Specification document

**CVen7**

****

Project manager: Benel Aharon

Version 1.3

Prepared by Benel Aharon, Michael Levitin, Nastaran Motiee,

Re’em Levi, Ortal Bukra, Yarden Aliev Krief, Daniel Sukharev

Group 7

05.01.18

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Users 1

1.2 Goals and objectives 1

1.3 Summary of problems 2

1.4 Risks 3

2. Overall Description 3

2.1 General charecterstics 3

2.14 Glossary……………………………………………………………………………………………………………………………………4

2.2 Users 4

2.3 General description and overview of the system 5

2.4 User Interfaces 11-13

2.5 Proccess 13

2.5.1 Registration 13

2.5.2 Login 13

2.5.3 Status change………………………………………………………………………………………………………………………………………...13

2.5.4 Apply CV………………………………………………………………………………………………..…13

2.5.5 Editing……………………………………………………………………………………………………..13

2.5.6 Search……………………………………………………………………………………………..……….14

2.6 Transactions……………………………………………………………………………………..………....14

2.7 Modules………………………………………………………………………………………………....14-15

2.8 Procedures ………………………………………………………………..……………………………..…15

2.9 Routines……………………………………………,,……….………………………………………….15-16

2.16 Input forms…………………………………………………………………..……………………………..16

2.19 Data security………………………………………………..…………………………………………...…16

2.22 Interfaces and connectivity……………………………………………………………………….16

2.23 Special requirements……………………………………………………………………………...17

2.2.99 Future goals……………………………………………………………………………………..18

**3. Technology…………………………………………………………………………………...18**

3.1 Main hardware……………………………………………………………………………………..18

3.2 Central data storage………………………………………………………………………………...……..18-19

3.14 Third party software…………………………………………………………………………………………19

1.1 Clients

1.1.1 Main user

\*HR department-Client

1.1.2 Additional users

\* Candidate

1.2 Goals and Objectives

\*Creating a database that will be easy to search, update, manage and sort CV applications sent to our client. Access to the database will be available only to authorized personal (HR department).

\*The option for a possible candidate to submit his CV application including his photograph and upload it to our database easily.

\*option for candidates to add social profile link to the CV application

1.2.1 General Objectives

\*Managing, sorting, updating and review the database

\*A simplified and easy to use user interface

\*CV application applying by Candidate via secured connection

1.2.2 Practical goals

Authorized personal\HR department:

\*Secured login into the database

\*Searching the database by ID number, education or years of experience

\*Updating\changing the status of a candidate

\*Managing a secured and encrypted database of our candidates

\*Time based connection to the database after successful login attempt

Candidate:

\*Registering a personal candidate account

\*Applying the CV application (picture and social media link included) and sending it via our system In order to be reviewed by HR

\*Checking the status of the candidates application

\*Picture uploading to Cloudinary service-based on Cloudinary API

1.3.0 Summary of problems in the existing situation

\*Reviewing the database takes a long time and is difficult to manage

\*Many free vacancies because a right candidate wasn’t found

\*No universal and comfortable way of applying the candidates CV

\*Sorting and deleting spam application manually

1.3.1 Problems that the system solves / should solve

\*Easy management of the database

\*Finding the best candidate for a given vacancy

\*GUI system to upload a CV application

\*Fast and Intuitive system to all users

\*CV application are easier to write and upload

\*Candidate-HR transparency being possible by the candidate personal account

\*HR can check the candidate with the social media account link

1.3.2 Problems that the system creates / could create

\*Not enough memory on the clients server to hold all the database

\*Leaking of personal data of the candidate

\*Wasted time reviewing unworthy candidates sending their CV

\* Wasted time manually filtering spam

1.3.3 Problems to be fixed later

\*Not enough memory on the clients server to hold all the database

\*Spam filtration

1.4.1 Risks

\*Problems in a third party software or API that cannot be fixed and determent early

\*Personal candidate data leak

2.1 General characteristics

\*Easy and intuitive work with the database

\* Easy and intuitive CV application system for the candidates

2.1.1 Existing situations

\*Registration system to all users

\*Secured login to HR database-third party base

\*Secured login to candidate personal database-locally managed

\*Different database managing tools given to HR as initially needed

\*Time based connection to HR database to minimize chances of unamortized person reviewing the database

\* No universal and comfortable way of applying the candidates CV

\* Reviewing the database takes a long time and is difficult to manage

\*No secured login option to enter the database

\*Option to add a photo and upload it

\*There is an option foe candidate to add a link of social media

2.1.2 System nature and category

\*As of now only CLI based access or in a python platform

Future UI:

\*A General user interface (GUI)-For company members and candidates alike

\*JSON database

2.1.3 Constraints

\*Possible candidates unexperienced with personal computer

\*Access to the database as well as submitting a CV requires an internet location

2.1.4 Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| HR | Human resources |
| GUI | General user interface |
| CV | Curriculum vitae |
| User | HR department or candidate |
| Candidate | Possible new employee - CV applicator |
| API | [Application programming interface](https://en.wikipedia.org/wiki/Application_programming_interface) |
| Cloudinary | Cloud based picture database |
| DFD | Data flow diagram |
| EFD | Event flow diagram |
| Status | The candidate current status (i.e postponed, accepted…) |

2.2.1 Users

\*The clients HR department-in order to access and manage the database

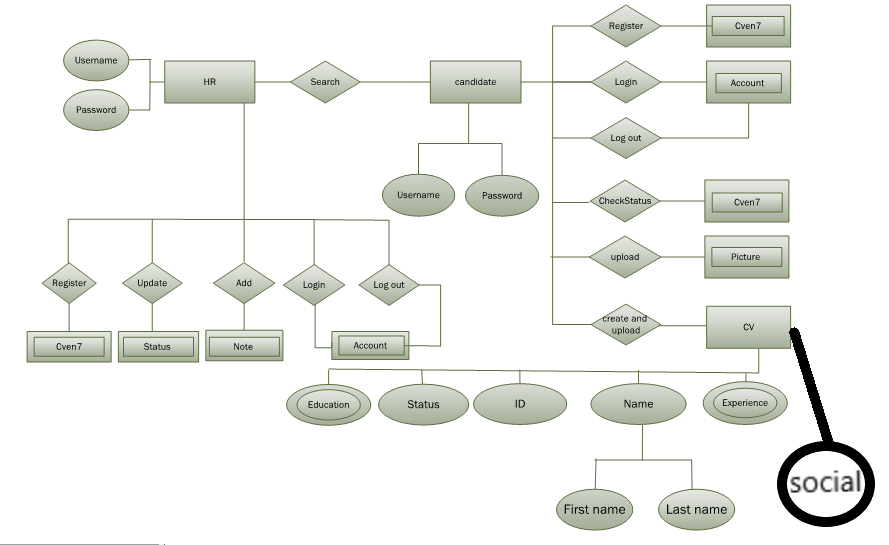
\*Candidates-in order the apply a CV application and status checking

2.2.2 Tangent systems

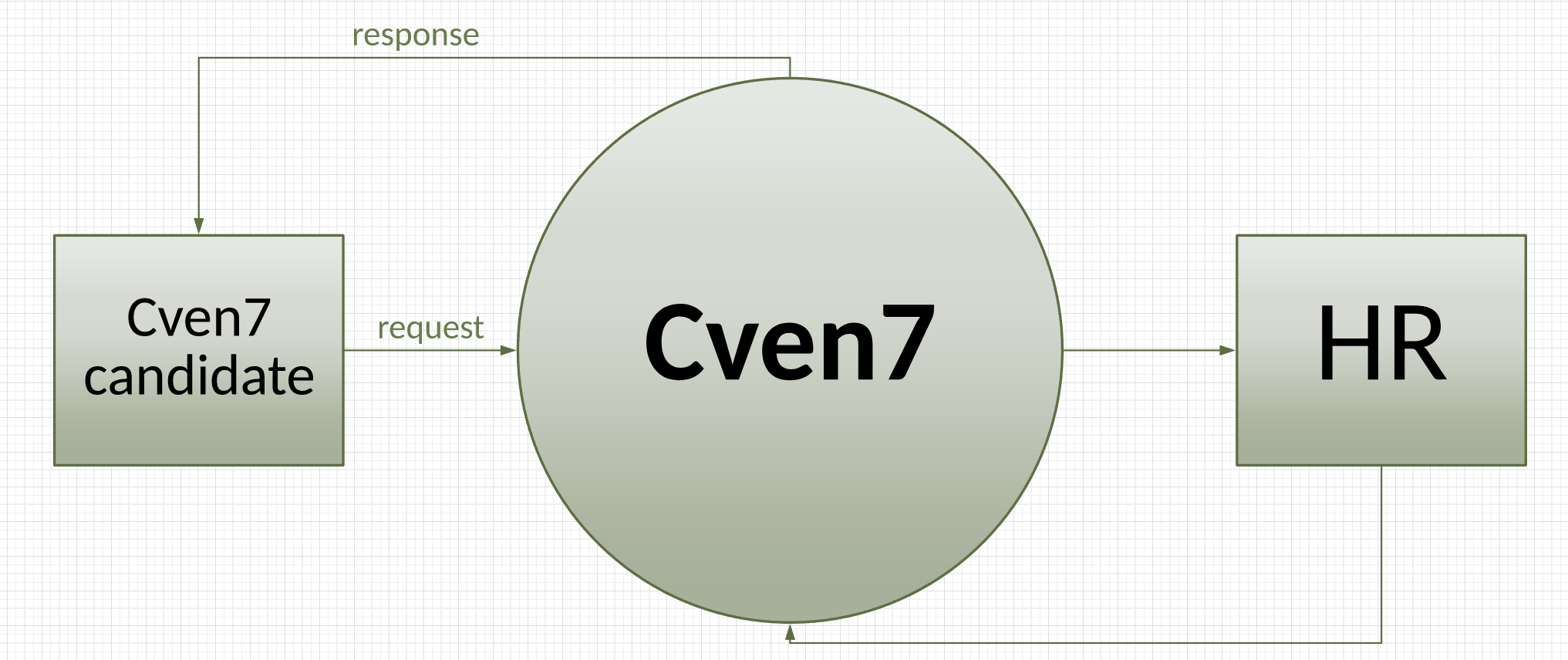
Not relevant at this stage

2.3.0 General description and overview of the system

ERD:

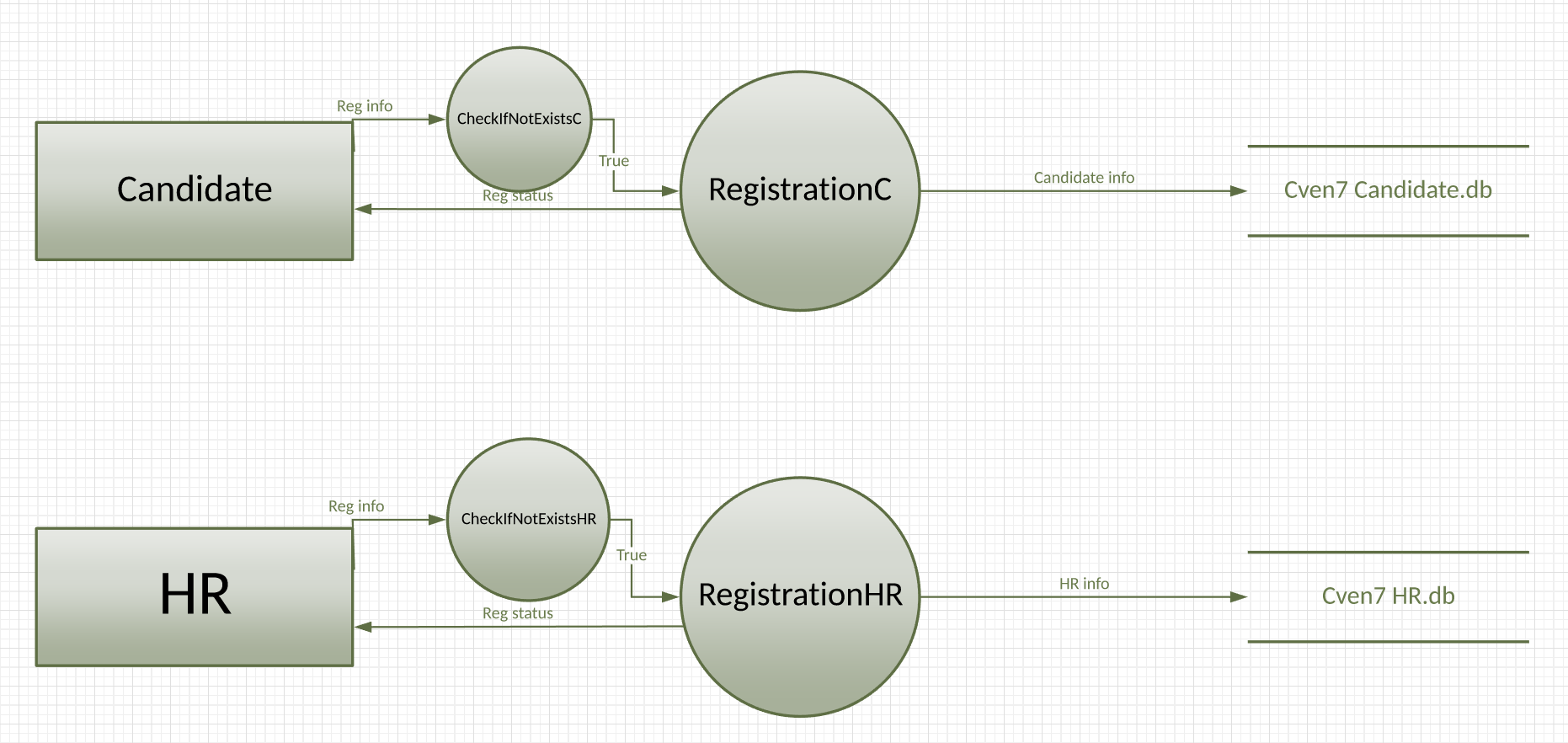


2.3.1 Initial system interaction:



Choosing login\registration and HR department\Candidate user type

2.3.2 Registration:

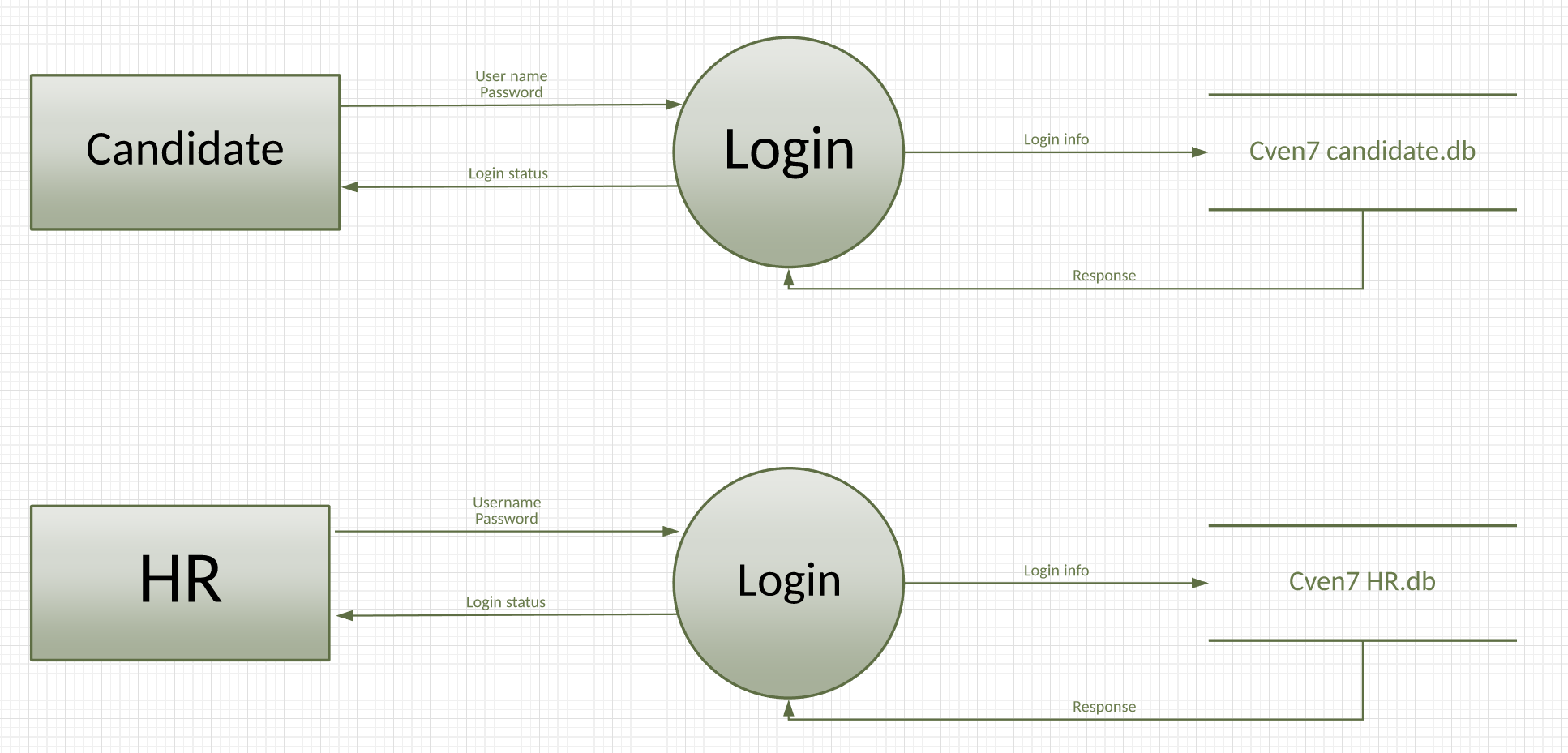


Registration to HR\Candidate account databases

Creating new account if yet un existing

Account database maintained via SQlite3 API

2.3.3 Login:

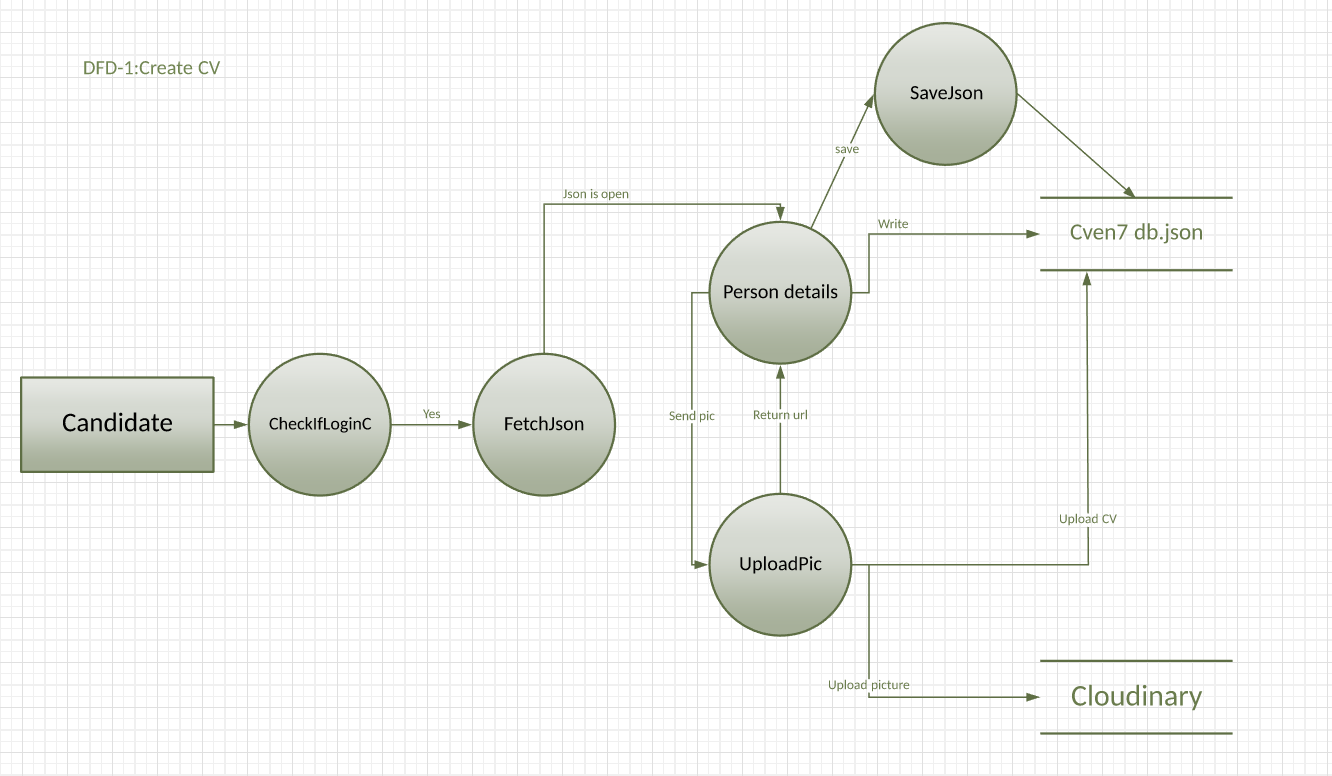


Login to HR\Candidate database

Connection enabled if User name and Password are correct

Secured SQlite3 login enabled

2.3.4 CV applying\creating



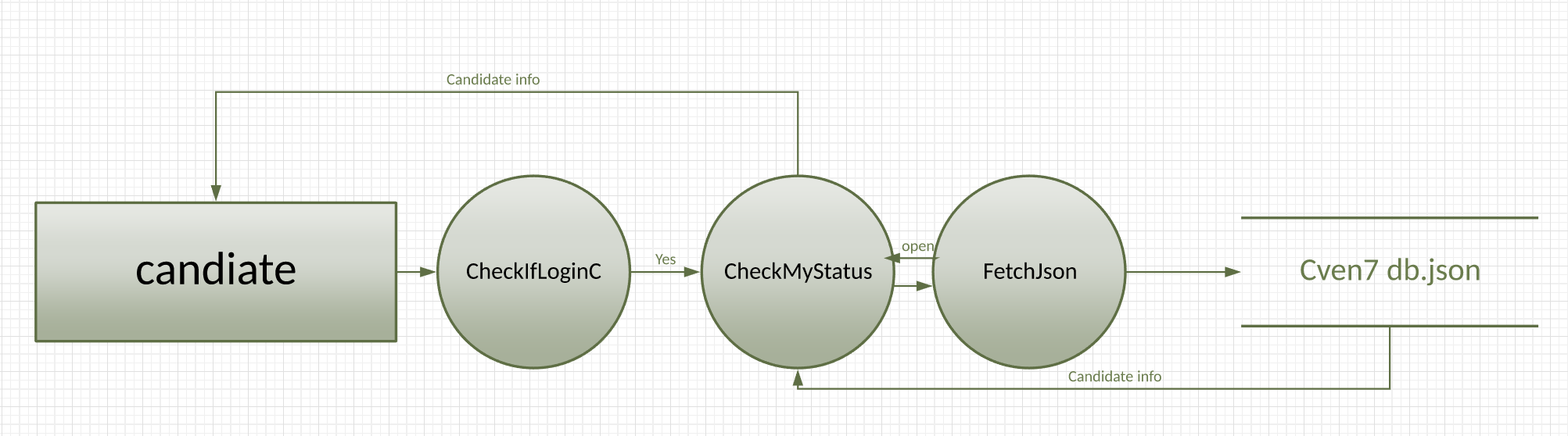
\*If login is enabled

Applying CV as JSON to database

Picture uploaded to Cloudinary database

Default candidate status applied

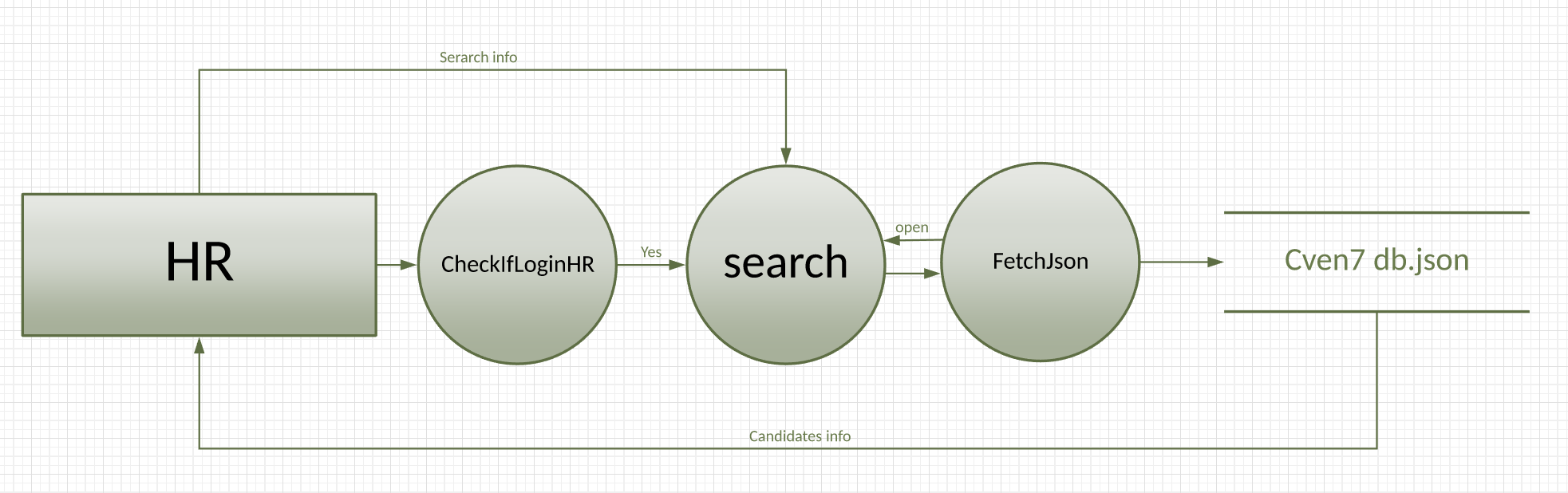
2.3.5 Status checking



\*If login is enabled

Returning current candidate status

2.3.6 Search

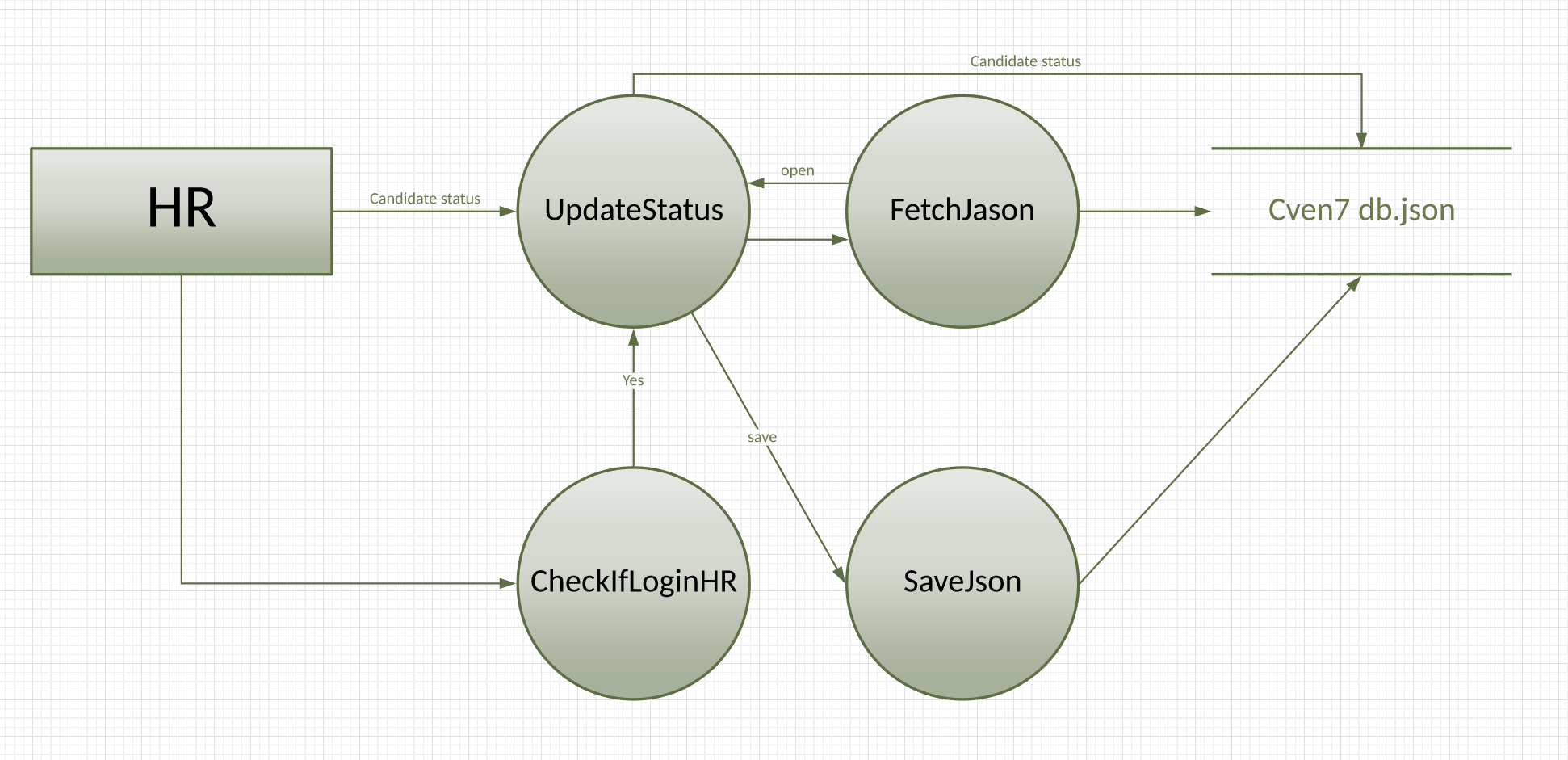


\*If HR login is enabled

Searching the candidate database via a given parameter(versatility depends on current software version)

Returning JSON

2.3.7 Update status



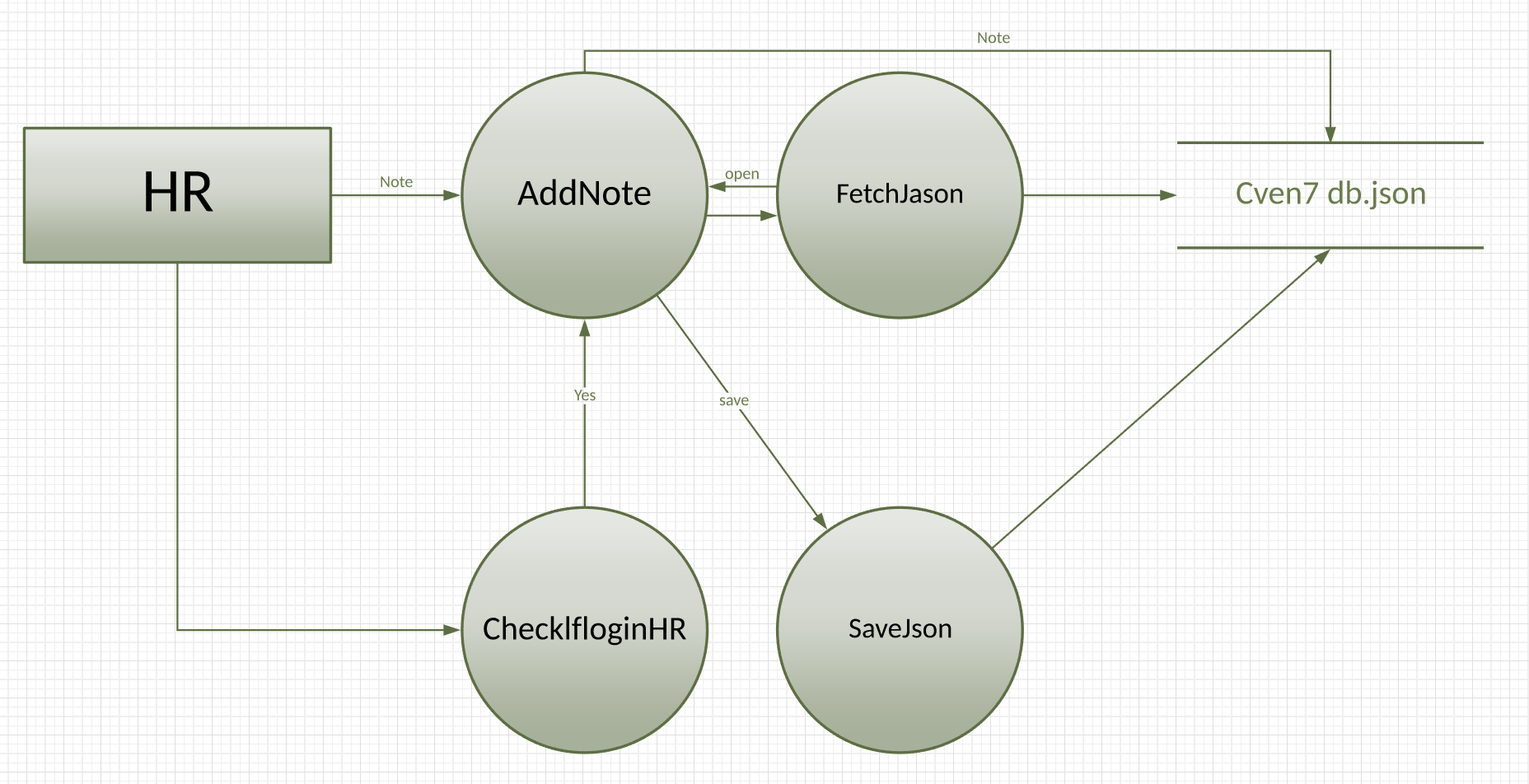
If HR login enabled

Changing the current status of the candidate

Current status options:

Postponed, Accepted, Declined, Reviewing

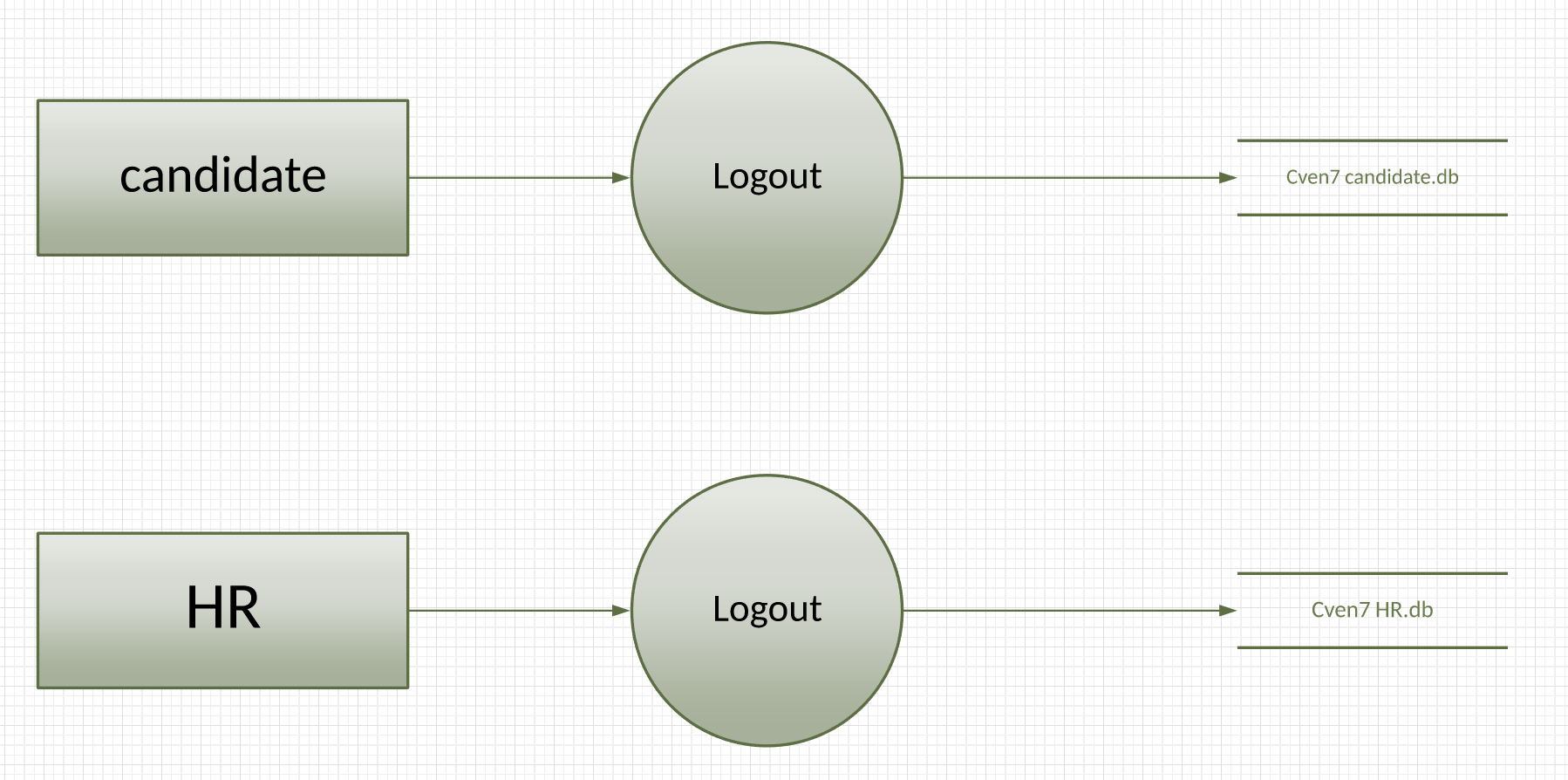
2.3.8 Add notes



If HR login enabled

Adding notes to a candidate in order to fastly notify different HR or Management personal of changes related to a candidate

2.3.9 Logout



System Logout of HR\Candidate account

Database connection over

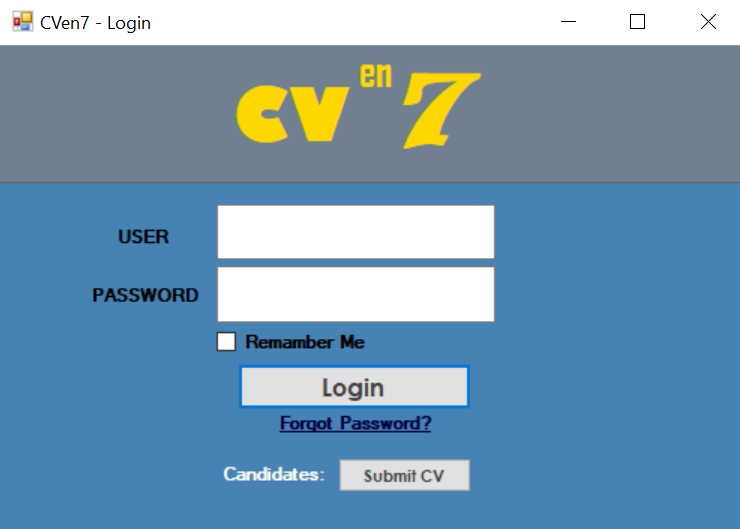
If user manually didn’t logout-automatic system logout enabled by the database (time depends on settings)

2.4 User interface

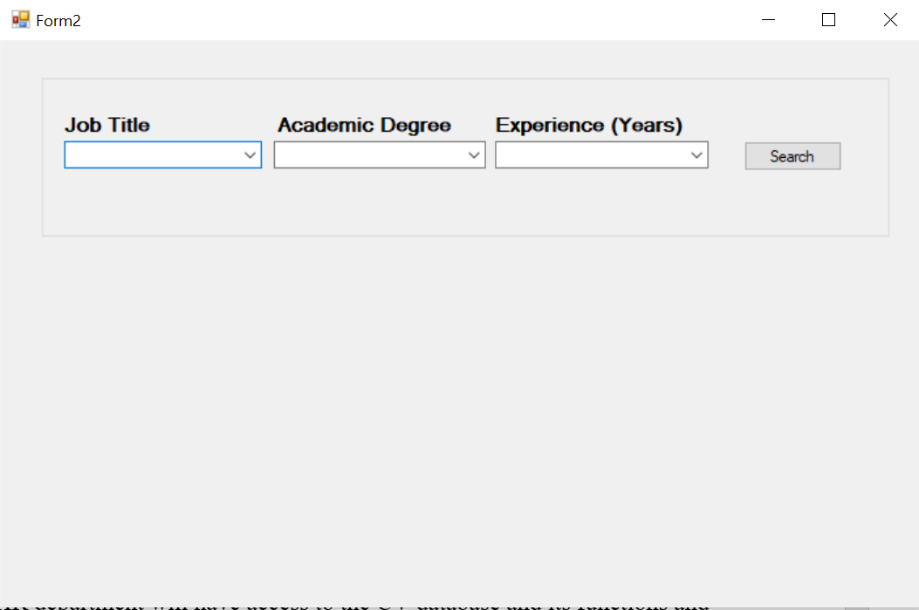
2.4.1 GUI **(previous version only)**

GUI based interface. Operated by mouse and keyboard.

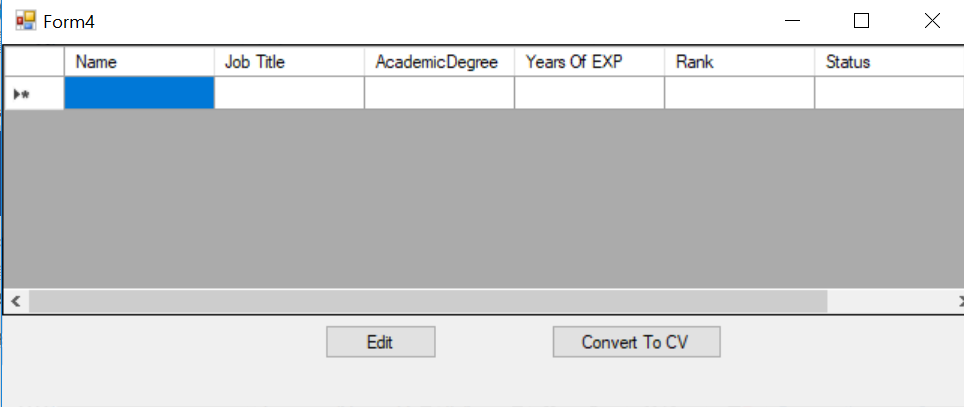
\*General user - Initial logging screen



\*HR - Database search windows



\*HR - Data table

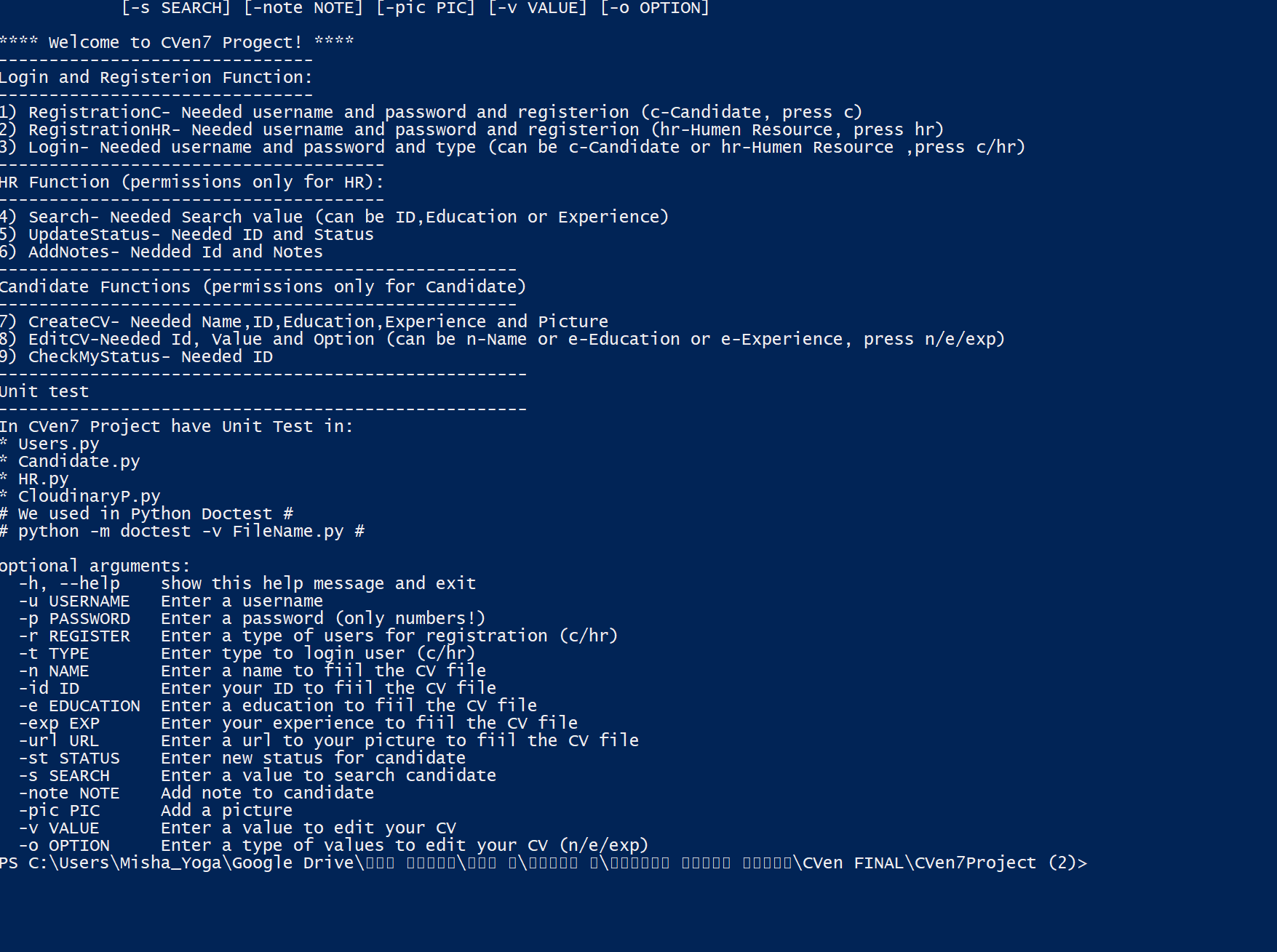


\*Candidate - CV application window

2.4.2 CLI (Current prototype UI)

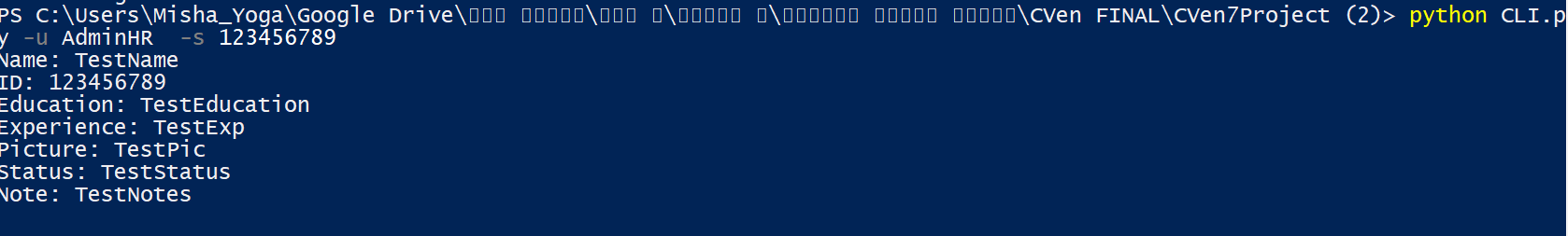
A CLI based user interface is used to interact with the system.

Using Windows powershell

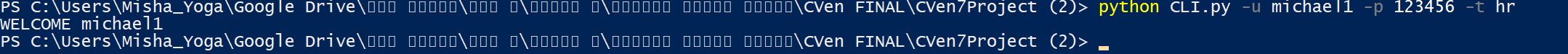


Examples

\*Searching the database:



\*Logging in



2.5 Processes

2.5.1 Registration

Registering a new account to HR or personal candidate database.

2.5.2 Logging

As a member of the HR department or candidate you are required to login into the system with your user name and password.

2.5.3 Candidate status change

Changing the status of a given candidate.

2.5.4 Applying CV-by a candidate

UI for a candidate to digitally apply the CV file

Uploading personal picture

2.5.5 Editing the CV document

Editing the CV file in order to make changes, mark and make notes

2.5.6 Search

Searching via the database using different parameters by need

2.6 Transactions

2.6.1 Applying CV application

As a candidate you send your own CV via the user interface (login required) to the HR database.

2.6.2 Changing the candidate status

\* Default status is “Under review”

As a HR department member you can change the status of the candidate and notify him.

2.6.3 Database transactions

When registering a new account to HR\Candidate department a new account is created and stored in the Database

2.6.4 Cloudinary

When a new CV is being created and uploaded including the candidates picture, The picture is then uploaded to the Cloudinary cloud service and stored there until a final decision regarding the candidate is made.

2.7 Modules

2.7.1 HR database module

\*New account registration

\*Ability to search CV by keywords , experience, education and so on

\*Change the candidate status

\*Sort and manage the database

2.7.2 Candidate database module

\*New account registration

\*Ability to apply a CV application

\*Personal status check

2.7.3 CV Database module

JSON based database containing CV and personal data of candidates.

2.8 Procedures

2.8.1 Testing

A python based unites is applied to the software.

2.9 Routines

2.9.1.2 Creating account

When creating an account for HR or Candidate the same algorithm is called to create and store the different account types in different databases.

2.9.1.2 Login

When logging to the system as an HR or Candidate the same algorithm is used to check if an account is existing and logging.

2.9.3 Third party procedures

\*Cloudinary – security procedures

\*Firebase(used in previous versions only)

2.16 Input forms

2.16.1 CV documents

The CV documents are created manually by a candidate user

Field include personal data, type of education, years of experience.

The CV is then saved as a JSON file and uploaded as an input to our database servers.

2.19 Data security

Database security is managed using SQLite3 to allow only authorized personal logged in to the system. Then is connection is made from the locally stored database and the account to search and review the database.

2.22 Interfaces and connectivity

2.22.1 Cloudinary

Cloudinary is the market leader in providing a comprehensive cloud-based image management solution.

Cloudinary is used to upload and store the candidates personal photograph sent when applying a new CV.

2.22.2 SQLite

SQLite 3.0 is used to view and manage the database locally stored in our system.

2.22.3 Powershell

Windows Powershell is used as a user interface (for the current version)

To manipulate the database and accounts.

2.22.4 Python

Current version of software is based on Python 3.4.

2.23 Special requirements

Future additional application needed:

\*More advanced and secured logging application.

\*A fully working graphic user interface compatible with windows based operation system.

\*A cloud based secured database in change of the current locally stored database.

\*More flexibility and tools as mentioned, for example a candidate direct message box to enable real time correspondence between a candidate and HR member.

\*Less run time when uploading data to a cloud based database.

2.2.99 Future goals

\*The option to rate different candidates by an algorithm that takes into consideration the education of the candidate as well years of experience and previous job titles.

\*Video call option from HR to candidate:

\*Only HR can make the initial call

\*Candidate option to receive\reject call

\*Candidate message box:

\*Receiving status updates

\*Receiving direct messages from HR department

\*Candidate updating of CV and personal data

\*Advertisement of job opportunities to possible candidates

\*Automatic reminder to HR personal of yet un-reviewed CV application

\*Filtering and maintain database spam(sent by spam robots) in order to save time from manually sorting unwanted spam application

3.0 General architecture

The current system architecture is an extension of the previous system.

Currently based on a CLI platform to interact with the database and the system.

3.1 Main hardware

Personal computer with a Microsoft Windows operation system.

Fast Ethernet connection to the internet.

Hard drive to contain the locally stored database.

3.2 Central data storage

\*HR database- including CV application and personal data sent by candidates is stored locally as database in JSON format

\*Candidate personal photographs- stored on a cloud service using Cloudinary cloud service

\*Accounts- HR\Candidate accounts are created and stored in a database using SQLite3 security protocol

3.14 Third party software

3.14.1 Service software

\*Google- used as a search engine and multi tool

\*Slack- used as a the platform for communication between members of our developing team

\*Github- used to store different versions and changes made to our software and stored in git

3.14.2 Application software

\*Microsoft Windows 7- the operating system being used

\*SQLite3- to encrypt and secure account connection to the database

\*Windows PowerShell- the CLI platform being currently used to interact with the software