

---

## Contents

<b>1</b>	<b>Planificación</b>	<b>1</b>
1.1	Objetivos . . . . .	1
1.2	Funcionalidades . . . . .	1
1.3	Cronograma . . . . .	1
1.4	Requisitos por versiones . . . . .	2
<b>2</b>	<b>Análisis/Diseño</b>	<b>2</b>
2.1	Base de datos . . . . .	2
2.2	Componentes principales del Sistema . . . . .	4

## 1 Planificación

### 1.1 Objetivos

La aplicación `page_alert` permite a usuarios conectarse a un sistema que se encargará de **monitorear** páginas web y de **avisarle** en caso de que alguna de las reglas (previamente definidas por el usuario) se cumpla.

### 1.2 Funcionalidades

- Login de usuario.
- Lista de páginas y reglas de usuario.
- Ayuda 'interactiva' para la creación de reglas.
- CRUD de páginas y reglas.
- Monitoreo recurrente de las páginas.
- Notificación por la vía configurado de los cambios encontrados.

### 1.3 Cronograma

---

Fase	Fechas	Hitos
------	--------	-------

---

| Análisis y diseño | 18-21 Jul | [ ] Diagramas UML (BD, JAVA)

## 1.4 Requisitos por versiones

### 1.4.1 0.1.0

- **Notificaciones**

- Email
- Push

- **Reglas de Monitoreo:**

- Toda la página: bodyHash
- Elemento/s concretos: [ 'css-selector' ]
- Precio (Precisa de ayuda interactiva al usuario)

## 2 Análisis/Diseño

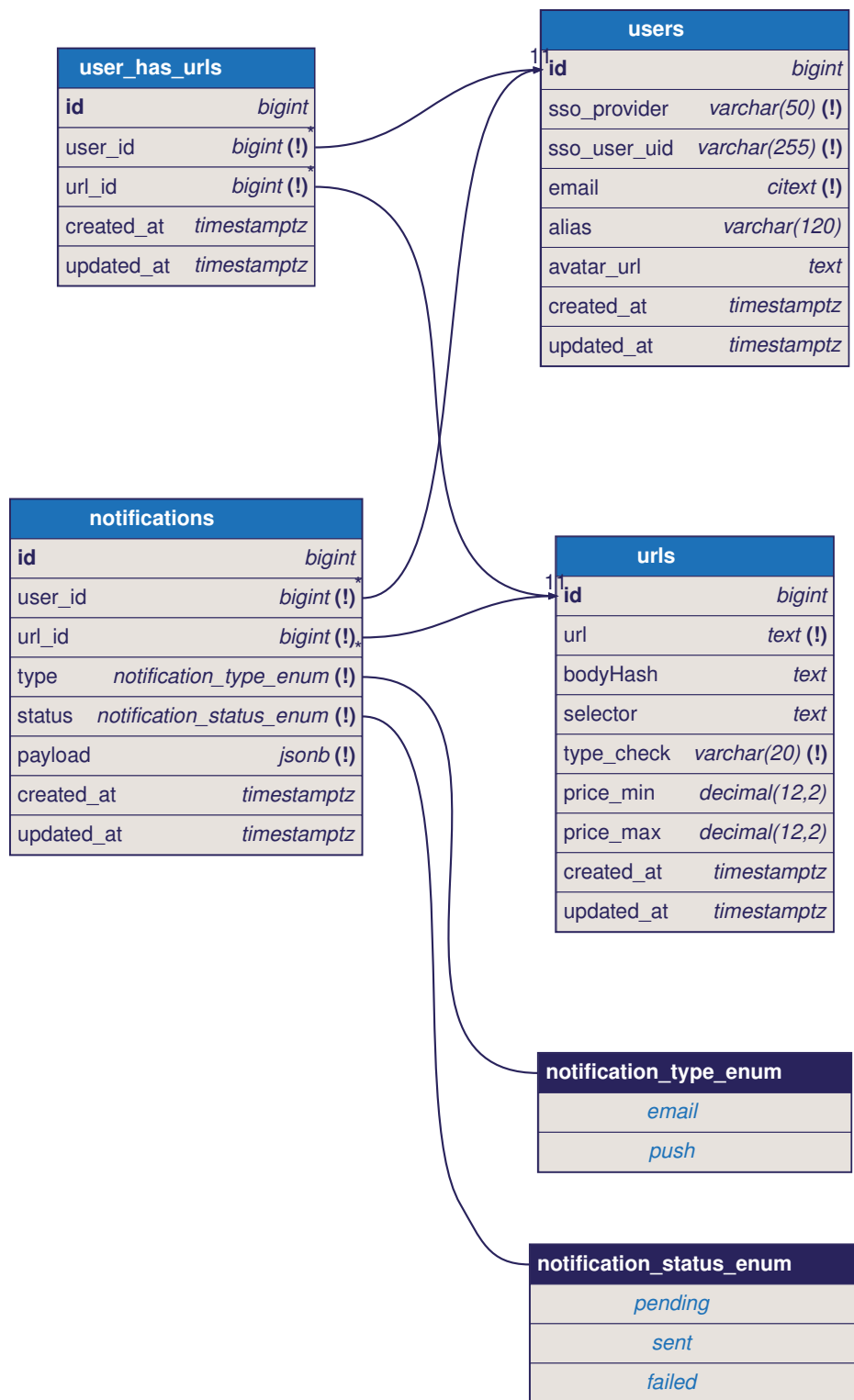
- Base de datos
- Componentes principales del Sistema

Herramientas que usaremos:

- Diagrama de Clases y flujo: PLANTUML (Integración automática con código)
- Diagramas de BD: dbdiagram.io (exportación de SQL)

### 2.1 Base de datos

Tras un análisis de las diferentes opciones de gestores de Base de Datos. Me decanto por **PostgreSQL** porque se asemeja mucho a MYSQL pero proporciona funcionalidades extra: colas de mensajes, crons, criptografía ...



**Figure 1:** Esquema BD

---

## 2.2 Componentes principales del Sistema

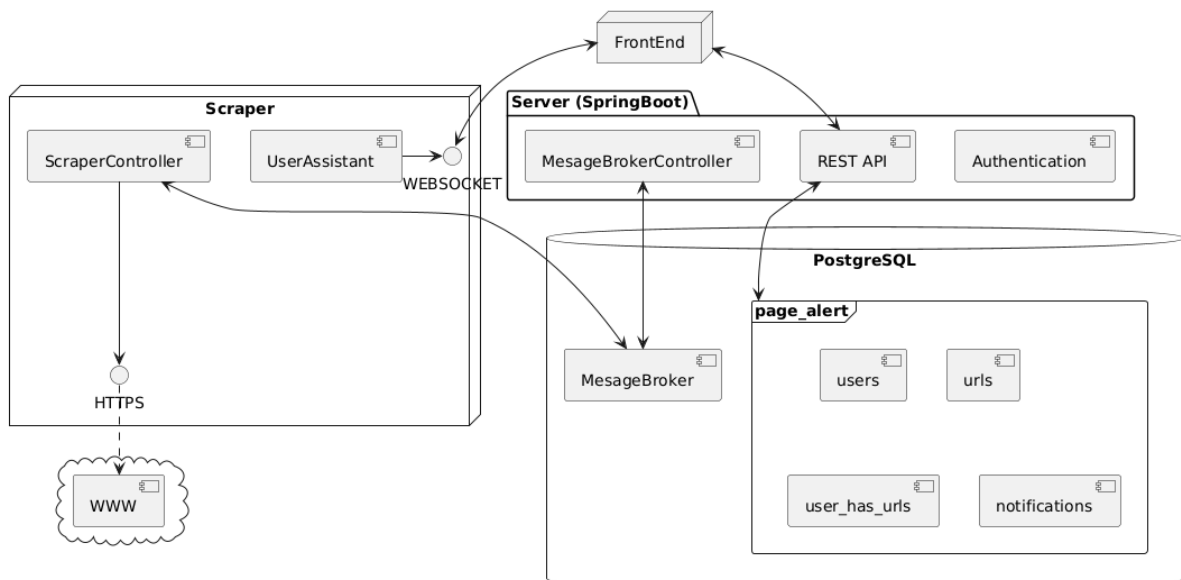
La Aplicación tendrá 4 componentes principales:

- **Scraper:**

- Encargado de actualizar el estado de las **URLS** en función de las reglas de monitoreo.
- Mediante WebSocket se encarga de validar con el usuario que las **reglas de monitoreo sean validas** y que el sistema es capaz de procesarlas.

- **Server:**

- Responsable de **autenticar** y manejar correctamente la **sesión del usuario**.
- Permite a los usuarios modificar sus datos de usuario y hacer un CRUD de URLS exponiendo una **API**.



**Figure 2:** Componentes principales