# Seminar 3

## Our progress and achievements


**pullOut team**

Renaldas Narbutas
Oleksii Morozov
Joris Namajūnas
Severyn Stankevič

**Supervisor**

Dr. Linas Bukauskas

Vilnius university

# Table of contents

# Terminology

- OCR - Optical character recognition
- NER - Named entity recognition
- CardDAV - Card Distributed Authoring and Versioning

# CardDAV server-client system

- Standard for communicating contact information between server and client.
- Stores information in vCard 3.0 format with .vtf extension

# vCard example

BEGIN:VCARD

VERSION:3.0

**FN**;CHARSET=UTF-8:Renaldas Narbutas

**N**;CHARSET=UTF-8:Narbutas;Renaldas;;;

**EMAIL**;CHARSET=UTF-8;type=WORK,INTERNET:renaldas.narbutas@mif.stud.vu.lt

**TEL**;TYPE=CELL:+37069918736

**ADR**;CHARSET=UTF-8;TYPE=WORK:;;Didlaukio g. 59;Vilnius;;;

**TITLE**;CHARSET=UTF-8:Student

**ORG**;CHARSET=UTF-8:Vilnius University

**URL**;CHARSET=UTF-8:https://www.google.com/

REV:2023-10-09T18:22:25.779Z

END:VCARD

# Diagram

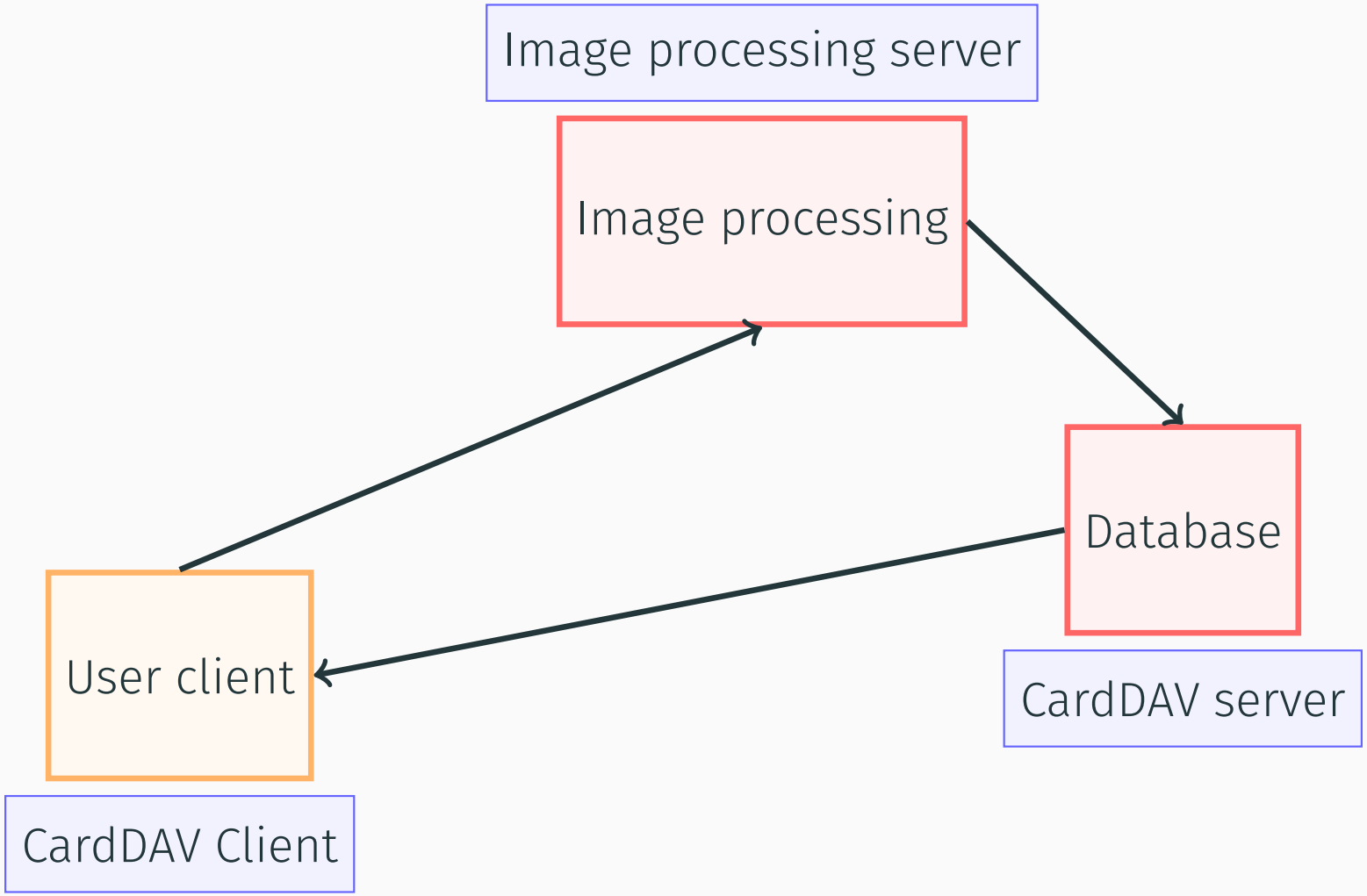


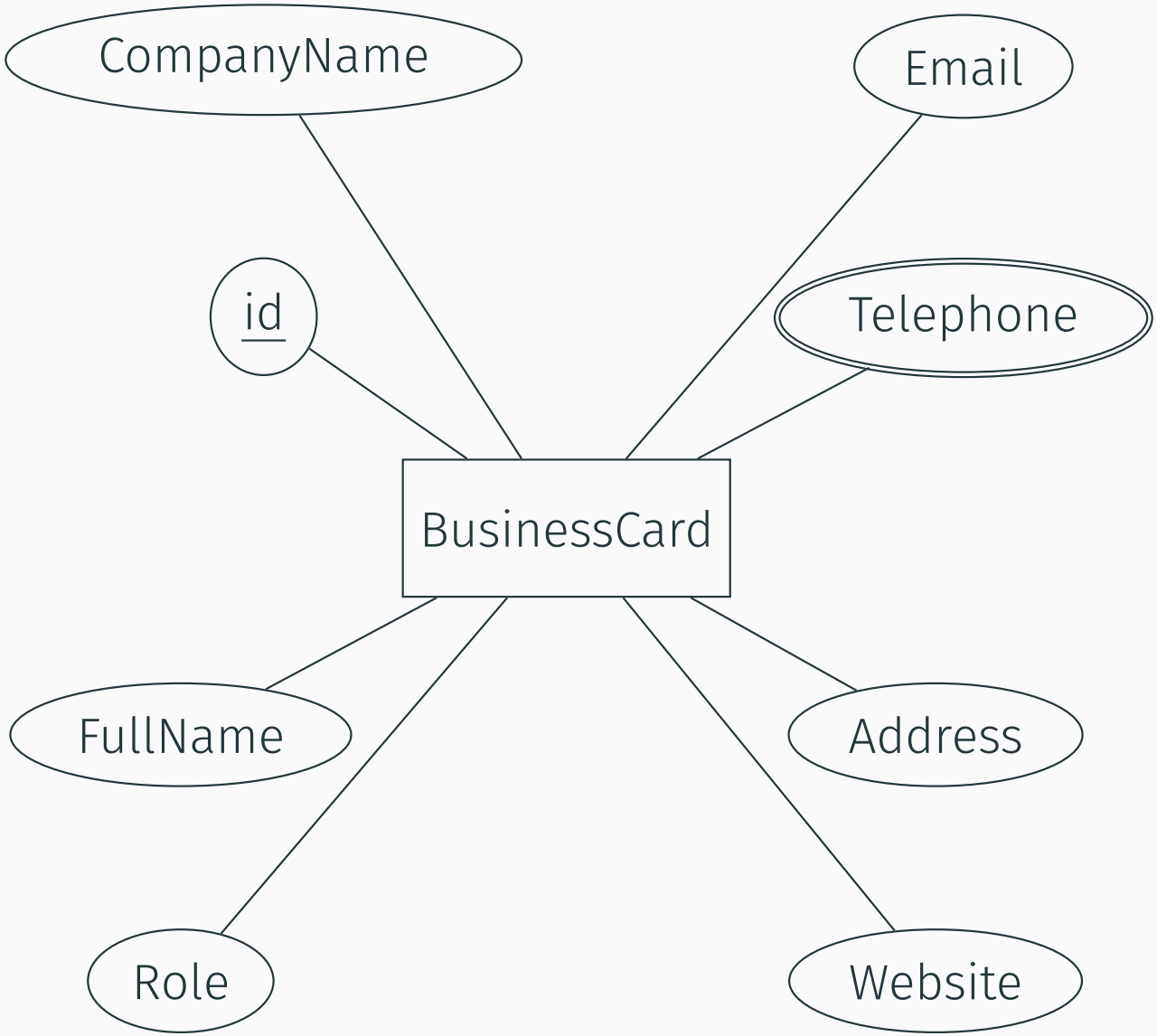

**Figure 1:** Server-Client system

# Card duplication

To avoid card duplication in database we will use hashes

| Explanation | Pseudo-code expression |
|---|---|
| get text from image and return string *text* | *text* ← *processText*() |
| get entity list from text | *entities* ← *getEntities*(*text*) |
| calculate *hash* from entities | *hash* ← *makeHash*(*entities*) |
| check if *hash* exists in *hashtable* | **if** *hash* ∉ *hashtable* **then**<br>  *hashtable*[*hash*] ← *true*<br>  *data* ← *entToVCard*(*entities*)<br>  *sendToDatabase*(*data*)<br>**else**<br>  *return* |

# E-R diagram


CompanyName
Email
id
Telephone
BusinessCard
FullName
Address
Role
Website

# Functional requirements

1. Accepts an images as input;
2. Support for common image formats: JPEG, PNG, and WEBP;
3. Scan image and recognize data fields: full name, phone number, company name, address, email, job title, URL;
4. Users able to view, edit, create groups and delete digital contacts;

# Non-functional requirements

1. OCR accuracy at least 90% on Lithuanian and English business cards with good quality picture;
2. Clear and well defined error messages;
3. The image processing server should be able to handle concurrent users input;

# Image processing

Image filtering steps:

1. Resize to width of 500px
2. Business card edge detection
3. Dilate image
4. Find 4 corners of the card
5. Fit cropping coordinates to original image
6. Scan text using Tesseract

# Original picture


**Figure 2:** Original photo

# Edge detection


**Figure 3:** Edge detection

# Dilation of edges

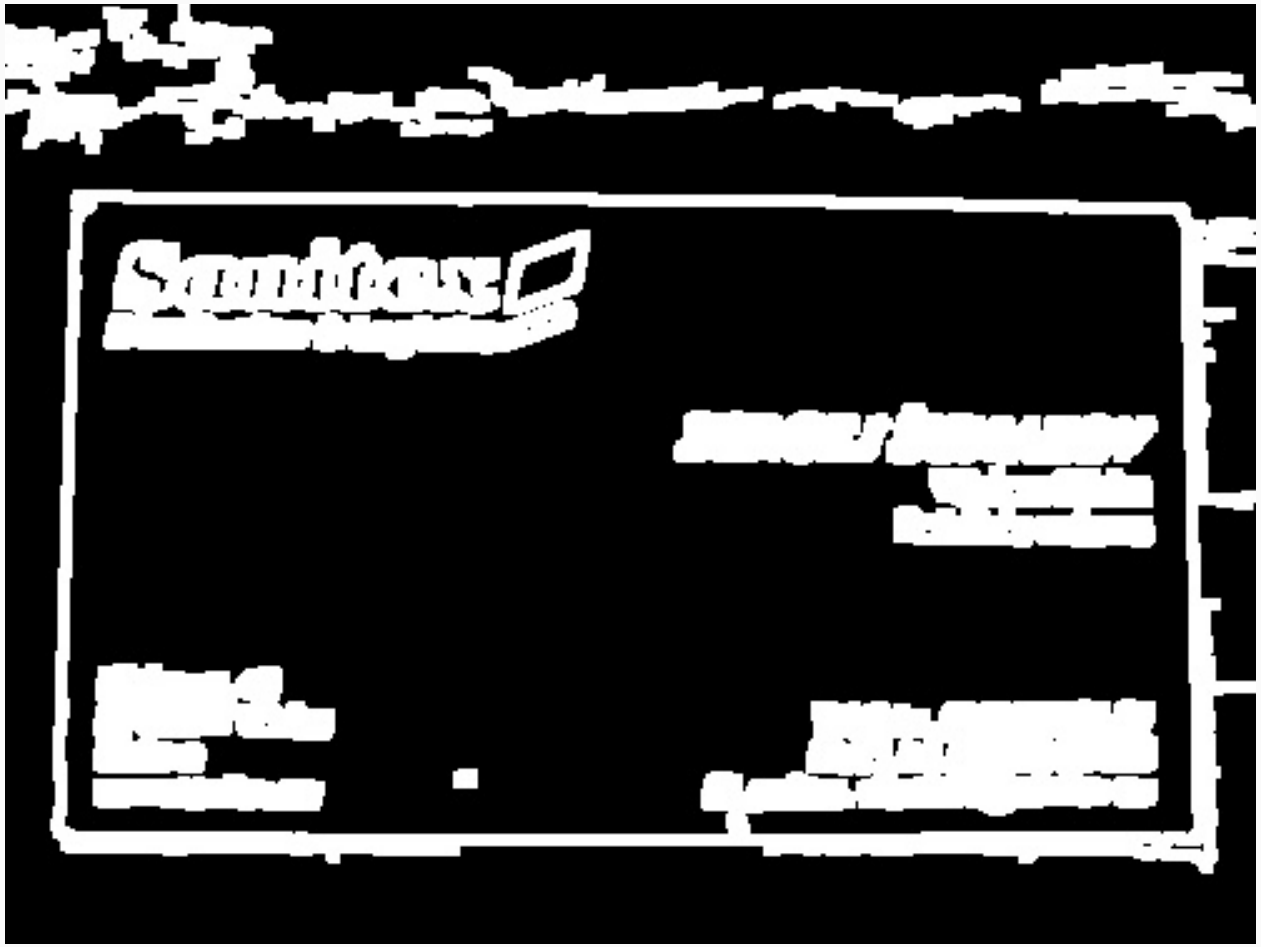


**Figure 4:** Dilated edges

# Contours


**Figure 5:** Contours drawn on original picture

# Scanned text

Sanitex 9

Distribution 8 Logistics /

SERGEJ ŠUMAKOV
Vilniaus filialas
Padalinio vadovas

Kirtimų g. 49

LT-02244 Vilnius | Tel./faks. (8-5) 260 21 87

Lietuva Mobil. tel. (8-656) 55 26
* — El. paštas: vil.retail1(dsanitex.eu |

www.sanitex.eu

# Entity recognition approaches

1. Preparing a list of possible values and comparing it to the given business card - **NONE**
2. Making regular expressions for patterns of text - **Email, Website, Telephone**
3. Training natural language processing model to recognize patterns and identify entities - **Company name, Address, Job title, Full name**

# Email

Regular expression used:

```
[\w.+-]+@[\w-]+\.[\w.-]+
```

- `[\w.+-]` - Match any word or character '.+-' one or more times
- `@` - Match '@' character
- `[\w.-]+` - Match word one or more times
- `\.[\w.-]+` - Match '.' then word or character '.-' one or more times

# Website

Regular expression used:

```
((http|https):\/\/)?
([\w_-]+(?:(?:\.[\w_-]+)+))
```

- `((http|https):\/\/)?` - Can begin with ”http://” or ”https://”
- `([\w_-]+(?:(?:\.[\w_-]+)+))` - Match word one or more times if it has dot and word one or more times later in the structure

# Telephone examples

```
1-718-444-1122
718-444-1122
(718)-444-1122
17184441122
7184441122
718.444.1122
1718.444.1122
1-123-456-7890
1 123-456-7890
1 (123) 456-7890
1 123 456 7890
+91 (123) 456-7890
```

```
+86 800 555 1234
1-800-555-1234
1 (800) 555-1234
(800)555-1234
(800) 555-1234
(800)5551234
800-555-1234
800.555.1234
18001234567
1 800 123 4567
1-800-123-4567
+18001234567
```

# Telephone

```
^(\+\d{1,2}\s?)?1?\-?\.?\s?
\(?\d{3}\)?[\s.-]?\d{3}[\s.-]?\d{4}$
```

```
(8-656) 55 265
+370 698 58 099
+370 69918736
```

# Spacy modeling system

It is a statistical language modeling system

> *Language modeling is the task of assigning a probability to sentences in a language. [...] Besides assigning a probability to each sequence of words, the language models also assigns a probability for the likelihood of a given word (or a sequence of words) to follow a sequence of words[1]*

# Entities for statistical model

- Company name
- Job title
- Full name of person
- Address

# Runtime

For one image to be processed On Debian 12 MIF VM with parameters:

- Memory: 1GB
- VCPU: 1
- CPU: 0.2

Image scanning took: **6.9sec**
Entity recognition took: **1.8sec**

In total: **8.7sec**

# Individual work

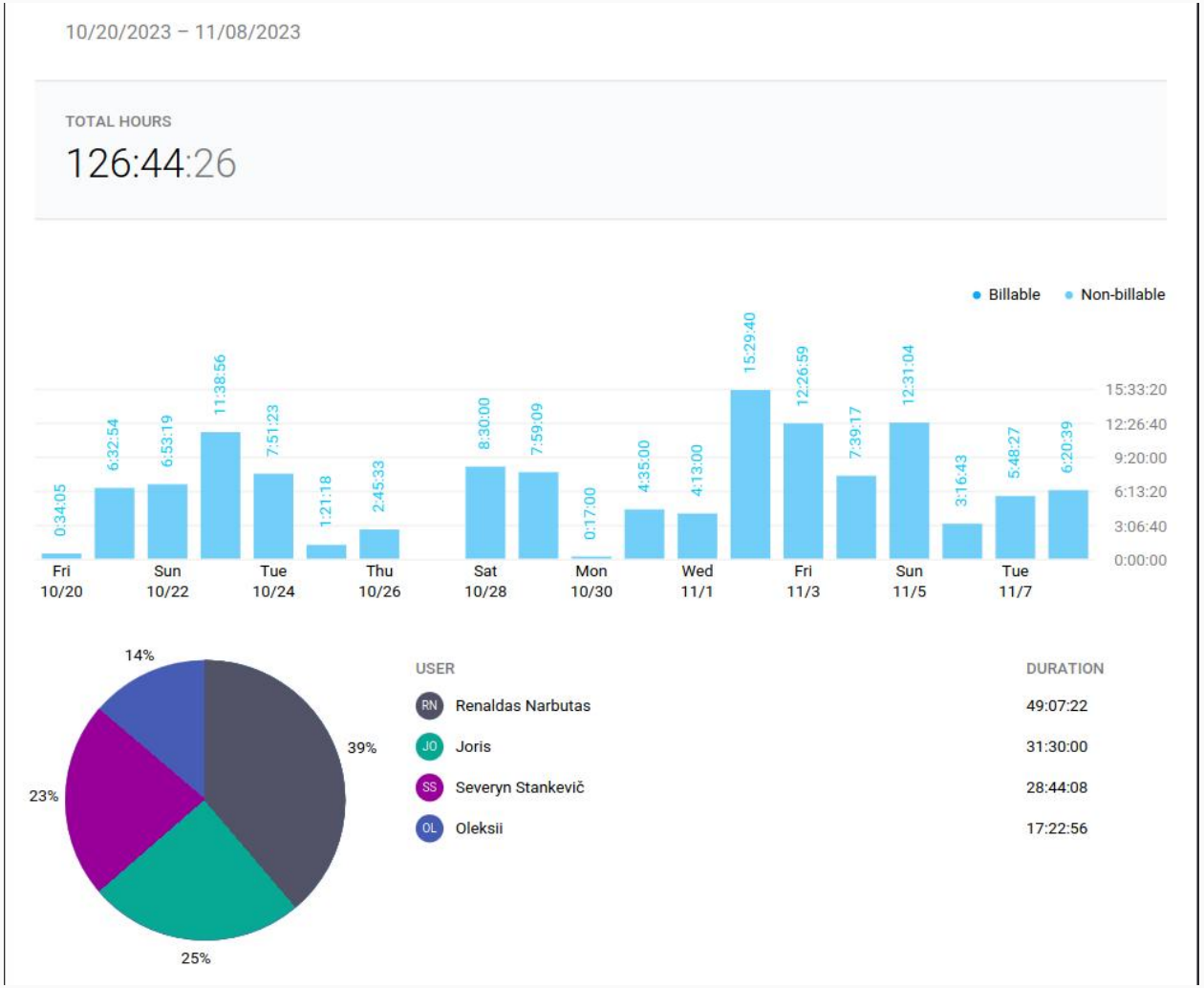



**Figure 6:** Individual time for work

# Team meetings


**Figure 7:** Team spent on meetings

# References


Yoav Goldberg.
Neural Network Methods in Natural Language Processing.
Morgan Claypool, 2017.