

# MIDSTATES IMS

## TEST PLAN

### INTRODUCTION

This is a test plan for the Midstates Inventory Application (IMS). Due to time constraints the test plan will be focused on manual testing of "the happy path". Meaning that I will not be focusing on all the ways that the application will break.

After school is finished, I do plan to conduct thorough testing on the application before it is released.

### BUSINESS BACKGROUND

The project is an IMS for Midstates Equipment and Supply. The IMS will allow the user to enter data about their equipment in a form. After the form is submitted, the data will upload to their site. The IMS will also include authentication for security purposes.

The Midstates website is currently static, and they do not have a way to change out the items they have in inventory easily. The client has limited experience in web development and frequently runs into issues when trying to update their existing static website. The layout of the site is not formatted like how it was initially and is not user friendly.

### TEST OBJECTIVES SCOPE

#### INCLUSIONS

- Manual Testing
- Functions and external interface of the Midstates Inventory Application.

#### EXCLUSIONS

- Automated Testing
- Non-functional testing such as stress, performance or logical database will not be tested.
- Security will not be tested
- Performance testing

## TEST PLAN

### TEST TYPES IDENTIFIED

#### BETA TESTING

Application will be deployed to a live environment to ensure there are no major failures in the application.

Testing will be manual.

#### ACCEPTANCE TESTING

To be performed by the client to verify if the end to end flow of the product meets business requirements. For this project, the client will only be able to provide initial feedback as all requirements for the client will not be met due to time constraints.

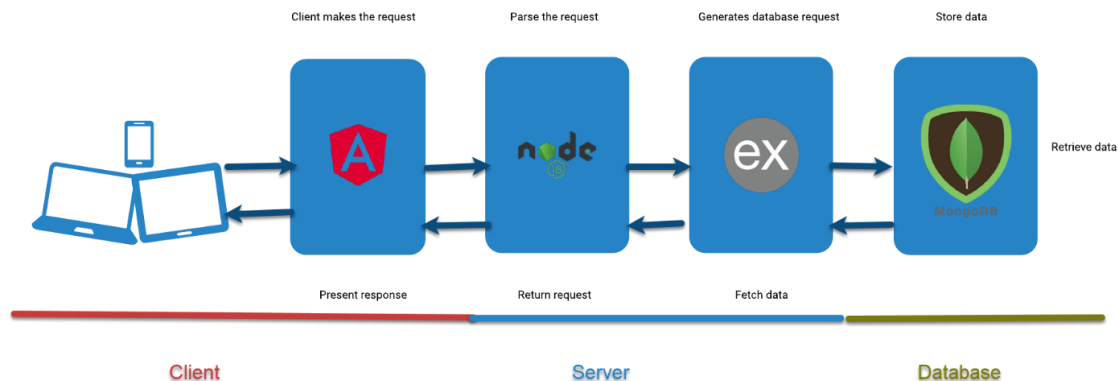
### PROBLEMS PERCEIVED

- Lack of experience with testing
- Limited time to complete tests
- Lack of an objective set of eyes. I am too close to the project and do not have a team to work with.
- Testing is not through enough resulting in numerous bugs.

To mitigate the problems, I plan to conduct more through testing after the course is over.

### ARCHITECTURE

The MEAN stack will be used. The MEAN stack includes MongoDB, Express, Angular, and Node.



Angular will handle the presentation and render the user interface with dynamic data, manage the user input, and communicate with backend services.

## TEST PLAN

The server side will handle the business logic with node.js and express.js. Node.js listens to requests and sends responses, executes server-side logic, and interacts with databases and files. Express is middleware-based and funnels the requests through functions and connects Angular with the backend.

The database that I will be using is MongoDB.

### ENVIRONMENT

#### TO BE DETERMINED

- Maximum user connections that the website can handle at the same time
- Hardware/Software Requirements
- Settings that may need to be configured

### ASSUMPTIONS

- Users will be on Windows 10
- Primary browser Google Chrome
- Maximum three users logged in at a time

### FUNCTIONALITY

- User can be added
- User can reset password
- User can login
- Inventory items can be added
- Inventory items can be updated
- Inventory items can be deleted