Other themes

HTML5 features

- Canvas - A complete drawing and animation API
- Geolocation
- Local storage - Save key/value pairs to a client-side data store implemented by browser
- WebSQL storage - A set of APIs to manipulate client-side databases using SQL
- Offline Apps - Run a web application in offline mode, disconnected from Internet
- WebSockets - Opens a persistent connection to the server
- WebWorkers - Provide “threads” for JavaScript execution

SEO – Search Engine Optimization

For a short intro, see

https://www.slideshare.net/bernardcharlebois/search-engine-optimization-seo-61163208

For a full discussione, see

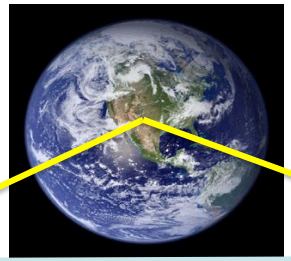
https://developers.google.com/search/docs/beginner/seo-starter-guide?hl=en

Server side frameworks

- ORM (Object Relational Mapping): Hibernate, JPA (https://hibernate.org/orm/documentation/6.0/)
- EJB (Enterprise Java Beans) (https://www.tutorialspoint.com/ejb/index.htm)
- Struts (https://struts.apache.org/index.html)
- Spring (https://spring.io/)
- ASP.NET (https://docs.microsoft.com/en-us/aspnet)

For

ORM - DAO



WORLD

MODEL

UML

ORM

ERA

ARCHITECTURE

DAO

DB

Actual storage

FS

platforms

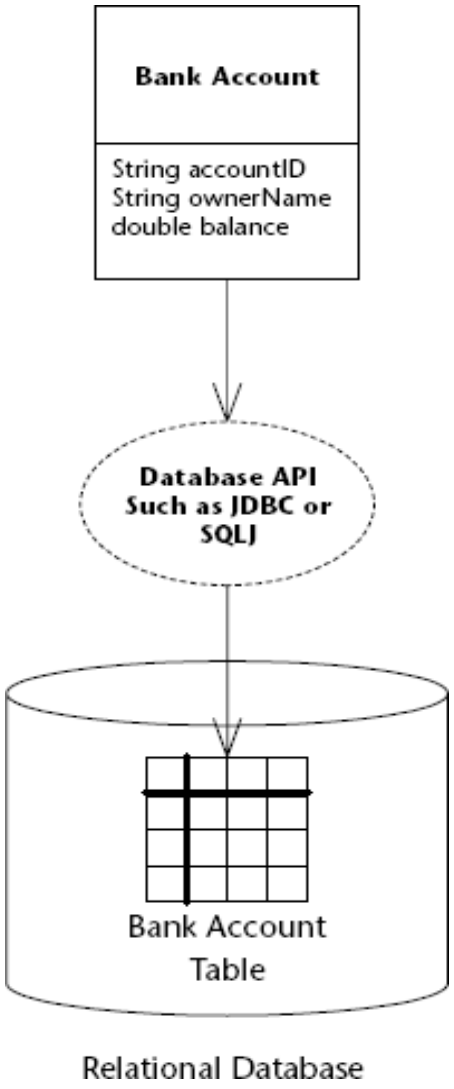
Data

Object

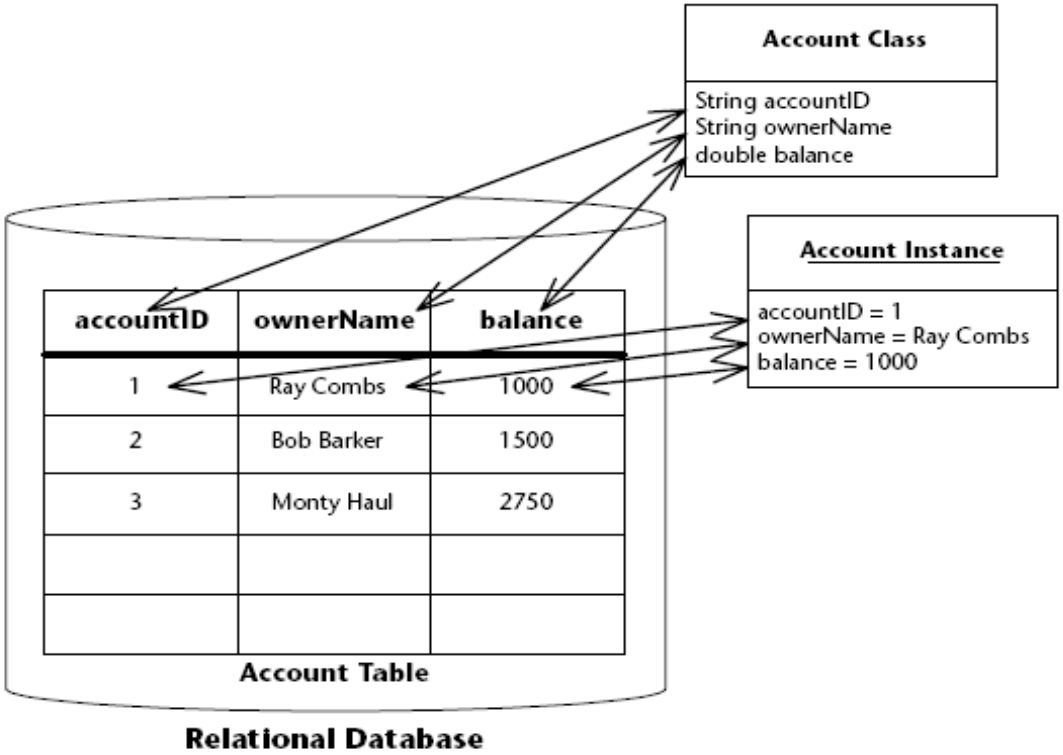
temp

tools

ORM



Object-Relational Mapping is NOT serialization!



New standards

- **WebAssembly** (Wasm) is an open standard that defines a portable binary-code format for executable programs, and a corresponding textual assembly language, as well as interfaces for facilitating interactions between such programs and their host environment.

https://developer.mozilla.org/en-US/docs/WebAssembly

For

UX

- The user experience (UX or UE) is how a user interacts with and experiences a product, system or service. It includes a person's perceptions of utility, ease of use, and efficiency.

For

WCAG (content accessibility guidelines)

- Guideline 1: Provide equivalent alternatives to auditory and visual content
- Guideline 2: Don't rely on colour alone
- Guideline 3: Use markup and style sheets, and do so properly
- Guideline 4: Clarify natural language usage
- Guideline 5: Create tables that transform gracefully
- Guideline 6: Ensure that pages featuring new technologies transform gracefully
- Guideline 7: Ensure user control of time sensitive content changes
- Guideline 8: Ensure direct accessibility of embedded user interfaces
- Guideline 9: Design for device independence
- Guideline 10: User interim solutions
- Guideline 11: Use W3C technologies and guidelines

For

WCAG

- Guideline 12: Provide context and orientation information
- Guideline 13: Provide clear navigation mechanisms
- Guideline 14: Ensure that documents are clear and simple

…

Guideline 2.1: Make all functionality available from a keyboard.

Guideline 2.2: Provide users enough time to read and use content.

….

Guideline 2.4: Provide ways to help users navigate, find content, and determine where they are.

…

Guideline 3.2: Make web pages appear and operate in predictable ways.

https://en.wikipedia.org/wiki/Web_Content_Accessibility_Guidelines

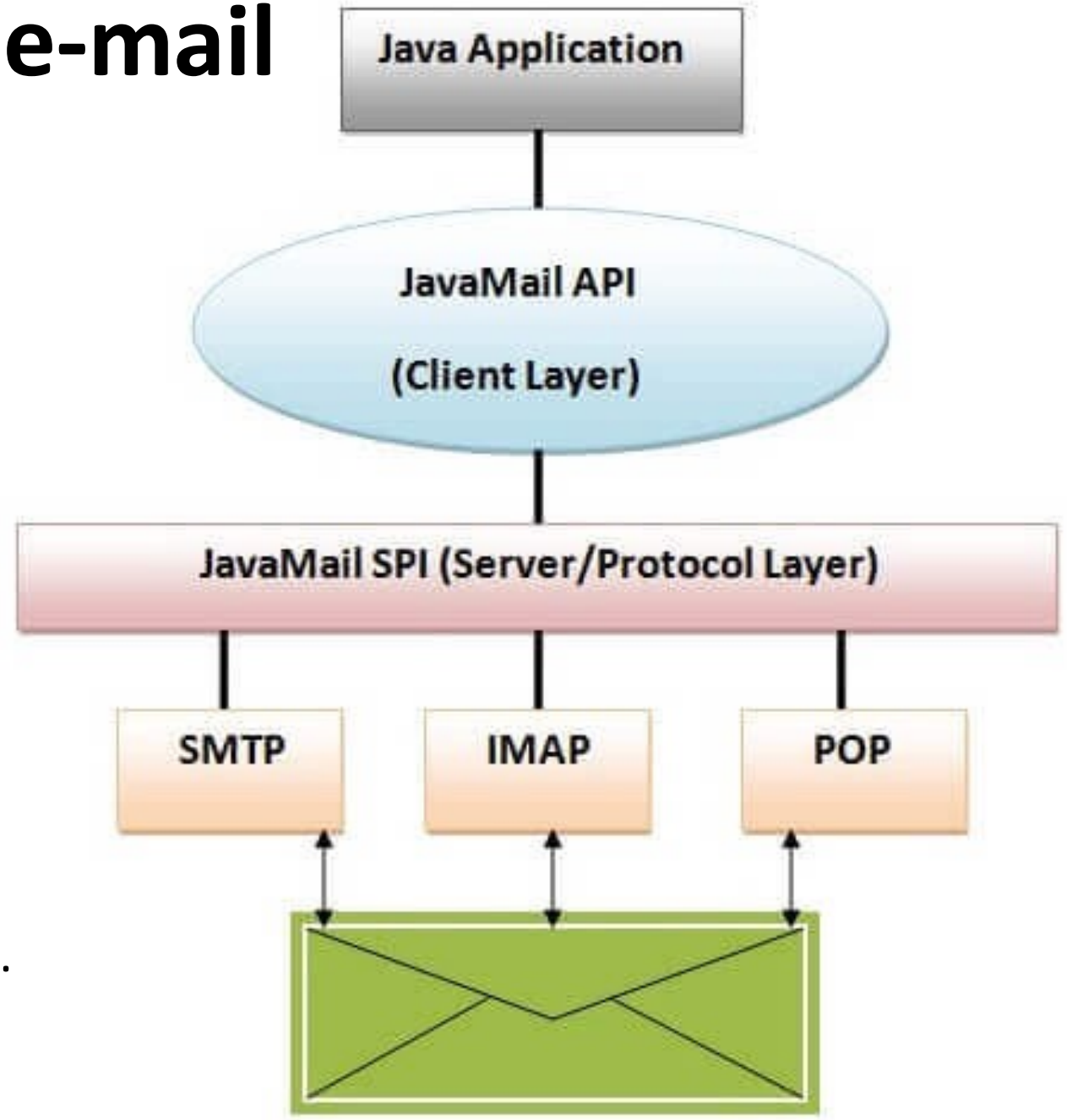
For

Enabling features on your web server

Send e-mail

How to send e-mail

- Install and use any SMTP server such as Postcast server, Apache James server, cmail server etc. (or)
- Use the SMTP server provided by the host provider e.g. my SMTP server is mail.javatpoint.com (or)
- Use the SMTP Server provided by other companies e.g. gmail etc.



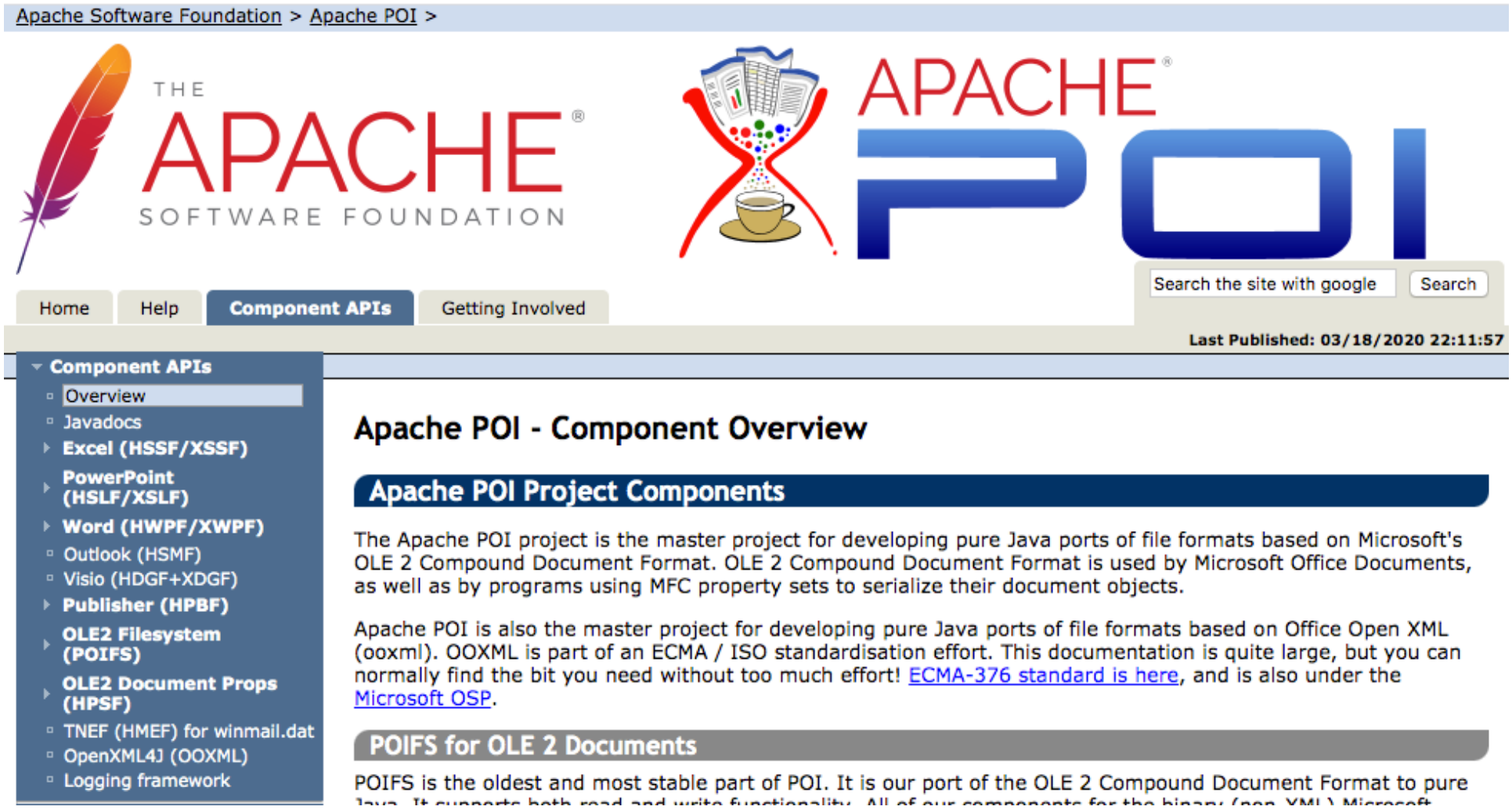
Some tutorials:

https://www.javatpoint.com/java-mail-api-tutorial

https://www.tutorialspoint.com/java/java_sending_email.htm

Enable exporting data from your site

Apache POI


Apache Software Foundation > Apache POI >
THE APACHE SOFTWARE FOUNDATION
APACHE POI
Search the site with google
Search
Home
Help
Component APIs
Getting Involved
Last Published: 03/18/2020 22:11:57
Component APIs
Overview
Javadocs
Excel (HSSF/XSSF)
PowerPoint (HSLF/XSLF)
Word (HWPF/XWPF)
Outlook (HSMF)
Visio (HDGF+XDGF)
Publisher (HPBF)
OLE2 Filesystem (POIFS)
OLE2 Document Props (HPSF)
TNEF (HMEF) for winmail.dat
OpenXML4J (OOXML)
Logging framework
Apache POI - Component Overview
Apache POI Project Components
The Apache POI project is the master project for developing pure Java ports of file formats based on Microsoft's OLE 2 Compound Document Format. OLE 2 Compound Document Format is used by Microsoft Office Documents, as well as by programs using MFC property sets to serialize their document objects.
Apache POI is also the master project for developing pure Java ports of file formats based on Office Open XML (ooxml). OOXML is part of an ECMA / ISO standardisation effort. This documentation is quite large, but you can normally find the bit you need without too much effort! ECMA-376 standard is here, and is also under the Microsoft OSP.
POIFS for OLE 2 Documents
POIFS is the oldest and most stable part of POI. It is our port of the OLE 2 Compound Document Format to pure

https://poi.apache.org/components/

writing excel documents

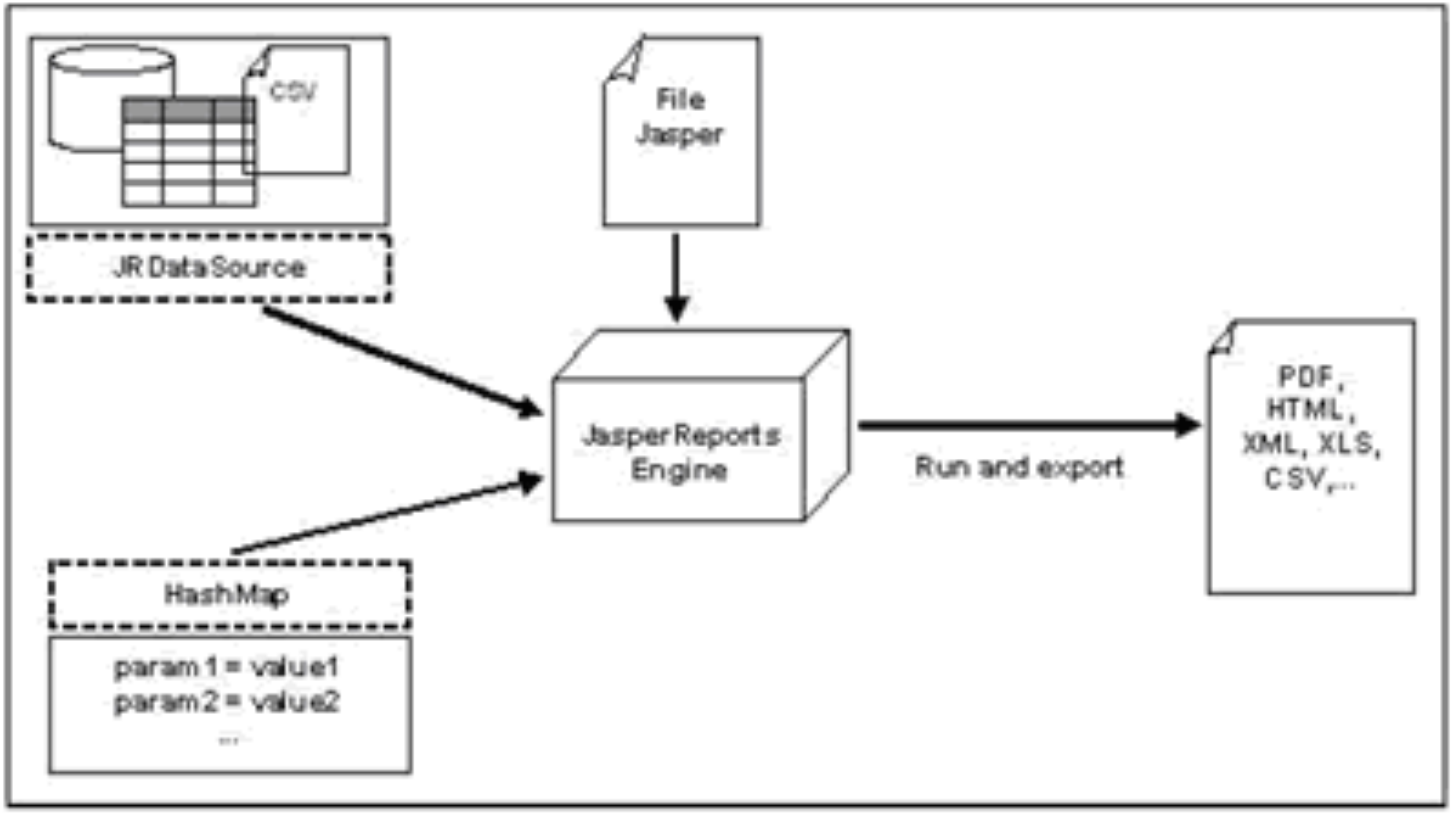
https://poi.apache.org/components/spreadsheet/index.html

text extraction

https://poi.apache.org/text-extraction.html

Jasper Report

Jasper Report architecture



References:

https://community.jaspersoft.com/wiki/jasperreports-library-features-and-highlights

https://www.tutorialspoint.com/jasper_reports/index.htm

Enable Searching in your site

Apache Lucene - Solr

Lucene: powerful indexing and search features, as well as spellchecking, hit highlighting and advanced analysis/tokenization capabilities.

https://lucene.apache.org/

Lucene and Solr

Lucene Core is a Java library providing powerful indexing and search features, as well as spellchecking, hit highlighting and advanced analysis/tokenization capabilities. The PyLucene sub project provides Python bindings for Lucene Core.

Solr™ is a high performance search server built using Lucene Core. Solr is highly scalable, providing fully fault tolerant distributed indexing, search and analytics. It exposes Lucene's features through easy to use JSON/HTTP interfaces or native clients for Java and other languages.

Solr features

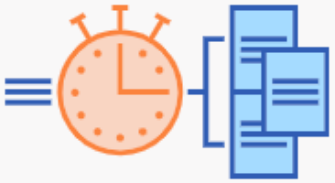
https://lucene.apache.org/solr/features.html

Advanced Full-Text Search Capabilities

Powered by Lucene™, Solr enables powerful matching capabilities including phrases, wildcards, joins, grouping and much more across any data type

Standards Based Open Interfaces - XML, JSON and HTTP

Solr uses the tools you use to make application building a snap



Near Real-Time Indexing

Want to see your updates now? Solr takes advantage of Lucene's Near Real-Time Indexing capabilities to make sure you see your content when you want to see it

Geospatial Search

Enabling location-based search is simple with Solr's built-in support for spatial search

Rich Document Parsing

Solr ships with Apache Tika built-in, making it easy to index rich content such as Adobe PDF, Microsoft Word and more.

Apache UIMA

Ready to enhance your content with advanced annotation engines? Solr integrates into Apache UIMA, making it easy to leverage NLP and other tools as part of your application.