JusticeAI Iteration Summary

Iteration 9

Team Members

Name	Student ID	GitHub ID
Lance Lafontaine	26349188	lancelafontaine
Arek Manoukian	21710389	arekmano
Sylvain Czyzewski	27066333	vynny
Mihai Damaschin	27177895	mihaiqc
Samuel Campbell	26457959	samuel-campbell
Taimoor Rana	26436110	taimoorrana1
Zhipeng Cai	21346482	choitwao

Project Summary

JusticeAI (ProceZeus) is a web chat bot that aims to facilitate access to judicial proceedings involving specific domains of law. Users will have the ability to converse with the chatbot, describing in detail the situation for which they wish to pursue litigation. The system, which will leverage the power of machine learning and natural language processing, will guide the user through a process wherein they'll be prompted with a series of questions relating to their potential case allowing the system to ultimately determine, based on provincial jurisprudence, whether the user has a valid case worth pursuing in the judicial system. Alternatively, the system may also suggest remedies in lieu of legal action if it is deemed unlikely to be in the user's best interest.

Velocity

The primary focus of iteration 9 was adding new machine learning prediction functionality for an additional claim category (retaking a rental), refactoring NLP data for better input classification by splitting landlord and tenant data into separate classifiers, and the addition of basic optical character recognition functionality allowing for leases to be parsed for data that will eventually be used to obtain facts, resulting in the user being able to skip answering certain questions normally asked by the chatbot.

During this iteration, we were able to complete **30** story points.

The following is a list of user stories that were completed in Iteration 9:

- #25 - Isolate the Document from a picture and align it (8 pts)
- #259 - Add a question to determine if the beta user is a legal professional (3 pts)
- #261 - Show precedents that are similar to the current claim (8 pts)
- #306 - SPIKE: Investigate & present how the RLQ data may be used (3 pts)
- #312 - DEV Story: Prepare Cyberjustice AI presentation (3 pts)
- #315 - NLP Data Revamp (5 pts)

Plan for Next Iteration

In the next iteration, we will be focusing on improving predictions by asking the user more questions. Rather than only asking a set number of predefined questions, we intend to implement a progress system wherein a user may obtain a prediction after answering a base set of questions, or may continue to answer questions for a more accurate prediction. In addition to this machine learning will be expanded by adding predictions for one more new claim category (tenant expulsion), and the previously finished retaking rental category will be implemented in the NLP system. Finally, will attempt to analyze handwriting using the OCR system to extract data from leases, and provide the user with legal definitions on the client side for more confusing legal terms.

Shown below are the stories that we will be working on for Iteration 10 for a total of **48** points:

- #37 - Legal Definitions (3 pts)
- #43 - Dashboard Reporting of Results (8 pts)
- #44 - Show Conversation Progress % (5 pts)
- #184 - Claim: Tenant Expulsion (8 pts)
- #186 - Claim: Retaking the rental (8 pts)
- #236 - PO STORY: Establish platform on various social medias to acquire FAQ from their users
- #263 - Handwriting Analysis for OCR (8 pts)
- #345 - DEV STORY: Fix CD (bug)
- #346 - I want to answer more questions to obtain a more accurate prediction (8 pts)

Noteworthy Achievements

- NLP intent classification significantly improved by adding more data and splitting tenant and landlord claim categories into separate classifiers

- OCR system is able to detect lease edges and is now able to move on to handwriting analysis
- New data provided by product owner has been analyzed to see if it can potentially be used to improve classification - using that data for the models may provide better results but would slow down the progress of the project significantly
- Upon receiving a prediction, a user now gets a list of similar case numbers which they may lookup to see how the prediction was obtained and how the proceedings went - the content of the proceedings themselves are still under NDA

Technology, Architecture and Library Changes

Iteration 9 marked the beginning of the implementation of OCR functionality into the application. A new microservice (task service) was added along with a few libraries to facilitate the implementation of lease information parsing with OCR.

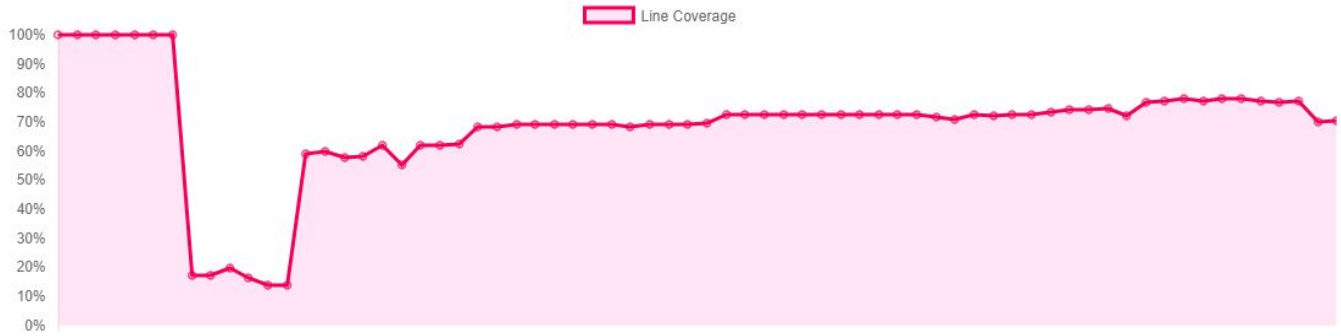
Component		
Tool	Language	Purpose
Task Service		
pytesseract	Python	Convenient Python bindings to the Tesseract library
tesseract	C++	An open-source multi-language OCR engine
OpenCV-Python	Python / C++	A computer vision and image processing library

Continuous Integration Processes and Naming/Coding Changes

There have been no changes to the CI processes or coding conventions in Iteration 9.

Unit Tests and Code Coverage

Line coverage after iteration 9 is currently at 70.39%.



Coverage has dropped by around 7% in this iteration due a couple of factors. Firstly, a major refactoring in the ML service resulted in several files that were used previously and that are not going to be used in the future being removed. Additionally, the introduction of a completely new service, the task service, currently used for OCR, resulted in the introduction of some boilerplate code which reduced our coverage percentage.

Iteration 9 Retrospective

When referring to “NLP” or “ML”, we are referring to the usual team compositions associated with the work on those two domains:
NLP = Natural Language Processing team
ML = Machine Learning team

What went well

- Note taking was very good when taking in criticism
- NLP refactoring went much smoother than previously anticipated
- Machine Learning was found to be very scalable

What went less well

- NLP wasn’t developed as quickly
- NLP doesn’t receive facts from ML fast enough so NLP is always doing work rushed
- Continuous Deployment is still broken

What we can do to fix it

- Allocate more resources to fixing Continuous Deployment