Python flask.request.form() Examples

The following are code examples for showing how to use *flask.request.form()*. They are from open source Python projects. You can vote up the examples you like or vote down the ones you don't like.

Example 1

Project: *everyclass-server* Author: *everyclass* File: *views.py* Mozilla Public License 2.0

```python
def register():
    """注册：第一步：输入学号"""
    if request.method == 'GET':
        return render_template('user/register.html')
    else:
        if not request.form.get("xh", None):  # 表单为空
            flash(MSG_EMPTY_USERNAME)
            return redirect(url_for("user.register"))

        _session_save_student_to_register_(request.form.get("xh", None))

        # 如果输入的学号已经注册，跳转到登录页面
        if User.exist(session[SESSION_STUDENT_TO_REGISTER].sid_orig):
            flash(MSG_ALREADY_REGISTERED)
            return redirect(url_for('user.login'))

        return redirect(url_for('user.register_choice'))
```

Example 2

Project: *ras-frontstage* Author: *ONSdigital* File: *create_message.py* MIT License

```python
def create_message(session):
    """Creates and sends a message outside of the context of an existing conversat
    survey = request.args['survey']
    ru_ref = request.args['ru_ref']
    party_id = session['party_id']
    form = SecureMessagingForm(request.form)
    if request.method == 'POST' and form.validate():
        logger.info("Form validation successful", party_id=party_id)
        sent_message = _send_new_message(party_id, survey, ru_ref)
        thread_url = url_for("secure_message_bp.view_conversation",
                             thread_id=sent_message['thread_id']) + "#latest-messa
        flash(Markup('Message sent. <a href={}>View Message</a>'.format(thread_url
        return redirect(url_for('secure_message_bp.view_conversation_list'))

    else:
        return render_template('secure-messages/secure-messages-view.html',
                               ru_ref=ru_ref, survey=survey,
                               form=form, errors=form.errors, message={})
```

Example 3

Project: *pysos* Author: *xmnlab* File: *views.py* GNU General Public License v3.0

```python
def register_view(self):
    form = RegistrationForm(request.form)
    if helpers.validate_form_on_submit(form):
```

```
            user = User()

            form.populate_obj(user)
            # we hash the users password to avoid saving it as plaintext in the db
            # remove to use plain text:
            user.password = generate_password_hash(form.password.data)

            db_session.add(user)
            db_session.commit()

            login.login_user(user)
            return redirect(url_for('.index'))
        link = '<p>Already have an account? <a href="' + url_for('.login_view') +
        self._template_args['form'] = form
        self._template_args['link'] = link
        return super(PySOSAdminView, self).index()
```

Example 4

```
def upload():
    f = request.files['file']
    assert f, "Where's my file?"

    filekey = request.form.get('filekey') or str(uuid.uuid1())
    assert RE_ALLOWED_FILEKEYS.match('filekey'), 'Unacceptable file key'

    permpath = getpath(filekey)

    content_range = (f.headers.get('Content-Range') or
                     request.headers.get('Content-Range'))

    if content_range:
        result, kwargs = handle_chunked(f, permpath, content_range)
    else:
        result, kwargs = handle_full(f, permpath)

    kwargs['filekey'] = filekey

    return jsonify(result=result, **kwargs)


# Flask endpoint
```

Example 5

```
def update_antiraid(server_id):
    log.info(request.form)
    data = dict(request.form)
    data.pop("_csrf_token",None)
    role = data.pop("mute_roles",None)
    db.delete("{}:AntiRaid:mute_roles".format(server_id))
    if role is not None:
        db.sadd('{}:AntiRaid:mute_roles'.format(server_id),*role)
    old_data = data.copy()
    data = {key:value[0] for key,value in data.items() if value[0].isdigit()}
    if len(old_data) != len(data):
        flash("At least one of those values has to be positive INTEGER",'warning')
```

```
        return
    db.hmset("{}:AntiRaid:Config".format(server_id),data)
    db.sadd("Info:AntiRaid",server_id)
    flash('Settings updated!', 'success')
    return dashboard(server_id=server_id)
```

Example 6

```
def update_discourse(server_id):
    log.info(request.form)
    domain = request.form.get('domain')
    api_key = request.form.get('api_key')
    username = request.form.get('username')
    channel = request.form.get('channel')
    msg_template = request.form.get("msg")

    if len(domain) == 0 or len(api_key) == 0 or len(username) == 0 or len(msg_temp
        flash("One of them need to be filled!", 'warning')
    else:
        db.hset("{}:Discourse:Config".format(server_id), "domain", domain.strip("/
        db.hset("{}:Discourse:Config".format(server_id), "api_key", api_key)
        db.hset("{}:Discourse:Config".format(server_id), "username", username)
        db.hset("{}:Discourse:Config".format(server_id), "channel", channel)
        db.hset("{}:Discourse:Config".format(server_id), "msg", msg_template)
        currently_topic = discourse(domain, api_key, username)
        if currently_topic is None:
            flash("There seem to be problem, please double check with domain,api k
        else:
            currently_topic = find_latest_topic(currently_topic)
            db.set("{}:Discourse:ID".format(server_id), currently_topic)
            flash('Settings updated!', 'success')
    return dashboard(server_id = server_id)
```

Example 7

```
def update_trust_level(server_id):
    def add_role(i):
        r =request.form.get("trust{}".format(i)).split(',')
        if len(r) > 0:
            db.sadd("{}:Discourse:trust_role{}".format(server_id,i),*r)
            return True
    isAssign = False
    for i in range(1,5):
        db.delete("{}:Discourse:trust_role{}".format(server_id, i)) #just in case
        gotAssign = add_role(i) #then add role. No need to make another loops
        if gotAssign:
            isAssign = True

    if isAssign:
        flash('Settings updated!', 'success')
        db.set("{}:Discourse:trust_bool".format(server_id),1)
    else:
        db.set("{}:Discourse:trust_bool".format(server_id), 0)

    return redirect(url_for("discourse.trust_level",server_id = server_id))
```

Example 8

```python
def update_profile(): #Update a setting.
    list_point = dict(request.form)
    list_point.pop('_csrf_token',None)
    path = "Profile:{}".format(session['user']['id'])
    warning = False
    warning_msg = "One of those have failed, Please double check {} "
    warning_list =[]
    for  x in list_point:
        print(x)
        if request.form.get(x) == "":
            db.hdel(path,x)
            continue
        elif x == "osu":
            results = osu_api.get_user(request.form.get(x))
            if results == []:
                warning = True
                warning_list.append(x)
                continue
        db.hset(path,x,request.form.get(x))
    if warning:
        flash(warning_msg.format(",".join(warning_list)), 'warning')
    else:
        flash('Settings updated!', 'success')
    return redirect(url_for('profile.profile'))
```

Example 9

```python
def create_token():
    key = current_app.config['priv_key']
    try:
        data = request.form
        if data.get('grant_type') != 'client_credentials':
            return _400('Wrong grant_type')

        client_id = data.get('client_id')
        client_secret = data.get('client_secret')
        aud = data.get('audience', '')

        if not is_authorized_app(client_id, client_secret):
            return abort(401)

        now = int(time.time())

        token = {'iss': 'runnerly-tokendealer',
                 'aud': aud,
                 'iat': now,
                 'exp': now + 3600 * 24}
        token = jwt.encode(token, key, algorithm='RS512')
        return {'access_token': token.decode('utf8')}
    except Exception as e:
        return _400(str(e))
```

Example 10

```python
def _send_new_message(party_id, survey, business_id):
    logger.info('Attempting to send message', party_id=party_id, business_id=busin
    form = SecureMessagingForm(request.form)

    subject = form['subject'].data if form['subject'].data else form['hidden_sub
    message_json = {
        "msg_from": party_id,
        "msg_to": ['GROUP'],
        "subject": subject,
        "body": form['body'].data,
        "thread_id": form['thread_id'].data,
        "business_id": business_id,
        "survey": survey,
    }

    response = conversation_controller.send_message(json.dumps(message_json))

    logger.info('Secure message sent successfully',
                message_id=response['msg_id'], party_id=party_id, business_id=busi
    return response
```

Example 11

6 vo

```python
def get_reset_password(token, form_errors=None):
    form = ResetPasswordForm(request.form)

    try:
        party_controller.verify_token(token)
    except ApiError as exc:
        if exc.status_code == 409:
            logger.warning('Token expired', api_url=exc.url, api_status_code=exc.s
            return render_template('passwords/password-expired.html', token=token)
        elif exc.status_code == 404:
            logger.warning('Invalid token sent to party service', api_url=exc.url,
                           token=token)
            abort(404)
        else:
            raise exc

    template_data = {
        "error": {
            "type": form_errors
        },
        'token': token
    }
    return render_template('passwords/reset-password.html', form=form, data=templ
```

Example 12

6 vo

```python
def post_reset_password(token):
    form = ResetPasswordForm(request.form)

    if not form.validate():
        return get_reset_password(token, form_errors=form.errors)

    password = request.form.get('password')
```

```python
    try:
        party_controller.change_password(password, token)
    except ApiError as exc:
        if exc.status_code == 409:
            logger.warning('Token expired', api_url=exc.url, api_status_code=exc.s
            return render_template('passwords/password-expired.html', token=token)
        elif exc.status_code == 404:
            logger.warning('Invalid token sent to party service', api_url=exc.url,
                           token=token)
            abort(404)
        else:
            raise exc

    logger.info('Successfully changed user password', token=token)
    return redirect(url_for('passwords_bp.reset_password_confirmation'))
```

Example 13

```python
def post_forgot_password():
    form = ForgotPasswordForm(request.form)
    form.email_address.data = form.email_address.data.strip()
    email = form.data.get('email_address')

    encoded_email = url_safe_serializer.dumps(email)

    if form.validate():

        try:
            party_controller.reset_password_request(email)
        except UserDoesNotExist:
            logger.info('Requesting password change for unregistered email in part
            return redirect(url_for('passwords_bp.forgot_password_check_email', em

        logger.info('Successfully sent password change request email')
        return redirect(url_for('passwords_bp.forgot_password_check_email', email=

    return render_template('passwords/forgot-password.html', form=form, email=ema
```

Example 14

```python
def api_update(webhook_id: int):
    with get_db() as DB:
        c = DB.cursor()
        c.execute(
            """
            UPDATE webhooks SET
            type = ?,
            endpointUrl = ?,
            authorizationHeader = ?,
            enabled = ?,
            timestamp = date('now')
            WHERE id == ?;
            """,
            (
                request.form["type"],
                request.form["endpoint_url"],
                request.form["authorization_header"],
```

```
            0 if "enabled" not in request.form else request.form["enabled"],
            webhook_id
        )
    )
    DB.commit()
    return redirect("/settings/webhooks")
```

Example 15

```
def api_create():
    with get_db() as DB:
        c = DB.cursor()
        c.execute(
            """
            INSERT INTO webhooks
            (type, endpointUrl, authorizationHeader, enabled)
            VALUES
            (?, ?, ?, ?);
            """,
            (
                request.form["type"],
                request.form["endpoint_url"],
                request.form["authorization_header"],
                0 if "enabled" not in request.form else request.form["enabled"]
            )
        )
        DB.commit()
    return redirect("/settings/webhooks")
```

Example 16

```
def test_add():
    if request.method == "POST":
        users = Users(
            request.form["username"],
            request.form["fullname"],
            request.form["email"],
            request.form["facebookurl"],
            request.form["instagram"],
            request.form["chapter"],
            request.form["password"],
        )
        session.add(users)
        session.commit()
        print(users)
        return redirect(url_for("show_all"))
    return render_template("adduser.html")
```

Example 17

```
def login():
    # ..
    form = LoginForm(request.form)
```

```
    if request.method == 'POST':
        if form.validate_on_submit():
            login_user(form.user)
            redirect_url = request.args.get('next') or url_for('admin.index')
            return redirect(redirect_url)
        else:
            flash_errors(form)
    # ..
    return render_template('accounts/login.html', form=form)
```

Example 18

```
def addNewUser():
    username = request.form["username"]
    email = request.form["email"]
    password = request.form["password"]

    info = {"userid":1,
            "name":username,
            "email":email,
            "password":password
            }
    return jsonify(status=addUser(info))
```

Example 19

```
def login():
    if 'loggedin' in session:
        return jsonify({"status":True})

    name = str(request.form["username"])
    password = str(request.form["password"])
    status=checkLogin(name,password)

    if status==True:
        session["loggedin"]=True
    return jsonify(status=status)
```

Example 20

```
def checkUser():
    _name = str(request.form["name"])
    return jsonify(present=checkUserPresence(_name))
```

Example 21

```
def password_strength_check():
    """AJAX 密码强度检查"""
    if request.form.get("password", None):
        # 密码强度检查
        pwd_strength_report = zxcvbn(password=request.form["password"])
```

```
        if pwd_strength_report['score'] < 2:
            return jsonify({"strong": False,
                            "score" : pwd_strength_report['score']})
        else:
            return jsonify({"strong": True,
                            "score" : pwd_strength_report['score']})
    return jsonify({"invalid_request": True})
```

Example 22

```
def js_set_preference():
    """AJAX更新偏好设置"""
    if request.form.get("privacyLevel", None):
        # update privacy level
        privacy_level = int(request.form["privacyLevel"])
        if privacy_level not in (0, 1, 2):
            logger.warn("Received malformed set preference request. privacyLevel v
            return jsonify({"acknowledged": False,
                            "message"     : "Invalid value"})

        PrivacySettings.set_level(session[SESSION_CURRENT_USER].sid_orig, privacy_
    return jsonify({"acknowledged": True})
```

Example 23

```
def handle_slack():
    pone = ponies.find_ponies(request.form['text'])
    if len(pone) == 0:
        return "No ponies found."
    url = pone[0]['representations']['small']
    return jsonify(attachments=[{
        "fallback": url,
        "image_url": "https:"+url,
        "title": request.form['text']
    }], text="", response_type="in_channel")
```

Example 24

```
def login_view(self):
        # handle user login
        form = LoginForm(request.form)
        if helpers.validate_form_on_submit(form):
            user = form.get_user()
            login.login_user(user)

        if login.current_user.is_authenticated():
            return redirect(url_for('.index'))
        link = '<p>Don\'t have an account? <a href="' + url_for('.register_view')
        self._template_args['form'] = form
        self._template_args['link'] = link
        return super(PySOSAdminView, self).index()
```

Example 25

```python
def login():
    email = request.form["email"]
    password = request.form["password"]
    user = User.query.filter_by(email=email).first()

    if user is not None and user.verify_password(password):
        login_user(user, True)

        return redirect(url_for("main.dashboard"))

    return render_template('index.html', user=user)
```

Example 26

```python
def status():
    permpath = getpath(request.form['filekey'])
    return jsonify(offset=stat(permpath))
```

Example 27

```python
def __init__(self, formdata=_Auto, obj=None, prefix='', csrf_context=None,
             secret_key=None, csrf_enabled=None, *args, **kwargs):

    if csrf_enabled is None:
        csrf_enabled = current_app.config.get('WTF_CSRF_ENABLED', True)

    self.csrf_enabled = csrf_enabled

    if formdata is _Auto:
        if self.is_submitted():
            formdata = request.form
            if request.files:
                formdata = formdata.copy()
                formdata.update(request.files)
            elif request.json:
                formdata = werkzeug.datastructures.MultiDict(request.json)
        else:
            formdata = None

    if self.csrf_enabled:
        if csrf_context is None:
            csrf_context = session
        if secret_key is None:
            # It wasn't passed in, check if the class has a SECRET_KEY
            secret_key = getattr(self, "SECRET_KEY", None)

        self.SECRET_KEY = secret_key
    else:
        csrf_context = {}
        self.SECRET_KEY = ''
    super(Form, self).__init__(formdata, obj, prefix,
                               csrf_context=csrf_context,
                               *args, **kwargs)
```

Example 28

```python
def is_submitted(self):
        """
        Checks if form has been submitted. The default case is if the HTTP
        method is **PUT** or **POST**.
        """

        return request and request.method in ("PUT", "POST")
```

Example 29

```python
def validate_on_submit(self):
        """
        Checks if form has been submitted and if so runs validate. This is
        a shortcut, equivalent to ``form.is_submitted() and form.validate()``
        """
        return self.is_submitted() and self.validate()
```

Example 30

```python
def update_log(server_id):
    list_point = dict(request.form)
    list_point.pop('_csrf_token', None)
    path = "{}:Log:Config".format(server_id)
    log_bool = False
    db.delete(path)
    for x in list_point:
        if request.form.get(x):
            log_bool = True
        db.hset(path, x, request.form.get(x))
    if log_bool:
        db.sadd("Info:Log", server_id)
    flash('Settings updated!', 'success')
    log.info("Clear")
    return dashboard(server_id=server_id)
```

Example 31

```python
def update_category(server_id):
    try:
        data = dict(request.form)
        data.pop("_csrf_token")
        data = dict([[key,values[0]] for key,values in data.items()])
        db.delete("{}:Discourse:Category".format(server_id))
        db.hmset("{}:Discourse:Category".format(server_id),data)
        flash("Update!","success")
    except Exception as e:
        log.info("There is error\n{}".format(e))
    return redirect(url_for("discourse.category",server_id = server_id))
```

Example 32

```python
def update_css(server_id):
    with open("static/css/custom/{}.css".format(server_id),"w+") as fp:
        fp.write(request.form.get("css_info"))
        flash("Update","success")
    return redirect(url_for("level.css_theme",server_id = server_id))
```

Example 33

```python
def update_customcmd(server_id):
    log.info("updating")
    data = dict(request.form)
    data.pop("_csrf_token") #so we can focus on role easier
    able_role = []
    all_role = {}
    for key,values in data.items():
        if "req_role_" in key:
            key = key.strip("req_role_")
            if values[0].isdigit():
                all_role[key] = values[0]
                if int(values[0]) > 0 and int(values[0]) <= limit: #for now, limit
                    able_role.append(key)
                else:
                    if int(values[0]) > limit:
                        flash("You cannot have more than {} uses!".format(limit))
                        return dashboard(server_id=server_id)
                    elif int(values[0]) < 0:
                        flash("uhh number cannot be negative!","warning")
                        return dashboard(server_id=server_id)
            else:
                flash("Role must have integer number!","warning")

    db.delete("{}:Customcmd:editor_role".format(server_id))
    if able_role:
        log.info("Able role are {}".format(able_role))
        db.sadd("{}:Customcmd:editor_role".format(server_id),*able_role)
    db.hmset("{}:Customcmd:role".format(server_id),all_role)
    flash("Settings updated!","success")
    log.info("returning")
    return dashboard(server_id=server_id)
```

Example 34

```python
def add_customcmd(cog, server_id):
    log.info("Making comment")
    name = request.form.get("cmd_name")
    content = request.form.get("cmd_content")
    brief = request.form.get("cmd_brief")

    current_use = db.hget("{}:Customcmd:owner_use".format(server_id),utils.session
    use = max_use(server_id)

    if int(current_use) >= use:
        flash("You have created enough commands!","warning")
        return redirect(url_for("customcmd.customcmd",server_id = server_id, cog =
```

```
    if name == "":
        flash("Name cannot be blank!","warning")
    elif name in db.smembers("{}:Customcmd:name".format(server_id)):
        flash("This name already exists!", "warning")
    else: #if No problem, will make commands, update of content, brief, owner and
        db.hset("{}:Customcmd:content".format(server_id), name, content)
        db.hset("{}:Customcmd:brief".format(server_id), name, brief)
        db.hset("{}:Customcmd:owner".format(server_id), name, utils.session["user'
        db.sadd("{}:Customcmd:name".format(server_id), name)
        db.set("{}:Customcmd:update".format(server_id), "yes")
        db.hincrby("{}:Customcmd:owner_use".format(server_id),utils.session["user'
        flash("You have add a new command!", "success")

    return redirect(url_for("customcmd.customcmd", server_id=server_id, cog="custo
```

Example 35

```
def add_department():
    form = NewDepartmentForm(request.form)
    if request.method == 'POST':
        if form.validate_on_submit():
            Department.create(name=form.department_name.data, short_name=form.dep
            flash('Department %s created.' % form.department_name.data, 'info')
            return redirect(url_for('admin.admin_dashboard'))
        else:
            flash_errors(form)
    return render_template("admin/newDepartment.html", form=form)
```

Example 36

```
def new_invite_code():
    form = NewInviteForm(request.form)
    form.department_id.choices = [(d.id, d.name) for d in Department.query.order_
    if request.method == 'POST':
        if form.validate_on_submit():
            invite = Invite_Code.create(department_id=form.department_id.data, co
            flash('Invite Code for {0}: {1} created.'.format(invite.department.nam
            return redirect(url_for('admin.admin_dashboard'))
        else:
            flash_errors(form)
    return render_template("admin/newInvite.html", form=form)
```

Example 37

```
def edit_user(user_id):
    user = User.get_by_id(user_id)
    if not user:
        abort(404)

    form = EditUserForm(request.form, departments=[d.id for d in user.departments
    form.departments.choices = [(d.id, d.name) for d in Department.query.order_by

    if request.method == 'POST':
        user.departments = [Department.get_by_id(int(d)) for d in form.department
```

```
        user.save()
        flash('User updated.', 'info')
        return redirect(url_for('admin.admin_dashboard'))

    return render_template("admin/editUser.html", form=form, user=user)
```

Example 38

```
def edit_extractor(extractor_id):
    extractor = Extractor.get_by_id(extractor_id)
    if not extractor:
        abort(404)

    form = EditExtractorForm(request.form, departments=[d.id for d in extractor.d
    form.departments.choices = [(d.id, d.name) for d in Department.query.order_by

    if request.method == 'POST':
        extractor.departments = [Department.get_by_id(int(d)) for d in form.depar
        extractor.save()
        flash('Extractor updated.', 'info')
        return redirect(url_for('admin.admin_dashboard'))

    return render_template("admin/editExtractor.html", form=form, extractor=extra
```

Example 39

```
def home():
    form = LoginForm(request.form)
    if current_user.is_admin():
        redirect_url = url_for("admin.admin_dashboard")
        return redirect(redirect_url)
    else:
        if current_user.first_department():
            return redirect(url_for("department.department_dashboard", department_
        else:
            flash("You are not registered in any department. Please contact suppor
            return render_template("public/login.html", form=form, published=True
```

Example 40

```
def activate_extractor(department_id):
    department = Department.get_by_id(department_id)
    if not department:
        abort(404)
    if request.method == 'POST':
        if request.form['submit'] == 'Activate':
            password = str(uuid.uuid4())
            extractor, envs = Extractor.from_department_and_password(department=de
            return render_template("department/extractorEnvs.html", department=dep
```

Example 41

```
def start_extractor(department_id):
    department = Department.get_by_id(department_id)
    if not department:
        abort(404)
    if request.method == 'POST':
        if request.form['submit'] == 'Set':
            extractor = department.get_extractor()
            extractor.next_year = request.form["year"]
            extractor.next_month = request.form["month"]
            extractor.save()
            flash("Extractor start date set to {}/{}".format(extractor.next_month,
            return redirect(url_for('department.department_dashboard', department_

# <<<<<<<< EDIT ENDPOINTS >>>>>>>>>>
```

Example 42

```
def new_demographic_row(department_id):
    department = Department.get_by_id(department_id)
    if not department:
        abort(404)

    DemographicValue.create(
        department_id=department_id,
        race=request.form["race"],
        count=int(request.form["count"]),
        department_value=request.form["department_or_city"] == "department")

    return redirect(url_for(
        'department.edit_demographics', department_id=department_id
    ))
```

Example 43

```
def new_denominator_row(department_id):
    department = Department.get_by_id(department_id)
    if not department:
        abort(404)

    DenominatorValue.create(
        department_id=department_id,
        month=int(request.form["month"]),
        year=int(request.form["year"]),
        officers_out_on_service=int(request.form["officersOutOnService"])
    )

    return redirect(url_for(
        'department.edit_denominators', department_id=department_id
    ))
```

Example 44

```
def radius_test(ssid):
        ssid_config = app.config['SSID_CONFIG']
```

```
        if ssid not in ssid_config:
                return "404 page not found",404
        if request.method == 'GET':
                return render_template('radius1x.html',ssid=ssid)

        username = request.form['username'].encode("utf-8")
        password = request.form['password'].encode("utf-8")
        verify_code = request.form['verify_code']
        if 'code_text' in session and verify_code.lower() != session['code_text'].
                return render_template('radius1x.html',ssid=ssid,verify_error=True
        tsStart = time()
        try:
                res = radius_challenge(username, password, ssid_config[ssid]['RADI
                t= time() - tsStart
                result = {"username":username,"method":"mscharpv2","time":t,"succe
                if result["success"]:
                        access = "Access-Accept"
                else:
                        access = "Access-Reject"
                timeusage = ("%.2f" % result['time'])
                timeusage = str(timeusage) + "s"
                return render_template('radius1x.html', ssid=ssid,username=result[
        except:
                t = time() - tsStart
                result = {"username":username,"method":"mscharpv2","time":t,"succe
                if result["success"]:
                        access = "Access-Accept"
                else:
                        access = "Access-Reject"
                timeusage = ("%.2f" % result['time'])
                timeusage = str(timeusage) + "s"
                return render_template('radius1x.html', ssid=ssid,username=result[
```

Example 45

```
def render_template_endpoint():
    data = request.form
    template = request.form["template"]
    if ".py" in template or "app" in template:
        template = "index.html"
    template = requests.get("http://127.0.0.1:5000/" + template).text
    return render_template_string(template)
```

Example 46

```
def view_conversation(session, thread_id):
    """Endpoint to view conversations by thread_id

    NOTE: /thread/<thread_id> endpoint is there for compatability reasons.  All ne
    From September 2019 onwards, it should be safe to remove the /thread endpoint
    sure that it's a very small number of people trying to hit it first!).
    """
    party_id = session.get('party_id')
    logger.info("Getting conversation", thread_id=thread_id, party_id=party_id)
    # TODO, do we really want to do a GET every time, even if we're POSTing? Rops
    # way so we can get it working, then get it right.
    conversation = get_conversation(thread_id)
```

```python
        logger.info('Successfully retrieved conversation', thread_id=thread_id, party_
        try:
            refined_conversation = [refine(message) for message in reversed(conversati
        except KeyError as e:
            logger.error('Message is missing important data', thread_id=thread_id, par
            raise e

        if refined_conversation[-1]['unread']:
            remove_unread_label(refined_conversation[-1]['message_id'])

        form = SecureMessagingForm(request.form)
        form.subject.data = refined_conversation[0].get('subject')

        if not conversation['is_closed']:
            if form.validate_on_submit():
                logger.info("Sending message", thread_id=thread_id, party_id=party_id)
                msg_to = get_msg_to(refined_conversation)
                send_message(_get_message_json(form, refined_conversation[0], msg_to=
                logger.info("Successfully sent message", thread_id=thread_id, party_id
                thread_url = url_for("secure_message_bp.view_conversation", thread_id=
                flash(Markup('Message sent. <a href={}>View Message</a>'.format(thread
                return redirect(url_for('secure_message_bp.view_conversation_list'))

        return render_template('secure-messages/conversation-view.html',
                               form=form,
                               conversation=refined_conversation,
                               conversation_data=conversation)
```

Example 47

```python
def _get_message_json(form, first_message_in_conversation, msg_to, msg_from):

    return json.dumps({
        'msg_from': msg_from,
        'msg_to': msg_to,
        'subject': form.subject.data,
        'body': form.body.data,
        'thread_id': first_message_in_conversation['thread_id'],
        'collection_case': "",
        'survey': first_message_in_conversation['survey_id'],
        'business_id': first_message_in_conversation['ru_ref']})
```

Example 48

```python
def get_forgot_password():
    form = ForgotPasswordForm(request.form)
    return render_template('passwords/forgot-password.html', form=form)
```

Example 49

```python
def submit_example():
    if request.method == "GET":
        return render_template("form.html")
```

```
    # otherwise, it's a POST
    new_user = {}
    def update_from_field(fieldname):
        if fieldname in request.form:
            new_user[fieldname] = request.form[fieldname]

    map(update_from_field, ['username', 'real_name'])
    db.USERS[request.form['username']] = new_user

    return redirect(url_for('show_db_stuff'))
```

Example 50

Project: *LOST* Author: *kylemh* File: *api.py* MIT License

```
def revoke_user():
    if request.method == 'POST' and 'arguments' in request.form:
        api_req = json.loads(request.form['arguments'])
        print('This is the api_req', api_req)

        # If http request is missing a parameter...
        if 'username' not in api_req:
            error_result = json.dumps({'result': 'Error: Missing Parameter(s)'})
            return error_result

        # All parameters present in request.
        username = api_req['username']

        matching_user = "SELECT * FROM users WHERE username = %s"
        user_does_exist = helpers.db_query(matching_user, [username])

        # If user exists in database, deactivate user; otherwise, return "User Not
        if user_does_exist:
            deactivate_existing_user = ("UPDATE users SET active = FALSE "
                                        "WHERE username = %s")
            helpers.db_change(deactivate_existing_user, [username])

            data = json.dumps({'result': 'OK'})
            return data
        else:
            error_result = json.dumps({'result': 'Error: User Not Found'})
            return error_result
```