PIM: Design HTML structure and CSS styling for the appearance and behaviour of the application

Design Sketch Of The Login Page
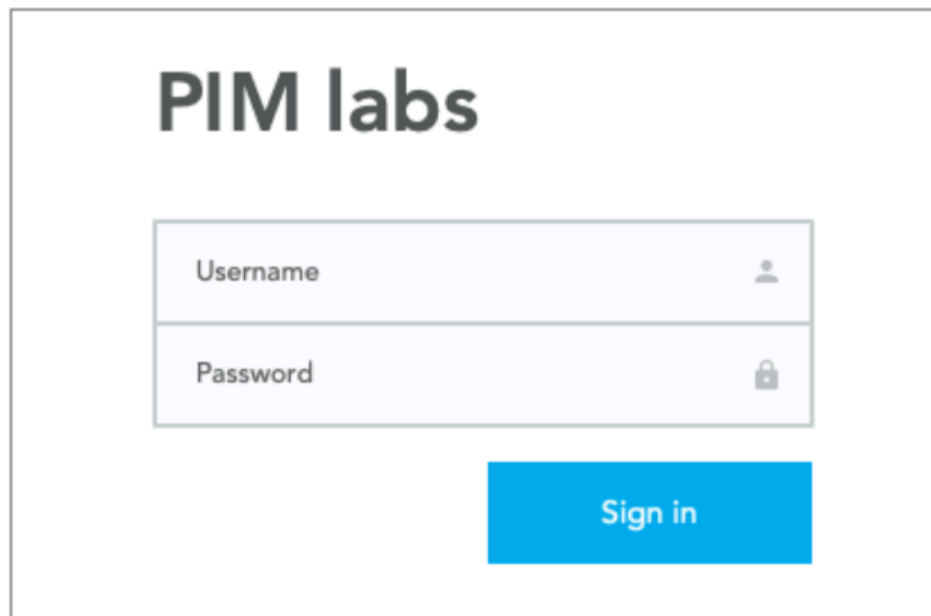

Fig. 6 PIM login view

Top-Level Structure

<main class="login-container"> — *Outer container that wraps and centers all login page elements.*

Logo Area

<div class="logo"> — *Logo section that displays the brand identity.*

- **Image Logo** — *<img src="logo.png" alt="App Logo"> — main branding graphic.*
- **Optional Text Logo** — *<h1> — used if no image is available.*

Title Section

<h2 class="login-title"> — *Page heading that communicates the form's purpose ("Login").*

Form Section

<form class="login-form"> — *Login form that contains username/password inputs and actions.*

Username Field

<div class="form-group">

- **Label** — *<label for="username">Username</label> — describes the input's purpose.*

- **Input** — *<input type="text" id="username" name="username"> — accepts username.*

Password Field

<div class="form-group">

- **Label** — *<label for="password">Password</label> — describes the input's purpose.*
- **Input** — *<input type="password" id="password" name="password"> — hides typed characters for privacy.*

Login Button

<button class="btn-login" type="submit"> *— Primary action button that submits the login form.*

Extra Links Section

<div class="extra-links"> *— Supplementary navigation links.*

- **Forgot Password?** — *<a href="/forgot-password"> — link for password recovery.*
- **Sign up** — *<a href="/signup"> — link to registration page.*

```
<main.login-container>
    <div.logo>
        <img> (logo image)
        <!-- or -->
        <h1> (text logo)
    <h2.login-title> ("Login")
    <form.login-form>
        <div.form-group>
            <label> ("Username")
            <input[type="text"]> (username input)
        <div.form-group>
            <label> ("Password")
            <input[type="password"]> (password input)
        <button.btn-login> ("Login")
        <div.extra-links>
            <a> ("Forgot Password?")
            <a> ("Sign up")
```
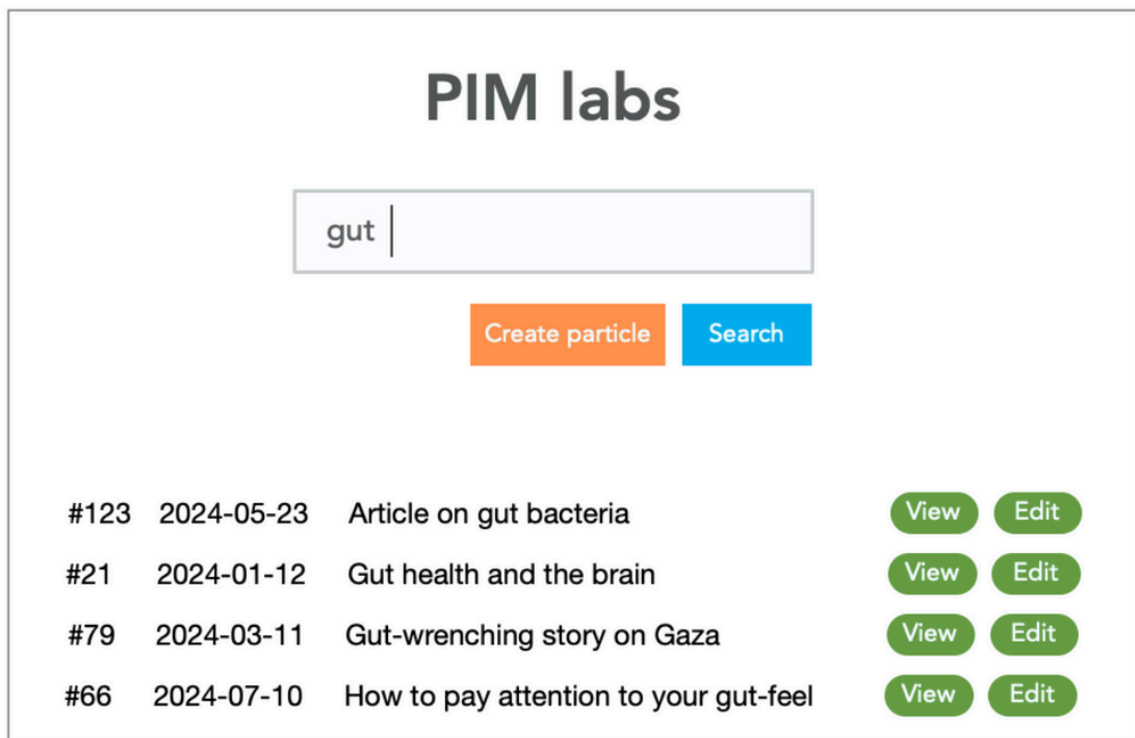
Design Sketch Of The PIM Search Page

Fig. 7 PIM search page

Top-Level Structure

- **<main class="container">**
- *Page wrapper — holds all visible content in a centered, max-width layout.*

Header Area

- **<h1 class="app-title">PIM labs</h1>**
- *App title — branding and page heading.*
- **<form id="quickCreate" class="row">**
- *Quick create/search form — groups search input & action buttons.*

Search Controls

- **<input id="q" class="input">**
- *Search field — type keyword or new title.*
- **<button class="btn btn-create">Create particle</button>**
- *Creates a new article if search yields no match.*
- **<button class="btn btn-search">Search</button>**
- *Executes the search request.*

Results List

- **<section id="results">**
- *Search results container.*
- **<ul class="results">**
- *List of search results.*

Result Card

- **<li class="result-card">**
- *One search result row — contains ID, date, title, and action buttons.*

- **`<span class="id">#123</span>`**
- *Unique article ID.*
- **`<time class="date">2024-05-23</time>`**
- *Article creation date.*
- **`<button class="title-link">`**
- *Clickable title — expands/collapses preview.*
- **`<a class="btn pill view">View</a>`**
- *Opens full article view.*
- **`<a class="btn pill edit">Edit</a>`**
- *Opens editor to modify article.*

Expanded Preview

- **`<div class="preview">`**
- *Inline article snippet — shows content without leaving search page.*

Overall Hierarchy

```
<main.container>
  <header>
    <h1.app-title>
    <form#quickCreate.row>
      <input#q.input>
      <button.btn.btn-create>
      <button.btn.btn-search>
  <section#results>
    <ul.results>
      <li.result-card>*
        <span.id>
        <time.date>
        <button.title-link>
        <a.btn.pill.view>
        <a.btn.pill.edit>
        <div.preview.markdown> (visible when .result-card.expanded)
```
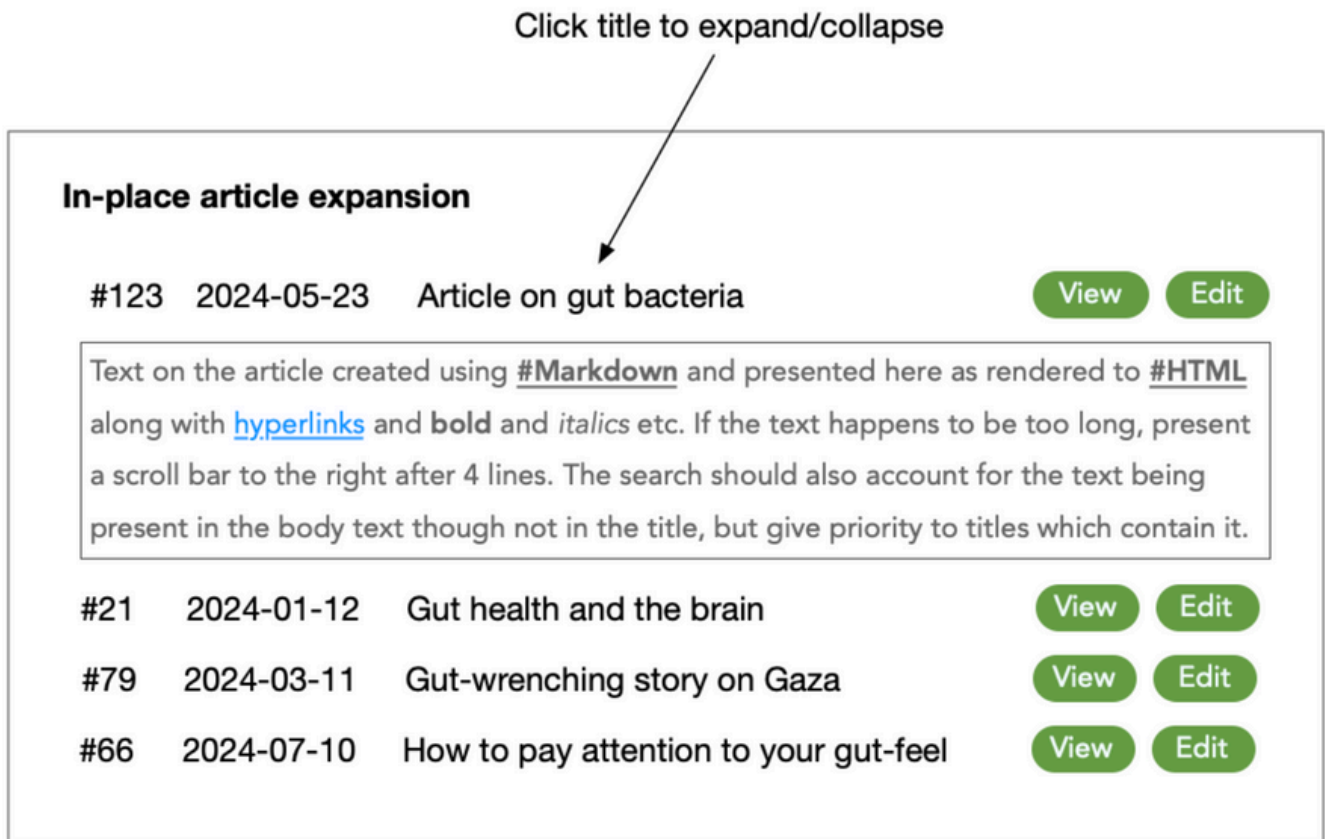
Design Sketch Of The PIM Search Page

Fig. 8 PIM search results list and in-place viewer

Interaction State

- **<li class="result-card [expanded]">**
- Result row supports a toggled state. When .expanded is present, the preview is visible; otherwise it's hidden.

Title Toggle

- **<button class="title-link" aria-expanded="false" aria-controls="preview-123">**
- Interactive control for expand/collapse. aria-expanded reflects state (true/false), and aria-controls points to the preview panel.

Preview Panel

- **<div id="preview-123" class="preview" hidden>**
- Collapsible detail region rendered in-line below the row's main fields. Initially hidden; becomes visible when the row is expanded.

Live Status

- **<p id="statusMsg" class="muted" aria-live="polite">**
- Announces status like "Expanded" / "Collapsed" for accessibility (optional but recommended).

Default Hidden Rule (CSS Hook)

- **.preview { display: none; } and/or hidden attribute**

- Ensures the preview is not visible until expanded.

Overall Hierarchy

```
<li class="result-card [expanded]">          ← Result row container (state: collapsed or expanded)

├─ <span class="id">#123</span>              ← Metadata: unique article ID

├─ <time class="date" datetime="2024-05-23">2024-05-23</time>
                                             ← Metadata: article creation date

├─ <button class="title-link"
           aria-expanded="false"
           aria-controls="preview-123"
           type="button">
       Article title here
   </button>
                                             ← Title toggle: expands/collapses preview

├─ <a class="btn pill view" href="#">View</a>
                                             ← Action: open full article view

├─ <a class="btn pill edit" href="#">Edit</a>
                                             ← Action: open article editor

└─ <div id="preview-123"
        class="preview markdown"
        role="region"
        aria-labelledby="title-123"
        hidden>
     <article class="particle">
       <header>...</header>
       <section class="body">Inline snippet content...</section>
       <footer class="tags">#tag1 #tag2</footer>
     </article>
   </div>
                                             ← Expandable content area (hidden unless .expanded is present)
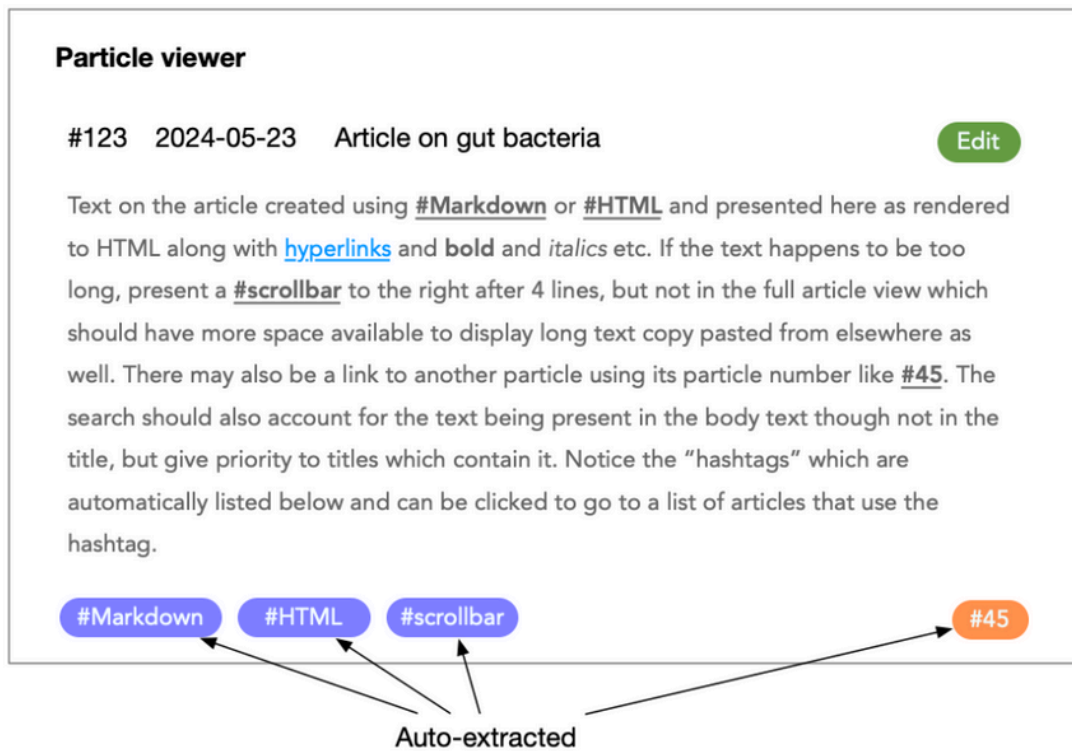```

## Design Sketch Of The PIM Particle Viewer

*Fig. 9* PIM particle viewer

## Main Container

**<article class="particle-viewer">** — *Semantic container for the complete particle.*

## Header Section

**<header class="particle-header">** — Contains metadata and the title.

- **Title** — *<h1 class="particle-title"> — Displays "Particle viewer".*
- **Metadata Group** — *<div class="particle-metadata"> — Bundles particle info:*
  - **Particle ID** — *<span class="particle-id"> — Unique identifier.*
  - **Date** — *<span class="particle-date"> — Publication date.*
  - **Article Title** — *<span class="article-title"> — Headline of the particle.*
  - **Edit Button** — *<button class="edit-btn"> — Opens editing mode.*

## Content Section

**<main class="particle-content">** — *Displays the full particle content.*

- **Article Body** — *<div class="article-body"> — Contains multiple <p> elements.*
  - *Supports **rich HTML formatting** (links, bold, italics embedded in paragraphs).*

## Footer Section

**<footer class="particle-footer">** — *Contains related information and tags.*

- **Hashtags Section** — *<div class="hashtags-section"> — Multiple hashtag <span> elements.*
- **Related Links** — *<div class="related-links"> — Contains <a> anchors to other particles.*

```
<article.particle-viewer>
    <header.particle-header>
        <h1.particle-title> ("Particle viewer")
        <div.particle-metadata>
            <span.particle-id> (particle ID)
            <span.particle-date> (publication date)
            <span.article-title> (article title)
            <button.edit-btn> ("Edit")
    <main.particle-content>
        <div.article-body>
            <p> (paragraph text, with links/bold/italics)
            <p> ...
    <footer.particle-footer>
        <div.hashtags-section>
            <span> (hashtag)
            <span> ...
        <div.related-links>
            <a> (related particle link)
            <a> ...
```
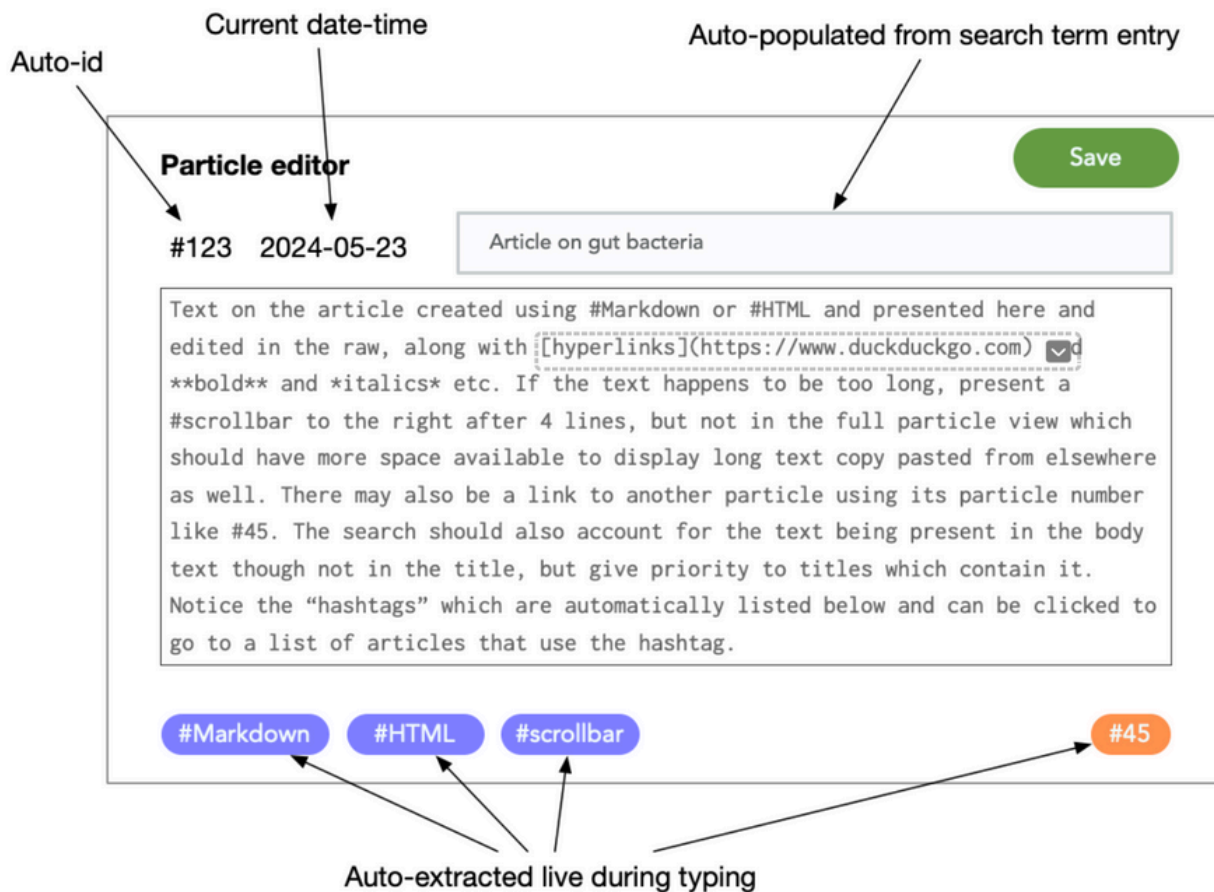
## Design Sketch Of The PIM Particle Editor

Fig. 10 PIM particle editor

## Top-Level Structure

**<main class="container particle-editor">**

- **Role**: Main content area for the editor.
- **Function**: Holds all editor controls and fields in a centered layout.

## Header Section

**<header>**

**<span class="id">#123</span>**

**<time class="date" datetime="2024-05-23">2024-05-23</time>**

**</header>**

- **Auto-ID**: Unique particle identifier (read-only, system-generated).
- **Current date-time**: Automatically filled based on creation or last edit date.

## Title Field

**<input class="title-input" type="text" value="Article on gut bacteria">**

- **Function**: Editable title of the particle.
- **Behavior**: Auto-populated with the title from the search term or existing particle title.

## Save Button

**<button class="btn btn-save">Save</button>**

- **Function**: Commits changes to the particle.
- **Placement**: Positioned next to the title for quick access.

## Body Text Area

**<textarea class="body-editor">...</textarea>**

- **Function**: Markdown/HTML-supported text field for the particle body.
- **Features**:
  - Supports formatting (**bold**, *italics*, #hashtags, hyperlinks).
  - Scrollbar appears after a set number of lines.
  - Auto-detection of hashtags within the text.

## Hashtag List

**<div class="tag-list">**

**<span class="tag">#Markdown</span>**

**<span class="tag">#HTML</span>**

**<span class="tag">#scrollbar</span>**

**<span class="tag orange">#45</span>**

**</div>**

- **Function**: Displays hashtags detected live as the user types in the body.
- **Behavior**: Tags are clickable — clicking opens a list of other particles using the same tag.
- **Color coding**: Example — special highlight for certain tags (e.g., #45 in orange).