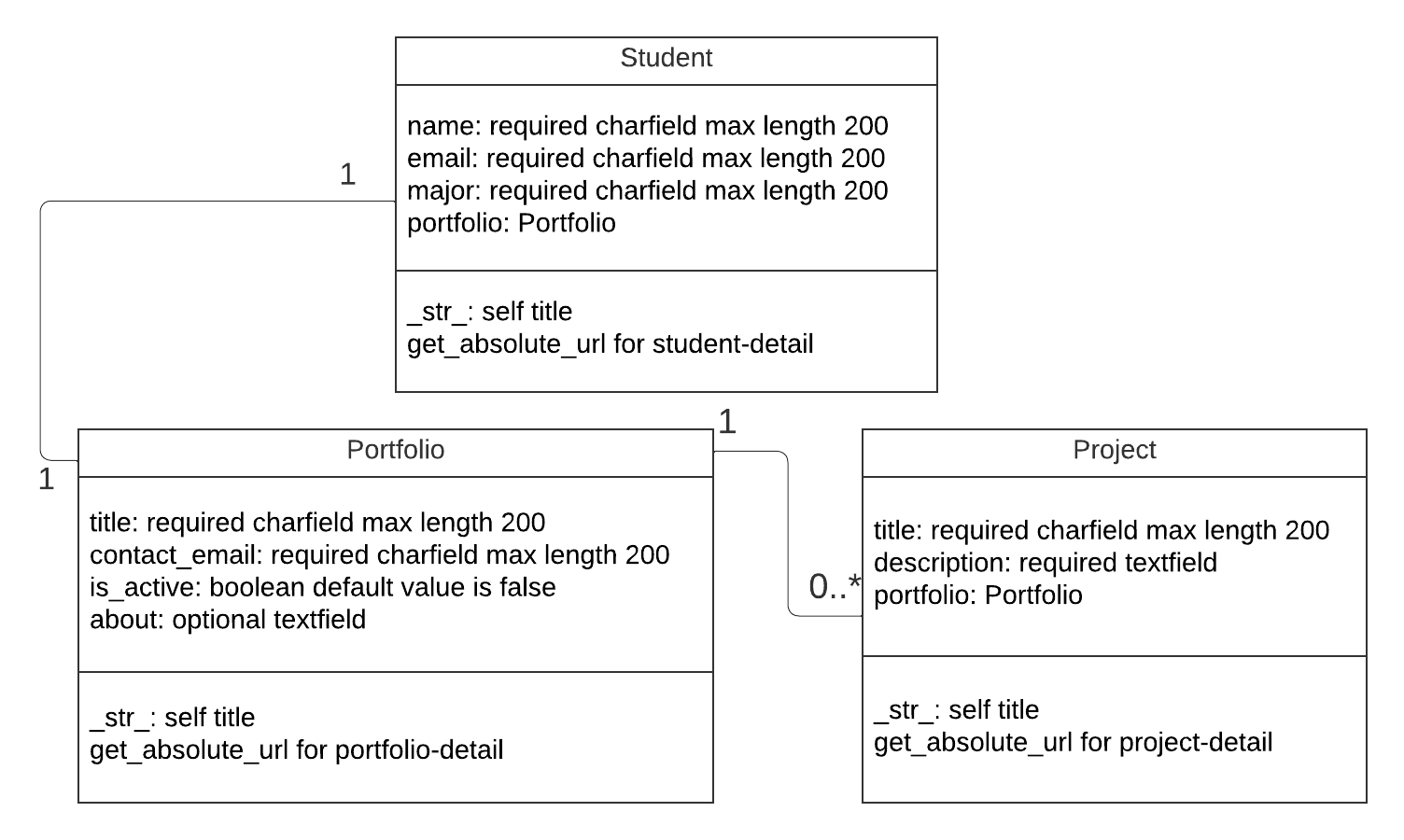
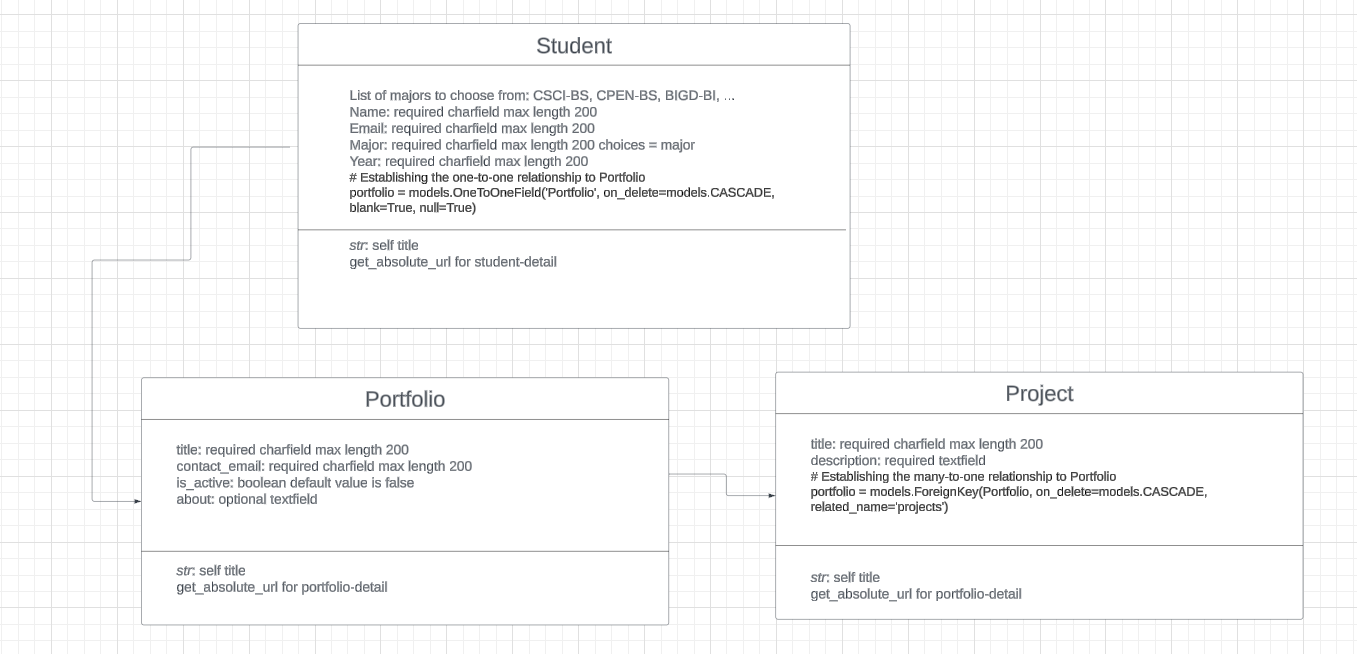
## **Models and Relationships**

UML diagram for the required and optional fields, default fields and the field types.



Created using <https://www.lucidchart.com/>



My personal created diagram.

1. Update the student model and create a portfolio and project model so that the fields are set up as shown above
   1. What is required? What type of field? Name, email, and major are required and need a charfield while year needs an int field. You need a one to one relationship with the portfolio because each student should only have one portfolio.
   2. What field do you add to the Student class to create the relationship for a Student model to have one to one portfolio.

# Establishing the one-to-one relationship to Portfolio

portfolio = models.OneToOneField('Portfolio', on\_delete=models.CASCADE, blank=True, null=True)

* 1. What field do you add to the Project class to create the relationship for a portfolio to be able to have multiple projects?

# Establishing the many-to-one relationship to Portfolio

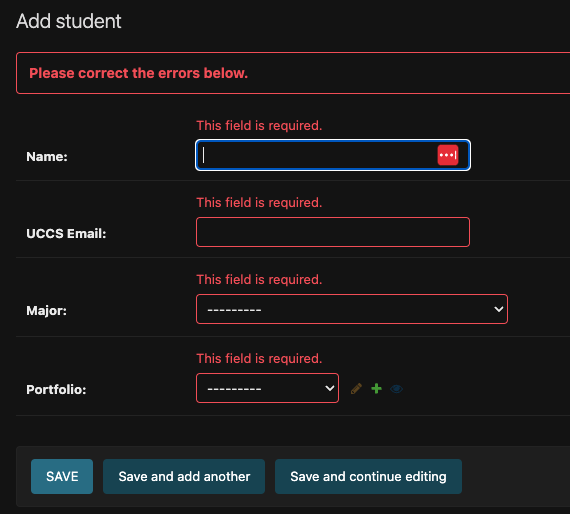
portfolio = models.ForeignKey(Portfolio, on\_delete=models.CASCADE, related\_name='projects')

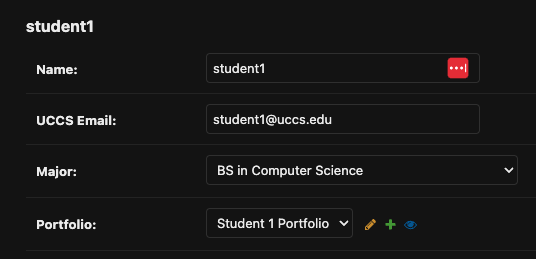
* 1. What order must the classes must be put in the models.py? Student, portfolio, then project.
  2. How do you define the str and get\_absolute\_url methods for each model?

Str returns a string that is meaningful and useful to the admin which helps represent the object and the get\_absolute\_url your URLs configuration that expects a primary key 'pk' to fetch the correct instance.

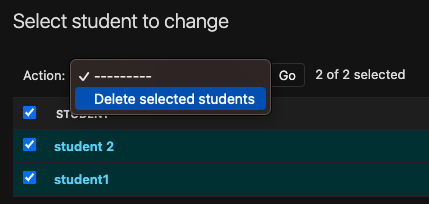
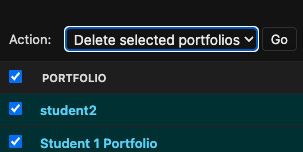
When creating your classes and relationships your database can get messed up. Here are some things I found helpful. Explore these as you add the relationships below so practice this before you create relationships in your customer app project.

* Explore the relationships after migrating class updates in the django admin app. See what is required and how the relationships work between models.





* Delete the instances in the database in the django admin app if you need to update the class model due to a mistake before migrating again.

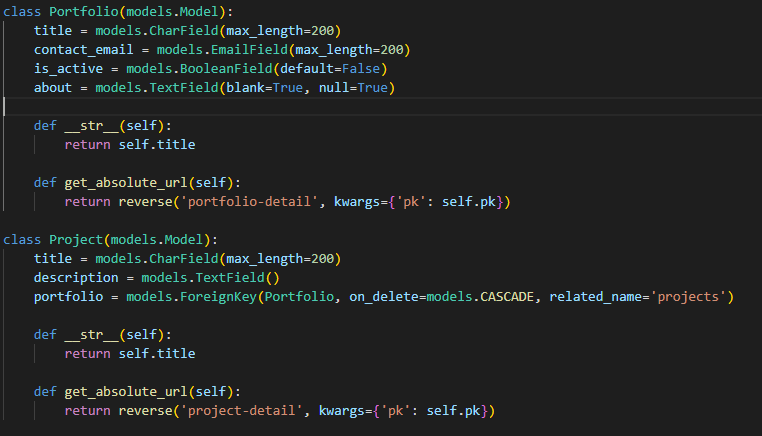
* Here is how to [Reset Database in Django](https://www.delftstack.com/howto/django/django-reset-database/) if you can’t do the deletions above due to database issues. Depending on the data you delete from the tables you may need to restart the server and create a new superuser. Don’t worry if the database gets messed up as this happens when developing an app and working with a database. That is why you have a local database associated with your own development environment.
* Read [Django One-To-Many Relationship](https://www.pythontutorial.net/django-tutorial/django-one-to-many/). If you come across the following when migrating you may need to do option 1 to fix the situation.

It is impossible to add a non-nullable field 'portfolio' to a student without specifying a default. This is because the database needs something to populate existing rows.

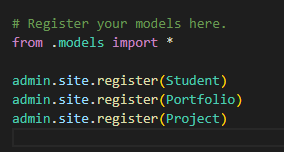
Please select a fix:

1) Provide a one-off default now (will be set on all existing rows with a null value for this column)

2) Quit and manually define a default value in models.py.

I went ahead and added the portfolio and project model to models.py and this is how it looks. 

You also need to make sure your updating the admin.py so that it will register.



Resources I found helpful:

<https://docs.djangoproject.com/en/5.0/topics/db/models/>

<https://docs.djangoproject.com/en/5.0/topics/db/examples/>  
<https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Models>