

NHẬP MÔN CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN MÔN HỌC #1 REPORT

Tài liệu này mô tả nội dung đồ án môn học cho môn học Nhập môn Công nghệ thông tin theo hướng chuyên ngành Công nghệ phần mềm – Hệ thống thông tin và Mạng máy tính

Nhóm <20CLC05-02>

<20127122> - <Phạm Thái Bình>

<20127206> - <Vũ Đình Duy Khánh>

<20127478> - <Huỳnh Tuấn Duy>

<20127492> - <Liêu Gia Hạo>

<20127636> - <Lương Văn Thông>

<20127650> - <Nguyễn Thị Phương Trang>



Khoa Công nghệ Thông tin
Đại học Khoa học Tự nhiên TP HCM
Tháng Thg11-20

MỤC LỤC

1	Tổng quan.....	3
	Thông tin nhóm.....	3
	Thông tin đồ án.....	3
2	Nội dung đồ án	4
2.1	Sử dụng ứng dụng Google Calendar	4
2.2	Tìm hiểu về GitHub.....	13
2.2.1.	Chức năng.....	13
2.2.2.	Cách hoạt động	13
2.2.3.	Hướng dẫn cách tải mã nguồn lên GitHub với nhiều phiên bản từ các thành viên trong nhóm	14
3	Đánh giá	24
3.1	Đánh giá mức độ hoàn thành.....	24
3.2	Đánh giá tỷ lệ đóng góp	24

1

Tổng quan

Thông tin nhóm

MSSV	Họ tên	Email	Vai trò
20127122	Phạm Thái Bình	20127122@student.hcmus.edu.vn	Nhóm Trưởng
20127206	Vũ Đình Duy Khánh	20127206@student.hcmus.edu.vn	Thủ Quỹ
20127478	Huỳnh Tuấn Duy	20127478@student.hcmus.edu.vn	Thành Viên
20127492	Liêu Gia Hạo	20127492@student.hcmus.edu.vn	Thành Viên
20127636	Lương Văn Thông	20127636@student.hcmus.edu.vn	Thành Viên
20127650	Nguyễn Thị Phương Trang	20127650@student.hcmus.edu.vn	Thư Ký

Thông tin đồ án

Tên đồ án	Nội dung
Sử dụng ứng dụng Google Calendar	<ul style="list-style-type: none"> - Lập lịch làm việc nhóm - Chia sẻ lịch làm việc cho các thành viên - Gửi email thông báo lịch cho các thành viên - Các thành viên nhóm tự thêm công việc cần làm vào mục Task trong tài khoản Gmail cá nhân
Tìm hiểu về Github	<ul style="list-style-type: none"> - Chức năng - Cách hoạt động - Hướng dẫn cách tải mã nguồn lên GitHub với nhiều phiên bản từ các thành viên

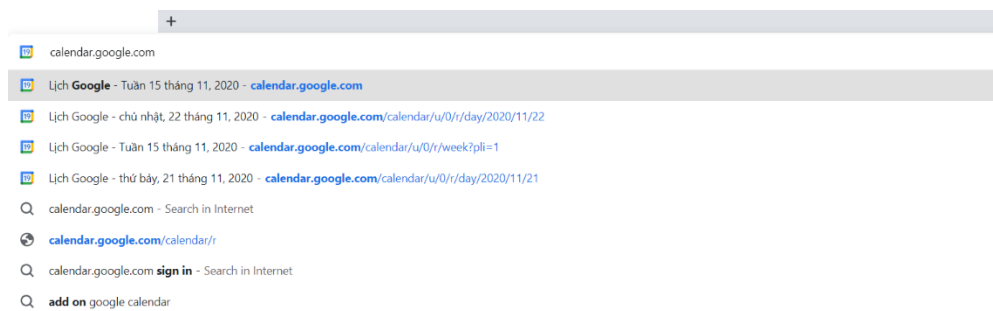
2

Nội dung đồ án

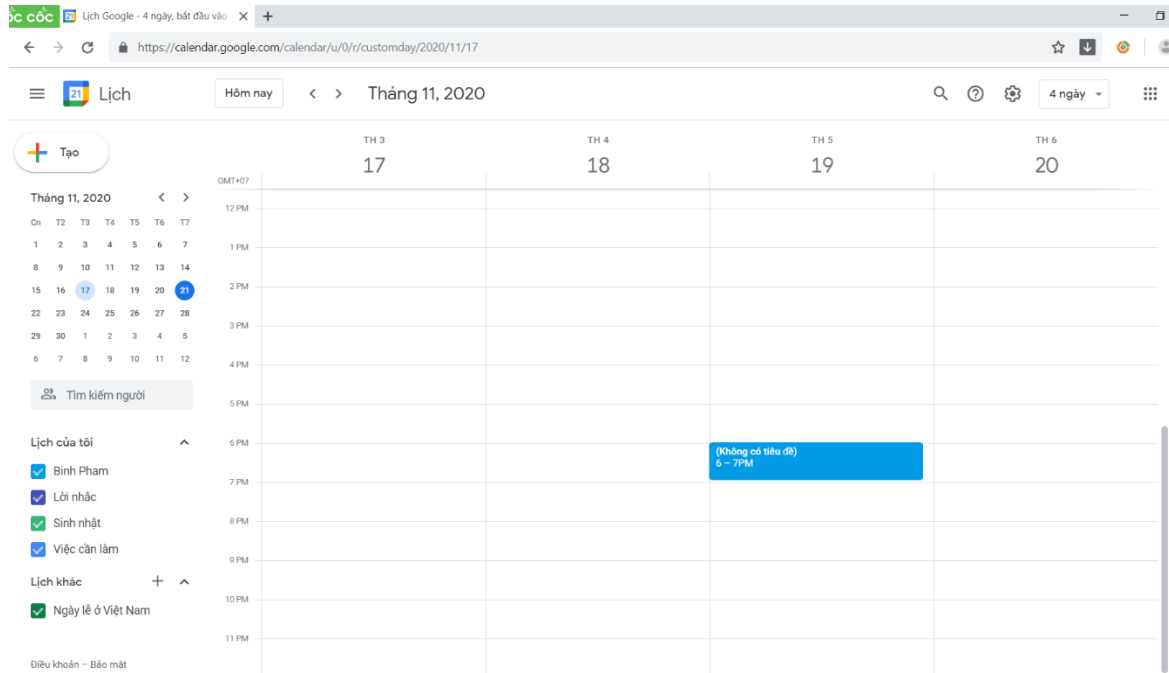
2.1

Sử dụng ứng dụng Google Calendar

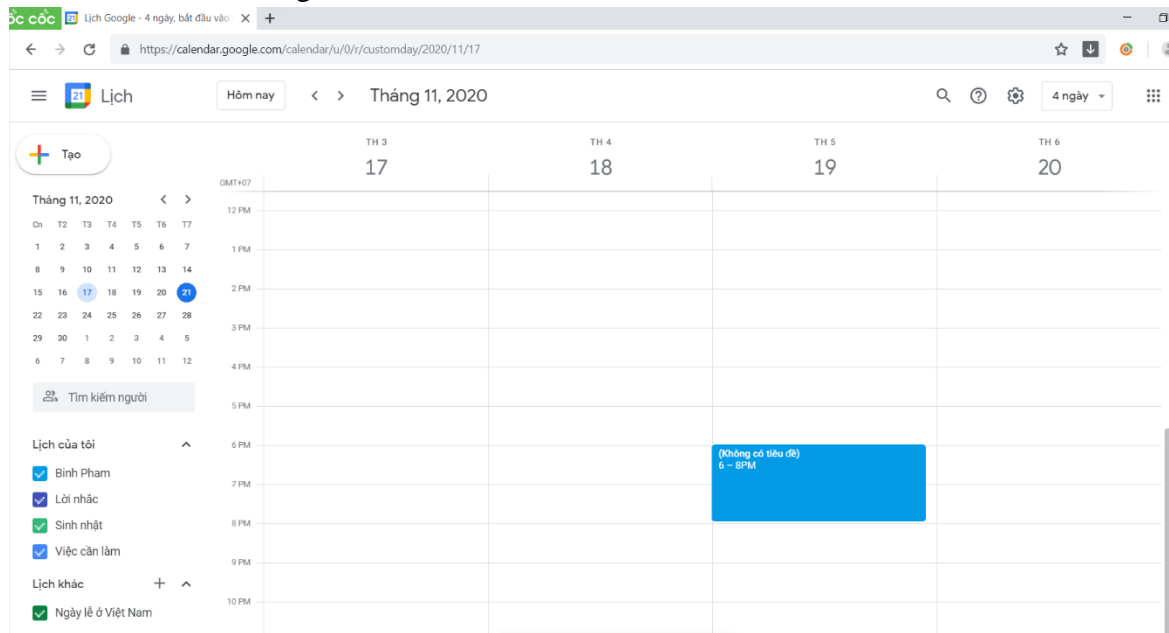
1. Bước 1: Truy cập vào đường link: calendar.google.com



2. Bước 2: Chọn mốc thời gian cho từng kế hoạch để nộp bài



3. Bước 3: Đặt mốc thời gian



4. Bước 4: Bấm vào chỗ mình đã chọn và phân công công việc

× Phân công nhiệm vụ cho từng thành viên

19 thg 11, 2020 6:00PM tới 8:00PM 19 thg 11, 2020 Múi giờ

☐ Cả ngày ☐ Không lặp lại

Chi tiết sự kiện

Thêm vị trí

Thông báo 30 phút

Thêm thông báo

binhpham.1512202@gmail.com

Bản Chế độ hiển thị mặc định

- Nguyễn Thị Phương Trang: tổng hợp thông tin
 - Vũ Đình Duy Khánh: tìm hiểu chức năng và cách hoạt động GitHub
 - Huỳnh Tuấn Duy: hướng dẫn cách tải mã nguồn lên GitHub với nhiều phiên bản (COMMIT MESSAGE, FOLLOWER)
 - Liêu Gia Hào: hướng dẫn cách tải mã nguồn lên GitHub với nhiều phiên bản (FEED, CHECK MODIFICATIONS)
 - Lương Văn Thông: hướng dẫn cách tải mã nguồn lên GitHub với nhiều phiên bản (SHOW HISTORY & MERGE SOURCE)
 Phạm Thái Bình: viết nền tảng cấu trúc kiến trúc

Khách

Thêm khách

Quyền của khách

☐ Sửa đổi sự kiện
☒ Mời những người khác
☒ Xem danh sách khách mời

5. Bước 5: Sau khi phân công kế hoạch chọn Lưu

The image displays two screenshots of the Google Calendar event creation interface. Both screenshots are in Vietnamese and show the 'Chi tiết sự kiện' (Event details) section.

Top Screenshot: 'Hoàn thành phần tìm hiểu về GitHub'

- Title:** Hoàn thành phần tìm hiểu về GitHub
- Date/Time:** 21 thg 11, 2020, 8:00PM tới 10:00PM, 21 thg 11, 2020, Múi giờ
- Location:** Thêm vị trí
- Duration:** 30 phút
- Guests:** binhpham.1512202@gmail.com
- Permissions:**
 - ☐ Sửa đổi sự kiện
 - ☒ Mời những người khác
 - ☒ Xem danh sách khách mời
- Notes:**
 - Các bạn Khánh, Hao, Duy, Thông nộp file word
 - Ban Trạng tổng hợp thông tin

Bottom Screenshot: 'Nộp bài báo cáo trên moodle'

- Title:** Nộp bài báo cáo trên moodle
- Date/Time:** 23 thg 11, 2020, 8:00PM tới 12:00AM, 23 thg 11, 2020, Múi giờ
- Location:** Thêm vị trí
- Duration:** 30 phút
- Guests:** binhpham.1512202@gmail.com
- Permissions:**
 - ☐ Sửa đổi sự kiện
 - ☒ Mời những người khác
 - ☒ Xem danh sách khách mời
- Notes:**
 - Nhóm trưởng nộp

Google - Lịch Google - Chỉ tiết sự kiện

https://calendar.google.com/calendar/u/0/r/eventedit/MWdhdWFmCXFqNGRzJHJvdnBIZDpYWVWdHdWfmluaHBoYW0uMTUxMjhwMkBT

Gửi tổng hợp cho nhóm trưởng

Lưu Thao tác khác

22 thg 11, 2020 6:00PM tới **10:00PM** 22 thg 11, 2020 Múi giờ

☐ Cả ngày ☐ Không lặp lại

Chi tiết sự kiện Tìm một Thời điểm

Thêm hội nghị truyền hình trên Google Meet

Thêm vị trí

Thông báo 30 phút X

Thêm thông báo

binhpham.1512202@gmail.com

Bạn Chế độ hiển thị mặc định

Bạn Trang gửi file tổng hợp thông tin cho nhóm trưởng

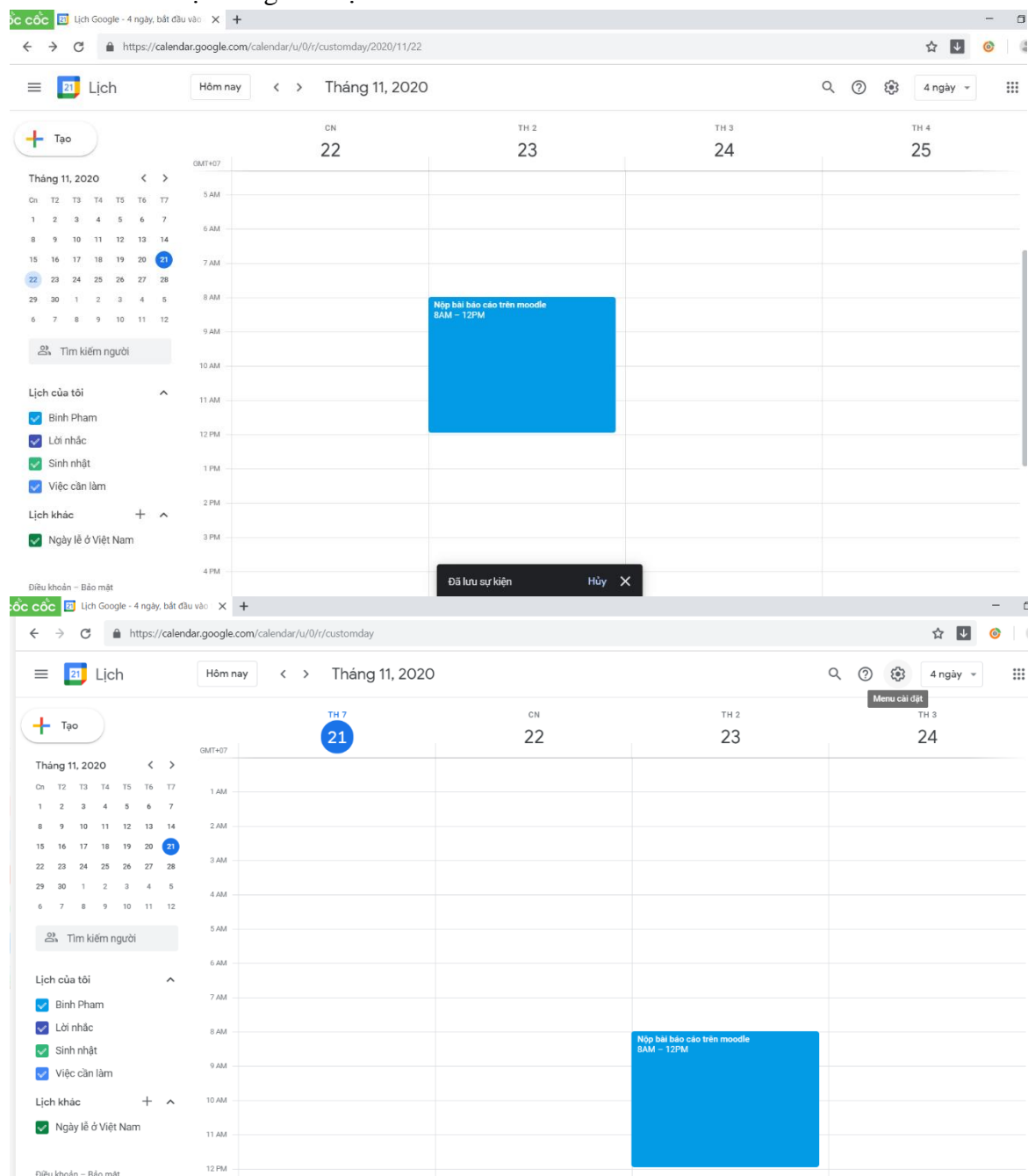
Khách

Thêm khách

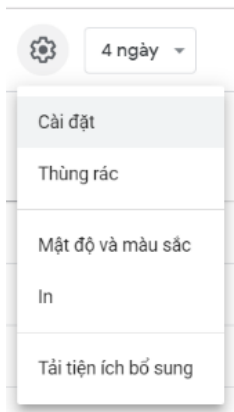
Quyền của khách

- ☐ Sửa đổi sự kiện
- ☒ Mời những người khác
- ☒ Xem danh sách khách mời

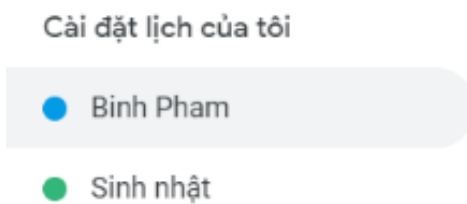
6. Bước 6: Thu được bảng kế hoạch



7. Bước 7: Nếu hoàn thành lịch cho các thành viên thì bấm vào chỗ CÀI ĐẶT




8. Bước 8: Bấm vào chỗ CÀI ĐẶT LỊCH CỦA TÔI



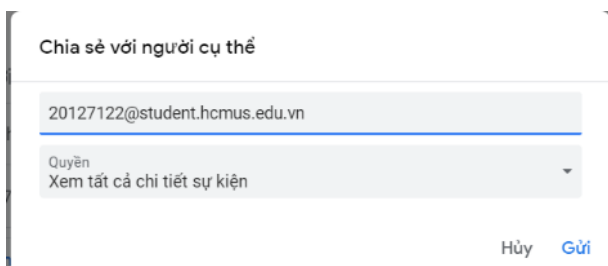
9. Bước 9: bấm vào CHIA SẺ VỚI NGƯỜI CỤ THỂ → THÊM NGƯỜI

Chia sẻ với người cụ thể

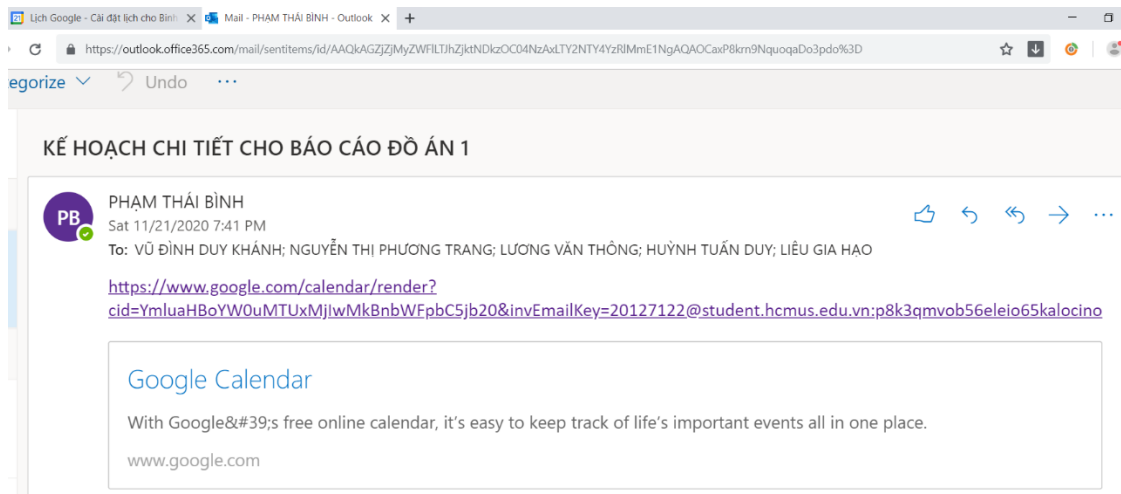
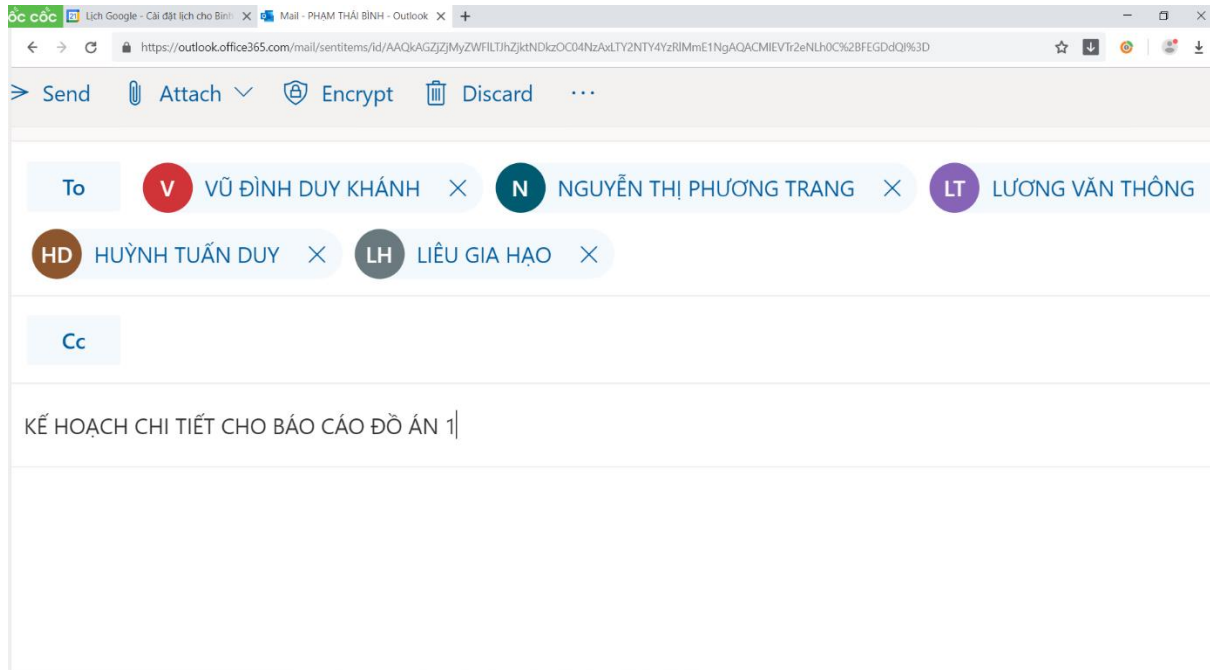
 binhpham.1512202@gmail.com (Chủ sở hữu)

[+ Thêm người](#)

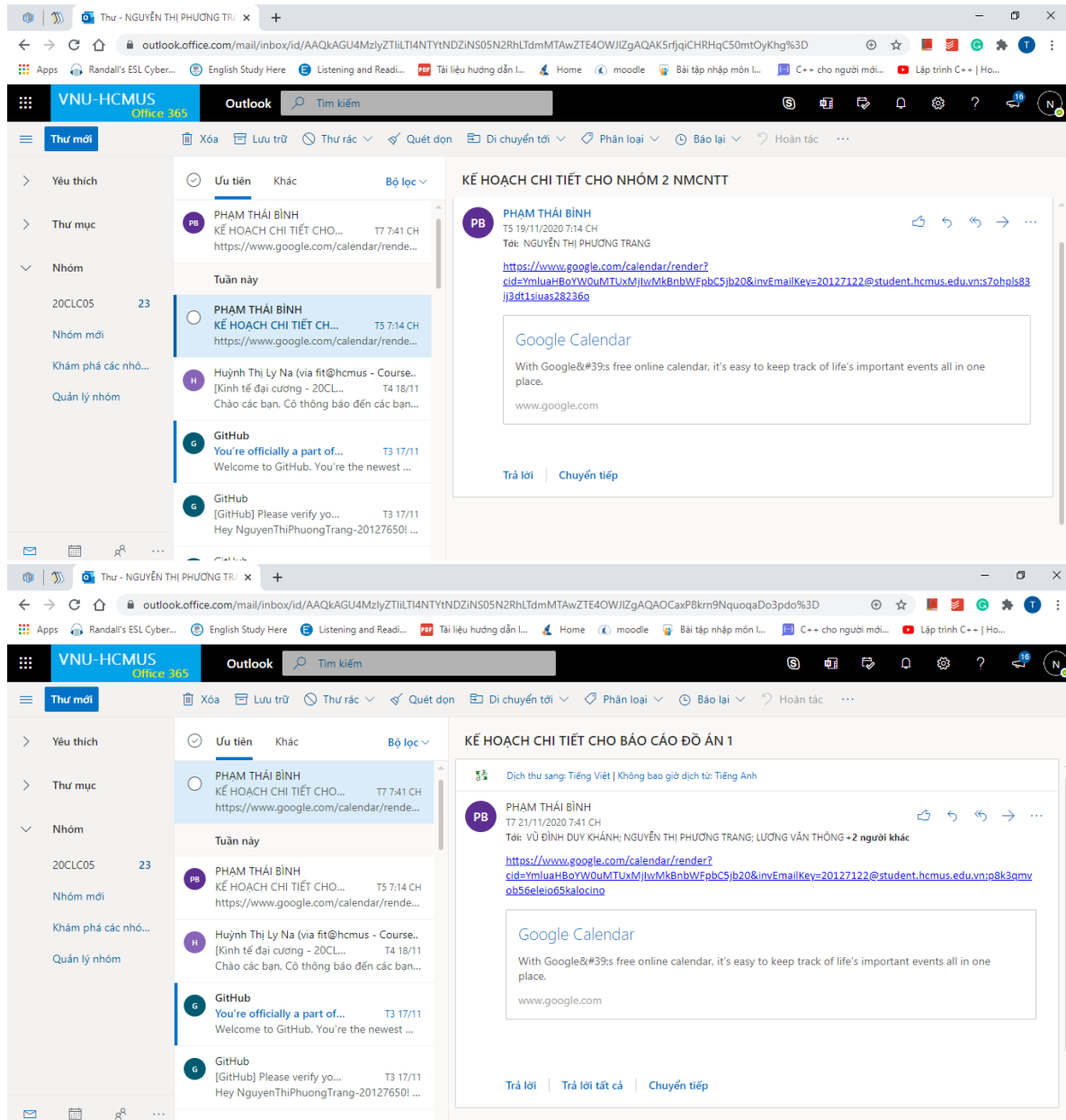
10. Bước 10: NHẬP ĐỊA CHỈ MAIL SINH VIÊN CHO CÁC THÀNH VIÊN. Sau đó bấm GỬI



11. Bước 11: Vào địa chỉ gmail sinh viên gửi cho các thành viên



12. Bước 12: Các thành viên kiểm tra email và đã nhận được thông báo



2.2

Tìm hiểu về GitHub

2.2.1. Chức năng

- GitHub được coi là một mạng xã hội dành cho lập trình viên lớn nhất và dễ dùng nhất với các tính năng cốt lõi như:

1. Wiki, issue, thống kê, đổi tên project, project được đặt vào namespace là user.
2. Watch project: theo dõi hoạt động của project của người khác. Xem quá trình người khác phát triển phần mềm thế nào, project phát triển ra sao.
3. Follow user: theo dõi hoạt động của người khác.

- Có 2 cách tiếp cận GitHub:

+ Tạo project của riêng mình

+ Contribute cho project có sẵn: fork project có sẵn của người khác, sửa đổi, sau đó đề nghị họ cập nhật sửa đổi của mình (tạo pull request).

2.2.2. Cách hoạt động

a) Remote repository

- Hiểu đơn giản là một nơi để lưu local repository lên Internet - nhưng nó không lưu tất cả mà chỉ lưu những cái người dùng push lên.
- Qua đó, người dùng có thể dễ dàng lấy nó về để sử dụng bất cứ lúc nào, và bất cứ ở đâu và có thể share cho những người khác cùng vào sử dụng.
- Nó có thể là một dịch vụ máy chủ Repository bất kỳ như: Github, Bitbucket,...

b) Git structures

- Trong các hệ thống quản lý phiên bản Version Control System thì dữ liệu sẽ được lưu trữ ở hai nơi:
 - Workspace hay Working Tree
 - Local repository
- Nhưng với Git thì có thêm một lựa chọn nữa đó là khu vực trung gian gọi là Staging Area hay Index - và đây cũng chính là lợi thế lớn của Git.

Staging Area là khu vực sẽ lưu trữ những thay đổi của bạn trên tập tin để nó có thể commit, vì đối với Git muốn commit tập tin nào thì tập tin ấy phải nằm trong Staging Area. Một tập tin khi nằm trong Staging Area thì sẽ có trạng thái là Staged.

- Như vậy Git Structures sẽ có 3 phần

1. Workspace/ Working Tree
2. Staging Area/ Index
3. Local repository

Tương ứng với sự thay đổi từng trạng thái của 1 tập tin từ trên xuống dưới.

- Khi làm việc với Git thì các file của chúng ta sẽ có 2 trạng thái chính đó là Tracked và Untracked

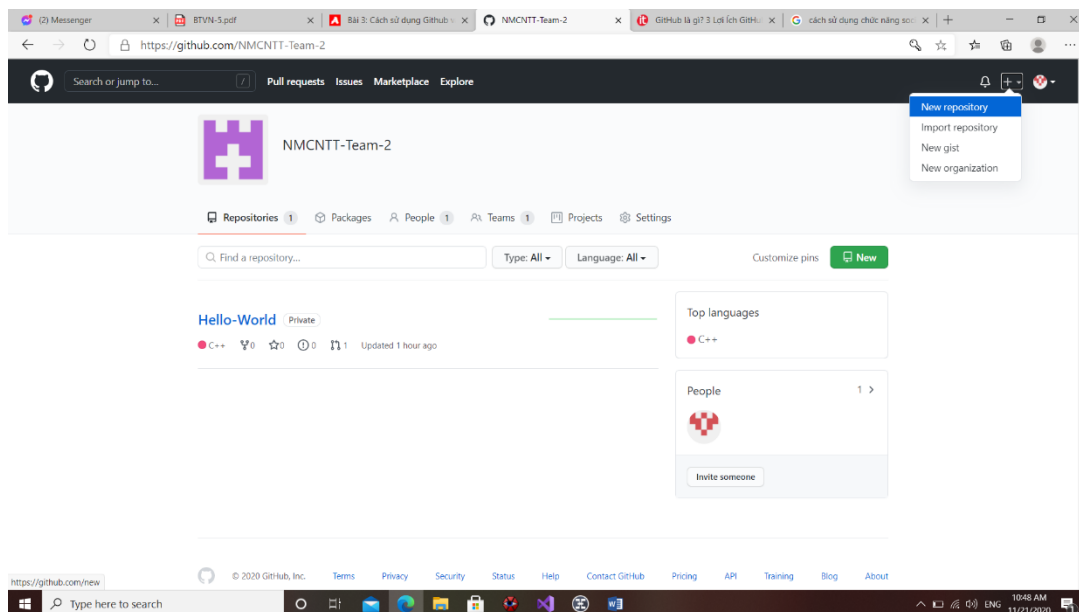
- Tracked: là tập tin đã được đánh dấu theo dõi trong Git để bạn làm việc với nó. Trong Tracked sẽ có thêm các trạng thái phụ khác là: Unmodified (chưa chỉnh sửa), Modified (đã chỉnh sửa nhưng chưa đưa vào Staging Area), Staged (đã nằm trong Staging Area và sẵn sàng commit)
- Untracked: là các tập tin mới được tạo ra và chưa được Git đánh dấu theo dõi.

2.2.3. Hướng dẫn cách tải mã nguồn lên GitHub với nhiều phiên bản từ các thành viên trong nhóm

a) Commit Message

I. Tạo 1 repository:

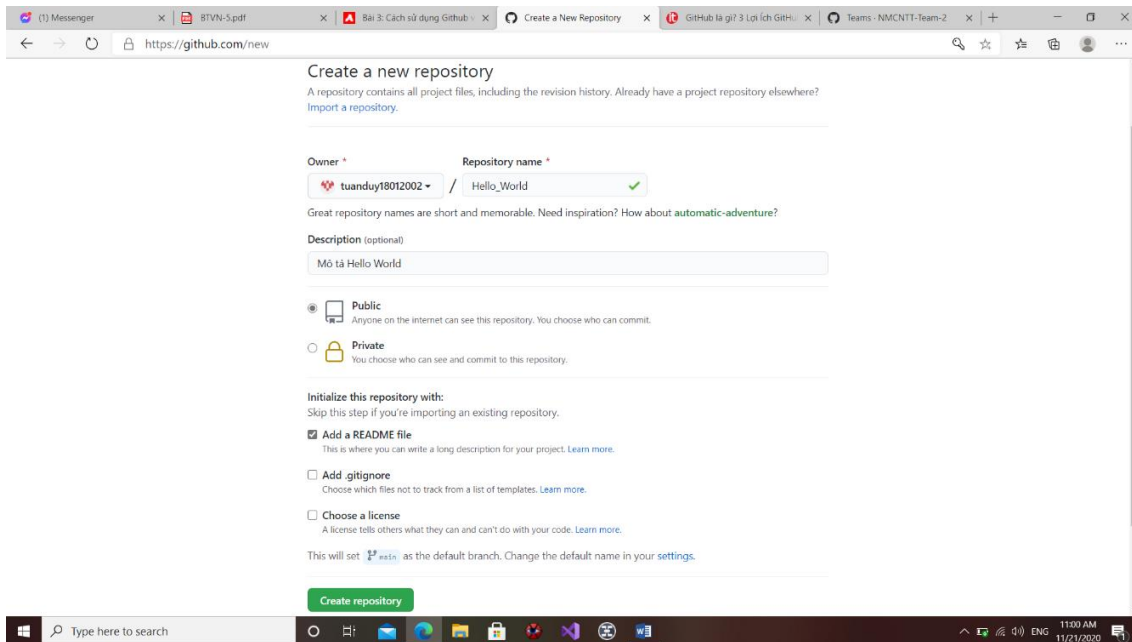
1. Bước 1: Bấm vào  sau đó chọn New repository.



2. Bước 2:

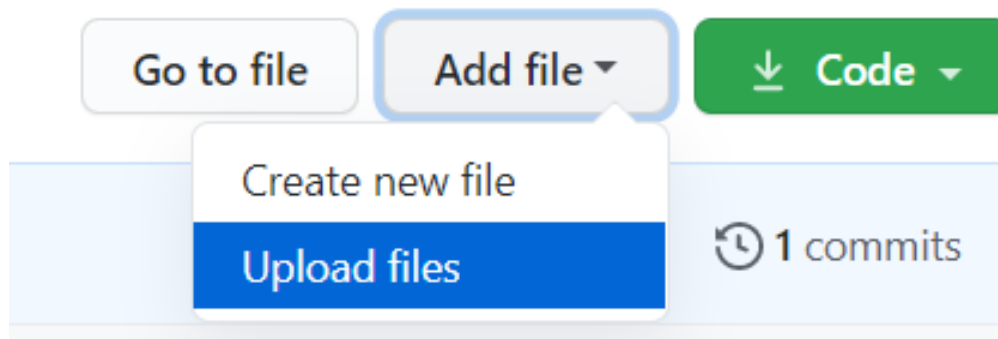
- Điền tên Repository vào ô Repository name
- Điền mô tả về repository vào ô Description
- Chọn chế độ Public (công khai) hoặc Private (cá nhân)

3. Bước 3: Nhấn vào nút Creat Repository ở phía dưới



II. Cách upload file lên Repository:

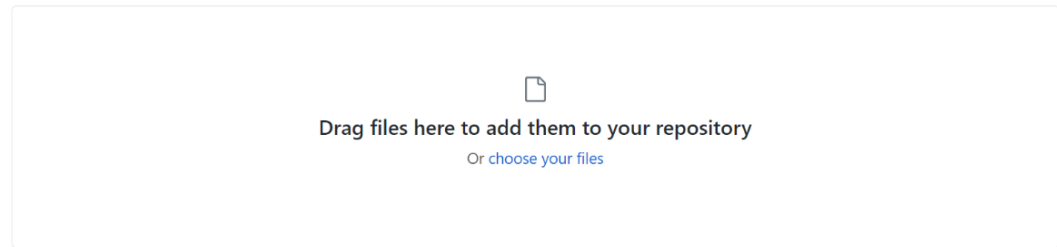
1. Bước 1: Chọn Add file → Upload files



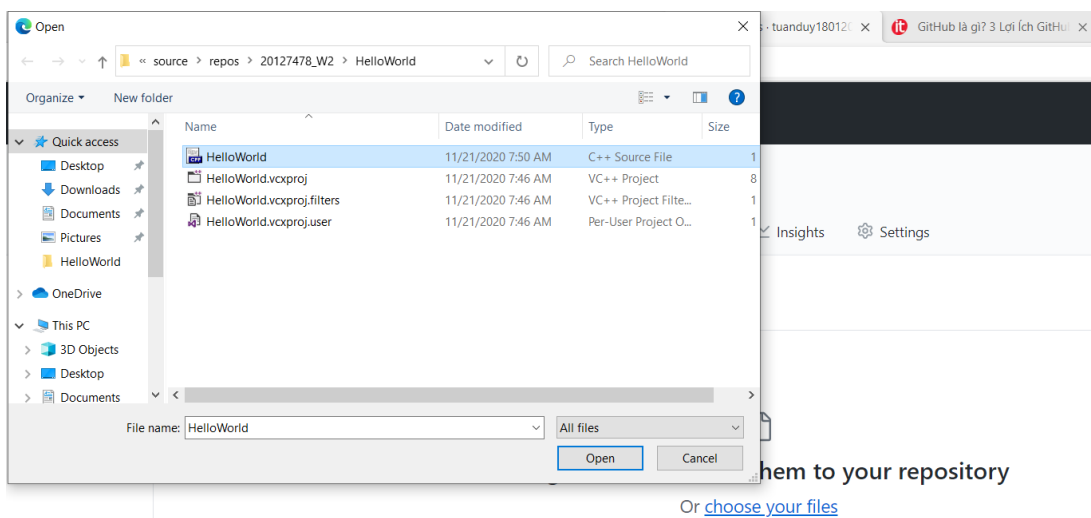
2. Bước 2:

- Chọn choose your file

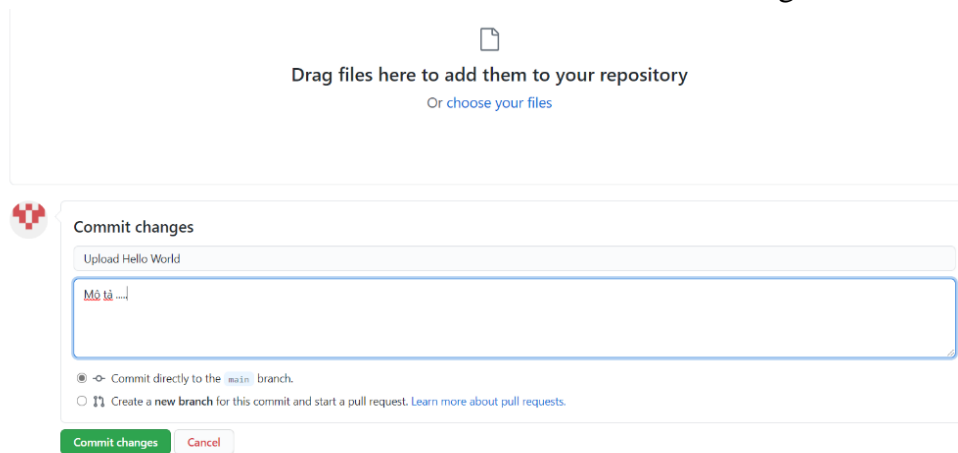
Hello_World /



- Chọn file cần upload rồi nhấn Open



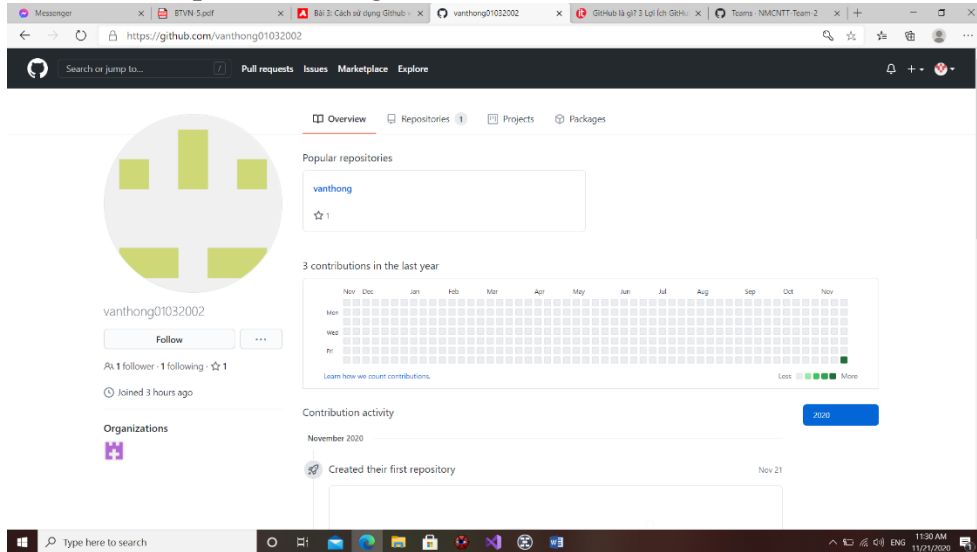
3. Bước 3: Điền vào tiêu đề file và mô tả ở ô Commit changes



4. Bước 4: Chọn nút Commit changes để hoàn thành upload

b) Follower

1. Bước 1: Vào profile của người cần follow



2. Bước 2: Nhấn vào nút Follow ở phía dưới Avatar

c) Feed

GitHub cung cấp một số tài nguyên dòng thời gian ở định dạng Atom. API nguồn cấp dữ liệu liệt kê tất cả các nguồn cấp dữ liệu có sẵn cho người dùng đã xác thực:

- **Dòng thời gian : Dòng thời gian công khai toàn cầu của GitHub**
- **Người dùng :** Dòng thời gian công khai cho bất kỳ người dùng nào, sử dụng mẫu URI
- **Người dùng hiện tại công khai :** Dòng thời gian công khai cho người dùng đã xác thực
- **Người dùng hiện tại :** Dòng thời gian riêng tư cho người dùng đã xác thực
- **Tác nhân người dùng hiện tại :** Dòng thời gian riêng tư cho hoạt động do người dùng đã xác thực tạo
- **Tổ chức người dùng hiện tại :** Dòng thời gian riêng tư cho các tổ chức mà người dùng đã xác thực là thành viên.
- **Cổ vấn bảo mật :** Tập hợp các thông báo công khai cung cấp thông tin về các lỗ hổng liên quan đến bảo mật trong phần mềm trên GitHub.

Lưu ý : Nguồn cấp dữ liệu riêng tư chỉ được trả lại khi xác thực qua Basic Auth vì URI nguồn cấp dữ liệu hiện tại sử dụng mã thông báo xác thực cũ hơn, không thể thu hồi.

d) Check modification

Người dùng nhận được thông báo cho các cuộc trò chuyện trong kho mà họ xem bao gồm:

- Vấn đề và nhận xét của họ
- Yêu cầu kéo và nhận xét của họ
- Nhận xét về bất kỳ cam kết nào

Thông báo cũng được gửi cho các cuộc trò chuyện trong kho chưa được xem khi người dùng tham gia, bao gồm:

- **@mentions**
- Phát hành bài tập
- Cam kết tác giả hoặc cam kết của người dùng
- Bất kỳ cuộc thảo luận nào mà người dùng tích cực tham gia

Tất cả các lệnh gọi API thông báo đều yêu cầu notification hoặc repo phạm vi API. Làm điều này sẽ cấp quyền truy cập chỉ đọc vào một số vấn đề và nội dung cam kết. Bạn vẫn sẽ cần repo phạm vi để truy cập các vấn đề và cam kết từ các điểm cuối tương ứng của chúng.

Thông báo trở lại dưới dạng "chủ đề". Một chuỗi chứa thông tin về cuộc thảo luận hiện tại của một vấn đề, yêu cầu kéo hoặc cam kết.

Thông báo được tối ưu hóa để thăm dò ý kiến với Last-Modified tiêu đề. Nếu không có thông báo mới, bạn sẽ thấy một 304 Last Modified phản hồi, giữ nguyên giới hạn tốc độ hiện tại của bạn. Có một X-Poll-Interval tiêu đề chỉ định tần suất (tính bằng giây) bạn được phép thăm dò. Trong thời gian máy chủ tải cao, thời gian có thể tăng lên. Vui lòng tuân theo tiêu đề.

Lý do thông báo

Khi truy xuất phản hồi từ API thông báo, mỗi trọng tải có một khóa có tiêu đề reason. Chúng tương ứng với các sự kiện kích hoạt thông báo.

Đây là danh sách những người có thể reason nhận được thông báo:

Tên lý do	Sự miêu tả
assign	Bạn đã được chỉ định cho vấn đề.
author	Bạn đã tạo chủ đề.
comment	Bạn đã nhận xét về chủ đề.
invitation	Bạn đã chấp nhận lời mời đóng góp vào kho lưu trữ.
manual	Bạn đã đăng ký chủ đề (thông qua sự cố hoặc yêu cầu kéo).
mention	Bạn đã được đề cập cụ thể @ trong nội dung.
review_requested	Bạn hoặc một nhóm mà bạn là thành viên đã được yêu cầu xem xét một yêu cầu kéo.
security_alert	GitHub đã phát hiện ra một lỗ hổng bảo mật trong kho lưu trữ của bạn.
state_change	Bạn đã thay đổi trạng thái luồng (ví dụ: đóng sự cố hoặc hợp nhất một yêu cầu kéo).
subscribed	Bạn đang xem kho lưu trữ.
team_mention	Bạn đã ở trong một đội đã được đề cập.

Lưu ý rằng điều reason này được sửa đổi trên cơ sở từng luồng và có thể thay đổi, nếu reason thông báo sau này khác.

Ví dụ, nếu bạn là tác giả của một vấn đề, các thông báo tiếp theo về vấn đề đó sẽ có một reason số author. Nếu bạn đang sau đó @mentioned về cùng một vấn đề, các thông báo bạn lấy sau đó sẽ có một reason số mention. Các reason di tích như mention, bất kể bạn đang từng đề cập một lần nữa.

Liệt kê thông báo cho người dùng đã xác thực

Liệt kê tất cả các thông báo cho người dùng hiện tại, được sắp xếp theo cập nhật gần đây nhất.

GET /notifications

Thông số

TÊN	Kiểu	TRONG	SỰ MIÊU TẢ
ACCEPT	chuỗi	tiêu đề	Nên đặt thành . application/vnd.github.v3+json
ALL	boolean	truy vấn	Nếu true, hiển thị thông báo được đánh dấu là đã đọc.
PARTICIPATING	boolean	truy vấn	Nếu true, chỉ hiển thị các thông báo mà người dùng đang trực tiếp tham gia hoặc được đề cập.
SINCE	chuỗi	truy vấn	Chỉ hiển thị các thông báo được cập nhật sau thời gian nhất định. Đây là một dấu thời gian trong ISO 8601 định dạng: YYYY-MM-DDTHH:MM:SSZ.
BEFORE	chuỗi	truy vấn	Chỉ hiển thị các thông báo được cập nhật trước thời gian nhất định. Đây là một dấu thời gian trong ISO 8601 định dạng: YYYY-MM-DDTHH:MM:SSZ.
PER_PAGE	số nguyên	truy vấn	Kết quả trên mỗi trang (tối đa 100)
PAGE	số nguyên	truy vấn	Số trang của kết quả cần tìm nạp.

Mẫu mã

Shell

```
curl \
  -H "Accept: application/vnd.github.v3+json" \
  https://api.github.com/notifications
```

JavaScript ([@octokit / core.js](#))

```
await octokit.request('GET /notifications')
```

Phản hồi mặc định

```
Status: 200 OK
[
  {
    "id": "1",
    "repository": {
      "id": 1296269,
      "node_id": "MDEwO1JlcG9zaXRvcnkxMjk2MjY5",
      "name": "Hello-World",
      "full_name": "octocat/Hello-World",
      "owner": {
        "login": "octocat",
        "id": 1,
        "node_id": "MDQ6VXNlcjE=",
        "avatar_url": "https://github.com/images/error/octocat_happy.gif",
```

```

    "gravatar_id": "",
    "url": "https://api.github.com/users/octocat",
    "html_url": "https://github.com/octocat",
    "followers_url": "https://api.github.com/users/octocat/followers",
    "following_url": "https://api.github.com/users/octocat/following{/other_user}",
    "gists_url": "https://api.github.com/users/octocat/gists{/gist_id}",
    "starred_url": "https://api.github.com/users/octocat/starred{/owner}/{/repo}",
    "subscriptions_url": "https://api.github.com/users/octocat/subscriptions",
    "organizations_url": "https://api.github.com/users/octocat/orgs",
    "repos_url": "https://api.github.com/users/octocat/repos",
    "events_url": "https://api.github.com/users/octocat/events{/privacy}",
    "received_events_url": "https://api.github.com/users/octocat/received_events",
    "type": "User",
    "site_admin": false
  },
  "private": false,
  "html_url": "https://github.com/octocat/Hello-World",
  "description": "This your first repo!",
  "fork": false,
  "url": "https://api.github.com/repos/octocat/Hello-World",
  "archive_url": "https://api.github.com/repos/octocat/Hello-World/{archive_format}/{/ref}",
  "assignees_url": "https://api.github.com/repos/octocat/Hello-World/assignees{/user}",
  "blobs_url": "https://api.github.com/repos/octocat/Hello-World/git/blobs{/sha}",
  "branches_url": "https://api.github.com/repos/octocat/Hello-World/branches{/branch}",
  "collaborators_url": "https://api.github.com/repos/octocat/Hello-World/collaborators{/collaborator}",
  "comments_url": "https://api.github.com/repos/octocat/Hello-World/comments{/number}",
  "commits_url": "https://api.github.com/repos/octocat/Hello-World/commits{/sha}",
  "compare_url": "https://api.github.com/repos/octocat/Hello-World/compare/{base}...{head}",
  "contents_url": "https://api.github.com/repos/octocat/Hello-World/contents/{+path}",
  "contributors_url": "https://api.github.com/repos/octocat/Hello-World/contributors",
  "deployments_url": "https://api.github.com/repos/octocat/Hello-World/deployments",
  "downloads_url": "https://api.github.com/repos/octocat/Hello-World/downloads",
  "events_url": "https://api.github.com/repos/octocat/Hello-World/events",
  "forks_url": "https://api.github.com/repos/octocat/Hello-World/forks",
  "git_commits_url": "https://api.github.com/repos/octocat/Hello-World/git/commits{/sha}",
  "git_refs_url": "https://api.github.com/repos/octocat/Hello-World/git/refs{/sha}",
  "git_tags_url": "https://api.github.com/repos/octocat/Hello-World/git/tags{/sha}",
  "git_url": "git:github.com/octocat/Hello-World.git",
  "issue_comment_url": "https://api.github.com/repos/octocat/Hello-World/issues/comments{/number}",
  "issue_events_url": "https://api.github.com/repos/octocat/Hello-World/issues/events{/number}",
  "issues_url": "https://api.github.com/repos/octocat/Hello-World/issues{/number}",
  "keys_url": "https://api.github.com/repos/octocat/Hello-World/keys{/key_id}",
  "labels_url": "https://api.github.com/repos/octocat/Hello-World/labels{/name}",
  "languages_url": "https://api.github.com/repos/octocat/Hello-World/languages",
  "merges_url": "https://api.github.com/repos/octocat/Hello-World/merges",
  "milestones_url": "https://api.github.com/repos/octocat/Hello-World/milestones{/number}",

```

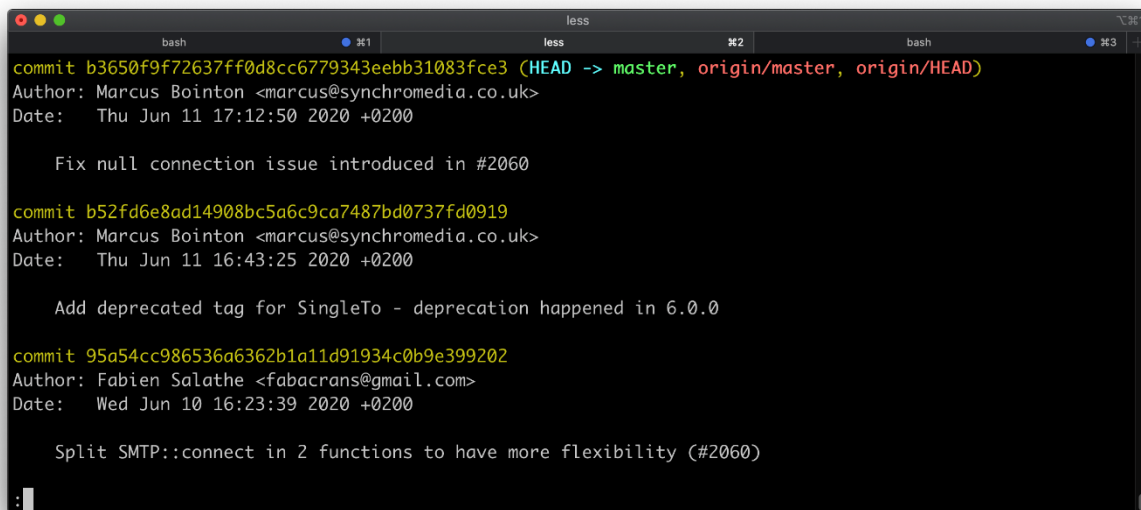
```

        "notifications_url": "https://api.github.com/repos/octocat/Hello-World/notifications{?since,all,participating}",
        "pulls_url": "https://api.github.com/repos/octocat/Hello-World/pulls{/number}",
        "releases_url": "https://api.github.com/repos/octocat/Hello-World/releases{/id}",
        "ssh_url": "git@github.com:octocat/Hello-World.git",
        "stargazers_url": "https://api.github.com/repos/octocat/Hello-World/stargazers",
        "statuses_url": "https://api.github.com/repos/octocat/Hello-World/statuses/{sha}",
        "subscribers_url": "https://api.github.com/repos/octocat/Hello-World/subscribers",
        "subscription_url": "https://api.github.com/repos/octocat/Hello-World/subscription",
        "tags_url": "https://api.github.com/repos/octocat/Hello-World/tags",
        "teams_url": "https://api.github.com/repos/octocat/Hello-World/teams",
        "trees_url": "https://api.github.com/repos/octocat/Hello-World/git/trees{/sha}",
        "hooks_url": "http://api.github.com/repos/octocat/Hello-World/hooks"
    },
    "subject": {
        "title": "Greetings",
        "url": "https://api.github.com/repos/octokit/octokit.rb/issues/123",
        "latest_comment_url":
        "https://api.github.com/repos/octokit/octokit.rb/issues/comments/123",
        "type": "Issue"
    },
    "reason": "subscribed",
    "unread": true,
    "updated_at": "2014-11-07T22:01:45Z",
    "last_read_at": "2014-11-07T22:01:45Z",
    "url": "https://api.github.com/notifications/threads/1"
}
]
    
```

Không sửa đổi	Status: 304 Not Modified
Yêu cầu xác thực	Status: 401 Not Modified
Bị cấm	Status: 403 Not Modified
Xác thực không thành công	Status: 422 Not Modified

e) Show history

- Tìm kiếm, xem lại, lọc các thông tin log lịch sử commit trên git bằng lệnh git log với các thiết lập khác nhau, như pretty, hiển thị đồ thị log
- Lệnh git log giúp bạn xem lại thông tin lịch sử commit, nhằm giám sát sự thay đổi của dự án. Lệnh git log có nhiều tham số để xuất ra, định dạng các thông tin hiển thị theo cách mong muốn. Bạn có thể định dạng cách các thông tin mỗi commit được in ra khi xem, cũng như có thể lọc thông tin nào đó.
- Mặc định thì lệnh git log liệt kê các commit theo thứ tự từ mới nhất đến cũ nhất, mỗi commit có các thông tin gồm: mã hash của commit, dòng thông báo, người tạo commit và ngày tạo commit.



```

commit b3650f9f72637ff0d8cc6779343eebb31083fce3 (HEAD -> master, origin/master, origin/HEAD)
Author: Marcus Bointon <marcus@synchronmedia.co.uk>
Date: Thu Jun 11 17:12:50 2020 +0200

    Fix null connection issue introduced in #2060

commit b52fd6e8ad14908bc5a6c9ca7487bd0737fd0919
Author: Marcus Bointon <marcus@synchronmedia.co.uk>
Date: Thu Jun 11 16:43:25 2020 +0200

    Add deprecated tag for SingleTo - deprecation happened in 6.0.0

commit 95a54cc986536a6362b1a11d91934c0b9e399202
Author: Fabien Salathe <fabacrans@gmail.com>
Date: Wed Jun 10 16:23:39 2020 +0200

    Split SMTP::connect in 2 functions to have more flexibility (#2060)
    
```

f) Merge Source

Lệnh merge này cho phép bạn hợp nhất những thay đổi vào một branch.

- Click vào Merge pull request để hợp nhất những thay đổi vào master branch.
- Click vào Confirm merge

3

Đánh giá

3.1

Đánh giá mức độ hoàn thành

Các công việc hoàn thành	Các công việc chưa hoàn thành / sẽ cập nhật thêm
Tìm hiểu về Google Calendar	
Tìm hiểu GitHub	

3.2

Đánh giá tỷ lệ đóng góp

MSSV	Họ tên	Công việc	Tỷ lệ	Ghi chú
20127122	Phạm Thái Bình	Tìm hiểu Google Calendar	100%	
20127206	Vũ Đình Duy Khánh	Tìm hiểu GitHub	100%	
20127478	Huỳnh Tuấn Duy	Tìm hiểu GitHub	100%	
20127492	Liêu Gia Hạo	Tìm hiểu GitHub	100%	
20127636	Lương Văn Thông	Tìm hiểu GitHub	100%	
20127650	Nguyễn Thị Phương Trang	Tổng hợp thông tin	100%	