

# System Verification and Validation Plan for Capstone 4ZP6A

## **StockSavvy: A User-Friendly Stock Market Solution**

Team No. 4

Jaskaran Singh; singj47@mcmaster.ca

Hosty Khurana; khurah2@mcmaster.ca

Shubham Ahuja; ahujas6@mcmaster.ca

**February 8, 2024**

## Table of Contents

<b>1. VERSION HISTORY .....</b>	<b>3</b>
<b>2. PROJECT DESCRIPTION.....</b>	<b>4</b>
<b>3. TESTING METHODOLOGIES .....</b>	<b>4</b>
<b>4. TESTING TOOLS AND ENVIRONMENTS .....</b>	<b>4</b>
<b>5. RISK ANALYSIS AND MITIGATION STRATEGIES .....</b>	<b>4</b>
<b>5.1 RISK OF DATA INCONSISTENCY: .....</b>	<b>4</b>
<b>5.2 RISK OF MODEL OVERFITTING: .....</b>	<b>5</b>
<b>5.3 RISK OF SECURITY VULNERABILITIES: .....</b>	<b>5</b>
<b>5.4 RISK OF POOR USER ADOPTION DUE TO USABILITY ISSUES: .....</b>	<b>5</b>
<b>6. COMPONENT TEST PLAN.....</b>	<b>5</b>
<b>6.1 DATA RETRIEVER COMPONENT.....</b>	<b>5</b>
<b>6.2 DATA CLEANER COMPONENT .....</b>	<b>5</b>
<b>6.3 LSTM (LONG SHORT-TERM MEMORY) COMPONENT .....</b>	<b>6</b>
<b>6.4 GUI COMPONENT .....</b>	<b>6</b>
<b>6.5 BACKEND INTEGRATION .....</b>	<b>6</b>
<b>6.6 MACHINE LEARNING MODEL .....</b>	<b>6</b>
<b>6.7 OVERALL SYSTEM TESTING .....</b>	<b>6</b>
<b>7. SCHEDULING AND MODEL TRAINING PLAN.....</b>	<b>6</b>
<b>7.1 WEEKLY MODEL TRAINING SCHEDULE.....</b>	<b>7</b>
<b>7.2 MONITORING AND CONTINUOUS IMPROVEMENT .....</b>	<b>7</b>

## 1. Version History

Date	Version	Contributors	Notes
02-02-24	0	Jaskaran Singh Hosty Khurana Shubham Ahuja	Initial System V&V Plan Created.
02-02-24	1	Shubham Ahuja	Project Description Added.
03-02-24	2	Jaskaran Singh	Testing Methodologies Added.
04-02-24	3	Hosty khurana	Testing Tools and Environments Added.
05-02-24	4	Jaskaran Singh Hosty Khurana	Risk Analysis and Mitigation Strategies Added.
07-02-24	5	Jaskaran Singh Hosty Khurana Shubham Ahuja	Component Test Plan Added.
08-02-24	6	Shubham Ahuja	Scheduling and Model training Plan Added.
08-02-24	7	Jaskaran Singh Hosty Khurana Shubham Ahuja	Final Editing and Review Completed .
29-03-24	7	Jaskaran Singh Hosty Khurana Shubham Ahuja	Final V&V Document .

## 2. Project Description

StockSavvy is an innovative stock market analysis tool designed to cater to the diverse needs of individual investors, financial advisors, financial institutions, regulators, and data providers. With its user-friendly interface and powerful features, StockSavvy enables efficient stock market analysis, assists in making informed investment decisions, integrates seamlessly for enhanced financial services, promotes investor education and protection, and identifies opportunities for investment and data enhancement through collaboration with data providers.

## 3. Testing Methodologies

- **Automated Testing:** Used for repetitive tasks and regression testing. For instance, automated unit tests verify individual components (e.g., data retrieval, LSTM training) for consistency after modifications.
- **Manual Testing:** Critical for areas where human judgment is crucial, such as GUI testing and overall user experience. Manual testing also includes exploratory testing to uncover issues not covered by automated tests.
- **Black-Box Testing:** Used mainly for functional testing, where the internal workings of the system are not known to the tester. This approach is useful for testing user interfaces and overall system functionality.

## 4. Testing Tools and Environments

- **Unit Testing:** Python's unittest framework for backend testing; for ML components, use pytest along with numpy.testing for asserting numerical outputs.
- **GUI Testing:** Selenium for automated browser-based tests of the web GUI.
- **Performance Testing:** JMeter or similar tools for stress testing the backend and measuring response times.
- **Continuous Integration/Continuous Deployment (CI/CD) Environment:** Use Jenkins or GitLab CI for automating the testing process as part of the CI/CD pipeline.

## 5. Risk Analysis and Mitigation Strategies

### 5.1 Risk of Data Inconsistency:

Due to reliance on external data sources (Yahoo Finance), there's a risk of data inconsistency or unavailability.

**Mitigation:** Implement redundant data sources and establish data validation checks.

### **5.2 Risk of Model Overfitting:**

The ML model may be overfit to the training data or fail to generalize to future, unseen data.

**Mitigation:** Apply cross-validation, utilize a dedicated test set for final evaluation, and adopt early stopping during training. Regularly update the model with new data to maintain its generalizability over time.

### **5.3 Risk of Security Vulnerabilities:**

User data and financial information are sensitive and could be targeted.

**Mitigation:** Regular security audits, adherence to best security practices, and data encryption.

### **5.4 Risk of Poor User Adoption Due to Usability Issues:**

The application might not meet user expectations in terms of usability and functionality.

**Mitigation:** Regular user feedback sessions and usability testing to guide interface design.

## **6. Component Test Plan**

### **6.1 Data Retriever Component**

#### **Unit Tests:**

- Test Case 1: Verify successful data retrieval from Yahoo Finance.
- Test Case 2: Handle network failure scenarios.
- Test Case 3: Validate the format and schema of the retrieved data.
- Test Case 4: Implement checks to prevent data leakage between training, validation, and test sets. This includes verifying windowing functions and ensuring that the correct partitions are used for each phase of model evaluation.

#### **Performance Tests:**

- Test the response time for data retrieval under various network conditions.
- Assess the reliability of the data retrieval process over multiple iterations.

### **6.2 Data Cleaner Component**

#### **Unit Tests:**

- Test Case 1: Ensure accurate handling of missing or corrupted data values.
- Test Case 2: Validate the integrity and consistency of processed data.
- Test Case 3: Check the data formatting process for compatibility with downstream components.

#### **Performance Tests:**

- Evaluate the processing time for data cleaning with datasets of varying sizes and complexities.

### **6.3 LSTM (Long Short-Term Memory) Component**

#### **Unit Tests:**

- Test Case 1: Verify the training process on historical data.
- Test Case 2: Assess the accuracy of stock price predictions.
- Test Case 3: Validate model retraining triggers and processes.

#### **Performance Tests:**

- Measure prediction accuracy using RMSE and other relevant metrics.
- Test the model's performance with unbalanced datasets and varied market conditions.

### **6.4 GUI Component**

#### **Unit Tests:**

- Test Case 1: Validate functionality of login and user creation processes.
- Test Case 2: Check the stock selection process and accuracy of displayed data.
- Test Case 3: Ensure proper rendering and functionality of analysis viewer and other UI elements.

#### **Performance Tests:**

- Measure GUI loading times and responsiveness on different devices and operating systems.

### **6.5 Backend Integration**

#### **Unit Tests:**

- Test the seamless integration and data flow between backend components.
- Validate the stability and consistency of integrated functionalities.

#### **Performance Tests:**

- Evaluate system response time and resource usage under normal and peak load conditions.

### **6.6 Machine Learning Model**

#### **Performance Tests and Metrics:**

- Implement cross-validation to evaluate predictive accuracy.
- Use appropriate metrics, like precision and recall, in addition to RMSE, considering the dataset's nature.

### **6.7 Overall System Testing**

#### **End-to-End Tests:**

- Simulate real-user scenarios to validate the complete system workflow from data retrieval to analysis presentation.

#### **Stress Tests:**

- Determine system stability and performance under high-load and stress conditions.

## **7. Scheduling and Model Training Plan**

## **7.1 Weekly Model Training Schedule**

StockSavvy's machine learning model, particularly the LSTM component, will undergo weekly retraining to ensure up-to-date accuracy with the latest market data. This process includes automated data fetching from Yahoo Finance, preprocessing to fit the model's requirements, and retraining the model with the new data. The training cycle is set to occur at the start of each week, ensuring that the model consistently learns from the most recent trends and patterns in the stock market.

## **7.2 Monitoring and Continuous Improvement**

Post-training, the model's performance will be evaluated using key metrics like RMSE (Root Mean Square Error) to ensure its efficacy in stock price prediction. Any significant deviations in performance will trigger an immediate review and potential adjustments to the model. This cycle of regular training, monitoring, and improvement is crucial for maintaining the robustness and reliability of StockSavvy's predictive capabilities, thereby providing consistent and valuable insights to its users.