```python
#app/main/forms.py
from flask_wtf import FlaskForm
from wtforms import StringField, TextAreaField, SubmitField, BooleanField, SelectField
from wtforms.validators import Length, DataRequired, Email, Regexp, ValidationError
from ..models import User, Role


class EditProfileForm(FlaskForm):
    name = StringField('Real Name', validators=[Length(0,64)])
    location = StringField('Location', validators=[Length(0,64)])
    about_me = TextAreaField('About Me')
    submit = SubmitField('Submit')


class EditProfileAdminForm(FlaskForm):
    email = StringField('Email', validators=[
                DataRequired(), Length(0, 64),
                Email()])
    username = StringField('Username', validators=[
                DataRequired(), Length(0, 64),
                Regexp('^[A-Za-z][A-Za-z0-9_.]*$', 0,
                'Username must have only letters, numbers, dots or underscores')])
    # confirmed = BooleanField('Confirmed')
    role = SelectField('Role', coerce=int)
    name = StringField('Real Name', validators=[Length(0, 64)])
    location = StringField('Location', validators=[Length(0, 64)])
    about_me = StringField('About Me')
    submit = SubmitField()


    def __init__(self, user, *args, **kwargs):
        super(EditProfileAdminForm, self).__init__(*args, **kwargs)
        self.role.choices = [(role.id, role.name)
                        for role in Role.query.order_by(Role.name).all()]
        self.user = user

    def validate_email(self, field):
        if field.data != self.user.email and \
                User.query.filter_by(email=field.data).first():
            raise ValidationError('Email already registered.')

    def validate_username(self, field):
        if field.data != self.user.username and \
                User.query.filter_by(username=field.data).first():
            raise ValidationError('Username already in use')
```