
Report #3: Final Report

PROJECT JOBIT

**Introduction to Software Engineering
(CSC13002)**

**GVHD: Cô Nguyễn Thị Minh Tuyền
GVTH: Thầy Nguyễn Đức Huy**

Group name: Group 18 - Team 32GB

STT	Họ và Tên	MSSV
1	Trần Phú Nguyên (NT)	1612448
2	Trần Ngô Anh Nguyên	1612440
3	Nguyễn Lê Thị Mỹ Linh	1612334
4	Trần Thị Lạng	1612331
5	Trần Thị Lý	1612367

Ho Chi Minh City, 29/12/2018

Report 3# Final Report

Revision History

Date	Version	Description	Author
<29/11/2018>	<1.0>	<Final Report>	<Nguyễn Lê Thị Mỹ Linh>

Individual Contributions Breakdown

MSSV	Họ tên	Vai trò	Phần trăm đóng góp
1612448	Trần Phú Nguyên	Project Manager Developer Designer	100%
1612440	Trần Ngô Anh Nguyên	Developer Designer	90%
1612334	Nguyễn Lê Thị Mỹ Linh	Developer Designer	90%
1612331	Trần Thị Lạng	Developer Tester Designer	90%
1612367	Trần Thị Lý	Developer Designer	90%

Table of Contents

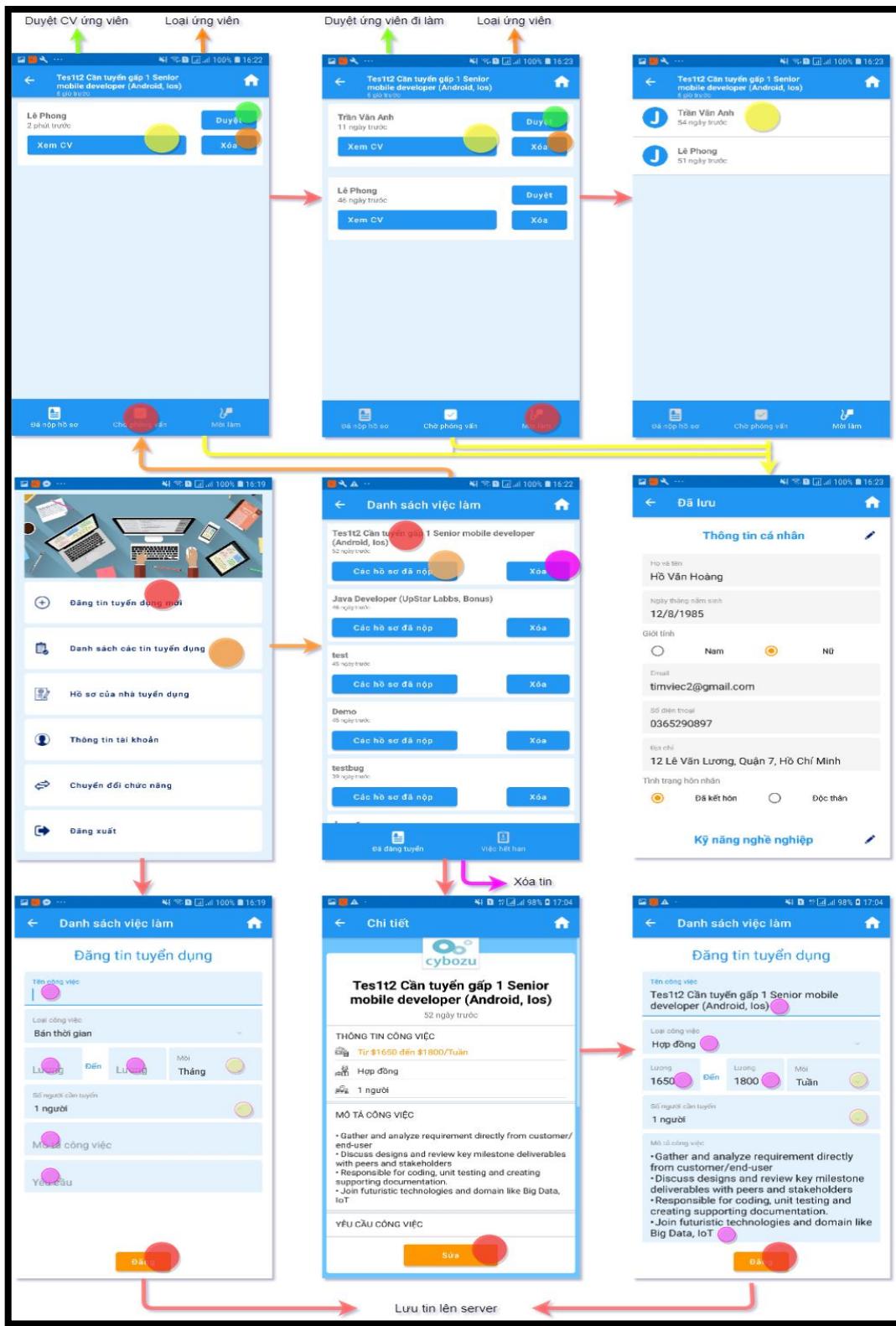
Nội dung

I. Summary of Changes	6
II. Customer Statement of Requirements	6
1. REQ-0: Người dùng	6
2. REQ-1: Người tìm việc	7
3. REQ-2: Nhà tuyển dụng	8
4. REQ-3: Admin	9
III. Glossary of Terms	10
IV. System Requirements	11
V. Functional Requirements Specification	16
1. Stakeholders	16
2. Actors and Goals	16
3. Use case	17
a. Casual Description	17
b. Use Case Diagram	25
c. Traceability Matrix	26
d. Use case Specification	27
4. System Sequence Diagrams	31
VII. Class Diagram and Interface Specification	33
1. Người Tìm Việc Và Nhà Tuyển Dụng	33
2. Admin	43
VIII. System Architecture and System Design	46
1. Architectural Styles	46
2. Identifying Subsystems	46
3. Mapping Subsystems to Hardware	47
4. Persistent Data Storage	47

Report 3# Final Report

5. Network Protocol.....	47
6. Global Control Flow.....	47
7. Hardware Requirements.....	48
IX. Algorithms and Data Structures(if applicable).....	49
1. Algorithms	49
2. Data Structures	49
X. User Interface Design and Implementation	50
1. Structure and Navigation	50
2. Specification	61

Report 3# Final Report



..61

XI. Design of Tests	62
1. List Test case.....	62
2. Phạm vi test.....	64

Report 3# Final Report

3. Kế hoạch test.....	65
XII. History of Work, Current Status, and Future Work	65
XIII. References.....	69

I. Summary of Changes

- Không thay đổi

II. Customer Statement of Requirements

1. REQ-0: Người dùng

ST T	ID	Chức năng	Mô tả
1	REQ-0.1	Đăng ký tài khoản	Người dùng có thể đăng ký bằng cách điền đầy đủ thông tin.
2	REQ-0.2	Đăng nhập	Người dùng đăng nhập vào app khi có tài khoản sẵn (tài khoản đã tồn tại trên firebase) - Người tìm việc: Có thể không đăng nhập vẫn có thể sử dụng ứng dụng nhưng chỉ có thể tìm kiếm và xem thông tin các công việc, không thể

Report 3# Final Report

			<p>apply. Để apply công việc hoặc xem hồ sơ, CV thì phải đăng nhập nếu đã có tài khoản hoặc đăng ký nếu chưa có tài khoản.</p> <ul style="list-style-type: none"> - Nhà tuyển dụng: Bắt buộc phải đăng nhập trước khi tạo thông tin công ty để tuyển dụng - Admin: Bắt buộc phải đăng nhập mới có thể thực hiện được các chức năng của Admin.
3	REQ-0.3	Đăng xuất	<p>Người dùng có thể đăng xuất tài khoản đang sử dụng và đăng nhập bằng tài khoản khác hoặc là đăng ký một tài khoản mới</p> <ul style="list-style-type: none"> - Admin: Chỉ có thể đăng nhập và sử dụng 1 tài khoản, không thể đăng nhập hoặc đăng ký bằng tài khoản khác.

2. REQ-1: Người tìm việc

ST T	ID	Tên chức năng	Mô tả
1	REQ-1.1	Tạo CV	Người dùng có thể tạo CV mới để có thể Apply vào công ty hoặc có thể đăng tải CV nếu có sẵn CV.
2	REQ-1.2	Edit CV	Người dùng có thể chỉnh sửa thông tin CV của mình để phù hợp với các vị trí ứng tuyển khác nhau.
3	REQ-1.3	Tìm kiếm công việc	Người dùng có thể tìm kiếm công việc bằng cách nhập tên công việc, chức danh hoặc từ khóa công ty và địa điểm làm việc để tìm kiếm công việc. Sau đó nhấn nút Tìm Việc.
4	REQ-1.4	Đánh giá nhà tuyển dụng	Người dùng có thể đánh giá nhà tuyển dụng tùy theo mức độ hài lòng của mình về nhà tuyển dụng.
5	REQ-1.5	Xóa tin	Người dùng có thể xóa tin mà mình đã lưu trong mục việc làm của tôi.
6	REQ-1.6	Apply	Người dùng có thể Apply một công việc mà mình muốn nhưng để Apply được công việc thì bắt buộc

Report 3# Final Report

			người dùng đã đăng nhập vào hệ thống và đã có CV rồi. Khi Apply thành công, thông tin công việc sẽ được lưu vào mục Các công việc đã Apply
7	REQ-1.7	Xem thông tin nhà tuyển dụng	Người dùng có thể xem thông tin công ty khi người dùng muốn tìm hiểu thêm thông tin về công ty

3. REQ-2: Nhà tuyển dụng

ST T	ID	Chức năng	Mô tả
1	REQ-2.1	Đăng ký thông tin công ty	Nhà tuyển dụng phải đăng ký thông tin công ty trước khi sử dụng các chức năng khác như đăng tin, xem thông tin ứng viên.
2	REQ-2.2	Chỉnh sửa thông tin công ty	Nhà tuyển dụng có thể chỉnh sửa (thêm, xóa) các thông tin cần thiết của công ty sao cho thu hút, phù hợp với nhiều ứng viên
3	REQ-2.3	Đăng tin tuyển dụng	Nhà tuyển dụng có thể đăng tin tuyển dụng cho các vị trí trong công ty
4	REQ-2.4	Gõ tin tuyển dụng	Nhà tuyển dụng có thể gõ tin đã đăng nếu như tin đã đủ người Apply hoặc tin đã hết hạn hoặc có sự thay đổi gì trong công việc.
5	REQ-2.5	Xem tin đã đăng	Nhà tuyển dụng có thể xem các tin tuyển dụng đã đăng, người tuyển dụng cũng có thể sửa các tin này để phù hợp với yêu cầu của công ty nếu có thay đổi.
6	REQ-2.6	Xem danh sách ứng viên	Nhà tuyển dụng được quyền xem danh sách các ứng viên đã nộp CV vào công ty.
7	REQ-2.7	Duyệt ứng viên	Nhà tuyển dụng có thể chọn Duyệt nếu thấy ứng viên đó phù hợp hoặc có thể Xóa thông tin ứng viên nếu đã đủ số lượng tuyển dụng, hoặc ứng viên không thích hợp.

4. REQ-3: Admin

ST T	ID	Chức năng	Mô tả
1	REQ-3.1	Cảnh cáo	<ul style="list-style-type: none"> - Khi người tìm việc cảnh cáo về bình luận của một người tìm việc khác về công ty tuyển dụng, cảnh cáo này được gửi về cho admin, nó nằm trong phần cảnh cáo người tìm việc. - Người tìm việc cũng có thể cảnh cáo nhà tuyển dụng. Cảnh cáo này được gửi về cho admin và nằm trong phần cảnh cáo nhà tuyển dụng. - Với mỗi cảnh cáo, admin có thể biết được ai là người cảnh cáo, ai là người bị cảnh cáo, nội dung cảnh cáo, lịch sử bị cảnh cáo và hồ sơ của người đang bị cảnh cáo. - Admin có thể gửi thông báo cảnh cáo đến người bị cảnh cáo và có thể xóa hồ sơ của người bị cảnh cáo.
2	REQ-3.2	Duyệt hồ sơ nhà tuyển dụng	<p>Sau khi nhà tuyển dụng hoàn tất hồ sơ và lưu, hồ sơ này sẽ được gửi đến cho admin để admin phê duyệt:</p> <ul style="list-style-type: none"> - Admin có thể bấm vào “hồ sơ cần phê duyệt” để xem chi tiết hồ sơ của nhà tuyển dụng. Khi đó admin có quyền duyệt hoặc không duyệt hồ sơ này. - Nhà tuyển dụng không được đăng tin tuyển dụng khi admin chưa duyệt hồ sơ hoặc bấm “Hủy” trong phần

Report 3# Final Report

			phê duyệt hồ sơ. Trường hợp admin đã bấm “Duyệt” trong phần phê duyệt hồ sơ thì nhà tuyển dụng có tất cả các quyền như nhà tuyển dụng bình thường.
3	REQ-3.3	Xem thống kê số lượng user	Admin sẽ biết được số lượng người tìm việc và nhà tuyển dụng sử dụng app. Admin chỉ có thể xem, không thể chỉnh sửa số liệu thống kê
4	REQ-3.4	Xem hồ sơ user	Admin có quyền xem danh sách các user và chỉ được xem thông tin profile user (email, name, avatar). Admin không có quyền xem các thông tin cá nhân khác.
5	REQ-3.5	Xóa account user	Khi nhà tuyển dụng bị người tìm việc bị cảnh cáo quá 3 lần (số lần cảnh cáo này là cảnh cáo từ user lẫn admin), admin có quyền xóa account này. Nếu số lần bị cảnh cáo < 3 thì admin không có quyền xóa hồ sơ này.

III. Glossary of Terms

- Firebase: Dùng để lưu trữ dữ liệu, xác thực người dùng
- Firebase Realtime Database: Lưu trữ dữ liệu dưới dạng JSON và đồng bộ realtime đến mọi kết nối client.
- Admin: Người duyệt hồ sơ công ty và cảnh cáo những user bị tố cáo
- Use case là đối tượng người dùng muốn nhận được từ hệ thống. Nó được đặt tên giống Động từ hoặc Động từ + cụm danh từ.
- Class Diagram là một bản vẽ quan trọng của thiết kế phần mềm, nó cho thấy cấu trúc và quan hệ giữa các thành phần tạo nên phần mềm
- Activity: là một thành phần ứng dụng cung cấp một màn hình mà với nó người dùng có thể tương tác để thực hiện một điều gì đó, chẳng hạn như quay số điện thoại, chụp ảnh, gửi gmail hoặc xem bản đồ. Mỗi hoạt động được cho trong một cửa sổ là nơi để vẽ giao diện người dùng của nó. Cửa

Report 3# Final Report

số này thường lắp đầy màn hình, nhưng có thể nhỏ hơn màn hình và nổi bên trên các cửa sổ khác

- MVP: MVP là một mô hình kiến trúc hướng giao diện người dùng, được thiết kế để tạo thuận lợi cho việc kiểm thử đơn vị (unit testing) và tăng tính tách biệt giữa tầng dữ liệu và tầng hiển thị dữ liệu trong mô hình MVC
- Client-Server: Là mô hình mà trong đó các máy chủ (server) cung cấp và quản lý các nguồn lực và dịch vụ cho máy khách (client)

IV. System Requirements

STT	ID	Chức năng	Mô tả	Priority Weight
-----	----	--------------	-------	--------------------

Report 3# Final Report

1	REQ -1.6	Apply vào công việc cần ứng tuyển	<ul style="list-style-type: none"> ● Đã đăng nhập vào ứng dụng ● Nếu chưa đăng nhập thì khi nhấn nút Apply sẽ hiện ra trang đăng nhập/đăng ký: <ul style="list-style-type: none"> · Nếu đã có tài khoản rồi có thể đăng nhập vào hoặc có thể đăng ký một tài khoản mới tùy vào mục đích sử dụng · Nếu chưa có tài khoản phải đăng ký một tài khoản qua Gmail hoặc liên kết với Facebook, Google. ● Đã có CV, nếu không có CV: <ul style="list-style-type: none"> · Tạo một CV mới · Tải lên một CV đã có sẵn ● Tin ứng tuyển có tồn tại mới có thể apply. 	0
2	REQ -2.3	Đăng tin tuyển dụng	<ul style="list-style-type: none"> ● Phải đăng nhập và tạo thông tin công ty mới có thể đăng tin. ● Phải điền đầy đủ thông tin cần thiết khi đăng tin tuyển dụng thì tin mới được duyệt 	1
3	REQ -2.7	Duyệt ứng viên	<ul style="list-style-type: none"> ● Phải có ứng viên apply vào vị trí cần tuyển. ● Khi bấm vào nút “Duyệt”, hồ sơ ứng viên sẽ được chuyển qua mục chờ phỏng vấn và hệ thống sẽ gửi thông báo về cho người tìm việc: Hồ sơ đã được chấp nhận. ● Khi bấm “Xóa” thì sẽ có thông báo về cho người tìm việc (người mà đã apply vào) và công việc này vẫn nằm trong mục đã apply của người tìm việc nhưng cập nhật lại tình trạng là: đã bị từ chối. Lúc này, người tìm việc có thể xóa công việc này, không lưu nó trong mục công việc đã apply của mình nữa. 	2

Report 3# Final Report

4	REQ -1.1	Tạo CV	<ul style="list-style-type: none"> • Cần phải đăng nhập mới có thể tạo CV • Nếu chưa đăng nhập thì khi bấm vào Tạo CV, sẽ hiện ra trang đăng nhập/đăng ký. • Nếu đã có tài khoản rồi có thể đăng nhập vào hoặc có thể đăng ký một tài khoản mới tùy vào mục đích sử dụng • Nếu chưa có tài khoản phải đăng ký một tài khoản qua Gmail hoặc liên kết với Facebook, Google. 	3
5	REQ -3.2	Duyệt hồ sơ nhà tuyển dụng	<ul style="list-style-type: none"> • Nếu admin bấm “Duyệt” hồ sơ thì hệ thống sẽ đưa thông tin hồ sơ này từ mục “Hồ sơ cần phê duyệt” sang mục “Nhà tuyển dụng” và gửi thông báo đã được phê duyệt về cho nhà tuyển dụng bằng Firebase Cloud Message. • Nếu admin bấm “Hủy” thì hệ thống sẽ xóa hồ sơ này trong phần “chờ phê duyệt” và gửi thông báo về cho nhà tuyển dụng. Nhà tuyển dụng có thể tiếp tục gửi yêu cầu phê duyệt lên cho admin. 	4
6	REQ -1.3	Tìm kiếm công việc	<ul style="list-style-type: none"> • Cần phải có thông tin trong ô tìm kiếm hoặc địa điểm mới có thể tìm kiếm công việc • Khi 2 ô tìm kiếm hoặc địa điểm có dữ liệu thì khi nhấn nút Tìm Việc mới hiện ra danh sách công việc tương ứng. 	5
7	REQ -3.1	Cảnh cáo	<ul style="list-style-type: none"> • Nếu nhà tuyển dụng bấm vào mục gửi cảnh cáo về cho user thì hệ thống sẽ tăng số lần bị cảnh cáo của user lên 1 và thực hiện gửi thông báo cảnh cáo bằng Firebase Cloud Message. 	6

Report 3# Final Report

8	REQ -2.1	Đăng ký thông tin công ty	<ul style="list-style-type: none"> ● Phải đăng nhập vào ứng dụng thì mới có thể đăng ký thông tin công ty. ● Chỉ khi điền đầy đủ các thông tin trong phần đăng ký thông tin thì mới có thể đăng ký thành công. 	7
9	REQ -1.2	Sửa CV	<ul style="list-style-type: none"> ● Cần phải tồn tại CV mới có thể sửa. ● Nút sửa CV chỉ hiện lên khi đã tồn tại CV ● Sau khi sửa CV và lưu thì hệ thống sẽ cập nhật lại CV trên database 	8
10	REQ -2.6	Xem danh sách ứng viên	<ul style="list-style-type: none"> ● Phải có ứng viên apply vào công việc ● Có thể duyệt hồ sơ của ứng viên hoặc xóa hồ sơ ● Khi bấm xóa thì sẽ có thông báo về cho người tìm việc (người mà đã apply vào) và công việc này vẫn nằm trong mục đã apply của người tìm việc nhưng cập nhật lại tình trạng là: đã bị từ chối. Lúc này, người tìm việc có thể xóa công việc này, không lưu nó trong mục công việc đã apply của mình nữa. ● Có thể xem CV của ứng viên đã apply vào công việc. 	9
11	REQ -2.4	Gõ tin tuyển dụng	<ul style="list-style-type: none"> ● Tin tuyển dụng này còn tồn tại thì mới được quyền gõ tin. ● Nếu tin tuyển dụng này đã có người apply mà chưa được phê duyệt thì sẽ có thông báo về cho người tìm việc (người mà đã apply vào) là tin này đã bị hết hạn và công việc này vẫn nằm trong mục đã apply của người tìm việc nhưng cập nhật lại tình trạng là: tin tuyển dụng không tồn tại. Lúc này, người tìm việc có thể xóa công việc này, không lưu nó trong mục công việc đã apply của mình nữa. 	10

Report 3# Final Report

12	REQ -3.5	Xóa account user	<ul style="list-style-type: none"> Hệ thống phải kiểm tra xem admin có đủ điều kiện để xóa account này hay không, nếu không thì sẽ thông báo lỗi, nếu đủ thì sẽ xóa account user này trong database và hiện thông báo xóa thành công hay không. 	11
13	REQ -1.5	Xóa tin	<ul style="list-style-type: none"> Khi nhấn vào nút Việc làm của tôi thì bắt buộc phải đăng nhập. Nếu đã có tài khoản thì đăng nhập vào hoặc không có thì có thể đăng ký một tài khoản mới. Có tồn tại tin trong mục “Việc làm của tôi” Để xóa tin thì trong mục “Việc làm của tôi” phải có tin đã lưu thì mới có thể xóa được 	12
14	REQ -2.5	Xem tin đã đăng	<ul style="list-style-type: none"> Có ít nhất một tin tuyển dụng đã được đăng trước đó mới có thể xem tin đã đăng. Các tin đã đăng còn tồn tại hoặc chưa bị xóa Có thể sửa thông tin đã đăng nếu nhà tuyển dụng có nhu cầu. Có thể xóa tin đã đăng nếu có yêu cầu gì phát sinh (Công việc không cần tuyển ứng viên nữa, công việc này đã bị hủy bỏ,...). Có thể xem thông tin của ứng viên apply vào tin này. 	13
15	REQ -2.2	Chỉnh sửa thông tin công ty	<ul style="list-style-type: none"> Phải có thông tin công ty trước đó mới có thể chỉnh sửa lại thông tin công ty cho phù hợp. Các thông tin của công ty không được để trống hoặc không hợp lệ. Nếu để trống hoặc không hợp lệ thì sẽ không lưu thành công thông tin đã chỉnh sửa. 	14

Report 3# Final Report

16	REQ -1.7	Xem thông tin nhà tuyển dụng	<ul style="list-style-type: none"> Tài khoản của nhà tuyển có tồn tại, chưa bị xóa. Nếu tài khoản này không còn tồn tại thì hệ thống sẽ hiện lên 1 thông báo. 	15
17	REQ -1.4	Đánh giá nhà tuyển dụng	<ul style="list-style-type: none"> Chỉ được đánh giá nhà tuyển dụng khi đã đăng nhập và một tài khoản chỉ được đánh giá mỗi nhà tuyển dụng một lần. Nếu người tìm việc đã đánh giá nhà tuyển dụng rồi, thì hệ thống sẽ cập nhật lần đánh giá gần nhất. 	16
18	REQ -3.3	Xem thống kê số lượng user	<ul style="list-style-type: none"> Hệ thống sẽ gửi 2 con số thống kê về để hiển thị số liệu thống kê lên cho admin xem (bao gồm số lượng người tìm việc và số lượng nhà tuyển dụng đang sử dụng app). Con số này sẽ được hệ thống cập nhật khi có account user mới được tạo. 	17
19	REQ -3.4	Xem hồ sơ user	<ul style="list-style-type: none"> Hệ thống sẽ gửi dữ liệu về để admin xem. 	18

V. Functional Requirements Specification

1. Stakeholders

- Các stakeholders: những nhà tuyển dụng, người tìm việc, admin, người bảo trì hệ thống, những người làm trong các bộ phận quản lý nhân sự của các công ty, lĩnh vực IT.

2. Actors and Goals

STT	Người/ thiết bị	Vai trò
1	Nhà tuyển dụng	<p>Đăng tin tuyển dụng giúp tạo cơ sở dữ liệu cho app.</p> <p>Duyệt hồ sơ tuyển dụng giúp thực hiện một trong những chức năng chính của app.</p> <p>Là 1 trong 2 đối tượng chính sử dụng app.</p>
2	Người tìm việc	Là 1 trong 2 đối tượng chính sử dụng app.

Report 3# Final Report

		Đối tượng này sử dụng chức năng quan trọng nhất của app là tìm việc và apply. Hồ sơ của đối tượng này là database quan trọng.
3	Admin	Quản lý các hồ sơ, tố cáo của user.
4	Smartphone chạy hệ điều hành Android, có kết nối Internet.	Là thiết bị để chạy app. Thiết bị phải kết nối Internet thì ứng dụng mới có thể chạy được, nếu không có Internet thì không thể tương tác với app.

3. Use case

a. Casual Description

- **Người sử dụng app**

STT	Tên Usecase	Tóm tắt	Dòng sự kiện chính	Dòng sự kiện phụ
1	Đăng ký tài khoản	Usecase này dùng để đăng ký tài khoản	Usecase bắt đầu khi người dùng muốn đăng ký một tài khoản để sử dụng các chức năng của app: + Khi đăng ký một tài khoản thì thông tin của người dùng sẽ được lưu lại trên cơ sở dữ liệu (Firebase)	Nếu tài khoản đăng ký đã tồn tại trên cơ sở dữ liệu thì người dùng phải nhập lại thông tin để đăng ký. Khi dùng đăng ký tài khoản thành công thì sẽ có thông báo là đăng ký tài khoản thành công để thông báo cho người dùng Admin không thể đăng ký một tài khoản mới

Report 3# Final Report

2	Đăng nhập	Usecase này dùng đăng nhập một tài khoản vào hệ thống để sử dụng các chức năng của app.	Usecase bắt đầu khi người dùng đăng nhập vào app Người dùng có thể đăng nhập bằng Gmail hoặc Facebook hoặc tài khoản đã đăng ký.	Người dùng chỉ đăng nhập thành công khi tài khoản đã tồn tại trên firebase Người dùng phải nhập đầy đủ thông tin username và password thì mới có thể đăng nhập thành công Khi người dùng là người tìm việc không cần đăng nhập vẫn có thể sử dụng một số chức năng của app. Đối với nhà tuyển dụng và admin thì bắt buộc phải đăng nhập trước khi sử dụng app
3	Đăng xuất	Usecase để đăng xuất tài khoản ra khỏi app	Khi người dùng muốn đăng xuất tài khoản ra khỏi app hoặc muốn thay đổi một tài khoản khác để tiếp tục sử dụng app	Người dùng chỉ có thể đăng xuất nếu như dùng đã đăng nhập vào app

- Người tìm việc

STT	Tên Usecase	Tóm tắt	Dòng sự kiện chính	Dòng sự kiện phụ
1	Tạo CV	Người tìm việc tạo CV để có thể	Người dùng có thể tạo CV mới	Nếu tạo CV mới mà người

Report 3# Final Report

		apply vào các công ty.	bằng cách điền đầy đủ thông tin cần thiết trên form có sẵn. Người dùng có thể đăng tải CV lên để apply nếu có sẵn CV thích hợp.	dùng không điền đủ thông tin thì phải điền lại mới có thể tạo CV thành công.
2	Edit CV	Người dùng có thể chỉnh sửa thông tin CV của mình để phù hợp với các vị trí ứng tuyển khác nhau.	Người dùng có thể thay đổi các thông tin trong CV bằng cách sửa, xóa hoặc thêm thông tin vào CV.	Không được phép để trống các thông tin cần phải có trong CV. Nếu xóa thông tin mà không thêm vào thì cập nhật CV không thành công
3	Tìm kiếm công việc	Người dùng có thể tìm kiếm công việc theo nhu cầu của bản thân trên thanh công cụ “Tìm Kiếm” trên app.	Người dùng tìm kiếm công việc bằng cách nhập tên công việc, chức danh hoặc từ khóa công ty và địa điểm làm việc để tìm kiếm công việc. Sau đó nhấn nút Tìm Việc	Người dùng bắt buộc phải nhập vào ô tìm kiếm. Nếu để trống mà vẫn nhấn nút “Tìm kiếm” thì hệ thống sẽ báo “Vui lòng nhập công việc cần tìm”
4	Đánh giá nhà tuyển dụng	Người dùng có thể đánh giá nhà tuyển dụng tùy theo mức độ hài lòng của mình về nhà tuyển dụng.	Người dùng đánh giá bằng cách xếp theo mức độ của sao. Cao nhất là 5 sao và thấp nhất là 1 sao đồng thời có thể đánh	Không có dòng sự kiện khác

Report 3# Final Report

			giá bằng comment trên trang của nhà tuyển dụng.	
5	Xóa tin	Người dùng có thể xóa tin mà mình đã lưu trong mục việc làm của tôi.	Người dùng chọn mục “Đã lưu” và nhấn vào tin mình muốn xóa để chọn “Xóa”	Không có dòng sự kiện khác
6	Apply	Người dùng có thể Apply một công việc mà mình thấy phù hợp với bản thân.	<p>Để apply được trước hết người dùng phải bắt buộc đã đăng nhập vào hệ thống và tài khoản đó phải đã tồn tại trên firebase.</p> <p>Khi đã có tài khoản để apply người dùng cần phải có CV. Nếu chưa có CV có thể tạo CV trực tiếp trên app hoặc tải CV có sẵn lên.</p> <p>Sau khi điền đầy đủ thông tin và nộp CV thì nhấn “Apply”. Khi apply thành công, thông tin công việc sẽ được lưu vào mục “Các công việc đã apply”</p>	<p>Nếu nhấn nút “Apply” mà chưa có tài khoản thì hệ thống sẽ tự động chuyển qua activity Đăng nhập, đăng ký.</p> <p>Nếu nhấn nút Apply mà chưa có CV thì hệ thống sẽ tự động chuyển sang activity Tạo CV.</p>
7	Xem thông tin	Người dùng có thể xem thông tin	Người dùng phải chọn xem	Không có dòng sự kiện khác.

Report 3# Final Report

	nhà tuyển dụng	công ty khi người dùng muốn tìm hiểu thêm thông tin về công ty	Tin tuyển dụng trước sau đó chọn “Xem chi tiết” mới có thể xem thông tin nhà tuyển dụng	
--	----------------	--	---	--

- Nhà tuyển dụng

STT	Tên Usecase	Tóm tắt	Dòng sự kiện chính	Dòng sự kiện phụ
1	Đăng ký thông tin	Usecase này dùng để đăng ký các thông tin công ty	Nhà tuyển dụng phải điền đầy đủ các thông tin mới có thể đăng ký thành công. Khi nhà tuyển dụng đăng ký thành công sẽ có thông báo và thông tin của nhà tuyển dụng sẽ được lưu trữ trên cơ sở dữ liệu (firebase)	Chỉ khi nhà tuyển dụng đăng ký một tài khoản hoặc đã đăng nhập mà chưa cung cấp thông tin công ty thì mới có thể đăng ký thông tin công tin. Mỗi nhà tuyển dụng chỉ có thể đăng ký thông tin một lần duy nhất. Nếu nhà tuyển dụng muốn chỉnh sửa thông tin công ty thì có thể vào phần hồ sơ nhà tuyển dụng và chỉnh sửa. Chỉ khi người dùng đã đăng ký thông tin công ty thì mới có thể thực

Report 3# Final Report

				hiện các chức năng như đăng tin, xem danh sách ứng viên,...
2	Chỉnh sửa thông tin công ty	Usecase dùng để chỉnh sửa thông tin công ty.	Khi người dùng chỉnh sửa thông tin công ty, thông tin công ty sẽ tự động cập nhập lại trên cơ sở dữ liệu	Khi người dùng chỉnh sửa thông tin công ty thành công thì sẽ có thông báo cho nhà tuyển dụng biết.
3	Đăng tin tuyển dụng	Usecase dùng để đăng tin tuyển dụng khi nhà tuyển dụng có nhu cầu.	Khi nhà tuyển dụng đăng tin thành công thì tin tuyển dụng sẽ được lưu trữ trên cơ sở dữ liệu (firebase). Các tin đã đăng sẽ được hiển thị trong phần các tin đã đăng của nhà tuyển dụng	Nếu nhà tuyển dụng điền thiếu thông tin thì việc đăng tin sẽ không thành công
4	Gỡ tin tuyển dụng	Usecase này dùng để gỡ tin tuyển dụng	Khi người tuyển dụng gỡ tin thì tin đó sẽ bị xóa dữ liệu trên cơ sở dữ liệu (firebase). Người tìm việc sẽ không thể tìm thấy tin đó trong danh sách các công việc	Không có dòng sự kiện khác
5	Xem tin đã đăng	Usecase này dùng để xem các tin tuyển dụng mà nhà tuyển dụng đã đăng	Nhà tuyển dụng có thể xem tất cả các tin đã đăng, ngoài ra cũng có thể xem danh	Nhà tuyển dụng có thể xóa tin đã đăng ở phần này

Report 3# Final Report

			sách các ứng viên đã apply vào công việc đó	
6	Xem danh sách ứng viên	Usecase này dùng xem danh sách các ứng viên đã apply vào công việc đã đăng	Nhà tuyển dụng có thể xem CV của ứng viên apply vào công việc. Nhà tuyển dụng có thể duyệt ứng viên hoặc không	Không có dòng sự kiện khác
7	Duyệt ứng viên	Usecase này dùng để duyệt các ứng viên đã apply vào công ty	Khi nhà tuyển dụng duyệt ứng viên thì thông tin ứng viên sẽ được chuyển qua mục chờ phỏng vấn. Khi nhà tuyển dụng duyệt ứng viên đó thì sẽ có thông báo gửi về cho người tìm việc là nhà tuyển dụng đã duyệt ứng viên đó. Thông tin ứng viên đó sẽ bị xóa trong tin tuyển dụng đó và được chuyển vào mục chờ phỏng vấn. Trong mục công việc của tôi của người tìm việc thì tin tuyển dụng đó sẽ được lưu vào mục chờ phỏng vấn Nếu người tuyển dụng	Nhà tuyển dụng có thể xem trực tiếp hồ sơ của ứng viên đã apply vào công việc đó để xem xét có duyệt hay không duyệt ứng viên đó.

Report 3# Final Report

			không duyệt hồ sơ của ứng viên, có thể xóa hồ sơ ứng viên khi đó sẽ có thông báo về cho người tìm việc đã apply vào công việc đó.	
--	--	--	---	--

- Admin

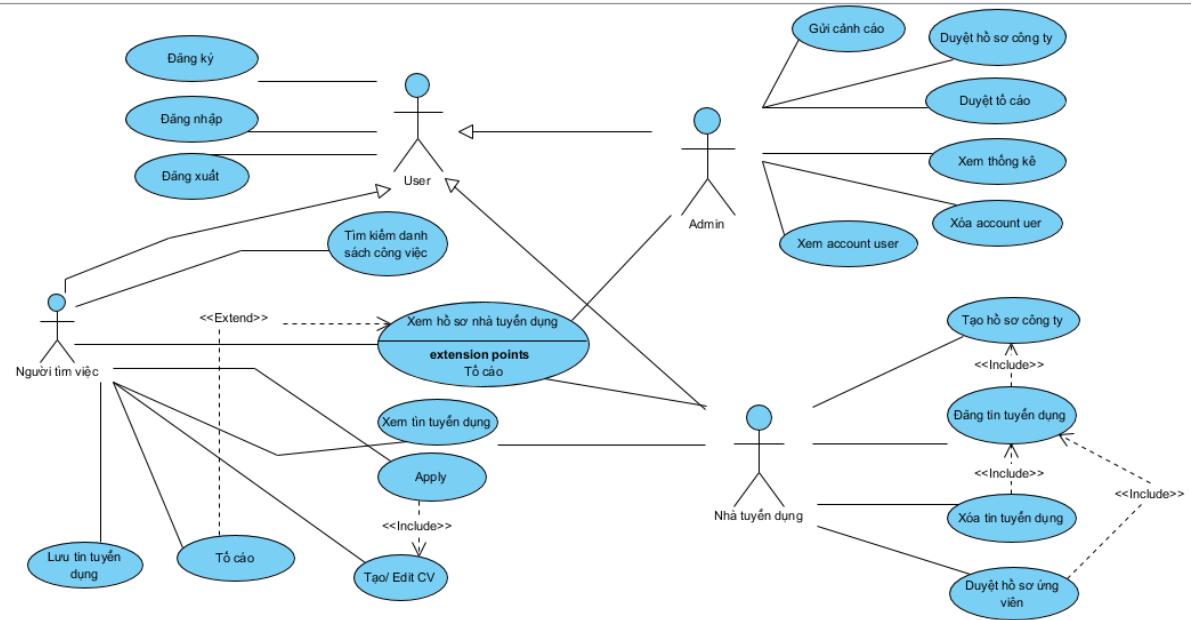
STT	Tên Usecase	Tóm tắt	Dòng sự kiện chính	Dòng sự kiện phụ
1	Cảnh cáo	Cảnh cáo những hồ sơ tố cáo	Admin nhận được thông báo có tố cáo mới. Admin có thể xem chi tiết tố cáo, gửi thông báo cảnh cáo đến người bị cảnh cáo hoặc có thể bỏ qua tố cáo này.	Nếu admin gửi cảnh cáo thì phải nhập lời cảnh cáo, nhắc nhở trước khi gửi. Nếu admin không nhập thì cảnh cáo sẽ không gửi được.
2	Duyệt hồ sơ nhà tuyển dụng	Admin duyệt hồ sơ của công ty tuyển dụng	Admin nhận thông báo có hồ sơ tuyển dụng mới. Admin có thể duyệt hoặc không duyệt hồ sơ tuyển dụng đó	Nếu admin không duyệt hồ sơ tuyển dụng thì nhà tuyển dụng không thể đăng tin tuyển dụng mới
3	Xem thống kê số lượng user	Admin có thể lấy số lượng user từ hệ thống cho việc thống kê	Admin xem số lượng user sử dụng app được phân loại theo loại người dùng: nhà tuyển dụng, người tìm việc	Không có dòng sự kiện khác

Report 3# Final Report

4	Xem hồ sơ user	Admin xem hồ sơ user	Admin bấm vào tên user (tên user này xuất hiện trong danh sách những user bị tố cáo, còn những user không có tên trong danh sách tố cáo thì admin không thể tìm và xem thông tin), admin chỉ có thể được xem tên, email và avatar của user, những thông tin khác thì không được phép xem.	Không có dòng sự kiện khác
5	Xóa account user	Admin xóa tài khoản của user	Admin bấm xóa tài khoản user (nếu thỏa điều kiện: user bị tố cáo từ 3 lần trở lên, tên của những user này chỉ xuất hiện trong danh sách bị tố cáo, admin không thể tự tìm user để xóa)	Không có dòng sự kiện khác

b. Use Case Diagram

Report 3# Final Report



Hình 1: Hình usecase

c. Traceability Matrix

- REQ-0: Người dùng

STT	System Requirements	Use case	Priority Weight
1	Đăng nhập	Đăng nhập	0
2	Đăng ký	Đăng ký tài khoản	1
3	Apply	Apply	2
4	Đăng tin tuyển dụng	Đăng tin tuyển dụng	3
5	Duyệt ứng viên	Duyệt ứng viên	4
6	Tạo CV	Tạo CV	5
7	Duyệt hồ sơ nhà tuyển dụng	Duyệt hồ sơ nhà tuyển dụng	6
8	Tìm kiếm công việc	Tìm kiếm công việc	7
9	Cảnh cáo	Cảnh cáo	8
10	Đăng ký thông tin công ty	Đăng ký thông tin công ty	9
11	Sửa CV	Edit Cv	10
12	Xem danh sách ứng viên	Xem danh sách ứng viên	11

Report 3# Final Report

13	Gõ tin tuyển dụng	Gõ tin tuyển dụng	12
14	Xóa account user	Xóa account user	13
15	Xóa tin	Xóa tin	14
16	Xem tin đã đăng	Xem tin đã đăng	15
17	Chỉnh sửa thông tin công ty	Chỉnh sửa thông tin công ty	16
18	Xem thông tin nhà tuyển dụng	Xem thông tin nhà tuyển dụng	17
19	Đánh giá nhà tuyển dụng	Đánh giá nhà tuyển dụng	18
20	Xem thống kê số lượng user	Xem thống kê số lượng user	19
21	Xem hồ sơ user	Xem hồ sơ user	20
22	Đăng xuất	Đăng xuất	21

d. Use case Specification

Đối tượng người dùng	STT	Tên Usecase	Tóm tắt	Dòng sự kiện chính	Dòng sự kiện phụ
Người sử dụng app	1	Đăng ký tài khoản	Usecase này dùng để đăng ký tài khoản	Usecase bắt đầu khi người dùng muốn đăng ký một tài khoản để sử dụng các chức năng của app: + Khi đăng ký một tài khoản thì thông tin của người dùng sẽ được lưu lại trên cơ sở dữ liệu (Firebase)	Nếu tài khoản đăng ký đã tồn tại trên cơ sở dữ liệu thì người dùng phải nhập lại thông tin để đăng ký. Khi dùng đăng ký tài khoản thành công thì sẽ có thông báo là đăng ký tài khoản thành công để thông báo cho người dùng

Report 3# Final Report

					Admin không thể đăng ký một tài khoản mới
	2	Đăng nhập	Usecase này dùng để đăng nhập một tài khoản vào hệ thống để sử dụng các chức năng của app.	Usecase bắt đầu khi người dùng đăng nhập vào app. Người dùng có thể đăng nhập bằng Gmail hoặc Facebook hoặc tài khoản đã đăng ký.	<p>Người dùng chỉ đăng nhập thành công khi tài khoản đã tồn tại trên firebase</p> <p>Người dùng phải nhập đầy đủ thông tin username và password thì mới có thể đăng nhập thành công</p> <p>Khi người dùng là người tìm việc không cần đăng nhập vẫn có thể sử dụng một số chức năng của app.</p> <p>Đối với nhà tuyển dụng và admin thì bắt buộc phải đăng nhập trước khi sử dụng app</p>
Người tìm việc	3	Tạo CV	<p>Người tìm việc tạo CV để có thể apply vào các công ty.</p> <p>Người dùng có thể tạo CV mới bằng cách điền đầy đủ thông tin cần thiết trên form có sẵn.</p> <p>Người dùng có thể đăng tải CV lên để apply nếu có sẵn CV thích hợp.</p>	<p>Người dùng có thể tạo CV mới bằng cách điền đầy đủ thông tin cần thiết trên form có sẵn.</p> <p>Người dùng có thể đăng tải CV lên để apply nếu có sẵn CV thích hợp.</p>	<p>Nếu tạo CV mới mà người dùng không điền đủ thông tin thì phải điền lại mới có thể tạo CV thành công.</p>

Report 3# Final Report

4	Apply	<p>Người dùng có thể Apply một công việc mà mình thấy phù hợp với bản thân.</p> <p>Để apply được trước hết người dùng phải bắt buộc đã đăng nhập vào hệ thống và tài khoản đó phải đã tồn tại trên firebase.</p> <p>Khi đã có tài khoản để apply người dùng cần phải có CV. Nếu chưa có CV có thể tạo CV trực tiếp trên app hoặc tải CV có sẵn lên.</p> <p>Sau khi điền đầy đủ thông tin và nộp CV thì nhấn “Apply”. Khi apply thành công, thông tin công việc sẽ được lưu vào mục “Các công việc đã apply”</p>	<p>Nếu nhấn nút “Apply” mà chưa có tài khoản thì hệ thống sẽ tự động chuyển qua activity Đăng nhập, đăng ký.</p> <p>Nếu nhấn nút Apply mà chưa có CV thì hệ thống sẽ tự động chuyển sang activity Tạo CV.</p>		
Nhà tuyển dụng	5	Đăng tin tuyển dụng	<p>Use case dùng để đăng tin tuyển dụng khi nhà tuyển dụng có nhu cầu.</p> <p>Khi nhà tuyển dụng đăng tin thành công thì tin tuyển dụng sẽ được lưu trữ trên cơ sở dữ liệu (firebase).</p> <p>Các tin đã đăng sẽ được hiển thị trong phần các tin đã đăng của nhà tuyển dụng</p>	<p>Nếu nhà tuyển dụng điền thiếu thông tin thì việc đăng tin sẽ không thành công</p>	
	6	Duyệt ứng viên	Use case này dùng để	Khi nhà tuyển dụng duyệt ứng	Nhà tuyển dụng có thể xem trực

Report 3# Final Report

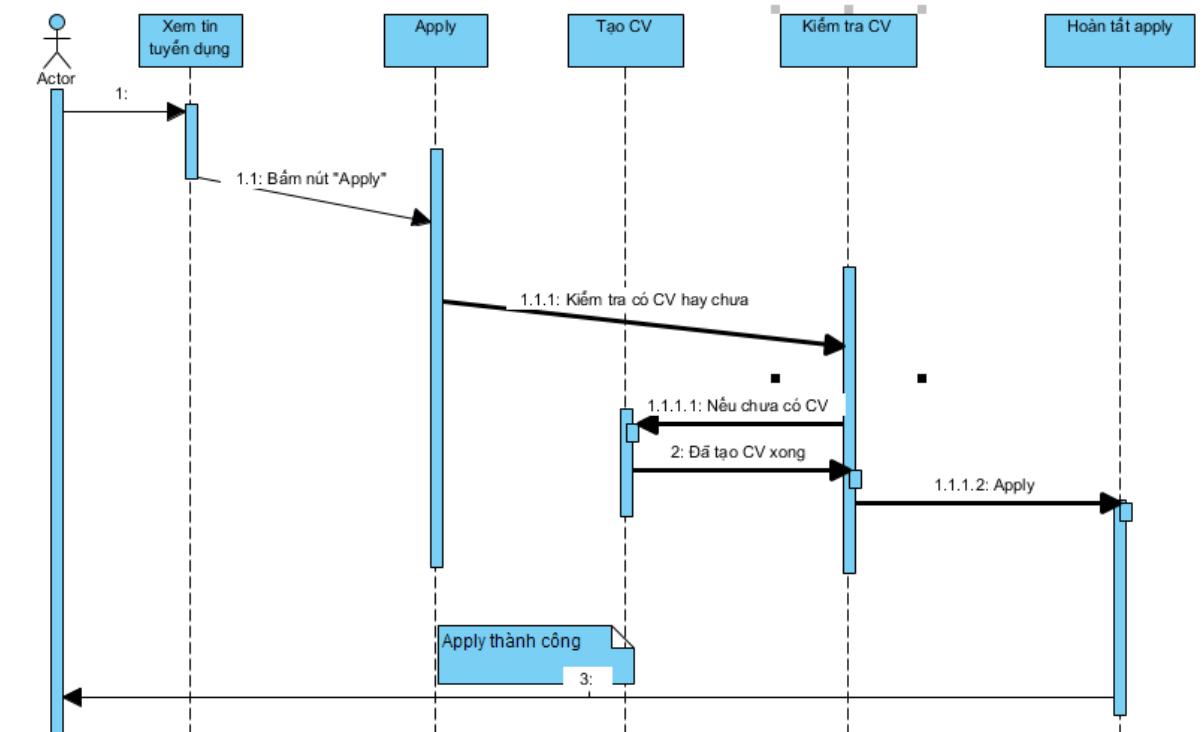
			đã duyệt các ứng viên đã apply vào công ty	<p>viên thì thông tin ứng viên sẽ được chuyển qua mục chờ phỏng vấn.</p> <p>Khi nhà tuyển dụng duyệt ứng viên đó thì sẽ có thông báo gửi về cho người tìm việc là nhà tuyển dụng đã duyệt ứng viên đó.</p> <p>Thông tin ứng viên đó sẽ bị xóa trong tin tuyển dụng đó và được chuyển vào mục chờ phỏng vấn.</p> <p>Trong mục công việc của tôi của người tìm việc thì tin tuyển dụng đó sẽ được lưu vào mục chờ phỏng vấn</p> <p>Nếu người tuyển dụng không duyệt hồ sơ của ứng viên, có thể xóa hồ sơ ứng viên khi đó sẽ có thông báo về cho người tìm việc đã apply vào công việc đó.</p>	tiếp hồ sơ của ứng viên đã apply vào công việc đó để xem xét có duyệt hay không duyệt ứng viên đó.
Admin	7	Cảnh cáo	Cảnh cáo những hồ sơ tố cáo	<p>Admin nhận được thông báo có tố cáo mới.</p> <p>Admin có thể xem chi tiết tố cáo, gửi thông báo cảnh cáo đến người bị cảnh cáo hoặc có</p>	<p>Nếu admin gửi cảnh cáo thì phải nhập lời cảnh cáo, nhắc nhở trước khi gửi. Nếu admin không nhập thì cảnh cáo sẽ</p>

Report 3# Final Report

				thẻ bỏ qua tố cáo này.	không gửi được.
8	Duyệt hồ sơ nhà tuyển dụng	Admin duyệt hồ sơ của công ty tuyển dụng	Admin nhận thông báo có hồ sơ tuyển dụng mới. Admin có thẻ duyệt hoặc không duyệt hồ sơ tuyển dụng đó	Admin nhận thông báo có hồ sơ tuyển dụng mới. Admin có thẻ duyệt hoặc không duyệt hồ sơ tuyển dụng đó	Nếu admin không duyệt hồ sơ tuyển dụng thì nhà tuyển dụng không thể đăng tin tuyển dụng mới

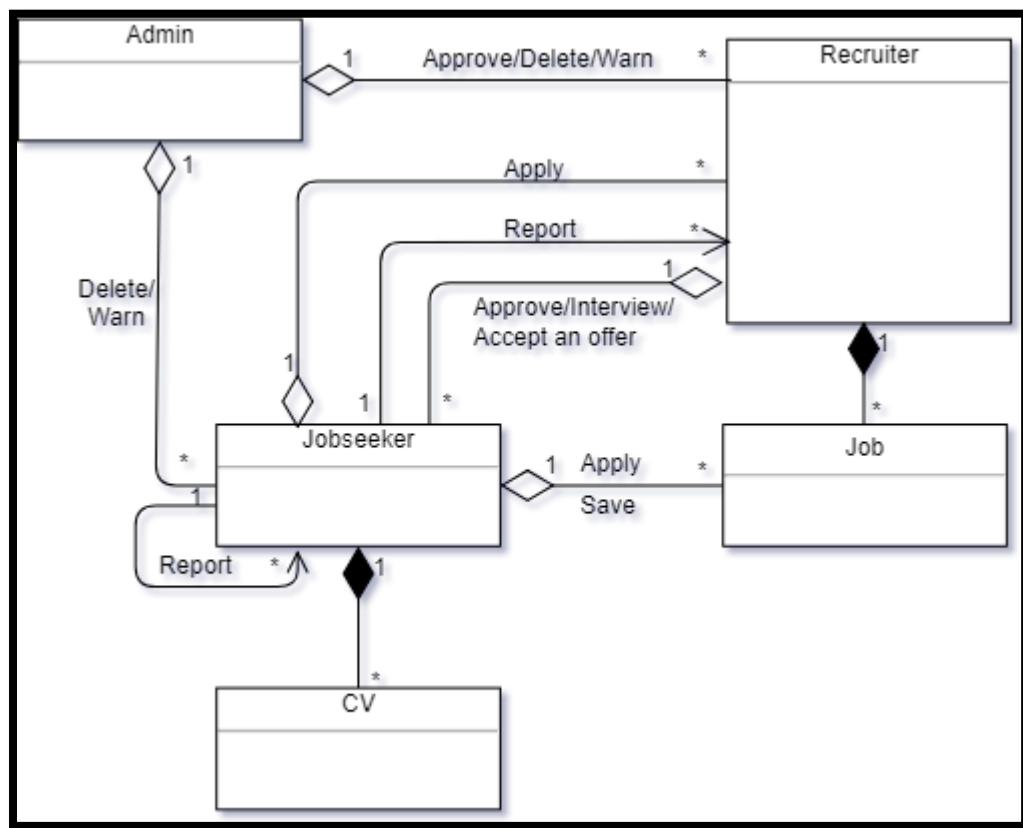
4. System Sequence Diagrams

Most important: Apply



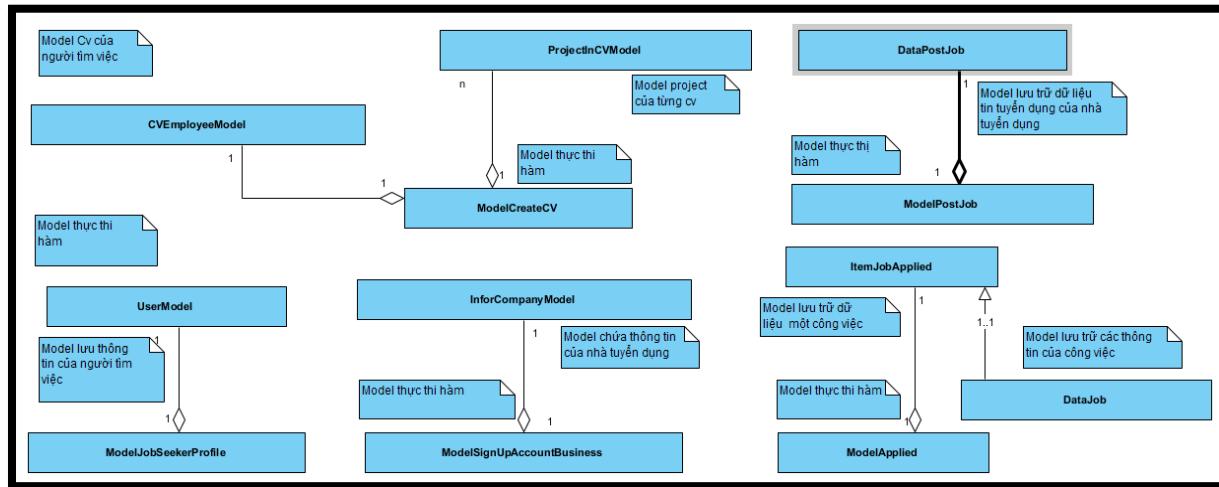
Hình 2: Sequence diagram cho usecase Apply

VI. Domain Analysis

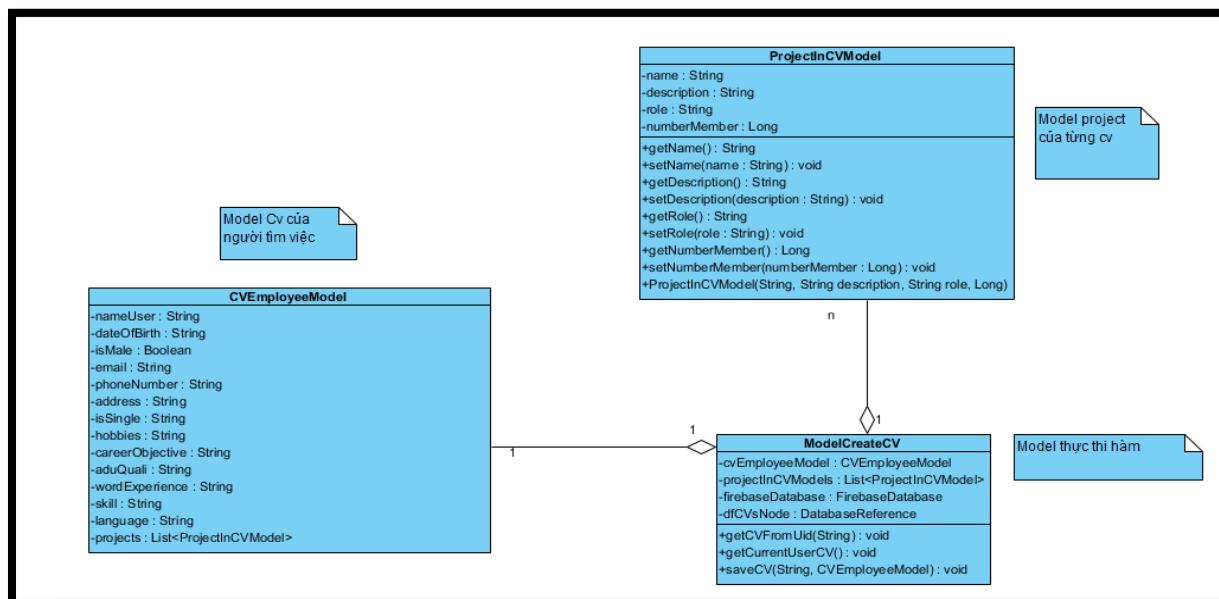


VII. Class Diagram and Interface Specification

1. Người Tìm Việc Và Nhà Tuyển Dụng

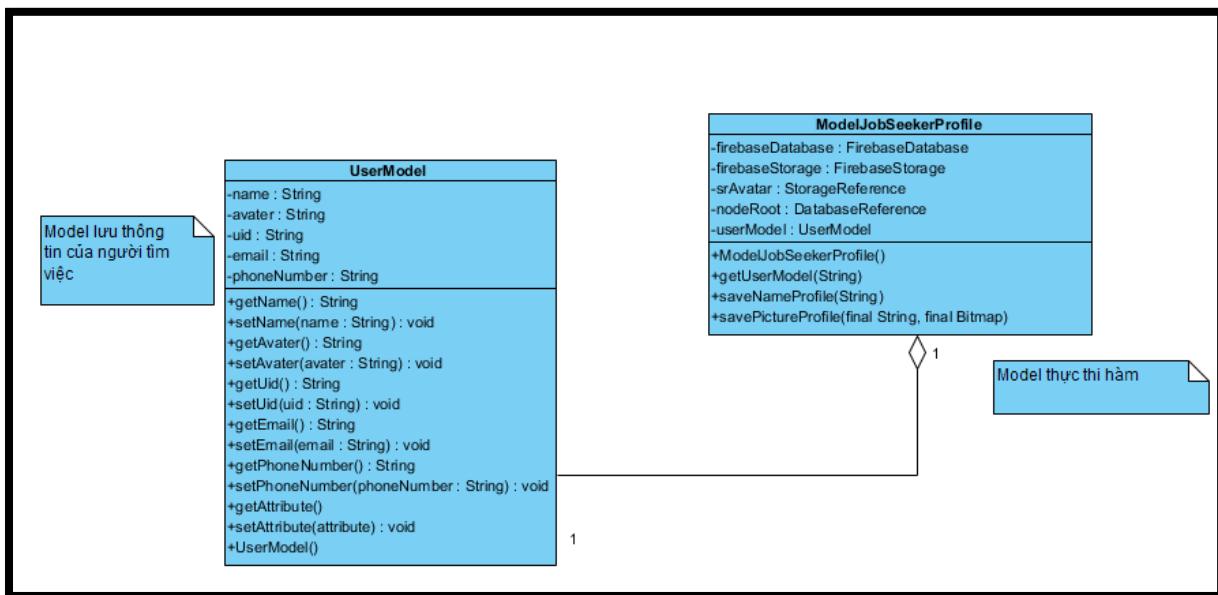


Hình 3. Tổng quan mối quan hệ giữa các class thao tác dữ liệu với Firebase

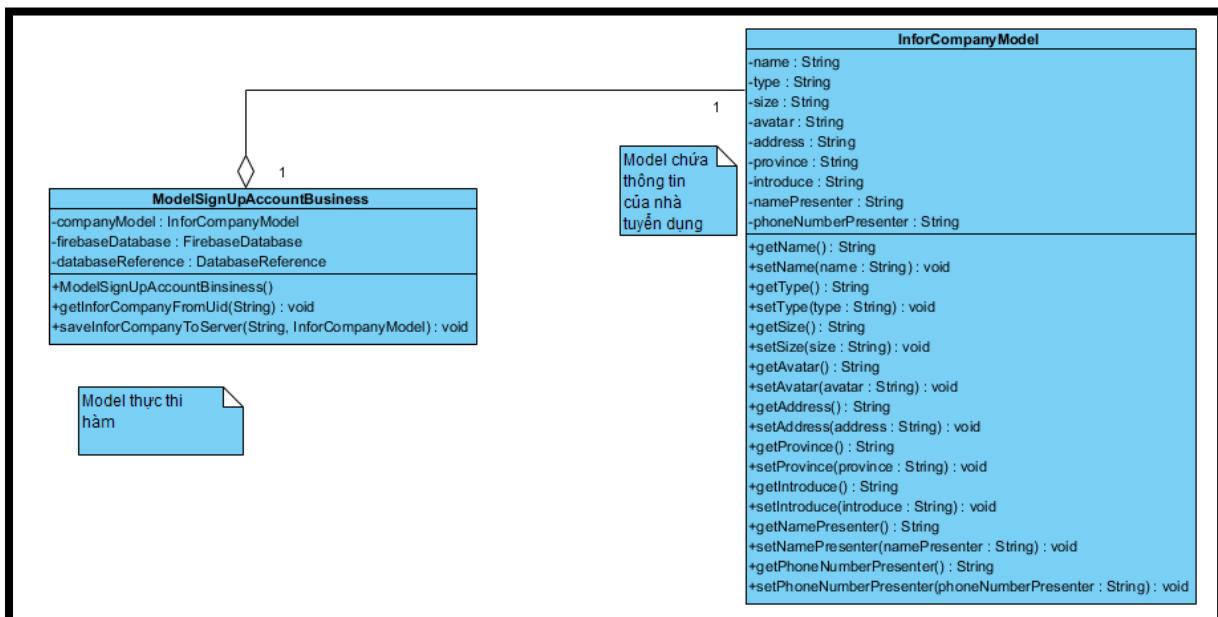


Hình 4. Mối quan hệ giữa class CVEmployeeModel, ProjectInCVMModel và ModelCreateCV

Report 3# Final Report

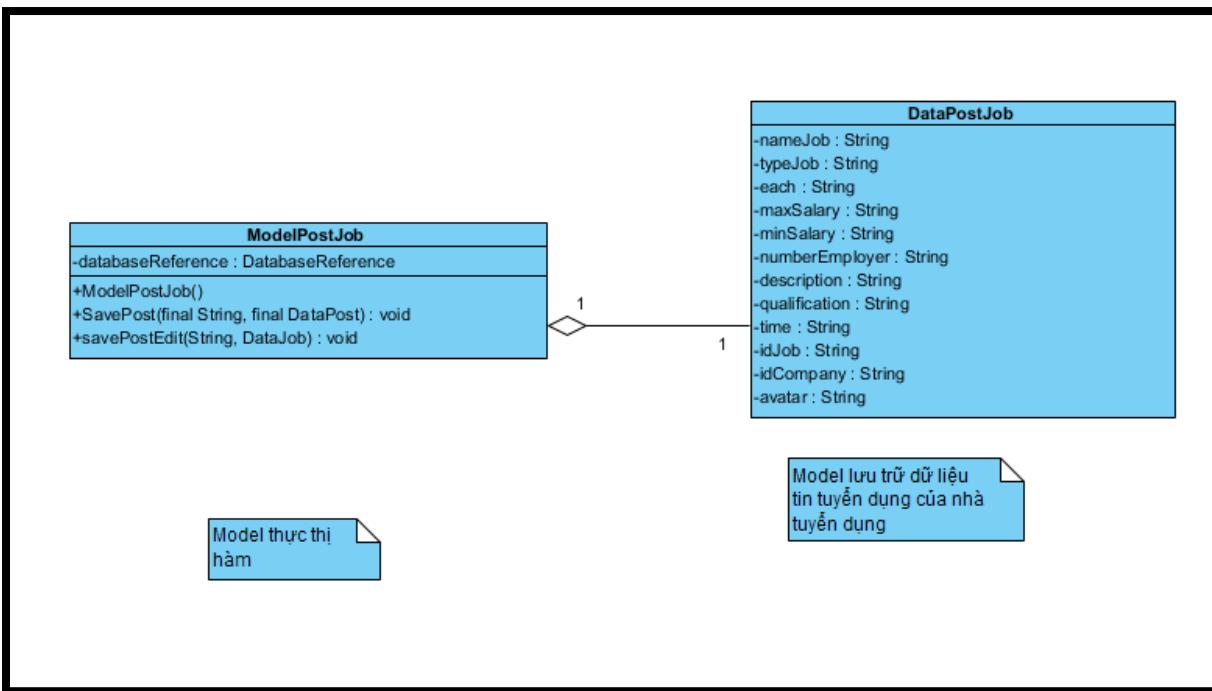


Hình 5. Mối quan hệ giữa class UserModel và ModelJobSeekerProfile

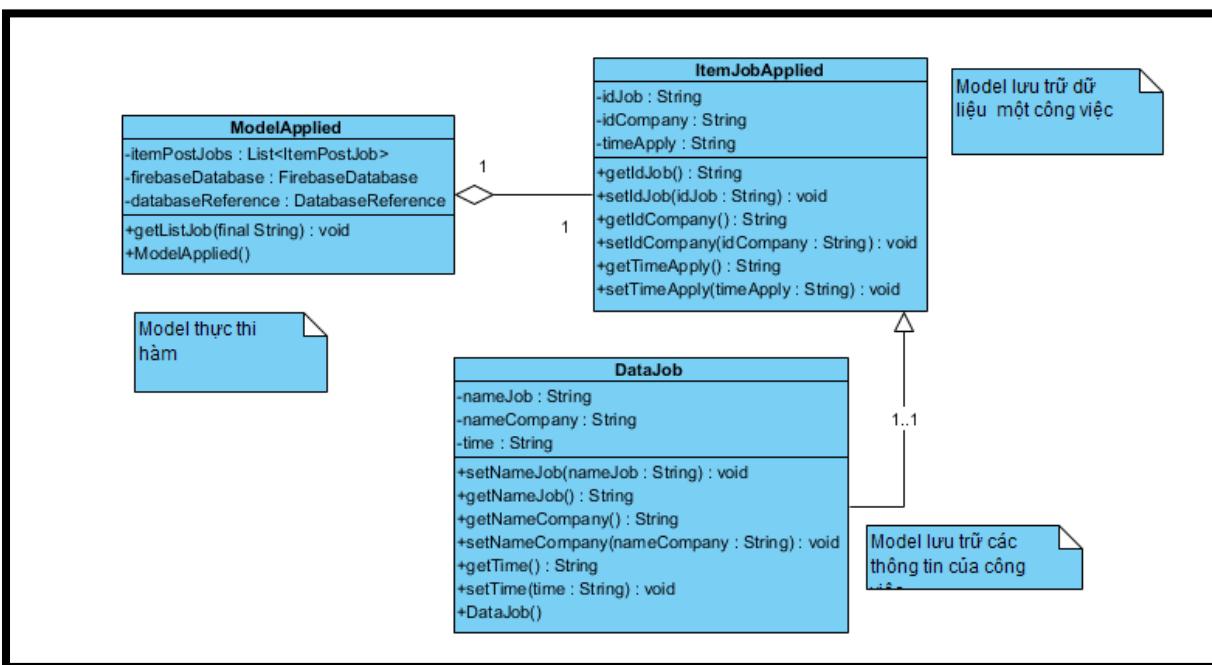


Hình 6. Mối quan hệ giữa ModelSignUpAccountBusiness và InforCompanyModel

Report 3# Final Report

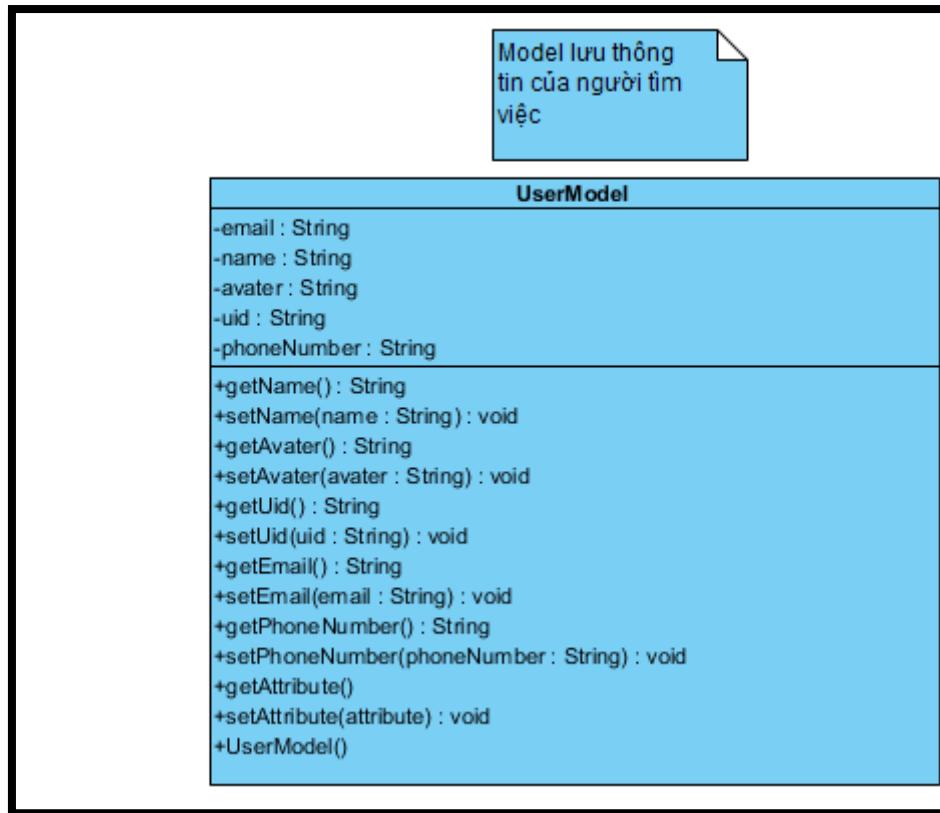


Hình 7: Mối quan hệ giữa class ModelPostJob, DataPostJob



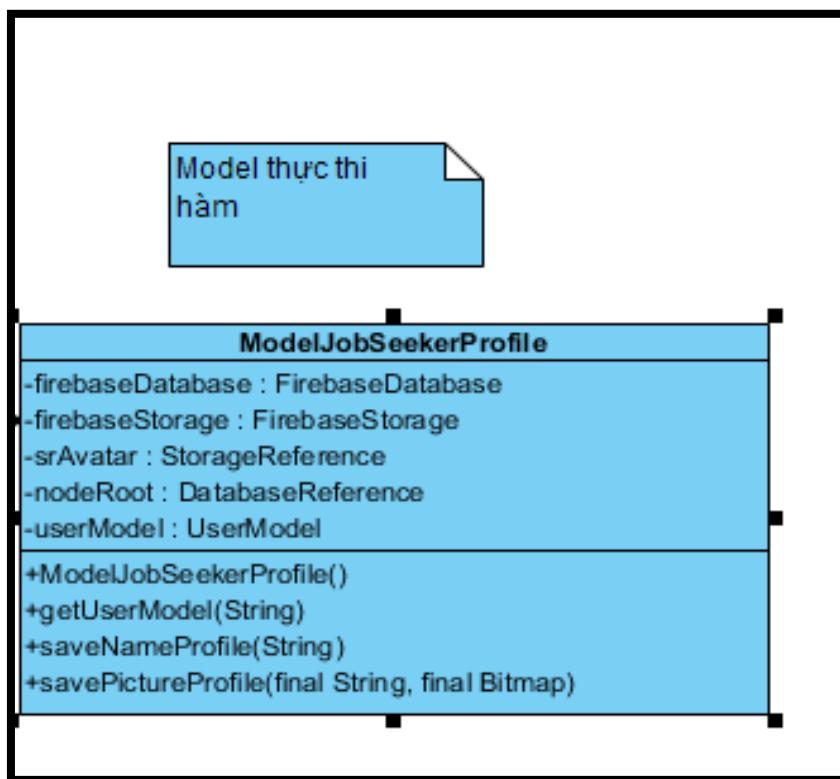
Hình 8. Mối quan hệ giữa class ModelApplied, DataJob và ItemJobApplied

Report 3# Final Report



Hình 9. UserModel

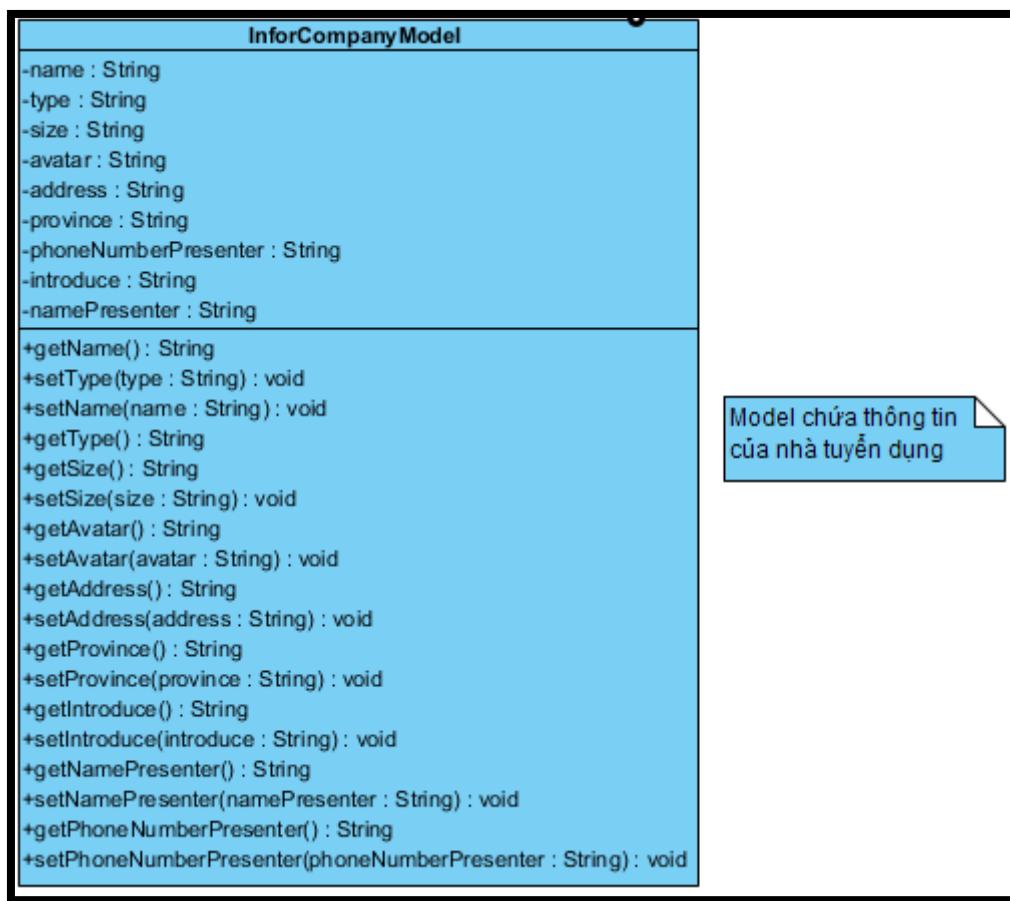
- Class UserModel dùng để lưu trữ tất cả các thông tin của người tìm việc



Hình 10. ModelJobSeekerProfile

Report 3# Final Report

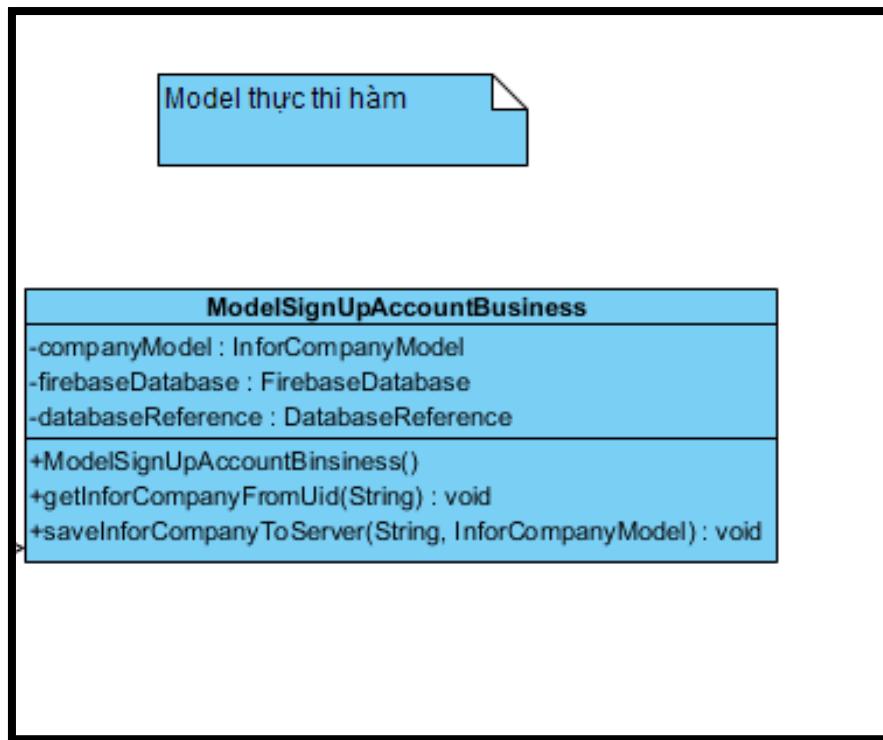
- Class ModelJobSeekerProfile dùng để thực thi các hàm, class này sẽ thực hiện việc lưu trữ thông tin người tìm việc lên Firebase.(hàm saveNameProfile và savePictureProfile)



Hình 11. InfoCompanyModel

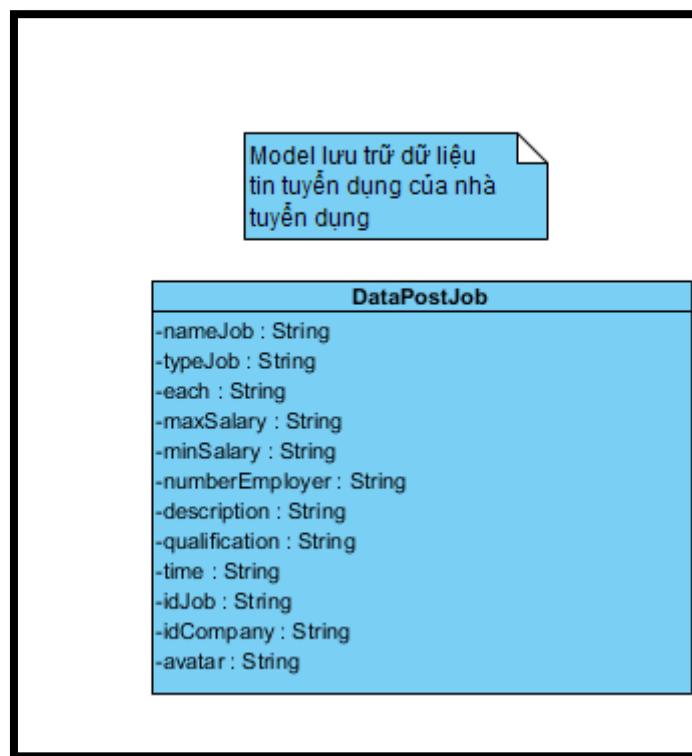
- Class Model lưu trữ các thông tin của nhà tuyển dụng, khi nhà tuyển dụng đăng ký một tài khoản thì sẽ cần cung cấp các thông tin trên.

Report 3# Final Report



Hình 12. ModelSignUpAccountBusiness

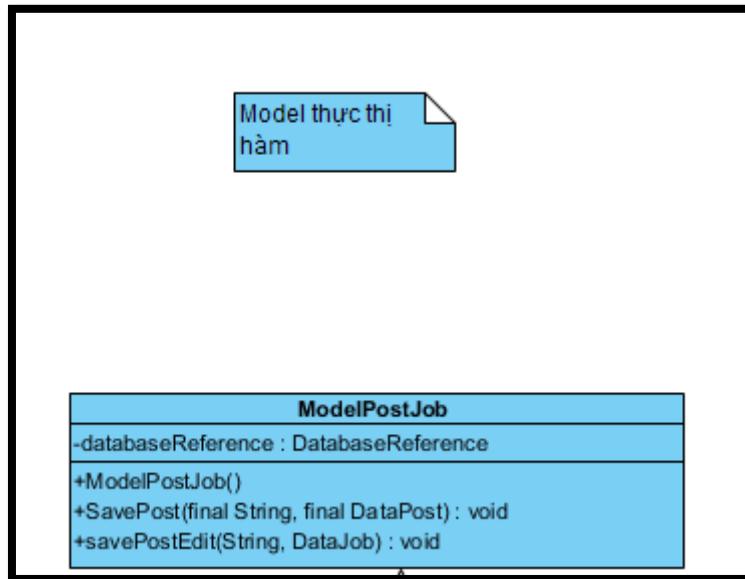
- Class Model để thực hiện việc lưu dữ liệu của công ty lên Firebase thông qua hàm saveInforCompanyToServer.



Hình 13. DataPostJob

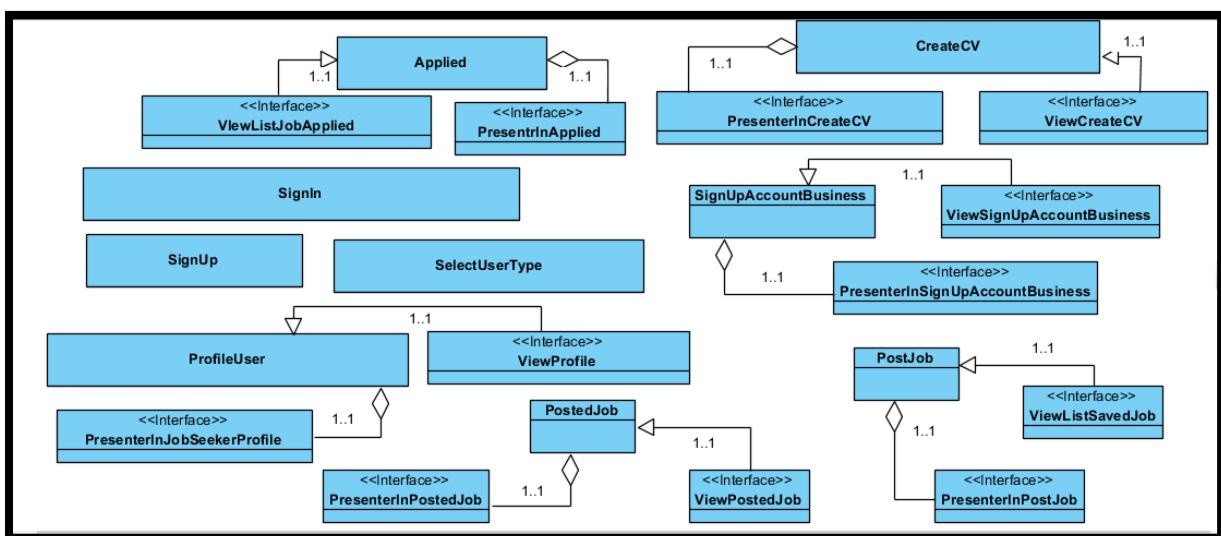
Report 3# Final Report

- Class model dùng để lưu trữ các thông tin về công việc mà công ty muốn đăng tuyển



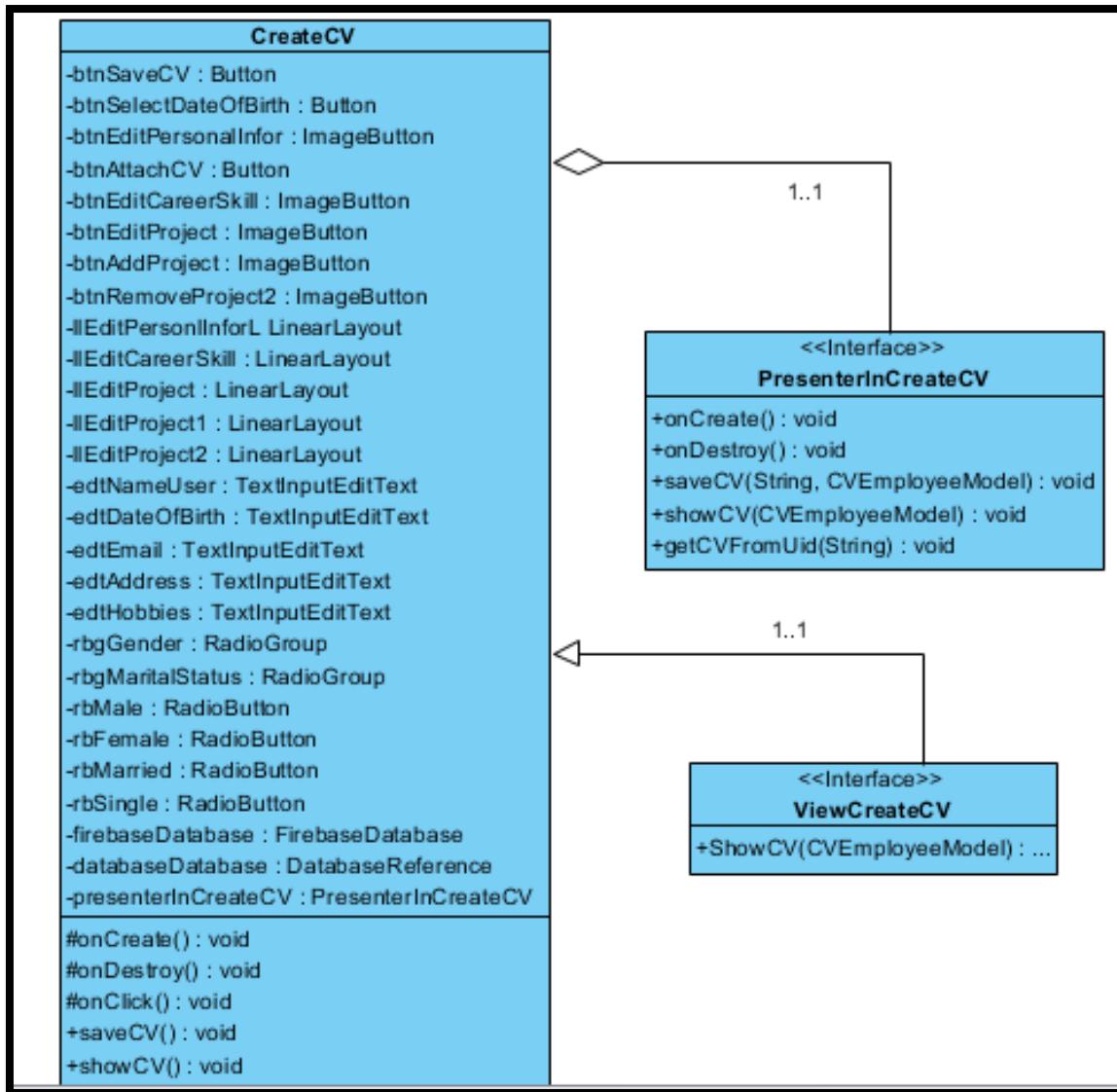
Hình 14. ModelPostJob

- Class ModelPostJob dùng để thực hiện lưu trữ thông tin của công việc lên Firebase thông qua hàm SavePost.



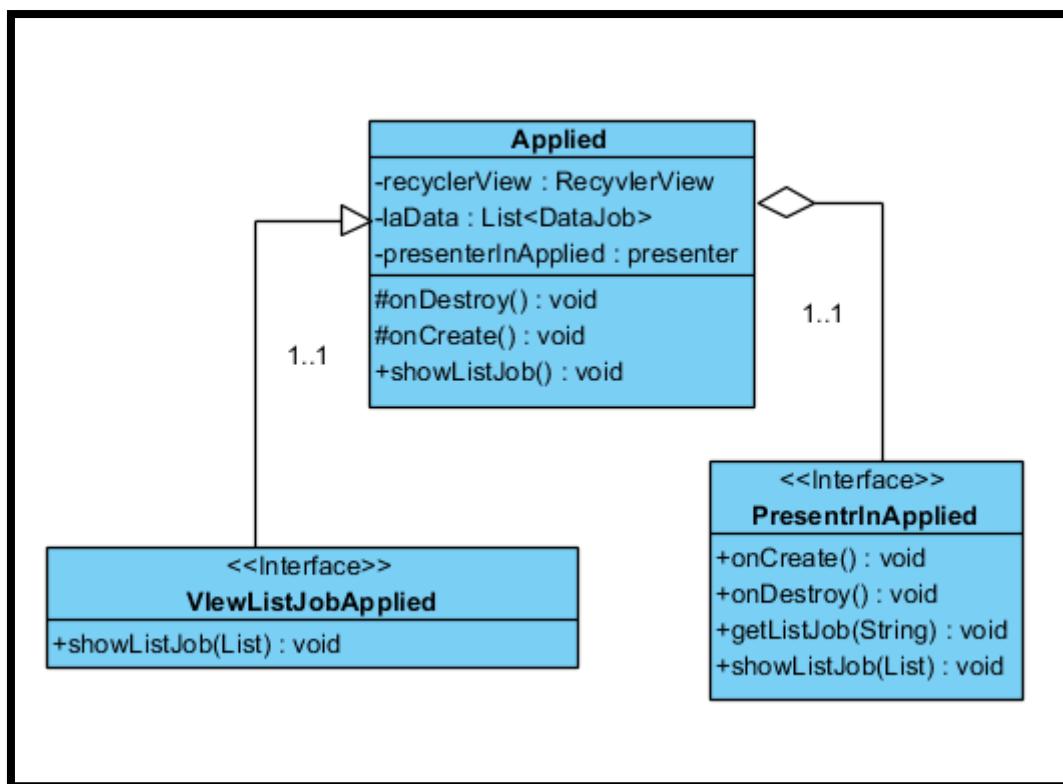
Hình 15. Tổng quan các class xử lý dữ liệu thực hiện với app

Report 3# Final Report



Hình 16. Class CreateCV

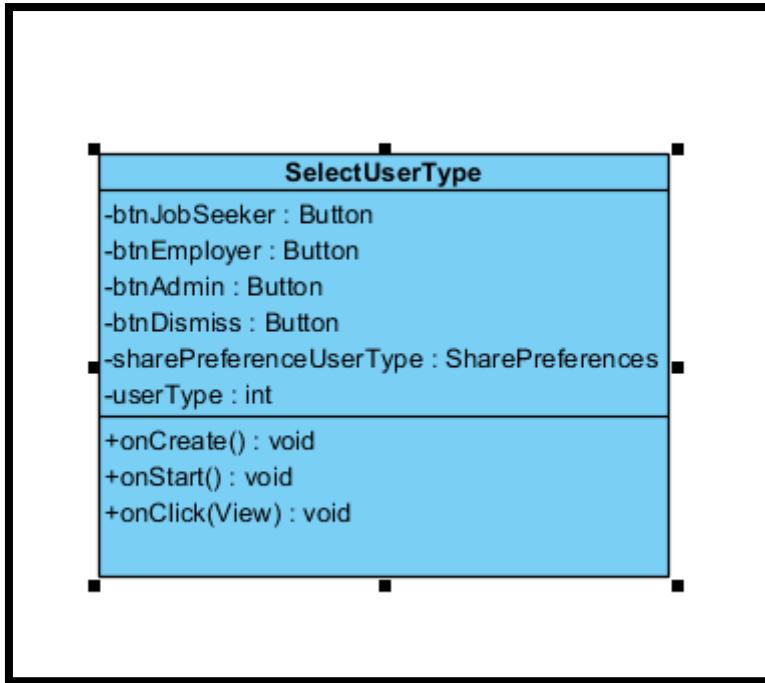
- Class tạo cv của người tìm việc, implement từ interface ViewCreateCV, ViewCreateCV có hàm showCV để lấy thông tin của người tìm việc và hiển thị ra thông qua hàm showCV trong class CreateCV.
- Hàm saveCV trong class CreateCV để lưu thông tin của người tìm việc lên Firebase. PresenterInCreateCV dùng làm trung gian để giao tiếp giữa Model và View thông qua hàm saveCV , showCV và getCVFromUid.GetCVFromUid dùng để thao tác với Firebase và lấy thông tin về.



Hình 17. Class Applied

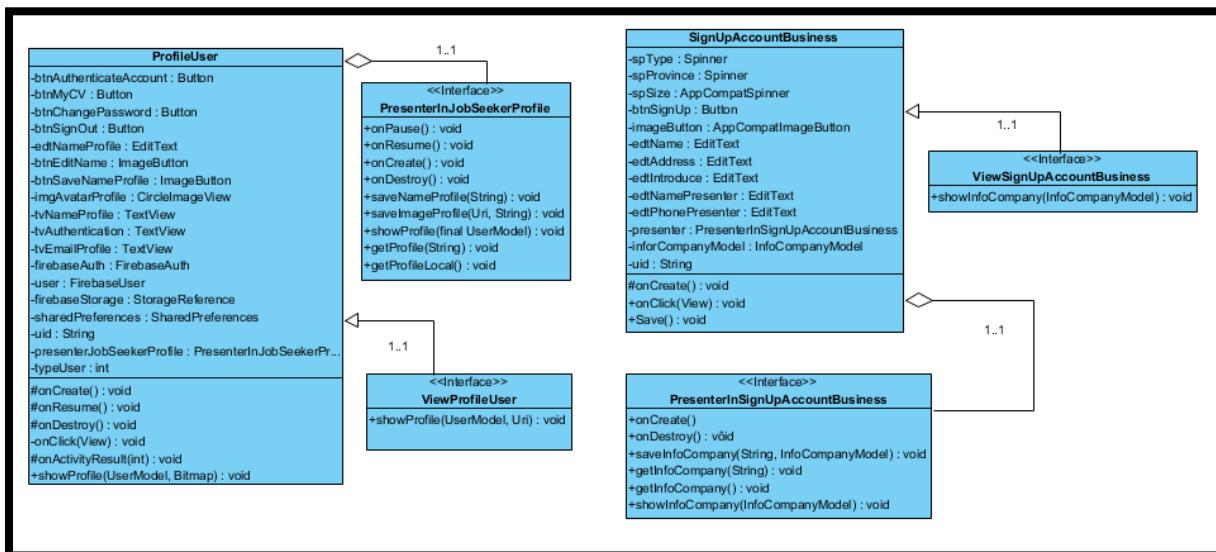
- Class **Applied** dùng giúp cho người tìm việc có thể apply vào công việc mình mong muốn, implement từ interface **VlewListJobApplied**, **VlewListJobApplied** có hàm `showListJob` để lấy danh sách các công việc mà người tìm việc đã apply và hiển thị ra thông qua hàm `showListJob` trong class **Applied**.
- **PresenterInApplied** dùng làm trung gian để giao tiếp giữa Model và View thông qua hàm `showListJob` và `getListJob`. `getListJob` dùng để thao tác với Firebase và lấy thông tin về. Khi người dùng nhập **Applied** thì công việc sẽ được lưu trên Firebase.

Report 3# Final Report



Hình 18. Class UserType

- Class SelectUserType dùng để phân quyền người dùng, là người tìm việc hay nhà tuyển dụng hay là Admin. Mỗi đối tượng có các đặc điểm riêng biệt. Khi một tài khoản chỉ được sử dụng cho một đối tượng.
- Đối với Admin thì chỉ được quyền đăng nhập, không được quyền đăng ký tài khoản

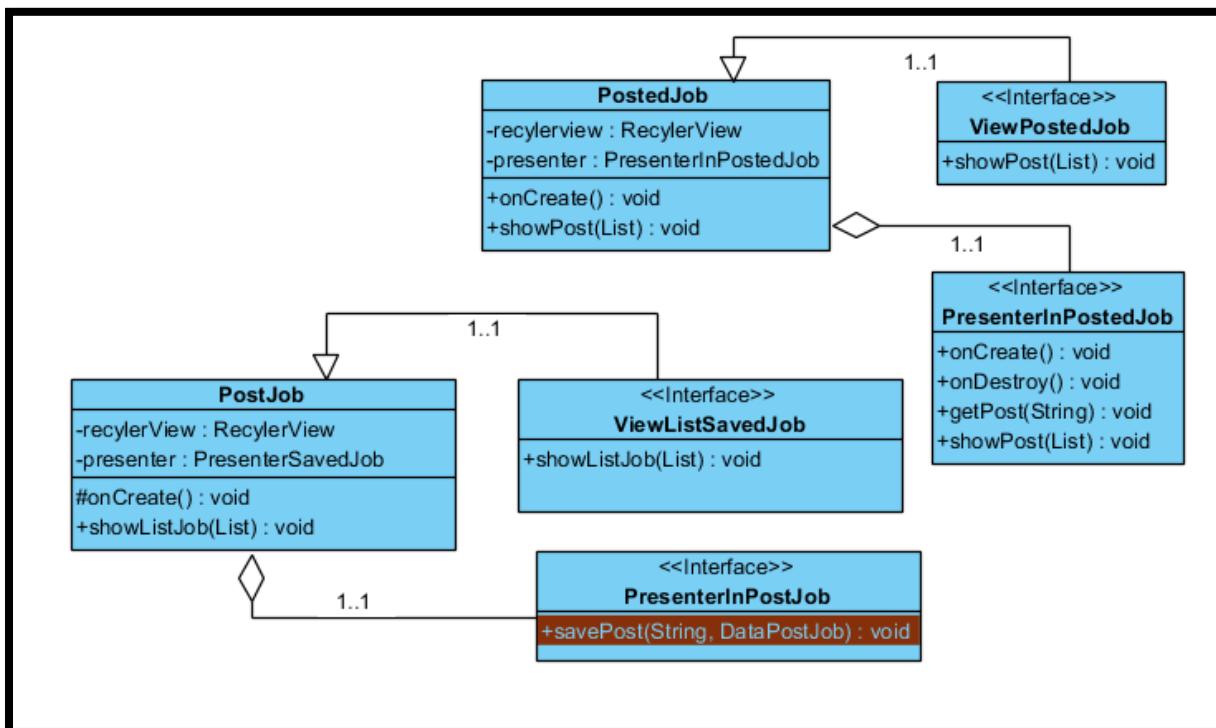


Hình 19. Class ProfileUser và SignUpAccountBusiness

- Đây là 2 class dùng để lưu thông tin người tìm việc và nhà tuyển dụng. Khi nhà tuyển dụng hoặc người tìm việc đăng nhập hoặc đăng ký ứng dụng sẽ yêu cầu người dùng nhập những thông tin cần thiết.

Report 3# Final Report

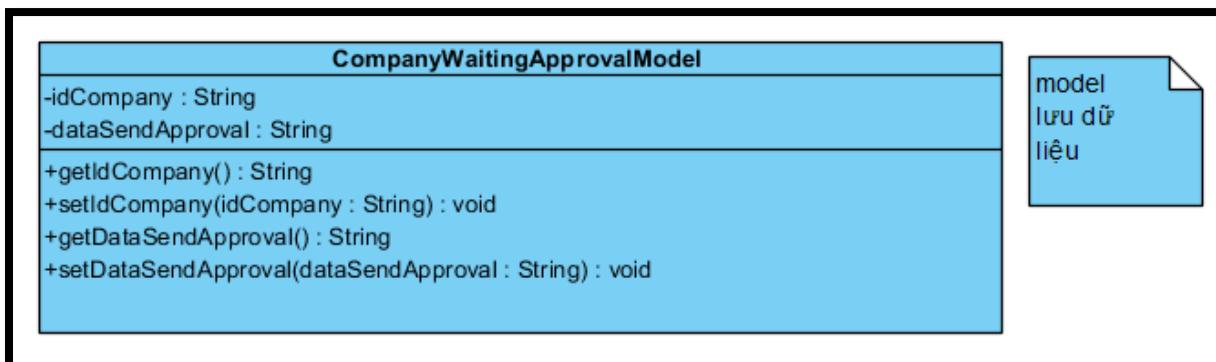
- Các thông tin này được đưa lên Firebase để lưu trữ và quản lý. View và Presenter có chức năng tương tự như những class trên



Hình 20. Class PostedJob và PostJob

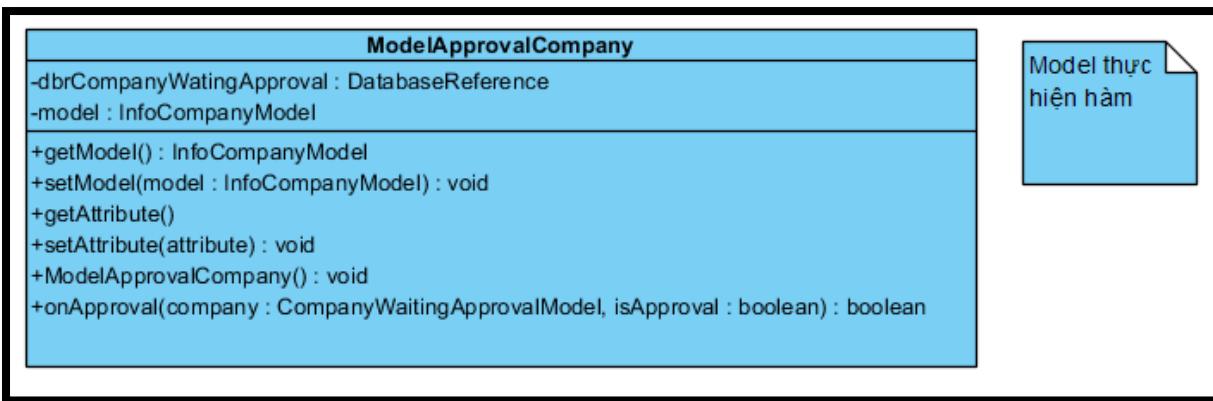
- Đây là 2 class PostedJob và PostJob. Các View và Presenter chức năng tương tự các lớp trên dùng để xử lý. 2 class này dùng để lưu trữ dữ liệu tin tuyển dụng của nhà tuyển dụng và lấy dữ liệu trên Firebase về hiện thị lên App.
- Các dữ liệu lấy được từ Firebase dùng để xử lý các chức năng khác như nhà tuyển dụng xem hồ sơ người ứng tuyển nộp vào từng tin đã đăng, sửa thông tin đã đăng...

2. Admin



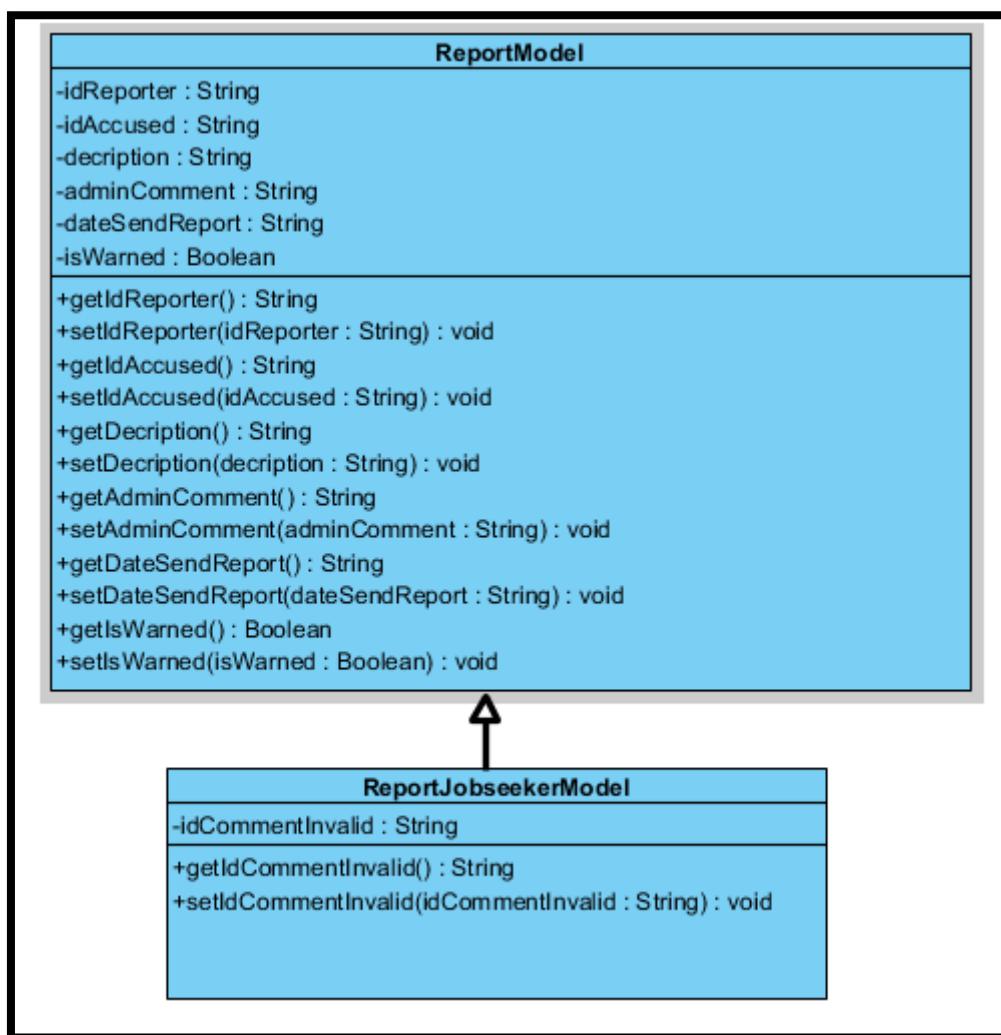
Hình 21. Class model công ty chờ admin duyệt hồ sơ.

Report 3# Final Report



Hình 22. Class ModelApprovalCompany

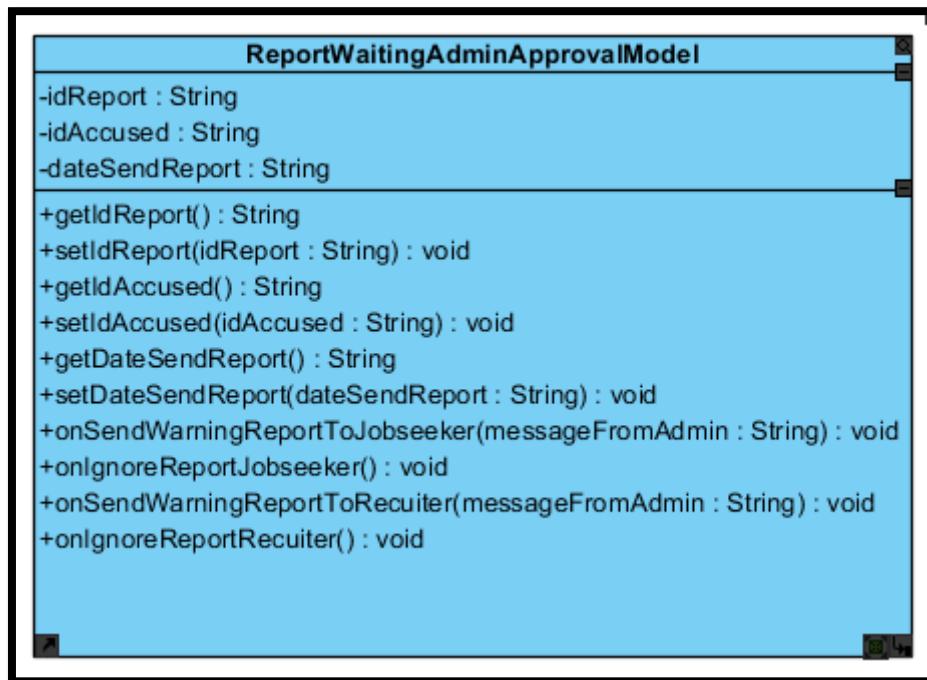
- Class model hồ sơ công ty cần duyệt, class này dùng để thực hiện hàm `onApproval` (xử lý phê duyệt công ty từ admin).



Report 3# Final Report

Hình 23. Class ReportModel và ReportJobSeekerModel

- Class model tố cáo (bên trên, ReportModel) và model tố cáo người tìm việc (bên dưới, kế thừa model tố cáo) . Hai class này là model lưu trữ dữ liệu.



Hình 24. Class ReportWaitingAdminApprovalModel

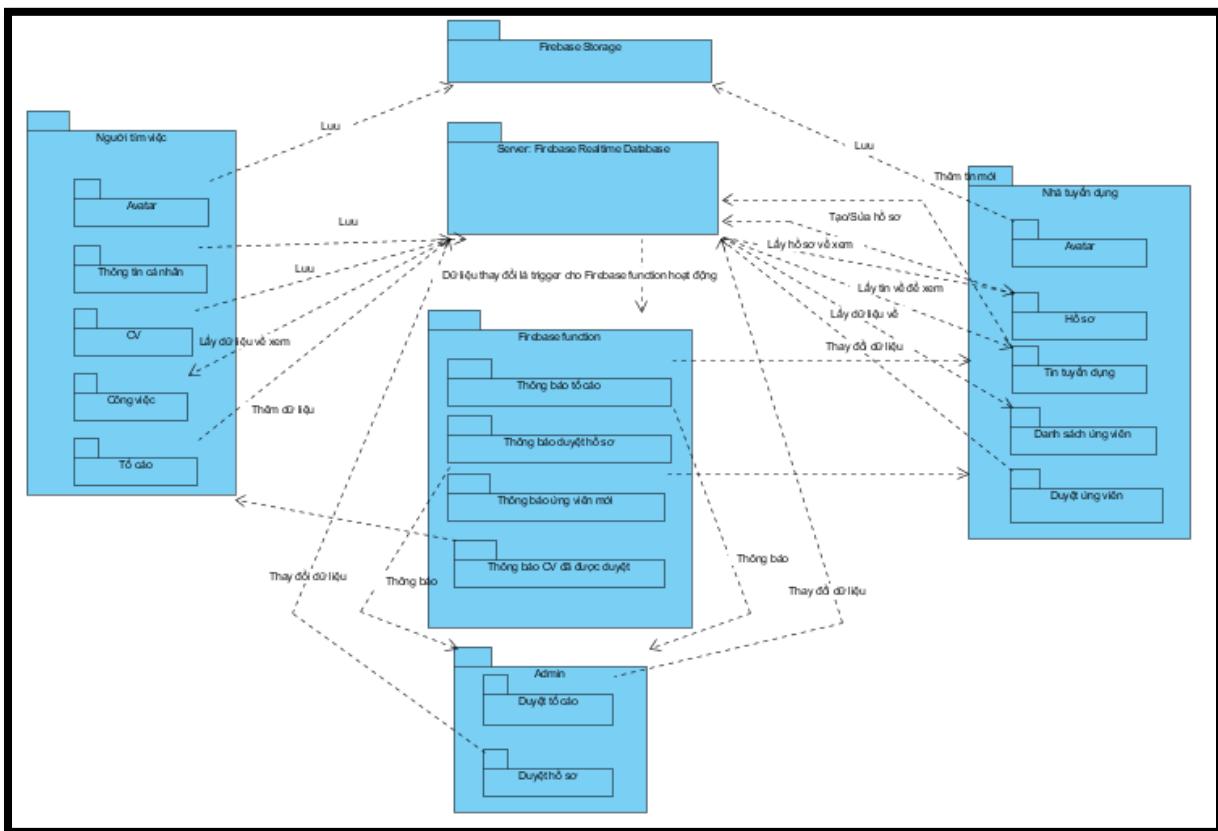
- Class model tố cáo đang chờ duyệt, dùng để lưu thông tin tố cáo đang chờ admin xử lý, class này có các hàm xử lý dữ liệu.

VIII. System Architecture and System Design

1. Architectural Styles

- Kiến trúc MVP (Model – View – Presenter):
 - + MVP là một mô hình kiến trúc hướng giao diện người dùng, được thiết kế để tạo thuận lợi cho việc kiểm thử đơn vị (unit testing) và tăng tính tách biệt giữa tầng dữ liệu và tầng hiển thị dữ liệu trong mô hình MVC.
 - + Mô hình MVP cho phép tách tầng trình diễn (Presenter) ra khỏi tầng dữ liệu (Model), vì vậy tương tác với giao diện được tách biệt với cách chúng ta biểu diễn nó trên màn hình (View), hay nói cách khác, tất cả logic khi người dùng tương tác được tách ra và đưa vào tầng trình diễn. Thiết kế lý tưởng nhất là với cùng một logic được áp dụng cho nhiều View khác nhau và hoán đổi được cho nhau.
- ⇒ Trong đó:
 - + Model là một interface xác định cách mà dữ liệu được hiển thị trong giao diện người dùng.
 - + View là một giao diện người dùng thụ động hiển thị dữ liệu (Model) và tiếp nhận tương tác người dùng và truyền tới để Presenter xử lý tương tác.
 - + Presenter được ví như middle-man. Khi người dùng tương tác với View, Presenter tiếp nhận tương tác người dùng và update Model. Khi Model được update hay có thay đổi, Presenter lấy dữ liệu từ Model, định dạng và đưa tới View để hiển thị.
- Kiến trúc Client-Server
 - + Là mô hình mà trong đó các máy chủ (server) cung cấp và quản lý các nguồn lực và dịch vụ cho máy khách (client)
 - + Kiểu kiến trúc này có một hoặc nhiều máy khách kết nối với máy chủ trung tâm thông qua mạng hoặc Internet.
 - + Kiến trúc Client-Server cũng có thể được gọi là mạng mô hình tính toán vì tất cả các yêu cầu và dịch vụ là được phân phối qua mạng.

2. Identifying Subsystems



Hình 25: Sơ đồ subsystem (gồm server- Firebase Realtime Database, Firebase function, Firebase Storage và Admin)

3. Mapping Subsystems to Hardware

- Không áp dụng

4. Persistent Data Storage

- Không áp dụng

5. Network Protocol

- Không áp dụng

6. Global Control Flow

- Quá trình lấy dữ liệu và hiển thị lên view (Activity, Fragment) (Mô hình MVP):
 - + Trong Model chạy 1 tiến trình (gọi là tiến trình A) chạy song song với tiến trình chính, đặt 1 callback trong tiến trình A. Khi dữ liệu được lấy hoàn tất sẽ kích hoạt callback gửi dữ liệu cho Activity hiển thị. Khi đó tiến trình A sẽ kết thúc.

- Quá trình hiển thị và lưu hình ảnh vào bộ nhớ:

Report 3# Final Report

- + Trong hàm onBindViewHolder chạy 1 tiến trình(gọi là tiến trình B) để download ảnh trên server về khi quá trình lấy dữ liệu hoàn tất ta sẽ hiện thị lên view. Sau đó tạo một tiến trình để lưu hình ảnh vào bộ nhớ (gọi là tiến trình C) (ngay khi tạo xong tiến trình C, tiến trình B kết thúc) và khi lưu ảnh xong tiến trình C sẽ kết thúc.
- Quá trình lưu dữ liệu trên server và vào bộ nhớ:
 - + Khi người dùng tiến hành lưu dữ liệu (nhấn button) thì sẽ tạo 2 tiến trình để lưu dữ liệu lên server và vào bộ nhớ. Khi lưu xong tiến trình lưu dữ liệu vào bộ nhớ sẽ tự kết thúc, tiến trình lưu dữ liệu trên server sẽ báo thành công hoặc thất bại về và ta sẽ hiển thị lên view cho người dùng biết và sau đó kết thúc.

7. Hardware Requirements

- Phiên bản android >= 5.0 (Lollipop (API 21)).
- Màn hình >= 5 inch.
- Cần bộ nhớ để chạy và lưu trữ dữ liệu. Sau 1 tuần dùng app bộ nhớ mà app sử dụng trong máy chỉ 8MB rất nhỏ so với bộ nhớ của điện thoại hiện nay. Vì vậy app chỉ cần bộ nhớ máy trống ~50MB là app hoạt động ổn định(Trong điều kiện chỉ có 1 app JobIT hoạt động).
- Cần cấp quyền truy cập bộ nhớ cho app để truy cập, lưu trữ dữ liệu trong bộ nhớ trong.
- Cần có mạng >= 128 Kbps.

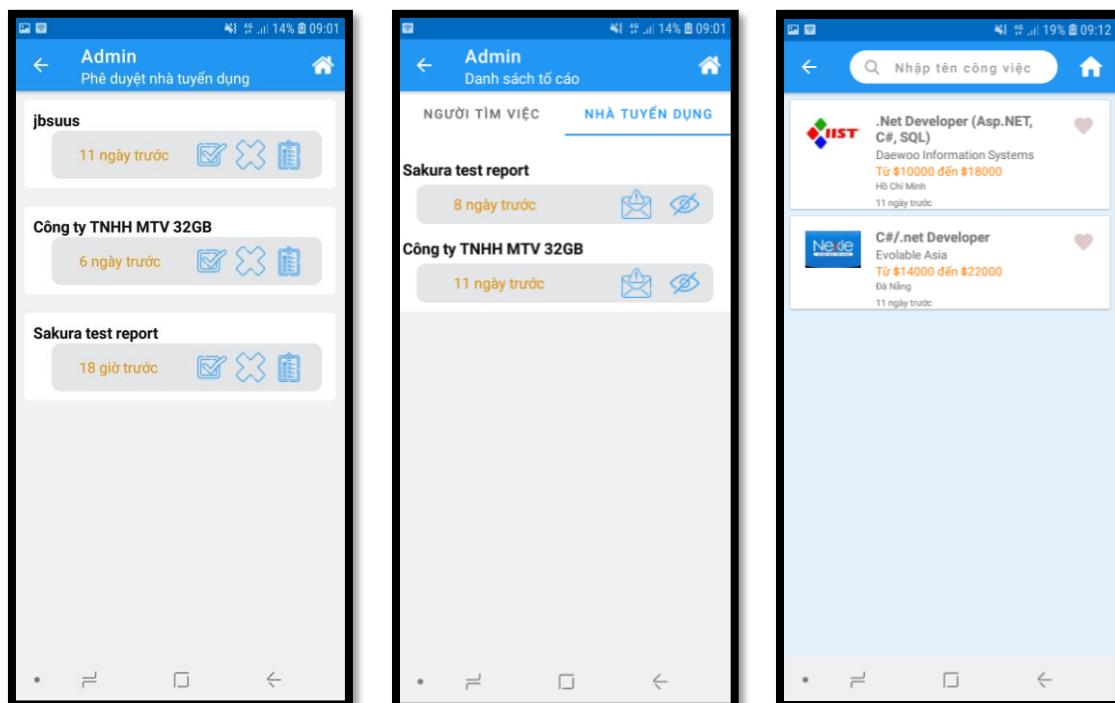
IX. Algorithms and Data Structures(if applicable)

1. Algorithms

- Không sử dụng

2. Data Structures

- Đồ án sử dụng cấu trúc dữ liệu danh sách (lists) và cây (trees).
- Cấu trúc dữ liệu lists dùng để lưu list các đối tượng (Ví dụ: list các công việc, list ứng viên, list hồ sơ công ty cần duyệt...). Vì số lượng phần tử trong mỗi danh sách không biết trước và thay đổi liên tục vì thế cấu trúc dữ liệu dạng list là hợp lý và linh hoạt. Cấu trúc này cho phép ta tạo ra một mảng các phần tử với số lượng phần tử có thể thay đổi được, việc thêm/ xóa/ cập nhật/ truy cập từng phần tử trong danh sách cũng được thực hiện dễ dàng.

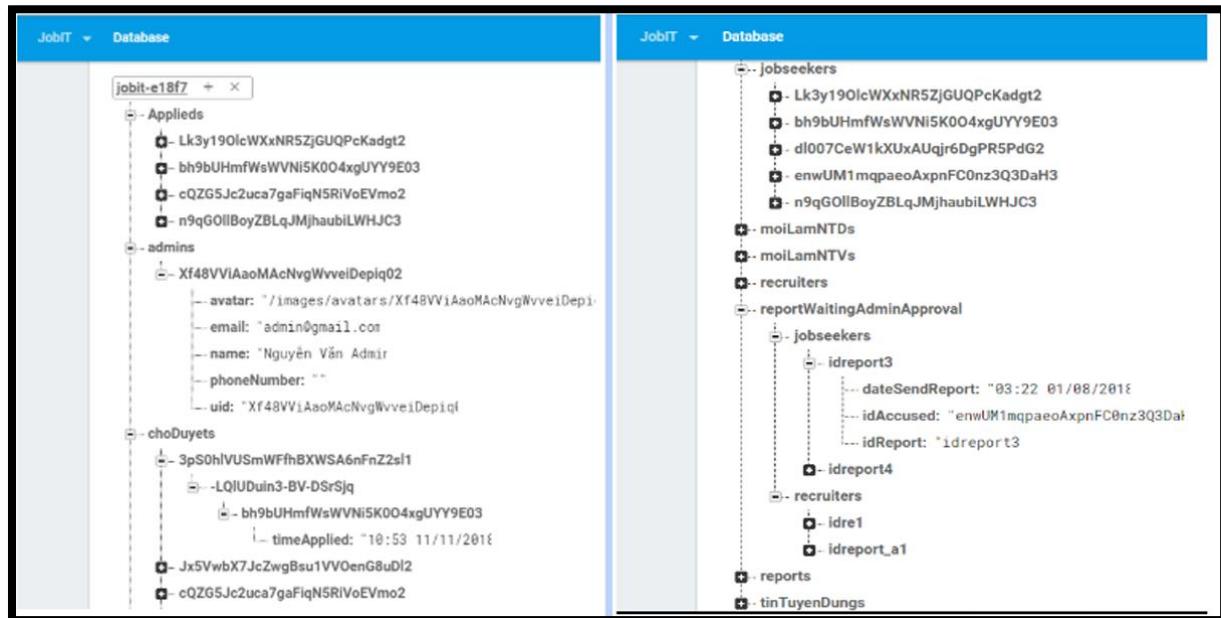


Hình 26. Hình ảnh minh họa dữ liệu dạng lists

- Cấu trúc dữ liệu dạng cây dùng để lưu data của toàn hệ thống. Hệ thống sử dụng Firebase Realtime Database để lưu database. Trên Firebase Realtime Database, dữ liệu được tổ chức theo dạng tree. Cấu trúc dữ liệu tree phù hợp với app do database của app JobIT được tổ chức phân tầng,

Report 3# Final Report

độ sâu không quá 5, có thể truy vấn dữ liệu một cách logic và phù hợp với môi trường Firebase Realtime Database.

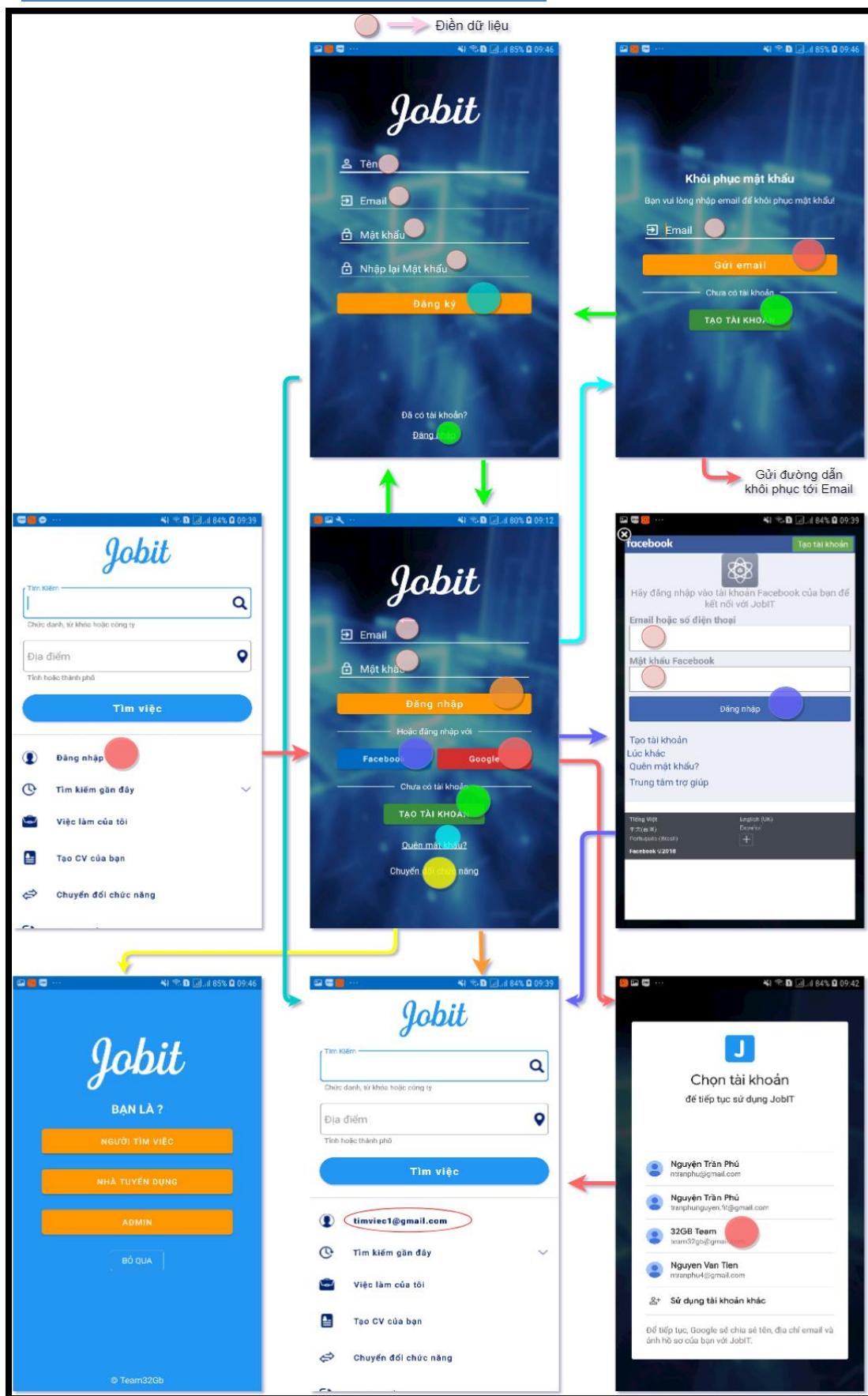


Hình 27. Hình ảnh minh họa cấu trúc dữ liệu dạng trees

X. User Interface Design and Implementation

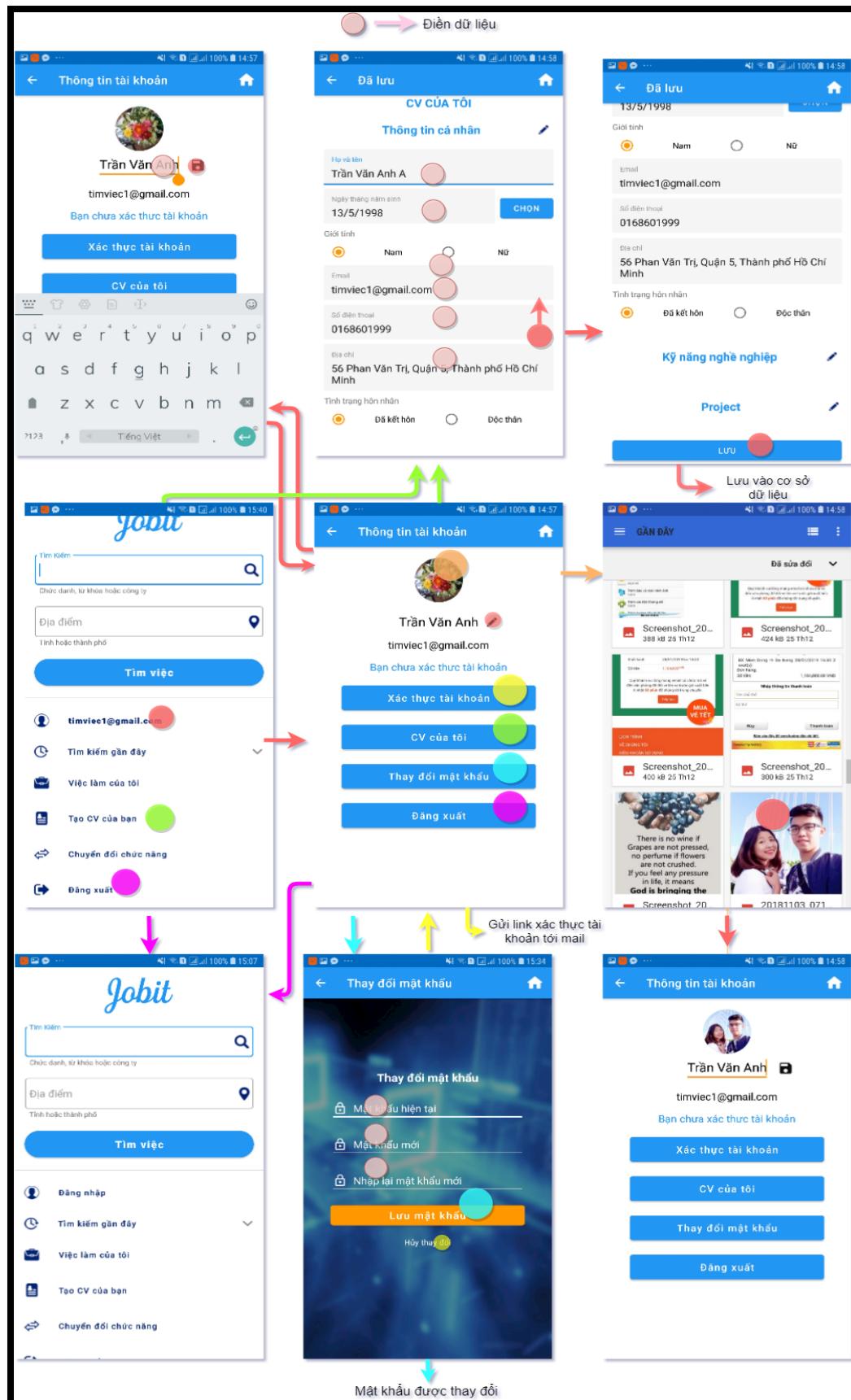
1. Structure and Navigation
 - a. Người tìm việc

Report 3# Final Report



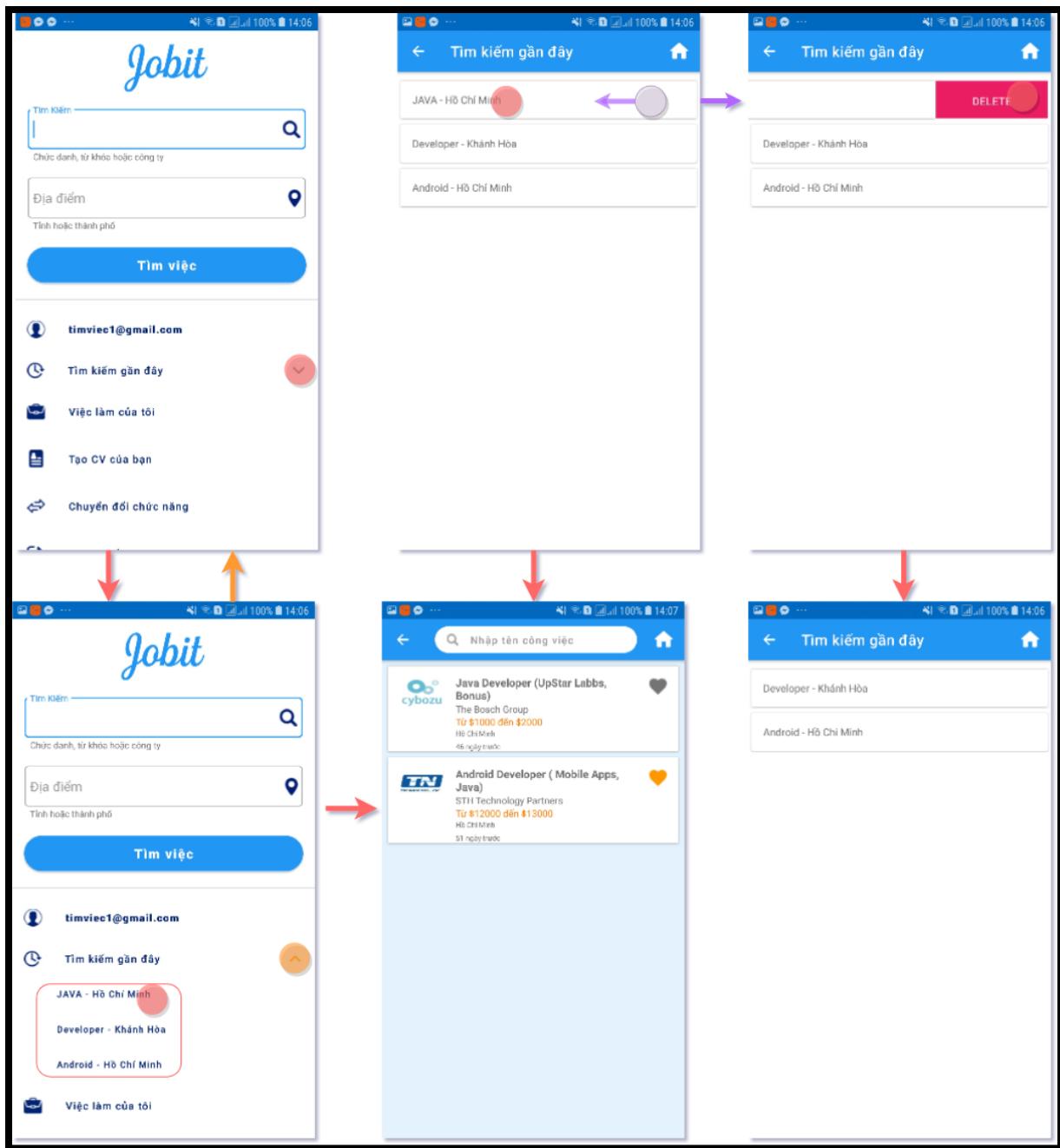
Hình 28: Tài khoản

Report 3# Final Report



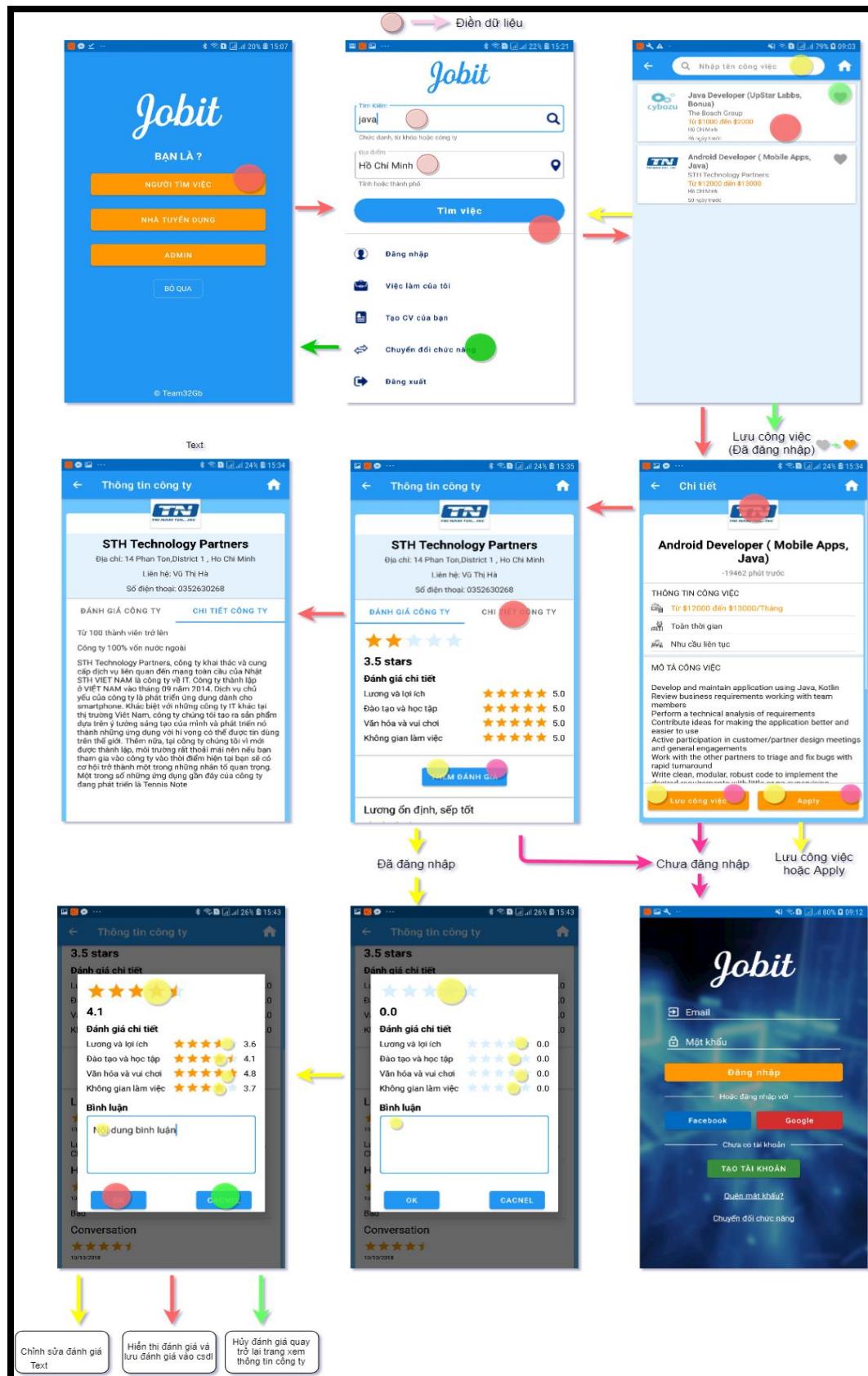
Hình 29: Chính sửa thông tin Account

Report 3# Final Report



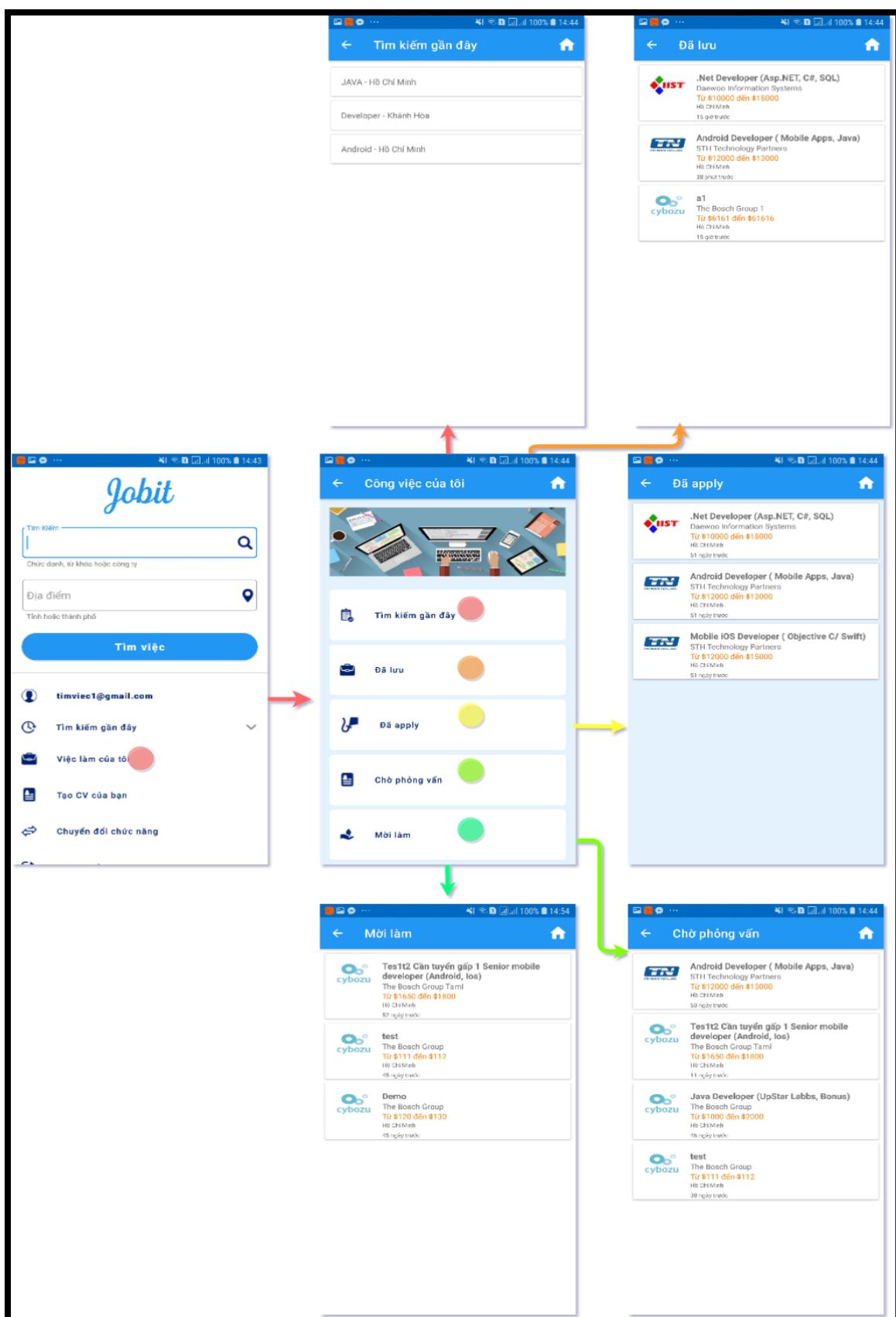
Hình 30: Tìm kiếm gần đây

Report 3# Final Report



Hình 31: Xem thông tin công ty

Report 3# Final Report



Hình 32: Việc làm của tôi

Report 3# Final Report

b. Nhà tuyển dụng



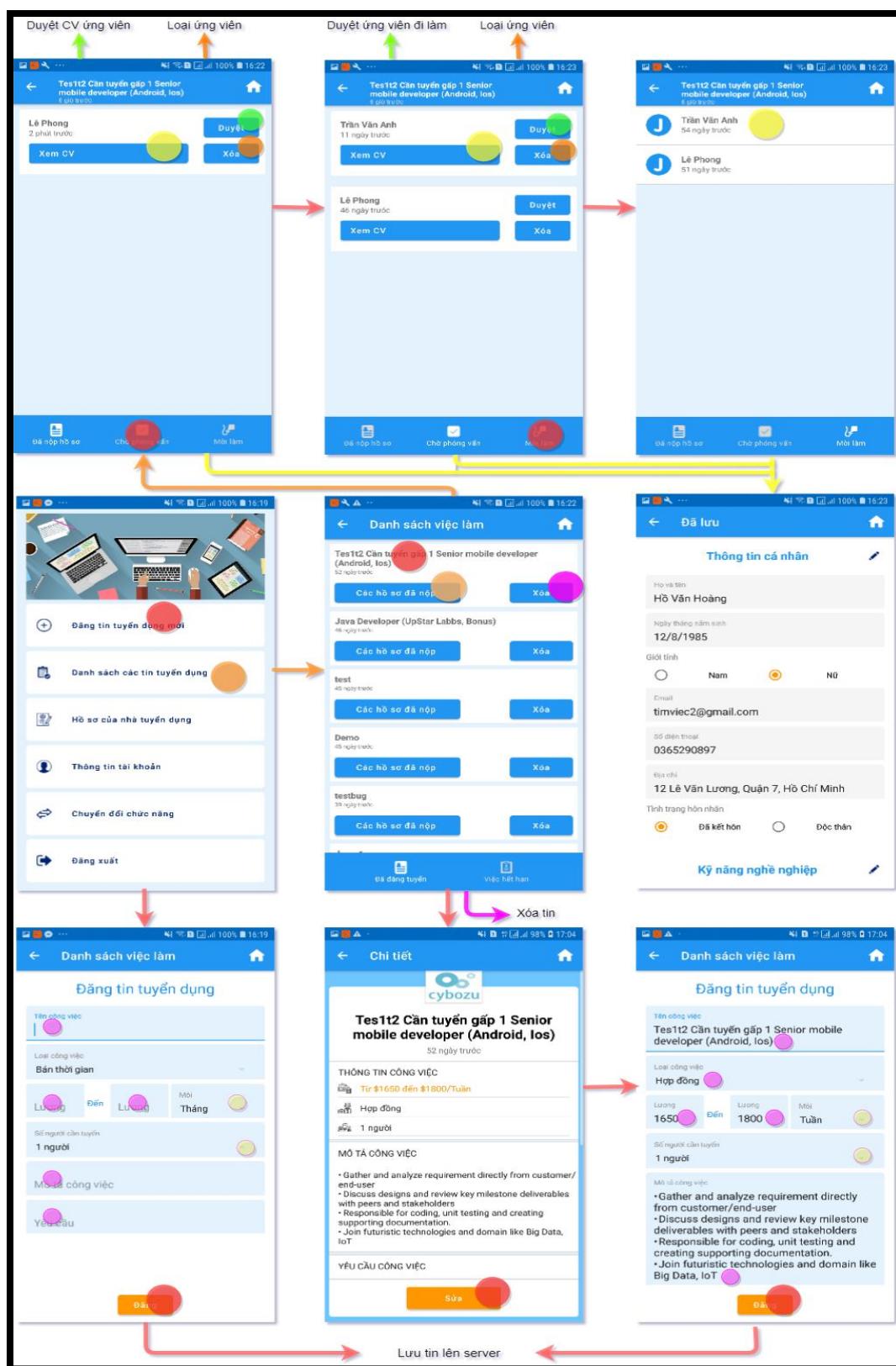
Hình 33: Nhà tuyển dụng

Report 3# Final Report



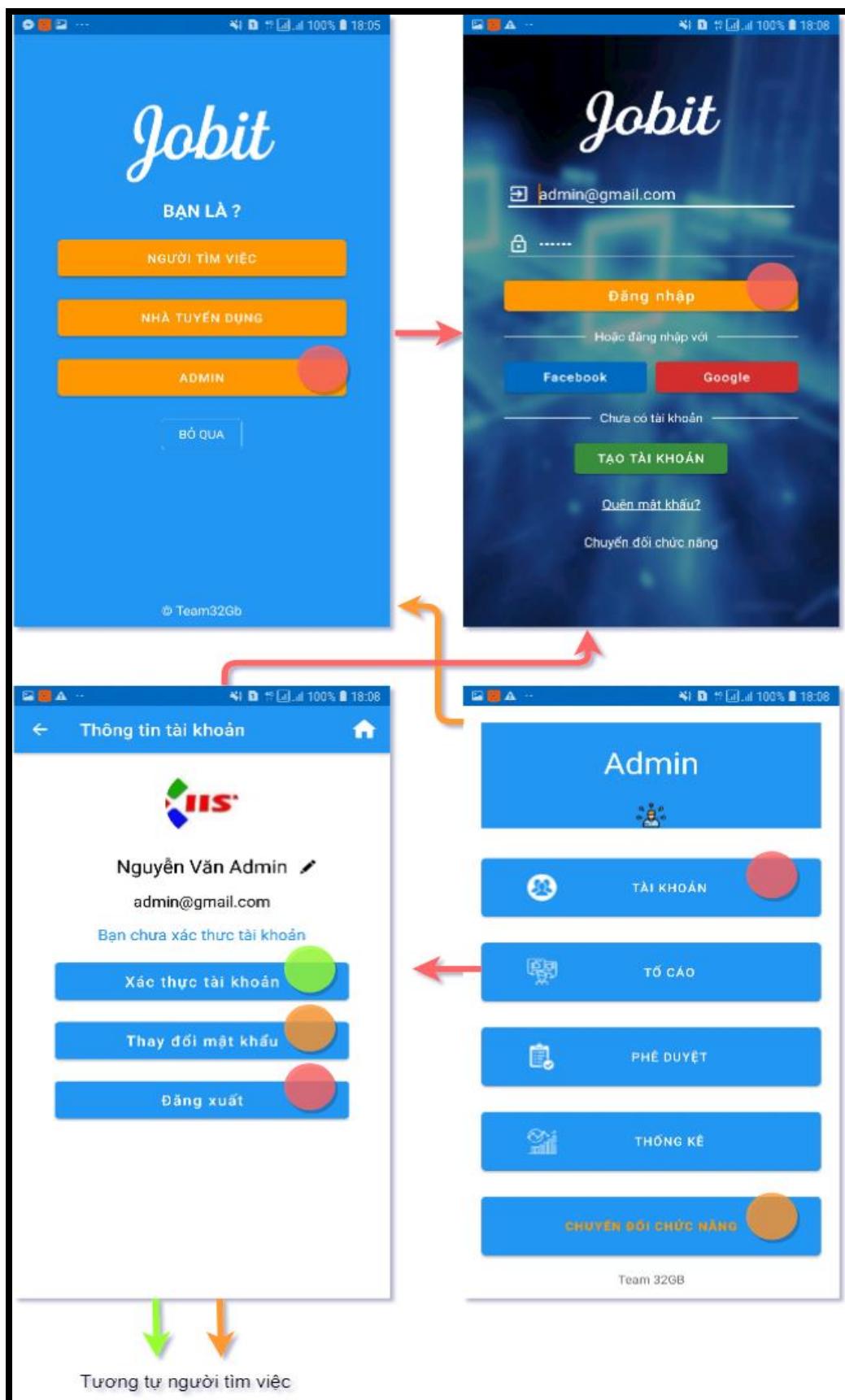
Hình 34: Thông tin nhà tuyển dụng

Report 3# Final Report



Hình 35: Đăng tin và xem danh sách ứng viên
c. Admin

Report 3# Final Report



Hình 35: Tài khoản admin

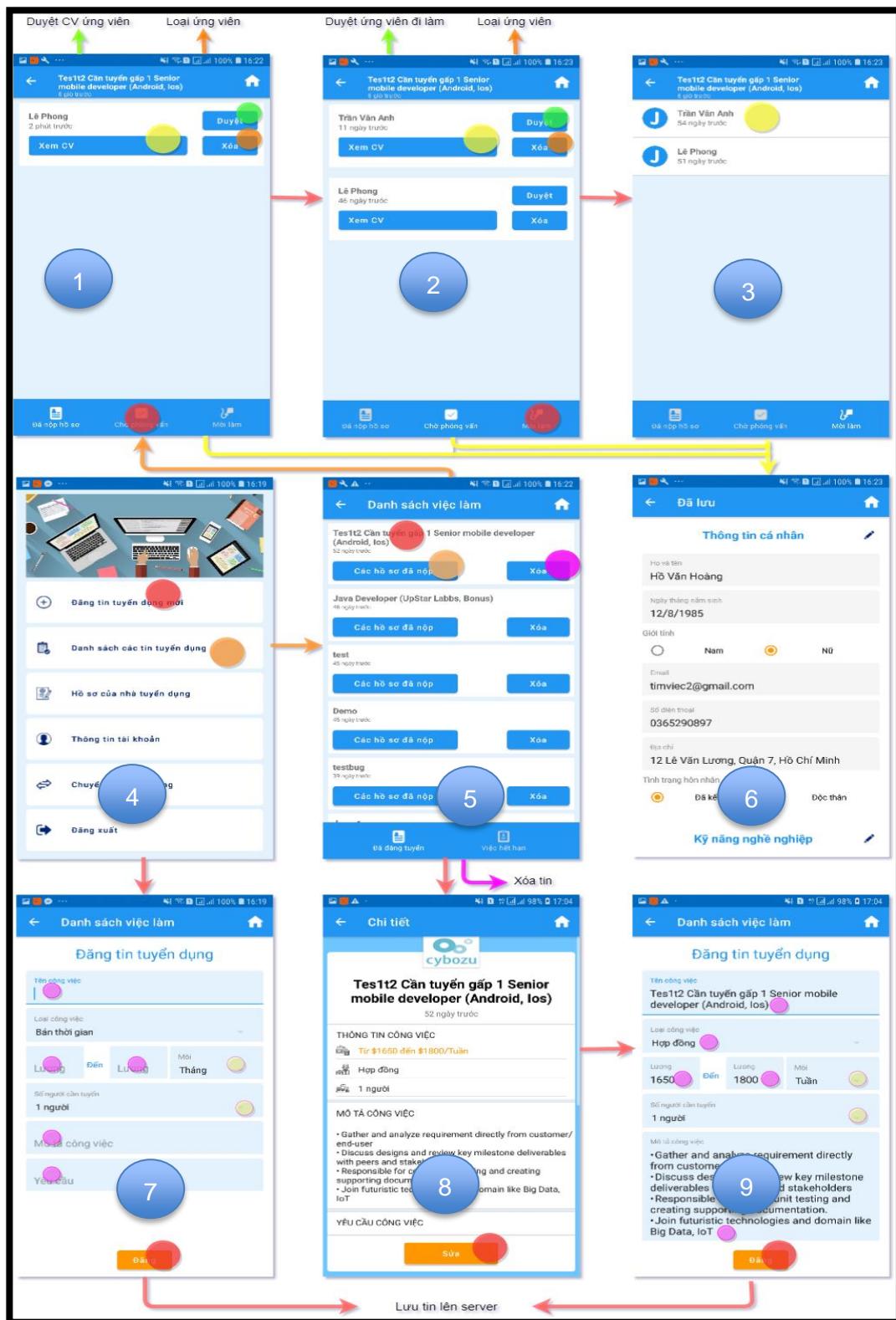
Report 3# Final Report



Hình 36: Admin duyệt hồ sơ và cảnh cáo

Report 3# Final Report

2. Specification



Hình 37: Đăng tin và duyệt ứng viên

- Hình 1: Khi nhà tuyển dụng bấm nút “Duyệt” thì ứng viên sẽ chuyển qua phần chờ phỏng vấn.

Report 3# Final Report

- Hình 2: Nếu nhà tuyển dụng bấm “Duyệt” thì ứng viên sẽ được chuyển qua phần mời làm (Hình 3). Ngược lại nếu người dùng bấm nút “Xóa” tại hình 1 thì ứng viên sẽ bị loại khỏi công việc đó. Ngoài ra, nhà tuyển dụng có thể xem được CV của ứng viên nộp vào bằng cách nhấn nút “xem CV” khi đó sẽ chuyển qua hình 6.
- Hình 6: Chứa thông tin chi tiết CV của người tìm việc đã apply vào công việc.
- Hình 4: Khi nhà tuyển dụng có nhu cầu đăng một tin tuyển dụng mới, người dùng nhấn nút “Đăng tin tuyển dụng” thì sẽ chuyển qua hình 7
- Hình 7: Nhà tuyển việc nhập các thông tin cần thiết vào các EditText thì tin tuyển dụng đó sẽ được lưu trên server đồng thời tin tuyển dụng đã đăng sẽ được hiển thị trong hình 5.
- Hình 5: Nhà tìm việc có thể xem danh sách các tin đã đăng bằng cách nhấn nút “Danh sách các tin đã đăng ở hình 4”, nhà tuyển dụng có thể xem chi tiết tin đã đăng bằng cách nhấn vào các tin, khi đó sẽ chuyển qua hình 8. Nhà tìm việc có thể sửa thông tin các tin đã đăng bằng cách nhấn nút sửa khi đó sẽ chuyển qua hình 9.
- Hình 9: Nhà tìm việc sửa thông tin đã đăng, khi sửa xong và nhấn nút đăng thì tin tuyển dụng sẽ được cập nhật lại trên server và trong danh sách các tin đã đăng.

XI. Design of Tests

1. List Test case

Tên testcase	Usecase liên quan	Ngữ cảnh	Input	Output mong đợi	Các bước thực hiện
Đăng ký	Chọn đối tượng người dùng (phân quyền người dùng)	Khi người dùng muốn tạo tài khoản để sử dụng các chức năng của app	TH1: Tên, gmail, mật khẩu, nhập lại mật khẩu	Đăng ký thành công và chuyển qua giao diện home Lưu thông tin trên cơ sở dữ liệu (Firebase)	Chọn đối tượng người dùng -> Chọn button tạo tài khoản -> Nhập dữ liệu cần thiết (Nhập lại mật khẩu)

Report 3# Final Report

			TH2: Nhập mật khẩu ít hơn 6 ký tự	Có xuất hiện thông báo mật khẩu phải dài hơn 6 ký tự	phải trùng với mật khẩu trước đó) -> Nhấn đăng ký
			TH3: Mật khẩu trong phần nhập lại khác với mật khẩu đã nhập	Có xuất hiện thông báo mật khẩu phải trùng với mật khẩu đã nhập	
			TH4: Nhập gmail sai định dạng	Có xuất hiện thông báo gmail sai định dạng	
			TH5: Nhập dữ liệu không đầy đủ mà nhấn nút đăng ký	Có xuất hiện thông báo nhập dữ liệu còn thiếu	
Quên mật khẩu	Người dùng nhập mật khẩu sai quá 3 lần	Khi người dùng thực hiện chức năng đăng nhập vào app	Nhập vào gmail và gmail phải tồn tại trên cơ sở dữ liệu	Có đường link gửi về gmail khi người dùng bấm vào đường link có 2 editText để người dùng nhập lại mật khẩu	Nhấn vào “Quên mật khẩu” -> nhập gmail -> kiểm tra gmail -> Nhập vào đường link -> nhập mật khẩu mới -> kiểm tra mật khẩu -> thay đổi mật khẩu thành công

Report 3# Final Report

Đánh giá nhà tuyển dụng	<ul style="list-style-type: none"> - Xem thông tin nhà tuyển dụng - Đăng nhập 	<p>Khi người tìm việc xem thông tin nhà tuyển dụng và muốn đánh giá nhà tuyển dụng.</p>	- TH1: Đánh giá sao và không comment	Thông báo lỗi: Comment không được rõ ràng	<ol style="list-style-type: none"> 1. Đăng nhập bằng tài khoản người tìm việc 2. Tìm việc 3. Xem thông tin nhà tuyển dụng 4. Đánh giá 5. Xem hiển thị đánh giá
			-TH2: Đánh giá sao và có comment	-Lưu đánh giá lên CSDL -Hiển thị đánh giá lên một cách chính xác so với input	
Tìm kiếm		<p>-Người tìm việc muốn tìm việc</p> <p>-Có 2 keyword cần nhập vào trước khi tìm kiếm: tên công việc và địa điểm</p>	TH1: Keyword rỗng	Thông báo lỗi: "Vui lòng nhập keyword"	<ol style="list-style-type: none"> 1. Nhập keyword 2. Nhấn button "Tìm kiếm" 3. Xem kết quả tìm kiếm
			TH2: Nhập tên công việc nhưng không nhập địa điểm	Hiển thị list công việc liên quan đến keyword công việc vừa nhập trên tất cả các địa điểm	
			TH3: Không nhập tên công việc và chỉ nhập tên địa điểm	Hiển thị list tất cả công việc trên địa điểm là keyword vừa nhập	

2. Phạm vi test

- Test giao diện
 - + Sự tương tác giữa các view
 - + Sự tương tác giữa người dùng với view
- Test hệ thống

Report 3# Final Report

- + Sự tương tác giữa các class
- + Sự tương tác giữa các component với API
- + Sự tương tác thích trên các thiết bị

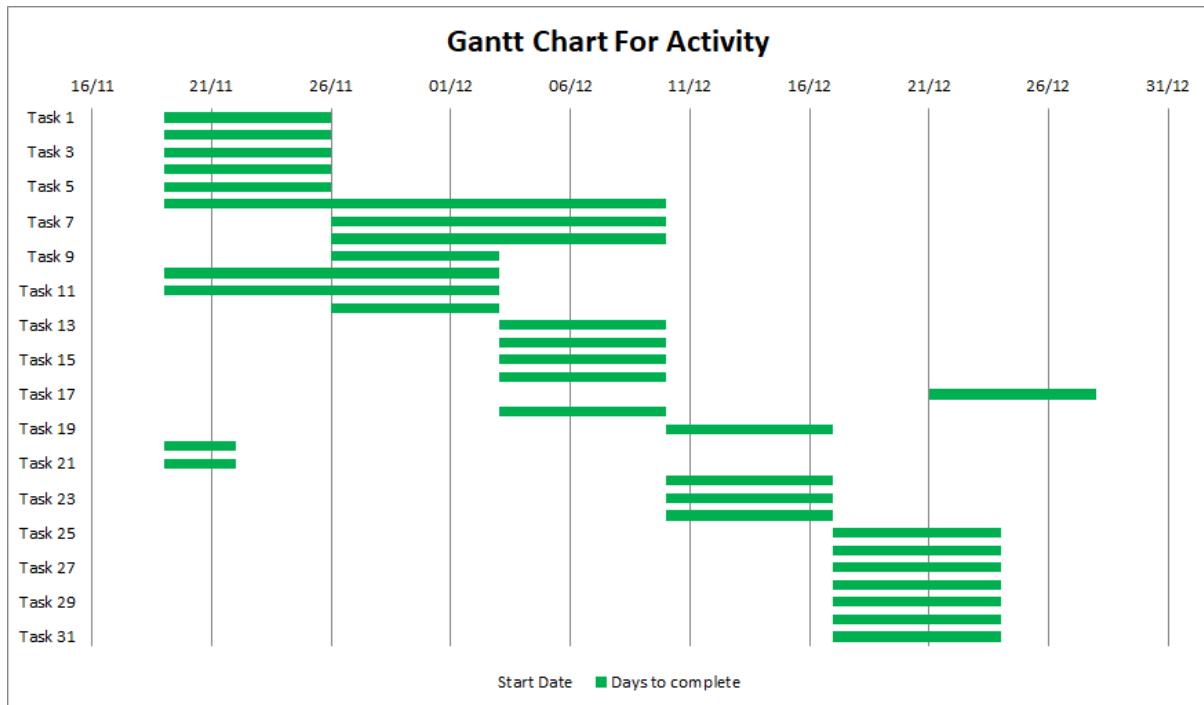
3. Kế hoạch test

- Ngôn ngữ sử dụng: Java
- Framework: iUnit, kiểm thử tự động
- IDE: Android
- Kiểm thử đơn vị:
 - + Kiểm tra các class đơn lẻ, phương thức đơn lẻ
 - + Kiểm tra sự chuyển đổi giữa các trạng thái
- Kiểm thử component
- User testing: Build ra bản apk, đưa cho người khác sử dụng trên các thiết bị khác nhau.

XII. History of Work, Current Status, and Future Work

History of work

Report 3# Final Report



Task	Mô tả	Người thực hiện
1	Làm báo cáo System Design	Tất cả
2	Làm chức năng xác thực người dùng	Nguyễn
3	Tạo cơ sở dữ liệu	Linh, Lạng
4	Làm chức năng lấy ảnh/ lưu ảnh avatar	Nguyễn
5	Làm chức năng tìm kiếm công việc	Nguyễn
6	Làm chức năng đánh giá	Nguyễn
7	Làm chức năng cho người tìm việc tố cáo người tìm việc	Linh
8	Làm chức năng cho người tìm việc tố cáo nhà tuyển dụng	Linh
9	Làm chức năng người tìm việc lưu tin tuyển dụng	Linh
10	Làm chức năng đăng tin tuyển dụng	Linh
11	Làm chức năng xóa tin tuyển dụng đã lưu	Linh

Report 3# Final Report

12	Chỉnh sửa giao diện người tìm việc	Linh, Lạng
13	Chỉnh sửa giao diện nhà tuyển dụng	Nguyễn, Lạng
14	Chỉnh sửa giao diện login	Nguyễn
15	Làm chức năng tạo hồ sơ nhà tuyển dụng	Linh
16	Làm chức năng sửa hồ sơ nhà tuyển dụng	Linh
	Làm chức năng xem hồ sơ nhà tuyển dụng	Nguyễn
17	Test	Lạng
18	Làm chức năng apply	Nguyễn, Linh
19	Tìm hiểu các công nghệ, kỹ thuật cần thiết để áp dụng vào đồ án	Nguyễn
20	Tìm hiểu để đồng bộ nhiều máy cùng làm việc với 1 project trên firebase	Nguyễn
21	Tìm hiểu và triển khai mô hình MVP cho nhóm	Nguyễn
22	Làm chức năng duyệt hồ sơ người tìm việc	Linh
23	Làm chức năng xem hồ sơ người tìm việc	Linh
24	Làm chức năng gửi thông báo có ứng viên apply	Nguyễn
25	Làm chức năng gửi thông báo hồ sơ apply đã được duyệt	Nguyễn
26	Làm chức năng gửi thông báo hồ sơ phê duyệt từ admin	Lý
27	Làm chức năng gửi thông báo tố cáo từ admin	Lý
28	Làm chức năng xóa/gửi tố cáo từ admin	Lý
29	Làm chức năng phê duyệt hồ sơ của admin	Lý
30	Làm chức năng thống kê của admin	Lý
31	Chỉnh sửa giao diện admin	Lý, Lạng

Report 3# Final Report

Future work

XIII. References

Mô hình MVP:

<https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93presenter>

Firebase:

<https://firebase.google.com/>

Kiến trúc Client-Server

https://en.wikipedia.org/wiki/Client%E2%80%93server_model