Full Stasis

Vincent Sanchez

CST-452 Capstone Project Final Architecture & Design

Grand Canyon University

Instructor: Professor Mark Reha

Revision: 2

Date: 4/17/23

**ABSTRACT**

This project is used is going to be used to color coordinated a person daily

schedule with certain requirements in mind. These colors will be used to help people that

have many different tasks in front of them get organized without letting their mental

health plummet in response. This will include having specific color events that are not

supposed to be moved or item that you do not have control over. For example, an

external event or responsibility that was planned by someone else or has an expected

attendance. This can be school, work, church, Jury duty, Items that are not able to be

moved and if missed with have a negative effect. There will also be a section for items

that should happen within a day this will be its own color. Using this part in combination should help those with busy schedules from getting

burned out or overwhelmed by all the tasks in front of them. All it takes is some planning

and determination and with this app it will be more assessable.

|  |
| --- |
| History and Signoff Sheet |

**Change Record**

|  |  |  |
| --- | --- | --- |
| **Date** | **Author** | **Revision Notes** |
| 11/7/22 | Vincent Sanchez | Initial draft created |
| 4/17/23 | Vincent Sanchez | Updated Diagrams |
|  |  |  |

|  |
| --- |
| **Overall Instructor Feedback/Comments** |

|  |
| --- |
| **Overall Instructor Feedback/Comments** |

**Integrated Instructor Feedback into Project Documentation**

Yes  No

**TABLE OF CONTENTS**

Design Overview 4

Detailed High-Level Solution Design 5

Detailed Technical Design 6

Appendix A – Technical Issue and Risk Log 7

Appendix B – References 8

Appendix C – External Resources 9

**Design Introduction**

The purpose of this document is to present the final design decision for Full Stasis. This will include the technologies and software that has been chosen for the development of the application through careful thought and consideration. This document will also contain diagrams that depict the flow of the application and the components pieces that are used to create the various screens for this mobile application. Each pieces of this document shall make the development of this application easier as each item to consider has been documented including the use of a framework, a compatible database and it’s modules along with the versions of them.

1. Use the template to list the project deliverables that are to be included external to this Design Specification (Data Dictionary, API Design, etc.).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Deliverable Acceptance Log | | | | | |
| ID | Deliverable Description | Comments | Evaluator (internal or external as applicable) | Status | Date of Decision |
| 1 | Project Proposal | This document holds the initial idea for the application Full stasis as well as the schedule for the Design and development | Internal | In progress | 11/7/22 |
| 2 | Milestone requirement | This document holds the requirement for this application and the technologies that will be used | Internal | In progress | 11/7/22 |
| 3 | Project User stories | This document holds the user stories that determine the functional and non-functional requirements for this application | Internal | In progress | 11/7/22 |

Detailed High-Level Solution Design

|  |  |  |
| --- | --- | --- |
| Proof of Concepts | |  |
| **Description** | **Rationale** | **Results** |
| 1.Prototype of application using firebase | This application was to prove that a app similar to the initial design is possible with all the technology list below | The Test was successful as the application was able to be use with firebase and each of the component where able to be navigated to |
| 2 – Design a page with a login button to test the authentication service | This was done to see if the included authentication service from firebase could be used for Full Stasis | The result of this experiment was that within the firebase console you can determine what 3rd party app you can use for a login like Google, Apple, or Facebook |

|  |
| --- |
| Hardware and Software Technologies |
| 1 -Visual Studio Code 2022(1.72.0) |
| 2 – ReactJS (18.2.0) |
| 3 – JSX (18.2.0) |
| 4 – Draw.IO (20.3) |
| 5 -MacBook Air 2020 (IOS 16) |
| 6 -iPhone 12(IOS 16) |
| 7 -Samsung Z Flip 4(Android 13) |
| 8 – React Native Firebase (Library V6) |
| 9 -Axios(1.1.3) |
| 10 -Firebase Realtime Database (31.1.0) |
| 11 -Google Logging (10.3.0) |
| 12 -Firebase Authentication (v21.1) |
| 13 -Firebase crashlynics(v18.3.2) |

**Logical Solution Design:**  
  
Graphical user interface, diagram

Description automatically generated

In the application the way that the logical diagram is laid out is you will have your host device, which is a mobile phone, this phone will interact with the presentation layer in which will show the component and the hold the state the is currently on the React Native application. As for the backend, Axios will handle the http client work as it will be used to retrieve the props from the front end and this data will be sent to the firebase database using key value pairs. This does not require a business or data layer for this information to be written or retrieved from the database as React Native can interact with directly with Firebase.

**Physical Solution Design:**Diagram

Description automatically generated

In the physical Diagram, we can see that the main portion is the cloud provider Firebase on the Sparks Tier. This tier allots us one virtual CPU core and 1 gigabyte of Ram. As for our other modules we are using Firebase’s Realtime database which provides us with 10GiB of storage on the Sparks tier. Another module that we are using is Firebase Authentication that is based on the state from React Native to determine if the application has a user logged in and allows a user to use 3rd party site to create login info such as use Google Account to login. For Logging on the application Google Logging was the chosen module as it can be integrated with Firebase very simply. Lastly, the module that is being used to monitor the application and determine if it has crashed is Firebase Crashlytics which will send a notification if the application is not responding correctly.

Detailed Technical Design

**General Technical Approach:**

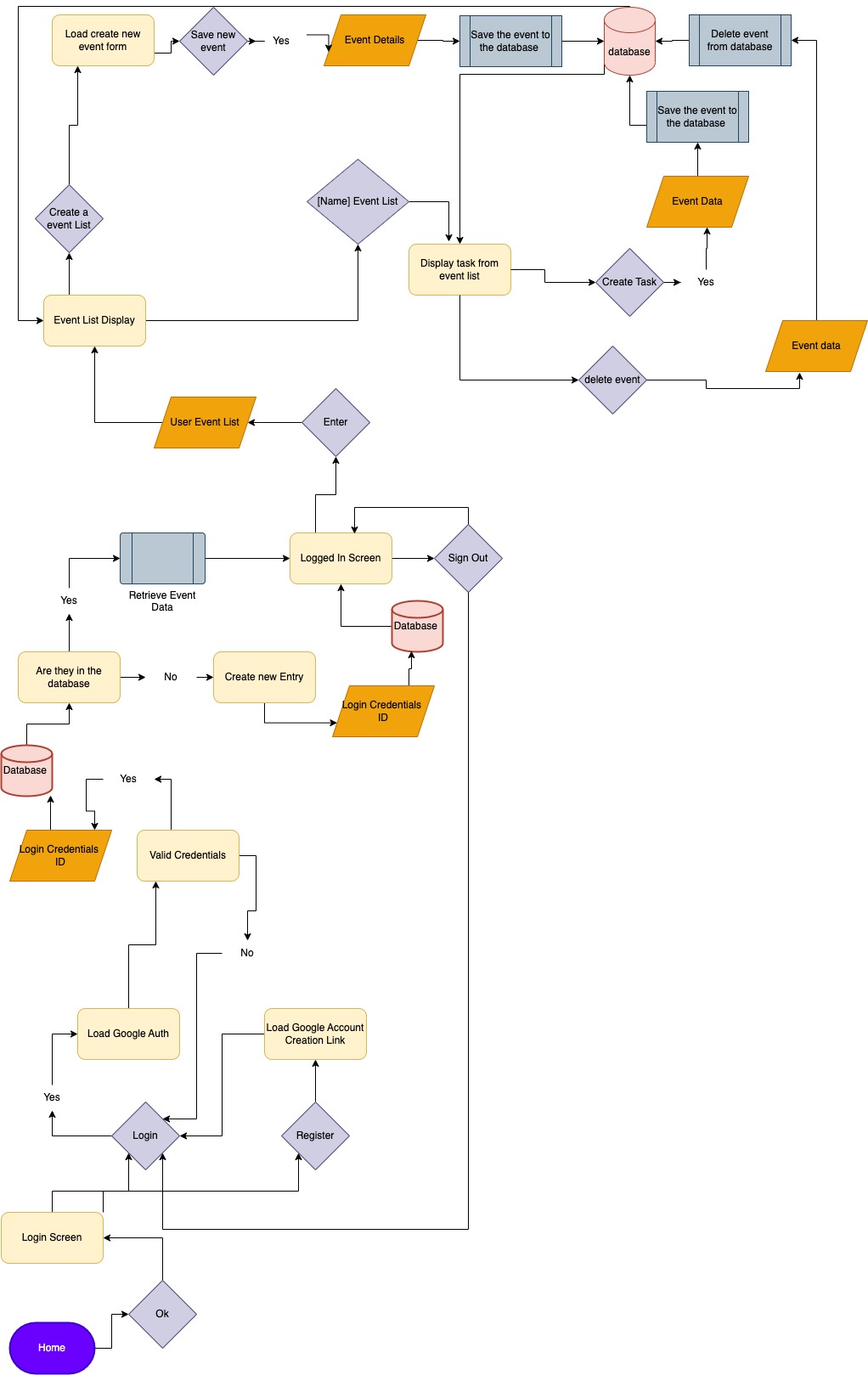
The approach for this application was contemplated in each stage of the documentation for this application. From the initial project proposal to the application requirements and now the final design. In each of these stages, the determination of whether React Native with a database using firebase was a workable solution for the creation or development of Full stasis was considered. After some proof of concept and familiarization of the framework was done it was determined that the chosen technologies should work for the intended use.

**Key Technical Design Decisions:**

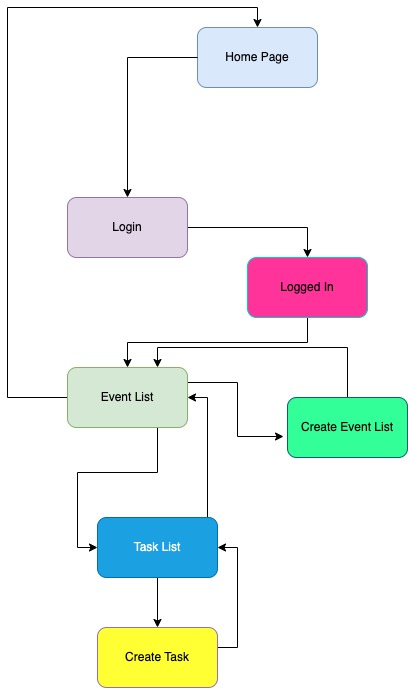
The framework that this application is going to for both the front end and the back end is React Native. The reason that this was chosen was it has compatibility be moved to work for both Android and IOS devices. This is favorable over other framework use Kotlin or swift is that those frameworks when using those languages create apps that are for specific platforms while React Native using JSX can run on both IOS and Android.

**Schema Design:**

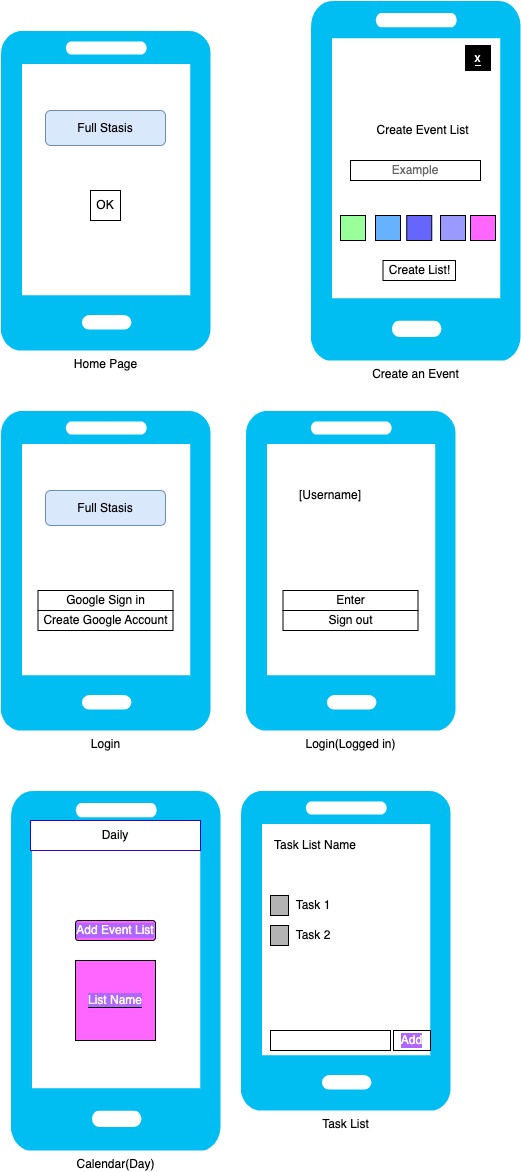
  
  
In the Shema Diagram above these are the key values that will be used to store data in the NoSQL database on Firebase for Full Stasis. The key values will allow the user to create events connected to their Google User ID that are saved to the database so that when you enter the application again it can retrieve their created event list for them to view or delete.

**Flow Charts/Process Flows:**  
  
The flow chart above shows the possible way that a person can go through the application and has a visual representation of how data can be read, written, and deleted from the database.

**Sitemap Diagram:**

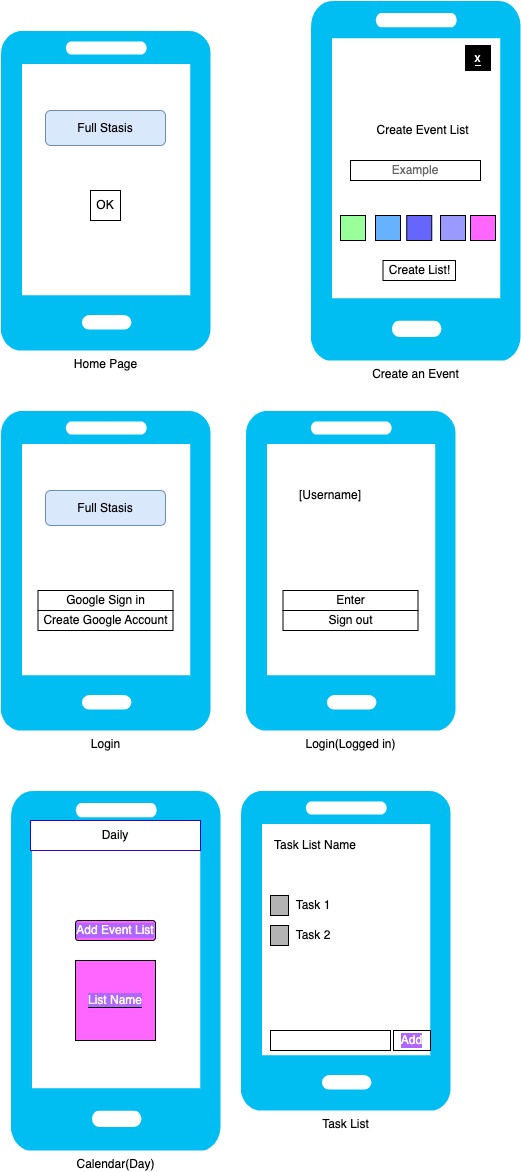


In this Sitemap Diagram, we are able to see all the path that are able to be taken in the application.

**User Interface Diagrams:**  


In the User Interface diagram above each of the different pages are pictured for how the mobile application can be used.

**Component Design:**



In the diagram above we can see all of the various components that make up the screens that were described in the wireframes.

**Service API Design:**

N/A

This application does not need an API design diagram as an API is not being used to retrieve, write, or update the data from within the Firebase Realtime database.

**NFR’s (Security Design, etc.):**

The non-Functional requirement that will be supported by this application is the load time for each scene of the mobile should take less than 5 seconds to load the content on the page.

The Event List must show a loading animation while retrieving information from database. Lastly, the application must maintain its state even if it is interrupted.

**Operational Support Design:**

The logging in this application is going to be done by using firebase’s module for logging information. This logging module will be used to report the status of the application. This includes if the application is not operating as, it or if it is generating exceptions. The monitoring of this logged info can be viewed in the firebase console for the specification application.

**Other Documentation:**

The specific devices that will be supported by this application are the following models:  
For Android: Android that are running Android 12 or newer (These are most android device made in the last two year)  
  
This application will not be placed in either the Apple “App Store” or Google’s “Play Store”.

Appendix A – Technical Issue and Risk Log

1. Use the template to identify and monitor project issues and risks.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Issues and Risk Log | | | | | | | | |
| **Issue or Risk** | **Description** | **Project Impact** | **Action Plan/Resolution** | **Owner** | **Importance** | **Date Entered** | **Date to Review** | **Date Resolved** |
| I/R | What is the issue or risk? | How will this impact scope, schedule, and cost? | How do you intend to deal with this issue? | Who manages this issue? |  |  |  |  |
| R | Unable to understand React Native Interface | The impact if this happens means  a new framework will  need to be picked, which will cause a setback in schedule, and an increase of cost due to more working hour needed to get acquainted with a new framework | Have the main developer begin and complete a course in a Learning Software based on the current intended framework | Vincent Sanchez | High | 11/9/22 | 11/12/22 | 11/14/22 |
| R | Firebase will not work for this intended use | The backend of the application will to be change out for an alternative like AWS or Azure | Have the main developer conduct research of the use of the React Native framework and firebase | Vincent Sanchez | High | 10/3/22 | 11/1/22 | 11/14/22 |

Appendix B – References

Google. (n.d.). *Firebase Android SDK Release notes*. Google. Retrieved November 20, 2022, from https://firebase.google.com/support/release-notes/android#latest\_sdk\_versions

Appendix C – External Resources

|  |  |
| --- | --- |
| **GIT URL:** | [*https://github.com/ironson22/Full-Stasis.git*](https://github.com/ironson22/Full-Stasis.git) |
| **Hosting URL:** | *N/A* |