|  |
| --- |
| **FPT- APTECH COMPUTER EDUCATION** |
| eProject Report |
| Team 3 |
|  |
| |  |  | | --- | --- | | **Team 3** | | | **Group Member** | Nguyễn Xuân Trường  Đặng Thị Thu Hà  Trần Công Vũ  Nguyễn Chí Công  Vũ Trường | | **Batch** | Bot work | | **Center** | FPT APTECH HN | | **Semester** | 4 | |
|  |

*<Remember to update the table of contents>*

**Contents**

[1.](#_gjdgxs) Acknowledgements 4

[2.](#_30j0zll) Introduction 4

[3.](#_1fob9te) Problem Definition 4

[3.1.](#_3znysh7) Problem Abstraction 4

[3.2.](#_2et92p0) The Current System 4

[3.3.](#_tyjcwt) The Proposed System 4

[3.4.](#_3dy6vkm) Boundaries of the System 4

[3.5.](#_1t3h5sf) Development Environment 4

[4.](#_4d34og8) Requirements and Business Flows 5

[4.1.](#_2s8eyo1) Customer Requirements 5

[4.2.](#_17dp8vu) Activity Diagram(Optional) 5

[4.3.](#_3rdcrjn) Use Case Diagram 5

[4.4.](#_26in1rg) Use Case Specification (optional) 5

[4.5.](#_lnxbz9) Other Concerns<Optional> 6

[5.](#_35nkun2) Design 7

[5.1.](#_1ksv4uv) System Architecture 7

[5.2.](#_44sinio) Class Diagram 7

[5.3.](#_2jxsxqh) Class Diagram Explanation 7

[5.4.](#_z337ya) Sequence Diagram 7

[5.5.](#_3j2qqm3) Collaboration Diagram (Optional) 7

[5.6.](#_1y810tw) State Diagram (Optional) 7

[5.7.](#_4i7ojhp) Entity Relationship Diagram 7

[5.8.](#_2xcytpi) Database Design 7

[5.9.](#_1ci93xb) Algorithms (optional) 8

[5.10.](#_3whwml4) Others (optional) 8

[6.](#_2bn6wsx) System Prototype 9

[6.](#_qsh70q) Management and Project Planning 10

[6.1.](#_3as4poj) Management Approach 10

[6.2.](#_1pxezwc) Project Plan 10

[6.3.](#_49x2ik5) Task Sheet 10

[6.4.](#_2p2csry) Meeting Minutes (Optional) 10

[7.](#_147n2zr) Checklists 11

[7.1.](#_3o7alnk) Check List of Validation 11

[7.2.](#_23ckvvd) Submission Checklist 11

[8.](#_ihv636) Screenshots 11

[9.](#_32hioqz) Coding Convention 11

[10.](#_1hmsyys) User Guide 12

[11.](#_41mghml) Other Concerns<Optional> 13

[Appendix 14](#_2grqrue)

[Glossary [Optional] 14](#_vx1227)

[References [Optional] 14](#_3fwokq0)

[Others<Optional> 14](#_1v1yuxt)

# Acknowledgements

# Introduction

*+ Bot cv is an online recruitment platform that helps find jobs in the digital age.*

*+ Helping students and employees wishing to change jobs or find a new environment that suits themselves.*

*+Make it easy for employers of companies to find suitable candidates for their companies*

# Problem Definition

## Problem Abstraction

*+There are 3 types of account (admin, recruitment, candidate).*

*+Admin: The website manager, managing everything related to information of employers and candidates.*

*+Recruitment: An employer is the person who posts recruiting articles and jobs on the website in order to find suitable candidates.*

*+Candidate: Candidates are those in need of finding a stable, good job through posts and job vacancies to develop themselves as a stable source of income.*

## The Current System

*The system is managed by many users, ensuring stability.*

## The Proposed System

*The system is written in the C # programming language over a cross-platform dotnet platform and uses Postgres DB*

## Boundaries of the System

*There is no function of payment via visa card*

## Development Environment

*backend: C # dotnet core + postgres*

*frontend: ReactJs*

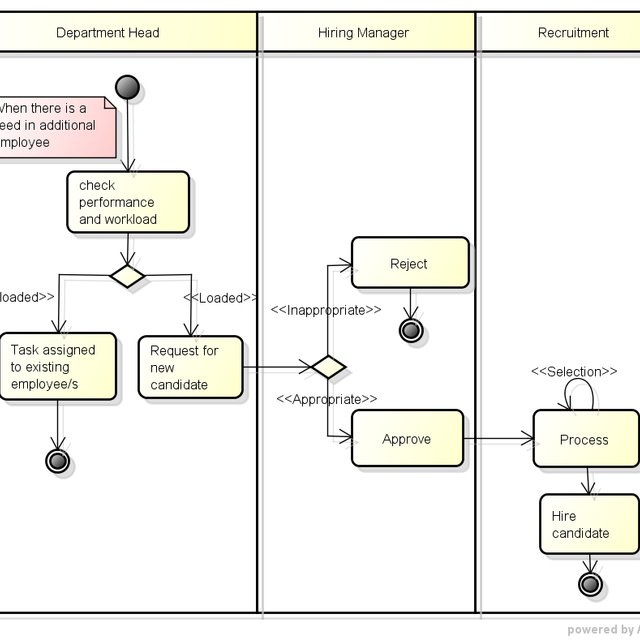
# Requirements and Business Flows

* *admin :*
* *Filter verify candidates*
* *Employer account management (in progress)*
* *Candidate account management*
* *Post news articles*
* *Recruiment :*
* *Post recruitment articles to find candidates*
* *Post new news.*
* *Log in to the system.*
* *Receive application notification from candidates*
* *Accept the application request, the candidate will be added to the Job.*
* *Search for suitable candidates (in progress)*
* *Candidate :*
* *See available Jobs that employers have posted*
* *See job details*
* *Apply for a job that suits you*

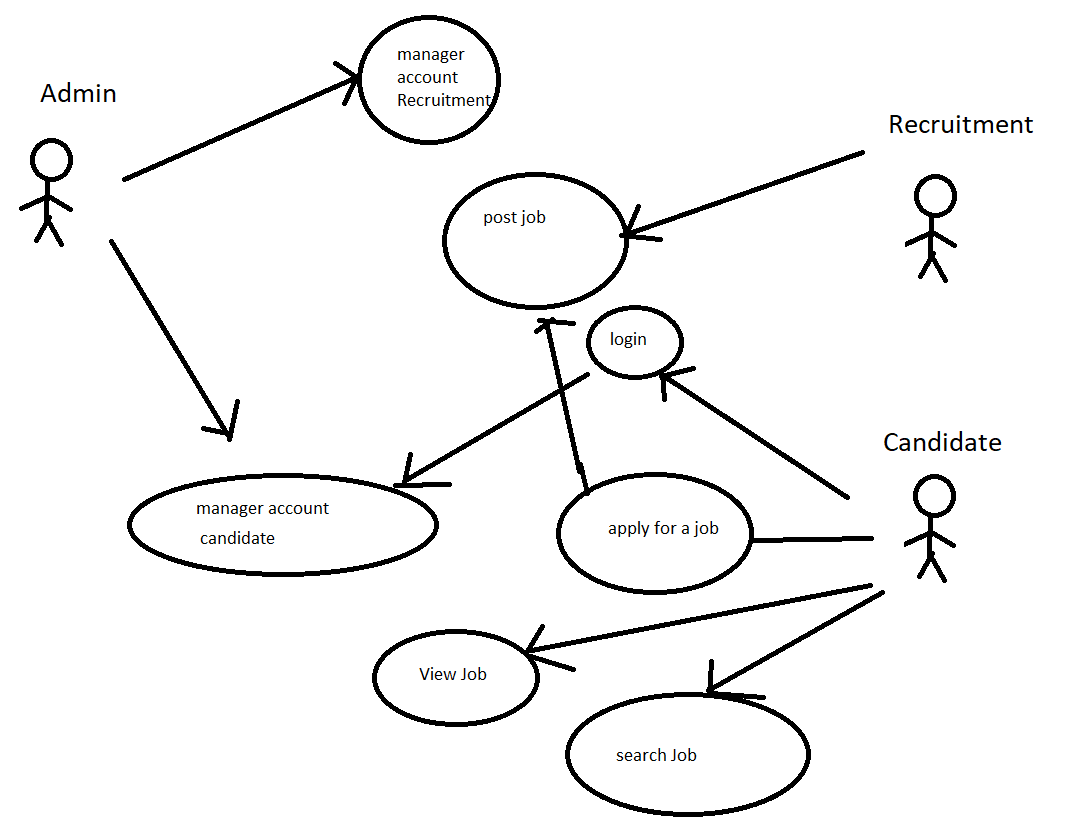
## Customer Requirements

* *Admin wants to manage multiple accounts and calculate revenue per day.*

## Activity Diagram

**

## Use Case Diagram

**

## Use Case Specification (optional)

*<Write down all non-trivial use cases. This should reflect what you get when your team does the system analysis. Use the template to write the detailed specification for use cases>*

*<Use case temlpate:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **USE CASE SPECIFICATION** | | | | |
| **Use-case No.** | <UC001> | **Use-case Version** | | <1.0> |
| **Use-case Name** | <Name> | | | |
| **Author** | <Members> | | | |
| **Date** | Dd/mm/yyyy | **Priority** | <High\Normal\Low> | |
| **Actor:**  <Lit all actors>  **Summary:**  *<Briefly describe the use case>*  **Goal:**  *<Briefly describe the goal of use case>*  **Triggers**  *<What leads this use case?>*  **Preconditions:**  *<List the required pre-conditions for this use case>*  **Post Conditions:**  *<List the required post-condition for this use case>*  **Main Success Scenario:**  *<List the main steps for this use case to reach the goal successfully >*    **Alternative Scenario:**  *<List the other steps for this use case to reach the goal in some alternatives condition >*  **Exceptions:**  *<list the exceptions of this use case>*  **Relationships:**  *<List the relationships that use case relates to>*  **Business Rules:**  *<Any concern about the business>* | | | | |

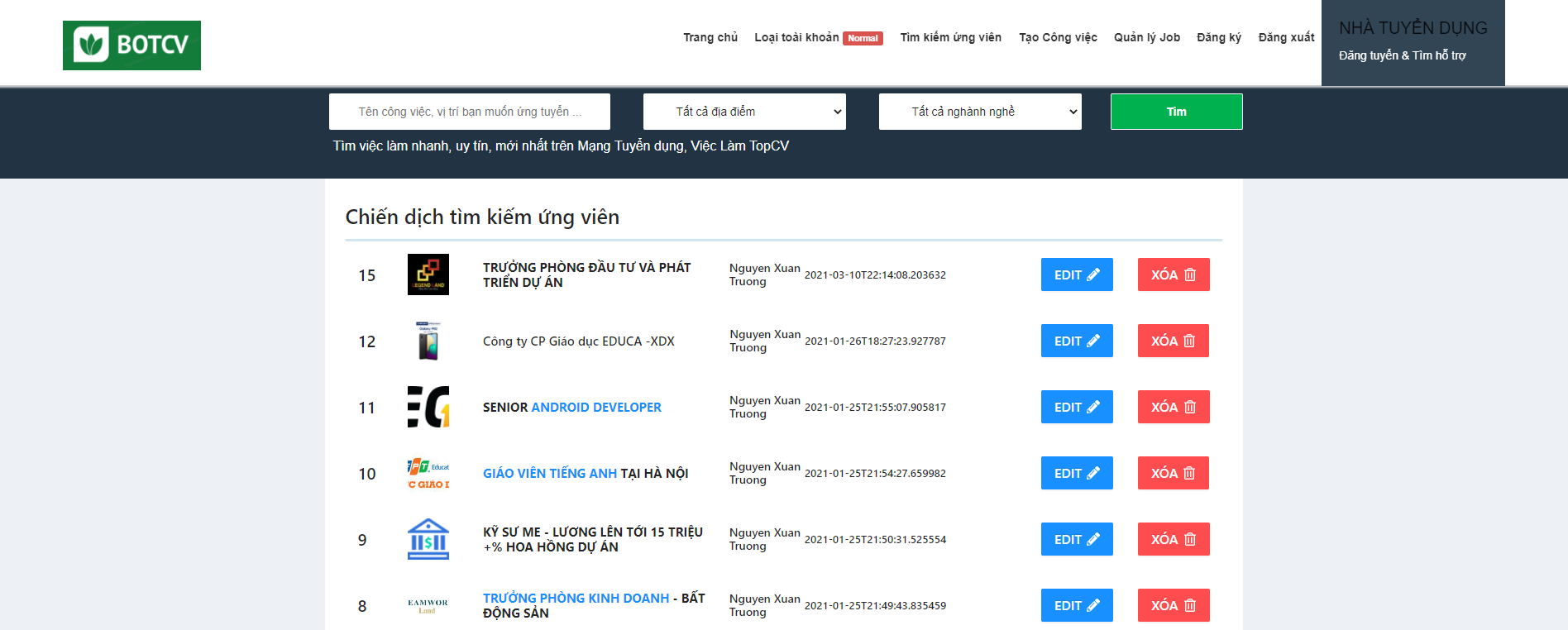
## Other Concerns<Optional>

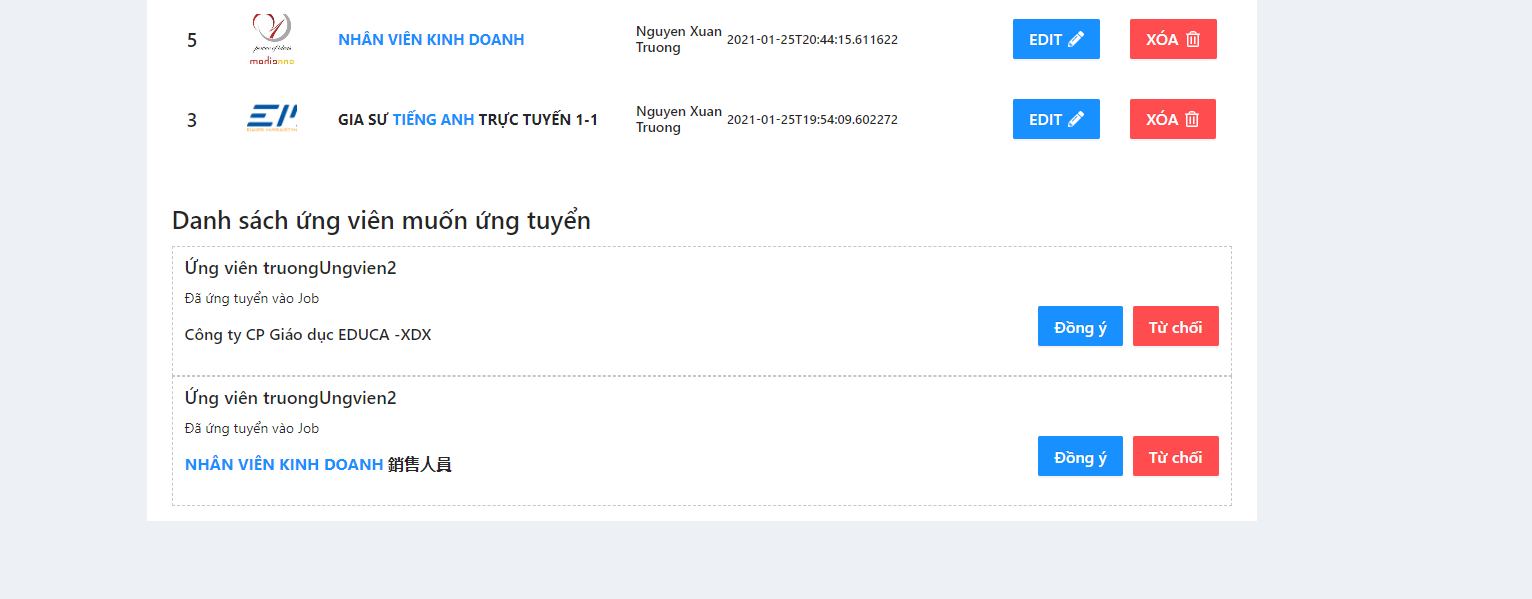
*<You can list here all other concerns about the business or the requirements if needed>*

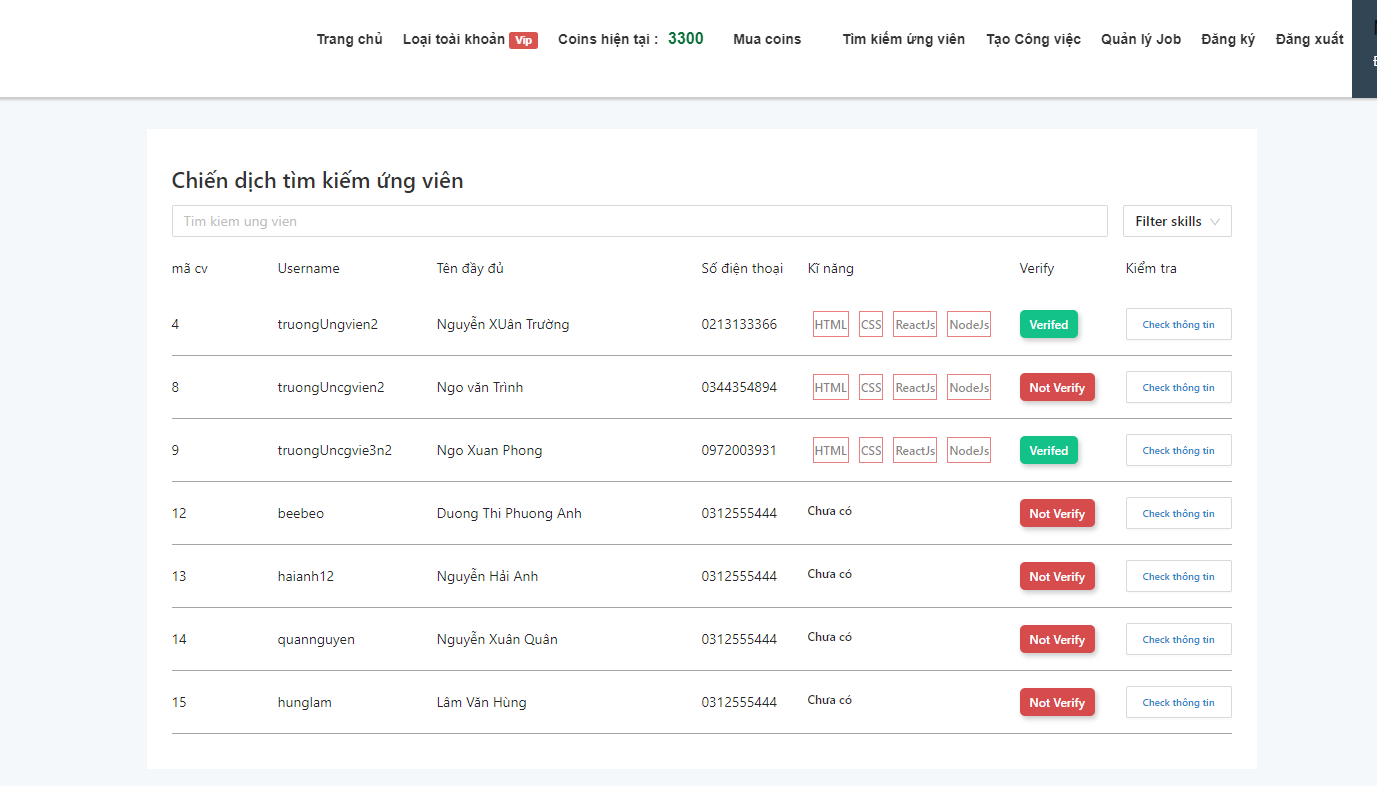
# Design

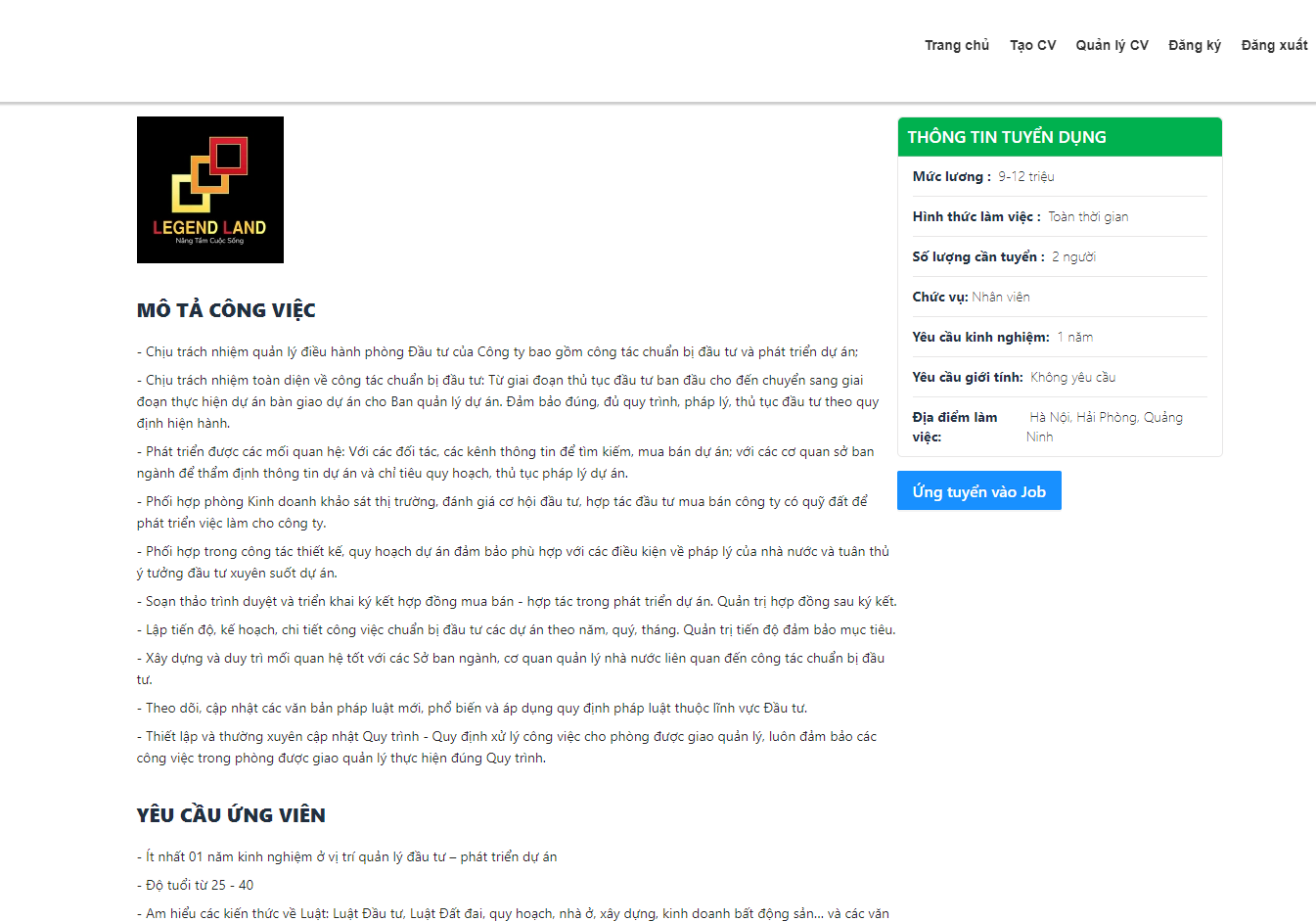
* Home page

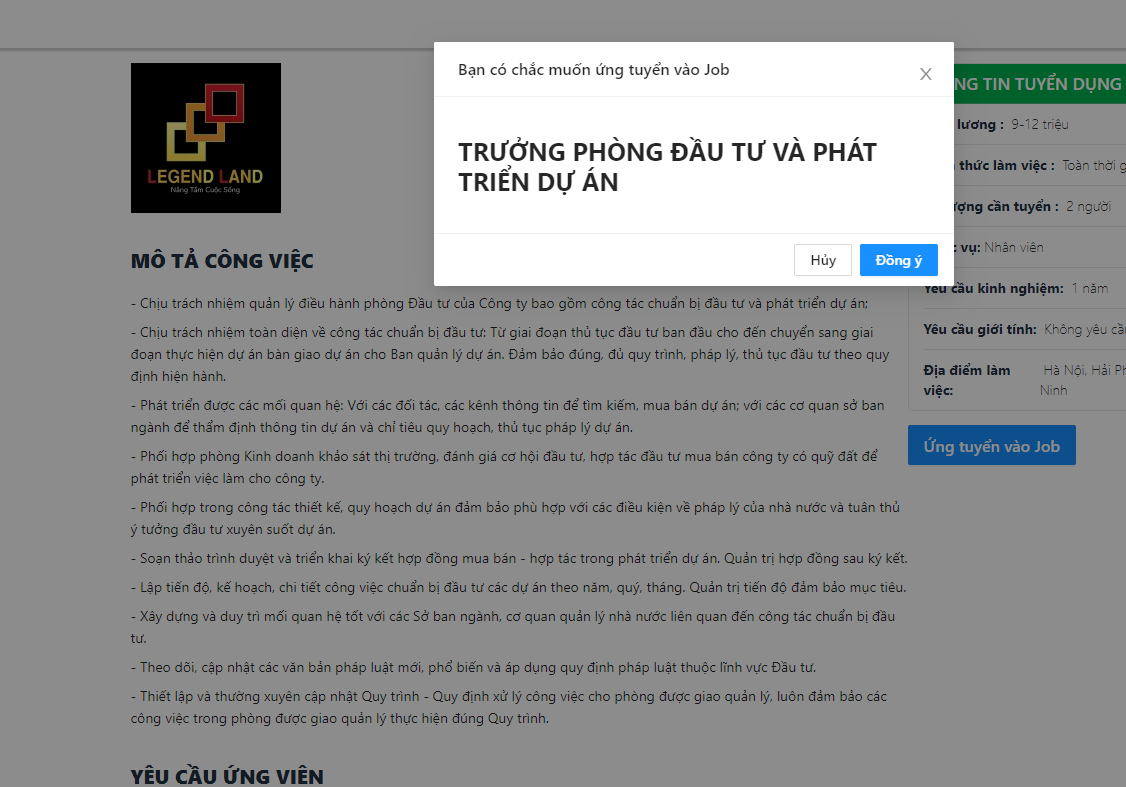
**

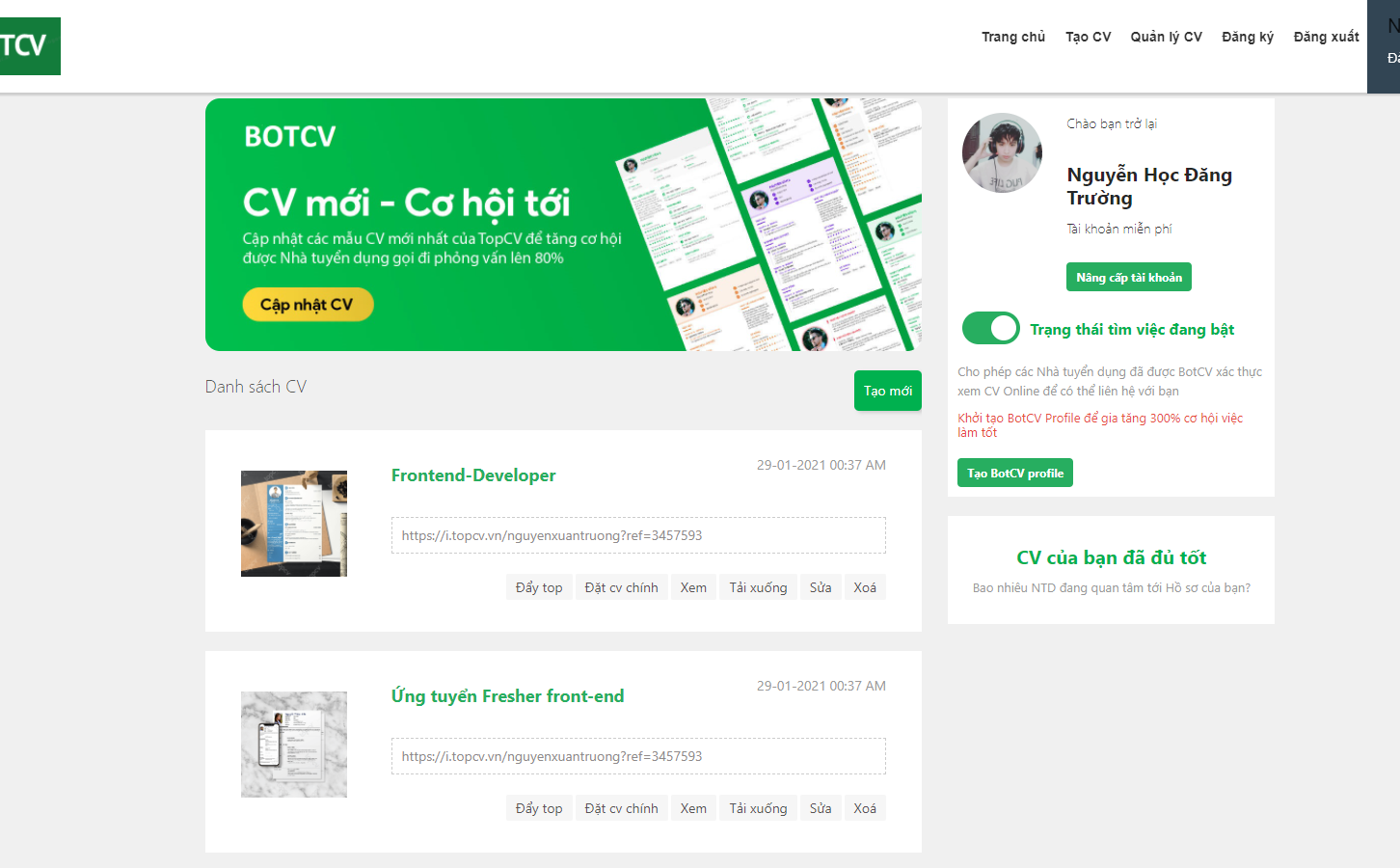
**

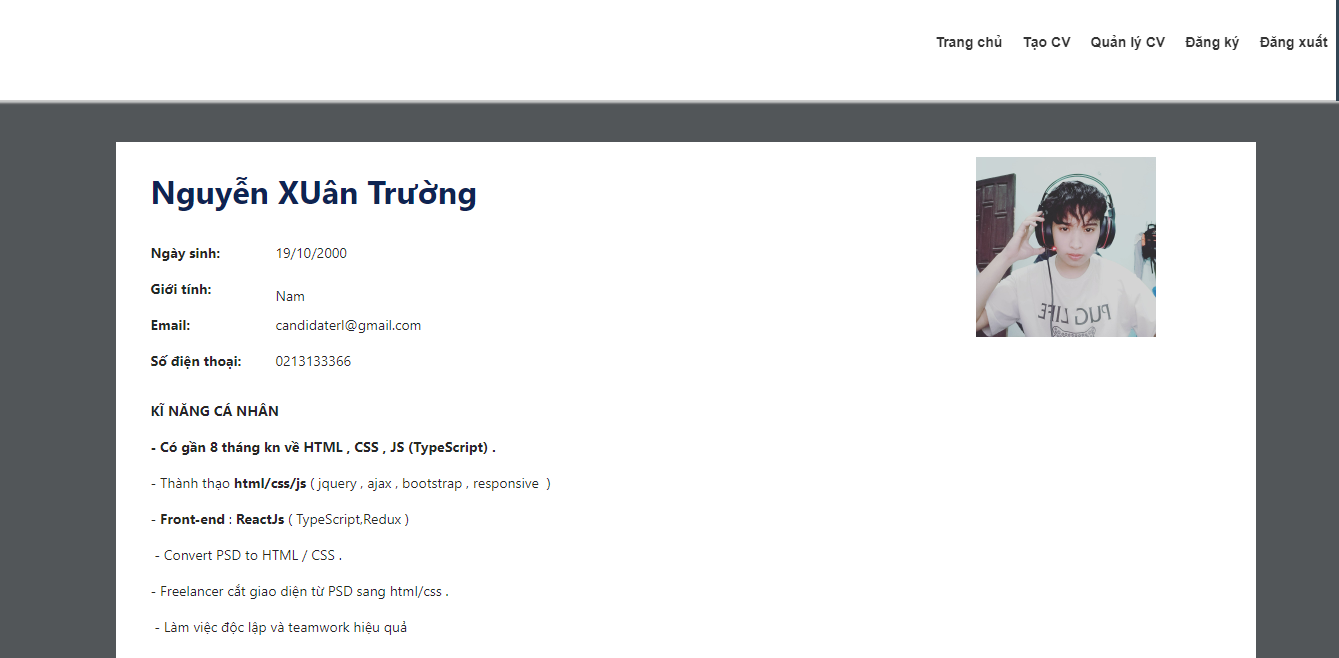
**

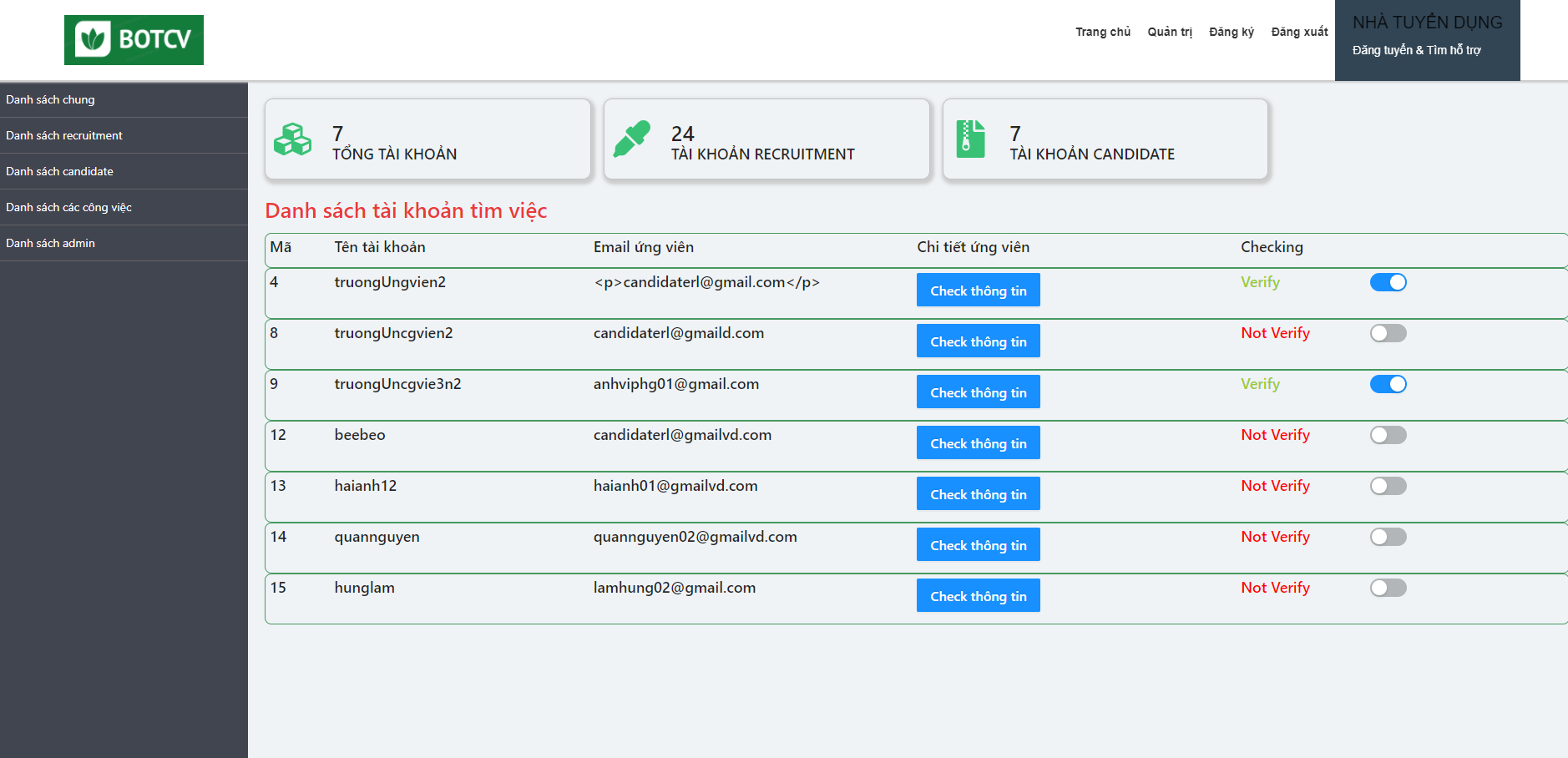
**

**

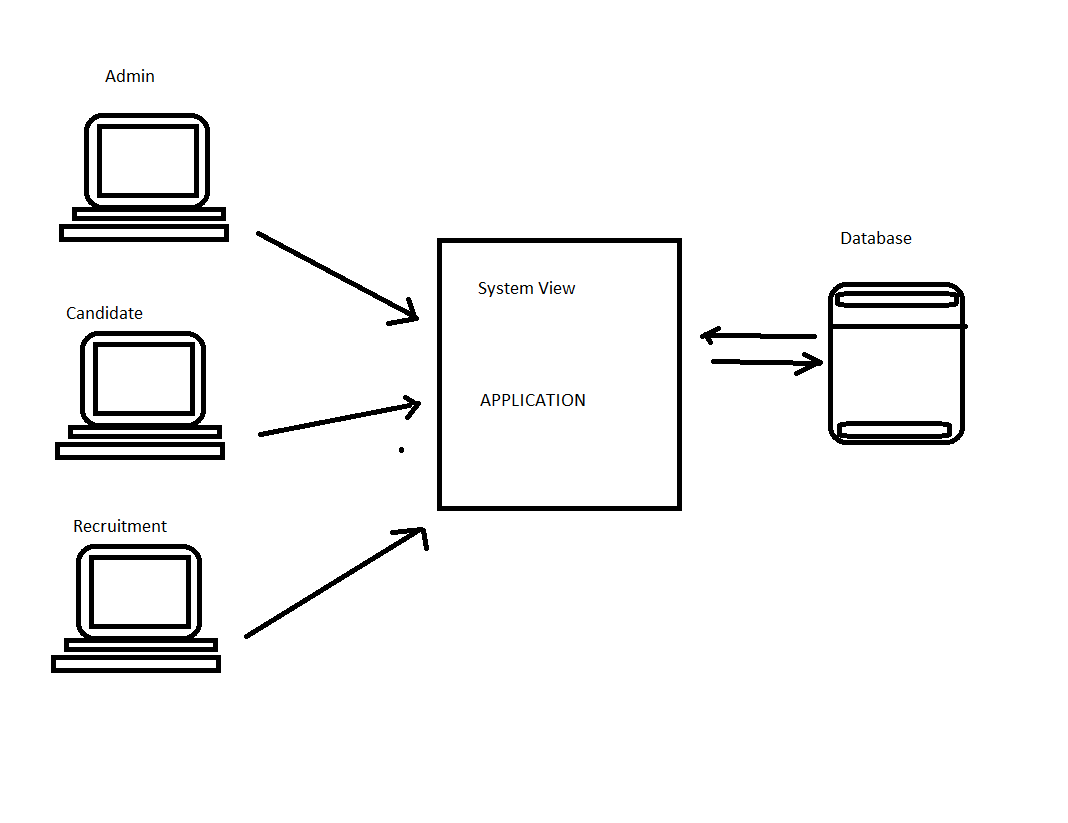
**

**

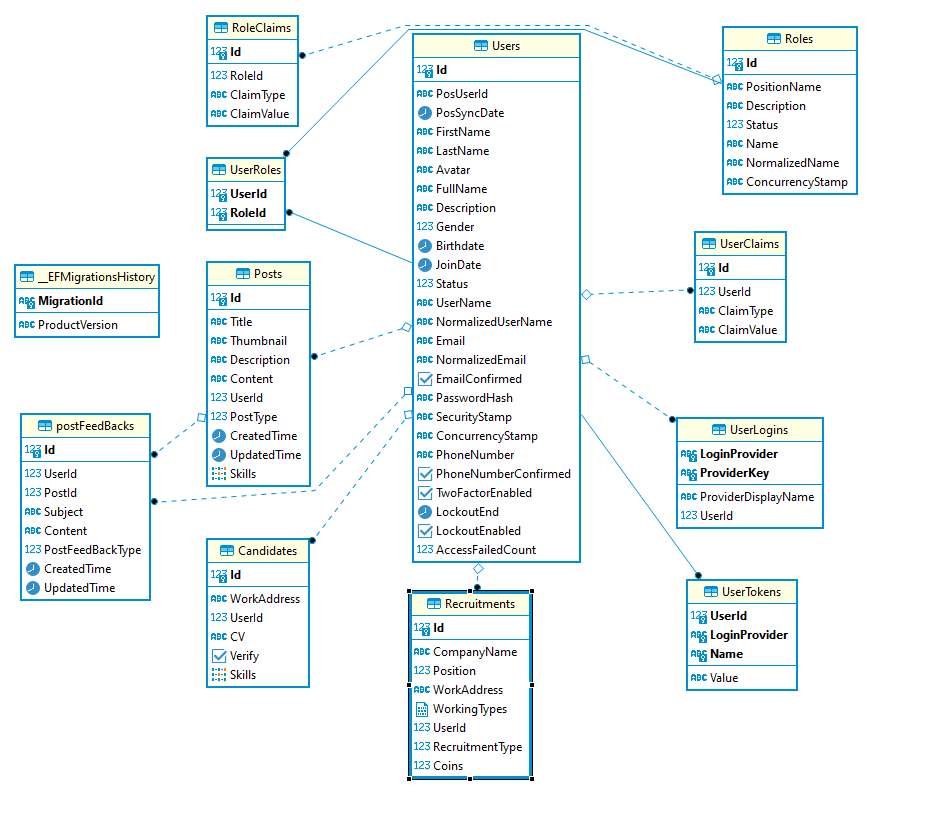
**

**

## System Architecture

**

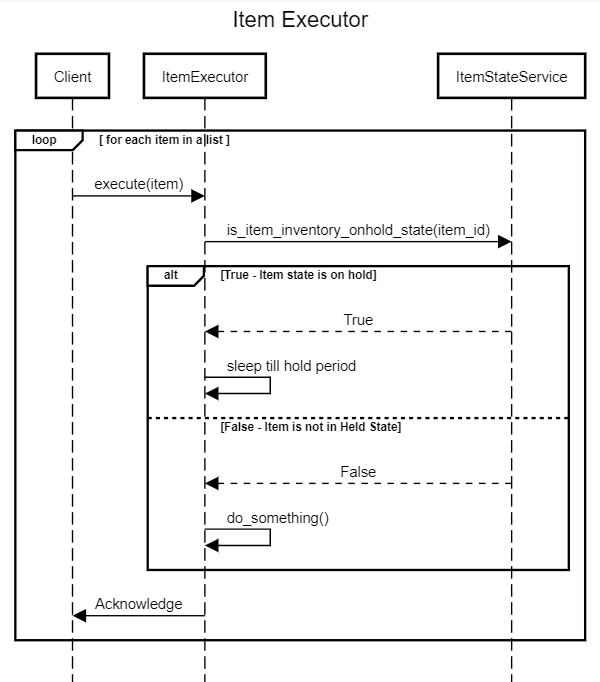
## Class Diagram

**

## Class Diagram Explanation

* *UserLogin : login account*
* *postFeedBacks: recruitment to receive candidate recruitment notices*
* *Post : Manage recruitment posts*
* *Users:Manage candidate list*
* *Roles: Decentralize login system*
* *Recruitments: Management of the recruitment list*
* *Candidates: Management of the candidate list*

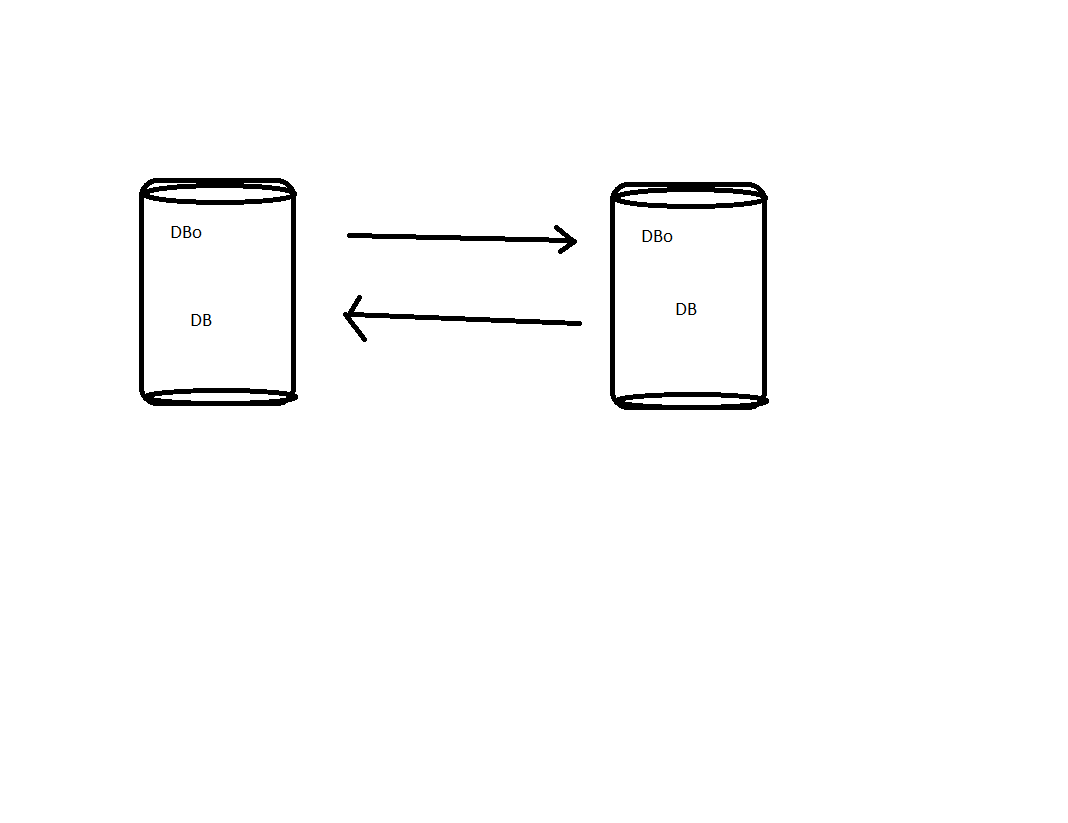
## Sequence Diagram (Optional)

**

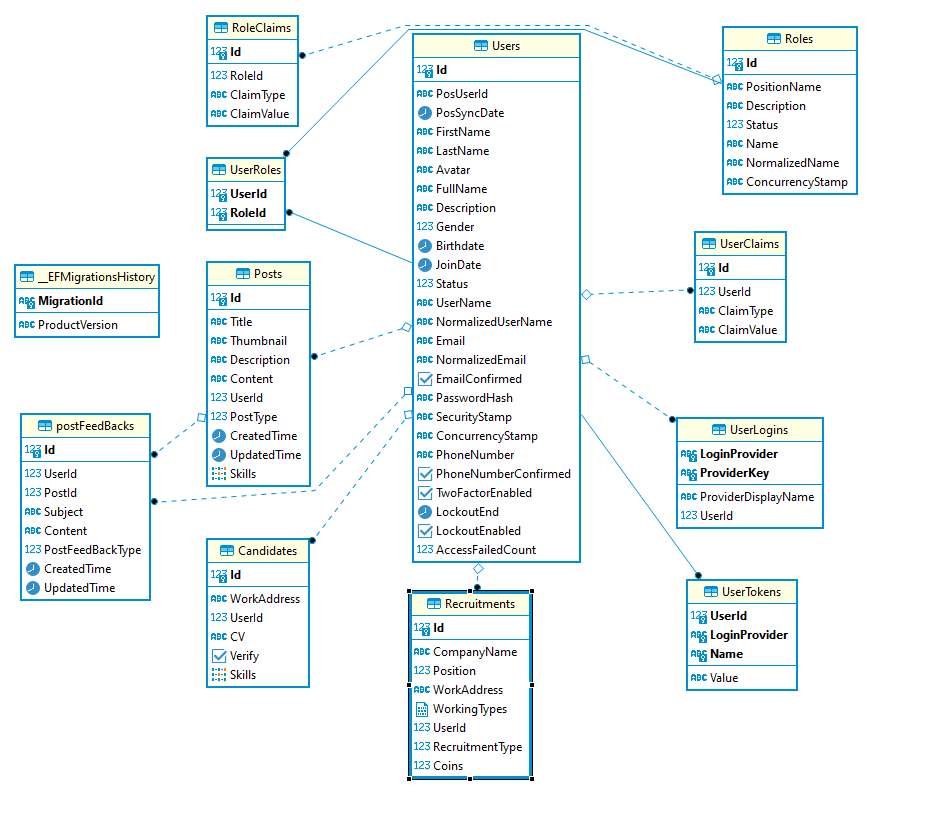
## Collaboration Diagram (Optional)

## State Diagram (Optional)

## Entity Relationship Diagram

**

## Database Design

**

## Algorithms (optional)

*Not available*

## Others (optional)

*Not available*

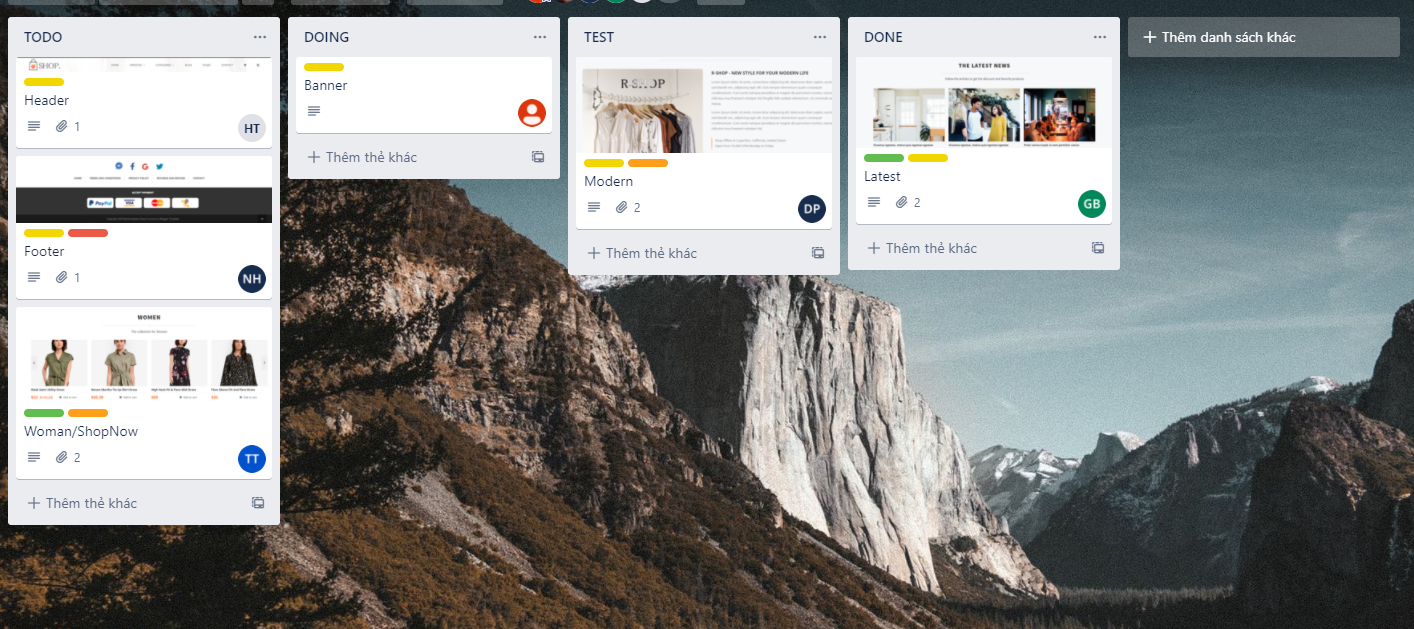
# System Prototype

# Management and Project Planning

## Management Approach

*- Management by github and teamwork with trello.*

## Project Plan

**

## Task Sheet

*- Complete login function.*

*- Complete the candidate's application function.*

*- Complete the job creation function.*

*- Complete the function of receiving employer notifications.*

*- Complete candidate search function.*

## Meeting Minutes (Optional)

* *Not available*

# Checklists

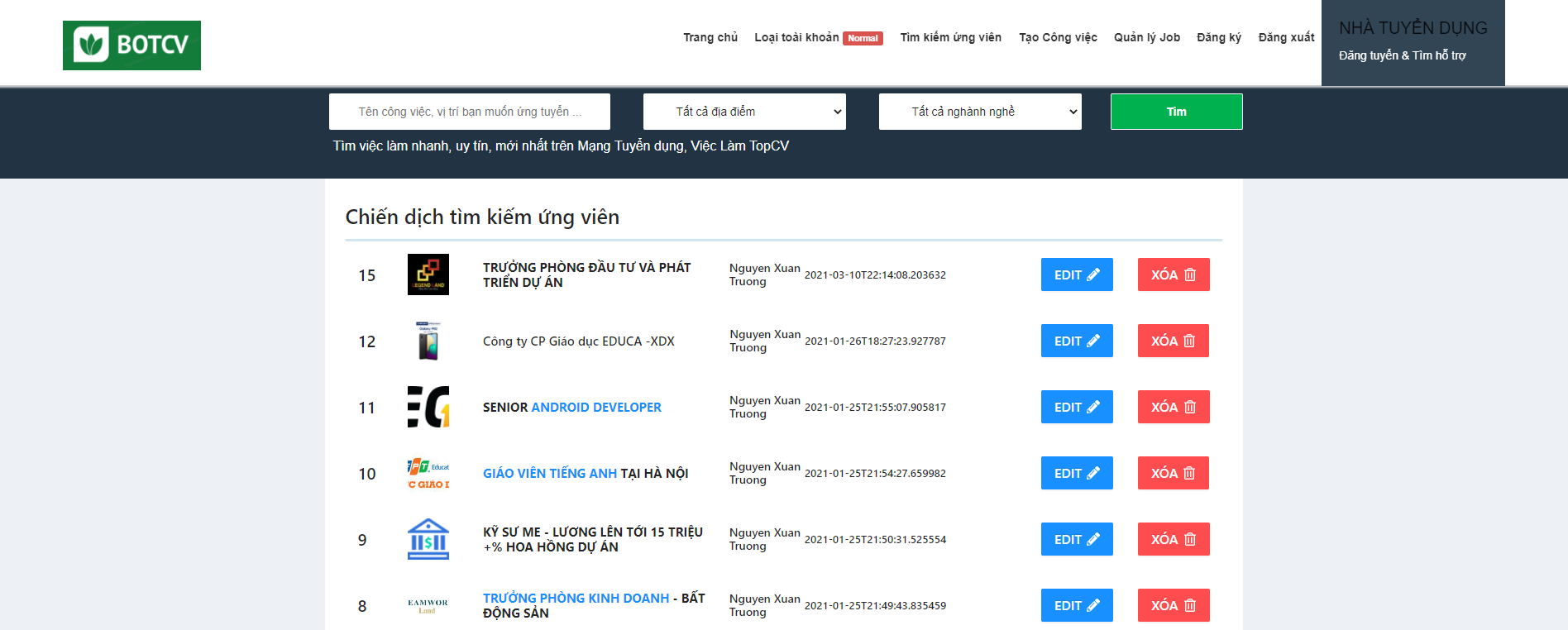
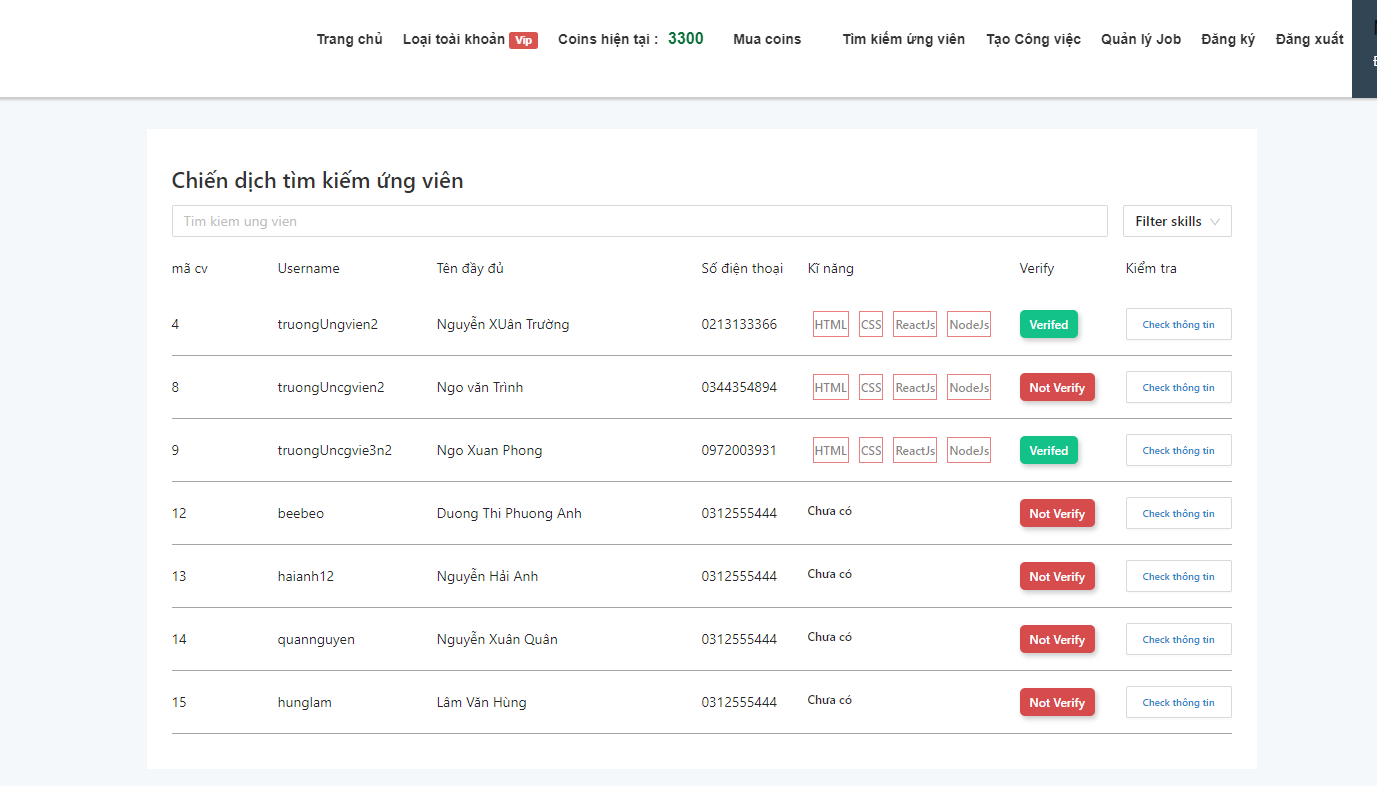
## Check List of Validation

*Not available*

## Submission Checklist

*Not available*

# Screenshots

1. 
2. 
3. 

# Coding Convention

* *Using Javascript Convention . Capitalization, BEM CSS HTML .*

# User Guide

* *Login or registration, then access table manager.*

# Other Concerns<Optional>

*- Lack of a few functions such as creating ads to increase income.*

*- Revenue report function by day, week, quarter.*

*- Post comment function.*

# Appendix

Not available

## Glossary [Optional]

*Not available*

## References [Optional]

Not available

## Others<Optional>

Not available