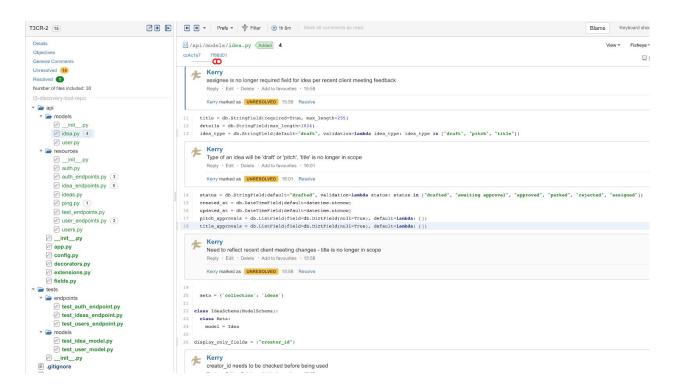**Backend Code Review:**

1. t3-discovery-tool/api/models/idea.py
   [Defect] line 26: the creator_id should be checked before use.
   [Suggestion] Check the creator_id and handle error

2. t3-discovery-tool/api/models/idea.py
   [Defect] line 10: assignee is no longer a required field for idea per recent client meeting feedback.
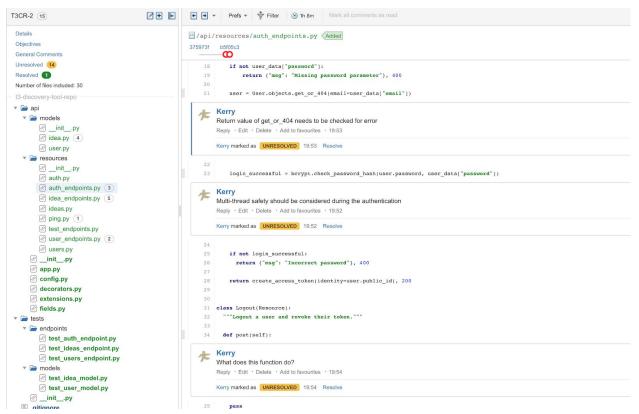   [Suggestion] Remove it

3. t3-discovery-tool/api/models/idea.py
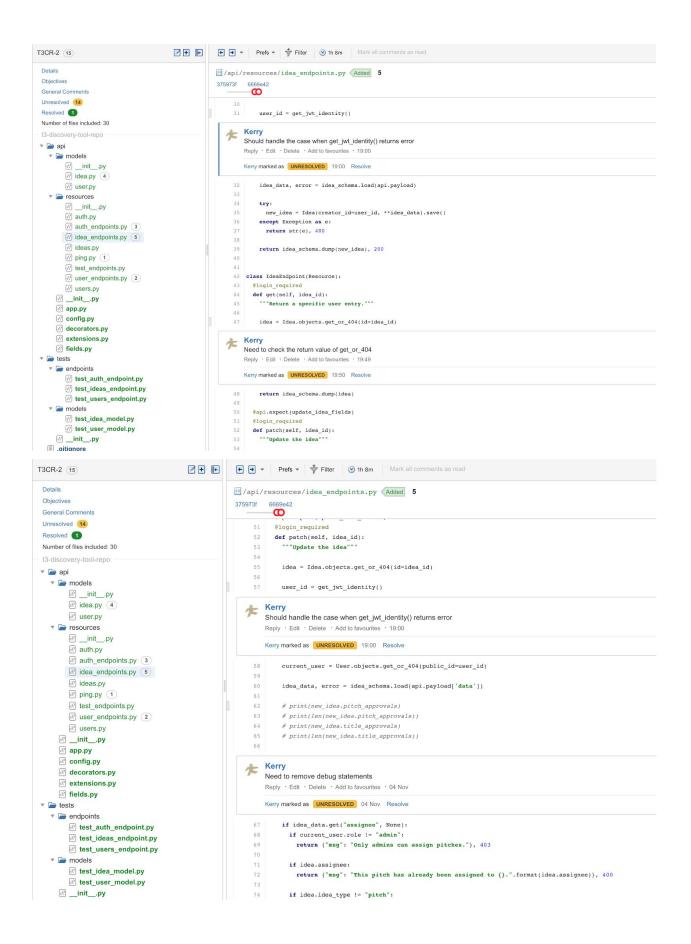   [Defect] line 13: Type of idea will only be 'draft' or 'pitch'.
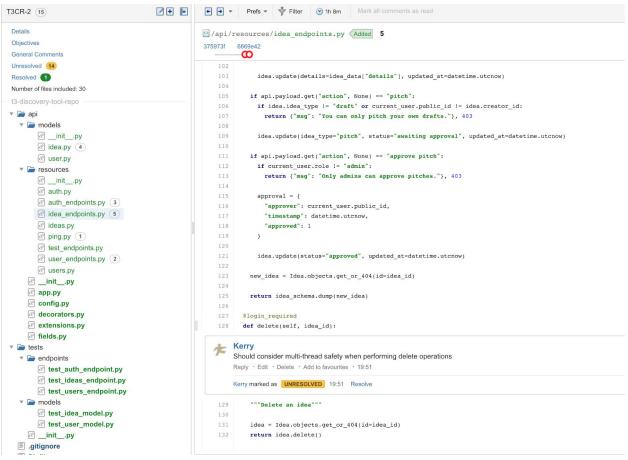   [Suggestion] Remove 'title'

4. t3-discovery-tool/api/models/idea.py
   [Defect] line 18: Need to reflect recent client meeting changes - title is no longer in scope
   [Suggestion] Remove 'title'

The screenshot shows a code review interface:

T3CR-2 (15)

- Details
- Objectives
- General Comments
- Unresolved **14**
- Resolved **1**
- Number of files included: 30

t3-discovery-tool-repo

- ▼ 📁 api
  - ▼ 📁 models
    - __init__.py
    - idea.py (4)
    - user.py
  - ▼ 📁 resources
    - __init__.py
    - auth.py
    - auth_endpoints.py (3)
    - idea_endpoints.py (5)
    - ideas.py
    - ping.py (1)
    - test_endpoints.py
    - user_endpoints.py (2)
    - users.py
  - __init__.py
  - app.py
  - config.py
  - decorators.py
  - extensions.py
  - fields.py
- ▼ 📁 tests
  - ▼ 📁 endpoints
    - test_auth_endpoint.py
    - test_ideas_endpoint.py
    - test_users_endpoint.py
  - ▼ 📁 models
    - test_idea_model.py
    - test_user_model.py
  - __init__.py
  - gitignore

Prefs ▾ Filter 1h 8m Mark all comments as read

📄 /api/resources/auth_endpoints.py (Added)

375973f   b5f05c3

```
18      if not user_data["password"]:
19          return {"msg": "Missing password parameter"}, 400
20
21      user = User.objects.get_or_404(email=user_data["email"])
```

Kerry
Return value of get_or_404 needs to be checked for error
Reply · Edit · Delete · Add to favourites · 19:53

Kerry marked as **UNRESOLVED** 19:53 Resolve

```
22
23      login_successful = bcrypt.check_password_hash(user.password, user_data["password"])
```

Kerry
Multi-thread safety should be considered during the authentication
Reply · Edit · Delete · Add to favourites · 19:52

Kerry marked as **UNRESOLVED** 19:52 Resolve

```
24
25      if not login_successful:
26          return {"msg": "Incorrect password"}, 400
27
28      return create_access_token(identity=user.public_id), 200
29
30
31  class Logout(Resource):
32      """Logout a user and revoke their token."""
33
34      def post(self):
```

Kerry
What does this function do?
Reply · Edit · Delete · Add to favourites · 19:54

Kerry marked as **UNRESOLVED** 19:54 Resolve

```
35      pass
```

5. t3-discovery-tool/api/resources/auth-endpoints.py
   [Defect] line 23: Multi-thread safety should be considered during the authentication.
   [Suggestion] Implement a locking mechanism for multiple-user scenario

6. t3-discovery-tool/api/resources/auth-endpoints.py
   [Defect] line 21: The return result from get_or_404() should be checked.
   [Suggestion] Implement logic for other branches

7. t3-discovery-tool/api/resources/auth-endpoints.py
   [Defect] line 34: What does this function do?
   [Suggestion] Seek answer from the author

Details
Objectives
General Comments
Unresolved 14
Resolved 1
Number of files included: 30

t3-discovery-tool-repo
- api
  - models
    - __init__.py
    - idea.py (4)
    - user.py
  - resources
    - __init__.py
    - auth.py
    - auth_endpoints.py (3)
    - idea_endpoints.py (5)
    - ideas.py
    - ping.py (1)
    - test_endpoints.py
    - user_endpoints.py (2)
    - users.py
  - __init__.py
  - app.py
  - config.py
  - decorators.py
  - extensions.py
  - fields.py
- tests
  - endpoints
    - test_auth_endpoint.py
    - test_ideas_endpoint.py
    - test_users_endpoint.py
  - models
    - test_idea_model.py
    - test_user_model.py
  - __init__.py
- .gitignore

Prefs ▾   Filter   1h 8m   Mark all comments as read

/api/resources/idea_endpoints.py (Added) 5

375973f   6669e42

```
30
31      user_id = get_jwt_identity()
```

Kerry
Should handle the case when get_jwt_identity() returns error
Reply · Edit · Delete · Add to favourites · 19:00

Kerry marked as UNRESOLVED 19:00 Resolve

```
32      idea_data, error = idea_schema.load(api.payload)
33
34      try:
35          new_idea = Idea(creator_id=user_id, **idea_data).save()
36      except Exception as e:
37          return str(e), 400
38
39      return idea_schema.dump(new_idea), 200
40
41
42  class IdeaEndpoint(Resource):
43      @login_required
44      def get(self, idea_id):
45          """Return a specific user entry."""
46
47          idea = Idea.objects.get_or_404(id=idea_id)
```

Kerry
Need to check the return value of get_or_404
Reply · Edit · Delete · Add to favourites · 19:49

Kerry marked as UNRESOLVED 19:50 Resolve

```
48          return idea_schema.dump(idea)
49
50      @api.expect(update_idea_fields)
51      @login_required
52      def patch(self, idea_id):
53          """Update the idea"""
54
```

Prefs ▾   Filter   1h 8m   Mark all comments as read

/api/resources/idea_endpoints.py (Added) 5

375973f   6669e42

```
51      @login_required
52      def patch(self, idea_id):
53          """Update the idea"""
54
55          idea = Idea.objects.get_or_404(id=idea_id)
56
57          user_id = get_jwt_identity()
```

Kerry
Should handle the case when get_jwt_identity() returns error
Reply · Edit · Delete · Add to favourites · 19:00

Kerry marked as UNRESOLVED 19:00 Resolve

```
58          current_user = User.objects.get_or_404(public_id=user_id)
59
60          idea_data, error = idea_schema.load(api.payload['data'])
61
62          # print(new_idea.pitch_approvals)
63          # print(len(new_idea.pitch_approvals))
64          # print(new_idea.title_approvals)
65          # print(len(new_idea.title_approvals))
66
```

Kerry
Need to remove debug statements
Reply · Edit · Delete · Add to favourites · 04 Nov

Kerry marked as UNRESOLVED 04 Nov Resolve

```
67          if idea_data.get("assignee", None):
68              if current_user.role != "admin":
69                  return {"msg": "Only admins can assign pitches."}, 403
70
71              if idea.assignee:
72                  return {"msg": "This pitch has already been assigned to {}.".format(idea.assignee)}, 400
73
74              if idea.idea_type != "pitch":
```

8. t3-discovery-tool/api/resources/idea-endpoints.py
   [Defect] line 31,57: Should consider get_jwt_identity() error here and solve it.
   [Suggestion] Implement error handling

9. t3-discovery-tool/api/resources/idea-endpoints.py
   [Defect] line 47: Should check the return value of Idea.objects.get_or_404()
   [Suggestion] Implement error handling

10. t3-discovery-tool/api/resources/idea-endpoints.py
    [Defect] line 67: Other branches should be considered.
    [Suggestion] Implement logic for other branches

11. t3-discovery-tool/api/resources/idea-endpoints.py
    [Defect] line 128: Multi-thread safety should be considered when doing delete operations
    [Suggestion] Implement a locking mechanism for multiple-user scenario

12. t3-discovery-tool/api/resources/idea_endpoints.py
    [Defect] line 62: Need to remove debug statements
    [Suggestion] Remove them

Prefs ▾ | Filter | ⏱ 1h 8m | Mark all comments as read

◻ /api/resources/`user_endpoints.py` (Added) 2

375973f   e223fb3

```
 3
 4    from flask_restplus import Resource
 5
 6    from ..extensions import api, bcrypt
 7    from ..models import User, user_schema, users_schema
 8    from ..decorators import admin_required, login_required
 9    from ..fields import user_fields, user_credential_fields, update_user_fields
10
11
12    class UsersEndpoint(Resource):
```

Kerry
The class name UsersEndpoint may be confusing with UserEndpoint, better change it
Reply · Edit · Delete · Add to favourites · 18:59

Kerry marked as UNRESOLVED 18:59 Resolve

```
13        @admin_required
14        def get(self):
15            """Return a list of users."""
16
17            return users_schema.dump(User.objects.all())
18
19        @api.expect(user_fields)
20        @admin_required
21        def post(self):
22            """Create a new user."""
23
24            user_data, error = user_schema.load(api.payload)
25            user_data["public_id"] = uuid.uuid4()
26
27            try:
```

Prefs ▾ | Filter | ⏱ 1h 9m | Mark all comments as read

◻ /api/resources/`user_endpoints.py` (Added) 2

375973f   e223fb3

```
37            return str(e), 400
38
39        return user_schema.dump(new_user), 200
40
41
42    class UserEndpoint(Resource):
43        @login_required
44        def get(self, user_id):
45            """Return a specific user entry."""
46
47            user = User.objects.get_or_404(public_id=user_id)
48            return user_schema.dump(user)
49
50        @api.expect(update_user_fields)
51        @admin_required
52        def put(self, user_id):
53            """Update a user entry."""
54
55            user_data, error = user_schema.load(api.payload['data'])
56
57            user = User.objects.get_or_404(public_id=user_id)
58            user.update(updated_at=datetime.utcnow, **user_data)
59
60            return user_schema.dump(user)
61
62        @admin_required
63        def delete(self, user_id):
```

Kerry
Multi-thread safety should be considered when doing delete operations
Reply · Edit · Delete · Add to favourites · 18:59

Kerry marked as UNRESOLVED 18:59 Resolve

```
64            """Delete a specific user entry."""
65
66            user = User.objects.get_or_404(public_id=user_id)
67            return user.delete()
```

13. t3-discovery-tool/api/resources/user-endpoints.py
    [Defect] line 12: The class name UsersEndpoint may be confused with user-Endpoint, better to change to another name.
    [Suggestion] Rename the class name to something more distinct

14. t3-discovery-tool/api/resources/user-endpoints.py
    [Defect] line 63: Multi-thread safety should be considered when doing delete operations.
    [Suggestion] Implement a locking mechanism for multiple-user scenario

15. Files structure
    [Defects] Need more clear file structure. There are many separated files in the backend baseline, and it is difficult for other developer to understand these codes at first.
    [Suggestion] we can divide the whole backend part into three file folders such as model part, view part, and controller part. In this way, users or other developer could understand the function of code part easily