# **MOVE CARE: Công cụ hỗ trợ giao dịch, tìm kiếm trong kết nối việc làm trên nền tảng Web và ứng dụng di động**

## TRAN THI ANH THI1\*, DOAN THE LONG2, VU QUANG HUY3

*1Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*

*\*tranthianhthi@iuh.edu.vn*

**Tóm tắt.** Nhằm hỗ trợ các ứng cử viên và nhà tuyển dụng dễ dàng kết nối để thực hiện các giao

dịch thỏa thuận, công cụ Move Care ra đời. Công cụ cho phép các nhà tuyển dụng đăng các tin tức tuyển dụng, tìm kiếm các ứng cử viên phù hợp với công việc mà nhà tuyển dụng đưa ra. Công cụ cũng cho phép các ứng cử viên tìm việc có thể đăng ký thông tin cá nhân để tìm việc. Ngoài ra, thông qua Move Care, ứng cử viên có thể được đề xuất với các nhà tuyển dụng có nhu cầu phù hợp với năng lực của ứng cử viên. Hệ thống cũng cho phép nhà tuyển dụng và ứng cử viên trao đổi thông tin qua lại nhằm đạt được thỏa thuận phù hợp. Hệ thống cũng cho phép nhà tuyển dụng và ứng cử viên thiết lập các hợp đồng thỏa thuận mà cả hai đều chấp nhận. Move Care được xây dựng trên hai phân hệ Web và App cho thiết bị di động. Chúng tôi đã sử dụng các công nghệ như NextJS, MUI, React, Native, Django, Restframework để xây dựng cho cả hai phân hệ trên. Chúng tôi còn sử dụng công nghệ mã băm JWT để tăng tính bảo mật hệ thống. Trong tương lai, chúng tôi sẽ mở rộng

**Từ khóa.**Ứng dụng di động, Ứng dụng Web, Bảo mật hệ thống, giao dịch thỏa thuận

**MOVE CARE: Tool to support negotiation transactions in job recruitment based on Web platform and mobile application**

**Abstract.** To help candidates and employers easily connect to carry out agreement transactions, the Move Care tool is built to meet the above requirements. The tool allows employers to post recruitment news and search for suitable candidates for the jobs offered by employers. The tool also allows job candidates to register personal information to search for jobs. In addition, through Move Care, candidates can be proposed to employers whose needs match the candidate's qualifications. The system also allows employers and candidates to exchange information back and forth to reach a suitable agreement. The system also allows employers and candidates to establish contractual agreements that are acceptable to both. Move Care is built on two subsystems: Web and App for mobile devices. We have used technologies such as NextJS, MUI, React Native, Django, and Rest framework to build both of the above modules. We also use JWT hash technology to increase system security. In the future, we will expand functions to meet the needs of employers and job candidates.

**Keywords.** Mobile applications, Web applications, System security,Negotiation transactions, helper,

**1 GIỚI THIỆU**

Xã hội ngày một hiện đại, kéo theo đó là vô vàn các ứng dụng và dịch vụ mới ra đời giúp ích cho con người rất nhiều.Trong đó không thể không kể đến ngành công nghệ thông tin là một ngành khoa học đang trên đà phát triển mạnh và được ứng dụng rộng rãi trong cuộc sống. Có rất nhiều tiện ích mà hiện tại chỉ cần có internet và vài thao tác đơn giản là hoàn thành ngay. Vì thế con người dần càng không thích những cái quá phực tạp, việc chạy đến văn phòng của một công ty mô giới nào đó để tìm kiếm một người phù hợp để hỗ trợ mình dọn dẹp nhà cửa hay nấu ăn quá phức tạp và rất mất thời gian. Nhân ra được nhu cầu đó nhóm chúng em muốn tạo ra một hệ thống quản lý giúp việc không chỉ phục vụ đơn thuần cho việc quản lý nhân sự, hơn thế nữa qua đó phục vụ cho nhu cầu tìm kiếm người giúp việc tùy theo sở thích như theo tháng, theo ngày, nấu ăn, chăm con…. mà luôn an tâm về chất lượng cũng như uy tín của người mình sẽ thuê. Vì sẽ làm việc thông qua phía công ty chứ không phải là một trang bán thông tin hay rao vặt. Tiếp cận với những người có chuyên môn trong lĩnh vực mình tìm kiếm.

Thông qua quá trình phân tích, đánh giá nhu cầu về việc quản lý nhân lực của công ty về mảng dịch vụ giúp việc. Thêm vào đó là cầu nối giữa hai tệp khách hàng người lao động và người cần sử dụng lao động trong lĩnh vực giúp việc. Tạo ra hệ sinh thái gồm ứng dụng và website để đáp ứng được nhu cầu đó. Bằng những công nghệ trong lập trình phát triển phần mềm như NexJS (phía web), React Native( giành cho mobile) còn phía Back-end là Django với mô hình client – server mạnh mẽ. Chúng tôi tin rằng mình có thể chung tay vào bước tiến công nghệ của thời đại, kỷ nguyên số, chỉ cần có điện thoại trong tay, không gì là không thể.

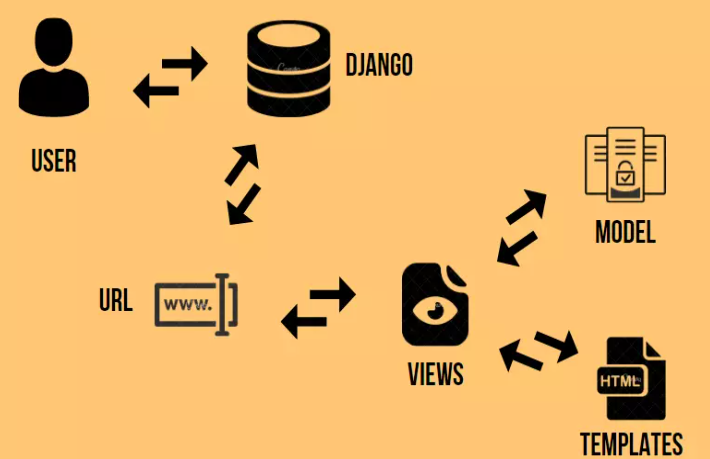
**2 CƠ SỞ LÝ THUYẾT**

**2.1 Django**

Django là một web framework và nó chú trọng vào việc phát triển web. Nó hỗ trợ mọi thứ xung quanh nên chỉ cần tập chung vào viết code.

Web framework này được triển khai dựa trên mô hình MVT với M (Model): dữ liệu muốn trình bày , V (View): xử lý các request và trả về nội dung kèm template, T (Template): tập html chứa bố cục web ta muốn xây dựng.

Ngoài ra web framework này tuân thủ 2 nguyên tắc DRY (Don’t repeat yourself) và KISS (Keep it short and simple) vì vậy nó sẽ giúp nhà phát triển thiết kế một cách nhanh chóng, hỗ trợ các thư viện/module cần thiết, đảm bảo bảo mật, khả năng mở rộng tốt và tính linh hoạt. [1]

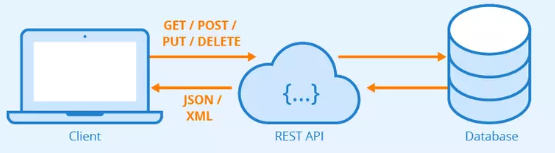


Hình 2.1 Mô hình MVT của Django.[2]

**2.2 Django REST framework**

Django REST framework là một công cụ hỗ trợ đắc lực trong việc xây dựng WebAPI.

Một vài lý do nên sử dụng REST: Hỗ trợ dữ liệu ORM và non-ORM với Serialization. Tài liệu phong phú, cộng đồng hỗ trợ lớn. Được sử dụng trong các công ty quốc tế như Mozilla, Red Hat, Heroku.[3]



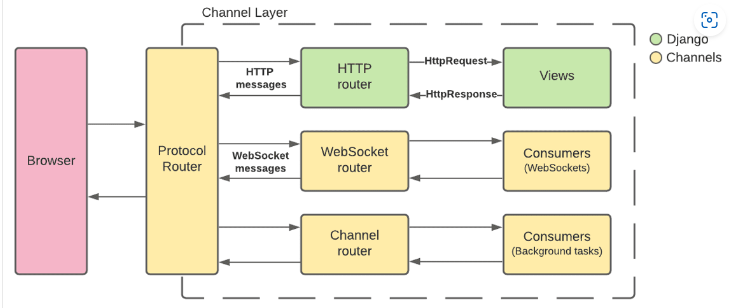
Hình 2.2 Cách hoạt động của Restful.[3]

**2.3 Django Channels**

Channels kết hợp hỗ trợ chế độ xem không đồng bộ gốc của Django, cho phép các dự án của Django xử lý không chỉ HTTP mà còn các giao thức yêu cầu kết nối lâu dài (WebSockets, MQTT, chatbots, v.v ).

Nó thực hiện điều này trong khi vẫn bảo toàn tính chất đồng bộ và dễ sử dụng của Django, cho phép bạn chọn cách bạn viết mã của mình.

Nó cung cấp tích hợp với hệ thống xác thực của Django, hệ thống phiên và hơn thế nữa, giúp việc mở rộng dự án chỉ HTTP của bạn sang các giao thức khác dễ dàng hơn bao giờ hết.[4]



Hình 2.3 Thiết lập channels cơ bản.[5]

**2.4 JSON Web Token ( JWT )**

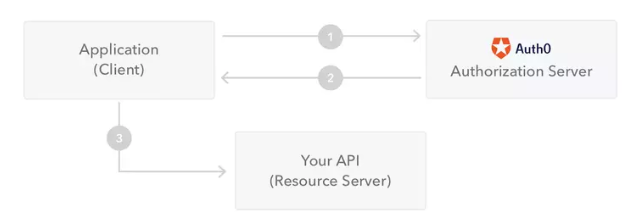
JWT là một tiêu chuẩn mở (RFC 7519) nhằm xác minh thông tin an toàn giữa các bên Client-Server dưới dạng JSON object. JWT có thể được ký bằng cách sử dụng một secret (với thuật toán HMAC) hoặc cặp public/private key dùng chuẩn RSA hoặc ECDSA.

JSON Web Tokens bao gồm 3 phần được phân tách bằng dấu chấm (.):, Header, Payload, Signature.

Với Header: chứa 2 phần là mã token (JWT) và thuật toán (HMAC, SHA256 hoặc RSA).

Payload: chứa các claims, 1 claims thường chứa typically, thông tin user và các dữ liệu bổ sung. Có 3 loại claims: registered, public, và private claims.

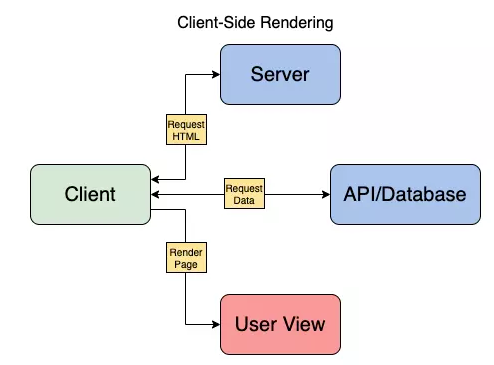
Signature: chứa header được mã hóa, payload được mã hóa, một secret, thuật toán được chỉ định trong header và sign.[6]



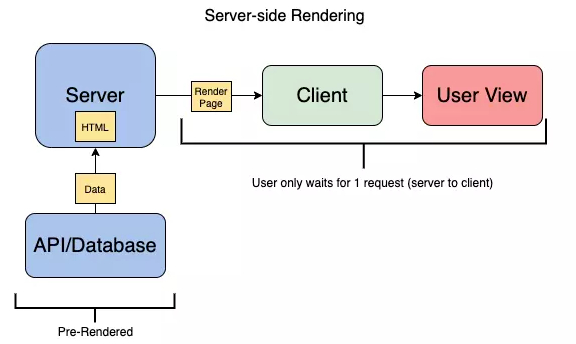
Hình 2.4 Cách JWT được lấy và sử dụng để truy cập API hoặc resource.[6]

**2.5 Next.js**

Next.js là một open-source React front-end framework được bổ sung các tính năng như Server Side Rendering (SSR) và Static Site Generation (SSG). Next.js được xây dựng dựa trên thư viện React.[7]



Hình 2.5.1 Phía Client.[7]



Hình 2.5.2 Phía server.[7]

**2.6 Material UI**

Material UI là một thư viện các React Component đã được tích hợp thêm cả Google's Material Design.

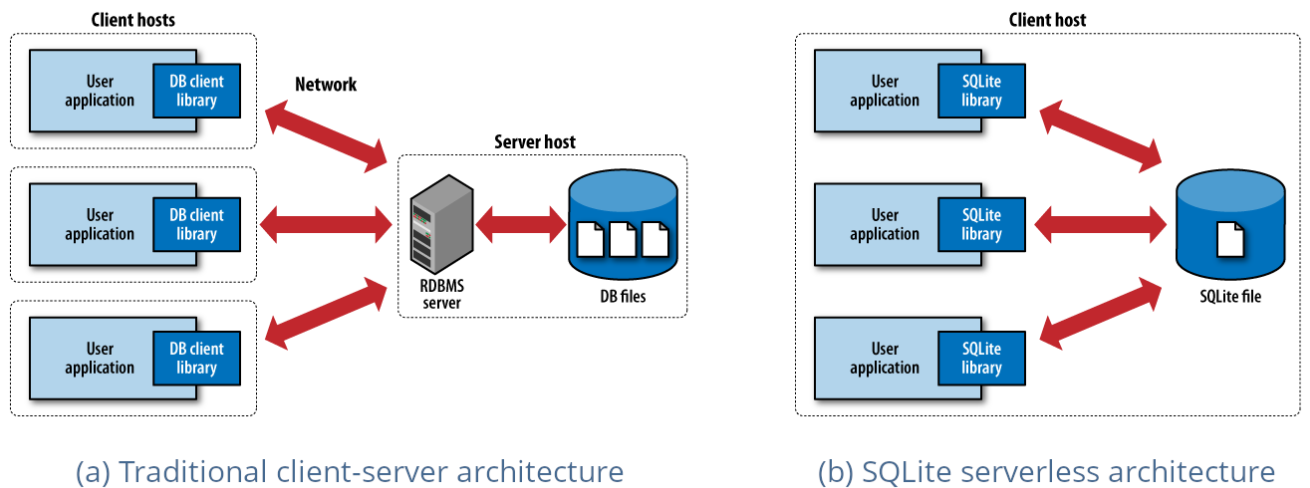
Material UI đem đến một giao diện hoàn toàn mới, với những button, textfield, toogle... được design theo một phong cách mới lạ, thay vì việc nhà nhà người người dùng Bootstrap như hiện nay.[8]

**2.7 React Native**

React Native là một framework mã nguồn mở được sáng tạo bởi Facebook. Nó được sử dụng để phát triển ứng dụng di động Android, iOS, Web và UWP bằng cách cho phép các nhà phát triển sử dụng React cùng với môi trường ứng dụng gốc. [9]

**2.8 SQLite**

SQLite là hệ quả trị cơ sở dữ liệu (DBMS) quan hệ tương tự như Mysql, ... Đặc điểm nổi bật của SQLite so với các DBMS khác là gọn, nhẹ, đơn giản, đặt biệt không cần mô hình server-client, không cần cài đặt, cấu hình hay khởi động nên không có khái niệm user, password hay quyền hạn trong SQLite Database. Dữ liệu cũng được lưu ở một file duy nhất. [10]

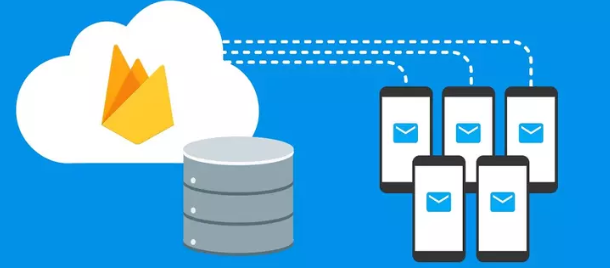
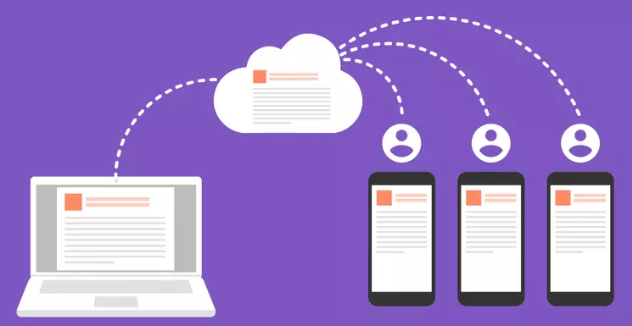


Hình 2.8.1 sự khác nhau giữa Sqlite và hệ quản trị cơ sở dữ liệu client-server truyền thống[1]

**2.9 Firebase**

Firebase ra đời dưới dạng Backend-as-a-Service với chức năng thời gian thực. Sau khi được Google mua lại vào năm 2014 thì Firebase nhanh chóng phát triển thành nền tảng phát triển ứng dụng đa năng của di động và website như ngày nay. Nền tảng này là sự kết hợp giữa cloud với hệ thống máy chủ của Google để tập trung chính cho 2 đối tượng là: Develop & test your app và Grow & engage your audience.

Firebase cung cấp cho chúng ta những API đơn giản, mạnh mẽ và đa nền tảng trong việc quản lý, sử dụng database, bởi vậy giờ đây chúng ta chỉ cần gọi API và phần server đã có Firebase lo12]



Hình 2.9.1 Realtime database.[12] Hình 2.9.2 Authentication.[12]



Hình 2.9.3 Firebase cloud messaging .[12] Hình 2.9.4 Remote Config.[12]

**2.10 Vercel**

Vercel là một nền tảng đám mây (cloud platform) được tạo ra để phục vụ việc phát triển và triển khai ứng dụng web nhanh chóng và dễ dàng. Nền tảng này cho phép bạn xây dựng, triển khai và quản lý các ứng dụng web mà không cần quan tâm đến việc cấu hình hệ thống máy chủ.

Vercel hỗ trợ các ứng dụng web được viết bằng nhiều ngôn ngữ lập trình, như JavaScript, TypeScript, Python, Go và nhiều ngôn ngữ khác.[13]

**3 MOVE CARE: Hỗ trợ kết nối và quản lý trong lĩnh vực giúp việc**

**3.1 Các thành phần trong hệ thống**

Hệ thống của MOVE CARE được thiết kế theo mô hình Client - Server, trong đó, phía Server được thiết kế theo bằng ngôn ngữ Python sử dụng Django và phía Front-End sử dụng NextJS còn phía di động sẽ dùng React Native lập trình đa nền tảng (*Hình 3.1*).

![A diagram of a software company

Description automatically generated

Hình 3.1: Kiến trúc hệ thống MOVE CARE

* 1. **Sơ lược công nghệ sử dụng**

Về phía Frontend chúng tôi sử dụng NextJS một framework sử dụng ngôn ngữ Javascipt, các tính năng như server-side rendering, static site generation, code splitting và pre-fetching, tất cả đều tích hợp sẵn trong framework giúp xây dựng hệ thống một cách nhanh chóng. Về Backend, chúng tôi sử dụng framework NestJS hỗ trợ một số tính năng hữu ích như Middleware, Exception Filters, Authentication và Authorization, WebSocket.



Hình 3.2: Sơ đồ Use case hệ thống MOVE CARE

**3.3 Phân tích thiết kế hệ thống**

Hệ thống hỗ trợ cho hai đối tượng chính gồm:

+ Nhân viên: hỗ trợ quản lý người dùng, và quản lý các bài viết, giao dịch

+ Khách hàng:

a. Người thuê giúp việc: hỗ trợ người thuê đặt những dịch vụ có sẵn hoặc tự tạo ra bài viết mới dựa trên nhu cầu riêng của bạn thân.

b. Người làm việc: nhận những công việc mà người thuê đặt, và cũng có thể đăng bài để tự tìm công việc phù hợp với mình

A diagram of a computer program

Description automatically generated with medium confidence

Hình 3.3 Sơ đồ lớp của MOVE CARE

**3.4 Mô hình triển khai**

Vercel được thiết kế để cung cấp tốc độ và hiệu suất cao nhất cho các ứng dụng web và trang web tĩnh. Nó sử dụng một mạng lưới các máy chủ toàn cầu để đưa nội dung gần hơn với người dùng cuối và tối ưu hóa tốc độ tải trang web. Nó còn tự động CI/CD cơ bản, cập nhật mã nguồn từ github.Trong mô hình triển khai cho MOVE CARE, chúng tôi sử dụng Vercel để triển khai mô hình web sử dụng NextJS (*Hình 3.4*)

A diagram of a computer

Description automatically generated

Hình 3.4.1: Sơ đồ triển khai Front-end

A diagram of a software

Description automatically generated

Hình 3.4.2: Sơ đồ triển khai Backend

Về phần Back-end chúng tôi sử dụng công nghê Django để cung cấp và xử lý các yêu cầu từ phía

Front-end thông qua API, ở phần tương tác với cơ sở dữ liệu chúng tôi có sử dụng Django Rest Framework, một công cụ mạnh mẽ khi làm việc với các cơ sở dữ liệu và tạo ra các API với số lượng mã nguồn ít nhất. sau đó sử dụng PythonAnywhere triển khai lên server ( Hình 3.5)

* 1. **Giao diện của MOVE CARE** 
     1. **WEB**

|  |  |
| --- | --- |
| Hình 3.5.1.1: Trang chủ | Hình 3.5.1.3: Danh sách các bài viết |
| Hình 3.5.*1.*2: màn hình thống kê | |

Với giao diện web hiện đại được thiết kế theo phong cách tối giản, có thể hỗ trợ cho nhân viên, người làm trong bộ phận quản lý và vận hành dễ dàng làm quen và sử dụng, bên cạnh đó chúng tôi còn rất chú trọng đến trải nghiệm cá nhân của người dùng, có thể tùy chỉnh chế độ sáng/tối, màu sắc hay bố cục của trang web theo ý muốn.

**3.5.2 MOBILE**

|  |  |  |
| --- | --- | --- |
| **A screenshot of a phone number  Description automatically generated**  Hình 3.5.2.1: Màn hình đăng nhập | **A person holding a camera  Description automatically generated**  Hình 3.5.2.2: Màn hình chính | **A screenshot of a phone  Description automatically generated**  Hình 3.5.2.3: Màn hình giao dịch |
| **A screenshot of a phone  Description automatically generated**  Hình 3.5.2.4: Màn hình đăng bài | Hình 3.5.2.5: Màn hình thông báo | **A screenshot of a phone  Description automatically generated**  Hình 3.5.2.6: Màn hình bài đăng |

Về phần ứng dụng, chúng tôi triển khai bằng React native hỗ trợ lập trình di động đa nền tảng cả Android và IOS để tiệm cận với tệp khách hàng lớn hơn, với giao diện đơn giản và các Icon bắt mắt, dễ dàng sử dụng, kể cả với những khách hàng lớn tuổi ít tiếp xúc với công nghệ .

**4 TRIỂN KHAI THỰC TẾ**

**4.1 Cấu hình phần cứng, phần mềm**

Yêu cầu phần cứng: Ram 1GB, SSD 4GB, hệ điều hành Window 10+

Phần mềm: sử dụng Python 3.9.7, Nodejs v18.18.2

**4.2 Giao diện hiện thực màn hình**

Để thuận tiện cho việc quản lý đăng bài, quản lý tài khoản, v.v. Chúng tôi thiết kế màn hình chức năng hiển thị chung để dễ quan sát và thao tác với chức năng chính là đăng bài.

Tiếp theo chúng tôi thiết kế quy trình đăng bài với 2 dạng người sử dụng là người cần thuê người giúp việc và người giúp việc.

|  |  |
| --- | --- |
| Hình 4.2.1: Có thể chọn những dịch vụ có sẵn | Hình 4.2.2: hoặc tự tạo bài viết theo nhu cầu của bản thân |
| Hình 4.2.3 Bài viết sẽ được kiểm duyệt bởi nhân viên | |
| Hình 4.2.4: Chi tiết của giao dịch cũng sẽ được nhân viên quản lý và giám sát | |

**4.3 Kiểm thử chức năng**

Chúng tôi đã tiến hành kiểm thử phần mềm bằng phương pháp kiếm thử hộp đên với một số Test Case cơ bản.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Test Case name** | **Input** | **Kết quả mong đợi** | **Kết quả thực tế** |
| 1 | Đăng nhập | TK và MK đúng | Đăng nhập thành công | Đăng nhập thành công |
| 2 | Đăng bài | Nhập thiếu thông tin | Báo cáo lỗi thiếu thông tin | Báo cáo lỗi thiếu thông tin |
| 3 | Đăng nhập ở Web với tài khoản của khách hàng | TK và MK của khách hàng | Đăng nhập không thành công | Đăng nhập không thành công |
| 4 | Đặt dịch vụ từ bài viết tìm việc | ấn vào bài viết sau khi đăng nhập | Đưa đến màn hình tiến hành giao dịch | Đưa đến màn hình tiến hành giao dịch |
| 5 | Hủy dịch vụ đã đặt | Đăng nhập thành công với tài khoản đã đặt dịch vụ | Hủy thành công | Hủy thành công |

**5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

Bài báo này, chúng tôi viết để trình bày giải pháp giúp cho việc kết nối giữa những người cần tìm người giúp việc với những người có khả năng cung cấp dịch vụ hỗ trợ giúp việc. MoveCare là một ứng dụng đa nền tảng (web, Android, iOS) được thiết kế với cơ sở dữ liệu là SQLite, MoveCare cung cấp một nền tảng linh hoạt và tiện lợi để tạo và quản lý các yêu cầu hỗ trợ công việc. Chức năng chính của MoveCare là cho phép người cần tìm người giúp việc đăng bài tìm kiếm theo các yêu cầu cụ thể, bao gồm thời gian và loại công việc cần hỗ trợ. Người giúp việc có thể dễ dàng đăng bài để cung cấp thông tin về khả năng giúp việc của mình và sẵn sàng làm việc vào các thời điểm cụ thể. Ngoài ra, MoveCare cũng cung cấp tính năng chat trực tuyến để người dùng có thể trao đổi và nhận được hỗ trợ từ nhân viên hỗ trợ. Điều này giúp tăng cường sự thuận tiện và tương tác giữa người dùng và nhà cung cấp dịch vụ. Với giao diện thân thiện và tính năng linh hoạt trên các nền tảng web, Android và iOS, MoveCare hứa hẹn là một công cụ hữu ích giúp nâng cao sự tiện lợi trong việc tìm kiếm và cung cấp dịch vụ hỗ trợ giúp việc, đáp ứng nhu cầu của cả người cần tìm người giúp việc và những người có khả năng cung cấp dịch vụ. Tuy nhiên do sự hạn chế về thời gian và sự khảo sát chưa đầy đủ nên phần mềm còn gặp nhiều hạn chế. Một số quy trình nghiệp vụ còn chưa bám sát với thực tế. Trong tương lai chúng tôi sẽ tìm hiểu kỹ hơn về quy trình nghiệp vụ cũng như phát triển MoveCare trở thành một phần mềm được áp dụng rộng rãi hơn.

**TÀI LIỆU THAM KHẢO**

[1]. Tìm hiểu về Django - framework hỗ trợ Python trong lập trình web - <https://viblo.asia/p/tim-hieu-ve-django-framework-ho-tro-python-trong-lap-trinh-web-QpmlexbkZrd> accessed 15/01/2020

[2]. Django là gì? Hướng dẫn cài đặt Django dành cho người mới - <https://vietnix.vn/django-la-gi/> accessed 08/07/2022

[3]. Viết API sử dụng Django REST framework với PostgreSQL - <https://viblo.asia/p/viet-api-su-dung-django-rest-framework-voi-postgresql-RQqKL9r4Z7z> accessed 14/02/2020

[4]. Django channels hướng dẫn cơ bản - <https://aithietke.com/django-channels-huong-dan-co-ban/> accessed 27/10/2021

[5]. Introduction to Django Channels - <https://testdriven.io/blog/django-channels/> accessed 08/02/2023

[6]. JSON Web Token là gì? - <https://viblo.asia/p/json-web-token-la-gi-aWj533go56m> accessed 23/06/2020

[7]. Tìm hiểu về Next.js (Phần 1) - <https://viblo.asia/p/tim-hieu-ve-nextjs-phan-1-V3m5WQkwZO7> accessed 20/09/2021

[8]. Tìm hiểu về Material UI - [https://viblo.asia/p/tim-hieu-ve-material-ui](https://viblo.asia/p/tim-hieu-ve-material-ui-jdWrvwg8vw38) accessed 28/06/2015

[9]. Tổng quan về React Native - [https://viblo.asia/p/tong-quan-ve-react-native](https://viblo.asia/p/tong-quan-ve-react-native-gwd43kOr4X9) accessed 27/01/2023

[10]. SQLite là gì? - <https://viblo.asia/p/sqlite-la-gi-E375zVVR5GW> accessed 20/05/2021

[11]. Django (4) - Làm việc với cơ sở dữ liệu - <https://legiacong.blogspot.com/2021/05/django-4-lam-viec-voi-co-so-du-lieu.html> accessed 04/05/2021

[12]. Tìm hiểu sơ lược về Firebase - <https://viblo.asia/p/tim-hieu-so-luoc-ve-firebase-Eb85oeOmZ2G> accessed 21/03/2021

[13]. Auto Deploy NodeJS App Using Vercel - [https://viblo.asia/p/auto-deploy-nodejs-app-using-vercel](https://viblo.asia/p/auto-deploy-nodejs-app-using-vercel-EbNVQx5oLvR) accessed 30/05/2023