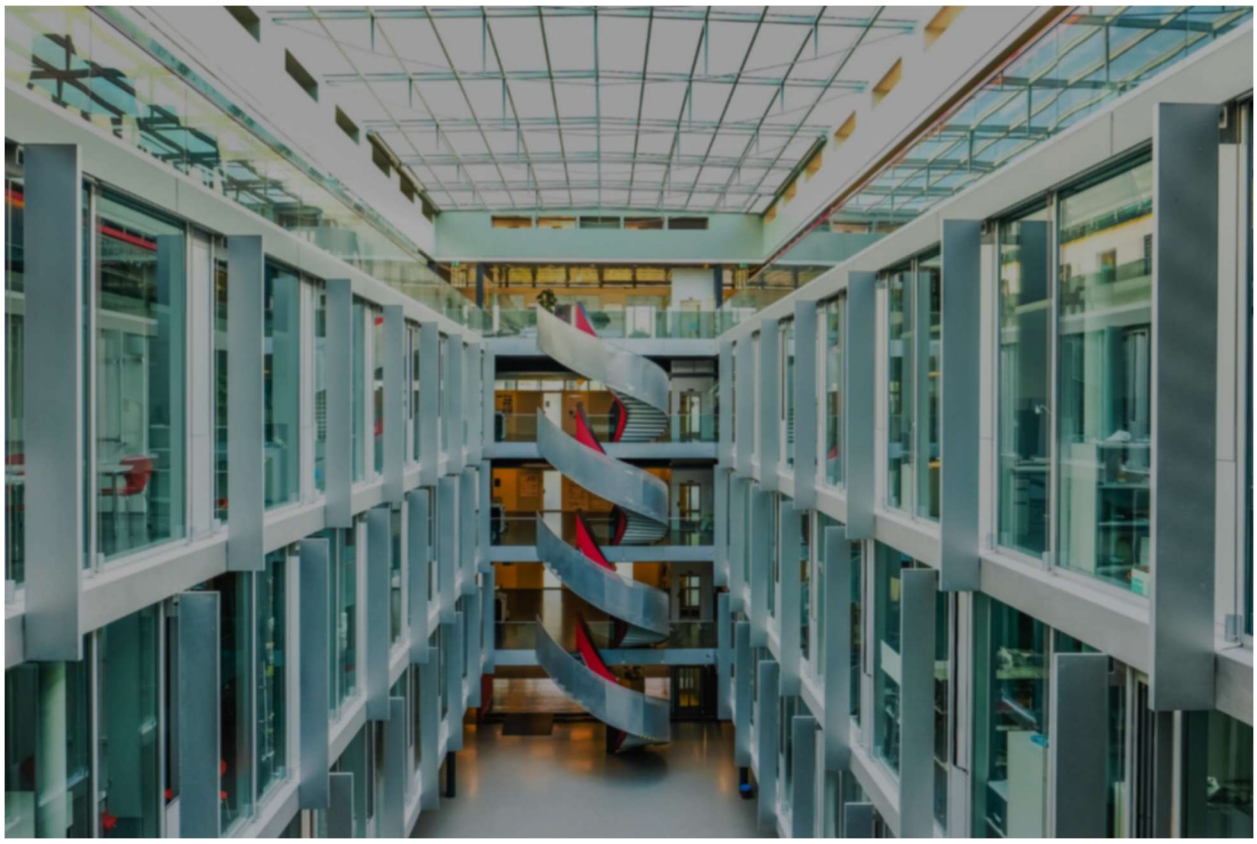

Scaladex: Displaying information from POM files - GSoC 2024 Proposal

Scala Center

15.03.2024

Siddharth Ingle
Savitribai Phule Pune University, India

Name and Contact Information

Name: Siddharth Ingle

Email: skingle97@gmail.com

GitHub: SiddharthAtos, skingle

Location: London, United Kingdom

Timezone: GMT

Typical working hours: 6 – 8 hours from 9 am to 6 pm (local time)

Synopsis

The Scala ecosystem thrives on information accessibility, and Scaladex stands as a pivotal resource in this domain. However, the current scope of Scaladex's artifact pages does not fully exploit the wealth of metadata available in Maven pom files. This project aims to enhance Scaladex by extracting and presenting additional information from pom files, enriching the user experience and empowering developers with deeper insights into Scala artifacts.

What it means to accomplish

- **Enhance the Artifact Model:** Integrate new fields such as scaladocUrl and versionScheme as optional attributes to capture additional metadata from Maven pom files.
- **Improved Artifact Representation:** Update the Artifact case class to accommodate the new fields, ensuring seamless integration within Scaladex's data structure.
- **Database Integration:** Implement necessary database schema changes and modifications to support storing and querying the newly added fields, ensuring data consistency and integrity.
- **Serialization:** Develop serializers, leveraging libraries like circe, to facilitate smooth handling of complex data types such as developers' information.

- **Testing:** Rigorously test the reading and writing functionalities to ensure accurate extraction and storage of data from Maven pom files, including handling scenarios with missing or optional information.
- **Scheduled Job Implementation:** Implement a scheduled job mechanism to periodically scan existing artifacts, identifying, and retrieving missing information from Maven pom files, thereby keeping Scaladex up-to-date and comprehensive.

How will it benefit Scaladex?

- **Enriched Artifact Pages:** Users will access more comprehensive and informative artifact pages, providing deeper insights into Scala artifacts' characteristics and dependencies.
- **Enhanced User Experience:** Developers relying on Scaladex for artifact discovery and evaluation will benefit from a richer dataset, aiding in informed decision-making and project selection.
- **Improved Data Integrity:** With comprehensive database support for additional metadata, Scaladex ensures robustness and reliability in storing and presenting artifact information.
- **Maintenance Automation:** The scheduled job mechanism automates the process of updating and enriching artifact information, reducing manual intervention and ensuring Scaladex's relevance and accuracy over time.

Mentors

- Adrien Piquerez (GitHub: @adpi2, Email: adrien.piquerez@epfl.ch)
- Kannupriya Kalra (LinkedIn)

Deliverables

- Extracting information from POM files received from Sonatype to display them in the artifact page of Scaladex.
- Tests for reading and writing of the new fields into the database

Required Deliverables

Phase I

- Enhance Artifact Model - Artifact Model with new fields (**milestone 1**).
- Integrate with PomConvert Module. (**milestone 2**).

Phase II

- Database Integration and Testing (**milestone 3**).
- Serialization and API Refinement (**milestone 4**).

Phase III

- Scheduled Job Implementation (**milestone 5**).
- User Interface Integration (**milestone 6**).
- Documentation, Optimization, and Final Testing.

Schedule

Community Bonding period - Pre-development Phase (May 1 - May 26)

- **Strengthen Scala Proficiency:** Dedicate time to enhancing Scala programming skills, with a focus on advanced topics such as for-comprehensions and working with Scala's Future API, which are likely to be extensively utilized in the project.

- **Familiarize with Relevant Libraries:** Acquaint oneself with akka-http, doobie, circe and Twirl, as these frameworks and libraries are integral components of the Scaladex project. This involves studying documentation, exploring sample code, and perhaps working through tutorials to gain a solid understanding of their usage.

- **Dive into the Codebase:** Set up development environment - install necessary tools and dependencies. Invest time in comprehensively understanding the existing Scaladex codebase by reading through relevant documentation, exploring source files, and studying previous contributions. This will provide valuable context and insights into project architecture, coding conventions, and existing functionalities.

- **Establish Communication with Mentors:** Set up communication channels with mentors and establish regular check-ins. Initiate discussions with project mentors to clarify project details, establish milestones, and align expectations. Communicate any questions or concerns regarding the project scope, technical requirements, or timeline to ensure a clear understanding and set a solid foundation for collaboration.

Enhance Artifact Model (May 27 - June 2)

- Implement changes to the Artifact Model to include new fields for scaladocUrl, versionScheme, and developers.
- Define corresponding data structures and update database schema.
- Develop serialization mechanisms for handling complex data types.
- Write unit tests to ensure proper functionality of the Artifact Model with new additions.

Integrate with PomConvert Module (June 3 - June 16)

- Modify the PomConvert module to extract and map information from pom files to the enhanced Artifact Model.
- Implement logic to fill in optional fields like scaladocUrl and versionScheme during artifact metadata extraction.
- Conduct extensive testing to verify accurate parsing and integration of pom file data.

Database Integration and Testing (June 17 - June 30)

- Update the database schema to accommodate new fields in the Artifact table.
- Implement read and write operations for the new fields in the database layer.
- Develop comprehensive test suites to validate database interactions, ensuring data integrity and consistency.
- Conduct integration tests to verify end-to-end functionality with Scaladex's existing database infrastructure.

Serialization and API Refinement (July 1 - July 14)

- Fine-tune serialization mechanisms for seamless handling of complex data types like developers' information.
- Refactor existing API endpoints to support retrieval and display of additional metadata fields.
- Conduct thorough testing to ensure API endpoints return expected results with the enhanced data model.

Scheduled Job Implementation (July 15 - July 28)

- Design and implement a scheduled job mechanism for periodically updating artifact information.
- Develop logic to identify and retrieve missing metadata from existing artifacts using pom files.
- Integrate the scheduled job into Scaladex's workflow, ensuring seamless execution and minimal performance impact.

User Interface Integration (July 29 - August 11)

- Modify Scaladex's user interface to display newly added metadata fields on artifact pages.
- Ensure proper formatting and presentation of scaladoc URLs, version schemes, and developer information.
- Conduct user acceptance testing to gather feedback on the updated UI and iterate based on user responses.

Documentation, Optimization, and Final Testing (August 12 - August 26)

- Document the implemented changes, including updated API endpoints, database schema modifications, and integration procedures.
- Optimize codebase for performance and scalability, addressing any bottlenecks or inefficiencies.
- Conduct final rounds of testing, including regression testing and performance profiling, to ensure readiness for deployment.
- Prepare for project submission, including code cleanup, documentation review, and finalization of deliverables.

Availability

Throughout the official GSoC timeline, extending from May 1st to August 19th, I can commit around 18-20 hours weekly. My weekends are entirely free, allowing for further engagement if needed.

Publishing the code

Pull Request Strategy:

- I plan to submit one pull request (PR) for each milestone achieved throughout the project.
- These PRs will be directed to a designated branch specifically created for this project.
- Once all milestones are completed, this project branch will undergo merging into the main branch.

Adherence to Best Practices:

- Throughout my contributions, I aim to adhere to best practices such as maintaining clear and concise commit messages.
- I'll ensure thorough documentation where necessary to enhance the project's usability and maintainability.
- Actively participating in code reviews and discussions will be a priority to ensure the quality and coherence of the project codebase.

Responsiveness to Feedback:

- I commit to remaining responsive to feedback from project maintainers and community members.
- I'll iterate on my contributions as needed to align with project objectives and standards, ensuring continuous improvement and alignment with community expectations.

About Me

Education

University: Savitribai Pune University, Pune, India

Major: Computer Science (B.E)

Graduation year: 2018

Why this project is important to me

As a newcomer to open-source contribution and a recent enthusiast of Scala, the project involving enhancing Scaladex by extracting information from POM files is particularly meaningful to me.

This project perfectly aligns with my current skill level and learning goals, offering a hands-on opportunity to delve into the Scala ecosystem and contribute to a practical tool like Scaladex. It enables me to apply theoretical knowledge in a real-world setting, accelerating my growth as a Scala developer. Specifically, diving into indexing within Scaladex provides a unique chance to understand how information is organized and retrieved efficiently, enhancing my understanding of data structures and algorithms.

Beyond personal development, contributing to Scaladex allows me to give back to the open-source community, fostering innovation and collaboration on a global scale. Knowing that my contributions can benefit countless developers worldwide is incredibly rewarding.

Moreover, working on this project connects me with experienced developers and mentors, providing valuable guidance, feedback, and networking opportunities within the Scala community. Collaborating with like-minded individuals enhances my skills and accelerates my development as a Scala practitioner.

In essence, this project represents a significant milestone in my journey as a Scala developer and open-source contributor, offering the chance to learn, grow, and make a meaningful impact while connecting with a vibrant community of technology enthusiasts.

Past Experience

In my journey as a Software Developer, my enthusiasm for exploring new horizons led me to immerse myself in various projects, each offering unique challenges and learning opportunities. My introduction to the world of open-source contribution ignited a spark within me, driving me to leverage my skills and passion for functional programming in Java and Scala towards collaborative endeavors.

During my tenure at HMRC, I immersed myself in the dynamic world of REST APIs and fault-tolerant microservices. Here, I was tasked with crafting scalable solutions, where I adeptly utilized Java, Scala, and the Play framework to bring innovative ideas to life. My dedication to excellence was evident in every facet of my work, from the creation of services for intricate business logic to ensuring adherence to SOLID principles. Embracing the ethos of Test-Driven Development (TDD), I iteratively refined solutions, ensuring robustness and reliability. Additionally, I seamlessly integrated connectors for downstream API calls, facilitating seamless communication between various components of the system. Further, my responsibilities extended to content localization, enhancing the accessibility and user experience of our applications.

Notably, I spearheaded the creation of RESTful API endpoints, streamlining data access and manipulation. Furthermore, I ventured into crafting UI pages using the Twirl templating engine within the Play framework, marrying functionality with aesthetic appeal. Throughout this journey, I leveraged the power of functional programming using Scala, unlocking new possibilities for software development. As I reflect on this experience, I am invigorated by the prospect of exploring more Scala projects, eager to delve deeper into its nuances and contribute meaningfully to future endeavors.

I orchestrated backup solutions on Azure stack and automated deployment processes on OpenShift. Harnessing tools like Jenkins, I streamlined deployment pipelines, slashing deployment time by a remarkable margin.

Each experience has been a stepping stone, fueling my eagerness to delve deeper into the realm of open-source contribution. With a solid foundation in Java, Scala, and cross-functional teamwork, I am poised to embark on new endeavors, driven by a fervent desire to make meaningful contributions to the open-source community. Additionally, my journey includes learning Akka through a course from Rock the JVM, although I haven't had the opportunity to apply it in a professional setting yet.