**02/01/2021**

Orange City, FL 32763

**CLM - POC**

**Ver: 1.0, @SSG**

# STAKEHOLDERS

## Internal

SME’s, Partner Channels, Architects, Senior Managers and Engineers

## External

Consultants, External Partners etc. covered under the CL NDA

# 

# OVERVIEW

Contract management is the process of overseeing contracts entered into with vendors, partners, customers, and/or employees to ensure the compliance, accuracy, and validity of commercial agreements. This process is risky and error-prone, and training employees to handle contract administration is complex and expensive.

Research from the International Association for Contract & Commercial Management (IACCM) suggests that 9.2 percent of revenue is lost as a result of poor contract management. Improper contract management can leave revenue holes, create approval delays, decrease productivity, and lead to inaccurate information. In business, this means dissatisfied customers, compliance problems, and ultimately, the loss of business.

In order to avoid these issues, reduce risk, and achieve compliance in contract lifecycle management (CLM), more and more organizations are turning to automation. In fact, implementing contract management software is becoming an essential aspect of every organization’s digital transformation program.



# 

# PURPOSE

TBD

# SCOPE

TBD

# PREREQUISITES

TBD

# Requirements of the Proposed System

1. Use existing contracts to seed and train the proposed system
2. REQUEST Stage - Based on a user request - assemble a document template intelligently with all the required clauses that are required by the requested topic / template-type
3. DRAFT Stage - Allow editing and re-arranging of clauses while maintaining the context of the topic (template type)
4. NEGOTIATION Stage - allow a workflow to facilitate multi user editing while allowing the originator to approve or reject the proposed changes
5. APPROVAL Stage - the proposed system should be able verify the presence of the necessary clauses (which are required as part of the topic / template-type) in their original form or in an edited form (which could have happened during the NEGOTIATION stage).   
   Could Have: Approval flow can be improved by the proposed system by differentiating edited clauses and their implications based on the context / topic / template-type.
6. COMPLIANCE Stage - TBD (not required at this stage)

# PROCEDURE

Notes: These are not the final steps. Details within each steps can vary depending upon the situation, infrastructure availability, performance of created model etc.

1. Contract documents are in text format. All the contracts are available in the directory file system. Directories and files are named properly. Directory structure should take care of document type, country, industry etc
2. Analysis of sections and information under each contract type. To understand document structure, content style, templates types, variation in template of same type, common section in different template etc.
3. **Template Creation**
   1. Using **SpaCy / NLTK / StandfordNLP libraries** perform NER parsing of each contract.
   2. Using different hyper parameters tune these
   3. **Evaluate the performance** of each model and select the best model for final NER parsing working.
   4. Remove NER info from the contract and create a template.
   5. Show the contract on screen. If there is anything which model could not handle then remove that NER text manually. (optional step to ensure 100% correct template)
   6. If this template is not existing in the system, then then save the template in csv file, else without saving template move to next contract.
   7. While saving template use the meta data of folder structures and file name as different attributes in the csv file.
4. **Multi-class Classification Model** (This model will be used to Validate details of new sections in the template )
   1. To create a model for multi class classification scan all the templates
   2. Break each section and section body of the template
   3. Save section, section body, contract type and other information in a another csv file.
   4. Perform pre-processing steps on the csv file (liking cleaning text, ensuring correct record created etc). This can be done using **Pandas or Regex lib.**
   5. Identify how many classes are required. Use this formation to select classification algo.
   6. Tokenization and Embedding of text using **Keras library**
   7. Using **LSTM/ KNN/ Decision Tree/ SVM / NB** create a classification model.
   8. Test the performance of model.
5. **Contract Creation**
   1. Create one user interface which shows type of templates, city, state etc and ask user to select the details to create contracts
   2. A list of templates with preview displayed on the screen. User select one template.
   3. Ask user to enter parties details to create a contract
   4. Using the selected template and party details a new document is created and displayed on the screen.
   5. Everything can be modified in the document except section name
   6. Against each section there is option to delete the section and related body.
   7. This screen also have option to add sections
   8. When section name is entered. Then user has to enter the body of the section.
   9. Sections can be recorded on the screen.
6. **Contract Validatio**n
   1. Screen as one button to validate the contract.
   2. When this bottom is clicked multi-class classification model will be used to check A- Whether the section name is relevant to the contract type or not. B- Details entered in the body of section and section name are in sync or not.
   3. A green to red scale bar will show how consistent contract, section, body info are.
   4. Screen will have a save contract button.
   5. Screen will also have save template button.
   6. If new section is entered or sections are rearranged then this can be saved as new template.
   7. Content of this new template is not part of the MCC Model.
   8. After sometime we can retrain the MCC model to incorporate the new template.
   9. Even without learning new template can be part of preview (as mentioned in previous steps)

# RISKS

Proposing wrong template, assembling the template with unimportant and unrelated verbiage etc could imply legal complications.

# REFERENCES

<https://conga.com/resources/blog/how-clm-software-improves-five-stages-contract-management>