\*\* Introduction \*\*  
  
Ensek Energy Corp application is implemented as a web application which help caters to specific requirements and process for users. This solution is implemented as part of the overall user journey and to provide well defined user experience.

The current scope of this strategy is to focus on given user requirement on navigating around the application.

\*\*Test Lifecycle\*\*

The standard agile testing lifecycle which will be followed consist of :

Requirement Analysis 🡪Test Planning 🡪 Test Case Development 🡪 Environment Setup 🡪 Test Execution 🡪 Test Validation 🡪 Test Closure

The requirement analysis is done the current version of the Ensek Application UI features which should define the test strategy and the scope of the test cases.

The test plan is created as a part of test planning to document the test strategies and methodology which will be used to validate the Ensek Application.

The testcases are developed to cover most of the UI features. The Acceptance criteria and Expected Results are created to validate the product feature

The Environment Setup is conducted with the help of the ensek development team which its url configurations is created to cater for test users .

The test cases execution will be performed by different testers involved in all streams of project around the Ensek Application.

The test cases will be executed and validated against the expected results for the product features.

In the event of any issues against any product feature and the expected result varies from the predefined expectations, Then it will be logged as a bug or defect.

One the defects or bugs are fixed and validated against the expected results, then the defects can be closed and regression testing can be initiated and sign off to deploy the Ensek Application into the next higher environment.

Test Scope & Approach

The scope for testing this application will be as follows :  
\*\* Functional Testing

\*\* Usability Testing

\*\* Operational Testing

The due to the minimal domain knowledge of the application, the approach to this testing will be an exploratory testing approach .

\*\*Test Automation\*\*

Automation Technique

To automate the frontend of the application, the testing framework of choice will be cypress due to the less effort to set up and the language of choice for the automation script will be typescript / JavaScript since the front end is most likely to be written in either of those languages, Cypress interact with . Also the API automated test can also be written using the same tool.

If the API endpoints are public facing and accessible without a need for access keys then the API endpoint test can be automated using Jest.

Some of the end to end test will need to be written to validate the flow of various user journey such as

Register

Log in

Buy Some Energy

Sell Some Energy ( which is down at the time of writing this)

Integration test also needs to be considered as to check that the API endpoints operations are directly reflected in the database which is very important whilst developing the application, The creation of the API could be the first task before the integration of front end code , it is very important to check the server side interaction with the database.

Cypress Locators

Cypress supports various locators such as tags ,id, class, attributes, text, e.t.c . Also Cypress also supports Xpaths selectors which with the help of third party plugin such as XPath, Cypress Recorder, UIPath and a host of others . For this application we can use a customized CSS locators to interact with UI elements.