

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  
KHOA CÔNG NGHỆ THÔNG TIN**

---



# **ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI**

## **XÂY DỰNG WEBSITE TO DO LIST**

Giảng viên hướng dẫn : Bùi Ngọc Dũng  
Sinh viên thực hiện : Trần Ngọc Lâm  
Lớp : CNTT6 – K60  
Mã sinh viên : 191200592

**Hà Nội – 2023**

## LỜI CẢM ƠN

Xin gửi lời cảm ơn chân thành và sâu sắc đến các thầy cô giảng dạy và hướng dẫn trong quá trình thực hiện đồ án này. Sự hỗ trợ, chỉ dẫn và động viên từ phía quý thầy cô đã đóng góp không nhỏ vào việc hoàn thiện và thành công của đề tài. Đầu tiên, xin gửi lời cảm ơn đến thầy Bùi Ngọc Dũng - người đã đồng hành và truyền đạt những kiến thức quý báu về công nghệ thông tin và lĩnh vực phát triển ứng dụng web. Cũng xin gửi lời cảm ơn đến các thầy cô khác trong khoa/khối ngành công nghệ thông tin, những người đã chia sẻ kiến thức, kinh nghiệm và sự đóng góp ý kiến trong quá trình thảo luận và thực hiện đồ án. Sự tận tâm và sự chia sẻ không điều kiện của quý thầy cô là nguồn động lực lớn để vượt qua những khó khăn và hoàn thiện đề tài một cách tốt nhất. Với lòng biết ơn chân thành, kính chúc quý thầy cô luôn dồi dào sức khỏe, thành công trong sự nghiệp giảng dạy và hướng dẫn, và tiếp tục lan tỏa những kiến thức và tinh thần sáng tạo đến các thế hệ sinh viên tiếp theo.

Hà Nội, Ngày 20 tháng 05 năm 2023

Sinh viên thực hiện

Lâm

Trần Ngọc Lâm

## Mục lục

<b>LỜI CẢM ƠN.....</b>	<b>2</b>
<b>DANH MỤC CÁC TỪ VIẾT TẮT.....</b>	<b>5</b>
<b>DANH MỤC HÌNH ẢNH.....</b>	<b>6</b>
<b>DANH SÁCH BẢNG BIỂU .....</b>	<b>8</b>
<b>LỜI NÓI ĐẦU .....</b>	<b>9</b>
<b>CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....</b>	<b>10</b>
1.    Đặt vấn đề .....	10
2.    Lý do chọn đề tài .....	11
3.    Mục tiêu, nhiệm vụ đề tài .....	11
4.    Phương pháp nghiên cứu .....	12
5.    Đối tượng nghiên cứu .....	12
6.    Phạm vi nghiên cứu .....	12
7.    Tổng quan về công nghệ.....	13
7.1.    React JS và Firebase <sup>[1]</sup> .....	13
7.2.    React Testing Library <sup>[2]</sup> .....	14
8.    Khảo sát .....	15
8.1.    Khảo sát hiện trạng .....	15
8.2.    Mục tiêu khảo sát.....	15
8.3.    Kết quả khảo sát thực tế .....	16
9.    Kết quả dự kiến đạt được.....	18
<b>CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....</b>	<b>19</b>

<b>1.</b>	<b>Phân tích hệ thống .....</b>	<b>19</b>
1.1.	Tham khảo một số website tương tự.....	19
1.2.	Mô tả .....	20
1.3.	Tác nhân tham gia hệ thống .....	20
1.4.	Yêu cầu chung và chức năng của hệ thống .....	21
1.5.	Phân tích bài toán .....	22
<b>2.</b>	<b>Sơ đồ phân rã chức năng(BFD) .....</b>	<b>22</b>
<b>3.</b>	<b>Sơ đồ thiết kế bằng UML .....</b>	<b>24</b>
3.1.	Biểu đồ ca sử dụng.....	24
3.1.1.	Biểu đồ ca sử dụng của hệ thống .....	24
3.1.2.	Quản lý project .....	24
3.1.3.	Quản lý task.....	26
3.1.4.	Tìm kiếm.....	28
3.1.5.	Thống kê .....	30
3.2.	Xây dựng biểu đồ trình tự.....	32
3.2.1.	Thêm project .....	32
3.2.2.	Sửa project .....	32
3.2.3.	Xoá project .....	33
3.2.4.	Tìm kiếm.....	33
3.2.5.	Thêm task .....	34
3.2.6.	Sửa task.....	34
3.2.7.	Hoàn thành task.....	35
3.2.8.	Thống kê .....	35
<b>4.</b>	<b>Thiết kế cơ sở dữ liệu.....</b>	<b>36</b>
4.1.	Thực thể dữ liệu .....	36
<b>5.</b>	<b>Sơ đồ ứng dụng.....</b>	<b>41</b>
<b>CHƯƠNG 3: GIAO DIỆN HỆ THỐNG VÀ KIỂM THỬ.....</b>		<b>42</b>
<b>KẾT LUẬN.....</b>		<b>54</b>
<b>TÀI LIỆU THAM KHẢO .....</b>		<b>55</b>

## DANH MỤC CÁC TỪ VIẾT TẮT

Viết tắt	Ý nghĩa
BFD	Bussiness Function Diagram
UML	Unified Modeling Languge

## DANH MỤC HÌNH ẢNH

<b>Hình 2.1.1:</b> Website tương tự .....	19
<b>Hình 2.2.1:</b> Sơ đồ phân rã chức năng .....	23
<b>Hình 2.3.1:</b> Biểu đồ ca sử dụng của hệ thống.....	24
<b>Hình 2.3.2:</b> Biểu đồ ca sử dụng quản lý project .....	24
<b>Hình 2.3.3:</b> Biểu đồ ca sử dụng quản lý task.....	26
<b>Hình 2.3.4:</b> Biểu đồ ca sử dụng tìm kiếm.....	28
<b>Hình 2.3.5:</b> Biểu đồ usecase thống kê .....	30
<b>Hình 2.3.6:</b> Biểu đồ trình tự - Thêm Project.....	32
<b>Hình 2.3.7:</b> Biểu đồ trình tự - Sửa Project.....	32
<b>Hình 2.3.8:</b> Biểu đồ trình tự - Xoá Project .....	33
<b>Hình 2.3.9:</b> Biểu đồ trình tự - Tìm kiếm.....	33
<b>Hình 2.3.10:</b> Biểu đồ trình tự - Thêm task.....	34
<b>Hình 2.3.11:</b> Biểu đồ trình tự - Sửa task.....	34
<b>Hình 2.3.12:</b> Biểu đồ trình tự - Hoàn thành task .....	35
<b>Hình 2.3.13:</b> Biểu đồ trình tự - Thống kê .....	35
<b>Hình 2.4.1:</b> Thực thể dữ liệu Todoist .....	36
<b>Hình 2.4.2:</b> Collection trong Todoist.....	37
<b>Hình 2.4.3:</b> Document trong Todoist.....	38
<b>Hình 2.4.4:</b> Các trường của project .....	39
<b>Hình 2.4.5:</b> Các trường của tasks.....	40
<b>Hình 2.4.6:</b> Các trường của users .....	40
<b>Hình 2.5.1:</b> Sơ đồ kiến trúc ứng dụng.....	41
<b>Hình 3.1:</b> Giao diện trang chính .....	42
<b>Hình 3.2:</b> Chức năng tìm kiếm 1 .....	43
<b>Hình 3.3:</b> Chức năng tìm kiếm 2 .....	44
<b>Hình 3.4:</b> Thêm task nhanh 1.....	44
<b>Hình 3.5:</b> Thêm task nhanh 2.....	45
<b>Hình 3.6:</b> Thêm task nhanh 3.....	45
<b>Hình 3.7:</b> Thêm task nhanh 4.....	46
<b>Hình 3.8:</b> Thêm task nhanh 5 .....	46
<b>Hình 3.9:</b> Chức năng dark mode.....	47
<b>Hình 3.10:</b> Chức năng thống kê - Task chưa hoàn thành .....	48
<b>Hình 3.11:</b> Chức năng thống kê - Task phải hoàn thành .....	48
<b>Hình 3.12:</b> Chức năng thống kê - Task đã hoàn thành .....	49
<b>Hình 3.13:</b> Chức năng thêm dự án mới .....	49
<b>Hình 3.14:</b> Chức năng sửa dự án .....	50
<b>Hình 3.15:</b> Chức năng xoá dự án .....	50

<b>Hình 3.16:</b> Chức năng thêm nhiệm vụ .....	51
<b>Hình 3.17:</b> Chức năng sửa nhiệm vụ .....	52
<b>Hình 3.18:</b> Kết quả sau khi kiểm thử.....	53

## DANH SÁCH BẢNG BIỂU

<b>Bảng 1.8.1:</b> Khảo sát nghiệp vụ .....	16
<b>Bảng 2.3.1:</b> Đặc tả ca sử dụng quản lý project .....	25
<b>Bảng 2.3.2:</b> Đặc tả ca sử dụng quản lý task .....	27
<b>Bảng 2.3.3:</b> Đặc tả ca sử dụng tìm kiếm.....	29
<b>Bảng 2.3.4:</b> Đặc tả ca sử dụng thống kê .....	31



## LỜI NÓI ĐẦU

Trong cuộc sống hằng ngày, chúng ta thường đối mặt với nhiều tác vụ và công việc cần phải quản lý. Từ việc hoàn thành nhiệm vụ trong công việc, quản lý thời gian cho học tập, đến việc tổ chức cuộc sống cá nhân, chúng ta luôn cần một phương pháp hiệu quả để giữ cho mọi việc được gọn gàng và tiến triển suôn sẻ. Trong trạng thái bận rộn của cuộc sống hiện đại, việc quản lý tác vụ trở thành một thách thức không nhỏ.

Đồ án này tập trung vào xây dựng một Website Todoist - một ứng dụng quản lý tác vụ trực tuyến. Mục tiêu chính của đề tài này là thiết kế và phát triển một ứng dụng mang tính thực tế và ứng dụng rộng rãi, giúp người dùng có thể dễ dàng quản lý và theo dõi các công việc hàng ngày của mình.

Trong đồ án này, em sẽ tập trung vào việc phân tích, thiết kế và triển khai Website Todoist. Em sẽ nghiên cứu về các yêu cầu và chức năng cần thiết để xây dựng một hệ thống quản lý tác vụ mạnh mẽ và linh hoạt. Đồng thời, em cũng sẽ tìm hiểu về công nghệ và các phương pháp phát triển phù hợp để xây dựng một ứng dụng ổn định và tương thích trên nhiều nền tảng.

Em tin rằng việc xây dựng Website Todoist sẽ mang lại nhiều giá trị cho người dùng trong việc quản lý tác vụ và công việc hàng ngày của họ. Đồ án này cũng đóng góp vào sự phát triển và ứng dụng của công nghệ thông tin trong việc cải thiện hiệu suất làm việc cá nhân và tạo ra một môi trường làm việc hiệu quả.

Với mục tiêu trên, báo cáo sẽ được trình bày gồm các chương sau:

Chương 1: Tổng quan về đề tài

Chương 2: Phân tích và thiết kế hệ thống

Chương 3: Giao diện hệ thống và kiểm thử

Em hy vọng rằng báo cáo đồ án này sẽ cung cấp cái nhìn tổng quan về quá trình xây dựng và triển khai Website Todoist, đồng thời gợi mở thêm những khía cạnh và tiềm năng trong việc ứng dụng công nghệ thông tin vào việc quản lý tác vụ và công việc cá nhân.

Tuy nhiên, do thời gian thực hiện đồ án có hạn và kiến thức còn nhiều hạn chế, nên đề tài vẫn còn rất nhiều những thiếu sót, điểm bất cập và các vấn đề chưa thể xử lý triệt để. Em rất mong nhận được sự góp ý của quý thầy cô và các bạn để báo cáo hoàn thiện hơn. Em xin chân thành cảm ơn!

# CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

## 1. Đặt vấn đề

Đề án này nhằm tìm hiểu và phân tích về ngành công nghệ thông tin, một lĩnh vực đóng vai trò quan trọng và không thể thiếu trong thế giới hiện đại. Ngành công nghệ thông tin đã tạo ra một cuộc cách mạng công nghệ, ảnh hưởng đến mọi khía cạnh của cuộc sống, kinh tế và xã hội. Công nghệ thông tin không chỉ là một công cụ hỗ trợ, mà còn là một ngành nghề đầy thách thức và triển vọng với sự phát triển không ngừng.

Công nghệ thông tin tập trung vào nghiên cứu, phát triển và ứng dụng các công nghệ liên quan đến xử lý thông tin và truyền thông. Các lĩnh vực chính của công nghệ thông tin bao gồm khoa học máy tính, phần mềm, mạng và hệ thống thông tin. Đây là các lĩnh vực quan trọng để xây dựng và quản lý các hệ thống thông tin, phát triển ứng dụng, phân tích dữ liệu và bảo mật thông tin.

Cùng với đó, trong quá trình công nghiệp hoá, hiện đại hoá hiện nay thì việc quản lý công việc cũng như các nhiệm vụ cần làm đang thực sự cần thiết trong bối cảnh con người đang quay cuồng với cuộc sống bận rộn. Việc có quá nhiều nhiệm vụ trong thời gian gần và tương lai xa mà bản thân không thể nhớ hết thì việc ghi chép lại là hết sức cần thiết. Nhưng lại phát sinh thêm vấn đề là không phải lúc nào ta cũng có thể viết hết một mạch trình tự các công việc cần làm ra theo trật tự và việc quản lý tờ note cũng khó khăn nếu không cẩn thận. Do đó cần phải có một thứ để quản lý, lưu lại những nhiệm vụ cũng như công việc cho cá nhân, tập thể giúp cho người dùng dễ dàng, thuận tiện hơn trong những việc cần làm.

Hiện nay với đời sống hiện đại khiến con người càng gánh trên vai nhiều trách nhiệm, nhiều công việc hơn. Việc không thể hình thành nên danh sách việc cần làm sẽ khiến rơi vào tình trạng mất phương hướng, mất thời gian, mất công sức,... khiến cho mọi thứ diễn ra không được hiệu quả. Việc áp dụng công nghệ thông tin vào việc lưu trữ, quản lý nhiệm vụ, công việc sẽ giúp cho cá nhân, tổ chức có thể tránh được việc khiến mọi thứ diễn ra không hiệu quả, giúp cho con người thuận tiện hơn trong việc quản lý đời sống, thay vì sử dụng phương pháp thủ công thì có thể sử dụng phương pháp thông minh hơn để dễ dàng quản lý. Đồng thời, áp dụng công nghệ sẽ giúp dễ dàng quản lý được việc lưu lại, phân công những nhiệm vụ cho từng công việc cụ thể.

Trong khuôn khổ của đề án này sẽ tập trung nghiên cứu về website quản lý tác vụ. Một trong những website quản lý tác vụ phổ biến và được sử dụng rộng rãi là Todoist. Todoist là một ứng dụng và trang web quản lý công việc được phát triển để giúp người dùng tổ chức công việc, lập kế hoạch và theo dõi tiến độ công việc của mình.

Todoist cung cấp một giao diện thân thiện và dễ sử dụng, cho phép người dùng tạo danh sách các nhiệm vụ, thiết lập hạn chế thời gian, gắn kết các tập tin và ghi chú, và thậm

chỉ chia sẻ công việc với nhóm làm việc. Nó cung cấp các tính năng linh hoạt như nhắc nhở, ưu tiên, lịch trình, và theo dõi tiến độ, giúp người dùng quản lý công việc một cách hiệu quả.

Tổng quan, đồ án này sẽ khám phá sâu hơn về ngành công nghệ thông tin và tập trung vào nghiên cứu và phân tích về website quản lý tác vụ, với Todoist là điển hình. Hy vọng rằng kết quả của đồ án sẽ mang lại những góc nhìn mới và đóng góp vào việc cải thiện trải nghiệm người dùng và hiệu suất công việc trong lĩnh vực quản lý tác vụ.

## 2. Lý do chọn đề tài

Nhận thức được lợi ích của việc áp dụng công nghệ thông tin và những khó khăn trong công tác quản lý, lưu trữ những công việc, nhiệm vụ,..., việc chọn đề tài Todoist cho đồ án là do sự phổ biến, tính đa dạng và tiềm năng nghiên cứu và cải tiến của nó. Và vì nó mang lại nhiều lợi ích và giải quyết một trong những vấn đề quan trọng nhất trong cuộc sống hiện đại - quản lý thời gian và công việc một cách hiệu quả. Với niềm tin rằng việc nghiên cứu và đề xuất các cải tiến cho Todoist sẽ mang lại giá trị đáng kể cho người dùng và góp phần vào phát triển của lĩnh vực quản lý tác vụ trong ngành công nghệ thông tin. Ngoài ra Todoist, cũng phù hợp với mục tiêu đồ án, nhằm nghiên cứu và phân tích các phương pháp quản lý công việc hiệu quả

Đối với các cá nhân và tổ chức, quản lý thời gian và công việc là yếu tố quan trọng để đạt được sự thành công và hiệu quả. Với Todoist, người dùng có thể dễ dàng tạo ra các danh sách công việc, phân loại chúng theo ưu tiên, gắn kết các mục tiêu và thiết lập hạn ché thời gian. Đặc biệt, Todoist cung cấp tính năng nhắc nhở và thông báo, giúp người dùng không bỏ sót công việc quan trọng và tuân thủ kế hoạch đã đề ra.

Thông qua việc nghiên cứu và phân tích Todoist, hy vọng đưa ra các đề xuất cải tiến và mở rộng, nhằm nâng cao trải nghiệm người dùng, tối ưu hóa hiệu suất và đáp ứng nhu cầu quản lý công việc đa dạng. Đồng thời, việc triển khai và đánh giá các cải tiến được đề xuất sẽ cung cấp một cái nhìn sâu hơn về khả năng áp dụng và hiệu quả của các phương pháp quản lý công việc trong Todoist.

## 3. Mục tiêu, nhiệm vụ đề tài

### - Mục tiêu:

- Phân tích đề tài, quy trình nghiệp vụ, thiết kế các chức năng, cơ sở dữ liệu để hoàn thành ứng dụng Todoist - Quản lý nhiệm vụ, đáp ứng nhu cầu của người dùng trong việc quản lý công việc hằng ngày, từ cá nhân đến doanh nghiệp, tạo ra một nền tảng linh hoạt trực quan và dễ sử dụng, giúp người dùng tối ưu hoá thời gian và năng suất, đồng thời tăng cường sự tổ chức
- Đề tài đảm bảo có chức năng:
  - Quản lý project gồm thêm, sửa, xoá project

- Quản lý task gồm thêm, sửa, đánh dấu hoàn thành task
- Phân loại task theo thời gian và project
- Chế độ dark mode
- Chức năng tìm kiếm task
- Thống kê những task chưa hoàn thành, phải hoàn thành, đã hoàn thành
- Sử dụng React Testing Library để kiểm thử các thành phần React
- Nhiệm vụ: Tìm hiểu các công nghệ liên quan, khảo sát nghiệp vụ của đề tài để xây dựng hệ thống, phân tích yêu cầu, thiết kế giao diện người dùng, xây dựng cơ sở dữ liệu, phát triển tính năng và chức năng, tối ưu hoá hiệu suất, bảo mật và bảo vệ dữ liệu, kiểm thử và sửa lỗi.

#### 4. Phương pháp nghiên cứu

- Phân tích chức năng: Đã phân tích các chức năng và tính năng của Todoist để hiểu cách thức hoạt động và lợi ích của chúng trong quản lý công việc. Cũng đã xem xét những khía cạnh đơn giản, linh hoạt
- Khảo sát người dùng: Đã tiến hành khảo sát người dùng để thu thập thông tin về trải nghiệm và ý kiến của người dùng về Todoist. Khảo sát này giúp đánh giá hiệu suất và hài lòng của người dùng đối với ứng dụng.
- Phân tích so sánh: Đã tiến hành phân tích so sánh Todoist với các ứng dụng quản lý công việc khác để đánh giá các ưu điểm và nhược điểm của Todoist. Điều này giúp xác định được sự đặc biệt và giá trị của Todoist so với các sản phẩm tương tự.

#### 5. Đối tượng nghiên cứu

- Đối tượng
  - Trong đề án này, tập trung nghiên cứu về đối tượng sử dụng và hướng dẫn sử dụng ứng dụng Todoist. Todoist là một ứng dụng quản lý công việc và nhắc nhở hàng đầu, được sử dụng rộng rãi trên nhiều nền tảng. Đối tượng nghiên cứu bao gồm cả người dùng cá nhân
  - Người dùng cá nhân: Todoist cung cấp một giao diện đơn giản và trực quan cho người dùng cá nhân để quản lý và tổ chức công việc cá nhân. Đối tượng nghiên cứu bao gồm những người sử dụng Todoist để quản lý công việc hàng ngày, lập kế hoạch dự án và đặt nhắc nhở để duy trì sự tổ chức và hiệu suất làm việc.

#### 6. Phạm vi nghiên cứu

- Trong đề án này, tập trung vào nghiên cứu và phân tích ứng dụng Todoist - một ứng dụng quản lý công việc và nhắc nhở hàng đầu. Xác định phạm vi nghiên cứu của dự án, nhằm hiểu rõ hơn về khả năng và giới hạn của Todoist trong việc quản lý công việc cá nhân

- Quản lý công việc cá nhân:
  - Tạo, chỉnh sửa và xóa công việc: Nghiên cứu cách người dùng có thể tạo, chỉnh sửa và xóa các công việc trong Todoist. Điều này bao gồm việc đặt ngày hết hạn, ưu tiên và phân loại công việc để tăng cường sự tổ chức và hiệu suất làm việc cá nhân.
  - Đặt nhắc nhở và theo dõi tiến độ: Nghiên cứu cách sử dụng tính năng nhắc nhở và theo dõi tiến độ công việc trong Todoist để giúp người dùng duy trì sự tổ chức và hoàn thành công việc một cách hiệu quả.
- Ưu điểm của phạm vi nghiên cứu
  - Tính ứng dụng rộng rãi: Todoist là một ứng dụng quản lý công việc phổ biến và được sử dụng trên nhiều nền tảng khác nhau. Vì vậy, phạm vi nghiên cứu cung cấp lợi ích cho nhiều người dùng cá nhân
  - Tính linh hoạt và dễ sử dụng: Todoist cung cấp giao diện đơn giản và dễ sử dụng, cho phép người dùng dễ dàng tạo, quản lý và theo dõi công việc. Phạm vi nghiên cứu của đã tập trung vào cách sử dụng các tính năng và công cụ này một cách hiệu quả và tiện lợi.

## 7. Tổng quan về công nghệ

### 7.1. React JS và Firebase<sup>[1]</sup>

Sử dụng ReactJS và Firebase để phát triển đề tài Todoist mang lại nhiều lợi ích và tiềm năng phát triển ứng dụng hiệu quả. ReactJS là một thư viện JavaScript phổ biến và mạnh mẽ cho việc xây dựng giao diện người dùng, trong khi Firebase là một nền tảng đám mây đa chức năng cung cấp các dịch vụ cơ sở dữ liệu và xử lý yêu cầu từ phía máy chủ.

Kết hợp giữa ReactJS và Firebase, ta có thể xây dựng một ứng dụng Todoist mạnh mẽ với giao diện người dùng phản ứng, đồng bộ dữ liệu thời gian thực và tích hợp xác thực người dùng. ReactJS giúp tạo ra giao diện người dùng linh hoạt và dễ quản lý, trong khi Firebase cung cấp các dịch vụ backend mạnh mẽ để lưu trữ và đồng bộ dữ liệu. Sự kết hợp giữa hai công nghệ này tạo ra một sự phát triển linh hoạt và hiệu quả cho đề tài Todoist, mang lại trải nghiệm người dùng tốt và giúp người dùng quản lý công việc một cách hiệu quả.

Với ReactJS, ta có thể tạo ra các thành phần UI linh hoạt như danh sách công việc, bảng điều khiển và các tính năng tương tác. Cấu trúc dựa trên thành phần của ReactJS giúp tách biệt logic và hiển thị, giúp việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn.

Firebase cung cấp một hệ thống cơ sở dữ liệu thời gian thực, cho phép dữ liệu trong ứng dụng Todoist được đồng bộ ngay lập tức giữa các người dùng và các thiết bị khác nhau. Điều này đảm bảo rằng các công việc, danh sách và trạng thái được cập nhật một

cách tức thì và người dùng luôn có được phiên bản mới nhất của dữ liệu. Firebase cũng cung cấp tích hợp xác thực người dùng, cho phép người dùng đăng nhập, đăng ký tài khoản và quản lý thông tin cá nhân một cách an toàn.

Với sự kết hợp của ReactJS và Firebase, ứng dụng Todoist sẽ mang lại trải nghiệm tốt cho người dùng. Người dùng có thể tạo, chỉnh sửa và xóa công việc, thêm ghi chú, đặt lịch nhắc nhở, và theo dõi tiến độ công việc một cách dễ dàng. Dữ liệu sẽ được lưu trữ và đồng bộ ngay lập tức trên nền tảng Firebase, đảm bảo rằng mọi thay đổi sẽ được cập nhật ngay tức thì trên các thiết bị và tài khoản người dùng. Điều này giúp người dùng có thể truy cập vào danh sách công việc của mình từ bất kỳ đâu và bất kỳ thiết bị nào mà họ sử dụng.

Tổng kết lại, sử dụng ReactJS và Firebase để phát triển đề tài Todoist mang lại nhiều lợi ích cho người dùng. ReactJS cung cấp khả năng xây dựng giao diện người dùng linh hoạt và dễ quản lý, trong khi Firebase đảm bảo việc lưu trữ, đồng bộ dữ liệu và tích hợp xác thực người dùng. Sự kết hợp này tạo nên một ứng dụng Todoist hiệu quả, giúp người dùng quản lý công việc một cách thuận tiện, đồng thời đảm bảo tính bảo mật và quyền riêng tư của thông tin cá nhân.

## 7.2. React Testing Library<sup>[2]</sup>

Việc sử dụng React Testing Library trong ứng dụng Todoist mang lại nhiều ý nghĩa và lợi ích đáng kể trong việc đảm bảo chất lượng và đáng tin cậy của ứng dụng. React Testing Library là một thư viện kiểm thử phổ biến và mạnh mẽ cho ứng dụng React, giúp thực hiện các bài kiểm tra tự động về giao diện người dùng và tương tác với thành phần React.

Một trong những ý nghĩa quan trọng của việc sử dụng React Testing Library trong ứng dụng Todoist là đảm bảo tính ổn định và đúng đắn của giao diện người dùng. Kiểm thử giao diện giúp phát hiện và ngăn chặn các lỗi hiển thị, truy cập và tương tác người dùng có thể xảy ra trong quá trình sử dụng ứng dụng. Điều này đảm bảo rằng người dùng sẽ có trải nghiệm mượt mà và không gặp các vấn đề không mong muốn.

Thứ hai, việc sử dụng React Testing Library giúp đảm bảo tính nhất quán và đúng đắn của logic xử lý dữ liệu trong ứng dụng. Bằng cách thực hiện các bài kiểm thử cho các thành phần và hành vi của ứng dụng, chúng ta có thể xác minh rằng các hàm và quy trình xử lý dữ liệu hoạt động đúng và trả về kết quả chính xác. Điều này giúp đảm bảo tính đáng tin cậy và chính xác của các tính năng và chức năng trong Todoist.

Thêm vào đó, việc sử dụng React Testing Library cũng tạo ra một tập thử nghiệm tự động cho ứng dụng Todoist, giúp tiết kiệm thời gian và công sức trong việc kiểm tra và xác minh chất lượng của ứng dụng. Khi có sự thay đổi trong mã nguồn hoặc tính năng, các bài kiểm thử tự động giúp đảm bảo rằng không có lỗi nào được tạo ra và ứng dụng vẫn hoạt động như mong đợi.

## 8. Khảo sát

### 8.1. Khảo sát hiện trạng

- Ứng dụng to-do list đơn giản: Đây là những ứng dụng đơn giản như Todoist<sup>[3]</sup> và MicrosoftTodo<sup>[4]</sup>, cho phép người dùng tạo danh sách công việc và đặt nhắc nhở.
- Ứng dụng quản lý dự án: Đây là các ứng dụng như Trello và Asana, cho phép người dùng tạo và quản lý dự án, phân công công việc và theo dõi tiến độ.
- Kết quả khảo sát người dùng:
  - Tính năng quan trọng nhất: Đa số người dùng cho biết tính năng quan trọng nhất của các ứng dụng này là khả năng tạo danh sách công việc, đặt nhắc nhở và phân loại công việc theo mức độ ưu tiên.
  - Điểm mạnh: Người dùng đánh giá cao tính đơn giản và dễ sử dụng của các ứng dụng quản lý tác vụ. Họ cũng đánh giá cao tính linh hoạt và tích hợp của các ứng dụng liên quan.
  - Điểm yếu: Một số người dùng gặp khó khăn trong việc đồng bộ hóa dữ liệu giữa các thiết bị và gặp vấn đề về hiệu suất khi làm việc với các ứng dụng có nhiều tính năng phức tạp.

### 8.2. Mục tiêu khảo sát

- Địa điểm khảo sát: Trực tuyến với cộng đồng sử dụng Todoist
- Quá trình khảo sát với các chủ đề:
  - Khảo sát về thực trạng sử dụng
  - Những bất cập trong việc quản lý tác vụ

### 8.3.Kết quả khảo sát thực tế

**Bảng 1.8.1:** Khảo sát nghiệp vụ

PHIẾU PHỎNG VẤN	
<p>Người hỏi: Trần Ngọc Lâm Ngày: 25/4/2023</p>	<p>Dự án: Khảo sát về công việc quản lý tác vụ, nhiệm vụ Người trả lời: Vũ Chí Bảo Vị trí: Thực tập sinh tại Samsung</p>
Câu hỏi	Câu trả lời
Bạn đã dùng đến những cách nào để có thể lưu trữ và quản lý tác vụ, nhiệm vụ mà không dùng đến ứng dụng quản lý?	Tôi đã dùng đến một vài cách trong đó cách thông dụng nhất mà mọi người đều dùng đó là dùng giấy ghi chú, sổ tay, danh sách hoặc bảng viết lại và dán vào nơi bản thân chú ý nhất, dễ nhìn thấy nhất. Mỗi một project thì sẽ dùng một màu giấy ghi chú khác nhau để bản thân dễ kiểm soát và nhanh chóng nhận ra. Và có thể viết ra đó danh sách công việc cụ thể và đánh dấu khi hoàn thành, có thể sắp xếp theo mức độ ưu tiên và thứ tự ưu tiên để theo dõi tiến độ của mình
Việc sử dụng giấy ghi chú như vậy liệu bản thân bạn có khó kiểm soát được những tác vụ cần làm hay không?	Điều khó khăn khi dùng giấy ghi chú đó là sẽ khiến không gian xung quanh tôi trông có vẻ lộn xộn và rối mắt bởi những mẫu giấy ghi chú dán khắp mọi nơi. Điều đó khiến tôi khó theo dõi tiến độ công việc và còn do nó không có sự linh hoạt trong việc thay đổi, sắp xếp hoặc di chuyển tác vụ một cách linh hoạt. Tiếp đó, nó thiếu tính di động, bạn cần mang theo giấy ghi chú và dụng cụ viết khi di chuyển để tiếp tục quản lý công việc. Hơn nữa nếu chẳng may vì một lý do nào đó khiến mẫu giấy ghi chú bị mất hoặc làm cho không thể nhận diện được nữa thì thực sự sẽ là một vấn đề nan giải
Liệu việc sử dụng phần mềm ứng dụng để quản lý những nhiệm vụ, tác vụ có khiến cho người dùng dần từ bỏ việc sử dụng giấy ghi chú để chuyển sang điện tử hoá những ghi chú hay không?	Rất có khả năng! Việc sử dụng những tờ giấy ghi chú suy cho cùng thì cũng rất lãng phí tài nguyên môi trường và cũng làm không gian xung quanh không được gọn gàng cho lắm. Việc sử dụng ứng dụng phần mềm quản lý sẽ tăng độ tiện lợi và dễ dàng quản lý, kiểm soát thêm sửa xóa tác vụ cũng như dự án hơn. Nó còn giúp chúng ta đồng bộ hoá dữ liệu trên nhiều thiết bị và nền tảng khác nhau. Điều này giúp cho ta truy cập và cập nhật bất kỳ nhiệm vụ nào và từ bất kỳ đâu với bất kỳ thiết bị nào mà được sử dụng. Điện tử hoá



	<p>ghi chú giúp giảm thiểu rủi ro mất mát thông tin và tăng tính khả dụng của dữ liệu. Hơn nữa nó có thể linh hoạt theo nhu cầu cá nhân. Người dùng có thể thêm tính năng bổ sung như phân loại nhiệm vụ, ghi chú chi tiết,... Điều này mang lại sự linh hoạt và tùy chỉnh cao hơn so với giấy ghi chú truyền thống. Tuy nhiên, việc sử dụng giấy ghi chú vẫn có những ưu điểm riêng, như cảm giác vật lý và sự linh hoạt trong việc tạo và sắp xếp ghi chú. Quyết định sử dụng phương pháp nào phụ thuộc vào sở thích và nhu cầu của mỗi người dùng.</p>
<p>Bạn đánh giá về hiệu quả sử dụng phần mềm quản lý dự án qua những phương pháp nào?</p>	<p>Để đánh giá mức độ hiệu quả trong việc sử dụng phần mềm quản lý công việc, dự án tôi quan tâm đến 4 thứ quan trọng:</p> <ul style="list-style-type: none"> <li>- Đánh giá hiệu quả qua bình quân số lượng dự án: giúp đánh giá năng suất và mức độ tương thích của phần mềm và tính chất công việc</li> <li>- Đánh giá hiệu quả qua số lượng thời gian/ ngân sách được tối ưu trên từng dự án: đánh giá mức độ hiệu quả, tối ưu chi phí của phần mềm trong thực tế hoạt động.</li> <li>- Đánh giá hiệu quả qua mức độ gắn kết: là một thông số đo lường gián tiếp. Thường, các phần mềm quản lý dự án sẽ đóng luôn vai trò như một công cụ công tác và hỗ trợ tiến hành công việc suôn sẻ, qua đó nâng cao tính gắn kết</li> <li>- Đánh giá hiệu quả khả năng tích hợp và tự động hóa của phần mềm: kiểm tra khả năng linh hoạt ứng dụng của công cụ, nhằm đánh giá mức độ tương thích của nó với hệ thống các phần mềm và quy trình làm việc hiện tại</li> </ul>
<p>Bạn đã thực sử dụng phần mềm quản lý công việc và dự án của mình hiệu quả chưa?</p>	<p>Một hệ thống quản lý công việc và dự án yếu kém là “con sâu bệnh” ăn mòn từ gốc rễ của nhiều doanh nghiệp. Tình trạng xảy ra do việc đo lường những tác động của sự thiếu hiệu quả trong các hoạt động thường nhật gần như là bất khả thi. Trong thực tế, nhiều doanh nghiệp đã tìm ra được lời giải cho bài toán này: đặt trọn niềm tin vào một phần mềm quản lý công việc và dự án cấp tiến</p>
<p>Bạn có đánh giá về Todoist như thế nào so với những ứng dụng quản lý công việc khác mà trước đây bạn đã</p>	<p>Todoist đã nhận được đánh giá tích cực từ cộng đồng người dùng và những người sử dụng đã tận hưởng các tính năng và giao diện người dùng dễ sử dụng của nó.</p>

từng trải nghiệm?	Đánh giá của mỗi người về Todoist có thể phụ thuộc vào nhu cầu, sở thích cá nhân và cách họ quản lý công việc của mình. Với Todoist thì việc phân loại tác vụ luôn diễn ra đồng thời, phục vụ cho mục đích cá nhân thì luôn được tối ưu tiên
Với mục đích sử dụng cho cá nhân bạn mong muốn trình quản lý tác vụ sẽ đáp ứng yêu cầu của bạn như nào?	Đương nhiên việc thêm sửa xóa dự án cũng như tác vụ, nhiệm vụ là những điều cần bản mà tôi mong muốn rồi. Thêm vào đó có thể giúp phân loại ra các dự án để có thể dễ dàng kiểm soát và nhận diện cũng là một chức năng hay. Thêm vào đó có thể giúp được chúng ta kiểm tra được deadline của nhiệm vụ, nhắc nhở khi gần đến hạn thì thực sự sẽ giúp ích rất nhiều
Todoist đã có sự ảnh hưởng đến cuộc sống và công việc hàng ngày của bạn như thế nào? Bạn có thể chia sẻ một trải nghiệm cụ thể với em không?	Todoist giúp người dùng tổ chức, lập kế hoạch và quản lý các nhiệm vụ một cách hiệu quả. Với khả năng tạo danh sách, đặt hạn chót và ưu tiên, người dùng có thể sắp xếp công việc một cách rõ ràng và dễ dàng theo dõi tiến độ. Người dùng còn có thể tập trung vào các nhiệm vụ quan trọng giúp tăng cường năng suất làm việc và hoàn thành công việc một cách hiệu quả hơn. Chức năng nhắc nhở và lịch trình cũng giúp người dùng không bỏ sót nhiệm vụ quan trọng. Tuy nhiên với người thì có một trải nghiệm khác nhau thể nên sẽ có những cách khác nhau để sử dụng

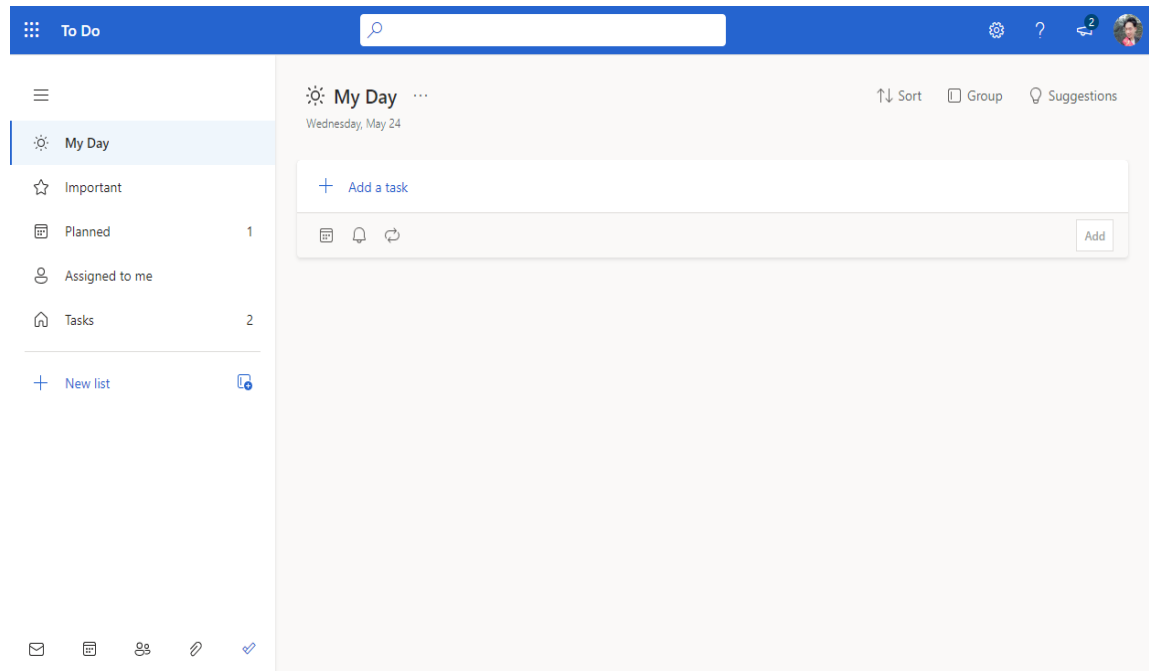
## 9. Kết quả dự kiến đạt được

- Tính năng đa dạng: Dự kiến cung cấp một loạt các tính năng quan trọng để người dùng có thể quản lý công việc một cách linh hoạt. Điều này bao gồm tạo danh sách công việc, thiết lập nhắc nhở, phân loại công việc theo thẻ và nhãn, và chia sẻ công việc với người khác.
  - Giao diện người dùng thân thiện: Đặt mục tiêu tạo ra một giao diện người dùng đơn giản, dễ sử dụng và thân thiện với người dùng. Người dùng sẽ có trải nghiệm mượt mà và dễ dàng tương tác với ứng dụng.
  - Hiệu suất và đáng tin cậy: Cam kết đạt được một mức độ hiệu suất cao và đáng tin cậy trong việc xử lý dữ liệu và đồng bộ hóa giữa các thiết bị. Điều này đảm bảo rằng người dùng có thể truy cập và cập nhật công việc của mình một cách nhanh chóng và mượt mà.
- ⇒ Kết quả dự kiến của ứng dụng Todoist đóng vai trò quan trọng trong việc cung cấp một công cụ quản lý tác vụ toàn diện và hiệu quả cho người dùng. Mong rằng ứng dụng sẽ giúp người dùng tổ chức công việc, quản lý thời gian và nâng cao năng suất làm việc. Đồng thời, ứng dụng cũng tạo ra trải nghiệm người dùng tốt, thân thiện và tiện lợi.

## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 1. Phân tích hệ thống

#### 1.1. Tham khảo một số website tương tự



Hình 2.1.1: Website tương tự

- Sự tương đồng giữa hai ứng dụng:
  - Tính năng quản lý tác vụ: Đã áp dụng các tính năng quản lý tác vụ quan trọng từ Microsoft Todo<sup>[4]</sup>, bao gồm tạo danh sách công việc, đặt thời hạn và ghi chú. Điều này giúp xây dựng một ứng dụng Todoist đáp ứng được nhu cầu quản lý công việc của người dùng.
  - Giao diện người dùng: Đã áp dụng các yếu tố thiết kế giao diện từ Microsoft Todo<sup>[4]</sup>, bao gồm bố cục rõ ràng, màu sắc tươi sáng và biểu tượng thân thiện. Điều này giúp tạo ra một giao diện người dùng thân thiện, dễ sử dụng và hấp dẫn cho người dùng của ứng dụng Todoist<sup>[3]</sup>.
- Sự khác biệt giữa hai ứng dụng:
  - Giao diện: Cả hai đều có giao diện đơn giản, gọn gàng nhưng MicrosoftTo tích hợp các thành phần của hệ sinh thái Microsoft như Office 365, Outlook và Microsoft Teams để tạo liên kết giữa công việc và các ứng dụng khác.
  - Tính năng: Microsoft ToDo cung cấp các tính năng tương tự như ToDoist, nhưng có tích hợp tốt hơn với các ứng dụng Microsoft như Outlook và Teams. Người dùng có thể tạo nhiệm vụ từ email trong Outlook và tạo công việc từ các cuộc họp hoặc trò chuyện trong Teams.

- Đối tượng: Todoist hướng đến các cá nhân và nhóm làm việc tổ chức cá nhân, cung cấp một giao diện đơn giản và linh hoạt để quản lý công việc. Microsoft ToDo phù hợp với các tổ chức và doanh nghiệp đã sử dụng các dịch vụ Microsoft khác, giúp tạo liên kết và tăng cường hiệu suất làm việc trong môi trường Microsoft
- ⇒ Tóm lại, cả Todoist và Microsoft ToDo đều cung cấp các tính năng quản lý công việc và lập kế hoạch. Tuy nhiên, Todoist tập trung vào trải nghiệm đơn giản và linh hoạt, trong khi Microsoft ToDo tích hợp tốt với hệ sinh thái Microsoft. Sự lựa chọn giữa hai ứng dụng phụ thuộc vào yêu cầu và môi trường làm việc của người dùng.

## 1.2.Mô tả

Todoist<sup>[3]</sup> là một ứng dụng quản lý tác vụ trực tuyến, cho phép người dùng tạo, theo dõi và quản lý danh sách công việc cá nhân một cách hiệu quả. Với giao diện đơn giản và dễ sử dụng, Todoist<sup>[3]</sup> cung cấp các công cụ và tính năng linh hoạt để người dùng có thể lập kế hoạch, sắp xếp và theo dõi tiến độ công việc của mình.

Todoist<sup>[3]</sup> cho phép người dùng tạo danh sách công việc. Người dùng có thể thiết lập ngày hết hạn và đánh dấu công việc đã hoàn thành. Một trong những ưu điểm nổi bật của Todoist<sup>[3]</sup> là tính linh hoạt và tích hợp. Todoist<sup>[3]</sup> có sẵn trên nhiều nền tảng, bao gồm web, di động và máy tính để bàn, giúp người dùng truy cập và quản lý công việc từ bất kỳ thiết bị nào và ở bất kỳ địa điểm nào.

Trên thị trường, Todoist<sup>[3]</sup> đã trở thành một trong những ứng dụng quản lý tác vụ phổ biến nhất và được tin dùng bởi hàng triệu người trên toàn thế giới. Tính linh hoạt, đơn giản và tích hợp của Todoist<sup>[3]</sup> đã giúp người dùng nâng cao năng suất làm việc, quản lý thời gian và tổ chức công việc một cách hiệu quả.

Ngoài Todoist<sup>[3]</sup>, còn có nhiều ứng dụng quản lý tác vụ khác như Asana, Trello, Microsoft ToDo, Notion, và nhiều hơn nữa. Các ứng dụng này cung cấp các tính năng và công cụ đa dạng để người dùng quản lý công việc, dự án và tương tác với các thành viên khác trong nhóm làm việc.

Trên cơ sở đó, việc chọn một ứng dụng quản lý tác vụ phù hợp sẽ phụ thuộc vào nhu cầu và yêu cầu cụ thể của từng người dùng hay tổ chức. Việc tìm hiểu và áp dụng ứng dụng quản lý tác vụ phù hợp có thể giúp cải thiện sự tổ chức, quản lý và hiệu suất làm việc, góp phần đóng vai trò quan trọng trong thành công của cá nhân và tổ chức.

## 1.3.Tác nhân tham gia hệ thống

- Tác nhân tham gia là những thành phần quan trọng trong hệ thống, đóng vai trò trong việc thực hiện các hoạt động và tương tác với người dùng

- Người dùng: Người dùng là tác nhân chính trong Todoist. Họ có thể tạo, chỉnh sửa và quản lý danh sách công việc, đặt các thời hạn, đánh dấu công việc hoàn thành và thực hiện các tương tác khác trong ứng dụng. Tác nhân này là người sử dụng cuối cùng của hệ thống và ảnh hưởng trực tiếp đến trải nghiệm và hiệu suất của Todoist.
- Hệ thống quản lý công việc: Đây là một tác nhân quan trọng trong Todoist, có trách nhiệm lưu trữ, quản lý và xử lý các công việc và thông tin liên quan. Hệ thống quản lý công việc xử lý các yêu cầu từ người dùng, đảm bảo tính toàn vẹn và sẵn sàng của dữ liệu, và thực hiện các hoạt động như đặt thời hạn, xếp hạng và đánh dấu công việc đã hoàn thành.
- Cơ sở dữ liệu: Cơ sở dữ liệu là một tác nhân quan trọng trong Todoist, chịu trách nhiệm lưu trữ và quản lý các dữ liệu liên quan đến danh sách công việc, người dùng, cài đặt và các thông tin khác. Cơ sở dữ liệu đảm bảo tính toàn vẹn và độ tin cậy của dữ liệu, cho phép truy xuất nhanh chóng và xử lý hiệu quả các hoạt động của Todoist.

#### **1.4. Yêu cầu chung và chức năng của hệ thống**

- Hệ thống Todoist nhằm giúp người dùng tổ chức và quản lý công việc một cách hiệu quả, đồng thời cung cấp các tính năng và chức năng giúp tăng cường năng suất làm việc
  - Quản lý công việc: Hệ thống Todoist cho phép người dùng tạo, chỉnh sửa và xóa các công việc. Người dùng có thể đặt thời hạn, đánh dấu công việc đã hoàn thành và gán nhãn để phân loại công việc theo nhóm.
  - Lập kế hoạch và lịch trình: Hệ thống Todoist cho phép người dùng lập kế hoạch và lịch trình công việc theo ngày, tuần, tháng hoặc quý. Người dùng có thể thiết lập nhắc nhở để nhận thông báo về các công việc sắp tới và đảm bảo không bỏ sót bất kỳ deadline nào.
  - Giao diện đơn giản và trực quan: Hệ thống Todoist thiết kế giao diện người dùng đơn giản và trực quan, giúp người dùng dễ dàng sử dụng và điều hướng trong ứng dụng. Giao diện thân thiện và trực quan giúp người dùng tiết kiệm thời gian và tăng cường trải nghiệm người dùng.
  - Thống kê và báo cáo: Todoist cung cấp các tính năng thống kê và báo cáo, giúp người dùng theo dõi tiến độ công việc, hiệu suất làm việc và phân tích dữ liệu để cải thiện năng suất làm việc.
- Nhiệm vụ: Quản lý project cũng như task, theo dõi tiến trình nhiệm vụ:
  - Thêm, sửa, xóa project
  - Quản lý task trong từng project bao gồm việc thêm, sửa, đánh dấu đã hoàn thành task

- Phân loại và hiển thị deadline cho từng task
- Thống kê những nhiệm vụ chưa hoàn thành, phải hoàn thành và đã hoàn thành
- Tìm kiếm task để dễ dàng đánh dấu hoàn thành những nhiệm vụ

### 1.5. Phân tích bài toán

Việc phân tích bài toán là một bước quan trọng trong quá trình thiết kế và phát triển một ứng dụng, giúp xác định rõ ràng các yêu cầu và chức năng cần có để đáp ứng nhu cầu của người dùng.

Mục tiêu chính của Todoist là cung cấp một nền tảng quản lý tác vụ đơn giản và hiệu quả cho người dùng. Để đạt được mục tiêu này, Todoist phải đáp ứng các yêu cầu và chức năng sau:

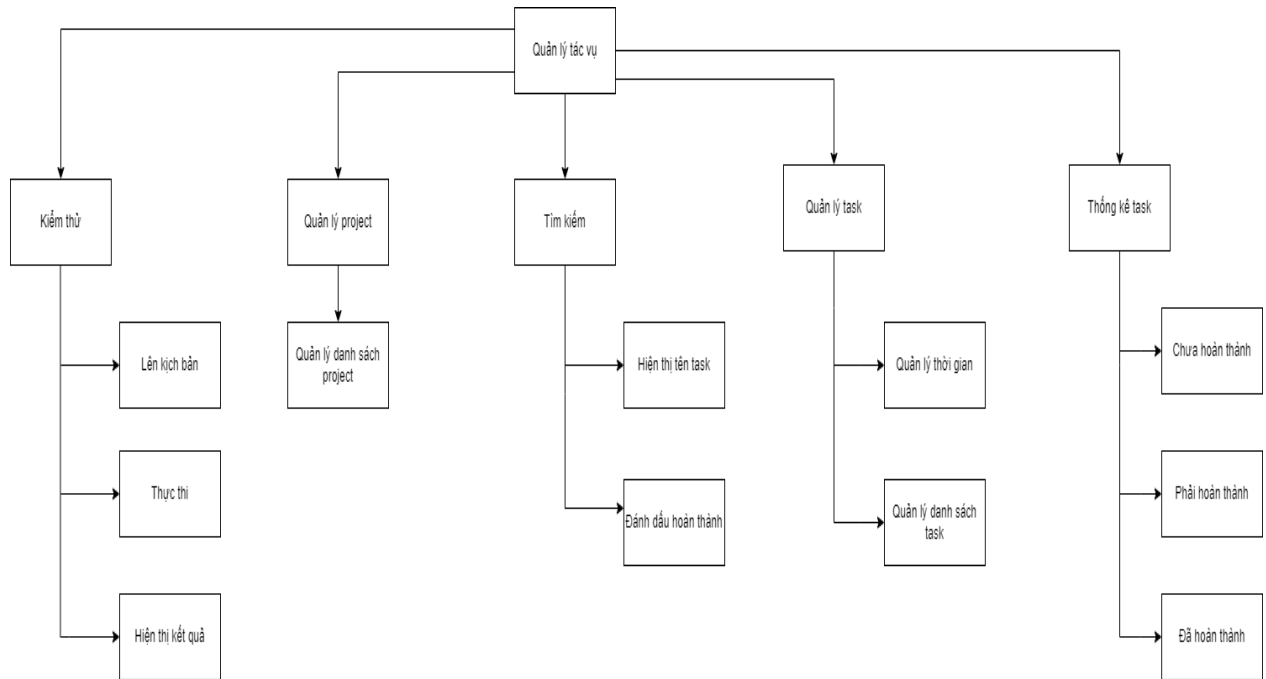
- Tạo danh sách công việc: Người dùng có thể tạo danh sách các công việc cá nhân và sắp xếp chúng theo các danh mục hoặc nhãn khác nhau để dễ dàng quản lý.
- Gán ngày hết hạn: Người dùng có thể thiết lập ngày hết hạn cho mỗi công việc để xác định mức độ quan trọng và thời gian hoàn thành.
- Phân loại công việc: Todoist cho phép người dùng gán nhãn và phân loại công việc theo các tiêu chí như dự án, mục tiêu. Điều này giúp người dùng tìm kiếm và lọc công việc một cách nhanh chóng và thuận tiện.
- Tìm kiếm công việc: Giúp người dùng có thể tìm kiếm một cách thuận lợi những công việc mình đã tạo theo tên và có thể đánh dấu hoàn thành ngay tại đây
- Thống kê: Đưa ra cho người dùng một cách tổng quan nhất về số lượng nhiệm vụ cũng như liệt kê các nhiệm vụ chưa hoàn thành, phải hoàn thành và đã hoàn thành để có thể kiểm soát cũng như đánh dấu lại những nhiệm vụ của mình

Qua quá trình phân tích bài toán, đã nhận thấy rằng Todoist được thiết kế để đáp ứng nhu cầu quản lý tác vụ cá nhân và công việc của người dùng một cách linh hoạt, tiện lợi và hiệu quả. Các chức năng và tính năng của Todoist được tối ưu hóa để giúp người dùng có thể tổ chức và quản lý công việc một cách thông minh và tiết kiệm thời gian.

## 2. Sơ đồ phân rã chức năng(BFD)

Xác định chức năng nghiệp vụ là bước đầu tiên của việc phân tích hệ thống. Để phân tích yêu cầu thông tin của tổ chức ta phải biết được tổ chức đó thực hiện những nhiệm vụ, chức năng gì. Từ đó, tìm ra các dữ liệu, các thông tin được sử dụng và tạo ra trong các chức năng. Đồng thời, cũng phải tìm ra những hạn chế, mối ràng buộc đặt lên các chức năng đó.

Mô hình phân rã chức năng là công cụ biểu diễn việc phân rã có thứ bậc đơn giản các công việc cần thực hiện. Mỗi công việc được chia ra làm các công việc con, số mức chia ra phụ thuộc kích cỡ và độ phức tạp của hệ thống

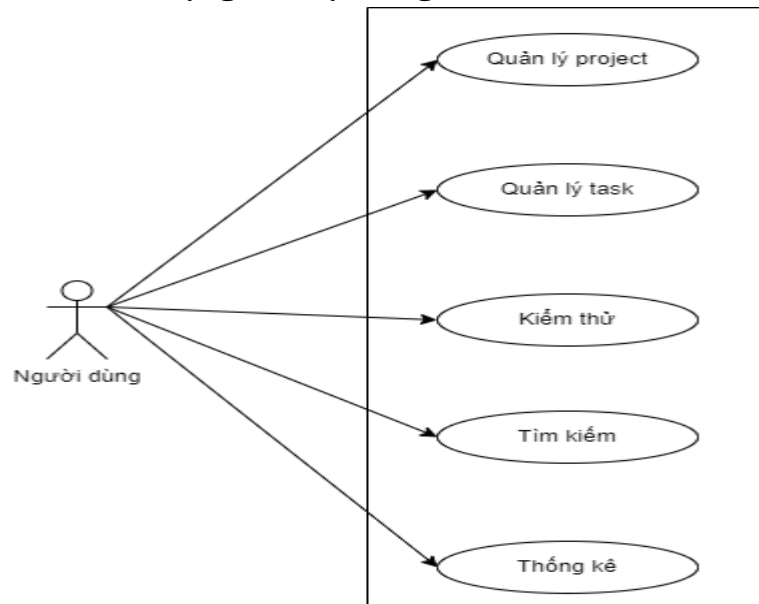


**Hình 2.2.1:** Sơ đồ phân rã chức năng

### 3. Sơ đồ thiết kế bằng UML

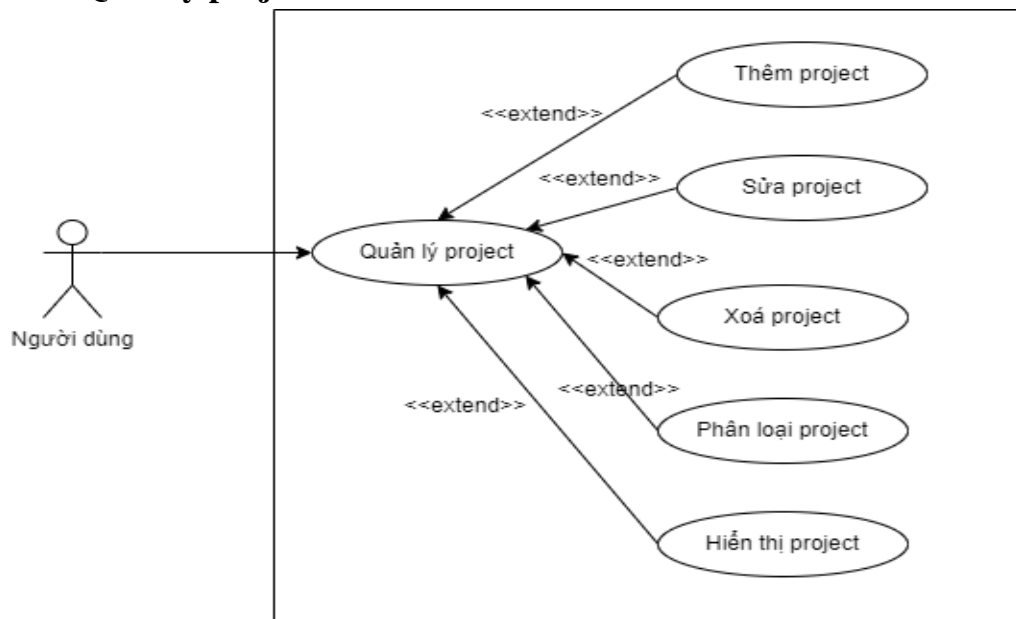
#### 3.1. Biểu đồ ca sử dụng

##### 3.1.1. Biểu đồ ca sử dụng của hệ thống



Hình 2.3.1: Biểu đồ ca sử dụng của hệ thống

##### 3.1.2. Quản lý project



Hình 2.3.2: Biểu đồ ca sử dụng quản lý project

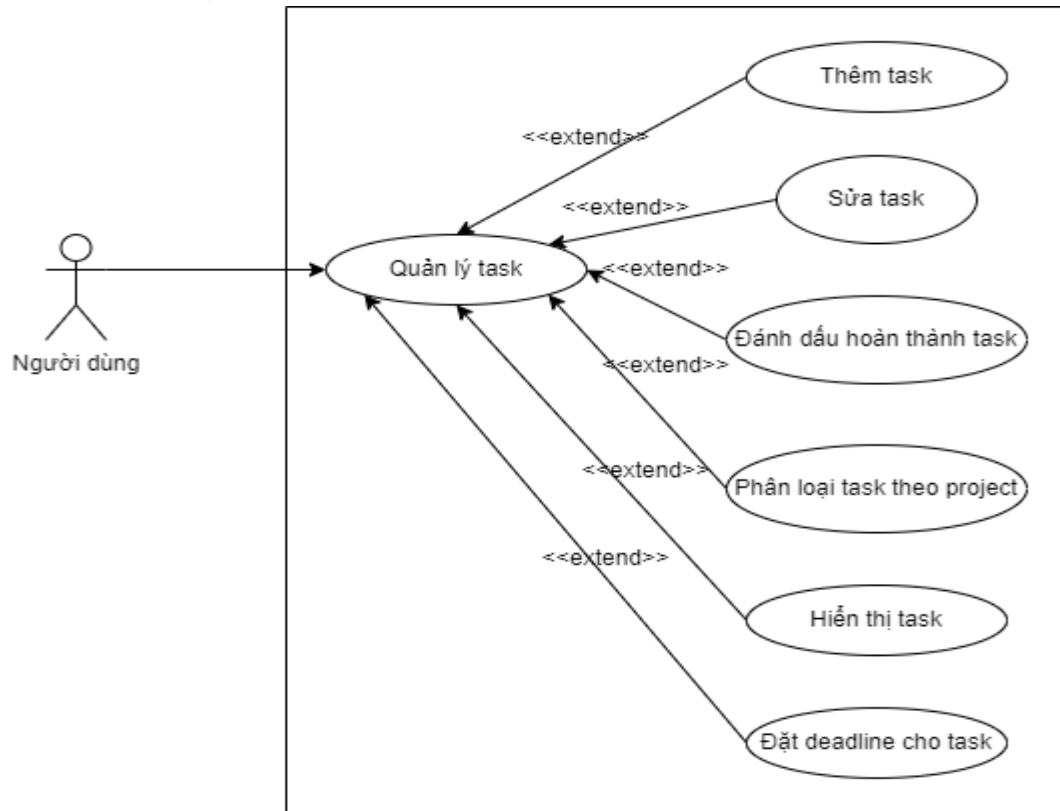


- Đặc tả ca sử dụng

**Bảng 2.3.1:** Đặc tả ca sử dụng quản lý project

Use Case	Quản lý project
Actor	Người dùng
Brief Description	Cho phép người dùng quản lý project của mình
Pre-conditions	Người dùng có userId nằm trong quyền thay đổi project này
Basic Flows	Người dùng có thể quản lý project của mình với chức năng thêm, sửa, xóa Sau mỗi lần thay đổi, ứng dụng sẽ phân loại project và hiển thị ra cho người dùng quản lý

### 3.1.3. Quản lý task



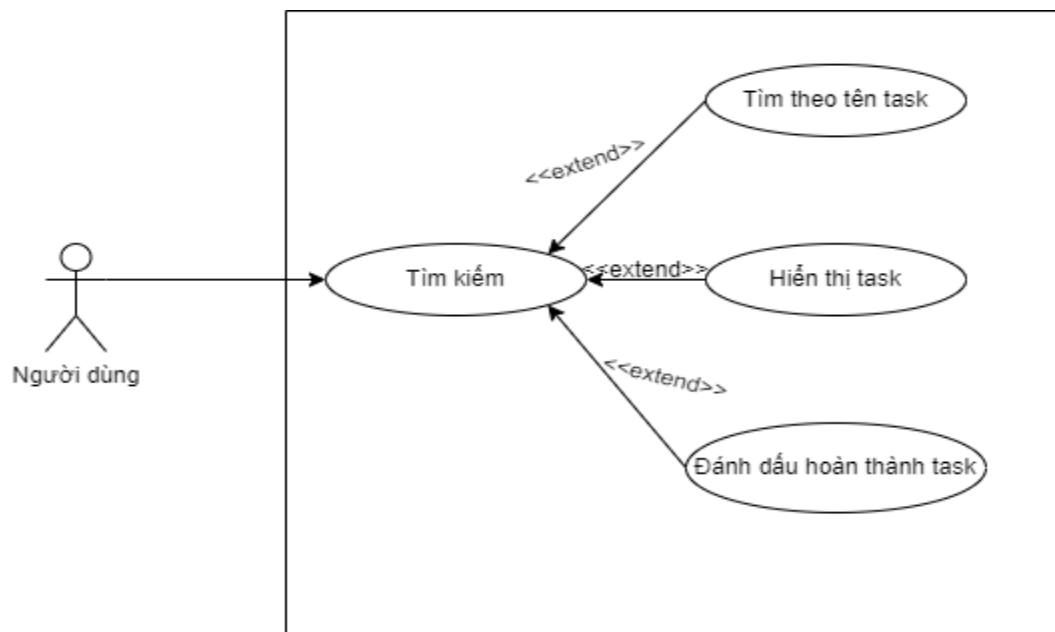
**Hình 2.3.3:** Biểu đồ ca sử dụng quản lý task

- Đặc tả ca sử dụng

**Bảng 2.3.2:** Đặc tả ca sử dụng quản lý task

Use Case	Quản lý task
Actor	Người dùng
Brief Description	Cho phép người dùng quản lý task của mình
Pre-conditions	Người dùng có userId nằm trong quyền thay đổi task này
Basic Flows	<p>Người dùng có thể quản lý task của mình với chức năng thêm, sửa, xóa</p> <p>Sau mỗi lần thay đổi, ứng dụng sẽ phân loại task và hiển thị ra cho người dùng quản lý</p>

### 3.1.4. Tìm kiếm



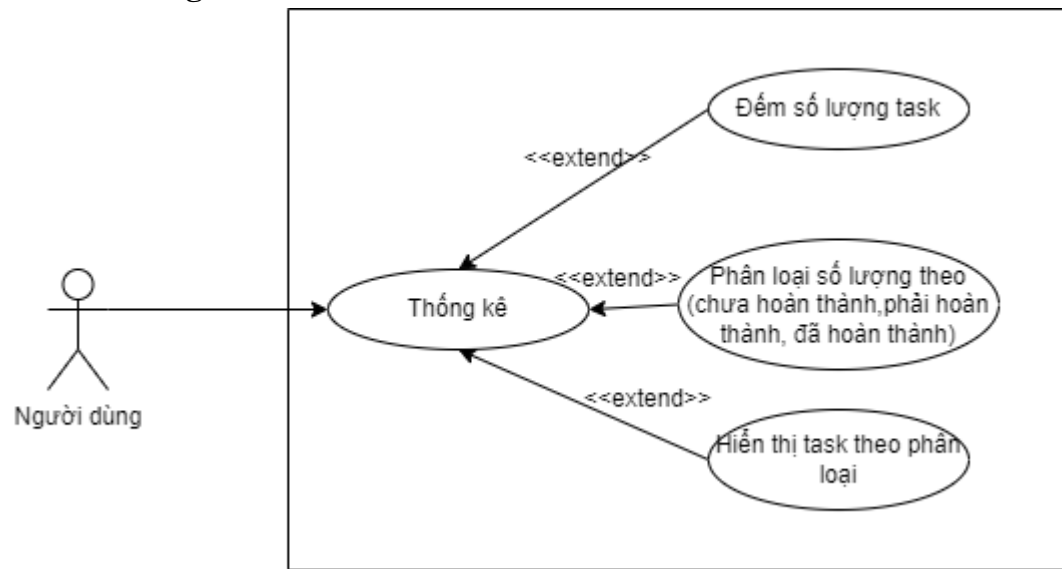
**Hình 2.3.4:** Biểu đồ ca sử dụng tìm kiếm

- Đặc tả ca sử dụng

**Bảng 2.3.3:** Đặc tả ca sử dụng tìm kiếm

Use Case	Tìm kiếm
Actor	Người dùng
Brief Description	Cho phép người dùng quản lý tìm kiếm theo tên task và đánh dấu hoàn task nếu muốn
Basic Flows	<p>Người dùng nhập vào input chức năng sẽ tự động tìm kiếm theo tên task và hiển thị ra cho người dùng</p> <p>Với mỗi task phù hợp với input tìm kiếm thì sẽ có ô CheckBox và tên task, người dùng có thể click vào để đánh dấu hoàn thành task</p>

### 3.1.5. Thống kê



**Hình 2.3.5:** Biểu đồ usecase thống kê

- Đặc tả ca sử dụng

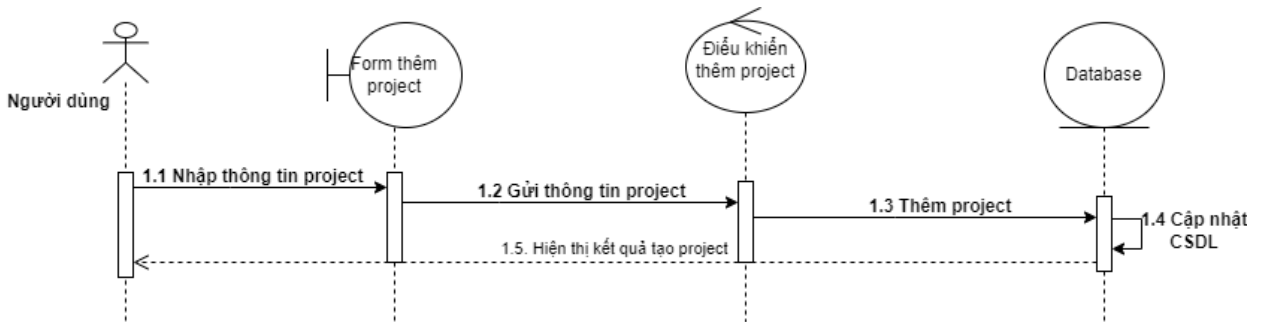
**Bảng 2.3.4:** Đặc tả ca sử dụng thống kê

Use Case	Thống kê
Actor	Người dùng
Brief Description	Cho phép người dùng xem thống kê những task mà mình đã xử lý với task chưa hoàn thành, phải hoàn thành, đã hoàn thành
Pre-conditions	Người dùng tạo task và xử lý task để chức năng có thể thống kê
Basic Flows	Người dùng nhấn vào chức năng thống kê sau đó hệ thống sẽ đếm số lượng task phụ thuộc vào mục chọn (chưa hoàn thành, phải hoàn thành, đã hoàn thành) để hiển thị lên cho người dùng

### 3.2. Xây dựng biểu đồ trình tự

#### 3.2.1. Thêm project

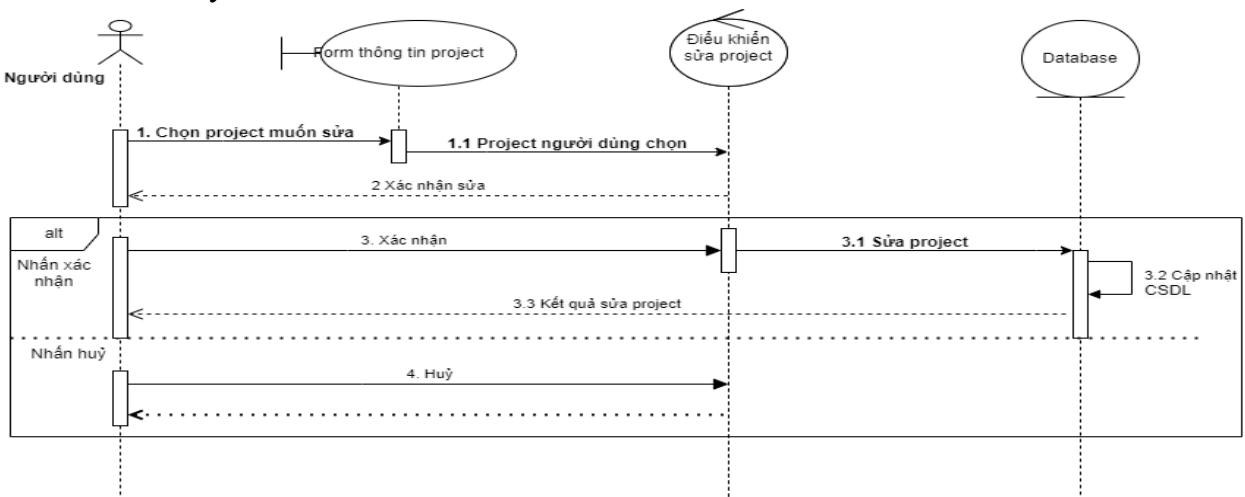
- Khi nhấn nút thêm project một form sẽ hiển thị với input nhập vào tên project rồi nhấn Thêm sẽ gửi dữ liệu lên database sau đó hiển thị lại lên cho người dùng



Hình 2.3.6: Biểu đồ trình tự - Thêm Project

#### 3.2.2. Sửa project

- Khi người dùng hover vào tên project trên màn hình, chức năng ‘Sửa project’ sẽ hiện ra và người dùng có thể nhập vào đây tên project cần sửa rồi nhấn nút ‘Sửa’, database sẽ được yêu cầu update và giao diện sẽ cập nhật lại và hiển thị cho người dùng. Nếu không thì người dùng nhấn ‘Huỷ’ và kết quả sẽ không thay đổi

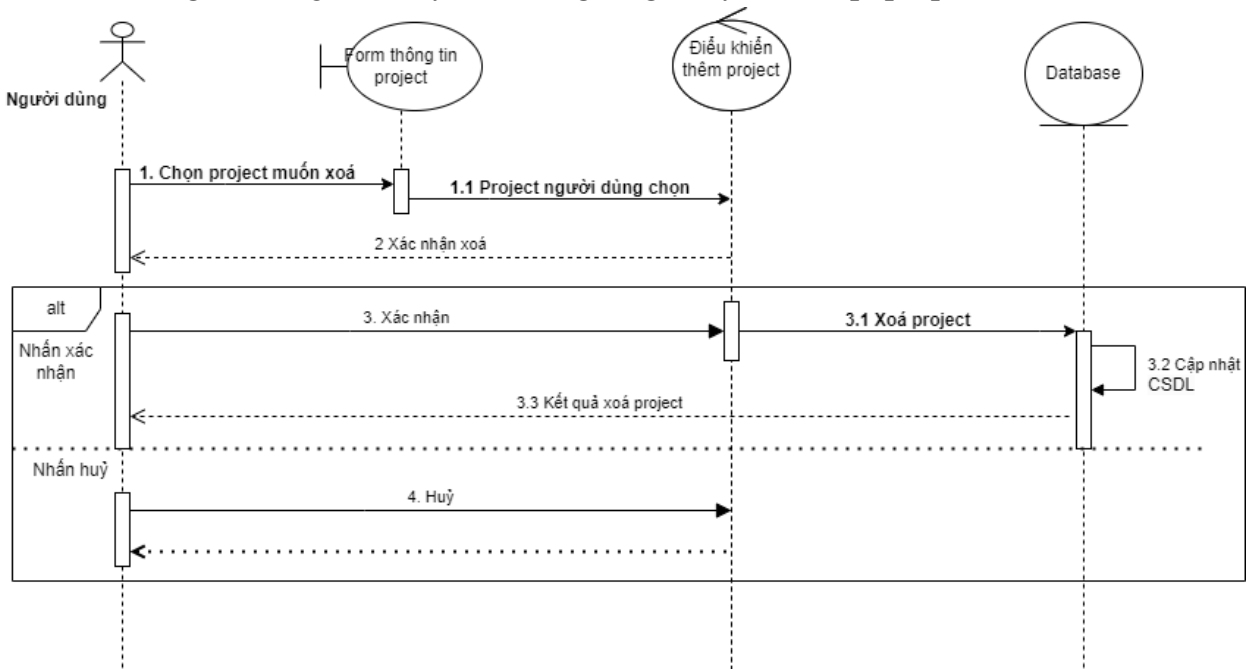


Hình 2.3.7: Biểu đồ trình tự - Sửa Project



### 3.2.3. Xóa project

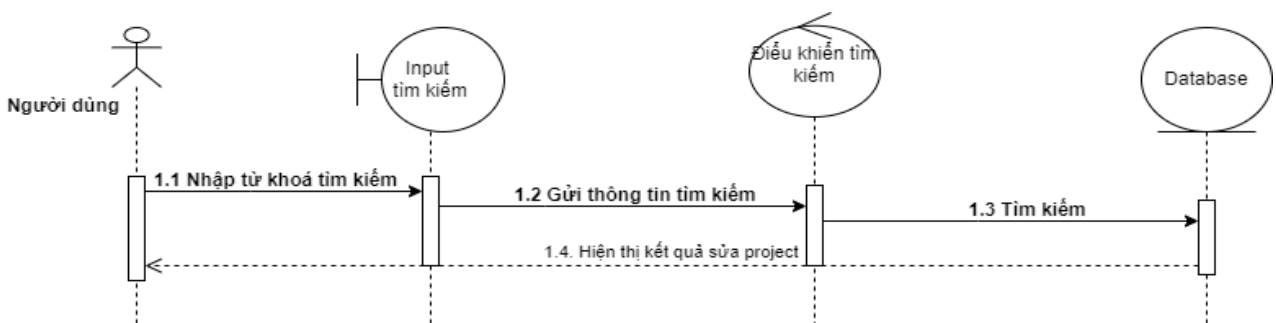
- Khi người dùng hover vào tên project một biểu tượng thùng rác sẽ hiện ra và người dùng click vào biểu tượng để sử dụng chức năng xóa, một pop-up sẽ hỏi người dùng có thực sự muốn xóa project này không, nếu ấn ‘Xóa’, database sẽ xử lý và delete dữ liệu, giao diện sẽ hiển thị lại kết quả cho người dùng. Nếu người dùng ấn ‘Huỷ’ sẽ không có gì thay đổi, và pop-up sẽ ẩn đi



Hình 2.3.8: Biểu đồ trình tự - Xóa Project

### 3.2.4. Tìm kiếm

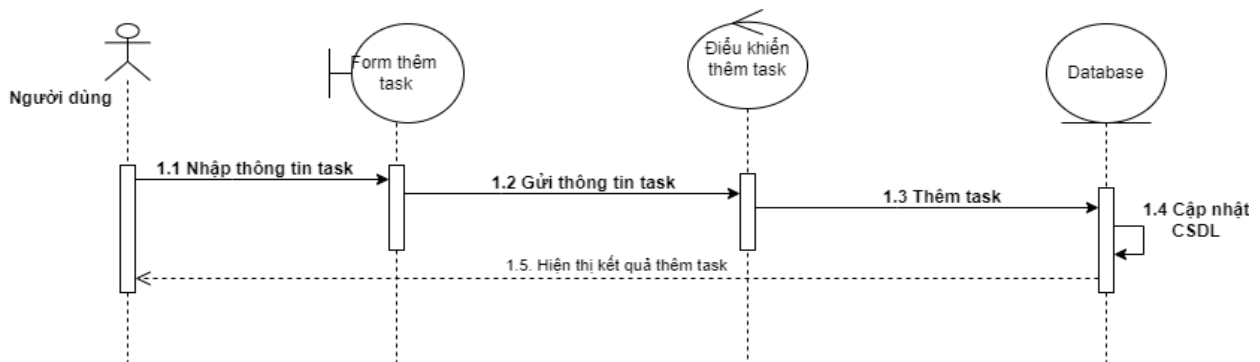
- Người dùng nhập vào input tìm kiếm từ khóa, với mỗi từ người dùng nhập chức năng sẽ chạy và tìm trong database tên task tương tự với đầu vào và hiển thị lên kết quả cho người dùng



Hình 2.3.9: Biểu đồ trình tự - Tìm kiếm

### 3.2.5. Thêm task

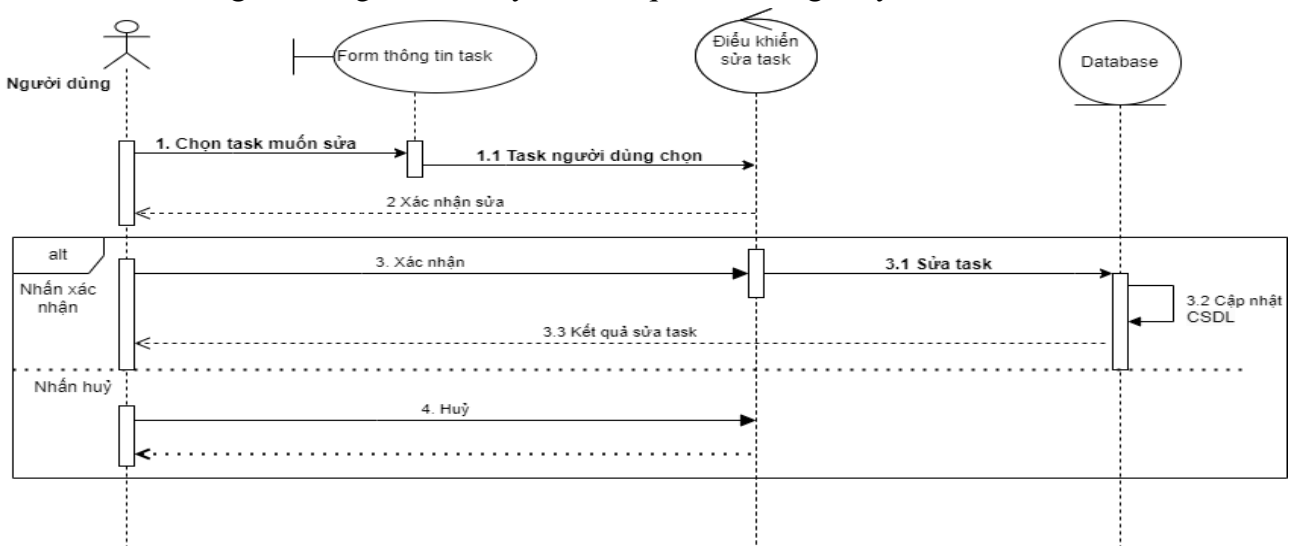
- Khi người dùng nhận vào nút ‘Thêm nhiệm vụ’ sẽ có input để người dùng nhập vào tên task muốn tạo, bên cạnh đó người dùng có thể chọn Project quản lý Tasks đó và deadline của Tasks sau đó nhấn nút ‘Thêm nhiệm vụ’. Cơ sở dữ liệu sẽ cập nhật lại với Task vừa tạo và sẽ hiển thị lại lên giao diện cho người dùng



Hình 2.3.10: Biểu đồ trình tự - Thêm task

### 3.2.6. Sửa task

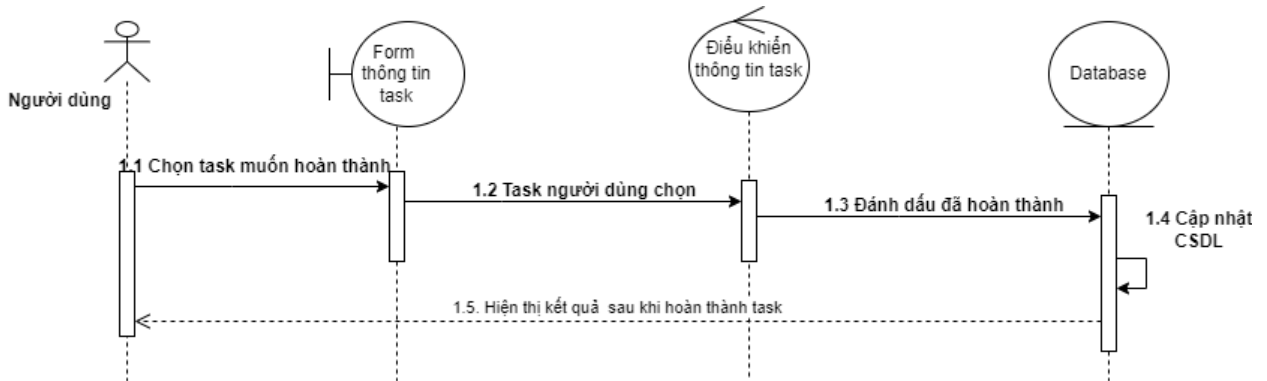
- Khi người dùng nhấn vào chức năng sửa Task trên màn hình, chức năng ‘Sửa tasks’ sẽ hiện ra và người dùng có thể nhập vào đây tên task cần sửa rồi nhấn nút ‘Sửa’, chọn Project của task và deadline cho Task, database sẽ được yêu cầu update và giao diện sẽ cập nhật lại và hiển thị cho người dùng. Nếu không thì người dùng nhấn ‘Huỷ’ và kết quả sẽ không thay đổi



Hình 2.3.11: Biểu đồ trình tự - Sửa task

### 3.2.7. Hoàn thành task

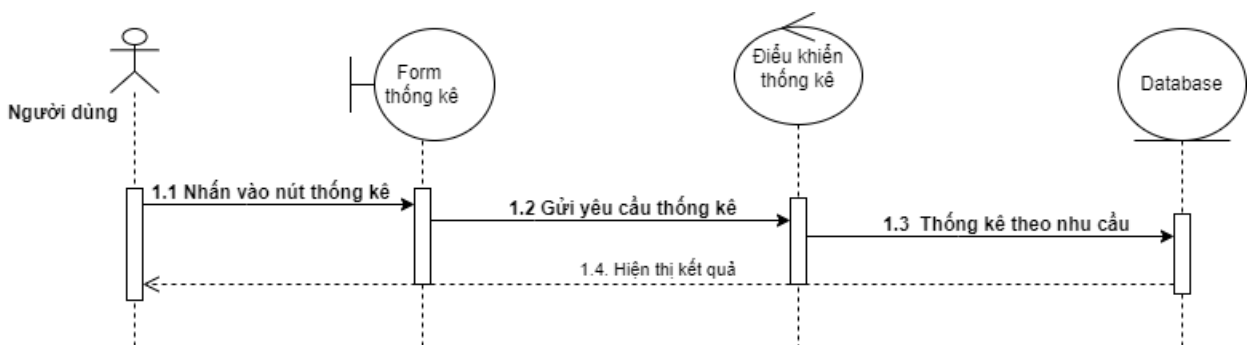
- Khi người dùng nhấn vào ô CheckBox, task sẽ được đánh dấu archived=true, và cập nhật vào cơ sở dữ liệu, kết quả sẽ được hiển thị lại trên giao diện



Hình 2.3.12: Biểu đồ trình tự - Hoàn thành task

### 3.2.8. Thống kê

- Khi người dùng nhấn vào nút thống kê, một giao diện thống kê sẽ hiện ra và có những chức năng thống kê như task chưa hoàn thành, phải hoàn thành và đã hoàn thành. Yêu cầu sẽ tự động gửi và lấy dữ liệu trong database rồi hiển thị ra kết quả cho người dùng



Hình 2.3.13: Biểu đồ trình tự - Thống kê

## 4. Thiết kế cơ sở dữ liệu

### 4.1. Thực thể dữ liệu

Tasks	
PK	<u>DocID</u>
	task
	archived
	date
	projectid
	userid

Users	
PK	<u>DocID</u> <u>userid</u>
	firstName
	lastName

Projects	
PK	<u>DocID</u> <u>projectId</u>
	name
	userID

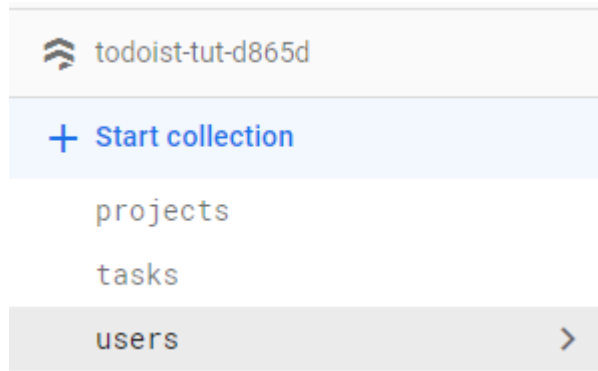
Hình 2.4.1: Thực thể dữ liệu Todoist

Mô tả thực thể:

- Mỗi một bảng dữ liệu là một collection, đây là một khái niệm rất quan trọng trong Firestore, một dịch vụ cơ sở dữ liệu NoSQL của Firebase. Với mỗi tên collection là duy nhất, ta sẽ chứa được tập các document được truy cập và quản lý độc lập trong collection

Trong Todoist, ta sử dụng 3 collection để lưu trữ dữ liệu:

- Projects sẽ lưu lại các thuộc tính của những dự án (sẽ được giới thiệu sau) và sẽ là thành phần chứa các Tasks được quản lý bởi Users. Việc này có thể giúp nhiều Users quản lý và tham gia phát triển được cùng một Projects
- Tasks sẽ lưu lại những thuộc tính của nhiệm vụ (sẽ giới thiệu sau) và cũng được quản lý bởi Users. Nhiều Users khác nhau cũng có thể quản lý hay phát triển Tasks song song, đồng bộ với nhau. Tuy nhiên mỗi một thành phần Tasks thì chỉ được quản lý trong một Projects
- Users sẽ lưu lại thuộc tính của người dùng (sẽ được giới thiệu sau), giúp quản lý được thông tin của người dùng - tác nhân chính điều khiển hệ thống cũng như tham gia phát triển Projects và Tasks



Hình 2.4.2: Collection trong Todoist

- Trong collection sẽ có một loạt document để có thể lưu trữ fields, mỗi documents thì chỉ thuộc về một collection cụ thể.

Mỗi document là đơn vị cơ bản để lưu trữ và quản lý dữ liệu trong Firestore, một dịch vụ cơ sở dữ liệu NoSQL của Firebase.

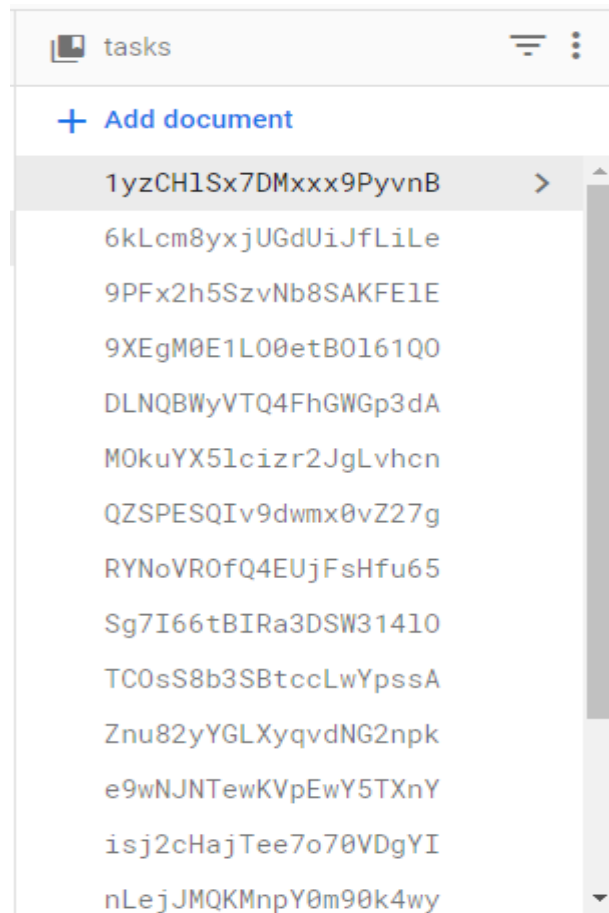
Mỗi document là một bản ghi JSON độc lập và chứa dữ liệu theo cấu trúc key - value. Key sẽ đại diện cho thuộc tính hoặc thông tin cụ thể. Value sẽ đại diện cho nội dung của Key và sẽ được tổ chức dưới dạng JSON

Mỗi document trong Firestore được xác định bằng một ID duy nhất. ID này có thể được tạo ra tự động bởi Firestore hoặc được chỉ định bởi người dùng và được gọi là **docID**. Trong ứng dụng Todoist thì đã để công việc này cho Firestore quản lý và sẽ chỉ tập trung vào các trường ID khác. Cả hai ID này sẽ đóng vai trò quan trọng trong việc xác định và truy cập vào document trong cơ sở dữ liệu của Todoist

Các document này còn có cấu trúc động nghĩa là mỗi document có thể có các trường khác nhau hoặc số lượng trường khác nhau. Tuy nhiên trong Todoist này thì mọi document đều có thuộc tính, trường như nhau nhưng điều này có thể sẽ hữu ích khi cần lưu trữ dữ liệu có tính linh hoạt và có thể mở rộng trong tương lai

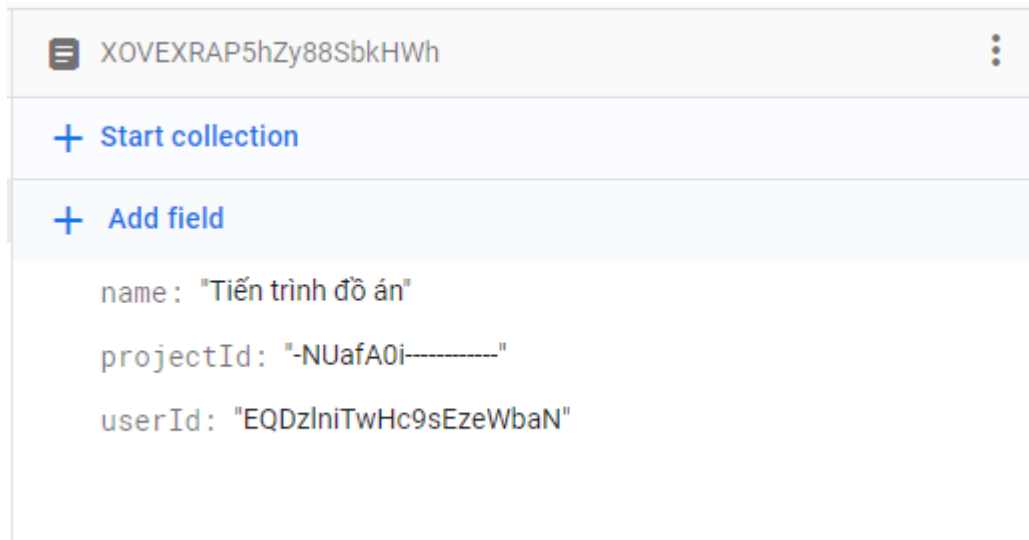
Một trong những ưu điểm và cũng là thứ quan trọng trong Todoist cũng như Firestore là khả năng đồng bộ dữ liệu thời gian thực. Khi dữ liệu trong document thay đổi, nó sẽ tự động cập nhật và đồng bộ trên tất cả các thiết bị hoặc ứng dụng đang kết nối. Điều này thực sự cần thiết cho ứng dụng Todoist để giúp người dùng quản lý tác vụ của mình một cách tối ưu nhất tránh tình trạng bất đồng bộ dữ liệu với nhau. Nó còn đặc biệt quan trọng hơn nếu ứng dụng phát triển trong đa nền tảng và việc giúp nhiều người dùng cũng tham gia tổ chức, phát triển Projects hoặc Tasks với nhau

Việc sử dụng documents trong Firestore mang lại nhiều lợi ích. Bạn có thể lưu trữ thông tin chi tiết về các đối tượng, sản phẩm, người dùng và nhiều hơn nữa. Dữ liệu trong documents có thể được truy cập và cập nhật dễ dàng từ các thiết bị và ứng dụng khác nhau, đảm bảo tính nhất quán và đồng bộ giữa các nền tảng.



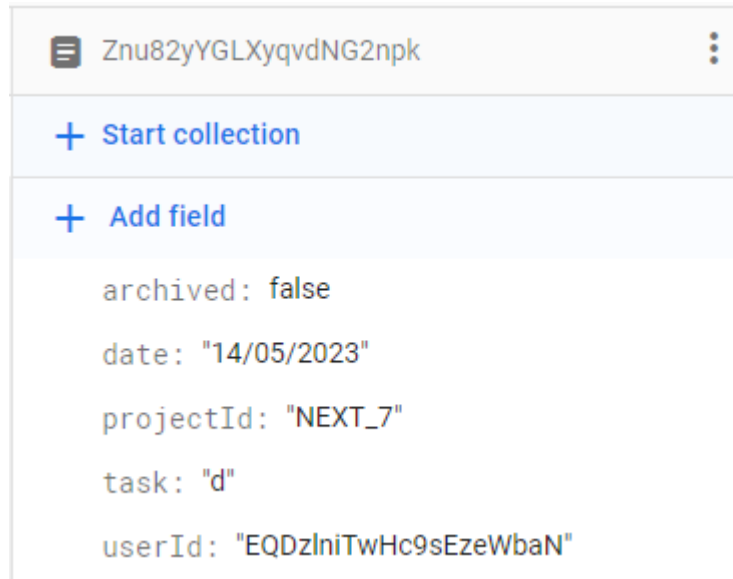
**Hình 2.4.3:** Document trong Todoist

- Trong Firebase, "fields" (trường dữ liệu) là các thành phần cơ bản của một document trong Firestore, dịch vụ cơ sở dữ liệu NoSQL của Firebase. Fields đại diện cho các thuộc tính hoặc thông tin cụ thể trong document và chứa các giá trị tương ứng. Dưới đây là mô tả về các fields trong Todoist:
  - Projects bao gồm:
    - `projectId(String)`: lưu trữ lại ID project do người dùng tạo ra và tự định nghĩa giúp lưu trữ lại `projectId` để sử dụng nhằm mục đích phân loại các tasks theo từng project
    - `Name(String)`: lưu trữ lại tên của project là một trường quan trọng của Projects để có thể lưu trữ dữ liệu và hiển thị lên giao diện
    - `userId(String)`: lưu trữ lại ID của người dùng giúp kiểm soát được Projects nào được người dùng nào quản lý. Ở đây chỉ có một người dùng nên tạm thời để mặc định



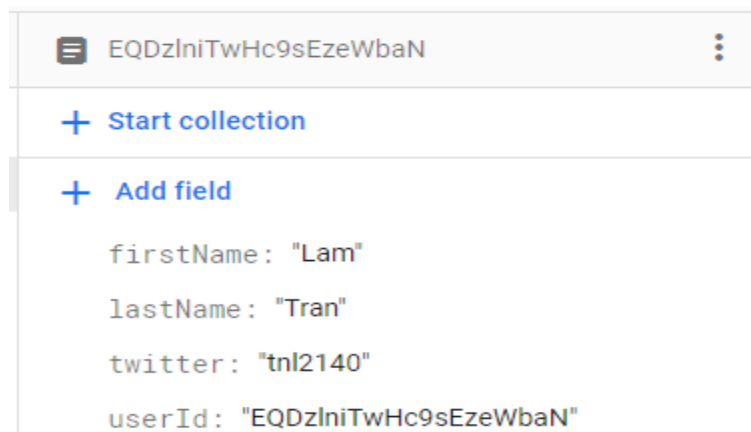
**Hình 2.4.4:** Các trường của project

- Tasks bao gồm:
  - Archived(boolean): đây là trường đánh dấu lại trạng thái hoàn thành của tasks. Nếu trạng thái bằng true thì sẽ bị ẩn khỏi giao diện và đánh dấu đã hoàn thành task này rồi, hãy tiếp tục hoàn thành task khác
  - Date(string): lưu lại deadline của tasks với những mục chọn như (hôm nay, ngày mai, tuần tới và không thời hạn) sẽ được viết dưới dạng DD/MM/YYYY
  - projectId(String): đây là trường lưu lại ID project của task giúp cho người dùng phân loại task theo project, trong trường hợp không có project nào được chỉ định thì sẽ lưu lại theo (INBOX, TODAY, NEXT\_7)
  - task(String): lưu trữ lại tên của task là một trường quan trọng của Tasks để có thể lưu trữ dữ liệu và hiển thị lên giao diện
  - userId(String): lưu trữ lại ID của người dùng giúp kiểm soát được Tasks nào được người dùng nào quản lý. Ở đây chỉ có một người dùng nên tạm thời để mặc định



**Hình 2.4.5:** Các trường của tasks

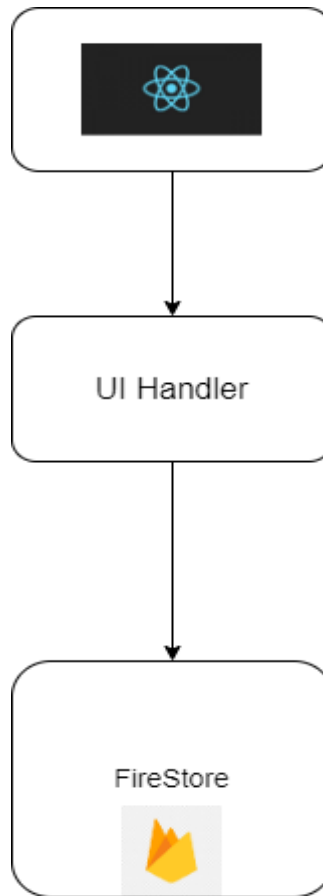
- Users bao gồm:
  - firstName(String): là tên của người dùng
  - lastName(String): là họ của người dùng
  - userId: là ID của người dùng, một trường quan trọng để có thể quản lý được tài khoản này có thể phát triển hay thay đổi Projects và Tasks nào trong hệ thống



**Hình 2.4.6:** Các trường của users



## 5. Sơ đồ ứng dụng



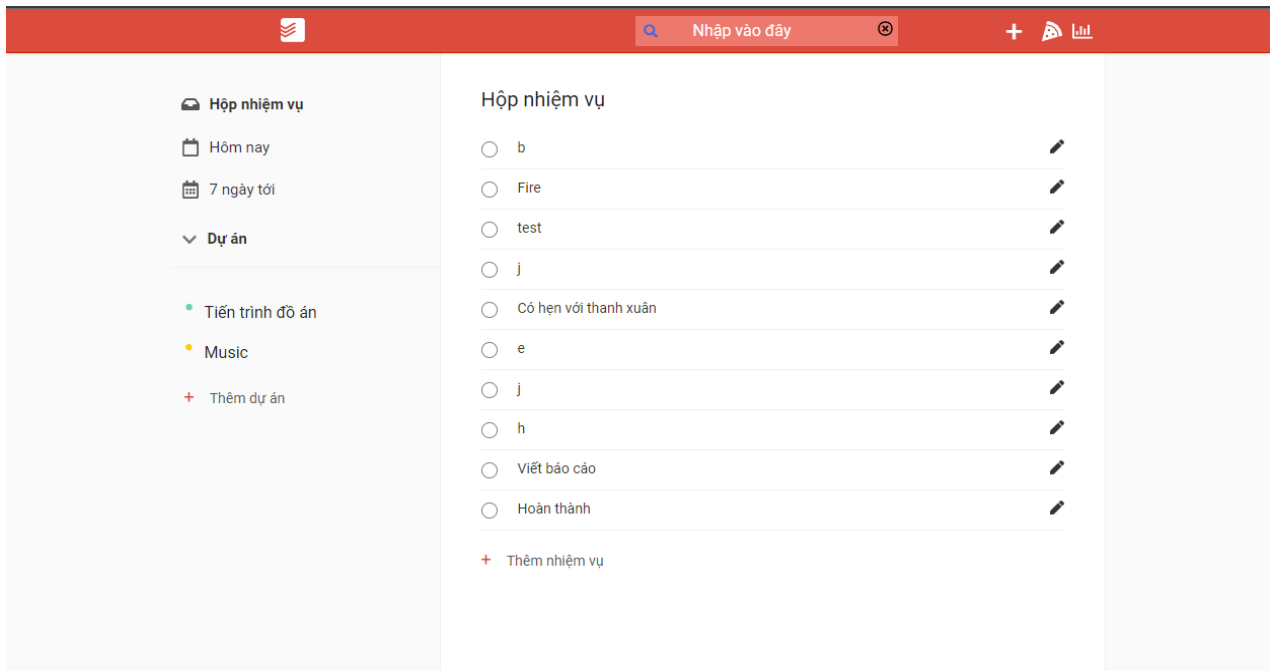
**Hình 2.5.1:** Sơ đồ kiến trúc ứng dụng

Ứng dụng Todoist được xây dựng bằng cách

- Giao diện người dùng (UI): Được xây dựng bằng ReactJS, gồm các thành phần như nút bấm, trường nhập liệu, danh sách công việc, v.v. Giao diện này cho phép người dùng tương tác với các chức năng của Todoist.
- UI Handler: Đây là một tầng trung gian giữa giao diện người dùng và logic xử lý. UI Handler nhận các sự kiện và yêu cầu từ giao diện người dùng và gọi đến để xử lý. UI Handler đại diện cho các hành động cụ thể được thực hiện trên giao diện người dùng, chẳng hạn như thêm công việc, sửa công việc, xóa công việc, v.v.
- Firestore: Firestore là một cơ sở dữ liệu do Firebase cung cấp. Nó được sử dụng để lưu trữ và quản lý dữ liệu liên quan đến công việc trong Todoist. Khi UI Handler gọi yêu cầu xử lý dữ liệu, nó sẽ gọi đến Firestore để lưu trữ hoặc truy vấn dữ liệu.

Sơ đồ kiến trúc này giúp phân chia rõ ràng các trách nhiệm và logic xử lý trong ứng dụng Todoist, từ giao diện người dùng, qua UI Handler, đến cơ sở dữ liệu Firestore. Điều này giúp tăng tính mô-đun hóa, dễ bảo trì và phát triển ứng dụng một cách hiệu quả.

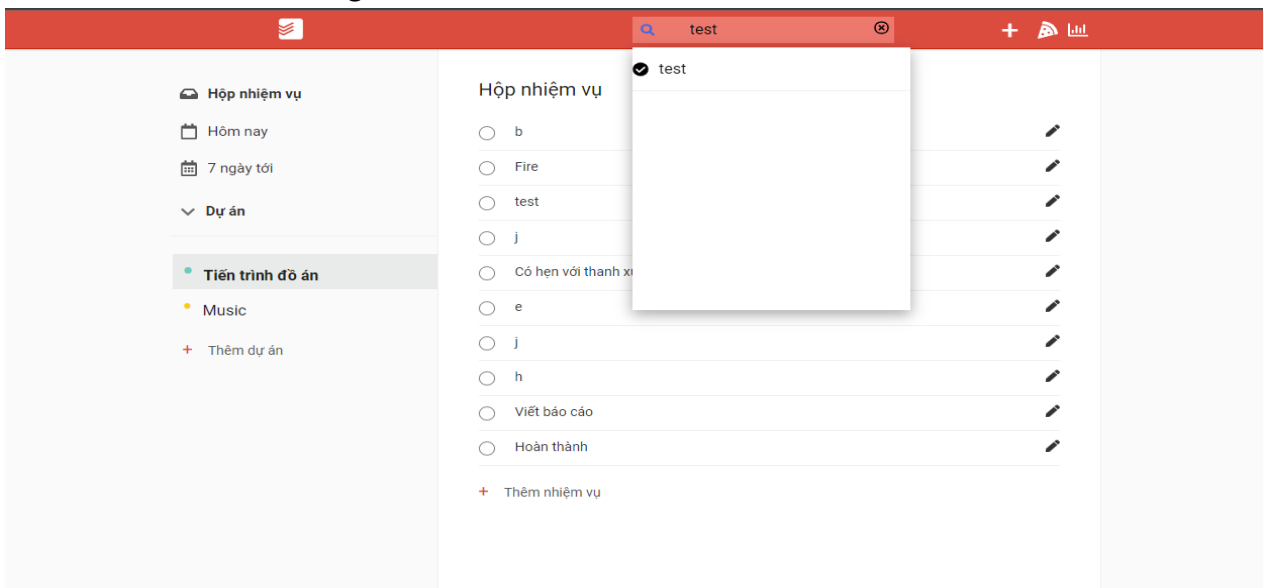
## CHƯƠNG 3: GIAO DIỆN HỆ THỐNG VÀ KIỂM THỬ



**Hình 3.1:** Giao diện trang chính

- Ứng dụng:
  - Những chức năng chính của Todoist:
    - Trên phần Header (bên trên):
      - Chức năng tìm kiếm theo tên task giúp người dùng dễ dàng tìm kiếm các task và có thể đánh dấu hoàn thành tại đây để dễ dàng hơn trong việc quản lý task
      - Chức năng thêm task nhanh giúp người dùng tạo một task nhanh chóng với đầy đủ thành phần như tên task, loại project, deadline
      - Chức năng dark mode giúp người dùng có thể chuyển giao diện sang màu đen trắng để phù hợp với nhu cầu người sử dụng
      - Chức năng thống kê task giúp người dùng có thể quan sát và theo dõi được những nhiệm vụ của mình chỉ với một nút bấm
    - Phần sidebar (bên trái):
      - Phân loại task theo ngày (Hộp nhiệm vụ (những nhiệm vụ không có thời hạn), Hôm nay (những nhiệm vụ có deadline là hôm nay), 7 ngày tới (những nhiệm vụ có deadline trong vòng bảy ngày tới))

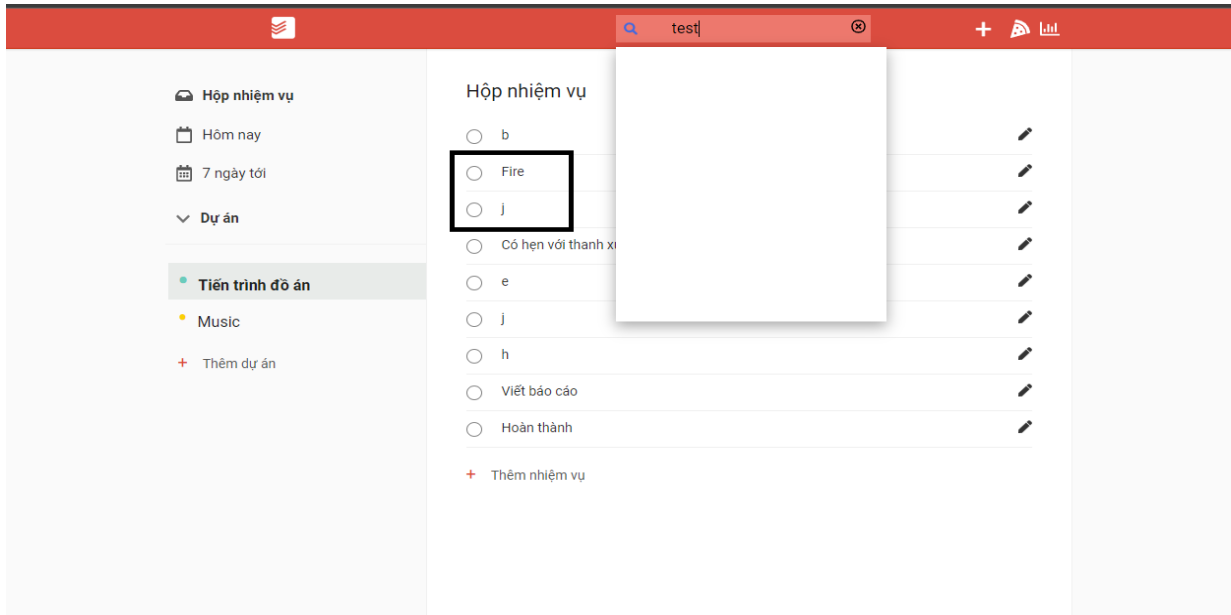
- Liệt kê những project đã tạo trong ứng dụng để có thể dễ dàng quản lý và thay đổi theo nhu cầu người dùng
- Chức năng thêm project giúp người dùng có thể tạo ra project mới để quản lý các task
- Phần content:
  - Hiện thị ra các task đã thêm theo từng Project hoặc deadline
  - Hộp Checkbox giúp người dùng có thể đánh dấu những nhiệm vụ đã hoàn thành bằng cách click vào và task sẽ biến mất khỏi ứng dụng
  - Chức năng thêm nhiệm vụ cũng tương tự với chức năng thêm nhiệm vụ nhanh nhưng đã chọn sẵn project được thêm vào hoặc người dùng có thể tùy chỉnh theo nhu cầu
  - Hình cây bút bên cạnh task giúp người dùng có thể chỉnh sửa tên, deadline, loại project của task một cách nhanh chóng
- Giới thiệu về chức năng:
  - Chức năng tìm kiếm:



**Hình 3.2:** Chức năng tìm kiếm 1

Mô tả: Khi nhập vào input tìm kiếm theo tên sẽ lọc ra task có chữ cái tương tự, người dùng có thể click vào một trong những task liệt kê ở bên dưới sau khi tìm kiếm xong để có thể đánh dấu hoàn thành task đó ngay tại đây

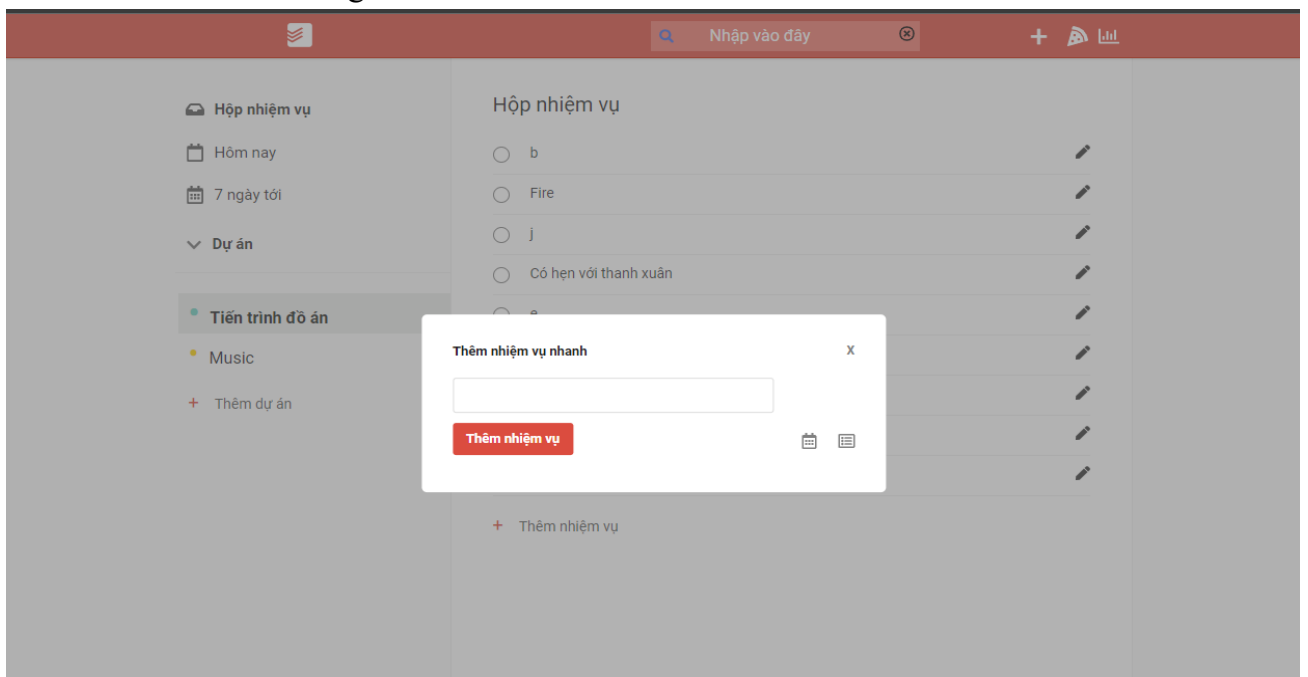
Ví dụ với task ‘test’



**Hình 3.3:** Chức năng tìm kiếm 2

Cả phần tìm kiếm và phần hiển thị giao diện đều không còn task ‘test’ do đã được đánh dấu hoàn thành

■ Chức năng thêm task nhanh:



**Hình 3.4:** Thêm task nhanh 1

Mô tả: Nhập vào input tên task và chọn deadline và loại project. Nhấn vào button ‘Thêm nhiệm vụ’ sẽ đưa task với những thông tin trên vào vị trí tương ứng. Nếu không chọn deadline sẽ tự động đưa vào ‘Hộp nhiệm vụ’. Nếu không chọn project sẽ gán vào mục (Hộp nhiệm vụ, Hôm nay, 7 ngày tới) tùy thuộc vào deadline của task.

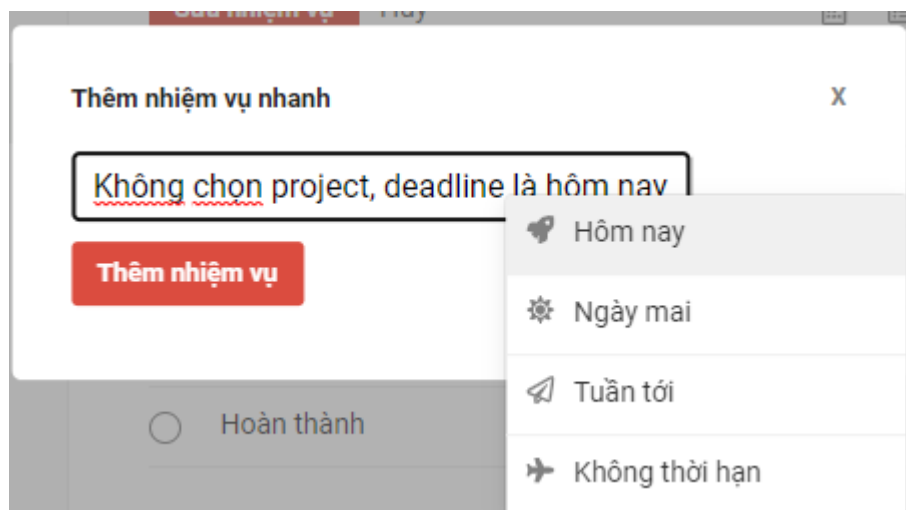
Ví dụ nếu người dùng không chọn deadline hay project

**Hình 3.5:** Thêm task nhanh 2

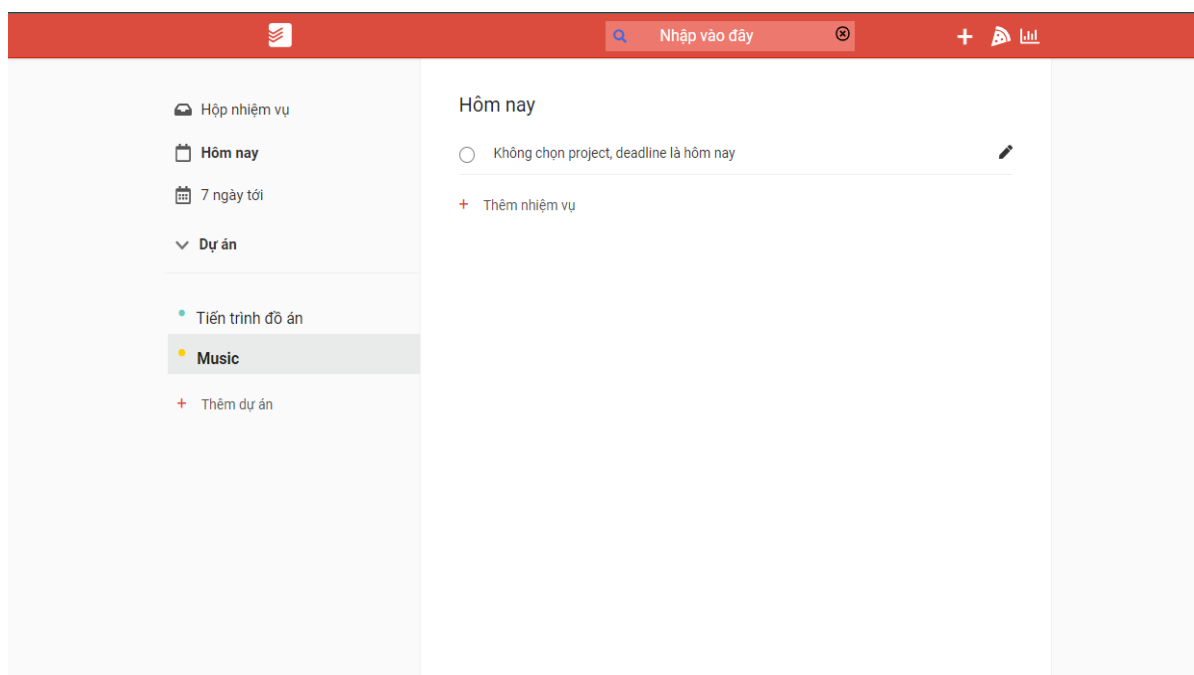
**Hình 3.6:** Thêm task nhanh 3

Kết quả sẽ được hiển thị trong ‘Hộp nhiệm vụ’

Ví dụ nếu người dùng không chọn project, và chọn deadline là hôm nay



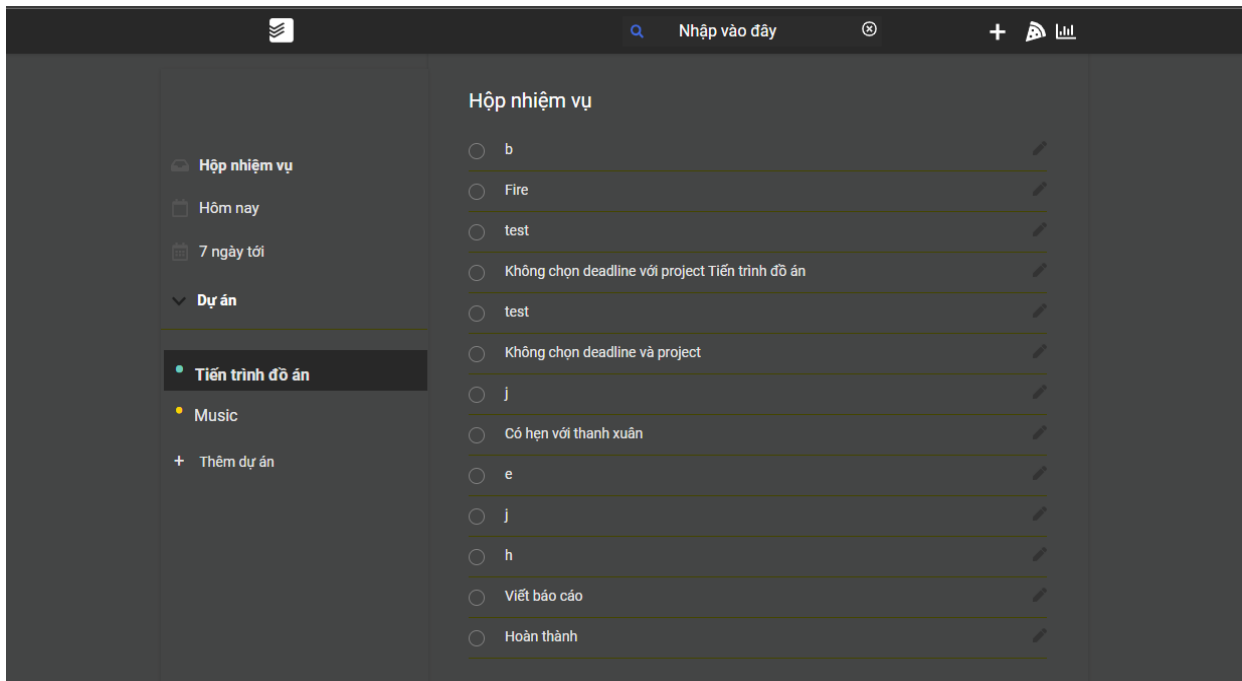
**Hình 3.7:** Thêm task nhanh 4



**Hình 3.8:** Thêm task nhanh 5

Khi nhấn vào mục ‘Hôm nay’ task sẽ được hiển thị ở đây

■ Chức năng dark mode:



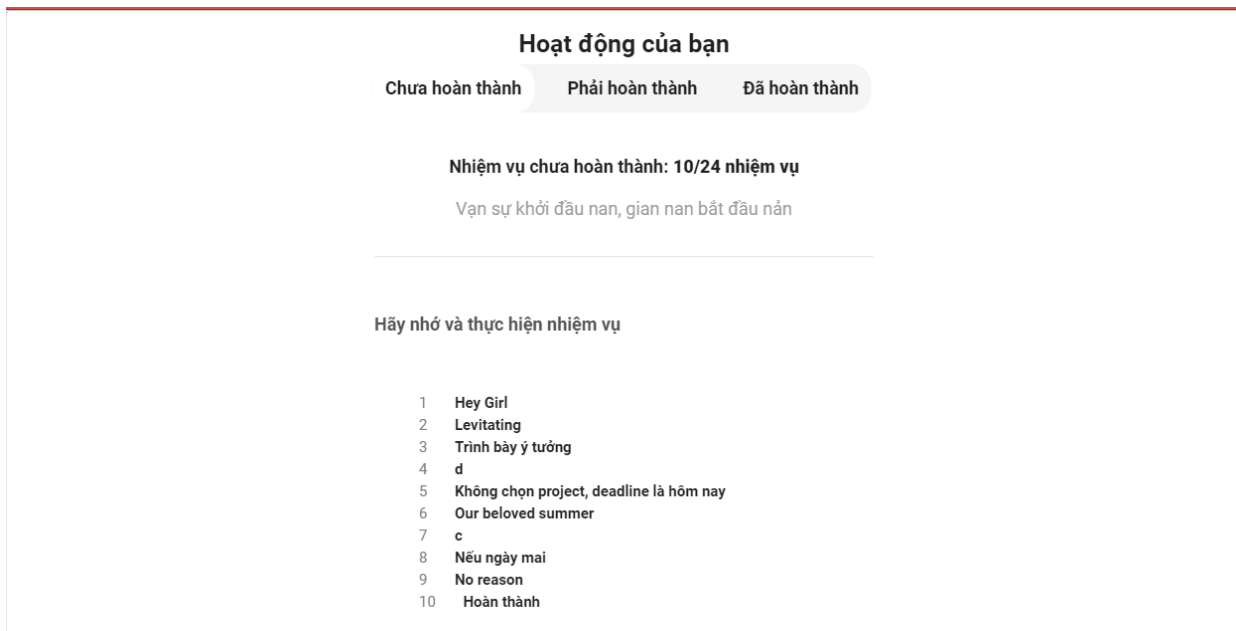
**Hình 3.9:** Chức năng dark mode

Mô tả: Khi nhấn vào nút dark mode sẽ chuyển giao diện về nền đen chữ trắng

■ Chức năng thống kê task:

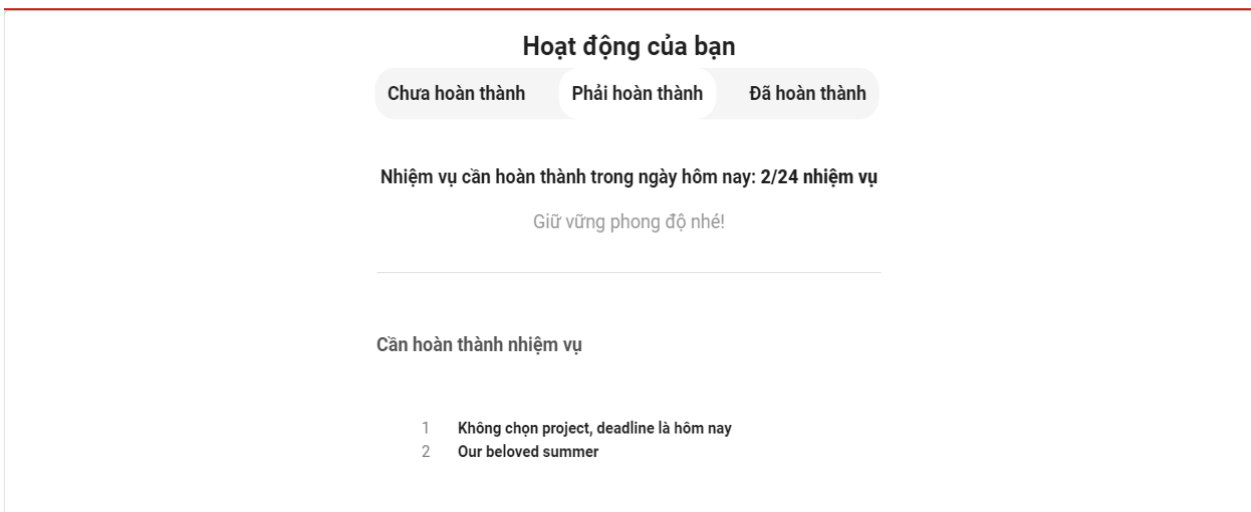
Mô tả: Khi nhấn vào nút thống kê task, sẽ có 3 mục thống kê chính đó là : Chưa hoàn thành, Phải hoàn thành, Đã hoàn thành

- Chưa hoàn thành: sẽ đếm tất cả số lượng task đã tạo nhưng chưa đánh dấu hoàn thành và sẽ liệt kê 10 task chưa hoàn thành theo thứ tự deadline giảm dần (nghĩa là nhiệm vụ có deadline xa nhất rồi đến những nhiệm vụ có deadline gần nhất)



**Hình 3.10:** Chức năng thống kê - Task chưa hoàn thành

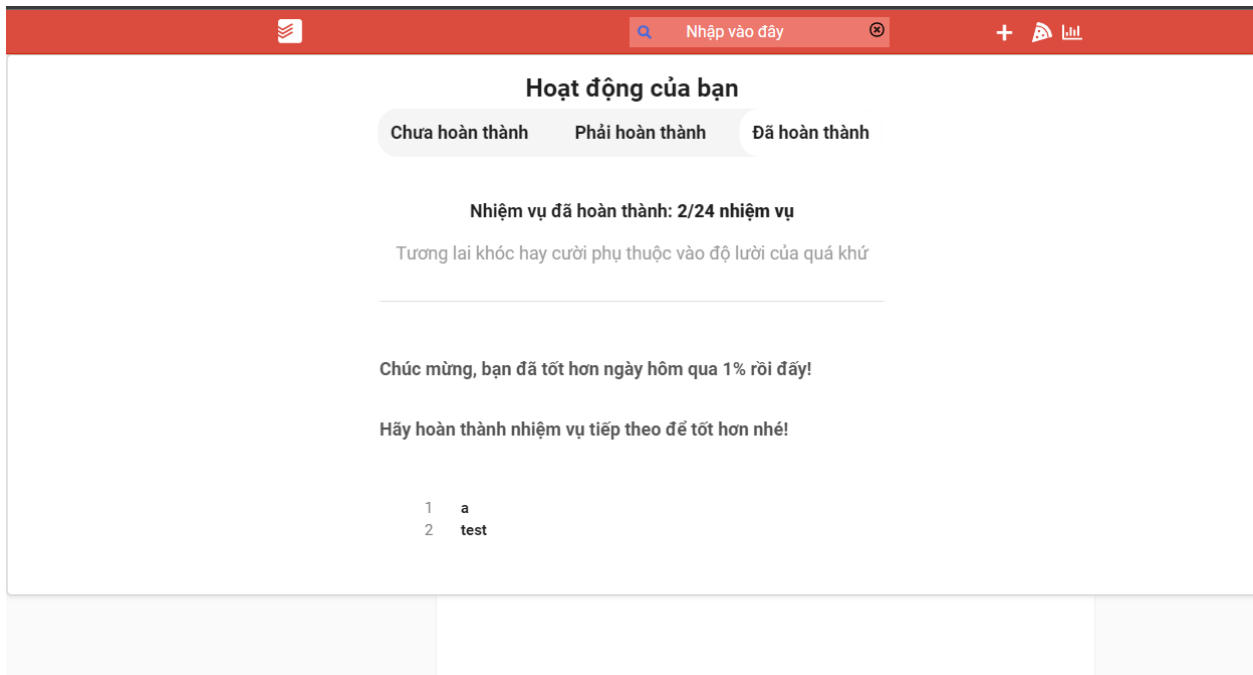
- **Phải hoàn thành:** sẽ đếm số lượng task phải hoàn thành trong ngày hôm nay (nghĩa là deadline là hôm nay) và liệt kê ra những task như vậy để người dùng có thể theo dõi và chủ động hoàn thành task



**Hình 3.11:** Chức năng thống kê - Task phải hoàn thành



- Đã hoàn thành: sẽ đếm số lượng task đã hoàn thành và liệt kê ra những task như vậy để người dùng có thể theo dõi tiến trình của mình

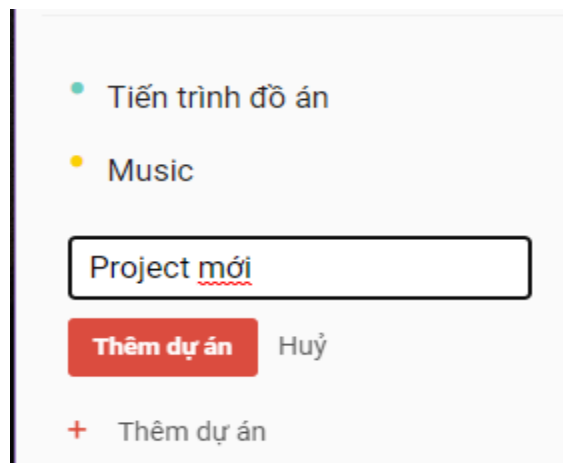


**Hình 3.12:** Chức năng thống kê - Task đã hoàn thành

■ Chức năng thêm dự án:

Mô tả: Khi nhấn vào nút '+ Thêm dự án' sẽ hiển thị một pop-up giúp người dùng thêm một dự án mới vào để quản lý

Ví dụ thêm 'Project mới' thì ngay lập tức ứng dụng sẽ hiển thị lên project vừa tạo

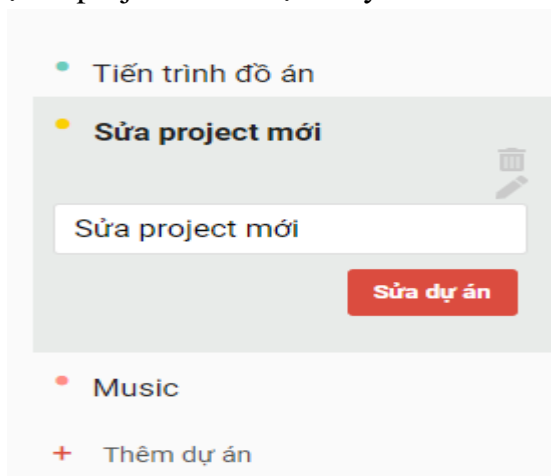


**Hình 3.13:** Chức năng thêm dự án mới

■ Chức năng sửa dự án:

Mô tả: Khi hover vào project bất kỳ sẽ hiện ra một input nhập vào tên muốn sửa của project đó

Ví dụ sửa ‘Project mới’ thành ‘Sửa Project mới’ thì ngay lập tức ứng dụng sẽ hiển thị lên project vừa được thay đổi

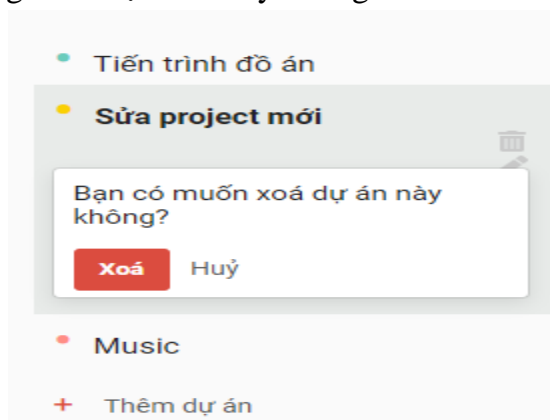


**Hình 3.14:** Chức năng sửa dự án

■ Chức năng xoá dự án:

Mô tả: Sau khi hover vào project bất kỳ, nhấn vào biểu tượng thùng rác sẽ hiện ra pop-up xác nhận xoá project. Người dùng có thể xác nhận xoá hoặc không với project được chọn

Ví dụ: Nhấn nút xoá bên cạnh ‘Sửa project mới’ thì hộp thoại sẽ hiện ra để người dùng xác nhận xoá hay không



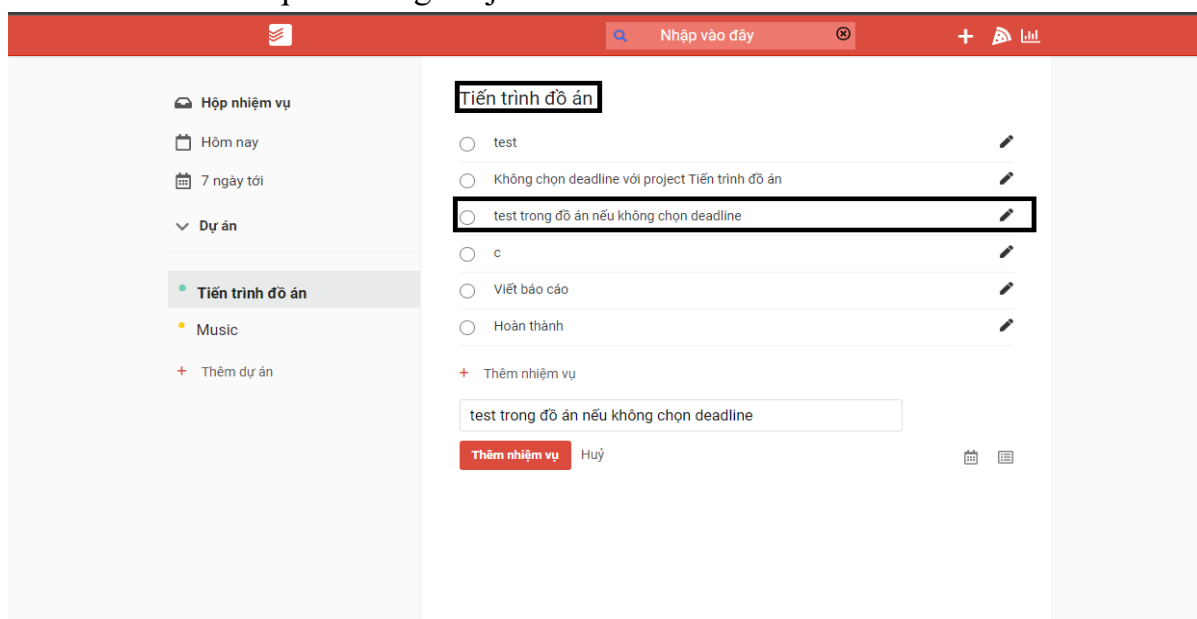
**Hình 3.15:** Chức năng xoá dự án

### ■ Chức năng thêm nhiệm vụ

Mô tả: Tương tự với chức năng thêm nhiệm vụ nhanh nhưng ở đây người dùng sẽ được tự động chọn sẵn project ở nơi mà người dùng click vào nút thêm nhiệm vụ hoặc có thể tùy chỉnh nơi muốn lưu task

Ví dụ: Người dùng thêm một task ‘test trong đồ án nếu không chọn deadline’ với project ở đây là ‘Tiến trình đồ án’ và deadline không chọn gì

Kết quả ở trong Project ‘tiến trình đồ án’:



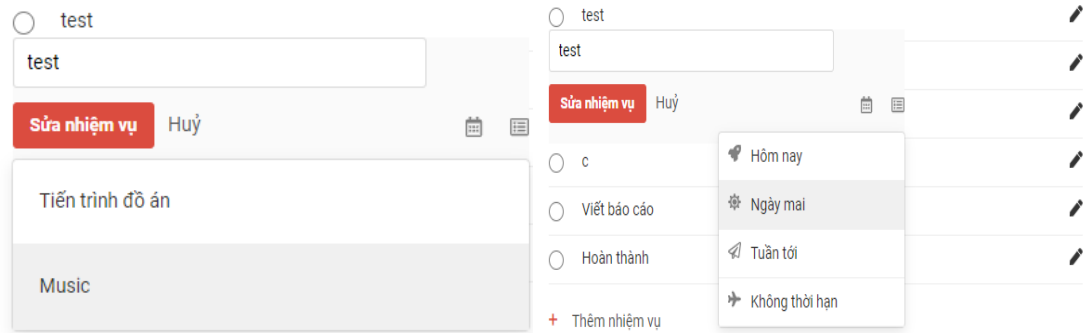
**Hình 3.16:** Chức năng thêm nhiệm vụ

Kết quả cũng hiển thị ở trong các mục deadline như (Hộp nhiệm vụ, Hôm nay, 7 ngày tới)

### ■ Chức năng sửa nhiệm vụ:

Mô tả: Người dùng nhấn vào hình cây bút bên cạnh task thì sẽ hiện ra pop-up với input là tên nhiệm vụ và nút đặt deadline, project của task để thay đổi theo nhu cầu của người dùng.

Ví dụ: Người dùng muốn sửa task ‘test’ trong ‘Tiến trình đồ án’ sang ‘Music’ và deadline là ngày mai. Người dùng chỉ cần nhấn vào nút chọn deadline và project tương ứng



**Hình 3.17:** Chức năng sửa nhiệm vụ

■ Chức năng đánh dấu hoàn thành nhiệm vụ:

Mô tả: Người dùng nhấn vào ô tròn nhỏ trước nhiệm vụ để có thể hoàn task của mình một cách thuận tiện. Nhiệm vụ sẽ biến mất khi được người dùng đánh dấu

- Kiểm thử bằng React Testing Library:
  - Việc này giúp ứng dụng Todoist được kiểm thử giao diện người dùng, đảm bảo rằng các phần tử được hiển thị đúng cách và có các thuộc tính, lớp phù hợp
  - Trong React Testing Library sẽ có các kịch bản khác nhau để có thể thực thi, kiểm thử các chức năng của ứng dụng. Bằng cách sử dụng các hàm, phần tử cơ bản để kiểm tra có tồn tại hay không, có hoạt động chính xác hay không sẽ làm tăng sự tin cậy và chất lượng của ứng dụng, đồng thời giúp phát hiện và khắc phục các lỗi sớm hơn trong quá trình phát triển

Kết quả khi chạy sẽ trả về folder chứa các file kiểm tra code coverage của Todoist. Điều này giúp xác định các phần chưa được kiểm tra hoặc chưa kiểm tra đầy đủ, từ đó có thể quan sát và tăng cường chất lượng cũng như độ tin cậy của Todoist

## All files / src App.js

100% Statements 4/4 100% Branches 3/3 100% Functions 1/1 100% Lines 4/4

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

```
1 import React, { useState } from "react";
2 import PropTypes from "prop-types";
3 import { Header } from "../components/layout/Header";
4 import { Content } from "../components/layout/Content";
5 import { ProjectsProvider, SelectedProjectProvider } from "../context";
6
7 1x export const App = ({ darkModeDefault = false }) => {
8 2x   const [darkMode, setDarkMode] = useState(darkModeDefault);
9
10 2x   return (
11     <SelectedProjectProvider>
12       <ProjectsProvider>
13         <main
14           data-testid="application"
15           className={darkMode ? "darkmode" : undefined}
16         >
17           <Header darkMode={darkMode} setDarkMode={setDarkMode} />
18           <Content />
19         </main>
20       </ProjectsProvider>
21     </SelectedProjectProvider>
22   );
23 };
24
25 1x App.propTypes = {
26   darkModeDefault: PropTypes.bool,
27 };
28
```

Code coverage generated by [istanbul](#) at 2023-05-07T04:46:09.594Z

Hình 3.18: Kết quả sau khi kiểm thử

## KẾT LUẬN

Todoist là một ứng dụng quản lý công việc hiệu quả và đa nền tảng, đã cung cấp một loạt các tính năng hữu ích và giao diện thân thiện người dùng. Đồ án đã trình bày chi tiết về phân tích yêu cầu, thiết kế và triển khai Todoist, cùng với việc sử dụng công nghệ Firebase để xây dựng phần backend

Trong quá trình phát triển Todoist, đã giúp có cái nhìn tổng quan và tìm hiểu và áp dụng các nguyên tắc quản lý dự án và phát triển phần mềm. Sự phân chia rõ ràng và cấu trúc hóa tác vụ đã giúp làm việc một cách có hiệu suất và tổ chức. Sử dụng ReactJS đã giúp tăng tính linh hoạt và khả năng tái sử dụng của mã nguồn.

Sử dụng Firebase làm nền tảng backend đã mang lại nhiều lợi ích cho Todoist. Firebase cung cấp tính năng quản lý người dùng, xác thực, lưu trữ dữ liệu và thông báo đến từng người dùng. Điều này giúp tập trung vào phát triển chức năng chính của ứng dụng mà không phải lo lắng về cơ sở hạ tầng.

Trên cơ sở triển khai và kiểm thử, Todoist đã chứng tỏ tính ổn định và khả năng hoạt động đáng tin cậy. Các tính năng quan trọng như tạo, sửa đổi và xóa task, phân loại task theo dự án và thời gian, cũng như khả năng đánh dấu hoàn thành task đã được triển khai và hoạt động một cách chính xác.

Tuy nhiên, đồ án cũng nhận thấy một số hạn chế của Todoist. Giao diện người dùng còn một số cải thiện để tăng tính trực quan và thân thiện hơn. Ngoài ra, tính năng chia sẻ task và tính năng nhắc nhở có thể được bổ sung để nâng cao trải nghiệm người dùng. Trong quá trình phát triển, đã gặp khó khăn trong việc tối ưu hóa hiệu suất và đảm bảo tính ổn định của ứng dụng

Tổng kết lại, việc thực hiện đồ án Todoist đã đem lại cho nhiều kiến thức và kỹ năng trong việc phát triển ứng dụng web, quản lý dự án và sử dụng công nghệ Firebase. Đồ án cũng giúp nhận thức được các thách thức và cơ hội khi làm việc trên một dự án thực tế. Hy vọng rằng Todoist sẽ tiếp tục được phát triển và cải thiện để mang lại giá trị tốt nhất cho người dùng trong việc quản lý công việc hàng ngày. Một lần nữa, xin cảm ơn các thầy, cô trong khoa đã giúp đỡ, tạo điều kiện trong suốt quá trình học tập.

## TÀI LIỆU THAM KHẢO

Tài liệu tham khảo tại các website như sau:

[1] Tìm hiểu về firebase, hướng dẫn cách làm 1 project ghi chú đơn giản với React Native + Firebase - \_Dugnt

<https://ntdung1128.wordpress.com/2018/09/13/tim-hieu-ve-firebase-huong-dan-cach-lam-1-project-ghi-chu-don-gian-voi-react-native-firebase/>

[2] Viết unit test cho React với React Testing Library - @nguyen.trong.toan

<https://viblo.asia/p/viet-unit-test-cho-react-voi-react-testing-library-Az45brx65xY>

[3] Ứng dụng Todoist

<https://todoist.com/>

[4] Ứng dụng MicrosoftTodo

<https://todo.microsoft.com/>

- Xây Dựng Website với ReactJS – F8  
<https://fullstack.edu.vn/courses/reactjs>
- React Todo App with Firebase – Code Commerce  
<https://www.youtube.com/watch?v=drF8HbnW87w&t=1018s>
- React Testing Library Introduction – The Net Ninja  
<https://www.youtube.com/watch?v=7dTTFW7yACQ>