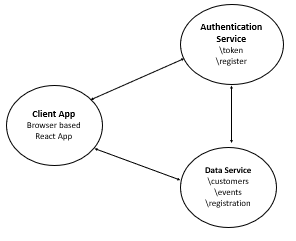
**Developed By Hampton Black and Hunter Thomas**

**Diagram:**



**Component List:**

* Authentication Service
* Data Service
* Client Application

**Component Interaction:**

* The client application gets JWT token from the authentication service during login
* The client application accesses Customer, Event and Registration data via the Data API
* The client application adds the JWT token to all Data API requests
* The authentication service calls the Data API to verify user/password during token requests
* The authentication service calls the Data API to insert Customer records during register requests

**Authentication Service:**

Information:

* Path: \account
* Port = 8081

Client App:

* React App

Authentication Service:

* \account\token
* \account\register

Endpoints:

* Token endpoint:
  + Returns a JWT token that connects to the data service
* Register endpoint:
  + Adds the name, email, and password to the customer data in the endpoint of the customer.

Security:

* Token is not required to access the authentication service endpoints

**Data Service:**

Information:

* Path: \api
* Port: 8080

Endpoints:

* \account\customers
* \account\events
* \account\registrations

Security:

* A servlet filter will be used to intercept calls to the api and check requests for JWT tokens

Customers, Events, and Registration Endpoints:

* Supports CRUD management
* Supports GET, POST, PUT, and DELETE requests
* JWT Token required for all requests

Registration Endpoints:

* Supports CRUD management
* Supports GET, POST, PUT, and DELETE requests
* JWT Token required for all requests
* Customers can register for events

**Work Plan (high level):**

* Create Customer endpoints
* Test Customer endpoints
* Create authentication API’s token endpoint
* Test token endpoint
* Secure the Data API service with filter that checks for JWT tokens
* Get token and test Customer endpoints including token
* Test APIs using the React client application

**Work Plan (task backlog):**

* Create local git repos
* Data API Code
* Auth API Code
* Implement Data API’s root endpoint
* Implement the Data API’s Customer GET endpoint
* Test created endpoints with postman REST tools
* Implement PUT, POST, and DELETE Customer API endpoints
* Test created endpoints with Postman
* Test customer endpoints using React client
* Implement Events and Registrations endpoints
* Test created endpoints with Postman
* Test created endpoints using React client
* Create Authentication API
* Implement token endpoint
* Test token endpoint using postman
* Test token endpoint using React client (the client’s login function uses the token endpoint)
* Add authentication filter to the Data API to require a JWT token to access services
* Try accessing Data API with authentication filter in place using Postman
* Test login and access to customers, events, and registrations records using the react client