

DevLog

Devlog for PyLearnGTK

Arseni Yamnii

StackOverflow

Git

Linux

Python3

LaTex

GTK

2020

Contents

1	PreDeveloping	1
April 29, 2020	create DevLog	2
May 2, 2020	LaTex	3
May 3, 2020	UML day one	4
May 5, 2020	Little tis about UML	5
May 7, 2020	UML class diagram	5
May 8, 2020	Ready with UM-class, and note about Sphinx	5
May 10, 2020	Planning	6
May 11, 2020	Scope/Statement of work	6
May 13, 2020	Scope/Statement of work:Part 2	6

1 PreDeveloping

April 29, 2020 create DevLog

here il starting write dev log with all my thinks(misly in russian) about this project

May 2, 2020 LaTeX

Start using Latex And so... I create my template for DevLog. It is like diary for application. Here i write some tips and triks)).

I think sometimes i create script to integrate this tis to Documentation. Maby with some Tags, etc.

Yes, it is long project... with UML and texts. But it pretty shit functionals. BUT mayby sometimes it get more functional, to got more popularity.

TODO:

- Create UML diagram with TikzUML. Yes in latex.....
- Find examples to python with GTK
- Create GTK UI
- create tests to program

May 3, 2020 UML day one

Few days ago i find latex package TIKZ-UML. this package was created for create UML diagrams in latex. I think it is the best tool for create uml, because for transfer source to another achine, you only need to transfer few files. .tex file, and library tikz-uml. And then compile to format what you want.

TODO:

- crete folder for code examples
- look throught tikz-uml example code, and create own UML diagrams for project. I dont know why, but for new expierance

May 5, 2020 Little tis about UML

UML got 15 diagrams :

- class - DONE
- component
- composite structure
- collaboration
- deployment
- object
- package
- profile
- activity
- state machine
- use case
- communication
- interaction overview
- sequence
- timing

I need learn all about this 15 diagrams, and create exampples, or working diagrams to my project. For new expiarence.

May 7, 2020 UML class diagram

I create i think good class diagram to first try. When i learn GTK, and UML more, i recreate it diagram. Tikz-uml goood))

May 8, 2020 Ready with UM-class, and note about Sphinx

Few day ago, i see, that Linux kernel used Sphinx as auto-documentation tool. I need learn more about sphinx UPD: Right now i undestand, that i dont use Software Development Cycle. I need that.

- planning
- analysis
- design
- implementation
- testing and integration
- maintenance

May 10, 2020 Planning

Planning it is Scope of Work (SOW). It is just few questions, and i must answer to them. But! What is the questions? I just gogle "scope of work questions", and find many questions for SOW. I ust read them all (only fro first page of google), and create fram them all y questions. Soething like Universal SOW questions. Every queston ust be unic.

May 11, 2020 Scope/Statement of work

Scope of work questions (in english and russian): русский sorry. i write it in english. But now i know how to write in many languages in one latex file

May 13, 2020 Scope/Statement of work:Part 2

Scope of work:

- Deliverables
what we doing. all stuff. Product/service. All things.
- Milestones
Major project stages
- Timeline
project timeline. We can use Gantt chart
- Reports
Whoch reprts we can do from all project stages

It is from <https://www.projectmanager.com/training/write-scope-work>
Oh fuck. It is just part of Statement of work. Statement of work i learn from <https://www.projectmanager.com/blog/statement-work-definition-examples>
Statement:

- **Intro:**
Begin with explaining what work is being done. Also, who is involved in the project? State these parties. This will lead to a standing offer, which cements prices for products or services purchased for the project, and a more formal contract that goes into greater detail.
- **What Is the Purpose of the Project:**
Start with the big question: why are you initiating this project? What's the purpose of doing the project? Create a purpose statement to lead off this section and provide a thorough answer to these questions, such as what are the deliverables, objectives and return on investment.
- **Scope of Work:**
What work needs to be done in the project? Note it here, including what hardware and software will be necessary. What is the process you'll use to complete the work? This includes outcomes, time involved and even general steps it'll take to achieve that.
- **Where Will the Work Be Done:**
The team you employ will have to work somewhere. The project might be site specific, at a central facility or some, if not all, the team members could work remotely. Either way, here is where you want to detail that and where the equipment and software used will be located.
- **Tasks:**
Take those general steps outlined in the scope of work and break them down into more detailed tasks. Be specific and don't leave out any action that is required of the project to produce its deliverables. If you want to, break the tasks down into milestones or phases.
- **Milestones:**
Define the amount of time that is scheduled to complete the project, from the start date to the proposed finish date. Detail the billable hours per week and month, and whatever else relates to the scheduling of the project. Again, specificity counts. For example, if there's a maximum amount of billable hours for vendors and/or contracts, note it here.
- **Deliverables:**
What are the deliverables of the project? List them and explain what is due and when it is due. Describe them in detail, such as quantity, size, color and whatever might be relevant.
- **Schedule:**
Include a detailed list of when the deliverables need to get done, beginning with which vendor will be selected to achieve this goal, the kickoff, what the period of performance is, the review stage, development, implementation, testing, close of the project, etc.

- Standards and Testing:
If there are any industry standards that need to be adhered to, list those here. Also, if there will be testing of the product, list who will be involved in this process, what equipment is needed and other resources.
- Define Success:
Note what the sponsor and/or stakeholder expects as a successful project completion.
- Requirements:
List any other equipment that is needed to complete the project and if there is a necessary degree or certification required of team members. Also, note if there will be travel or other aspects of the project not already covered.
- Payments:
If the budget has been created, then you can list the payments related to the project, and how they will be delivered, up front, over time or after completion. For example, you can pay after the completion of a milestone or on a fixed schedule, whichever is more financially feasible.
- Other:
There will be other parts of the project that are not suited to the above categories, and this is the place where you can add them so that everything is covered. For example, are there security issues, restrictions around hardware or software, travel pay, post-project support, etc?
- Closure:
This will determine how the deliverables will be accepted, and who will deliver, review and sign off on the deliverables. Also, it deals with the final admin duties, making sure everything is signed and closed and archived.