哈尔滨工业大学 计算学部

2024年秋季学期《开源软件开发实践》

Lab3：自选开源项目编程

学号	姓名	联系方式
2021130014	李璉峻	17645005180/1391843714@qq.com
		Email/手机号码
		Email/手机号码

目 录

1 实验要求.....↑↑↑↑↑↑1
2 实验内容1 掌握基本Markdown语法 .....↑↑↑↑↑↑1
3 实验内容2 参与开源项目 .....↑↑↑↑↑↑1
3.1 分析所选项目.....↑↑↑↑↑↑1
3.2 项目参与者贡献过程分析.....↑↑↑↑↑↑1
3.3 贡献代码.....↑↑↑↑↑↑1
4 小结.....↑↑↑↑↑↑1

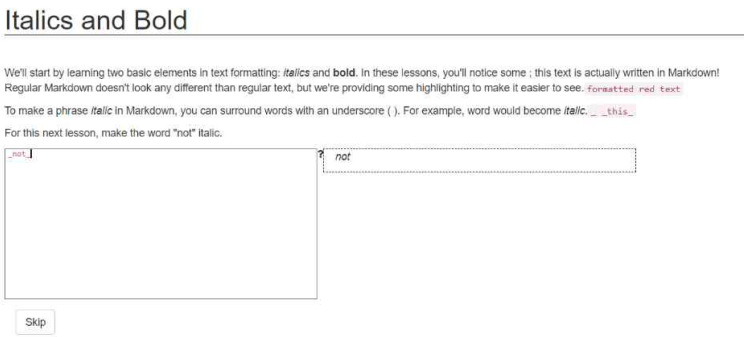
[文档全部完成之后，请更新上述区域]

1 实验要求

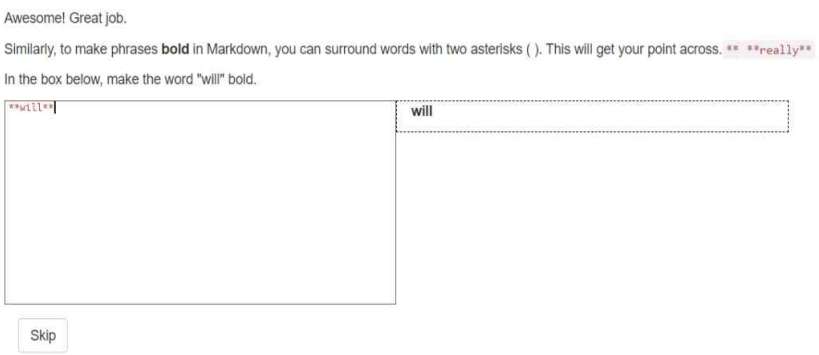
这次实验的目标是能够根据开源项目提供的文档，对项目进行了解和分析、掌握基本的Markdown语法、实际参与某开源项目，贡献代码。所以首先掌握Markdown语法，然后参与实际开源项目。

2 实验内容1 掌握基本Markdown语法

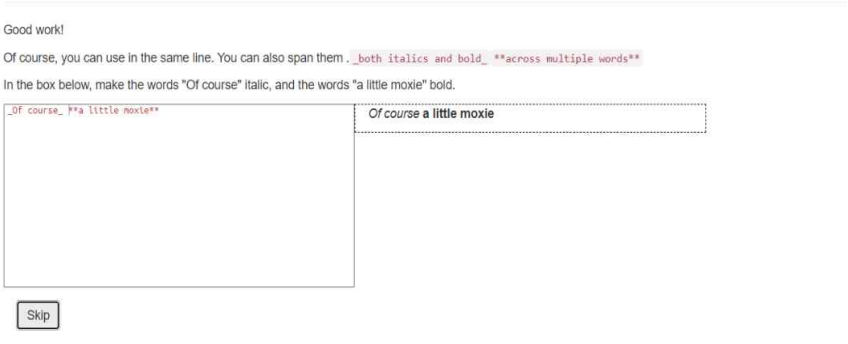
1. Italics and Bold



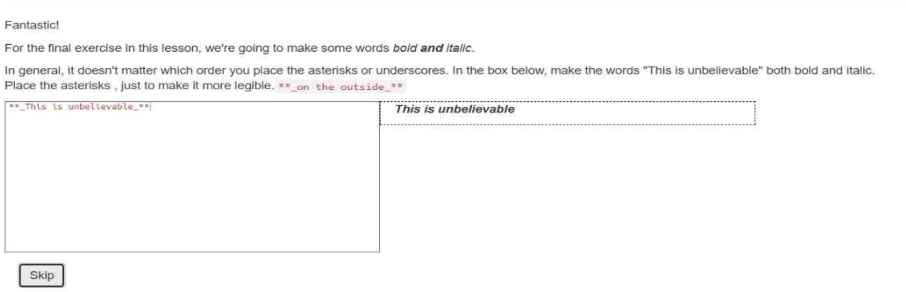
代码：_not_



代码：**will**



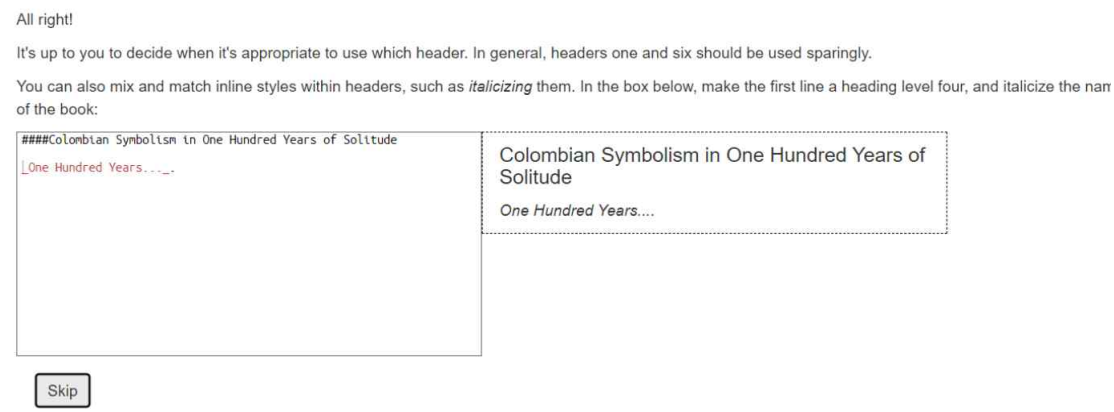
代码：_of course_ **a little moxie”



代码：**_This is unbelievable_**

2.Header

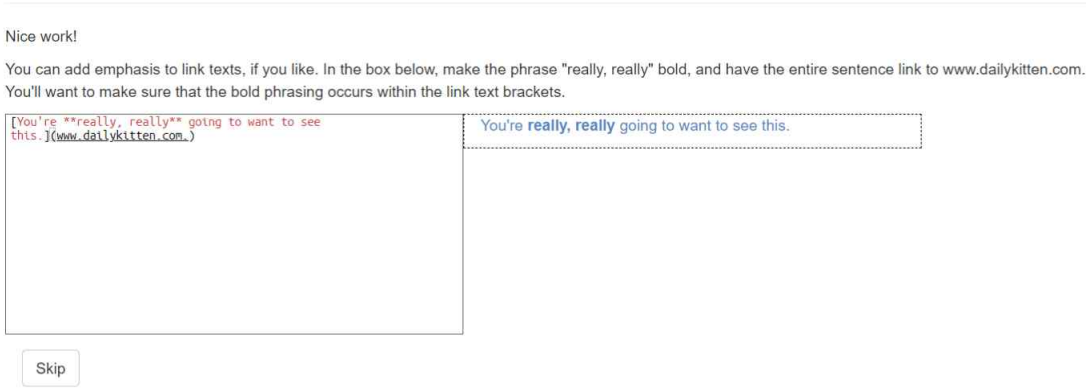
代码：#Header one ##Header Two ###Header Three



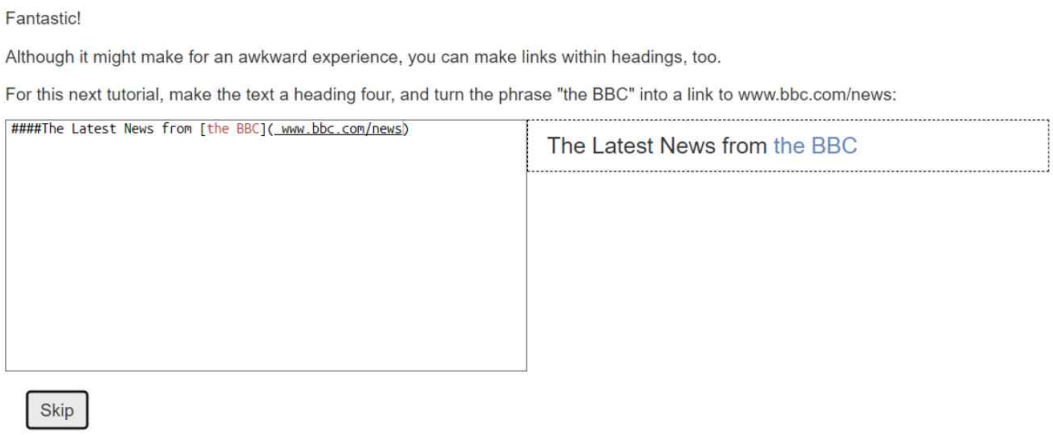
代码：####Colombian Symbolism in One Hundred Years of Solitude.
One hundred Years

3.Links

代码：[Search for it](www.google.com)



代码：[You're **really, really** going to want to see this.](www.dailykitten.com.)



代码：####The Latest News from the [BBC](www.bbc.com/news)

That's all there is to writing inline links.

The other link type is called a *reference* link. As the name implies, the link is actually a reference to another place in the document. Here's an example of what we mean:

```
Here's [a link to something else][another place].
Here's [yet another link][another-link].
And now back to [the first link][another place].

[another place]: www.github.com
[another-link]: www.google.com
```

The "references" above are the second set of brackets: `[another place]` and `[another-link]`. At the bottom of a Markdown document, these brackets are defined as proper links to outside websites. An advantage of the reference link style is that multiple links to the same place only need to be updated once. For example, if we decide to make all of the `[another place]` links go somewhere else, we only have to change the single reference link.

Reference links don't appear in the rendered Markdown. You define them by providing the same tag name wrapped in brackets, followed by a colon, followed by the link.

In the box below, we've started writing out some reference links. You'll need to finish them up! Call the first reference tag "a fun place", and make it link to www.zombo.com; make the second link out to www.stumbleupon.com.

```
Do you want to [see something fun][a fun place](www.zombo.com)?

Well, do I have [the website for you](www.stumbleupon.com)[another fun place]!
```

Do you want to [see something fun]a fun place?

Well, do I have the website for you[another fun place]!

代码：Do you want to [see something fun][a fun place](www.zombo.com)?
Well, do I have [the website for you](www.stumbleupon.com)[another fun place]!

4.Images

Images

If you know how to create links in Markdown, you can create images, too. The syntax is nearly the same.

Images also have two styles, just like links, and both of them render the exact same way. The difference between links and images is that images are prefaced with an exclamation point (`!`).

The first image style is called an *inline image link*. To create an inline image link, enter an exclamation point (`!`), wrap the alt text in brackets (`[ ]`), and then wrap the link in parentheses (`( )`). (Alt text is a phrase or sentence that describes the image for the visually impaired.)

For example, to create an inline image link to https://octodex.github.com/images/bannekat.png, with an alt text that says, Benjamin Bannekat, you'd write this in Markdown: `![Benjamin Bannekat](https://octodex.github.com/images/bannekat.png)` .

In the box below, turn the link to an image, and fill out the alt text brackets to say "A pretty tiger":

```
[!A pretty tiger](https://upload.wikimedia.org/wikipedia/commons/5/56/Tiger.50.jpg)
```

?

!A pretty tiger

Skip

代码：[A pretty tiger](https://upload.wikimedia.org/wikipedia/commons/5/56/Tiger.50.jpg)

Wonderful!

Although you don't *need* to add alt text, it will make your content accessible to your audience, including people who are visually impaired, use screen readers, or do not have high speed internet connections.

For a reference image, you'll follow the same pattern as a reference link. You'll precede the Markdown with an exclamation point, then provide two brackets for the alt text, and then two more for the image tag, like this: `![The founding father][Father]` At the bottom of your Markdown page, you'll define an image for the tag, like this: `[Father]: http://octodex.github.com/images/founding-father.jpg` .

In the box below, we've started placing some reference images; you'll need to complete them, just like the last lesson. Call the first reference tag "Black", and make it link to `https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg` ; make the second image link out to `http://icons.iconarchive.com/icons/google/noto-emoji-animals-nature/256/22221-cat-icon.png` .

```
[Black cat]https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg

[Orange cat][Orange]http://icons.iconarchive.com/icons/google/noto-emoji-animals-nature/256/22221-cat-icon.png

[Black] https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg
```

[Black cat]https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg

[Orange cat]
[Orange]http://icons.iconarchive.com/icons/google/noto-emoji-animals-nature/256/22221-cat-icon.png

[Black]
https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg

Skip

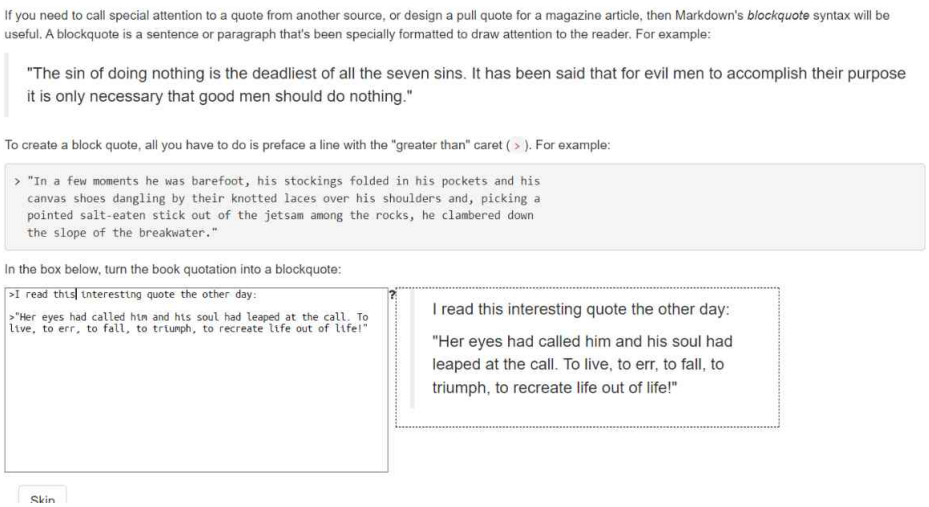
代码：

[Black cat]https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg

[Orangecat][Orange]http://icons.iconarchive.com/icons/google/noto-emoji-animals-nature/256/22221-cat-icon.png

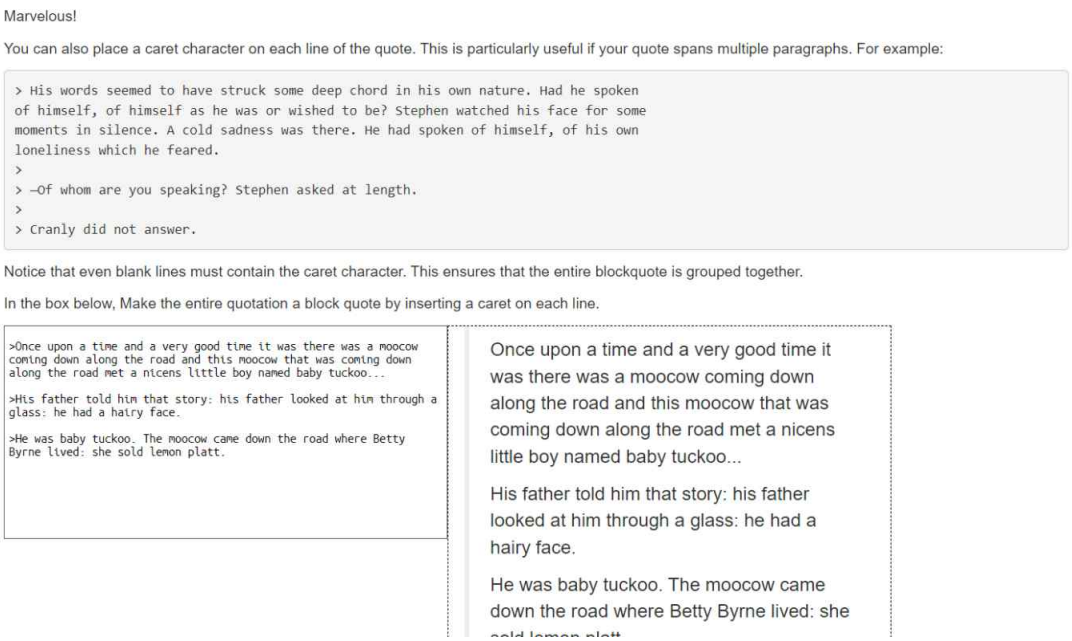
[Black] https://upload.wikimedia.org/wikipedia/commons/a/a3/81_INF_DIV_SSI.jpg

5.Blockquotes



代码：>I read this interesting quote the other day:

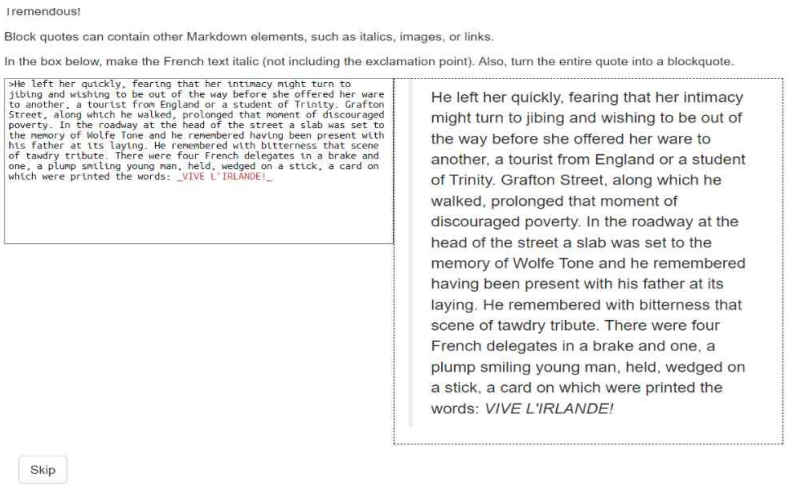
>"Her eyes had called him and his soul had leaped at the call. To live, to err, to fall, to triumph, to recreate life out of life!"



代码：>Once upon a time and a very good time it was there was a moocow coming down along the road and this moocow that was coming down along the road met a nicens little boy named baby tuckoo...

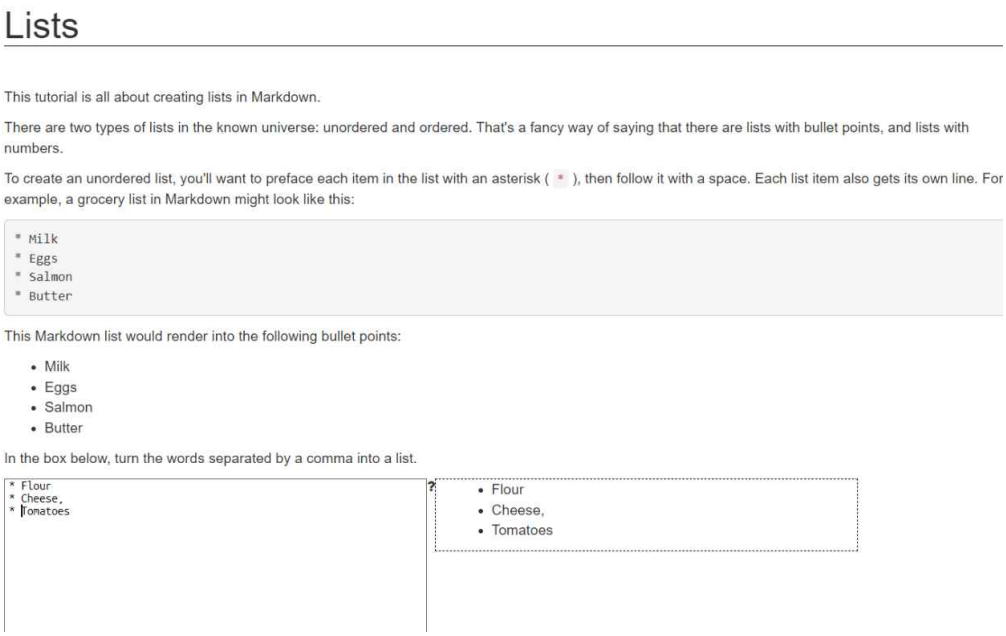
>His father told him that story: his father looked at him through a glass: he had a hairy face.

>He was baby tuckoo. The moocow came down the road where Betty Byrne lived: she sold lemon platt.



代码：>He left her quickly, fearing that her intimacy might turn to jibing and wishing to be out of the way before she offered her ware to another, a tourist from England or a student of Trinity. Grafton Street, along which he walked, prolonged that moment of discouraged poverty. In the roadway at the head of the street a slab was set to the memory of Wolfe Tone and he remembered having been present with his father at its laying. He remembered with bitterness that scene of tawdry tribute. There were four French delegates in a brake and one, a plump smiling young man, held, wedged on a stick, a card on which were printed the words: _VIVE L'IRLANDE!_

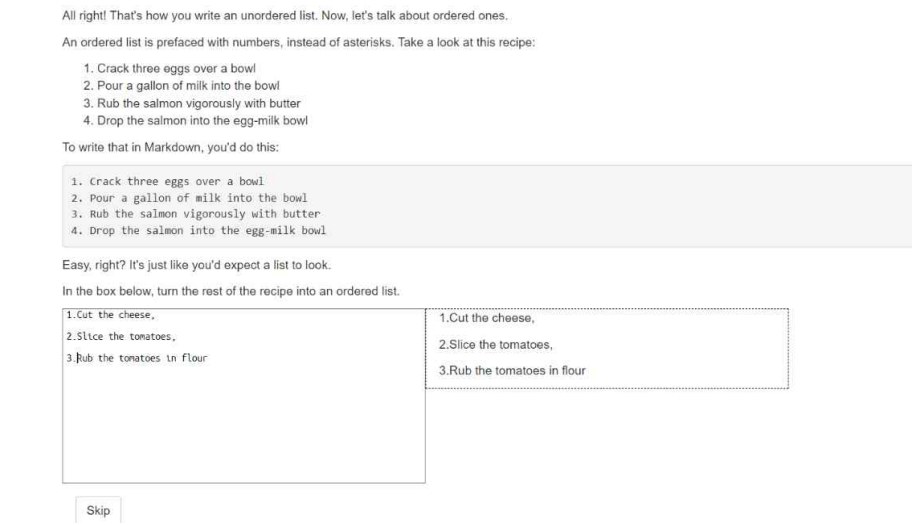
6.Lists



代码：* Flour,

* Cheese,

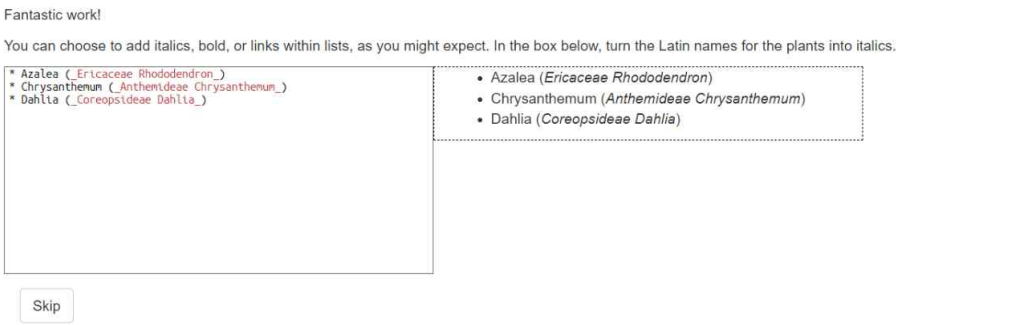
* Tomatoes



代码：1.Cut the cheese,

2.Slice the tomatoes,

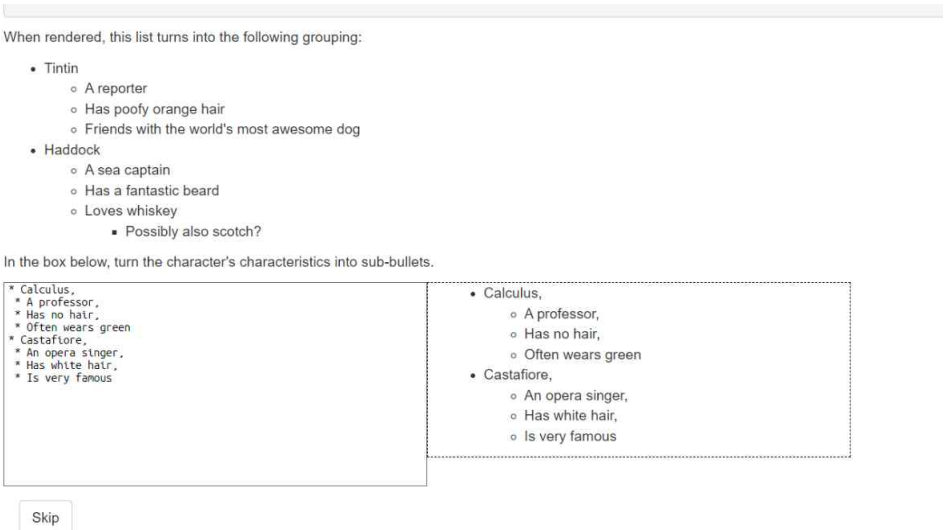
3.Rub the tomatoes in flour



代码：* Azalea (_Ericaceae Rhododendron_)

* Chrysanthemum (_Anthemideae Chrysanthemum_)

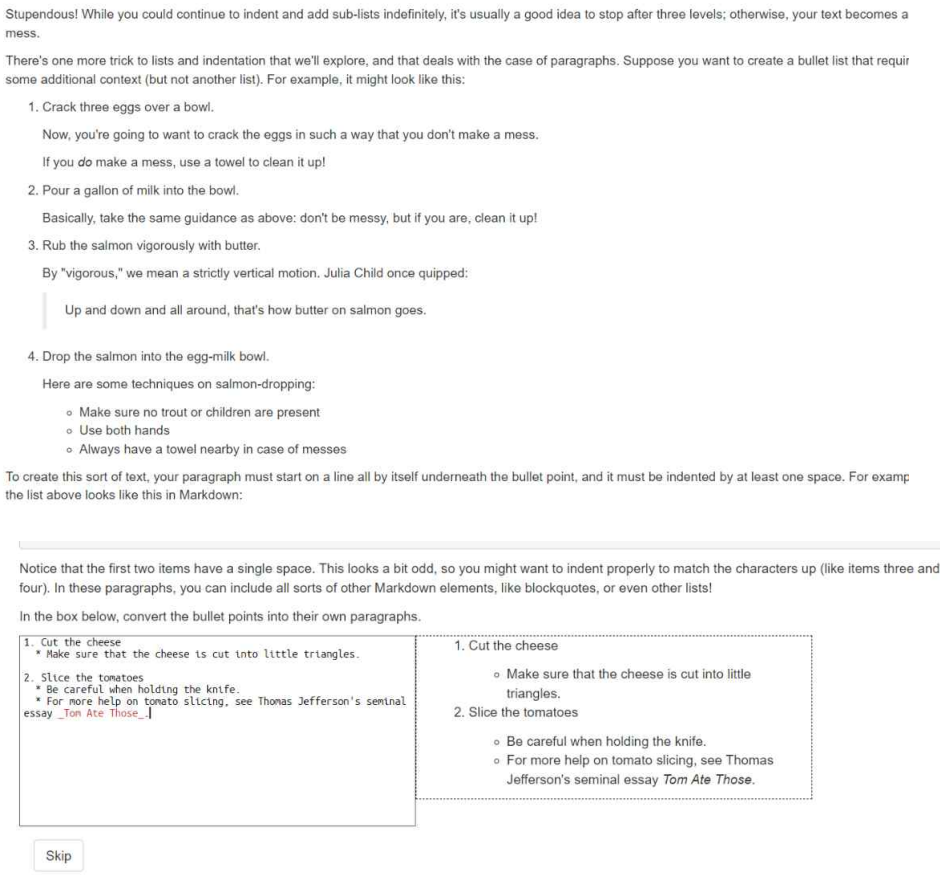
* Dahlia (_Coreopsideae Dahlia_)



代码：* Calculus,

* A professor,
* Has no hair,
* Often wears green

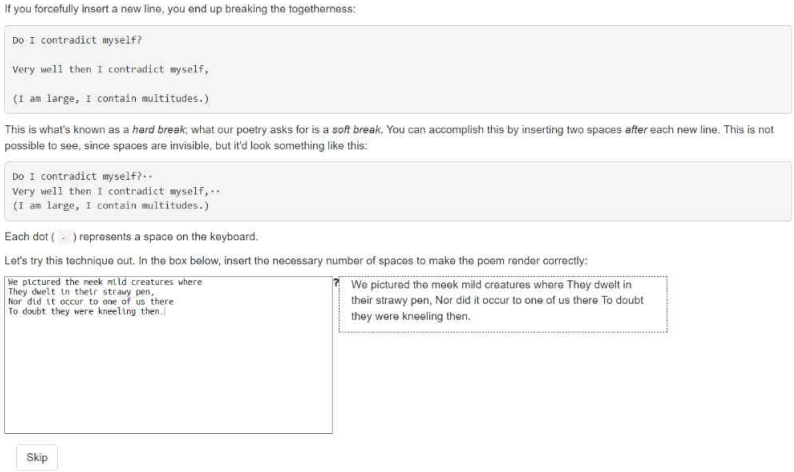
* Castafiore,
* An opera singer,
* Has white hair,
* Is very famous

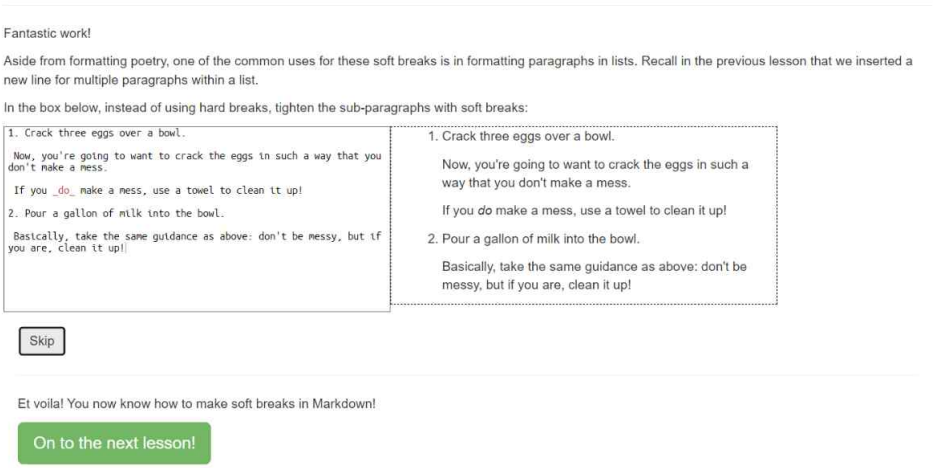


代码：1. Cut the cheese
* Make sure that the cheese is cut into little triangles

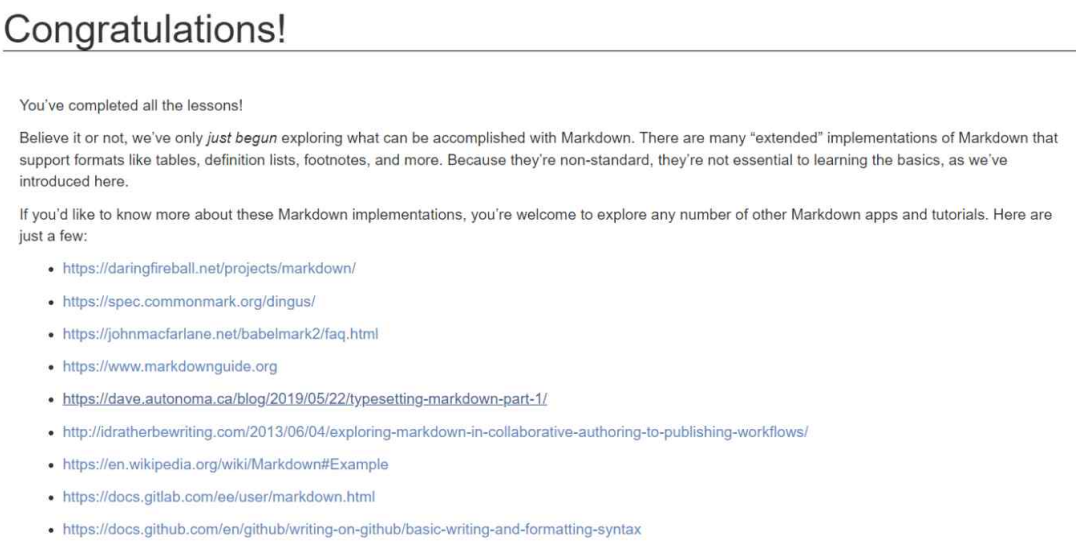
2. Slice the tomatoes
* Be careful when holding the knife.
* For more help on tomato slicing, see Thomas Jefferson's seminal essay _Tom Ate Those_.

7.Paragraphs





8.Conclusion

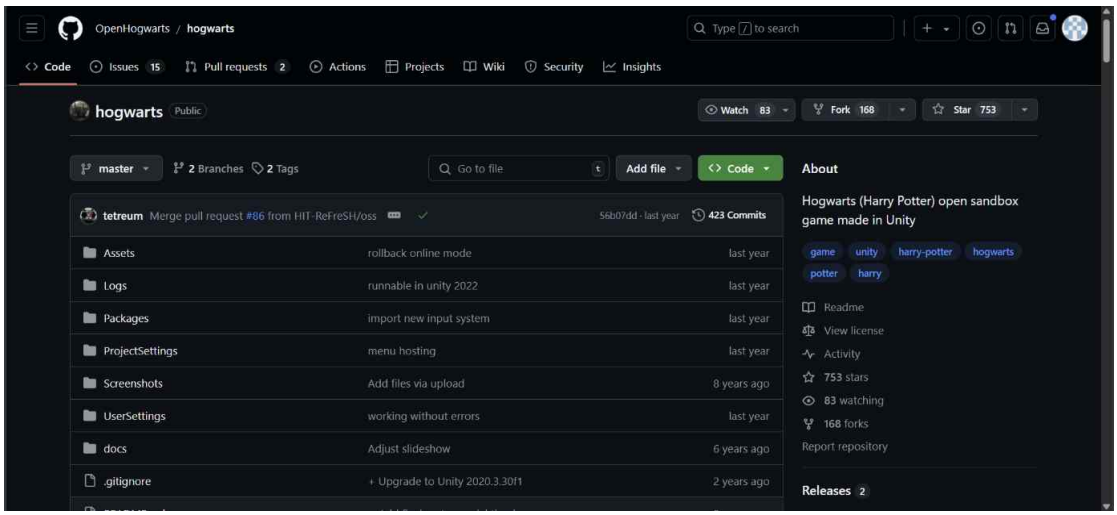


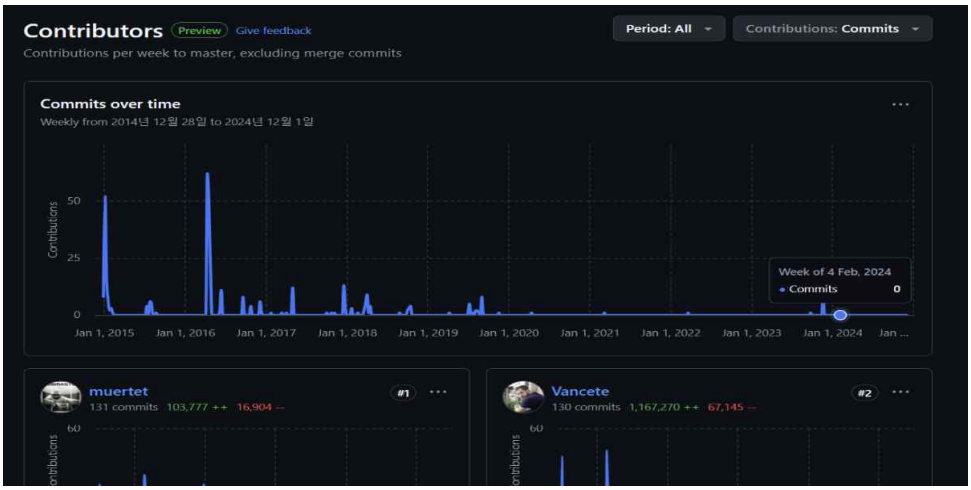
完整结束。

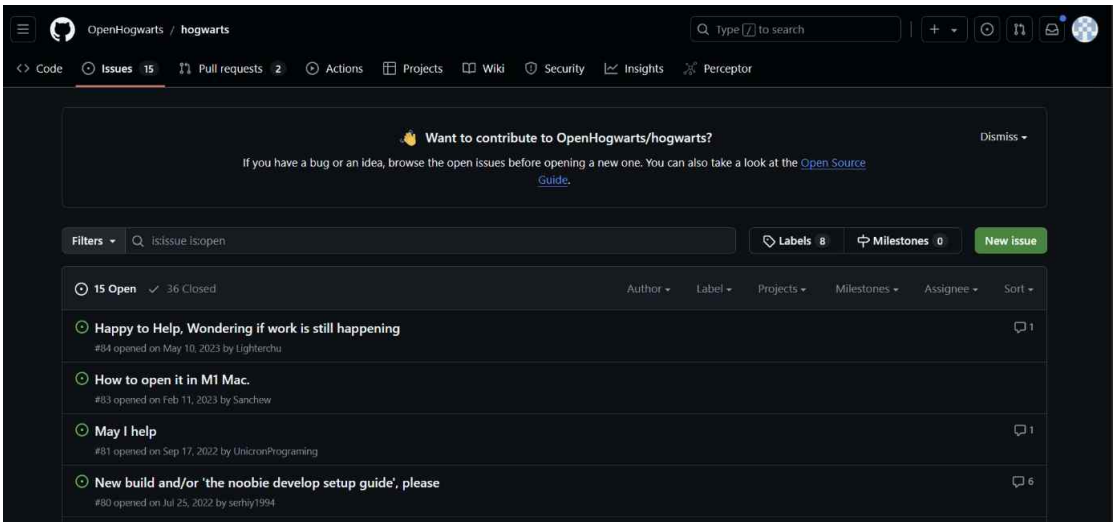
3 实验内容2 参与开源项目

3.1 分析所选项目

项目地址：**https://github.com/OpenHogwarts/hogwarts**





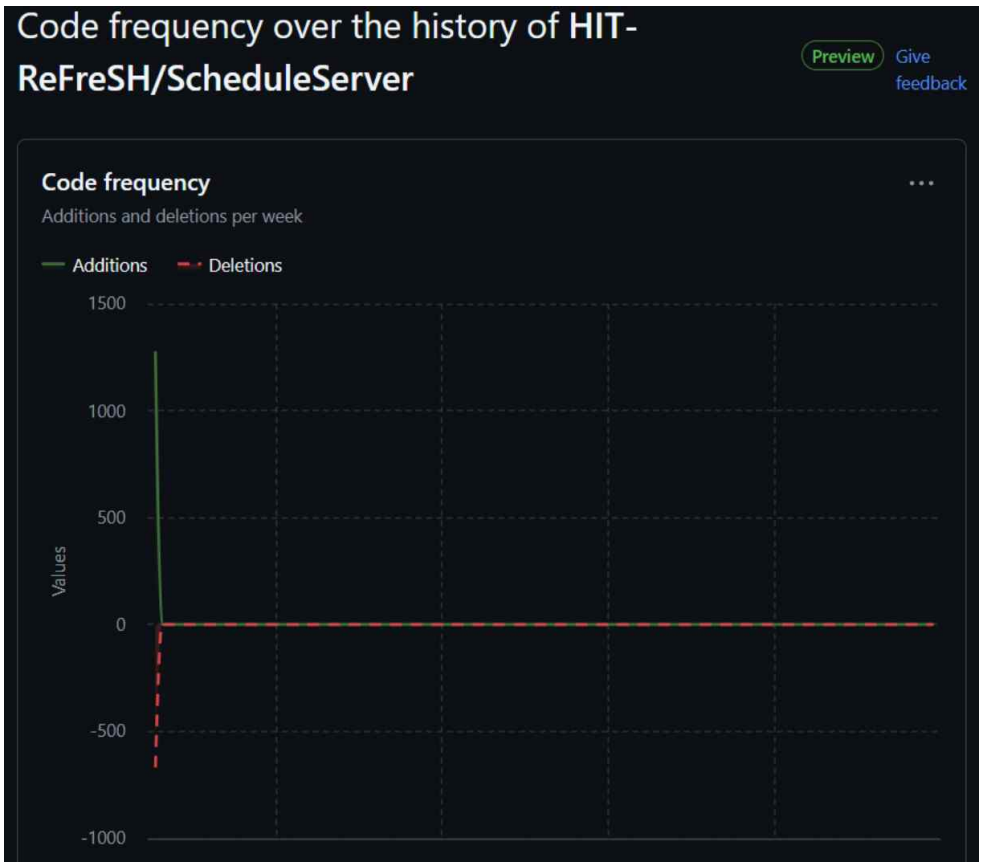


OpenHogwarts项目旨在利用Unity引擎开发一款以哈利波特世界为背景的开放式沙盒游戏。项目的结构由Assets、logs、Packages、Projects、Usersettings。这项目遵守MIT许可。分析项目活跃度，这项目的star是756个，Fork是166个，主要贡献者是Kardall, Vancete, Seyji, finchen，开发并改善多样的功能。commit比较定期的。

3.2 项目参与者贡献过程分析

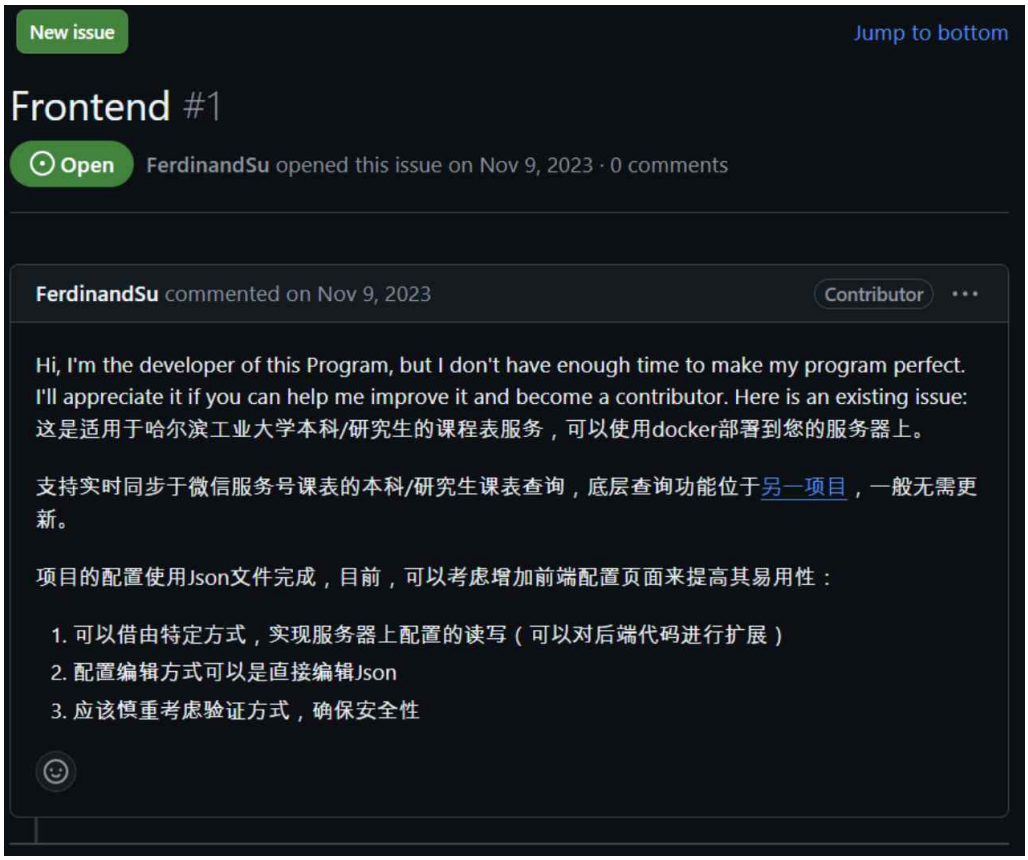
我分析的项目是HIT-ReFreSH/ScheduleServer。这项目还没PR，所以不能提交PR的过程。这项目的贡献方针可分为4个阶段。第一是Fork和Clone。将项目进行fork后，将clone放入本地机器中。第二是创建分支，创建新的分支已进行更改。第三是测试和评审，测试更改并填写评审信息，最后是创建完整PR，将更改提交到PR。

选择的贡献者是FerdinandSu。



[Code Frequency]

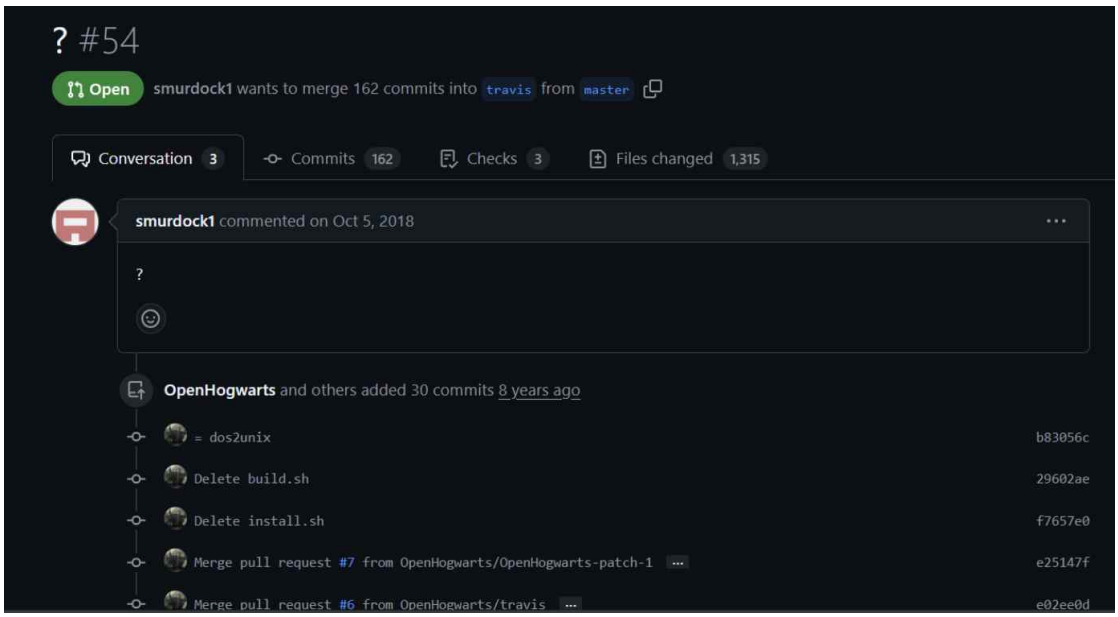
下的图片是他写的项目介绍。



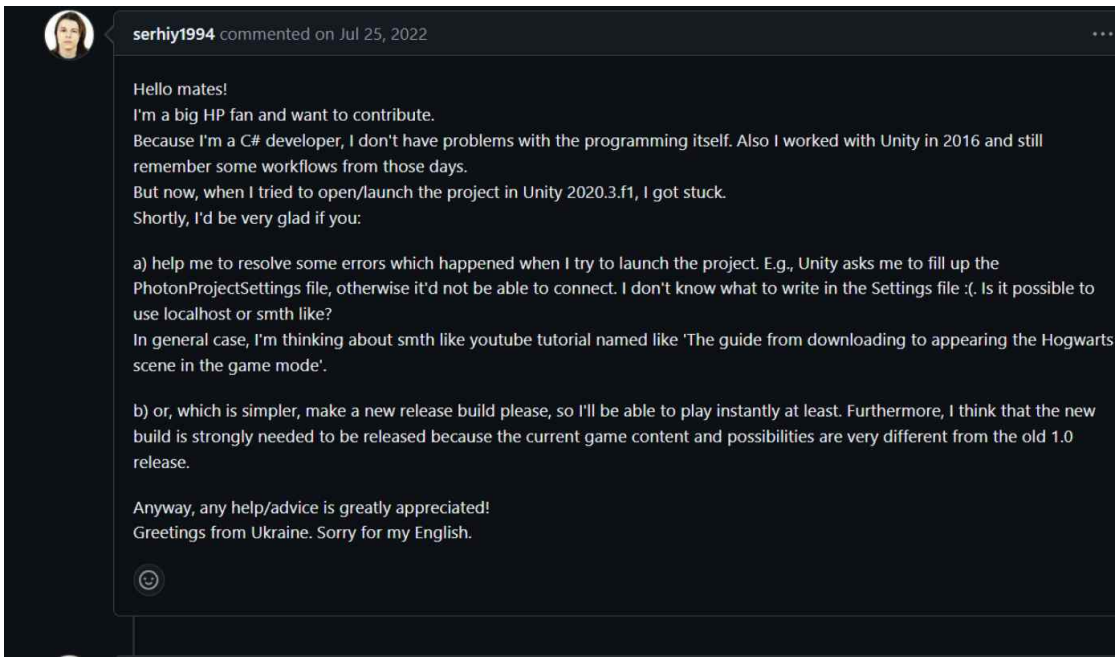
我看这个项目没有开发进度。 其依据是没有Issue交流,也没有PR的状态。 读了项目介绍，觉得是非常好的项目，只是有点可惜。 希望这个项目的参与者能够完成开发。

3.3 贡献代码

Issue 地址：**https://github.com/OpenHogwarts/hogwarts/issues/80**
PR地址：**https://github.com/OpenHogwarts/hogwarts/pulls**



[PR页]



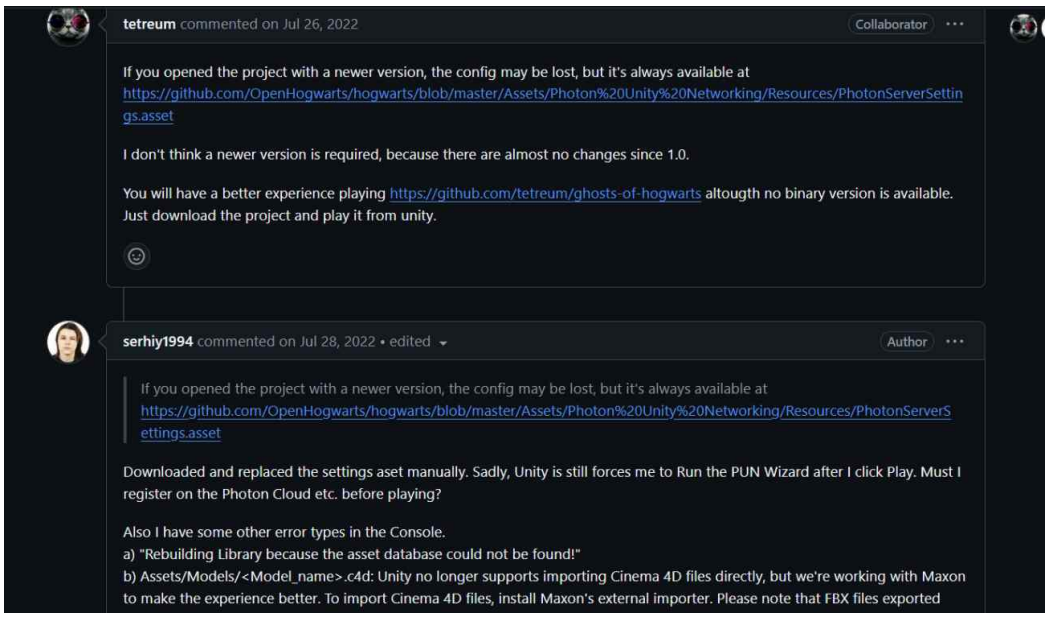
serhiy1994 commented on Jul 25, 2022

Hello mates!
I'm a big HP fan and want to contribute.
Because I'm a C# developer, I don't have problems with the programming itself. Also I worked with Unity in 2016 and still remember some workflows from those days.
But now, when I tried to open/launch the project in Unity 2020.3.f1, I got stuck.
Shortly, I'd be very glad if you:

a) help me to resolve some errors which happened when I try to launch the project. E.g., Unity asks me to fill up the PhotonProjectSettings file, otherwise it'd not be able to connect. I don't know what to write in the Settings file :(. Is it possible to use localhost or smth like?
In general case, I'm thinking about smth like youtube tutorial named like 'The guide from downloading to appearing the Hogwarts scene in the game mode'.

b) or, which is simpler, make a new release build please, so I'll be able to play instantly at least. Furthermore, I think that the new build is strongly needed to be released because the current game content and possibilities are very different from the old 1.0 release.

Anyway, any help/advice is greatly appreciated!
Greetings from Ukraine. Sorry for my English.



tetreum commented on Jul 26, 2022 Collaborator

If you opened the project with a newer version, the config may be lost, but it's always available at https://github.com/OpenHogwarts/hogwarts/blob/master/Assets/Photon%20Unity%20Networking/Resources/PhotonServerSettings.asset

I don't think a newer version is required, because there are almost no changes since 1.0.

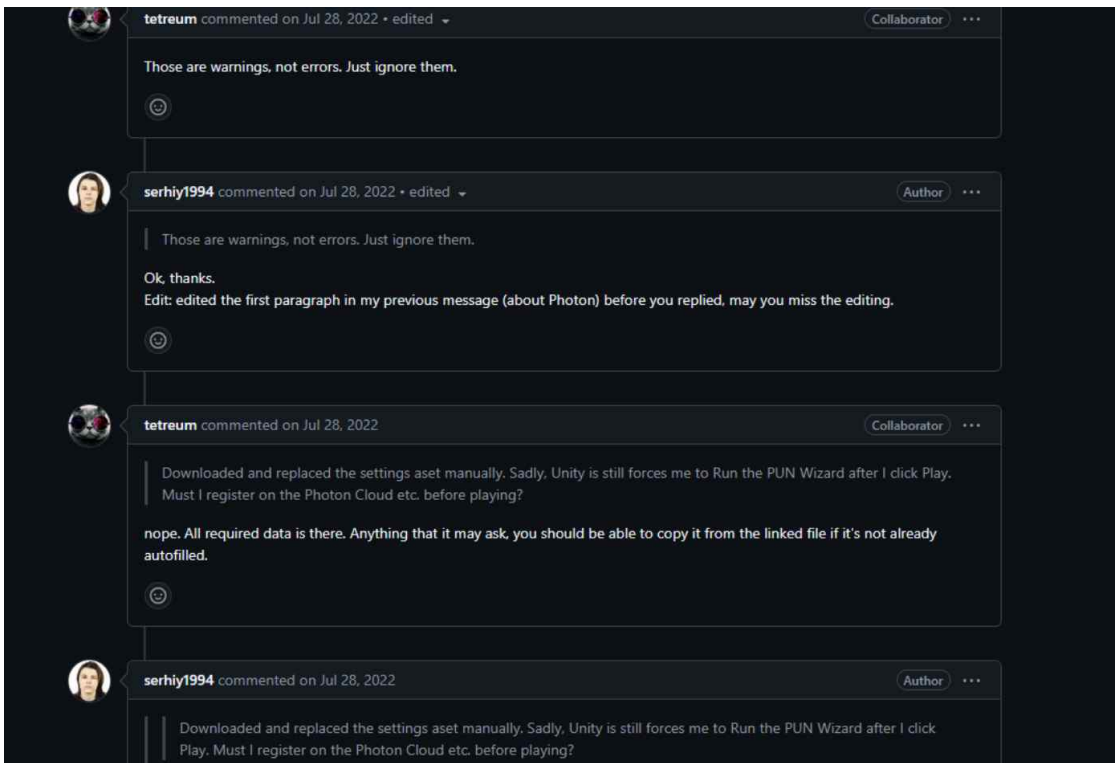
You will have a better experience playing https://github.com/tetreum/ghosts-of-hogwarts altougth no binary version is available. Just download the project and play it from unity.

serhiy1994 commented on Jul 28, 2022 • edited Author

> If you opened the project with a newer version, the config may be lost, but it's always available at https://github.com/OpenHogwarts/hogwarts/blob/master/Assets/Photon%20Unity%20Networking/Resources/PhotonServerSettings.asset

Downloaded and replaced the settings aset manually. Sadly, Unity is still forces me to Run the PUN Wizard after I click Play. Must I register on the Photon Cloud etc. before playing?

Also I have some other error types in the Console.
a) "Rebuilding Library because the asset database could not be found!"
b) Assets/Models/<Model_name>.c4d: Unity no longer supports importing Cinema 4D files directly, but we're working with Maxon to make the experience better. To import Cinema 4D files, install Maxon's external importer. Please note that FBX files exported



tetreum commented on Jul 28, 2022 • edited Collaborator

Those are warnings, not errors. Just ignore them.

serhiy1994 commented on Jul 28, 2022 • edited Author

> Those are warnings, not errors. Just ignore them.

Ok, thanks.
Edit: edited the first paragraph in my previous message (about Photon) before you replied, may you miss the editing.

tetreum commented on Jul 28, 2022 Collaborator

> Downloaded and replaced the settings aset manually. Sadly, Unity is still forces me to Run the PUN Wizard after I click Play. Must I register on the Photon Cloud etc. before playing?

nope. All required data is there. Anything that it may ask, you should be able to copy it from the linked file if it's not already autofilled.

serhiy1994 commented on Jul 28, 2022 Author

> Downloaded and replaced the settings aset manually. Sadly, Unity is still forces me to Run the PUN Wizard after I click Play. Must I register on the Photon Cloud etc. before playing?

[Issue页面]

选择了Issue是New build and/or 'the noobie develop setup guide', please。这Issue的解决方法首先为了初学者编写开发设置指南。指南的内容要包含Unity2020.3安装并设置方法、项目下载及打开、如何配置 PhotonServerSettings.asset 文件等的内容。然后创建并提交最新的构建，让新贡献者更容易实施项目。

下述是核心代码：

```
// PhotonServerSettings.asset 文件配置示例
PhotonServerSettings serverSettings = new PhotonServerSettings();
serverSettings.AppSettings.AppIdRealtime = "YOUR_APP_ID";
serverSettings.AppSettings.Server = "localhost";
PhotonNetwork.PhotonServerSettings = serverSettings;
```

4 小结

该实验首先熟悉Markdown语法,然后选择项目,分析贡献者、项目的Issue、PR。 通过这次实验,似乎很清楚应该以何种方式分析项目,Issue部分和PR部分是什么,起到什么作用。 Issue部分是项目参与者重要的沟通手段。 这是可以共享项目的改善点、好奇点等并进行反馈的空间。 因此,我认为对于第一次使用GitHub的人来说,这是非常必要的实验。 这是本人参与的项目,如果不能与参与者进行沟通,项目开发将面临困难。 如果说第二个实验是对基本分析工具的介绍，那么我认为这次实验是与之前的实验联系起来，使分析工具更加自由自在。 但是，我似乎还没有习惯使用Hypercrx等分析工具。