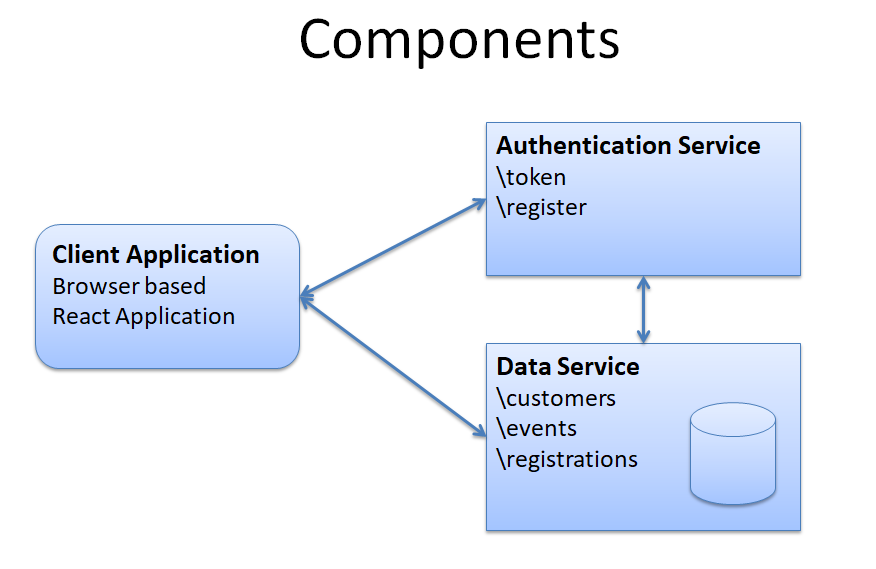
## Component Diagram



## 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

* Service information:
  + context path = \account
  + port = 8081
* Endpoints
  + \account\token
  + \account\register
  + \account\
* Root Endpoint (\account\)
  + Returns a message confirming that the service is up and running.
* Token Endpoint
  + Responds to requests for JWT tokens
  + Takes username and password as parameters
  + Checks that username exists in customer store
  + Checks that password matches the one listed for the customer in the customer store
  + Returns a JWT token suitable for connecting to the data service
* Register endpoint
  + Takes name, email and password and adds them to the Customer data store via the Customer endpoint of the data service
* Security
  + A token is not required to access the authentication service endpoints

## Data Service

* Service information
  + context path = \api
  + port = 8080
* Backend Data
  + Will use embedded hsql database
* Endpoints
  + \account\customers
  + \account\events
  + \account\registrations
* Root Endpoint (\api\)
  + Returns a message confirming that the service is up and running.
* Security
  + A servlet filter will be used to intercept calls to the api and check requests for JWT tokens
* Customers Endpoint
  + Supports CRUD management of Customer records
  + Supports GET, PUT, POST and DELETE requests
  + A JWT token is required for all requests
* Events Endpoint
  + Supports CRUD management of Events records
  + Supports GET, PUT, POST and DELETE requests
  + A JWT token is required for all requests
* Registrations Endpoint
  + Supports CRUD management of Registrations records
  + Allows customers to register for events
  + Supports GET, PUT, POST and DELETE requests
  + A JWT token is required for all requests

## 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 for
  + 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 tool
* 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