

App.py					
Route	Success Criteria	Test Method	Result	Comment	Solution
@app.route	Redirects to the home page	Navigate to the app	Pass	This is the default route on page visit	
@app.route("/home") def index():	Navigates to the home page – if the user is logged in they are greeted with a custom message and the about text is no longer visible	Navigate to the home page whilst logged in and logged out	Pass		
@app.route("/events") @app.route("/get_events") def get_events():	Navigates to the events page and shows all events in the db (if there are any)	Navigate to the events page	Pass	If no events in db – an svg is displayed	
@app.route("/contact")	Navigates to the contact page	Navigate to the contact page	Pass		
@app.route("/register", methods=["GET", "POST"]) def register():	A user is registered in the database and redirected to the reports page – notifies the user if their username already exists	Register a new user  Attempt to register a current username	Pass		
@app.route("/login", methods=["GET", "POST"]) def login():	Log the user into the app – validates the password and user against records in the db  Warns the user if their credentials are invalid	Login with correct credentials  Login with invalid credentials	Pass		
@app.route("/logout") def logout():	Logs the user out of the app and redirects the to the home page  Their session token is destroyed	Logout of app	Pass		
@app.route("/create-report", methods=["POST"]) def createReport():	Creates a new report in the db and returns a success message	Create new report Console the response Inspect the db	Pass		
@app.route("/reports/<username>", methods=["GET", "POST"]) def reports(username):	Checks the db for reports by the username  Returns all reports	Login to the app – auto redirected to the reports page	Pass		
@app.route("/update-report",	Updates the specific	Update a report	Pass		

methods=["POST"])	report selected by the user  Updates the document in the db  Returns a json response with a success message	Console the response Inspect the db			
@app.route("/delete-report", methods=["POST"]) def deleteReport():	deletes the specific report selected by the user  deletes the document in the db  Returns a json response with a success message	Delete a report Console the response Inspect the db	Pass		
@app.route("/upload-image", methods=["POST"]) @cross_origin() def upload_file():	Posts images to Cloudinary storage  Returns a response containing a success / error message  A URL for the image is returned  A json response is sent to the front end	Attach an image to a report  Check the Cloudinary dashboard  Console,log the json response	Pass		
@app.errorhandler(404) def four_oh_four_error(e):	Navigates to a custom 404 page if the user visits an invalid url	Navigate to an invalid url	Pass		
@app.errorhandler(500) def five_oh_oh_error(e):	Navigates to a custom 500 page	Commented out code / used an exception handler /	Pass		