File: D:\project\app.py

```
from flask import Flask, render_template
from extensions import mysql
from routes.auth_routes import auth_bp

app = Flask(__name__)

# Secret Key Configuration
app.secret_key = '123456789'

# MySQL Configuration
app.config['MYSQL_HOST'] = '127.0.0.1'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = 'admin@555'
app.config['MYSQL_DB'] = 'user_database'

# Initialize MySQL with the app
mysql.init_app(app)

# Register blueprints
app.register_blueprint(auth_bp)

@app.route('/')
def home():
    return render_template('index.html')

if __name__ == '__main__':
    app.run(debug=True)
```

File: D:\project\auth.py

```
from flask import Blueprint, render_template, request, redirect, session

from flask_mysqldb import MySQL

import MySQLdb.cursors

import re



auth = Blueprint('auth', __name__)



# Ensure MySQL instance is accessible globally

mysql = None



def init_mysql(app_mysql):

    global mysql

    mysql = app_mysql



@auth.route('/signup', methods=['GET', 'POST'])

def signup():

    if request.method == 'POST' and 'username' in request.form and 'password' in request.form and 'role' in request.form:

        username = request.form['username']

        password = request.form['password']

        role = request.form['role']



        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

        cursor.execute('SELECT * FROM users WHERE username = %s', (username,))
        account = cursor.fetchone()

        if account:
            msg = 'Account already exists!'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'Username must contain only characters and numbers!'
        elif not username or not password or not role:
            msg = 'Please fill out the form!'
        else:
            cursor.execute('INSERT INTO users (username, password, role) VALUES (%s, %s,
%s)', (username, password, role))
            mysql.connection.commit()
            msg = 'You have successfully registered!'
            return redirect('/auth/signin')

        return render_template('auth/signup.html', msg=msg)

    return render_template('auth/signup.html')

@auth.route('/signin', methods=['GET', 'POST'])
def signin():
    if request.method == 'POST' and 'username' in request.form and 'password' in
request.form:
        username = request.form['username']
        password = request.form['password']

        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM users WHERE username = %s AND password = %s',
(username, password))
        account = cursor.fetchone()

        if account:
            session['loggedin'] = True
            session['id'] = account['id']
            session['username'] = account['username']
            session['role'] = account['role']

            if account['role'] == 'Student':
                return redirect('/dashboard/student')
            elif account['role'] == 'Teacher':
                return redirect('/dashboard/teacher')
        else:
            msg = 'Incorrect username/password!'
            return render_template('auth/signin.html', msg=msg)

    return render_template('auth/signin.html')


@auth.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    session.pop('role', None)
    return redirect('/auth/signin')
```

File: D:\project\extensions.py

```
from flask_mysqldb import MySQL



# Initialize MySQL object

mysql = MySQL()
```

File: D:\project\.git\COMMIT_EDITMSG

```
Initial commit
```

File: D:\project\.git\config

```
[core]

 repositoryformatversion = 0

 filemode = false

 bare = false

 logallrefupdates = true

 symlinks = false

 ignorecase = true

[remote "origin"]

 url = https://github.com/Danish1092/Student_Performance_Analyzer.git

 fetch = +refs/heads/*:refs/remotes/origin/*

[branch "main"]

 remote = origin

 merge = refs/heads/main
```

File: D:\project\.git\description

```
Unnamed repository; edit this file 'description' to name the repository.
```

File: D:\project\.git\HEAD

```
ref: refs/heads/main
```

File: D:\project\.git\index

Error reading file D:\project\.git\index: 'utf-8' codec can't decode byte 0xde in position 14: invalid continuation byte

File: D:\project\.git\hooks\applypatch-msg.sample

```
#!/bin/sh

#

# An example hook script to check the commit log message taken by

# applypatch from an e-mail message.

#

# The hook should exit with non-zero status after issuing an

# appropriate message if it wants to stop the commit.  The hook is

# allowed to edit the commit message file.

#

# To enable this hook, rename this file to "applypatch-msg".


. git-sh-setup

commitmsg="$(git rev-parse --git-path hooks/commit-msg)"

test -x "$commitmsg" && exec "$commitmsg" ${1+"$@"}

:
```

File: D:\project\.git\hooks\commit-msg.sample

```
#!/bin/sh

#

# An example hook script to check the commit log message.
# Called by "git commit" with one argument, the name of the file
# that has the commit message.  The hook should exit with non-zero
# status after issuing an appropriate message if it wants to stop the
# commit.  The hook is allowed to edit the commit message file.
#
# To enable this hook, rename this file to "commit-msg".

# Uncomment the below to add a Signed-off-by line to the message.
# Doing this in a hook is a bad idea in general, but the prepare-commit-msg
# hook is more suited to it.
#
# SOB=$(git var GIT_AUTHOR_IDENT | sed -n 's/^\(.*>\).*$/Signed-off-by: \1/p')
# grep -qs "^$SOB" "$1" || echo "$SOB" >> "$1"

# This example catches duplicate Signed-off-by lines.

test "" = "$(grep '^Signed-off-by: ' "$1" |
  sort | uniq -c | sed -e '/^[  ]*1[  ]/d')" || {
 echo >&2 Duplicate Signed-off-by lines.
 exit 1
}
```

File: D:\project\.git\hooks\fsmonitor-watchman.sample

```
#!/usr/bin/perl

use strict;
use warnings;
use IPC::Open2;

# An example hook script to integrate Watchman
# (https://facebook.github.io/watchman/) with git to speed up detecting
# new and modified files.
#
# The hook is passed a version (currently 2) and last update token
# formatted as a string and outputs to stdout a new update token and
# all files that have been modified since the update token. Paths must
# be relative to the root of the working tree and separated by a single NUL.
#
# To enable this hook, rename this file to "query-watchman" and set
# 'git config core.fsmonitor .git/hooks/query-watchman'
#
my ($version, $last_update_token) = @ARGV;

# Uncomment for debugging
# print STDERR "$0 $version $last_update_token\n";

# Check the hook interface version
if ($version ne 2) {
 die "Unsupported query-fsmonitor hook version '$version'.\n" .
     "Falling back to scanning...\n";
}

my $git_work_tree = get_working_dir();
my $retry = 1;

my $json_pkg;
eval {
 require JSON::XS;
 $json_pkg = "JSON::XS";
 1;
} or do {
 require JSON::PP;
 $json_pkg = "JSON::PP";
};

launch_watchman();

sub launch_watchman {
 my $o = watchman_query();
 if (is_work_tree_watched($o)) {
  output_result($o->{clock}, @{$o->{files}});
 }
}

sub output_result {
 my ($clockid, @files) = @_;

 # Uncomment for debugging watchman output
 # open (my $fh, ">", ".git/watchman-output.out");
 # binmode $fh, ":utf8";
 # print $fh "$clockid\n@files\n";
 # close $fh;

 binmode STDOUT, ":utf8";
 print $clockid;
 print "\0";
 local $, = "\0";
 print @files;
}

sub watchman_clock {
 my $response = qx/watchman clock "$git_work_tree"/;
 die "Failed to get clock id on '$git_work_tree'.\n" .
  "Falling back to scanning...\n" if $? != 0;

 return $json_pkg->new->utf8->decode($response);
}

sub watchman_query {
 my $pid = open2(\*CHLD_OUT, \*CHLD_IN, 'watchman -j --no-pretty')
 or die "open2() failed: $!\n" .
 "Falling back to scanning...\n";

 # In the query expression below we're asking for names of files that
 # changed since $last_update_token but not from the .git folder.
 #
 # To accomplish this, we're using the "since" generator to use the
 # recency index to select candidate nodes and "fields" to limit the
 # output to file names only. Then we're using the "expression" term to
 # further constrain the results.
 my $last_update_line = "";
 if (substr($last_update_token, 0, 1) eq "c") {
  $last_update_token = "\"$last_update_token\"";
  $last_update_line = qq[\n"since": $last_update_token,];
 }
 my $query = <<" END";
  ["query", "$git_work_tree", {$last_update_line
   "fields": ["name"],
   "expression": ["not", ["dirname", ".git"]]
  }]
 END

 # Uncomment for debugging the watchman query
 # open (my $fh, ">", ".git/watchman-query.json");
 # print $fh $query;
 # close $fh;

 print CHLD_IN $query;
 close CHLD_IN;
 my $response = do {local $/; <CHLD_OUT>};

 # Uncomment for debugging the watch response
 # open ($fh, ">", ".git/watchman-response.json");
 # print $fh $response;
```

```
 # close $fh;

 die "Watchman: command returned no output.\n" .
 "Falling back to scanning...\n" if $response eq "";
 die "Watchman: command returned invalid output: $response\n" .
 "Falling back to scanning...\n" unless $response =~ /^\{/;

 return $json_pkg->new->utf8->decode($response);
}

sub is_work_tree_watched {
 my ($output) = @_;
 my $error = $output->{error};
 if ($retry > 0 and $error and $error =~ m/unable to resolve root .* directory (.*) is
not watched/) {
  $retry--;
  my $response = qx/watchman watch "$git_work_tree"/;
  die "Failed to make watchman watch '$git_work_tree'.\n" .
      "Falling back to scanning...\n" if $? != 0;
  $output = $json_pkg->new->utf8->decode($response);
  $error = $output->{error};
  die "Watchman: $error.\n" .
  "Falling back to scanning...\n" if $error;

  # Uncomment for debugging watchman output
  # open (my $fh, ">", ".git/watchman-output.out");
  # close $fh;
  # Watchman will always return all files on the first query so

  # return the fast "everything is dirty" flag to git and do the

  # Watchman query just to get it over with now so we won't pay

  # the cost in git to look up each individual file.

  my $o = watchman_clock();

  $error = $output->{error};



  die "Watchman: $error.\n" .

  "Falling back to scanning...\n" if $error;



  output_result($o->{clock}, ("/"));

  $last_update_token = $o->{clock};



  eval { launch_watchman() };

  return 0;

 }



 die "Watchman: $error.\n" .

 "Falling back to scanning...\n" if $error;



 return 1;

}



sub get_working_dir {

 my $working_dir;

 if ($^O =~ 'msys' || $^O =~ 'cygwin') {
  $working_dir = Win32::GetCwd();

  $working_dir =~ tr/\\/\//;

 } else {

  require Cwd;

  $working_dir = Cwd::cwd();

 }


 return $working_dir;

}
```

File: D:\project\.git\hooks\post-update.sample

```
#!/bin/sh

#

# An example hook script to prepare a packed repository for use over

# dumb transports.

#

# To enable this hook, rename this file to "post-update".


exec git update-server-info
```

File: D:\project\.git\hooks\pre-applypatch.sample

```
#!/bin/sh

#

# An example hook script to verify what is about to be committed

# by applypatch from an e-mail message.

#

# The hook should exit with non-zero status after issuing an
# appropriate message if it wants to stop the commit.

#

# To enable this hook, rename this file to "pre-applypatch".



. git-sh-setup

precommit="$(git rev-parse --git-path hooks/pre-commit)"

test -x "$precommit" && exec "$precommit" ${1+"$@"}

:
```

File: D:\project\.git\hooks\pre-commit.sample

```
#!/bin/sh

#

# An example hook script to verify what is about to be committed.

# Called by "git commit" with no arguments.  The hook should

# exit with non-zero status after issuing an appropriate message if

# it wants to stop the commit.

#

# To enable this hook, rename this file to "pre-commit".



if git rev-parse --verify HEAD >/dev/null 2>&1

then

 against=HEAD

else

 # Initial commit: diff against an empty tree object

 against=$(git hash-object -t tree /dev/null)

fi
```

```
# If you want to allow non-ASCII filenames set this variable to true.

allownonascii=$(git config --type=bool hooks.allownonascii)



# Redirect output to stderr.

exec 1>&2



# Cross platform projects tend to avoid non-ASCII filenames; prevent

# them from being added to the repository. We exploit the fact that the

# printable range starts at the space character and ends with tilde.

if [ "$allownonascii" != "true" ] &&

 # Note that the use of brackets around a tr range is ok here, (it's

 # even required, for portability to Solaris 10's /usr/bin/tr), since

 # the square bracket bytes happen to fall in the designated range.

 test $(git diff-index --cached --name-only --diff-filter=A -z $against |

   LC_ALL=C tr -d '[ -~]\0' | wc -c) != 0

then

 cat <<\EOF

Error: Attempt to add a non-ASCII file name.



This can cause problems if you want to work with people on other platforms.



To be portable it is advisable to rename the file.



If you know what you are doing you can disable this check using:



  git config hooks.allownonascii true

EOF
```

```
 exit 1
fi

# If there are whitespace errors, print the offending file names and fail.
exec git diff-index --check --cached $against --
```

File: D:\project\.git\hooks\pre-merge-commit.sample

```
#!/bin/sh
#
# An example hook script to verify what is about to be committed.
# Called by "git merge" with no arguments.  The hook should
# exit with non-zero status after issuing an appropriate message to
# stderr if it wants to stop the merge commit.
#
# To enable this hook, rename this file to "pre-merge-commit".

. git-sh-setup
test -x "$GIT_DIR/hooks/pre-commit" &&
        exec "$GIT_DIR/hooks/pre-commit"
:
```

File: D:\project\.git\hooks\pre-push.sample

```
#!/bin/sh

# An example hook script to verify what is about to be pushed.  Called by "git
# push" after it has checked the remote status, but before anything has been
# pushed.  If this script exits with a non-zero status nothing will be pushed.
```

```
#
# This hook is called with the following parameters:
#
# $1 -- Name of the remote to which the push is being done
# $2 -- URL to which the push is being done
#
# If pushing without using a named remote those arguments will be equal.
#
# Information about the commits which are being pushed is supplied as lines to
# the standard input in the form:
#
#   <local ref> <local oid> <remote ref> <remote oid>
#
# This sample shows how to prevent push of commits where the log message starts
# with "WIP" (work in progress).

remote="$1"
url="$2"

zero=$(git hash-object --stdin </dev/null | tr '[0-9a-f]' '0')

while read local_ref local_oid remote_ref remote_oid
do
 if test "$local_oid" = "$zero"
 then
  # Handle delete
  :
```

```
 else
  if test "$remote_oid" = "$zero"
  then
   # New branch, examine all commits
   range="$local_oid"
  else
   # Update to existing branch, examine new commits
   range="$remote_oid..$local_oid"
  fi

  # Check for WIP commit
  commit=$(git rev-list -n 1 --grep '^WIP' "$range")
  if test -n "$commit"
  then
   echo >&2 "Found WIP commit in $local_ref, not pushing"
   exit 1
  fi
 fi
done

exit 0
```

File: D:\project\.git\hooks\pre-rebase.sample

```
#!/bin/sh
#
# Copyright (c) 2006, 2008 Junio C Hamano
#
```

```
# The "pre-rebase" hook is run just before "git rebase" starts doing
# its job, and can prevent the command from running by exiting with
# non-zero status.
#
# The hook is called with the following parameters:
#
# $1 -- the upstream the series was forked from.
# $2 -- the branch being rebased (or empty when rebasing the current branch).
#
# This sample shows how to prevent topic branches that are already
# merged to 'next' branch from getting rebased, because allowing it
# would result in rebasing already published history.

publish=next
basebranch="$1"
if test "$#" = 2
then
 topic="refs/heads/$2"
else
 topic=`git symbolic-ref HEAD` ||
 exit 0 ;# we do not interrupt rebasing detached HEAD
fi

case "$topic" in
refs/heads/??/*)
 ;;
*)
```

```
 exit 0 ;# we do not interrupt others.

 ;;

esac



# Now we are dealing with a topic branch being rebased

# on top of master.  Is it OK to rebase it?



# Does the topic really exist?

git show-ref -q "$topic" || {

 echo >&2 "No such branch $topic"

 exit 1

}



# Is topic fully merged to master?

not_in_master=`git rev-list --pretty=oneline ^master "$topic"`

if test -z "$not_in_master"

then

 echo >&2 "$topic is fully merged to master; better remove it."

 exit 1 ;# we could allow it, but there is no point.

fi



# Is topic ever merged to next?  If so you should not be rebasing it.

only_next_1=`git rev-list ^master "^$topic" ${publish} | sort`

only_next_2=`git rev-list ^master           ${publish} | sort`

if test "$only_next_1" = "$only_next_2"

then

 not_in_topic=`git rev-list "^$topic" master`
 if test -z "$not_in_topic"
 then
  echo >&2 "$topic is already up to date with master"
  exit 1 ;# we could allow it, but there is no point.
 else
  exit 0
 fi
else
 not_in_next=`git rev-list --pretty=oneline ^${publish} "$topic"`
 /usr/bin/perl -e '
  my $topic = $ARGV[0];
  my $msg = "* $topic has commits already merged to public branch:\n";
  my (%not_in_next) = map {
   /^([0-9a-f]+) /;
   ($1 => 1);
  } split(/\n/, $ARGV[1]);
  for my $elem (map {
    /^([0-9a-f]+) (.*)$/;
    [$1 => $2];
   } split(/\n/, $ARGV[2])) {
   if (!exists $not_in_next{$elem->[0]}) {
    if ($msg) {
     print STDERR $msg;
     undef $msg;
    }
    print STDERR " $elem->[1]\n";
   }
  }
 ' "$topic" "$not_in_next" "$not_in_master"
 exit 1
fi

<<\DOC_END

This sample hook safeguards topic branches that have been
published from being rewound.

The workflow assumed here is:

 * Once a topic branch forks from "master", "master" is never
   merged into it again (either directly or indirectly).

 * Once a topic branch is fully cooked and merged into "master",
   it is deleted.  If you need to build on top of it to correct
   earlier mistakes, a new topic branch is created by forking at
   the tip of the "master".  This is not strictly necessary, but
   it makes it easier to keep your history simple.

 * Whenever you need to test or publish your changes to topic
   branches, merge them into "next" branch.

The script, being an example, hardcodes the publish branch name
to be "next", but it is trivial to make it configurable via
$GIT_DIR/config mechanism.
```

With this workflow, you would want to know:

(1) ... if a topic branch has ever been merged to "next". Young topic branches can have stupid mistakes you would rather clean up before publishing, and things that have not been merged into other branches can be easily rebased without affecting other people. But once it is published, you would not want to rewind it.

(2) ... if a topic branch has been fully merged to "master". Then you can delete it. More importantly, you should not build on top of it -- other people may already want to change things related to the topic as patches against your "master", so if you need further changes, it is better to fork the topic (perhaps with the same name) afresh from the tip of "master".

Let's look at this example:

```
     o---o---o---o---o---o---o---o---o---o "next"
    /       /           /           /
   /   a---a---b A     /           /
  /   /               /           /
        /   /   c---c---c---c B         /
       /   /   /             \         /
      /   /   /   b---b C     \       /
     /   /   /   /             \     /
    ---o---o---o---o---o---o---o---o---o---o---o "master"
```

A, B and C are topic branches.

* A has one fix since it was merged up to "next".

* B has finished. It has been fully merged up to "master" and "next",
  and is ready to be deleted.

* C has not merged to "next" at all.

We would want to allow C to be rebased, refuse A, and encourage
B to be deleted.

To compute (1):

```
 git rev-list ^master ^topic next
 git rev-list ^master        next
```

 if these match, topic has not merged in next at all.

To compute (2):

```
 git rev-list master..topic
 if this is empty, it is fully merged to "master".

DOC_END
```

File: D:\project\.git\hooks\pre-receive.sample

```
#!/bin/sh
#
# An example hook script to make use of push options.
# The example simply echoes all push options that start with 'echoback='
# and rejects all pushes when the "reject" push option is used.
#
# To enable this hook, rename this file to "pre-receive".

if test -n "$GIT_PUSH_OPTION_COUNT"
then
 i=0
 while test "$i" -lt "$GIT_PUSH_OPTION_COUNT"
 do
  eval "value=\$GIT_PUSH_OPTION_$i"
  case "$value" in
  echoback=*)
   echo "echo from the pre-receive-hook: ${value#*=}" >&2
   ;;
  reject)
   exit 1
  esac
  i=$((i + 1))
 done
fi
```

File: D:\project\.git\hooks\prepare-commit-msg.sample

```
#!/bin/sh
#
# An example hook script to prepare the commit log message.
# Called by "git commit" with the name of the file that has the
# commit message, followed by the description of the commit
# message's source.  The hook's purpose is to edit the commit
# message file.  If the hook fails with a non-zero status,
# the commit is aborted.
#
# To enable this hook, rename this file to "prepare-commit-msg".

# This hook includes three examples. The first one removes the
# "# Please enter the commit message..." help message.
#
# The second includes the output of "git diff --name-status -r"
# into the message, just before the "git status" output.  It is
# commented because it doesn't cope with --amend or with squashed
# commits.
#
# The third example adds a Signed-off-by line to the message, that can
# still be edited.  This is rarely a good idea.

COMMIT_MSG_FILE=$1
```

```
COMMIT_SOURCE=$2

SHA1=$3


/usr/bin/perl -i.bak -ne 'print unless(m/^. Please enter the commit message/..m/^#$/)'

"$COMMIT_MSG_FILE"


# case "$COMMIT_SOURCE,$SHA1" in

#  ,|template,)

#    /usr/bin/perl -i.bak -pe '

#       print "\n" . `git diff --cached --name-status -r`

#   if /^#/ && $first++ == 0' "$COMMIT_MSG_FILE" ;;

#  *) ;;

# esac


# SOB=$(git var GIT_COMMITTER_IDENT | sed -n 's/^\(.*>\).*$/Signed-off-by: \1/p')

# git interpret-trailers --in-place --trailer "$SOB" "$COMMIT_MSG_FILE"

# if test -z "$COMMIT_SOURCE"

# then

#   /usr/bin/perl -i.bak -pe 'print "\n" if !$first_line++' "$COMMIT_MSG_FILE"

# fi
```

File: D:\project\.git\hooks\push-to-checkout.sample

```
#!/bin/sh


# An example hook script to update a checked-out tree on a git push.

#

# This hook is invoked by git-receive-pack(1) when it reacts to git
# push and updates reference(s) in its repository, and when the push
# tries to update the branch that is currently checked out and the
# receive.denyCurrentBranch configuration variable is set to
# updateInstead.
#
# By default, such a push is refused if the working tree and the index
# of the remote repository has any difference from the currently
# checked out commit; when both the working tree and the index match
# the current commit, they are updated to match the newly pushed tip
# of the branch. This hook is to be used to override the default
# behaviour; however the code below reimplements the default behaviour
# as a starting point for convenient modification.
#
# The hook receives the commit with which the tip of the current
# branch is going to be updated:
commit=$1

# It can exit with a non-zero status to refuse the push (when it does
# so, it must not modify the index or the working tree).
die () {
 echo >&2 "$*"
 exit 1
}

# Or it can make any necessary changes to the working tree and to the
# index to bring them to the desired state when the tip of the current
# branch is updated to the new commit, and exit with a zero status.
```

```
#
# For example, the hook can simply run git read-tree -u -m HEAD "$1"
# in order to emulate git fetch that is run in the reverse direction
# with git push, as the two-tree form of git read-tree -u -m is
# essentially the same as git switch or git checkout that switches
# branches while keeping the local changes in the working tree that do
# not interfere with the difference between the branches.

# The below is a more-or-less exact translation to shell of the C code
# for the default behaviour for git's push-to-checkout hook defined in
# the push_to_deploy() function in builtin/receive-pack.c.
#
# Note that the hook will be executed from the repository directory,
# not from the working tree, so if you want to perform operations on
# the working tree, you will have to adapt your code accordingly, e.g.
# by adding "cd .." or using relative paths.

if ! git update-index -q --ignore-submodules --refresh
then
 die "Up-to-date check failed"
fi

if ! git diff-files --quiet --ignore-submodules --
then
 die "Working directory has unstaged changes"
fi
```

```
# This is a rough translation of:
#
#   head_has_history() ? "HEAD" : EMPTY_TREE_SHA1_HEX
if git cat-file -e HEAD 2>/dev/null
then
 head=HEAD
else
 head=$(git hash-object -t tree --stdin </dev/null)
fi

if ! git diff-index --quiet --cached --ignore-submodules $head --
then
 die "Working directory has staged changes"
fi

if ! git read-tree -u -m "$commit"
then
 die "Could not update working tree to new HEAD"
fi
```

File: D:\project\.git\hooks\sendemail-validate.sample

```
#!/bin/sh

# An example hook script to validate a patch (and/or patch series) before
# sending it via email.
#
# The hook should exit with non-zero status after issuing an appropriate
# message if it wants to prevent the email(s) from being sent.
#
# To enable this hook, rename this file to "sendemail-validate".
#
# By default, it will only check that the patch(es) can be applied on top of
# the default upstream branch without conflicts in a secondary worktree. After
# validation (successful or not) of the last patch of a series, the worktree
# will be deleted.
#
# The following config variables can be set to change the default remote and
# remote ref that are used to apply the patches against:
#
#   sendemail.validateRemote (default: origin)
#   sendemail.validateRemoteRef (default: HEAD)
#
# Replace the TODO placeholders with appropriate checks according to your
# needs.

validate_cover_letter () {
 file="$1"
 # TODO: Replace with appropriate checks (e.g. spell checking).
 true
}

validate_patch () {
 file="$1"
 # Ensure that the patch applies without conflicts.
 git am -3 "$file" || return
 # TODO: Replace with appropriate checks for this patch
 # (e.g. checkpatch.pl).
 true
}

validate_series () {
 # TODO: Replace with appropriate checks for the whole series
 # (e.g. quick build, coding style checks, etc.).
 true
}

# main -------------------------------------------------------------------------

if test "$GIT_SENDEMAIL_FILE_COUNTER" = 1
then
 remote=$(git config --default origin --get sendemail.validateRemote) &&
 ref=$(git config --default HEAD --get sendemail.validateRemoteRef) &&
 worktree=$(mktemp --tmpdir -d sendemail-validate.XXXXXXX) &&
 git worktree add -fd --checkout "$worktree" "refs/remotes/$remote/$ref" &&
 git config --replace-all sendemail.validateWorktree "$worktree"
else
 worktree=$(git config --get sendemail.validateWorktree)
fi || {
 echo "sendemail-validate: error: failed to prepare worktree" >&2
 exit 1
}
```

```
unset GIT_DIR GIT_WORK_TREE

cd "$worktree" &&



if grep -q "^diff --git " "$1"

then

 validate_patch "$1"

else

 validate_cover_letter "$1"

fi &&



if test "$GIT_SENDEMAIL_FILE_COUNTER" = "$GIT_SENDEMAIL_FILE_TOTAL"

then

 git config --unset-all sendemail.validateWorktree &&

 trap 'git worktree remove -ff "$worktree"' EXIT &&

 validate_series

fi
```

File: D:\project\.git\hooks\update.sample

```
#!/bin/sh

#

# An example hook script to block unannotated tags from entering.

# Called by "git receive-pack" with arguments: refname sha1-old sha1-new

#

# To enable this hook, rename this file to "update".

#

# Config
# ------
# hooks.allowunannotated
#   This boolean sets whether unannotated tags will be allowed into the
#   repository.  By default they won't be.
# hooks.allowdeletetag
#   This boolean sets whether deleting tags will be allowed in the
#   repository.  By default they won't be.
# hooks.allowmodifytag
#   This boolean sets whether a tag may be modified after creation. By default
#   it won't be.
# hooks.allowdeletebranch
#   This boolean sets whether deleting branches will be allowed in the
#   repository.  By default they won't be.
# hooks.denycreatebranch
#   This boolean sets whether remotely creating branches will be denied
#   in the repository.  By default this is allowed.
#

# --- Command line
refname="$1"
oldrev="$2"
newrev="$3"

# --- Safety check
if [ -z "$GIT_DIR" ]; then
 echo "Don't run this script from the command line." >&2
 echo " (if you want, you could supply GIT_DIR then run" >&2
 echo "  $0 <ref> <oldrev> <newrev>)" >&2

 exit 1

fi



if [ -z "$refname" -o -z "$oldrev" -o -z "$newrev" ]; then

 echo "usage: $0 <ref> <oldrev> <newrev>" >&2

 exit 1

fi



# --- Config

allowunannotated=$(git config --type=bool hooks.allowunannotated)

allowdeletebranch=$(git config --type=bool hooks.allowdeletebranch)

denycreatebranch=$(git config --type=bool hooks.denycreatebranch)

allowdeletetag=$(git config --type=bool hooks.allowdeletetag)

allowmodifytag=$(git config --type=bool hooks.allowmodifytag)



# check for no description

projectdesc=$(sed -e '1q' "$GIT_DIR/description")

case "$projectdesc" in

"Unnamed repository"* | "")

 echo "*** Project description file hasn't been set" >&2

 exit 1

 ;;

esac



# --- Check types

# if $newrev is 0000...0000, it's a commit to delete a ref.
```

```
zero=$(git hash-object --stdin </dev/null | tr '[0-9a-f]' '0')
if [ "$newrev" = "$zero" ]; then
 newrev_type=delete
else
 newrev_type=$(git cat-file -t $newrev)
fi

case "$refname","$newrev_type" in
 refs/tags/*,commit)
  # un-annotated tag
  short_refname=${refname##refs/tags/}
  if [ "$allowunannotated" != "true" ]; then
   echo "*** The un-annotated tag, $short_refname, is not allowed in this repository"
>&2
   echo "*** Use 'git tag [ -a | -s ]' for tags you want to propagate." >&2
   exit 1
  fi
  ;;
 refs/tags/*,delete)
  # delete tag
  if [ "$allowdeletetag" != "true" ]; then
   echo "*** Deleting a tag is not allowed in this repository" >&2
   exit 1
  fi
  ;;
 refs/tags/*,tag)
  # annotated tag
  if [ "$allowmodifytag" != "true" ] && git rev-parse $refname > /dev/null 2>&1
  then
   echo "*** Tag '$refname' already exists." >&2
   echo "*** Modifying a tag is not allowed in this repository." >&2
   exit 1
  fi
  ;;
 refs/heads/*,commit)
  # branch
  if [ "$oldrev" = "$zero" -a "$denycreatebranch" = "true" ]; then
   echo "*** Creating a branch is not allowed in this repository" >&2
   exit 1
  fi
  ;;
 refs/heads/*,delete)
  # delete branch
  if [ "$allowdeletebranch" != "true" ]; then
   echo "*** Deleting a branch is not allowed in this repository" >&2
   exit 1
  fi
  ;;
 refs/remotes/*,commit)
  # tracking branch
  ;;
 refs/remotes/*,delete)
  # delete tracking branch
  if [ "$allowdeletebranch" != "true" ]; then
```

```
   echo "*** Deleting a tracking branch is not allowed in this repository" >&2

   exit 1

  fi

  ;;

 *)

  # Anything else (is there anything else?)

  echo "*** Update hook: unknown type of update to ref $refname of type $newrev_type"
>&2

  exit 1

  ;;

esac


# --- Finished

exit 0
```

File: D:\project\.git\info\exclude

```
# git ls-files --others --exclude-from=.git/info/exclude

# Lines that start with '#' are comments.

# For a project mostly in C, the following would be a good set of

# exclude patterns (uncomment them if you want to use them):

# *.[oa]

# *~
```

File: D:\project\.git\logs\HEAD

```
0000000000000000000000000000000000000000         f249163981c2ff17bc5829fcf1e1890ec9a5d472

Danish1092 <syeddanish1092@gmail.com> 1733225409 +0530 commit (initial): Initial commit

f249163981c2ff17bc5829fcf1e1890ec9a5d472         0000000000000000000000000000000000000000
```

```
Danish1092 <syeddanish1092@gmail.com> 1733225427 +0530 Branch: renamed refs/heads/master
to refs/heads/main
0000000000000000000000000000000000000000        f249163981c2ff17bc5829fcf1e1890ec9a5d472
Danish1092 <syeddanish1092@gmail.com> 1733225427 +0530 Branch: renamed refs/heads/master
to refs/heads/main
f249163981c2ff17bc5829fcf1e1890ec9a5d472        271853e2aec26945f68da5303ffd79472ee8d4c7
Danish1092 <syeddanish1092@gmail.com> 1733297384 +0530 commit: Initial commit
```

File: D:\project\.git\logs\refs\heads\main

```
0000000000000000000000000000000000000000        f249163981c2ff17bc5829fcf1e1890ec9a5d472
Danish1092 <syeddanish1092@gmail.com> 1733225409 +0530 commit (initial): Initial commit
f249163981c2ff17bc5829fcf1e1890ec9a5d472        f249163981c2ff17bc5829fcf1e1890ec9a5d472
Danish1092 <syeddanish1092@gmail.com> 1733225427 +0530 Branch: renamed refs/heads/master
to refs/heads/main
f249163981c2ff17bc5829fcf1e1890ec9a5d472        271853e2aec26945f68da5303ffd79472ee8d4c7
Danish1092 <syeddanish1092@gmail.com> 1733297384 +0530 commit: Initial commit
```

File: D:\project\.git\logs\refs\remotes\origin\main

```
0000000000000000000000000000000000000000        271853e2aec26945f68da5303ffd79472ee8d4c7
Danish1092 <syeddanish1092@gmail.com> 1733297398 +0530 update by push
```

File: D:\project\.git\objects\03\647e49b0f61540b12c9b617c399c12b92480a0

Error reading file D:\project\.git\objects\03\647e49b0f61540b12c9b617c399c12b92480a0: 'utf-8' codec can't decode byte 0x85 in position 15: invalid start byte

File: D:\project\.git\objects\06\a432ceb88a995d19cd7643876a0fcf9f5a1157

Error reading file D:\project\.git\objects\06\a432ceb88a995d19cd7643876a0fcf9f5a1157: 'utf-8' codec can't decode byte

0x85 in position 2: invalid start byte

File: D:\project\.git\objects\16\68d1aa999e79e5f18bedd7537ed24ec7c1f606

Error reading file D:\project\.git\objects\16\68d1aa999e79e5f18bedd7537ed24ec7c1f606: 'utf-8' codec can't decode byte 0xa9 in position 19: invalid start byte

File: D:\project\.git\objects\25\36959348f77cfdc107988635f3a4e524e754f8

Error reading file D:\project\.git\objects\25\36959348f77cfdc107988635f3a4e524e754f8: 'utf-8' codec can't decode byte 0x85 in position 15: invalid start byte

File: D:\project\.git\objects\26\efeafcaef70dec4f7aa456e3d340df4099b792

Error reading file D:\project\.git\objects\26\efeafcaef70dec4f7aa456e3d340df4099b792: 'utf-8' codec can't decode byte 0xca in position 3: invalid continuation byte

File: D:\project\.git\objects\27\1853e2aec26945f68da5303ffd79472ee8d4c7

Error reading file D:\project\.git\objects\27\1853e2aec26945f68da5303ffd79472ee8d4c7: 'utf-8' codec can't decode byte 0x9d in position 2: invalid start byte

File: D:\project\.git\objects\31\4b3c4a62ddf3c7e77276de0810642e661eebbc

Error reading file D:\project\.git\objects\31\4b3c4a62ddf3c7e77276de0810642e661eebbc: 'utf-8' codec can't decode byte 0xc5 in position 2: invalid continuation byte

File: D:\project\.git\objects\33\9ce40c7c673d44778f3644019eb7289808d85e

Error reading file D:\project\.git\objects\33\9ce40c7c673d44778f3644019eb7289808d85e: 'utf-8' codec can't decode byte 0x8f in position 3: invalid start byte

File: D:\project\.git\objects\34\407a3d0847ba93d8afc5886e5629286e79a504

Error reading file D:\project\.git\objects\34\407a3d0847ba93d8afc5886e5629286e79a504: 'utf-8' codec can't decode byte 0x8e in position 3: invalid start byte

File: D:\project\.git\objects\3b\43eb4f49f0946ff691593f505b7d25edf4fede

Error reading file D:\project\.git\objects\3b\43eb4f49f0946ff691593f505b7d25edf4fede: 'utf-8' codec can't decode byte 0xc8 in position 18: invalid continuation byte

File: D:\project\.git\objects\4a\52c57d4e3bc1d9e811afca11d6979d578a1940

Error reading file D:\project\.git\objects\4a\52c57d4e3bc1d9e811afca11d6979d578a1940: 'utf-8' codec can't decode byte 0x92 in position 3: invalid start byte

File: D:\project\.git\objects\58\e09d69db43519bb7db539903c80108110f6fa1

Error reading file D:\project\.git\objects\58\e09d69db43519bb7db539903c80108110f6fa1: 'utf-8' codec can't decode byte 0x95 in position 2: invalid start byte

File: D:\project\.git\objects\5a\3371002974e2bd460519791d23711c9952bb0d

Error reading file D:\project\.git\objects\5a\3371002974e2bd460519791d23711c9952bb0d: 'utf-8' codec can't decode byte 0xc8 in position 18: invalid continuation byte

File: D:\project\.git\objects\5e\8ea40cdee89cdf48df6875bb484d94051822b8

Error reading file D:\project\.git\objects\5e\8ea40cdee89cdf48df6875bb484d94051822b8: 'utf-8' codec can't decode byte 0xd5 in position 2: invalid continuation byte

File: D:\project\.git\objects\5f\258c905c4a01ac0cd0a8655965d5f0d9a0f935

Error reading file D:\project\.git\objects\5f\258c905c4a01ac0cd0a8655965d5f0d9a0f935: 'utf-8' codec can't decode byte 0xbb in position 3: invalid start byte

File: D:\project\.git\objects\6f\9991f28f9fd8ff58d9475604d55ef4c9eb4821

Error reading file D:\project\.git\objects\6f\9991f28f9fd8ff58d9475604d55ef4c9eb4821: 'utf-8' codec can't decode byte 0xff in position 7: invalid start byte

File: D:\project\.git\objects\7c\bbb16ca13f756abbbcb68dd9bbc9639dc749aa

Error reading file D:\project\.git\objects\7c\bbb16ca13f756abbbcb68dd9bbc9639dc749aa: 'utf-8' codec can't decode byte 0xd0 in position 21: invalid continuation byte

File: D:\project\.git\objects\83\a64e911ad09d58f96caaf1566a7a14cbbc633d

Error reading file D:\project\.git\objects\83\a64e911ad09d58f96caaf1566a7a14cbbc633d: 'utf-8' codec can't decode byte 0x85 in position 15: invalid start byte

File: D:\project\.git\objects\a6\5d17fc8f92cd706593206d7130d9b48d17f847

Error reading file D:\project\.git\objects\a6\5d17fc8f92cd706593206d7130d9b48d17f847: 'utf-8' codec can't decode byte 0xcf in position 4: invalid continuation byte

File: D:\project\.git\objects\ac\ca7f740a9602da261265fb0831c11589d86ff2

Error reading file D:\project\.git\objects\ac\ca7f740a9602da261265fb0831c11589d86ff2: 'utf-8' codec can't decode byte 0xca in position 23: invalid continuation byte

File: D:\project\.git\objects\ad\0147773378ff6b3bf129882fe1cc8a5bee723a

Error reading file D:\project\.git\objects\ad\0147773378ff6b3bf129882fe1cc8a5bee723a: 'utf-8' codec can't decode byte 0xad in position 2: invalid start byte

File: D:\project\.git\objects\c3\046d54c72777e4618d703042cdca1510701d10

Error reading file D:\project\.git\objects\c3\046d54c72777e4618d703042cdca1510701d10: 'utf-8' codec can't decode byte 0x85 in position 15: invalid start byte

File: D:\project\.git\objects\ca\6658a580d3f0ae3cc31467048ae0c9880e1a05

Error reading file D:\project\.git\objects\ca\6658a580d3f0ae3cc31467048ae0c9880e1a05: 'utf-8' codec can't decode byte 0x95 in position 2: invalid start byte

File: D:\project\.git\objects\cd\9695a4682225150a9ea8f98fd8ed1c892bf597

Error reading file D:\project\.git\objects\cd\9695a4682225150a9ea8f98fd8ed1c892bf597: 'utf-8' codec can't decode byte 0xca in position 3: invalid continuation byte

File: D:\project\.git\objects\d1\d7c6cac55220287b24717beaae48b29d2c7dc3

Error reading file D:\project\.git\objects\d1\d7c6cac55220287b24717beaae48b29d2c7dc3: 'utf-8' codec can't decode byte 0xb4 in position 9: invalid start byte

File: D:\project\.git\objects\d3\a69e88ae42d7554b6f11ec002673fe6bf5007b

Error reading file D:\project\.git\objects\d3\a69e88ae42d7554b6f11ec002673fe6bf5007b: 'utf-8' codec can't decode byte 0xed in position 2: invalid continuation byte

File: D:\project\.git\objects\df\9c57b48ada498562440a1529b15248ec6d6e17

Error reading file D:\project\.git\objects\df\9c57b48ada498562440a1529b15248ec6d6e17: 'utf-8' codec can't decode byte 0x95 in position 2: invalid start byte

File: D:\project\.git\objects\df\bbc13b4fa3d9827394f2f645cab2bc2772ce61

Error reading file D:\project\.git\objects\df\bbc13b4fa3d9827394f2f645cab2bc2772ce61: 'utf-8' codec can't decode bytes in position 2-3: invalid continuation byte

File: D:\project\.git\objects\e1\bee97aa351aef27d0f6f9ac2b6efab7c9561cf

Error reading file D:\project\.git\objects\e1\bee97aa351aef27d0f6f9ac2b6efab7c9561cf: 'utf-8' codec can't decode byte 0xd4 in position 6: invalid continuation byte

File: D:\project\.git\objects\e9\aa71c0cec8517b930227da076a4ff9e811ca2c

Error reading file D:\project\.git\objects\e9\aa71c0cec8517b930227da076a4ff9e811ca2c: 'utf-8' codec can't decode byte 0x91 in position 3: invalid start byte

File: D:\project\.git\objects\f2\49163981c2ff17bc5829fcf1e1890ec9a5d472

Error reading file D:\project\.git\objects\f2\49163981c2ff17bc5829fcf1e1890ec9a5d472: 'utf-8' codec can't decode byte 0x9d in position 2: invalid start byte

File: D:\project\.git\objects\f2\a712bc65659b478dddda2c7012f9a382908ddd

Error reading file D:\project\.git\objects\f2\a712bc65659b478dddda2c7012f9a382908ddd: 'utf-8' codec can't decode byte 0xc5 in position 2: invalid continuation byte

File: D:\project\.git\objects\f7\5ab0b4fe633405b984596bb96cfbc7f1322fe7

Error reading file D:\project\.git\objects\f7\5ab0b4fe633405b984596bb96cfbc7f1322fe7: 'utf-8' codec can't decode byte 0x8e in position 3: invalid start byte

File: D:\project\.git\objects\fe\9bbc86fcacb560026a55be5a6f5e2c5c7cf8a1

Error reading file D:\project\.git\objects\fe\9bbc86fcacb560026a55be5a6f5e2c5c7cf8a1: 'utf-8' codec can't decode byte 0xcd in position 2: invalid continuation byte

File: D:\project\.git\refs\heads\main

```
271853e2aec26945f68da5303ffd79472ee8d4c7
```

File: D:\project\.git\refs\remotes\origin\main

```
271853e2aec26945f68da5303ffd79472ee8d4c7
```

File: D:\project\routes\auth_routes.py

```
from flask import Blueprint, render_template, request, redirect, session

from extensions import mysql

import MySQLdb.cursors

import re


auth_bp = Blueprint('auth', __name__)


@auth_bp.route('/signup', methods=['GET', 'POST'])

def signup():

       if request.method == 'POST' and 'username' in request.form and 'password' in
request.form and 'role' in request.form:

        username = request.form['username']

        password = request.form['password']

        role = request.form['role']


        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

        cursor.execute('SELECT * FROM users WHERE username = %s', (username,))

        account = cursor.fetchone()


        if account:

            msg = 'Account already exists!'

        elif not re.match(r'[A-Za-z0-9]+', username):

            msg = 'Username must contain only characters and numbers!'

        elif not username or not password or not role:

            msg = 'Please fill out the form!'

        else:

            cursor.execute('INSERT INTO users (username, password, role) VALUES (%s, %s,
%s)', (username, password, role))

            mysql.connection.commit()

            msg = 'You have successfully registered!'

            return redirect('/signin')


        return render_template('signup.html', msg=msg)


    return render_template('signup.html')


@auth_bp.route('/signin', methods=['GET', 'POST'])

def signin():

      if request.method == 'POST' and 'username' in request.form and 'password' in

request.form:

        username = request.form['username']

        password = request.form['password']


        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

          cursor.execute('SELECT * FROM users WHERE username = %s AND password = %s',

(username, password))

        account = cursor.fetchone()


        if account:

            session['loggedin'] = True

            session['id'] = account['id']

            session['username'] = account['username']

            session['role'] = account['role']
            if account['role'] == 'Student':
                return redirect('/student_dashboard')
            elif account['role'] == 'Teacher':
                return redirect('/teacher_dashboard')
        else:
            msg = 'Incorrect username/password!'
            return render_template('signin.html', msg=msg)

    return render_template('signin.html')


@auth_bp.route('/student_dashboard')
def student_dashboard():
    if 'loggedin' in session and session.get('role') == 'Student':
        return render_template('student_dashboard.html', username=session['username'])
    return redirect('/signin')


@auth_bp.route('/teacher_dashboard')
def teacher_dashboard():
    if 'loggedin' in session and session.get('role') == 'Teacher':
        return render_template('teacher_dashboard.html', username=session['username'])
    return redirect('/signin')


@auth_bp.route('/upload_marks/<sem>')
def upload_marks(sem):
    # Render the upload marks template with the selected semester
    return render_template(f'upload_marks/{sem}.html')
```

File: D:\project\routes\dashboards.py

```
from flask import Blueprint, render_template, session, redirect



dashboard_bp = Blueprint('dashboard', __name__)



@dashboard_bp.route('/student_dashboard')

def student_dashboard():

    if 'loggedin' in session and session['role'] == 'Student':

        return render_template('student_dashboard.html', username=session['username'])

    return redirect('/signin')



@dashboard_bp.route('/teacher_dashboard')

def teacher_dashboard():

    if 'loggedin' in session and session['role'] == 'Teacher':

        return render_template('teacher_dashboard.html', username=session['username'])

    return redirect('/signin')
```

File: D:\project\routes\__pycache__\auth_routes.cpython-312.pyc

Error reading file D:\project\routes\__pycache__\auth_routes.cpython-312.pyc: 'utf-8' codec can't decode byte 0xcb in position 0: invalid continuation byte

File: D:\project\static\style.css

```
/* General form styling */

body {

    font-family: Arial, sans-serif;

    background-color: #f4f4f8;

    margin: 0;
    padding: 0;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

}


form {

    background: white;

    padding: 30px;

    border-radius: 10px;

    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.2);

    width: 400px;

    display: flex;

    flex-direction: column;

    gap: 15px;

}


label {

    font-size: 14px;

    font-weight: bold;

    margin-bottom: 5px;

}


/* Input field styling */

input {

    padding: 10px;
    font-size: 14px;

    border: 1px solid #ccc;

    border-radius: 5px;

}



input:focus {

    border-color: #007bff;

    outline: none;

}



/* Role selector container */

.role-selector {

    display: flex;

    justify-content: space-between;

    margin-top: 10px;

}



/* Role button styling */

.role-btn {

    flex: 1;

    padding: 10px;

    font-size: 14px;

    border: 2px solid #000;

    background: white;

    color: black;

    cursor: pointer;

    text-align: center;
    border-radius: 5px;

    margin: 0 5px;

    font-weight: bold;

    transition: background-color 0.3s, color 0.3s;

}


.role-btn:hover {

    background: #f0f0f0;

}


.role-btn.active {

    background: black;

    color: white;

    border-color: black;

}


/* Button styling */

button[type="submit"] {

    background-color: #28a745;

    color: white;

    padding: 10px;

    font-size: 16px;

    border: none;

    border-radius: 5px;

    cursor: pointer;

}
```

```
button[type="submit"]:hover {

    background-color: #218838;

}


/* Error message styling */

p {

    margin-top: 15px;

    color: red;

    font-size: 14px;

    text-align: center;

}
```

File: D:\project\static\teacher_style.css

```
body {

    margin: 0;

    font-family: Arial, sans-serif;

}


.ribbon {

    background-color: #995ce9;

    padding: 10px 20px;

    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);

    display: flex;

    align-items: center;

}


.ribbon .logo {
    display: flex;

    align-items: center;

    margin-right: 20px;

}


.ribbon .logo img {

    height: 40px;

    margin-right: 10px;

}


.ribbon .logo span {

    color: white;

    font-weight: bold;

}


.ribbon ul {

    list-style-type: none;

    margin: 0;

    padding: 0;

    display: flex;

    justify-content: center;

    flex-grow: 1;

    font-size: 20px;

}


.ribbon ul li {

    position: relative;
    margin: 0 15px;

}


.ribbon ul li a {

    text-decoration: none;

    color: white;

    font-weight: bold;

    transition: color 0.3s;

    padding: 10px;

    display: flex;

    align-items: center;

    cursor: pointer;

}


.ribbon ul li a:hover {

    color: #c112f7;

}


.ribbon ul li .arrow {

    margin-left: 5px;

    font-size: 0.8em;

}


.ribbon ul li .dropdown {

    display: none;

    position: absolute;

    background-color: white;
    color: black;

    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);

    border-radius: 5px;

    padding: 10px 0;

    min-width: 200px;

    z-index: 100;

}


.ribbon ul li .dropdown a {

    display: block;

    padding: 10px 15px;

    text-decoration: none;

    color: #814be6;

    font-weight: normal;

}


.ribbon ul li .dropdown a:hover {

    background-color: #c30fe7ea;

    color: white;

}


.ribbon ul li .nested-dropdown {

    display: none;

    position: absolute;

    background-color: white;

    color: black;

    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);
    border-radius: 5px;

    padding: 10px 0;

    min-width: 200px;

    left: 100%;

    top: 0;

}

/* General Styles */

body {

    margin: 0;

    font-family: Arial, sans-serif;

    background-color: #f8f9fa;

    color: #333;

}


h2 {

    color: #0056b3;

    margin-bottom: 10px;

}


p, ul {

    line-height: 1.6;

}


/* Ribbon Navigation Styles */

.ribbon {

    display: flex;

    align-items: center;
    background-color: #0056b3;

    padding: 10px 20px;

    color: #fff;

    justify-content: space-between;

    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

}


.ribbon .logo {

    display: flex;

    align-items: center;

}


.ribbon .logo img {

    height: 40px;

    margin-right: 10px;

}


.ribbon .logo span {

    font-size: 20px;

    font-weight: bold;

}


.ribbon ul {

    list-style: none;

    margin: 0;

    padding: 0;

    display: flex;
```

```
}

.ribbon ul li {
    position: relative;
    margin: 0 15px;
}

.ribbon ul li a {
    color: #fff;
    text-decoration: none;
    font-size: 16px;
    cursor: pointer;
    padding: 5px 10px;
    transition: background-color 0.3s;
}

.ribbon ul li a:hover {
    background-color: #004494;
    border-radius: 5px;
}

.ribbon .arrow {
    margin-left: 5px;
    font-size: 12px;
}

/* Dropdown Menu Styles */
```

```
.dropdown, .nested-dropdown {

    display: none;

    position: absolute;

    background-color: #fff;

    color: #333;

    min-width: 200px;

    padding: 10px;

    border: 1px solid #ddd;

    border-radius: 4px;

    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

    z-index: 1000;

}


.dropdown a, .nested-dropdown a {

    display: block;

    padding: 5px 10px;

    color: #333;

    text-decoration: none;

    font-size: 14px;

    transition: background-color 0.3s;

}


.dropdown a:hover, .nested-dropdown a:hover {

    background-color: #f1f1f1;

    border-radius: 3px;

}
```

```
.nested-dropdown {

    left: 100%;

    top: 0;

    margin-left: 5px;

}


/* Content Section */

.content {

    padding: 20px 40px;

    background-color: #fff;

    margin: 20px auto;

    max-width: 1000px;

    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

    border-radius: 8px;

}


.content section {

    margin-bottom: 30px;

}


.content ul {

    padding-left: 20px;

}


.content ul li {

    margin-bottom: 8px;

}
```

```
/* Responsive Design */
@media (max-width: 768px) {
    .ribbon {
        flex-direction: column;
        align-items: flex-start;
    }

    .ribbon ul {
        flex-direction: column;
    }

    .ribbon ul li {
        margin: 5px 0;
    }

    .content {
        padding: 20px;
        margin: 10px;
    }
}
.content-section {
    padding: 20px;
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    border: 1px solid #ddd;
    margin-top: 20px;
    border-radius: 5px;

}

.content-section h1 {

    font-size: 24px;

    color: #333;

}

.content-section ul {

    list-style-type: disc;

    margin-left: 20px;

}

.content-section p {

    font-size: 16px;

    line-height: 1.5;

}

<style>

    /* Carousel styling */

    .carousel {

        position: relative;

        width: 100%;

        max-width: 600px;

        margin: 20px auto;

        overflow: hidden;

        border: 2px solid #ddd;

        border-radius: 10px;

    }



    .carousel-images {
        display: flex;

        transition: transform 0.5s ease-in-out;

        width: 500%; /* Adjust based on the number of images */

    }


    .carousel-images img {

        width: 80%; /* Reduce width to 80% of its container */

        max-height: 300px; /* Set a max height to reduce size */

        object-fit: contain; /* Ensures the image fits well inside the box */

        margin: 0 auto; /* Centers the image horizontally */

        flex-shrink: 0;

    }



 /* Semesters Container */

.semesters-container {

    display: grid;

    grid-template-columns: repeat(4, 1fr); /* Adjust number of columns */

    gap: 10px;

    padding: 10px;

    background-color: #ffffff;

    border-radius: 5px;

    box-shadow: 0px 4px 6px rgba(0, 0, 0, 0.1);

    max-width: 400px; /* Set a maximum width */

}


/* Button Styling */
```

```
.semesters-container button {
    padding: 12px 20px;
    border: 1px solid #ddd;
    border-radius: 5px;
    cursor: pointer;
    background-color: #f9f9f9;
    color: #6a0dad; /* Purple text color */
    font-weight: bold;
    text-align: center;
    transition: background-color 0.3s ease, color 0.3s ease;
    font-size: 16px;
}

.semesters-container button:hover {
    background-color: #f2f2f2;
    color: #0056b3; /* Change text color on hover */
}

</style>
```

File: D:\project\static\images\aa.py

```
import os
from fpdf import FPDF

# Function to escape characters that FPDF cannot handle
def escape_special_characters(content):
    # Replace unsupported characters with '?'
    return ''.join([char if ord(char) < 256 else '?' for char in content])


# Function to write code from any file type to PDF
def write_code_to_pdf(folder_path, output_pdf):
    # Create an instance of FPDF
    pdf = FPDF()
    pdf.set_auto_page_break(auto=True, margin=15)
    pdf.add_page()

    # Set font for the PDF (monospace font for code)
    pdf.set_font("Courier", size=10)

    # Loop through all files in the directory
    for root, dirs, files in os.walk(folder_path):
        for file in files:
            file_path = os.path.join(root, file)

            # Write the file name as a header
            pdf.set_font("Arial", 'B', 12)
            pdf.cell(200, 10, f"File: {file_path}", ln=True, align='L')

            # Read the content of the file
            try:
                with open(file_path, 'r', encoding='utf-8') as f:
                    content = f.read()

                # Escape any special characters that may not be supported by FPDF
                content = escape_special_characters(content)

                # Write the file content to the PDF
                pdf.set_font("Courier", size=10)
                pdf.multi_cell(0, 10, content)
                pdf.ln(10)  # Add space between files
            except Exception as e:
                # If there's an error reading the file (e.g., binary files)
                pdf.set_font("Arial", size=10)
                pdf.multi_cell(0, 10, f"Error reading file {file_path}: {str(e)}")
                pdf.ln(10)  # Add space between files

    # Output the PDF to a file
    pdf.output(output_pdf)
    print(f"PDF generated: {output_pdf}")

# Example usage
folder_path = r'D:\project'  # Replace with your project folder path (use raw string for
Windows paths)
output_pdf = 'project_code.pdf'  # The output PDF file
write_code_to_pdf(folder_path, output_pdf)
```

File: D:\project\static\images\STD1.jpg

Error reading file D:\project\static\images\STD1.jpg: 'utf-8' codec can't decode byte 0xff in position 0: invalid start byte

File: D:\project\static\images\STD3.jpg

Error reading file D:\project\static\images\STD3.jpg: 'utf-8' codec can't decode byte 0xff in position 0: invalid start byte

File: D:\project\static\images\STD4.webp

Error reading file D:\project\static\images\STD4.webp: 'utf-8' codec can't decode byte 0x89 in position 0: invalid start byte

File: D:\project\static\images\STD5.png

Error reading file D:\project\static\images\STD5.png: 'utf-8' codec can't decode byte 0x89 in position 0: invalid start byte

File: D:\project\static\images\styles.css

```
body {

    margin: 0;

    font-family: Arial, sans-serif;

}



.ribbon {

    background-color: #995ce9;

    padding: 10px 20px;

    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);

    display: flex;

    align-items: center;

}



.ribbon .logo {

    display: flex;

    align-items: center;

    margin-right: 20px;

}
```

```
.ribbon .logo img {

    height: 40px;

    margin-right: 10px;

}


.ribbon .logo span {

    color: white;

    font-weight: bold;

}


.ribbon ul {

    list-style-type: none;

    margin: 0;

    padding: 0;

    display: flex;

    justify-content: center;

    flex-grow: 1;

    font-size: 20px;

}


.ribbon ul li {

    position: relative;

    margin: 0 15px;

}


.ribbon ul li a {

    text-decoration: none;
    color: white;

    font-weight: bold;

    transition: color 0.3s;

    padding: 10px;

    display: flex;

    align-items: center;

    cursor: pointer;

}


.ribbon ul li a:hover {

    color: #c112f7;

}


.ribbon ul li .arrow {

    margin-left: 5px;

    font-size: 0.8em;

}


.ribbon ul li .dropdown {

    display: none;

    position: absolute;

    background-color: white;

    color: black;

    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);

    border-radius: 5px;

    padding: 10px 0;

    min-width: 200px;
    z-index: 100;

}



.ribbon ul li .dropdown a {

    display: block;

    padding: 10px 15px;

    text-decoration: none;

    color: #814be6;

    font-weight: normal;

}



.ribbon ul li .dropdown a:hover {

    background-color: #c30fe7ea;

    color: white;

}



.ribbon ul li .nested-dropdown {

    display: none;

    position: absolute;

    background-color: white;

    color: black;

    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.2);

    border-radius: 5px;

    padding: 10px 0;

    min-width: 200px;

    left: 100%;

    top: 0;
```

```
}

/* General Styles */

body {

    margin: 0;

    font-family: Arial, sans-serif;

    background-color: #f8f9fa;

    color: #333;

}


h2 {

    color: #0056b3;

    margin-bottom: 10px;

}


p, ul {

    line-height: 1.6;

}


/* Ribbon Navigation Styles */

.ribbon {

    display: flex;

    align-items: center;

    background-color: #0056b3;

    padding: 10px 20px;

    color: #fff;

    justify-content: space-between;

    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
```

```
}

.ribbon .logo {
    display: flex;
    align-items: center;
}

.ribbon .logo img {
    height: 40px;
    margin-right: 10px;
}

.ribbon .logo span {
    font-size: 20px;
    font-weight: bold;
}

.ribbon ul {
    list-style: none;
    margin: 0;
    padding: 0;
    display: flex;
}

.ribbon ul li {
    position: relative;
    margin: 0 15px;
```

```
}

.ribbon ul li a {
    color: #fff;
    text-decoration: none;
    font-size: 16px;
    cursor: pointer;
    padding: 5px 10px;
    transition: background-color 0.3s;
}

.ribbon ul li a:hover {
    background-color: #004494;
    border-radius: 5px;
}

.ribbon .arrow {
    margin-left: 5px;
    font-size: 12px;
}

/* Dropdown Menu Styles */
.dropdown, .nested-dropdown {
    display: none;
    position: absolute;
    background-color: #fff;
    color: #333;
    min-width: 200px;
    padding: 10px;
    border: 1px solid #ddd;
    border-radius: 4px;
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    z-index: 1000;
}

.dropdown a, .nested-dropdown a {
    display: block;
    padding: 5px 10px;
    color: #333;
    text-decoration: none;
    font-size: 14px;
    transition: background-color 0.3s;
}

.dropdown a:hover, .nested-dropdown a:hover {
    background-color: #f1f1f1;
    border-radius: 3px;
}

.nested-dropdown {
    left: 100%;
    top: 0;
    margin-left: 5px;
}
```

```
/* Content Section */
.content {
    padding: 20px 40px;
    background-color: #fff;
    margin: 20px auto;
    max-width: 1000px;
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    border-radius: 8px;
}

.content section {
    margin-bottom: 30px;
}

.content ul {
    padding-left: 20px;
}

.content ul li {
    margin-bottom: 8px;
}

/* Responsive Design */
@media (max-width: 768px) {
    .ribbon {
        flex-direction: column;
        align-items: flex-start;

    }


    .ribbon ul {

        flex-direction: column;

    }


    .ribbon ul li {

        margin: 5px 0;

    }


    .content {

        padding: 20px;

        margin: 10px;

    }

}

.content-section {

    padding: 20px;

    font-family: Arial, sans-serif;

    background-color: #f4f4f4;

    border: 1px solid #ddd;

    margin-top: 20px;

    border-radius: 5px;

}

.content-section h1 {

    font-size: 24px;

    color: #333;
```

```
}

.content-section ul {

    list-style-type: disc;

    margin-left: 20px;

}

.content-section p {

    font-size: 16px;

    line-height: 1.5;

}

<style>

    /* Carousel styling */

    .carousel {

        position: relative;

        width: 100%;

        max-width: 600px;

        margin: 20px auto;

        overflow: hidden;

        border: 2px solid #ddd;

        border-radius: 10px;

    }


    .carousel-images {

        display: flex;

        transition: transform 0.5s ease-in-out;

        width: 500%; /* Adjust based on the number of images */

    }
    .carousel-images img {

        width: 80%; /* Reduce width to 80% of its container */

        max-height: 300px; /* Set a max height to reduce size */

        object-fit: contain; /* Ensures the image fits well inside the box */

        margin: 0 auto; /* Centers the image horizontally */

        flex-shrink: 0;

    }

</style>
```

File: D:\project\static\images\VTU.png

Error reading file D:\project\static\images\VTU.png: 'utf-8' codec can't decode byte 0x89 in position 0: invalid start byte

File: D:\project\templates\index.html

```
<!DOCTYPE html>

<html lang="en">

<head>

    <title>Home</title>

    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

</head>

<body>

    <h1>Welcome</h1>

    <a href="/signup">Sign Up</a> | <a href="/signin">Sign In</a>

</body>

</html>
```

File: D:\project\templates\signin.html

```
<!DOCTYPE html>
```

```
<html lang="en">

<head>

    <title>Sign In</title>

    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

</head>

<body>

    <form method="post">

        <label>Username:</label>

        <input type="text" name="username" required>

        <label>Password:</label>

        <input type="password" name="password" required>

        <button type="submit">Sign In</button>

    </form>

    {% if msg %}

        <p>{{ msg }}</p>

    {% endif %}

</body>

</html>
```

File: D:\project\templates\signup.html

```
<!DOCTYPE html>

<html lang="en">

<head>

    <title>Sign Up</title>

    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

</head>

<body>
```

```
    <form method="post">
        <label>Username:</label>
        <input type="text" name="username" required>

        <label>Password:</label>
        <input type="password" name="password" required>

        <label>Signup As:</label>
        <div class="role-selector">
                             <button  type="button"  class="role-btn"  id="student-btn"
onclick="selectRole('Student')">Student</button>
                             <button  type="button"  class="role-btn"  id="teacher-btn"
onclick="selectRole('Teacher')">Teacher</button>
        </div>
        <input type="hidden" name="role" id="role-input" value="">

        <button type="submit">Sign Up</button>
    </form>

    {% if msg %}
        <p>{{ msg }}</p>
    {% endif %}

    <script>
        function selectRole(role) {
            document.getElementById('role-input').value = role;
            const studentBtn = document.getElementById('student-btn');
            const teacherBtn = document.getElementById('teacher-btn');

            // Highlight the selected button
            if (role === 'Student') {
                studentBtn.classList.add('active');
                teacherBtn.classList.remove('active');
            } else {
                teacherBtn.classList.add('active');
                studentBtn.classList.remove('active');
            }
        }
    </script>
</body>
</html>
```

File: D:\project\templates\student_dashboard.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Student Dashboard</title>
</head>
<body>
    <h1>Welcome, {{ username }}</h1>
    <a href="/logout">Logout</a>
</body>
</html>
```

File: D:\project\templates\teacher_dashboard.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Teacher Dashboard</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='teacher_style.css') }}">
    <style>
        /* Carousel styling */
        .carousel {
            position: relative;
            width: 100%;
            max-width: 600px;
            margin: 20px auto;
            overflow: hidden;
            border: 2px solid #ddd;
            border-radius: 10px;
        }

        .carousel-images {
            display: flex;
            transition: transform 0.5s ease-in-out;
            width: 500%; /* Adjust based on the number of images */
        }

        .carousel-images img {
            width: 90%; /* Reduce size to 90% */
            max-height: 300px; /* Set a maximum height */
            margin: 0 auto; /* Center images horizontally */
            object-fit: cover; /* Ensures the image fits within its container */
            flex-shrink: 0;
        }

        /* Dropdown menu styling */
        .dropdown, .nested-dropdown {
            display: none;
            background-color: #f9f9f9;
            padding: 10px;
            border: 1px solid #ddd;
            border-radius: 5px;
            position: absolute;
        }

        .nested-dropdown {
            margin-left: 20px;
        }
    </style>
</head>
<body>
    <div class="ribbon">
        <!-- Logo -->
        <div class="logo">
            <img src="{{ url_for('static', filename='images/VTU.png') }}" alt="Logo">
            <span>STUDENT PERFORMANCE ANALYZER</span>
        </div>


        <!-- Navigation -->
        <ul>
            <li>
                <a onclick="showContent('home', this)">Home</a>
            </li>
            <li>
                <a onclick="toggleDropdown('about', this)">About <span
class="arrow">?</span></a>
                <div id="about" class="dropdown">
                    <a onclick="showMission()">Mission</a>
                    <a onclick="showContent('vision')">Vision</a>
                    <a onclick="showContent('faculty-info')">Faculty Info</a>
                </div>
            </li>
            <li>
                <a onclick="toggleDropdown('attendance', this)">Attendance <span
class="arrow">?</span></a>
                <div id="attendance" class="dropdown">
                    <a href="#1st-year-attendance">1st Year</a>
                    <a href="#2nd-year-attendance">2nd Year</a>
                    <a href="#3rd-year-attendance">3rd Year</a>
                </div>
            </li>
            <li>
```

```
                <a onclick="toggleDropdown('performance', this)">Performance <span class="arrow">?</span></a>
                <div id="performance" class="dropdown">
                    <div class="semesters-container">
                        <!-- Semesters in two rows -->
                        <button onclick="goToSemester(1)">1st Sem</button>
                        <button onclick="goToSemester(2)">2nd Sem</button>
                        <button onclick="goToSemester(3)">3rd Sem</button>
                        <button onclick="goToSemester(4)">4th Sem</button>
                        <button onclick="goToSemester(5)">5th Sem</button>
                        <button onclick="goToSemester(6)">6th Sem</button>
                        <button onclick="goToSemester(7)">7th Sem</button>
                        <button onclick="goToSemester(8)">8th Sem</button>
                    </div>
                </div>
            </li>
            <li>
                <a href="#solutions">Solutions</a>
            </li>
            <li>
                <a href="#upload">Upload</a>
            </li>
        </ul>
    </div>

    <!-- Content Section -->
    <div id="content" class="content-section">
```

```
        <p>Welcome! Click on the navigation options to view content.</p>
    </div>

    <script>
        // Show content dynamically and hide dropdowns for other sections
        function showContent(section) {
            const dropdowns = document.querySelectorAll('.dropdown, .nested-dropdown');
            const content = document.getElementById('content');

            // Hide all dropdowns
            dropdowns.forEach(dropdown => dropdown.style.display = "none");

            if (section === 'home') {
                content.innerHTML = `
                    <h1>Welcome to the Student Performance Analyzer</h1>
                    <div class="carousel">
                        <div class="carousel-images">
                              <img src="{{ url_for('static', filename='images/STD1.jpg')
}}" alt="Image 1">
                              <img src="{{ url_for('static', filename='images/STD2.jpg')
}}" alt="Image 2">
                              <img src="{{ url_for('static', filename='images/STD3.jpg')
}}" alt="Image 3">
                             <img src="{{ url_for('static', filename='images/STD4.webp')
}}" alt="Image 4">
                              <img src="{{ url_for('static', filename='images/STD5.png')
}}" alt="Image 5">
```

```
                        </div>
                    </div>
                    <p>
                                The <strong>Student Performance Analyzer</strong> is a
comprehensive tool designed to assist students, educators,
                                and administrators in monitoring and evaluating academic
progress. It provides detailed insights into:
                    </p>
                    <ul>
                        <li>Attendance tracking and analysis</li>
                        <li>Subject-wise and overall performance metrics</li>
                        <li>Individual performance evaluation</li>
                        <li>Data-driven insights for academic improvement</li>
                    </ul>
                    <p>
                                Explore the tool to make informed decisions that enhance
learning experiences and outcomes.
                    </p>
                `;
                startCarousel(); // Initialize the carousel
            }
        }

        function showMission() {
            const dropdowns = document.querySelectorAll('.dropdown, .nested-dropdown');
            const content = document.getElementById('content');
            // Hide all dropdowns
            dropdowns.forEach(dropdown => dropdown.style.display = "none");

            // Set Mission content dynamically
            content.innerHTML = `
                <h1>Our Mission</h1>
                <p><strong>Holistic Student Profiling:</strong></p>
                    <p>Our mission is to go beyond test scores and grades by analyzing attendance, participation, emotional well-being, and extracurricular involvement...</p>
                <!-- Add other mission content as described -->
            `;
        }

        // Carousel functionality
        function startCarousel() {
            const carouselImages = document.querySelector('.carousel-images');
            let index = 0;

            setInterval(() => {
                index++;
                 const offset = -(index % 5) * 100; // Update offset (5 is the number of images)
                carouselImages.style.transform = `translateX(${offset}%)`;
            }, 5000); // Change image every 5 seconds
        }

        // Toggle dropdowns and hide other open dropdowns
        function toggleDropdown(id) {
            const dropdown = document.getElementById(id);
            const dropdowns = document.querySelectorAll('.dropdown');

            // Hide all other dropdowns
            dropdowns.forEach(d => {
                if (d !== dropdown) d.style.display = "none";
            });

            // Toggle the current dropdown
            dropdown.style.display = dropdown.style.display === "block" ? "none" :
"block";
        }

        // Toggle nested dropdowns
        function toggleNestedDropdown(id) {
            const nestedDropdown = document.getElementById(id);
            const nestedDropdowns = document.querySelectorAll('.nested-dropdown');

            // Hide all other nested dropdowns
            nestedDropdowns.forEach(nd => {
                if (nd !== nestedDropdown) nd.style.display = "none";
            });

            // Toggle the current nested dropdown
            nestedDropdown.style.display = nestedDropdown.style.display === "block" ?
"none" : "block";
```

```
        }

            // Function to navigate to semester upload page

            function goToSemester(sem) {

                const url = `templates/upload_marks/${sem}`;

                     window.location.href = url; // Redirect to the new page with the
semester info

            }

    </script>

</body>

</html>
```

File: D:\project\templates\upload_marks.html

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Upload Semester Data</title>

    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

</head>

<body>

    <div class="container">

        <h1>Upload Data for Semester {{ semester }}</h1>

        <form action="/process-upload" method="POST" enctype="multipart/form-data">

            <input type="hidden" name="semester" value="{{ semester }}">

            <label for="file">Select Excel File:</label>

            <input type="file" name="file" id="file" required>
            <button type="submit">Upload</button>
        </form>
    </div>

    <style>
        .container {
            max-width: 600px;
            margin: 0 auto;
            text-align: center;
        }

        input[type="file"] {
            margin: 20px 0;
        }

        button {
            padding: 10px 20px;
            border: none;
            background-color: #007bff;
            color: white;
            border-radius: 5px;
            cursor: pointer;
        }

        button:hover {
            background-color: #0056b3;
        }
    </style>

</body>

</html>
```

File: D:\project\__pycache__\app.cpython-312.pyc

Error reading file D:\project\__pycache__\app.cpython-312.pyc: 'utf-8' codec can't decode byte 0xcb in position 0: invalid continuation byte

File: D:\project\__pycache__\extensions.cpython-312.pyc

Error reading file D:\project\__pycache__\extensions.cpython-312.pyc: 'utf-8' codec can't decode byte 0xcb in position 0: invalid continuation byte