CLAIM CRAFT

Prototype Introduction

The Future of Insurance Claims Adjusting

By Alex and Yuvraaj

Table of contents

01 Overview

02 Background

03 Adjuster Report And Dashboard

04 Image-to-Text

05 Text-to-Completion

06 Conclusion

01

Claim Craft Overview

Powered by openAI

Claim Craft utilizes OpenAI's GPT-4 for Text-to-Completion and GPT-4-Turbo-Vision for Image-to-Text generation.

Automates Insurance Claims Process

Using AI and OCR technologies Claim Craft streamlines documentation, analysis and recommendations.

Interactive Dashboard

GUI that contains all saved Adjuster Reports

Future implementations will include

- Multifaceted Claim Tracking
- Contractor Portal

02

Background

Why an Insurance App?

- Family ties to insurance industry
- McKinsey research estimates that by 2030 more than half of current claims activities could be replaced by automation
  - $1.1 trillion in estimated annual value for the global insurance industry
  - $300 billion of which from AI-powered customer service offerings
- Lack of clear niche AI tool targeting insurance adjusters

McKinsey & Company

hosta AI

Tractable

Source: https://www.mckinsey.com/industries/financial-services/our-insights/insurer-of-the-future-are-asian-insurers-keeping-up-with-ai-advances

Background

Our aim

To modernize the insurance adjusting process by leveraging AI to improve efficiency, accuracy, and ease the workload, driven by challenges adjusters face in documentation and damage assessments

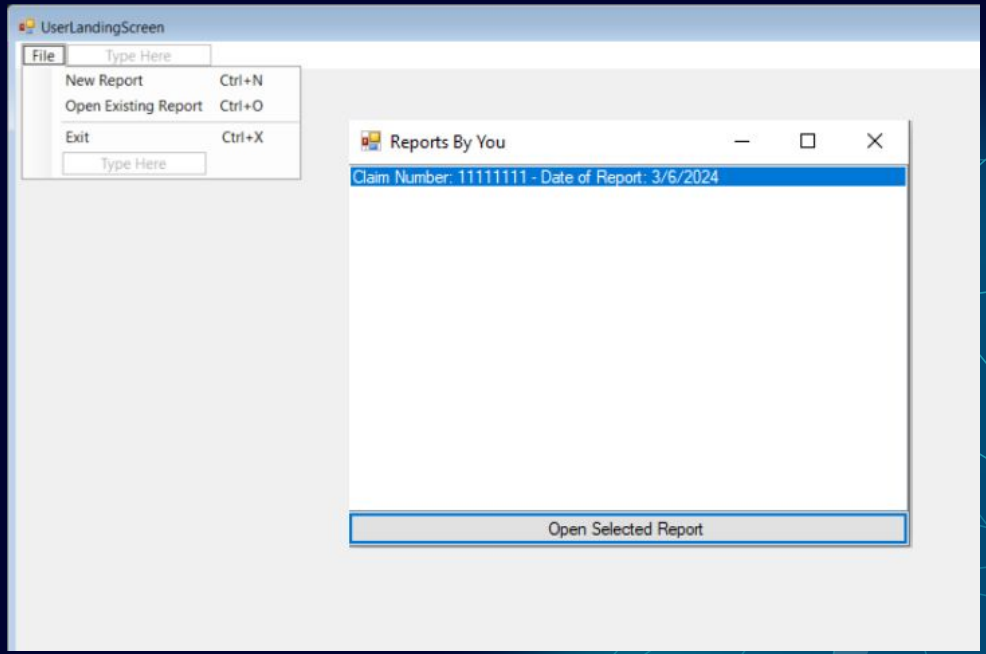
The goal

To simplify and enhance the insurance claim adjustment process for adjusters, streamlining workflows and improving accuracy through AI-generated damage descriptions and policy coverage assessments.

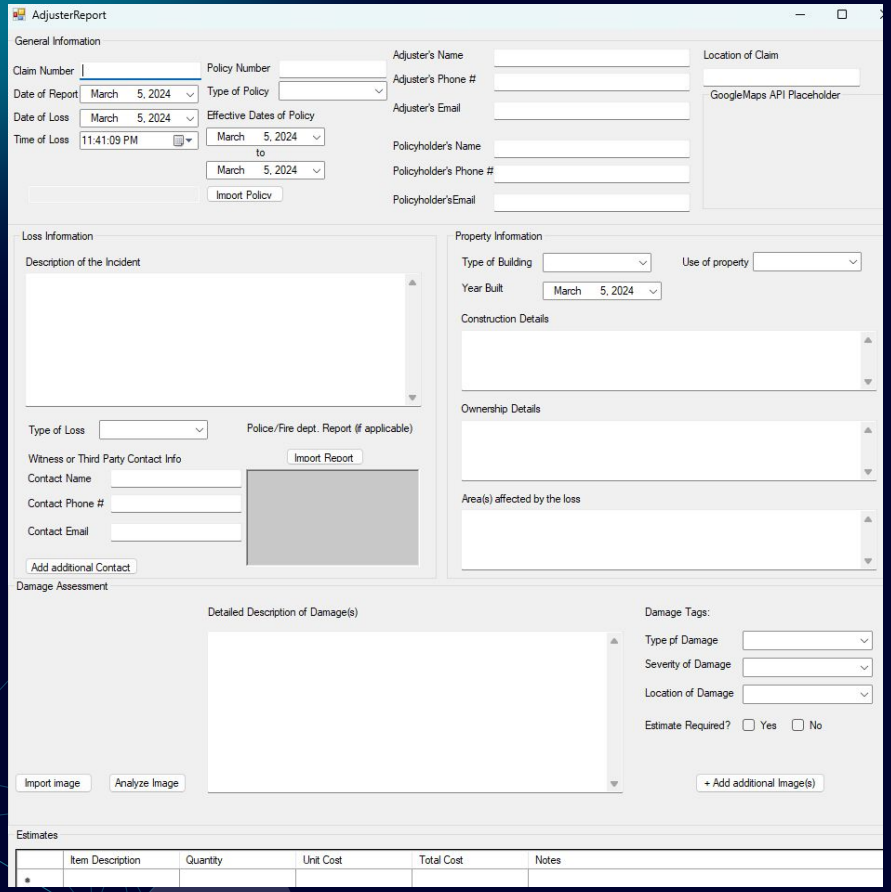
03

Adjuster Report And Dashboard

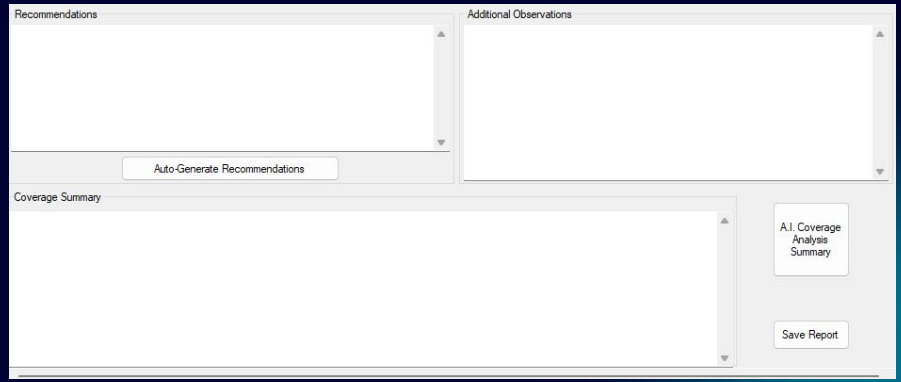
- Uses MongoDB to store Users, session, and adjuster field reports
  - Saves and retrieves form data elements
- Currently in Proof-of-Concept stage
- Future development will include:
  - Claim tracking GUI for incoming, WIP, and completed Adjuster Forms
  - Contractor tendering widget



Adjuster Report And Dashboard

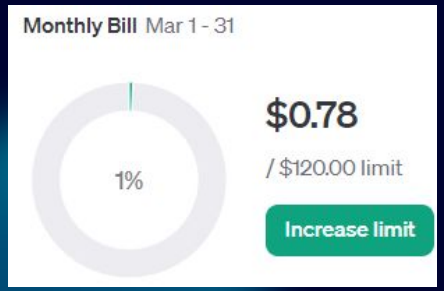


- Main innovation of Claim Craft Insurance app
- AI and OCR driven features
- Contains all standard elements of an insurance adjuster field report



04 Image-to-Text

- Uses GPT-4-Turbo-Vision API to analyze images into text
- Identifies in great detail damages seen in pictures
- Saves hours in administrative costs
- $0.01 per image




```
/// <summary>
/// Asynchronously analyzes the image using the OpenAI API.
/// </summary>
2 references | 0 changes | 0 authors, 0 changes
public async Task<string> AnalyzeImageWithOpenAI(byte[] imageBytes)
{
    // Retrieve the API key using SecretsManager
    string apiKey = secretsManager.GetSecret("OpenAIKey");

    // initialize the OpenAI API
    OpenAIAPI api = new OpenAIAPI(apiKey);

    // Convert byte[] to Base64 String
    string base64Image = Convert.ToBase64String(imageBytes);

    // creates link to model and sets tokens for response
    var chat = api.Chat.CreateConversation();
    chat.Model = Model.GPT4_Vision;
    chat.RequestParameters.MaxTokens = 800;
    chat.RequestParameters.Temperature = 1;

    chat.AppendSystemMessage("You are an insurance adjuster document
    chat.AppendUserInput("In great detail, what is the summary of da
    string response = await chat.GetResponseFromChatbotAsync();
    Console.WriteLine(response);

    return response;
}
```

05 Text-to-Completion

- Uses GPT-4 API to take all form data and provide insurance coverage recommendation and summary
- Converts insurance policies to strings using OCR
- Saves even more hours in administrative costs
- $0.03 per analysis

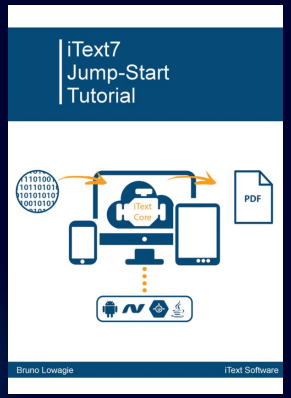
```
// initialize the OpenAI API
OpenAIAPI api = new OpenAIAPI(apiKey);

// creates link to model and sets tokens for response
var chat = api.Chat.CreateConversation();
chat.Model = Model.GPT4_Turbo;

// To be removed once max token chat appending helper function made for extra large chats
chat.AutoTruncateOnContextLengthExceeded = true;

// Sets parameters for API calls
chat.RequestParameters.Model = Model.GPT4_Turbo;
chat.RequestParameters.MaxTokens = 4000;
chat.RequestParameters.Temperature = .5;

chat.AppendSystemMessage($"You are an insurance adjuster documenting damages from an insurance claim. {openAIPrompt}. " +
    $"The following are elements taken directly from the insurance adjusters report: {adjusterFormTextString}. " +
    $"Here is the insureds' insurance policy: {policyTextString}. Please provide a recommendation" +
    $"of if they are covered and a summary given what you know about the insurance claim from the adjuster report.");

string OpenAIPolicyCoverageAnalysis = await chat.GetResponseFromChatbotAsync();

return OpenAIPolicyCoverageAnalysis;
```

05

Text-to-Completion

Other Technologies

- BouncyCastle library used to decrypt pdf file
- iText7 used to parse pdf into .txt file




```
string policyFilePath = insuredPolicyFile;
string insuredPolicyToText;

using (PdfDocument pdfDocument = new PdfDocument(new PdfReader(policyFilePath)))
{
    StringBuilder text = new StringBuilder();
    for (int i = 1; i <= pdfDocument.GetNumberOfPages(); ++i)
    {
        text.Append(PdfTextExtractor.GetTextFromPage(pdfDocument.GetPage(i)));
    }
    insuredPolicyToText = text.ToString();
}
```

06 Conclusion

Lots of opportunity for AI Efficiency in Niche Applications

From insurance adjusting to construction management, AI's adaptability has potential to transform numerous sectors, offering tailored solutions that optimize processes and drive tangible value in specialized domains. This means there's lots of money to be made.

Great learning opportunity to gain prompt engineering experience

As AI continues to impact various industries, the skills honed through developing and refining AI-driven applications become increasingly valuable, positioning engineers for success in a rapidly evolving landscape where AI expertise is in high demand.