**Req2: Manage the user session with refresh token**

In current inventory build oAuth2.0 pattern has been used to secure back end resources. On successful user authentication backend generates Back end generates token and refresh token in the following format.

*{*

*"access\_token": "7a1c927f-aae8-4a05-9a36-8b86460e8f98",*

*"token\_type": "bearer",*

*"refresh\_token": "71c8d47c-e9de-449f-a828-fa8fe9075da0",*

*"expires\_in": 119,*

*"scope": "read write trust",*

*"firstName": "Sridhar",*

*"lastName": "Jakkula",*

*"id": 4,*

*"username": "sri1@nowhere.com",*

*"token": "7a1c927f-aae8-4a05-9a36-8b86460e8f98"*

*}*

Access token is valid for 1 hour and refresh token is valid for hours. These values are configurable and access token validity will be adjusted to minutes in future build. In current reactJS token accesstoken is generated only once. Once accesstoken is expired user session will be timed out and user will be forced to sign in again.

Any resource request send 401 error when access token expires.

*{*

*"error": "invalid\_token",*

*"error\_description": "Access token expired: 7a1c927f-aae8-4a05-9a36-8b86460e8f98"*

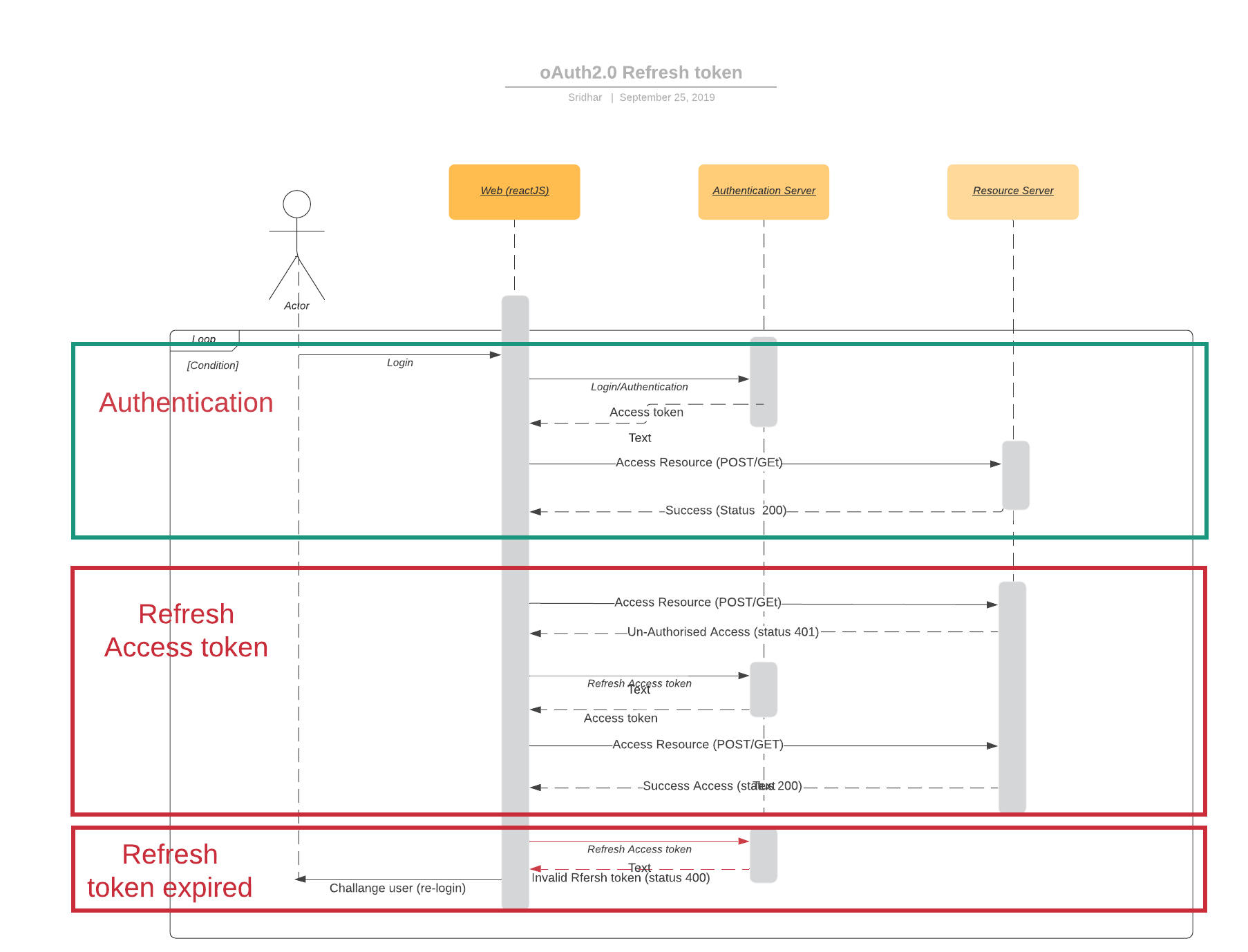
*}*

Required Change:

When accesstoken expires extend current reactjs code base to refresh accesstoken with refresh token to keep the session alive.

API Refrence:

Refer to Refresh token end point in POST MAN



Authentication has been already implemented in above sequence. Refreshing access token and refresh token expiry scenarios need to be implanted.

Refer function login(username, password) for Authentication implementation.

Inputs for implementation:

* Write a method handler to refresh access token
* When 401 error code received in response use this above method to refresh access token and retry same request again.

React JS ref: user.service.js

Testing Setting:

1. Change the access token to 2 minutes in config/AuthorizationServer.config
2. Recompile the code

* mvn clean
* mvn install

1. Restart back end server

java -jar target/Inventory-0.0.1-SNAPSHOT.java