Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

Aplicaciones con Tecnología a Internet

Seminario

Tópicos a considerar para el desarrollo de una aplicación Web orientada al área de Recursos humanos de cualquier organización

Seminario

presentado ante la ilustre

Universidad Central de Venezuela

por los bachilleres

**Kevin D. Miranda F.**, C.I.: 26.150.260

**Luis E. Campos M.**, C.I.: 18.033.084

Tutor

**Prof. José Sánchez**

Caracas, Junio de 2018

# Índice General

[**Índice General**](#_n65g8kdd5l1e) **1**

[**Índices de figuras**](#_l728o4phtdxn) **9**

[**Capítulo 1**](#_rzvivz6m822q) **11**

[**Parte I - Introducción**](#_g3kkem2vbas9) **11**

[**Capítulo 2**](#_lxjj5zt4ek6a) **14**

[**Parte II - Marco teórico**](#_f1edosaknp4f) **14**

[2.1 Antecedentes](#_sae91j301sv6) 14

[2.2 Características positivas y negativas de los antecedentes](#_ps2x7aawypjo) 16

[2.3 Problema de investigación](#_afauz8lvu05v) 17

[**Capítulo 3**](#_5rdeqxo4v6pr) **18**

[Sistemas de recursos humanos](#_fjeaauoq9nr8) 18

[3.1 Sistema de Recursos Humanos](#_731o4cfi6xj7) 18

[3.2 Funciones Claves del Sistema de Recursos Humanos](#_jnxthjt42zgo) 18

[3.2.1 Reclutamiento y Selección](#_ff4af6wtkrwj) 19

[3.2.2 Inducción](#_vfnagu7a4ohq) 19

[3.2.3 Capacitación](#_eaqnck6c0yzy) 19

[3.2.4 Remuneraciones](#_y5mx79iunkfg) 20

[3.2.5 Motivación y clima laboral](#_576l2ap77x3w) 20

[3.3 Portales Web que proporcionan la búsqueda de empleo](#_y8dlrrreudfj) 20

[Figura 1 - Interfaz del sitio web bumeran.com](#_5u0ldxsk23ay) 21

[3.3.1 Bumeran](#_5vsd4wvs2ndt) 21

[Figura 2 - Interfaz del sitio web computrabajo.com](#_rffi0fcq9iic) 22

[3.3.2 CompuTrabajo](#_g0vivfbm8fgn) 22

[Figura 3 - Interfaz del sitio web empleate.com](#_oxlf0tdxtskm) 23

[3.3.3 Empléate](#_7vlfox8xcadq) 23

[Figura 4 - Interfaz del sitio web buscojobs.com](#_5lc02mbqk56j) 24

[3.3.4 BuscoJobs](#_vcoz7oioplba) 24

[Figura 5 - Interfaz del sitio web trabajando.com](#_mc9y4df3cqcw) 25

[3.3.5 Trabajando.com](#_3cgidykl4qhr) 25

[3.4 Inferencias sobre los antecedentes](#_zae8hpdxgplc) 26

[**Capítulo 4**](#_i02fp0dg1cec) **27**

[Proceso de contratación de personal en las organizaciones.](#_tamq87yoh8ba) 27

[4.1 Organización](#_e6jw1a1hi5wy) 27

[4.2 Proceso de Contratación](#_notfdsdbbm48) 27

[4.2 Proceso de selección](#_yx35kwvlro3h) 28

[4.3 Procesos de reclutamiento de empresas nacionales](#_pmdzomtygkk3) 30

[4.3.1 Farmatodo](#_3e6cjn4gukxo) 31

[4.3.2 Sudeban](#_prp438qzb2ne) 32

[4.3.3 Cantv](#_alrtjzswx9r8) 33

[4.3.4 Strix Technologies](#_obfetu5vulal) 34

[4.4 Procesos de reclutamiento de empresas internacionales](#_kxsxjxqkcnnl) 35

[4.4.1 Bairesdev](#_lloo7lm4g7p5) 35

[4.4.2 Crossover](#_fk71zlebz2q3) 37

[4.4.3 Google](#_onui6agtfqtx) 38

[**Capítulo 5**](#_my03ajyc77in) **43**

[Framework de Desarrollo Web](#_86f1uzkvdyuw) 43

[5.1 Framework](#_1kdvdp8vbfj) 43

[5.2 Framework de Desarrollo Web](#_sma8f92zigwp) 43

[5.3 Componentes de un Framework de Desarrollo Web](#_30h86bcq5uvf) 43

[5.3.1 Librerías](#_igz112u1c3eh) 44

[5.3.2 APIs (Application Program Interface)](#_oof217q1xum1) 44

[5.3.3 Scaffolding](#_lsdtru14qm74) 44

[5.3.4 AJAX](#_lbb5s21tu8ut) 44

[Figura 6 - Ejemplo de una llamada AJAX](#_a6e8tsf0qu9y) 45

[5.3.5 Caching](#_uodzqfbpn2cj) 45

[5.3.6 Seguridad](#_eh9m6wm4h1ls) 45

[5.4 Objetivos Principales de un Framework](#_iqgad7rsqc5p) 46

[**Capítulo 6**](#_8rbgxnbtvkf) **47**

[Sistemas manejadores de contenido](#_ms0an4oyfgwv) 47

[6.1 Sistema Manejador de Contenido](#_c8nfhw5k4dzt) 47

[6.2 Organización y Estructura de un CMS](#_6h2bwxfup5te) 48

[6.3 Flujo de Trabajo de un CMS](#_sg5rw2jf0nro) 48

[6.3.1 Flujo de un CMS](#_rrge5smglzay) 49

[Figura 7 - Flujo de trabajo, funcionamiento de un CMS](#_6xxxnu7frvga) 49

[**Capítulo 7**](#_qeyjdj9z7yrl) **51**

[Aplicaciones configurables o parametrizables](#_ixbjow7m5q87) 51

[7.1 Aplicaciones configurables para organizaciones o clientes](#_kq94b87ank02) 51

[7.1.1 Apariencia](#_1qehuu2w9ec2) 51

[Figura 8 - Ejemplo de Aplicación configurable en apariencia](#_m36c77l50roy) 52

[Figura 9 - Ejemplo de aplicación configurable en apariencia](#_ek6338eyya23) 53

[7.1.2 Plantilla o diseño](#_1cu139qkt1fy) 53

[Figura 10 - Ejemplo de cómo Google permite distribuir sus características](#_5jrn6uf7fxg) 54

[Figura 11 - Ejemplo de una aplicación cuyo contenido puede ser desplazado](#_xubl8gpszblr) 55

[7.1.3 Contenido](#_51ytsfq55by0) 55

[Figura 12 -Ejemplo configuración de etiquetas al generar un contenido](#_nzc1h2tj09mm) 56

[Figura 13 - Ejemplo de configuración de prioridad y tamaño de las etiquetas](#_63g60gqggkhr) 57

[7.1.4 Navegación](#_p3liow92hfhz) 57

[Figura 14 - Ejemplo de navegación dependiendo del rol de usuario](#_4sw9usbrsdop) 58

[7.1.5 Interacciones](#_jh60wne74olj) 58

[Figura 16 - Ejemplo de definición de reglas de negocio](#_3srcmbhe8se) 60

[7.2 Evaluar cuando una aplicación debe ser configurable](#_r6lqz09m3fq2) 60

[**Capítulo 8**](#_8c8ls8afbhs8) **61**

[Frameworks de desarrollo Web vs Sistemas manejadores de contenido](#_z0g6tccdxp3z) 61

[8.1 Framework de Desarrollo web para el desarrollo de aplicaciones configurables](#_w6t6f6pr7tv8) 61

[8.1.1 Ventajas](#_93wfxrhsb6nv) 61

[8.1.2 Desventajas](#_ki078m69564l) 63

[8.2 Sistema manejador de contenidos para el desarrollo de aplicaciones configurables](#_o7ub8v5838xl) 63

[8.2.1 Ventajas](#_45cjulc0q0wx) 64

[8.2.2 Desventajas](#_os8axj4v7ugd) 64

[8.3 Framework de desarrollo Web vs Sistema manejador de contenido para el desarrollo de aplicaciones configurables](#_6ep4hg757ddt) 65

[8.3.1 Seguridad](#_2lvcsox0gz4e) 65

[8.3.2 Personalización](#_l9zulcmybct6) 66

[8.3.3 Flexibilidad](#_p3gjr78vfc2e) 67

[8.3.4 Actualizaciones](#_im3at7oyy15e) 67

[**Capítulo 9**](#_7417lkl66hym) **68**

[**Parte III. Marco metodológico**](#_3tw5u3xnj5lm) **68**

[9.1.1 Todo el equipo](#_dbmwf9619ppf) 68

[9.1.2 Planificación del esquema de trabajo](#_3vx7brc4h0t5) 69

[9.1.3 Iteración de planificación](#_vj7quwx04j7t) 70

[9.1.4 Pruebas al cliente](#_z6s2eziwmd) 70

[9.1.5 Pequeños entregables](#_5lbblxs8bk9x) 71

[9.1.6. Diseño simple](#_jxzsshgi8ob0) 72

[9.1.7 Programación en parejas](#_ztixdy0n9fa) 73

[9.1.8 Pruebas de desarrollo](#_96h29wxc66z7) 74

[9.1.9 Mejora de diseño](#_er5ro14ecxlr) 74

[9.1.10 Integración continúa](#_1bl2ujcs5o57) 76

[9.1.11 Código de la propiedad colectiva](#_vocx42r41vd) 77

[9.1.12 Codificación standard](#_3gfnp49x23yt) 78

[9.1.13 Metáfora](#_uvynneptdt5n) 78

[9.1.14 Desarrollo sostenible](#_o68bpc5f1d2t) 79

[9.2 ¿En qué se diferencia de otras metodologías?](#_lfvh4dyrjoou) 79

[**Capítulo 10**](#_rui756w1powh) **80**

[Seguimiento y control del presente estudio](#_h7d4tgir2mnd) 80

[10.1 Proyecto](#_85xh38rtdp42) 80

[10.2 Gerencia del proyecto](#_93awpwoqtl18) 80

[10.3 Metodología de gestión de proyectos a utilizar](#_6e9dffkc8a7g) 81

[10.3.1 Gestión del alcance del proyecto](#_hdgfk4sje3vq) 81

[10.3.1.1 Planificar Gestión del Alcance](#_tw3wf5hp91r6) 81

[10.3.1.2 Recopilar Requisitos](#_5w0m7qbfz75q) 82

[10.3.1.3 Definir el Alcance](#_ni1wsyw0j9wi) 82

[10.3.1.4 Crear la EDT/WBS](#_tj652lqyri0v) 82

[10.3.1.5 Validar Alcance](#_16kvz7km2js4) 83

[10.3.1.6 Controlar el Alcance](#_ju561bp14s10) 83

[10.3.2 Gestión del tiempo](#_hh8e2r9lcfwl) 83

[10.3.2.1 Planificar la Gestión del Cronograma](#_qw3gqwsn6bdx) 83

[10.3.2.2 Definir las actividades](#_8bd39ytrj19g) 84

[10.3.2.3 Secuenciar las actividades](#_74p1surqqdjl) 84

[10.3.2.4 Estimar los recursos de las actividades](#_ytwbjvc6zlzo) 84

[10.3.2.5 Estimar la duración de las actividades](#_ruwfeexon78d) 84

[10.3.2.6 Desarrollar el Cronograma](#_hllr6ey8j339) 85

[10.3.2.7 Controlar el Cronograma](#_5syz5m3zijzv) 85

[10.3.3 Gestión del costo del proyecto](#_rqa7si3x1a13) 85

[10.3.3.1 Planificar la Gestión de los Costos](#_27vw1yusucqh) 86

[10.3.3.2 Estimar los Costos](#_fqzfjlmx9wgf) 86

[10.3.3.3 Determinar el presupuesto](#_3dzd8lp04zeq) 86

[10.3.3.4 Controlar los Costos](#_onq93row5z0q) 86

[10.3.4 Gestión de calidad](#_9pohx373hnc2) 87

[10.3.4.1 Planificar Gestión de Calidad](#_2zwh3pcoov16) 87

[10.3.4.2 Realizar Aseguramiento de Calidad](#_ch7fpx5re6bk) 87

[10.3.4.3 Controlar la Calidad](#_ufri5zfl93p2) 88

[10.2.5 Gestión de las comunicaciones](#_xxs4hqlz8t17) 88

[10.2.5.1 Planificar la Gestión de las Comunicaciones](#_55hpfxgt8h8f) 88

[10.2.5.2 Gestionar las Comunicaciones](#_uazbtlax7vyx) 89

[10.2.5.3 Controlar las Comunicaciones](#_mdsdrysd8e63) 89

[Consideraciones éticas y legales](#_ttkzkghama97) 89

[**Parte IV.**](#_gdhw78vf19g1) **90**

[Propuesta de Trabajo Especial de Grado](#_6fjat4pkd6c5) 90

[**Capítulo 11**](#_a4tidlb53ryz) **90**

[Análisis y Propuesta](#_qhw3r3u2ijwa) 90

[11.1 Caso de estudio](#_lspf6wxmub25) 90

[11.2 Situación actual](#_gptmaircyswc) 90

[11.2.1 Inconvenientes para el candidato](#_9t6prd2y4sdn) 93

[11.2.2 Inconvenientes para las organizaciones](#_ixok4ls9lpt7) 95

[**Capítulo 12**](#_1nh9q0om92eh) **98**

[Propuesta Trabajo Especial de Grado](#_r9tkl7evxm00) 98

[12.1 Justificación](#_htxgqg5ijlc9) 100

[12.2 Objetivo General](#_cnuewxs5njym) 103

[12.3 Objetivos específicos.](#_4526m8h790mz) 103

[12.3.1 Analizar y definir los requerimientos para la aplicación Web](#_i3iygwlap6xg) 103

[12.3.2 Realizar el diseño de la aplicación Web.](#_1riifdfvsoit) 104

[12.3.3 Definir los componentes de la aplicación Web que serán configurables o parametrizables de forma detallada.](#_rkbwxm72zt30) 105

[12.3.4 Desarrollar la aplicación Web](#_4xriagp9jnkd) 105

[12.3.5 Realizar pruebas sobre la aplicación Web](#_u07mr5y9edfe) 106

[12.3.6 Elaborar los manuales de instalación y de uso de las aplicaciones](#_gl07nsol7jrz) 106

[12.4 Alcances y Limitaciones](#_hl0qzghiwpje) 106

[12.4.1 Alcance](#_bw3f6w8rvpul) 106

[12.4.2 Limitaciones](#_wpah3pnyqd1g) 107

[12.4.2.1 Factores culturales o sociales](#_8z7khy7l5hs) 108

[12.4.2.2 Falta de instrumentos para mediciones técnicas](#_j1a8fvyk06me) 109

[12.4.2.3 Características de las muestras](#_22m99uym9l3b) 109

[12.4.2.4 Sesgo del Sujeto](#_gd6x6sggkcn2) 109

[12.4.2.5 Disposición del candidato](#_rqe899wjnltw) 110

[**Capítulo 13**](#_qx2zuc4su0gt) **111**

[Tecnologías propuestas](#_t77ac3yak7xy) 111

[13.1 Tecnologías del lado del cliente (Capa de presentación)](#_4i2rc8rj0grk) 111

[13.1.1 HTML 5 Lenguaje de Marcas de Hipertexto](#_qa4bz3gjoo39) 111

[13.1.2 HTML5 vs HTML4.01](#_eq62l14nxbf0) 112

[Figura 18 - Estructura de un sitio Web sencillo con HTML5](#_615y6gk2nixr) 114

[Figura 19 - Diferencias entre la estructura del código HTML4.01 y HTML5](#_z0g295j1octh) 115

[13.1.2 CSS3 Hojas de Estilo en Cascada](#_bj4pi6vx8ci3) 115

[Figura 20 - Diferencia entre los múltiples selectores HTML](#_ul2oylu6mck6) 116

[13.1.3 Bootstrap](#_glob1g9dj8xp) 117

[Figura 21 - Grid System del framework Bootstrap para dispositivos pequeños.](#_v8w09nf6ramv) 118

[Figura 22 - Diferencia entre la visualización de navegación entre los diferentes tamaños de dispositivos](#_luv2zvdagurp) 118

[13.1.4 JavaScript](#_pi6n41aqdgpp) 118

[Figura 23 - Ejemplo del llamado a un archivo con extensión js.](#_ooj5jqml4r8b) 120

[Figura 24 - Ejemplo del código Javascript embebido en el código HTML.](#_3ckyyrbw9jw) 120

[13.2 Tecnologías del lado del servidor (Núcleo de la aplicación)](#_lqitcsh3gobp) 120

[13.2.1 PHP](#_wfitq9nklb85) 120

[13.2.2 Laravel](#_nd0hx6mrdzh4) 121

[Figura 25 - Icono oficial de Laravel](#_xwpfi67kh99) 122

[13.2.2.1 Evolución del Framework](#_2a1m2fl03p95) 122

[1.- Laravel](#_60e425srczv0) 122

[2.- Laravel 2](#_kyfv3q59mm7f) 122

[3.- Laravel 3](#_je0ent7ettmj) 123

[4.- Laravel 4](#_72bzec6cp757) 123

[5.- Laravel 5](#_1ncm389swiv2) 123

[13.2.2.2 Características principales de Laravel](#_snp5esodxp8x) 125

[1.- Paquetes](#_hkz01xnrzsiw) 125

[2.- Eloquent ORM](#_1m7evlpz0iw8) 125

[3.- Lógica de la Aplicación](#_3j0sy41bhhlw) 126

[4.- Enrutamiento inverso](#_f0937x6rgasi) 126

[5.- Controladores RESTful](#_62wtdquncxfq) 126

[6.- Carga automática de Clases](#_myurgw6wxqrx) 126

[7.- View Composers](#_rm8nodegf8rj) 127

[8.- IoC Container (Contenedor de Control de Inversión)](#_p6gn9nqi2ba7) 127

[9.- Migraciones](#_br1fcwqtxz7p) 127

[10.- Pruebas Unitarias](#_qsvtvqtqj9v3) 127

[11.- Paginación Automática](#_8ytvj06nyp85) 128

[**Capítulo 14**](#_zdi7mwkns8x9) **129**

[Patrones de Diseño](#_u7ginie8yxyw) 129

[14.1 ¿Porqué utilizar patrones de diseño?](#_vtvf9pheqbp) 129

[14.2 Desventajas de utilizar patrones de diseño](#_vmxetffdkmo) 130

[1. Patrón en capas:](#_32zk4nj0tlkm) 130

[Figura 26 - Diagrama de patrón en capas](#_61nvmuc64oqx) 131

[2. Patrón cliente-servidor](#_lwifw24buqrq) 131

[Figura 27 - Diagrama de patrón Cliente Servidor](#_pz0zf0hqphf9) 132

[3. Patrón maestro-esclavo](#_i2jdc319k6lo) 132

[Figura 28 - Diagrama de patrón maestro esclavo](#_u3v3ccrwbdjx) 133

[4. Patrón de filtro de tubería](#_c2zhwlwmtz6b) 133

[Figura 29 - Diagrama de filtro tubería](#_3knrysg0r6af) 134

[5. Patrón del agente mediador](#_kpvqhqkcbdex) 134

[Figura 30 - Diagrama de patrón mediador](#_9lap4do2u2dh) 135

[6. Patrón de punto a punto (peer to peer)](#_8yvyvcro9wbu) 135

[Figura 31 - Diagrama de patrón punto a punto](#_uwh3fscqrr43) 136

[7. Patrón de bus de evento](#_3c4btbw6pcvu) 136

[Figura 32 - Diagrama de patrón bus de eventos](#_8tbvsqopa1ly) 137

[8. Patrón de modelo-vista-controlador](#_tbsu8k7avn14) 137

[Figura 33 - Diagrama de patrón modelo vista controlador](#_wds5nghnp3s0) 138

[9. Patrón de pizarra](#_cgx4i6wor4ir) 138

[Figura 34 - Diagrama de patrón pizarra](#_2g7dfru0pwx6) 139

[10. Patrón de intérprete](#_7zd63nfhxb7o) 140

[Figura 35 - Diagrama de patrón intérprete](#_51upu7wwfffr) 140

[Tabla Comparativa 1 - comparación](#_ma3eorujzxg4) 144

[Figura 36 - Diagrama de nivel de abstracción](#_r1ctl0bpmwp1) 144

[**Capítulo 15**](#_8u7jjz63z236) **144**

[Sistema Manejador de Base de Datos](#_64q8f1j8bsgv) 145

[15.1 PostgreSQL](#_vvnoeykg385h) 145

[15.2 Características](#_yl13e5332co1) 146

[1.-MVCC](#_o967r2jo3oa3) 146

[2.-Hot-Standby](#_i1zfw59lavjo) 146

[3.-Flexibilidad](#_ttsxcpqzsor) 146

[4.-Multiplataforma](#_tl526xwpe3kr) 147

[5.-pgAdmin](#_uvy1stg6mdkx) 148

[Figura 37 - Interfaz pgAdmin](#_eupb969g6ky3) 148

[Figura 38 - Query Tool, muestra gráfica del plan de ejecución](#_v9v2q96p0jmu) 149

[15.3 Ventajas](#_wqaup2d0vu70) 149

[15.4 Desventajas](#_225af0soals2) 150

[15.5 Comparación de otros Sistemas Manejador de Base de Datos](#_8575zbt42o0s) 151

[15.5.1 PostgreSQL vs MySQL](#_9lgcx5128va0) 151

[15.5.2 PostgreSQL vs Oracle](#_s72wh5wsu088) 152

[**Capítulo 16**](#_7c78w9s92ujn) **152**

[Servicios Web](#_jts79evzc2oz) 153

[16.1 Tendencias Principales Backend](#_bs2mncmm992h) 153

[16.2 ¿Porque REST y no SOAP?](#_iqqsdxf8ftxp) 156

[16.3 Ventajas y desventajas](#_52hj42pfjdxa) 156

[16.4.1 Seguridad](#_uit67r90yr54) 156

[16.4.2 Desempeño](#_uk1346vwnbi7) 157

[16.4.3 Complejidad](#_ele49zjv9jdv) 157

[Test Driven Development (TDD)](#_2xmhdffadrol) 159

[17.1 ¿Por qué usar TDD?](#_8fmmu4tdzfk4) 159

[17.2 ¿Cómo se instala?](#_6q0v96qjo5hg) 160

[17.3 ¿Cómo se utiliza?](#_3g4rn9e5aksj) 160

[17.4 Depuración de errores en TDD](#_mei8lsg036qw) 161

[17.5 Ventajas](#_dtiu7qgqk1bj) 161

[17.6 Desventajas](#_rj7wpnx8nqz4) 162

[**Referencias Bibliográficas y Digitales**](#_vn84sikthx5q) **164**

# Índices de figuras

[Figura 1 - Interfaz del sitio web bumeran.com](#_5u0ldxsk23ay) [20](#_5u0ldxsk23ay)

[Figura 2 - Interfaz del sitio web computrabajo.com](#_rffi0fcq9iic) [21](#_rffi0fcq9iic)

[Figura 3 - Interfaz del sitio web empleate.com](#_oxlf0tdxtskm) [22](#_oxlf0tdxtskm)

[Figura 4 - Interfaz del sitio web buscojobs.com](#_5lc02mbqk56j) [23](#_5lc02mbqk56j)

[Figura 5 - Interfaz del sitio web trabajando.com](#_mc9y4df3cqcw) [24](#_mc9y4df3cqcw)

[Figura 6 - Ejemplo de una llamada AJAX](#_a6e8tsf0qu9y) [45](#_a6e8tsf0qu9y)

[Figura 7 - Flujo de trabajo, funcionamiento de un CMS](#_6xxxnu7frvga) [49](#_6xxxnu7frvga)

[Figura 8 - Ejemplo de Aplicación configurable en apariencia](#_m36c77l50roy) [53](#_m36c77l50roy)

[Figura 9 - Ejemplo de aplicación configurable en apariencia](#_ek6338eyya23) [54](#_ek6338eyya23)

[Figura 10 - Ejemplo de cómo Google permite distribuir sus características](#_5jrn6uf7fxg) [55](#_5jrn6uf7fxg)

[Figura 11 - Ejemplo de una aplicación cuyo contenido puede ser desplazado](#_5uoat1vtbdo2) [56](#_5uoat1vtbdo2)

[Figura 12 -Ejemplo configuración de etiquetas al generar un contenido](#_nzc1h2tj09mm) [57](#_nzc1h2tj09mm)

[Figura 13 - Ejemplo de configuración de prioridad y tamaño de las etiquetas](#_63g60gqggkhr) [58](#_63g60gqggkhr)

[Figura 14 - Ejemplo de navegación dependiendo del rol de usuario](#_4sw9usbrsdop) [59](#_4sw9usbrsdop)

[Figura 15 - Ejemplo de definición de flujo de información](#_93q46gbmmgb2) [61](#_93q46gbmmgb2)

[Figura 16 - Ejemplo de definición de reglas de negocio](#_zalyoqe438iv) [61](#_zalyoqe438iv)

[Figura 17 - Estructura de un sitio Web sencillo con HTML4.01](#_a2038isdhctz) [112](#_a2038isdhctz)

[Figura 18 - Estructura de un sitio Web sencillo con HTML5](#_615y6gk2nixr) [113](#_615y6gk2nixr)

[Figura 19 - Diferencias entre la estructura del código HTML4.01 y HTML5](#_z0g295j1octh) [114](#_z0g295j1octh)

[Figura 20 - Diferencia entre los múltiples selectores HTML](#_mm4dqgcunvrp) [116](#_mm4dqgcunvrp)

[Figura 21 - Grid System del framework Bootstrap para dispositivos pequeños.](#_v8w09nf6ramv) [117](#_v8w09nf6ramv)

[Figura 22 - Diferencia entre la visualización de navegación entre los diferentes tamaños de dispositivos](#_mrnttl447n70) [118](#_mrnttl447n70)

[Figura 23 - Ejemplo del llamado a un archivo con extensión js.](#_ooj5jqml4r8b) [119](#_ooj5jqml4r8b)

[Figura 24 - Ejemplo del código Javascript embebido en el código HTML.](#_3ckyyrbw9jw) [119](#_3ckyyrbw9jw)

[Figura 25 - Icono oficial de Laravel](#_xwpfi67kh99) [121](#_xwpfi67kh99)

[Figura 26 - Diagrama de patrón en capas](#_61nvmuc64oqx) [130](#_61nvmuc64oqx)

[Figura 27 - Diagrama de patrón Cliente Servidor](#_pz0zf0hqphf9) [131](#_pz0zf0hqphf9)

[Figura 28 - Diagrama de patrón maestro esclavo](#_u3v3ccrwbdjx) [132](#_u3v3ccrwbdjx)

[Figura 29 - Diagrama de filtro tubería](#_3knrysg0r6af) [133](#_3knrysg0r6af)

[Figura 30 - Diagrama de patrón mediador](#_9lap4do2u2dh) [134](#_9lap4do2u2dh)

[Figura 31 - Diagrama de patrón punto a punto](#_uwh3fscqrr43) [135](#_uwh3fscqrr43)

[Figura 32 - Diagrama de patrón bus de eventos](#_8tbvsqopa1ly) [136](#_8tbvsqopa1ly)

[Figura 33 - Diagrama de patrón modelo vista controlador](#_oasldqhhf08d) [137](#_oasldqhhf08d)

[Figura 34 - Diagrama de patrón pizarra](#_2g7dfru0pwx6) [139](#_2g7dfru0pwx6)

[Figura 35 - Diagrama de patrón intérprete](#_51upu7wwfffr) [140](#_51upu7wwfffr)

[Tabla Comparativa 1 - comparación](#_ma3eorujzxg4) [143](#_ma3eorujzxg4)

[Figura 36 - Diagrama de nivel de abstracción](#_r1ctl0bpmwp1) [144](#_r1ctl0bpmwp1)

[Figura 37 - Interfaz pgAdmin](#_eupb969g6ky3) [148](#_eupb969g6ky3)

[Figura 38 - Query Tool, muestra gráfica del plan de ejecución](#_v9v2q96p0jmu) [149](#_v9v2q96p0jmu)

# Capítulo 1

# Parte I - Introducción

La necesidad de contratar personal en las empresas viene dado por diferentes causas como expansión de la empresa, generación de nuevos puestos de trabajo, rotación de personal, un requerimiento o simple necesidad de fuerza laboral, y lo cual depende de las capacidades y aptitudes del talento humano, por lo cual es de vital importancia realizar una selección adecuada de candidatos que contribuyan al éxito de las organizaciones.

En la actualidad, la mayoría de los usuarios revisan portales de empleo a través de internet, y pueden visualizar información en relación a las vacantes de su interés. El usuario decide postularse, y de ser seleccionado comienza un proceso de evaluación por parte de la empresa.

Dicho proceso involucra la interacción de la empresa y del usuario, y en el cual se origina una serie de inconvenientes para ambas partes, originando pérdidas de tiempo, dinero, de recursos humanos aptos para dichas vacantes, frustración, problemas de competitividad en las empresas, entre otras.

Entre las causas se puede determinar la creación de procesos ineficiente en la selección, evaluación y contratación de talento humano que se encuentra en la búsqueda de oportunidades laborales; y en donde algunas empresas deciden crear sus propios procesos de selección, evaluación y contratación de personal, y en otros casos, contratan a otras empresas para realizar dichas labores, pero incluso, las empresas especializadas en la búsqueda y selección de talento, han perdido a excelentes profesionales, por utilizar procesos convencionales que no les permite manejar un volumen considerable de candidatos, de forma eficaz.

Por lo antes expuesto, se propone el desarrollo de una aplicación web, que sea configurable, que permita generar y ofrecer información acerca de las diferentes vacantes ofrecidas por una empresa, que permita definir y personalizar las diferentes etapas de de selección y evaluación de potenciales candidatos, que permitiendo atender a una gran cantidad de candidatos de manera simultánea, y proporcionales una respuesta rápida; en cada una de las etapas del proceso de selección; permitiendo que las empresas puedan descartar a diversos candidatos, de forma rápida y eficiente, sin originar los inconvenientes descritos; y que administre la información obtenida de los candidatos de manera que permita generar métricas que la empresa pueda utilizar para tomar mejores decisiones acerca de sus procesos de selección.

Por dicha razón se ha optado por la propuesta de utilizar tecnologías del lado del cliente como Bootstrap, JavaScript, y jQuery; así como tecnologías del lado del servidor como el lenguaje de programación Php, y el Framework Laravel, utilizado el patrón MCV, y patrones de diseño para su construcción, utilizando la metodología de desarrollo Extremo XP, ya que permiten un desarrollo rápido robusto y con mucho soporte documental de como aplicarlo, además de proveer una abstracciones útiles entre las fases del proyecto, aplicables al desarrollo y progreso de la aplicación, y que puede escalar sin importar la cantidad de personas e independientemente del propósito u de negocio de la aplicación.

Adicionalmente, se considera, el utilizar la metodología de gerencia de proyectos establecida por el PMI (Project Management Institute), para generar conocimientos y experiencia en dicha metodología para los autores del presente trabajo; y para permitir un control y seguimiento del desarrollo de la presente investigación y del trabajo especial de grado, que permita incrementar la posibilidad de que los trabajos establecidos finalicen en los tiempos estimados.

El presente documento está estructurado de la siguiente manera:

* **Parte I - Introducción:** Sección en la que se expone el problema, de los métodos convencionales de reclutamiento y sus procesos. En resumen, la propuesta para solucionar dichos problemas.
* **Parte II - Marco Conceptual:** se exponen los aspectos teóricos relacionados con los procesos, metodologías, patrones y tecnologías a utilizar para la implementación de la aplicación web, desde una perspectiva imparcial, y con la finalidad de presentar tanto las ventajas y desventajas de cada propuesta.
* **Parte III - Propuesta de TEG:** Se plantea el contexto del problema, la situación actual, la solución propuesta, la justificación de la solución planteada, así como el objetivo general y los objetivos específicos que se desean lograr.

# Capítulo 2

# Parte II - Marco teórico

La presente sección ilustra los antecedentes y las bases conceptuales que sirven como base para el desarrollo del trabajo de investigación, lo cual es de vital importancia útil para elaborar ideas que van naciendo de la investigación; por tal motivo, se trata en detalle el estado actual de aplicaciones relacionadas con la gestión de las personas como recursos humanos, conceptos sobre características de la aplicación propuesta y conceptos relacionados, haciendo referencia a investigaciones realizadas por diversos autores, que sustentaron y contribuyeron al desarrollo del presente TEG, por aspectos teóricos que ayudaron a facilitar la comprensión de la información presentada en capítulos posteriores; y aspectos legales asociados al estudio del presente emprendimiento; descritos a continuación:

## 2.1 Antecedentes

En Venezuela y en otras partes del mundo se han realizado diversas aplicaciones web relacionadas con la gestión de contrataciones y filtrado de personas por capacidades profesionales. Por ejemplo, en Venezuela se cuenta con páginas web como Bumeran, compuTrabajo, Beebee, etc.

Que son plataformas que ayudan a sus usuarios a encontrar empleo, tienen bastante cosas en común, pero en general se destacan por tener procedimientos cerrados e invariantes, que no ofrecen flexibilidad y que pueden resultar tediosos por la cantidad requerida de información bajo cualquier rol que se utilicen esas plataformas.

El proceso de reclutamiento y selección de empresas consta de una serie de etapas que se aplica a cada candidato que desee aplicar para el cargo ofrecido.

Inicialmente, la empresa debe agendar y manejar el tiempo de tal forma que puedan atender a la mayor cantidad de candidatos en un día determinado, sin embargo, muchas empresas no poseen una forma de trabajo eficiente que permita evaluar a una gran cantidad de candidatos en el menor tiempo posible debido a que, generalmente, dichas evaluaciones se realiza por una sola persona que dedica parte de su tiempo a aplicar el proceso de selección, evaluación y contratación de a un candidato a la vez lo que causa que el proceso sea lento lo que puede ocasionar que los candidatos decidan retirarse del proceso.

Las empresas realizan impresiones de cuestionarios y encuestas que son aplicados a cada candidato generando pérdidas económicas si el candidato decide abandonar el proceso de selección, o si no es seleccionado para continuar, y si el candidato es seleccionado, se encuentre con el escenario de que dicho candidato ya aceptó una propuesta laboral de otra empresa debido a la lentitud del proceso y falta de feedback a lo largo del proceso. La lentitud del proceso y falta de feedback deja al candidato en incertidumbre en determinar la posibilidad que tiene en ser seleccionado y poder incursionar en el vacante lo que causa que la empresa pierda candidatos potenciales durante el proceso.

Las empresas, para reclutar al candidato ideal para la vacante, realizan diversos filtros que son aplicados a cada candidato a evaluar, lo que ocasiona que los candidatos se sientan incómodos ya que deben solicitar permisos laborales para asistir presencialmente a cada una de las etapas en el proceso de reclutamiento y selección, permisos que, son recuperados en tiempo de oficina o generan pérdidas monetarias en transporte, impresión de documentos como currículos, certificados, títulos, entre otros; esto ocasiona que los candidatos se retiren del proceso de selección por considerarlo lento, costoso y molesto.

## 2.2 Características positivas y negativas de los antecedentes

Estos portales en su mayoría, por su largo tiempo de existencia ya han segmentado un gran número de usuarios, poseen diseños atractivos y funcionales, así como también, recolectado una gran cantidad de información de miles de usuarios, poseen filtros, motores de búsqueda y clasificaciones por categorías, todos ellos, con el propósito de hacerle a los usuarios las búsquedas más simples, todos esas características son inspiración y guía de cómo ayudar a los usuarios a conseguir lo que quieren, en este caso empleo.

Aunque, cabe destacar que, poseen algunas características que causan controversia, como lo sería la rigidez de sus estructuras, no permiten ninguna personalización, teniendo como característica común ser repositorios de curriculums y formularios de datos personales, y por la parte empresarial, poco más que un anuncio de periódico, no ofrecen herramientas como: pruebas, formularios hechos a la medida para el puesto de trabajo, preguntas cruciales, etc. eso justo es lo que queremos aportar con nuestra propuesta de trabajo de grado, hacer herramientas flexibles, que ayuden a seleccionar con mayor precisión los postulantes que desean una vacante y posean los requisitos necesarios para llenarla satisfactoriamente.

## 2.3 Problema de investigación

1. Facilitar la conexión entre los participantes y las instituciones que posean vacantes en roles que los usuarios puedan aprovechar.
2. Velar por el uso adecuado de los recursos ordinarios y extraordinarios de los participantes de los concursos para las vacantes propuestas por las empresas.
3. Mantener una base de datos de los recursos personales de muchos individuos, para poder así contactar aquellos que puedan exceder cumpliendo con los con las vacantes.
4. Velar porque las actividades que se publiquen en la aplicación sean legítimas y cumplan con un mínimo de información para que los usuarios puedan tomar buenas decisiones para postularse.
5. Estudiar y evaluar las proposiciones de cualquier institución que pueda aportar una significativa mejora a los procesos de contratación de personal, y hacer las mejoras correspondientes a los recursos y herramientas del sistema.
6. Divulgar el potencial investigativo y creador de la Facultad de Ciencias, así como los resultados de investigaciones ya realizadas, y los aportes que la universidad puede hacer para solucionar problemas de reclutamiento de personal.

# Capítulo 3

## Sistemas de recursos humanos

En el presente capítulo se explican el sistema de recursos humanos, como participa dicho sistema en la organización y se describen las funciones claves que cumplen.

## 3.1 Sistema de Recursos Humanos

El sistemas de recursos humanos inicialmente eran encargados únicamente al área de contratación y gestión de contratos dentro de las organizaciones pero surgió la necesidad de un sistema para la selección, formación personal y nóminas. En la actualidad el sistema de recursos humanos se encargan de múltiples actividades además de las mencionadas desarrollando nuevas herramientas para la gestión del personal, como es la evaluación de desempeño, competencias. gestión de talentos, bonificaciones, retroalimentación con el personal de la empresa y el reclutamiento y selección de nuevo personal para ocupar los puestos vacantes de la organización.

## 3.2 Funciones Claves del Sistema de Recursos Humanos

Toda empresa que aspire a crecer necesita tener un Sistemas de Recursos Humanos altamente calificados capaces de ejecutar las siguiente funciones vitales para una empresa detallados a continuación:

## 3.2.1 Reclutamiento y Selección

Se planifica con anticipación las necesidades para analizar los puestos vacantes y determinar las características y requisitos mínimos necesarios que debe poseer los candidatos potenciales que opten por postularse donde, utilizando técnicas de reclutamiento como el análisis del currículum vitae de los postulados, cuestionarios, entrevistas, test aptitudinales y revisión de referencias, se selecciona a aquellos candidatos que cumplan con los criterios para realizar las ofertas salariales y beneficios cuyo fin es incluir el nuevo recurso humano en el puesto vacante.

## 3.2.2 Inducción

Se realiza un acompañamiento durante los primeros días para transmitir la cultura de la organización, presentar el equipo de trabajo y el ambiente donde va a ejecutar sus actividades, presentar las medidas de seguridad aplicadas por la organización en caso de alguna emergencia, repasar las funciones y actividades que se esperan y finalmente aclarar las dudas del nuevo empleado en la organización.

## 3.2.3 Capacitación

El personal perteneciente a la organización puede requerir formación adicional ya sea para mejorar el desempeño de sus funciones o para realizar nuevas funciones, el sistema de recursos humanos debe planificar y ofrecer sindicatos, charlas, centros o profesores particulares que ofrecen cursos aplicados en el negocio de diversos como puede ser idiomas, herramientas informáticas, técnicas de ventas, nuevas tecnologías o redes sociales.

## 3.2.4 Remuneraciones

Se definen los sueldos y liquidación según las normas correspondientes y además determina las compensaciones de forma que sean equitativas, reflejen resultados y ayude a mantener la motivación de sus empleados.

## 3.2.5 Motivación y clima laboral

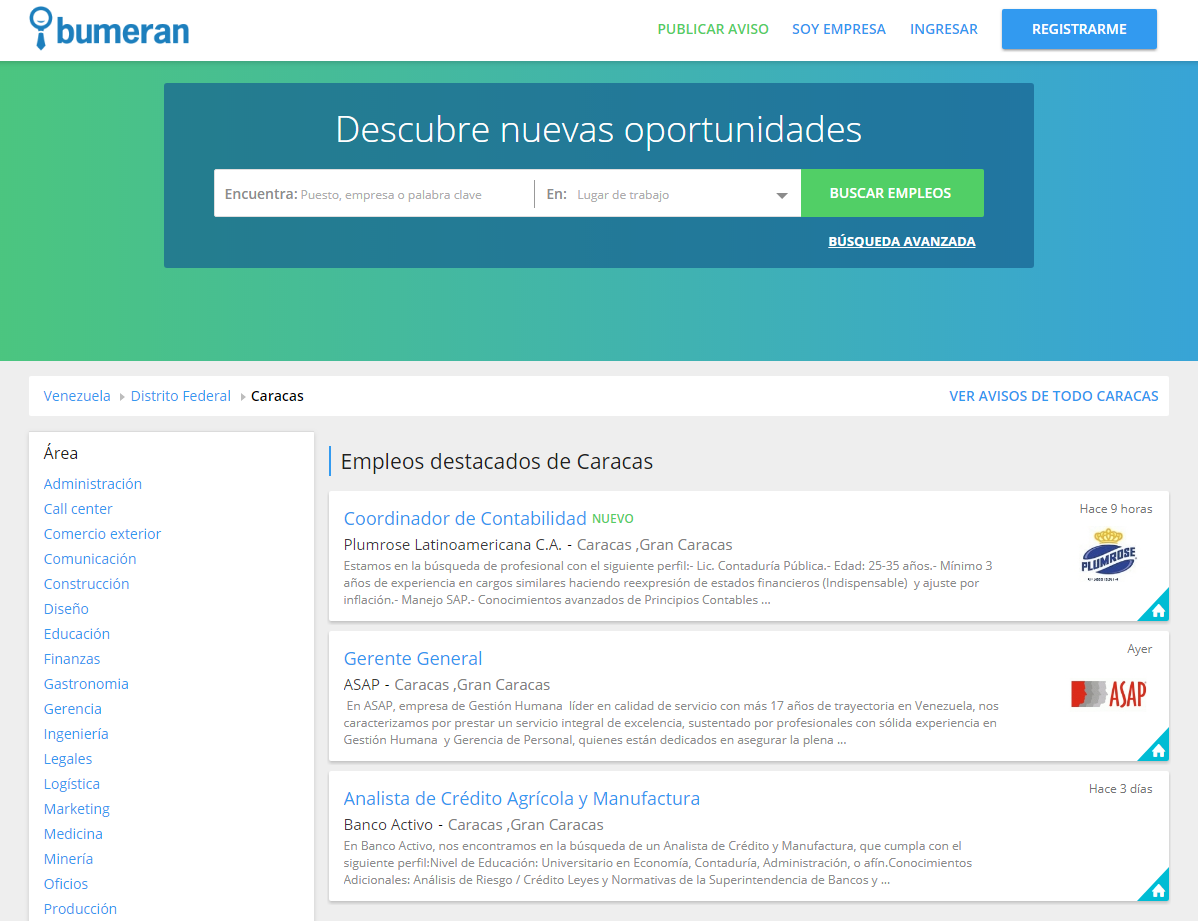
En este se asegura que el ambiente de trabajo sea agradable para todos los empleados que pertenecen a ella para mantener la motivaciòn y moral alta.

# 

Una vez que las empresas han identificado, cuáles son sus necesidades, y además, han determinado lo que se les puede ofrecer a los potenciales candidatos, proceden a la publicación de sus anuncios, en diversos portales Web, como los expresados a continuación:

## 3.3 Portales Web que proporcionan la búsqueda de empleo

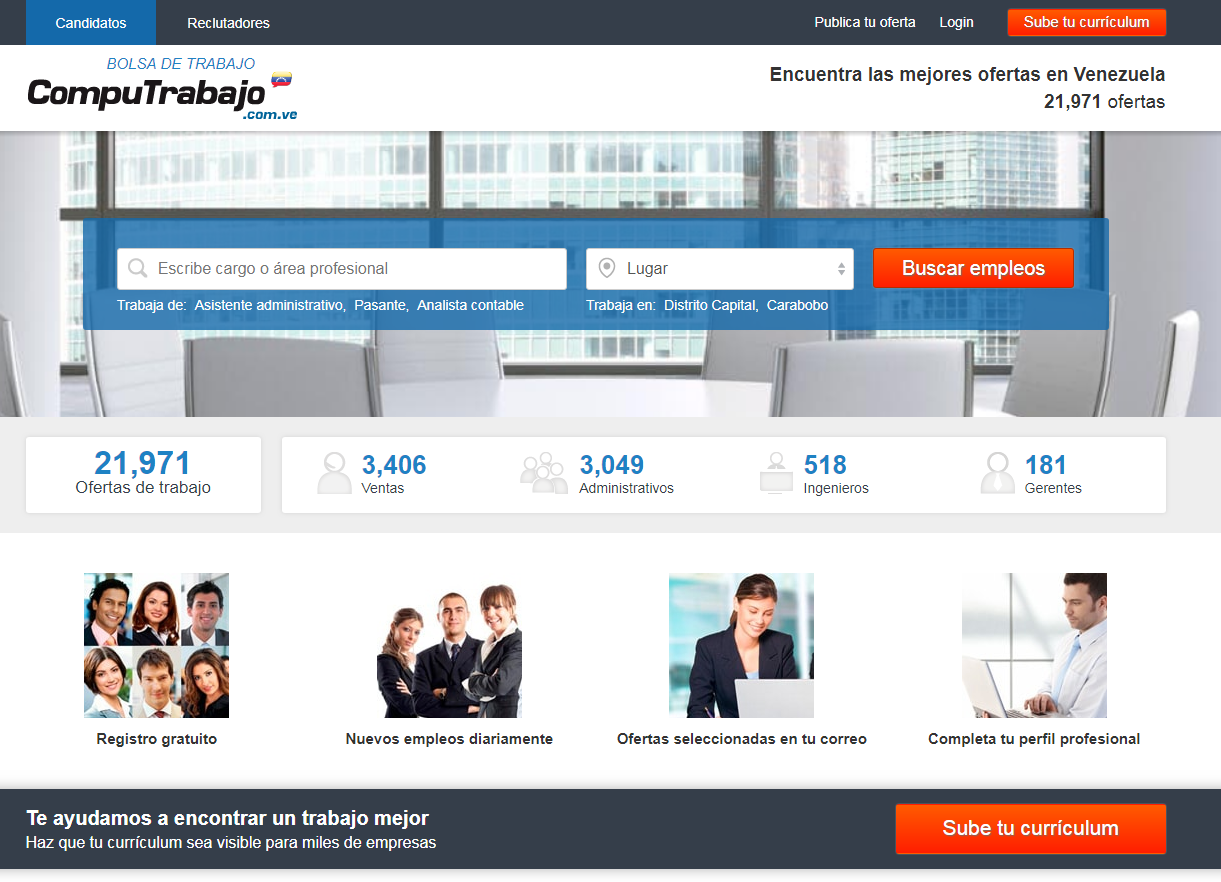
Estos ejemplos en especial son válidos para el territorio Venezolano, y poseen buena reputación, a continuación una breve descripción de cada una, basándose en su reputación y algunas reseñas.



### Figura 1 - Interfaz del sitio web bumeran.com

## 3.3.1 Bumeran

Una opción donde cargar tu Currículum Vitae online y visualizar ofertas de empleo por cargo o provincia (entre otros filtros) es Búmeran. Con una interfaz bastante sencilla, solo tienes que marcar la opción de si prefieres cargar tu CV o si eres una empresa para comenzar a utilizarlo. Además de las ofertas normales, también puedes buscar por filtros como “part-time”, “full-time” o “pasantías”, entre otros.



### Figura 2 - Interfaz del sitio web computrabajo.com

## 3.3.2 CompuTrabajo

Desde su página web CompuTrabajo se denomina como “la mayor red de trabajo de habla hispana”, y lo cierto es que cuenta con presencia en varios países. Puedes registrarte y buscar empleo por localización, cargos profesionales y categorías; además de que te llegarán las ofertas más ajustadas a tu perfil a tu propio correo electrónico. Con más de 12.000 ofertas de trabajo en Venezuela, el registro en esta web es completamente gratuito.

****

### Figura 3 - Interfaz del sitio web empleate.com

## 3.3.3 Empléate

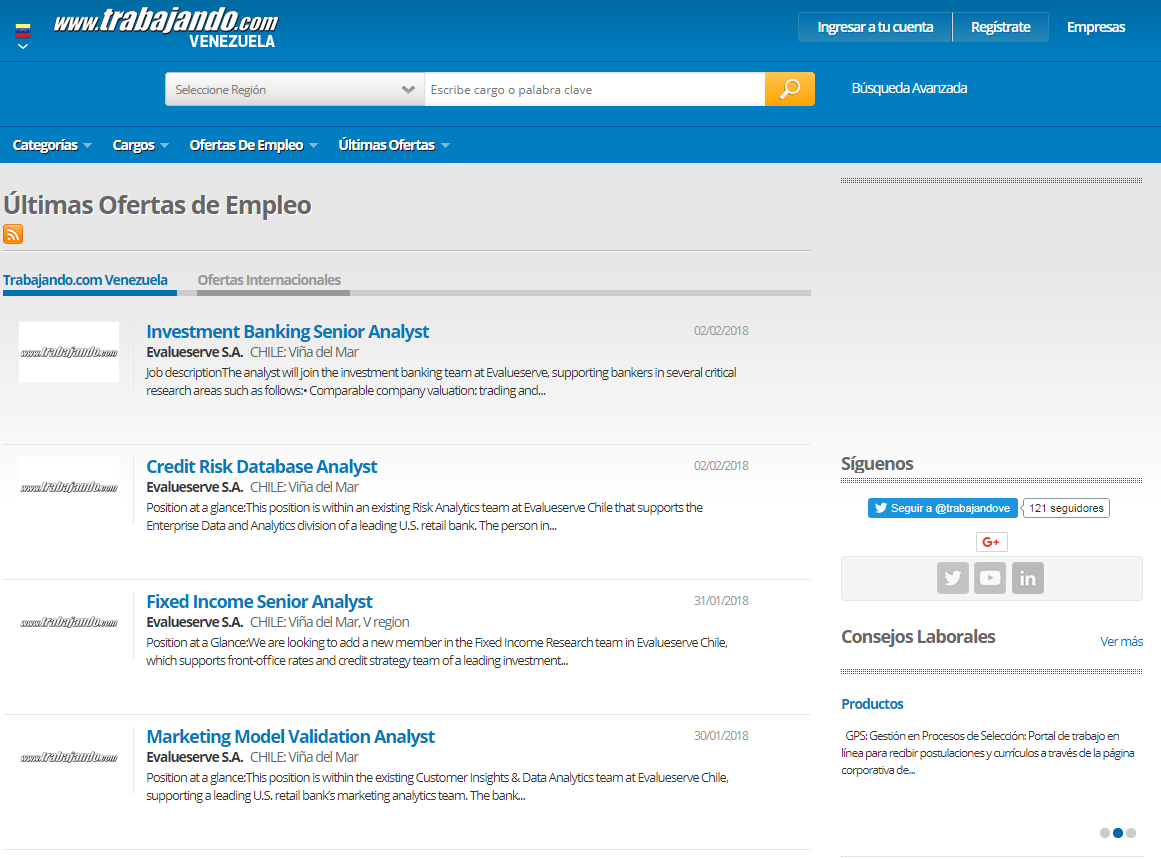
Con más de 2.000.000 de usuarios y 80.000 empresas suscritas, esta web cuenta con miles de ofertas de empleo que se actualizan todo el tiempo. Regístrate en la plataforma y busca tu próximo trabajo por palabras clave, ciudad, cargo, estado y más categorías. Además de la posibilidad de buscar trabajo, Empléate cuenta con otros recursos como la capacitación a través de cursos online de pago.

****

### Figura 4 - Interfaz del sitio web buscojobs.com

## 3.3.4 BuscoJobs

Con presencia en más de veinte países, otra de las webs de búsqueda de empleo líderes en su rubro es BuscoJobs. Aquí también puedes buscar trabajo según las categorías de cargo o área profesional y ciudad, además de ofrecer otras interesantes herramientas que te ayudarán a posicionarte entre las empresas como el “Test de perfil laboral”.

****

### Figura 5 - Interfaz del sitio web trabajando.com

## 3.3.5 Trabajando.com

Con presencia en 10 países de habla hispana, la misión de Trabajando.com su moto es “Dar mayor eficiencia, flexibilidad y transparencia al mercado laboral, generando día a día más oportunidades de trabajo para las personas y mejores candidatos para las empresas.”. a pesar de ser una página de diseño sencillo, es funcional y ampliamente usado como plataforma de ayuda.

## 

## 3.4 Inferencias sobre los antecedentes

Todos los antecedentes tienen maneras de crear un perfil del usuario que posteriormente pueden usar como curriculum interno dentro del portal, este punto es muy importante ya que da al portal datos con los que evaluar los diferentes perfiles profesionales de las personas que usan la plataforma, esta característica es clave para el desarrollo de sistemas de búsqueda de personal, así como también la particularidad de ser heterogéneos en su propósito y divididas según el rol con que se desea usar la plataforma, si como persona o como empresa, por esta particularidad las empresas también tienen mecanismos para describir sus necesidades y que clase de profesionales necesitan para llenar esos puestos. Ambos mecanismos son tan claves para este tipo de portales que incluso en nuestro trabajo debemos incluir estas dos características, lo que diferencia a las páginas más eficientes, es el uso que hacen de tales repositorios de información, igual que la usabilidad que prestan a los usuarios para hacer las tareas que ofrece el sistema lo menos tediosas posibles y así estimular el crecimiento de los datos, nuestro aporte específico viene del hecho de querer filtrar un poco la cantidad de aspirantes para ciertos puestos, haciendo preguntas específicas del dominio del puesto al que se están postulando y que son configurables por el organismo que publicó la oferta de trabajo.

Las búsquedas, filtros, publicar propuestas laborales, etc. son tan comunes para estos portales y tan centrales a su funcionamiento que es difícil describirlos como puntos fuertes para alguno de estos portales en específico. Lo realmente notable es los datos y análisis que hacen con la información de los usuarios registrados.

# Capítulo 4

## Proceso de contratación de personal en las organizaciones.

En este capítulo se explica el proceso de contratación de personal que son aplicadas por las organizaciones de forma detallada incluyendo ejemplos de empresas.

## 4.1 Organización

Una organización es un conjunto de personas y tareas estructuradas administrativamente para lograr metas u objetivos. Para las organizaciones, una de las prioridades es establecer fuentes de suministros de recursos humanos para ejecutar las funciones requeridas por la organización, por ello, las fuentes de recursos humanos se denominan fuentes de reclutamiento ya que una de sus funciones es representar los objetivos los cuales determinarán las técnicas de reclutamiento que se van a aplicar.

## 4.2 Proceso de Contratación

El proceso de contratación de personal en las organizaciones inicia con la identificación de los requerimientos necesarios que deben satisfacer los candidatos para que ocupen los puestos eficientemente denominado Perfil del Puesto. Este perfil se define elaborando profesiogramas, el cual es un documento técnico que organiza la relación, actuación y dependencia desde el punto de vista de gestión de talento humano, seguridad ocupacional y salud laboral, donde se representan los requisitos para el puesto de trabajo y se señalan de forma cuantitativa y cualitativa.

Posterior a la definición de los requisitos se procede a realizar el reclutamiento, donde se identifica, selecciona y se mantiene las fuentes por el cual los candidatos pueden encontrar información sobre la organización, el puesto vacante, los requisitos y aptitudes para que éste pueda aplicar al puesto. Este reclutamiento puede ser por medio del reclutamiento interno, donde la organización intenta llenar las vacantes reubicando, ya sea ascendiendo o transfiriendo a los empleados que ya se encuentren en la organización; o por medio del reclutamiento externo donde la organización intenta llenar las vacantes utilizando las técnicas de reclutamiento como lo son folletos, carteles, contactando universidades, conferencias, publicación en revistas, etc.

Luego, se procede a la fase de selección, el cual consiste en conocer de los candidatos potenciales, sus aptitudes y cualidades con el fin de asignarle el puesto vacante acorde a sus características o rechazar al candidato si no posee las características necesarias.

## 4.2 Proceso de selección

La fase de selección inicia principalmente cuando el departamento de recursos humanos recibe los curriculums de los candidatos potenciales y estas son analizados para determinar las características y aptitudes en base a nivel de educación, experiencias laborales, cursos, títulos adquiridos, referencias laborales, entre otros.

Una vez seleccionado los candidatos cuyas aptitudes correspondan al perfil del puesto de la organización se procede a las entrevistas, donde por medio de una comunicación interpersonal (Entrevistador-entrevistado) permite al reclutador obtener información valiosa sobre el candidato, a través de preguntas, demostraciones, simulaciones o cualquier técnica aplicada por la organización que permita categorizar y evaluar la idoneidad del candidato para el puesto de trabajo.

Posterior a la entrevista, se realizan pruebas psicológicas a los candidatos que continúan en el proceso, estas pruebas se dividen en: Psicométricos donde se evalúan las cualidades psíquicas del individuo como medición de inteligencia (comparación de la edad mental con la edad cronológica), rasgos de personalidad (reacción ante distintas situaciones), y rasgos proyectivos (aspiraciones futuras); de Personalidad donde se evalúan las principales características del carácter del candidato para determinar factores como la adaptación a los cambios, capacidades de iniciativa, reacción ante situaciones de presión, motivación, estabilidad emocional, etc; Psicotécnicos donde se evalúan las capacidades y aptitudes intelectuales del candidato en relación al puesto ofertado por la organización, aptitudes como la memoria, percepción, capacidad de comunicarse, capacidad numérica, etc.

A su vez, dependiendo del tipo de vacante que se está ofertando, se realizan pruebas de capacidad, donde se evalúa la capacidad intelectual por medio de pruebas técnicas, donde el candidato puede demostrar su experiencia y conocimientos de un tópico, prueba con simulaciones para ejecutar sus funciones ante ciertas situaciones o acontecimientos que pudiesen ocurrir al ocupar el cargo.

Finalmente, aquel o aquellos candidatos que fueron seleccionados se les hace entrega de una propuesta salarial, esta propuesta realizada por la organización presenta el salario, beneficios sociales, seguros o cualquier otro tipo de beneficios hecha habitualmente por escrito con la intención de incorporar al candidato a la organización. En este punto el candidato analiza la propuesta y evalúa las condiciones para determinar si la acepta o opta por retirarse de la selección al no cumplir con su expectativa.

## 4.3 Procesos de reclutamiento de empresas nacionales

Esta información fue obtenida principalmente utilizando la metodología focus groups que consiste en realizar preguntas sencillas a empleados actuales de dichas empresas, dichas respuestas a pesar de no ser detalladas, sirven como base para entender los modelos de reclutamientos utilizados, dejando en casos entrever su apoyo en tecnología o por el contrario, la falta de herramientas tecnológicas en las mismas, todo esto sin embargo, dando un puesto justo a la intervención humana que muchos de estos procesos conllevan y que son irremplazables.

## 4.3.1 Farmatodo

Farmatodo es una empresa pionera en implantación y desarrollo del concepto de farmacias de autoservicio en Venezuela , dedicada a comercialización directa de medicamentos, artículos de cuidado personal, de uso diario en el hogar, belleza y alimentos. Iniciaron en 1918, originalmente como una droguería en el estado lara, reconocidos a nivel nacional.

En su página web puedes encontrar los enlaces que te llevan a su pagina de busqueda de empleo, inicialmente solo te piden que envies tu curriculum a un correo electrónico que especifican en su página de reclutamiento, al cabo de unos días te responderán con instrucciones para asistir a una entrevista en un centro de reclutamiento, usualmente un hotel aledaño y conocido de la región a la que aplicaste, la empresa no hace una entrevista personalizada al principio, simplemente asisten a una charla con otros aspirantes en la cadena de farmacias donde se realiza una charla con información relevante sobre la empresa para todos los aspirantes y a continuación, dividirán los participantes en grupos de pocas personas, usualmente 4 o 5, para aplicarles exámenes de resolución de problemas básicos. Al finalizar dichos exámenes, aquellos quienes obtengan la puntuación mínima son entrevistados de forma individual por turnos, para hacerle preguntas personales y preguntas hipotéticas para medir las reacciones del aspirante a situaciones de stress y medir su capacidad de resolución de problemas

Al finalizar esa batería de pruebas, los aspirantes son despedidos de las instalaciones y deben esperar a los resultados del proceso de selección para determinar si aprobaron o no para el cargo en el que estaban interesados, de ser así, la empresa abre un contrato con el aspirante, sino, se hacen contraofertas según los resultados obtenidos, dependiendo del grado de educación del aspirante se pueden contar con más opciones, aunque se puede saltar parte de este proceso por medio de recomendaciones de personas con cargos gerenciales dentro de la empresa.

## 4.3.2 Sudeban

La superintendencia de bancos es un organismo del Estado que se encarga de regular y supervisar a las Instituciones del Sector Bancario, con un talento humano motivado y comprometido, a través de la aplicación de las mejores prácticas nacionales e internacionales, que contribuyan con la estabilidad del Sector y el Desarrollo Nacional. Esta descripción según se puede encontrar en su página web, es la que se encuentra en la misión de esta institución.

Este organismo usa medios de reclutamiento bastante ortodoxos, por lo general, tercerizan proyectos a empresas pequeñas para solucionar necesidades de la institución, de llevarse a cabo un proyecto exitosamente, se toman personas de tales empresas y son recomendados a los supervisores del proyecto dentro de sudeban, estos son llamados por recursos humanos de sudeban,donde los aspirantes entregan sus curriculums y son entrevistados para medir sus expectativas, al terminar este proceso, junto con las las recomendaciones de los jefes de proyecto, los gestores del proyecto deciden si estos individuos son útiles para otros proyectos que la institución tiene planificados, de ser así estos individuos son contratados por cumplir con las expectativas siendo en general un reclutamiento a base de recomendación.

El segundo método muy común de ingreso es a través de su programa de pasantías, donde se hace trabajo por un período de prueba y de ser exitoso puede quedar recomendado por el encargado del proyecto dentro de Sudeban. Finalmente, el tercer método consiste en entregar un curriculum en las oficinas de Sudeban, donde será evaluado por las personas de recursos humanos a ser aprobado, posteriormente será enviado a un jefe de división del organismo y este decidirá si proceder con la contratación. Este procedimiento por lo general tiene un tiempo de espera significativo ya que estos currículos son archivados, evaluados y pasan a una lista de espera de aspirantes antes de ser vistos y evaluados para proyectos que se presenten en la institución.

## 4.3.3 Cantv

La Compañía Anónima Nacional Teléfonos de Venezuela (Cantv), ente adscrito al Ministerio del Poder Popular para Ciencia, Tecnología e Innovación, junto a sus filiales Movilnet y Caveguías, es la primera empresa de telecomunicaciones en Venezuela que tiene como objetivo fundamental fomentar la inclusión social y la disminución de la brecha al acceso de tecnologías digitales, facilitando así el alcance de todos a los servicios de telecomunicaciones. Esto, según lo describe su portal web, es lo que lo posiciona como una de las empresas más reconocidas en el territorio nacional y controla virtualmente el monopolio de las telecomunicaciones en el país.

Su proceso de reclutamiento comienza desde su página web, donde cada aspirante envía su currículum al correo electrónico descrito en dicha página y posteriormente, se contacta a cada aspirante para realizar un conjunto de pruebas de coeficiente intelectual y psicológicas dependiendo de tu perfil profesional y de la vacante a la que el aspirante se postule.

Aquellos aspirantes que pasen las pruebas son contactados nuevamente para proceder a realizar las entrevistas donde se discuten los honorarios profesionales y describir las responsabilidades del puesto disponible, así como la expectativa salarial, personas bajo tu cuidado, todos la información que necesaria para tomar la decisión y finalizar la contratación.

## 4.3.4 Strix Technologies

Esta empresa de tecnología, está establecida en venezuela hace más de 15 años, ofrece soluciones de software, hardware y consultoría tecnológica a empresas nacionales o internacionales que no requieran de la presencia del personal y estén dispuestas a trabajar de manera remota, actualmente trabaja con empresas en Puerto Rico, España, Colombia y en el interior de Venezuela. A pesar de ser una empresa pequeña, se mantienen en constante búsqueda de personal para realizar proyectos nuevos y trabajar en proyectos ya existentes.

Su proceso de reclutamiento se realiza por medio de recomendación de los empleados que actualmente trabajan para la empresa o por medio de contacto por su página web. Se llevan a cabo dos entrevistas el mismo dia al contactar al aspirante; una personal donde se discuten los proyectos y la visión de la empresa o las necesidades de los diferentes proyectos para determinar el interés del aspirante de emprender un proyecto nuevo o integrarse a uno existente, la segunda donde se realiza una evaluación técnica donde se supervisa la capacidad del aspirante de solucionar problemas y evaluar su proceso de pensamiento tomando en consideración su velocidad, comprensión y su perfil técnico para determinar si es el adecuado o si necesita afinar alguna habilidad. después de dichas pruebas, si el aspirante fue exitoso, se discuten sus roles según su perfil técnico y se prepara un contrato, sea freelance o a tiempo completo según las preferencias del aspirante. De no contarse con medios para evaluar al aspirante, se contrata un profesional que dé el visto bueno para decidir si las habilidades están acorde a lo descrito en su perfil profesional,curriculum, etc.

Cabe destacar que los anteriores procesos en su mayoría son a poca escala y solo pueden aplicarse a un número reducido de personas a la vez, o incluso de manera personal únicamente, es por ello que nos vimos forzados evaluar empresas internacionales también.

## 4.4 Procesos de reclutamiento de empresas internacionales

A continuación se detalla cómo las empresas internacionales realizan todo el proceso de reclutamiento y selección.

## 4.4.1 Bairesdev

Bairesdev es una empresa que ofrece servicios tecnológicos que emplea el Top 1% de Latino América que se especializan en el desarrollo de software, outsourcing de software, pruebas unitarias, mantenimiento de software y capacitación para personal de Tecnología.

Esta empresa se basa en el Top 1%, esto es, seleccionan y reclutan a sus vacantes del mejor talento a través del siguiente proceso de selección y reclutamiento detallado a continuación:

El proceso inicia a través de un conjunto de algoritmos únicos cuya función es de rechazar o asignar los aspirantes a vacantes abiertas o inclusiva a futuras vacantes que estarán disponible en un determinado lapso de tiempo. Para aquellos ingenieros que fueron seleccionados, el sistema envía, de forma automática un conjunto variado de pruebas (desde ocho hasta doce pruebas) dependiendo de la posición que el aspirante espera tomar, además de los exámenes de coeficiente intelectual e inglés.

Una vez los aspirantes culminen la evaluación el sistema asigna una puntuación acorde a las respuestas enviadas y aquellos aspirantes que superen el límite de la posición o rol el equipo de Recursos Humanos procede a realizar una serie de entrevistas para evaluar en el aspirante su experiencias, habilidades de comunicación, compromiso, personalidad y cultura. Aún quienes superen las entrevistas deben realizar una serie de evaluaciones escritas, que a diferencia de las evaluaciones preliminares, no solo se evalúan los resultados sino las preguntas realizadas, la manera en que se plantea la solución y el nivel de detalle en cómo se resuelve cada uno de los ejercicios planteados.

Posterior a las evaluaciones escritas, aquellos aspirantes que superan el límite de la vacante o rol se procede a realizar la entrevista técnica, en este punto se realizan preguntas muy específicas al aspirante con la finalidad de evaluar su capacidad de generar soluciones en tiempo real. Aquellos que finalizan dichas entrevistas de forma exitosa son contratados por la empresa Bairesdev.

## 4.4.2 Crossover

Crossover es una empresa que ofrece servicios para la construcción y administración de los mejores equipos para el negocio, por medio de una selección rígida del mejor talento disponible para el negocio. Este proceso de selección filtra el gran flujo de aspirantes con el fin de seleccionar aquel más apto para el rol. Dicho proceso de selección se detalla a continuación:

Fundamentalmente, el proceso inicia cuando el aspirante realiza su postulación en la aplicación, una vez sube su curriculum, el algoritmo selecciona un conjunto de evaluaciones o pruebas realizados para evaluar los fundamentos y la capacidad de resolución de problemas antes distintos escenarios presentados.

Aquellos aspirantes que completen las pruebas de forma satisfactoria pasan a un proceso de evaluación a través de una simulación de un escenario real de trabajo con el fin de evaluar las capacidades técnicas, habilidades de comunicación, manejo de la situación ante escenarios de alto estrés, capacidad de análisis, creatividad (factor que Crossover considera indispensable para la carrera) y manejo del tiempo. Posterior a ello se procede a realizar las entrevistas por un agente de Recursos Humanos, con el fin de obtener las experiencias, éxitos, fallos y qué aprendió de ellos, compromiso y cultura. Para aquellos aspirantes seleccionados se procede a realizar el proceso de verificación de los antecedentes y contactar referencias personales y empresariales con el fin de obtener desde otro punto de vista, las cualidades que encontraron en el aspirante.

El proceso es extenso, tomando desde una a tres semanas, sin embargo, la ventaja de realizar el proceso en Crossover es que permiten realizar cada etapa del proceso según la disponibilidad del aspirante, pudiendo continuarlo a cualquier hora del día y en cualquier día de la semana según la disponibilidad del aspirante.

## 4.4.3 Google

Google es una empresa mundialmente conocida fundada el septiembre del año 1998, que, a pesar de ser conocido mayormente por su buscador, la empresa también se enfoca en el área de servicios relacionados con el Internet (Google Earth, Google Analytics, etc.), desarrollo de software, dispositivos electrónicos.

Google afirma que el talento es más apreciado que la experiencia, es decir, Google tiene una historia de buscar los mejores graduados en distintas universidades y asigna en ellos responsabilidades de suma importancia desde el principio sin importar la experiencia laboral, esto es para determinar el valor y compromiso del aspirante ya que es más fácil enseñar experiencias, que cultivar talento.

El proceso cuando el aspirante realiza su postulación en el sitio Web y se realiza un cierto número de filtros sobre todos los aspirantes buscando cuatro características esenciales para el puesto, estas son:

* Competencias funcionales, en el cual se analizan las capacidades del aspirante y el encaje con el rol que se dispone a ocupar.
* Cultura o “Googlyness”, en el cual se analiza el encaje del aspirante en el ambiente de trabajo de Google, descrita como informal, ameno y muy comunicativa.
* Liderazgo, en el cual analizan la capacidad del aspirante de asumir las responsabilidades asociados al rol y guiar a su equipo de trabajo con el fin de alcanzar las metas.
* Habilidades Cognitivas, el cual analizan cómo el aspirante piensa y analiza los distintos escenarios propuestos y la creatividad y detalle de cómo los resuelve con el fin de proponer una solución.

Una vez realizado el filtro se procede a realizar las entrevistas a los candidatos seleccionados, sin embargo, a diferencia a lo que se observa en la mayoría de las empresas la realiza un comité de selección, es decir, las entrevistas son realizados por un grupo extenso de profesionales y a su vez su posible manager, compañeros de trabajo e inclusive los subordinados. Una vez realizado la entrevista el comité se encarga de compactar toda la información obtenida del aspirante y proceden a realizar una decisión.

Según los datos estadísticos proporcionados por Google, los aspirantes que son seleccionados y se entrega la oferta de trabajo,no cuenta con un período de prueba, esto se debe a que el proceso riguroso que Google asegura que el candidato no se retirará de la empresa.

**4.4.4 Facebook**

Facebook es una Red Social mundialmente conocida cuya visión es facilitar el conexión entre las personas y compartir recursos multimedia a través del mismo. Esta fue creada por Mark Zuckerberg en el año 2004 y al día de hoy es la Red Social más grande del mundo con más de un billón de usuarios a nivel mundial.

Al igual que las empresas anteriores, Facebook se enfoca en reclutar personal que tienen una inclinación a realizar acciones, creatividad, y análisis oportuna ante situaciones que se presenten en el día a día y la forma en que propone las soluciones.

El proceso de reclutamiento inicia por medio del sitio Web en el cual cada aspirante que desee ocupar una vacante puede subir su curriculum para la revisión y posterior a ello se realizan una serie de filtros buscando las características mencionadas anteriormente y una vez seleccionados los aspirantes se procede a realizar una serie de entrevistas organizadas de la siguiente manera:

La primera entrevista se realiza por teléfono con el reclutador quien determina la experiencia profesional del aspirante y la pasión que tiene por pertenecer a la empresa. La segunda entrevista se realiza igualmente por teléfono, sin embargo, a diferencia del primero, esta busca identificar las habilidades técnicas y de comunicación. Esta última lo realiza un empleado de Facebook que tenga el cargo en el cual el candidato está aplicando.

Una vez finalizada las primeras dos entrevistas, se procede a realizar entrevistas cara a cara el cual incluye un tour por las oficinas de Facebook para que el aspirante pueda identificar la cultura de la empresa y ambiente de trabajo.

Posterior al tour, se procede a ser un número indeterminado de entrevistas personales, el número varía según el rol a ocupar, por ejemplo, para roles de desarrollo de software se realizan evaluaciones de código donde se determina qué tan rápido, y preciso puede el aspirante resolver problemas ante escenarios inesperados. Finalmente el/los aspirantes seleccionados pasan a recibir una oferta para ocupar el rol al cual se postularon.

**4.4.5 Microsoft**

Microsoft es una compañía tecnológica que desarrolla, diseña, licencia y provee soporte para computadores personales, servidores, dispositivos electrónicos y servicios tecnológicos. Conocido mundialmente por sus productos más utilizados como lo son el sistema operativo Microsoft Windows, el conjunto de aplicaciones de Microsoft Office (Excel, Power Point, Word, entre otros) y los navegadores de Internet, Internet Explorer y Edge.

El proceso de selección inicia a través de la página Web, el aspirante puede realizar seguimiento del estatus de su postulación durante todo el proceso de selección a través de la Página Web, para determinar cómo se estructura el proceso, fue necesario tomar experiencias personales de distintas personas que se postularon, dicho análisis permitió sintetizar el proceso de la siguiente manera:

En general el proceso se realiza por teléfono o por Skype, donde el reclutador realiza preguntas técnicas para determinar la experiencia profesional del postulante, y para roles que requieran conocimientos de desarrollo de software realiza un conjunto de evaluaciones en el cual se presenta un escenario y el aspirante debe desarrollar código listo para producción en un corto período de tiempo mientras se detalla el razonamiento de la solución propuesta. Esto se repite con los aspirantes que van siendo seleccionados aumentando la complejidad y el nivel de conocimiento requerido hasta quedar con un número reducido de aspirantes, se procede a realizar la entrevista con el personal de Recursos Humanos para determinar el encaje cultural del aspirante y la pasión del aspirante por la empresa.

# Capítulo 5

## Framework de Desarrollo Web

En el presente capítulo se explican los conceptos y objetivos de los frameworks de desarrollo Web para el desarrollo de las aplicaciones.

## 5.1 Framework

Un framework es una estructura de software que se componen de librerías y componentes personalizables para facilitar el desarrollo de una aplicación o componente de software.

## 5.2 Framework de Desarrollo Web

Análogamente, un framework de desarrollo Web es un tipo de framework cuyos componentes apoyan al desarrollo de sitios, aplicaciones y servicios webs dinámicos por medio de librerías que ofrecen funcionalidades estándares como la conexión con base de datos y mecanismos para facilitar la integración con otras herramientas para implementar las capas de negocio y presentación.

## 5.3 Componentes de un Framework de Desarrollo Web

Todo framework incluye un conjunto de componentes que facilitan el desarrollo de características y funcionalidades a la aplicación web, detalladas a continuación.

## 5.3.1 Librerías

Las librerías son una colección de recursos utilizados para el desarrollo de aplicaciones donde se incluye configuración de datos, documentación, código preescrito y rutinas, clases o tipos de datos.

## 5.3.2 APIs (Application Program Interface)

Un API o interfaz de programación de aplicaciones es un conjunto de rutinas, funciones y procedimientos útiles para conectar dos software entre sí para realizar la comunicación con la fuente de datos (back-end) con los datos a mostrar (front-end) ya sea en formato XML o JSON.

## 5.3.3 Scaffolding

Scaffolding es una técnica de programación soportado por los frameworks MVC (Modelo, Vista, Controlador) donde se especifica como y cuando se tiene acceso a la fuente de datos.

## 5.3.4 AJAX

La mayoría de los frameworks de desarrollo web incorporan la tecnología Asynchronous JavaScript And XML, mejor conocido como AJAX, el cual permite a los sitios web actualizarse de forma asíncrona realizando un intercambio de datos con el servidor sin la necesidad de recargar la página.



### Figura 6 - Ejemplo de una llamada AJAX

## 5.3.5 Caching

El proceso de Caching consiste en el almacenamiento de datos en la memoria caché, la memoria cache es un área de almacenamiento temporal que guarda archivos cargados de sitios webs de forma que al momento de abrir el mismo sitio nuevamente, esta cargue de forma más rápida incrementando el rendimiento.

## 5.3.6 Seguridad

Todo framework de desarrollo web provee mecanismos de seguridad, estos mecanismos buscan resguardar la información de personajes que no tengan acceso a ella, como información privada de los usuarios registrados, de la aplicación u organización en general.

## 5.4 Objetivos Principales de un Framework

* Acelerar los tiempos de desarrollo de software por medio de librerías integradas con funcionalidades ya implementadas y probadas de forma exitosa.
* Facilita la reutilización de código a lo largo del desarrollo.
* Promover a la utilización de buenas prácticas de desarrollo de software, como lo son el uso de patrones de diseños.

# Capítulo 6

## Sistemas manejadores de contenido

En el presente capítulo se explican qué es sistema de manejador de contenido o CMS (Content Management System), como funciona y las ventajas que proveen el uso de dicho sistema.

## 6.1 Sistema Manejador de Contenido

Un sistema manejador de contenido o CMS (Content Management System) es un programa o aplicación alojado en línea que permite administrar, editar, crear y eliminar contenidos en un sitio web, permitiendo al usuario controlar el contenido que se muestra. Los datos o contenidos mostrados en el sitio web se alojan en una o varias bases de datos permitiendo:

* Modificar el contenido sin la necesidad de realizar modificaciones dentro aplicación.
* Facilita la reutilización del contenido en distintas vistas, reduciendo los costos de mantenimiento y tasa de errores.
* Facilita el crecimiento de la aplicación.

Los sistemas manejadores de contenido permiten proveer a usuarios con distintos roles y permisos de cualquier nivel para realizar modificaciones de contenido, datos o información que se muestra dando la ventaja de adaptarse a los nuevos formatos o tecnología de forma eficiente.

## 6.2 Organización y Estructura de un CMS

La información se encuentra organizadas en documentos y archivos, y a su vez, cada documento y archivo se encuentra dividido en uno o varios esquemas, esto permite clasificar la información o contenido que puede ser recuperado con facilidad para aplicarse en una plantilla o vista. La estructura de estos datos permiten que se puedan desarrollar procesos automatizados de mantenimiento de forma periódica o ante eventos que desencadenan el proceso.

Todo el contenido se almacena en un formato nativo que permite a los desarrolladores utilizar herramientas de software para para la creación del contenido y mantenimiento del mismo. Cuando se recupera la información, el proceso de ensamblaje agrupa el contenido del documento o archivo de acuerdo al esquema asociado y lo formatea de acuerdo a la plantilla o vista en la que se muestre.

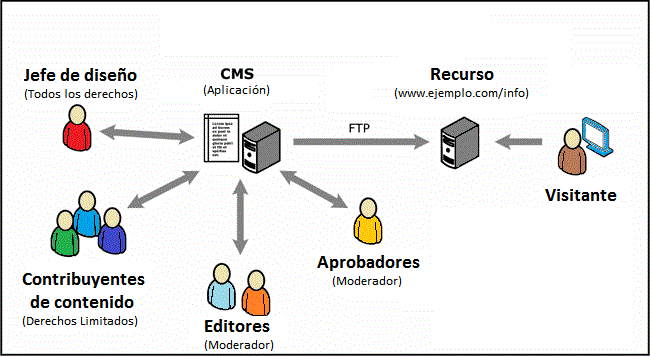
## 6.3 Flujo de Trabajo de un CMS

Los CMS se basan en el uso de flujos de trabajos para la administración de la información. En general todos los CMS incluye flujos de trabajo por defecto, sin embargo, es posible realizar los flujos de forma personalizada de acuerdo a las necesidades de la organización.

Los Flujos de Trabajo nos permiten determinar cómo se estructura el contenido, quién la estructura, como se sincroniza y finalmente cómo fluye el contenido en nuestra aplicación.

## 6.3.1 Flujo de un CMS

En todo CMS existe un Flujo de trabajo fundamental para la administración del contenido, como se muestra en la siguiente imagen.



### Figura 7 - Flujo de trabajo, funcionamiento de un CMS

**6.3.2 Funcionamiento**

El jefe del diseño es quien se encarga de realizar el diseño, estructura, funcionalidades y accesibilidad creando un conjunto de plantillas donde se incluyen campos y regiones editables, características personalizables de navegación y de diseño dentro del sistema manejador de contenido. Debido a esto, el jefe de diseño tiene todos los derechos para la creación, edición y eliminación de los contenidos.

Los contribuyentes de contenido son quienes se encargan de tomar las plantillas generadas por el jefe de diseño e incluyen el contenido o información en ellas para ser mostrado hacia el usuario. Debido a esto, los contribuyentes tienen derechos limitados en los que se incluye la edición y eliminación de elementos contenidos en las plantillas.

Los editores son quienes se encargan de hacer revisiones a al conjunto de contenidos generados por contribuyentes y realizan ediciones tanto estructural como de diseño en caso de ser necesario. Debido a esto, los editores tienen únicamente derechos de moderación para realizar la edición de los elementos contenidos en las plantillas generadas por los contribuyentes.

Los aprobadores son quienes se encargan de realizar la última revisión del contenido y, por medio de los lineamientos dictados por la empresa, se evalúa si dicho contenido es aceptable o no para mostrarse en el sitio web. Debido a esto, los aprobadores tienen derechos de moderación para aceptar o rechazar el contenido generado en las fases anteriores. Una vez que el contenido es diseñado, editado y revisado, cualquier usuario podrá acceder al recurso por medio del URL.

Analizando las desventajas obtenidas en los puntos anteriores en relación al proceso de contratación en diversas empresas, en los portales Web de colocación de empleos, en los sistemas administradores de contenido, se planteó la necesidad de abordar el tema de aplicaciones configurables o parametrizables.

# Capítulo 7

## Aplicaciones configurables o parametrizables

En toda aplicación ya sea simple o compleja como administración de aprendizaje, procesamiento de información, ingeniería de software y otras herramientas similares, es común proporcionar a los usuarios que interactúen con la aplicación la flexibilidad y libertad de realizar modificaciones a nivel de interfaz de usuario, para satisfacer las necesidades de accesibilidad y usabilidad a cada tipo de usuario que ingrese a la aplicación.

Uno de los principales objetivos de un desarrollador al momento de realizar un proyecto es asegurarse que la experiencia sea óptima para cualquier tipo de usuario, realizando la aplicación de forma que sea fácil para los usuarios configurar el producto como lo deseen tomando en cuenta el impacto que la personalización, individualización y configuración específica tendrá en la experiencia final del usuario.

## 7.1 Aplicaciones configurables para organizaciones o clientes

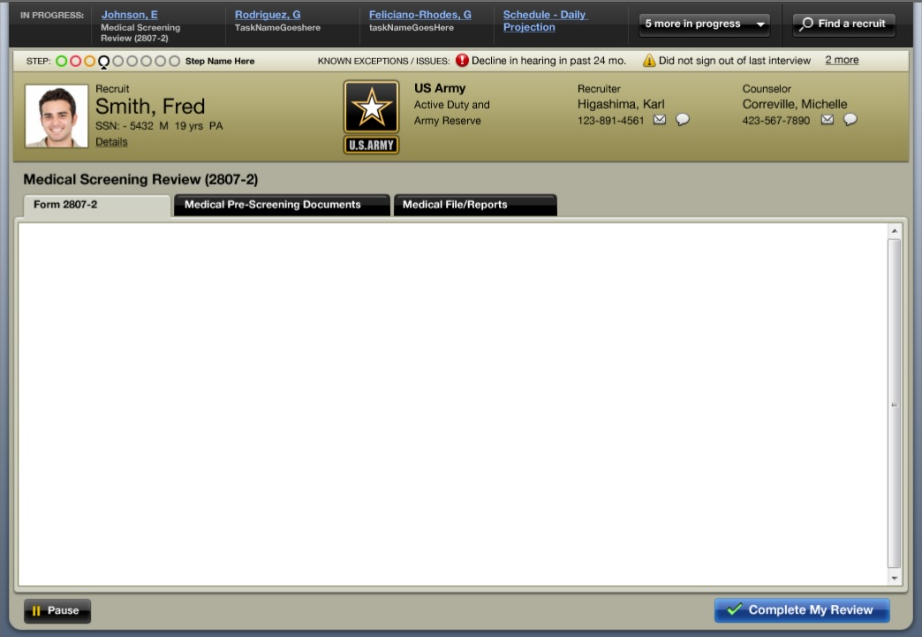
Es importante en una aplicación configurable en el que una organización o cliente requiera de realizar cambios o personalizar la aplicación tenga acceso a los siguientes aspectos:

## 7.1.1 Apariencia

En cuanto a apariencia, para una organización es importante tener la posibilidad de realizar cambios en la forma que se muestra el contenido basado en datos, atributos de usuario (roles, clientes con mayores beneficios) o dominios, a su vez, las aplicaciones utilizados como servicios permiten el cambio de la marca o logotipo por la de la organización.



### Figura 8 - Ejemplo de Aplicación configurable en apariencia



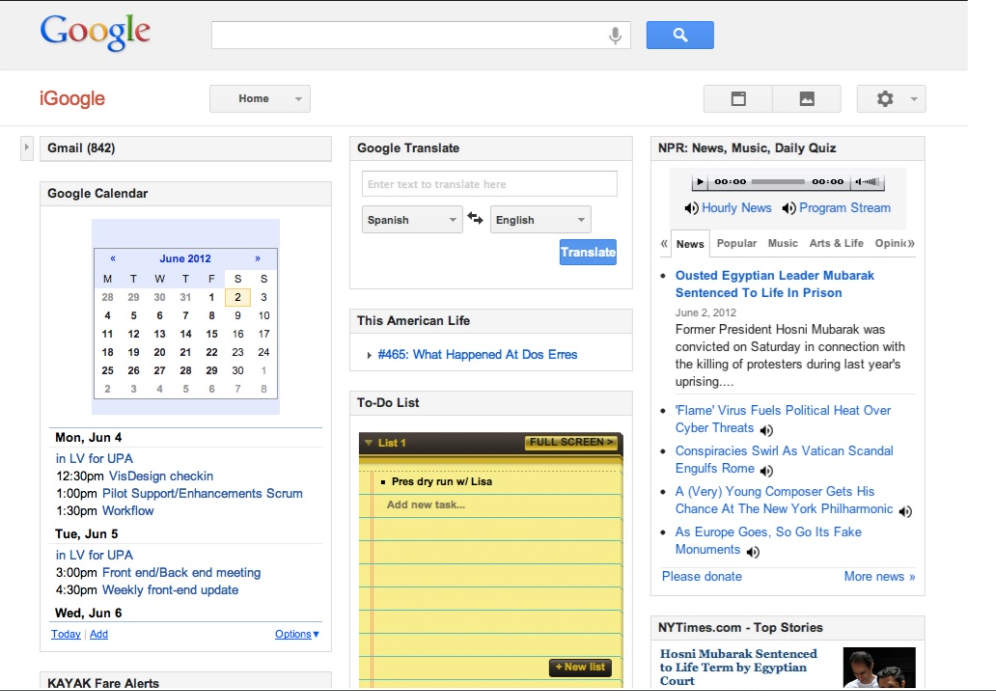
### Figura 9 - Ejemplo de aplicación configurable en apariencia

## 7.1.2 Plantilla o diseño

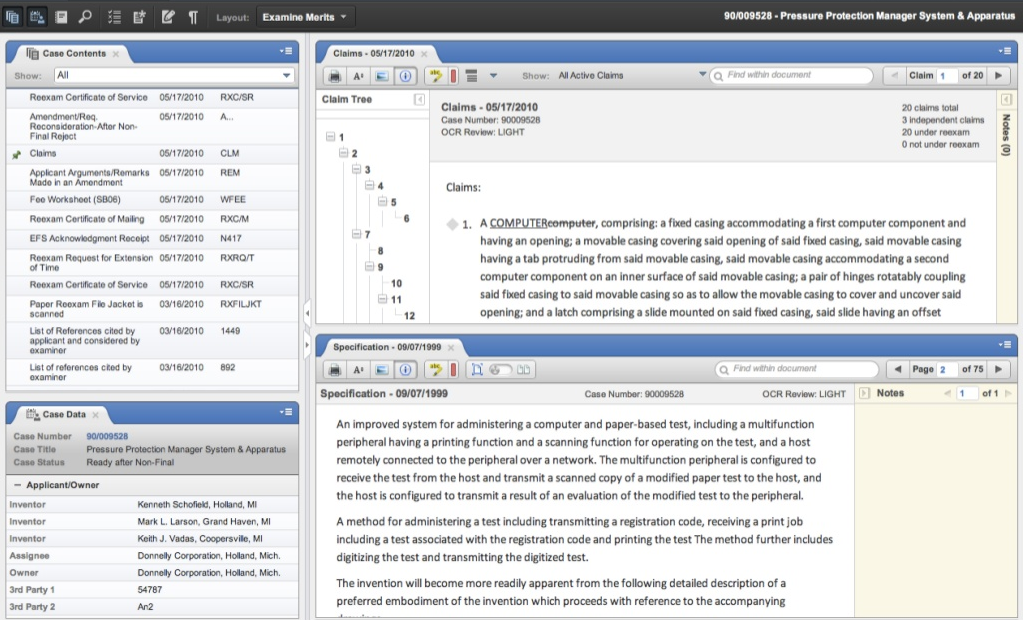
En este aspecto, se dan alternativas al usuario de modificar la posición en que las diferentes funcionalidades o características de la aplicación se muestran, a su vez modificar la forma en que se muestra el contenido ya sea por cantidad de contenido en pantalla, o posición.

Adicionalmente, en este aspecto se tiene la compatibilidad del diseño de la aplicación entre los distintos tamaños de los dispositivos y ambientes de trabajo.

Es importante resaltar que todo cambio realizado por la organización o cliente debe ser guardado entre cada sesión y debe existir la posibilidad de revertir los cambios a una plantilla estándar o una versión anterior.



### Figura 10 - Ejemplo de cómo Google permite distribuir sus características

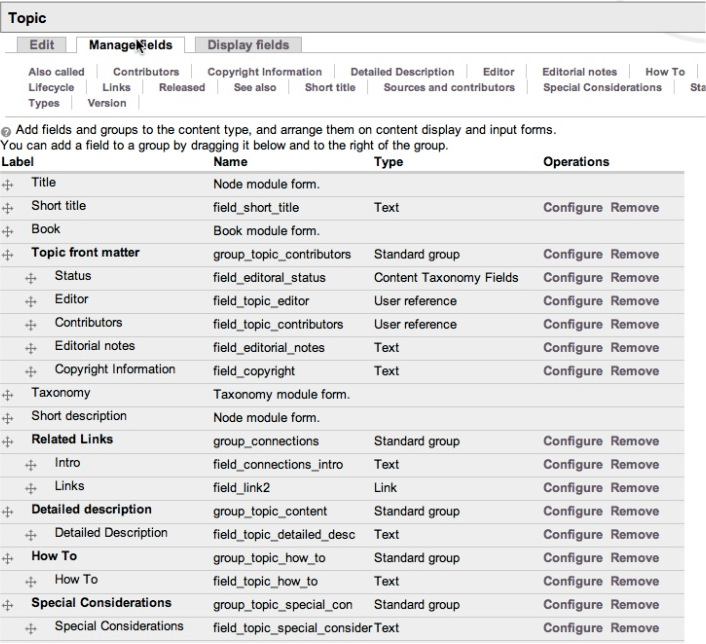


### Figura 11 - Ejemplo de una aplicación cuyo contenido puede ser desplazado

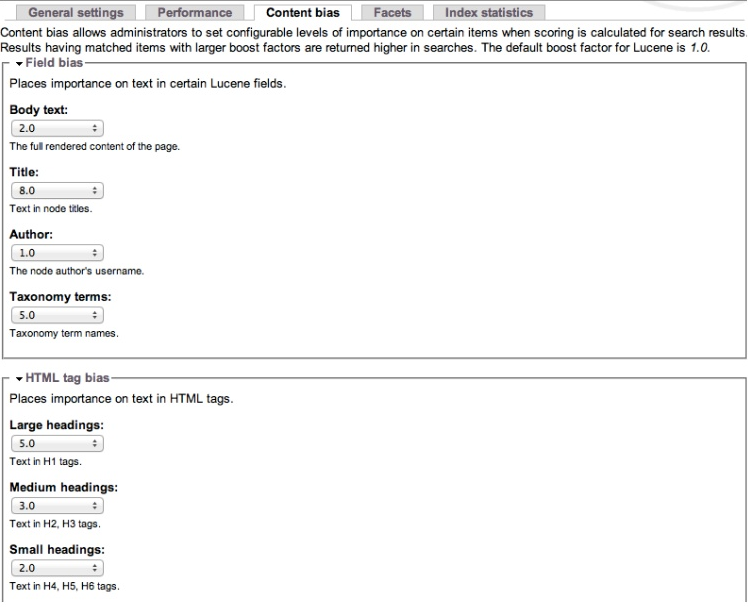
## 7.1.3 Contenido

En este aspecto se permite a la organización generar su propio contenido para ser mostrado en la aplicación. Esto se logra proporcionando una plantilla o interfaz en el que la organización pueda llenar un conjunto de campos u opciones de acuerdo a las necesidades y configurar el lenguaje en el que se muestra el contenido por medio de configuraciones de etiquetas globales .

La complejidad de este aspecto parte de que la interfaz tiene que ser usable para que la organización pueda generar el contenido, esto se logra implementando soporte durante la creación de contenido, como entrenamiento, instrucciones en cada paso durante la creación, tutoriales cortos, y a su vez, realizando todas las validaciones necesarias para garantizar que lo que se esté creando cumpla con lo establecido; teniendo en cuenta de la dependencia lógica de los datos que pueden o no estar disponibles para la organización al momento generar el contenido.



### Figura 12 -Ejemplo configuración de etiquetas al generar un contenido



### Figura 13 - Ejemplo de configuración de prioridad y tamaño de las etiquetas

## 7.1.4 Navegación

Para las organizaciones es importante que las reglas de negocio y las terminologías se vean reflejadas en la navegación a través de la aplicación, por lo que la organización debe tener la posibilidad de dividir a los usuarios por roles.

Los roles de usuario son componentes de una aplicación que definen qué es lo que se puede ver y qué tipo de modificación puede realizar, como borrar, modificar y generar contenidos y permiten asignar acceso a un conjunto de módulos según las funciones del rol.



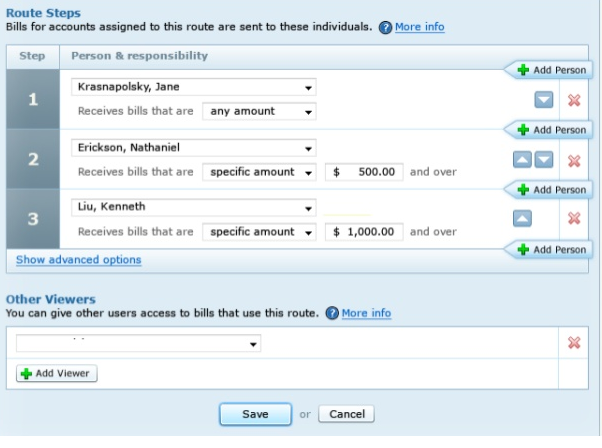
### Figura 14 - Ejemplo de navegación dependiendo del rol de usuario

## 7.1.5 Interacciones

En este aspecto se define el flujo de la información de forma que se pueda configurar como los usuarios, dependiendo del rol asignado, ven la información. Esto se realiza generando flujos de información (parecido a un flujo de trabajo) donde se define qué información se ve, bajo qué condiciones la información será mostrada y en qué orden se muestra.

Además del flujo, también se proveen herramientas para generar mensajes y notificaciones que serán enviados a usuarios suscritos a la aplicación por medio de correo electrónico o mensajes push en el caso de dispositivos móviles.

Finalmente se definen las reglas de negocio, donde se asigna qué información recibe un usuario ante ciertos acontecimientos o eventos que ocurran en la aplicación.



**Figura 15 - Ejemplo de definición de flujo de información**

### 

### Figura 16 - Ejemplo de definición de reglas de negocio

## 7.2 Evaluar cuando una aplicación debe ser configurable

Una vez analizado todos los aspectos, hay que tomar en consideración el beneficio que trae si se desarrolla características que permitan al usuario personalizar la interfaz. Esta evaluación se realiza analizando los costos de implementación versus el beneficio obtenido, el valor que añade la personalización al usuario final de acuerdo al objetivo final de la aplicación y la seguridad de la información que los usuarios pueden manejar al momento de personalizar la aplicación. Para el tópico de este trabajo de Seminario es imprescindible que la aplicación final sea configurable, ya que las organizaciones que ingresen a ella deben tener la posibilidad de generar evaluaciones, información detallada sobre las ofertas de trabajo, inducciones, requerimientos y propuestas salariales.

# Capítulo 8

## Frameworks de desarrollo Web vs Sistemas manejadores de contenido

En este capítulo se analiza las ventajas y desventajas tanto de un framework de desarrollo Web como el de un sistema manejador de contenido (CMS) para el desarrollo de aplicaciones configurables.

## 8.1 Framework de Desarrollo web para el desarrollo de aplicaciones configurables

Un framework es un conjunto de librerías que hacen posible la generación de una aplicación utilizando código creado y probado de forma exitosa, esto nos trae una variedad de ventajas y desventajas detallados a continuación.

## 8.1.1 Ventajas

Por medio de un un framework, es más rápido el desarrollo de la Interfaz de programación de aplicaciones, esto facilita la forma en que se transfiere los datos de la aplicación y el cómo se procesa ya que el framework provee control total sobre los datos y provee compatibilidad con múltiples formatos. Todo esto provee facilidad al momento de generar plantillas configurables por los usuarios, determinando el qué y cómo puede personalizar su interfaz y cómo se maneja el pase de mensajes o información entre el servidor y la aplicación.

Al usar un framework se tiene más flexibilidad al momento de desarrollar las funcionalidades de la aplicación, las librerías contenidas en un framework son adaptables a las necesidades de cada desarrollador y proveen funcionalidades bases que pueden expandirse por medio de complementos o addons desarrollados por el propio desarrollador o proporcionado por la comunidad.

Finalmente un framework provee múltiples ambientes de desarrollo, estos ambientes refieren a dónde se ejecuta una aplicación de software. En el desarrollo de una aplicación, es importante separar los ambientes de desarrollos con el fin de tener control y asegurar la calidad del producto final. Entre los ambientes de desarrollo más utilizados tenemos el ambiente de desarrollo, donde los desarrolladores implementan características o funcionalidades a la aplicación; ambiente de pruebas o QA testing donde un conjunto de tester por medio de pruebas unitarias controladas prueban la aplicación validando la funcionalidad y los estándares de calidad; y finalmente el ambiente de producción donde el usuario final podrá ingresar a la aplicación final e interactuar con ella.

Estos ambientes se pueden generar con facilidad gracias a la compatibilidad integrada de los framework con distintos manejadores de base de datos como MySQL, Postgress, etc. y browsers web como Google Chrome, Firefox, etc.

## 8.1.2 Desventajas

Una de las desventajas de un framework es el hecho de realizar constantes mantenimientos a la aplicación de acuerdo a las actualizaciones y mejoras del framework. Estas actualizaciones traen mejoras de rendimiento y velocidad de respuesta, funciones adicionales añadidas o provee nuevas implementaciones para facilitar el desarrollo de una aplicación, sin embargo, muchos de estas actualizaciones pueden causar que ciertas funciones dejen de funcionar y se tiene que realizar cambios importantes del código ya implementado y probado.

A pesar de la libertad y flexibilidad que otorga un framework al momento de realizar el desarrollo de una aplicación, es importante que el desarrollador tenga un nivel alto de experticia para sacar provecho a todas la propiedades que el framework tiene para ofrecer.

## 8.2 Sistema manejador de contenidos para el desarrollo de aplicaciones configurables

Un sistema manejador de contenido es una aplicación web que se construye sobre una plataforma para administrar contenidos que pueden ser configurables, esto trae una variedad de ventajas y desventajas detalladas a continuación:

## 8.2.1 Ventajas

Generar una aplicación Web utilizando un sistema manejador de contenido es rápido ya que la instalación de las mismas incluye un conjunto de plantillas y temas predefinidos preparados para ser utilizados por el desarrollador, permitiendo definir qué campos son configurables por el usuario final.

Gracias a la comunidad de los diferentes sistemas manejadores de contenidos se pueden descargar e integrar todo tipo de addons y plugin de forma automática fácilmente configurables. El uso de un sistema manejador de contenido no requiere experiencia previa por lo que no es necesario tener experiencia alguna del sistema para usarlo e integrarlo con la aplicación web a desarrollar.

## 8.2.2 Desventajas

Una de las desventajas principales al utilizar un sistema manejador de contenido es la poca escalabilidad de las aplicaciones que se desarrollan en ella. Esto se debe a que las plantillas y plugin integradas en la aplicación no son escalables, es decir, añadir nuevas funcionalidades a un plugin existente es difícil o en algunos casos imposibles de lograr debido a las limitaciones inherentes de cada una de ellas.

Otra desventaja es la seguridad del sistema manejador de contenido, en especial los que son open-source (código abierto), ya que al aumentar la popularidad puede generar interés a ladrones informáticos o hackers para analizar y buscar vulnerabilidades en el sistema. Estas vulnerabilidades pueden ocurrir no solo por el sistema en general, si no por los plugins desarrollado por la comunidad.

## 8.3 Framework de desarrollo Web vs Sistema manejador de contenido para el desarrollo de aplicaciones configurables

Luego de realizar el análisis de cada sistema junto a sus ventajas y desventajas, se pueden tomar varios aspectos que se debe tomar en consideración al momento de desarrollar una aplicación configurable, detallados a continuación.

## 8.3.1 Seguridad

Uno de los factores más importantes a tomar en cuenta a la hora de desarrollar una aplicación configurable es la seguridad de la información que se transmite en ella.

Debido a que los sistemas manejadores de contenidos, en su mayoría, son open-source, los hackers o ladrones informáticos realizan el análisis del código fuente para encontrar vulnerabilidades, por lo que es necesario realizar un monitoreo constante de la versión del CMS ya que es importante realizar actualizaciones y parches a tiempo.

Por otro lado los frameworks tienen su propio código personalizado es mucho más difícil encontrar fallas de seguridad, adicionalmente, los frameworks vienen con funciones integradas como parametrización de consultas SQL para proteger contra ataques de SQL Injection y scripts que permiten el cifrado y protección de datos contra los ataques más comunes.

## 8.3.2 Personalización

La instalación de un sistema manejador de contenido viene con un conjunto de funcionalidades y características que puedes fácilmente integrar en la aplicación web, sin embargo, puede existir limitaciones en dichas características causando complicaciones entre lo que se quiere hacer y lo que permite el CMS, por lo que se debe adaptar las funcionalidades a las necesidades específicas como por ejemplo limitar lo que el usuario puede personalizar en la aplicación.

Con un framework se debe construir todo desde cero y requiere un alto nivel de experticia para implementarlas, sin embargo, permite crear características distintivas y únicas por lo que se puede moldear las funcionalidades para que cubran las expectativas del desarrollador dando libertad para permitir la personalización esperada de la aplicación por parte del usuario final.

## 8.3.3 Flexibilidad

Dependiendo del tipo de aplicación web, puede existir la necesidad de incorporar características o funcionalidades que no son comunes en un desarrollo de aplicación web. En el caso del sistema manejador de contenidos, si el componente no existe en los temas o plugins desarrollados por la comunidad es necesario realizar un desarrollo personalizado, ese desarrollo debe ser resuelto internamente o por medio de freelancing o agencias, incrementando los costos y tiempo de desarrollo. En cambio, un framework puede moldearse a las necesidades del desarrollador, por lo que un framework es más flexible para la integración de funcionalidades o características configurables para la aplicación web.

## 8.3.4 Actualizaciones

Finalmente, los sistema manejadores de contenidos suelen recibir actualizaciones que incluyen mejoras de rendimiento, seguridad, añadido de plantillas, plugins, funcionalidades y características. Dicha actualización se realiza de forma automática por medio de setups liberados por los creadores, facilitando la compatibilidad entre versiones.

En cambio los frameworks de desarrollo web son actualizados con menor regularidad, y sin embargo, luego de cada actualización es importante validar la compatibilidad con versiones anteriores ya que ciertas funciones quedan obsoletas y por lo tanto retiradas del framework, causando que la aplicación deje de funcionar hasta que se realice una refactorización del código que sea compatible con la nueva versión del framework.

# Capítulo 9

# Parte III. Marco metodológico

Para todo desarrollo de un proyecto es esencial planificar la forma en que se van a abordar los tiempos de desarrollo, el personal o recursos necesarios durante cada fase de desarrollo, conjunto de pruebas unitarias a realizar, estándares de calidad a través de metodologías de desarrollo y gestión y planificación de proyectos que se detallan a continuación.

**9.1 Metodología de Desarrollo: Programación Extrema XP**

Extreme Programming por su descripción en ingles (XP) es un marco de desarrollo de software ágil que tiene como objetivo producir software de mayor calidad y una mejor calidad de vida para el equipo de desarrollo. XP es el más específico de los marcos ágiles con respecto a las prácticas de ingeniería apropiadas para el desarrollo de software.

## 9.1.1 Todo el equipo

Todos los contribuyentes a un proyecto de XP se sientan juntos, miembros de un equipo. Este equipo debe incluir un representante comercial, el "Cliente", que proporciona los requisitos, establece las prioridades y dirige el proyecto. Lo mejor es que el Cliente o uno de sus asistentes sea un usuario final real que conozca el dominio y lo que se necesita. El equipo, por supuesto, tendrá programadores. El equipo puede incluir probadores, que ayudan al Cliente a definir las pruebas de aceptación del cliente. Los analistas pueden servir como ayudantes para el Cliente, ayudando a definir los requisitos. Por lo general, hay un entrenador que ayuda al equipo a seguir el camino y facilita el proceso. Puede haber un gerente, proporcionando recursos, manejando la comunicación externa, coordinando actividades. Ninguno de estos roles es necesariamente propiedad exclusiva de un solo individuo: todos los miembros de un equipo de XP contribuyen de cualquier forma que puedan. Los mejores equipos no tienen especialistas, solo colaboradores generales con habilidades especiales.

## 9.1.2 Planificación del esquema de trabajo

La planificación de XP aborda dos preguntas clave en el desarrollo de software: predecir lo que se logrará antes de la fecha de vencimiento y determinar qué hacer a continuación. El énfasis está en dirigir el proyecto, lo que es bastante sencillo, más que en una predicción exacta de lo que se necesitará y cuánto tiempo tomará, lo cual es bastante difícil. Hay dos pasos clave de planificación en XP, que abordan estas dos preguntas:  
  
La planificación de versiones es una práctica en la que el cliente presenta las características deseadas a los programadores, y los programadores estiman su dificultad. Con las estimaciones de costos disponibles, y con el conocimiento de la importancia de las características, el Cliente establece un plan para el proyecto. Los planes de lanzamiento inicial son necesariamente imprecisos: ni las prioridades ni las estimaciones son realmente sólidas, y hasta que el equipo comience a trabajar, no sabremos cuán rápido irán. Sin embargo, incluso el primer plan de lanzamiento es lo suficientemente preciso para la toma de decisiones y los equipos de XP revisan el plan de lanzamiento regularmente.

## 9.1.3 Iteración de planificación

La planificación por iteración es la práctica mediante la cual el equipo recibe instrucciones cada dos semanas. Los equipos de XP crean software en "iteraciones" de dos semanas, entregando software útil en funcionamiento al final de cada iteración. Durante la Planificación de iteración, el Cliente presenta las características deseadas para las próximas dos semanas. Los programadores los dividen en tareas y estiman su costo (en un nivel de detalle más detallado que en la Planificación de versiones). En función de la cantidad de trabajo realizado en la iteración anterior, el equipo se registra para lo que se llevará a cabo en la iteración actual.  
Estos pasos de planificación son muy simples, sin embargo, brindan muy buena información y un excelente control de la dirección en manos del cliente. Cada dos semanas, la cantidad de progreso es completamente visible. No hay "noventa por ciento hecho" en XP: una historia característica se completó, o no fue. Este enfoque en la visibilidad da como resultado una pequeña y agradable paradoja: por un lado, con tanta visibilidad, el Cliente está en posición de cancelar el proyecto si el progreso no es suficiente. Por otro lado, el progreso es tan visible, y la capacidad de decidir qué se hará a continuación es tan completa, que los proyectos de XP tienden a ofrecer más de lo que se necesita, con menos presión y menos estrés

## 9.1.4 Pruebas al cliente

Como parte de la presentación de cada característica deseada, el cliente de XP define una o más pruebas de aceptación automatizadas para mostrar que la función está funcionando. El equipo crea estas pruebas y las usa para demostrarse a sí mismas, y al cliente, que la función se implementa correctamente. La automatización es importante porque con el paso del tiempo, las pruebas manuales se saltan. Eso es como apagar las luces cuando la noche se pone más oscura.  
  
Los mejores equipos de XP tratan las pruebas de sus clientes de la misma manera que lo hacen con las pruebas de programación: una vez que la prueba se ejecuta, el equipo la sigue funcionando correctamente a partir de entonces. Esto significa que el sistema solo mejora, siempre entallando hacia adelante, nunca retrocediendo.

## 9.1.5 Pequeños entregables

Los equipos de XP practican versiones pequeñas de dos maneras importantes:

En primer lugar, el equipo lanza un software en ejecución y probado, que proporciona el valor comercial elegido por el Cliente en cada iteración. El Cliente puede utilizar este software para cualquier propósito, ya sea de evaluación o incluso de lanzamiento a usuarios finales (altamente recomendado). El aspecto más importante es que el software es visible y se entrega al cliente al final de cada iteración. Esto mantiene todo abierto y tangible.

En segundo lugar, los equipos de XP también lanzan a sus usuarios finales con frecuencia. Los proyectos de XP Web se publican con la misma frecuencia que todos los días, en proyectos internos mensualmente o con mayor frecuencia. Incluso los productos envueltos se envían cada tres meses.

Puede parecer imposible crear buenas versiones tan a menudo, pero los equipos de XP de todo el mundo lo hacen todo el tiempo. Consulte la integración continua para obtener más información al respecto, y tenga en cuenta que estos lanzamientos frecuentes se mantienen confiables debido a la obsesión de XP con las pruebas, como se describe aquí en Pruebas de clientes y Desarrollo.

## 9.1.6. Diseño simple

Los equipos de XP crean software para un diseño simple pero siempre adecuado. Comienzan de manera simple, y a través de las pruebas de los programadores y la mejora del diseño, lo mantienen de esa manera. Un equipo de XP mantiene el diseño exactamente adecuado para la funcionalidad actual del sistema. No hay movimiento perdido, y el software siempre está listo para lo que viene después.

El diseño en XP no es algo que ocurre una sola vez, o una cosa inicial, es algo que ocurre todo el tiempo. Hay pasos de diseño en la planificación de versiones y la planificación de iteraciones, además de que los equipos participan en sesiones de diseño rápido y revisiones de diseño a través de la refactorización, durante el transcurso de todo el proyecto. En un proceso incremental e iterativo como Extreme Programming, un buen diseño es esencial. Es por eso que hay tanto enfoque en el diseño a lo largo de todo el desarrollo.

## 9.1.7 Programación en parejas

Todo el software de producción en XP está construido por dos programadores, sentados uno al lado del otro, en la misma máquina. Esta práctica garantiza que todos los códigos de producción sean revisados ​​por al menos otro programador, y resulta en un mejor diseño, mejores pruebas y mejores códigos.

Puede parecer ineficiente tener a dos programadores haciendo el "trabajo de un programador", pero lo contrario es cierto. La investigación sobre la programación de pares muestra que el emparejamiento produce mejores códigos casi al mismo tiempo que los programadores que trabajan solos. Así es: ¡dos cabezas realmente son mejores que una!

Algunos programadores se oponen a emparejar la programación sin siquiera intentarlo. Toma algo de práctica hacerlo bien, y debes hacerlo bien durante unas semanas para ver los resultados. El noventa por ciento de los programadores que aprenden programación de pares lo prefiere, por lo que lo recomendamos a todos los equipos.

El emparejamiento, además de proporcionar mejores códigos y pruebas, también sirve para comunicar el conocimiento en todo el equipo. Al cambiar de pareja, todos obtienen los beneficios del conocimiento especializado de todos. Los programadores aprenden, sus habilidades mejoran, se vuelven más valiosas para el equipo y la empresa. El emparejamiento, incluso por sí solo fuera de XP, es una gran victoria para todos.

## 9.1.8 Pruebas de desarrollo

La Programación Extrema está obsesionada con la retroalimentación, y en el desarrollo de software, una buena retroalimentación requiere buenas pruebas. Los mejores equipos de XP practican el "desarrollo basado en pruebas", trabajando en ciclos muy cortos para agregar una prueba y luego hacerla funcionar. Casi sin esfuerzo, los equipos producen código con casi el 100% de cobertura de prueba, lo que es un gran paso adelante en la mayoría de las tiendas. (Si sus programadores ya están haciendo pruebas más sofisticadas, más potencia para usted. ¡Sigan así, solo puede ayudar!)

No es suficiente escribir pruebas: tienes que ejecutarlas. Aquí, también, la Programación Extrema es extrema. Estas "pruebas de programador" o "pruebas unitarias" se recopilan todas juntas, y cada vez que un programador publica algún código en el repositorio (y las parejas suelen publicar dos veces al día o más), cada una de las pruebas del programador debe ejecutarse correctamente. Cien por ciento, todo el tiempo! Esto significa que los programadores obtienen retroalimentación inmediata sobre cómo les está yendo. Además, estas pruebas proporcionan una ayuda inestimable a medida que se mejora el diseño del software.

## 9.1.9 Mejora de diseño

Extreme Programming se enfoca en entregar valor comercial en cada iteración. Para lograr esto en el transcurso de todo el proyecto, el software debe estar bien diseñado. La alternativa sería reducir la velocidad y finalmente atascarse. Así que XP usa un proceso de mejora continua del diseño llamado Refactoring, del título del libro de Martin Fowler, "Refactorización: mejorar el diseño del código existente".

El proceso de refactorización se centra en la eliminación de la duplicación (una señal segura de diseño deficiente) y en aumentar la "cohesión" del código, al tiempo que reduce el "acoplamiento". La alta cohesión y el bajo acoplamiento han sido reconocidos como los sellos de un código bien diseñado durante al menos treinta años. El resultado es que los equipos de XP comienzan con un diseño bueno y simple, y siempre tienen un diseño bueno y simple para el software. Esto les permite mantener su velocidad de desarrollo, y de hecho generalmente aumenta la velocidad a medida que avanza el proyecto.

Refactoring es, por supuesto, fuertemente respaldado por pruebas exhaustivas para asegurarse de que a medida que el diseño evoluciona, nada se rompe. Por lo tanto, las pruebas de los clientes y las pruebas del programador son un factor habilitador crítico. Las prácticas de XP se apoyan mutuamente: son más fuertes juntas que por separado.

## 9.1.10 Integración continúa

Los equipos de Extreme Programming mantienen el sistema completamente integrado en todo momento. Decimos que las construcciones diarias son para los débiles: los equipos de XP construyen varias veces al día.

El beneficio de esta práctica se puede ver al pensar en proyectos de los que quizás hayas oído hablar (o incluso haber participado) en los que el proceso de compilación fue semanal o menos frecuente y generalmente condujeron al "infierno de la integración", donde todo se rompió y no se conocía el por qué.

La poca integración conduce a problemas serios en un proyecto de software. En primer lugar, aunque la integración es fundamental para el envío de un buen código de trabajo, el equipo no tiene experiencia en ello y, a menudo, se delega a personas que no están familiarizadas con todo el sistema. En segundo lugar, el código integrado con poca frecuencia suele ser, diría normalmente, un código defectuoso. Los problemas se infiltran en el tiempo de integración y no son detectados por ninguna de las pruebas que tienen lugar en un sistema no integrado. En tercer lugar, el proceso de integración débil conduce a largas congelaciones de código. La congelación de códigos significa que tiene largos períodos de tiempo en los que los programadores podrían estar trabajando en funciones importantes que pueden enviarse, pero que esas funciones deben retenerse. Esto debilita su posición en el mercado o con sus usuarios finales.

## 9.1.11 Código de la propiedad colectiva

En un proyecto de Programación Extrema, cualquier par de programadores puede mejorar cualquier código en cualquier momento. Esto significa que todo el código obtiene el beneficio de la atención de muchas personas, lo que aumenta la calidad del código y reduce los defectos. También hay otro beneficio importante: cuando el código es propiedad de individuos, las características requeridas a menudo se colocan en el lugar equivocado, ya que un programador descubre que necesita una función en algún lugar del código que no le pertenece. El propietario está demasiado ocupado para hacerlo, por lo que el programador coloca la función en su propio código, donde no pertenece. Esto lleva a un código feo y difícil de mantener, lleno de duplicaciones y con baja (mala) cohesión.

La propiedad colectiva puede ser un problema si las personas trabajan ciegamente en un código que no entendieron. XP evita estos problemas a través de dos técnicas clave: el programador prueba los errores de captura, y la programación de pares significa que la mejor manera de trabajar con códigos desconocidos es emparejarse con el experto. Además de garantizar buenas modificaciones cuando sea necesario, esta práctica difunde el conocimiento en todo el equipo.

## 9.1.12 Codificación standard

Los equipos de XP siguen un estándar de codificación común, por lo que todo el código del sistema se ve como si hubiera sido escrito por una única persona (muy competente). Los detalles del estándar no son importantes: lo importante es que todo el código parece familiar, en apoyo de la propiedad colectiva.

## 9.1.13 Metáfora

Los equipos de Extreme Programming desarrollan una visión común de cómo funciona el programa, lo que llamamos la "metáfora". En el mejor de los casos, la metáfora es una simple descripción evocativa de cómo funciona el programa, como "este programa funciona como una colmena de abejas, sale al polen y lo trae de vuelta a la colmena" como una descripción de una información basada en agentes sistema de recuperación.

Algunas veces una metáfora suficientemente poética no surge. En cualquier caso, con o sin imágenes vívidas, los equipos de XP utilizan un sistema común de nombres para asegurarse de que todos entiendan cómo funciona el sistema y dónde buscar la funcionalidad que está buscando, o para encontrar el lugar adecuado para poner. la funcionalidad que está a punto de agregar

## 9.1.14 Desarrollo sostenible

Los equipos de Extreme Programming están a largo plazo. Trabajan duro y a un ritmo que puede mantenerse indefinidamente. Esto significa que trabajan horas extras cuando es efectivo, y que normalmente trabajan de tal manera que maximicen la productividad semana tras semana. En estos días se comprende bastante bien que los proyectos de la marcha de la muerte no son productivos ni producen software de calidad. Los equipos de XP están ahí para ganar, no para morir.

## 9.2 ¿En qué se diferencia de otras metodologías?

Extreme Programming es una disciplina de desarrollo de software basada en valores de simplicidad, comunicación, retroalimentación y valentía. Funciona al reunir a todo el equipo en presencia de prácticas simples, con suficientes comentarios para que el equipo pueda ver dónde están y para ajustar las prácticas a su situación única.

# Capítulo 10

## Seguimiento y control del presente estudio

En el presente capítulo se explican los conceptos y metodologías referentes al estándar PMBOK (Project Management Body of Knowledge) para la administración de proyectos.

## 10.1 Proyecto

Un proyecto es un conjunto de objetivos que se llevan a cabo con la finalidad de crear un producto, servicio o resultado único que puede ser tangible o intangible que sea duradero. Los proyectos son temporales, es decir, que existe un principio y un final bien definidos, dicho final es alcanzado cuando se cumplen todos los objetivos o cuando se alcanzan los objetivos que pueden cumplirse. Esto se debe a que, durante la realización del proyecto, se puede determinar que algunos objetivos no pueden ser cumplidos, no podrán cumplirse a tiempo o ya no se consideran necesarios para el proyecto. Igualmente, un proyecto puede culminar sin cumplir los objetivos cuando el cliente o la persona que solicitó el proyecto decide terminar con el mismo al considerar que ya no es necesario su existencia.

## 10.2 Gerencia del proyecto

La Gerencia de proyectos es la disciplina de organizar y administrar los recursos de forma tal que un proyecto dado sea terminado completamente dentro de las restricciones de alcance, tiempo y coste planteados a su inicio.

Dada la naturaleza única de un proyecto, en contraste con los procesos u operaciones de una organización, administrar un proyecto requiere de habilidades y competencias específicas.

## 10.3 Metodología de gestión de proyectos a utilizar

La Metodología a utilizar para el desarrollo del proyecto es el Project Management Institute, constituido por un conjunto de fundamentos, estándares y pautas para la dirección de proyectos utilizando buenas prácticas o soluciones evaluadas por varias organizaciones cuyo consenso validan el valor y la utilidad de las mismas. Del proceso de gestión del proyecto se realizarán los siguientes:

## 10.3.1 Gestión del alcance del proyecto

En este se determinan los procesos necesarios para definir y controlar qué objetivos serán incluidas para completar el proyecto de forma exitosa. La Gestión del alcance del proyecto se divide en seis procesos las cuales son:

## 10.3.1.1 Planificar Gestión del Alcance

En este proceso, se crea un plan de gestión de alcance en el que se documenta cómo es definido, validado y controlado el alcance del proyecto; ésta proporciona una guía y dirección sobre cómo será gestionado el alcance durante el desarrollo de un proyecto. Este proceso inicia con el análisis de la información contenida en la constitución del proyecto, plan y procesos de la organización y cualquier otro factor ambiental que pueda afectar a la organización, con la finalidad de reducir los riesgos de que ocurra una modificación de los alcances del proyecto.

## 10.3.1.2 Recopilar Requisitos

En este proceso es se identifican, documentan y gestionan todas las necesidades y requisitos del personal interesado para alcanzar los objetivos del proyecto; ésta proporciona la base para definir el alcance del proyecto incluyendo el del producto. En general, las organizaciones clasifican los requisitos en diferentes tipos dependiendo de las necesidades, tales como requisitos de negocio, técnicos, de transición, funcionales y no funcionales, etc.

## 10.3.1.3 Definir el Alcance

En este proceso se desarrolla una descripción detallada como los límites del producto o servicio mediante el análisis de los requisitos recopilados para especificar cuáles de dichos requisitos son incluidos y cuáles son excluidos del alcance del proyecto. Este proceso se desarrolla inicialmente describiendo en alto nivel la visión para el proyecto global y a medida que se realice iteraciones, se describen la planificación y requisito de forma detalladas.

## 10.3.1.4 Crear la EDT/WBS

En este proceso se subdividen los entregables y el trabajo del proyecto en componentes pequeños y fáciles de manejar usando EDT o Estructura de Descomposición del Trabajo lo que permite proporcionar una visión más estructurada de lo que debe entregarse. Dichos componentes son denominados paquetes de trabajo y en ellas se agrupan los artefactos, productos o entregables que son el resultado de un conjunto de actividades realizadas y no la actividad específica.

## 10.3.1.5 Validar Alcance

En este proceso se formaliza la aceptación de los entregables que se hayan completado, lo que aporta objetividad y mejor posibilidad de que el producto o servicio sea aceptado por medio de la evaluación de cada entregable individual. A diferencia del Control de Calidad, las salidas obtenidas como lo son la documentación de requisitos, datos de desempeño y bases del alcance son las que constituyen la base para realizar la validación y aceptación final.

## 10.3.1.6 Controlar el Alcance

En este proceso se monitorea el estado del alcance y las gestión de los cambios que puedan presentarse, lo que permite asegurar que todos los cambios, acciones preventivas o acciones correctivas se procesen a través del proceso de Control de Cambios.

## 10.3.2 Gestión del tiempo

En este se determinan los procesos necesarios para gestionar la terminación en el plazo del proyecto. La gestión del tiempo se divide en siete procesos las cuales son:

## 10.3.2.1 Planificar la Gestión del Cronograma

En este proceso se establecen los procedimientos, políticas y documentación para planificar, desarrollar, ejecutar y controlar el cronograma del proyecto, a su vez se establecen la forma en que se informa sobre posibles contingencias relativas al cronograma y cómo se manejan las mismas, proporcionando una guía y dirección de cómo se gestiona el proyecto.

## 10.3.2.2 Definir las actividades

En este proceso se identifican y documentan las acciones a realizar para generar los entregables del proyecto lo que permite el desglose de los paquetes de trabajo generado con los EDT en actividades que proporcionan una base para estimar, programar, ejecutar, monitorear y controlar el trabajo del proyecto.

## 10.3.2.3 Secuenciar las actividades

En este proceso se identifican y documentan las relaciones entre las actividades a desarrollar del proyecto, generando una secuencia lógica para obtener la máxima eficiencia al tener en cuenta todas las restricciones del proyecto.

## 10.3.2.4 Estimar los recursos de las actividades

En este proceso se estiman el tipo, cantidades y características de materiales, personal, equipos, suministros y complementos requeridos para llevar a cabo cada una de las actividades, para estimar los costos y duración de la manera más precisa.

## 10.3.2.5 Estimar la duración de las actividades

En este proceso se estiman la cantidad de períodos de trabajos necesarios para finalizar las actividades individuales con los recursos ya asignados, esta constituye una entrada fundamental del desarrollo del cronograma ya que establece la cantidad de tiempo necesario para finalizar cada una de las actividades.

## 10.3.2.6 Desarrollar el Cronograma

En este proceso se analizan las secuencias de actividades, los requisitos de recursos y personal, duraciones y restricciones del cronograma para generar el modelo de programación del proyecto con fechas planificadas para completar las actividades del proyecto. Este proceso es iterativo ya que requiere el repaso y revisión de las estimaciones de duración y recursos para que el personal asignado pueda confirmar la fecha de inicio y fin para así evitar conflictos con los cronogramas de los recursos o actividades asignadas en otros ámbitos del proyecto.

## 10.3.2.7 Controlar el Cronograma

En este proceso se monitorea el estado de las actividades del proyecto para actualizar y gestionar los cambios del cronograma dado el caso que sea necesario para cumplir el plan del proyecto, lo que permite detectar desviaciones y establecer las acciones correctivas y preventivas para minimizar el riesgo.

## 10.3.3 Gestión del costo del proyecto

En este se determinan los procesos necesarios para planificar, estimar, presupuestar, financiar, gestionar y controlar los costos de modo que el proyecto finalice dentro del presupuesto aprobado. La gestión del costo del proyecto se divide en cuatro procesos las cuales son:

## 10.3.3.1 Planificar la Gestión de los Costos

En este proceso se establece las políticas, procedimientos y documentación requerido para planificar, gestionar, ejecutar el gasto y controlar los costos del proyecto proporcionando una guía y dirección sobre cómo se gestiona los costos del proyecto a lo largo del desarrollo.

## 10.3.3.2 Estimar los Costos

En este proceso se desarrolla una estimación aproximada de los recursos monetarios requeridos para completar las actividades del proyecto. Esta estimación de costo se basan en la información existente de un momento determinado teniendo en cuenta el balance entre costo y riesgos, compartición de recursos y alquiler de equipos en caso de ser necesario.

## 

## 10.3.3.3 Determinar el presupuesto

En este proceso se suman los costos estimados de las actividades o paquetes de trabajo para establecer una línea de costos autorizados contemplando todos los fondos para ejecutar el proyecto, al establecer dicha línea, se puede controlar y monitorear el desempeño del proyecto.

## 10.3.3.4 Controlar los Costos

En este proceso se monitorea el estado del proyecto para actualizar sus costos y gestionar cambios lo que permite detectar desviaciones con respecto al plan con el objetivo de tomar acciones preventivas y correctivas para minimizar el riesgo.

## 10.3.4 Gestión de calidad

En este se determinan los procesos que establecen las políticas de calidad, objetivos y responsabilidades de calidad para que el proyecto cumpla con las necesidades. La gestión de calidad se divide en tres procesos las cuales son:

## 10.3.4.1 Planificar Gestión de Calidad

En este proceso se identifican los requisitos y estándares de calidad para los entregables del proyecto así como también la documentación de cómo se demuestra que el proyecto cumple con dichos requisitos y estándares lo que proporcio guía a los participantes del proyecto a cómo se gestiona y valida la calidad a lo largo del desarrollo. El desarrollo de este plan debe realizarse en paralelo a los planes de cronograma y costos.

## 10.3.4.2 Realizar Aseguramiento de Calidad

En este proceso se realiza la auditoría de los requisitos de calidad y los resultados obtenidos por medio de las medidas de control de calidad lo que facilita la mejora de los procesos de calidad para garantizar que se utilicen los estándares de calidad adecuadas. Este proceso para garantizar la calidad busca construir confianza en que las salidas futuras o en curso se completarán de forma que cumplan los requisitos y expectativas establecidos.

## 10.3.4.3 Controlar la Calidad

En este proceso se monitorea y registra los resultados de la ejecución de las actividades de calidad para evaluar el desempeño y sugerir cambios necesarios lo que permite identificar las causas de una calidad ineficiente del proceso o entregable y aplicar acciones para eliminarlas, y valida que los entregables cumple con los requisitos del personal interesado para la aceptación final.

## 10.2.5 Gestión de las comunicaciones

En este se determinan los procesos necesarios para asegurar que la planificación, recopilación, creación, distribución, almacenamiento, recuperación, gestión, control, monitoreo y disposición final de la información del proyecto sean oportunos y adecuados, la comunicación genera puentes entre los diferentes interesados que pueden tener diferentes antecedentes y experiencias cuyas perspectivas impactan o influyen en la ejecución o resultado del proyecto. La gestión de las comunicaciones se divide en tres procesos las cuales son:

## 10.2.5.1 Planificar la Gestión de las Comunicaciones

En este proceso se desarrolla un enfoque y plan adecuados para la comunicación tomando en cuenta las necesidades y requisitos de información del personal interesado, identificando el enfoque a utilizar para comunicarse de forma eficaz y eficiente.

## 10.2.5.2 Gestionar las Comunicaciones

En este proceso se recopilan, distribuyen, almacenan, recuperan y realizan las disposición final de la información del proyecto lo que permite un flujo de comunicación eficaz y eficientes entre los interesados, adicionalmente, este proceso permite al personal interesado generar solicitudes de información adicional, en caso de ser necesario, de aclaración y debate usando el modelo de Emisor-Receptor que incorpora ciclos de retroalimentación para dar oportunidades de interacción o participación a los interesados.

## 10.2.5.3 Controlar las Comunicaciones

En este proceso se monitorean y controlan las comunicaciones a lo largo del ciclo de vida del proyecto para garantizar que la necesidad de información de los interesados se cumpla de forma eficaz y eficiente lo que asegura un flujo óptimo de información entre todos los participantes.

## Consideraciones éticas y legales

En cuanto a la información documental consultada, se tuvo especial cuidado y respeto de los derechos de autor en diversos tópicos de interés emitidos por terceras personas, por lo que se hace referencia al autor original de las mencionadas.

Cabe mencionar que aunado a lo expuesto anteriormente, se hace uso de las normas APA para la escritura de las referencias bibliográficas, y se indica el aporte de diversos autores que contribuyen a la realización de la presente investigación.

# Parte IV.

## Propuesta de Trabajo Especial de Grado

# Capítulo 11

## Análisis y Propuesta

## **1**1**.1 Caso de estudio**

Sistemas de recursos humanos, y el proceso de reclutamiento y selección de personal en las empresas.

## 11.2 Situación actual

Hoy en día, el éxito de las organizaciones y empresas depende en gran medida de una buena selección de candidatos que desean aportar sus conocimientos, experiencias y aptitudes para mejorar el funcionamiento y desarrollo de las tareas de la empresa. La necesidad de reclutar nuevos candidatos en las empresas viene dado por diferentes causas como requerir personal que contribuya a realizar, de forma eficaz y eficiente, una serie de actividades que contribuyan al logro de las diversas áreas de una organización y así alcanzar la misión y visión de dicha organización, o debido a factores externos tales como expansión de la empresa, lo que genera nuevos puestos de trabajos, pérdida de persona o despido, entre otros.

El proceso de reclutamiento de personal comienza a raíz de un requerimiento o solicitud que posee la empresa en desarrollar una serie de actividades, en un área determinada, donde la empresa carece de personal calificado para ejecutar dicha labor, donde posteriormente se procede a determinar:

1. El nombre de la vacante.
2. Las actividades y responsabilidades que deben desempeñarse en la vacante.
3. Requerimientos técnicos mínimos necesarios.
4. Requerimientos sociales e interpersonales necesarios.
5. Horario de trabajo disponible para la vacante.
6. Salarios
7. Número de vacantes.

Una vez que la empresa determina dichos datos sobre la vacante se procede a realizar la publicación de la vacante ya sea en el sitio web de la empresa o en sitios web donde potenciales candidatos pueden encontrar información de la vacante, donde posteriormente, se inicia un proceso de solicitud de una serie de candidatos para ocupar una vacante.

Dicho proceso, se ejecuta de forma presencial donde el candidato consulta información del puesto y entrega los documentos para su postulación. De esta forma, la empresa debe emplear tiempo en atender a cada candidato que desee postularse, y el candidato debe dirigirse a la empresa a entregar los documentos para su postulación, posterior a ello, de acuerdo a las necesidades de la empresa, ciertos candidatos serán seleccionados para continuar con el proceso de reclutamiento, donde se harán evaluaciones, entrevistas para conocer las expectativas del candidato en la empresa, etc.

Todos estos procesos consumen recursos (tiempo y costos) para ambos lados del proceso; la empresa suele realizar impresiones de cuestionarios o exámenes lo que genera pérdidas económicas si el candidato decide retirarse del proceso o no es seleccionado para continuar, requiere de mayor personal para poder proveer un feedback a cada candidato sobre el estado en el proceso de selección en un tiempo suficiente que no genere incertidumbre en el candidato sobre su selección, y por otro lado, cada candidato se siente incómodo con las diversas etapas que debe realizar durante el proceso de selección ya que deben solicitar permisos laborales para poder asistir a dichas etapas, todo esto causa una pérdida de dinero en transportes, impresión de los documentos necesarios por la empresa como currículos, certificaciones, títulos, entre otros lo que origina que un candidato abandone el proceso por ser lento y costoso.

Para cualquier empresa está sobreentendida la necesidad de contratar nuevo personal constantemente, sobre todo si se tiene un crecimiento de cualquier magnitud, o cualquier tarea sin atender que podría ser crucial para el desarrollo y progreso de la empresa, es por eso que se tiene la necesidad no solo de contratar, sino de captar candidatos que llenen dichos roles de la manera más eficiente posible, es simple estadística, mientras más candidatos se puedan procesar procesar, más probabilidades se tienen de contratar a un candidato que satisfaga las expectativas de la organización. Sin embargo, muchas organizaciones no poseen una forma de trabajo eficiente que permita evaluar a una gran cantidad de candidatos en tiempo eficiente, debido a que, generalmente dichas evaluaciones son realizadas por una sola persona que suele dedicar parte de su tiempo para aplicar estos procesos de selección, evaluación y contratación, generando tiempos de respuestas poco útiles para los candidatos.

Pero captar una gran cantidad de candidatos implica una gran cantidad de gasto de recursos para todos los involucrados, sea desde la empresa o desde el punto de vista de los candidatos, cualquiera que sea el gasto tiempo, dinero o material. Entre algunos de los inconvenientes para los usuarios y para las empresas, se pueden mencionar los siguientes.

## 11.2.1 Inconvenientes para el candidato

En la mayoría de los casos, es necesario que el candidato proporcione una documentación previa a la empresa, entre ellas se incluye el Currículum vitae impreso, copia de diplomas o títulos obtenidos, referencias laborales, certificaciones, entre otras.

Cada candidato que desee postularse debe invertir tiempo y dinero en realizar el traslado a la sede de la empresa, donde dependiendo la distancia entre el postulante y la sede, puede incurrir en costos adicionales de transporte.

Al momento de realizar la primera entrevista, el postulante puede retirarse del proceso al recibir información como temas asociados al horario laboral, la flexibilidad de dicho horario para permitir laborar y estudiar mientras emplea sus funciones en la organización, considerar el proceso largo y tedioso cuando la organización no especifica los beneficios y expectativas económicas que pueden ofrecer; todo esto acarreando una pérdida de dinero y tiempo importante por conceptos de transportes e impresión de documentos requeridos.

Posterior a la primera entrevista, las organizaciones suelen realizar una serie de evaluaciones técnicas, psicológicas, espaciales y demás aspectos que son importantes para la organización conocer del postulado antes de decidir, esto acarrea costos adicionales de transporte y tiempo de horas laborales que implica realizar solicitudes para asistir a dichos procesos para posteriormente recuperar el trabajo del día.

Luego de realizar los procesos realizados por la organización hacia cada postulados, cada una toma una cantidad de tiempo considerable en proporcionar una respuesta que le de información al postulado si fue seleccionado o no, generando incertidumbre lo que causa que los postulados abandonen el seguimiento y decidan realizar procesos de selección con otras organizaciones.

## 11.2.2 Inconvenientes para las organizaciones

En la mayoría de los casos, la información detallada de la vacante es proporcionado al candidato de forma oral, por lo que la organización debe

En algunos casos proporcionan información de la vacante al potencial candidato, de forma oral, y para lo cual deben invertir un tiempo considerable por cada candidato.

Existe la posibilidad que el candidato decida retirarse del proceso debido a que no se cumple con las expectativas salariales, flexibilidad en los horarios laborales o fijación de horario que beneficie tanto al candidato como a la organización; como también la organización puede determinar que el candidato no cumple con los requisitos para la vacante. Todo este proceso requiere una inversión de tiempo mínimo de una hora por cada candidato, por lo que sí existe una cantidad elevada de candidatos, el costo en tiempo llega a ser muy ineficiente para las organizaciones.

Siempre que un candidato supera la primera etapa del proceso y este decide continuar con el proceso de selección, las organizaciones proceden a realizar evaluaciones para conocer las destrezas profesionales de los candidatos, su nivel capacidad lógica y resolución de problemas, trabajo en equipo y bajo presión, entre otras; o entrevistas realizadas para conocer los valores, personalidad y aspectos de interés para el reclutador. Todo este proceso puede tomar hasta meses en algunas organizaciones, por lo que muchos candidatos no deciden esperar ese tiempo para tener una respuesta y se postulan a otras vacantes laborales en otras organizaciones.

Habitualmente, las organizaciones no preguntan a los potenciales candidatos las expectativas salariales, a lo cual, luego de pasar el largo proceso de evaluación y selección, la propuesta salarial determinada por la organización puede ser rechazada por el candidato, originando que, tanto la organización como el postulante, haya perdido tiempo y dinero en continuar todo el proceso hasta el final.

Por último, en la mayoría de los casos, las organizaciones fallan en notificar a los potenciales candidato si lograron superar una etapa o no de forma rápida y práctica, por lo que los candidatos asumen que no superaron el proceso y deciden postularse a otras vacantes o continuar procesos de selección en otras organizaciones.

De esta necesidad surge una propuesta que pretende mitigar el desperdicio de recursos y a su vez ofrecer una herramienta que beneficie a los involucrados, creando una aplicación web configurable que permita a las organizaciones definir la vacantes que requiere la organización, las responsabilidades asociadas a dicha vacante, determinar a qué persona reporta la vacante, los requisitos y cualidades tanto indispensables como optativos, información sobre flexibilidad y horarios de trabajo, propuesta salarial de dicha vacante. A su vez, definir las evaluaciones asociadas a la vacante junto a la puntuación de cada respuesta a modo de automatizar el proceso de evaluación y finalmente, realizar entrevistas en línea para adquirir información adicional sobre los candidatos.

En cuanto a los candidatos dar la posibilidad de postularse a las vacantes, realizar las evaluaciones relacionadas a la vacante que se desea ocupar y monitorear de forma activa la etapa en el que se encuentra, y el estado del mismo durante todo el proceso de selección. Por lo que se propone realizar un estudio de factibilidad del emprendimiento mencionado, a objeto de determinar, si la iniciativa de negocios propuesta es viable o no.

# Capítulo 12

## Propuesta Trabajo Especial de Grado

De lo expuesto anteriormente, surge la iniciativa y oportunidad de diseñar y desarrollar una aplicación web que permita a la empresa ofrecer información detallada de la vacante, y a los candidatos de postularse al proceso de selección y obtener respuesta de forma oportuna sobre el estado de su postulación durante todo el proceso.

Por parte de la organización, cada una tiene la posibilidad configurar en el portal la publicación de las vacantes disponibles, incluyendo información detallada donde se incluye la descripción completa, los requisitos específicos y opcionales, las responsabilidades y objetivos asociado y la propuesta salarial. Esto le permite a la organización exponer toda la información junto a la descripción de cada una de las etapas que conforman el proceso de selección para el puesto vacante, por lo que cada candidato, al revisar dicha publicación puede determinar, si la vacante se adecua a su aspiración del puesto que quisiera obtener y la disposición a participar en todo el proceso.

Por parte del candidato, cada uno puede ingresar al portal y revisar de forma secuencial todas las vacantes disponibles por las organizaciones, dando flexibilidad a filtrar por cargo de la vacante, responsabilidades, requisitos, entre otros. En caso de que se genere interés en una vacante, el candidato podrá postularse ingresando los datos necesarios juntos a curriculums y certificaciones en digital y proceder a realizar las distintas evaluaciones requeridas por la organización que oferta la vacante en su proceso de selección. Finalmente cada candidato podrá revisar en tiempo real el estado de su postulación y en qué fase del proceso se encuentra, teniendo dicha información disponible a cualquier hora.

Lo anteriormente mencionado otorga beneficios a ambos participantes del proceso, en cuanto a las organizaciones, permite publicar las vacantes por medio de la Aplicación Web alcanzando una mayor población, generar y configurar las evaluaciones de forma que, dependiendo del tipo de evaluación, se evalúe de forma automatizada y seleccione a los mejores candidatos para continuar con el proceso y proveer feedback a cada uno de los candidatos de forma automatizada, esto reduce los costos de impresión de folletos para la publicación de las vacantes y de las evaluaciones o cuestionarios, automatiza los procesos claves reduciendo el tiempo necesario para procesar a cada uno de los candidatos.

Por otro lado, los candidatos pueden obtener acceso a toda la información referente a la vacante para determinar si desea postularse o no, y en caso de postularse, puede introducir su curriculum junto a sus certificaciones de forma digital y realizar las evaluaciones y obtener pronta respuesta para determinar si continúa en el proceso o no, esto reduce los costos de impresión de dichos documentos, permite al candidato postularse y revisar el estado de su postulación a cualquier hora del día en la comodidad de su hogar reduciendo a su vez los costos generados por el transporte hacia la sucursal de la organización y las solicitudes de permisos laborales.

Finalmente la aplicación Web es la que se encarga de capturar la información proporcionada tanto por las organizaciones y candidatos y almacenarlo utilizando un sistema de base de datos, suministrando a su vez, análisis de los datos para generar información para facilitarle a la organización en el proceso de selección.

## 12.1 Justificación

El objetivo prioritario de la presente investigación, no es documentar ni informar cuáles son los gastos que afronta una empresa durante los procesos de reclutamiento, sino proponer una solución eficiente y de mayor flexibilidad que las herramientas convencionales que se encuentran en la actualidad, y demostrar que dichos procesos pueden ser optimizados de manera que, las empresas puedan agilizar sus procedimientos de contratación, y que a su vez les permite incrementar las posibilidades de seleccionar el candidato idóneo para la empresa contratante, y por otro lado eliminar o mitigar los inconvenientes mencionados anteriormente para los potenciales candidatos

Cabe resaltar que la contratación de personal es un proceso complejo donde existen una serie de prácticas que son de uso común, y que probablemente no sean las más eficientes, y de serlo, dichas prácticas pueden ser obsoletas y no adaptarse al dinamismo y a las necesidades del mercado actual.

Una contratación errónea acarrea gastos significativos, pérdidas y consecuencias que no son sencillas de manejar, como emocionales, legales, morales, monetarias, entre otras; es por ello que cualquier aporte que contribuya a tomar mejores decisiones durante el proceso de reclutamiento debe ser aprovechado.

El realizar el presente estudio y recopilar datos relacionados al reclutamiento, va a permitir analizar patrones o prácticas comunes que serán de utilidad para el desarrollar un sistema eficiente para el beneficio de las personas que sean involucradas, beneficios detallados a continuación.

1.-Facilitar el proceso de toma de decisión referente a si continuar o no el proceso de selección de cierto candidato por medio de un análisis automatizados que determina si la información se cumple con los requerimientos solicitados por la vacante.

2.-Analizar las fortalezas y debilidades de las estructuras y procedimientos de contratación actuales sobre las cuales el presente proyecto se basa para determinar y proponer condiciones de mejora.

3.-Determinar las mejores prácticas a tomar para que los procesos de selección que se planteen sean factibles para cada participante del proceso.

4.-Reducir el nivel de incertidumbre que se genera en el candidato a postularse al momento de someterse al proceso de selección y, a su vez, mantener informados a los interesados por la vacante.

5.-Justificar el uso de sistema basándonos en los recursos, tanto de tiempo y económico, que ahorra para sus usuarios .

6.-Determinar la plataforma tecnológica y dinámicas de trabajo óptimos que el sistema usará la captación, procesamiento y análisis de la información ingresada por los usuarios.

7.-Analizando los datos generados por los usuarios, se busca generar información que permitirán llevar a cabo la toma de decisiones sobre contrataciones con bases sólidas.

8.-Adquirir conocimientos que le serán de utilidad a los futuros gerentes de recursos humanos, sobretodo para evitar escenarios comunes que conlleva al desperdicio de recursos durante procesos de captación de personal obsoletos.

Lo anterior mencionado da flexibilidad a los candidatos en realizar el proceso de selección y, a su vez, el seguimiento de su estado durante todo el proceso desde la comodidad de su casa, sin restricciones de tiempo, lo que evita que el candidato deba solicitar permisos para retirarse de la oficina para realizar entrevistas con la incertidumbre de saber si dicha entrevista es beneficiosa o no.

Por otro lado, da flexibilidad a la empresa en realizar y configurar los cuestionarios para que estos sean evaluados de forma automática sin participación del personal de Recursos Humanos, y encuestas que pueden ser reutilizados en múltiples vacantes disponibles en la empresa. Además puede proveer a todos sus candidatos el feedback sobre su estado durante el proceso de selección de forma eficiente eliminando la incertidumbre que dichos candidatos puedan generar durante el proceso.

## 12.2 Objetivo General

Desarrollar una aplicación Web configurable para apoyar el procesamiento del proceso de reclutamiento y selección definido por cada organización.

## 12.3 Objetivos específicos.

Para los objetivos específicos optamos por dividir por puntos claves, quedando como se muestra a continuación.

## 12.3.1 Analizar y definir los requerimientos para la aplicación Web

1. Recopilar datos sobre los métodos comunes de búsqueda, evaluación y selección de personal utilizado por diversas empresas, utilizando la técnica de investigación documental (Google, Facebook, Microsoft, Crossover), y la técnica de Focus Group (consiste en reunir a un grupo de 5 a 6 personas y conversar de manera informal, acerca de los métodos de búsqueda y selección de personal de empresas en los que hayan tenido experiencia), y utilizando la experiencia de los autores y el tutor del presente trabajo especial de grado.
2. Evaluar las pérdidas económicas incurridas tanto por las organizaciones que van tienen la necesidad de realizar un proceso de selección para ocupar una vacante y de los usuarios que decidan postularse en la misma, así como también, los costos en tiempo generado por cada participante en el proceso.
3. Elaborar una dinámica de búsqueda evaluación y selección del personal que sea configurable que contribuya a automatizar el proceso de búsqueda, selección y contratación de personal.
4. Crear Diagramas que ilustran el intercambio de información entre el candidato y la organización durante cada etapa del proceso de selección y la información que genera la organización a la hora de evaluar a un candidato.

## 12.3.2 Realizar el diseño de la aplicación Web.

1. Analizar los requerimientos a forma de dar solución que permitan solventar los inconvenientes descritos en el planteamiento del problema en un ambiente Web usable.
2. Definir la arquitectura para el desarrollo de la aplicación Web incluyendo los diversos módulos que conforman el funcionamiento del mismo.
3. Elaborar diagramas de caso de uso, diagrama de clases, diagrama de secuencia.
4. Diseñar el modelo de datos y la arquitectura en la base de datos.
5. Realizar procedimientos flexibles y configurables para que puedan adaptarse a cualquier empresa que requiera contratación de personal, sin importar su escala y complejidad.
6. Construir prototipos de interfaces que constituye la aplicación Web.

## 12.3.3 Definir los componentes de la aplicación Web que serán configurables o parametrizables de forma detallada.

1. Definir los componentes que se pueden administrar, configurar y parametrizar de forma detallada
2. Definir cómo se puede administrar, configurar y parametrizar los componentes mencionados en el punto anterior.
3. Definir el flujo de trabajo por parte de ambos participantes identificando los puntos claves que influyen en la misma.
4. Seleccionar las tecnologías óptimas a utilizar para el desarrollo de la aplicación Web.

## 12.3.4 Desarrollar la aplicación Web

1. Desarrollar los requerimientos de la aplicación Web empleando las mejores prácticas de programación en las que incluye la documentación del código fuente, uso de patrones y antipatrones de diseño, técnicas de interacción Humano-Computador para agilizar el proceso de reclutamiento y selección de personal para la organización.

## 12.3.5 Realizar pruebas sobre la aplicación Web

1. Realizar pruebas unitarias sobre la aplicación utilizando la práctica de ingeniería de software TDD (Test Driven Development).
2. Mantener documentación que detalle los resultados de las pruebas obtenidas.
3. Generar lecciones aprendidas en relación al uso de TDD (Test Driven Development).

## 12.3.6 Elaborar los manuales de instalación y de uso de las aplicaciones

1. Elaborar un manual que detalle el paso a paso para la instalación de la aplicación Web.
2. Elaborar un manual detallado que describa los componentes y su forma de uso que permita al usuario utilizar de forma satisfactoria la aplicación Web.
3. Realizar una demostración de uso de la aplicación.

## 12.4 Alcances y Limitaciones

El trabajo a realizar contiene ciertos alcances y limitaciones que son considerados para el trabajo de investigación, detallados a continuación:

## 12.4.1 Alcance

El presente estudio tiene como alcance crear una aplicación Web que permita a los candidatos realizar búsquedas de vacantes disponibles publicados por las empresas y postularse a los mismos y a las empresas a publicar el detalle de sus vacantes para ser accedidos por los candidatos. En cuanto al proceso de reclutamiento se apoya a la empresa proveyendo una serie de procedimientos que permita filtrar de forma automatizada a los postulantes y encontrar aquellos adecuados a las necesidades de la posición.

Los aspectos puntuales que comprende la investigación están referidos a algunos estudios de los procedimientos que se aplican, y en ciertas prácticas de búsqueda laboral, para evitar sus mayores defectos, de tal modo que se puedan plantear procedimientos que aprovechen mejor los recursos disponibles.

Cabe destacar que se analizaran diferentes métodos comunes de reclutamiento, es imposible abarcar todas las prácticas ya que existen métodos variados o tan específicos para ciertas empresas que se haría absurdo, o por el contrario, usan combinaciones de técnicas tan ligeramente intercambiables que no vale la pena estudiarlas.

## 12.4.2 Limitaciones

El presente trabajo especial de grado se limita a la creación de una aplicación Web que presenta una solución genérica, que sea utilizable y configurable por una empresa, con el fin de agilizar los procesos de reclutamiento, evaluación, selección y contratación de personal; y para proporcionar flexibilidad y comodidad para los candidatos que decidan postularse a las vacantes publicadas por dichas empresas.

Entre las limitaciones que podrían afectar la calidad de la información obtenida para proceder a elaborar el diseño, el desarrollo y utilización de la aplicación Web, se mencionan las siguientes:

## 12.4.2.1 Factores culturales o sociales

Pueden darse casos en que la aplicación pueda que no satisfaga las necesidades organizacionales, esto se debe a que existen ciertas vacantes que requieren la aplicación de pruebas de tipo presencial en cada una de las etapas del proceso de evaluación y selección de personal, como ocurre en el caso del personal que labora en las embajadas, que casos son sometidos a exámenes de drogas y al polígrafo, para determinar la veracidad de sus respuestas a lo largo de cada una de las etapa del proceso de selección.

Entre otros de los factores pueden mencionarse la preferencia de algunas empresas de mantener un contacto cara a cara con los potenciales candidatos para determinar la personalidad y carácter de dichos candidatos, resistencia al cambio, entre otras.

## 12.4.2.2 Falta de instrumentos para mediciones técnicas

El comportamiento humano varía mucho entre individuo e individuo, por lo que el tomar conclusiones del estado emocional, nivel de frustración, y características similares es, cuando poco, un reto nada trivial, por lo tanto, la mayoría de los datos serán netamente basados en la interacción de las personas con la aplicación y es por ello que estos datos tienden a ser aproximados.

## 12.4.2.3 Características de las muestras

Los instrumentos y métodos de recolección de datos sobre la aplicación y sus características se realizan por lo general mediante el uso de encuestas, evaluaciones, cuestionarios, pruebas psicológicas, entre otros. Estos dados a su vez por personas con diferentes estratos, como por ejemplo, docentes, obreros, jefes de familia, comerciantes, etc, esa muestra de sujetos producen una calidad de información a recolectar heterogénea, y en consecuencia, necesitará ser interpretada.

## 12.4.2.4 Sesgo del Sujeto

Las respuestas que se obtendrán de las encuestas o entrevistas, dependen del grado de conocimiento que tengan los sujetos, de sus opiniones y al ser estas preguntas muy puntuales, la información que se obtiene de ellas será muy discreta.

## 12.4.2.5 Disposición del candidato

La información proporcionada por el candidato estará limitada por la disposición que tiene dicho candidato en brindar información personal por medio de la Web y las limitaciones de tiempo pertinentes a sus ocupaciones laborales para responder dichos instrumentos.

# Capítulo 13

## Tecnologías propuestas

En particular, los autores determinaron cuáles serían las tecnologías a utilizar en la capa de presentación o front-end, y cuáles serían las tecnologías a utilizar para el desarrollo del núcleo de la aplicación, o back-end, descritas a continuación:

## 13.1 Tecnologías del lado del cliente (Capa de presentación)

Las tecnologías del lado del cliente son ejecutadas e interpretadas del lado del cliente en una aplicación web, es decir, en el navegador web. En muchos casos el buen funcionamiento de estas tecnologías va a depender del tipo de navegador web y de la versión de cada uno de ellos. Por lo general estas tecnologías son utilizadas para mostrar información, darle formato a dicha información, solicitar datos, etc. Entre las tecnologías que se explican a continuación están: el Lenguaje de Marcas de Hipertexto, Hojas de Estilo en Cascada, JavaScript y jQuery.

## 13.1**.1 HTML 5** Lenguaje de Marcas de Hipertexto

El Lenguaje de Marcas de Hipertexto, mejor conocido como HTML (HiperText Markup Language), es un lenguaje de programación utilizado para el desarrollos de sitios web diseñado por el físico Tim Berners Lee en el año 1989, esta se publicó en un documento formal en el año 1991, convirtiéndose en un estándar para el desarrollo de páginas web en el año 1995 ya que ofrecía una estructura básica para la definición de como se muestra en contenido de una página web, como lo son las imágenes, formularios, textos generales, botones, entre otros.  
A partir del año 1996, la World Wide Web Consortium (W3C), organización que define la mayoría de los estándares para el desarrollo web, se ha encargado de publicar constantes revisiones (HTML 3.2, HTML 4, HTML 4.01) de este estándar, ampliándolo y depurándose con cada versión. HTML 5 fue la última revisión aprobada por esta organización, en el año 2014.

HTML 5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Incorpora nuevas etiquetas con mayor significado semántico, mejora la integración con las tecnologías de desarrollo web actual, permite desplegar objetos y animaciones en formato 3D (tridimensional), tiene un mejor soporte multimedia y ahorra la necesidad de tener que procesar medios de audio o video a través de complementos instalados en el navegador web. A continuación se muestra se detallan las principales diferencias entre HTML 4.01 y HTML 5

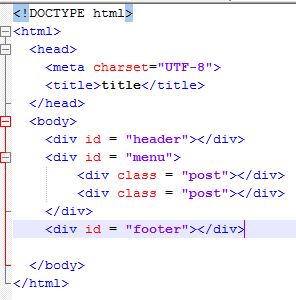
## 13.1.2 HTML5 vs HTML4.01

Una de las principales diferencias es la integración de etiquetas de audio y video a su vez la integración de gráficos vectoriales como SVG (Scalable Vector Graphics) y canvas, que, aunque se podía lograr en HTML4.01, era necesario utilizar varias tecnologías como VML (Vector Markup Language), Silver Light y Flash.

En HTML4.01 el cache del navegador puede utilizarse como un sistema de almacenamiento temporal, sin embargo, en HTML5 tanto la cache de aplicación, cache del navegador y base de datos SQL web puede utilizarse como almacenamiento temporal.

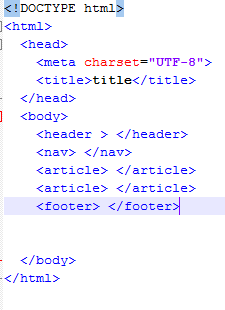
Otra de las principales diferencias de HTML5 es que ésta no esta basado en SGML (Standard Generalized Markup Language), estandar que define los lenguajes de marcados generalizados para documentos, lo que permitió la mejora de las reglas de parsing de la información, incrementando la compatibilidad con distintos sistemas.

Finalmente se implementaron etiquetas o elementos que permiten estructurar la página web de forma más organizada, reduciendo la cantidad de bloques <div> generados al desarrollar la página Web. Un ejemplo de esto se detalla a continuación:



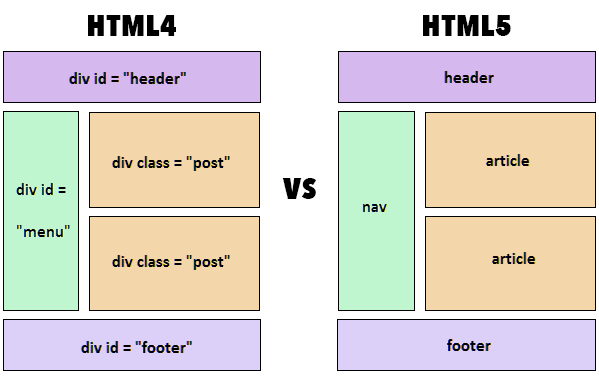
**Figura 17 - Estructura de un sitio Web sencillo con HTML4.01**

En la imagen se puede observar que para desarrollar un sitio web sencillo con un menú de navegación con varios artículos es necesario generar una numerosa cantidad de bloques <div>, esto puede presentarse a confusión al momento de añadir estilos por medio de CSS, en especial a medida que incrementa el tamaño de los componentes añadidos en la página Web. Dicha estructura escrita con HTML5 quedaría estructurado de la siguiente manera.



### Figura 18 - Estructura de un sitio Web sencillo con HTML5

Donde se puede observar que con HTML 5 se disminuye el uso de la etiqueta <div> (utilizada para definir un bloque de contenido), y se introducen etiquetas con mayor significado semántico para que la estructura de una página web sea más fácil de entender, en comparación a HTML4.01. En resumen se puede observar la diferencia entre ambas estructuras en la siguiente imagen.



### Figura 19 - Diferencias entre la estructura del código HTML4.01 y HTML5

## 13.1.2 CSS3 Hojas de Estilo en Cascada

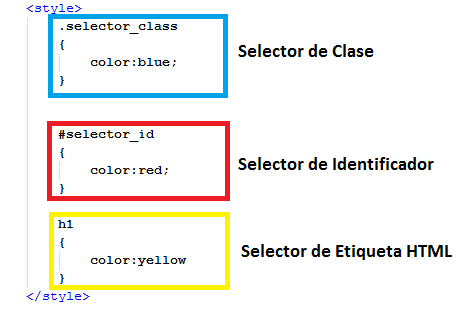
Mientras que HTML permite definir la estructura de una página web, las Hojas de Estilo en Cascada (Cascading Style Sheets - CSS) son las que ofrecen la posibilidad de definir las reglas y estilos de representación en diferentes dispositivos, ya sean pantallas de equipos de escritorio, portátiles, tabletas, celulares u otros dispositivos capaces de mostrar contenidos web

CSS fue diseñado por Håkon Wium Lie y Bert Bos entre 1994 y 1995 y se convirtió en un estándar en 1996, con el nombre de CSS 1. Al igual que con HTML, la W3C se ha encargado de publicar constantes revisiones de este lenguaje, siendo la versión CSS 2.1 el estándar actual que ofrece gran compatibilidad con la mayoría de los navegadores web del mercado.

En el año 2005 se comenzó a definir el sucesor de la versión CSS 2.1, al cual se lo conoce como CSS 3, pero actualmente sigue en desarrollo y sólo se han publicado algunos borradores. Esta versión a pesar de que aún no es compatible con todos los navegadores, ofrece una gran variedad de opciones muy importantes para las necesidades del diseño web actual, como funciones avanzadas de movimiento y transformación, opciones de bordes redondeados, sombras en textos y cajas, entre otras opciones, sin la necesidad de usar un editor gráfico.

La sintaxis del lenguaje CSS consta de elementos llamados reglas, las cuales se dividen en selectores y declaraciones, donde el selector es el identificador al que se le quiere dar la declaración de un conjunto de propiedades con sus valores. Los selectores se pueden aplicar tanto para etiquetas html, identificadores o clases, permitiendo aplicar reglas específicas a un conjunto específico de componentes del sitio web.

### Figura 20 - Diferencia entre los múltiples selectores HTML



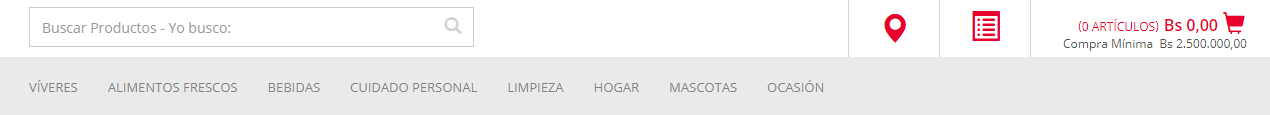
## 13.1.3 Bootstrap

Bootstrap, es un marco de trabajo (framework) que permite crear interfaces web con las tecnologías de CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web obtiene la capacidad adaptarse automáticamente al tamaño de una computadora de escritorio o portátil, una tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como diseño adaptativo (responsive design).

Bootstrap nos permite tener una aplicación Web bien organizada de forma visual rápidamente, ya que permite utilizar muchos elementos como iconos desplegables, elementos de HTML5, CSS y Javascript por medio del “grid system” o sistema de celdas que permite maquetar cómo se muestran los componentes según el espacio disponible en el dispositivo, a través de una división de doce columnas.

### Figura 21 - Grid System del framework Bootstrap para dispositivos pequeños.

Este sistema de Grid se integra fácilmente con librerías Javascript, esto permite cambiar la forma en que se muestra la navegación según el tamaño del dispositivo que ingresó a la aplicación.





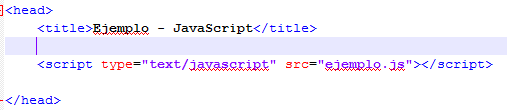
### Figura 22 - Diferencia entre la visualización de navegación entre los diferentes tamaños de dispositivos

## 13.1.4 JavaScript

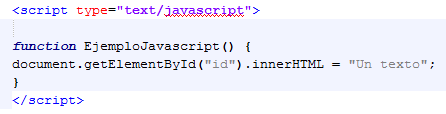
Es un lenguaje de programación interpretado basado en guiones (scripts), inicialmente desarrollado por Brendan Eich para luego ser una marca registrada por la empresa Sun Microsystem la cual no guarda ninguna relación con el lenguaje de programación compilado de alto nivel Java, utilizado principalmente para realizar desarrollos de páginas Web dinámicas, es decir, interfaces que por medio de una serie de disparadores (como por ejemplo el clic a un botón, elemento del sitio, etc.) asociados a eventos, activa una acción o serie de acciones para incorporar efectos como animaciones, despliegue de ventanas con información o mensajes útiles para el usuario final, entre otros.

Gracias a que éste lenguaje es interpretado no existe la necesidad de realizar procesos intermedios como la compilación para poder ser ejecutado, permitiendo realizar pruebas rápidamente en cualquier navegador que tenga permitido la carga de Javascript.

Existen dos formas de incluir código Javascript, ya sea que se encuentre embebido dentro del código HTML (habitualmente no recomendado para reducir duplicación de código y fomentar la reutilización, y a su facilitar el mantenimiento del mismo) o llamando a un archivo “.js”.



### Figura 23 - Ejemplo del llamado a un archivo con extensión js.

****

### Figura 24 - Ejemplo del código Javascript embebido en el código HTML.

## 13.2 Tecnologías del lado del servidor (Núcleo de la aplicación)

Las tecnologías del lado del servidor son ejecutadas e interpretadas del lado del servidor. Por lo general estas tecnologías son utilizadas para procesar la información generado por el cliente como registro de usuario, detalles de vacante, entre otros; y ejecutar funciones esenciales como validación de los datos de login, realizar dicho login, entre otros, por medio de un Servicio Web.

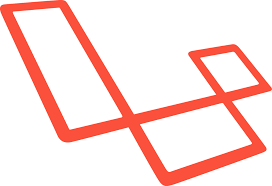
## 13.2.1 PHP

Para el desarrollo del proyecto se utilizará el lenguaje de programación de propósito general PHP, cuya sintaxis, por medio las etiquetas especiales “<?php” y “?>”, permiten ejecutar comandos que facilita el desarrollo de una aplicación Web. Lo que distingue PHP de Javascript es que el código HTML que se genera se realiza del lado del servidor y éste es enviado al cliente, por lo que el cliente solo tendrá acceso al resultado de ejecutar el script PHP.

Las principales ventajas de php se encuentra en la antigüedad del lenguaje, gracias a esto, PHP permite generar entornos de desarrollos de forma rápida y configurable ya que cuenta con una inmensa cantidad de librerías generadas por la comunidad que son totalmente autoinstalables que se integran a las aplicaciones sin mucha complejidad, paquetes que incluye compatibilidad con múltiples idiomas, facilidad en el acceso de base de datos incluyendo PostgreSQL. En cuanto a desventajas, debido a que el código PHP es interpretado del lado del servidor, a medida que aumenta las peticiones simultáneas sobre el servidor puede causar un colapso por lo que no permitiría a los clientes ingresar a la aplicación, aunado a esto, a pesar de que el lenguaje ha mejorado al paso de los años en el aspecto de seguridad, se debe tener un buen nivel de conocimiento para configurar el servidor PHP de forma que no existan brechas de seguridad que permitan a ladrones informáticos o hackers acceder a información sensible de forma no autorizada.

## 13.2.2 Laravel

Laravel es un framework open source (código abierto) creado por Taylor Otwell hecho para una arquitectura MVC, o modelo vista controlador, donde el modelo representa la forma en que se operan los datos del negocio, el controlador que se encarga de responder a eventos para adquirir información del modelo y finalmente la vista que es la forma que se presenta el modelo al usuario final; y además resuelve necesidades de eventos y autenticación de usuarios.

****

### Figura 25 - Icono oficial de Laravel

## 13.2.2.1 Evolución del Framework

A continuación se detalla el proceso de evolución del framework Laravel desde sus inicios hasta su versión actual de forma breve y las características que se fueron añadiendo a lo largo de su historia.

## 1.- Laravel

Laravel fue creado por Taylor Otwell en el año 2011 para proveer una alternativa al framework CodeIgniter, framework PHP utilizado para desarrollo Web que permite a los desarrolladores generar código siguiendo las buenas prácticas, sin embargo requería un nivel de dominio alto para su uso correcto, Laravel en cambio provee herramientas que son fáciles de manejar y customizar para cualquier tipo de desarrollador y a su vez le da libertad a desarrolladores expertos de modularizar los componentes de la aplicación.

## 2.- Laravel 2

Laravel 2 fue liberado el Septiembre de 2011 brindando mejoras en las estabilidad y a su vez, la inclusión de los controladores, componente que faltaba para convertirse framework MVC completo.

## 3.- Laravel 3

Liberado el Febrero del 2012, Laravel 3 buscaba ofrecer mayor compatibilidad con más manejadores de base de datos y permitir la migración de base de datos como forma de control de versión. En esta versión se desarrolló la línea de comandos Artisan, que contenía una lista de comandos que facilitaba la construcción y configuración de la aplicación web a desarrollar.

## 4.- Laravel 4

En Mayo del año 2013 el creador Taylor Otwell libera una nueva versión de Laravel llamado Laravel 4. Dicha versión incorpora nuevas mejoras y funcionalidades, como lo son la carga mediante el uso de colas de procesamiento flexibles, expansión en las operaciones para el procesamiento de datos y en las características de patrones basado en filtros, encolar eventos, añadir eventos a los modelos y se introdujo un sistema de correo. Todas estas mejoras permiten que las aplicaciones sean más modulares, y por consiguiente, facilita las pruebas unitarias.

## 5.- Laravel 5

El día 4 de Febrero del 2015 el creador libera una nueva versiòn de Laravel. Dicha versión añadió una gran variedad de nuevas características y funcionalidades que ofrecen al desarrollador mayor flexibilidad al momento de desarrollar las aplicaciones, entre las características principales de la versión Laravel 5 tenemos:

* La estructura del directorio de archivos fue cambiado al implementar el estándar “PSR-4 autoloading”, dicho estándar describe las especificaciones para realizar la carga de múltiples clases de forma automática por medio de una ruta, eso significa que todas las clases en un proyecto Laravel tendrán su propio espacio de nombres.
* Artisan realiza caching de rutas de acceso incrementando notablemente el rendimiento de las cargas y registro de las rutas, sin embargo, dicho cambio es más notable en aplicaciones grandes que en aquellas aplicaciones con un número inferior de rutas.
* Se añade la característica de Middleware de Enrutamiento, esta permite añadir capas adicionales a las ruta https, ejecutándose bloques de códigos adicionales a todas o a un grupo selecto de rutas en la aplicación.
* Se añadieron contratos que son un conjunto de interfaces que definen los servicios principales que ofrece el framework.
* Se añadieron paquetes para la integración de registro y acceso por medio de Facebook, Twitter, Google y Github llamado Socialite que lidia con el estándar Open Authorization (OAuth), estándar abierto que permite notificar al proveedor del recurso que el dueño del recurso, es decir, el usuario a autenticarse, que se otorga permiso a la aplicación de usar su información.
* Incorporación de un paquete llamado Flysystem desarrollado por Frank de Jonge el cual es una abstracción de un sistema de archivos que permite intercambiar fácilmente archivos locales con archivos remotos. Compatible con Amazon Web Services - S3 y Rackspace Cloud Files.

## 13.2.2.2 Características principales de Laravel

A continuación se detallan las principales características que provee el framework Laravel que ayudarán al desarrollo de la aplicación Web.

## 1.- Paquetes

Laravel incluye un sistema de paquete modular que permite incluir funcionalidades e integrarlos a la aplicación de forma automática ya sea importando de forma manual o descargando de un repositorio utilizando la línea de comandos Artisan.

## 2.- Eloquent ORM

ORM (Object-Relational Mapping) es una técnica empleada para transformar datos en forma de objeto con cualquier lenguaje orientado a objetos los datos obtenidos de una base de datos relacional, permitiendo manejar dicha información como un objeto y las ventajas del mismo como lo son herencia y polimorfismo. Eloquent ORM es la implementación de un ORM por medio de PHP ActiveRecord con la capacidad de aplicar restricciones tanto en relaciones como en cargas anidadas, otorgando un control total sobre la data.

## 3.- Lógica de la Aplicación

En Laravel, la lógica de la aplicación puede ser implementada utilizando controladores o directamente en la declaración de las rutas otorgando flexibilidad al momento de crear funcionalidades nuevas.

## 4.- Enrutamiento inverso

El enrutamiento inverso permite crear enlaces a rutas con nombre. Al crear enlaces, simplemente se indica el nombre de la ruta y Laravel insertará automáticamente el URI correcto.

## 5.- Controladores RESTful

Los controladores RESTful proveen una opción para separar la lógica de negocio para las solicitudes GET y POST.

## 6.- Carga automática de Clases

La carga automática de clases evita que se deba mantener una configuración de componentes a cargar, por lo que si se desea usar una biblioteca o modelo el framework automáticamente se encargará de realizar la carga de componentes y librerías necesarias.

## 7.- View Composers

Los View Composers son bloques de códigos que pueden ser ejecutados al momento que una vista esté cargando, eliminando la necesidad de que los controladores carguen data innecesaria para algunas vistas de la aplicación.

## 8.- IoC Container (Contenedor de Control de Inversión)

El Control de Inversión provee métodos para crear uno o varios objetos e instanciar estos de manera opcional en las vistas y permite acceder a dichos objetos desde cualquier lugar del código.

## 9.- Migraciones

En Laravel, las migraciones es una forma de control de versión integradas de los esquemas de base de datos, permitiendo actualizar los esquemas al momento de enviar un push request por medio de Artisan.

## 10.- Pruebas Unitarias

Laravel tiene implementado un conjunto de pruebas unitarias que garantizan que los cambios realizados a lo largo del desarrollo de la aplicación no afecte a componentes anteriormente desarrollados, a su vez permite realizar pruebas unitarias personalizadas por medio de Artisan.

## 11.- Paginación Automática

La paginación automática previene que la lógica de la aplicación se sobrecargue con demasiadas configuraciones de paginación, esto reduce la necesidad de hacer recuento de los registros de base de datos adquiridos y fragmentar la data utilizando offsets y limits. Sin embargo, Laravel también permite la configuración manual de paginación para dar flexibilidad y personalización al desarrollador.

# Capítulo 14

## Patrones de Diseño

Los patrones de diseño son conjuntos de soluciones que solventan problemas típicos, comunes o recurrentes, encontrados a la hora de desarrollar una aplicación. Aunque una aplicación sea única, tendrá partes comunes y por lo tanto retos y problemas comunes con otras aplicaciones: acceso a datos, creación de objetos, operaciones entre sistemas etc. En lugar de reinventar nuevas formas de solucionar el mismo, podemos utilizar o guíanos por algún patrón, ya que son soluciones probadas, estandarizadas, implementadas y comúnmente documentadas por comunidades de desarrolladores.

## 14.1 ¿Porqué utilizar patrones de diseño?

Desarrollar aplicaciones robustas y fáciles de mantener es algo que requiere diseño y estándares, se deben cumplir lineamientos. estos patrones, más que reglas de diseño son también un conjunto de prácticas o estándares, no son obligatorias, pero sí muy recomendadas. Siempre podemos decidir no aplicarlas o adaptarlas muy vagamente a la solución. pero si no lo hacemos, hay que ser conscientes de la razón de no aplicarlas con esmero y de sus posibles consecuencias a largo plazo, así como las limitaciones de crecimiento por no seguir un patrón definido.  
  
 Los patrones de diseño nos ayudan a cumplir muchos de estos principios, requerimientos o reglas de diseño. Programación SOLID, control de cohesión y acoplamiento o reutilización de código son algunos de los beneficios que podemos conseguir al utilizar patrones.

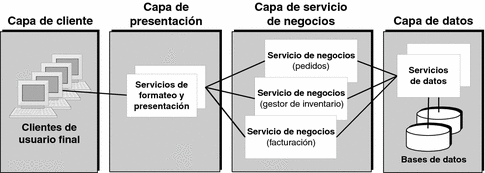
## 14.2 Desventajas de utilizar patrones de diseño

Las consecuencias de no utilizarlas son difícilmente posibles de evaluar, pero mientras más grande la aplicación, menos probable es que pueda llevarse a cabo sin algún conjunto de reglas para mantener la consistencia de la solución, cualquier sistema con robustez, o que requiera mantenimiento a largo plazo o de diferentes personas o que requiera escalar o reescribir segmentos grandes de módulos, puede considerarse destinado al fracaso si no se sigue nunca un patrón de diseño, los únicos sistemas capaces de sobrevivir sin diseño son programas pequeños o con funciones muy específicas. Por ejemplo, algunos patrones de diseño son:

## 1. Patrón en capas:

Este patrón se puede usar para estructurar programas que se pueden descomponer en grupos de subtareas, cada una de las cuales se encuentra en un nivel particular de abstracción. Cada capa proporciona servicios a la siguiente capa superior.  
  
 Las 4 capas más comúnmente encontradas de un sistema de información general son las siguientes.  
1.-Capa de presentación (también conocida como capa UI)  
2.-Capa de aplicación (también conocida como capa de servicio)  
3.-Capa de lógica de negocios (también conocida como capa de dominio)  
4.-Capa de acceso a datos (también conocida como capa de persistencia)

**Ejemplos:**  
 Aplicaciones de escritorio generales.  
 Aplicaciones web de comercio electrónico.

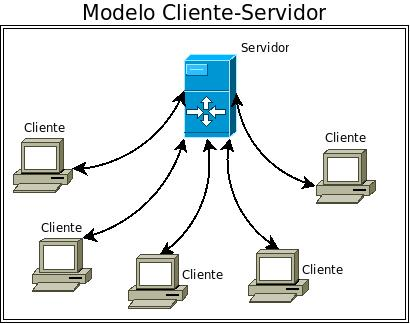


### Figura 26 - Diagrama de patrón en capas

## 2. Patrón cliente-servidor

Este patrón consiste en dos partes; un servidor y múltiples clientes. El componente del servidor proporcionará servicios a múltiples componentes del cliente. Los clientes solicitan servicios del servidor y el servidor proporciona servicios relevantes a esos clientes. Además, el servidor sigue escuchando las solicitudes de los clientes.

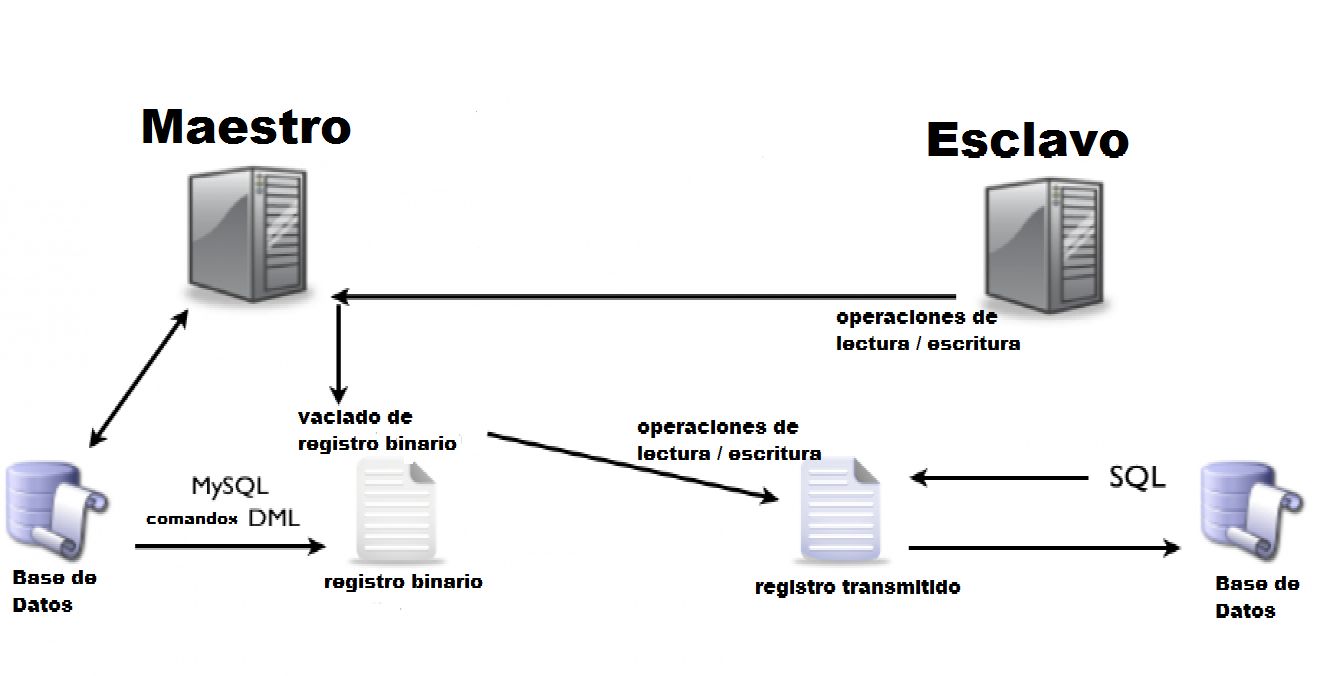
**Ejemplo:**  
Aplicaciones en línea como correo electrónico, uso compartido de documentos y banca.



### Figura 27 - Diagrama de patrón Cliente Servidor

## 3. Patrón maestro-esclavo

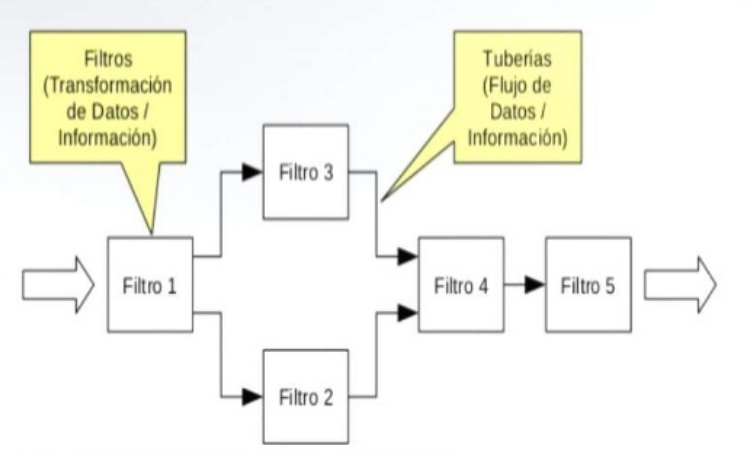
Este patrón consiste en dos partes; maestro y esclavos. El componente maestro distribuye el trabajo entre componentes esclavos idénticos y calcula el resultado final de los resultados que devuelven los esclavos.  
  
 **Ejemplo:**  
 En la replicación de la base de datos, la base de datos maestra se considera como la fuente autorizada y las bases de datos esclavas se sincronizan con ella.  
Periféricos conectados a un bus en un sistema informático (unidades maestra y esclava).



### Figura 28 - Diagrama de patrón maestro esclavo

## 4. Patrón de filtro de tubería

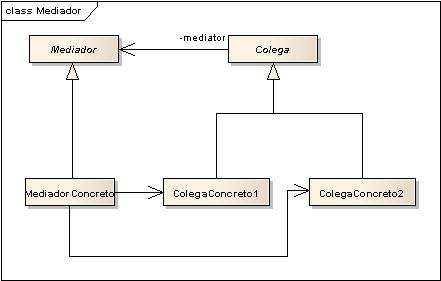
Este patrón se puede usar para estructurar sistemas que producen y procesan una secuencia de datos. Cada paso de procesamiento se incluye dentro de un componente de filtro. Los datos que se procesarán se pasan a través de las tuberías. Estas tuberías se pueden utilizar para el almacenamiento en búfer o con fines de sincronización.  
  
 **Ejemplo:**  
 Compiladores Los filtros consecutivos realizan análisis léxico, análisis sintáctico y generación de código. Flujos de trabajo en bioinformática.



### Figura 29 - Diagrama de filtro tubería

## 5. Patrón del agente mediador

Este patrón se usa para estructurar sistemas distribuidos con componentes desacoplados. Estos componentes pueden interactuar entre sí mediante invocaciones de servicios remotos. Un componente de intermediario es responsable de la coordinación de la comunicación entre los componentes.  
  
 Los servidores publican sus capacidades (servicios y características) a un intermediario. Los clientes solicitan un servicio del intermediario y el intermediario redirecciona al cliente a un servicio adecuado desde su registro.  
**Ejemplo:**  
Software de Message Broker como Apache ActiveMQ, Apache Kafka, RabbitMQ y JBoss Messaging.

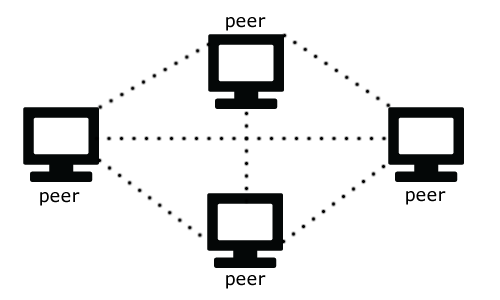


### Figura 30 - Diagrama de patrón mediador

## 6. Patrón de punto a punto (peer to peer)

En este patrón, los componentes individuales se conocen como pares. Los pares pueden funcionar tanto como un cliente, solicitando servicios de otros pares, y como un servidor, proporcionando servicios a otros pares. Un par puede actuar como un cliente o como un servidor o como ambos, y puede cambiar su rol dinámicamente con el tiempo.

**Ejemplo:**  
Redes de intercambio de archivos como Gnutella y G2)  
Protocolos multimedia como P2PTV y PDTP.



### Figura 31 - Diagrama de patrón punto a punto

## **7. Patrón de bus de evento**

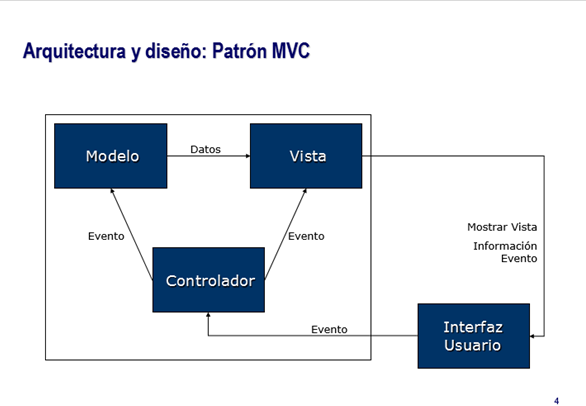
Este patrón trata principalmente con eventos y tiene 4 componentes principales; fuente de evento, escucha de evento, canal y bus de evento. Las fuentes publican mensajes en canales particulares en un bus de eventos. Los oyentes se suscriben a canales particulares. Los oyentes son notificados de los mensajes que se publican en un canal al que se han suscrito anteriormente.

**Ejemplo:**  
 Desarrollo de Android y servicios de notificación.



### Figura 32 - Diagrama de patrón bus de eventos

## **8. Patrón de modelo-vista-controlador**

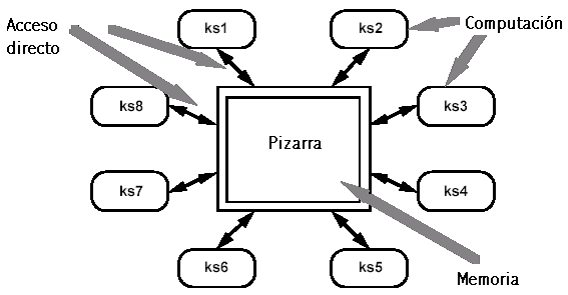
Este patrón, también conocido como patrón MVC, divide una aplicación interactiva en 3 partes, el Modelo el cual contiene la funcionalidad y los datos básicos, La vista el muestra la información al usuario (se puede definir más de una vista) y el controlador el cual maneja la entrada del usuario. Esto se hace para separar las representaciones internas de información de las formas en que se presenta y acepta la información del usuario. Desacopla los componentes y permite la reutilización eficiente del código.  
  
 **Ejemplo:**  
 Arquitectura para aplicaciones World Wide Web en los principales lenguajes de programación. Marcos web como Django, Laravel (seleccionado para el proyecto) y Rails.   


### Figura 33 - Diagrama de patrón modelo vista controlador

## **9. Patrón de pizarra**

Este patrón es útil para problemas para los que no se conocen estrategias de solución deterministas. El patrón de pizarra consta de 3 componentes principales. La pizarra el cual es una memoria global estructurada que contiene objetos del espacio de solución, la fuente de conocimiento el cual implementa módulos especializados con su propia representación componente de control el cual selecciona, configura y ejecuta módulos.

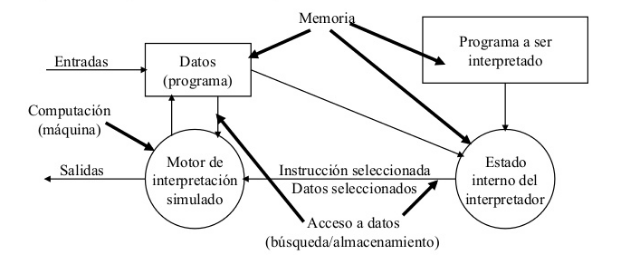
Todos los componentes tienen acceso a la pizarra. Los componentes pueden producir nuevos objetos de datos que se agregan a la pizarra. Los componentes buscan tipos particulares de datos en la pizarra, y pueden encontrarlos por coincidencia de patrones con la fuente de conocimiento existente.  
  
 **Ejemplo:**  
 Reconocimiento de voz, identificación y seguimiento del vehículo, identificación de la estructura proteica, sonar señala la interpretación.



### Figura 34 - Diagrama de patrón pizarra

## 10. Patrón de intérprete

Este patrón se usa para diseñar un componente que interpreta programas escritos en un lenguaje dedicado. Específica principalmente cómo evaluar las líneas de programas, conocidas como oraciones o expresiones escritas en un idioma particular. La idea básica es tener una clase para cada símbolo del idioma.  
  
 **Ejemplo:**  
 Lenguajes de consulta de base de datos como SQL, idiomas utilizados para describir los protocolos de comunicación.



### Figura 35 - Diagrama de patrón intérprete

|  |  |  |
| --- | --- | --- |
| Nombre | Ventajas | Desventajas |
| Capas | - Una capa inferior puede ser usada por diferentes capas superiores.  - Las capas hacen la estandarización fácil ya que definen los niveles claramente.  - Cambios pueden hacerse a cualquier capa sin afectar a las otras | - No es universalmente aplicable.  - Algunas Capas podrían ser omitidas en ciertas situaciones. |
| Cliente-Servidor | Es un buen modelo para servicios donde los clientes hacen peticiones, | - Servicios separados son a veces atendidos por diferentes hilos del servidor.  - La comunicación inter-procesos puede causar sobrecarga al servidor ya que diferentes clientes tienen diferentes estados. |
| Maestro-Esclavo | La Consistencia, la ejecución de un servicio es delegado a diferentes esclavos, con diferentes implementaciones. | - Los esclavos están aislados, no tienen estado compartido.  - La comunicación entre los esclavos y el maestro puede volverse un problema, por ejemplo en sistemas en tiempo real.  - este patrón sólo puede aplicado a problemas que pueden ser descompuestos en partes. |
| Filtro de tubería | Posee procesamiento concurrente. cuando la entrada y salida de de los procesos consisten en transferencias de datos los filtros se encargan de recibir y sesgar los datos.  - Es fácil añadir más filtros a los existentes así que el sistema puede ser extendido con facilidad.  Los filtros son reusables, pueden usarse en diferentes situaciones y recombinarse con otros filtros. | La eficiencia depende del filtro más lento.  - Datos algunas veces causan sobrecarga en procesamiento moviendo datos de un filtro a otro. |
| Agente mediador | Permite modificaciones dinámicas, creación, agregación, borrado y relocalización de objetos y hace la distribución de estos transparentes al desarrollador. | Requiere estandarización de la descripción de los servicios, y aveces algunas clases de homologación de los datos. |
| Punto a punto | Suporta computación descentralizada  - Muy robusta en caso de fallo de alguno de los nodos.  Altamente escalable en términos de recursos y poder de cómputo. | No hay garantía de calidad de servicio, ya que los nodos no son dedicados sino voluntarios.  - La seguridad es un problema difícil de garantizar.  - El desempeño depende del número de nodos |
| Bus de eventos | Nuevos Publicadores, Subscriptores y conecciones pueden ser agregadas fácilmente.  - Efectivo para aplicaciones altamente distribuidas. | Pueden haber problemas de escalabilidad, ya que los eventos todos viajan a través del mismo bus, |
| Modelo vista controlador | Hace sencillo tener múltiples vistas del mismo modelo, que puede ser conectado y desconectado en tiempo real. | Según aumenta la complejidad, puede llevar a actualizaciones innecesarias para las acciones de los usuarios. |
| Pizarra | Es fácil agregar nuevas aplicaciones.  Extender la estructura de datos en espacio es fácil | Modificar la estructura de los datos espaciales es difícil, ya que todas las aplicaciones son afectadas, puede necesitar sincronización y control de acceso. |
| Intérprete | Altamente Dinámico en comportamiento de ser posible  Bueno para el usuario final programable mente hablando (configurar, entender)  Aumenta la flexibilidad, porque reemplazar el programa interprete es fácil. | Debido a que el lenguaje interpretado es generalmente lento que los compilados, el desempeño puede ser un problema en problemas de mayor escala. |

### Tabla Comparativa 1 - comparación



### Figura 36 - Diagrama de nivel de abstracción

# Capítulo 15

## Sistema Manejador de Base de Datos

Según Silberschatz A. et al. (2002) un Sistema Manejador de Base de Datos (SMBD) “consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos” y su objetivo principal “es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente”.

En la actualidad existe una gran variedad de sistemas manejadores de bases de datos, muchos de ellos son propietarios o comerciales como Oracle, Microsoft SQL Server, IBM Informix, Sybase, etc. y otros no comerciales o de código abierto como MySQL o PostgreSQL, para efectos de este proyecto, optamos por utilizar el SMBD código abierto PostgreSQL detallando sus características a continuación.

## 15.1 PostgreSQL

PostgreSQL es un potente Sistema Manejador de Base de Datos objeto-relacional de código abierto, creado por Michael Stonebreaker bajo el nombre de Post Ingres y cuenta con un desarrollo activo con más de 15 años bajo una arquitectura que ha sido probada por los desarrolladores ganándose una sólida reputación por su fiabilidad, integración de los datos y facilidad de corrección ante situaciones de errores. Debido a que PostgreSQL trabaja de forma rápida y eficiente con grandes cantidades de datos y además ofrece un alto grado de concurrencia de usuarios que pueden acceder a la vez al sistema manteniendo la integridad de los datos, se considera una herramienta que favorece a usuarios con sistemas empresariales de gran tamaño.

Este Sistema Manejador de Base de Datos se encuentra distribuido bajo la licencia PostgreSQL, el cual se asimila a la de Distribución de Software de Berkeley(BSD) y a la del Instituto de Tecnología de Massachusetts (MIT), el cual permite a los usuarios realizar cualquier tipo de ajuste o modificación en el código fuente de la aplicación incluyendo la reventa de los binarios sin el código.

## 15.2 Características

A continuación se detallan las principales características de PostgreSQL que influyeron en la decisión de implementar el proyecto con el mismo

## 1.-MVCC

PostgreSQL implementa un sistema de control de concurrencias multiversión (Multi Version Concurrency Control), esto agrega una imagen del estado de la base de datos a cada transacción, permitiendo realizar transacciones eventualmente consistentes, es decir, garantiza que, si no se ha realizado modificaciones a los datos, eventualmente todos los accesos a dicha data tendrá el valor más actualizado, ofreciendo grandes ventajas de acceso y rendimiento.

## 2.-Hot-Standby

Gracias a que PostgreSQL no requiera del uso de bloqueo de lectura al momento de realizar una transacción de lectura en la base de datos se implementa un término llamado Hot-Standby, este término refiere a la acción de permitir que los clientes realicen búsquedas (únicamente de lecturas) en los servidores mientras se encuentre en modo de recuperación o espera, lo que permite hacer tareas de mantenimiento o recuperación sin bloquear completamente el sistema.

## 3.-Flexibilidad

PostgreSQL soporta múltiples lenguajes que pueden aplicarse para definir funciones personalizadas ,ofreciendo flexibilidad estos lenguajes son:

* PL/pgSQL
* PL/tcl
* PL/Perl
* PL/Python
* PL/PHP
* PL/Ruby
* PL/Java

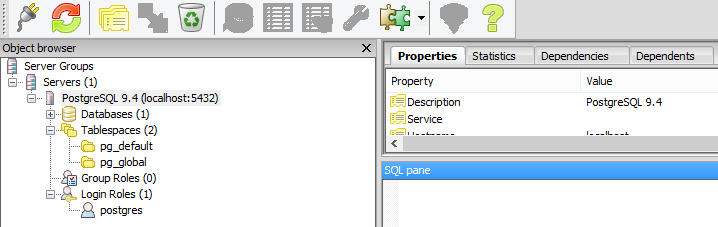
## 4.-Multiplataforma

PostgreSQL está disponible para múltiples plataformas y Sistemas Operativos y ofrece el código fuente del producto final según la plataforma desde el sitio oficial. Alguna de las builds oficiales liberados se encuentran en las siguientes plataformas:

* Mac OSX
* Windows
* Solaris
* Red Hat
* Debian
* Ubuntu

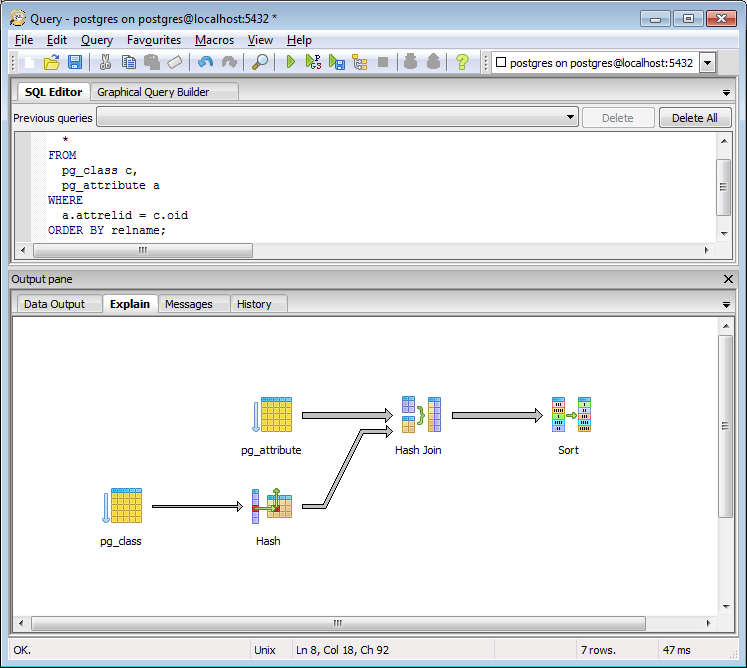
## 5.-pgAdmin

pgAdmin es una herramienta oficial de PostgreSQL para administrar las bases de datos creadas en dicho sistema manejador de forma fácil e intuitiva



### Figura 37 - Interfaz pgAdmin

Una de las herramientas más útiles implementadas en pgAdmin para el desarrollo del proyecto es la herramienta de Query Tool, dicha herramienta permite generar consultas de alto grado de complejidad de forma arbitraria, una vez generado dichos query, se da la opción de revisar el plan de ejecución para medir tiempos de respuesta y realizar las optimizaciones necesarias según la necesidad del proyecto, todo esto por medio de gráficas que facilita el entendimiento del plan de ejecución.



### Figura 38 - Query Tool, muestra gráfica del plan de ejecución

## 15.3 Ventajas

Tomando en cuenta las principales características de PostgreSQL se puede identificar las siguientes ventajas:

* Es código abierto.
* La velocidad de respuesta se mantiene al aumentar el tamaño de la base de datos, escenario que en otros sistemas manejadores empieza a reducir el rendimiento considerablemente.
* Proporciona estabilidad y confiabilidad.
* Tiene una gran capacidad de almacenamiento.
* Soporta gran número de peticiones simultáneas de a la base de datos manteniendo una consistencia eventual.
* El sistema se mantiene activo para acciones de lectura mientras se encuentre en estado de mantenimiento o recuperación.
* Puede operar sobre distintas plataformas.
* Provee un buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.
* Soporta los tipos de datos, cláusulas, funciones y comandos de tipo estándar SQL92/SQL99 y extendidos propios de PostgreSQL.

## 15.4 Desventajas

* En comparación con otros sistemas manejadores de base de datos, como por ejemplo MySQL, es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección.
* Cuenta con muchos foros oficiales de ayuda, pero no con una documentación de ayuda obligatoria.
* La sintaxis de algunos comandos o sentencias se diferencian de la línea de comandos de otros por lo que migrar a este sistema manejador requiere una curva de aprendizaje.
* Consuma una gran cantidad de recursos

## 15.5 Comparación de otros Sistemas Manejador de Base de Datos

Para determinar el porqué se elige PostgreSQL como el SMBD a utilizar en el proyecto a desarrollar se tomó en cuenta las diferencias entre los distintos SMBD objeto relacionales detallados a continuación:

## 15.5.1 PostgreSQL vs MySQL

En cuanto a usabilidad, en ambos sistemas es posible llevar a cabo tareas de todo tipo tanto en consola como a través de una interfaz gráfica, como lo son PgAdmin para PostgreSQL y MyAdmin para MySQL, sin embargo, la diferencia se encuentra en el enfoque que tiene cada sistema. En el caso de MySQL tiene un enfoque de trabajo rápido y sencilla, por lo que en ella las consultas suelen ser pequeñas y las modificaciones necesarias demoran poco tiempo, por lo que este sistema es ideal para base de datos pequeñas y medianas. PostgreSQL, por otra parte, brinda opciones más complejas para las consultas y modificaciones por lo que este sistema es ideal para base de datos grandes que contenga consultas complejas (donde participen múltiples tablas) y extensas.

En cuanto a rendimiento, como se mencionó anteriormente MySQL gana en proyectos pequeños o mediano porte cuya lógica no requiere de consultas complejas y de alto costo, en cambio, PostgreSQL es mejor empleado en proyectos de gran porte y tamaño, que requieran una base de datos robusta y accesible con consultas largas, complejas y frecuentes.

En cuanto a open source, ambos motores son libres, sin embargo, la licencia de MySQL es más restrictiva debido a que se encuentra bajo el poder de Oracle, quienes podrían tomar la decisión de privatizar al 100% en caso de que lo deseen.

## 15.5.2 PostgreSQL vs Oracle

En cuanto a usabilidad, Oracle tiene una mayor cantidad de APIs y enlaces entre el sistema o servicio web y base de datos en múltiples lenguajes de programación debido a su gran popularidad, esto da facilidad de integrar sistemas que ya estén desarrollados con una sistema de bases de datos de Oracle contando con un mayor soporte, en cambio, PostgreSQL al ser código abierto el soporte que presenta es lo que ofrece la comunidad ya sea en foros o grupos de discusión que se encuentren en la red.

En cuanto a rendimiento, Oracle posee una superioridad en el área, esto se debe a que Oracle es capaz de procesar una mayor cantidad de transacciones por segundo tanto en sistemas pequeños como en sistemas grandes y complejos, sin embargo, la licencia de Oracle no es gratuita por lo cual, para la necesidad de este proyecto utilizaremos el sistema gratuito que ofrece las características ideales para el desarrollo de la aplicación.

## 

# Capítulo 16

## Servicios Web

Para la aplicación a desarrollar es esencial contar con un Servicio Web que se encargue de recibir las peticiones de los clientes y dar respuesta de manera eficiente, utilizando estilos de arquitectura como lo son SOAP y REST, a continuación se detallarán estos últimos de forma comparativa para determinar cúal utilizar para el desarrollo.

## 16.1 Tendencias Principales Backend

Principalmente, explicaremos un poco la diferencia entre las dos tendencias principales de backend para traslado de datos entre cliente y servidor, en nuestro caso, laravel implementa REST asi que haremos la comparativa respecto al segundo principal recurso que se usa para hacer los intercambios de información entre cliente y servidor, SOAP.

SOAP, el segundo método más común para el intercambio de información entre el cliente y servidor; y REST no se pueden comparar directamente, ya que el primero es un protocolo (o al menos trata de serlo, por las reglas que requiere implementar) y el segundo es un estilo arquitectónico. Esta es probablemente una de las fuentes de confusión a su alrededor, ya que las personas tienden a llamar a REST cualquier API de HTTP que no sea SOAP.

Empujando un poco las cosas e intentando establecer una comparación, la principal diferencia entre SOAP y REST es el grado de acoplamiento entre las implementaciones de cliente y servidor. Un cliente SOAP funciona como una aplicación de escritorio personalizada, estrechamente acoplada al servidor. Existe un contrato rígido entre el cliente y el servidor, y se espera que todo se rompa si cualquiera de los lados cambia algo. Necesita actualizaciones constantes después de cualquier cambio, pero es más fácil determinar si se sigue el contrato.

Un cliente REST es más como un navegador. Es un cliente genérico que sabe cómo usar un protocolo y métodos estandarizados, y una aplicación debe encajar dentro de eso. No infringe los estándares de protocolo creando métodos adicionales, aprovecha los métodos estándar y crea las acciones con ellos en su tipo de medio. Si se hace bien, hay menos acoplamiento, y los cambios se pueden tratar con más gracia. Se supone que un cliente debe ingresar a un servicio REST sin conocimiento de la API, excepto por el punto de entrada y el tipo de medio. En SOAP, el cliente necesita conocimiento previo sobre todo lo que utilizará, o incluso no comenzará la interacción. Además, un cliente REST puede ampliarse mediante código a pedido suministrado por el propio servidor, el ejemplo clásico es el código JavaScript utilizado para impulsar la interacción con otro servicio en el lado del cliente.

Los siguientes puntos son cruciales para entender de qué se trata REST, y cómo difiere de SOAP:

* REST es independiente del protocolo. No está acoplado a HTTP. Al igual que puede seguir un enlace ftp en un sitio web, una aplicación REST puede usar cualquier protocolo para el que exista un esquema de URI estandarizado.
* REST no es una asignación de CRUD a los métodos HTTP. Lea esta respuesta para una explicación detallada sobre eso.
* REST es tan estandarizado como las partes que estás usando. La seguridad y la autenticación en HTTP están estandarizadas, por lo que es lo que se usa al hacer REST sobre HTTP.
* REST no es REST sin hipermedia y HATEOAS. Esto significa que un cliente solo conoce el URI del punto de entrada y se supone que los recursos deben devolver los enlaces que el cliente debe seguir. Esos sofisticados generadores de documentación que ofrecen patrones URI para todo lo que puede hacer en una API REST se pierden por completo. No solo están documentando algo que se supone que debe seguir el estándar, sino que cuando lo haces, estás conectando al cliente con un momento particular en la evolución de la API.

Con lo anterior en mente, se da cuenta de que, aunque REST podría no estar restringido a XML, para hacerlo correctamente con cualquier otro formato, pero se debe diseñar y estandarizar algún formato para sus enlaces. Los hipervínculos son estándar en XML, pero no en JSON. Hay proyectos de estándares para JSON, como HAL. (o la comunicación interna de laravel en nuestro caso).

## 16.2 ¿Porque REST y no SOAP?

Porque el framework de laravel, explicado anteriormente, implementa REST por defecto y es parte vital de su correcto comportamiento como stack para creación de aplicaciones. aunado a eso, la flexibilidad de la arquitectura para expandirse al dominio mobile (Si se aíslan los endpoints de API)

## 16.3 Ventajas y desventajas

* Formato de mensaje.
* Descubrimiento de servicio.
* El formato del mensaje es más fácil de entender.

El empaque de SOAP tanto para las solicitudes como para las respuestas es bastante pesado. Está el sobre SOAP que contiene un encabezado y una sección de cuerpo. El encabezado puede ser utilizado por varios filtros en la cadena de solicitud para realizar algún tipo de identificación, autorización, etc., sin embargo, XML es costoso de analizar, lo que produce una cierta penalización a la escalabilidad de su sistema. Solo cuánto depende de la capa de procesamiento SOAP en su pila.

El descubrimiento de servicios es donde probablemente tendrá más contienda. REST por su propia naturaleza proporciona puntos finales predecibles, y el contenido de la solicitud es una simple solicitud HTTP. El beneficio es que no hay gastos adicionales, y los usuarios finales pueden adivinar cómo hacer lo que necesitan una vez que entienden la estructura de URL de su sitio. Por supuesto, la gente ingenua y consciente de la seguridad lo verá como una debilidad. Después de todo, con SOAP, debe consumir un WSDL para saber cuáles son los puntos finales. Por supuesto, con SOAP se le dio el formato de mensaje completo para que pueda realizar ataques más específicos. Realizando una comparación con características específicas, tenemos:

## 16.4.1 Seguridad

Ninguno de los dos es intrínsecamente más seguro que el otro. Usan buenos principios de seguridad:

* Encriptan comunicaciones.
* Aseguran la autenticación y autorización a los usuarios antes de procesar.
* Buenos hábitos de codificación para evitar ataques directos.

## 16.4.2 Desempeño

Tanto el rendimiento bruto como la escalabilidad irán a REST debido a la solicitud que sigue protocolos HTTP simples. La mayoría de las pilas SOAP utilizan el análisis SAX (análisis basado en eventos) que mejora en gran medida la escalabilidad de las pilas de SOAP, pero hay un impacto mensurable en la sobrecarga. SOAP tiene la sobrecarga de procesamiento HTTP normal además de la tara de análisis XML. REST solo tiene la sobrecarga de procesamiento de HTTP.

## 16.4.3 Complejidad

Desde la perspectiva del sistema, REST gana. Hay menos piezas móviles, una cadena de solicitud más sencilla, etc. Eso significa que es más fácil de hacer confiable.

Desde la perspectiva del programador, SOAP puede ganar si el IDE o el marco que está utilizando proporciona un buen soporte para él. Esencialmente, con REST usted tiene la responsabilidad de realizar el trabajo de preproceso (autenticación / autorización / etc.) mientras que con SOAP gran parte de eso se puede lograr con una cadena de procesamiento conectable.

**Capítulo 17**

## Test Driven Development (TDD)

Test Driven Development es una práctica de ingeniería del software que cambia la forma en que se realiza el desarrollo de un proyecto, ya que en dicha práctica el proceso inicia con la definición del requerimiento y posterior a ello generar las pruebas unitarias (o casos de pruebas) antes de iniciar la escritura del código fuente, sin embargo, esto se hace en pequeñas partes, es decir, se genera una prueba unitaria para una función específica y luego se desarrolla, por lo que las funciones no se desarrollan hasta que exista una prueba unitaria que falle su ejecución en la versión actual del proyecto.

## 17.1 ¿Por qué usar TDD?

El objetivo de la práctica de Test Driven Development es evitar generar código innecesario. En general, los desarrollos se realizan primero generando el código y posterior a esto realizan las pruebas necesarias para validar el correcto funcionamiento del mismo, esto en general permite detectar si existen bugs, pero no da suficiente información si el código carece de ellos, por lo que la única manera efectiva de probar el nivel de confianza de una aplicación se logra dando una prueba convincente de su correctitud.

Esta práctica permite realizar pruebas mano a mano con el desarrollo de la aplicación, por lo que solo se genera código cuando alguna de las pruebas falla en ser procesado (sea por un requerimiento nuevo o un bug encontrado). Esto reduce la cantidad de defectos en los proyectos, mejora la calidad del software y aumenta la productividad en el equipo de desarrollo.

## 17.2 ¿Cómo se instala?

Test Driven Development no requiere instalación, puesto a que es una práctica de ingeniería de software que todo el equipo de desarrollo debe disponerse a emplear, sin embargo, es importante aplicar dicha práctica utilizando herramientas y frameworks lo suficientemente desarrollados que permitan generar un set de pruebas unitarias y probarlas sobre la aplicación.

## 17.3 ¿Cómo se utiliza?

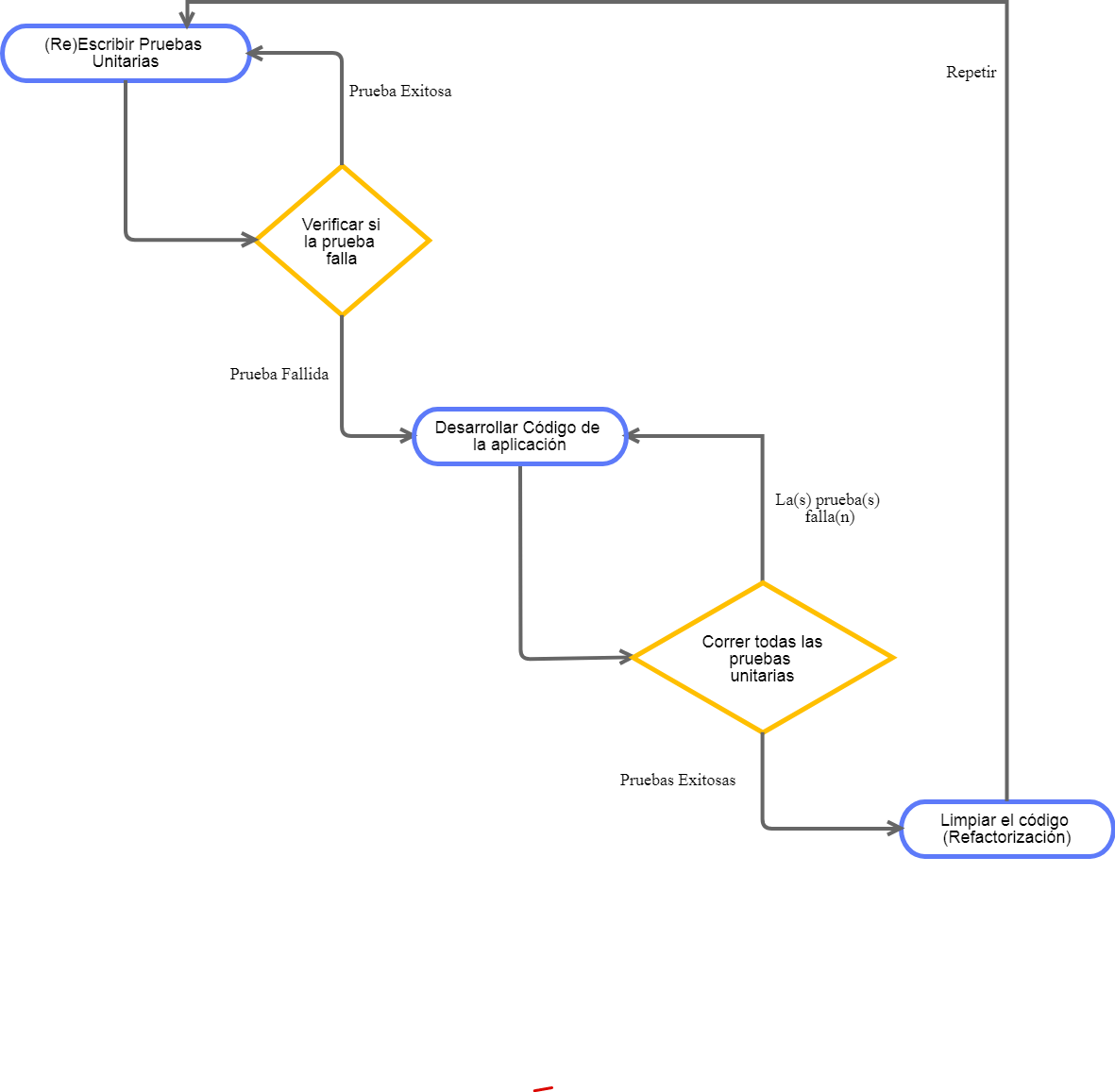
1.-Antes de escribir algún tipo de código primero debe realizar pruebas unitarias automáticas que será ejecutado por la aplicación, tomando en cuenta las posibles entradas, errores y salidas esperadas según sea el caso de la prueba.

2.-Correr el set de pruebas unitarias creadas anteriormente en la aplicación, ésta indudablemente fallará, indicando que es necesario realizar modificaciones a la aplicación.

3.-Se inicia el desarrollo de la nueva funcionalidad sobre la aplicación. Al tener el set de pruebas unitarias, mientras el código desarrollado no pase dichas pruebas de forma exitosa significa que no está lista y requiere ajustes. Se repite dicha iteración hasta que todas las pruebas se procesen satisfactoriamente

4.-Una vez que la aplicación pase todas las pruebas satisfactoriamente se procede a realizar refactorización para optimizaciones necesarias.

5.-Se repite el proceso ante nuevos requerimientos o proyectos.



**Figura 39 - Diagrama del proceso de TDD**

## 17.4 Depuración de errores en TDD

La depuración de errores ocurren de forma nativa utilizando la práctica de Test Driven Development debido a que cada cambio o desarrollo que se integra a una aplicación debe procesar de forma satisfactoria todas las pruebas unitarias para integrarlo a la aplicación en un ambiente productivo. Además, conociendo exactamente qué parte de la prueba ha fallado en procesarse, permite identificar en qué parte de los cambios se ha generado bugs en la aplicación.

## 17.5 Ventajas

Al momento de generar código nuevo de la aplicación se puede dar por garantizado que no queda trabajo por hacer al finalizar la funcionalidad desarrollada, gracias al set de pruebas que debe ser procesado satisfactoriamente.

Debido a la naturaleza del desarrollo utilizando esta práctica, permite al desarrollador enfocarse en una única funcionalidad, es decir, al validar que una prueba ha fallado en procesarse, el desarrollador se enfoca únicamente en hacer que dicha prueba se procese exitosamente, mejorando la productividad del equipo de desarrollo. Adicionalmente, esto garantiza que el trabajo final tendrá un resultado con una reducida cantidad de bugs, garantizando la calidad del software al momento de salir al ambiente productivo.

Finalmente, las pruebas son una fuente importante de documentación para un desarrollador, ya que ante la duda de como funciona una librería o framework se puede realizar un chequeo de la documentación de las pruebas, esto da información clave de cómo utilizar una clase, las entradas y salidas esperadas y el costo de añadir dicha funcionalidad.

## 17.6 Desventajas

Dependiendo de la complejidad de la funcionalidad que se desea desarrollar, se debe invertir una considerable cantidad de tiempo planificando las pruebas unitarias, para sistemas sencillos, dicho tiempo es casi irrelevante, sin embargo, para sistemas con un grado de complejidad alto, las pruebas unitarias son más difíciles de calcular, debido a que pueden existir varias entradas, respuesta a errores y salidas, y todas ellas deben ser generadas y documentadas.

Cuando el diseño de la aplicación no está clara desde el principio del desarrollo, puede generar que se requieran rediseñar las funcionalidades desarrolladas y probadas con anterioridad causando que se deba rediseñar todo el set de pruebas.

# Referencias Bibliográficas y Digitales

Bustamante, J. (2015). RECURSOS HUMANOS. Recuperado en Febrero 2018 de: <http://www.academia.edu/12139844/RECURSOS_HUMANOS_trabajo_monografico>

Laravel. (2018). Laravel Documentation. Recuperado en Febrero 2018 de: <https://laravel3.veliovgroup.com/docs>

Reicek. (2015). Laravel, el mejor framework en PHP. Recuperado en Febrero 2018 de: <https://platzi.com/blog/laravel-framework-php/>

PHP-FIG. (2018). PSR-4: Autoloader. Recuperado en Marzo 2018 de: <https://www.php-fig.org/psr/psr-4/>

Barnes, E. (2015). Laravel 5. Recuperado en Marzo de 2018 de: <https://laravel-news.com/laravel-5>

Wodehouse, C. (2015). What is a Framework?. Recuperado en Marzo de 2018 de: <https://www.upwork.com/hiring/development/understanding-software-frameworks/>

Battle, L & Chessman, L. (2012). Designing Configurable and Customizable Applications. Recuperado en Enero de 2018 de: <http://www.designforcontext.com/insights/designing-configurable-and-customizable-applications>

OpensourceCMS. (2014). Content Management Systems vs. Frameworks. Recuperado en Enero 2018 de: <https://www.opensourcecms.com/content-management-systems-vs-frameworks/>

Webb, D. (2016). Is Framework or CMS Better for Web Development?. Recuperado en Enero 2018 de: <https://www.b2interactive.com/blog/framework-vs-cms/>

Reicek. (2015). Qué es PostgreSQL y cuáles son sus ventajas. Recuperado en Marzo 2018 de: <https://platzi.com/blog/que-es-postgresql/>

Santiago. (2018). PostgreSQL vs MySQL. Recuperado en Marzo 2018 de: <https://guiadev.com/postgresql-vs-mysql/>

BairesDev. (2018). Top 1% Talent. Recuperado en Mayo 2018 de: <https://www.bairesdev.com/top-talent>

Crossover. (2018). The Qualification Process. Recuperado en Mayo 2018 de: <https://www.crossover.com/for-candidates/#qualification-process>

Google. (2018). How we hire. Recuperado en Mayo 2018 de: <https://careers.google.com/how-we-hire/>

Feloni, R. (2016). Facebook's head of recruiting explains the company's top 3 approaches to finding exceptional employees. Recuperado en Mayo 2018 de: <http://www.businessinsider.com/how-facebook-finds-exceptional-employees-2016-2#facebook-looks-for-talent-and-cultural-fit-3>