|  |
| --- |
| Eric Bowser  5154 S 5200 W, Kearns, 84118 · 435-494-8030  [ericryanbowser@gmail.com](file:///C:\Users\ericr\Documents\ericryanbowser@gmail.com) · [https://www.linkedin.com/in/eric-bowser-35028880/](https://www.linkedin.com/in/eric-bowser-35028880/%20)  <https://engineer-it-all.com/> |
| Full stack engineer looking for a new opportunity to expand craft, and take career to the next level in an environment where innovation and creativity is embraced |

# Experience

|  |
| --- |
| 2016 – 06/2022Software Engineer, **WILLIS TOWERS WATSON**   * Support Bounded Context(s) * Migrated from monolith pub / sub to HTTPS RESTful messaging * Migrated from on-prem PostgreSQL to Azure Cosmos DB * Internal tool to build dynamic applications * Responsibilities ranging from public facing web page to data layer   + Moving towards DDD – Sending events between bounded contexts * Healthcare specific conditional logic used based on CMS compliance and carrier-specific application rules |
| 2015 – 2016Software Developer, Tech Systems Contract to hire for Extend Health, LLC 2013 – 2015Software Developer, GOLDPoint Systems  * Develop and support new client code for onboarding new clients * Creation of unique SQL reports for client statistics * Owned or inherited unique client reports using C# Drawing libraries   + Custom rendered reports to financial clients |

# Education

|  |
| --- |
| 2011 - 2013Bachelor of Science, Weber State University  * Computer and Network Security * Software Development   + Both capstone paths (A little less networking and more development) |
| 2010 – 2011Associate Of Art, University of Phoenix  * IT fundamentals * introduction into software development |

# Skills

|  |  |
| --- | --- |
| * JavaScript * React JS * NPM / Node * T-SQL * NoSQL * Html and CSS | * Design Patterns * C# and .NET * JavaScript * Microservices * Bootstrap * Styled Components * Scripting |

**Front End - React.js Frontend SPA**

* + Declarative style (functional composition)
  + State management
    - React Hooks
      * Local state management
    - React Query
      * Extensive library for handling fetching and caching for a limited stale time
      * Custom hooks to import and access API endpoints
        + Prefetching
        + On demand
    - Custom Hooks
      * Own implementation to wrap intended logic, data-fetching, and communication with other imported web components
        + Implemented custom hook that would send API requests to another imported component of a different bounded context, and supply a user-friendly way to get specific data back from the request
  + Micro frontend
    - Front end solution that used a document database in Azure
      * Cosmos DB
      * Returns form / application data back to the front end
  + ASP.NET, IIS, and Kestrel
    - Micro frontend hosted in Kestrel
      * Builder pattern and bootstrap of application

**Styles**

* + - HTML, Styled-Components, and CSS
* HTML boilerplate and elements
* CSS styles, selectors, pseudo-combinators, and specificity
* Styled-Components (React)
* React Bootstrap for Components

**Microservices**

* HTTP Messaging and Restful API
  + ASP.NET API
  + Microservices and bounded contexts
  + Restful APIs – HTTP/HTTPS messaging
  + Swagger API Versioning

**Security**

* OAuth
* Windows Auth (NTLM)
* Bearer tokens / API key

T-SQL or Microsoft SQL Server

* Microservice endpoints for SQL
  + - Stored procedures
    - User-defined functions
    - .NET microservices to data layer
      * Dapper / ORM
      * Entity Framework
    - Views and Read-only
      * Read-only SELECT or Read access
      * Basic SELECT statements or table references
* **CLI**
  + Bash
    - Git bash CLI
    - Git source control commands for local repository and branch development
  + Cake
    - Build script automation tasks
    - executing pre-deploy build, package, and run tests
  + Powershell
    - Execute automation scripts
    - Database scripts / New migration scripts
* **Environment**
  + Node runtime environment
  + NPM – Node package manager (NPM packages)