

Transformant Processos de Negoci a Chatbots

Mohamed Benali Bendahmane

Director: Josep Carmona Vargas

Codirector: Lluís Padro Cirera

Departament del directors: Computer Science (CS)

Tutor de GEP: Barrabes Naval Fernando

Data de defensa: 03/07/2020 10:00

Centre: Facultat d'Informàtica de Barcelona (FIB)

Universitat: Universitat Politècnica de Catalunya (UPC)

Treball de Fi de Grau pertanyent al Grau en Enginyeria Informàtica, a l'especialitat d'Enginyeria del Software.

Quadrimestre Primavera 2019-2020



Resum

Els models BPMN (Business Process Modeling and Notation), els quals permeten representar els processos de negoci d'una empresa, no resulten fàcils d'entendre per part d'aquelles persones no qualificades. En l'àmbit laboral, la majoria de treballadors no tenen els coneixements per entendre'ls. Això fa que es requereixi algú que expliqui el funcionament del negoci. A més, no sempre és possible tenir algú a qui preguntar un dubte, el que pot causar que un treballador no sàpiga que fer. La problemàtica dels BPMN també pot aplicar-se a treballadors que sí entenen els models. Per exemple, un alt càrrec, que vulgui consultar un detall d'un determinat procés, pot perdre molt de temps buscant entre molts models per tal de trobar el que busca.

L'objectiu general del projecte és crear una aplicació que transformi processos de negoci en assistents virtuals. Concretament, el projecte construeix un prototip fent ús de les tecnologies més punteres i avalua les tecnologies segons la flexibilitat del llenguatge obtinguda. En aquest context, la flexibilitat del llenguatge s'entén com a la diversitat del vocabulari, que pot utilitzar una persona, en referir-se a un mateix concepte.

El prototip selecciona un model BPMN, el processa i genera frases de resposta. A més, genera frases amb significat similar, per tal d'augmentar la flexibilitat del llenguatge. Un cop acabat el processat, s'utilitza Dialogflow (una eina de modelatge d'assistents virtuals) per tal de construir l'assistent virtual. Finalment i durant una conversa amb l'assistent, s'utilitza un servidor per a emmagatzemar la informació pertinent.

Tenint en compte el prototip construït, el projecte conclou que dialogflow és una bona eina pel que fa a la generació d'assistents virtuals a partir d'un model BPMN. Tot i això, si es vol aconseguir un bon resultat en la flexibilitat del llenguatge, s'ha de dedicar grans esforços en la generació de frases amb significat similar. El projecte conclou que es requereix més recerca per aconseguir una major flexibilitat en el llenguatge.

Resumen

Los modelos BPMN (Business Process Modeling and Notation), los cuales permiten representar los procesos de negocio de una empresa, no resultan fáciles de entender por parte de aquellas personas no cualificadas. En el ámbito laboral, la mayoría de trabajadores no tienen los conocimientos para entenderlos. Esto hace que se requiera alguien que explique el funcionamiento del negocio. Además, no siempre es posible tener a alguien a quien preguntar una duda, lo que puede causar que un trabajador no sepa que hacer. La problemática de los BPMN también puede aplicarse a trabajadores que sí entienden los modelos. Por ejemplo, un alto cargo, que quiera consultar un detalle de un determinado proceso, puede perder mucho tiempo buscando entre muchos modelos a fin de encontrar lo que busca.

El objetivo general del proyecto es crear una aplicación que transforme procesos de negocio en asistentes virtuales. Concretamente, el proyecto construye un prototipo en uso de las tecnologías más punteras y evalúa las tecnologías según la flexibilidad del lenguaje

obtenida. En este contexto, la flexibilidad del lenguaje se entiende como la diversidad del vocabulario, que puede utilizar una persona, al referirse a un mismo concepto.

El prototipo selecciona un modelo BPMN, lo procesa y genera frases de respuesta. Además, genera frases con significado similar, a fin de aumentar la flexibilidad del lenguaje. Una vez terminado el procesamiento, se utiliza Dialogflow (una herramienta de modelado de asistentes virtuales) para construir el asistente virtual. Finalmente y durante una conversación con el asistente, se utiliza un servidor para almacenar la información pertinente.

Teniendo en cuenta el prototipo construido, el proyecto concluye que dialogflow es una buena herramienta en cuanto a la generación de asistentes virtuales a partir de un modelo BPMN. Sin embargo, si se quiere conseguir un buen resultado en la flexibilidad del lenguaje, se debe dedicar grandes esfuerzos en la generación de frases con significado similar. El proyecto concluye que se requiere de más investigación para conseguir una mayor flexibilidad en el lenguaje.

Abstract

Business Process Modeling and Notation (BPMN) models, which allow you to represent the business processes of a company, are not easy to understand by those who are not qualified. In the workplace, most workers do not have the knowledge to understand them. This requires someone to explain how the business works. Also, it is not always possible to have someone to ask a question to, which can cause a worker to not know what to do. The problem of BPMNs can also be applied to workers who do understand the models. For example, a senior official, who wants to consult a detail of a certain process, may waste a lot of time searching among many models in order to find what he is looking for.

The overall goal of the project is to create an application that transforms business processes into virtual assistants. Specifically, the project builds a prototype using the most advanced technologies and evaluates the technologies according to the flexibility of the language obtained. In this context, the flexibility of language is understood as the diversity of vocabulary, which a person can use, when referring to the same concept.

The prototype selects a BPMN model, processes it, and generates response sentences. In addition, it generates sentences with similar meanings, in order to increase the flexibility of the language. Once the processing is complete, Dialogflow (a virtual assistant modeling tool) is used to build the virtual assistant. Finally, during a conversation with the wizard, a server is used to store the relevant information.

Given the prototype built, the project concludes that dialogflow is a good tool for generating virtual assistants from a BPMN model. However, if you want to get a good result in language flexibility, you have to put a lot of effort into generating sentences with similar meaning. The project concludes that more research is required to achieve greater flexibility in language.

Acrònims

AIML - Artificial Intelligence Markup Language

API - Interfície de programació d'aplicacions.

BFS - Breadth-First Search

BPMN - Business Process Model and Notation

FIB - Facultat d'Informàtica de Barcelona.

HTTP - Hypertext Transfer Protocol

JAR - Java Archive

IDE- Integrated development environment.

SQL - Structured Query Language

UML - Unified Modeling Language

UPC - Universitat Politècnica de Barcelona.

TFG - Treball Final de Grau.

Glossari

Assistent virtual: Programa normalment instal·lat en un telèfon intel·ligent, que és capaç d'oferir serveis i realitzar tasques que li demana l'usuari.

Backtracking: Algorisme per trobar totes les combinacions.

Cache: Memòria ràpida en la qual s'emmagatzema les dades que són llegides amb més freqüència.

Camunda: Plataforma de codi obert, que permet modelar processos de negoci. També té una eina que permet processar-los.

Chatbot: Programa que simula mantenir una conversa amb un ésser humà.

Coverage del codi: Mesura utilitzada per descriure la proporció de codi cobert per les proves de funcionalitats.

Dialogflow: És un desenvolupador de tecnologies d'interacció entre humans i ordinadors de Google, basat en converses de llenguatge natural.

Docker: Un docker facilita el desplegament d'aplicacions encapsulant-les dintre de contenidors, anomenats dockers, proporcionant així una capa que permet executar l'aplicació en qualsevol sistema operatiu.

Fork (github): Còpia d'un repositori GitHub.

Framework: Entorn de treball que facilita el desenvolupament del codi.

Freeling: *Es tracta* d'una llibreria que permet analitzar el llenguatge.

Fulfillment (dialogflow): En el context on un usuari parla amb un chatbot, cada interacció amb entre ells causarà que es cridi un servei extern per decidir què fer.

Gantt: El diagrama de Gantt és una forma de representar una planificació.

Google Assistant: Assistant virtual desenvolupat per Google.

Heroku: És una plataforma que ofereix servidors remots (en internet) als desenvolupadors. Aquests poden personalitzar, i utilitzar els servidors per a executar-hi una aplicació.

Intent: En una conversa amb un chatbot, significa la intenció de l'usuari.

Java: És un llenguatge de programació, orientat a objectes, que ofereix una alta capacitat de manteniment de codi. El llenguatge és molt usat per la seva alta reutilització de codi.

JUNIT: Llibreria que serveix per provar funcionalitats

Llibreria: Conjunt de funcionalitats executables agrupades en un contenidor.

Patró: Tècnica de programació per resoldre problemes recurrents

PostgreSQL: És un software que implementa un sistema gestor de bases de dades relacionals. És una base de dades robusta i eficient. Igual que tots els gestors de bases de dades, PostgreSQL implementa tots els elements SQL relacionals.

SimpleNLG: Es tracta d'una eina la qual permet generar frases senzilles a partir d'una anàlisi sintàctic

SpinnerChief: Servei que donada una frase, dona frases sinònimes

Stakeholder: Persones o entitats que es veuen afectades per un projecte

String: Cadena de caràcters.

Testing: Proves de funcionalitats

Xatkit: És una eina de modelatge de chatbots.

Índex

1. Introducció i contextualització	10
1.1 Introducció	10
1.2 Identificació del problema	10
1.2.1 Transformació d'un model a un agent de conversa	11
1.2.2 Complexitat dels models BPMN	11
1.3 Stakeholders	11
1.3.1. Desenvolupador	11
1.3.2. Directors del projecte	12
1.3.3. Tutor de GEP	12
1.3.4. Actors beneficiats	12
2. Justificació de l'alternativa de resolució escollida	13
2.1 Treballs i productes relacionats	13
2.1.1 Manpower i similars	13
2.1.2 LoyJoy	13
2.1.3 Treball del mateix grup	14
2.2 Conclusions	14
3. Abast i obstacles del projecte	16
3.1 Abast	16
3.1.1 Objectius generals	16
3.1.2 Sub-objectius	16
3.1.3 Requisits no funcionals	17
3.2 Obstacles	17
3.3 Riscos	18
4. Planificació temporal inicial	19
4.1 Introducció	19
4.2 Descripció de les tasques	19
4.2.1 - Gestió del projecte	19
4.2.2 - Anàlisis	19
4.2.3 - Disseny, implementació i prova de les funcionalitats	20
4.2.4 - Documentació	20
4.2.5 - Seguiment	20
4.3 Recursos	22
4.3.1 Recursos humans	22
4.3.2 Recursos materials	22
4.3 Gestió del risc	23
4.4 Diagrama de Gantt	24
5. Gestió econòmica	27
5.1 Costos de Personal per Activitat (CPA)	27

5.2 Costos Genèrics (CG)	28
5.2.1 Hardware	29
5.2.2 Software	29
5.2.3 Altres costos	29
5.3 Contingències	29
5.4 Imprevistos	30
5.5 Costos totals	30
5.6 Control de gestió	30
6. Sostenibilitat i compromís social	32
6.1 Autoavaluació	32
6.2 Dimensió econòmica	32
6.3 Dimensió ambiental	33
6.4 Dimensió social	33
7. Planificació final i desviacions	34
8. Models BPMN i Assistants virtuals	36
8.1 Models BPMN	36
8.1.1 Cobertura del BPMN dintre del prototip	36
8.1.1.1 Participants	36
8.1.1.2 Esdeveniment inicial	36
8.1.1.3 Tasca	37
8.1.1.4 Flux de seqüència	37
8.1.1.5 Passarel·la exclusiva	38
8.1.1.6 Passarel·la paral·lela	39
8.1.1.7 Esdeveniment intermedi de missatge	39
8.1.1.8 Flux de missatge	40
8.1.1.9 Esdeveniment final	40
8.1.2 Exemple	40
8.2 Assistents virtuals	41
9. Arquitectura general del sistema	43
9.1 Fase de generació del chatbot	43
9.2 Fase d'execució del chatbot	44
9.3 UML	44
10. Anàlisis d'opcions	48
10.1 Lectura del model BPMN	48
10.2 Generació de frases	49
10.2.1 Introducció	49
10.2.2 Analitzador de frases	50
10.2.3 Construcció de frases	51
10.2.4 Generació de frases similars	52
10.3 Generació del Chatbot	54

10.3.1 DialogFlow	54
10.3.2 Xatkit	56
10.3.3 Luis	56
10.3.4 Decisió inicial: Xatkit	56
10.3.5 Nova decisió: DialogFlow	57
11. Recursos informàtics utilitzats	58
11.1 Llenguatge de programació	58
11.1.1 Java	58
11.1.2 SQL	59
11.2 Eines	59
11.2.1 IntelliJ	59
11.2.2 Maven	59
11.2.3 Spring Boot	59
11.2.4 Heroku	60
11.2.5 PostgreSQL	60
11.2.6 Docker	60
12. Generació del chatbot	61
12.1 Selecció del BPMN	61
12.2 Generació d'intencions	63
12.2.1 Tractar participants	63
12.2.1.1 Extracció d'informació per defecte	64
12.2.1.2 Esdeveniment inicial	66
12.2.1.3 Tasca	66
12.2.1.4 Passarel·la exclusiva	68
12.2.1.5 Passarel·la paral·lela	68
12.2.1.6 Esdeveniment intermedi	70
12.2.1.7 Esdeveniment final	70
12.2.2 Generar intencions per defecte	70
12.2.3 Generar consultes generals	71
12.3 Generació de frases	71
12.3.1 Analitzador de frases	72
12.3.2 Construcció de frases	72
12.3.3 Generació de frases similars/d'entrenament	74
12.4 Entrenament del chatbot	76
13. Execució del chatbot	77
14. Testing	78
15. Resultats	80
15.1 Zoo.bpmn	80
15.2 Employee	83
15.3 Cache.bpmn	86

15.4 Conclusions dels resultats obtinguts	88
16. Conclusions	90
17. Treball futur	91
18. Adaptació de les competències tècniques	93
18.1 CES1.2:	93
18.2 CES1.3:	93
18.3 CES1.7:	94
18.4 CES1.9:	94
18.5 CES2.1:	94
18.6 CES2.2:	95
18.7 CES3.1:	95
Apèndix A	96
Fitxers adjunts	96
Estructura de la carpeta	96
Requeriments	97
FreeLing	97
SpinnerChief	97
DialogFlow	97
JAR	97

1. Introducció i contextualització

En aquest apartat s'explica el context del projecte. Concretament, una introducció, definició de termes, formulació del problema i stakeholders.

1.1 Introducció

El projecte “**Transformant Processos de Negoci a Chatbots**” és un Treball Fi de Grau del Grau en Enginyeria en Informàtica el qual està enfocat a l'especialitat de l'Enginyeria del Software. El treball es realitza a la FIB¹, en la modalitat A (Centre). Es tracta d'un treball orientat a la recerca, dirigit pel grup “Process and Data Science”² del departament de Computer Science, sobre la generació de Chatbots a partir de Processos de Negoci.

El projecte consisteix en desenvolupar un algorisme de transformació d'un procés formal de negoci en un agent de conversa. Aquest TFG neix a partir d'un treball similar, en el mateix grup de recerca, el qual tenia com a problema la falta de flexibilitat en el llenguatge. Aquest TFG busca solucionar aquest problema.

Pel que fa a la visió final, la idea és que un empleat (que no té els coneixements per entendre documentació tècnica) pugui utilitzar el chatbot per tal d'informar-se dels processos que duu a terme l'empresa on treballa. L'empleat també podria fer consultes sobre tasques específiques.

Donat que la visió final sobrepassa l'abast del projecte, aquest es centra a realitzar una prova de concepte on s'utilitza les tecnologies més punteres (pel que fa a la generació de chatbots) i es valora la seva aportació pel que fa a la flexibilitat en el llenguatge obtingut.

Per tant, aquest projecte consisteix en desenvolupar un algorisme, de transformació d'un procés formal de negoci en un agent de conversa, i en valorar la flexibilitat en el llenguatge a l'hora de comunicar-se amb l'assistent virtual.

1.2 Identificació del problema

Aquest TFG té dos enfocaments. Per una banda, té una component de recerca on es busca transformar, automàticament i amb garanties, qualsevol model BPMN en un agent de conversa. Per l'altra banda, busca simplificar i facilitar l'enteniment dels models BPMN.

¹ "Facultat d'Informàtica de Barcelona |." <https://www.fib.upc.edu/ca>. S'hi ha accedit el dia 15 de maig. 2020.

² "PADS-UPC - Computer Science Department." <https://www.cs.upc.edu/~pads-upc/> . S'hi ha accedit el dia 15 de maig. 2020.

1.2.1 Transformació d'un model a un agent de conversa

L'estàndard BPMN (Business Process Modeling and Notation) permet representar els processos que duu a terme una empresa independentment de les activitats que aquesta faci. Aquests models són formals, i per tant, una màquina els pot interpretar. Tot i això, els models BPMN contenen text, el qual és més complicat d'interpretar.

Convertir, automàticament, un model BPMN en un assistent virtual, de manera que aquest entengui a l'usuari, és complicat. Hi ha diverses causes per les quals l'assistent virtual pot no comportar-se adequadament. En primer lloc, l'usuari pot formular un dubte de moltes formes, incloses expressions informals de carrer. En segon lloc, el model BPMN pot no seguir fidelment els estàndards. Finalment, reconèixer la intenció d'un usuari a partir d'una frase no és suficient, es requereix un context.

Tot plegat, sumat a la falta de bones eines de modelatge d'assistents virtuals, causa que es dificulti molt la generació automàtica de chatbots a partir de models BPMN.

1.2.2 Complexitat dels models BPMN

Els models BPMN (Business Process Modeling and Notation) no resulten fàcils d'entendre per part d'aquelles persones no qualificades. En l'àmbit laboral, la majoria de treballadors no tenen els coneixements per entendre'ls. Això fa que es requereixi algú que expliqui a la persona (sigui un client o un nouvingut) el funcionament del negoci. A més, no sempre és possible tenir algú a qui preguntar un dubte, el que pot causar que el treballador no sàpiga que fer.

La problemàtica dels BPMN també pot aplicar-se a treballadors que sí entenen els models. Per exemple, una gran empresa té molts processos en diverses àrees. A més, aquests processos poden tenir una complexitat elevada. Un alt càrrec, que vulgui consultar un detall d'un determinat procés, pot perdre molt de temps buscant entre molts models per tal de trobar el que busca.

1.3 Stakeholders

Per a cada stakeholder, es descriu el seu rol i el seu objectiu.

1.3.1. Desenvolupador

El rol de desenvolupador el realitza jo mateix, en Mohamed Benali Bendahmane.

- **Rol:** La meua feina consisteix en analitzar els BPMN, aprendre a crear assistents virtuals i veure si és viable convertir un model BPMN en un assistent virtual amb garanties. També dissenyar i desenvolupar el codi. Finalment, també gestiono d'aquest projecte i escric la memòria.
- **Objectiu:** Poder realitzar el meu rol satisfactòriament

1.3.2. Directors del projecte

Els directors del projecte són Josep Carmona i Lluís Padró, professors del Departament de CS (Computer Science) de la Universitat Politècnica de Catalunya³.

- **Rol:** Supervisar el projecte i donar ajut a l'estudiant, sobre el camp en el qual es treballa, en cas que sigui necessari.
- **Objectiu:** Que el projecte sigui satisfactori

1.3.3. Tutor de GEP

El tutor de GEP és en Barrabes Naval Fernando.

- **Rol:** Supervisar i puntuar la documentació de la fase inicial. També dona consells a l'alumne
- **Objectiu:** Ajudar a l'alumne a perfilar el seu TFG.

1.3.4. Actors beneficiats

Els actors beneficiats són alts càrrecs, treballadors sense coneixement de models BPMN i el grup de recerca.

Alts càrrecs de grans empreses, que si entenen els models BPMN.

- **Rol:** Fer ús de l'assistent virtual com a client
- **Objectiu:** No perdre el temps buscant un procés d'entre centenars de models.

Treballadors que no entenen els BPMN i no tenen ningú a qui acudir podran fer ús de l'aplicació per conversar i entendre el procés que estan realitzant.

- **Rol:** Fer ús de l'assistent virtual com a client
- **Objectiu:** Entendre millor el funcionament dels processos que es duen a terme.

El grup on es realitza el TFG és veu beneficiat per la feina feta.

- **Rol:** Aprofita la feina feta per les seves investigacions.
- **Objectiu:** Utilitza la feina que es realitzi en aquest TFG, en les investigacions que porta el propi grup.

³ "UPC - UPC Universitat Politècnica de" <https://www.upc.edu/ca>. S'hi ha accedit el dia 18 de juny. 2020.

2. Justificació de l'alternativa de resolució escollida

Aquest TFG és una prova de concepte i consisteix en transforma un model BPMN en un chatbot intentant millora la flexibilitat del llenguatge.

Per tal de fer el millor treball possible és molt important familiaritzar-se amb els estudis previs o productes alternatius. Per això, s'ha fet una anàlisi dels diferents treballs existents en el sector. És important destacar que cap producte fa el mateix que el d'aquest projecte.

2.1 Treballs i productes relacionats

A continuació s'exposen els productes relacionats, concretament Manpower, LoyJoy i el treball realitzat anteriorment en el mateix grup que impulsa el projecte.

2.1.1 Manpower i similars

Manpower⁴ és una empresa que ofereix uns serveis de reclutament i connecta a més de 600,000 persones, el que vol dir que gestiona una base de dades enorme.

A causa de les dades que Manpower té, l'empresa rep moltes consultes diàries d'empleats respecte a la seva situació laboral, el que genera una gran quantitat de sol·licituds. El seu assistent virtual respon a les preguntes més freqüents. Algunes d'aquestes preguntes són: Pròxima data de cobrament, sou, detalls de la nòmina o rebuts.

Cal destacar que existeixen molts chatbots similars al de Manpower, els quals han estat creats manualment i dissenyats pel cas específic de l'empresa en qüestió.

2.1.2 LoyJoy

LoyJoy⁵ ofereix un servei que consisteix en modelar un BPMN en un Chatbot. És a dir, et permet, a partir d'un model BPMN, generar un assistent virtual. El producte de LoyJoy permet, de forma manual, generar el flux i les frases de resposta de l'assistent virtual.

Cal destacar que els assistents virtuals que es poden crear amb LoyJoy, tots són creats manualment i dissenyats pel cas específic de l'empresa en qüestió. Això suposa que tota empresa hagi de destinar uns recursos d'inversió inicial. A canvi de posteriorment, treure'n beneficis.

⁴ "Caso de éxito: Chatbot para Manpower - iNBest.cloud." 23 de maig. 2019, <https://www.inbest.cloud/comunidad/caso-de-exito-de-chatbot-para-manpower>. S'hi ha accedit el dia 15 de maig. 2020.

⁵ "Conversational BPMN - LoyJoy." <https://www.loyjoy.com/en/product/conversational-bpmn>. S'hi ha accedit el dia 15 de maig. 2020.

2.1.3 Treball del mateix grup

En el mateix grup (on s'està fent aquest projecte), existeix un treball previ⁶ on es va intentar fer el mateix que en aquest projecte. És a dir, es va intentar generar, automàticament, un assistent virtual a partir d'un model BPMN. L'anterior projecte, era capaç de, automàticament, generar un assistent virtual a partir d'un BPMN senzill. Aquest assistent, podia descriure el procés de forma seqüencial (pas a pas), a més, era capaç de respondre a una única pregunta genèrica (ex: "Qui envia la tasca X").

El chatbot però, tenia un gran problema i era l'enteniment de l'usuari. El chatbot només era capaç d'entendre paraules fixes que coincidissin completament amb les paraules de l'usuari. Això feia que el chatbot moltes vegades no entenia a l'usuari. El que feia impossible una conversa fluida. Aquesta limitació és deguda a l'ús de l'AIML⁷, el qual és un llenguatge de marcat d'intel·ligència artificial. Aquest és un llenguatge relativament antic, el qual té moltes limitacions.

El projecte actual té l'objectiu de solucionar el problema de l'enteniment fent ús de les tecnologies més punteres actuals (com ara DialogFlow⁸) pel que fa a l'enteniment del llenguatge natural.

2.2 Conclusions

Com a conclusió podem determinar que no hi ha gaires productes relacionats. Únicament el treball del mateix grup s'hi assembla. Tot i això, aquest projecte és centra en determinar si és possible un diàleg fluid, amb les tecnologies més punteres, que precisament era en el que mancava l'anterior.

Tot i que el treball anterior està relacionat amb aquest, i està fet pel mateix grup, no és convenient reutilitzar-lo perquè la implementació anterior està influenciada per l'ús de l'AIML. Tot i això, és possible que hi hagi algunes idees abstractes (ex: com enfocar les respostes del chatbot) que es puguin reutilitzar.

Dins del mateix anàlisi de mercat s'ha observat que els productes relacionats no generen automàticament els chatbot. Per tant, hi ha una oportunitat de mercat que la resta d'aplicacions encara no han aconseguit explotar.

En aquest projecte s'usa DialogFlow el qual permet processar llenguatge natural. Es tracta de la millor eina gratuïta que dona suport a la generació de Chatbots. Per tant, és la millor opció gratis disponible actualment tal com s'explica en l'apartat [Generació del Chatbot](#).

⁶ "From Process Models to Chatbots - ResearchGate."

https://www.researchgate.net/publication/333442636_From_Process_Models_to_Chatbots. S'hi ha accedit el dia 15 de maig. 2020.

⁷ "AIML Foundation." <http://www.aiml.foundation/>. S'hi ha accedit el dia 15 de maig. 2020.

⁸ "Dialogflow." <https://dialogflow.com/>. S'hi ha accedit el dia 22 de juny. 2020.

En la Taula 1 es mostra una comparativa on es resumeixen les diferències entre les diferents opcions de l'apartat anterior i el TFG.

	Generació automàtica del Chatbot	Consultes sobre el procés	Gratis	Centrat en els BPMN	Diàleg fluid	Edició Posterior
ManPower	NO	SÍ	Autònom	NO	SÍ	SÍ, però Difícil
LoyJoy	NO	SÍ	NO	SÍ	SÍ	SI
Mateix Grup	SÍ	SÍ	SÍ	SÍ	NO	NO
TFG	SÍ	SÍ	SÍ	SÍ	SI	SÍ, però Difícil

Taula 1: Taula comparativa entre les diferents opcions. Elaboració pròpia

3. Abast i obstacles del projecte

En aquest apartat s'exposa l'abast del projecte, els obstacles i els riscos que poden sorgir.

3.1 Abast

En aquest apartat s'exposa els objectius, subobjectius i requisits del projecte.

3.1.1 Objectius generals

El projecte té dos objectius principals, un consisteix en desenvolupar un algorisme de transformació d'un procés formal de negoci (model BPMN) en un agent de conversa, fent ús de les tecnologies més punteres.

L'altre objectiu està relacionat amb el projecte anterior del mateix grup. Aquest consisteix en valorar l'aportació de les noves tecnologies en vers les que es van usar en el seu moment. Principalment, s'avaluara dialogflow com a tecnologia de modelatge d'assistents virtuals.

En Fig. 1 es pot veure l'arquitectura general de l'aplicació. En l'apartat Arquitectura general del sistema, s'explica més a fons.

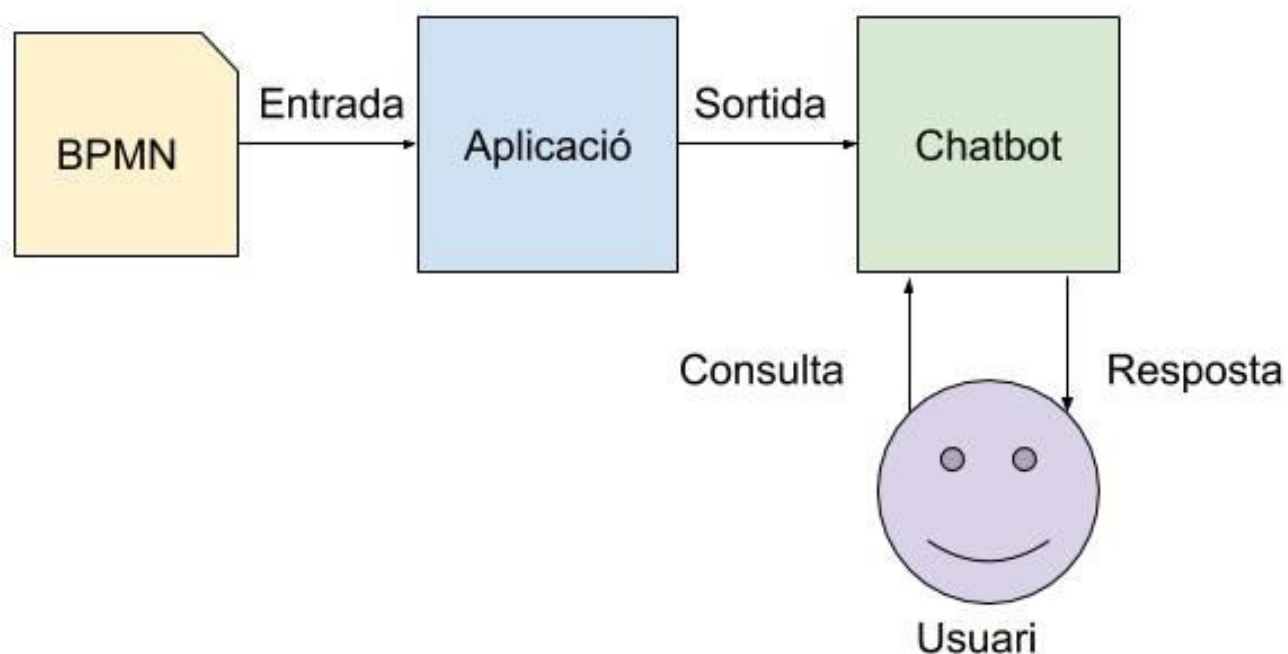


Figura 1: Arquitectura general de l'aplicació

3.1.2 Sub-objectius

Cada subobjectiu té una descripció i els seus criteris d'acceptació.

- Crea un aplicatiu senzill

- Existeix una Interfície gràfica que permet seleccionar un model bpmn tal que es genera un assistent virtual automàticament.
- Selecció de procés
 - El chatbot permet seleccionar sobre quin rol o procés vol fer la consulta.
- Explicació d'un procés
 - L'assistent virtual és capaç d'explicar pas a pas un procés.
- Consultes
 - L'assistent virtual accepta preguntes, genèriques, sobre un procés
 - L'assistent virtual respon correctament les preguntes generals.
- Diàleg Flexible
 - L'assistent virtual és capaç d'entendre la intenció de l'usuari a partir del significat de les paraules.
- Millora la flexibilitat de diàleg respecte el projecte anterior.
 - L'assistent virtual té millor capacitat d'enteniment que l'anterior projecte.
- Robustesa de DialogFlow
 - Entrenament de Dialogflow bo mitjançant frases d'entrenament

3.1.3 Requisits no funcionals

Mantenibilitat

Descripció: Mesura la facilitat amb la qual es pot donar manteniment al sistema i realitzar canvis.

Justificació: Per tal de desenvolupar nous requeriments, aïllar els defectes i les seves causes, corregir aquests defectes i atendre els canvis de requisits. A més, aquest projecte serà usat pel grup, per tant ha de ser mantenible.

Condicció de satisfacció: El desenvolupador pensa que és fàcil de mantenir.

Eficiència

Descripció: El sistema ha de respondre en un temps raonable

Justificació: Que l'usuari deixi la conversa perquè el temps de resposta és elevat.

Condicció de satisfacció: En mitjana el temps de resposta no supera els dos segons.

3.2 Obstacles

Aquest projecte té els següents obstacles:

- **Temps limitat:** La durada del projecte és d'un quadrimestre.
- **Tecnologies recents:** DialogFlow és una eina molt recent sobre la qual costarà aprendre com funciona.
- **Tecnologies desconegudes:** No he treballat mai en aquest camp, ni he usat cap de les eines (ex: DialogFlow), per tant requeria un temps extra d'aprenentatge.
- **Sense Pressupost:** Aquest projecte no té pressupost. Per tant, no es pot fer ús de la versió gratuïta de les eines.

- **Camp poc explorat:** El tema del TFG pertany a una temàtica poc estudiada.

3.3 Riscos

- **Eines no satisfactòries:** Les eines poden donar un rendiment inferior a l'esperat o no permetre implementar certes funcionalitats. El que podria implicar haver de descartar part de la feina feta i refer-la.
- **Integració de l'aplicatiu:** Fer l'aplicatiu pot complicar-se a l'hora d'integrar-lo amb DialogFlow.
- **Coronavirus:** El virus pot causar problemes, els quals poden augmentar la durada del projecte.

4. Planificació temporal inicial

4.1 Introducció

El projecte es realitza en el quadrimestre de primavera del curs 2019-2020. Oficialment comença el dia de la matrícula (3 de febrer) però el projecte es va començar a treballar l'1 de gener del 2020. S'espera tancar el projecte en el mateix quadrimestre en què s'inicia i el torn de lectura probablement serà al juny. Es dedicaran entre 20 i 30 hores setmanals (3 i 4 diàries) al projecte.

4.2 Descripció de les tasques

Per dur a terme aquest projecte s'ha determinat quines tasques s'han de fer, aquestes es mostren en la seva seqüència lògica. En general, com més gran és el codi, més tard es fa.

4.2.1 - Gestió del projecte

- [T1.1] - [Estudi d'eines de suport al projecte](#): Estudi del conjunt d'eines de tecnologies de la informació i comunicacions disponibles per realitzar el treball.
- [T1.2] - [Decidir l'abast](#): Es decideix l'abast del projecte.
- [T1.3] - [Planificació temporal](#): Es defineix la planificació temporal del projecte.
- [T1.4] - [Pressupost i sostenibilitat](#): Es calcula el cost, el pressupost i els recursos del projecte i s'elabora l'informe de sostenibilitat corresponent.
- [T1.5] - [Integració document final](#): Revisió i unificació de tots els documents

4.2.2 - Anàlisis

- [T2.1] - [Estudi d'eines de processament de llenguatge natural](#): Estudia les diferents eines que existeixen i determina la millor de totes.
- [T2.2] - [Definició de les funcionalitats](#): Definir les funcionalitats que tindrà el treball.
- [T2.3] - [Estudis de treballs similars existents](#): S'estudia els treballs existents per veure el que ofereixen.
- [T2.4] - [Definició de les tecnologies utilitzades](#): És seleccionen les tecnologies que s'utilitzarà segons les seves característiques.
- [T2.5] - [Estudi dels Chatbots](#): Estudia com funcionen els chatbots.
- [T2.6] - [Aprenentatge de Xatkit](#): S'estudia una eina de modelatge de Chatbots.
- [T2.7] - [Aprenentatge de DialogFlow](#): Estudia l'API de DialogFlow per veure les seves prestacions.
- [T2.8] - [Aprenentatge dels BPMN](#): S'estudia com funcionen i que expressen els BPMN (Business Process Modeling and Notation).
- [T2.9] - [Aprenentatge de Camunda](#): Aprenentatge de l'eina de processament dels BPMN.
- [T2.10] - [Avaluar millora de la flexibilitat](#): Avaluar l'impacte de les noves tecnologies a la flexibilitat del diàleg.

4.2.3 - Disseny, implementació i prova de les funcionalitats

- [T3.0] - **Xatkit**: Tota la feina que s'ha fet amb Xatkit i s'ha acabat descartant
- [T3.1] - **Processament del BPMN amb Camunda**: Lectura i enteniment dels diferents components d'un BPMN utilitzant Camunda.
- [T3.2] - **Selecció de procés**: El Chatbot permet seleccionar sobre quin rol o procés es vol fer la consulta.
- [T3.3] - **Explicació d'un procés**: El Chatbot és capaç d'explicar pas a pas un procés.
- [T3.4] - **Consultes**: El Chatbot accepta preguntes genèriques.
- [T3.5] - **Transformació a diàleg interpretable per DialogFlow**: Traduir tots els components del model BPMN en un diàleg interpretable per Dialog Flow.
- [T3.6] - **Robustesa de DialogFlow**: Dissenyar e implementar l'entrenament de Dialog Flow per aconseguir un diàleg fluid.
- [T3.7] - **Llenguatge natural**: Fer que el Chatbot utilitzi un llenguatge natural i flexible.
- [T3.8] - **Generació del Chatbot**: Genera el Chatbot
- [T3.9] - **Crea aplicatiu**: Es dissenya, implementa una aplicació que permet seleccionar un model bpmn tal que es genera un assistent virtual automàticament.

4.2.4 - Documentació

- [T4.1] - **Redacció de la memòria**: Redacció de la memòria del projecte.
- [T4.2] - **Preparació de la presentació oral**: Prepara la defensa del projecte.

4.2.5 - Seguiment

- [T5.1] - **Correus electrònics de seguiment**: Enviament de correus electrònics per tal de preguntar per dubtes i fixar reunions de seguiment.
- [T5.2] - **Reunió de seguiment**: Reunió amb els directors per discutir sobre el progrés i fer el seguiment del projecte.
- [T5.3] - **Reunió d'Esprint**: Reunions individuals per fer el seguiment i planifica els esprints.

En la Taula 2 es pot veure un resum de les tasques i la seva informació. Cada macro tasca conte els recursos que son necessaris per a totes les subtasques.

Codi	Nom descriptiu	Hores	Dependencies
T1	Gestió del projecte	75	
T1.1	Estudi d'eines de suport al projecte	10	
T1.2	Decidir l'abast	25	
T1.3	Planificació temporal	9	T1.2
T1.4	Pressupost i sostenibilitat	12	T1.3
T1.5	Integració document final	19	T1.4

T2	Analisis	137	
T2.1	Estudi d'eines de processat de llenguatge natural	4	
T2.2	Definició de les funcionalitats	10	T2.1
T2.3	Estudis de treballs similars existents	12	T2.2
T2.4	Definició de les tecnologies utilitzades	6	T2.3
T2.5	Estudi dels Chatbots	10	T2.4
T2.6	Aprenentatge de Xatkit	20	T2.4
T2.7	Aprenentatge de DialogFlow	30	T2.4
T2.8	Aprenentatge dels BPMN	20	T2.4
T2.9	Aprenentatge de Camunda	15	T2.8
T2.10	Evaluar millora de la flexibilitat	10	T3.9
T3	Disseny, implementació i prova de funcionalitats	172	
T3.0	Xatkit	17	T2.6
T3.1	Processat del BPMN amb Camunda	15	T2.9
T3.2	Selecció de procés	10	T3.1
T3.3	Explicació d'un procés	10	T3.2
T3.4	Consultes	10	T3.3
T3.5	Transforma a diàleg interpretable per DialogFlow	40	T3.1
T3.6	Robustesa de DialogFlow	30	T3.5
T3.7	Llenguatge natural	20	T3.5
T3.8	Generació del Chatbot	15	T3.5
T3.9	Crea aplicatiu	15	T3.8
T4	Documentació	70	
T4.1	Redacció de la memòria	40	T1.5
T4.2	Preparació de la presentació oral	30	T4.1
T5	Seguiment	12	

T5.1	Correus electrònics de seguiment	2	
T5.2	Reunió de seguiment	10	
T5.3	Reunió de Sprint	8	
Total		484	

Taula 2: Taula informativa sobre les tasques i la seva informació. Elaboració pròpia.

4.3 Recursos

En el projecte hi ha dos tipus de recursos, els humans i els materials.

4.3.1 Recursos humans

Una persona (l'alumne) s'encarregarà de desenvolupar tot el projecte. Durant el transcurs del projecte, l'alumne, tindrà els següents rols: Programador, analista, dissenyador i cap de projecte. En la Taula 3 es pot veure les tasques que realitza cada rol.

Rol	Tasques que realitza
Programador	Totes les del T3
Analista	Totes les del T2 excepte T2.2 i T2.4
Dissenyador	T2.2 i T2.4
Cap de projecte	Totes les del T1, T4 i T5

Taula 3: Taula informativa sobre els recursos humans i les tasques que realitzen

4.3.2 Recursos materials

En la Taula 4 es pot veure els recursos del projecte i en quines tasques s'utilitzen.

Codi	Nom	Utilitat	Tasques
R1	Portàtil	Hardware per realitzar el projecte	Totes
R2	Taiga ⁹	Software pel seguiment del progrés	T1.*
R3	Google Drive	Eines software per documentar, i redactar la memòria.	T1.1, T4.*
R4	PyCharm ¹⁰	Entorn de desenvolupament integrat per desenvolupar l'aplicació.	T3.0
R5	IntelliJ	Entorn de desenvolupament integrat per desenvolupar	T3.*

⁹ "Taiga.io." <https://taiga.io/>. S'hi ha accedit el dia 26 de juny. 2020.

¹⁰ "PyCharm: the Python IDE for Professional" <https://www.jetbrains.com/pycharm/>. S'hi ha accedit el dia 26 de juny. 2020.

		l'aplicació.	
R6	Servidor	Guardar l'aplicació	T3.9
R7	Correu de la FIB	Correu per comunicar-se amb els directors.	T5.1
R8	GitHub ¹¹	Repositori remot per emmagatzemar el codi.	T3.*
R9	Internet	Cerca d'informació i realitzar la documentació	T2.*
R10	draw.io ¹²	Eina software per realitzar diagrames	T4.1
R11	bpmn.io ¹³	Eina software per realitzar diagrames BPMN	T3.1, T4.1
R12	DialogFlow	Eina Software per programar un Chatbot	T3.5, T3.6
R13	Ganttproject ¹⁴	Eina software per elaborar el Gantt	T1.3, T4.1

Taula 4: Taula informativa sobre els recursos i les seves dependències. Elaboració pròpia

4.3 Gestió del risc

En la Taula 5 s'esmenten els obstacles identificats en el primer lliurament junt amb la probabilitat en què poden aparèixer, l'impacte que tenen en el projecte i l'estratègia per a resoldre'ls.

Risc	Probabilitat	Impacte	Estratègia de resolució
Quedarse sense temps	Mitja	Alt	Mentre es desenvolupa el projecte es controla si les fites dels Esprints es compleixen o no. En el cas que no es pugui replanifica la feina, llavors s'haurà de reduir l'abast
Tecnologies desconegudes requereixen més dedicació	Alta	Mig	S'ha sobreestimat la dedicació a aprendre com funcionen les tecnologies.
Eines no satisfactòries	Alta	Alt	Busca una altra eina que faci el mateix. En cas que no existeixi cap eina, l'alumne desenvolupa una eina amb les prestacions necessàries.
Coronavirus	Alta	Alt	Difícil de resoldre. Treballar desde casa. Les reunions via email i videoconferència.

¹¹ "GitHub." <https://github.com/>. S'hi ha accedit el dia 26 de juny. 2020.

¹² "Draw.io - Diagrams.net." <https://www.draw.io/>. S'hi ha accedit el dia 26 de juny. 2020.

¹³ "bpmn.io." <https://bpmn.io/>. S'hi ha accedit el dia 26 de juny. 2020.

¹⁴ "GanttProject." 22 d'abr.. 2020, <https://www.ganttproject.biz/>. S'hi ha accedit el dia 26 de juny. 2020.

Taula 5: Taula informativa sobre els riscos, el seu impacte i la seva resolució. Elaboració pròpia

Cal destacar el risc **“Eines no satisfactòries”**, el qual ja ha succeït. En el projecte es va utilitzar Xatkit, una eina de modelatge de Chatbots, però ha resultat ser una eina no satisfactòria. L'estratègia per resoldre aquest problema ha estat implementar jo mateix les funcionalitats.

En el cas que algun dels altres riscos succeeixi, l'estratègia de resolució hauria de ser prou bona per a fer-ne front i poder acabar el projecte a temps. Pel que fa als recursos, no suposarien cap problema.

4.4 Diagrama de Gantt

En aquest apartat es mostra el diagrama de Gantt associat a les tasques definides en apartats anteriors.

S'ha de tenir en compte que tot i que el projecte va començar l'1 de gener, hi havia altres assignatures i per tant es va dedicar setmanalment menys hores.

Pel que fa a la distribució de tasques, aquestes estan distribuïdes en fites. Aquestes fites són: **Inici de GEP** (17 de febrer), **Finalització de GEP** (16 de març), **robustesa de DialogFlow** (20 d'abril), **crea aplicatiu** (11 de maig) i **la defensa del TFG** (juny). Donada la naturalesa d'algunes tasques, aquestes estaran en més d'una iteració (memòria, reunions, seguiment, etc.).

En la Fig. 2 es pot veure la taula de tasques del Gantt.

Fecha de inicio	Fecha de fin	Nombre	CODI	Duración	Riesgo	Rol
21/12/19	21/12/19	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
17/02/20	16/03/20	▣ • Gestio de Projectes	T1	29	Bajo	Cap de Projecte
17/02/20	17/02/20	• Estudi d'eines de suport al projecte	T1.1	1	Bajo	Cap de Projecte
18/02/20	25/02/20	• Decidir l'abast	T1.2	8	Bajo	Cap de Projecte
26/02/20	2/03/20	• Planificació temporal	T1.3	6	Bajo	Cap de Projecte
3/03/20	9/03/20	• Pressupost i sostenibilitat	T1.4	7	Bajo	Cap de Projecte
10/03/20	16/03/20	• Integració document final	T1.5	7	Bajo	Cap de Projecte
9/01/20	9/01/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
9/01/20	9/01/20	• Reunió de Sprint	T5.3	1	Bajo	Cap de Projecte
1/01/20	18/05/20	▣ • Anàlisis	T2	139	Bajo	Analista
1/01/20	4/01/20	• Estudi d'eines de processat de lleng...	T2.1	4	Bajo	Analista
8/01/20	11/01/20	• Definició de les funcionalitats	T2.2	4	Bajo	Dissenyador
15/01/20	18/01/20	• Estudis de treballs similars existents	T2.3	4	Bajo	Analista
22/01/20	23/01/20	• Definició de les tecnologies utilitza...	T2.4	2	Bajo	Dissenyador
27/01/20	30/01/20	• Estudi dels Chatbots	T2.5	4	Bajo	Analista
30/01/20	5/02/20	• Aprenentatge de Xatkit	T2.6	7	Alto	Analista
18/02/20	18/02/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
18/02/20	18/02/20	• Reunió de Sprint	T5.3	1	Bajo	Cap de Projecte
17/02/20	1/03/20	• Aprenentatge dels BPMN	T2.8	14	Bajo	Analista
27/02/20	7/03/20	• Aprenentatge de Camunda	T2.9	10	Bajo	Analista
10/02/20	12/04/20	• Aprenentatge de DialogFlow	T2.7	63	Medio	Analista
15/05/20	18/05/20	• Evaluar millora de la flexibilitat	T2.10	4	Medio	Analista
30/01/20	14/05/20	▣ • Disseny, implementació i prova de les f...	T3	106	Medio	Programador
30/01/20	9/02/20	• Xatkit	T3.0	11	Alto	Programador
9/03/20	15/03/20	• Processat del BPMN amb Camunda	T3.1	7	Bajo	Programador
17/03/20	17/03/20	• Reunió de Sprint	T5.3	1	Bajo	Cap de Projecte
17/03/20	17/03/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
17/03/20	22/03/20	• Selecció de procés	T3.2	6	Medio	Programador
24/03/20	29/03/20	• Explicació d'un procés	T3.3	6	Medio	Programador
30/03/20	4/04/20	• Consultes	T3.4	6	Medio	Programador
2/04/20	2/04/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
9/03/20	17/04/20	• Transforma a diàleg interpretable p...	T3.5	40	Medio	Programador
16/04/20	16/04/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
29/04/20	29/04/20	• Reunió de Sprint	T5.3	1	Bajo	Cap de Projecte
19/04/20	30/04/20	• Robustesa de DialogFlow	T3.6	12	Medio	Programador
25/04/20	4/05/20	• Llenguatge natural	T3.7	10	Medio	Programador
30/04/20	6/05/20	• Generació del Chatbot	T3.8	7	Medio	Programador
7/05/20	7/05/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
8/05/20	14/05/20	• Crea aplicatiu	T3.9	7	Medio	Programador
20/05/20	20/05/20	• Reunió de Seguiment	T5.2	1	Bajo	Cap de Projecte
17/03/20	4/06/20	▣ • Documentació	T4	80	Bajo	Cap de Projecte
17/03/20	25/05/20	• Redacció de la memòria	T4.1	70	Bajo	Cap de Projecte
26/05/20	4/06/20	• Presentació oral	T4.2	10	Bajo	Cap de Projecte
1/01/20	29/05/20	▣ • Seguiment	T5	150	Bajo	Cap de Projecte
1/01/20	29/05/20	• Correus electrònics de seguiment	T5.1	150	Bajo	Cap de Projecte

Figura 2: Taula de les tasques del Gantt del projecte.

En la Fig. 3 es pot veure el Gantt. A l'esquerra està el codi de cada tasca. Amb colors estan diferenciades les diferents tasques segons la seva agrupació (T1, T2, T3, T4, T5).

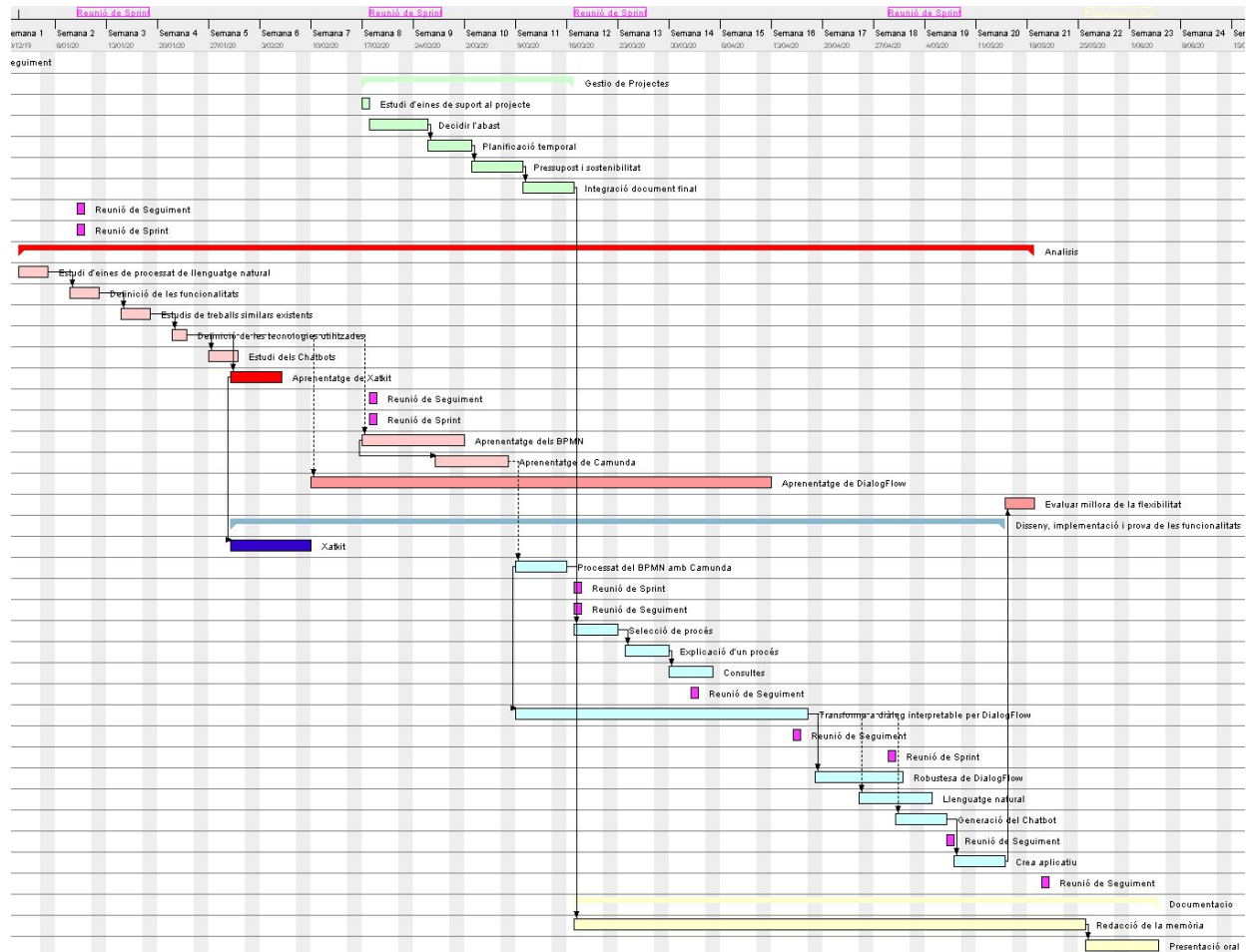


Figura 3: Diagrama de Gantt del projecte.

5. Gestió econòmica

En aquest apartat es calcula el pressupost del projecte tenint en compte costos humans, materials, contingències i imprevistos.

5.1 Costos de Personal per Activitat (CPA)

Aquest projecte és realitzat per una persona, estudiant de la FIB, que farà dels quatre rols que té aquest projecte. El salari brut per rol¹⁵, serà el que es veu en la Taula 6.

Rol	Cap de projecte	Analista	Dissenyador	Programador
Cost per hora (€)	9	9	9	9

Taula 6: Taula sobre la remuneració per hora de cada rol

En la Taula 7 es pot observar el cost estimat per tasca del projecte i les tasques que tenen cada rol.

Codi	Tasca	Hores	Rol	Cost(€)
T1	Gestió del projecte	75	Cap de projecte	675
T1.1	Estudi d'eines de suport al projecte	10	Cap de projecte	90
T1.2	Decidir l'abast	25	Cap de projecte	225
T1.3	Planificació temporal	9	Cap de projecte	81
T1.4	Pressupost i sostenibilitat	12	Cap de projecte	108
T1.5	Integració document final	19	Cap de projecte	171
T2	Anàlisi	137	Analista	1233
T2.1	Estudi d'eines de processat de llenguatge natural	4	Analista	36
T2.2	Definició de les funcionalitats	10	Dissenyador	90
T2.3	Estudis de treballs similars existents	12	Analista	108
T2.4	Definició de les tecnologies utilitzades	6	Dissenyador	54
T2.5	Estudi dels Chatbots	10	Analista	90
T2.6	Aprenentatge de Xatkit	20	Analista	180
T2.7	Aprenentatge de DialogFlow	30	Analista	270
T2.8	Aprenentatge dels BPMN	20	Analista	180
T2.9	Aprenentatge de Camunda	15	Analista	135

¹⁵ "Pràctiques en empresa | Facultat d'Informàtica de ... - UPC-FIB."

<https://www.fib.upc.edu/ca/empresa/practiques-en-empresa>. S'hi ha accedit el dia 19 de juny. 2020.

T2.10	Evaluar millora de la flexibilitat	10	Analista	90
T3	Disseny, implementació i prova de funcionalitats	172	Programador	1548
T3.0	Xatkit	17	Programador	153
T3.1	Processat del BPMN amb Camunda	15	Programador	135
T3.2	Selecció de procés	10	Programador	90
T3.3	Explicació d'un procés	10	Programador	90
T3.4	Consultes	10	Programador	90
T3.5	Transforma a diàleg interpretable per DialogFlow	40	Programador	360
T3.6	Robustesa de DialogFlow	30	Programador	270
T3.7	Llenguatge natural	20	Programador	180
T3.8	Generació del Chatbot	15	Programador	135
T3.9	Crea aplicatiu	15	Programador	135
T4	Documentació	70	Cap de projecte	630
T4.1	Redacció de la memòria	40	Cap de projecte	360
T4.2	Preparació de la presentació oral	30	Cap de projecte	270
T5	Seguiment	12	Cap de projecte	108
T5.1	Correus electrònics de seguiment	2	Cap de projecte	18
T5.2	Reunió de seguiment	10	Cap de projecte	90
T5.3	Reunió de Sprint	8	Cap de projecte	72
Total		484		4194

Taula 7: Cost de cada tasca segons qui les fa i quantes hores duren

Al cost total de la [Taula 7](#) se li ha d'aplicar la taxa de la seguretat social, que representa un 30% del salari brut, per tant el cost serà de $4194\text{€} * 1.3 = 5452,2\text{€}$.

5.2 Costos Genèrics (CG)

Els costos genèrics poden diferenciar-se en tres tipus, els costos de hardware, els costos de software, i altres costos com la llum, el gas, l'aigua, etc.

Pel càlcul dels costos s'ha tingut en compte que en 2020 hi ha 253 dies hàbils¹⁶, amb 8 hores laborables al dia. El cost per hora es calcula dividint el preu total pel nombre total d'hores de vida útil (4 anys * 253 dies/any * 8 hores/dia).

¹⁶ "Días laborables en año 2020 en España."

https://www.dias-laborables.es/dias_laborables_feriados_2020.htm. S'hi ha accedit el dia 19 de juny. 2020.

5.2.1 Hardware

En la Taula 8 es pot veure els costos del hardware amb una amortització en 4 anys. El portàtil Predator Helios 300 es fa servir durant tot el projecte, per tant són 484 hores.

Producte	Preu	Unitat	Vida útil (anys)	Cost per hora (€/h)	Hores	Cost
Predator Helios 300	825,00	1	4	0,10	484	49,32

Taula 8: Costos del Hardware

5.2.2 Software

Tot el software que es fa servir en el projecte és gratuït: GitHub, Google Drive, correu de la FIB, IntelliJ, DialogFlow, Taiga, draw.io, bpmn.io.

5.2.3 Altres costos

En la Taula 9 es pot observar els costos indirectes com el lloguer del local, el llum, el gas, l'aigua i la internet. No incloc el cost del trajecte (per les reunions), ja que les reunions es realitzen en dies on l'alumne té classe, i per tant, no suposen un cost afegit.

Producte	Preu (any)	Cost per hora	Hores	Cost
Establiment	4.000,00	1,98	484,00	956,52
Internet	375,00	0,19	484,00	89,67
Llum	320,00	0,16	484,00	76,52
Gas	250,00	0,12	484,00	59,78
Aigua	200,00	0,10	484,00	47,83
Total				1.230,33

Taula 9: Costos indirectes

5.3 Contingències

En el TFG es reserva un marge de seguretat del 10%. En la Taula 10, es pot observar els càlculs segons cada categoria.

Producte	Cost	Percentatge	Cost reservat
Costos de Personal per Activitat	5.452,20€	10%	545,22€
Hardware	49,32	10%	4,93€
Software	0	10%	0,00€
Altres costos	1.230,33	10%	123,03€
Total			673,18€

Taula 10: Contingències del projecte

5.4 Imprevistos

En la Taula 11 es pot veure cada imprevist, amb el nombre d'hores que suposaria, el cost monetari que suposaria, la probabilitat d'aparició i el cost ponderat. Per calcular el cost monetari per hora del projecte, s'ha dividit el cost monetari total entre el nombre d'hores que s'ha estimat que durarà el projecte: $5452,2 \text{ €} / 484 \text{ hores} = 11,26 \text{ €/h}$.

Notar “**Eines no satisfactòries**” amb el 100% de probabilitat, ja que aquest ja ha succeït.

Risc	Hores	Cost(€)	Probabilitat	Cost ponderat(€)
Retard(1 setmana de retard)	20,00	225,30	50,00%	112,65
Retard(2 setmanes de retard)	40,00	450,60	25,00%	112,65
Tecnologies desconegudes requereixen més dedicació	35,00	394,27	80,00%	315,42
Eines no satisfactòries	50,00	563,24	100,00%	563,24
Total				1.103,96

Taula 11: Imprevistos del projecte i el seu cost ponderat

5.5 Costos totals

El projecte tindrà un cost total de 8508,99 euros, tal com es pot veure en la Taula 12.

CPA	Hardware	Software	Altres costos	Contingències	Imprevistos	Total
5452,2	49,32	0,00	1.230,33	673,18	1.103,96	8.508,99

Taula 12: Cost total del projecte

5.6 Control de gestió

Per tal de validar que la planificació es va complint, i en cas de no es compleixi poder detectar on i perquè, es defineixen indicadors de control els quals ajudaran a supervisar els costos durant l'execució del projecte.

En el projecte es realitzarà un control al final de cada iteració (fita), per tal de validar el que s'ha gastat amb el que es va planificar. En aquest control compararà el cost real en vers el cost que es va planificar. Concretament, és calcularan i compararan, per a cada tasca, els següents indicadors:

- **Cost estimat:** Cost estimat sense contar contingències ni imprevistos.
- **Cost real:** Cost real de la tasca.
- **Desviació del cost:** Cost estimat - Cost real.

En el cas que existeixi una desviació important del cost, sigui positiva o negativa, s'intentarà descobrir el perquè. Per fer-ho, es calculara uns indicadors extres per descobrir l'origen de la desviació. Aquests indicadors són:

- **CE RRHH:** Cost estimat dels recursos humans
- **CR RRHH:** Cost real dels recursos humans
- **Desviació en el cost dels recursos humans:** $CE\ RRHH - CR\ RRHH$

6. Sostenibilitat i compromís social

6.1 Autoavaluació

La conclusió que he tret després de realitzar l'enquesta és que tinc un nivell mitjà-alt pel que fa al coneixement de la sostenibilitat. Crec que sóc capaç d'avaluar decentment els aspectes socials i ambientals de la sostenibilitat. L'aspecte econòmic crec que és el que menys entenc.

Crec que l'aspecte social i l'aspecte ambiental els sé avaluar molt bé. Sé veure els impactes, positius i negatius, que pot tenir una tecnologia en la societat i en el medi ambient. També crec que conec aquells indicadors importants a l'hora de valorar la sostenibilitat. Tant en l'àmbit social com en l'àmbit ambiental.

L'aspecte econòmic de la sostenibilitat és el que menys comprenc. No conec quins indicadors són necessaris per a avaluar correctament l'impacte econòmic d'una tecnologia o un servei TIC. Suposo que indicadors com la viabilitat econòmica o els guanys econòmics poden ser interessant. Tot i això, crec que no conec en profunditat (no ser valorar prou bé) indicadors com la transversalitat econòmica (l'accés a les tecnologies de forma independent de la situació econòmica).

Personalment crec, que comprenc els aspectes socials i ambientals perquè des de petit n'he rebut informació. En canvi, crec que no hi ha prou informació que permeti avaluar correctament l'aspecte econòmic. No crec que sigui un aspecte deixat de banda, sinó que és dóna molta més importància als aspectes ambientals i socials. Tot i això, gràcies a internet, és possible informar-se d'aquests temes de forma autònoma.

6.2 Dimensió econòmica

Per tal de planificar el projecte s'ha estimat els costos de recursos humans, costos genèrics com el hardware, el software i altres costos indirectes com el lloguer, el desplaçament o l'electricitat. També s'ha inclòs una estimació dels costos que suposaran les contingències i els imprevistos.

El cost del producte és estàndard tenint en compte la seva durada. El projecte es podria realitzar en menys temps si més d'una persona hi treballés, però no es podria reduir els recursos ni el cost del projecte.

Donat que l'únic projecte prou similar que existeix és el treball realitzat en el mateix grup on es realitza aquest TFG, el cost és similar. És rellevant destacar que també va ser realitzat per un estudiant de la FIB. La solució que dóna aquest TFG millora, econòmicament, l'altre solució perquè amb un cost similar, es millora la qualitat i la fluïdesa del diàleg.

Un cop acabat aquest TFG, el projecte serà utilitzat per al grup on es desenvolupa.

6.3 Dimensió ambiental

L'impacte ambiental que tindrà aquest projecte és mínim perquè no requereix cap tecnologia ni cap procés que contami. L'únic aspecte a considerar ambientalment és el consum energètic (ús d'un portàtil, entorn de treball, llums, etc.). Tot i això, aquests recursos són molt lleugers, i de totes maneres s'haurien consumit encara que no és fes el projecte. De totes maneres, és farà ús d'entorns de desenvolupament, framework, llibreries i tecnologies existents per tal d'accelerar el desenvolupament del projecte, reduint així el consum energètic.

La solució que proposa aquest TFG no té cap impacte, positiu o negatiu, en el medi ambient. No millora ni empitjora respecte a productes similars.

6.4 Dimensió social

Aquest projecte em dóna l'oportunitat de treballar i gestionar un projecte real. També em permet aprendre tecnologies noves com Dialog Flow o processos de negoci. A més, em permet aprendre com funcionen els assistents virtuals.

Actualment existeixen assistents virtuals que permeten consultar informació o procediments. Aquests però, són assistents virtuals fets a mida i per tant, s'han de programar de forma personalitzada. Amb aquest projecte es desenvoluparà una aplicació que permetrà automatitzar la generació dels assistents virtuals. Això facilitarà la feina a les empreses i els treballadors es veuran beneficiats al poder parlar amb un assistent virtual.

El TFG també té una component de recerca, on es busca veure la viabilitat de generar assistents virtuals fiables de forma automàtica. Això ajudarà al grup en avançar la recerca d'aquest tema.

7. Planificació final i desviacions

La planificació final ha canviat molt respecte a la planificació inicial, això és degut al fet que es va planificar malament el projecte. La mala planificació inicial és deu a què al principi el projecte estava molt verd, i no estava clar fins on arribaria. A més, no es tenia experiència prèvia i per tant, no es van fer errors de càlcul. Per exemple, la planificació inicial es va fer en cascada¹⁷, és a dir, primer és feia l'anàlisi, i després la implementació.

Donada la poca exactitud de la planificació inicial i de cara a poder fer una comparativa, s'ha ajuntat l'anàlisi i la implementació. Per exemple, tant l'aprenentatge de dialogflow com la implementació utilitzant dialogflow, s'ajunten en una única tasca. A més, tal com s'explica en l'informe de seguiment, donada la mala planificació inicial va ser necessari concretar algunes tasques en subtasques. Per poder fer una comparativa, aquestes subtasques també s'inclouen en les tasques originals. Finalment, la utilització de l'eina [Xatkit](#) va resultar ser insatisfactòria, cosa que va comporta una pèrdua de temps.

Tot i la mala planificació inicial, la majoria de funcionalitats s'han pogut dur a terme. Això és degut al fet que inicialment es van deixar dues setmanes per contingències. Tot i això, hi ha algunes desviacions. Aquestes són:

1. La component [passarel·la paral·lela](#) s'ha simplificat. Això és degut al fet que la seva implementació va ser més complicada de l'esperat
2. No s'ha implementat un xat on l'usuari pugui interactuar amb el chatbot. Això és degut al fet que es pot integrar amb un xat existent. Per exemple, l'assistent de google.
3. S'ha simplificat els [missatges](#). Això és degut al fet que l'estàndard BPMN té moltes formes d'expressar que s'envia un missatge. En el projecte només s'implementa una forma.

Cal destacar que tot i la mala planificació temporal inicial, els [objectius](#) del projecte no han canviat.

Donat que el diagrama de Gantt de la planificació inicial no s'assembla en res a l'execució real. No es farà el Gantt perquè no té sentit comparar-los. En canvi, es farà una comparativa entre les hores planificades i les hores treballades.

En la Taula 13, es pot veure una comparativa resum entre hores treballades i hores planificades. Tal com s'ha dit anteriorment, s'ajunta l'anàlisi i la implementació. A més, GEP s'inclou en la memòria. Finalment, i tal i com s'ha dit abans, donat que la planificació inicial

¹⁷ "Waterfall model - Wikipedia." https://en.wikipedia.org/wiki/Waterfall_model. S'hi ha accedit el dia 26 de juny. 2020.

no és va fer correctament, aquesta comparativa només serveix per veure les diferències respecte a els grups de tasques.

Tasca	Hores reals	Hores planificades	Diferencia(h)	Cost real
Xatkit	37	37	0	416,80 €
Processat del BPMN amb Camunda	86,5	80	6,5	974,41 €
Transforma a diàleg interpretable per DialogFlow	72,7	100	-27,3	818,96 €
Llenguatge natural	82,35	62	20,35	927,66 €
Crea aplicatiu	10,2	30	-19,8	114,90 €
Redacció de la memòria	126,35	125	1,35	1.423,32 €
Preparació de la presentació oral	40	30	10	450,60 €
Seguiment	13,5	20	-6,5	152,08 €
Total	468,6	484	-15,4	5.278,72 €

Taula 13: Comparativa entre les hores dedicades i planificades. També és mostra el cost real.

Per tant i pel que respecte el cost real del projecte, l'únic aspecte que ha canviat és el cost humà per activitat. La resta de costos es mantenen iguals. Al haber dedicat menys hores de les planificades, el cost total disminueix. En la planificació inicial, el cost humà per activitat total era **5452,2€**. Per tant, el cost real del projecte és el nou cost d'activitats sumat a la resta de costos genèrics. El cost del projecte passa de **8508,99€** a és **8.335,51€**.

8. Models BPMN i Assistants virtuals

Abans d'explicar el funcionament del programa, és necessari entendre que són els models BPMN i quins components es cobriran. A més, és necessari entendre com són els assistents virtuals.

8.1 Models BPMN

BPMN (***Business Process Model and Notation***) és una notació gràfica estandarditzada que permet el modelatge de processos de negoci. Es tracta d'un estàndard que permet a una empresa representar els processos que realitza.

8.1.1 Cobertura del BPMN dintre del prototip

Donat l'abast del projecte i donada l'extensió de l'estàndard BPMN, el prototip no cobreix la totalitat de l'estàndard, sinó que es cobreix els components suficients per a valorar la flexibilitat del llenguatge. Els components estan explicats en els següents subapartats.

8.1.1.1 Participants

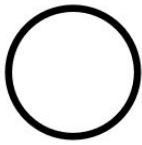
En la Fig. 4 es mostra la representació d'un participant. Un participant és un actor dintre d'un procés. Els participants poden ser persones o entitats. Per exemple, un empleat dintre d'una empresa. Cal destacar que els participants no tenen per què ser treballadors d'un negoci, sinó que també poden ser actors externs. Per exemple, un visitant en un Zoo és un actor implicat i per tant, també és un participant.



Figura 4: Component tipus participant

8.1.1.2 Esdeveniment inicial

En la Fig. 5 es mostra la representació d'un esdeveniment inicial. Un esdeveniment inicial és un esdeveniment, i per tant, indica un fet que succeeix. L'esdeveniment inicial indica l'inici d'un procés. Normalment tenen un text associat, el qual indica l'esdeveniment.



Start event

Figura 5: Component tipus event d'inici

8.1.1.3 Tasca

En la Fig. 6 es mostra la representació d'una tasca. Una tasca, a diferència d'un esdeveniment, descriu el que es fa. Sempre tenen un text associat el qual indica la tasca que es realitza. El text acostuma a començar amb un verb en infinitiu, el qual indica l'acció. Per exemple, "Drink water".

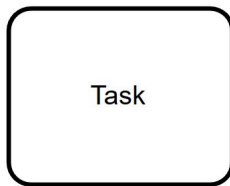


Figura 6: Component tipus tasca

8.1.1.4 Flux de seqüència

En la Fig. 7 es mostra la representació del flux de seqüència. Un flux de seqüència es representa amb la forma d'una fletxa. El component indica l'ordre en el qual es realitza el procés.



Figura 7: Component tipus flux de seqüència

En la Fig. 8 es pot veure que primer comença l'esdeveniment inicial "Start event", i després es realitza la tasca "Task".

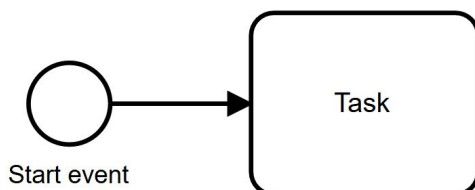


Figura 8: Exemple del flux de seqüència

8.1.1.5 Passarel·la exclusiva

Una passarel·la exclusiva controla el flux del procés segons una condició.

En la Fig. 9 es mostra la representació gràfica de la passarel·la exclusiva. La passarel·la té associada una pregunta. Segons la resposta, el procés de negoci continuarà per un costat o per un altre. El flux només pot continuar per un sol costat.

En el context del projecte, s'anomena passarel·la exclusiva d'obertura si aquesta bifurca el procés.

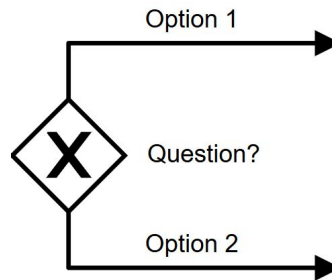


Figura 9: Component tipus pasarel·lela exclusiva

En la Fig. 10 es pot veure una passarel·la de tancament. Aquesta indica que els fluxos s'ajunten. És a dir, és una component que indica que no importa per quina branca es vingui, es continuarà per una sola.

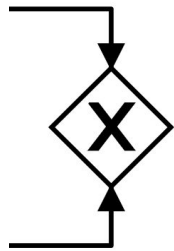


Figura 10: Representació de la la passarel·la de tancament. Ajunta el flux

En la Fig. 11 es pot veure un cas d'ús senzill. Primer de tot, hi ha la pregunta "Hungry?". Si la resposta és "Yes", es realitza la tasca "Eat food". Si la resposta és no, llavors no es realitza cap tasca. La passarel·la exclusiva del final indica que els fluxos s'ajunten. Per tant, no importa quina branca hagi escollit, a l'acabar, s'unifica el flux.

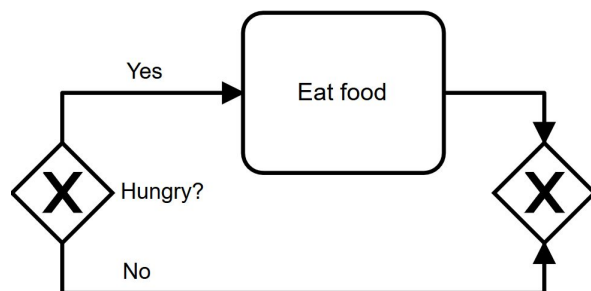


Figura 11: Exemple senzill d'ús d'una pasarel·la exclusiva

8.1.1.6 Passarel·la paral·lela

Una passarel·la paral·lela indica que totes les branques contingudes es poden executar en paral·lel. En la Fig. 12 es mostra la representació gràfica de la passarel·la paral·lela. En la Fig. 12 primer es pot realitzar la tasca “Eat” i després la tasca “Drink”. O es pot fer al revés, és a dir, primer la tasca “Drink” i després la tasca “Eat”. La passarel·la final (la de la dreta) indica que s’unifica el flux d’execució. És a dir, el participant no pot continuar si no ha acabat totes les branques contingudes. Per exemple, si el participant ha realitzat la tasca “Eat”, també ha de realitzar la tasca “Drink” si vol continuar.

En el context del projecte, s’anomena passarel·la paral·lela d’obertura si té aquesta inicia un context paral·lel. D’altra banda, s’anomena passarel·la paral·lela de clausura si unifica branques.

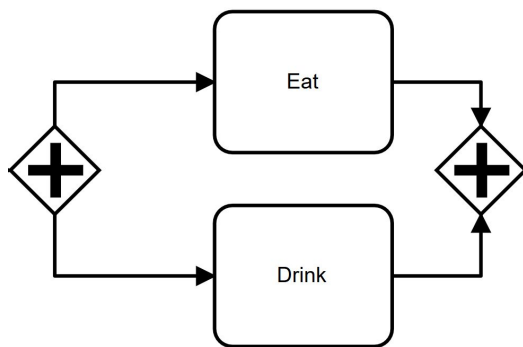


Figura 12: Component tipus passarel·la paral·lela

8.1.1.7 Esdeveniment intermedi de missatge

Un esdeveniment intermedi indica un fet que succeeix entre l'inici i el final del procés. En el cas de l'esdeveniment intermedi de missatge, també indica que és un missatge entre dos participants. Per tant, aquesta component és una forma d'indicar que dos participants es comuniquen durant el procés.

En la Fig. 13 es pot veure un exemple senzill on una persona envia un missatge a un banc. Cal destacar que la fletxa que s'utilitza és diferent de les de flux de seqüència. Aquesta component s'explica en l'apartat 7.1.1.8 Flux de missatge.

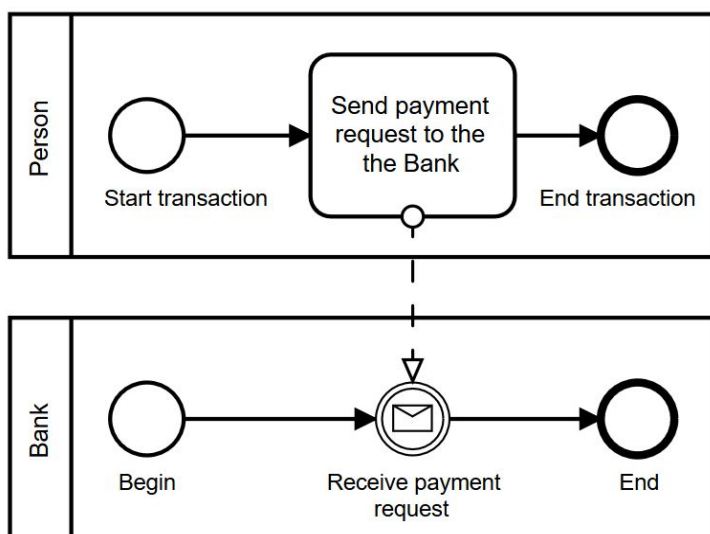


Figura 13: Esdeveniment intermig de missatge

8.1.1.8 Flux de missatge

En la Fig. 13 es mostra la representació del flux de missatge. Un flux de missatge es representa amb la forma d'una fletxa discontinua. El component indica que un participant envia un missatge a un altre.

8.1.1.9 Esdeveniment final

En la Fig. 14 és mostra la representació d'un esdeveniment final. Un esdeveniment final és un esdeveniment, i per tant, indica un fet que succeeix. L'esdeveniment final indica l'acabament d'un procés. En cas de tenir text associat, aquest dóna informació sobre com acaba.

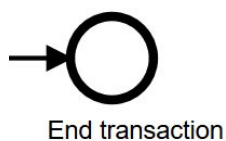


Figura 14: Component tipus esdeveniment final

8.1.2 Exemple

Per tal d'exemplificar les components descrites anteriorment, en aquest apartat s'explica un procés senzill.

En la Fig. 15 es pot veure un model BPMN senzill. Aquest inclou dos participants (empleat i departament), tasques, esdeveniments d'inici, intermedi i final, i dues passarel·les, una exclusiva i una altre paral·lela.

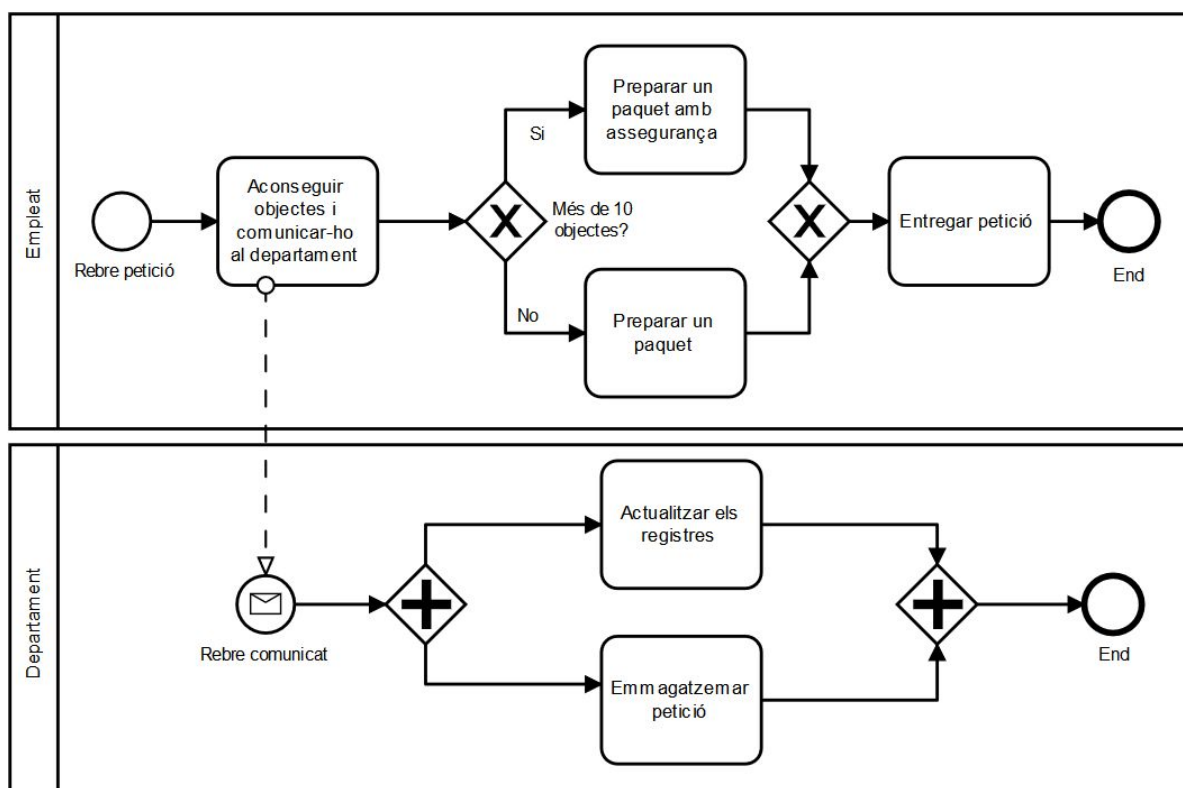


Figura 15: BPMN senzill que funciona com a exemple. Descriu el procés que realitza un empleat i el departament, quan és al rebre una petició

El procés comença quan un empleat rep una petició. A continuació, l'empleat aconsegueix uns objectes, i comunicar la petició al departament. Llavors l'empleat comprova si hi ha més de 10 objectes. En cas afirmatiu, prepara un paquet amb assegurança. En cas negatiu, només prepara el paquet. Sense importar en quin cas es trobi, entrega la petició i acaba. Pel que fa al departament, aquest comença quan un empleat comunica una petició. Llavors pot actualitzar els registres i emmagatzemar la petició o fer-ho al revés. Finalment, el departament acaba el procés.

8.2 Assistents virtuals

Els assistents virtuals han agafat molta força els darrers anys. Hi ha molts exemples d'assistents virtuals, com ara Siri¹⁸, Google Assistant¹⁹ o Alexa²⁰.

Avui en dia els assistents virtuals es caracteritzen per ser capaços de realitzar tasques generals com ara consultar el clima o reproduir la música que se li indica. Tot i això, la majoria d'assistents virtuals estan orientats a negoci. N'hi ha molts exemples: Assistents

¹⁸ "Siri - Apple." <https://www.apple.com/siri/>. S'hi ha accedit el dia 21 de juny. 2020.

¹⁹ "Google Assistant, your own personal Google." <https://assistant.google.com/>. S'hi ha accedit el dia 21 de juny. 2020.

²⁰ "Amazon Alexa - Apps on Google Play." <https://play.google.com/store/apps/details?id=com.amazon.dee.app&hl=en>. S'hi ha accedit el dia 21 de juny. 2020.

virtuals per demanar pizzes, per fer consultes amb una teleoperadora, per demanar detalls d'una transacció, per demanar l'estat del teu banc.

Els assistents virtuals enfocats a un negoci acostumen a ser molt guiats i senzills. L'objectiu dels assistents acostuma a ser contractar un servei. Per fer-ho, l'assistent guia a l'usuari donant-li a escollir entre diverses opcions. En la Fig. 16 es pot veure l'assistent virtual de KLM Royal Dutch Airlines²¹, el qual permet comprar un bitllet d'avió. L'assistent pregunta a l'usuari on anirà i des d'on sortirà. A partir d'aquesta informació, l'assistent realitza la compra del bitllet.

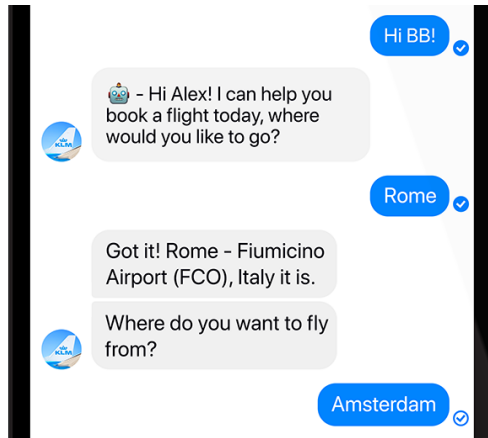


Figura 16: Assistent virtual de KLM Royal Dutch Airlines, el qual permet compra un bitllet d'un vol. Imatge extreta de <https://dialogflow.com/images/case-studies/klm/KLM-001.png>

En general la majoria d'assistents virtuals treballen sobre tres punts claus, el context, la intenció i la resposta.

- El context és la situació actual, és a dir, és la consideració del que s'ha dit anteriorment.
- La intenció és el concepte al qual es refereix l'usuari. La intenció es veu afectada segons el context.
- La resposta és la o les frases que dona l'assistent virtual com a resposta.

En general, un assistent virtual condiona la seva resposta segons el context i la intenció de l'usuari. És a dir, encara que un usuari digui la mateixa frase, en contextos diferents, això provocarà que el chatbot respongui de manera diferent. Per exemple, si un usuari pregunta: **"Teniu samarretes del Barça?"** i posteriorment diu **"Ensenyem fotos"**, l'assistent virtual ensenyarà fotos de la samarreta del Barça. En canvi, si l'usuari pregunta: **"Teniu samarretes del Reial Madrid?"** i posteriorment diu **"Ensenyem fotos"**, l'assistent virtual haurà d'ensenyar fotos del Reial Madrid. És a dir, tot i que l'usuari diu la mateixa frase **"Ensenyem fotos"**, a partir del context l'assistent virtual ha de saber que la intenció de l'usuari és diferent.

²¹ "KLM Royal Dutch Airlines - Dialogflow." <https://dialogflow.com/case-studies/klm>. S'hi ha accedit el dia 21 de juny. 2020.

9. Arquitectura general del sistema

L'arquitectura del sistema es divideix en dues fases, la fase de generació del chatbot i la fase d'execució del chatbot. A més, també és important considerar el diagrama de classes del sistema.

L'arquitectura del sistema funciona de la següent manera: Primer es genera un chatbot a partir d'un model BPMN. Un cop generat el chatbot, l'usuari ja pot interaccionar amb el chatbot via dialogflow o via Google Assistant.

9.1 Fase de generació del chatbot

La fase de generació del chatbot es refereix al moment de generar l'assistent virtual. És a dir, des de el moment en el qual es selecciona un model BPMN, fins que es construeix l'assistent virtual. En l'apartat Generació del chatbot s'explica a detall aquest procés. En la Fig. 17 es pot veure l'arquitectura de la generació del chatbot.

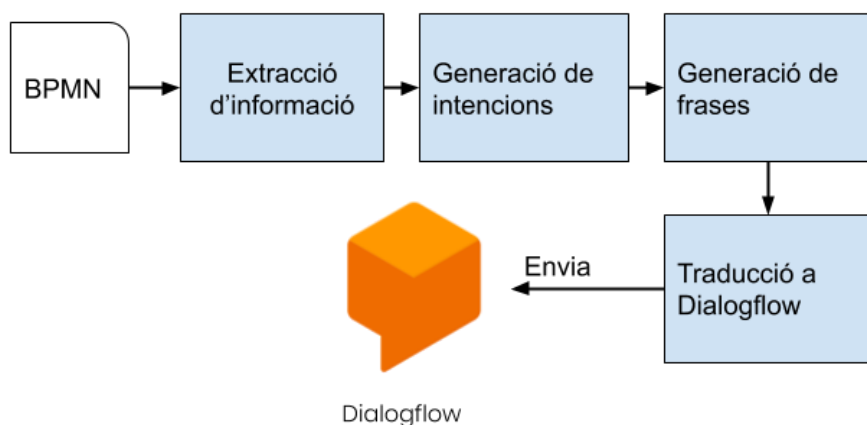


Figura 17: Arquitectura de la generació del chatbot

La fase de generació del chatbot comença amb la selecció d'un model BPMN. Seguidament, es llegeix i s'extreu la informació del model. A partir de la informació extreta es genera les intencions que l'usuari podrà indica a l'assistent virtual. Per tal d'augmentar la flexibilitat del llenguatge, es genera frases d'entrenament, és a dir, diferents formes d'indicar una mateixa idea. També es generen les frases de resposta. Finalment, s'utilitza dialogflow per tal de generar i entrenar l'assistent virtual. La lògica de l'assistent virtual es guarda en els servidors de dialogflow.

Cal destacar que s'ha integrat Google Assistant per poder utilitzar l'assistent de google. La integració, però, només està disponible per comptes de Google, autoritzats manualment.

9.2 Fase d'execució del chatbot

La fase d'execució es refereix al moment en el qual s'utilitza l'assistent virtual. És a dir, el moment en el qual un usuari es comunica amb l'assistent virtual. En l'apartat [Execució del chatbot](#) s'explica més a fons com es realitza l'execució i que succeeix quan s'utilitza el chatbot.

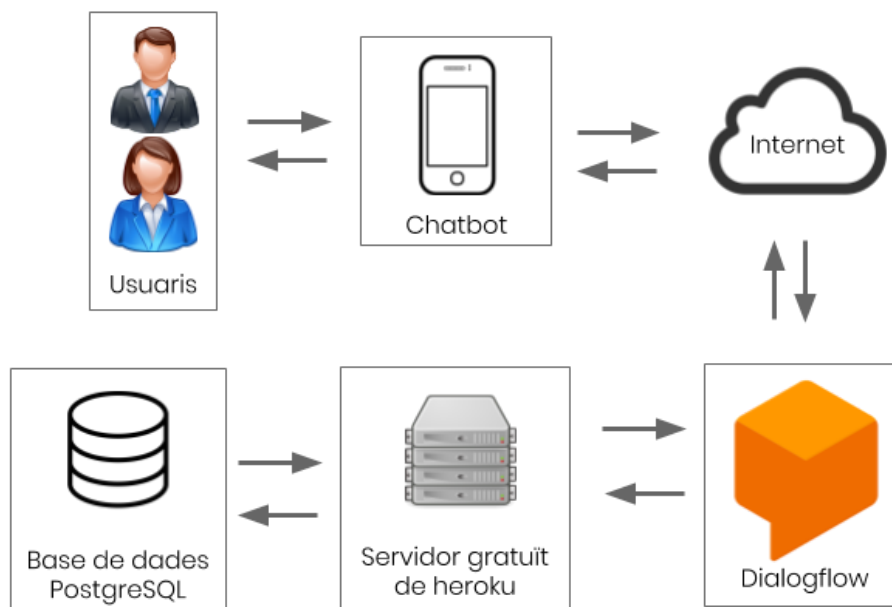


Figura 18: Arquitectura de la fase d'execució del chatbot

Una conversa amb el chatbot funciona de la següent manera. Primer l'usuari envia un missatge al chatbot, el qual pot ser el chatbot per defecte o una integració amb Google Assistant. Aquest missatge és enviat a dialogflow a través d'internet. És allà on es realitza el procés d'enteniment de l'usuari. Un cop dialogflow entén la intenció de l'usuari, s'envia aquesta informació a un servidor de heroku on fent ús de la base de dades, es decideix l'acció a realitzar. Finalment, es retorna la resposta que acaba arribant als usuaris.

9.3 UML

L'UML²² descriu les classes del sistema i com es relacionen entre elles. En la Fig. 19 es pot veure la jerarquia del concepte intenció. El diagrama només inclou els aspectes rellevants de les classes, els atributs, els mètodes i les seves associacions. Cada atribut que apareix,

²² "Class diagram - Wikipedia." https://en.wikipedia.org/wiki/Class_diagram. S'hi ha accedit el dia 25 de juny. 2020.

té els seus getters²³ i setters. En el diagrama hi ha una classe principal²⁴ “**myIntent**”, la resta són subclasses que hereten d'ella. Cada subclassa està associada a la intenció generada al tractar el component BPMN. Per exemple, Task és la intenció generada a partir de la component tasca d'un BPMN.

En el diagrama s'hi inclou els mètodes que s'usen en el moment de generar el chatbot. Les subclasses sobrecarregen²⁵ els mètodes de tal manera que quan s'executi, cada subclassa utilitzarà el seu propi mètode. Cal notar que el mètode “**translateIntoDialogFlow**”, el qual s'encarrega de generar la informació per fer funcionar el chatbot, no està sobrecarregat per cap subclassa. Això és degut al fet que s'utilitza un patró Plantilla²⁶. La idea és la següent:

En el moment de generar la informació pertinent, les subclasses utilitzaran el mètode **translateIntoDialogFlow** de la classe principal. Aquest utilitza els mètodes sobrecarregats de les subclasses. En cada mètode les subclasses trien el comportament a dur a terme. Per tant, les subclasses utilitzaran el codi definit en la superclasse, i els mètodes sobrecarregats. És important notar que en el mètode **translateIntoDialogFlow**, hi ha codi comú a tothom. Per tant, l'objectiu és permetre que cada subclassa pugui construir la informació de forma autònoma sense repetir codi. Fent així més mantenible el codi.

²³ "Getters and Setters in Java Explained - freeCodeCamp.org." 25 de gen.. 2020, <https://www.freecodecamp.org/news/java-getters-and-setters/>. S'hi ha accedit el dia 25 de juny. 2020.

²⁴ "Java Inheritance (Subclass and Superclass) - W3Schools."

https://www.w3schools.com/java/java_inheritance.asp. S'hi ha accedit el dia 25 de juny. 2020.

²⁵ "Overriding in Java - GeeksforGeeks." <https://www.geeksforgeeks.org/overriding-in-java/>. S'hi ha accedit el dia 25 de juny. 2020.

²⁶ "Template Method Design Pattern - SourceMaking."

https://sourcemaking.com/design_patterns/template_method. S'hi ha accedit el dia 25 de juny. 2020.

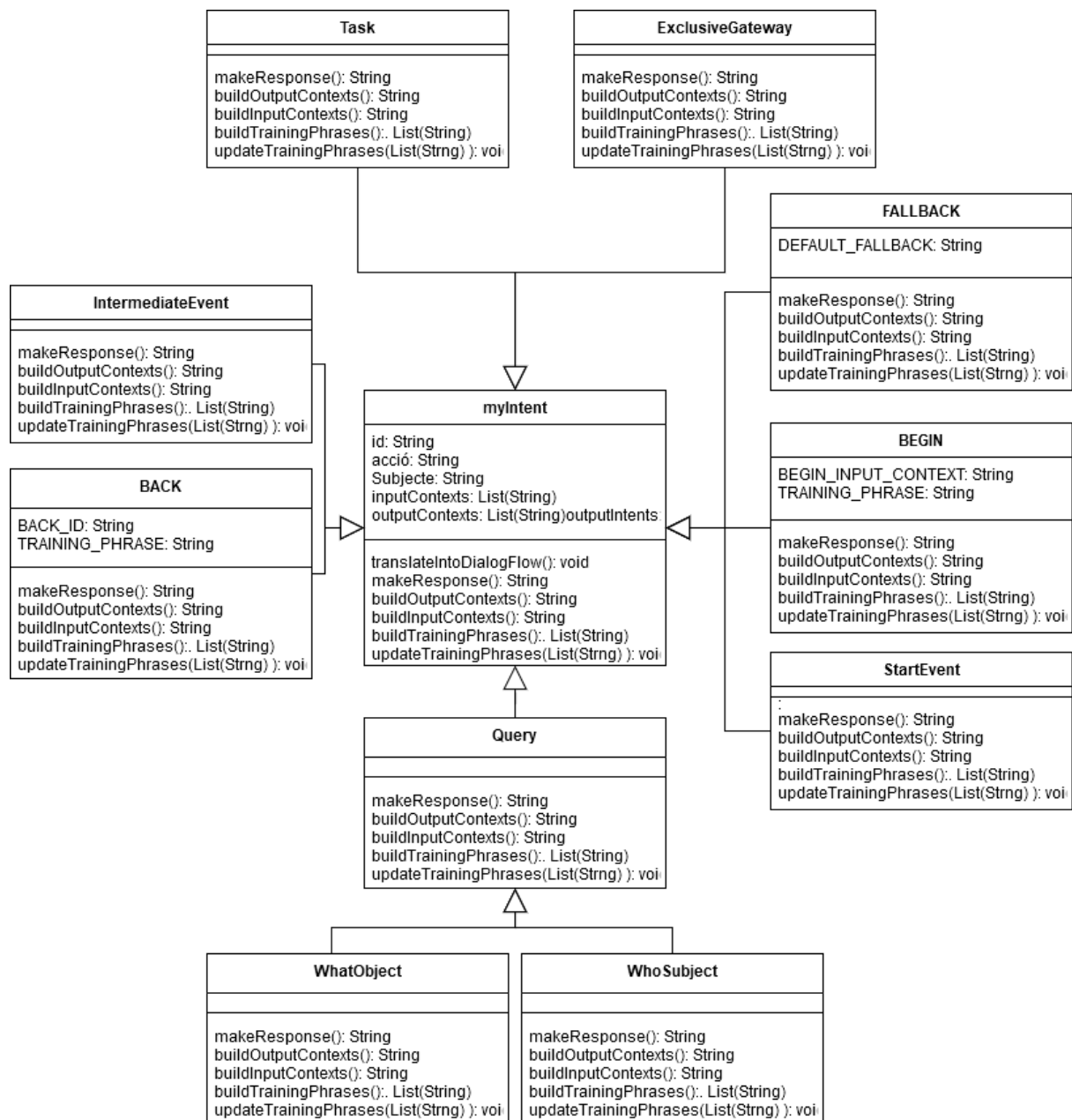


Figura 19: Jerarquia de classes de el concepte intenció. Només és mostra els metodes i atributs més rellevants. Els getters i setters són implícits

En la Fig. 20 es pot veure l'UML amb les parts rellevants de la generació d'una frase. Donat que un requisit no funcional del projecte és la mantenibilitat, s'utilitza el patró adaptador²⁷. Aquest utilitza Interfícies²⁸ per adaptar els serveis, fent-los això més canviabls. Tot i que no és mostra en l'UML, s'ha utilitzat la mateixa estratègia pel que fa a l'algoritme de processament. Per tant, seria fàcilment canviabls.

²⁷ "Adapter pattern - Wikipedia." https://en.wikipedia.org/wiki/Adapter_pattern. S'hi ha accedit el dia 25 de juny. 2020.

²⁸ "Java Interface - W3Schools." https://www.w3schools.com/java/java_interface.asp. S'hi ha accedit el dia 25 de juny. 2020.

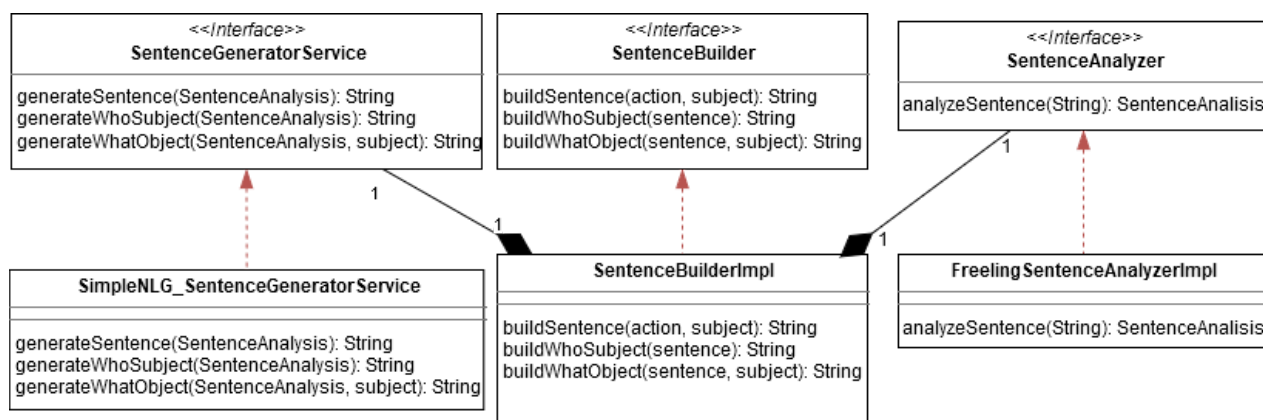


Figura 20: Diagrama UML de classes sobre la generació de una frase

Pel que fa la base de dades, només hi ha la taula intenció, que simula una pila d'execució. És a dir, enregistra les intencions de l'usuari. El contingut és el següent:

Intent(id: int, intenció:string)

10. Anàlisi d'opcions

En aquest apartat s'explica les opcions considerades i l'opció escollida per les fases de l'apartat 8. Les opcions considerades estan dividides en la lectura del model BPMN, generació de frases i generació del Chatbot.

10.1 Lectura del model BPMN

Per fer la lectura del model BPMN hi ha diverses opcions, aquestes són que l'estudiant faci la lectura o utilitzar una eina existent. Les eines més rellevants per la lectura dels models BPMN són Camunda²⁹ i Activiti³⁰.

La primera opció consisteix en el fet que l'estudiant programi la lectura des de zero. Això significa que la lectura està feta a mesura. Tot i això. Es perd molt de temps en construir i validar el lector. A més, té un major risc perquè pot resultar ser més complicat de l'esperat. Finalment, s'ha de considerar que aquesta opció no té el suport de cap comunitat.

La segona opció consisteix a utilitzar la llibreria Activiti. Activiti és una llibreria, d'ús gratuït, que permet llegir models BPMN. La llibreria no se centra només en els models BPMN, sinó que també dona suport a altres tipus de models. Aquesta opció comporta un risc baix. A més, utilitzar-la comporta estalviar temps, també es té el suport de la comunitat. Com a desavantatge. Aquesta eina no està feta a mesura i no està centrada en els models BPMN.

L'última opció consisteix a utilitzar la llibreria Camunda. Camunda és una llibreria, d'ús gratuït, que permet llegir models BPMN. La llibreria és un **"fork"** de l'opció Activiti. La llibreria arregla alguns errors fent-la més ràpida tal com es pot veure en un estudi³¹ que es va fer. A més, Camunda se centra més en els models BPMN³². Utilitzar camunda té els avantatges de què és una llibreria existent molt usada i contrastada, té el suport de la comunitat, està centrada en els models BPMN, s'estalvia feina i a més, la llibreria ha estat recomanada per part dels directors del TFG. Com a desavantatge, Camunda no està feta a mesura.

Tenint totes les opcions en consideració, l'opció escollida és Camunda perquè és l'opció més ràpida, més robusta i està centrada en els models BPMN. Té suport per part de la comunitat i suposa un risc baix. A més, ha estat recomanada pels directors del TFG.

²⁹ "Camunda BPM." <https://camunda.com/>. S'hi ha accedit el dia 15 de maig. 2020.

³⁰ "Activiti." <https://www.activiti.org/>. S'hi ha accedit el dia 15 de maig. 2020.

³¹ "Micro-Benchmarking BPMN 2.0 Workflow Management" <https://www.iaas.uni-stuttgart.de/publications/INPROC-2016-05-Micro-Benchmarking-BPMN-2.0-WfM-S-with-Workflow-Patterns.pdf>. S'hi ha accedit el dia 15 de maig. 2020.

³² "Reasons for Migration from Activiti to Camunda." 18 de set.. 2018, <https://www.attuneww.com/blogs/reasons-for-migration-from-activiti-to-camunda.html>. S'hi ha accedit el dia 15 de maig. 2020.

En la Taula 14 es pot veure un resum comparatiu entre les diferents opcions considerades.

	Estudiant	Activiti	Camunda
Opció Gratuïta(€)	SÍ	SÍ	SÍ
Fet a mesura	SÍ	NO	NO
Risc	Mig	Baix	Baix
Facilitat	Mig	Alta	Alta
Resultats	Bons	Bons	Bons
Suport dels creadors	SÍ	NO	NO
Suport de la comunitat	NO	SÍ	SÍ
Centrant als BPMN	SÍ	NO	SÍ
Recomanació dels directors	NO	NO	SÍ

Taula 14: Resum comparatiu entre els lectors de models BPMN

10.2 Generació de frases

Pel que fa a la generació de frases hi ha tres apartats a considerar: L'analitzador de frases, la construcció de frases i la generació de frases similars.

10.2.1 Introducció

La correctesa de les frases del chatbot no és fàcil d'aconseguir. Les frases de les tasques dels models BPMN estan en infinitiu. En la Fig. 21 es pot veure com els verbs "Process" i "Charge" estan en infinitiu. A més, el subjecte de l'acció, el "Bank", no està inclòs en la frase, sinó que engloba a totes les tasques que realitza. Per tant, per generar la frase "The bank charges the payment information", s'ha d'analitzar la frase "Process payment information" i extreure'n el verb, l'objecte i els complements. Després, fent ús de la informació extreta i el subjecte es pot construir la frase desitjada.

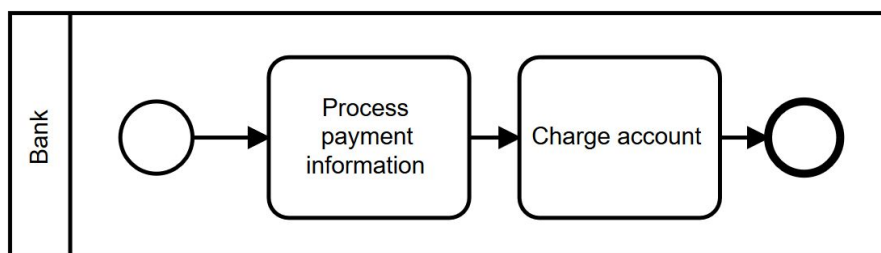


Figura 21: Model BPMN on es mostra com són les frases

Finalment, per tal d'aconseguir una flexibilitat en el llenguatge, s'ha de generar frases similars a la prèviament construïda. En l'exemple anterior, una frase similar de "The

bank charges the payment information” podria ser “The financial institution charges the transaction details”.

Per tant, generació de frases, hi ha 3 apartats a considerar, l'analitzador de frases, la construcció de frases i la generació de frases similars.

10.2.2 Analitzador de frases

Per analitzar les frases, s'han tingut en compte diverses opcions. Aquestes són, que el faci l'estudiant, utilitzar CoreNLP³³, utilitzar OpenNLP³⁴ o utilitzar Freeling³⁵.

La primera opció consisteix en el fet que l'estudiant realitza l'analitzador de frases. Es tracta d'una tasca molt complexa i que comporta un risc elevat. Donada la complexitat, es trigaria molt de temps en obtenir un resultat decent. Tot i això, existiria l'avantatge que l'analitzador estaria fet a mesura. Evidentment no hi hauria cap mena de suport per part de la comunitat.

La segona opció consisteix en utilitzar CoreNLP (Stanford). Aquesta eina és una llibreria molt utilitzada, robusta i amb un ampli suport de la comunitat. Proporciona més funcionalitats que la d'analitzar les frases. L'eina és gratuïta i té un alt suport per una gran varietat de llenguatges de programació. És la llibreria més usada en java juntament amb OpenNLP³⁶. I entre aquestes dues, CoreNLP es la que més activitat té³⁷. Com a desavantatge, els creadors no ofereixen molt suport. Tot i això, si es pot obtenir suport de la comunitat. Pel que fa al projecte, ofereix totes les funcionalitats necessàries.

La tercera opció consisteix en utilitzar OpenNLP (Apache). Aquesta eina és una llibreria molt utilitzada, robusta i amb un ampli suport de la comunitat. Proporciona més funcionalitats que la d'analitzar les frases. L'eina és gratuïta i open source. A més, té un ampli suport per una gran varietat de llenguatges de programació. És la llibreria més usada en java juntament amb CoreNLP. Com a desavantatge, els creadors no ofereixen molt suport, tot i això, si es pot obtenir suport de la comunitat. A més, OpenNLP no té una velocitat de desenvolupament tan ràpida com CoreNLP. Igual que CoreNLP, ofereix les funcionalitats que requereix aquest TFG.

L'última opció consisteix en utilitzar Freeling. Aquesta és una eina que permet analitzar frases i obtenir informació tal com el subjecte o el predicat de la frase. D'entre els avantatges de freeling es troba que és un projecte d'ús gratuït i open source, que usar-la

³³ "Stanford CoreNLP - Stanford NLP Group." <https://stanfordnlp.github.io/CoreNLP/>. S'hi ha accedit el dia 15 de maig. 2020.

³⁴ "Apache OpenNLP." <https://opennlp.apache.org/>. S'hi ha accedit el dia 15 de maig. 2020.

³⁵ "Welcome | FreeLing Home Page - TALP - UPC." <http://nlp.lsi.upc.edu/freeling/index.php/node/1>. S'hi ha accedit el dia 15 de maig. 2020.

³⁶ "Most used Java libraries, frameworks, and APIs in big data" 2 de març. 2017, <https://itnext.io/most-used-java-libraries-frameworks-and-apis-in-big-data-projects-part-2-72263d03d147>. S'hi ha accedit el dia 11 de maig. 2020.

³⁷ "CoreNLP vs Apache OpenNLP - Awesome Java - LibHunt." <https://java.libhunt.com/compare-corenlp-vs-apache-opennlp>. S'hi ha accedit el dia 12 de maig. 2020.

dóna continuïtat al grup de recerca i que es té el suport del grup. Com a desavantatge no és una eina tan usada com pot ser CoreNLP o OpenNLP, i per tant, no hi ha gran suport de la comunitat. Igual que CoreNLP i OpenNLP, ofereix les funcionalitats que requereix aquest TFG. Cal destacar que per utilitzar-la és fara ús d'un docker.

L'opció de crear des de zero l'analitzador és massa costosa, per tant es farà ús una llibreria existent. D'entre les llibreries esmentades anteriorment, totes serveixen per al que es vol aconseguir. Pel que fa al suport, CoreNLP i OpenNLP tenen molt suport per part de la comunitat. FreeLing en canvi, té més suport per part dels creadors. Per tant, considerant els avantatges i desavantatges de cada opció s'ha decidit usar FreeLing per tal de donar continuïtat als treballs del grup on es realitza aquest TFG.

En la Taula 15 es pot veure un resum comparatiu entre les diferents opcions considerades.

	Estudiant	CoreNLP	OpenNLP	Freeling
Opció Gratuïta(€)	SÍ	SÍ	SÍ	SÍ
Fet a mesura	SÍ	NO	NO	NO
Risc	Alt	Baix	Baix	Baix
Facilitat	Difícil	Mig	Mig	Mig
Resultats	Mitjans	Bons	Bons	Bons
Suport dels creadors	SÍ	NO	NO	SÍ
Suport de la comunitat	NO	SÍ	SÍ	NO

Taula 15: Resum comparatiu entre els analitzadors de frases

10.2.3 Construcció de frases

Per tal que el chatbot pugui parla i respondre amb frases correctes gramaticalment, s'ha de construir les frases de forma apropiada. Per això s'ha inclòs un generador de frases. Per al generador de frases hi ha diferents opcions: L'estudiant el programa, utilitzar SimpleNLG³⁸ i utilitzar RosaeNLG³⁹.

La primera opció consisteix en que l'estudiant programa el constructor de frases. Es tracta d'una tasca molt complexa i que comporta un risc elevat, el que podria augmentar el temps del projecte. Tot i això, existiria l'avantatge que estaria fet a mesura.

La segona opció consisteix en fer ús de l'eina SimpleNLG. Es tracta d'una eina la qual permet generar frases senzilles a partir d'una anàlisi sintàctic (Subjecte, Predicat, Objecte, Complement). Com a avantatges, l'eina és gratuïta, és robusta i bastant usada. Com a desavantatges, l'eina no esta feta a mesura i les limitacions de l'eina limiten les frases del chatbot.

³⁸ "simplenlg/simplenlg: Java API for Natural Language ... - GitHub."

<https://github.com/simplenlg/simplenlg>. S'hi ha accedit el dia 15 de maig. 2020.

³⁹ "Change log of RosaeNLG :: RosaeNLG // Docs."

<https://rosaelng.org/rosaelng/1.14.2/changelog.html>. S'hi ha accedit el dia 15 de maig. 2020.

L'última opció consisteix en fer ús de l'eina RosaeNLG. Es tracta d'una eina, open source, la qual permet generar frases senzilles a partir d'indicar-li el contingut de les frases. Per indicar l'estructura de les frases utilitza el llenguatge PUG⁴⁰. Tot i això, aquesta eina no ofereix suport dels creadors, és poc utilitzada, no té bona documentació, s'hauria d'aprendre PUG i seria difícil adaptar-la al projecte.

L'opció de crear des de zero el generador és massa costosa, per tant es farà ús una llibreria existent. D'entre les llibreries esmentades anteriorment, SimpleNLG és la que millor encaixa amb l'analitzador de frases. A més, SimpleNLG té molt suport per part de la comunitat. En canvi, RosaeNLG no en té tant. Per tant, considerant els avantatges i desavantatges de cada opció s'ha decidit usar SimpleNLG donat que és l'opció que millor s'ajusta al temps i a les necessitats del projecte.

En la Taula 16 es pot veure un resum comparatiu entre les diferents opcions considerades.

	Estudiant	RosaeNLG	SimpleNLG
Opció Gratuïta(€)	SÍ	SÍ	SÍ
Fet a mesura	SÍ	NO	NO
Risc	Alt	Mig	Baix
Facilitat	Baixa	Mig	Alta
Resultats	Mitjans	Bons	Bons

Taula 16: Resum comparatiu entre els constructors d'una frase

10.2.4 Generació de frases similars

Per tal d'obtenir un diàleg flexible, i per tal que de què el chatbot entengui la intenció de l'usuari, s'ha de generar un conjunt de frases similars. Aquestes frases s'utilitzen com a frases d'entrenament, és a dir, frases a partir de les quals l'assistent virtual entendrà la intenció de l'usuari. Per generar aquestes frases s'han considerat diverses opcions, aquestes són: L'estudiant fa el generador de frases similars, traducció de frases, utilitzar SpinnerChief⁴¹, utilitzar SpinBot⁴² o utilitzar QuillBot⁴³.

La primera opció consisteix en el fet que l'estudiant programa el generador. Això suposa que donada una frase, aquesta s'analitza, i s'obté sinònims per cada component de la frase. Llavors, s'avalua la correctesa d'aquests sinònims i finalment, es genera una nova frase respectant l'estructura i els temps verbals de la frase original. Tot plegat comporta una gran inversió de temps. Com a avantatge, estaria fet a mida.

La segona opció utilitzaria un servei de traducció. El procés per traduir una frase seria el següent: Primer de tot, la frase s'aniria traduint a altres idiomes. Seguidament, es

⁴⁰ "Pug - GitHub." <https://github.com/pugjs/pug>. S'hi ha accedit el dia 15 de maig. 2020.

⁴¹ "Spinner Chief 6." <https://www.spinnerchief.com/>. S'hi ha accedit el dia 15 de maig. 2020.

⁴² "Spinbot API - Spinbot.com - Article Spinning, Text Rewriting" <https://spinbot.com/api>. S'hi ha accedit el dia 15 de maig. 2020.

⁴³ "QuillBot | AI Paraphrasing Tool." <https://quillbot.com/>. S'hi ha accedit el dia 15 de maig. 2020.

traduiria la frase traduïda a l'idioma original. Finalment, s'utilitzaria la nova frase com a frase similar. Notar que aquesta opció aprofita el fet que normalment en traduir una frase de tornada a l'idioma font s'obté una frase diferent de l'original. És a dir, traduir una frase a un altre idioma, i tornar-la a traduir a l'idioma original dóna com a resultat una frase molt similar, però diferent, a la inicial. L'avantatge d'aquesta opció és que és fàcil d'implementar. Tot i això, té el risc de ser inestable

La tercera opció utilitza SpinnerChief com a eina per generar frases similars a partir d'una frase existent. El funcionament de SpinnerChief és el següent: Donada una frase existent, et retorna una frase similar. També pot donar opcions de paraules sinònimes. Per exemple, si s'envia la frase "Who requests the items?", l'API retorna "{Who|Who else} requests the {items|things|products}?". És a dir, ha substituït la paraula Who per Who|Who else i la paraula items per items|things|products. Per tant, aquesta eina es pot utilitzar per generar combinacions de frases similars tals com "Who requests the products?" o "Who else requests the items?". Notar que l'ús d'aquesta eina suposaria implementar una forma de generar frases a partir de les opcions que dóna SpinnerChief.

Com a desavantatge l'API té poca documentació i dóna poc suport. A més, la versió gratis és molt limitada (20 consultes diàries, 1000 paraules per consulta), i per tant, s'hauria d'implementar una forma de coordinar la generació de les frases d'entrenament. La versió Prèmium, sense limitacions, és una molt bona opció en cas que es pugui pagar el cost. Però no és el cas, ja que no hi ha pressupost.

La quarta opció és utilitzar SpinBot com a eina per generar frases similars a partir d'una frase existent. És una eina decent però pitjor que Spinnerchief, ja que no permet l'opció de retornar opcions per les paraules. Tampoc té versió gratuïta. Seria una opció decent en cas que es pogués pagar el cost. I en tot cas abans seria millor escollir a SpinnerChief.

L'última opció és utilitzar Quillbot com a eina per generar frases similars a partir d'una frase existent. És una eina decent però té el mateix problema que SpinBot, no dóna diverses opcions per les paraules. No té versió gratuïta. Seria una opció decent en cas que es pogués pagar el cost. I en tot cas abans seria millor escollir a SpinnerChief.

A l'hora de triar una eina i donat que el projecte no disposa de pressupost, només es tenen en consideració les opcions gratuïtes. Aquestes són l'opció d'implementar-ho l'estudiant, Sistema de traducció i l'opció de SpinnerChief.

Entre les opcions gratuïtes s'ha decidit implementar SpinnerChief perquè és l'opció gratuïta que permet generar més frases i de millor qualitat. És cert que SpinnerChief té el desavantatge que té una API limitada i cal processar la resposta (SpinnerChief dóna opcions per cada paraula) però en canvi té una versió gratuïta i és l'eina que dóna millors resultats. Finalment, SpinnerChief és millor opció que fer-ho l'estudiant, ja que comporta menys riscos i s'adapta millor al temps disponible que es pot dedicar a aquest apartat.

En la Taula 17 es pot veure un resum comparatiu entre les diferents opcions considerades

	Estudiant	Traducció	QuillBot	SpinBot	SpinnerChief
Opció Gratuïta(€)	SÍ	SÍ	NO	NO	SÍ
Fet a mesura	SÍ	NO	NO	NO	NO
Risc	Alt	Baix	Mig	Mig	Mig
Temps	Molt	Poc	Mig	Mig	Mig
Facilitat	Poca	Alta	Alta	Alta	Mitja
Resultats	Mitjans	Mitjans	Bons	Bons	Millors*

Taula 17: Comparativa entre les diferents opcions

10.3 Generació del Chatbot

Pel que fa a la generació del Chatbot hi ha diverses eines que permeten generar un chatbot, aquestes són Xatkit⁴⁴, Dialogflow⁴⁵, LUIS⁴⁶. En aquest apartat es comparen^{47 48} i es decideix quina eina utilitzar.

Cal mencionar que no s'ha considerat que l'estudiant programi el generador de chatbots, ja que es passa de l'abast del projecte. Tampoc s'ha considerat AIML.

10.3.1 DialogFlow

Dialogflow és una eina de modelatge de chatbots. Permet construir chatbots i integrar-los en els negocis. DialogFlow ofereix una versió gratuïta la qual és suficient per a aquest projecte.

Els chatbots de DialogFlow funcionen de manera similar a l'explicada en l'apartat [Assistents virtuals](#). És a dir, funcionen a partir de la **intenció** de l'usuari i el **context** de la conversació. Segons el context i la intenció de l'usuari, el chatbot es comporta diferent. És a dir, encara que un usuari digui la mateixa frase en contextos diferents, això provocarà que el chatbot respongui diferent.

A més, dialogflow et permet entrenar el chatbot proporcionant frases que l'usuari pot dir. Aquestes frases s'anomenen **frases d'entrenament**. El seu funcionament és el següent: A partir de les frases d'entrenament proporcionades, dialogflow n'extreu el significat semàntic. D'aquesta manera quan l'usuari digui una frase que tingui un significat similar a les frases proporcionades, el chatbot sabrà com ha de comportar-se.

⁴⁴ "Xatkit - Easiest way to get, host and monitor your own" <https://xatkit.com/>. S'hi ha accedit el dia 15 de maig. 2020.

⁴⁵ "Dialogflow." <https://dialogflow.com/>. S'hi ha accedit el dia 15 de maig. 2020.

⁴⁶ "LUIS.ai." <https://www.luis.ai/>. S'hi ha accedit el dia 15 de maig. 2020.

⁴⁷ "Dialogflow vs Lex vs LUIS vs Watson vs Chatfuel - Mining" 8 d'oct.. 2018, <https://miningbusinessdata.com/dialogflow-vs-lex-vs-luis-vs-watson-vs-chatfuel/>. S'hi ha accedit el dia 12 de maig. 2020.

⁴⁸ "Understanding the Differences Between Alexa, API.ai, WIT.ai" 17 d'abr.. 2017, <https://www.linkedin.com/pulse/understanding-differences-between-different-ai-platforms-abraham-kaing>. S'hi ha accedit el dia 12 de maig. 2020.

Cal destacar que els assistents virtuals de dialogflow estan basats en el que ells anomenen “**Intent**”. Els intents representen una **intenció** de l'usuari. Aquesta intenció té frases d'entrenament, frases de resposta, i contextos. Els contextos poden ser **d'entrada** i de **sortida**. Un context d'entrada és un context que ha d'estar actiu perquè l'**intent** que el conte pugui ser activat. Un context de sortida és un context que s'activa quan l'intent que el conte és activat. Els contextos de sortida tenen un **temps de vida** el qual està definit com el nombre d'interaccions amb l'usuari. És a dir, si un context de sortida té un temps de vida igual a 5, significa que un cop activat, està actiu durant 5 interaccions amb l'usuari. A la sisena interacció, el context deixarà d'estar actiu.

En general, els chatbots construïts per DialogFlow són chatbots senzills modelats de forma manual. En quant els chatbots empresarials, el comportament acostuma a ser el mateix: L'usuari demana un servei, el chatbot li dona opcions i l'usuari va triant. Finalment, i a partir de les dades extretes, és contracta el servei.

Pel que fa a els casos d'ús de la utilització de dialogflow per crear chatbots, el més famós és el de l'empresa Domino's Pizza ⁴⁹. Una conversa del seu chatbot pot ser la de la Fig. 22. En la figura es pot observar com el chatbot li dona dues opcions: “**Place an order**” o “**Track an order**”. L'usuari n'escull una. Seguidament, el chatbot pregunta el tipus de la petició, i una altra vegada li dona unes opcions. En aquest cas són “**Easy Order**”, “**Most recent Order**” i “**New Order**”. Finalment, l'usuari tria una d'aquestes opcions.

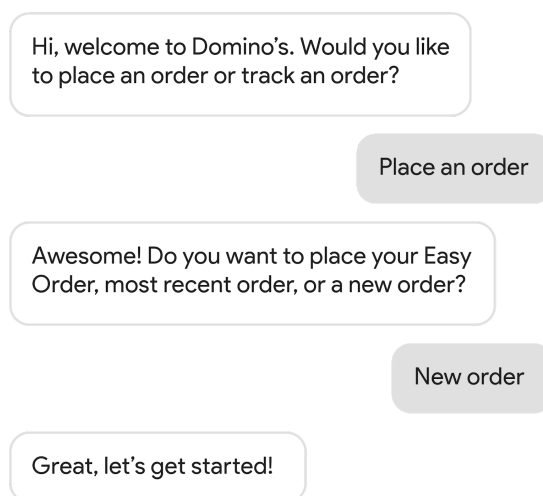


Figura 22: Conversa del chatbot de Domino's Pizza. Imatge extreta de <https://dialogflow.com/images/case-studies/dominos/dominos-001.png>

En conclusió, els chatbots creats amb dialogflow són chatbots, manuals, molt guiats i orientats a la contractació d'un servei. Cal destacar que totes les eines de modelat de

⁴⁹ "Case studies - Dialogflow." <https://dialogflow.com/case-studies>. S'hi ha accedit el dia 12 de maig. 2020.

chatbot ofereixen unes capacitats similars a dialogflow. Per tant, la generació automàtica mitjançant DialogFlow o qualsevol altra eina pot esdevenir un repte complicat.

10.3.2 Xatkit

Xatkit és una eina de modelatge de Chatbots que implementa DialogFlow. La idea darrere Xatkit aprofitar la potencia dialogflow sense haver d'implementar ni gestionar les crides a l'API de Dialogflow. El funcionament de Xatkit hauria de ser igual que el de DialogFlow. Per tant, tot el que es pot fer en DialogFlow hauria de ser possible en Xatkit.

Com a desavantatge, Xatkit és una tecnologia molt recent i encara no té una comunitat que la utilitza.

Cal destacar que Xatkit està relacionat amb el grup on es realitza aquest TFG, i per tant, es tindria el suport de Xatkit.

En conclusió, utilitza Xatkit hauria d'estalviar feina de cara al projecte, ja que permet fer ús de DialogFlow sense la necessitat d'integrar-lo.

10.3.3 Luis

LUIS (Language Understanding Intelligent Service) és molt similar a DialogFlow perquè també permet entrenar el chatbot a partir de les frases d'entrenament. De la mateixa manera, també permet entendre la intenció de l'usuari a partir del context.

Una de les grans diferències respecte dialogflow és que LUIS dóna més informació sobre el seu funcionament intern. Per cada frase que diu l'usuari, LUIS indica quines intencions ha detectat i quant n'està de segur. Això és un gran avantatge perquè permet dissenyar un chatbot més robust.

LUIS ofereix una versió gratuïta ⁵⁰ bastant limitada, però prou gran perquè es pugui treballar amb ella en aquest projecte.

10.3.4 Decisió inicial: Xatkit

S'ha decidit utilitzar Xatkit perquè es pot aprofitar el potencial de DialogFlow sense haver d'implementar tota la infraestructura per suportar-lo. A més, seria senzill integrar-lo amb altres serveis perquè Xatkit té planejat incloure altres processadors de llenguatge natural com LUIS o el propi que Xatkit està desenvolupant. Finalment menciona que és totalment gratuït, i que Xatkit està relacionat amb el grup on es realitza aquest projecte.

En la Taula 18 es pot observar un resum comparatiu entre les diferents opcions considerades.

⁵⁰ "Cognitive Services pricing—Language Understanding."
<https://azure.microsoft.com/en-us/pricing/details/cognitive-services/language-understanding-intelligent-services/>. S'hi ha accedit el dia 12 de maig. 2020.

	Xatkit	LUIS	DialogFlow
Opció Gratuïta(€)	SÍ	SÍ, però limitada	SÍ
Fet a mesura	NO	NO	NO
Risc	Mig	Mig	Mig
Temps	Mig	Alt	Alt
Facilitat	Mitja	Mitja	Mitja
Resultats	Bons	Bons	Bons
Suport dels creadors	SÍ	NO	NO
Suport de la comunitat	NO	SÍ	SÍ

Taula 18: Resum comparatiu entre les eines de modelat de chatbots

10.3.5 Nova decisió: DialogFlow

Xatkit ha resultat ser una eina no satisfactòria. En principi implementaven les funcionalitats de DialogFlow però no ha estat així. Per exemple, restringeixen algunes funcionalitats de dialogflow com ara la d'eliminar contextos en temps d'execució. Aquesta funcionalitat és vital per aquest projecte el qual genera el chatbot automàticament. També li ha mancat molta documentació bàsica (de com usar Xatkit) i els creadors de l'aplicació no han donat el suport necessari. Per tant, s'ha decidit canviar d'opció i implementar la infraestructura necessària per utilitzar DialogFlow.

En la Taula 19 es pot observar el nou resum comparatiu entre les diferents opcions considerades.

	Xatkit	LUIS	DialogFlow
Opció Gratuïta(€)	SÍ	SÍ, però limitada	SÍ
Fet a mesura	NO	NO	NO
Risc	Alt	Mig	Mig
Temps	Alt	Alt	Alt
Facilitat	Poca	Mitja	Mitja
Resultats	Mitjans	Bons	Bons
Suport dels creadors	Poc	NO	NO
Suport de la comunitat	NO	SÍ	SÍ

Taula 19: Nou resum comparatiu entre les opcions de modelat de chatbots

11. Recursos informàtics utilitzats

El projecte s'ha programat principalment en java. Pel que fa a les eines utilitzades aquestes han estat IntelliJ⁵¹, Maven⁵², Spring Boot⁵³, Heroku⁵⁴, PostgreSQL⁵⁵ i Docker⁵⁶.

Cal remarcar que els recursos descrits en aquest apartat són complementàries als esmentats en l'apartat [Anàlisis d'opcions](#). Concretament, aquest apartat se centra en entorns de treball, llenguatges de programació i infraestructures utilitzades.

11.1 Llenguatge de programació

Tot el codi s'ha realitzat en Java excepte la gestió amb la base de dades, que s'ha fet amb SQL.

11.1.1 Java ⁵⁷



Java és un llenguatge de programació, orientat a objectes, que ofereix una alta capacitat de manteniment de codi. El llenguatge és molt usat per la seva alta reutilització de codi.

S'ha decidit utilitzar aquest llenguatge perquè es busca que el codi sigui fàcil de mantenir. A més, el llenguatge ofereix una gran varietat d'elements de suport, el que pot disminuir el temps d'implementació. A més, Java és fàcilment portable a altres sistemes operatius.

⁵¹ "IntelliJ IDEA: The Java IDE for Professional" <https://www.jetbrains.com/idea/>. S'hi ha accedit el dia 22 de juny. 2020.

⁵² "Maven – Welcome to Apache Maven." <https://maven.apache.org/>. S'hi ha accedit el dia 22 de juny. 2020.

⁵³ "Spring Boot." <https://spring.io/projects/spring-boot>. S'hi ha accedit el dia 22 de juny. 2020.

⁵⁴ "Heroku." <https://www.heroku.com/>. S'hi ha accedit el dia 22 de juny. 2020.

⁵⁵ "Postgres - SQL Database Service | Heroku." <https://www.heroku.com/postgres>. S'hi ha accedit el dia 22 de juny. 2020.

⁵⁶ "Docker." <https://www.docker.com/>. S'hi ha accedit el dia 22 de juny. 2020.

⁵⁷ "Java (software platform) - Wikipedia." [https://en.wikipedia.org/wiki/Java_\(software_platform\)](https://en.wikipedia.org/wiki/Java_(software_platform)). S'hi ha accedit el dia 22 de juny. 2020.

11.1.2 SQL ⁵⁸



SQL (Structured Query Language) és un llenguatge de programació, dissenyat per administrar i recupera informació de bases de dades relacionals. Permet realitzar consultes per aconseguir i modificar informació de les bases de dades

S'utilitza ja que és el llenguatge, de bases de dades, més utilitzat i perquè en el projecte es requereix realitzar consultes sobre la informació emmagatzemada.

11.2 Eines

El projecte utilitza diverses eines per tal de desenvolupar el codi. Des de entorns de treball com pot ser IntelliJ, fins a servidors, amb base de dades, com pot ser heroku.

11.2.1 IntelliJ



IntelliJ IDEA és un entorn de desenvolupament integrat (IDE) pel desenvolupament de programes informàtics. Es tracta d'una excel·lent eina que inclou funcionalitats com poden ser suggeriments intel·ligents per l'autocompletació de codi, disseny intuïtiu o funcionalitats específiques segons el llenguatge utilitzat. A més, també ofereix suport pel que fa al control de versions del codi.

S'ha decidit utilitzar IntelliJ perquè ofereix moltes funcionalitats que fan més eficient la programació.

11.2.2 Maven



Maven és una eina de software per a la gestió i construcció de projectes. En el context del projecte, Maven gestiona les llibreries i el codi de tercers, de tal manera que facilita la integració al programa.

S'ha decidit utilitzar Maven perquè facilita molt la instal·lació i gestió del codi de tercers.

11.2.3 Spring Boot

⁵⁸ "SQL - Wikipedia." <https://en.wikipedia.org/wiki/SQL>. S'hi ha accedit el dia 22 de juny. 2020.



Spring Boot és un entorn de programari lliure utilitzat per crear aplicacions, que s'executaran en un servidor. Spring Boot separa la gestió del servidor de la lògica del codi.

S'ha decidit utilitzar Spring Boot perquè permet programar serveis, centrant-se només en programar l'aspecte lògic.

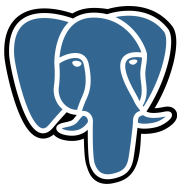
11.2.4 Heroku



HEROKU Heroku és una plataforma que ofereix servidors remots (en internet) als desenvolupadors. Aquests poden personalitzar, i utilitzar els servidors per a executar-hi una aplicació.

S'ha decidit utilitzar Heroku perquè ofereix un servidor, remot i gratuït, on és podrà executar l'aplicació feta amb Spring Boot.

11.2.5 PostgreSQL



PostgreSQL és un software que implementa un sistema gestor de bases de dades relacionals. És una base de dades robusta i eficient. Igual que tots els gestors de bases de dades, PostgreSQL implementa tots els elements SQL relacionals.

S'ha decidit utilitzar PostgreSQL per a tenir una base de dades robusta, la qual pugui emmagatzema tota la informació requerida.

11.2.6 Docker



Docker facilita el desplegament d'aplicacions encapsulant-les dintre de contenidors, anomenats dockers, proporcionant així una capa que permet executar l'aplicació en qualsevol sistema operatiu.

S'utilitza un docker perquè un dels programes que s'utilitza ([FreeLing](#)) està dintre d'un docker.

12. Generació del chatbot

En l'apartat Fase de generació del chatbot, és mostra l'arquitectura general de l'aplicació en el moment en el qual es genera el chatbot. En la Fig. 17 es pot veure el procés que realitza l'aplicació des de el moment en el qual es selecciona un model BPMN, fins al moment en el qual es construeix l'assistent virtual. En aquest apartat es desenvolupa més a fons el procés de generació de l'assistent virtual.

La fase de construcció comença amb la selecció d'un model BPMN. Seguidament, es llegeix i s'extreu la informació del model. A partir de la informació extreta es genera les intencions que l'usuari podrà indica a l'assistent virtual. Per tal d'augmentar la flexibilitat del llenguatge, es genera frases d'entrenament, és a dir, diferents formes d'indicar una mateixa idea. També es generen les frases de resposta. Finalment, s'utilitza dialogflow per tal de generar i entrenar l'assistent virtual. La lògica de l'assistent virtual es guarda en els servidors de dialogflow.

Al llarg dels següents subapartats, es descriu en més detall cada component d'aquesta arquitectura.

12.1 Selecció del BPMN

En la Fig. 23 es pot veure la interfície gràfica que permet seleccionar un model fent clic en **“Select BPMN model”** i generar el chatbot fent clic en **“Generate chatbot”**.

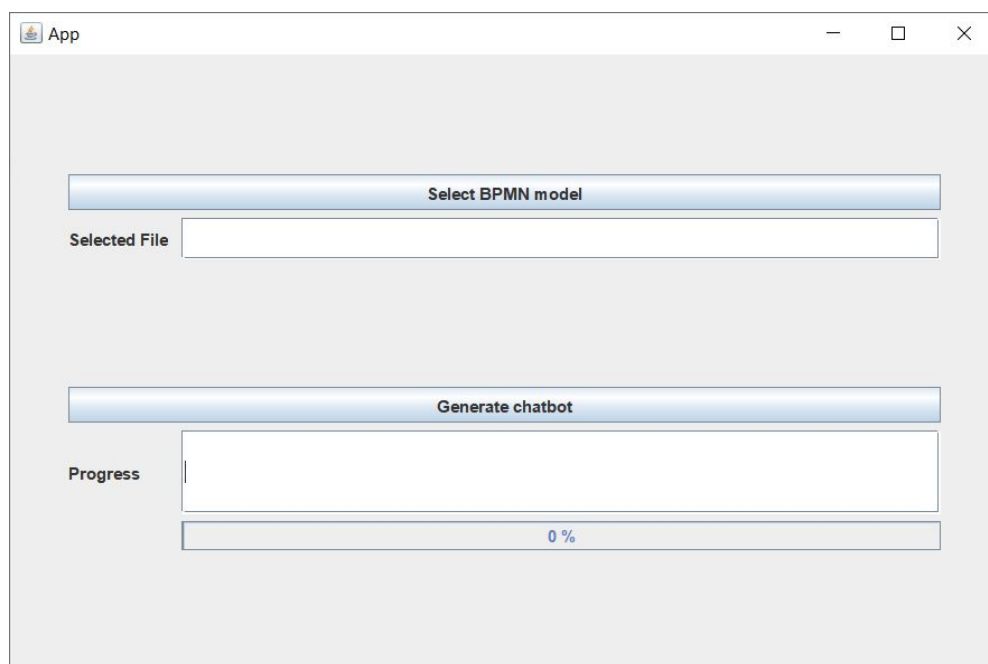


Figura 23: Interfície gràfica per seleccionar un model

El procés comença quan un usuari selecciona un model BPMN fent clic en “**Select BPMN model**”. Seguidament, s’obre una finestra on es pot seleccionar un model BPMN. En la Fig. 24 es pot veure una foto del resultat.

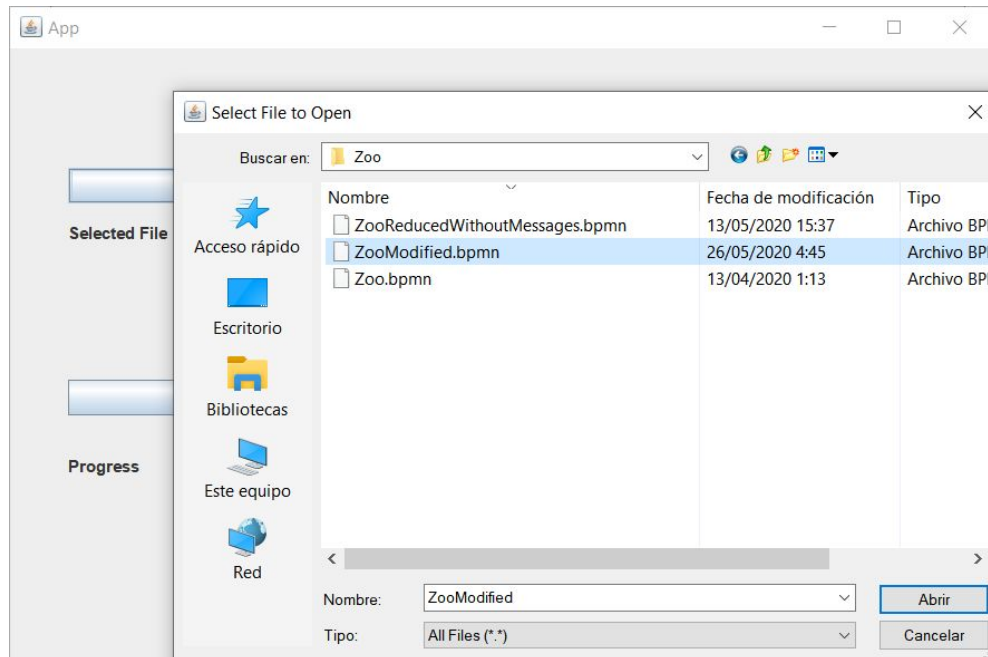


Figura 24: Selecció d'un arxiu BPMN mitjançant la interfície gràfica.

Un cop seleccionat un model BPMN, l'usuari veu una imatge com la de la Fig. 25.

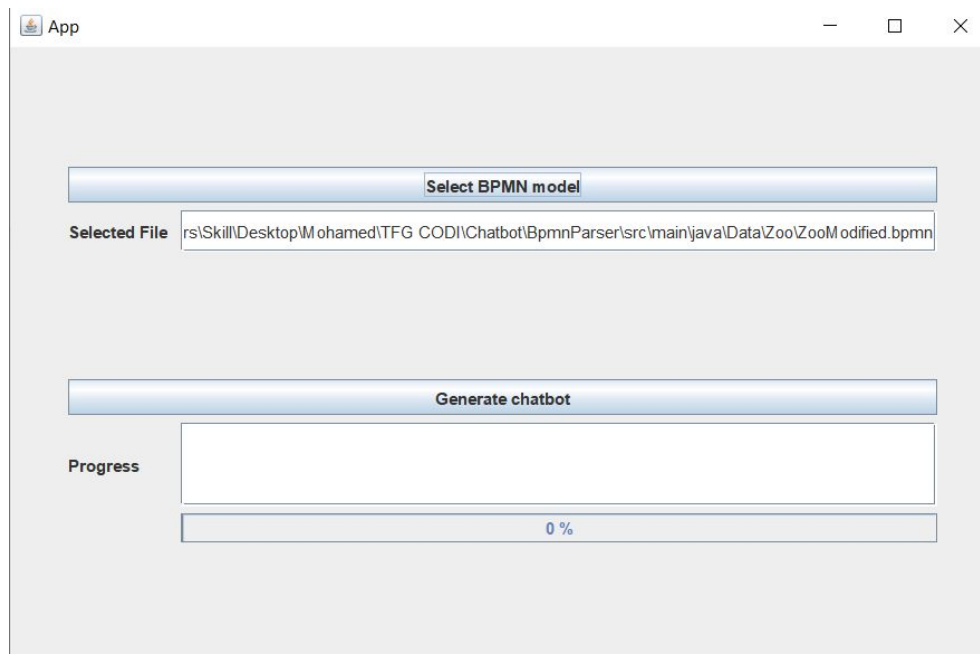


Figura 25: Fitxer bpmn seleccionat, preparat per generar el chatbot

Finalment l'usuari prem “**Generate chatbot**” i el programa llegeix el model BPMN. Aquest està guardat en un format estàndard. Per tant, s’ha utilitzat [Camunda](#) per realitzar aquest procés.

12.2 Generació d'intencions

Un cop llegit el model BPMN, el programa genera la informació necessària per fer funcionar el chatbot. Aquesta informació, tal com s'explica en l'apartat [DialogFlow](#), són les intencions. Aquestes intencions hauran de tenir contextos (entrada i sortida), frases d'entrenament i frases de resposta. Cal notar que les frases, tant d'entrenament com de resposta, es generen més tard, i per tant, en aquest pas l'algoritme es limita a extreure'n la informació necessària per generar-les. La informació que s'extreu per a cada intenció és la següent:

- **Identificador:** Identificar de la intenció. En general és el mateix que la component d'on prové.
- **Subjecte:** Qui realitza l'acció
- **Acció:** El que es realitza.
- **Frases d'entrenament:** Frases que denoten la intenció.
- **Contextos d'entrada:** Contextos que es requereixen per tal d'activar la intenció.
- **Contextos de sortida:** Contextos que seran activats quan s'activi la intenció.
- **Intencions posteriors:** Intencions que només poden succeir si s'activa la intenció actual.

Cal destacar que els contextos són la manera de gestionar el flux de la conversació. És a dir, una intenció només pot ser activada si tots els contextos d'entrada estan actius. Per tant, es pot activar i desactivar contextos per tal de controlar quines intencions es poden activar en un determinat moment de la conversa.

L'estratègia per aconseguir la informació es pot veure en la Fig. 26. Primer es tracta els participants, després es generen intencions per defecte i finalment es generen consultes generals que es poden preguntar en qualsevol moment.

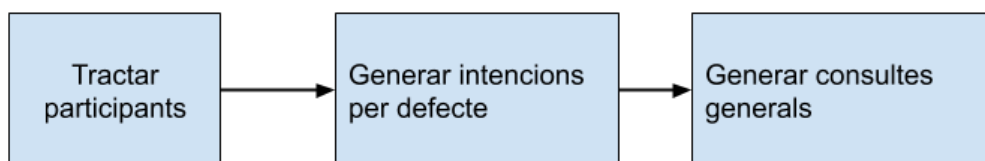


Figura 26: Arquitectura de la generació de intencions

En els següents subapartats s'explica cada fase de l'estratègia.

12.2.1 Tractar participants

Primer de tot, cada participant es tracta per separat. El tractament d'un participant consisteix en generar les intencions pel procés que el participant realitza. Concretament, es comença per l'[esdeveniment inicial](#) i es va recorrent tot el procés, component a component. Cada tipus de component es tracta de forma diferent, perquè components diferents tenen significats diferents. No és el mateix una tasca que una passarel·la exclusiva.

Tot i això, hi ha certa informació que s'extreu de forma **comuna a totes les components**.

12.2.1.1 Extracció d'informació per defecte

La informació que s'extreu de forma **comuna a totes les components** és l'identificador, el subjecte, l'acció i les frases d'entrenament. Aquesta informació s'extreu de la següent manera:

- **Identificador:** S'utilitza l'identificador del component.
- **Subjecte:** És guarda el nom del participant que realitza l'acció.
- **Acció:** És guarda l'acció, si en té. En el cas de les passarel·les exclusives és guarda la pregunta. En el cas d'esdeveniments, siguin d'inicials, intermedis o finals, és guarda en la descripció de l'esdeveniment associat.
- **Frases d'entrenament:** És guarda totes les frases contingudes en els [flux de seqüència](#) que incideixen en la component. Si un flux de seqüència no té text associat, s'assigna la frase "Next", com a frase d'entrenament.

En la Fig. 27 es pot veure un exemple senzill. Per a totes les components l'identificador seria l'empleat. Per l'**esdeveniment inicial**, l'acció seria "Rebre petició" i no tindria frases d'entrenament perquè no té cap flux de seqüència incident. Per la **tasca "Prepara un paquet"**, tindria l'acció "Prepara un paquet" i tindria una frase d'entrenament que seria "No". De la mateixa manera, la **tasca "Entrega petició"**, tindria "Entrega petició" com a acció. Pel que fa a la seva frase d'entrenament, i donat que hi incideix un flux de seqüència buit (sense text associat), la frase d'entrenament serà "Next". Anàlogament, l'**esdeveniment final** tindrà d'acció "End" i com a frase d'entrenament tindrà "Next". Pel que fa a, la **passarel·la exclusiva** tindrà d'acció "Preu per sobre de 100?" i frase d'entrenament "Next". Finalment, cal destacar que la passarel·la exclusiva sense text (la de la dreta), no es converteix en cap intenció.

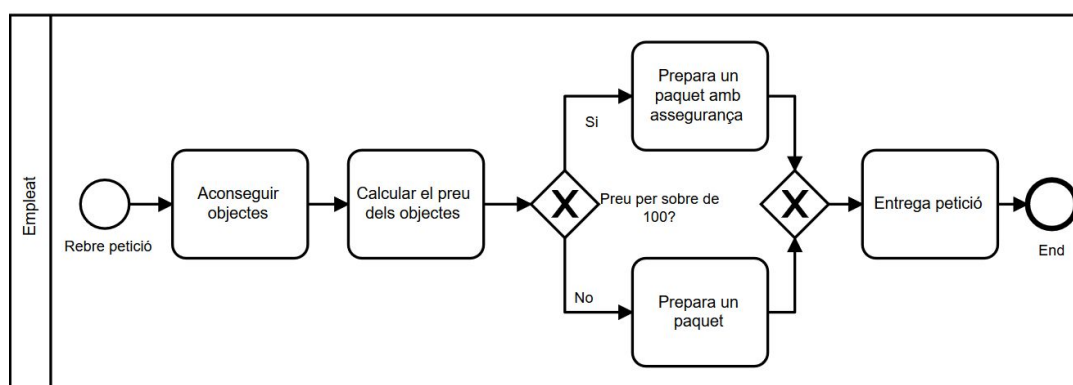


Figura 27: BPMN senzill utilitzat per exemplificar el procés d'extracció de la informació comuna

En la Fig. 28 es pot veure el gràficament el resultat final, només considerant la informació que s'extreu de forma comuna.

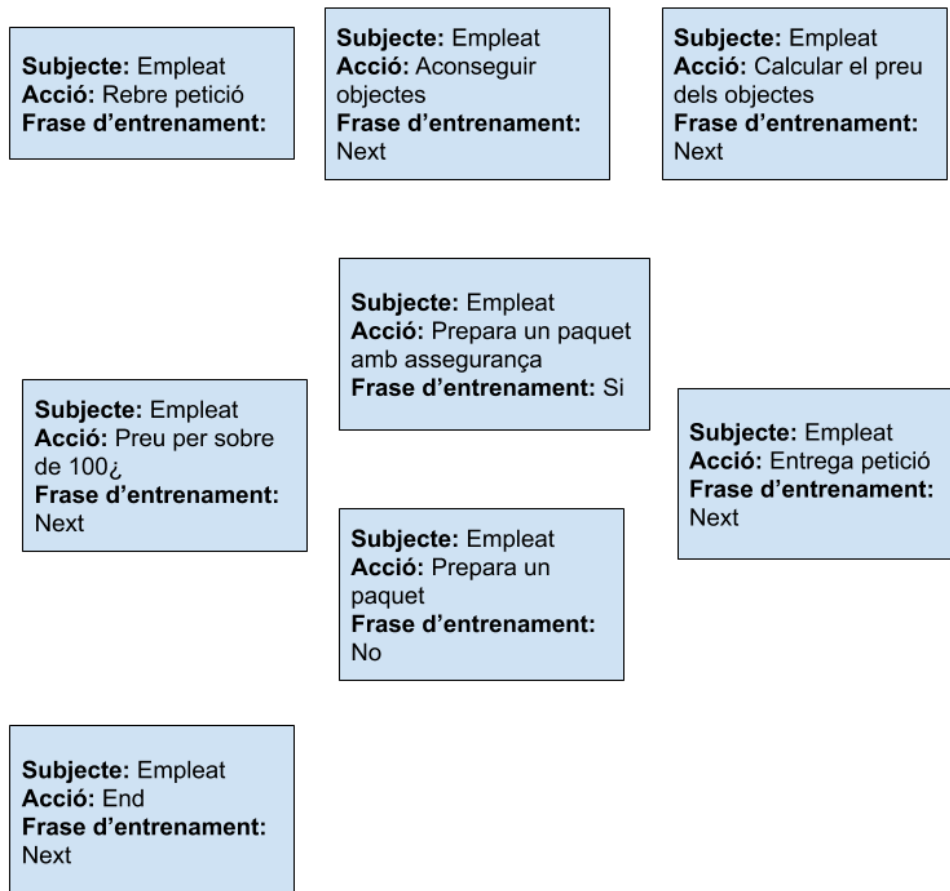


Figura 28: Intencions generades amb la informació comuna

A més de la informació que s'extreu de forma comuna, n'hi d'altre que pot varia depenent de la component. Aquesta informació és: **Contextos, d'entrada i de sortida, i intencions posteriors.**

En general, el context d'entrada és el context de sortida del component anterior, el context de sortida representa al mateix component i les intencions posteriors, seran les següents components. Tot i això, en el cas que la següent component sigui una [passarel·la exclusiva de tancament](#), el context és el de la passarel·la exclusiva, ja que és un punt, de control del flux, on s'ajunten dues o més branques.

En els següents apartats s'explica com es tracta cada component.

12.2.1.2 Esdeveniment inicial

Els esdeveniments inicials aconseguixen el subjecte i l'acció tal com s'explica anteriorment. Tot i això, l'obtenció de les frases d'entrenament es realitza de manera diferent, ja que els esdeveniments inicials van precedits de la intenció BEGIN. Donat que aquesta intenció demana sobre quin participant et vols informar, la frase d'entrenament serà la resposta a aquesta pregunta, és a dir, el subjecte. En l'exemple de la Fig. 29, seria l'empleat.

Pel que fa als contextos, l'esdeveniment inicial té a la intenció "BEGIN" com a context d'entrada. Com a context de sortida, es té a si mateix, com a posteriors intencions té a les següents components.

En la Fig. 29 es pot veure un exemple de generació d'una intenció a partir d'un esdeveniment d'inici.

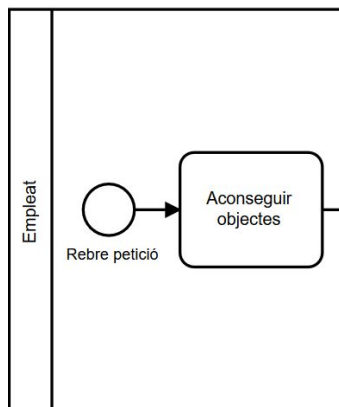


Figura 29: Exemple de la component inicial

El resultat seria el de la Fig. 30. El subjecte seria l'empleat. L'acció seria "Rebre petició", la frase d'entrenament també seria empleat, el context d'entrada seria "BEGIN", i el context de sortida seria l'identificador de l'esdeveniment d'inici.

Subjecte: Empleat
Acció: Rebre petició
Frase d'entrenament: Empleat
Context d'entrada: BEGIN
Context de sortida: Identificador de Rebre petició

Figura 30: Intenció generada a partir de l'esdeveniment inicial

12.2.1.3 Tasca

Les intencions de les tasques s'extreuen de la forma descrita en [Extracció d'informació per defecte](#).

En la Fig. 31 es pot veure un exemple de generació d'intencions a partir de tasques.

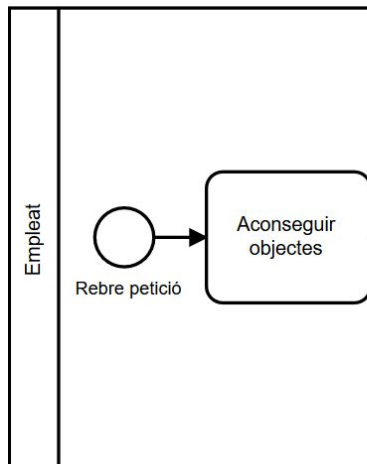


Figura 31: BPMN amb tasques d'exemple

El resultat seria el de la Fig. 32. En el cas de l'esdeveniment inicial, la informació seria la mateixa descrita en Esdeveniment inicial. Pel que fa a la tasca, tindria l'empleat com a subjecte, l'acció seria "Aconseguir objectes", la frase d'entrenament seria "Next", el context d'entrada seria l'identificador de l'esdeveniment inicial. El context de sortida seria l'identificador del context de sortida.

Subjecte: Empleat Acció: Rebre petició Frase d'entrenament: Empleat Context d'entrada: BEGIN Context de sortida: Identificador de Rebre petició	Subjecte: Empleat Acció: Aconseguir objectes Frase d'entrenament: Next Context d'entrada: Identificador de "Rebre petició" Context de sortida: Identificador de Aconseguir objectes
--	--

Figura 32: Intencions generades a partir de l'esdeveniment inicial i la tasca

Cal destacar que en el cas que una tasca envii un missatge a un altre participant, la tasca crearà una intenció. Aquesta intenció consisteix en el fet que si la conversa amb el chatbot es troba en el context de la tasca que conté el missatge, llavors el chatbot preguntarà a l'usuari si vol saber que succeeix amb el missatge. Per exemple, si un chatbot està explicant el model BPMN de la Fig. 33, i està en el context de la tasca "Send payment request to the bank", llavors el chatbot preguntarà a l'usuari si vol saber que fa el Bank. En cas que l'usuari digui que si, el chatbot començarà a explicar l'esdeveniment intermedi "Receive payment confirmation".

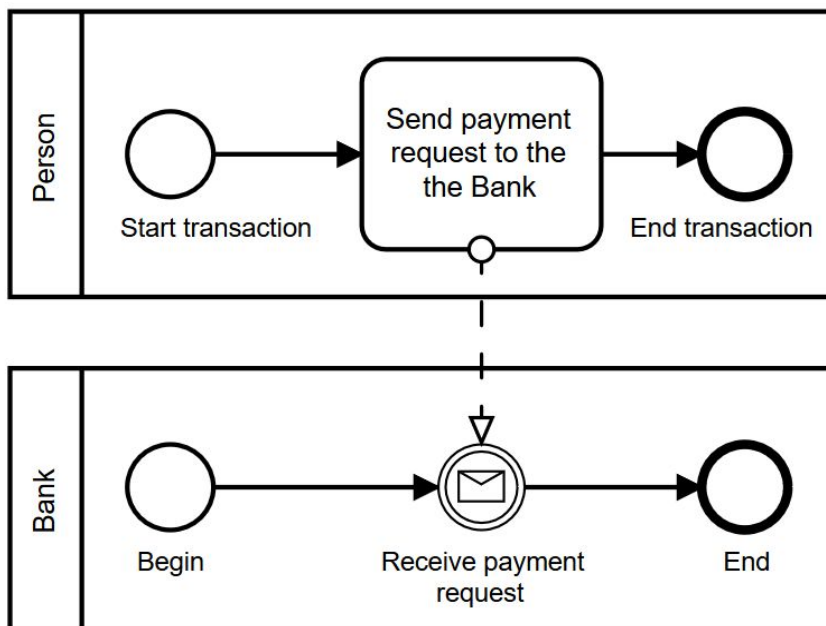


Figura 33: Exemple de una tasca enviant un missatge

12.2.1.4 Passarel·la exclusiva

En el cas de la passarel·la exclusiva, s'ha de diferenciar entre passarel·les que divideixen el flux o passarel·les que ajunten el flux. Aquestes últimes no generen cap mena d'intenció. Les primeres, en canvi, sí que generen una intenció. La intenció és generada amb la mateixa estratègia explicada en [Extracció d'informació per defecte](#).

12.2.1.5 Passarel·la paral·lela

Tal com s'explica en l'apartat [Passarel·la paral·lela](#), aquestes serveixen per descriure processos que es poden dur a la vegada. S'ha considerat dues estratègies per tractar les passarel·les. Les opcions considerades són convertir les branques en una estructura seqüencial i deixar que l'usuari pugui triar que vol saber abans.

Convertir les branques en una estructura seqüencial consisteix en explicar les branques una per una. Si un hi ha N branques dintre el context paral·lel, llavors s'explica primer la branca 1, després la branca 2..., fins a la branca N. Aquesta opció és fàcil d'explicar a usuari (no hi ha confusions) i és més fàcil d'implementar. Com a desavantatges, l'usuari no sap que es pot fer en paral·lel i tampoc pot triar l'ordre.

Deixar que l'usuari pugui triar que vol saber consisteix en donar la primera tasca de cada branca i fer que l'usuari pugui triar l'ordre de les branques. Això permet que l'usuari sap que es pot fer en paral·lel i també pot triar l'ordre. Com a desavantatges, aquesta opció és complicada d'implementar. A més, si el context paral·lel és molt llarg, no és senzill d'explicar a l'usuari. Per tant, l'usuari pot no saber on es troba i quedar confós.

La decisió ha estat realitzar la primera opció, ja que és una opció vàlida, que no confon a l'usuari i s'ajusta a l'abast del projecte. La segona opció seria una opció a considerar en un futur.

Pel que fa al tractament de la component, l'estratègia resumida es pot veure en la Fig. 34: Primer es busca la passarel·la paral·lela de tancament (on s'unifiquen les branques), per això s'utilitza un algorisme BFS⁵⁹. A continuació, és tracta la primera branca. Llavors, s'uneix final de la primera branca, amb l'inici de la segona (el primer component de la segona branca requereix el context de l'últim component de la primera). Llavors es tracta la segona branca. El procés es repeteix fins a arribar a l'última branca.

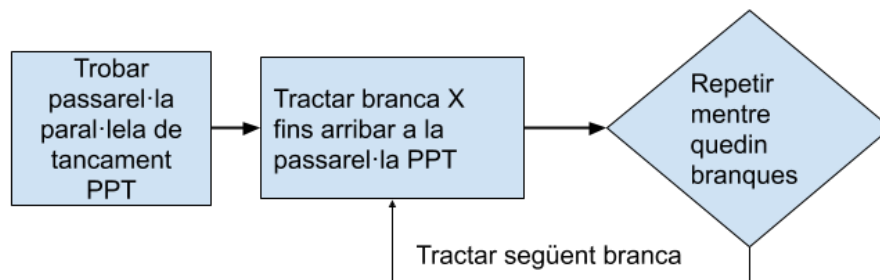


Figura 34: Estratègia seguida al tracta una passarel·la paral·lela

Per d'exemplificar el procés de generació d'intencions d'una passarel·la paral·lela, s'utilitza el model BPMN de la Fig. 35. Aquest procés tindria com a participant un Empleat.

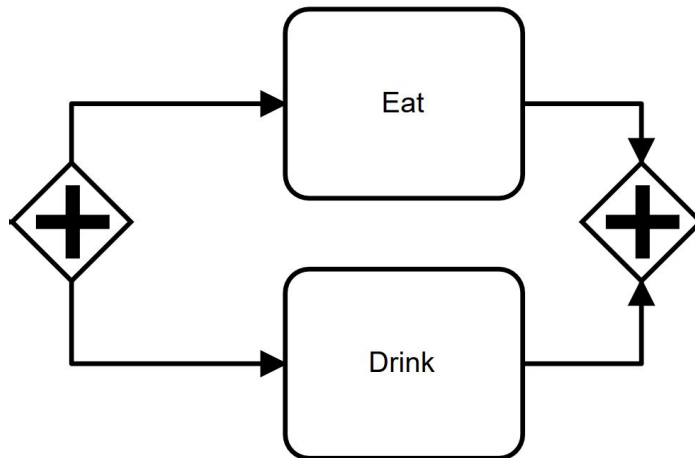


Figura 35: Exemple de passarel·la paral·lela

Primer de tot és buscaria la passarel·la paral·lela de tancament PPT. A continuació, es tractaria la primera branca, en aquest cas la tasca "Eat", fins a arribar a la PPT. Seguidament és connectaria la primera branca amb la segona. És a dir, la tasca "Eat" seria prèvia a la "Drink". Finalment, es tractaria la tasca "Drink".

El resultat seria el de la Fig. 36:

⁵⁹ "Breadth-first search - Wikipedia." https://en.wikipedia.org/wiki/Breadth-first_search. S'hi ha accedit el dia 23 de juny. 2020.

Subjecte: Employee Acció: Eat Frase d'entrenament: Next Context d'entrada: Identificador de la tasca previa a la passarel·la paral·lela Context de sortida: Identificador de "Eat"	Subjecte: Employee Acció: Drink Frase d'entrenament: Next Context d'entrada: Identificador de "Eat" Context de sortida: Identificador de "Drink"
--	---

Figura 36: Intencions generades a partir del context paral·lel

12.2.1.6 Esdeveniment intermedi

En el cas dels esdeveniments intermedis, la intenció és generada amb l'estratègia explicada en [Extracció d'informació per defecte](#). Tot i això, aquests es caracteritzen per tenir un [flux de missatge](#) incident. Per tant, els esdeveniments intermedis realitzen un procés que els altres components no fan. Dit d'una altra manera, els esdeveniments intermedis han de crear una intenció extra perquè l'usuari pugui gestionar el missatge.

En la Fig. 33 es pot veure que esdeveniment intermedi "Receive payment request" té dues fletxes incidents. Per tant, haurà de generar dues intencions. Una per quan un usuari estigui consultant el procés del "Bank", i passi de "Begin" a "Receive payment request" amb la intenció "Next". L'altra intenció serà quan l'usuari estigui consultat el procés de la persona, estigui en el context de "Send payment request to the Bank" i respongui si a la pregunta de si vol saber que succeeix amb el missatge. Llavors aquesta intenció serà la que anirà de "Send payment request to the Bank" a "Receive payment request".

12.2.1.7 Esdeveniment final

En el cas dels esdeveniments finals la intenció és generada amb l'estratègia explicada en [Extracció d'informació per defecte](#).

12.2.2 Generar intencions per defecte

Un cop tractats tots els processos, es genera unes intencions per defecte. Aquestes són les intencions **"BEGIN"**, **"FALLBACK"** i **"BACK"**. El motiu de la seva generació és perquè són intencions necessàries que no és deriven del model BPMN.

- **"BEGIN"** correspon a la intenció de començar a executar l'assistent virtual. Tot esdeveniment inicial va precedit per aquesta intenció. El motiu és perquè aquesta intenció pregunta sobre quin participant vols informar-te. Per tant, els esdeveniments inicials requereixen a aquesta intenció.
- **"FALLBACK"** no correspon a una intenció en si, sinó que es refereix a durant l'assistent no entén a l'usuari. És a dir, correspon a aquella intenció que s'activa quan cap altra intenció s'ha activat. Quan aquesta intenció s'activa, l'assistent virtual respon amb "Sorry, i didn't understand".
- **"BACK"** correspon a la intenció de tornar enrere en la conversació. És a dir, correspon a tornar a l'última intenció de l'usuari.

12.2.3 Generar consultes generals

A continuació, es realitza un postprocessament, on es genera les intencions de les preguntes generals del procés. Cal destacar que aquestes intencions només tenen la informació que posteriorment s'utilitza per generar les consultes generals.

El procediment per tota intenció, exceptuant aquelles originades per una passarel·la exclusiva, és generar dues intencions. Aquestes intencions són dues consultes generals que es poden realitzar en qualsevol moment, i per tant, no tenen contextos d'entrada ni de sortida. Les intencions creades són **preguntar qui realitza una acció**, i **preguntar quin objecte rep l'acció per part del subjecte**.

. El procés de generació de les preguntes generals, i de les frases, està explicat amb detall en l'apartat [Generació de frases](#).

12.3 Generació de frases

Un cop el model BPMN ha estat tractat, i totes les intencions han estat creades, toca construir les frases de resposta i generar frases d'entrenament (frases amb significat similar, les quals seran usades per millorar la flexibilitat del llenguatge).

En la Fig. 37 es pot veure les tres frases pel que respecte a la generació de frases.

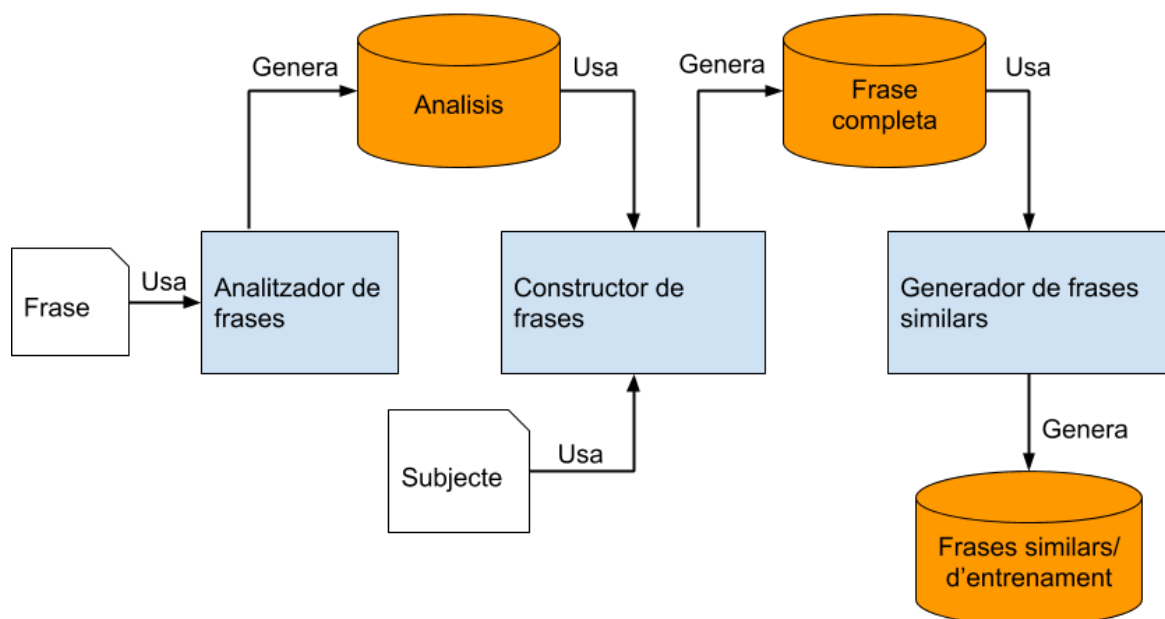


Figura 37: Arquitectura de la generació de frases

Primer de tot, es realitza una anàlisi de les accions d'una intenció. A continuació s'utilitza l'anàlisi i el subjecte de la intenció per construir les frases completes. Aquestes frases poden

ser frases de resposta, o frases d'entrenament. Finalment, a partir de les frases d'entrenament, i per augmentar la flexibilitat en el llenguatge, es generen frases similars. En els següents subapartats s'explica en detall cada una de les fases.

12.3.1 Analitzador de frases

Tal com s'explica en l'apartat [Analitzador de frases](#), l'eina per analitzar les frases és FreeLing. Per utilitzar aquesta eina s'ha fet ús d'un docker que conté l'analitzador.

El procediment d'analitzar les frases s'ha realitzat de la següent manera: Per cada intenció, s'ha obtingut l'acció que es realitza i s'ha enviat en una crida HTTP al docker de FreeLing. FreeLing retorna l'anàlisi de la frase. Aquesta anàlisi té tres components: **Predicat** (Verb), **objecte** i **complements**.

Per exemple, en la Fig. 38 es pot veure una tasca amb l'acció “**Send documents to marketing department**” associada.



Figura 38: Exemple on és mostra el text que FreeLing analitzara

L'anàlisi de l'acció, per part de FreeLing retorna:

Predicat: “Send”

Objecte: “finished documents”

Complement: “to marketing department”.

L'anàlisi és guarda per utilitzar-lo a l'hora de construir les frases completes.

12.3.2 Construcció de frases

Tal com s'explica en l'apartat [Construcció de frases](#), l'eina per construir les frases és SimpleNLG.

En la Fig. 39 es pot veure l'arquitectura de la construcció de frases. SimpleNLG utilitza l'anàlisi generat per FreeLing, i el subjecte (qui realitza l'acció) per generar una frase completa que tingui sentit.

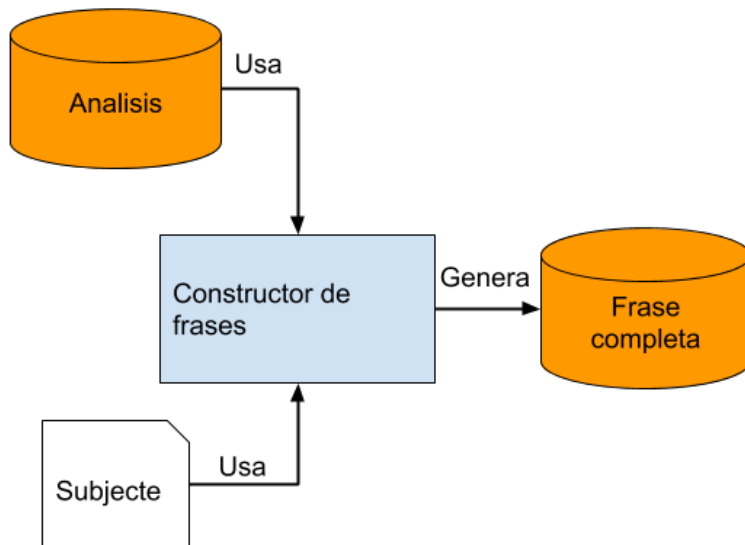


Figura 39: Arquitectura de la construcció de frases

Per exemplificar el procés s'utilitza el model BPMN de la Fig. 40.

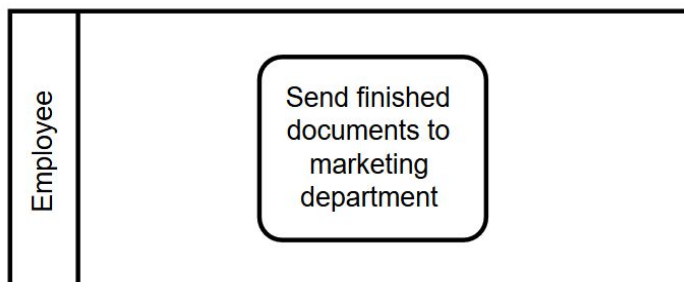


Figura 40: Model d'exemple de construcció d'una frase

FreeLing generaria l'anàlisi de la frase "Send finished documents to marketing department". Aquesta informació, sumada al subjecte (employee) seria:

Subjecte: "Employee",
Predicat: "Send",
Objecte: "finished documents",
Complement: "to marketing department".

A continuació, SimpleNLG utilitzaria aquesta informació per generar les frases pertinents. Aquestes frases són les **frases de resposta**, **qui realitza una acció**, i **quin objecte rep l'acció per part del subjecte**.

Les frases de resposta és generen en format en present simple i en tercera persona. En l'exemple la frase generada seria "**Employee sends finished documents to marketing department**"

Respecte a qui realitza una acció, la frase generada seria "**Who sends the finished documents to marketing department?**"

Pel que fa a quin objecte rep l'acció per part del subjecte, la frase generada seria **“What sends the employee to marketing department?”**

12.3.3 Generació de frases similars/d'entrenament

Tal com s'explica en l'apartat Generació de frases similars, l'eina per (donada una frase), generar frases similars és SpinnerChief. Aquesta eina té una gran limitació i és que està limitada a 20 crides de l'API, cada dia. Per tant, per tal de poder generar les frases similars per a totes les intencions, s'ha dissenyat una estratègia de sincronització per tal d'enviar totes les frases en una sola crida. El procés seguit es pot veure en la Fig. 41.

Primer de tot, s'ajunten totes les frases generades per SimpleNLG i s'envien, en una sola crida, a l'API de SpinnerChief. Aquesta, per totes les frases, retorna sinònims possibles per cada paraula. A partir d'aquestes opcions, un algorisme genera les frases similars. Finalment, i a partir de les frases similars, s'actualitza les frases d'entrenament de cada intenció.

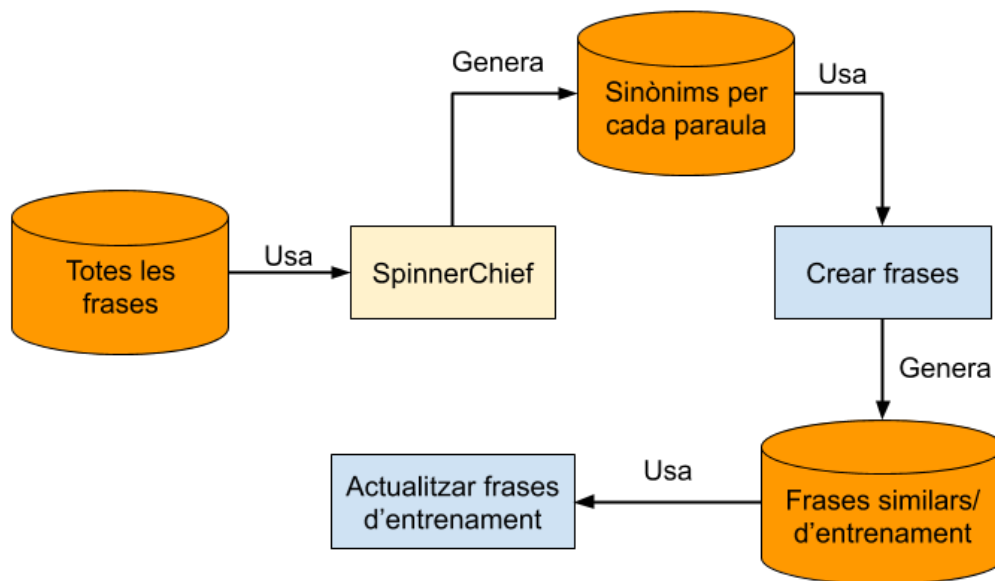


Figura 41: Arquitectura de la generació de frases similars

Donat que SpinnerChief retorna sinònims per cada paraula, s'ha seguit una estratègia per convertir aquests sinònims en frases. Inicialment, es va realitzar un algoritme backtracking⁶⁰ que generava totes les combinacions. Això però, era massa costos. Per tant, és va haver de dissenyar un algorisme que respectes aquesta restricció.

Per tal d'explicar l'algoritme de creació de frases, s'utilitzarà la frase **“Who ships a parcel with insurance?”** com a exemple. Si SpinnerChief processa aquesta frase, retornarà per cada paraula tres sinònims i la paraula original. Per tant, per cada paraula (**Who, ships, a, parcel, with, insurance**) retorna quatre opcions. Aquestes opcions es poden veure en la Taula 20.

⁶⁰ "Backtracking - Wikipedia." <https://en.wikipedia.org/wiki/Backtracking>. S'hi ha accedit el dia 25 de juny. 2020.

Paraula original	Sinònims			
Who	Who	That	Who else	Which
ships	ships	boats	delivers	cruises
a	a	the	a new	some sort of
parcel	parcel	package	courier	goods
with	with	along with	together with	using
insurance	insurance	insurance coverage	insurance policy	insurance plan

Taula 20: Taula on és mostra, per cada paraula original, quatre opcions. Cada fila representa una paraula, i les diferents opcions. En blau la paraula original. En vermell la segona opció, en groc la tercera opció i en verd la quarta opció.

La idea és generar frases de tal manera que la meitat de les paraules utilitzin la paraula original i l'altra meitat utilitzin els sinònims. Després es fa el mateix però al revés. És a dir aquelles paraules que utilitzaven la paraula original, ara utilitzen sinònims, i les paraules que utilitzaven sinònims, ara utilitzen la paraula original. S'ha decidit fer aquesta distinció entre paraules en posicions parelles i senars. És a dir, primer utilitzaran sinònims les posicions parelles, i les senars mantindran la paraula original. Després s'intercanviaran i les posicions parelles utilitzaran la paraula original i les senars utilitzaran els sinònims. Cal destacar que encara que totes les paraules estan canviades en la meitat de les frases, aquesta estratègia assegura que les frases sempre tinguin similitud.

En el següent exemple s'utilitza colors per diferenciar les frases que utilitzen sinònims de les que mantenen la paraula original. En blau la paraula original.

Les frases generades, quan les posicions parelles utilitzen sinònims, serian les següents:

- Who ships a parcel with insurance?
- Who boats a package with insurance coverage?
- Who delivers a courier with insurance policy?
- Who cruises a goods with insurance plan?

Les frases generades, quan les posicions senars utilitzen sinònims, serian les següents:

- That ships the parcel along with insurance?
- Who else ships a new parcel together with insurance?
- Which ships some sort of parcel using insurance?

Aquesta estratègia assegura que el nombre de frases és sempre adequat. També assegura que les frases sempre tenen una part fixa, i l'altre variable, les frases varien però mantenen certa cohesió. Per tant DialogFlow dona millors resultats.

12.4 Entrenament del chatbot

Tal com s'explica en l'apartat [Generació del Chatbot](#), l'eina per entrenar el chatbot és Dialogflow.

Un cop es té totes les intencions amb la seva informació la conversió és prou directa. Els contextos, tant d'entrada com de sortida, les respostes generades i les frases d'entrenament generades, són directament adaptades al format de dialogflow. Per entrenar el chatbot, s'utilitza l'API de dialogflow, i es carrega totes les intencions amb la seva informació. La lògica del chatbot queda emmagatzemada en el servidor de dialogflow. Cal destacar que s'ha realitzat una integració amb Google Assistant, per tal de poder utilitzar el chatbot l'assistent de google.

Un cop s'ha acabat de generar el chatbot, la interfície gràfica indica que el procés s'ha acabat i donarà un enllaç on poder utilitzar el chatbot. En la Fig. 42, es pot veure una foto un cop ha acabat la generació del chatbot.

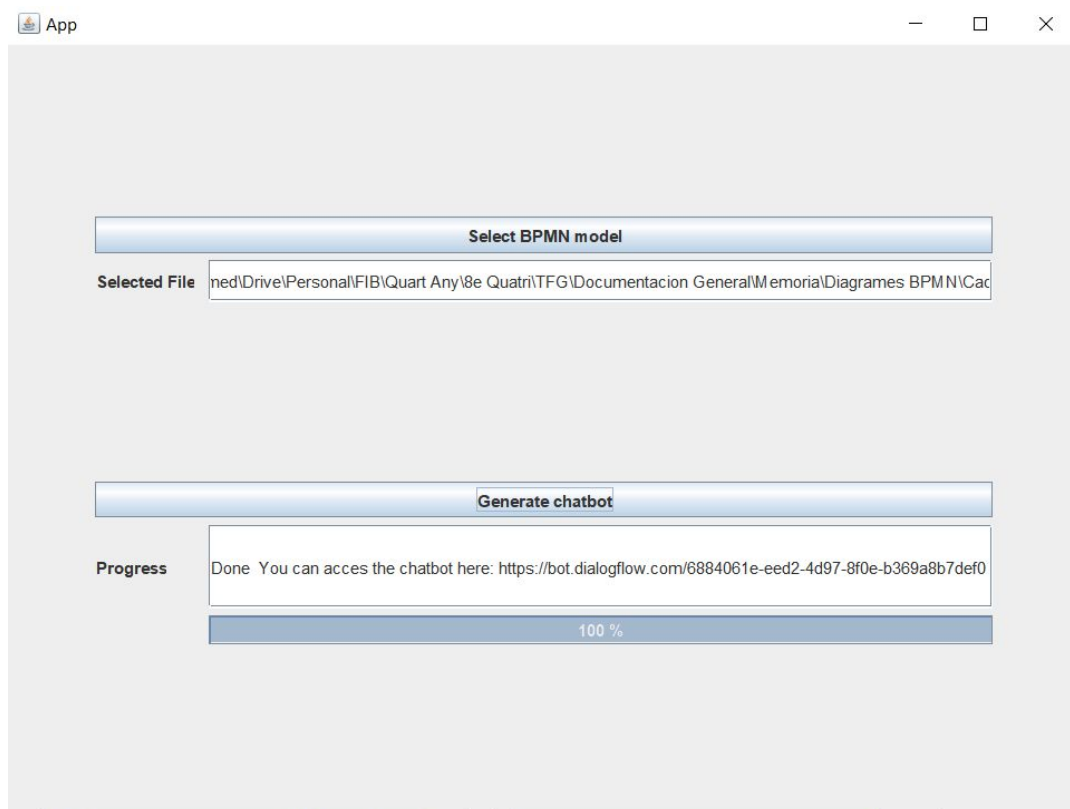


Figura 42: Imatge de la interfície gràfica quan s'acaba l'entrenament del chatbot

13. Execució del chatbot

L'execució del chatbot es refereix al moment en el qual s'utilitza l'assistent virtual. És a dir, el moment en el qual un usuari es comunica amb l'assistent virtual. Tal com s'explica en Fase d'execució del chatbot, l'arquitectura del sistema en el moment d'utilitzar-se és la de la Fig. 18.

Una conversa amb el chatbot funciona de la següent manera. Primer l'usuari envia un missatge al chatbot. Aquest missatge és enviat a dialogflow a través d'internet. És allà on es realitza el procés d'enteniment de l'usuari. Un cop dialogflow entén la intenció de l'usuari, s'envia aquesta informació a un servidor de heroku on fent ús de la base de dades, es decideix l'acció a realitzar. Finalment, es retorna la resposta que acaba arribant als usuaris.

El procés descrit es coneix com a "Fulfillment". Consisteix en el fet que cada cop que l'usuari interacciona amb el chatbot dialogflow contacta amb un servidor. És en el servidor on s'implementa la lògica del chatbot en temps d'execució.

Dintre del que és el projecte, el "Fulfillment" s'ha implementat per implementar la funcionalitat de tornar enrere, és a dir, de tornar a un context anterior de la conversa. Concretament, el paper del servidor és enregistrar l'historial d'intencions de l'usuari.

El funcionament del servidor és el següent: Cada cop que l'usuari parla amb el chatbot, dialogflow envia al servidor quina és la intenció de l'usuari. El servidor va enregistraant totes les intencions, creant així una pila d'intencions. A cada nova intenció, la pila creix. Si en algun moment, l'usuari té la intenció de tornar enrere, llavors la pila decreixerà i es tornarà al context de la conversa anterior.

14. Testing

Generar un chatbot automàticament no és senzill. Per tal de poder desenvolupar-lo de forma ordenada, és necessari definir tests automàtics que provin les funcionalitats que es van desenvolupant. Els tests automàtics són molt importants perquè són una forma de validar que un cop realitzat un canvi en el codi, la resta de funcionalitats continuen funcionant.

El testing s'ha realitzat i automatitzat amb JUNIT⁶¹. S'han realitzat un total de 85 tests. Aquests tests cobreixen principalment tests unitaris⁶² i d'integració⁶³. És a dir, asseguren que totes les funcionalitats funcionen bé per separat. També s'han realitzat alguns tests automàtics de sistema.

Pel que fa a les components provades, aquestes es divideixen en dos grups, la generació de frases i el processament del BPMN. De la part de generació de frases, les components provades són l'analitzador de frases, el constructor de frases i el generador de frases similars. També s'han realitzat tests d'integració que proven totes les components de generació de frases a la vegada. Pel que fa al processament del BPMN, les components provades són l'algorisme de processat i el tractament de les components BPMN.

Per tal de mesura l'efectivitat dels tests, existeix el "coverage". El coverage es refereix a la cobertura del codi. És a dir, indica quina és la proporció de codi que es prova. El coverage té tres indicadors, percentatge de classes, percentatge de mètodes, i percentatge de línies de codi. El percentatge de classes indica quina és la proporció de classes que s'ha provat. El percentatge de mètodes, indica quina és la proporció de mètodes que s'han provat. Finalment, el percentatge de línies de codi, indica la proporció de línies de codi provades. Entre els tres indicadors mencionats, el més rellevant és el percentatge de línies de codi. Aquest dona informació sobre la cobertura real del codi. Els altres dos indicadors, només diuen quantes classes/mètodes s'ha provat. Però no indiquen la proporció provada. Per tant, s'utilitza el percentatge de línies de codi per mesura el "coverage".

La cobertura (coverage) del codi, per cada component es pot veure en la Taula 21:

Components	Nombre de tests	Cobertura
Analitzador de frases	6	92%
Constructor de frases	9	95%

⁶¹ "JUnit.org - JUnit 5." <https://junit.org/junit5/>. S'hi ha accedit el dia 25 de juny. 2020.

⁶² "Unit Testing - Software Testing Fundamentals." <http://softwaretestingfundamentals.com/unit-testing/>. S'hi ha accedit el dia 25 de juny. 2020.

⁶³ "Integration Testing - Software Testing Fundamentals." <http://softwaretestingfundamentals.com/integration-testing/>. S'hi ha accedit el dia 25 de juny. 2020.

Generador de frases similars	12	72%
Integració de la generació de frases	11	83%
Processat del BPMN	15	98%
Tractament de components BPMN	32	91%

Taula 21: Nombre de tests i lineas cobertes pels tests, per cada component.

15. Resultats

En aquest apartat es mostren els resultats obtinguts pel que fa a la generació automàtica i a la flexibilitat en el llenguatge obtinguda. Per a cada model BPMN és mostra una captura de pantalla on es veu la conversa amb el chatbot.

15.1 Zoo.bpmn

El model BPMN utilitzat és el de la figura Fig. 43. El procés descriu el procediment que es realitza quan un visitant s'inscriu al club d'un zoo.

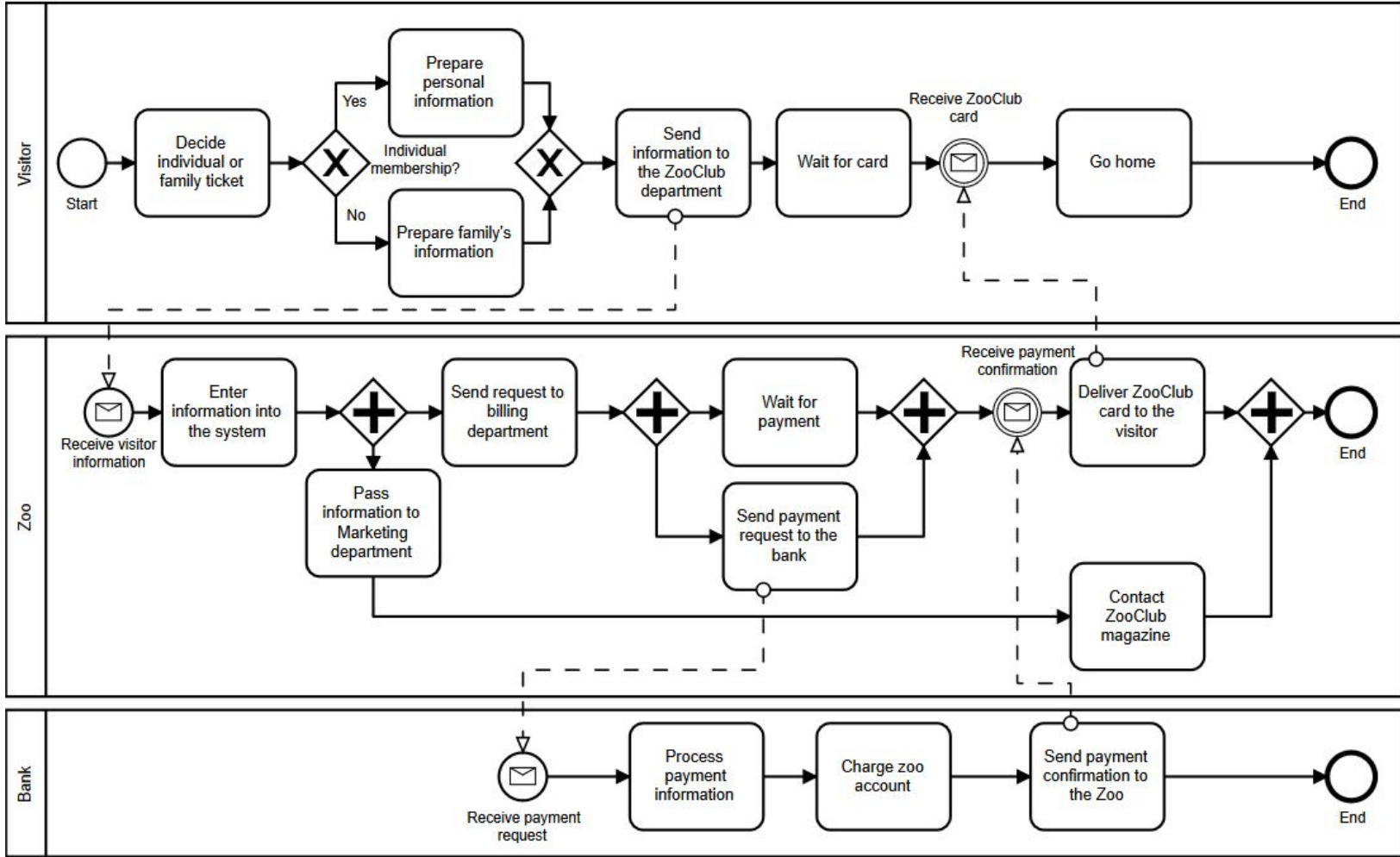


Figura 43: BPMN sobre un Zoo

En Fig. 44 i en Fig. 45 es pot veure unes captures de pantalla, de la conversa amb el chatbot. Al principi entenc a quin rol es refereix a l'usuari. A l'hora d'indicar-li al chatbot que continuï, aquest és capaç d'entendre correctament paraules com **“follow”** i **“next”**. Pel que fa a indicar que si, el chatbot entén **“yes”**, i també és capaç d'entendre **“i think so”** com a frase afirmativa.

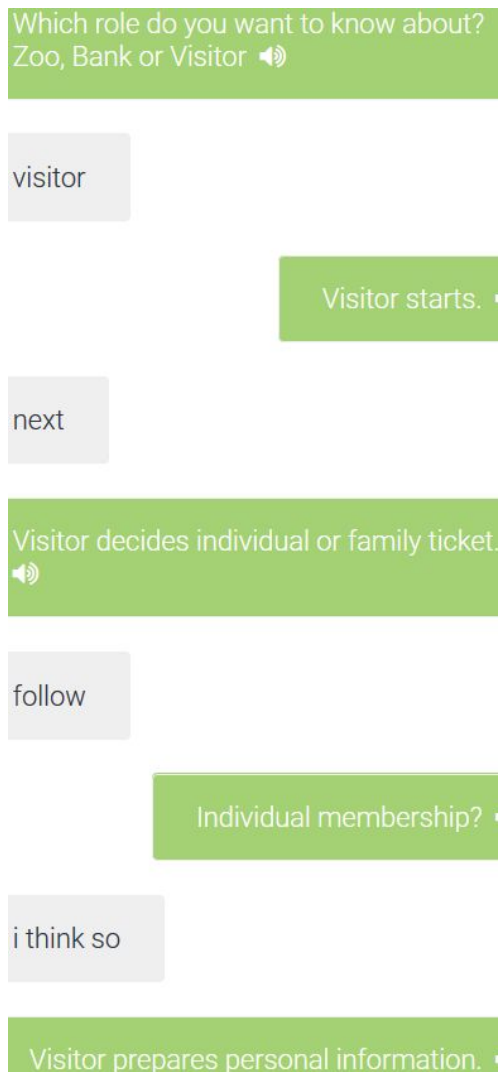


Figura 44: Interacció amb el chatbot

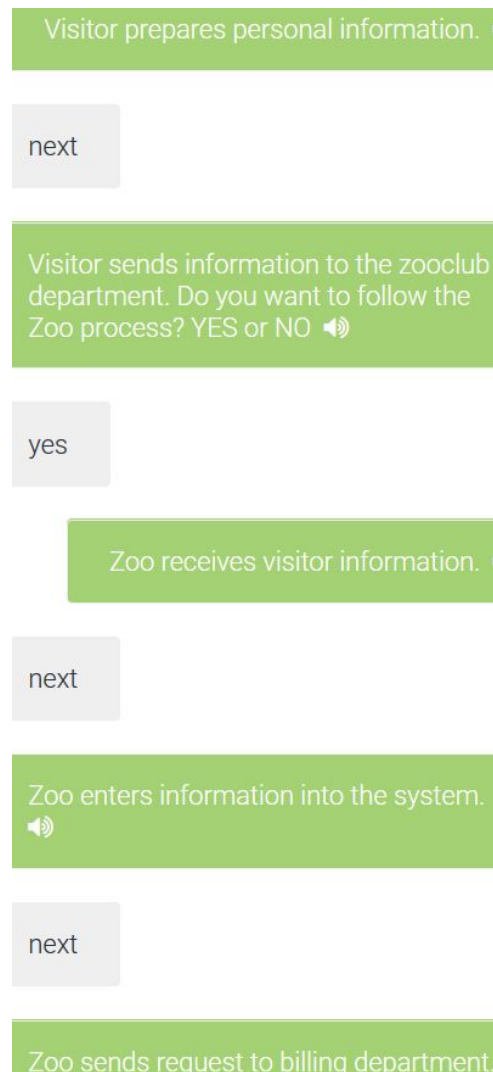


Figura 45: Interacció amb el chatbot 2

Pel que fa a les consultes generals, en Fig. 46 i en Fig. 47 es pot veure que algunes funcionen molt bé, amb flexibilitat en el llenguatge i d'altres, no funcionen tan bé. Per exemple, la pregunta del tipus de tiquet, es pregunta amb, **“Who choses the ticket type”**, **“Who is the person who decides the ticket type?”** i **“Who decides if the ticket is individual or not?”** i **“Who decides if the admission is individual or not?”** i l'entén sempre correctament. També les preguntes de **“Who waits for the card”** o **“Who processes the payment information”** són respostes correctament.

Tot i això hi ha d'altres que no les entén gaire bé. Per exemple, la pregunta **“What sends the visitor to the zoo”** o **“What does the bank receive?”** es responen incorrectament.

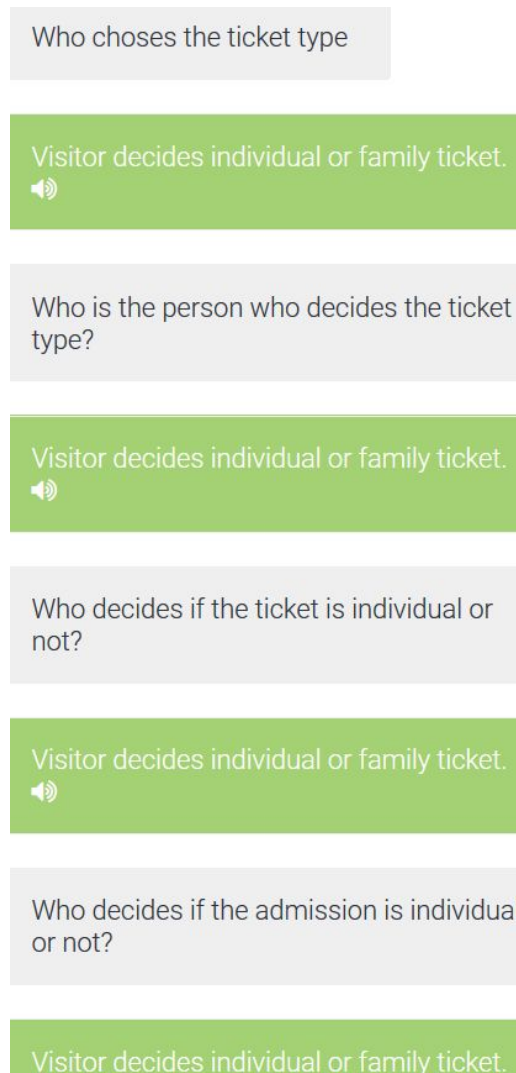


Figura 46: Interacció amb el chatbot 3

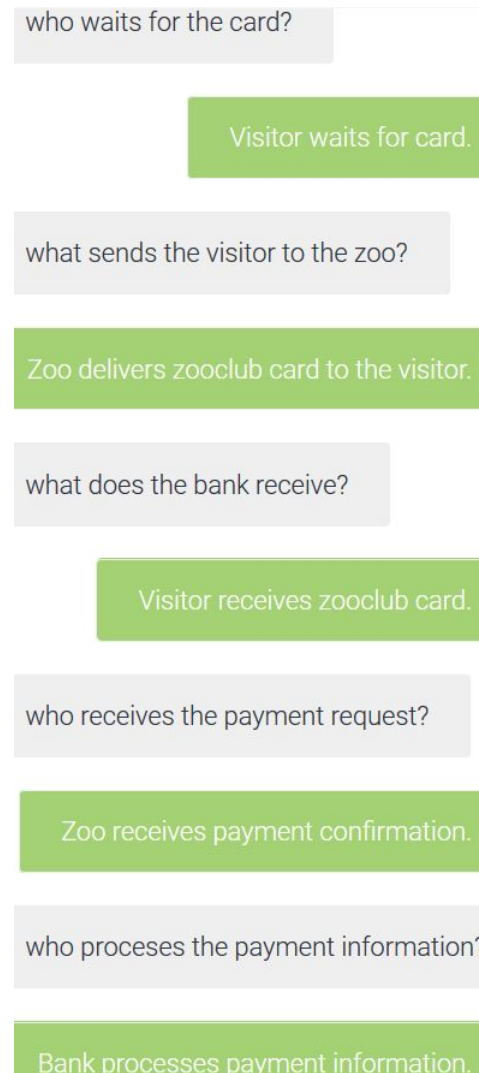


Figura 47: Interacció amb el chatbot 4

El rendiment d'aquest model són molt bons pel que fa a l'explicació pas a pas. Pel que fa a les consultes generals, en alguns casos funciona molt bé, amb flexibilitat en el llenguatge, i en altres casos falla. Això és perquè en el model BPMN hi ha moltes tasques similars. Molts comparteixen les mateixes paraules. Per tant, donat que la diferència entre algunes tasques és molt petita, la generació de frases similars no és prou bona per diferenciar-les.

15.2 Employee

En la Fig. 48 es pot veure un procés que descriu el procediment que es realitza quan un visitant s'inscriu al club d'un zoo.

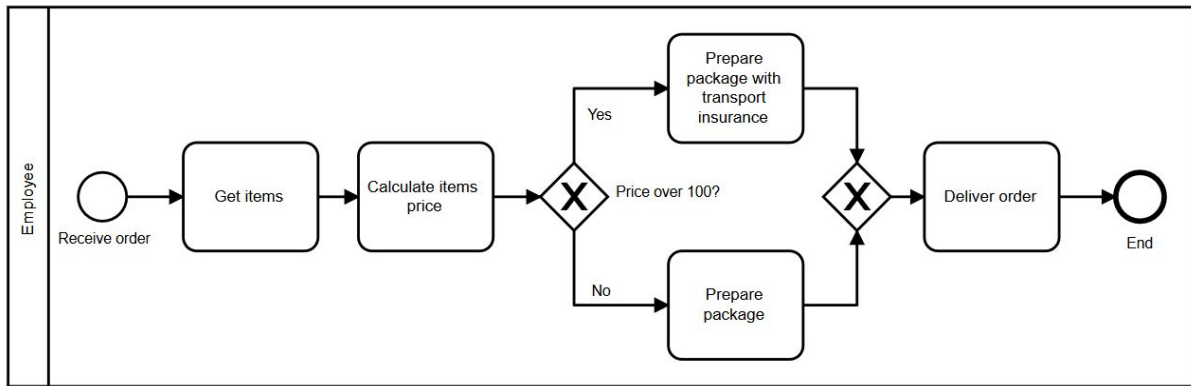


Figura 48: BPMN que és prova. Nom del fitxer: SampleEmployee.bpmn

En Fig. 49 i en Fig. 50 es pot veure unes captures de pantalla, de la conversa amb el chatbot. El chatbot és capaç d'explicar pas a pas el procediment. Pel que fa a indicar que no, el chatbot entén “**don't think so**” com a frase negativa. El chatbot també és capaç de tornar enrere per si l'usuari vol escollir un camí diferent.

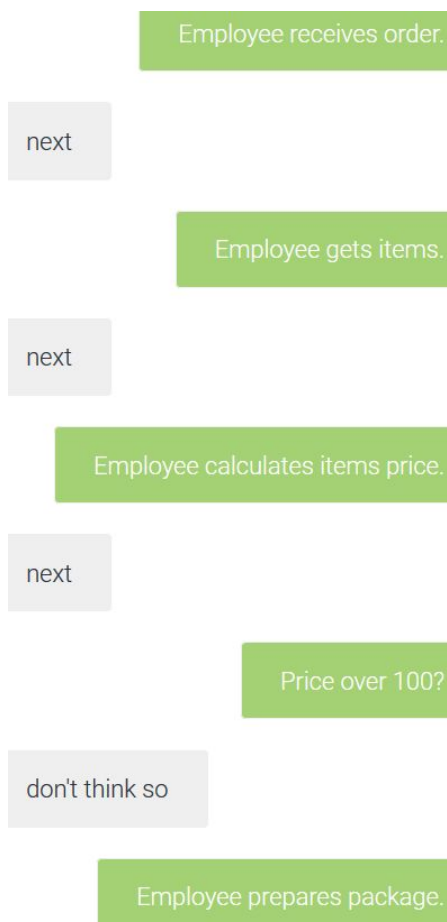


Figura 49: Interacció amb el chatbot 5

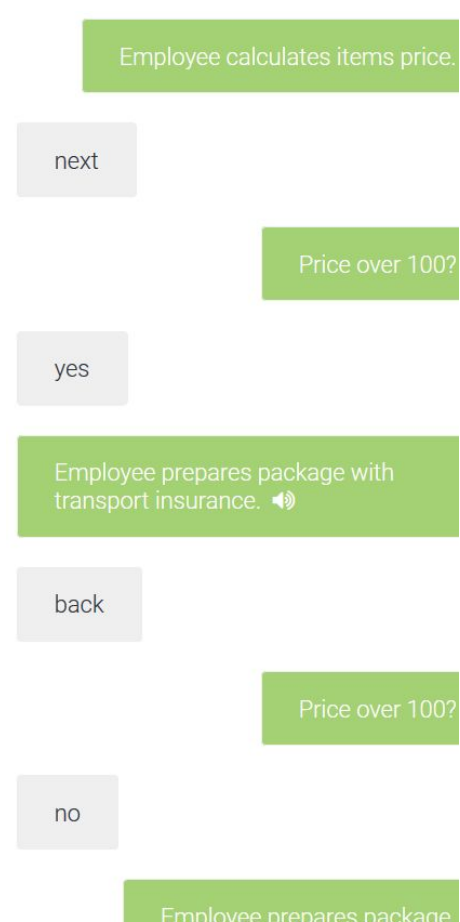


Figura 50: Interacció amb el chatbot 6

Pel que fa a les consultes generals, el chatbot ho fa molt bé, amb flexibilitat en el llenguatge. En la Fig. 51 es pot veure que ha entès correctament quatre frases diferents per referir-se al mateix. En la Fig. 52 n'hi ha un altre exemple.

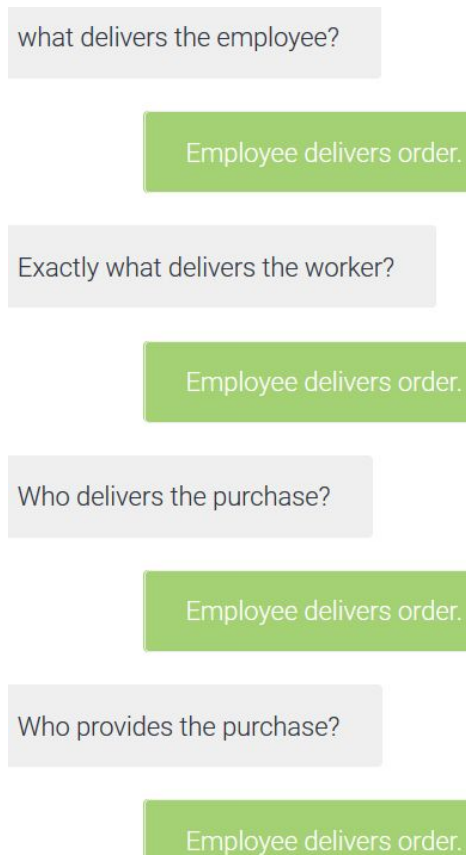


Figura 51: Interacció amb el chatbot 7

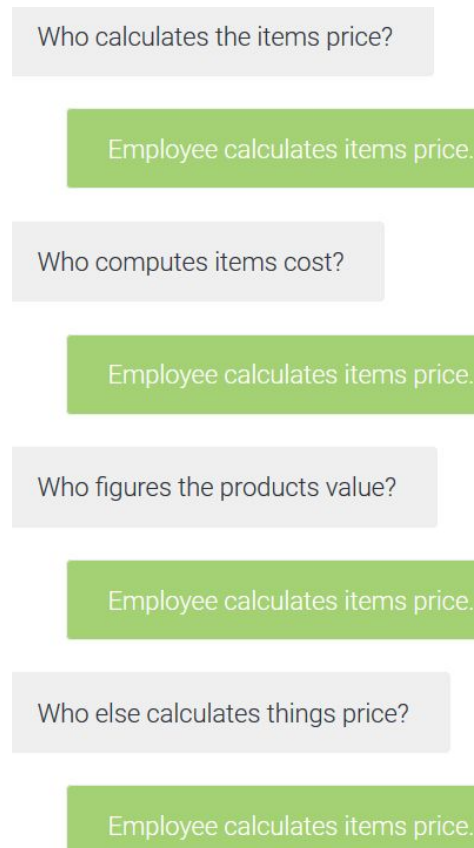


Figura 52: Interacció amb el chatbot 8

La millor en la flexibilitat del llenguatge és deu a què les tasques que es realitzen no són similars. Per tant, és possible aconseguir bona flexibilitat en el llenguatge.

15.3 Cache.bpmn

El BPMN de la Fig. 53 descriu el procés d'obtenir un bloc de caché.

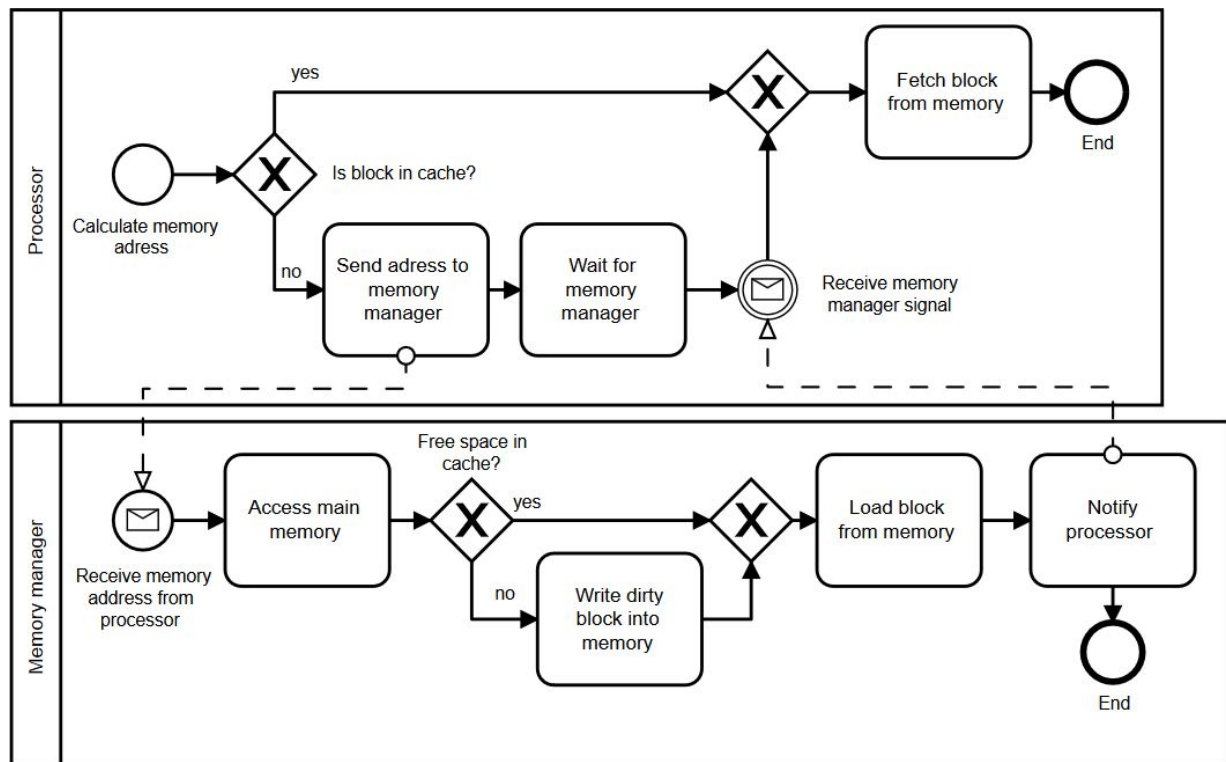


Figura 53: BPMN que es prova. Nom del fitxer cache.bpmn

En Fig. 54 i en Fig. 55 es pot veure unes captures de pantalla, de la conversa amb el chatbot. El chatbot és capaç d'explicar pas a pas el procediment.

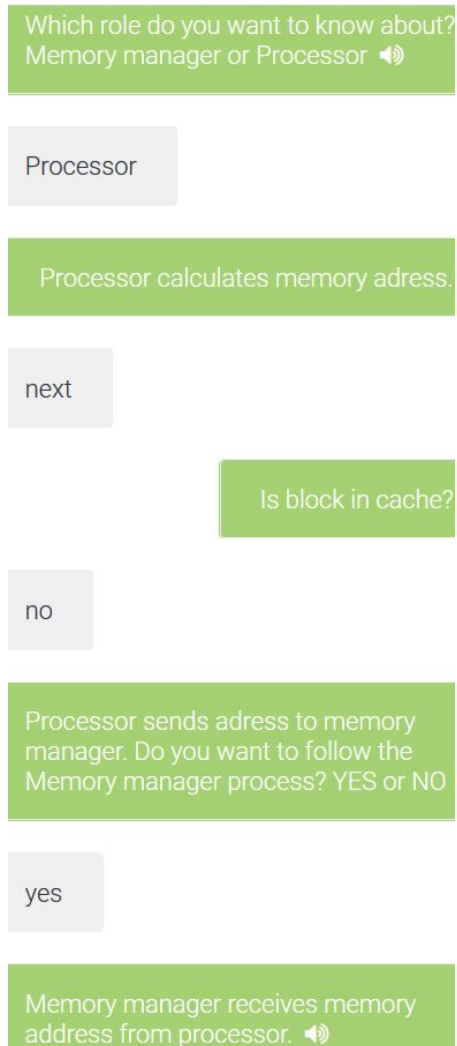


Figura 54: Interacció amb el chatbot 9

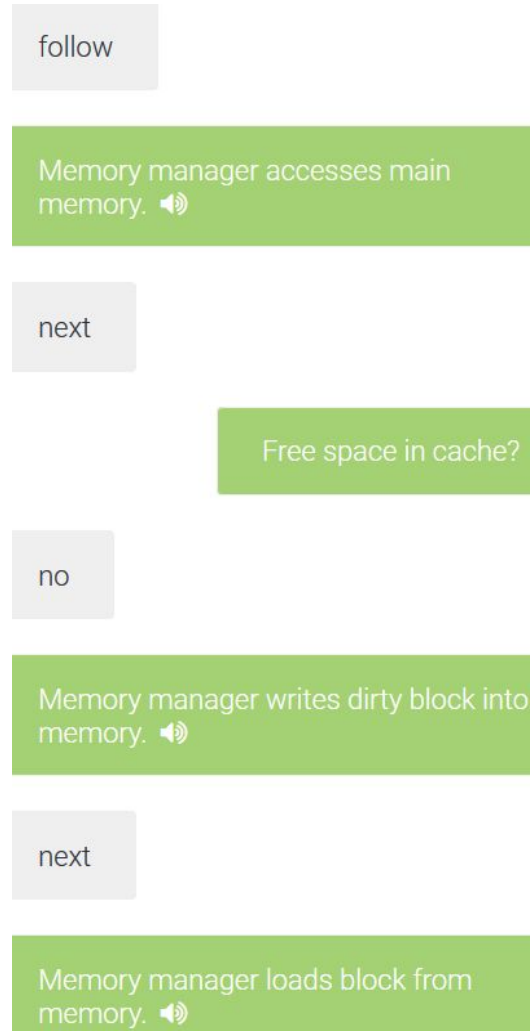


Figura 55: Interacció amb el chatbot 10

Pel que fa a les consultes generals, el chatbot ho fa prou bé, amb flexibilitat en el llenguatge. En la Fig. 56 es pot veure que ha entès correctament tres frases diferents per referir-se al mateix. En la Fig. 57 es pot veure un altre exemple.

who does the memory manager notify?

Memory manager notifies processor.

who does the storage manager inform?

Memory manager notifies processor.

who alerts the processor?

Memory manager notifies processor.

Figura 56: Interacció amb el chatbot 11

what writes the manager into memory?

Memory manager writes dirty block into memory. 🔊

what flushes the storage manager into memory?

Memory manager writes dirty block into memory. 🔊

who writes the dirty block into memory?

Memory manager writes dirty block into memory. 🔊

who writes the filthy block into storage?

Memory manager writes dirty block into memory. 🔊

Figura 57: Interacció amb el chatbot 12

La flexibilitat del llenguatge és deu a què les tasques que es realitzen no són similars. Per tant, és possible aconseguir bona flexibilitat en el llenguatge.

15.4 Conclusions dels resultats obtinguts

A partir dels resultats obtinguts, es pot concloure que el chatbot és capaç d'explicar pas a pas un el procés inclòs en el model BPMN. Primer l'usuari pot triar satisfactòriament el model del qual vol informar-se. Si l'usuari vol progressar en el model, ho pot indicar amb “**Next**”, “**follow**” i variacions d'aquestes paraules. Per contra, si l'usuari vol retrocedir, pot indicar-ho amb “**Back**” i variacions. Pel que fa al control de flux (anar per un camí o un altre), l'usuari ha pogut indicar-ho amb suficient flexibilitat. En el cas de l'afirmació, ho ha fet amb “**yes**”, “**of course**”, “**indeed**”, “**sure**”, “**i think so**” i “**affirmative**”. En el cas de la negació, ho ha pogut fer amb “**no**”, “**negative**” i “**don't think so**”.

Pel que fa a les consultes generals, aquestes funcionen prou bé, però també hi ha prou casos en els quals no s'aconsegueix respondre correctament. Les consultes que millor han funcionat són aquelles que no comparteixen res amb d'altres. També s'ha aconseguit una flexibilitat en el llenguatge prou bona. Aquesta és deu a què les tasques que es realitzen no són similars. Per tant, és possible aconseguir bona flexibilitat en el llenguatge.

Les consultes que pitjor han funcionat són les d'enviar i rebre coses. Aquestes poden ser confuses, ja que diferents consultes comparteixen els mateixos objectes.

16. Conclusions

Es pot concloure que s'ha aconseguit l'objectiu d'aquest projecte, que era avaluar la flexibilitat en el llenguatge fent ús de les eines actuals més punteres, pel que fa a genera chatbots automàticament, a partir de models BPMN.

De forma concreta, en el projecte s'ha fet una anàlisi profund de les alternatives per a cada component. Sempre s'ha prioritzat utilitzar les eines més punteres i gratuïtes que puguin millorar la flexibilitat en el llenguatge. Tot i això, també s'ha tingut en compte un temps d'execució raonable. A més, les tecnologies utilitzades s'han implementat de manera que siguin fàcils de canviar, en el cas que es decideixi que existeix una millor alternativa o és decideixi implementar-la des de zero.

Pel que fa a l'aplicació desenvolupada, aquesta pot seleccionar un model BPMN i genera un chatbot. Aquest chatbot és capaç d'explicar pas a pas el procés de negoci del model seleccionat. A més, també és capaç de respondre les consultes generals amb la suficient flexibilitat en el llenguatge.

Tot i que en general funciona prou bé, cal mencionar que l'aplicació no és perfecte. Els models més complexos causen que les consultes generals no acabin de funcionar del tot bé. Això és degut al fet que hi ha certs conceptes, repetits en moltes tasques, que causen que el chatbot no sàpiga diferencia correctament la intenció de l'usuari. La forma de solucionar-ho és millorant la qualitat de les frases d'entrenament. Això però, sobrepassa l'abast del projecte, per tant, donat el temps del qual s'ha disposat, el resultat ha sigut prou bo pel que fa a la precisió aconseguida a l'hora d'explicar pas a pas un procés i la flexibilitat obtinguda a l'hora de fer les consultes generals.

Per tant, considerant els resultats obtinguts, es conclou que dialogflow és una bona eina pel que fa a generar chatbots de forma automàtica, a partir de models BPMN. Funciona molt bé pel que respecta a explicar pas a pas un model, i funciona decentment amb la flexibilitat en el llenguatge. Tot i això, si es vol aconseguir una flexibilitat en el llenguatge major, a l'hora d'efectuar les consultes generals, serà necessari realitzar més recerca.

17. Treball futur

Donat que la conclusió del projecte és que es requereix més recerca, si es vol aconseguir una millor flexibilitat en llenguatge, hi ha cert treball que podria realitzar-se en aquesta línia. Per tant, es planteja la següent feina com a treball futur:

- **Millorar la generació de frases d'entrenament:** Consistiria a millorar la generació de frases d'entrenament. Concretament es tractaria de dissenyar un mètode generar frases similars. A més, donada una frase, es podria calcular quines són les frases que tenen un significat més similar.
- **Estendre la cobertura dels components BPMN:** En el projecte no s'ha cobert tot l'estàndard BPMN. Per tant, aquesta opció consisteix a cobrir tots els components de l'estendard BPMN.
- **Desenvolupar una eina de modelatge de chatbots:** Aquest projecte genera automàticament un chatbot a partir d'un model BPMN. Dit d'una altra manera, es pot modelar chatbots a partir de model BPMN. Per tant, aquesta opció consistiria en realitzar una eina que permeti genera chatbots utilitzen diagrames.
- **Utilitzar el servidor per entrenar el chatbot:** En el projecte s'ha implementat un servidor amb una base dades. La idea seria emmagatzemar en la base de dades informació sobre l'execució real del chatbot. A partir d'aquesta informació, el chatbot aniria aprenent dels seus errors.
- **Utilitzar el servidor per realitzar estudis empresarials:** Tal com s'ha dit anteriorment, en el projecte s'ha implementat un servidor. La idea seria emmagatzemar en la base de dades informació sobre l'execució real del chatbot. A partir d'aquesta informació, es tractaria de realitzar estudis per detectar quins són els dubtes que tenen i que consulten els empleats d'una empresa. Això podria permetre detectar els punts dèbils d'una empresa.
- **Utilitzar el chatbot com a eina d'ensenyament:** En l'apartat [Cache](#), s'utilitza un model BPMN el qual descriu, de forma simplificada, com funciona

la memòria caché. La idea seria que els estudiants poguessin comunicar-se amb el chatbot de manera que podrien preguntar i resoldre dubtes sobre una unitat d'un tema d'una assignatura. Donat que avui en dia no existeix res similar, es tractaria d'estudiar-ne la viabilitat.

18. Adaptació de les competències tècniques

18.1 CES1.2:

Donar solució a problemes d'integració en funció de les estratègies, dels estàndards i de les tecnologies disponibles. [En profunditat]

Tal com es pot veure en [Anàlisis d'opcions](#), per cada component de l'arquitectura general, s'ha considerat i s'ha analitzat moltes tecnologies. Cada una d'elles s'ha integrat al projecte d'una forma diferent. Freeling s'ha integrat mitjançant un docker, SimpleNLG a partir de les llibreries JAR i SpinnerChief i Dialogflow s'han integrat mitjançant crides a una API. Cal destacar també, que donada la limitació de l'API de SpinnerChief, s'ha dissenyat una manera de sincronitzar a totes les instàncies per realitzar una sola crida. Aquesta es pot veure en [Generació de frases similars](#).

A més, i tal com es pot veure en [UML](#), s'ha utilitzat un patró adaptador per cada una de les eines mencionades. D'aquesta manera és fàcil integrar altres eines, en el cas que sigui necessari.

El grau de satisfacció és “**En profunditat**” perquè l'objectiu és fer ús de les tecnologies més punteres, i per tant, s'ha de dissenyar un sistema per poder integrar les eines que es requereixen, de manera senzilla, sigui com sigui la seva integració.

18.2 CES1.3:

Identificar, avaluar i gestionar els riscos potencials associats a la construcció de software que es poguessin presentar. [Una mica]

Els riscos del projecte es van considerar al principi del projecte tal com es pot veure en [Gestió del risc](#). D'entre els riscos mencionats, els que estan associats a la construcció del software són quedar-se sense temps, que les tecnologies desconegudes fossin difícils d'integrar i que les eines no fossin satisfactòries. Per cada risc, es menciona l'estratègia de mitigació. A més, pel que fa a què les eines no fossin satisfactòries, es va decidir (tal com es pot veure en els [requisits no funcionals](#)) que un requisit no funcional del projecte fos la mantenibilitat. D'aquesta manera seria fàcil canviar una eina que no funcionés bé. Això s'aconsegueix, tal com es pot veure en [UML](#), mitjançant adaptadors.

El grau de satisfacció és “**Una mica**” perquè tot i que per una part s'ha realitzat un codi mantenible, per l'altra part el projecte realitza un prototip i per tant, no hi ha informació sensible que hagi de ser xifrada.

18.3 CES1.7:

Controlar la qualitat i dissenyar proves en la producció de software. [Una mica]

Tal com s'explica en [Testing](#), s'ha dissenyat proves automàtiques per tal de validar el correcte funcionament i la qualitat de cada component de l'algorisme de generació de chatbots.

El grau de satisfacció és “**Una mica**” perquè tot i que s'ha realitzat un coverage de codi alt, la majoria dels tests són tests unitaris i d'integració.

18.4 CES1.9:

Demostrar comprensió en la gestió i govern dels sistemes software. [Bastant]

Aquest projecte integra moltes tecnologies i serveis. Per això, es va decidir fer mantenible el sistema i crear adaptadors per cada eina. A més, s'ha realitzat tests per tal assegurar que cada component funciona correctament. Aquests tests s'han fet automàtics per tal de poder provar de manera ràpida i eficient tot el sistema. Si es fa un canvi, es pot detectar si una altra funcionalitat ha deixat de funcionar. Pel que fa a les dades personals, tot i ser un prototip, també s'ha considerat les polítiques de privacitat.

El grau de satisfacció és “**Bastant**”, ja que el projecte ha requerit entendre com integra tots els serveis, de manera incremental.

18.5 CES2.1:

Definir i gestionar els requisits d'un sistema software. [En profunditat]

En el projecte es va definir quins components eren necessaris i quina era la [cobertura](#) de l'estàndar BPMN de cara a l'elaboració del prototip. A més, també es va especificar les tasques que es durien a terme. Tal com es pot veure en [Planificació final i desviacions](#), en el transcurs del projecte algunes d'aquestes tasques van haver de ser més concretades. A més, també s'ha concretat quines consultes generals pot respondre el chatbot. Finalment, també es va definir els [requisits no funcionals](#) del prototip.

El grau de satisfacció és “**En profunditat**” perquè el projecte requereix tenir un abast ben definit, i cada un dels requisits s'ha treballat amb detall.

18.6 CES2.2:

Dissenyar solucions apropiades en un o més dominis d'aplicació, utilitzant mètodes d'enginyeria del software que integrin aspectes ètics, socials, legals i econòmics.

[Una mica]

El tractament de la informació sensible ha estat considerat a l'hora d'integrar el chatbot amb altres plataformes. Concretament, la integració del chatbot amb Google Assistant requereix crear una política de privacitat, on es defineixi quines dades es tracten i quin ús se'n fa.

El grau de satisfacció és **“Una mica”** perquè tot i que s'ha considerat l'aspecte legal, el projecte realitza un prototip i per tant, no es tracten dades de ningú. Tot i això, si en un futur el projecte progressa, a més de la política de privacitat, la llei de protecció de dades hauria de ser considerada a fons.

18.7 CES3.1:

Desenvolupar serveis i aplicacions multimèdia. [Bastant]

El projecte desenvolupa una aplicació amb una interfície gràfica la qual permet generar chatbots. Aquesta inclou molts serveis, els quals s'han d'executar en qualsevol moment. Per tant, s'han hagut de desenvolupar l'aplicació tenint-ho en compte.

Pel que fa a parlar amb el chatbot, s'ha integrat amb dialogflow, també s'ha integrat amb un servei perquè pugui ser utilitzat per telèfon (només veu) i s'ha integrat per google assistant. L'opció per telèfon però, consumeix diners, per tant, s'ha desactivat. Actualment, el chatbot pot executar-se en la pàgina de dialogflow o en la integració de google assistant.

El grau de satisfacció és **“Bastant”** perquè s'ha desenvolupat l'aplicació de manera que es pot parlar amb el chatbot via text, veu i mixt (google assistant).

Apèndix A

Fitxers adjunts

En aquest apartat es pot trobar una descripció de la estructura de les carpetes. També inclou els requeriments per executar el sistema.

Estructura de la carpeta

Partir des de l'arrel del projecte (carpeta BpmnParser). La jerarquia simplificada del projecte és la següent. Només és mostra les carpetes més rellevants.

- BpmnParser
 - out
 - Artifact
 - credentials.json
 - BpmnParser.jar
 - src
 - main
 - java
 - resources
 - test
- Requeriments
 - FreeLing
 - README-windows
 - SpinnerChief
 - README
 - Dialogflow
 - README

La carpeta “**out**” és on es guarda l'executable (fitxer.jar) Allà també hi ha les credencials de dialogflow necessàries per funcionar. En “**src/main/java**” conte les classes java. La carpeta “**test**” conté les proves, també esta la carpeta “**resources**”.

La carpeta “**requeriments**” conté les instruccions dels requisits que s'expliquen a continuació.

Requeriments

En aquest apartat s'explica que es necessita per fer funcionar l'aplicació.

FreeLing

FreeLing està en un docker, per tant és necessita instal·lar el docker per poder utilitzar freeling. En la carpeta “**Requeriments/FreeLing**” hi ha el fitxer **README-windows** el qual indica les instruccions d'instal·lació.

Aquest pas és imprescindible si és vol executar el JAR en un altre ordinador.

SpinnerChief

Per poder utilitzar SpinnerChief és necessari crear-se un compte per obtenir unes credencials les quals permeten fer funcionar l'API de SpinnerChief. En la carpeta “**Requeriments/SpinnerChief**” hi ha el fitxer README el qual indica les instruccions.

Normalment seria necessari seguir els passos descrits, el projecte però, ja té unes credencials, així que aquest pas no seria necessari. Tot i això s'ha d'anar amb compte ja que la API està limitada a 20 crides al dia.

DialogFlow

Per poder utilitzar dialogflow és necessari crear-se un compte per obtenir unes credencials les quals permeten fer funcionar l'API. En la carpeta “**Requeriments/DialogFlow**” hi ha el fitxer README el qual indica les instruccions.

Normalment seria necessari seguir els passos descrits, el projecte però, ja té unes credencials, així que aquest pas no seria necessari. Tot i això s'ha d'anar amb compte perquè si dues persones utilitzen les mateixes credencials, poden tenir interferències.

JAR

El JAR és el fitxer executable que iniciarà l'aplicació. Si tots els [requeriments](#) es compleixen, llavors funcionarà.

El JAR s'ha de generar a partir del fitxer “**BpmnParser/src/main/java/**”.

Les credencials de dialogflow han d'estar a la mateixa carpeta que el fitxer **jar**.

En l'annex ja hi ha un JAR. Per executar-lo simplement s'ha de fer doble clic. **Important:** Si no se instal·la freeling no funcionarà.