

Project Requirements Document, v1

Team Name: Excelsior

Project Name: Sibyl

Sponsor: Appfolio

Team Members:

Sahil Naik (Team Lead)

Rutvik Jha (Scribe)

Jake Yim

Skylar Winig

Robin Wang

Introduction

Background

Property managers collect an influx of accounting data across their portfolio of properties. Currently, millions of data points regarding their assets, liabilities, expenses, income, and equity are hidden deep in databases with no direct insights or gains being made. Generating insights or projections from this vast accounting data requires a lot of overhead and human labor from the property manager's side. As a result, property managers often face uncertainties with their assets as they are restricted in projecting their income and are forced to look at confusing and incomplete personal spreadsheets of their expenditures when trying to get a better understanding of their properties. When building a portfolio, it is crucial to have a comprehensive understanding of financial assets and to have the foresight to predict net gains or losses in incoming months. Sibyl aims to allow Property Managers to extract insights from their vast accounting databases and analyze them in a meaningful way without extra overhead costs, by leveraging Machine Learning and a UX focused Dashboard to visualize ML insights.

Objective

Our objective is to create a machine learning model to forecast property management expenditures and income. We intend to train our model using the vast plethora of property management data that is hidden or confusing to the managers themselves. Since our model will be pretrained on the exact same format and fields that we expect to create real forecasts on, we

will be able to produce accurate projections of the financial data. Based on our projections, property managers will gain insights into which areas of their business to focus on using our user experience focused dashboard. This is the secondary crux of our project. We plan to create a user experience centered dashboard to present our predictions in a clear and concise manner using dynamic graphs and charts. These graphs will be generated using our machine learning model's predictions and will allow property managers to quickly see their past reports and future forecasts to tackle the financial uncertainties in property management.

A stretch goal of ours is to enable other ML oriented teams at AppFolio to easily integrate the insights from their Machine Learning models onto our dashboard. We aim to accomplish this by supporting a wide range of visualization formats (continuous value insights, discrete insights, and graphing projection functions, etc) and to create a clearly documented API to integrate external model insights into our dashboard.

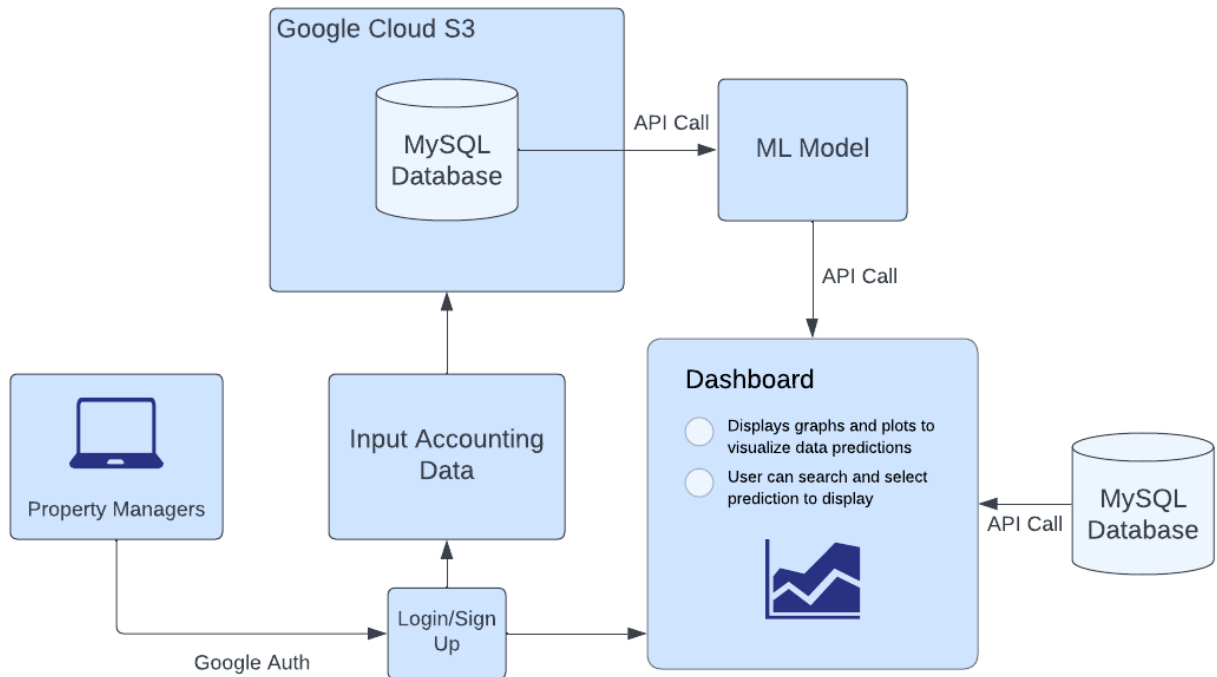
Current Solutions

Currently, the industry standard for visualizing data is modularization by black-boxing machine learning features and interfacing with the frontend components through API calls. We aim to adopt this design by creating a full-stack web application tasked to visualize the dashboard and python web application to generate ML insights. These two independent applications will interface with each other via API calls. This allows us to build a robust, future-proof solution, where future ML models that AppFolio might take on can easily visualize their insights on our dashboard. Or on the other side, future iterations of a dashboard can easily interface with our ML model through similar API calls.

Implementation Choices

We have chosen to develop our application using the Ruby on Rails framework. Our mentoring company AppFolio uses a Ruby-React stack and we hope that our project can eventually be incorporated into Appfolio's products. Using Rails will allow for seamless integration of our application. We have also decided to use React.js to help with the data visualization aspect of our project. React supports libraries such as Highcharts that allow for interactive and dynamic graphing of data within our web application frontend. We are choosing to use MySQL as our database system. MySQL is fast and reliable and can handle storing the accounting data given to us by our users.

System Architecture Diagram



Use Case/User Stories

1. As a user, I can login or sign up through Google OAuth so I can access and view my accounting data and visual projections.
2. As a user, I can recover my password if I forget it by using my email.
3. As a user, I can import my previous accounting information and data through the provided website interface
 - a. Scenario 1: The user imports correctly formatted data and the data is able to be processed through our ML model.
 - i. The website will display the data in a table that can be selected by the user to show predictions in graph form
 - b. Scenario 2: The user imports incorrectly formatted data and the ML application is unable to process the data.
 - i. A pop up window will indicate that the data is incorrect and how the user can correctly import the data.

- c. Scenario 3: The user copies over their database with APM entirely
 - i. The website will allow the user to allow a one-time copy of their accounting data with APM to Sibyl
- 4. As a user, I can choose what predictions to display based on previously imported data.
 - a. Scenario 1: The user selects a prediction our model is capable of producing. The dashboard displays this prediction in the form of a graph.
 - i. Once the prediction is displayed, the user will be able to identify the prediction our model has made.
 - b. Scenario 2: The user selects a prediction and our model is unable to come to a conclusion based on the given data.
 - i. A pop up window will explain the error and how to possibly fix the prediction request or ask the user to import more data
- 5. As a user, I can view the confidence in ML predictions, if it pertains to the particular model
- 6. As a user, I can use the predicted metrics to derive secondary metrics to help me understand the implications of the predicted metrics
 - a. Scenario 1: I can see how the predicted occurrence of maintenance requests impact my cash flow this month
 - b. Scenario 2: Based on the expected time of delivery for rent, I can view my daily account balance projections of next 2 weeks.
- 7. As a user, I would like a clear, non-complicated means to efficiently understand the insights extracted from my data.
 - a. Scenario 1: A UX focused design for the front end and the output of ML models should allow the user to easily understand the insights given to them without needing to be familiar with the complexities of ML models.
- 8. As a user, I can view a wholistic score for my accounting health and how I compare with my peers
 - a. Scenario 1: I can see how my maintenance related expenses compare to other properties
- 9. As a user, I can look at past history and see past trends

10. As a developer, I would like a comprehensive set of visualization formats that support a wide variety of ML models.
 - a. Scenario 1: An ML developer looking to integrate their models onto our visualization dashboard can select from various graphs or projections that display continuous value insights, discrete insights, insights in the form of projection functions, insights from sequential data, etc.
11. As a developer, I would like a clear usage guide for our APIs that allow ML developers to integrate insights from their models onto our visualization dashboard.
 - a. Scenario 1: An ML Developer can refer to a well documented usage guide maintained by the team behind Sibyl to clearly leverage our API to integrate the insights of their ML model with our visualization Dashboard.

Technologies Employed

1. Ruby on Rails
2. MySQL
3. ReactJS
4. GitHub