
Bill Vivino

Software Developer

ABOUT

I currently work as a contract/freelance software developer for various large and small-business clients, billing a full-time income in software development business receipts over the past year. I received the Robert H. Goddard Award for my work with NASA and achieved "top-rated" status in software development for my work on Upwork. I have developed many full-stack iOS, Android, and Web applications in native and cross-platform languages and frameworks on top of robust backends. I have also developed a Unity/C#-based app for the NASA visitor center and games for various clients. My apps are published in the Apple App Store, Google Play Store, and on the web. I have a B.S. in computer science from Rutgers University in 2004. I achieved a 4.0 GPA in my subject area.

EXPERIENCE

ScrubSync Surgical Platform – *Full-Stack Engineer*

<https://www.scrubsync.io/>

TypeScript | React | Node.js (Express/TSX) | PostgreSQL (Drizzle ORM) | FHIR (STU3 & R4) | WebSockets

MAR 2025 – PRESENT

- **Epic FHIR Integration (Scheduling & Chart Read/Write)** – Wired the app to Epic's public sandbox and hospital-tenant endpoints; stood-up `$find` / `$book` appointment flows, user-level OAuth scopes, and resilient token refresh logic that surfaces Epic surgeries inside ScrubSync's calendar.
- **Domain-Driven Postgres Schema** – Authored 30-plus tables and enums (e.g., `surgicalCases`, `caseInstruments`, `vendorImplantInventory`, `sterilizationRecords`) with Drizzle; enforced FK relationships and Zod insert schemas to guarantee typed inserts across the API and React Query cache.
- **Mass Merge & Refactor Rescue** – Resolved >40 merge-time runtime errors (duplicate exports, temporal-dead-zone refs, missing modules); consolidated duplicated `CASE_STATUS`, `surgicalCases`, and object-literal keys, restoring a clean build with zero Vite or esbuild warnings.

- **Real-Time Collaboration** – Hardened WebSocket layer: implemented exponential-back-off reconnect, token-based auth handshake, ping/pong keep-alive, and UI toast notifications that invalidate React-Query caches on “new_message” and “user_status” events.
- **Developer Productivity** – Scripted one-liner Git flows to cherry-pick single files from `origin/main`, bulk-restore missing components, and ratcheted CI linting to block duplicate property keys—cut build-breaks by 95 %.
- **Secure Auth Context Overhaul** – Re-architected AuthContext with rate-limited login/logout, Role-based theming (console users forced light mode), deep-link redirect preservation, and WebSocket bootstrap tied to `/api/auth/me` session lifecycle.

PFEP Field-Service Platform – *Full-Stack Mobile & API Engineer,*
International Filtration Services (ifs-filter.com)

React Native (Expo) | TypeScript | Node.js (Express) | Docker | SQL Server

MAR 2025 – PRESENT

- Architected a **hierarchical parts-tracking app** (Facility → Building → Level → Column → Unit → Process → Weekly Readings) used by industrial technicians to log filter data across 52-week cycles.
- **Full-stack data model & APIs:** Designed 7+ relational tables (Facilities, Buildings, Plant Levels, Column Locations, Unit Processes, Weekly Readings, etc.), authored idempotent T-SQL migration/seed scripts, and exposed REST endpoints `(/api/facilities/:facility/buildings/:building/...)` with Express—supporting nullable numeric fields, foreign-key cascades, and indexed look-ups for >50k rows.
- **Local & cloud parity:** Containerised SQL Server 2022 Developer via Docker on Apple Silicon, scripted CLI piping for schema management, and mirrored production Azure SQL—enabling offline development without firewall friction.
- **React Native front-end:** Built dynamic FlatList UIs that lazy-load each hierarchy level, implemented secure auth/token refresh with Expo AuthSession, and added universal logout in the header across all nested stacks.
- **Performance & UX optimisations:** Debounced search queries, used expo-router for deep linking, and tuned Flexbox layouts (logo spacing, safe-area insets) for consistent iOS/Android presentation.
- **Team enablement:** Documented VS Code “SQL Server” extension workflows so non-DB engineers could browse tables, run ad-hoc queries, and verify seed data interactively—eliminating error-prone shell pipelines.

MainlineV1 – Dual-GNSS Yield Calculator (iOS) - Lead Swift / GIS

Engineer JAN 2025 – PRESENT

- Integrated Trimble GNSS SDK (TMM reference) to stream real-time position data from two external receivers; exposed **Set Start Point** actions for each device and captured boom-to-boom width on every **Calculate Yield** press.
- Computed treated-area polygons with the shoelace formula using successive width × distance segments, feeding live application-rate and coverage metrics to the UI.
- Wrapped the Objective-C demo from the vendor SDK in a pure-Swift MVVM layer, keeping interop minimal while matching TMM behavior and UX.
- Devised hardware-independent test harnesses that replay NMEA logs, allowing most development without a physical receiver (hardware shipped only for final field validation).

CRAFT - the Social Beer App - Lead Mobile and Web Application Developer,

Android(Kotlin) | iOS(Swift) | Web(React) | Firebase(Node.js) | Full-Stack | Founder

JAN 2019 - PRESENT

- Founding member of a craft beer social media startup. I have entirely developed its Android, iOS, and Web applications and the backend.
- App Store Link: <https://apps.apple.com/us/app/craft-the-social-beer-app/id6444923219>
- Google Play Store Link: https://play.google.com/apps/testing/com.vivinobrothers.craft_thesocialbeerapp
- Website: <https://craft--the-social-beer-6b1e6.web.app/>
- The app began as an extension of the MessageKit API, which allowed users to send messages and save their profiles—created 2019-2020. Original version based on UIKit framework.
- I used Firebase SDK for app user data, beer data, brewery data, beer cellar data, message data, and user authorization.
- Custom emoji actions developed.
- Developed Firebase functions written in Typescript/Node.js to make periodic updates to the data structures in the database.
- Firebase functions via HTTP control, sending push notifications to other users and email updates to administrators when users add beers or suggest edits.
- Created backend Firebase functions (Typescript/Node.js) to calculate crowdsourced beer ratings, updated after each user entered a new rating.

WeatherSats AR App, NASA/Andrew Morgan Consulting (Recipient of Robert H.

Goddard Award) - Unity/C# Developer

MAY 2022 - PRESENT (contract, part-time)

- You can download this app (for free) on the App Store at the following URL:
<https://apps.apple.com/us/app/nasa-weathersats-ar/id6444034201>
- Google Play Store link:
https://play.google.com/store/apps/details?id=com.JPSS.WeatherSatsAR&hl=en_US&gl=US
- Added the AR models to the app and coded object manipulation scripts.
- Added and enhanced the AR features
- Built UI and UI behavior for the AR side of the app. (App has both AR and 3-D sections)
- Added additional UI features.
- Bug-fixed the AR and 3D systems and scene loader.
- PlasticSCM source control and Azure DevOps.
- Provided video updates to NASA on app development progress.
- Prepared and delivered the app to NASA for publishing on the Google Play Store and App Store.
- Responsible for app maintenance and feature updates.
- App is an exhibit at Goddard Space Flight Center in Maryland.

===== CLOSED PROJECTS =====

Living Snow Project – *Mobile App Engineer - iOS & Android*

<https://wp.wvu.edu/livingsnowproject/>

MAR 2025 – JUN 2025

- Developed and launched a bilingual mobile application enabling community scientists to report glacial snow algae sightings across global regions.
- Implemented dynamic multilingual support using i18n and Expo Localization for seamless language switching based on device settings.
- Designed and integrated an interactive geospatial map view to display polygon-based alpine probability models and real-time user-submitted data using react-native-maps.
- Built form flows capturing complex environmental metadata with nested conditional logic and validation (e.g., glacier exposure, snowpack thickness, impurities).
- Engineered offline-first capabilities and photo upload handling with fallbacks for poor connectivity in remote alpine regions.
- Collaborated with domain experts to visualize shapefile-derived data, converting them to GeoJSON and integrating performant rendering strategies for large geometries.
- Managed builds and distribution via Expo Application Services (EAS), including provisioning profiles, OTA updates, and internal testing tracks.

Smoothspeak Application – *Mobile Application Developer*

Swift | Kotlin | React Native

JAN 2025 - MAY 2025

- **Integrated Native Share Extensions:** Developed and implemented native share extensions on iOS and Android, enabling users to seamlessly share images and multimedia content directly from the system share sheet into the [SmoothSpeak app](#).
- **Deep Linking & App Launch Optimization:** Designed and executed deep linking strategies using custom URL schemes and Universal Links to automatically launch the main React Native application from the share extension, ensuring a fluid user experience.
- **Background Processing & Uploads:** Leveraged native background processing techniques—including iOS background URLSession and Android WorkManager—to process and upload shared content even when the main app was in the background or closed.
- **Robust Authentication Flows:** Coordinated, secure token management between the share extension and main app, redirecting unauthenticated users to the login screen, thereby maintaining seamless integration and data integrity.
- **Cross-Platform Collaboration:** Worked closely with cross-functional teams to integrate native modules with the React Native core, ensuring smooth interoperability between native code and JavaScript and optimizing overall app performance.

WaterSprout Mobile Application – *Mobile Application Developer*

SwiftUI | Kotlin | Bluetooth Low Energy (BLE) | Core Bluetooth | Firebase |

Javascript | MVVM

NOV 2024 - JUN 2025

- **Developed BLE Communication Protocol:** Designed and implemented a robust Bluetooth interface for iOS and Android, utilizing Core Bluetooth and Android BLE APIs to send and receive ASCII-formatted commands and parse real-time device responses.
- **Firmware Integration:** Implemented Over-The-Air (OTA) firmware update functionality, managing binary file transfers via BLE using custom protocols and CRC validation for secure updates.
- **Interactive Scheduling Interface:** Created a visually engaging scheduling view in SwiftUI and Kotlin, allowing users to define activation days and times with real-time conflict detection and resolution.

-
- **Cross-Platform UI Design:** Developed reusable UI components for features like PIN entry, water management, and device setup, ensuring consistency across iOS and Android platforms.
 - **Device Configuration:** Built intuitive interfaces for configuring device-specific parameters like activation duration, volume per activation, and rest durations, including validation for HEX and BASE16 values.
 - **State Management & Data Persistence:** Utilized MVVM architecture to sync real-time BLE status updates with app views, storing user configurations securely in UserDefaults and Firebase.
 - **PDF Logging System:** Added functionality to log all BLE interactions and errors into a PDF file stored on the user's device, improving debugging and traceability.
 - **Dynamic Peripheral Management:** Implemented device selection and secure PIN management, requiring user authentication before connecting to peripherals, with support for multiple device sessions.

Carlton Richards/Pryon, “ActionCoach” Chatbot with Generative AI (Pryon SDK) – Mobile Application Developer

React Native | Auth0 | Pryon SDK | Expo | Firebase | TypeScript

SEP 2024 - JAN 2025

- Designed and implemented a [cross-platform chatbot application](#) in React Native, integrating **Pryon SDK** for real-time generative AI responses and **Auth0** for secure authentication.
- Developed a **user-friendly conversational interface** using React Native components, allowing users to ask questions and receive real-time AI-powered responses.
- Integrated **secure token management** with Auth0, ensuring seamless user authentication and authorization flows.
- Implemented **state management** to handle dynamic conversation updates, enabling progressive rendering of AI responses and interactive citations with TextTrackCue.
- Enhanced the user experience by incorporating **real-time video thumbnails** and clickable citations using Vimeo API for deeper contextual engagement.
- Leveraged Expo's managed workflow for efficient development, integrating **custom fonts** and **dynamic theming** to align with branding.
- Built a **custom input field** with real-time feedback, allowing users to efficiently submit queries and scroll through AI-generated answers.

-
- Ensured seamless cross-platform compatibility on **iOS, Android, and Web**, handling device-specific challenges such as video playback and responsive UI.
 - Employed **modern JavaScript best practices**, including async/await, hooks like useEffect and useState, and dynamic theming using useColorScheme.
 - Collaborated with APIs for **token generation** and conversational response, optimizing performance and user interactivity.

Highlights

- **Tech Stack:** React Native, Pryon SDK, Auth0, Expo, TypeScript, Firebase, Vimeo API.
- **Key Features:**
 - **Real-time AI Chatbot:** Progressive rendering of AI responses with generative functionality.
 - **Secure Authentication:** Fully integrated secure login/logout flows using Auth0.
 - **Interactive Media Integration:** Videos and citations enriched user interaction.

New Digital Commune - Lead iOS Developer (Swift/SwiftUI/Firebase/Node.js)

MAY 2024 - MAR 2025

- App Store Link:
<https://apps.apple.com/us/app/new-digital-commune/id6474153217>
- **SwiftUI Development:** Addressed issues with view updates in response to view model changes, ensuring smooth and responsive UI for iOS applications.
- **Media Handling in SwiftUI:** Implemented advanced media handling capabilities using AVKit, Nuke, and Firebase, including video and image caching with custom classes like CachingPlayerItem.
- **GIF Animation Feature:** Developed a new GIF video clip animation feature for the toolkit and profile feeds, enhancing user engagement and providing dynamic content previews of users' video posts.
- **Backend Integration:** Worked on Firebase Functions integration for push notifications and database management, ensuring seamless data synchronization and efficient database management to generate curated feeds.
- **Firebase Deployment and Management:** Regularly deployed and managed Firebase functions, ensuring smooth and continuous integration and deployment workflows.
- **Curated Feeds:** Developed and optimized functions to initialize and update curated feeds for users, handling large-scale data processing using Cloud Pub/Sub for efficient batch processing.

- **Complex JSON Handling:** Managed and manipulated complex JSON structures in Firestore databases, ensuring accurate data representation and retrieval.
- **Logging and Error Handling:** Implemented detailed logging and error handling mechanisms in Firebase Functions to ensure robust and traceable application performance.
- **Analytics React Web App Development:** Developed a comprehensive analytics dashboard using React Native and Firebase, enabling real-time tracking and visualization of user engagement metrics, including time spent on the app by minute, hour, day, week, and month. Created and optimized Firebase Cloud Functions to securely handle backend logic, including custom queries to BigQuery for advanced data analysis.

Pickleball Match - *Full Stack Developer (iOS/Swift, React/Javascript, PostgreSQL/Javascript, AWS)*

JAN 2024 - DEC 2024

- <https://www.pickleballmatch.ca/>
- **Backend Integration:** Worked on the iOS app with a PostgreSQL backend, ensuring seamless data synchronization and efficient database management.
- **AWS Deployment:** Deployed and managed the Pickleball Match website on AWS, ensuring high availability and performance.
- **Backend Updates:** Updated and optimized the PostgreSQL/JavaScript backend, improving performance and scalability.
- **Database Management:** Managed the PostgreSQL database using pgAdmin, ensuring data integrity and optimizing queries for better performance.
- **React Website Updates:** Made significant updates to the React-based website, enhancing user experience and adding new features.
- **Swipe Feature Implementation:** Implemented a right/left swipe feature in the app, improving user interaction and engagement.
- **Push Notifications:** Integrated Apple Push Notifications in the iOS app and PostgreSQL/Javascript backend, ensuring timely and relevant updates for users.
- **Real-Time Data with Pusher:** Implemented real-time updates and notifications using Pusher, handling user connections and ensuring timely information delivery.
- **UI Enhancements:** Updated the UI of the iOS app, providing a modern and intuitive user experience.
- **Complex JSON Handling:** Managed and manipulated complex JSON structures in PostgreSQL databases, ensuring accurate data representation and retrieval.

Andrew Morgan Consulting - *Azure Developer Associate / Power BI Consultant*

MAR 2023 – JUL 2023

Client: Internal team at NASA

-
- Designed, developed, and deployed a custom **Power BI dashboard** to support data-driven decision-making for a NASA project team, integrating multiple data sources and aligning KPIs with mission goals.
 - Collaborated with stakeholders to define key metrics, automate data pipelines, and optimize report performance using **DAX, Power Query**, and **Azure Data Services**.
 - Created secure, scalable backend solutions using **Azure Data Factory, Azure SQL**, and **Blob Storage** to ensure real-time data refresh and high availability.
 - Developed role-level security and user-access management aligned with NASA's compliance protocols.
 - Provided documentation and technical handoff for internal analysts and engineers, supporting long-term maintainability of the dashboard solution.

Top Gear Tractor, Equipment Registration App - *React Native/Typescript Developer (iOS/Android/Cross-Platform)*

JAN 2024-DEC 2024

- <https://testflight.apple.com/join/tJ7qXVbv>
- **React Native Development:** Actively involved in React Native development, including writing TypeScript code for various screens and components in the app.
- **User Interface (UI) Design:** Worked on the UI design of the app, implementing features such as input fields, dropdowns, and placeholders. Additionally, you have customized the styling of components to meet specific design requirements.
- **Debugging and Troubleshooting:** Encountered and resolved issues related to running the Android emulator, connecting to ADB, and configuring Expo to work with the emulator.
- **Collaboration with DALL-E:** Integrated DALL-E, an AI model, into the app to generate placeholder images for tag photos, enhancing the visual appeal of the application.
- **Exploration of Expo and Development Environment:** Explored various aspects of the Expo development environment, including QR code sharing, port configuration, and connection to the Android emulator, demonstrating a strong commitment to improving your development skills.

Lucid, SportsKinetics Discus Throwing App - *iOS Developer*

(iOS/Apple Watch/Firebase/Javascript)

JAN 2024-DEC 2024

- <https://throwingapp.com/>
- <https://testflight.apple.com/join/ht9xsNdd>
- **Integration with Firebase for Authentication:** Implemented user authentication using Firebase Auth, supporting various methods such as email/password and social logins.
- **Development of Account Deletion Feature:** Implemented functionality for users to delete their accounts, utilizing Firebase Functions to handle the deletion process.
- **Utilization of Firebase Functions:** Setup and deployment of Firebase Functions to execute server-side logic, such as account deletion, securely and efficiently.

- **Security Considerations:** Reviewed security measures to protect user data and authentication flows, including using secure connections (HTTPS) and Firebase's built-in security rules.
- **User Interface and Experience:** Discussed the design and implementation of the user interface for the authentication flow, ensuring it's user-friendly and accessible.
- **Feature Implementation for Deleting Throws:** Developed a feature allowing users to delete recorded discus throws, enhancing user control over their data.
- Implemented a user interface swipe action for users to initiate the delete operation on specific throw records.
- **Backend Integration for Throw Deletion:** Set up the necessary backend logic to delete throw records from the database securely upon user request.
- I ensured that the deletion process was linked to the user's account to prevent unauthorized access and deletion of data.
- **Maintenance and Update of App Website:** Used SSH to download and maintain the app html and image files.

b+Positive App, *Flutter/Dart Developer*

JAN 2024 - DEC 2024

- **Project Overview:** Updated and enhances client's existing mobile application using Flutter framework for both iOS and Android platforms.
- **Feature Implementation:** Re-Implemented existing features including user authentication, onboarding screens, in-app purchases, and push notifications using Firebase services and third-party plugins.
- **UI/UX Design:** Designed and implemented intuitive user interfaces following Material Design guidelines, ensuring a seamless user experience throughout the application.
- **Integration and Testing:** Integrated external APIs and services. Conducted thorough testing to ensure app functionality, performance, and compatibility across various devices.

Cellf Defense, *iOS Developer*

MAY 2023 - SEP 2024

- **Project Continuation:** Successfully assumed control of CellDefense, a critical phone radiation app, after the original developer transitioned to Apple, ensuring the project's continuity and stability.
- **Modified Cell Tower Search:** Developed a creative workaround to replace the CTCarrier API, which was removed due to Apple's privacy policy changes. This involved querying Firebase for all MCC/MNC codes for the user's particular carrier in their country and filtering cell towers based on their nearest location to the user, ensuring that the app continued to provide accurate radiation exposure information.
- **Enhanced User Interface:** Implemented improvements to the app's user interface using UIKit, creating a carrier selection wheel in both the onboarding and settings menu.

Venn Interactive, *Stupid Simple Macros App - iOS/Android/Unity/C# Developer*

SEP 2023 - DEC 2023

-
- Venn Interactive owns one of the top food tracking apps on the App Store. The app broke the top 10. This contract involved a variety of coding tasks for the 8.5/8.6 app updates.
 - Nutrition Profile UI completion
 - Old Nutrition Profile Conversion
 - Add Profile Setup option to setup wizard
 - Give users a way to redo the setup wizard for their nutrition profiles.
 - Finalize Nutrition Goal Viewer UI
 - New Nutrition Profile Completion
 - Created a C# parser to import new food data from OpenFoodFacts.org.
 - <https://apps.apple.com/us/app/stupid-simple-macro-tracker/id1210995590>

ModaVision (via Gun.io), Virtual Fitting Room AR/Vision iOS App - iOS Developer

JUL 2023 - JAN 2024

- App Store Link: <https://apps.apple.com/us/app/modavision/id6473058672>
- MODA is a group of NYU Stern student entrepreneurs trying to build a cutting-edge virtual fitting room app.
- Through Gun.io, they hired me for a 160-hour contract. I am building their customer-facing e-commerce app using SwiftUI.
- I used the TrueDepth Camera and Apple APIs to “measure” the body through the iPhone camera.
- The app pulls in clothing size data from the clothing manufacturer and compares it to the body measurements to assess how well the clothes fit.

Word Game App - Unity/C# Developer

JAN 2023 - DEC 2024

- The client was looking for a C# (Unity) programmer to finish an app that was about 80% complete.
- The game is a word game wherein you form words to play and win the game by scoring higher than the opponent(s).
- I utilized my expertise in AI and algorithm complexity/efficiency to create a more challenging AI opponent.
- Incorporated LINQ and knowledge of algorithmic complexity to maximize AI strategy.
- Made updates to the graphical user interface, added sound effects, and user-friendly features.

Flappy.io, Apple Vision Video Game iOS App - iOS Developer

MAY 2023 - DEC 2023

- This app is a remake of the “Flappy Bird” video game, but using the Apple Vision human skeleton detection.
- Player jumps, making the bird “fly up” on the screen to avoid obstacles.
- Made adjustments to the functions that detect the player jumping.
- Built the tutorial UI and game over UI/functionality.
- App uses SpriteKit/UIKit/Apple Vision API
- <https://apps.apple.com/us/app/flappy-ai/id6449951841>

Apple Vision Dance Game iOS App - *iOS Developer/Python/AWS Backend Developer*

MAR 2023 - DEC 2023

- This game uses the Apple Vision human detection to perform an analysis of a dancer as to how well they dance.
- Developed and deployed a Flask-based scoring API on an AWS EC2 instance for real-time body motion analysis, supporting the Apple Vision-powered iOS dance app.
- Set up secure domain routing via Route 53, Nginx reverse proxy, and Let's Encrypt SSL to ensure HTTPS compliance for production traffic.
- Integrated Gunicorn as a WSGI server for scalable, production-grade request handling.
- Created a custom scoring pipeline using SciPy, NumPy, and Pandas to analyze joint movement data against music BPM and return normalized performance scores with annotated visual plots.
- Supported SwiftUI app integration with CORS-enabled endpoints and JSON-based request/response validation.

Lost Works, AR App - *Unity/C# Developer*

MAY 2023 - JUNE 2023

- Rework of UI.
- Update the app to include the current version of Vuforia.
- I have implemented a model target for tracking.

AIA Spurs, Korea - Person Segmentation Web AR App - *Web AR/Unity/Node.js Developer*

APR 2023 - MAY 2023

- <https://vjos.zappar.io/8580527391267934165/1.2.5/>
- Created/bug-fixed a person segmentation web app that sits on an already-created online hub for a Korean company.
- The web app allows users to choose from two backgrounds and take a selfie with their chosen background.
- From there they can share it or download the image.
- They will then be able to click back to the hub to continue exploring the site.
- Tech stack: Unity, WebGL, Zappar, Universal AR, Zappar CLI, Node.js, Javascript, TensorFlow
- Created a simple REST API endpoint to remove the background from the image using Node.js and TensorFlow.js

All-Star Sports, Eureka Canvas - Sports Edition App - *iOS Developer*

JAN 2022 - JAN 2023

- Official App Store Release:
<https://apps.apple.com/us/app/eureka-canvas-sports-edition/id1643624044>

- Developed a sports video markup app for a sports coach client.
- Integrated body position analysis code from Apple's 2020 WWDC Vision example.
- Full video scrolling/slider/ +/-5, play/pause, 1/8, 1/4, 1/2, 1x, 2x buttons implemented.
- PencilKit API.
- StoreKit API for the app subscription.
- Custom shapes, arrows, and sports markup features.
- Ability to save and load canvases and game scenes.
- Use of trigonometry functions to calculate vector angles for shooting balls.
- Future updates will use Apple's Vision API to track the ball's path from video analysis.

Artistry AR - *Web AR Developer*

MARCH 2023 - MARCH 2023

- Wrote a Node.js server using Express, hosted on AWS to convert videos coming from Android devices from webm to mp4 format so that they could be shared via WhatsApp and other popular social networking apps.
- Made small adjustments to the controls for placement and positioning of the models in AR.
- Used WebRTC framework (RecordRTC) and MediaRecorder API to record the dancer from the canvas on the screen.

Pioneer Mobile Applications - *iOS/Android Developer (part-time contracts)*

JAN 2023 - DEC 2023

- TradePit-iOS <https://apps.apple.com/us/app/tradepit/id1616285544>
 - Made various updates to the UI and push notifications.
 - Implemented "sort by newest item" feature
 - Bottom sheet picker for users to select the state they live in.
 - Disable banner and sound push notifications when in the middle of a text message thread with another user.
 - UI enhancements.
 - Installation of Crashlytics
- LottoLytics (iOS/Android)
 - Implemented a bottom picker that shows lottery games available in each state. Saves the user's last selected state choice in user defaults. iOS and Android app versions.
- Rapid Response Co. (iOS)
 <https://apps.apple.com/us/app/rapid-response-co/id1580988008>
 - Implemented a settings controller that accesses UserDefaults to save app settings.
 - Leveraged settings to change the map stack views so that users can decide which map type is on top and how many maps to display.

SoulandForm, LLC (<https://soulandform.com/>) - *iOS Developer*

JAN 2022 - JUNE 2022

- You can download this app on the App Store at the following URL:
<https://apps.apple.com/us/app/soul-and-form/id1633902006>
- Built a SwiftUI-based animation video game with a Firebase backend for an Upwork client from scratch. This contract turned into a business partnership.
- I coded the app from scratch using SwiftUI and Firebase.
- Responsible for all UI and backend calls and Firebase security rules.
- Coded StoreKit functionality for in-app purchase subscriptions.
- Gameplay - players draw animations and challenge other players to find the "link" or "rule" between the animations.

PlanetXR (<https://www.planetxr.net/>) - *iOS Developer (part-time)*

AUG 2022 - APRIL 2023

- <https://apps.apple.com/us/app/planet-xr-1-spatial-ar-world/id1605504634>
- Created a transparent tutorial video overlay on the main app screen.
- Timed UI text to coordinate with the transparent video.
- Used the view alpha channel to fade the tutorial and text in and out for a cinematic effect.
- Coded app logic to display the tutorial only in applicable scenarios.
- Updated the Google Maps view to display simultaneously with the AR camera in the bottom third of the screen to enhance the user's ability to find objects in the AR world while navigating in AR.
- Updated the map view system to include AR models and AR "air messages" based on their thumbnail or text data.
- Participation in Jira system for software project management
- Fixed a large variety of software bugs, including the GraphQL resolver for emoji comments.
- Currently working on a model upload website in React Native/Node js, which connects with the GraphQL/Node js backend.

Upwork - *Freelance iOS Developer/Freelance Unity Developer (part-time, contract/freelance)*

JAN 2022 - PRESENT

- *Veggie Ninja, iOS/Machine Learning (Apple Vision) Developer*
- *TrackHero* (<https://trackhero.com/>), *iOS Consultant*
 - Served as a consultant to help the client make API calls using Combine.
 - Implementation of the Stripe Purchase API
 - Helped solve issues with concurrency, loading data models, and SwiftUI view management.
- *Arrows Video Game, Sole Unity Developer*
 - A Unity/C# WebGL third-person shooter game for a client called "Arrows."
 - <https://play.unity.com/mg/other/webgl-builds-154388>
- *LEGO Sky Island Escape (Unity Port)*
 - I made a port of a Unity-based LEGO game to iOS.
 - I coded the user control logic and combined pre-built game levels, customizing some of the game levels.

- Public beta is open through TestFlight here:
<https://testflight.apple.com/join/BDWu8n5P>.

SimFit App - *iOS/WatchOS Developer, Internship*

AUG 2022 - FEB 2023

- Lead developer on a workout-tracking WatchOS app that links the user's progress to video game play on a linked iOS device.
- Responsible for developing the user interface behavior using SwiftUI.
- Development of the backend communication and database structure using Firebase. Created view controllers and leveraged a database manager I created for my other applications (Craft - the Social Beer App and SoulandForm) to create an auth and data system for user profiles in this app.
- Use of HealthKit, WatchKit, and WatchConnectivity to link the WatchOS app to the main app.
- User records workout data on the watch app which is sent via WatchConnectivity messaging to change the speed of the player in real-time in the video game, which I also created. (It is a simple racing game.)

IBM - *Summer Internship, Computer Science*

JUN 2002 - AUG 2002

Lucent Technologies - *Summer Internship, Computer Science*

JUN 2001 - AUG 2001

SKILLS/FRAMEWORKS/LIBRARIES

- | | |
|---------------------------------|--------------------|
| ● Swift | ● Express |
| ● C# | ● StoreKit |
| ● JavaScript/TypeScript | ● AVKit |
| ● Node.js | ● MessageKit |
| ● Objective-C | ● MapKit |
| ● Flutter/Dart | ● PencilKit |
| ● React/React Native/JavaScript | ● RealityKit |
| ● WatchKit | ● ARKit |
| ● HealthKit | ● AFrame |
| ● WatchConnectivity | ● CoreML |
| ● SwiftUI | ● Vision |
| ● UIKit | ● React |
| ● Unity | ● Bootstrap |
| ● Firebase | ● CSS |
| ● Firebase Functions | ● HTML |
| ● GraphQL | ● Game Development |
| ● AWS | ● LINQ |

EDUCATION

Rutgers University, New Brunswick, NJ - *Bachelor of Science, Computer Science, Bachelor of Arts, Music*

SEPT 2000 - MAY 2004

- Graduation from the Honors Program
- 4.0 GPA In Computer Science courses

CERTIFICATIONS

Data Storage in Microsoft Azure for Associate Developers (Azure Developer AZ-204 Certificate)

Coursera

DEC 2023

- Credential URL: [3Y3Y9APNQ2JA](https://www.coursera.org/account/accomplishments/certificate/3Y3Y9APNQ2JA)

Material Components for Flutter Basics, Coursera/Google Cloud

JUL 2023

- Credential ID: WPEW2SFY6RYE
- Credential URL: <https://www.coursera.org/account/accomplishments/certificate/WPEW2SFY6RYE>

Getting Started With Flutter Development, Coursera/Google Cloud

JUL 2023

- Credential ID: K55Q2XRPSTNA
- Credential URL: <https://coursera.org/verify/K55Q2XRPSTNA>

Connect Your Services with Microsoft Azure Service Bus (Azure Developer AZ-204 Certificate)

Coursera

JUL 2023

- Credential ID: LJRHYGH7ZH28
- Credential URL: <https://www.coursera.org/account/accomplishments/certificate/LJRHYGH7ZH28>

Create Serverless Applications (Azure Developer AZ-204 Certificate),

Coursera

JUL 2023

- Credential ID: UFBFHNJRZEUQ
- Credential URL: <https://coursera.org/verify/UFBFHNJRZEUQ>

Programming With JavaScript, Coursera

APR 2023

- Credential ID: CHH4F24CB6KN
- Credential URL: <https://coursera.org/verify/CHH4F24CB6KN>

Introduction to Back-End Development, Coursera

DEC 2022

- Credential ID: LVTBP9N764AU
- Credential URL: <https://coursera.org/verify/LVTBP9N764AU>

Introduction to Front-End Development, Coursera

AUG 2022

- Credential ID: WQ3JRSV5GD8G
- Credential URL: <https://coursera.org/verify/WQ3JRSV5GD8G>

Unity Essentials Pathway, Unity Technologies

JAN 2021

- Credential URL:
<https://www.credly.com/badges/ab4e6d89-f757-443b-988e-583492d9dac4>

AWARDS

- Robert H. Goddard Award, NASA, August 29, 2023
- 772 Coding Score, CodeSignal
- CodeSignal, Level 42
- LinkedIn Swift Badge