1. Name
Stanislaw Jaroszynski

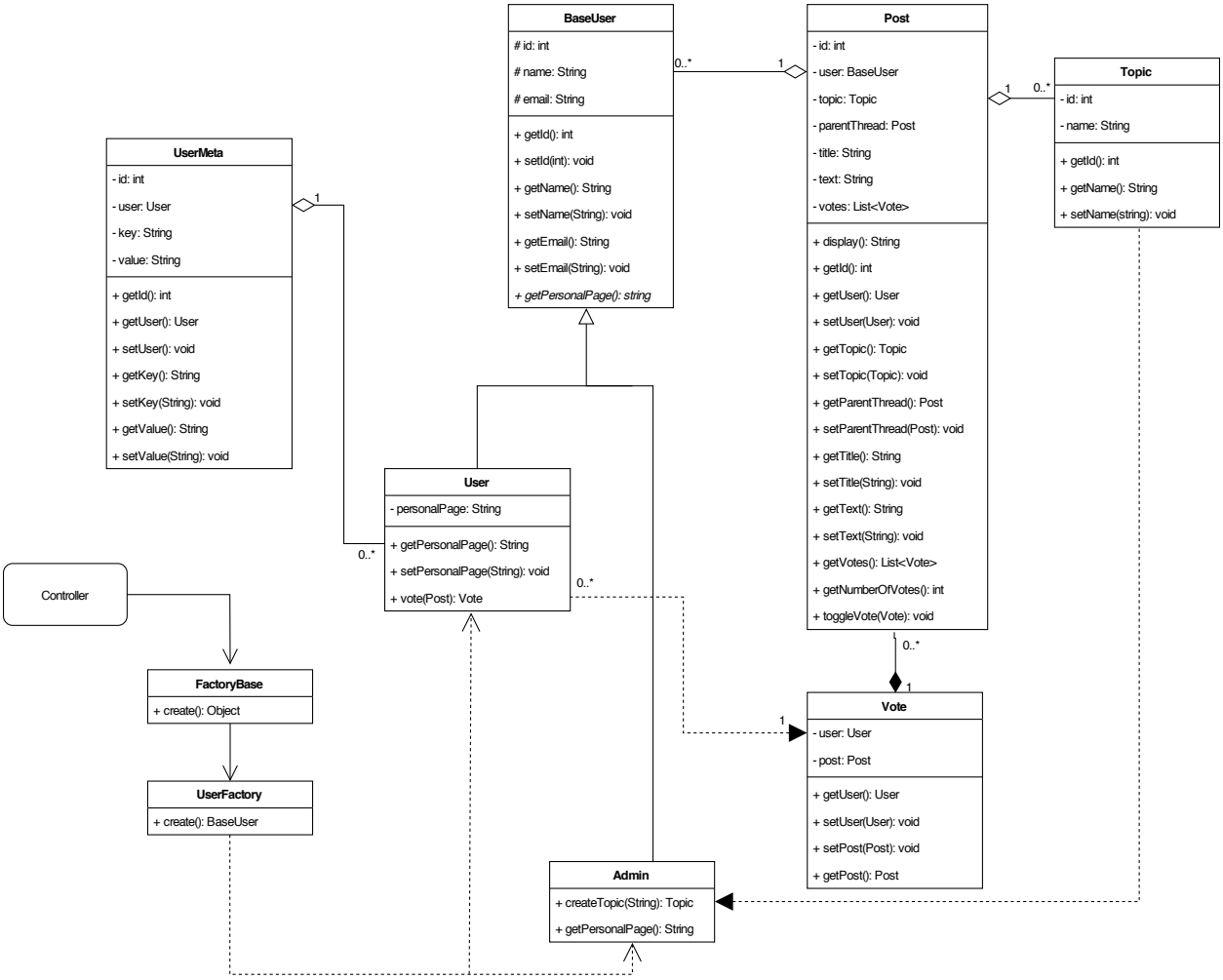
2. Project Description
Forum website that will allow users to discuss topics of interest. The Admin can create new topics. Users can discuss the topics within posts as well as interact with each others profiles.

3. Features that were implemented

ID	Description	Actors
U-01	Users can sign up	User
U-02	Users/admin can log in	User, Admin
U-03	Admin can create new topics	Admin
U-04	Users can create new posts	User, Admin
U-05	Users can browse posts	User, Admin
U-06	Users can view a post	User, Admin
U-07	Users can reply to posts	User, Admin
U-08	Users can edit their personal page	User
U-09	User can view others personal pages	User, Admin
U-10	Users can vote on posts	User
U-11	Users can see their posts	User, Admin
U-12	Users can see their replies	User, Admin

4. Features that were not implemented
All of the planned features were implemented

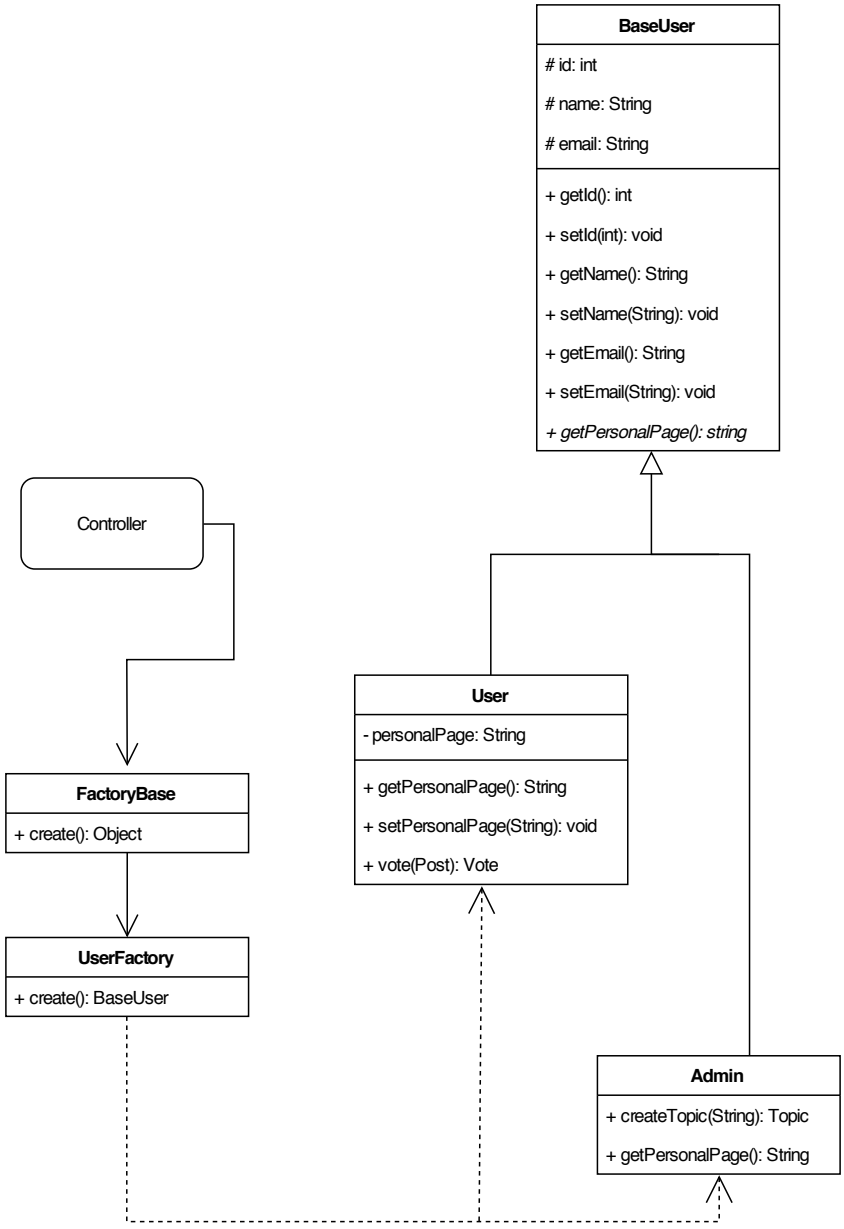
5. Final Class Diagram



What changed?

I added in the UserFactory to streamline the creation of User objects. I also added a UserMeta class to store additional metadata per user in the database without having to add columns to the database user model.

6. Factory Design Pattern



I selected this design pattern because it replaced having to explicitly call class constructors for User and Admin throughout my codebase. Depending on the user returned from the database I need to create either of the subclasses and by using the factory design pattern, I can automatically do this.

I implemented the pattern by creating a UserFactory class which is a ConcreteFactory. It has a create(userDatabaseModel):BaseUser method. This method takes a user model from the database, determines which subclass should be created and then returns a BaseUser that was initialized to the correct subclass.

7. What have I learned?

By stepping through the process of creating, designing, and implementing this system, I learned how much clearer code can be when the full potential of object-oriented programming is realized. Design patterns are the building blocks of a stable code base. I find that when I take design patterns into account at the very beginning of a project, as the project grows larger it is surprisingly easy to expand.