xVH/CtPF3/QHl/wC/if8AxVeb9Xrfyv7j6v8AtPA/8/o/ejlq09A1n+xNUS6aBbhFPzRscZ/Gtb/hWni7/oDy/wDfxP8A4qj/AIVp4u/6A8v/AH8T/wCKpPDVWrOD+4zqY/L6sHCdWNn/AHkUvEPiP+3NaF9HbLbAHIQHPNXdb8Zrq+gwaemnxwNGMNIGzmj/AIVp4u/6A8v/AH8T/wCKo/4Vp4u/6A8v/fxP/iqn6nU/kf3HN7XKfc/eR9zb3v8Agh4f8ZromlXFnLYJcmVSquWxtrlLuJLy7a4k3Bm6qrYBrq/+FaeLv+gPL/38T/4qj/hWni7/AKA8v/fxP/iqqOGqxd4wf3FxxGVQnKaqRvLV+8v8zi5bCNlPlkq3VSTwv0pYrFY0AMjk7cNg4DV2f/CtPF3/AEB5f+/if/FUf8K08Xf9AeX/AL+J/wDFVp7PEWtyv7i/reWc3N7SP3o5D7LF/dPXPX2xXWeGPF6+HrWeCSxS6EilVJbG2n/8K08Xf9AeX/v4n/xVH/CtPF3/AEB5f+/if/FVnLDVp/FF/cTWxOV16fs51I2/xJfqZuna6bHXv7Ra3WVS2WiJwCKn8SeJf7e1VbuK1W2C4IQHOT61b/4Vp4u/6A8v/fxP/iqP+FaeLv8AoDy/9/E/+KqfqlS9+R/cL6zlftVW9pHmSt8S2+8dq3jVdT8Ow6YunxwvHw0obOfwrlK6n/hWni7/AKA8v/fxP/iqP+FaeLv+gPL/AN/E/wDiqawtWO0H9xWHxWWYeLjSqxSvf4l/mctRXU/8K08Xf9AeX/v4n/xVH/CtPF3/AEB5f+/if/FVX1et/K/uOj+08D/z+j96OWq5pGmT6xq9tp9qhaSeQIMDp6n8OtdNZfCnxXdzBZLFbZc8vNKuB+RJr1nwR8PbLwjGZ2f7VfyLh5iMBR6KO31rpw+Bq1JrmVkeTmXEWDwtF+ympTeyWv3nV2lutrZw26fdijVB9AMVNRRX1Wx+QNtu7OP+LP8AySLxP/2Dpv8A0E18GV95/Fn/AJJF4n/7B03/AKCa+DKBH6R0UUUAFGM0UUAJtHpRtHpS0UCshNo9KNo9KWigLITaPSjaPSlooCyE2j0prPGjKrsqs5woJxuPtT6xdftb2W6026sLYXJtZy7xmQISCpHBPFAWRs7R6UbR6Vwup6V4g3Xd2HljUlncpdOfMQ4xGEXO0jnkVU0zSPEN9apPH9pt0d28gTXbhrYbupB5bI9aAsj0U7QQDgZ6e9G0elcPa+Htcm1myu9Tj3JZyptH2stkAEF8dPTg13NAWQm0elG0elLRQFkJtHpRtHpS0UBZCbR6UbR6UtFAWQYAooooGFFFFAHH/Fn/AJJF4n/7B03/AKCa+DK+8/iz/wAki8T/APYOm/8AQTXwZQB+kdFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHH/Fn/AJJF4n/7B03/AKCa+DK+8/iz/wAki8T/APYOm/8AQTXwZQB+kdFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHH/Fn/AJJF4n/7B03/AKCa+DK+8/iz/wAki8T/APYOm/8AQTXwZQB+kdFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHH/Fn/AJJF4n/7B03/AKCa+DK+8/iz/wAki8T/APYOm/8AQTXwZQB+kdFfFf8Aw0T8Sf8AoNQ/+AUX/wATR/w0T8Sf+g1D/wCAcX/xNAH2pRXxX/w0T8Sf+g1D/wCAcX/xNH/DRPxJ/wCg1D/4Bxf/ABNAH2pRXxX/AMNE/En/AKDUP/gHF/8AE0f8NE/En/oNQ/8AgHF/8TQB9qUV8V/8NE/En/oNQ/8AgHF/8TR/w0T8Sf8AoNQ/+AcX/wATQB9qUV8V/wDDRPxJ/wCg1D/4Bxf/ABNH/DRPxJ/6DUP/AIBxf/E0AfalFfFf/DRPxJ/6DUP/AIBxf/E0f8NE/En/AKDUP/gHF/8AE0AfalFfFf8Aw0T8Sf8AoNQ/+AcX/wATR/w0T8Sf+g1D/wCAcX/xNAH2pRXxX/w0T8Sf+g1D/wCAcX/xNH/DRPxJ/wCg1D/4Bxf/ABNAH2pRXxX/AMNE/En/AKDUP/gHF/8AE0f8NE/En/oNQ/8AgHF/8TQB9qUV8V/8NE/En/oNQ/8AgHF/8TR/w0T8Sf8AoNQ/+AcX/wATQB9qUV8V/wDDRPxJ/wCg1D/4Bxf/ABNH/DRPxJ/6DUP/AIBxf/E0AfalFfFf/DRPxJ/6DUP/AIBxf/E0f8NE/En/AKDUP/gHF/8AE0AfalFfFf8Aw0T8Sf8AoNQ/+AcX/wATR/w0T8Sf+g1D/wCAcX/xNAH1J8Wf+SReJ/8AsHTf+gmvgyvQtY+OXj3XtGu9K1PVopbS7iaKZBaxruUjBGQuRXntABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAf//Z)![A picture containing graphical user interface

Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD4RXhpZgAATU0AKgAAAAgABAE7AAIAAAAPAAAISodpAAQAAAABAAAIWpydAAEAAAAeAAAQ0uocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAERhdmlkIEF0a2luc29uAAAABZADAAIAAAAUAAAQqJAEAAIAAAAUAAAQvJKRAAIAAAADNjkAAJKSAAIAAAADNjkAAOocAAcAAAgMAAAInAAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAADIwMjI6MTE6MTQgMjA6Mzg6NDEAMjAyMjoxMToxNCAyMDozODo0MQAAAEQAYQB2AGkAZAAgAEEAdABrAGkAbgBzAG8AbgAAAP/hCyFodHRwOi8vbnMuYWRvYmUuY29tL3hhcC8xLjAvADw/eHBhY2tldCBiZWdpbj0n77u/JyBpZD0nVzVNME1wQ2VoaUh6cmVTek5UY3prYzlkJz8+DQo8eDp4bXBtZXRhIHhtbG5zOng9ImFkb2JlOm5zOm1ldGEvIj48cmRmOlJERiB4bWxuczpyZGY9Imh0dHA6Ly93d3cudzMub3JnLzE5OTkvMDIvMjItcmRmLXN5bnRheC1ucyMiPjxyZGY6RGVzY3JpcHRpb24gcmRmOmFib3V0PSJ1dWlkOmZhZjViZGQ1LWJhM2QtMTFkYS1hZDMxLWQzM2Q3NTE4MmYxYiIgeG1sbnM6ZGM9Imh0dHA6Ly9wdXJsLm9yZy9kYy9lbGVtZW50cy8xLjEvIi8+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczp4bXA9Imh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8iPjx4bXA6Q3JlYXRlRGF0ZT4yMDIyLTExLTE0VDIwOjM4OjQxLjY5NDwveG1wOkNyZWF0ZURhdGU+PC9yZGY6RGVzY3JpcHRpb24+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iPjxkYzpjcmVhdG9yPjxyZGY6U2VxIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpsaT5EYXZpZCBBdGtpbnNvbjwvcmRmOmxpPjwvcmRmOlNlcT4NCgkJCTwvZGM6Y3JlYXRvcj48L3JkZjpEZXNjcmlwdGlvbj48L3JkZjpSREY+PC94OnhtcG1ldGE+DQogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgIDw/eHBhY2tldCBlbmQ9J3cnPz7/2wBDAAcFBQYFBAcGBQYIBwcIChELCgkJChUPEAwRGBUaGRgVGBcbHichGx0lHRcYIi4iJSgpKywrGiAvMy8qMicqKyr/2wBDAQcICAoJChQLCxQqHBgcKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKir/wAARCAGjANEDASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6/9oADAMBAAIRAxEAPwD2xtQuF8OxSrcJFKseVVcAsQDx0xjj8cVJp2pXc+vGGWTMRVvlwB0VT06jqee9bxijIwUUj0xShEDlgoDHvigDivife39ro9lBZTG2hurgR3FwDjYuPXsOv5Vh/DnUZW8VSWum3M9xpU1sZSkzbmhcHHJ7ZwfwNem3Vpb31s9vdwpNE4wyOMg1Bpmjado8TR6ZaRWysct5a4zQBdooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigDlPiFFrjeHTN4fuHieElpkj+86ex9qy/hxNrUOg3Gp+Jb5hYkbohcHlQOrZPb2rvyMjB6VHJbwywGGSJGiIxsI4x9KAOU0PxUPEmvJJZS7bJHliWI9ZMKDvP58V5J8UPCv/AAsD4j3GkSSLFdKfKtJyv3Ds3BSepUn8s19A2+nWdq++2tYom9UQCj+zrM3f2o2sX2jOfN2Dd+dAHw7/AMKY+If/AEKl9+S/40V910UAFFcx8Qdb1Xw/4TmvdEszcz7gjMP+WCHrJjviqvhLUNTisbK0Gh3BtZBua+kvkl3buS5x1yT29aAOxooooAKK4v4mXk9ppWkrbyXirPqsMUq2T7ZXQhsqpyOuPWuZ0zxxqWlaPiET3IvdVktbEXwaWWBETLiQL8xOQcD3oA9aorzsePtcureJbTTLe3uorKW8uluw6gqj7QEHB5xnnoKgXxvqPm3d5ZWPnTvYWU3l72ZU83duITPOPQcmgD0uivNh4w1y91fw41vPZC3uZp0ukiV/n2KT0bBU8dD0NR2nxP1O6gurxdKVrQQTSQsIpAEMecB2PBzjt0oA9NorN8PXV/faHbXeqpBHcToJNkBJVVIBA571pUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAjKrqVcBlYYII61W07TbXSbJbTT4RDApJVF6Lk5OPzqbzk9T+VHnJ6n8qAJKKj85PU/lR5yep/KgCG+0201E2xvIhIbWdbiHP8ADIucH9azbrwdol4tws1kv+kXAuXKsVIlxjeCOhx6Vsecnqfyo85PU/lQBxvif4ex6va2ltpht7ZIInh3SozOFY5OCGGfocitWLwRoq2CW81t5rC3hgaXcQzCIfIcjoRz0rd85PU/lR5yep/KgDHTwdoccNpHHZBRZzGeFgx3Bz1JPU575psXgnQYXuTHZAC5VlkTeduG+9gZwM+1bXnJ6n8qPOT1P5UALBDHbW8cEK7Y41CIvoAMAU+o/OT1P5UecnqfyoAkopnnJ7/lR5ye/wCVAD6KZ5ye/wCVHnJ7/lQA+imecnv+VHnJ7/lQA+imecnv+VHnJ7/lQA+imecnv+VHnJ7/AJUAPopnnJ7/AJUecnv+VAD6KZ5ye/5Uecnv+VAD6KZ5ye/5UUAPwKMCiigAwKMCiigAwKMCiigAwKMCiigAwKQ7VGWwB6mmTzJbwtLK21VGSTXMT6jcapOVAZYM8KO/1qXJIaVzdn1S1hU7GErj+FOayL3Wr9ZlMMSRxdyRk1LaWCxjOBjvU0tupypGVNUtQIoPEsap/psRU9mQZBps3jHT4m/1crD1ArOv4VMixjHoKxtShVYzHABkD5nNJ6AdbbeMtDuHEZvFhkP8MvFbcUkc0YeF1dD0ZTkGvANQ+ziQ7mVjnqam0jxRe6LcIbK7fy1PMTHKkfSo5yuU98xRiub8NeNNP8QRiPesN0OsbHr9K6SrTuRsGKMUUUwDFGKKKADFGKKKADFGKKKADFFFFABRRRQAUUUUAFFFFABRRUVzOttbSTPwqKSaAOd8Q3puboWcZ+RDlz6n0ryb4kfFXVPh3rlna2OlRTWkse9pZM/Mc8gV6BNcF5mmP3pGLYrH8W6RpXiLQ3ttZiR0Aypbqp9Qe1c6kr3Zry6WIvBfxg0bxlCiwyfZL7HzW0jYP4HvXfw3iyrya+LfEHhK40HVDPoNw88aNlDGfnT8q9n+FfjDWtW8PtHrqSLJAwRJnGDIK0c+VXEo3dj0jWb5Ybjcr8AcmvGPiN8XYrNH0zQmEtx0eUHKp/ia0/ijc65caI8ehh3Z22yeWfmC15Po3hiKwdb/AMRjO05EGc5PvSUlLVjcbaI2fBsuu6ik99qssslrIMJ5nc+orptMuHh1RbZzujl4ANVIfF9telYIkSJEG1VUYwKjvJSs0NxCfmVgQRUSeo0tDo75pNOkWe0do3RuccYNezfDbxXJ4j0Zo7tt1zbYDN/eHY14/eyC+smk7lAT+Vdr8C7OTbqV42fLJWNfc9aIN8wpbHr9FFFdBkFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABWL4quDBo5A/5aMFNbVZuv2H9oaPLEoy6jcn1FTO7i7FR3OIZgArH+Fa4b4g313PpjW1gWDk/Ns64rrbuV0hIOQ2MY75rBSxknnxP/GcEd64VPU6nE830HwzezXiT3atCi4JViTvrtrrVrPR4UVl6HhF610kmiQ2NkTbDnrzzmvJ/HdpOJN7k+YDnCV1cvPuYXtsdVdfEHS7hVtmtfJZ+N5IrndZ8PzX1q15ZTOXYZIC/LiuDtFkuL5ZJw4QHBJr13T9Uih0OJVcsNu32NU4qOpPNfQ8ijsZ4dQw2Q6HDYFdTatmAI/JqXVpDPOXIwvdsYzWfHKqthWArmlK7NYrQ6uymb7DKoBJZNqj1PQV9A+BNBXw74Rs7TbtmZBJMfVjya8m+Edlb614i23KB47OISKp/ibPBr3zp0raktLmc30CiiitzMKKKKACiiigAooooAKKKKACiiigAooooAKKKKACikyPWjcPWgV0efeNI47C+83gbzvX61yS6sDlidoB6jvXp/i3w0PEenCOOTyp4zlG9favI9U8N6vpty6S2cxVP4lUsp/EV59SnKMrrY7ISTjqa0muwJalTKc7cn2rzDXtVjuXlkVDI+7G72rS1E3f7xdrJn1GKq6dCAhJVCT94MM1tCrZamcodjik+2zsQICB0HFbtlezw2UavuiMT4YA8nPet4RQIzPMAD2AFc9fSrNdMVXagP51cqia0JUO5a1Py5QPMlLnGeTWO8S7fkz9asDzbqQRxRPM7HAVFJJr0rwN8HtQ1SaO68QxtaWX3vKPEj+2OwrBRlJmnMkdD8BdDngsrzVp1ISbEcZI+8B1Nex1XsLC20yxis7KJYoIl2oijoKsV2Rjyqxzyd3cKKKKoQUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUGiigDxvxn4n8UeH/Ek1sNSxE/7yIIgwFJ4FINX8eDw8mtpeeZaEFjgAkAHqRiqnxZbPjED0t1/ma7zwsqN8KYxJjb9mkzn8a8eKlOtOPM9D7yrOlh8Bh63sotyaT0MzwN8R5dXvk0zWVQTv/qplGAx9CK9DlkjjQtMyqvcsQBXzb4eLjxLYGHO4XCkY+tb/iDX7nxR4wFncXLw2QuPJVEOABnGfrTo41qn7+rvYjMMgp1MX+4fLHlu/L0R7MINI1BuI7S4I642tUMnh3QowZJNNtFHdjGBXiWvWlz4J8WNHp11Kojw8bk9QfWtbx5rmqanY6Xcuzx2VxAGKocBn75rX64kpc0dUcH+r7lOn7KreE1dPr32PSn0DwldMS1np7leuCvFcb4x07TrLSTqHhe00x7aF9lx+5Vzu/GuW8QQ6bcx6XH4YR3uHhAnSIEnd7+9dxqmjDRPhDJbSRCOYoHlH+0TQ6sqkZK1rLdBTy+jhJ0qkpczlK3K102d/wBDlPC1/wCItWuZItDjsIWiXczGBEAH1xXSaVrni+z8XWVprDLcWtySu6FQU+uQK4fwo+tD7eughSTAfO3ddvt716Z8ObhLTwXA+pSrFumZY2lOM5PQZrDCylUtdv8AQ9PN6VLCqbjTg1okkveV1udrLdQwD9/Kkf8AvsBTZb22h2+bPGm7pucDNeAeN/tEPi68gku5Z1V8qXbOAecU7xhpt3ptxZNd30l009usgLfw+wraWOa5vd2OCjw5Cfs71vjV9j36W6ggQPNNGinoWcAGpI5FlQPGwZT0KnINeAarp183gvTtWutQlmSRjEkLHhAOld38HrqebRbyGWRnSKYbATnbkCtaeKc6vs3GxxYvJY0MI8TGpzWdnp8j0aiiiu4+eCiiigAooooAKKKKACiiigAooooAKQ52nHWlooA8V8VeFvFmveIJ7yXTVxnYnluMbR06mp49M8fL4eXRIrMR2uCuQyhiD2zmvY8UYri+px5nLmep9Cs+rezhTdOLUdtNjznwP8N30e7Go6yyPcKP3cSHIT3J9awPFfw61a21yS+0SL7RBJJ5gVWAZDnPevZaMVTwdJwUF0MqeeYyGIliG021Zp7WPHT4O8TeMNXhudehWzijRUZjjJA9vWqPj+xa18RWWm6dMZhDEqxW4/gP8ua9wxXLeJfAmn+IrtLxpJLa6QYE0R5NY1MH7jUNWztweef7RGVdKMIppJLRXPMLnX/F2ixJNPbiyUnaH+zquT+Vb0uq+JfGngUQQWK3DvKUllUhcge1bp+GKXbp/a2tXt7Eh4jduK7PT9NtdLsY7SxiWKGMYVRSp4eq21NtRZpi80wcYwlQpxdRO97NL8dTx7w5oHjPw1eyXNnpKuZE2MsjAgj86v3Ok+MPEWsWMWrWC2tlDIH2xEBVxz69a9cwKMCtY4OKjyqTscVTPatSo6rpx5mrXs7nhXiDwl4m1TxFcXI0p8O+FwwxgcDnNXvGPh3xLq13Z40tmWC3WPMbA5OOa9nwKMCpeBg01d6mkeIcRGUGoR9xWW545qHhrxNN4D0/Tf7Ny0ErMyq43Adq6D4W6NrGjR3sep2Rt4pCGQsRknp0Feh4FGMVpHCxjUVRN3Ry1s4rVsNLDOKSbv8AjcKKKK6zxQooooAKKKKACiiigAooooAKKKKACiiigAoorn/FutXWi2VvJZ7d0swRiYy+BgngAj0qZyUI8zNaNKVaoqcd2dBRXF23j9NsKXVlI0hhEkrxfdGckfnj14pR47LMW+xuojUlogwYscAjn8ay+sUu51/2bir/AAnZ0VxT+P8AypGd7HdAY1MeyTLFixGOnTitTTvFX9opdyR2EqJaxB2LMASSM7cfhTVam3ZMzngcRCPNKOh0NFcS/wAQY7qP/iX20gZXXJfoQen507R/Gl3d6lHFe2qrFMURTG2djNnr69KX1im3ZM0eXYmMXKUbWO0orjJ/F91Y6pcC5WGS1jneFY0BEg2jOc5p7ePolXH2B/MU/vF8wYUZAyD360e3p9WT/Z+JtdRudhRXEWPjG+MMlzcwCWNpvKijRQmTk4+bJ9PSrdh4un1S8tYrayMIkkXzN53fIVJzx05FCr03sOeX4iF7rY6yiiitzgCiiigAooooAKKKKACiiigAooooAKKKr3l4lpHluWPRfWgcYuTsixRXPSancyNkPtHotM+33X/PZvzrPnR1rBz7nSVFNbQ3Gzz4kk2NuXcM4PrWB9vuv+ezfnR9vuv+ezfnRzoawk1szUfQ9NYgiyhVwpRXVACoPXH51V07wppWn20kK2yzCQksZQGJyMVV+33X/PZvzo+33X/PZvzqPcvexsqWIUXFT0NRtC0t4wjWFuVCbADGPu9cVYgsLS2R0t7eONXADBVAyBWH9vuv+ezfnR9vuv8Ans351XNFdCHh6zVnL8zXGkaeowLOAdOkY7dKIdI0+Agw2cCEMGBVAOfWsj7fdf8APZvzo+33X/PZvzo5o9g+r1v5vzNf+yNP+0Sz/Y4PNmBEj7BlvrVC+8JaVexwRmAQxQtuCRAKCc5qv9vuv+ezfnR9vuv+ezfnSfI9Gio0sRB3jM04tB0qGQvFp9urk5LCMZJ9amj0yxhkV4rWFGTG0qgGMVjfb7r/AJ7N+dH2+6/57N+dO8V0JeHrPeX5nSUVzf2+6/57N+dH2+6/57N+dPnRH1Ofc6SiubGoXQP+tatCy1XzHEdxgE8BqammRPCzir7mpRRRVnKFFFFABRRRQAUUUUAFc9qUhkv3z0X5RXQ1garAY7wvj5X5BqJ7HXhGvaalKiiisT1QpKWigDnp/FkEGt3NhLEqpaqGllaXnGM5C4yaD400fz0TzJdrK7NIYmATaATnPsRU914YtLq+ubl5rlPtShZo0lwr8Y6VFN4N0qfd5iSkPu3DzDggqFI/ICtv3Ryf7R0sWLLxPpeoTpDaTtJI6lgoibgDrnjj8ajbxdoy/wDL0WPcLGxI69gPY1Lpnh2y0mYS2ofzBGY8u2cjOf6Vn2/gqyWGYXDN501w0xeAlMbuNv0x/Ol+6KvXstrl2TxXo8cRka7ynPKoTnGM9vcVX1PxXFYahaW4gZknj815XDKEXOPQ889OKT/hCdJExdVmC9oxKdq9Og/AVf1DQbLU5RJdK5YR+X8rY4zn+Yo/dpi/fuPRMrr4u0h3VEnkZmYqoELckde1KninTjpEOoymaKGZyiBoiWJHXgfSmP4R0x0tkImC27l1USHkk55qb/hG7H7DFaHzTFDIzoC54LAgj6c0/wB15gvrGuwyTxZo8TlXuiD5YkH7tuRx0456imR+MNFlyBdMpHGHiZcnOMDI5Oar/wDCC6OZ/MxPnbtx5px2/wABU0ng7SZmDSRSEhi4+c8Etuz+dH7rzFfE36G8DkAilpAMAClrE6wpKWhVLsFUZJOAKAOls5DLZxs3UrzU9RW0Xk20cfdRzUtdKPBlbmdgooooJCiiigAooooAKjngS4jKSDI/lUlFA02ndGNJojbv3Ugx/tCm/wBizf30rboqORHQsVV7mJ/Ys399KP7Fm/vpW3RRyIPrVUxP7Fm/vpR/Ys399K26KORB9aqmJ/Ys3/PRKP7Fm/56JW3RRyIPrVUxP7Fm/vpR/Ys399K26KORB9aqmJ/Ys399KP7Fm/vpW3RRyIPrVUxP7Fm/vpR/Ys399K26KORB9aqmJ/Ys399KP7Fm/vpW3RRyIPrVUxBosueZExV+002O1O4ne/qe1XKKaikTKvUmrNhRRRVGAUUUUAFFFFABRRRQAVVvr1bSMd3boKtVzupOXv5M/wAJwKmTsjow9NVJ2ew2S/uZGyZWHspxTPtU/wDz2k/76qKisbs9VQiuhL9qn/57Sf8AfVH2qf8A57Sf99Vl63qX9kaLdX/l+b9njL7M4z+NZEfjK3ihh/tCF0klG4eQrMqrkKCSwBHJ9KpRnJXRnKdKDtI6v7VP/wA9pP8Avqj7VP8A89pP++q5KHx7pTCEXKzW8sxYBGUHADFcn6kVdbxVp0emW9/N58VvcEiNnjxnAz+vam4TXQFUovZo6D7VP/z2k/76o+1T/wDPaT/vquaTxlpTSbCZ1ydqlosBjkAgfQnmp7XxNZX1xNBZrM7xRtIxMeFGMjk/UUck+wlUovZo3vtU/wDz2k/76o+1T/8APaT/AL6rlbfxtpzRRi7EkUzQmUqq7hx2HerFj4qtdQv1gt4J9jW5n81gMYB6Uck+wKpRfVHRfap/+e0n/fVH2qf/AJ7Sf99Vy/8Awm+ki2knb7QqoRndHgkHuOenFI/jjSkmKBbpwP41hyvGM857ZFHJU7B7Sh3R1P2qf/ntJ/31R9qn/wCe0n/fVYWleJtN1m9mtbGVnkhGTlcAjOOK2KlqSdmaR9nJXjYl+1T/APPaT/vqj7VP/wA9pP8AvqoqKV2VyR7Eou7gHImf860LLVWLiO55B4DelZVJQpNETowmrNHW0VBYuZLKJm67anroPFas7BRRRQIKKKKACiiigArB1a3Md0ZAPlfnPvW9TJYkmjKSLuBqZK6NqNT2crnK0ta0miAt+6lwPQimf2I//PZf++ay5JHpfWaXcxL6yg1GxltLtN8My7XXJGR9RWePDGmbcPFJJ8oXMkzscbg2Mk+oFdX/AGI//PZf++aP7Ef/AJ7L/wB81SU1sTKrQk7s5IeFNIWZZVtmDqWPErDOTkg88jPOKdf+HrW90200/AjtbeRHCY3ZCnIGT0rq/wCxH/57L/3zR/Yj/wDPZf8Avmn+8J9ph7NdzkpfCmjzSTO9qd0xycSMNpzklefl5HarFhoOn6YJBZwFPMTY5Llsjn1Pua6X+xH/AOey/wDfNH9iP/z2X/vml+8Y1Uw6d1Y5BPB+ipN5iWrK23bxK2OmOmevvViz8OabYOrW0DIyoyZ8xjkN1zzzXT/2I/8Az2X/AL5o/sR/+ey/980/3jEqmHW1jkZvCOizoEktDgKFG2VgQB2yD709fC+kqhQWvBBB+du+M9/YV1f9iP8A89l/75o/sR/+ey/980fvA9ph730Ob0/Q7DS5pJLGExGT7w3kj8ATgVo1p/2I/wDz2X/vmj+xH/57L/3zUuMm7suNejFWTMyitP8AsR/+ey/980f2I/8Az2X/AL5pckh/WKXczKVEaRwiDJJwBWkNEbPMwx/u1ftdPiteV+Zv7xpqD6kTxVNL3dSa3i8m3SP+6MVJRRWx5bd3cKKKKBBRRRQAUUUUAFFFU7++FogC8yN0HpSbsVGLm7IuUVzEl1PKcvK30zTPNk/56N+dR7RHYsHLqzqqK5XzZP8Ano350ebJ/wA9G/Oj2g/qb7nVUVyvmyf89G/OjzZP+ejfnR7QPqb7nVUVyvmyf89G/OjzZP8Ano350e0D6m+51VFcr5sn/PRvzo82T/no350e0D6m+51VFcr5sn/PRvzo82T/AJ6N+dHtA+pvudVRXK+bJ/z0b86PNk/56N+dHtA+pvudVRXK+bJ/z0b86PNk/wCejfnR7QPqb7nVUVyomlByJG/OtCy1V1kCXB3KeNx6imppkTwkoq6dzaooByOKKs4wooooAKKKKACiiigArnNRcvfyZ7HAro6w9WtjHceaB8r9fY1E9jrwjSqamcKWkpaxPVCkJwKWkoA5K08Yzsk893aEwrcG3jSGNgzNv2j5mIU/hSH4gWkZkeWyuFh2p5ZBXc7MSMYzxjBrbj8OaTFM0qWMYZn8w8nG7Oc4+tI/hvSJFw9hERtC9McA5H6mt+al2OTkxHSSK1p4qtr22u7iC0uvKtUDuzKBnIzgDOc4qqvjvTXAeOC5eHqZQo2qucA9ehPpW0NKtorSeC0jW3E64ZkAJ6Yzz7VWg8MaVDb2kTWkcv2VNsbOvPr/ADqU6fVFONeySaKaeMrKWeCGG2uZJJyoRVC/xZ9T7VFP4puLfWb+GS3H2SxALMsblm4z977oP1rVtvDulWc4mtrGKOQNvDAdDSzeH9LuLuS5ms0aaUYdsn5vqKL0+wctdrdGQnjqxJcSWd3EVBwH2fMRjI+970yDxmt9qVnFZRfurhkB8xcMuQc9D7Vsy+HdInUiWwhYEk8juev8qdb+H9KtXje3sYo2iIKED7uOn86rmpdETyYjrJGhS0UVgdYUUUUAFJS06ONppAiDJJoE3ZXZ0Vi5eyiLddtWKjgiEMCRj+EYqSuhHhSs5NoKKKKZIUUUUAFFFFABTZI1lQq4DKeoNOooDYzZNFiZsxuyD0xmmf2Gv/PY/wDfNapOKTcPUfnU8qN1iKq6mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mX/Ya/8APY/980f2Gv8Az2P/AHzWpuHqPzo3D1H50cqD6xV7mWNDTPMxx/u1dtrKG1H7sZbux61PuHrS0+VEyrVJK0mFFFFMyCiiigAooooAKKKKACqOo3/2VQkfMjfpV6uc1Fi2oS57HAqZOyOnDU1OevQhkmklbMjsx9zTMn1oorA9ayQZPrRk+tFc5rmtX1l4gsrCzC+XNE8jsYfMIwQP7y4HPvVRi5OyIqTjTV5HR5PrRk+tcbJ46kOwQaYQTKq5eUYKFyhPHQ5FTWXjlNQuIobbS7gmaby42ZtqsOecke3Sr9lO17GSxFFu1zrMn1oyfWuR1vxo1hNeWtvaHzYSUSVnBBfbuxt64x3pB48iWMCbT5BLHnzwJF+QDAyPXqOKPZT7A8RRTs2dfk+tGT61xz/EGCOIs+nT7mI8pQ4PmZYjsOOlX77xYlnb2kosJnFxEZWUsFMag4OQfrS9lNdBqvSd7M6LJ9aMn1rk38d2+1PJspJHc4VfMA53baqTePZna3+zacUZnBZJJV+dMHoexyKpUaj6EvFUUr3O3yfWjJ9a44/EO2LSeVp1zIsaA7hwNxxxnGB1611lvI01tHJJH5TOoJTdnb7ZqJQlHc1hUp1PhdyXJ9aMn1ooqDSyAMQeCRV+y1N4nCTMWQ9z2qhSU02iZ04zVmjrQcjI6UVXsGLWMRbrtxViug8OSs2gooooEFFFFABRRRQAViavask3nqPlbr7GtukZQ6lWAIPUGk1dGtKo6cuY5Oit6TR7d2ypZPYGo/7Eh/56P+lZcjPRWKpmLVS80uwv5EkvbOC4eP7jSxhiv0zXS/2JD/z0f9KP7Eh/56P+lHLJA8TSe5zJ0nTimw2NuVxjHljHXP8APmlh0rT7eXzILKCN9+/csYB3ev1rpf7Eh/56P+lH9iQ/89H/AEp8shfWKP8ASOTTQNMW+ubt7WOWa5OZGkUN2xge1TPpGmyOrPYW7Mrb1JiHB9a6b+xIf+ej/pR/YkP/AD0f9KLTF7eh/SOZfSNNkjMclhbshGCpiGCM5/nT30+zkREe1hZUXYoKD5V9B7V0f9iQ/wDPR/0o/sSH/no/6Ucsh/WKP9I5ddF0xJmlXT7ZZGOWcRDJNJJomlzLtl062dfRoga6n+xIf+ej/pR/YkP/AD0f9KLT7i9vQ7fgcwdH01pDI1hbFyoQt5QzgdBV0YAwOBW1/YkP/PR/0o/sSH/no/6UuWT3GsTRWxi0Vtf2JD/z0f8ASj+xIf8Ano/6UcjH9apGLUkMTTyiNBkmtcaLCDzI5/KrkFrFbLiJcep7mhQfUieLgl7u4+GMRQpGvRRin0UVseZuFFFFABRRRQAUUUUAFFFZ+p35twIov9Yw5PoKTdkXCDnLlReZ1X7zAfU0nnR/89F/76Fcuzu7ZdiT6k0nPrUe0O1YPvI6nzo/+ei/99Cjzo/+ei/99CuW59aOfWl7Qf1JfzHU+dH/AM9F/wC+hR50f/PRf++hXK596Mn1o9oH1JfzHVedH/z0X/voUedH/wA9F/76Fctz60mT60e0D6kv5jqvOj/56L/30KPOj/56L/30K5bn1o59aPaB9SX8x1PnR/8APRf++hR50f8Az0X/AL6Fctz60c+tHtA+pL+Y6nzo/wDnov8A30KPOj/56L/30K5bn1o59aPaB9SX8x1PnR/89F/76FHnR/8APRf++hXLc+tHPrR7QPqS/mOpE0Z6Op/Gn1yfPrVyy1GS3kCyMWjPUHtTUyJ4RpXi7nQUUgIZQR0NLWhwhRRRQAUUUUAFFFFABXN6gSdQlz64rpKxtXtGEnnoMqfvexqJ7HXhZJT16mZRSUVieqLSHgUUUAcNY+LNRAuJZyly32o28UBaNAP3m0H5ct+YpreP7mIvJJYxMjLH5UYlOQzEg5OOnFdotjaJIZFtYFdjksIlyT9cUNZWrjD20LAjBBjB49K39pT6xOT2NbpMw7DxTJeWd/dy2S28NpGGO6XJLFc44HA96itvGaT6ANSe1VALkW7jzflGf4s4966QQQqjKsSBW4YBRg/WmizthAYRbwiI9YxGNp/Cp5odi+Srb4jkv+FgKGBaxHlmEyKVmyzEZ4AA9u+KbF49uZrfzE0pPliMjbrjjAIAx8vPWuvFnbKwZbeEEDaCIxwPShbO2RdqW8KrjGBGAKfPT/lI9nXv8f4HIy+P/IaJJrGMSmUxyIs5OOcZHy8/pSDx9OmHn0pRDgNlJyzYIOONvXiuvaztXYF7aFiDkExg4P5Uv2a34xDEMdPkFHPT/lD2Vf8An/AyvDOvt4gsZLhrX7PtfA+fcGGM+grbqOKGKBSsEaRqTkhFAGfwp9ZSab0OiCkopSd2LRSUUixaSipYIHuJhHGM56n0oE2krs39PYtYxE/3as0yKMRRKi9FGKfXSeFJ3k2gooooJCiiigAooooAKCARg8iiigClJpVtI2dpX/dNM/sa29X/ADq488Uf+skVfqab9rt/+e0f/fQqbI2VSrbRsq/2Nber/nR/Y1t6v+dWvtdv/wA9o/8AvoUfa7f/AJ7R/wDfQotEftK3dlX+xrb1f86P7GtvV/zq19rt/wDntH/30KPtdv8A89o/++hRaIe0rd2Vf7GtvV/zo/sa29X/ADq19rt/+e0f/fQo+12//PaP/voUWiHtK3dlX+xrb1f86P7GtvV/zq19rt/+e0f/AH0KPtdv/wA9o/8AvoUWiHtK3dlX+xrb1f8AOj+xrb1f86tfa7f/AJ7R/wDfQo+12/8Az2j/AO+hRaIe0rd2Vf7GtvV/zo/sa29X/OrX2u3/AOe0f/fQo+12/wDz2j/76FFoh7St3ZV/sa29X/Oj+xrb1f8AOrX2u3/57R/99Cj7Xb/89o/++hRaIe0rd2Vho9sD/Gfxq3DbxW67YkCikF1ATgTRk/7wqUHI4ppLoRKdR/EwooopmYUUUUAFFFFABRRRQAVm6pftD+5hOHIyT6VpVzeoEnUJc+tRN2R1YaCnPXoVySxyxyfeiiisT1goopD0oAKWvP7HxPqSm5Y3KXVx9rMCQSyrhF8zaDtVQwwPUmkk8c6rB5k0ltbsrJHsj5AQliGYkkccfqK29jLocn1qmt7noNFcxp3iW9u7HULy5tYYYrSMFVVixLFc9uMfSqieNLh/Cp1PyLdJBciAl3Pl4OPmyD70vZyvY0eIgld+p2VFcKvj65NxbxfYocyRksNxHPOCCf4ePSlh8dXkq2eLW3zO5RsFjz2AHf6jNP2MyPrVI7miuFTx5qEjSr/ZIQRqWLuW24B2k/TP6Uq+NtSeN3jsrUxxR7nl3Ptb59oYf7Pej2Mw+tU/6R3NFYnhO+udR0MXF5KssjSuNy/dIDcY9q26yaadjohJTipIKKKKRQVasr6S2kAJJjPVaq0lNNomUVJWZ1isGUMvIIyKWq2nkmwiz/dqzXQeFJWbQUUUUCCiiigAooooAKyNXs2L/aIxkY+YDt71r0daTV0aU6jpy5kclRXRyadbSNkx4P8AsnFM/sm1/uN/30az5Geh9ch2OforoP7Jtf7jf99Gj+ybX+43/fRo5GH1yn5nPbFBztGfpRtU9QPyrof7Jtf7jf8AfRo/sm1/uN/30aORi+tU+xz+B6fpUFxZ293GsdxCsio4dQR0I6Gun/sm1/uN/wB9Gj+ybX+43/fRo5GH1qk+hz21fQflRsXjgcdOOldD/ZNr/cb/AL6NH9k2v9xv++jRyMPrVPsc9tHoPyo2rjGB+VdD/ZNr/cb/AL6NH9k2v9xv++jRyMf1un2OfAAGAMfQUV0H9k2v9xv++jR/ZNr/AHG/76NHIw+t0zn6K6D+ybX+43/fRo/sm1/uN/30aORh9cp+Zz9TW1s91MEQcdz6VtjSrUH7hP1Y1ZjiSJdsahR6AUKHcieMVvdQsaCONUXoowKdRRWp5wUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQB//2Q==)![Graphical user interface

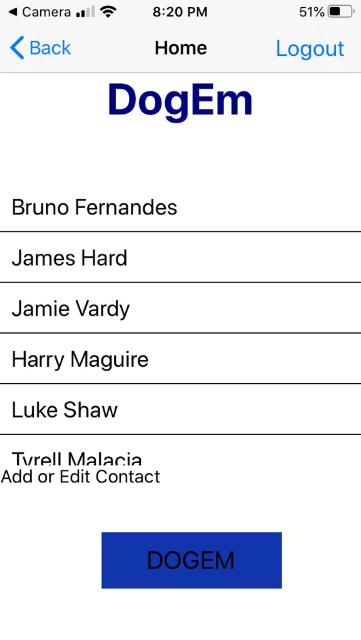
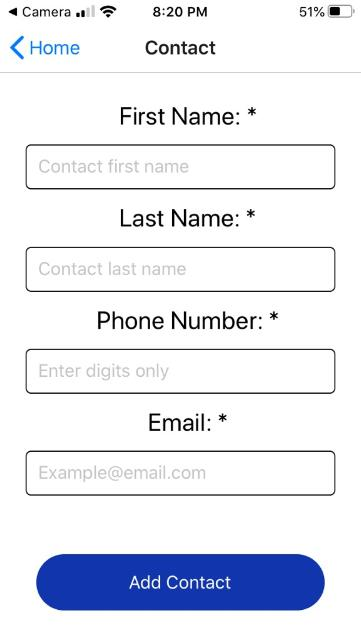
Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD4RXhpZgAATU0AKgAAAAgABAE7AAIAAAAPAAAISodpAAQAAAABAAAIWpydAAEAAAAeAAAQ0uocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAERhdmlkIEF0a2luc29uAAAABZADAAIAAAAUAAAQqJAEAAIAAAAUAAAQvJKRAAIAAAADNzcAAJKSAAIAAAADNzcAAOocAAcAAAgMAAAInAAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAADIwMjI6MTE6MTQgMjA6MzY6MjkAMjAyMjoxMToxNCAyMDozNjoyOQAAAEQAYQB2AGkAZAAgAEEAdABrAGkAbgBzAG8AbgAAAP/hCyFodHRwOi8vbnMuYWRvYmUuY29tL3hhcC8xLjAvADw/eHBhY2tldCBiZWdpbj0n77u/JyBpZD0nVzVNME1wQ2VoaUh6cmVTek5UY3prYzlkJz8+DQo8eDp4bXBtZXRhIHhtbG5zOng9ImFkb2JlOm5zOm1ldGEvIj48cmRmOlJERiB4bWxuczpyZGY9Imh0dHA6Ly93d3cudzMub3JnLzE5OTkvMDIvMjItcmRmLXN5bnRheC1ucyMiPjxyZGY6RGVzY3JpcHRpb24gcmRmOmFib3V0PSJ1dWlkOmZhZjViZGQ1LWJhM2QtMTFkYS1hZDMxLWQzM2Q3NTE4MmYxYiIgeG1sbnM6ZGM9Imh0dHA6Ly9wdXJsLm9yZy9kYy9lbGVtZW50cy8xLjEvIi8+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczp4bXA9Imh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8iPjx4bXA6Q3JlYXRlRGF0ZT4yMDIyLTExLTE0VDIwOjM2OjI5Ljc3MzwveG1wOkNyZWF0ZURhdGU+PC9yZGY6RGVzY3JpcHRpb24+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iPjxkYzpjcmVhdG9yPjxyZGY6U2VxIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpsaT5EYXZpZCBBdGtpbnNvbjwvcmRmOmxpPjwvcmRmOlNlcT4NCgkJCTwvZGM6Y3JlYXRvcj48L3JkZjpEZXNjcmlwdGlvbj48L3JkZjpSREY+PC94OnhtcG1ldGE+DQogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgIDw/eHBhY2tldCBlbmQ9J3cnPz7/2wBDAAcFBQYFBAcGBQYIBwcIChELCgkJChUPEAwRGBUaGRgVGBcbHichGx0lHRcYIi4iJSgpKywrGiAvMy8qMicqKyr/2wBDAQcICAoJChQLCxQqHBgcKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKir/wAARCAGxANADASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6/9oADAMBAAIRAxEAPwCL4V/BTT9Nex1Lx5p4vby6VJYdPlP7uFWBwZF6M3H3TwM4wT096fVYdI/0ZbJYLaOBnijjG3AGONoGBycYFaM2mW092tzIreYuOjEA46HHtmo30a0lkZpVeQspXDOSBkAE/XigCawuhe2qyjAbJDKM/KR25rz/AF7xrrUniS+sNDaC3isPlLTJkzSdkHucHH0r0O1tY7OExw5wWLMWOSxPcmuX8Q/DzT9d1T7eLme0mZ1eTyTw5HQ/X3oA0/COvN4k8N2+oyReVIxKSKOm4HBxW3VPSdLttG0yGxsU2QxDAzySe5PvVygAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACvMvFnxJ1jQfEj6fFpkaxoRsMmSZR6jFem1QvtE0/Ury1ur22SWa1bdEzDoaAIk1tLfw5Hqusp9hzEHkjZslSf4fr7Vzniix0fxV4H1O48T6FbXgtVl2Ryj502jja45U+4Irb8Q+FLLxMYhqMs/lxfdjjk2rn1x61ZOg2zaJc6ZI8jw3IYSMzfMd3XmgDTopssixRPI/3UUsfoK4Kx8a6zLb6Vrd1bWg0XVrpbeKNN3nRByQjk5wc45GO9AHf0Vxr/ABF0+6tp1sIbqOZra5ktZZocRyvCDuAOecYpNE+Iun3Ohi41dbiznh05L2YzQ7RKhABaPnkbjjHuKAOzormYvHWnSWl1I1pfxz2rIslo8H735/uHGcYPPOexqKP4iaPMlh9nivJpL6SWKOKOHLI0ZAcNzxjPWgDq6K4G5+IgufEGn2+lRTJZTJdM9xNB8kvlISChzyMj2zVlPiVpdvZwG8S6mf7LDc3E0Fv+7jWTox54GR05oA7WiuYn8f6TDqxsfLu5AtwlqbhIsxCVsYXdnryK6egAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAEdQ6MjDKsMEeorjLLwBLavYWcuryTaNptz9ptbIxAFWBJUM/Uhc8D6V2mRRkUAcYvw9Qadp9r9vbFkt6A2wfN9oDA/luqqnw0NxYyW2r6tJdBdNXTrVkiCGFAysG923Iv5V3uRRkUAcLd/D271EXN1f6z52ozywMZfIAiKQ52oU7gliTU3h/4erol5ZXD3/nm1lu5CoiCBvPxkYHTGK7TIoyKAOEtfhxPBPaJJrLSWdglxHaw+SAUWYEHLd8Zpw+GsY0i9sf7QbF3p1vYl/LHyiLPzfjmu5yKMigDyy+8La0PFSWWnR3H9mPqcV9K0ip5YKhdxDZ3c7emOvevU6MijIoAKKMijIoAKKMijIoAKKMijIoAKKMijIoAKKMijIoAKKMijIoAKKMijIoAKKMijIoAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAj8lPSjyU9KkooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooqnqWox6fb72+Zzwq+ppN2DctlgoyxAHqao3GrQQttjzK2M4Tp+dYJkuNVfdcMw9FHAFaMFkEjwRggcUk7jasV4/E1wJm861VU3YADc4rQXxFpxTLylDjlSvNUZrRD85XlaxJbRZpXJO1RyxqrAdH/wlukbwrXBT3ZSBWja6hZ3q5tLmKYf7Dg15TrEQKnZ8kfTJ71zAlNlcCa1naJ1OQyuazcrDsfQ1FeWeG/iTNbuINak+0RZwJQPmX6+temWl5b39sk9pKssbjIZTmrUkxNWJ6KKKYgooooAKKKKACiiigAooooAKKKKACiiigAooooAbI6xRs7nCqMk1xd3dG/vJLqQExoDsT2FdB4hu1h0/wAnPzzHaB7VysU4SXaO36VhOV5WNIrS55xpn7RmmR69Npur6VJZQpKYxNu3FcHHzCvWtM8R2mqW0dxYXCXEEgyro2RXifxT+Gulaq8mqWDpaXr/ADOR92Q+4/rXlPh3xP4k+HusqkLSGDfh4DlkkHtVp3+EVu59o3E6m2ZgcZFcbqGtQafFNJdyiOFMszE4FQW3iF9Q0SK6UNH5sYfY3UZHSvBvihf+ItW1/wDsyGGcWQwV2g7XPqTUqpzOxXJbUn8dfF661LUDaeHSViVtvmlc7voKvWEuoy6NBLqJZLxlyw6Z/CsLw3pmkeFmF1qwS5vsZVT92P8A+vXUf2vBqS74yOfSlK1tBJPqaOkyLqFi3mja8RwWFa/h7xff+F9aCJKzWxb542PBFcz4euza6xPA3McyHIqbXY9hjkT14NRexdj6hsbyO/sYbqA5jlQMv41PWD4Jt5bXwXpkU+d4gUnPvW9XStjAKKKKYBRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHHeJLrfraRf881/nWLJMIUZupPNbXiyzMGpR3qj5ZF2t7GuO1WaVk8uLljwQK4ZyakzpirxR5r471DUdQ1wRQiVrZRgLEO9T+FtBeNBPqgDk8oG5212dpoS3zgSkoU5ynFGs2UlnbmO2G4kdd3QVrB80bEy0ZS/4SXStPuTHdbnAGCicgVm6jrGma+pj00iNs4Ift715nr0txZ6nL+9chj0q34LBj1NJppMHdwG7itfZqxHON8SaLeWjEkrNF1Mm0D/AOvWfoglhY8/LnivVNf1C3ltTG2yQDj5gOnpiuClUCQlQEHYDispy6FRV9TUtnAuEkHWut0XST4k17SdPIJQvvmx2Ra4axuAkyliCqnJFfQnwo0W2j0BdaKA3N2CAf7iA9KiC5mVLRHfxRrDEkaDCoAoHtTqKK7DnCiiigAooooAKKKKACiiigAooooAKKKKACiijNAFbULVbuxlicA5XjPY15PqNzHZXUqMASThs9sV7D1rzDxt4Nvf7Ql1HT1aWBhuMacsp78elcuIg37yN6UktGZFvqaLIPmEYPbFU/EmuQrp7ssynPy5HU/SufuDPAreYjhsYG4YrA1CWa4ZEkH7us6c+Xcucb7GFr15CJD5MRfnPPes+1nmSaOZoXCqctxXafYoTbjNuhYDAbHNRXMVpa2bF8ZYcqBXR7WNjDkdyKK9knuHRpAMgMuVB4rIvolMxZnLH60wSlHMgOC3AHoKW2tLjU7xbexgknmY8JGhY1zyd2bLRFeOJzcokJJLHGMda+tfA2mSaR4L060nG2RYgWHoTzXn/wAOfhIbKaHWPEqfv0O6K16hT6t717ABjpXRSg46synK+gUUUVsZhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAGfrkuoQ6NcSaQsbXarmMS/dryOD4n+Kri7S1iW2eZ32KvldTnp1r2a7OLOY/7B/lXzroBz4ysj63a/8AoVebjJTjOCi7XPrcho0KtCvKrTUuVXV/R/5HZXfxF8WaFfLDrmnwDPO0pjcPYg16N4a8R2nibSVvLTKkHbJG3VG9K434xwxnSLGUgeYsxUHvjFcv4C19/D2l61dgblSNdqnoXJwKiNadGu6c3dG1TAUMflscVRpqE72stnrY9oubCyuV23VtDIv+2oNZU3gjw3c583Sbc7vRcfyryjTU1nxt/aNzPq0yy20fmLEGOD7AdqteAPGepWesJpt3NJcwTZVVkOSjdsGtI4yEpK8bJ7M4qvD9WnTm4VE5Q+Jduu56MfAHhhVwdNjA/wB4/wCNUbn4aeEFjMlxZhEzklpiB/OuBt9WuNe8bm28T300FtvZTH5hjVMdBVbSrO61nxkulRXVzd6dHcHP7wsoQHrS+tRfwx3div7BlBN1atrR5nppbyfVkfiSysfD2tvbP4b04x/ehZi7bl7H71b+hah4lttLF5oHhfT4bdxw8UQBb9c1D8Xoki1mxVBgLBtH0BrNi1rXRo2i23kNBZJOvlTISPMOehrmlUlCrKLk7LsexSwVCrgaVanShzPe9+l/PfQ9J8BeKb7xFa3a6pCsVxbSbW2rgfTFdduGcVzviFZbbwveyaVMlpdiPzGdFG48c/8A668h8MXmv6t4jEdpqcgupI2/eSsWAGK7ZYh0HGnL3mzwaWWxzCNXE02qcY9NdP8AgH0BuGcZo3DOO9fOun6nrdv4mRINQl+1tN5W5nJBJOOR6VJ4guNc0XxNKl1qc0l1EwbzFc4PfpWf9oLl5uVnX/qzP2ns/bK9r7M+h6Kp6RcPdaPaTy8vJErN9cVcr0U7q58pKLjJxfQKKKKZIUUUUAFFFFABRRRQAUUUUAZHie8vLLQLiXTrNrycjaI1688Z/CvDbHQvENjqUN7Ho90zxSCQAxHBIOa+icZpNorkrYZVpKTdrHtZfm0sDSlTjTUube9/uPFdei8YeNruCOfSJLeOM/KpG1QT3JNdXa/DgW3ga60wyK19cYdpOwYdB9K7/aKWiOFgm5Sd2yqudV5QhTpRUIxd0l3PBtFt/EfhDV5iNInmMkZiZQhKsD3yK2vB3gTUVvn1rVEFn5QaSGN+u7sT6AV69gU2aFJ4HikGVdSpHtWcMFGLV3e2x0VuIKtWMkoKLlZSa6pHztDe3seuXd1JZR6s5cq5eMyLnPXit3TPiJqWkXyQf2VZ20ZYB4o4DG2K6pfhxqOk30s3hvW2tElOSjpmp9M+GinVxqfiC/bULgMG27dqkj1rlhh8RF+7o7+R7VfM8sqxbqJSVtFre/psjkvHcGs+JNcWe20e7EMcYVDszu75q3pGra9pmh2+nTeEpLv7M26KSRT8p9cYr18IAAAMAUbRXX9Uam5qTuzwnnalQjh5UU4x21Z47Zx+IbqHXNYv7G5M08XkLCEI69wPQVm+BtO1fTPEa3baTdMqRODmMr29690wKMCj6mrp8z0K/t2Sp1KapK09OuitY+erTStaHiaK7Gk3ZYXIk2mIj+LPWrvjbTNY1DxXdTjSroK5AXEZYEY9RXvGBRgVH1GPLy83W5v/AKyVPbKt7JXStuzL8MmY+G7H7VA1vKIlDRv1GK1aKK9CKsrHzFSXPNyta4UUUUyAooooAKKKKACiiigAooooAKKKKACisfxFrn9g2kVy0QkiZ9jktjbkcfrgViWnxDtXhEl7btFnaoWM7jvwSw5x0xWUq1OMuWTOungsRVh7SEbo7OiuYsvGlvMWWaCUkO2TCm4IgPDNQ3jzTFQHyboseQgjGSuM7uvTApe2p2vcf1HE83LyM6eiuY/4TzSv3YZbhHdtuxkGQOOevTmprzxbb2OqzWc1rO2xUKNGAfMLdgKftqfcX1PEXtyM6Giua1PxTJDFZS6bbxzpcxvIfNcpgKMkdDzTU8d6W0Qd0nQFcg7Mhm4yo9+aPbU72bBYPENXUbnT0Vx7+ODPqkNrY2hCsD5hnVtykHBGFB/M1OPHlg4Iit7ktj5SyAKTg45z3waXt6fcp4HEpX5TqaKraddNe6dBcOnltIgYp6VZrVO6ucck4uzCiiimIKKKKACiiigAooooAKKKKACiiigAooooArX+n2up2pt76FZoiQ21vUHIqi3hjSGUj7Gq5lMxKsQd54JyK16Klxi90aRq1IK0ZNGSfDGkmZZPsi7lJP3jzn19abH4V0eIfLZL35LE9RjH0wa2KKXJDsV7et/M/vMk+GdJMiSC0CuhBDKxB4GPX2p974e03UJ2murffIyhSwYg4HTpWnRRyR7C9vV35n95l3PhzS7uG3hntQY7cYjUMRgdxx1qG58KaXOJSlskckibQwGQvuB0zxW1RQ6cHuhxxFaO0mc5p/grTbKFPM8yeZSSZS5UnJzjg9Kur4Z0lFAWzUBcYwTxjOP5mtaihU4LZDlia03eUmMiiWGFY4xhVGACc0+iirMNwooooAKKKKACiiigAooooAKKK5zW/HGj6HIYZ5jNOvWKEbiPqegppN7G1GhVry5KUW35HR0V54fi1YZ4064x/vij/hbdj/0Dp/8AvsVfs59j0f7FzD/n0/wPQ6K88/4W3Y/9A6f/AL7FH/C27H/oHT/99ij2c+wf2LmH/Pp/geh0V55/wtux/wCgdP8A99ij/hbdj/0Dp/8AvsUezn2D+xcw/wCfT/A9Dorzz/hbdj/0Dp/++xR/wtux/wCgdP8A99ij2c+wf2LmH/Pp/geh0V55/wALbsf+gdP/AN9ij/hbdj/0Dp/++xR7OfYP7FzD/n0/wPQ6K88/4W3Y/wDQOn/77FH/AAtux/6B0/8A32KPZz7B/YuYf8+n+B6HRXnn/C27H/oHT/8AfYo/4W3Y/wDQOn/77FHs59g/sXMP+fT/AAPQ6K88/wCFt2P/AEDp/wDvsUf8Lbsf+gdP/wB9ij2c+wf2LmH/AD6f4HodFeef8Lbsf+gdP/32KP8Ahbdj/wBA6f8A77FHs59g/sXMP+fT/A9Dorz+L4s6a0gEtjcIvqCDiut0fxBp2uwGTTrlZCPvIeGX6ipcJLc5sRl+Kw0earBpGnRRRUnCFFFFAHJ/EDxHJoWiBLRtt1dEojDqo7n/AD614qzM7l3JZmOSSeteg/FoP/aVgT9zymx9c8/0rz2u2kkon6Zw/h4UsFGcd5asKKKK0PfCiiigAooooAKKKiuSwtZTHndtOMdaAbsrktFY807SwxHNwAIv4FYHzPeoxJdrJ+9E5uDn7pYKBt9MY61HMYOuk7WNyisCB9TSaPf5zxSTKDnOU4/ka36cZXLp1PaX0sFFFFUaBRRRQAUUUUAFXNK1S60bUY7yykKSRnPXhh3B9qp0Ub6EzhGpFwkrpn0XpOox6tpNvfQ/cnQNj0PcfnVyuV+G4kHgi18zoXcr9Nxrqq8+Ss7H49iqcaVedOOybQUUUUjnOZ8c+HG8Q6JttwPtVud8X+16r+NeIzRSW8zRTo0ciHDKwwQa+lKxdZ8KaRrh331opl/56odrfmOtbU6nLoz6XKM7+pR9jVV4dLbo8Cor2A/CzRM/fuh/20H+FH/Cq9E/56XX/fwf4Vt7aB9J/rJgPP7jx+ivYP8AhVeif89Lr/v4P8KP+FV6J/z0uv8Av4P8KPbQD/WTAef3Hj9Fewf8Kr0T/npdf9/B/hR/wqvRP+el1/38H+FHtoB/rJgPP7jx+ivYP+FV6J/z0uv+/g/wo/4VXon/AD0uv+/g/wAKPbQD/WTAef3Hj9Fewf8ACq9E/wCel1/38H+FH/Cq9E/56XX/AH8H+FHtYB/rJgPP7jx+ivYP+FV6J/z0uv8Av4P8KP8AhVeif89Lr/v4P8KPbQD/AFkwHn9x4/RXsH/Cq9E/56XX/fwf4Uf8Kr0T/npdf9/B/hR7aAf6yYDz+48for2D/hVeif8APS6/7+D/AAo/4VXon/PS6/7+D/Cj20A/1kwHn9x4/RXsH/Cq9E/56XX/AH8H+FH/AAqvRP8Anpdf9/B/hR7aAf6yYDz+48frR0PRLvXtSjtLJCcn53xwi+pr1GP4XaEjgt9pkH91pMZ/IV1Gm6TY6Rb+Rp9skCdwo5P1PeplWVtDjxfE1FU2sOm5eeiRJp1jFpumwWduP3cKBF98VZoorlPhJScm5S3YUUUUEhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAjMFUknAHrXC698TrHT52t9Mh+2SKcGQnCA+3rT/ibrsmnaPFY2zlJLskOQeQg6/nmvIK6KdNNXZ9fkuS0sTT+sYjVPZfqzuz8WNWzxZ2gH0b/ABpP+Fsav/z6Wn/fLf41wtFbezj2Pp/7Gy//AJ9I7r/hbGr/APPpaf8AfLf40f8AC2NX/wCfS0/75b/GuFoo9nHsH9jZf/z6R3X/AAtjV/8An0tP++W/xo/4Wxq//Ppaf98t/jXC0Uezj2D+xsv/AOfSO6/4Wxq//Ppaf98t/jR/wtjV/wDn0tP++W/xrhab5qYJ3rgdTuHFHs4dg/sfL/8An0jvP+Fsav8A8+lp/wB8t/jR/wALY1f/AJ9LT/vlv8a4PzFJADLk9OetBkUHBZQeuM0ckOwv7Hy//n2jvP8AhbGr/wDPpaf98t/jR/wtjV/+fS0/75b/ABrgvOjxnzEx/vCnB1Y4VlJ9AaOSHYP7Hy//AJ9o7v8A4Wxq/wDz6Wn/AHy3+NH/AAtjV/8An0tP++W/xrhaKPZx7D/sbL/+fSO6/wCFsav/AM+lp/3y3+NH/C2NX/59LT/vlv8AGuFoo9nHsH9jZf8A8+kd1/wtjV/+fS0/75b/ABo/4Wxq/wDz6Wn/AHy3+NcLRR7OPYP7Gy//AJ9I72L4s6kHHnWNs69wu5T+ea7bw14207xGfJjzb3QGTDJ3+h714ZUtvcS2lzHcW7mOWNgysOoIqZUotaHHi+H8HWptUo8sujX6n0lRWb4e1Qa1oFpfcBpU+cDsw4P6itKuN6H5xUhKnNwlutAooooIPLPi1BIL6wnIPltGyA+4Of6153Xvvirw/H4j0V7RiElU74XP8Lf4V4bqWl3mkXjWuoQtFIp7jg+4PcV2UpJxsfo/D2Mp1cKqF/ej08u5UoozRmtj6UKKM0ZoAKKM0ZoARhlSB6Vzw0e6jUZQPyDiMKCPmyeDwTXRZo4qXG5jUoxqNOXQx2srgTlkt1bcVKs4X5AOo4PH4Uy302eK4iZ4A3I3lirDHP4g/pW3xRxS5Cfq8b3MV9LkRYNkA+SV2YRqh4PTrxUltp80epC4YKELMcAAEZ6ZPf6VrcUZFHINUIJ3CijNGas3CijNGaACijNGaACijNdB4W8J3niO+TCNHZqf3sxHGPQepqW0ldmFevTw9N1Krsken/DqCSDwTaeZkby7gH0LHFdRUVtbx2lrHbwKEjiUIqjsBUtcDd3c/IcRV9tWlU7tsKKKKRgFVrvT7S/j8u9top0/uyIG/nVmigabi7p2ZhHwX4dP/MIt/wAqP+EK8O/9Ai3/ACrdop3Z0fXMT/z8l97ML/hCvDv/AECLf8qP+EK8O/8AQIt/yrdoouw+uYn/AJ+S+9mF/wAIV4d/6BFv+VH/AAhXh3/oEW/5Vu0UXYfXMT/z8l97ML/hCvDv/QIt/wAqP+EK8O/9Ai3/ACrdoouw+uYn/n5L72YX/CFeHf8AoEW/5Uf8IV4d/wCgRb/lW7RRdh9cxP8Az8l97ML/AIQrw7/0CLf8qP8AhCvDv/QIt/yrdoouw+uYn/n5L72YX/CFeHf+gRb/AJUf8IV4d/6BFv8AlW7RRdh9cxP/AD8l97ML/hCvDv8A0CLf8qP+EK8O/wDQIt/yrdoouw+uYn/n5L72YX/CFeHf+gRb/lR/whXh3/oEW/5Vu0UXYfXMT/z8l97MWPwf4ficMmk22R0ymf51rxRJDGEiRUVeiqMAU+ii7MqlapU+OTfqwooopGYUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUm9f7w/Ojev94fnQAtFJvX+8Pzo3r/eH50ALRSb1/vD86N6/3h+dAC0Um9f7w/Ojev8AeH50ALRSb1/vD86N6/3h+dAC0Um9f7w/Ojev94fnQAtFJvX+8Pzo3r/eH50ALRSb1/vD86N6/wB4fnQAtFJvX+8Pzo3r/eH50ALRSb1/vD86N6/3h+dAC0Um9f7w/Ojev94fnQAtFJvX+8Pzo3r/AHh+dAC0Um9f7w/Ojev94fnQB8NmkoooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA//Z)**6.1.3. Chloe Chung’s Prototype**



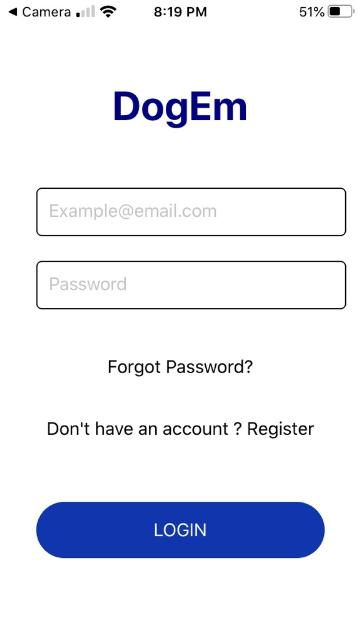
**6.1.4. Zane Atkinson’s Prototype**

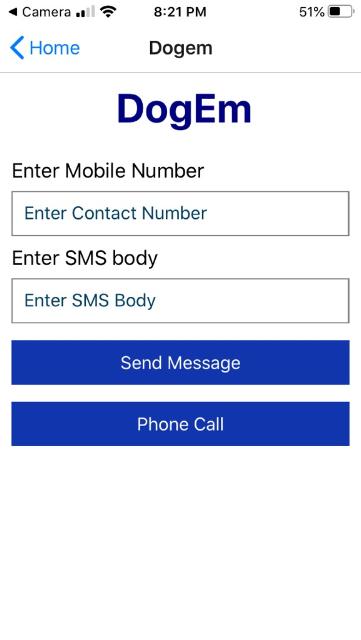
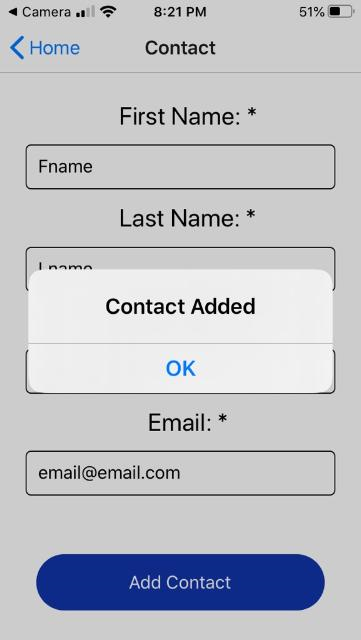


**6.2. Development Phase: The Final Prototype**

Graphical user interface, text

Description automatically generatedGraphical user interface, text, application

Description automatically generatedThe final prototype was a pre-version 1 of our app. We created this to create a foundation for future development efforts. The objective was to get a sense of what our app would look and behave like. The final prototype uses Deo’s UI with additional pages and features added by Chloe and Melissa.



**7. Version Control**

**7.1. How the DogEm Team Uses Version Control**

Version/Source Control is the process of tracking and maintaining changes made to code. It is an integral part of the software development process. In order to facilitate best practices in our source code management, the DogEm team has done the following:

**1. Storing, Managing, and Navigating Source Code with GitHub and GitHub Pages**:

Storing and managing with GitHub: We have a central location for all of our repositories at [DogEm Repositories (github.com)](https://github.com/dogemapp?tab=repositories). Changes made to any and all source code has been tracked in their respective GitHub repo(s).

Navigating with GitHub Pages:

We have created an organized directory for navigating all of our source code at [DogEm App source code directory.](https://dogemapp.github.io/sourcecodedir.html) This is a way to organize our various repos for greater ease of use for our users.

The specific category for DogEm’s versions is called “DogEm Versions” and it lists version 1 and version 2 of DogEm. It allows users to navigate to each version’s repo using a link.

**2. An explanation of the development process – Versioning:**

What is versioning: Software versioning is the practice of assigning unique version names (or in this case, numbers) to distinct software states.

Versioning scheme we employed:

DogEm v<major version>.<minor version>.<patch>

Ex. DogEm v1.0.0 is the very first version of our application.

Why we used versioning:

It is a systematic way of developing an app, because as functionality is added, the versioning keeps track of the app’s growth. In addition to that, it makes it easier to adapt to new requirements.

Versioning is useful for keeping track of new releases/changes for both developers AND the clients.

In the event of failure, developers can revert to an older version.

**3. How we will use versioning for the DogEm App:**

We will begin with a small initial version that encapsulates only our phase 1 requirements called DogEm v.1.0.0.

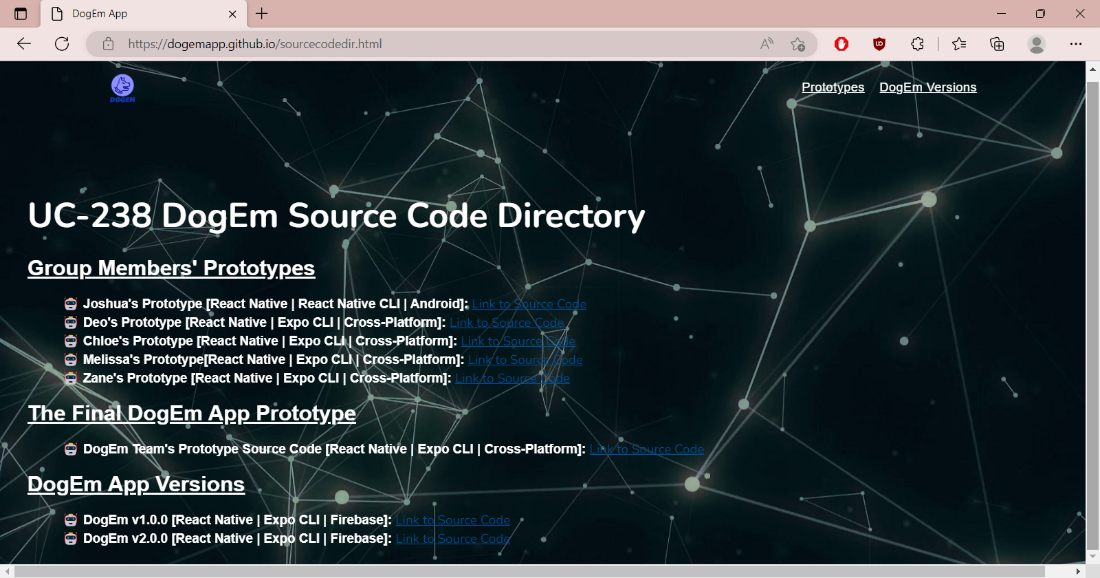
Phase 2 requirements will become our DogEm v.2.0.0 and beyond.

Any modifications to DogEm will be tracked in the version history and will abide by the versioning scheme detailed above. For example, if a patch is added to DogEm v.1.0.0, it will be versioned DogEm v.1.0.1, added to GitHub as a repo, and will be posted onto our source code directory under the category “DogEm App Versions” as DogEm v.1.0.1.

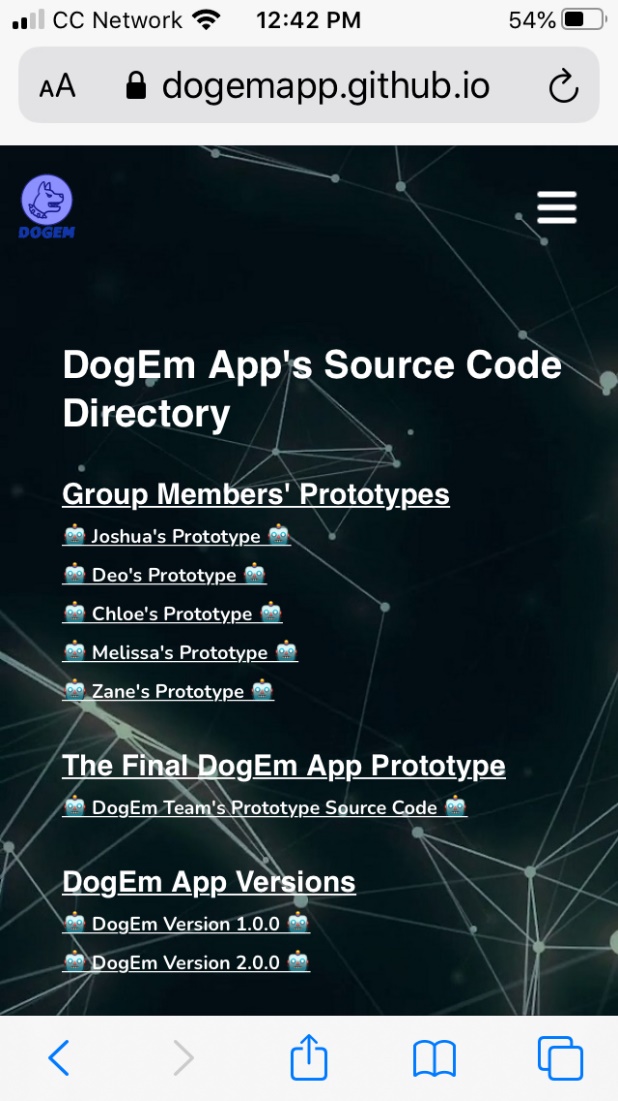
**7.2. The Source Code Directory**

The source code directory is a handy tool for navigating important repos in our GitHub. It was designed to resemble a table of contents. Its organized format ensures that instead of having to parse through various repos, people wanting to survey our code base can do so from a user-friendly landing page. A web version and a mobile version was created.

The web version can be found at [DogEm Source Code Directory (Web)](https://dogemapp.github.io/sourcecodedir.html)



The mobile version can be found in the [DogEm Source Code Directory (Mobile)](https://dogemapp.github.io/dogemmobilesrccodedir.html)



**8. Results**

**8.1. Discussion and Demo of the Final Product**

The following video is a complete overview of our app. The version being demonstrated is DogEm Version 1.

[Video-UC-238](https://www.youtube.com/watch?v=-e5LMo3oBuk&ab_channel=Chlo%C3%A9Chung)

[](https://www.youtube.com/watch?v=-e5LMo3oBuk&ab_channel=Chlo%C3%A9Chung)

We also have a brief walkthrough of version 2 showing sample ads:

[DogEm Version 2](https://www.youtube.com/watch?v=ftcKaXDysO8&list=PL7wQWqxrho3yiNOc7X8nbR46ML2VSAdQE&index=3&ab_channel=Chlo%C3%A9Chung)

[](https://www.youtube.com/watch?v=ftcKaXDysO8&list=PL7wQWqxrho3yiNOc7X8nbR46ML2VSAdQE&index=3&ab_channel=Chlo%C3%A9Chung)

**8.2. DogEm Calling vs. Conventional Mobile Calling**

**Why use the DogEm Calling Function in Favor of Your Device’s Calling Capability? – Timing and Keystroke Comparison**

Android and iOS Calling (15 phone calls)

The comparison below was made between Android/iOS default calling function and Dogem in order to generate repeated phone calls.

Initial Action: Enter contacts phone number. Click on the calling button. (10 seconds).

Subsequent actions:

Accessing the recent phone calls list in the dialer menu. (1-3 keystroke) \*15

Clicking on the most recent phone call to generate the phone call (1 keystroke) \*15

Repeat until the contact responds. The number of keystrokes will vary depending on how many times the user wishes to call the contact, which is 15 in this case.

Total Keystrokes required: Initial (10 seconds) +Accessing the recent phone calls list (1.5 keystrokes) \*15+Clicking on the most recent phone call to generate the phone call (1\*15) = 37.5 keystrokes

DogEm Calling (15 phone calls)

Initial Action: Enter contacts phone number. Click on the calling button in the Dogem menu (10 seconds).

Subsequent actions:

Clicking on the calling button when prompted by the device (1 keystroke) \*15

Total keystrokes required: Initial (10 seconds) + Clicking on the calling button when prompted (1 keystroke) \*15 = 15 keystrokes.

Using our app, the user only needs to click on the calling button when prompted by the user. The remaining actions will be generated by Dogem.

**8.3. DogEm Messaging vs. Conventional Mobile Messaging**

**Why use the DogEm Messenger Function in Favor of Your Device’s Messaging Capability? – Timing and Keystroke Comparison**

Android and iOS Messaging (240 MMS messages)

Initial Action: Enter contacts’ emails and phone numbers, as well as filling in the text field with their message. Then click on the send button. (5-30 seconds)

Subsequent actions:

Copy the message you wish to send contacts (single press, then another press for “select all”, then another press for “copy” x3 seconds)

For each message, the user must press into the text field (single press) then select paste (single press) then hit send (single press) (x3 seconds \* 239)

Total keystrokes required: Initial (5-30 seconds for the first message, keystroke number is unknown) + copy operation (3 \* 1) + repeated messaging (3 \* 239) = 725 to 747 seconds total.

DogEm Messaging (240 MMS messages)

Initial Action: Enter contacts’ emails and phone numbers, as well as filling in the text field with their message. Then click on the “Send Message” button. Then a new MMS opens with the fields filled and the user hits the send button. (5-30 seconds, additional 1 second for hitting the second send button)

Subsequent action: Repeatedly hit the send button. The sms body automatically re-fills, so the paste operation is not required. (1 second per messaging action)

Total keystrokes required: Initial (5-30 seconds + 1 for the first message, keystroke number is unknown) + (239 \* 1) = 245 to 270 seconds total.

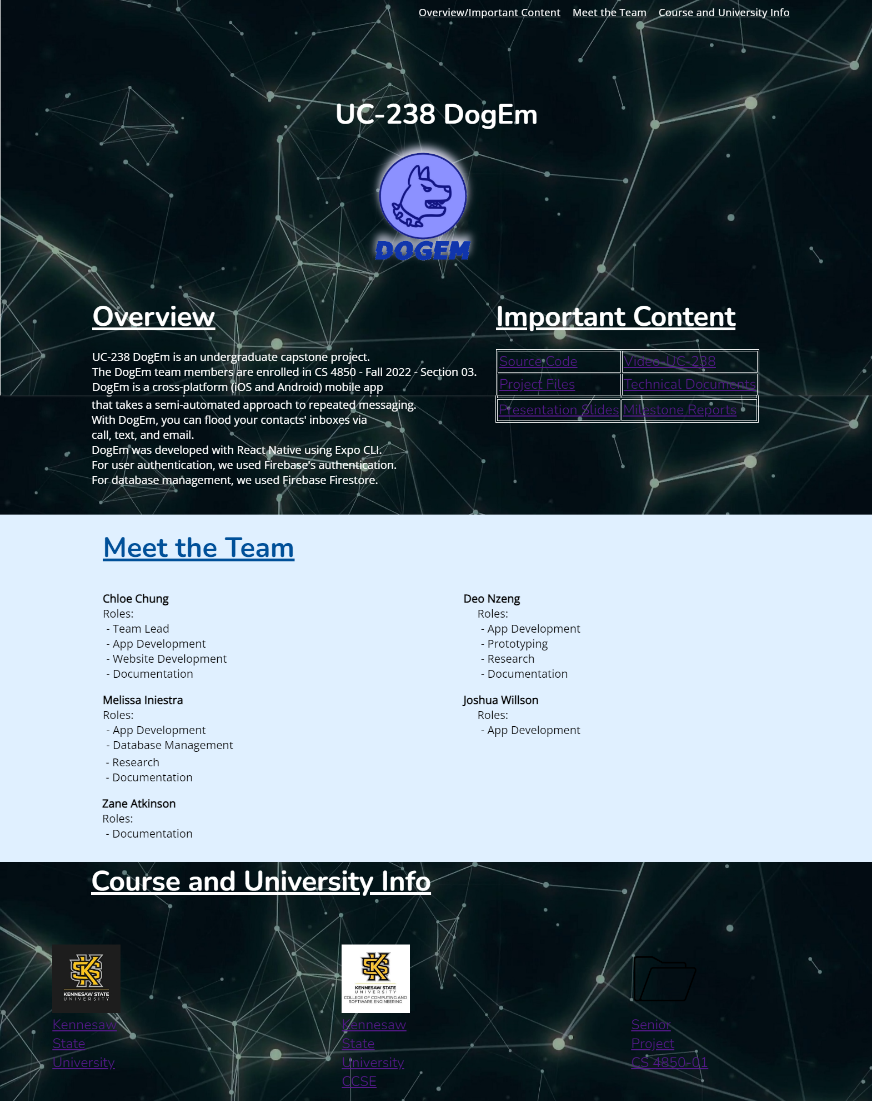
For the repeated messaging function, DogEm messenger requires 1/3 of the keystrokes and therefore roughly 1/3 of the time.

**8.4. DogEm Website (Homepage)**

TheDogEm website is a landing page for our project. The site is packed with important information including the name of our project, our app logo, an overview of our project, various content links, a “meet the team” section, and course and university information.

Chloe produced a web version of the site and a mobile version of the site. In addition to the web and mobile versions of the homepage, web and mobile versions of the source code directory were also created. The rationale behind this was to ensure that the site is attractive and easy to use on both web and mobile. Links and screenshots are below.

The web version of the DogEm website can be found at <https://dogemapp.github.io/>



The mobile version of the DogEm app can be accessed via this QR code:





**9. Conclusion**

After an arduous semester-long journey, the team successfully accomplished our various goals. We were tasked with creating a cross-platform mobile application capable of flooding a user-entered contacts list’s email inbox, default texting app’s inbox, and default calling app’s inbox. Not only did we make it so that our app sends a massive volume of messages and calls, we made it so that it does so quickly and reliably. Our testing indicated that DogEm is capable of sending messages in 1/3 of the time of standard messaging and it is capable of sending calls in ½ the time of standard calling. Our app has two versions that possess the same functionality: DogEm v1 and DogEm v2. DogEm v1 is our paid-tier which is an ad-free experience. DogEm v2 is our free-tier which accounts for monetization by including in-app ads in our UI. In addition to our two app versions, a series of other deliverables were produced for the project: a mobile website for our app, a mobile source code directory for our app, a web version of our website, a web version of our source code directory, four different member prototypes, a final prototype, demo videos, C-day materials, and this report.

**Appendices**

**Appendix 1 – Weekly Logs**

**A.1.1. Week 1: 8/16/22 - 8/22/22.**

**Week 1 Description:** Week 1 was the first week of the semester. The class was asked to organize into groups and select a project. We were assigned to the DogEm project which was where we had to make a mobile app that could email, call, and text someone until they responded. In addition to that, we were also instructed to decide how we were going to communicate and collaborate on deliverables. Melissa Iniestra created a GroupMe for day-to-day communication and a Microsoft Teams for collaboration on deliverables.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Planning Phase | Determine Group Members | Team |
|  | Project Selection | Team |
|  | Establish Group Communication and Collaboration | Melissa Iniestra |

**A.1.2. Week 2: 8/23/22 - 8/29/22.**

**Week 2 Description:** For week 2, we made a list of requirements and a Gantt chart that shows the hours we worked on each part of the project. It also had dates for each milestone and had the current status of each section of the app that showed whether it’s completed or not. We were able to use this document to assign parts to everyone.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Planning Phase | Requirements | Team |
|  | Initial Gantt Chart | Melissa |
|  | Gantt Chart | Chloe, Melissa, Zane, Deo |
|  | Initial mockup | Deo |
|  | Presentation mockup | Chloe |
|  | **Meeting:** Produce deliverables | Team |

A.1.3. Week 3: 8/30/22 - 9/05/22.

**Week 3 Description:** In week 3, we did mostly research on how to use react-native and expo in Visual Studio Code. Chloe created a GitHub repository for our source code. We looked into how to make a phone call in a mobile app. We decided to start working on our own prototypes and created our own repositories to upload our finished prototypes at a later date. The initial GitHub pages site was also created.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Planning Phase | GitHub setup | Chloe |
|  | Research calling function | Deo |
|  | Gather resources for setting up local development environments. | Melissa and Deo |
|  | Setup tech required on local PC’s | Team |
|  | Initial website creation | Chloe |
|  | **Meeting:** Progress Report | Team |

A.1.4. Week 4: 9/06/22 - 9/12/22.

**Week 4 Description:** We continued to do more research on react native and started to look more in depth on how to email and text through an app using react native. We each provided an update on our research and talked about what resources to use moving forward. The Project Selection deliverable was submitted.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Planning Phase | Brainstorming | Team |
|  | **Deliverable:** Project Selection | Chloe, Melissa, Deo, Zane |
|  | **Meeting:** Group Update | Chloe, Melissa, Deo, Zane |
|  | Work on prototypes | Team |

A.1.5. Week 5: 9/13/22 - 9/19/22.

**Week 5 Description:** Everyone shared their prototypes. All of us looked at each one and discussed which one we should use as the primary product. We began working on the Project Plan. For the project, we discussed our deadlines and reassured each other on what our roles were and talked about our test plan.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Planning Phase | **Meeting:** Share Prototypes | Chloe, Melissa, Deo, Zane |
|  | We delegated parts for the Project Plan deliverable. | Team |
|  | Review prototypes to decide which will be the final prototype (verdict is next week). | Chloe, Melissa, Deo, Zane |

**A.1.6. Week 6: 9/20/22 - 9/26/22**

**Week 6 Description:** We decided to try to combine some of our code from our prototypes into our final prototype. Deo’s prototype was the group’s favorite, Melissa’s prototype featured React Navigation, and Chloe’s had an initial version of the messaging functionality. Ultimately, this was when we decided what the app layout would look like page-by-page. We also determined what functionality we wanted to include, or not, and what the minimum viable product would look like.Each of us gave an update on what new resources we found. We laid out a plan on how we were going to do for the project plan deliverable and completed the deliverable together during a Teams meeting.

We also drafted a new requirements doc and test plan for our application to guide our development of the final prototype.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Initial Phase | UI prototype | Deo |
|  | Prototype with React Navigation | Melissa |
|  | Prototype introducing messaging using expo-sms | Chloe |
|  | Adding functionality to the UI prototype | Chloe, Melissa, and Deo |
|  | Editing the DogEm site | Chloe |
|  | Initial phase test plan | Team |
|  | Initial phase requirements doc | Team |
|  | Initial phase mockup | Chloe Chung |
|  | **Deliverable:** Project Plan | Team |
|  | **Meeting:** Progress report | Chloe, Deo, Melissa, Zane |

A.1.7. Week 7: 9/27/22 - 10/03/22.  
**Week 7 Description:** In week 7, we continued to work on the Milestone 1 readiness report. We discussed what we were going to present to the class. We also discussed how much functionality of the app should be completed by the time we present. We also decided on the content that will be on each page, for instance, what fields would the user be required to enter for registration, contacts page, and login page. We made a lot of adjustments to the app this week to make each page look consistent and user-friendly. Then we gave another update on how much progress we made and talked about new resources that we discovered that could be beneficial to use.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Initial Phase | General App Development | Chloe, Melissa, Deo |
|  | Research | Chloe, Melissa, Zane, Deo |
|  | **Meeting:** Progress Report | Chloe, Melissa, Zane, Deo |

A.1.8. Week 8: 10/04/22 - 10/10/22.

**Week 8 Description:** We talked more about the Milestone 1 readiness report. For the Milestone 1 readiness presentation we decided that we were going to share all our prototypes but instead we decided that we were just going to show one since it would’ve been time consuming to go through them all. Then we realized we also needed to talk about the proof of concept and know what future plans were going to be, so we started discussions about implementing a database into the app.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Initial Phase | App Development | Chloe, Deo, Melissa |
|  | Database research | Team |
|  | **Meeting:** Progress report | Team |

A.1.9. Week 9: 10/11/22 - 10/17/22.

**Week 9 Description:**  The team added additional pages and functionality from other prototypes to ensure that the final prototype aligned more closely with our requirements.

The team also had a meeting to gather materials for our M1 presentation. Chloe produced a series of demo videos for that as well.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Initial Phase | App Development | Chloe, Melissa, Deo |
|  | **Meeting:** Preparation for our M1 presentation. | Team |
|  | Demo videos | Chloe |

A.1.10. Week 10: 10/18/22 - 10/24/22.

**Week 10 Description**: In week 10, we started to get more involved with the database. We decided the most time-efficient implementation with some level of security through the use of authentication would be using Firebase. Firebase provides authentication through Firebase Auth and also provides a database through Firebase Firestore. The authentication was set up during this week to allow for registration. The chosen authentication was email and password.

The deliverables due this week were milestone 1 readiness (M1) and milestone 1 (M1).

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Development Phase | Developed User Authentication w/ Email and Password | Melissa Iniestra |
|  | DogEm Page: Add Contact feature | Chloe Chung |
|  | Development phase 1 mockup | Chloe Chung |
|  | **Deliverable:** M1 Readiness | Chloe, Melissa, Deo, Zane |
|  | **Deliverable:** M1 | Chloe, Melissa, Deo, Zane |
|  | **Meeting:** Progress Report and M1 completion | Chloe, Melissa, Deo, Zane |

A.1.11. Week 11: 10/25/22 - 10/31/22.

**Week 11 Description**: In week 11, we continued to develop the app. More specifically, we determined that we need to approach the app differently. The main changes we determined were needed involved changing the layout of the app. For instance, instead of having “contacts” saved through the database and connected to a user, we decided to have contacts be added during the DogEm Page and not involve the database or the contact page. The contacts will be temporarily stored for the duration of the log in of the user. The welcome page was subsequently used for another function other than the original intended use, showing a list of the contacts. During this week, we had not yet determined what the welcome page would be changed to.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Development Phase | Developed restrictions to user input in the registration page | Melissa Iniestra |
|  | Customized Error Messages to add security to the app | Melissa Iniestra |
|  | DogEm Page: Flexible Contacts List | Chloe Chung |
|  | Changed the layout of the App | Chloe Chung, Deo Intshakal A Nzeng |
|  | **Meeting:** Report on app changes | Team |

A.1.12. Week 12: 11/01/22 - 11/07/22.

**Week 12 Description**: This week was devoted to working on the app. Melissa connected DogEm to Firestore. For data management, she created a Firestore database, added the Firestore configuration to the app, and modified various pages to facilitate effective database CRUD operations. Deo made changes to the Calling function. He attached the contacts list to the calling function, created an interval for calling (every 10 seconds), and added error handling to the calling function.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Development Phase | Creating the Firestore Database | Melissa Iniestra |
|  | Connect DogEm to Firestore | Melissa Iniestra |
|  | Modifying pages to facilitate CRUD operations against Firestore | Melissa Iniestra |
|  | Calling function changes: attach the contacts list to calling | Deo Intshakal A Nzeng |
|  | Calling function changes: setting a 10 second interval between calls | Deo Intshakal A Nzeng |
|  | Calling function changes: error handling | Deo Intshakal A Nzeng |
|  | Messaging function changes: error handling | Chloe Chung |
|  | Messaging function changes: Contacts List Modifications | Chloe Chung |
|  | **Meeting:** Discuss App Changes | Team |

A.1.13. Week 13: 11/08/22 - 11/14/22.

**Week 13 Description**: The team finally finished our app this week. Joshua added a page named “Onboarding” that has an embedded YouTube video that shows a video walkthrough of our app and a play/pause button. Chloe used firebase auth to display the email of the current user on top of the onboarding page. Melissa reviewed the app and did some code cleanup by deleting an unnecessary button and adding some placeholder text to a field. Chloe made the Phase 2 (Version 2.0.0) of the app by adding a sample ad banner on top of every page to simulate the free-tier experience with ads. The group then held a meeting to discuss the finished product. Chloe made a mobile version of the site and source code directory for C-day preparation.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Development Phase | App finishing touches: Editing Components | Melissa Iniestra |
|  | Make mobile version of the website and source code directory. | Chloe |
|  | Onboarding page | Joshua Willson |
|  | Onboarding page welcome message | Chloe Chung |
|  | Development phase 2 mockup | Chloe Chung |
|  | Phase 2/Version 2.0.0 of DogEm | Chloe Chung |
|  | **Meeting:** Discuss versions 1.0.0 and 2.0.0 of the app. | Team |

A.1.14. Week 14: 11/15/22 - 11/21/22.

**Week 14 Description:** We continued to work on the M2 Readiness Report and the C-Day submission this week. Our M2 Readiness Report required us to do a presentation in class where we showed our final products (app, site, source code directory, and videos), we showed what we needed to revise, and we presented our C-day materials. After the presentation, Chloe made a screen recording of the final DogEm app product. Melissa wrote a script for describing the app and then she made a voiceover to complete the video. Also, after the presentation, Chloe edited the C-day poster and flash sheet to incorporate suggestions made by the professor and students during the M2 Readiness Presentation. After completing our deliverables, we submitted our video, poster, and flash sheet to the C-Day site. On the 19th, the team had a meeting in which we completed a required form for C-day together.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Development Phase | **Presentation:** Milestone 2 Readiness Report | Chloe, Melissa, Deo, Zane |
|  | **Meeting:** M2 and C-day Preparation | Team |
|  | **Deliverable:** Milestone 2 Readiness Report | Team |
|  | C-Day Video Script | Melissa Iniestra |
|  | C-Day Video Voiceover | Melissa Iniestra |
|  | C-Day Video Screen Recording | Chloe Chung |
|  | C-Day Final Poster | Chloe Chung |
|  | C-Day Final Flash Sheet | Chloe Chung |
|  | C-Day Submission | Team |
|  | C-Day Form | Chloe, Melissa, Deo, Zane |

A.1.15. Week 15: 11/22/22 - 11/28/22.

**Week 15 Description:** We worked on the Milestone 2 – Final Report Draft and discussed C-Day. The team collaborated on the final report draft via teams and we had a virtual meeting to discuss our plans for C-day. Chloe made modifications to the website.

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| Development Phase | **Meeting:** C-Day Discussion | Team |
|  | Edits to the website | Chloe Chung |
|  | **Deliverable:** Milestone 2 – Final Report Draft | Chloe, Melissa, Zane, Deo |
|  | **Deliverable:** Dept Eval Website | Chloe Chung |

A.1.16. Week 16: 11/29/22 - 12/01/22.

**Week 15 Description:** C-day

|  |  |  |
| --- | --- | --- |
| **Phase:** | **Activity:** | **Accomplished By:** |
| N/A | C-Day Event | Chloe, Zane, Joshua |
|  | **Deliverable:** Final Report | Chloe, Melissa, Zane, Deo |
|  | **Deliverable:** C-Day verification | Chloe Chung |

**Appendix 2 – Suggestions for the Future**

The DogEm team had three major features we wanted but were unable to implement.

**They are as follows:**

Stopping messaging or calling on contact response. This would have been a powerful feature that would have added tremendous value to the app. While the app was built to pester contacts, being able to have it stop upon receiving a response would make it more flexible.

Malicious user reporting. We chose not to deploy this app to the app store because we did not have proper mechanisms to ensure that users would not be able to use this app maliciously. Allowing for reporting of malicious users would make DogEm less susceptible to abuse.

User profile page: Due to limited time, the DogEm team was unable to produce an important page: user profile page. We wanted to implement a page that would allow users to see view and edit their information.

**References**

*Choosing the right database for your react native app*. Codemagic blog. (2021, January 31). Retrieved November 29, 2022, from https://blog.codemagic.io/choosing-the-right-database-for-react-native-app/

*Expo documentation*. Expo Documentation. (n.d.). Retrieved November 29, 2022, from https://docs.expo.dev/

Google. (n.d.). *Get started with Cloud Firestore  |  Firebase*. Google. Retrieved November 29, 2022, from https://firebase.google.com/docs/firestore/quickstart

Google. (n.d.). *Get started with Firebase Authentication on websites*. Google. Retrieved November 29, 2022, from https://firebase.google.com/docs/auth/web/start

Google. (n.d.). Google. Retrieved November 29, 2022, from https://firebase.google.com/

Microsoft. (2021, November 3). *Download visual studio code - MAC, linux, windows*. RSS. Retrieved November 29, 2022, from https://code.visualstudio.com/download

Node.js. (n.d.). *Download*. Node.js. Retrieved November 29, 2022, from https://nodejs.org/en/download/

React navigation RSS. (n.d.). Retrieved November 29, 2022, from https://reactnavigation.org/docs/getting-started

*Setting up the development environment · REACT NATIVE*. React Native RSS. (2022, September 5). Retrieved November 29, 2022, from https://reactnative.dev/docs/environment-setup

*Using Firebase*. Expo Documentation. (n.d.). Retrieved November 29, 2022, from https://docs.expo.dev/guides/using-firebase/