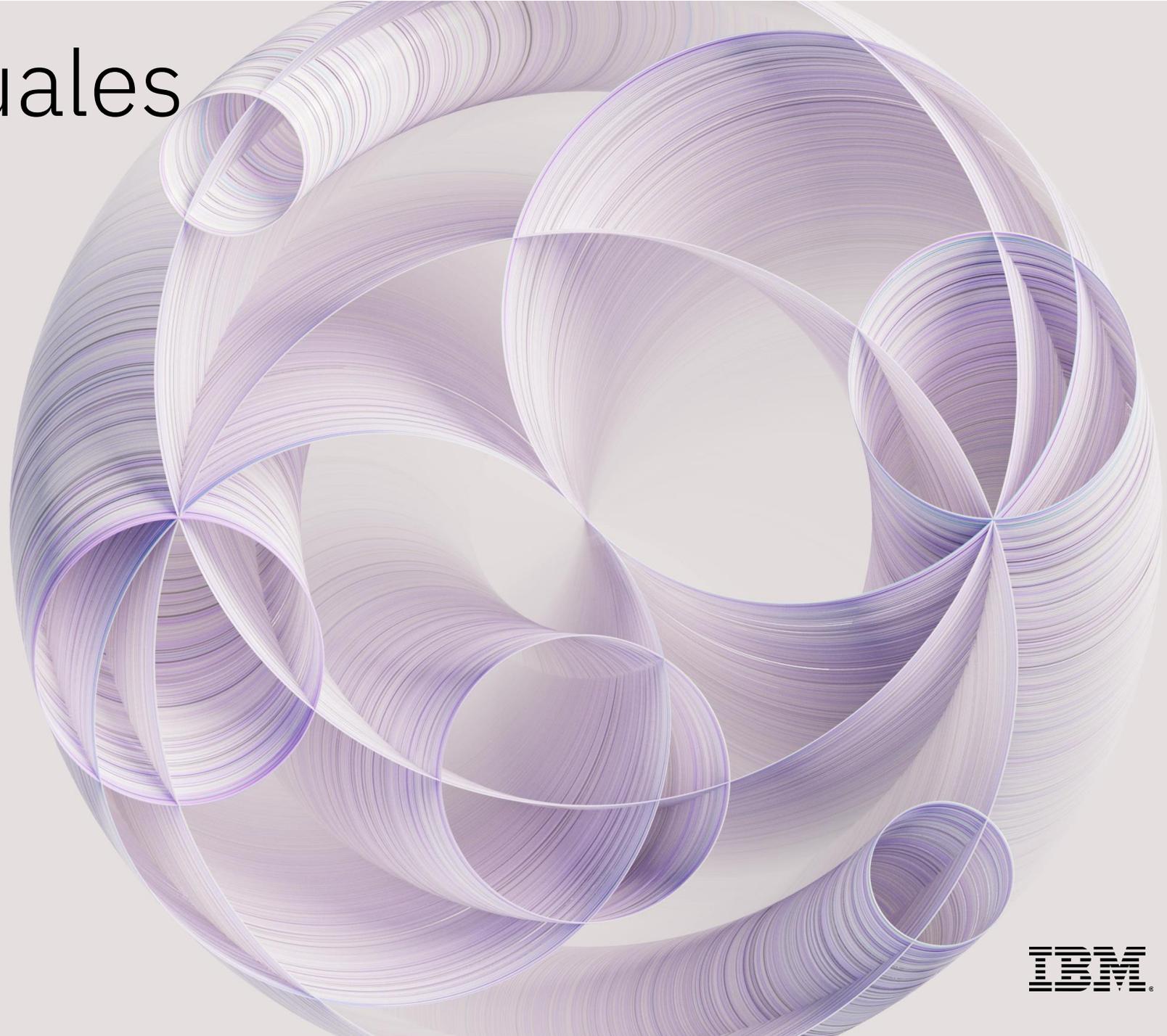


# Asistentes Virtuales **watsonx**

**Victoria Gómez**  
IBM Head of AI





**Año 2011**



Siguiente nivel:  
IA Generativa

**watsonx**





# watsonx Assistant es líder

- Sólida plataforma “no-code” basada en altas capacidades de NLP, voz e IA conversacional
- Integración completa con agentes y CCaaS
- Opciones de implementación flexibles
- Análisis de conversaciones avanzado

# ¿Por qué IBM **watsonx** ?

## Confianza

---

- IBM **no utiliza datos de cliente** para entrenar sus modelos de IA
- **Gobierno y transparencia**
- IBM proporciona **indemnización** por sus modelos de IA

## Abierto

---

- **Flexibilidad** de elección de los modelos de IA
- **Multi-modelo:** distintos modelos de IA en una misma conversación

## Híbrido

---

- **On Prem, SaaS, IaaS**, para una máxima protección de los datos

# Cientes a nivel mundial **confían** en nosotros

**+1 billion**

Mensajes procesados cada mes con watsonx Assistant

**+10.000**

clientes utilizando asistentes de IBM en este momento

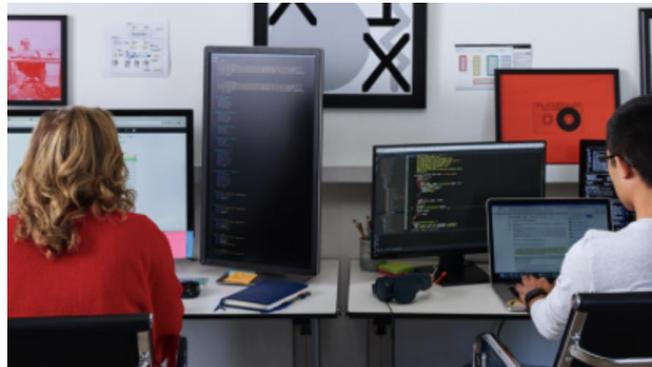


# Casos de Uso

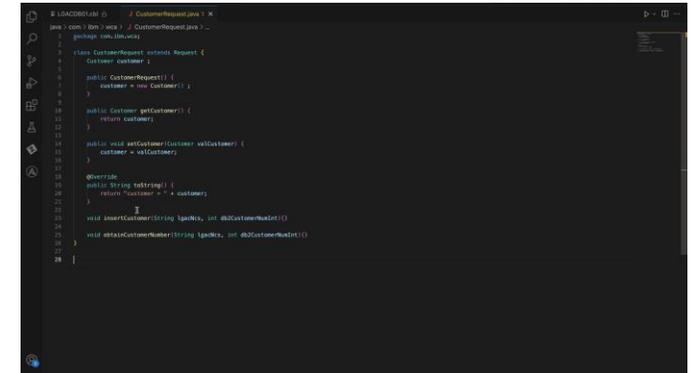
Experiencia de Cliente



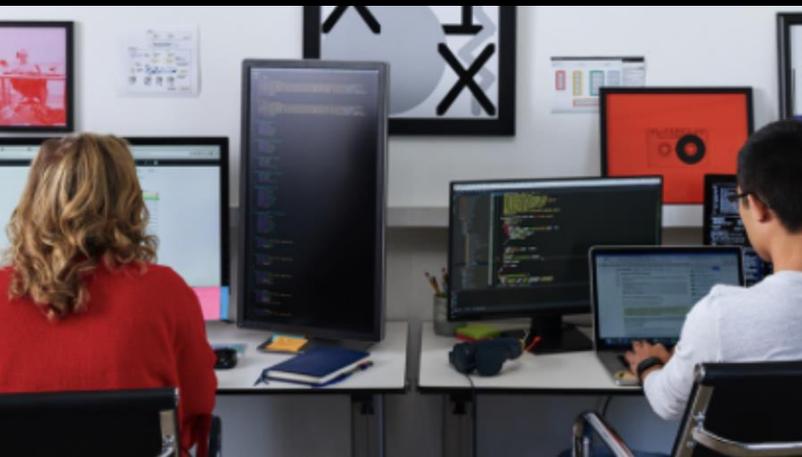
Productividad del Empleado



Gestión del conocimiento



# Productividad del Empleado



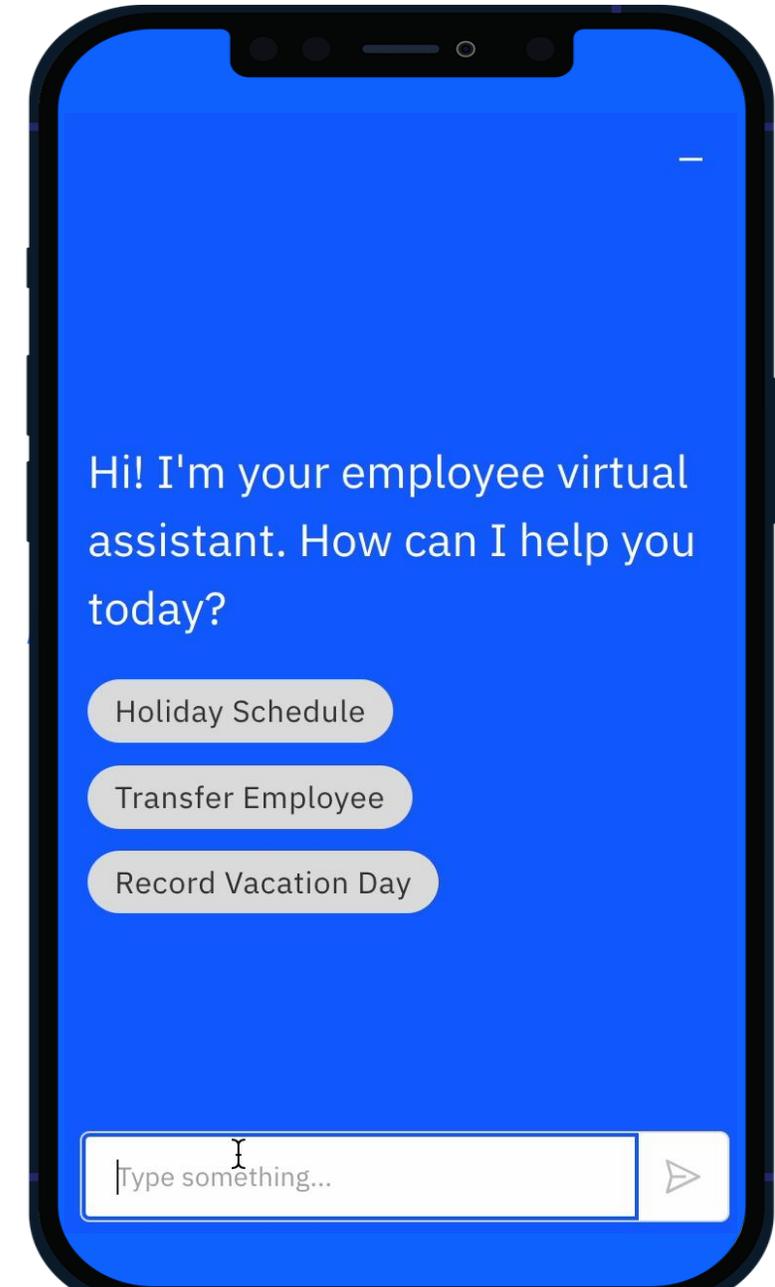
Consultas y ejecución de trámites RRHH

Asistente integrado con múltiples automatizaciones de todos los procesos HR



94% de las tareas de RRHH de IBM automatizadas a través de AskHR

73% sentimiento positivo en el feedback



# Gestión del Conocimiento

Asistente virtual con interfaz sencilla

Consultas en lenguaje natural



Reducción de tiempo en consultas y búsqueda

Información 24x7

Menor curva de aprendizaje

```
java > com > bin > vca > J CustomerRequest.java > ? CustomerRequest > ? InsertCustomer(String, int)
1 package com.ibm.vca;
2
3 class CustomerRequest extends Request {
4     Customer customer;
5
6     public CustomerRequest() {
7         customer = new Customer();
8     }
9
10    public Customer getCustomer() {
11        return customer;
12    }
13
14    public void setCustomer(Customer valCustomer) {
15        customer = valCustomer;
16    }
17
18    @Override
19    public String toString() {
20        return "Customer = " + customer;
21    }
22
23    void insertCustomer(String lgaNcs, int @b2CustomerNumInt);
24
25    void obtainCustomerNumber(String lgaNcs, int @b2CustomerNumInt);
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
Java Method generation
Java Method: insertCustomer
Java Class: CustomerRequest
Generated from COBOL file: LGAC0001
Generated from COBOL Paragraph: INSERT-CUSTOMER

void insertCustomer(String lgaNcs, int @b2CustomerNumInt)
ErrorHandler errorHandler = new ErrorHandler();

/* Insert the into customer table based on customer number */
if (lgaNcs.equals("M")) {
    try {
        String sql = "INSERT INTO CUSTOMER (CUSTOMERNUMBER, FIRSTNAME, LASTNAME,
        DATEOFBIRTH, HOUSENUMBER, POSTCODE, MOBILEPHONE,
        EMAILADDRESS VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
        PreparedStatement ps = @b2ConnectLink.connectLink.prepareStatement(sql,
        Statement.RETURN_GENERATED_KEYS);
        ps.setInt(1, @b2CustomerNumInt);
        ps.setString(2, this.customer.getFirstName());
        ps.setString(3, this.customer.getLastName());
        ps.setDate(4, this.customer.getDateOfBirth());
        ps.setString(5, this.customer.getHouseNumber());
        ps.setString(6, this.customer.getHouseNumber());
        ps.setString(7, this.customer.getPostalCode());
        ps.setString(8, this.customer.getPhoneNumber());
        ps.setString(9, this.customer.getPhoneNumber());
        ps.setString(10, this.customer.getEmailAddress());
        ps.executeUpdate();
    } catch (SQLException e) {
        errorHandler.handleError(e);
    }
}

Insert
Rate the Java generated by watson.
```

**watsonx**

