

PyZombis: A community course to teach Python

Sub-Organization: Python Argentina

About Me:

Personal Details:

- Name: Atharva Kulkarni
- GitHub: <https://github.com/IronVenom>
- Country: India

Education:

- Institution: [Indian Institute of Technology, Bhubaneswar](#)
- Major: Electrical Engineering
- Program: Dual Degree (BTech. and MTech. with specialization in Power Electronics and Drives.)
- Current Academic Year: Sophomore (4th Semester)
- Expected Graduation Year: 2022

Personal Background and Programming Experience:

I am an undergraduate sophomore with a huge interest in web development, open source software development, machine learning, and cybersecurity. I am an aspiring Python Developer, and have been using it over a year, and have used a variety of libraries - OpenCV, NumPy, Matplotlib, Scikit-learn, Beautiful Soup, Click, etc. I have also made a Discord Bot using Python and Discordpy, and a CLI application for my institute. I had participated in the Halite 3 Artificial Intelligence Bot contest created by Two Sigma and had secured a rank of 1961 out of 4014 participants. I mostly use Windows 10 as my operating system and Sublime Text as my code editor.

I am also proficient in C++ and have done a fair amount of competitive programming on platforms such as Hackerrank and Codechef. I have participated in the ACM ICPC 2018 regionals. Currently, I am working on a GUI development project, using Qt C++ to make the GUI and with PostgreSQL as the database management system to analyze the transition time of transistors in integrated circuits.

I am a self-taught web developer and am constantly learning new technologies. I have a fair amount of knowledge of HTML, CSS, Bootstrap, and JavaScript, and am currently focusing on

the MERN stack. I am accustomed to using Git and Github and have been using Git as a version control software for a year.

I am an avid learner and am always eager to learn new technologies.

Code Contribution:

I have opened the following pull requests in PyAr/PyZombis:

- [Migrated the lectures to work on Runestone Serve.](#)
- [Embedded all the videos in the lectures topic wise.](#)
- [Translated lecture 10 to Spanish using Sphinx Internationalization](#)
- [Created a Quiz using Activecode, taking questions from the provided quizzes.](#)
- [Created Interactive examples using Codelens in lecture 10](#)

I have also published an online static build to check the current progress:

<https://ironvenom.github.io/>

Apart from this, I have also made a critical comment on automatic subtitles generation for the videos in English - [Automated Subtitles Generation](#).

Open Source Experience:

The pull requests that I have made in PyAr/PyZombis is the first time that I have contributed to an organization apart from my institute. However, I have enough experience using Git and GitHub, as I have been using it in projects over the past year. This project will also provide me a chance to learn more about Open Source Development.

Project: PyZombis

Organization: [Python Argentina](#)

Mentors: [Mariano Reingart](#), [Nahuel](#)

Project Abstract:

The aim of the project is to teach Python to people of all ages and background in a lucid, creative and interactive manner, in the form of an interactive course created using the Runestone framework, and integrating the YouTube API and Sphinx Documentation with it. The aim is also to create a massive online community course, so as to teach Python in Spanish. This project has the potential to reach over thousands of students and people of all ages and teach them Python in a lucid and easy manner so that they are able to apply the concepts in real life

situations, which is of paramount importance. It involves creating an entire Runestone setup and serving lectures, which would be translated to Spanish through Sphinx Internationalization. The quizzes and assignments will be created using Runestone's components, so as to provide highly interactive environments for the people to learn and challenge themselves. The lectures will be accompanied by examples and interactive exercises made using Runestone's components, which will help in providing a thorough explanation of the concepts discussed in the lectures. Videos will be added to support the content, along with automatic subtitle generation.

The project is based on the original Brazilian MOOC [Python para Zumbis](#) created by [Fernando Masanori](#) that has 500k+ reproductions on [Youtube](#) and has taught basic Python to over 70k students. Original course materials are mainly ad hoc Presentations (PPTX) and Videos. Source documents were already converted to ReST files so they are easier to adapt and translate. Original exercises and assignments had to be run in student's machines (local python installation was required), and corrections were mainly community-based. This project will also provide an enhancement via online interactive exercises and automatic grading, to give more feedback and hints to students, lowering effort from community members (teaching assistants).

My Motivation:

I have a huge interest in programming, and I like to learn new things every day. This is also the reason why I have joined my institute's programming club. With fields like Machine Learning, Artificial Intelligence, Deep Learning, Internet of Things, etc. progressing at a rapid rate these days, the knowledge of programming, hence becomes crucial. Programming is not limited only to software development; it is applied science. In every sphere of human life, programming plays a crucial role. I love to teach people, and with this project, I hope to reach out to thousands of students and provide them a chance to learn Python in an easy and awesome manner, and become a part of the Open Source Python community, and remain a contributor in the long term.

My Goals:

- **Interactive demonstrations and adding comment boxes.**

Overview:

The aim will be to add interactive examples, exercises and comment boxes using various components of Runestone.

Deliverables:

The interactive examples and exercises can be added in the Restructured Text files using the components like the Codelens component. An example of this can be shown in the pull request [#18](#) that I have made. We can use various other components as well, like the Multiple Choice component to add multiple choice questions after the lectures, so as to test the user's understanding of the concept that has been taught. The ShowEval Trace mode component and ShowEval Replace mode component can be used to show and explain the workflow of the

code. Comment boxes can be added using the Disqus comment box component. Polls can be created as well, for purposes such as taking reviews or ratings from the user using the Polls component.

- **Adapting assignments to use Runestone's Components**

Overview:

The aim is to migrate all the existing quizzes and assignments to use Runestone's components, and connecting them to the Runestone book.

Deliverables:

The quizzes and assignments that are currently Python files, can be added to the Runestone book by creating Restructured Text files, and then we can embed the questions, functions and sample unit tests using components such as ActiveCode. An example of this can be shown in the pull request [#17](#) that I have made. Apart from ActiveCode, we can add multiple choice questions using the Multiple Choice component and drag and drop problems using the Parsons Problems component. We can add questions involving error detection using the Clickable Area component as well.

- **Embedding videos and automatic subtitles translation**

Overview:

The aim is to add videos relevant to the lecture in that lecture itself and then generate automatic subtitles for the same using YouTube's API in Spanish as well as in English.

Deliverables:

The videos can be embedded in the lectures using the video and YouTube components of Runestone. I have done this in the following pull request - [#15](#). For automatic subtitles generation, I have tried two approaches, which can be seen in the comment - [Comment regarding subtitles](#). I have tried to incorporate the required language in the video id of the video and also tried to change the `playersvars` variable in the `onYouTubeIframeAPIReady()` function present in the `runestonevideo.js` file, which is created in the `_static` folder by default. I intend to create separate JavaScript files for this purpose, which will involve redirecting from the `runestonevideo.js` file and the API call will be made from these new files while taking into consideration that the Runestone structure is not disturbed.

- **Internationalization of lectures**

Overview:

The current lectures and quizzes are in Portuguese, and the aim is to translate them to Spanish and English.

Deliverables:

This can be done using Sphinx Internationalization and GNU gettext utilities. An example of this can be seen in the pull request that I have made - [#16](#), in the .po file. Using Sphinx Internationalization we can generate the HTML files and Restructured Text files, which can be then later on integrated into our Runestone framework. Using gettext, we can create the .pot files. Then using Sphinx Internationalization we build the .po files, and then using either sphinx-intbuild or msgfmt utility, we build the .mo files. After this, we use msgmerge utility to add any remaining strings from the .pot files to the .po files. And then we add the translations to the .po files in the msgstr strings, and then create .mo files again. After that, we build the document using batch create (.\make.bat html). This procedure can be followed for all the files, thereby giving us the translated lectures and quizzes. Constant integration of the generated files with the original Runestone Framework will be required.

- **Serving the created Runestone Book and creating an Instruction Manual**

The final step would be to bring together all the integrations and build the book and serve it. An instruction manual is also to be created. It will contain a detailed explanation of how the book works.

Weekly Timeline:

- **May 7 - May 26:**

- ❖ Read the Runestone Documentation and learn how to tackle external integrations.
- ❖ Read the Sphinx Documentation for creating complicated Restructured Text files.
- ❖ Read how to work with Sphinx Internationalization and GNU gettext utilities more thoroughly and learn how to include undetected words externally.
- ❖ Read the YouTube API documentation and find a way to integrate it with Runestone externally so that the API call can be made independent of the runestonevideo.js file.
- ❖ Communicate with the mentors and discuss more innovative ideas that can be added in the Runestone Book. Get involved with the PyAr community and get to know more about the mentors such as their timezone, preferred medium of communication, etc.

- **Week - 1 (May 27 - June 2):**

- ❖ Start adding Interactive Examples and Exercises to the Restructured Text files.

- **Week - 2 (June 3 - June 9):**

- ❖ Complete adding Interactive Examples and Exercises to the Restructured Text files.
- ❖ Start adding quizzes and assignments to the Runestone book.

- **Week - 3 (June 10 - June 16):**

- ❖ Complete adding the quizzes and assignments.
- ❖ Run the first trial of the book to check for errors.

- **Week - 4 (June 17 - June 23):**

- ❖ Start translating the lectures to Spanish and English

- ❖ Add additional unit tests for the quizzes and assignments.
- **Week - 5 (June 24 - June 26):**
- ❖ Phase 1 Evaluation
- **Week - 5 (June 27 - June 30):**
- ❖ Work on translating the lectures through Sphinx Internationalization.
- ❖ Integrating the generated documents with the Runestone framework without loss of meaning.
- ❖ Adding missing words externally.
- **Week - 6 (July 1 - July 7):**
- ❖ Complete translating the lectures.
- ❖ Integrating the generated documents with the Runestone framework without loss of meaning.
- ❖ Integrating the Runestone components to work with the newly generated files.
- ❖ Start translating the quizzes and assignments.
- **Week - 7 (July 8 - July 14):**
- ❖ Complete translating the quizzes and assignments to Spanish and English.
- ❖ Run the second trial of the book to check for errors and making sure that all the integrations work together.
- **Week - 8 (July 15 - July 21):**
- ❖ Start embedding the lectures in the Restructured Text files, making sure that Sphinx Translation is not disturbed.
- **Week - 9 (July 22 - July 24):**
- ❖ Phase 2 evaluation
- **Week - 9 (July 25 - July 29):**
- ❖ Complete embedding the lectures
- ❖ Start working on code to generate automatic subtitles
- **Week - 10 (July 30 - August 5):**
- ❖ Work on the code for generation of automatic subtitles and integrate it with the YouTube API and Runestone framework.
- ❖ Run the third trial to make sure that everything is working.
- **Week - 11 (August 6 - 12):**

- ❖ Complete the subtitles generation code.
- ❖ Run complete trial of the book to see if everything works perfectly.

- **Week 12 (August 13 - 19):**
 - ❖ Checking for any errors and finishing up the project.
 - ❖ Creating documentation for usage.

- **Week 13 (August 20 - 25):**
 - ❖ Wrapping up and submitting the project.

Will I be able to complete this project?

Yes, I am very sure that I will be able to complete the project successfully. I have been working on this project for a month and have been in constant contact with the mentors. My motivation to teach programming has been a boosting factor for me to work on this project. I know that I will get to learn a lot of new stuff while working on it. I will strictly adhere to the timeline that I have proposed and deliver on all the goals that I have promised. With my skills and willingness to learn, I believe that I am well suited for this project and will be able to complete it successfully.

How will this benefit the community?

Since the aim of the project PyZombis is to create a massive open community course, it will benefit people of all ages, and help them to learn programming in an easy manner. With coding exercises, examples, assignments and tons of interactive features, the book aims to teach people programming in a way that will help them to apply it in their day to day lives.

Other Commitments:

I am applying to Google Summer of Code for the first time, and I will work on this project only. I have no other vacation plans or internships in the summer, and I will be fully concentrating on completing this project. I will be able to devote 40 to 45 hours of work per week. If emergency situations arise, then I will take measures to make up for the lost time. I am also eligible to receive stipend from Google.