**INTERSHIP PROJECT REPORT**

(PROJECT TERM AUGUST – DECEMBER 2019)

MACHINE LEARNING TEAM

SUBMITTED BY

**DILIP CHALAMALASETTY**

**REG NUMBER: 11611851**

COURSE CODE: CSE 447 – INDUSTRY CO -OP PROJECT -1

**UNDER THE GUIDANCE OF**

KIRTI BALA | ASSISTANT PROFESSOR, LPU

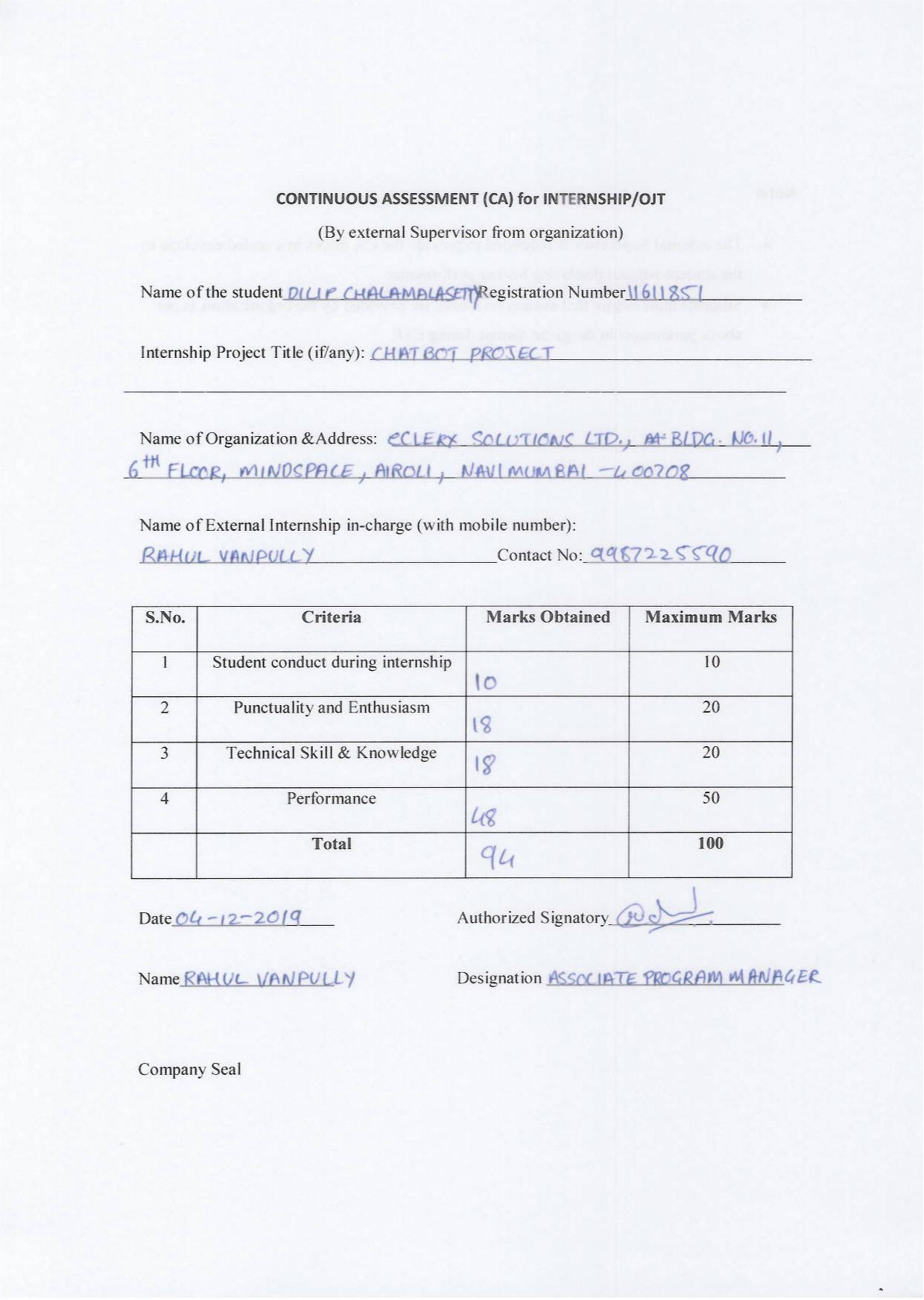
AND

RAHUL VANAPALLY| ASSOCIATE PROGRAM MANAGER, ECLERX

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

A black sign with white text

Description automatically generated



**MARKS GIVEN BY THE EXTERNAL SUPERVISOR**

**DECLARATION**

I hereby declare that the project work entitled “**SARA (Chatbot)**” is an authentic record of our own work carried out as requirements of Internship for the award of B.Tech degree in Computer Science and Engineering from Lovely Professional University, Phagwara, under the guidance of Associate Professor Kirthi Bala and Mr. Rahul Vanapully (Associate Process Manager, Eclerx ) during (August to December 2018). All the information furnished in this internship project report is based on my own intensive work and is genuine.

Name of Student: **DILIP CHALAMALASETTY**

Registration Number: 11611851

**DILIP CHALAMALASETTY**

(Signature of Student)

Date: 30 Nov 2019

**CERTIFICATE**

This is to certify that the declaration statement made by the student is correct to the best of my knowledge and belief. He has progressed well with his internship under my guidance and supervision. The present work is the result of his original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The internship is fit for the submission and partial fulfillment of the conditions for the award of B.Tech degree in Computer Science and Engineering(Hons.) from Lovely Professional University, Phagwara.

**Signature:**

**KIRTHI BALA**

**Associate Professor**

**School of Computer Science and Engineering,** Lovely Professional University, Phagwara, Punjab.

Date: 30 NOV 2019

**ACKNOWLEDGEMENT**

I take this opportunity to express our gratitude and respect to all those who have helped me throughout our working period on the real time company environment. Doing internship in Eclerx help us a lot to understand the new technology and how to grow in a corporate world. My special thanks is to our mentor Mr. Rahul Vanapally(Associate Process Manager), who helped me a lot to show us the right path how to work in a company and to Learn the various aspects of application.

I owe my regards to the entire faculty of the department of Computer Science at LPU from where I learnt the basics of Computer Science and I express my sincere thanks to all our course mates who supported us in the project through various informal discussions which were very valuable to the successful completion of the project.

**-Dilip Chalamalasetty**

Table of Contents

CHAPTER 18

ABOUT ORGANIZATION8

CHAPTER 210

2.1 TECHNOLOGIES USED TO DEVELOP THIS PRODUCT10

2.2 Softwares required to install and each use10

2.2.1 Angular cli11

2.2.1.1 Uses of angular in project10

2.2.2 nodejs and koa server11

2.2.2.1 Uses of nodejs and koa in project10

2.2.3 Elastic search and kibana10

2.2.3.1 uses of elastic search and kibana in project10

2.2.4 Mongodb and robo3t10

2.2.4.1 use of mongodb and robo 3t in project10

2.2.5 Anaconda10

2.2.5.1 Spacy10

2.2.5.1.1 Uses of spacy in project10

2.2.5.2 Rasa10

2.2.5.2.1 File structure in rasa10

2.2.5.2.2 use of rasa in project10

Table of Diagrams

CHAPTER 18

ABOUT ORGANIZATION8

CHAPTER 210

2.1 TECHNOLOGIES USED TO DEVELOP THIS PRODUCT10

2.2 Softwares required to install and each use10

2.2.1 Angular cli11

2.2.1.1 Uses of angular in project10

2.2.2 nodejs and koa server11

2.2.2.1 Uses of nodejs and koa in project10

2.2.3 Elastic search and kibana10

2.2.3.1 uses of elastic search and kibana in project10

2.2.4 Mongodb and robo3t10

2.2.4.1 use of mongodb and robo 3t in project10

2.2.5 Anaconda10

2.2.5.1 Spacy10

2.2.5.1.1 Uses of spacy in project10

2.2.5.2 Rasa10

2.2.5.2.1 File structure in rasa10

2.2.5.2.2 use of rasa in project10

**SECTION 1: INTRODUCTION**

**About Organization**

Eclerx helps businesses work smarter by its innovative business process management, change management, data-driven insights and advanced analytics powered by subject matter experts and smart automation. And it has Eclerx Digital which is the trusted brand of choice to worlds largest global brands for creative production, eCommerce/web operation and analytics and insights service.

It has digital delivery employees at five production hubs which are there in Mumbai, Pune, Chandigarh, Verona and Phuket. It was founded in 2000 by Anjan Malik and PD Mundhra.

eClerx provides critical business operations services to over fifty global fortune 500 client, including several of the world leading companies across financial services, cable and telecommunication, retail, fashion, media, & entertainment, travel and leisure,software and high-tech.

eClerx Digital team of 3000+ full-time digital delivery employees at our five production hubs in Mumbai, Pune, Chandigarh, Verona and Phuket apply deep digital expertise to effectively support the most demanding global clients by employing a follow the sun delivery model. eClerx Digital’s innovative delivery model drives the “metrics that matter” for our clients: improved acquisition, conversion and retention and overall lifetime value of your customer 24x7x365.

eClerx Marketing- Maximize your marketing efforts to build awareness, generate leads,and increase sales.

eClerx eCommerce- Remove the friction from the user experience to reduce cart abandonment and increase average order value

.

eClerx Business Intelligence & Analytics - Make sense of and use your data for personalization, consistently accurate forecasting, and to direct resources towards profitable activities.

The industries Eclerx serve include: financial services, cable and telecommunications, retail, fashion, media and entertainment, manufacturing, travel and leisure, software and high tech.

The popular clients which Eclerx provides its products and services are Dell, Citibank, Autodesk, HSBC, Paypal, Adobe, Timberland, Comcast, etc.

**CHAPTER 2: INTRODUCTION TO PROJECT UNDERTAKEN**

**SARA** isa chatbotthat is on development phase. It is developing in a way that it is helpful for end customers interactions to answer their questions. Our Team is trying to make it more interactive so that the end customers will benefit from it. Its services are provided from Eclerx client website.

**2.1 TECHNOLOGIES USED TO DEVELOP THIS PRODUCT:**

|  |  |
| --- | --- |
| CATEGORY | TECHNOLOGIES |
| WEB DEVELOPMENT | Angular 8, Nodejs, Koa server |
| NLP AND NLU | Spacy, Rasa |
| NO SQL DATABASES | Elasticsearch, MongoDB |

**2.2 SOFTWARE REQUIRED TO INSTALL:**

* Angular 8 CLI
* Node runtime environment
* Elastic search
* Kibana
* MongoDB
* Robo 3t
* Anaconda

**2.2.1 Angular CLI:** This Framework is used to design the web layout for the project, Typescript is the preferred language in the Angular 8. Advantages are mentioned below

* Component-based architecture that provides a higher quality of code
* Google Long-Term Support
* Seamless updates using Angular CLI
* High Performance
* Loved by millions of developers
* Unit-test friendly
* Reusability

**2.2.1.1 Use of this software in project:**

* By using this framework, we create the interface of the chatbot.
* Routing played the key role in navigating and rendering the components using router outlets.
* Sharable data is accessible through user data services.
* Created ts (type script) files for every component to implement the functionality.
* Imported plugins for making chats and api connections
* Create the admin portal for the chatbot, through which admin can see all the chatlogs of all the users.
* Admin portal is divided into four sections
  1. Landing home page
  2. Dashboard
  3. Chatlogs
  4. Statistics
  5. Priority messages
  6. Profile page

**2.2.2 Node Runtime environment:** Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

* nodejs provide the non-blocking I/O
* Initially javascript is restricted to client-side programming only which is run by the browser only but nodejs changes everything about the javascript giving javascript power of programming both for client and server side.
* Node js supports npm packages which attracts a greater number of developers to develop packages for the nodejs.

**2.2.2.1 Use of nodejs and Koa in Project:**

* By using nodejs we made the api for chatbot appilication
* List of api’s
  + Login – which made available by running the nodejs and koa server and accepts get and posts requests from the angular.
  + Register – Which is used to register the user, The user data is taken from the register form which is made by using angular, and the data is post to the api url .
  + Chatlogs – This api returns the chatlogs of user.

**2.2.3 Elastic search and kibana**

* Elastic search is no sql database
* Whie indexing the documents it offers many different options for the developer as compared to other sql databases.
* Kibana is environment for writing and executing the elastic search queries

**2.2.3.1 Use of elastic search in our project:**

* Aggregations concept of elastic search mainly bucket aggregation is used to index the document indexed as a number of groups based in the criteria.
* While users typing the message in the text box which is developed by the angular ,elastic search server triggered with query every single second to get the best suited suggestions to the user and display that suggestions as an pop attached window which is very helpful to the chatbot user .

**2.2.4 Mongodb and robo3t:**

* Mongodb is an nosql database
* Robo3t is the environment in which we can write the query and see all the documents and collections.

2.2.4.1 Use of MongoDB in our project:

* Answer , questions, chatlogs all are stored in the mongodb.
* Admin data also stored in mongodb
* All the apis which are designed by using node js used to store the information in mongodb only by using mongoose package.

**2.2.5 Anaconda**: Anaconda come with huge number of pavkages the two most important packages which our team used in the project is Spacy and Rasa.

**2.2.5.1 About Spacy**

* Spacy package is NLP package is mostly used for pre-processing of the text like
  1. Tokenization
  2. Parts of Speech Tagging
  3. Dependency Parsing
  4. Lemmatization
  5. Named entity detection

**2.2.5.1.1 Tokenization:**

* Segmenting text into words, punctuations marks etc.
* There are different kinds of tokenizers that are supported by spacy comity, some of them are listed below.
  1. White space tokenizer
     + The whitespace tokenizer divides text into terms whenever it encounters any whitespace character.
  2. Standard tokenizer
     + The standard tokenizer divides text into terms on word boundaries, as defined by the Unicode Text Segmentation algorithm. It removes most punctuation symbols. It is the best choice for most languages.
  3. Letter tokenizer
     + The letter tokenizer divides text into terms whenever it encounters a character which is not a letter.
  4. Lowercase tokenizer etc
     + The lowercase tokenizer, like the letter tokenizer, divides text into terms whenever it encounters a character which is not a letter, but it also lowercases all terms.

**2.2.5.1.2 Parts of Speech Tagging**

* Assigning word types to tokens, like verb or noun.
* Spacy have three kind of models en\_core\_web\_sm, en\_core\_web\_md, en\_core\_web\_lg the size of the models increases in the increases order the main difference between each model is the features provided.
* Pos tagging is the process of identifying the parts of speech of the particular word this tag is identified by the model which is previously trained on the large amount of the data
* These parts of speech tag is very useful for further text pre-processing steps.

**2.2.5.1.3 Dependency Parsing**

* Assigning syntactic dependency labels, describing the relations between individual tokens, like subject or object.

import spacy

nlp = spacy.load("en\_core\_web\_sm")

doc = nlp("Apple is looking at buying U.K. startup for $1 billion")

for token in doc:

    print(token.text, token.pos\_, token.dep\_)

**output:**

Apple PROPN nsubj

is AUX aux

looking VERB ROOT

at ADP prep

buying VERB pcomp

U.K. PROPN compound

startup NOUN dobj

for ADP prep

$ SYM quantmod

1 NUM compound

billion NUM pobj

**2.2.5.1.4 Lemmatization:**

* Assigning the base forms of words. For example, the lemma of “was” is “be”, and the lemma of “rats” is “rat”.

import spacy

nlp = spacy.load("en\_core\_web\_sm")

doc = nlp("Apple is looking at buying U.K. startup for $1 billion")

for token in doc:

    print(token.text, token.lemma\_, token.pos\_, token.tag\_, token.dep\_,

            token.shape\_, token.is\_alpha, token.is\_stop)

* **Output**

TEXT        LEMMA   POS     TAG     DEP     SHAPE   ALPHA   STOP

Apple       apple   PROPN   NNP     nsubj   Xxxxx   True    False

is          be      VERB    VBZ     aux     xx      True    True

looking     look    VERB    VBG     ROOT    xxxx    True    False

at          at      ADP     IN      prep    xx      True    True

buying      buy     VERB    VBG     pcomp   xxxx    True    False

U.K.        u.k.    PROPN   NNP     compoundX.X.    False   False

startup     startup NOUN    NN      dobj    xxxx    True    False

for         for     ADP     IN      prep    xxx     True    True

$           $       SYM     $       quantmod $      False   False

1           1       NUM     CD      compound d      False   False

billion     billion NUM     CD      pobj    xxxx    True    False

**2.2.5.1.5 Named Entity Detection:**

A named entity is a “real-world object” that’s assigned a name – for example, a person, a country, a product or a book title. spaCy can **recognize**[various types](https://spacy.io/api/annotation#named-entities) of named entities in a document, by asking the model for a **prediction**. Because models are statistical and strongly depend on the examples they were trained on, this doesn’t always work perfectly and might need some tuning later, depending on your use case.

**2.2.5.1.6 Use of Spacy in Project:**

* Spacy is used to in the pipeline of the rasa.
* All the pre processing work will be handled by the spacy.
* All the pre-processing steps are mentioned above.
* We should need to configure depending upon our need.

**2.2.5.2 About Rasa:**

* Rasa is the NLU (Natural Language Understanding) package which is mainly used to find the intent, entity extraction and find the action that need to take.

**2.2.5.2.1 File structure of Rasa :**

* Domain.yml – this file contains all the list of intents, entities, slots, templates, forms and etc, these are only declared in this section.
* Data/core/stories.md – This is the core data which on our model will be trained , this consists of blocks of stories each block consists of

Story name

\*intent 1

-action need to be taken

\*intent 2

- action need to be taken

* Data/nlu/nlu.md – This file consists of all the sentences that our chatbot need to face in future in simple words these are the questions which chatbot handles and answers, this file is also used to train our model at training state.

The main use of this file in our project is it gives intent classification ability to our rasa model.

* Actions.py – this is file where we write code to our customizable actions, like switching on the electrical appliances , booking ticket, and etc..
* Endpoints.yml – this is the files where we configure the endpoints like for example mongodb endpoint which is tracker store which is used to store the chat logs of all the users directly after the chat.and we also configure our customize actions server in this file only.
* Config.yml – This is the Important file to the rasa project, this contains the pipeline of the project, pipeline means what are the different stages that the user inputted message goes beforing answered by bot.