Die approbierte Originalversion dieser Diplom-/Masterarbeit ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Visual Exploration and Comparison of Multiple Resumes

Focus on Time and Space

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medieninformatik und Visual Computing

eingereicht von

Velitchko Filipov
Matrikelnummer 0726328

an der Fakultät für Informatik
der Technischen Universität Wien
Betreuung: Univ.Prof.in Dr.in Silvia Miksch

Wien, 1. Mai 2017

Velitchko Filipov	Silvia Miksch

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Visual Exploration and Comparison of Multiple Resumes

Focus on Time and Space

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Media Informatics and Visual Computing

by

Velitchko Filipov
Registration Number 0726328

to the Faculty of Informatics

at the TU Wien

Advisor: Univ.Prof.in Dr.in Silvia Miksch

Vienna, 1st May, 2017

Velitchko Filipov

Silvia Miksch

Erklärung zur Verfassung der Arbeit

Velitchko Filipov
Vienna

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 1. Mai 2017

Velitchko Filipov

Danksagung

Ich möchte meiner Betreuerin Prof. Silvia Miksch danken. Ich danke Ihnen für die Führung und die Freiheit, meine Ideen auszudrücken und auszuführen. Ich möchte auch meinen herzlichen Dank aussprechen und die enorme Unterstützung von meiner Familie und meinen Freunden erkennen, vor allem meinen Eltern, die geduldig waren und mich während meiner Ausbildung motiviert haben.

Vielen Dank.

Acknowledgements

I would like to acknowledge my supervisor Prof. Silvia Miksch. I thank you for the guidance and freedom to express and execute my ideas. I would also like to extend my sincere gratitude and recognize the tremendous support from my family and friends, especially my parents who have been patient, kept me motivated, and guided me throughout my education.

Thank you.

Kurzfassung

Daten, die in Lebensläufen vorhanden sind, haben sowohl zeitliche als auch räumliche Dimensionen, wie etwa die Berufserfahrung oder Bildung einer Person. Diese Information ist umfassend - es ist sehr klar, wann, wo und wie lange eine Person gearbeitet oder studiert hat. Es wird immer schwieriger, einen klaren Überblick zu erhalten, wenn man mehrere Lebensläufe, insbesondere ihre chronologischen Informationen, vergleichen will. Das Umschalten zwischen den verschiedenen Lebensläufen um jedes einzelne Ereignis zu vergleichen, ist sehr ineffizient. Die häufigste und intuitivste Art, Ereignisse aus mehreren Lebensläufen zu vergleichen, wäre, sie in einem Seite-an-Seite Modus zu sehen. Viele Ähnlichkeiten können aus diesem Modus schnell erkannt werden, aber in den Fällen, wo mehr als zwei Lebensläufe gleichzeitig verglichen werden müssen, verliert man den Überblick. Die Menge an Informationen steigt erheblich und der Vergleich von Ereignissen wird zu einer schwierigen Aufgabe. Wir schlagen das Design und die Implementierung einer Webapplikation vor - CV3, die die Möglichkeit anbietet, mehrere Chronologien gleichzeitig zu vergleichen und die Ergebnisse klar visualisieren kann. Dabei ist unsere Ziel einen sauberen Überblick der Ähnlichkeiten bzw. Unterschiede der verschiedenen Lebensläufen zu erhalten. Unser Ansatz unterstützt Benutzer_innen bei der Filterung von chronologischen Ereignissen über alle Lebensläufe hinweg, erkennt und extrahiert relevante Informationen aus Lebensläufen und visualisiert ihre Ähnlichkeiten effizient in einer einzigen klaren Übersicht.

Abstract

Information present in resumes has temporal as well as spatial dimensions, such as a person's work experience or education. This information by itself is comprehensible - it is very clear when, where, and for how long a person has worked or studied. It becomes increasingly difficult to maintain a clear overview when comparing multiple resumes, specifically their chronological information. Navigating back and forth between resumes, switching between views and attempting to compare every single event is highly inefficient. The most common and intuitive way to compare events from multiple resumes, would be to view them in a side-by-side fashion. Many similarities can be quickly recognized by doing this, but in cases where more than two resumes need to be compared simultaneously, the overview becomes cluttered, the amount of information increases substantially and comparing events becomes a difficult task. We propose the design and implementation of a web application - CV3 - that is capable of comparing multiple chronologies and visualizing the output in a clear manner, whilst maintaining a clean overview. Our approach supports users in filtering timeline events across all resumes, recognizing and extracting relevant information from resumes and visualizing their similarities efficiently in a single overview.

Contents

CHAPTER 1

Introduction

Card et al., [CMS99, p. 7] define Information Visualization (IV) as

> "...the use of computer supported interactive visual representations of data or information to amplify human cognition."

Thomas and Cook [TC06, p. 10] describe Visual Analytics (VA) as

> "... the science of analytical reasoning facilitated by an interactive visual interface."

One of the most important principles in IV for effectively representing data is the information seeking mantra of Shneiderman [Shn96] overview first, zoom and filter, then details on demand. This principle is crucial in the context of effectively visualizing and comparing large amounts of data such as that represented in multiple resumes.

1.1 Problem Description

The search for employment in the last decade has become more electronic due to the increasing amount of people having access to the internet. As a result of this, a magnitude of electronic Electronic Curriculum Vitae (E-CV) builders have been developed and provide users with the option to showcase their personal information, skill sets and work/education histories in a clean and visually pleasing manner. Such services are offered by e.g., Represent.IO (www.represent.io), VisualCV (www.visualcv.com), and DoYouBuzz (www.doyoubuzz.com) to name a few.

The new E-CV trend has also lead to more employers accepting resumes online via e-mail or other mediums. People working as hiring managers or in the Human Resources (HR)

department of a company typically go through hundreds of resumes on a daily basis to find the best suited candidate for a given position. This means that they need to read through hundreds of Curriculum Vitae (CV)'s and compare their work and education experience, skill sets, languages, and other information quickly as well as maintain an overview of the strengths and weaknesses of each candidate. According to a recent study, using eye tracking software conducted by Nanji [Nan], in the short time that recruiters spend reading resumes they will focus mostly on current and previous companies, position start and end dates as well as education. Hiring managers can quickly navigate through and extract relevant information from a single resume, due to the fact that they are accustomed to the structure and format of a standard CV. Issues arise when they need to concurrently compare multiple resumes. The most common and intuitive way to compare this information is to have the resumes side-by-side and go through each of the sections of the CV's. Many similarities, differences, and relevant information can be quickly recognized from a side-by-side view of the resumes. This requires the reader to switch attention back and forth between the resumes and is not at all an efficient way of comparing such information. If there are more than two resumes to compare simultaneously, the overview is quickly lost and the amount of data to compare increases substantially. The use of IV and VA techniques can aid the task of comparing and visualizing the multifaceted data that is present in resumes.

In an article by Lauren Weber [Web], she states that the current state of affairs for analyzing, filtering, and finding resumes of candidates best suited for a given position is to use an Applicant Tracking System (ATS), which filters candidate CV's based on a set of keywords, skills, former employers, and years of experience. Such a system only results in a set of candidates best suited for the position.

This leads to one of the main issues with current ATS's - the fact that they lack any visualization functionality and cannot provide an overview of employment or education histories. Furthermore, these systems do not have an option of comparing general information, languages, skill sets, or work/education history from multiple resumes. This is a problem, because hiring managers receive a set of candidates well suited for the position from an ATS, but they still need to analyze the resume data manually, compare it with other resumes in order to find the best applicant. This process is highly inefficient, inherently slow, and can be greatly improved by implementing a framework which can provide the user with a clear overview of the applicant's information, such as languages, skills, and education/work history using visualization techniques as well as options to further filter and compare specific events. Such a solution will assist the users in their task to compare multiple candidates and supports interactive and visual exploration of chronological and other data present in resumes.

1.2 Aim of the Thesis

The goal of this thesis is to design and develop a web application that utilizes VA and IV techniques to provide the user with a clear overview of the resumes as well

as a set of functions to support the user in her/his tasks of comparing and extracting relevant information from resumes. This solution will further provide the means for the visualization and comparison of a large number of resumes simultaneously. The approach is based on the information seeking mantra of Shneiderman [Shn96] by providing a clear overview, zoom and filter functions, and details on demand.

The main research question of this thesis is:

- Can VA and IV support the visual exploration and comparison of chronological events across several instances of resumes in a meaningful way?

In order to determine which visualization techniques and interaction methods we will be using in our approach, we need to answer the following questions as well:

- What kind of tasks do users have?
- What kind of results do users expect?
- What kind of data do the users work with?
- How can we best visualize that data to support the user in her/his tasks?
- How can the information seeking mantra of Shneiderman [Shn96] be supported?

1.3 Structure of the Thesis

The remainder of this thesis is structured into five chapters, which correspond to the steps taken into the design and development process of the solution we propose. The chapters are:

- In section 2 we perform literary research and review state of the art and existing approaches to visualizing multiple chronologies as well as analyze similar solutions and critically discuss the advantages and disadvantages of each.
- In section 3 we describe our chosen research methods and discuss the design requirements of our suggested solution.
- In section 4 we provide an overview of our approach, highlight and showcase the implemented features. We also discuss the infrastructure of the application as well as implementation and feature details.
- In section 5 we evaluate our solution by conducting a small scale user study and highlight potential issues as well as improvements based on the user's feedback.
- In section 6 we conclude the thesis, provide answers to the research questions and hypotheses previously stated, and discuss what features could be implemented in future work.

CHAPTER 2

State of the Art

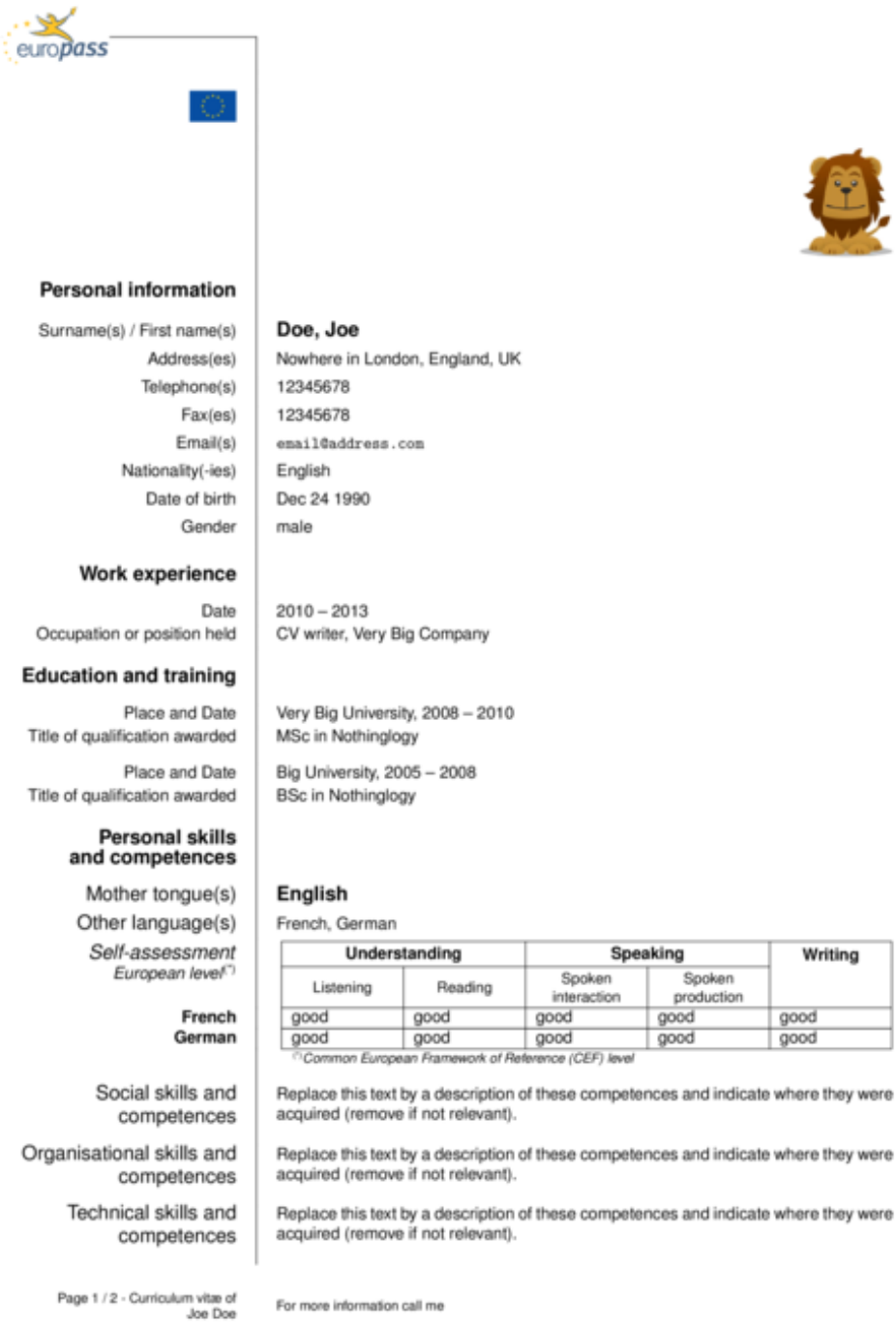
This chapter will establish the current state of visualization techniques for resumes and ATS's. As mentioned previously the current state of affairs for analyzing, filtering, and finding resumes best suited for a given position is to use an ATS. In her article Lauren Weber [Web] quotes Ed Struzik, an International Business Machines Corp. expert, that roughly ninety percent of large companies use ATS's to search for qualified candidates from large applicant pools. These systems help employers by analyzing resumes and CV's, surfacing candidates that best match the position's requirements and filtering out those who do not. They do not however provide any intuitive way of comparing multiple candidates simultaneously or means for visualizing data that is present in resumes. Using appropriate IV techniques can help emphasize an applicant's strengths and assist recruiters in efficiently extracting relevant information from resumes. In this section we would like to analyze the different types of information is present in resumes and discuss state of the art techniques for visualizing that information. We will also describe and compare some of the more popular open source ATS's.

2.1 Resume Data

A typical resume is comprised of several sections, each having a different type of data associated with it. In figure 2.1 a typical EuroPass resume is depicted - EuroPass is the de facto standard for a resume. In this section we will analyze the data structure, types, and contents of the multiple sections of an exemplary resume.

The sections of a resume, as can be seen in the EuroPass template, are:

- General and Personal Information
  - Fore- and Surname
  - E-Mail



europass

Personal information

Surname(s) / First name(s)	**Doe, Joe**
Address(es)	Nowhere in London, England, UK
Telephone(s)	12345678
Fax(es)	12345678
Email(s)	email@address.com
Nationality(-ies)	English
Date of birth	Dec 24 1990
Gender	male

Work experience

Date	2010 – 2013
Occupation or position held	CV writer, Very Big Company

Education and training

Place and Date	Very Big University, 2008 – 2010
Title of qualification awarded	MSc in Nothinglogy
Place and Date	Big University, 2005 – 2008
Title of qualification awarded	BSc in Nothinglogy

Personal skills and competences

Mother tongue(s) **English**

Other language(s) French, German

Self-assessment *European level*(*)	Understanding		Speaking		Writing
	Listening	Reading	Spoken interaction	Spoken production	
French	good	good	good	good	good
German	good	good	good	good	good

(*) *Common European Framework of Reference (CEF) level*

Social skills and competences	Replace this text by a description of these competences and indicate where they were acquired (remove if not relevant).
Organisational skills and competences	Replace this text by a description of these competences and indicate where they were acquired (remove if not relevant).
Technical skills and competences	Replace this text by a description of these competences and indicate where they were acquired (remove if not relevant).

Page 1 / 2 - Curriculum vitæ of Joe Doe

For more information call me

Figure 2.1: An example of the EuroPass resume template. Image from EuroPass (www.europass.cedefop.europa.eu/).

  - Location
  - Phone number
  - Social Profiles
  - Birth date
  - Website

- Education history
  - Such information can be schools or universities attended, internships, and training. Each entry has a title, description, start and end date, qualification, and location associated with it. (This data contains temporal as well as spatial information)

- Work history
  - Such information can be positions held at a company, projects, or collaborations the person has participated in. Each entry has a title, description, start and end date, company, position, and location associated with it. (This data contains temporal as well as spatial information)

- Languages
  - Each entry is associated with the name of the language as well as a level of proficiency. (There are six levels of proficiency for a language - A1, A2, B1, B2, C1, C2).

- Professional Skills
  - Such traits are value-added skills essential to any career. These can have a level of competence associated with them.

- Other Skills
  - This section of the resume typically contains social skills and hobbies. These skills do not generally have a level of competence associated with them.

From the different sections we can formalize the information present in resumes as belonging to any of the following three types of data summarized in table 2.1.

In the following section we will discuss different IV techniques that apply to the aforementioned types of data.

Data Type	Resume Information
Nominal Data	General, Personal Information & Other Skills
Ordinal Data	Languages & Professional Skills
Temporal & Spatial Data	Education & Work history

Table 2.1: Data types and information associated with resumes

2.2 Vislualization Techniques

2.2.1 Nominal Data

Nominal or textual data is present throughout the sections of a resume, mostly prominent in the 'personal information' and 'other skills' sections. There are numerous techniques for visualizing textual data such as e.g., TextArc depicted in figure 2.2 and Word Clouds illustrated in figure 2.3.

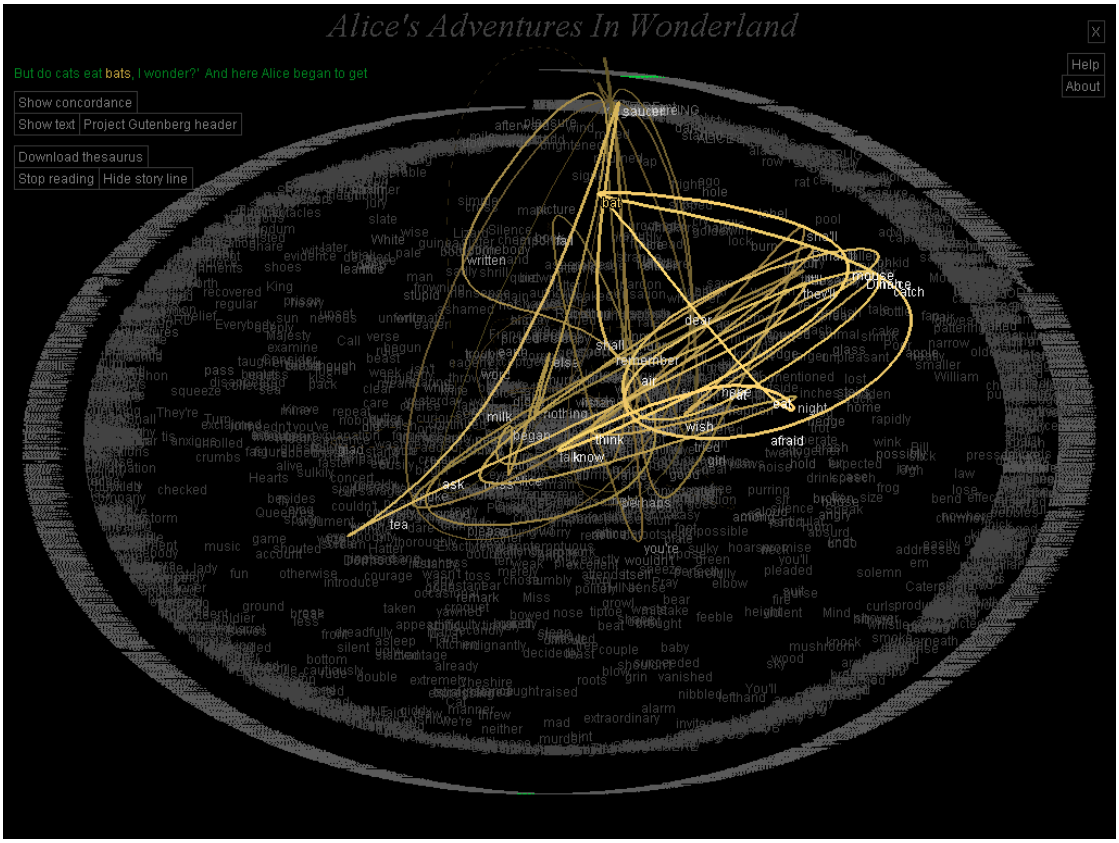


Figure 2.2: An example of the TextArc visualization technique. Image generated from TextArc (www.textarc.org).

Figure 2.3: An example of a word cloud visualization. Image generated from WordItOut (www.worditout.com).

The TextArc visualization technique surveyed by [PSB10] consists of two levels. The first level displays the original text around the periphery of the visualization area. The second level generates an interconnected graph of terms in the middle of the visualization area. The two levels are linked together, meaning that a user is able to select any term in the middle area and can quickly see its context in the full text that is displayed along the periphery. This approach is most useful when applied to full text visualization to explore relationships between different text segments, but in the context of resume visualization it is difficult to interpret this specific visualization technique.

The Word Cloud visualization technique is conceptually similar to a histogram, with the additional flexibility that the relative importance of each word can be highlighted by varying font-size, color, or orientation. This approach is well suited for visualizing a set of words, such as the data present in the 'other skills' section of a resume. An evaluation of the usability of Word Clouds was performed by [SKK+08]. The result of the evaluation is that this visualization technique is easily comprehensible, intuitive, and requires no previous knowledge or expertise in the field of IV.

In the context of resume visualization the most useful and intuitive method for text visualization is the Word Cloud. The visual representation of text using a Word Cloud can be interpreted with ease - recruiters can quickly extract and grasp our competencies and skills from this solution.

2.2.2 Ordinal Data

Ordinal data is a type of data where the variables have a category associated with them. A level of measurement is used to describe that the data has a sense of order, but it is uncertain if the distances between the different categories are equal. For example the level of qualification obtained from an educational institution is considered ordinal data. The order of the values is an important distinction of ordinal data.

There exist a multitude of visualization techniques for ordinal data the most common being bars and graphs. In figure 2.4 an example of a bar chart is given where one dimension of the data is nominal (language) and the other is ordinal (proficiency). This visualization provides a clear, comprehensive, and intuitive representation of the data, in specific the languages and their corresponding proficiency. Other approaches can also be used to visualize the same data in a different fashion, such as the use of doughnut chart in figure 2.5.

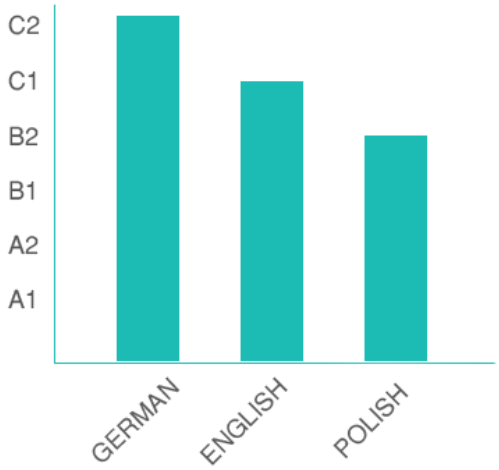


Figure 2.4: An example of a bar chart visualization.

Figure 2.5: An example of a doughnut chart visualization.

Another possible visualization of ordinal data is a star rating approach illustrated in figure 2.6. The star rating technique is like that of product ratings from online shops.

The number of stars corresponds to the proficiency level of the language in this example.

Figure 2.6: An example of a star rating visualization.

In the context of resume visualization, we can use both bar and doughnut charts to represent the professional skills and language sections of a CV. The use of these approaches provides the user with a clear overview of the professional skill set or languages of a person along with the corresponding proficiency of those skills or languages. These visualizations are intuitive and can be interpreted with ease by recruiters, which is why they are best suited for visualizing ordinal data in the context of resumes.

2.2.3 Temporal and Spatial Data

Certain sections of a resume, like work and education history, have temporal as well as spatial information associated with them. In these sections we will refer to the entries as events. Events have a start date (required), end date (optional), location, title, and description associated with them. If the end date is not specified we can assume that it is an ongoing event. Furthermore, we can differentiate between events that span an interval of time or have occurred in a specific point in time. An event that has occurred in a specific point in time could be an award or a course that took one day to complete. The most common and intuitive visualization technique for representing time-oriented data is the use of timelines illustrated in figure 2.7.

Timelines display a sequence of events on a horizontal axis depicting time. This visualization method provides the user with a quick and clear overview of the data represented in the work and education sections of a resume. One of the main drawbacks is that only a small subset of the event's information can be displayed on a timeline in a non-interactive environment - we can see the start and end date, along with the title of an event but the description, location, and the type of event is not visible. This limits the usage of the approach.

Other possible visualization techniques for temporal and spatial data exist as well, including PlanningLines by Aigner et al., [AMTB05], Mosaic Charts by Luz et al., [LM11], LifeFlow by Wongsuphasawat et al., [WGP+11], and VAiRoma by Cho et al, [CDW+16]. These approaches are used mostly in the context of project management, clinical environments to track patient histories, and exploring trends and topics in history.

These techniques are well suited for representing and visualizing temporal and spatial data associated with events. We will discuss these solutions in the following section 2.3.

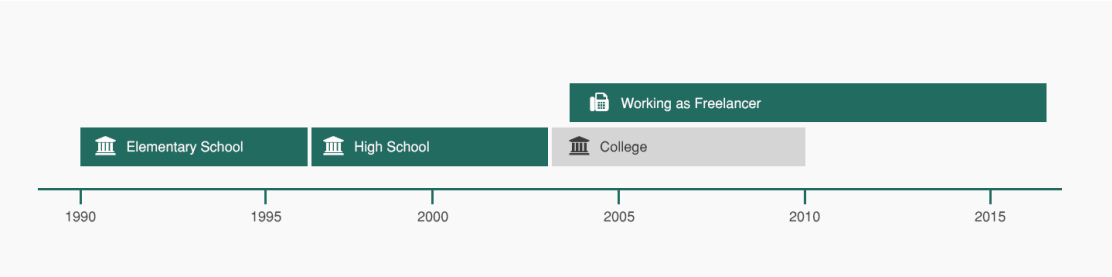


Figure 2.7: An example of a timeline visualization.

2.3 Literature Review

There exist multiple approaches outside the context of resumes that tackle the problem of representing multiple chronologies or event sequences and use specific visualization techniques to provide a clear overview. These solutions are described in the following subsections and we discuss their advantages and disadvantages related to visualizing and comparing CV information.

2.3.1 PlanningLines

PlanningLines is an approach that provides a solution to temporal uncertainties associated with events in the context of project management by introducing novel glyphs called PlanningLines. These glyphs allow for the representation of time attributes as an interval rather than an instant [AMTB05]. A PlanningLine visually encodes the start, end, and duration of an event as well as the uncertainties associated with it by extending time attributes in the following manner:

- start interval
  - earliest starting time
  - latest starting time
- end interval
  - earliest finishing time
  - latest finishing time
- duration
  - minimum duration

– maximum duration

An example of a PlanningLine glyph and a simple project plan using the glyphs are illustrated in figures 2.8 and 2.9.

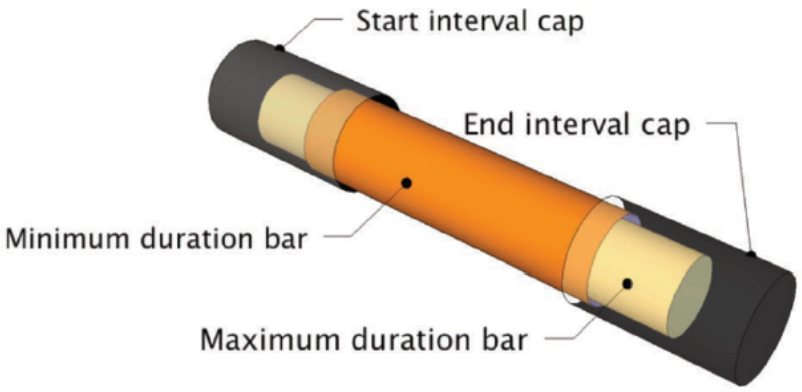


Figure 2.8: PlanningLine glyph representing uncertainties associated with events. Image from Aigner et al., [AMTB05].

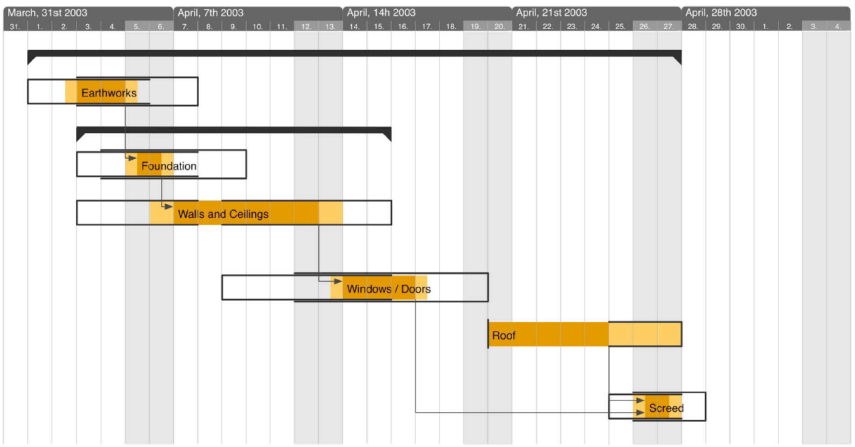


Figure 2.9: Simple project plan using PlanningLines. Image from Aigner et al., [AMTB05].

This visualization technique provides a lot of information at a quick glance and succeeds in encoding uncertainties associated with events in the context of project management. PlanningLines provides a clean and clear overview of multiple events and intuitively represents indeterminacies that might be present in some events. This approach does not however apply well to resume visualization, specifically the work and education histories, as events do not have any uncertainty associated with them. In the context of resumes

there are other attributes associated with events such as location, event type, position, and company, which cannot be encoded with the PlanningLines approach.

2.3.2 Mosaic Chart

This approach proposes the use of a mosaic chart as a visualization technique in the context of visualizing and comparing project schedules. This solution is capable of aggregating multiple chronologies spanning a common period of time and visualizing the output in a single overview. Figures 2.10 and 2.11 illustrate a mosaic chart on the left side and the corresponding timeline representation on the right side.

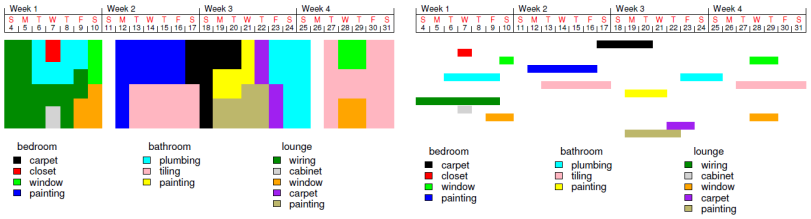


Figure 2.10: Example of a Mosaic Chart. Image from Luz et al., [LM11].

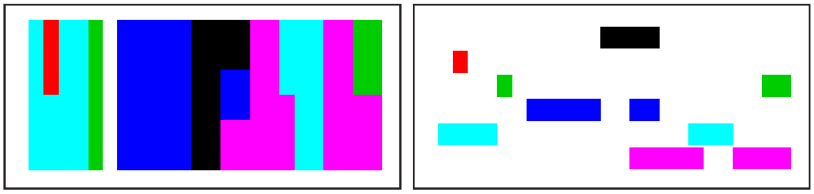


Figure 2.11: Example of a Mosaic Chart. Image from Luz et al., [LM11].

Unlike a timeline, which grows vertically on demand, a temporal mosaic is a form of visualization that starts with a fixed height and allocates space proportionally to the number of overlapping events in a given time interval. The event types are visualized through colors and not lines as represented in a timeline view. Temporal mosaics preserve the representation of overview, even as the available drawing area is reduced, and facilitate the detection of concurrent and overlapping events, even as the number of events to be represented increases according to the authors Luz et al., [LM11].

One of the drawbacks of this technique is that it lacks interactivity, which means most of the event's meta-information, that is relevant in the context of resume visualization, is lost in the overview. Furthermore, even though the overview is clear and intuitive the possibility of filtering event sequences and providing the user with details on demand is lacking. Mosaic charts are well suited for the detection of event overlaps and schedule comparison, but do not support the user in exploring and analyzing information associated with the events.

2.3.3 LifeFlow

LifeFlow by Wongsuphasawat et al., [WGP+11] is a scalable application capable of visualizing a large number of event sequences to support users in analyzing, exploring, and discovering patterns that are present in medical records. It provides an interactive visual overview of event sequences and supports the information seeking mantra of overview first, zoom and filter, then details on demand. An example of the LifeFlow application is depicted in figure 2.12. The type of event is color-coded (Arrival - blue, Emergency - pink, ICU - red, Floor - green, Discharge-Alive - light blue, and Die - black bar) and the height of the bars is proportional to the number of records in the data set. This approach is specifically tailored for patient medical records, although according to the authors it can be adapted to other fields such as incident management, log analysis, or the study of human activities in general.

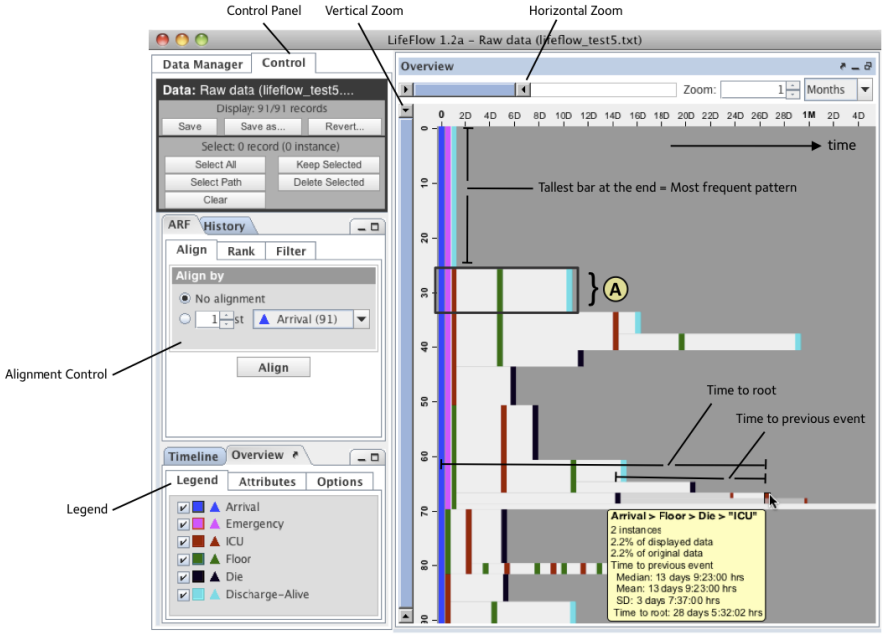


Figure 2.12: LifeFlow application overview. Image from Wongsuphasawat et al., [WGP+11].

This solution does not provide features to intuitively compare multiple data sets - the simplest way to do so is to run multiple instances of the application and compare the data side by side. This is not an efficient method of comparing data, because it requires the user to switch attention back and forth between multiple views and if more than two data sets need to be compared simultaneously the overview quickly becomes cluttered.

In general this approach is very well suited for exploring, analyzing, detecting, and understanding patterns of patient transfer data. In the context of resume visualization certain aspects of this technique can be adapted, such as the color coding and interactivity

of the events represented in the main view.

2.3.4 VAiRoma

VAiRoma by Cho et al, [CDW+16] is a VA system that provides users with an interactive visual interface with multiple linked views. This solution assists users in exploring, discovering, and comparing events, places, and times throughout Roman history.

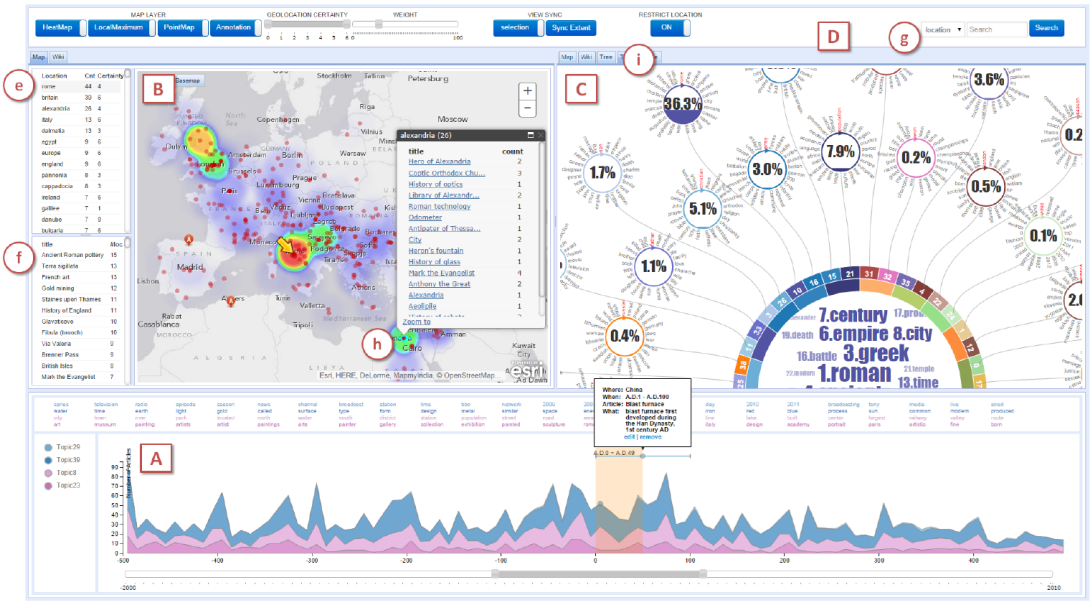


Figure 2.13: Overview of VAiRoma Interface. The interface has three main views: Timeline view (A), Geographic view (B) and Topic view (C). Image from Cho et al., [CDW+16].

VAiRoma illustrated in figure 2.13 is a web-based application that consists of three primary views - timeline view, geographic view, and topic view. Each of the views utilizes different visualization techniques and the views are interlinked to allow users to gain insights through analyzing and exploring large amounts of Wikipedia articles.

The timeline view (A) uses a stacked graph approach to visualize trends and topics over time. Each point in time represents the number of articles related to a certain topic. The timeline view is interactive and provides users features to zoom in or out and select a time period using mouse clicking or dragging.

The geographic view (B) utilizes three different layers to visualize the spatial data - heatmap layer, points layer, and pin layer.

The heatmap layer depicts the density of geolocations that are mentioned in articles. The heatmap layer is color-coded, with red areas indicating that the region is more frequently

mentioned. This layer intuitively visualizes geographic hot spots that are associated with the selected topic and time period.

The point layer depicts the exact locations in the geographic view compared to the heatmap which illustrates a general overview of the region. The point layer is interactive and displays information consisting of all Wikipedia articles that contain the selected location and time period.

The pin layer displays pins that indicate the locations selected by the users. This layer is useful in cases where several points on the map overlap. Users can click on pins to display Wikipedia articles associated with the selected location and time period.

Furthermore, VAiRoma offers a topic view (C), which utilizes multiple visualization techniques for displaying topic hierarchies, content, and topic weights. Users can select to view the topic data either as a tree view or a circluar view illustrated in figures 2.14 and 2.15.

The tree topic view offers a hierarchical visualization of topics and summarizes the main themes. Each topic in this view is associated with a unique number and color, which allows for them to be easily distinguished from each other. The tree topic view is interactive and offers users options to filter the collection to find articles that are most relevant to that topic. When selecting a topic the geographic and timeline views are also updated to display information associated with that topic.

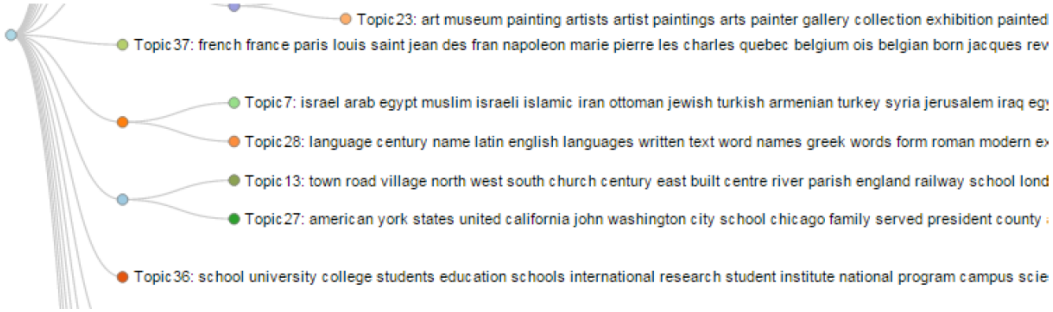


Figure 2.14: Tree topic visualization. Image from Cho et al., [CDW+16].

The circular topic view manages to visually encode the topic hierarchy well, whilst preserving a clear overview. The circular topic view is based on the sunburst graph visualization technique and offers users interactions to further explore information associated with each topic. The circular topic view is interactive and upon zooming into a certain level displays a word cloud of the selected topic's keywords.

The VAiRoma solution provides multiple linked views and different visualization techniques adapted for temporal and spatial data. We believe this approach supports the information seeking mantra of Schneiderman [Shn96] and provides users with a unique interactive experience to explore and discover relationships between events, topics, and

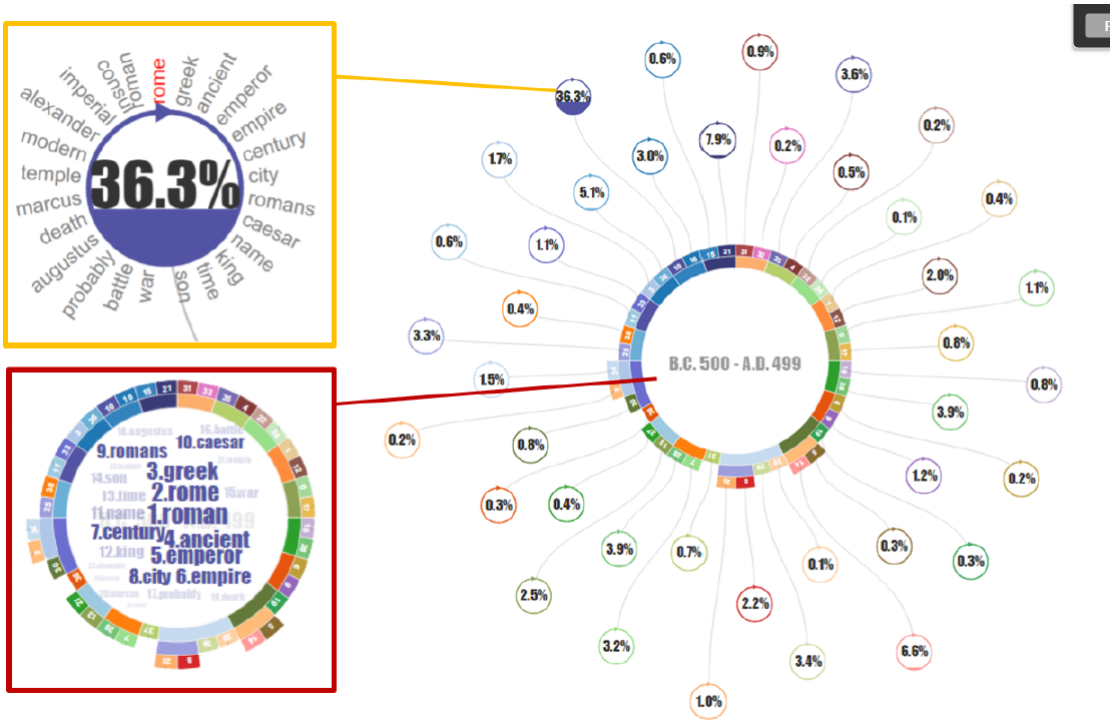


Figure 2.15: Circular topic vislualization. Image from Cho et al., [CDW+16].

places in Roman history. VAiRoma provides a clear overview, uses intuitive visualization techniques, and offers users options to select different visualization techniques. We believe that this approach is well suited for visualizing spatial and temporal information and can be adapted for the comparison and visualization of multiple instances of resumes.

2.3.5 Interactive Visual Profiling of Musicians

Stefan Jänicke et al., [JFS16] propose the development of an interactive visual profiling system for musicians, which utilizes IV and VA techniques to support users in iteratively determining similar musicians. The interface consists of various columns for visualizing the multifaceted data of a musician's attributes and is depicted in figure 2.16. The different columns represent meta-data related to a musician's musical and other professions, where they worked, and their denominations. Consistent color-coding is used throughout the interface to provide an intuitive means of visually distinguishing different musicians.

The column explorer consists of three different views, including the musician's lifetime data, a relationship graph, and a map. In the following we will discuss the different views.

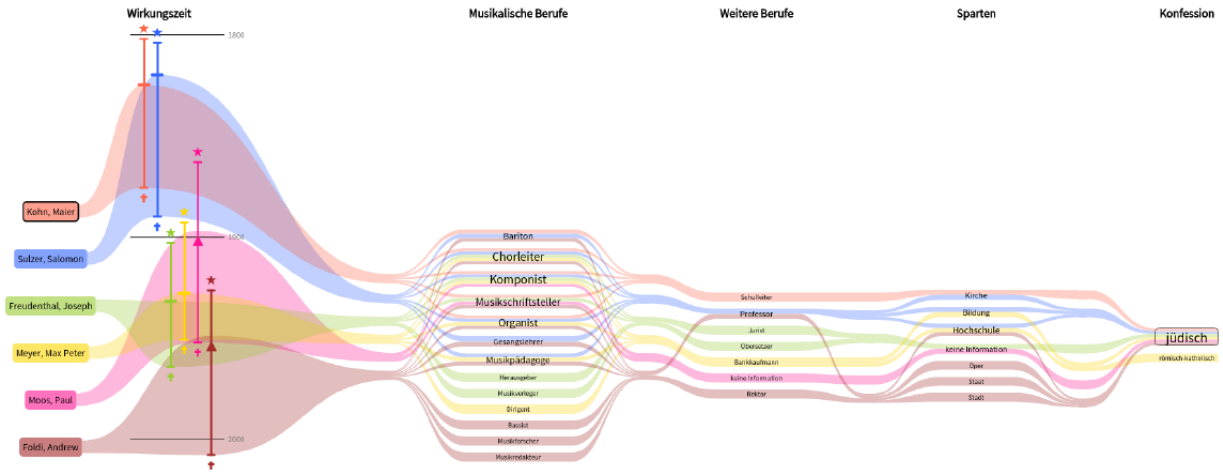


Figure 2.16: The column explorer visualizes a musician's meta data. Image taken from [JFS16].

Lifetime Data

The lifetime data of a musician, displayed in figure 2.17, is visualized as a vertical timeline with symbols depicting the date of birth (star) and death (cross).

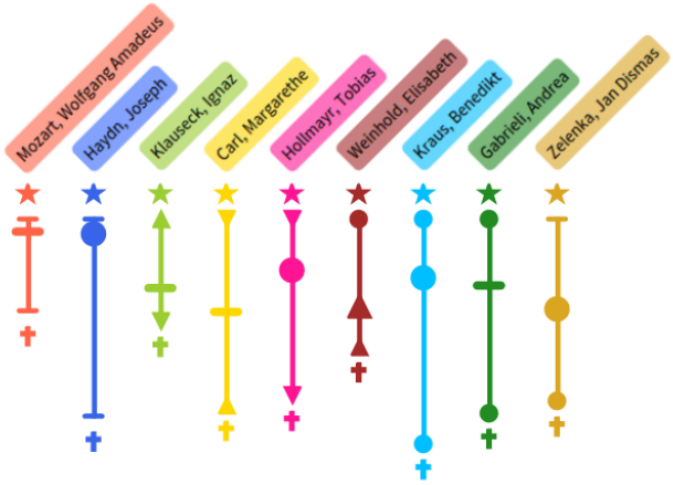


Figure 2.17: A vertical timeline displaying a musician's lifetime data. Image taken from [JFS16].

Due to uncertainties related with the datings of a musician's lifetime events, the authors opt to visually encode this information by using additional symbols, such as a horizontal line for *precise* dating, a circle for *around* datings, a triangle for *before* datings, and

an upside down triangle for *after* datings. The use of these symbols is well suited for the visualization a musician's lifetime events and the corresponding uncertainty in a constrained space. We believe that the use of symbols provides an intuitive and clear overview of a musician's lifetime data, however this visualization technique is not interactive and does not provide users with the possibility of viewing detailed information or comparing events.

Relationship Graph

The relationship graph of the profiling system provides an interactive visual view of a musician's social network. This visualization technique assists users in discovering, exploring, and analyzing the relationships between different musicians. The relationship graph is illustrated in figure 2.18 and provides more detailed information on demand by using mouse hover and click events.

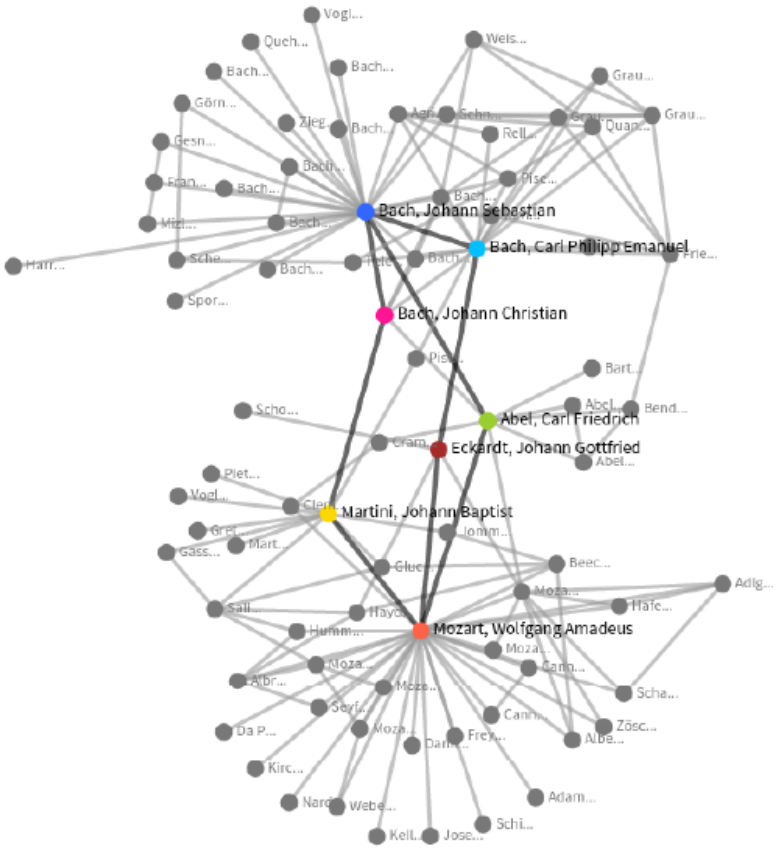


Figure 2.18: Relationship graph visualizing a musician's social network. Image taken from [JFS16].

Hovering over a gray vertex will display the full name of them musician and hovering

over an edge will display information about the roles of the connected musicians and their relationship.

Map

The map of the visual profiling system, illustrated in figure 2.19, displays all places of activity for the selected musicians. The idea behind this visualization technique is to provide an intuitive means of displaying and interpreting the information associated with an activity region and to support users in the discovery, exploration, and comparison of different activity regions. The locations of activity for musicians are displayed as color-coded markers plotted on the map. This allows users to intuitively distinguish between the different musicians, as each is associated with a unique color throughout the interface.

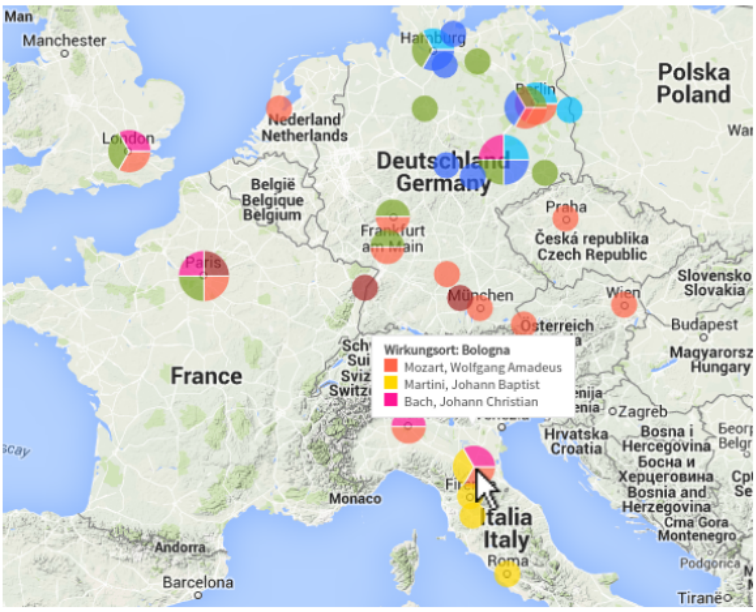


Figure 2.19: Map visualizing all regions of activity for the selected musicians. Image taken from [JFS16].

In cases where multiple musicians have the same region of activity, the information is consolidated into a pie chart and details, regarding the name of the location and a list of related musicians, are displayed on mouse hover effects. The markers on the map are displayed at lower opacity to avoid the loss of the underlying geographic information. We believe that this visualization technique manages to encode the spatial information related to a musician's region of activity in an intuitive and clear manner, whilst providing a clean overview. Furthermore the interactivity of the map provides users with details on demand.

In summary we believe that the interactive visual profiling system proposed by Stefan Jänicke et al., [JFS16] provides an intuitive overview of a musicians lifetime data, relationships and regions of activity. The IV and interaction techniques used support the information seeking mantra of Shneiderman [Shn96]. In the context of resume visualization and comparison we can adopt and modify certain aspects of the column explorer to better support users in exploring, analyzing, and comparing lifetime data across multiple instances of resumes.

2.3.6 Space Time Cube

In contrast to the aforementioned approaches and solutions the Space-Time-Cube technique offers visualization of the spatial and temporal dimensions of a sequence of events. This approach is based on Hägerstrand's time geography [Häg70] - the study of space-time behavior of human individuals. In their daily life each individual follows a trajectory through space and time. Hägerstrand sees both space and time as inseparable, and this becomes clear if one studies the graphic representation of his ideas, the Space-Time-Cube as depicted in figure 2.20.

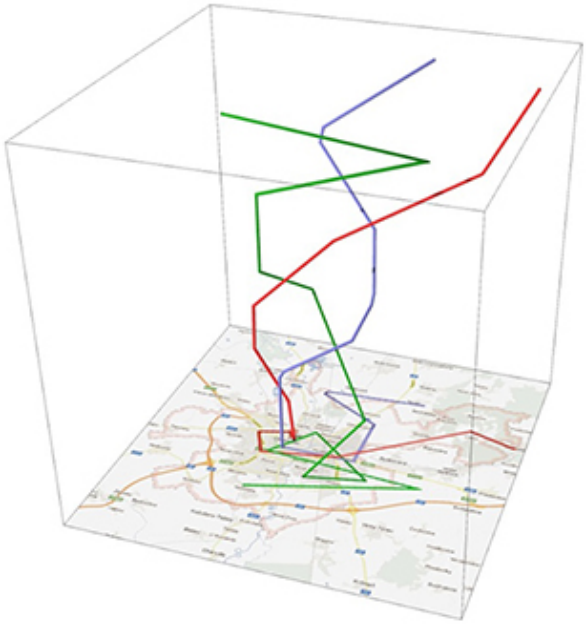


Figure 2.20: Space Time Cube approach overview. Image from www.intechopen.com.

This technique offers an interesting visualization of the spatial and temporal information in a three-dimensional cube, where the vertical axis (z-axis) is the variation in time, the horizontal axes (x- and y-axis) are a representation of the underlying geography and the paths depict the individual timelines or sequences of events, each associated with a

location. Each path or timeline is associated with a different color, which helps the users differentiate the sequences of events.

The Space-Time Cube is most suitable for visualizing paths of multiple individuals, groups, or other objects moving through space and has a wide range of applications such as e.g., real-time monitoring of runners, Geographic Information System (GIS), or a sequence of historic events.

The Space-Time Cube solution manages to visualize the spatio-temporal information that is present in sequences of events in an intuitive manner and supports the information seeking mantra of Shneiderman - overview first, zoom and filter, then details on demand. There are some drawbacks to this solution as well - mostly related to the usability of the technique. The most prominent pitfalls of the Space-Time Cube is that the overview quickly gets cluttered when attempting to visualize a large number of timelines and that it requires an interactive environment. Furthermore, the interaction in three-dimensions is not intuitive for most applications.

2.4 Applicant Tracking Systems

As mentioned earlier in this chapter ATS's are the state of the art when it comes to analyzing, filtering, and finding candidates best suited for a given position. In this section we will give a short overview of a couple of the open source solutions for ATS's and discuss the advantages and disadvantages of each related to visualizing and comparing resume data.

2.4.1 OpenCATS

OpenCATS (www.opencats.org) is an open-source web-based ATS designed to assist recruiters in managing job postings and candidate selection. The application provides the user with options to explore candidates, positions, and companies.

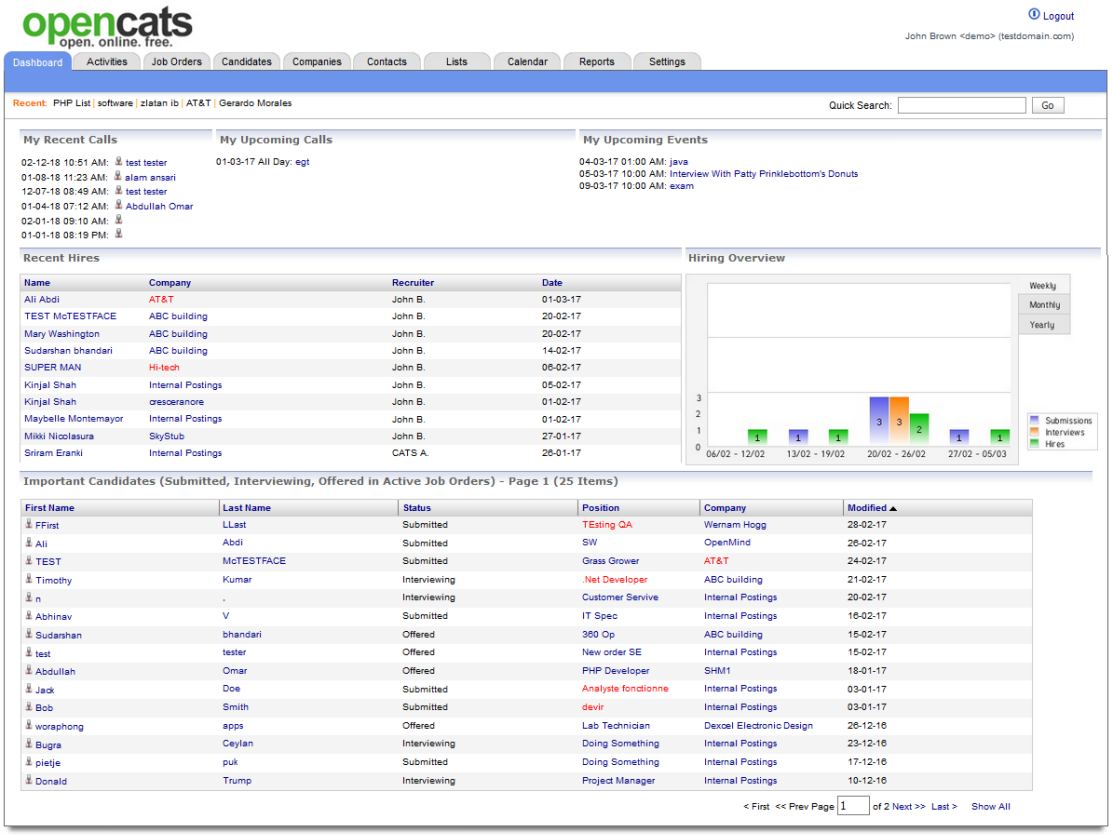


Figure 2.21: The OpenCATS System: Overview of the dashboard.

Upon entering the application the user is presented with a dashboard depicted in figure 2.21 providing an overview of recent hires, important candidates, hiring overview, and upcoming calls, or events. The dashboard displays important information regarding the current state of the hiring process. It is noticeable that in the overview one IV technique

is used, and that is a bar chart, illustrating the hiring overview for the last week, month, and year.

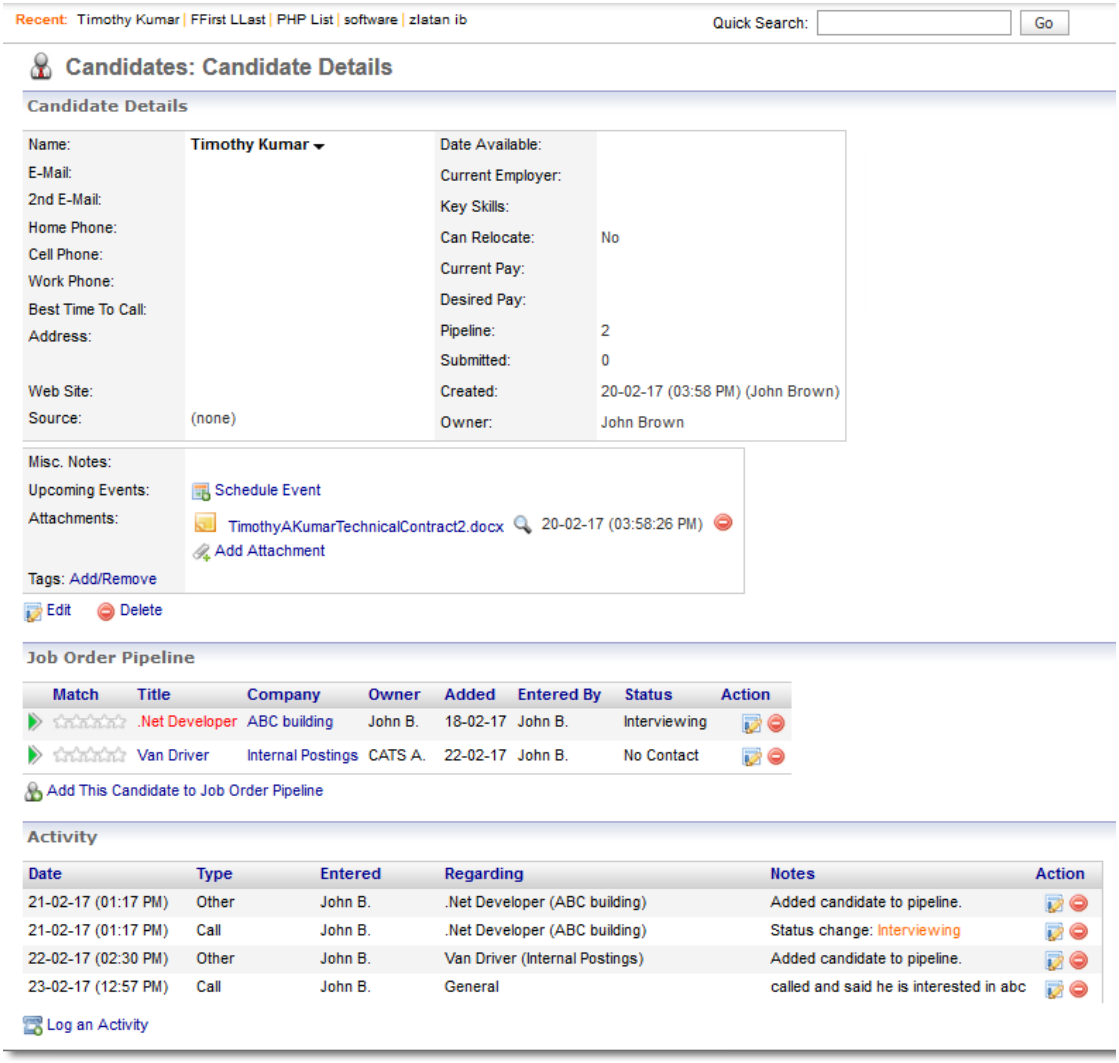


Figure 2.22: The OpenCATS System: Detailed view of a candidate.

The important candidates section offers a list of applicants along with details about their name, status of the application, current position and company. This list does not provide any visualization techniques and cannot be modified to display more information. When clicking on a candidate's name from the list the user is redirected to a detailed view illustrated in figure 2.22, which highlights general and personal information, jobs the applicant has applied for, an overview of the candidate's activities, and a miscellaneous section for attachments, such as a resume or cover letter. We can notice that no visualization methods are used for any of the data present in the detail view and information regarding the applicant's previous work or education history, languages,

professional, and other skills is completely missing.

An important feature that this solution lacks is the ability to quickly compare and provide an overview of each of the candidate's strengths and weaknesses, which is one of the main tasks that recruiters have. Currently multiple candidates can be compared in the application by opening multiple tabs and viewing the applicant's side-by-side, but this is not an efficient method of comparing large amounts of candidates as it requires attention switching and the user quickly loses overview.

In summary this solution lacks any type of IV methods, except bar charts summarizing the hiring process and does not really support the user in the task of exploring, analyzing, and comparing multiple candidates.

2.4.2 Yawik

Another example of a popular open-source web-based ATS is Yawik (www.yawik.org), which offers an online portal for applicant tracking, job posting, and administration.

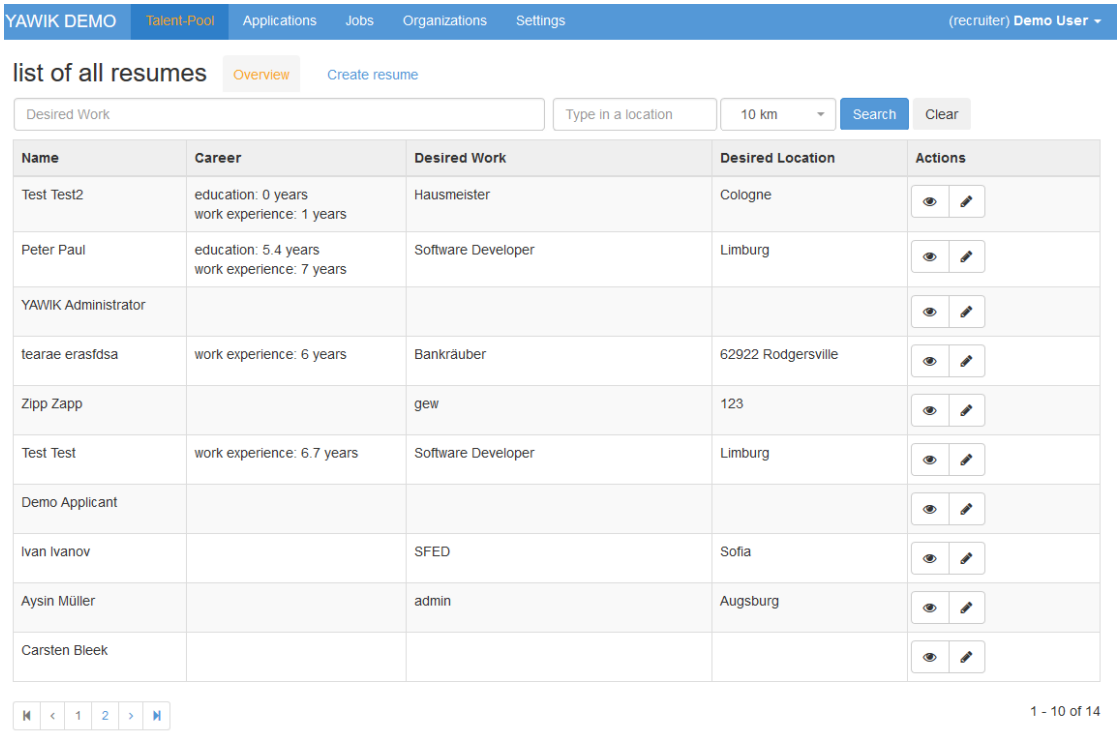


Figure 2.23: The Yawik System: Overview of the dashboard.

The application provides an overview of the talent pool - a list of all candidates along with their career experiences, desired work and location, and actions to view or edit the applicants. The list of all resumes is illustrated in figure 2.23 and provides options to

filter applicants depending on their location and desired work. The application provides no overview of recent hires or other useful information in contrast to OpenCATS.

Yawik provides users with a set of features to explore and analyze applicants, job postings and companies. When clicking on an applicant we are navigated to a detailed view, where we can see more information about that candidate. The detail view is depicted in figure 2.24 and displays sections on the candidates' personal information, attachments, work experience, education, and training as well as an overview of the current state of the application. The detail view offers more information about the applicant compared to OpenCATS, but sections about languages and professional skills are still lacking. The detail view presents the resume data as is, without implementing any IV techniques, which leads to recruiters possibly missing important information because it was not emphasized or displayed correctly. Furthermore, Yawik does not provide users with the ability to compare multiple candidates, leaving recruiters to perform a side-by-side comparison of the strengths and weaknesses of each applicant.

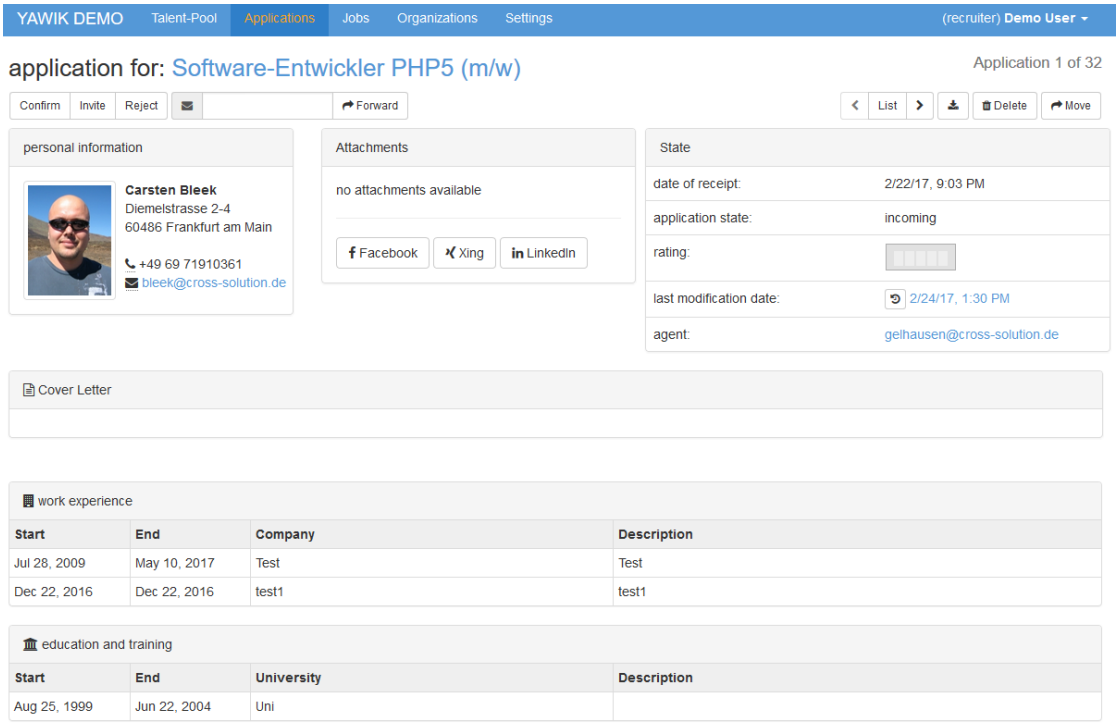


Figure 2.24: The Yawik System: Detailed view of a candidate.

In conclusion Yawik offers a clear overview of the applicants and displays more relevant data in the detail view of each candidate but does not provide any means of visualizing information at all, which in turn makes it difficult for recruiters to recognize and extract important information. The application additionally lacks features to support the users

in comparing multiple resumes.

CHAPTER 3

Methodology

This chapter describes the approaches and methods used to conduct our thesis. The aim of this thesis is to analyze state of the art visualization techniques used in the spatio-temporal domain and find how they can be applied to the context of resume data, specifically visualizing multiple chronologies and allowing for the comparison of multiple instances of resumes. The methods we use are to reflect on current issues with state of the art visualization techniques and to improve on the feature set of ATS's described in section 2. In the following subsections we will discuss the research methodology that we have used to answer the hypothesis of this thesis and formalize the design requirements and models that our solution will have.

3.1 Research Methodology

In this section we will reiterate the main research question and sub-hypotheses of the thesis and describe our chosen research methods used to answer those questions.

3.1.1 Research Questions

The main research question of this thesis is:

- Can VA and IV support the visual exploration and comparison of chronological events across several instances of resumes in a meaningful way?

Answers to the following questions will determine the visualization and interaction techniques we will use in our proposed solution:

- What kind of tasks do users have?

- What kind of results do users expect?
- What kind of data do the users work with?
- How can we best visualize that data to support the user in her/his tasks?
- How can the information seeking mantra of Shneiderman [Shn96] be supported?

3.1.2 Research Methods

The goal of the project is to create a tool capable of assisting users (for example employees in the HR department) in the process of comparing and analyzing a number of resumes simultaneously, whilst preserving a clear overview of multiple chronologies and providing details on demand for each event. We expect this approach to greatly improve the efficiency of the hiring process and assist the user(s) in finding the best suited candidate for a given position. We plan on accomplishing this by designing and implementing a framework capable of utilizing IV techniques and provide users with the functionality to compare a number of resumes simultaneously. We expect to achieve these results by applying the design triangle framework proposed by Miksch et al., [MA14] illustrated in figure 3.1.

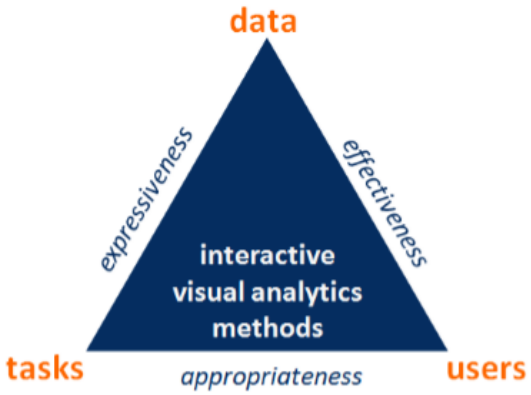


Figure 3.1: The design triangle framework. Image from Miksch et al., [MA14].

We consider the three main aspects of designing and developing such an application, namely the characteristics of the data, the users and their domain of expertise, and the user's tasks. The first step towards the implementation of the application is to answer the following questions:

- What kind of data are the users working with?
- Who are the users?
- What are the tasks of the users?

For the purpose of collecting information and data, that will help answer the research questions of this thesis, we use both qualitative, quantitative, and literary research methods.

We further propose to validate the design and visualization models by applying a nested model approach proposed by Tamara Munzner [Mun09], which consists of four nested layers depicted in figure 3.2. Where the four layers are:

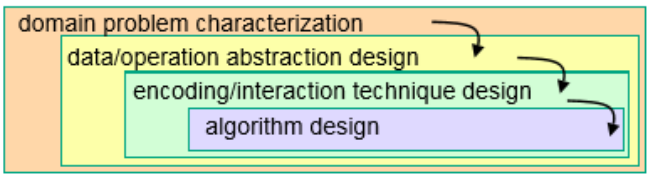


Figure 3.2: The nested model approach. Image from Tamara Munzner [Mun09].

- To characterize the tasks and data in the vocabulary of the problem domain
- To abstract into operations and data types
- To visually encode the data and interaction techniques
- To implement algorithms that efficiently execute these techniques

In the following section we will describe the design requirements, determining user expectations, and state the important characteristics that our solution must provide to assist users in their tasks.

3.2 Design Requirements

We have compiled a list of tasks that the users have when comparing and analyzing candidates for a given position. We abstract the requirements from the design and implementation in this step, the purpose of this section is to outline the user's needs and provide functions to support the successful execution of the their tasks. The tasks along with a short description is provided in the following list:

- Search
  - Search should allow users to filter candidates based on their current location, location associated with events, their skillset, age, languages, and occupation. The search is performed by applying a user given query to the set of all CV's and returning a subset of resumes matching that query. The results can then be further filtered, explored, or compared by the users.

- Add
  - The application should provide an intuitive way of adding resumes and uploading them to a database. The purpose of this feature is to allow users to easily add and upload the contents of a resume to the database. Users can upload a profile picture along with multiple certifications, honors, awards, events, skills, languages, and the other sections of the resume described in section 2.1.
- Edit
  - Resumes should be editable while adding or after being uploaded. We would like to provide the users with an intuitive and effortless way of editing, adding, and removing the contents of a resume before and after it has been uploaded to the database.
- View
  - The proposed solution should provide a straightforward way of viewing applicants and visualizing their information. This feature should allow users to view a specific candidate and explore the sections of the resume in more detail. The detail view should also provide techniques to encode the resumes' data visually in an intuitive and clear manner. This view should further support the information seeking mantra of Shneiderman.
- List
  - Another requirement is the feature to list all candidates and prove users with the means to filter them based on a query. The list should display an overview of the candidates with additional details related to their certifications and education.
- Compare
  - One of the most essential features that the application needs to support is to provide functions to interactively compare and allow for the exploration and analysis of multiple instances of resumes. This functionality is crucial as it facilitates the visual exploration, comparison, and discovery of information and assists users in finding the best suited candidate for a given position.

The implementation and design of the aforementioned requirements are described in detail in the next chapter 4.

3.3 Models

In our solution we combine the advantages of two implementation models - the Waterfall and Rapid Application Development models. Both of these models are described in the following subsections.

3.3.1 Waterfall Model

The Waterfall model, introduced by Bell et al., [BT76], is a popular life cycle model for software engineering, often considered a classic approach. This model is depicted in figure 3.3 and describes a development method that fixed and linear, meaning that there are distinct goals in each phase of development. Each phase needs to be completed before advancing to the next one. Advantages of this model include:

- Each stage has a set deadline.
- Each phase is planned in detail.
- Projects are delivered on time.
- Product can proceed through the development process.

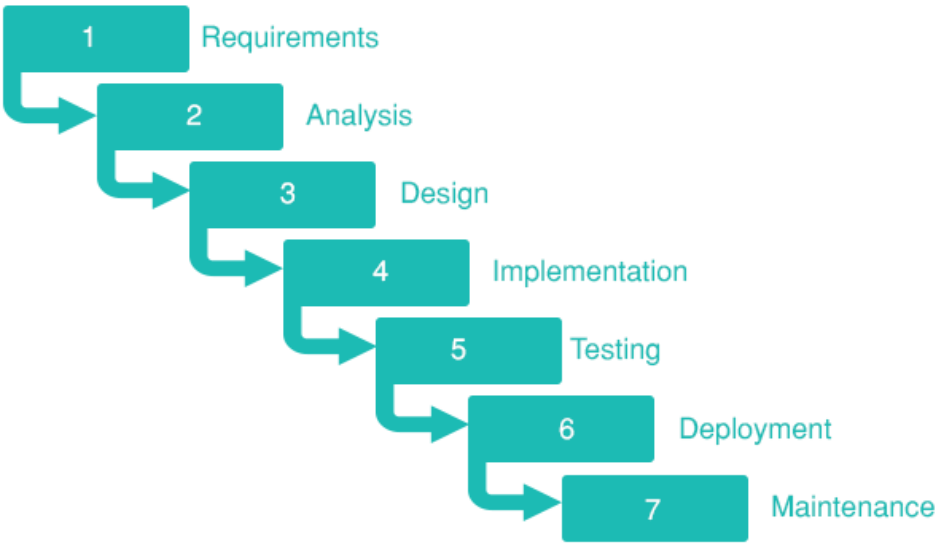


Figure 3.3: The Waterfall model.

We employ this model in our solution to define the stages of early development, plan the details, set deadlines for each phase of development, and provide an option to refine features in a later stage based on user feedback.

3.3.2 Rapid Applciation Development Model

The Rapid Application Development (RAD) model, illustrated in figure 3.4, was introduced by James Martin [Mar91] and refers to any programming language that offers possibilities to quickly implement features compared to traditional frameworks such as C/C++. Such languages are called Rapid Development Languages (RDLs) and reduce project costs by shortening the amount of time needed to build a product.

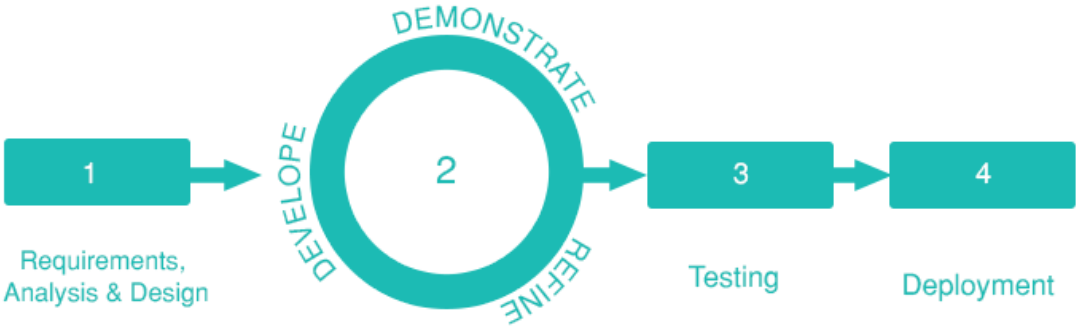


Figure 3.4: The Rapid Application Development model.

The RAD model proposes that products can be developed faster by:

- Using workshops and focus groups to compile design requirements.
- Prototyping and user testing of designs and interactions.
- Re-using software and application components.
- Following a schedule that defers design improvements to the next iteration of the product.

We utilize the advantages of the RAD model in our implementation stage by quickly implementing and testing prototypes in order to assess the advantages and disadvantages of each change.

CHAPTER 4

Conceptual Design and Implementation

This section describes the implementation of the prototype and provides technical information on the frameworks used and features supported by the application. We begin by giving a short overview of the CV3 application and explain the User Interface (UI) and interaction elements, then continue with an introduction to the architecture and implementation details of the prototype. We have named our solution CV3 because it is based on a previous project for resume visualization we have designed and developed, called CV2.

4.1 Overview

CV3 is our proposed solution to visualizing and comparing multiple instances of resumes in an interactive environment. The application offers users the possibility to visually explore, analyze, and compare the education and work histories, skillsets, and languages of candidates.

4.1.1 Views and Interaction Elements

Our solution consists of multiple views providing an overview, details on demand, and the possibility to add new or edit existing resumes in the database. The application consists of the following views:

- Add/Edit View
  - Views to add new resumes to the database or edit existing ones.
- Detail View

  - Detailed view of a single resume.
- List View
  - Lists all the resumes currently in the database.
- Compare View
  - View which enables the comparison of multiple resumes.

In this subsection we will go over the different views and discuss what features and interaction elements they have.

Adding new resumes

The view to add new resumes to the system is depicted in figure 4.1 and displays the inputs for the different sections of a resume that we have described in section 2.1. The implementation of the resume model can be found in the Appendix section A.1 In the following we will briefly go over the different parts of this view and describe them.

(A) General and personal information

View A in the add view provides inputs for the general and personal information section of the resume. Candidates can upload a profile picture, enter their name, occupation, date of birth, add social profiles, and provide contact details such as address, email, and phone number. There is also an 'About me' field where applicants can write a short summary of who they are, what they do, and what interests them.

(B) Languages and proficiency

View B is where the candidates would input information on what languages they know along with the corresponding proficiency levels. We use the standard Common European Framework of Reference for Languages (CEFRL) to distinguish between six language proficiency levels ranging from A1 (beginner) to C2 (native speaker).

(C) Timeline for education and work histories

View C provides applicants with inputs for their certifications, honors, awards, and details on their completed education. Additionally the 'Timeline' view offers the possibility to enter events, such as work or education histories. An event has the following properties:

- Title
  - The title will be displayed on the events in the timeline view.
- Description

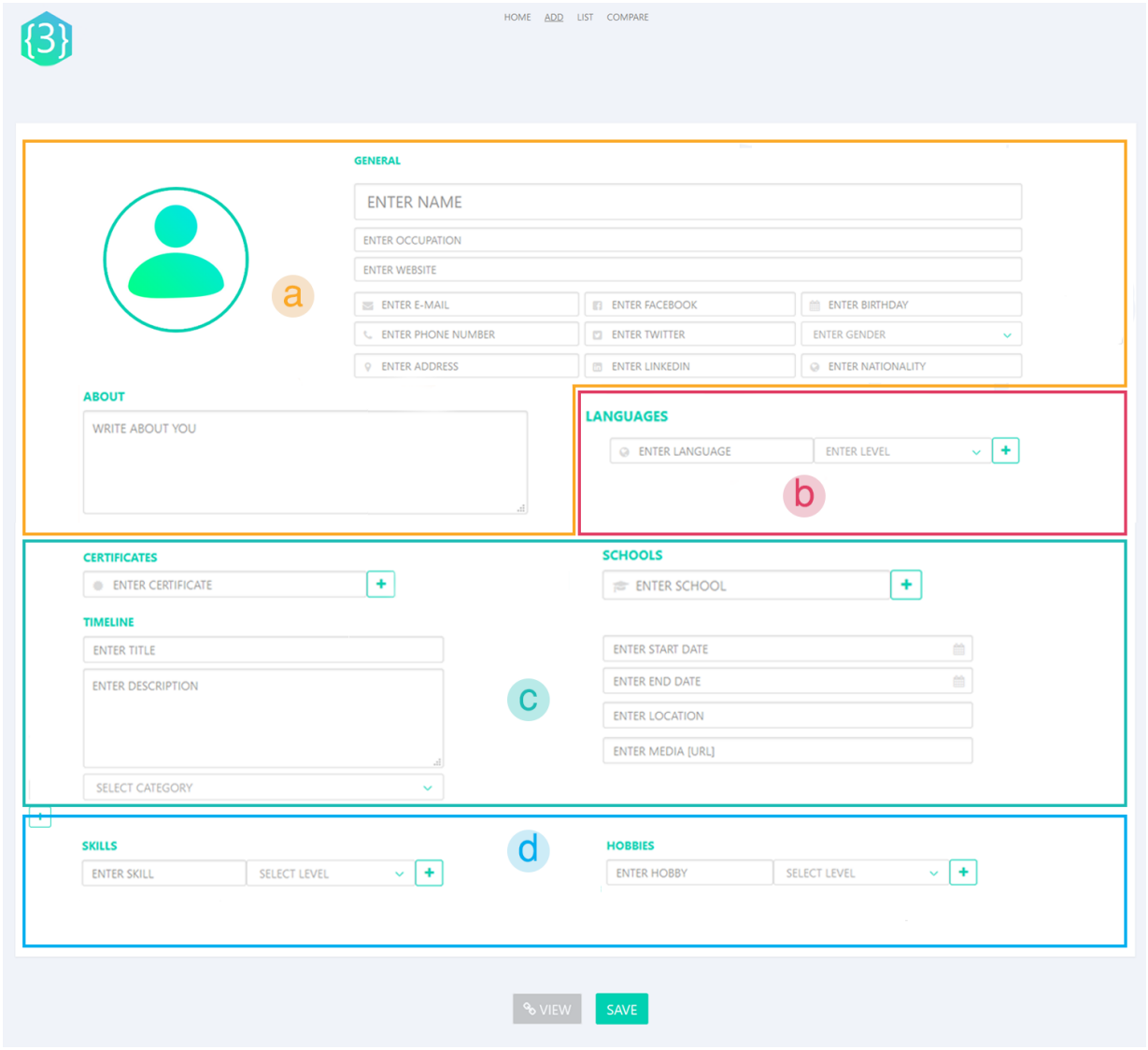


Figure 4.1: View to add new resumes consisting of multiple views for a) general and personal information, b) languages and proficiency, c) a timeline to enter work and/or education events, and d) professional and other competencies.

  - Applicants can provide a brief description of the event.

- Start date
  - Start date is required.
- End date
  - The end date is an optional input, if not selected we assume that the event is ongoing.
- Location
  - Each event is usually associated with a location, as to where it occurred.
- Media
  - Candidates can add media associated with an event, such as for example a picture.
- Category
  - The category can be work, education, internship, or project. If the category 'Work' or 'Internship' is selected the applicant is provided with two additional input fields where information on the position and company can be given.

(D) Professional and other competencies

View D offers the possibility for applicants to showcase their professional and personal skills, traits, and hobbies. The professional skills are associated with a level of knowledge. The proficiency is based on the office of HR at the National Institutes of Health (NIH) and has five levels of knowledge, which range from novice to expert.

The edit view, depicted in figure 4.2 has the same structure as the add view with the addition that we load the candidate's data we want to edit from the database and display it. Information can be edited by hovering over the entries, clicking the edit icon, and then saving the changes. An example of how the editing works is shown in figure 4.3.

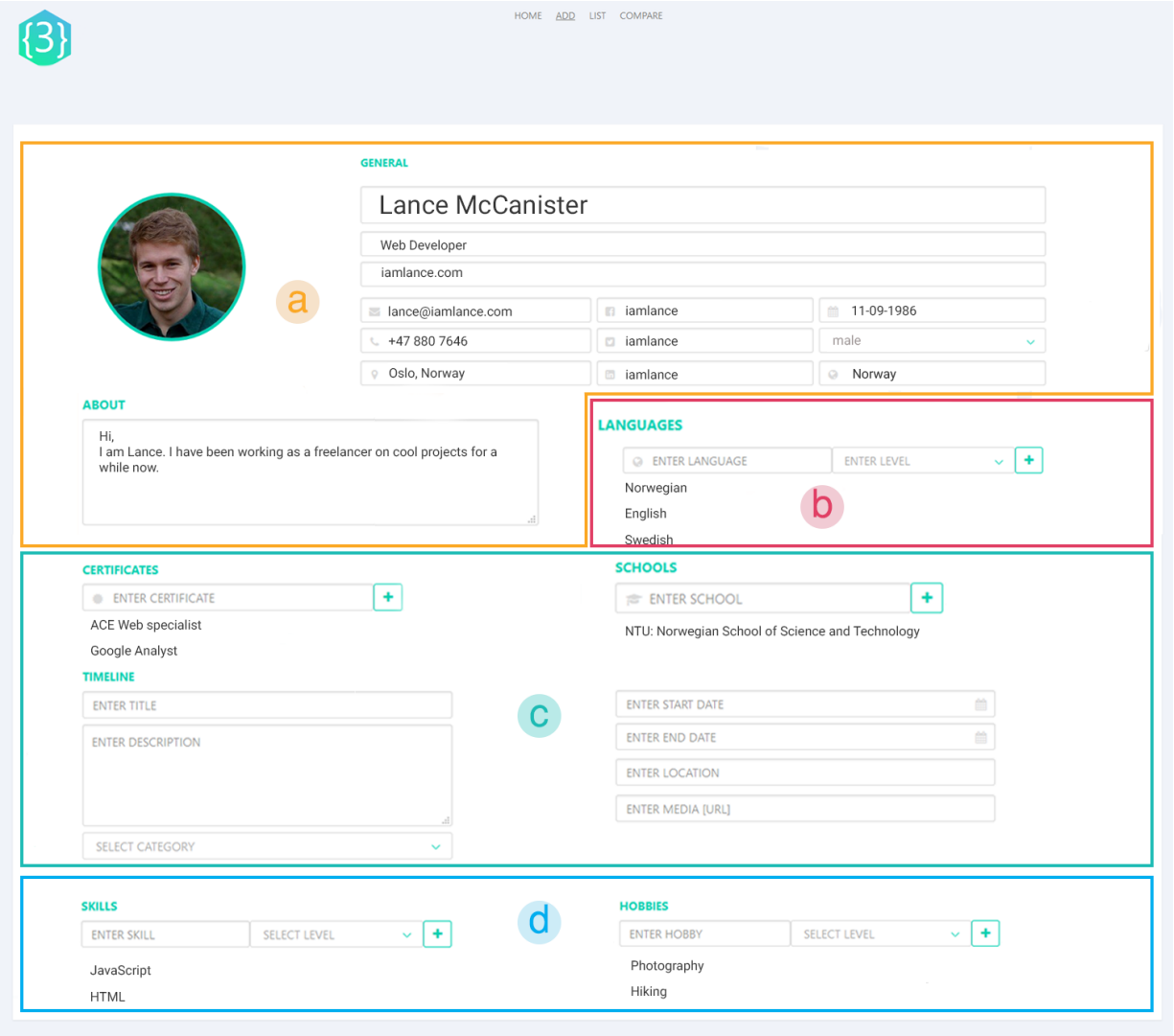


Figure 4.2: Edit view with sections for a) general and personal information, b) languages and proficiency, c) a timeline to enter work and/or education events, and d) professional and other competencies.

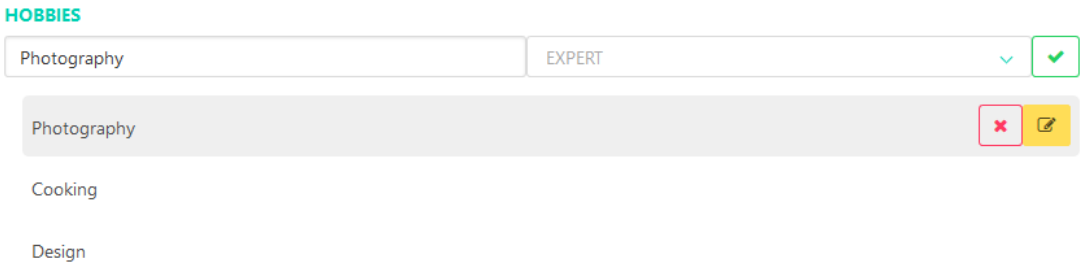


Figure 4.3: In the edit view entries can be edited by hovering over them and pressing the pencil icon. The changes are saved by clicking the green check mark.

Detail view of resumes

The detailed view of an applicant illustrated in figure 4.4 displays the visual resume of a single candidate. This view is divided into multiple components, using specific visualization techniques for the different types of data. In the following we describe the different views and their associated visualizations.

(A) General and personal information

In view A we have opted for a structured tabular view of the candidates general information, which includes attributes such as name, occupation, date of birth, social networks, and contact information. The information in this view is purely textual and should provide a clear overview of the candidate's information without the overuse of IV techniques.

(B) Languages and proficiency

View B provides a visualization of the candidate's languages and proficiency. We use a doughnut chart visualization to encode the information. The name of the language is displayed on the bottom and the proficiency is displayed in the center of the doughnut. The surrounding circle uses varying opacity to visually encode the proficiency as well - a full opacity circle is the highest proficiency possible (C2), whereas other levels of knowledge have smaller opaque areas.

(C) Timeline for work and education histories

View C offers inputs for certifications, honors, awards, and details on a candidate's completed education. Furthermore, this view provides a timeline visualization of the work and education histories of an applicant. The timeline view is composed of two parts - the top part is a slider displaying meta-information associated with a selected event, including start and end dates, title, description, and media (in this case a picture), the bottom

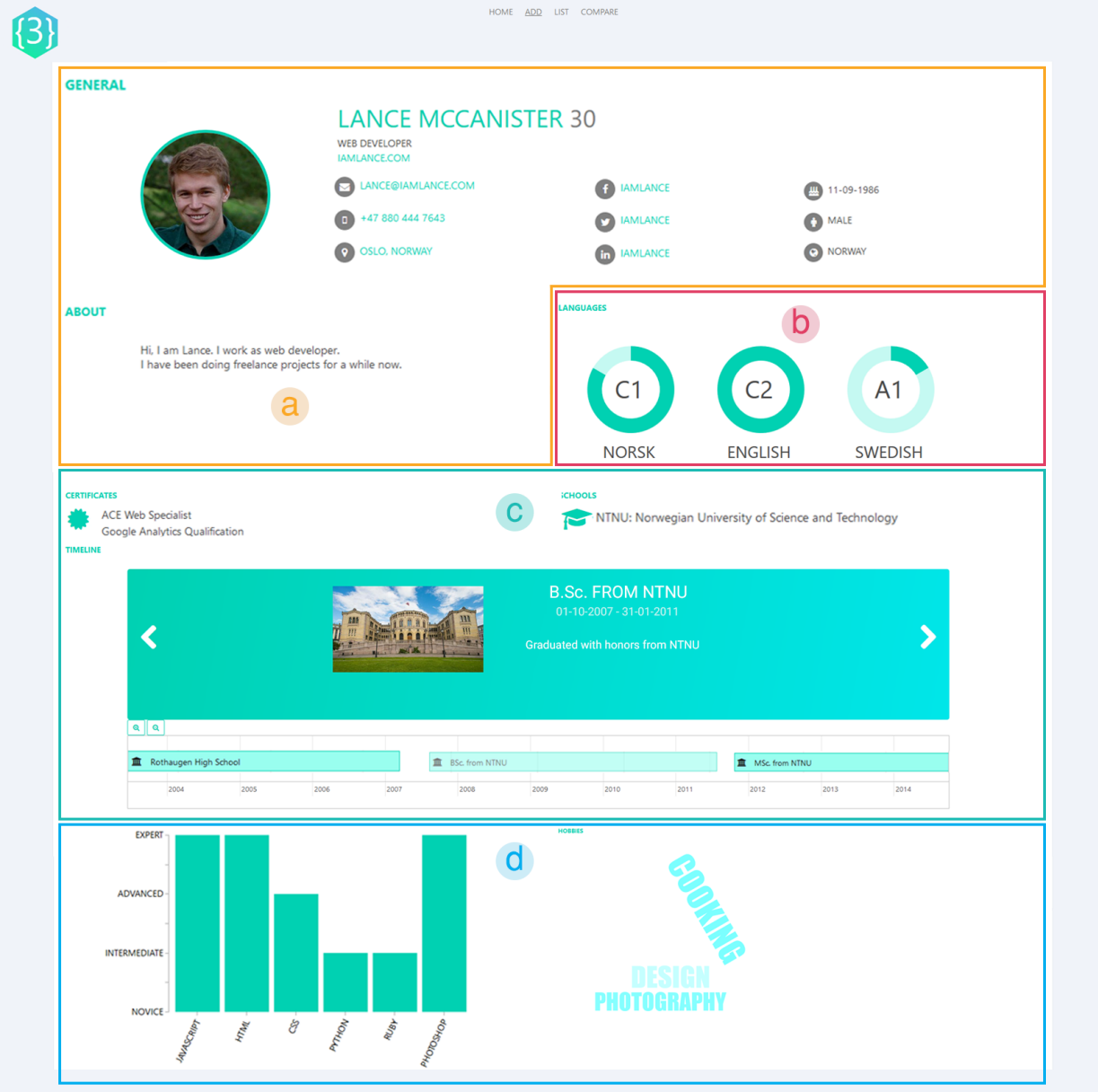


Figure 4.4: View the details of existing resumes consisting of multiple components with different visualization techniques depending on the type of data. a) general and personal information, b) languages and proficiency, c) a timeline for work and/or education events, and d) professional and other competencies.

part is the timeline view, which represents events as bars along the horizontal (time) axis and their length depicts the duration of the event. We additionally use symbols to visually encode the category of each event.

(D) Professional and other competencies

In view D we use two different visualization techniques for professional competencies and other skills or hobbies. As professional skills are typically associated with proficiency, we visualize these in a bar graph where the height of the bars corresponds to the level of knowledge. For the other skills or hobbies view we have opted to visualize the data there by using a word cloud technique. The size and orientation of the words are generated at random, whereas the colors and their shades are chosen so that they are consistent with the design of the rest of the application.

List view of resumes

The list view, depicted in figure 4.5, provides users with information about all applicants entered in the database and displays details, including a profile picture, occupation, location, certifications, awards, and completed education.

The view consists of a main area, which illustrates an overview of each candidate, a quick list containing the subset of applicants to be compared, a search bar offering full text search and advanced search features along with a set of functions to add, edit, view, or delete candidates. In the following we will describe the different elements of the list view and their features in more detail.

(A) Search bar

The search bar illustrated in detail in figure 4.6 provides the users with the ability to filter candidates using full-text or advanced search features. The full-text search will return any results, where the query matches one or more of the attributes from the candidate's profile. We have additionally implemented an advanced search feature which allows users to filter the results based on the applicant's occupation, location, and skillset. For example a user could use the advanced search to gather results on all candidates, whose occupation is a programmer with C/C++ experience, living in Paris.

(B) Functions

The buttons depicted in figure 4.7 provide users with the functionality to add candidates to a quick list for comparison, edit their information, view a detailed profile, or delete them from the database. Once added to the quick list the respective candidate is highlighted in the list view and the add button is replaced by a remove one, allowing users to remove them from the quick list.

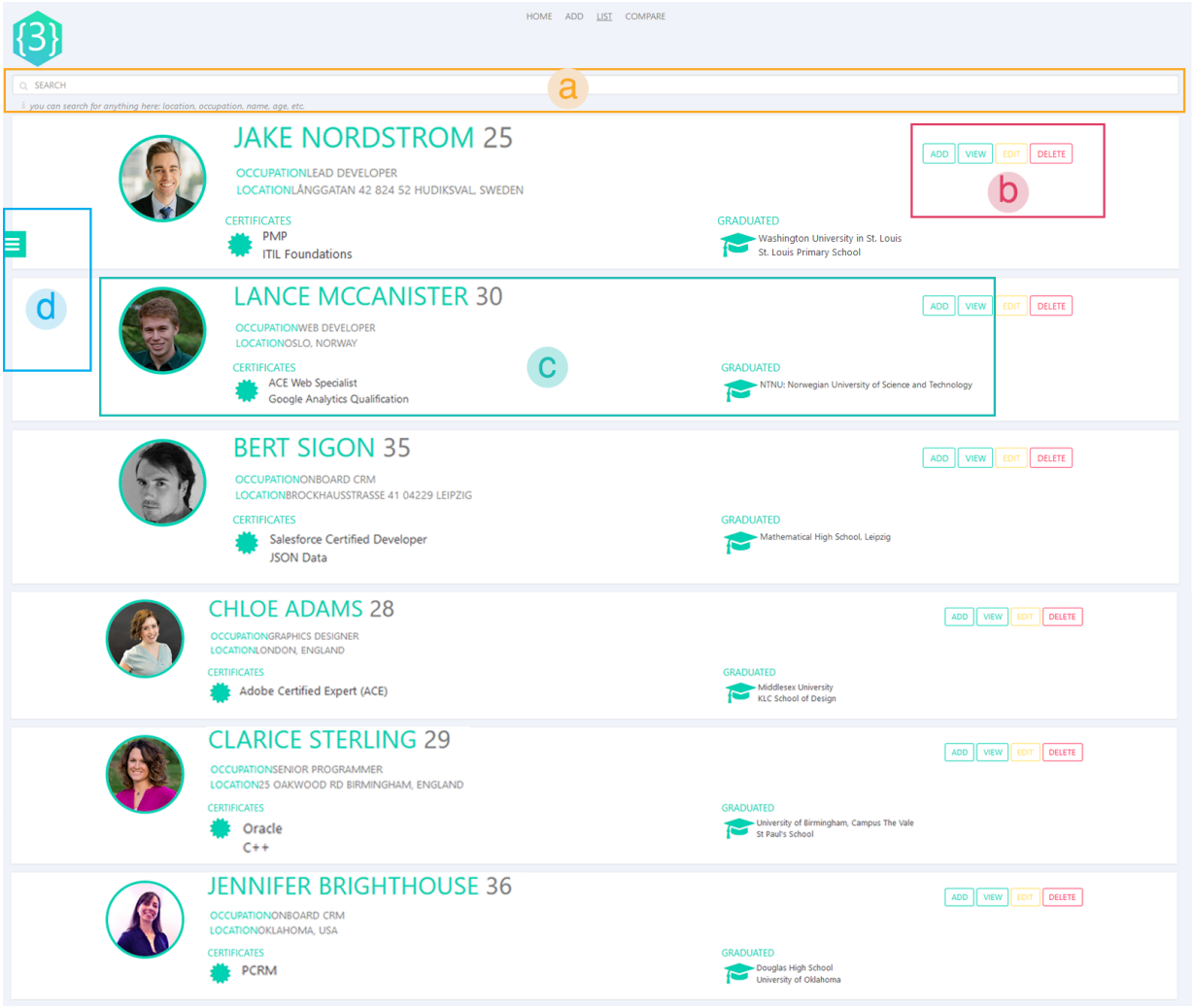


Figure 4.5: List view for all resumes in the database. a) A search bar, b) Functions to compare, view or edit entries, c) Overview of each candidate, d) Quick list.

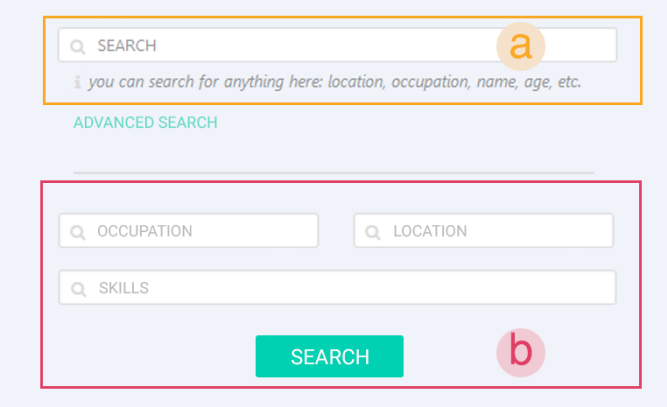


Figure 4.6: Search bar with a) full-text search and b) advanced search.

Figure 4.7: Add, View, Edit and Delete functions.

(C) Candidate overview

In the list each entry provides a general overview of the candidate illustrated in figure 4.8. We chose to display information regarding the candidate's name, age, occupation, location, certifications, awards, and completed education as a result of interviewing hiring managers and employees in the HR department. This information is considered important and therefore instead of providing a simple list with the applicant's names we have opted to display these additional attributes in the list view.

Figure 4.8: Overview of a candidate in the list view.

(D) Quick list

The quick list, illustrated in figure 4.9, provides a means of displaying the selected candidates in cases where there might be a large amount of results present in the list

view. The selected applicants are also highlighted in the main area displaying the results. The reason for this is so that users can visually distinguish between selected and not selected candidates.

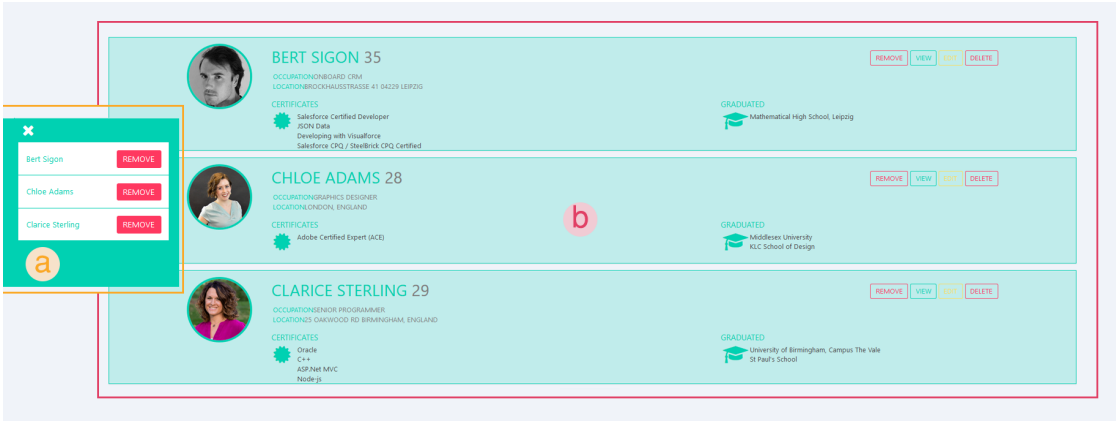


Figure 4.9: Quick list. a) List with selected candidates, b) Selected candidates are highlighted in the list view.

We have implemented the quick list feature so users can have a quick and clear overview of the selected candidates without the need to scroll through the large list of results. The quick list is located on the left side of the application, can be opened or closed on mouse click events and provides an intuitive way to remove candidates as well.

Comparative view of resumes

In this section we will describe the main part of our proposed solution - the compare view, illustrated in figure 4.10. We use consistent color-coding throughout this view to allow candidates to be easily and intuitively distinguished from each other. The compare view is composed of multiple components that we will describe in the following.

(A) Overview

The overview, illustrated in figure 4.11, displays general information about each candidate, including their name, age, years of experience working, number of languages known, and the last position they have held in a company.

We have visualized this information using cards, which are a convenient method of displaying content composed of different objects in the context of web applications. Users can remove candidates from the compare view by clicking the close icon in the top right of each card. Details on languages, such as the name of the language and proficiency are provided on mouseover effects - this can be seen in the following figure 4.12.

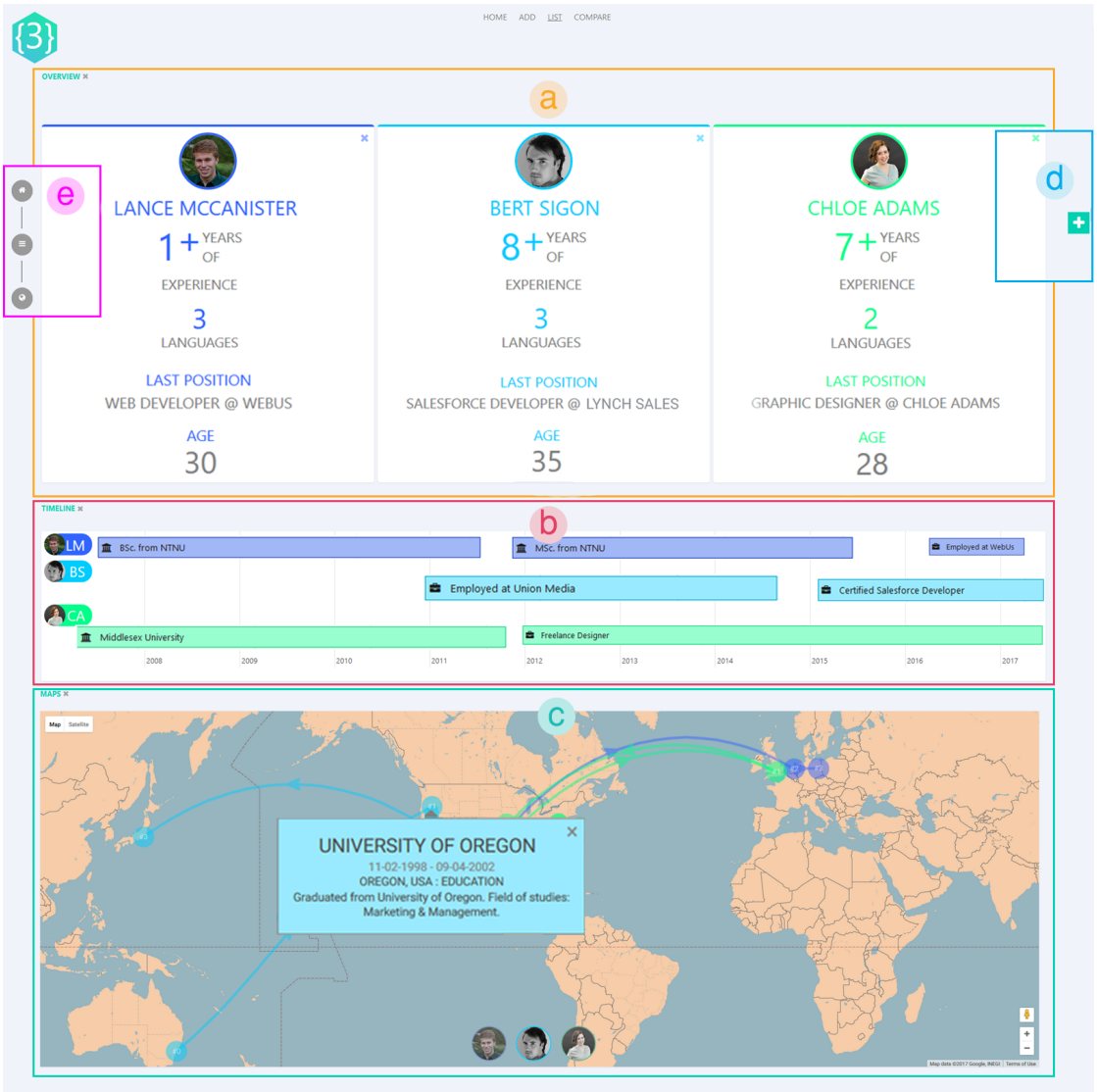


Figure 4.10: Comparative view for selected resumes. a) Gives an overview of general information for each candidate, b) provides a common timeline visualization, c) provides a spatial visualization of the events, d) offers users the possibility of interactively adding more candidates, e) is the quick navigation.

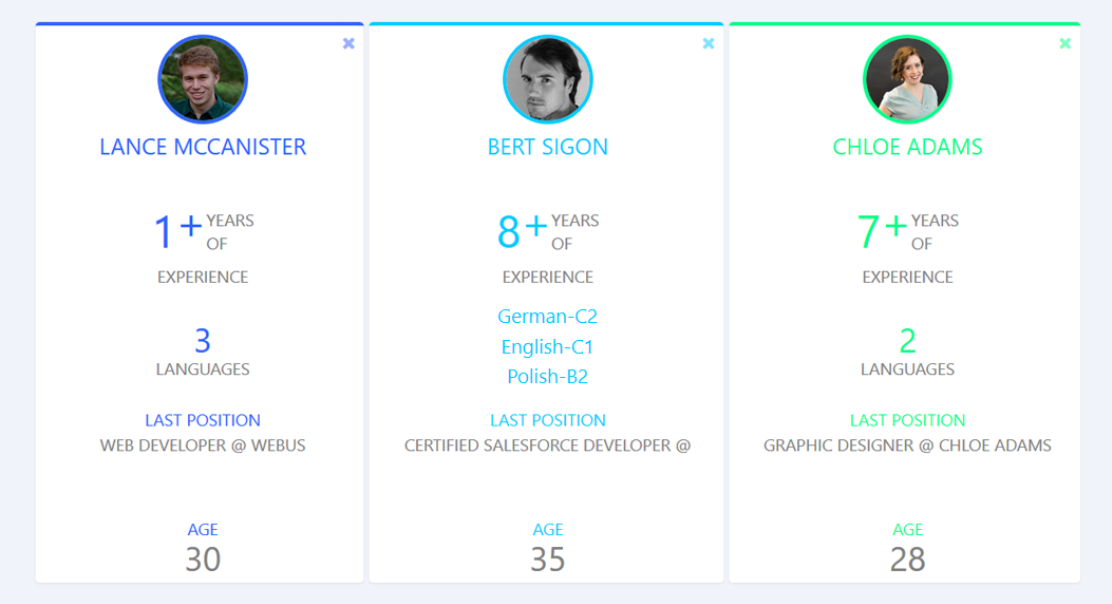


Figure 4.11: Overview displaying general information about the candidates. Languages can be seen on mouse over effects.

Figure 4.12: Languages can be seen on mouse over effects.

(B) Timeline

In view B we have implemented a timeline visualization, which is depicted in figure 4.13. We map all events of the selected candidates to a common timeline representation and visually encode the category of each event by using symbols. Each candidate's timeline is color-coded and on the left side we display their profile picture along with the initials of the their name. On mouseover we display the details associated with each event. Such details include the event's title, a short description, location, start and end date, and optional media. An example of the event's details this is illustrated in figure 4.14.

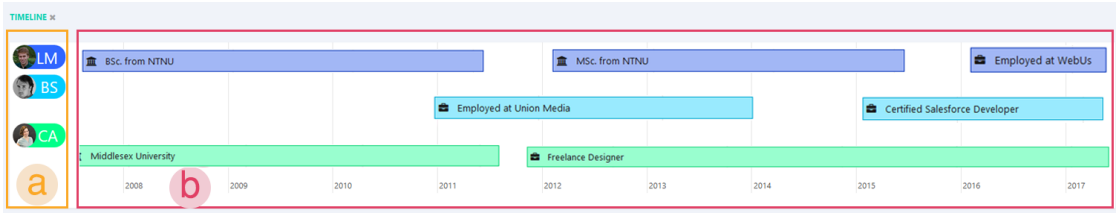


Figure 4.13: Timeline - Candidates are color-coded and events are displayed as bars, where the length depicts their duration. a) Legend with candidates and their initials, b) The main area of the timeline.

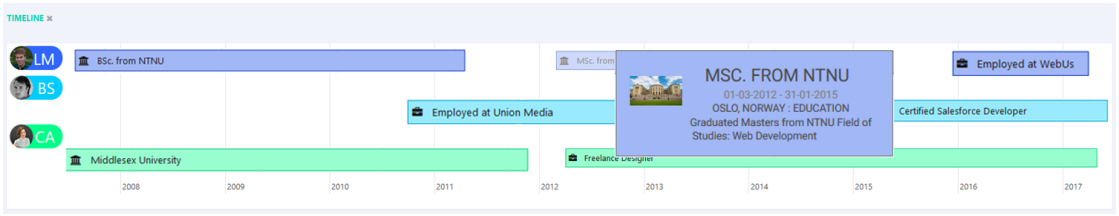


Figure 4.14: Timeline - Event details and meta-information are displayed upon interaction via mouse click or hover effects.

We have also extended the timeline to allow users to quickly filter events that have occurred in a specific location or have a specific category associated with them. This visualization technique is interactive, offers event details on demand, and is an intuitive representation of multiple candidate's chronologies.

(C) Maps

The maps view offers visualization of the spatial information associated with each event. We visualize each event as a marker, which is color-coded, and plot them on the map as illustrated in figure 4.15. The events of a single candidate are connected by lines, which display an arrow depicting the direction of movement throughout time. The markers react to mouse click events and display details associated with the corresponding event upon interaction. We also display a legend in lower part of the map, showing which

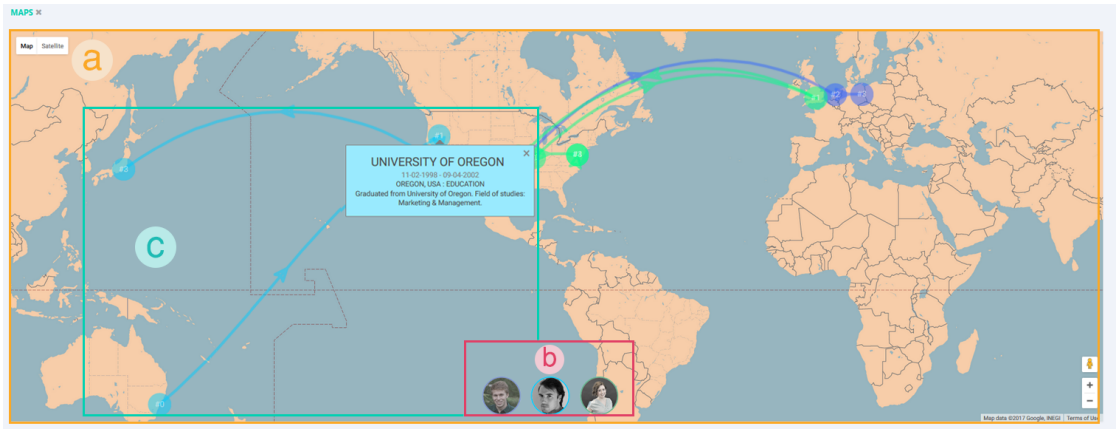


Figure 4.15: Events are plotted and connected on the map. a) Main map area, b) Legend with candidates, c) Individual path with events of a candidate.

candidate's chronology is currently visualized. Clicking on any of the candidates in the legend would highlight their path, while displaying other candidates at lower opacity.

The maps and timeline views are linked views, meaning that clicking an event in either view would in turn highlight the corresponding event in the other view as well. This interaction is useful for users so that they can gain insights on when and where an event has occurred. A detailed view of a single applicants' chronology plotted on the map is illustrated in figure 4.16.

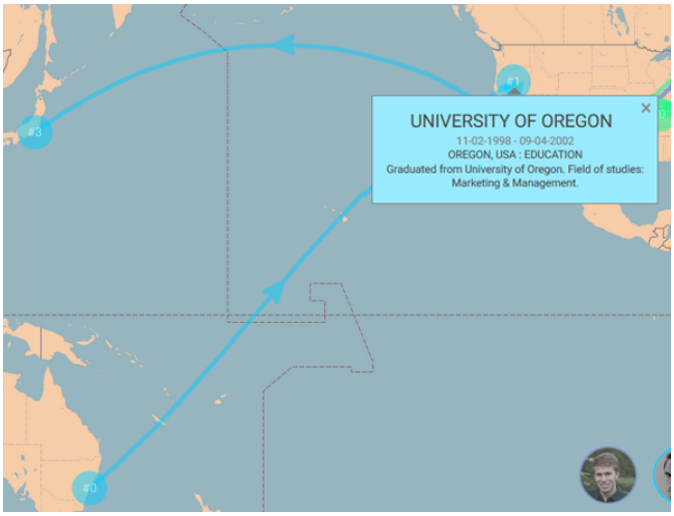


Figure 4.16: Detail view of a candidates path and event details on mouse click interaction.

(D) Quick add

The quick add feature offers users the option to add more candidates to the compare view without having to navigate back and forth between views. The quick add is located on the right side of the application, depicted in figure 4.17 and is opened by clicking on the button with the plus sign on it - this will expand into a list of candidates that can be interactively added. Once a new candidate is added to the compare view - the overview, timeline, and maps views of the application are updated. The quick add feature also allows for the search of specific candidates and displays information regarding each candidate's name, location, and occupation along with a profile picture.

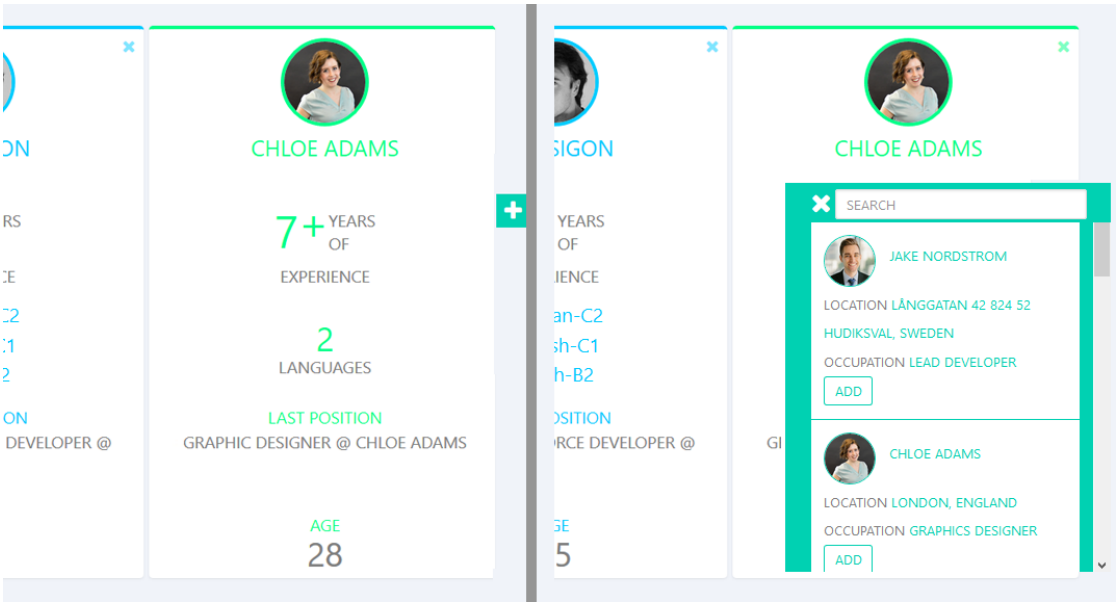


Figure 4.17: Quick add list can be opened on mouse click. On the left the button to toggle the list is displayed. On the right the quick add list is displayed.

4.2 Technical details

In this section we will describe the main frameworks used in our solution and the technical details of the application as well as go over the server-client communication.

4.2.1 Frameworks

MEAN

In our solution we use the MEAN stack, which is composed of the following components:

- MongoDB
- ExpressJS
- AngularJS
- NodeJS

In the following sections we will go over the aforementioned components in more detail, explaining their functionality, and how we use them in our implementation.

MongoDB

MongoDB is a free and open-source cross-platform document-oriented database and can be downloaded for free from `https://mongodb.com/`. It is classified as a NoSQL database and uses documents with schemas that are similar to the JavaScript Object Notation (JSON) file format. We have chosen this database, because the document structure is easily extendable, matches the resume format extremely well, and the database offers seamless integration with the other frameworks used in our solution. In our implementation we use MongoDB for Create, Read, Update, and Delete (CRUD) operations and managing resumes.

ExpressJS

ExpressJS is a free and open-source web application framework for NodeJS. It is designed for building web applications, Application programming interfaces (API), and is the de facto standard server framework for NodeJS. ExpressJS can be downloaded for free from `https://expressjs.com/`. In our solution we use ExpressJS to manage Hypertext Transfer Protocol (HTTP) requests between the client and server, as well as manage the database and file upload.

AngularJS

AngularJS is a JavaScript-based open-source front-end web application framework and can be downloaded for free from `https://angularjs.org/`. Angular utilizes the

Model-View-Controller (MVC) pattern, which provides a clear separation of the data, UI, and interaction components. In our solution we use Angular to implement a Single Page Application (SPA), which provides a similar user experience to that of a desktop application. All necessary resources are loaded once at the start of the application and the contents are updated as the user interacts with the application, which provides a more fluid and responsive user experience than traditional web applications do.

NodeJS

NodeJS is an open-source, cross-platform JavaScript run-time environment for executing JavaScript code server-side. The NodeJS library is available for download from `https://nodejs.org`. NodeJS allows for rapid development and creation of web servers. In our solution we use Node to manage our server-side and client-side dependencies, as well as create and configure a web server, which handles HTTP requests and connections to the database.

Other notable frameworks

VisJS

VisJS is a dynamic browser-based visualization library and is available for download from `http://visjs.org/`. This framework allows for the handling of large amounts of dynamic data, enables the manipulation, and interaction with the data. We use this library in our solution to manage, visualize, and represent the time-oriented data present in work and education histories of resumes.

D3JS

D3 stands for Data-Driven Documents and is a JavaScript library for manipulating documents based on data. The library can be downloaded from `https://d3js.org/`. D3JS utilizes the capabilities of web standards such as HyperText Markup Language (HTML), Scalable Vector Graphics (SVG), and Cascading Style Sheets (CSS) to provide powerful visualizations of data. We use this framework to create visualizations of ordinal and nominal data, such as the professional skills and languages sections of a resume.

Bulma

Bulma is an open-source modern CSS framework that is based on Flexbox. The Flexbox Layout is a new layout module in CSS version 3 made to improve the items align, directions, and order in the container even when they are with dynamic or even unknown size. We use Bulma in our solution to configure and design the layout of the application, the interaction elements, and components.

4.2.2 Architecture

In this section we will describe the different components of our solution. We will discuss in detail how it is that they load, process, and visualize the data. In general we use a client-server architecture, where the server is capable of supporting multiple clients and handling their requests. A typical client-server communication model is depicted in figure 4.18.

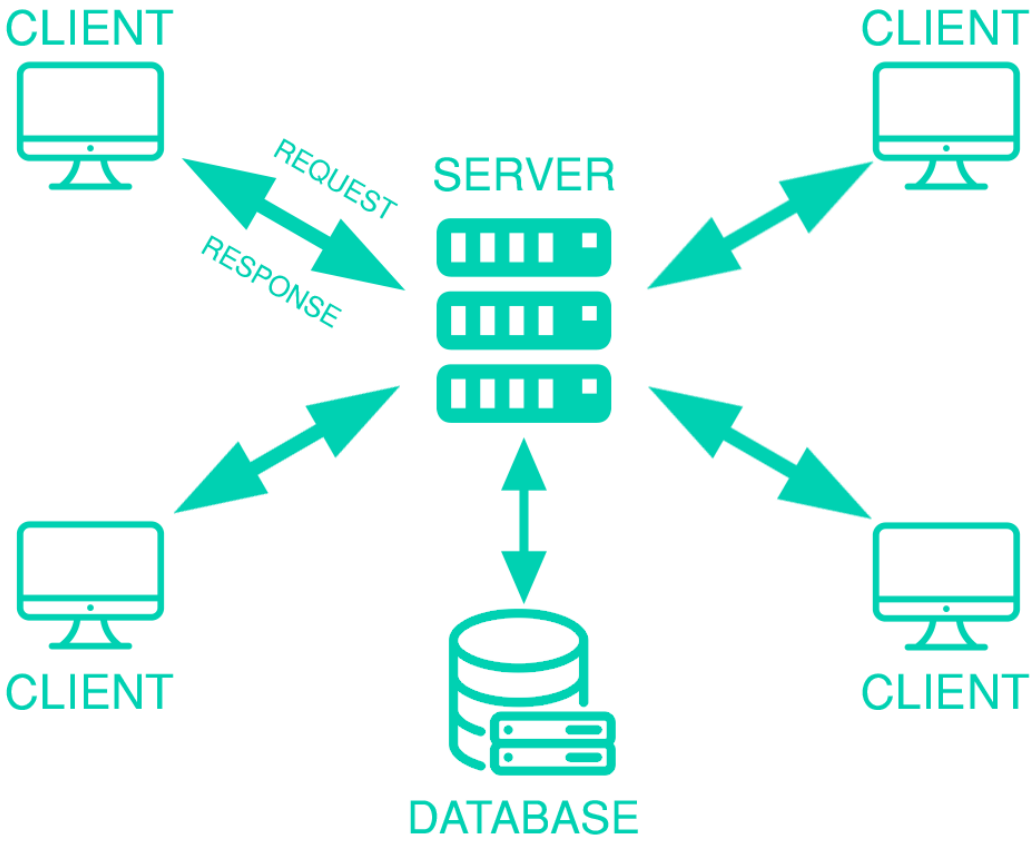


Figure 4.18: Example of a client-server communication model.

The server accepts incoming requests from the clients, queries the database for results and returns those results in a response to the clients. In the following sections we will describe in detail the implementation of the server and client components.

Server

The server is responsible for managing the database and returning results depending on the client's request. In the following we will describe the server's API.

The server's directory structure is illustrated in figure 4.19 and highlights the different assets, components, models, and controllers that it manages.

Figure 4.19: Server directory structure.

- In the `/assets` sub-directory we store image resources (such as e.g., icons) that are used throughout the application.
- Our controllers managing creating, reading, updating, and deleting resumes from the database are stored in `/controllers`.
- The schema definition of a resume is located in `/models`.
- The application's routing is defined in `/routes`.
- In the `/uploads` directory we store files and attachments, such as profile pictures.
- The different views that our implementation provides are organized in the `/views` sub-directory.
- The files `bower.json` and `package.json` files contain the required dependencies and are managed from NodeJS.
- The `server.js` file is the main file responsible for starting and stopping the server.

In the following subsections we will explain the HTTP requests used, list the API calls that the server provides, and describe what they are responsible for.

- `GET`

  - The `GET` method is used to retrieve information from the server using a given Uniform Resource Identifier (URI). Requests using `GET` should only retrieve data and should have no other effect on the data.

- `POST`
  - A `POST` request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.

- `PUT`
  - `PUT` replaces all the current representations of the target resource with the uploaded content.

- `DELETE`
  - `DELETE` removes all the current representations of the target resource given by URI.

GET /

This API call performs a `GET` request on the base directory `/` and returns the index page of the application.

GET /:name

Upon navigating to any of the sub-pages of the application we perform a `GET` request on a partial file that will be injected into the main index file. For example when navigating to the list view the client sends `list` as the `:name` parameter in the URI. The server processes the request and renders the list view which is located in the `/views/partials` directory.

GET /api/cvs

A `GET` request to the `/api/cvs` URI will return a full list of the resumes present in the database. The resumes are encoded in JSON.

GET /api/cvs/:id

A `GET` request to the `/api/cvs/:id` URI will return a single result, which is the resume that matches the ID requested from the client.

POST /api/cvs

A `POST` request to the `/api/cvs` URI will create a new resume and save it in the database with a new ID generated from the database manager.

DELETE /api/cvs/:id

A `DELETE` request to the `/api/cvs/:id` URI will query the database for a resume matching the ID parameter passed from the client and remove it from the database. This action is irreversible.

PUT /api/cvs/:id

A `PUT` request to the `/api/cvs/:id` URI will query the database for a resume matching the ID parameter passed from the client and update the contents of that resume with the new information that the has been sent to the server.

GET /api/db/count

A `GET` request to the `/api/db/count` URI will return the number of resumes currently present in the database.

POST /api/upload

A `POST` request to the `/api/upload` URI is used to upload files and attachments to the `/uploads` directory. Such files can be for example a profile picture.

Client

The client is responsible for managing, parsing, processing, and visualizing the data it receives from the server. The client communicates user interactions with the server and manages the UI of our implementation. In the following section we will describe the client's directory structure, depicted in figure 4.20.

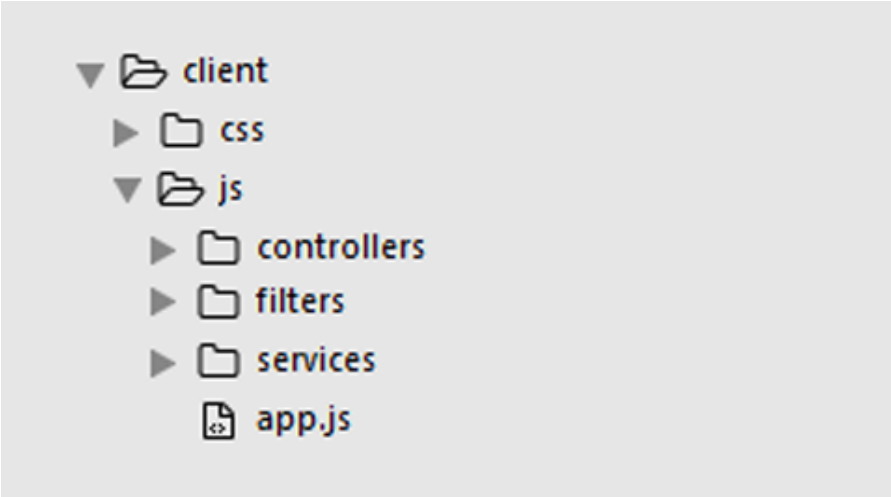


Figure 4.20: Client directory structure.

- In the `/css` sub-directory we store files that contain rules for the design, look and feel of the application.
- The `/js` sub-directory is where JavaScript files, including controllers, filters, services, and the main configuration file of our proposed solution are located.
- The controllers that manage the data and interactions of the different views are located in the `/js/controllers` sub-directory.
- Filters that are responsible for the search and advanced search features of the application can be found in the `/js/filters` sub-directory.
- The `/js/services` sub-directory contains functions that we use throughout the controllers. Instead of having redundant code in the controllers, we have compiled a set of reusable functions that parse and process data and handle the file upload.
- The `app.js` file is the main file for managing and configuring the client. It defines global variables, injects dependencies, and configures the client-side application routing.

The client offers five controllers that are responsible for the detail, edit, create, list, and compare views respectively. In the following subsections we will describe the different controllers in more detail.

View Controller

The view controller defines a list of functions responsible for the loading, parsing, processing, and visualizing of resume information for a single candidate. Upon receiving the JSON encoded resume from the server, we initialize the different components of the detail view and utilize features from the VisJS and D3JS libraries for visualizing the language, timeline, professional, and other skills views. The view controller is also responsible for managing the user interactions defined by the UI elements.

Edit Controller

The edit controller loads a resume into the edit view of the application. Users can then proceed to edit the different sections of the resume by updating or deleting existing information or by adding new data. The edit controller makes requests to the server in order to receive the data requested by the client and to save the modified data back to the database.

Create Controller

The create controller is responsible for creating a new resume entry in the database. This controller defines the interaction of UI elements, validation, parsing, and processing of information that is entered by the user when creating a resume. The controller uses

services defined in the `/js/services` sub-directory for uploading and parsing the data. Upon creation of a new resume the user can then navigate to the detail or edit view.

List Controller

The list controller performs a request to the server and receives a set of all resumes present in the database. This controller is responsible for parsing, processing, and visualizing that information in the list view and provides a set of functions for the detailed viewing, editing, or deleting of resumes. The list controller further offers users features to add a subset of those resumes to a secondary list with the primary objective of comparing those candidates.

Compare Controller

The compare controller receives a set of resumes that will be compared. A color scheme is defined in the controller, which applies a unique color for each candidate - this results in consistent color-coding of the applicants throughout the compare view, which makes them easily distinguishable from each other. The controller is further responsible for computing the years of experience for each candidate, visualizing the overview, common timeline, and plotting each candidate's events on the map view. The compare controller also defines a set of functions that allow the users to interactively remove or add resumes to the compare view. This results in a unique interactive experience, which supports users in the task of visually exploring multiple instances of resumes.

4.3 Summary

In this chapter we have introduced our proposed solution - CV3 - for visualizing, exploring, and comparing multiple resumes with a focus on temporal and spatial data. We have described in detail the different views, user interaction elements, and components that our application provides, as well as listed and described the dependencies used. We have also explained how the client and server communicate, the different types of requests, and the responsibilities of the different controllers. In the next chapter we will evaluate our proposed solution by conducting a small scale user study.

CHAPTER 5

Evaluation

In this chapter we will describe our evaluation of our proposed solution, where we will gather information from user surveys and observations that will serve as evidence to prove the hypotheses of this thesis.

> "Can VA and IV support the visual exploration and comparison of chronological events across several instances of resumes in a meaningful way?"

We will further discuss issues, that we have discovered from user feedback, and describe the proposed improvements to the application. This chapter is structured into three sections. In section 5.1 we will describe the evaluation methods that we have used. The results of the evaluation and user feedback are presented in section 5.2. Based on the feedback we have implemented improvements to our proposed solution, which are described in section 5.3.

5.1 Evaluation Method

For the evaluation of our proposed solution we have decided to perform a small scale user study. Before conducting the evaluation we have explained the purpose of the application, different components, and interaction elements to the users. We have abstracted the scope of our user study to solely evaluate the proposed solution in terms of IV and VA techniques for the purpose of providing an interactive experience to explore and analyze multiple instances of resumes.

For our user study we had six participants, half of which are expert users, those with previous experience in comparing and analyzing resumes and the other half are non-expert users. The expert users have previously worked with ATS's and have experience working in the HR department. All of the participants had no previous knowledge or experience

with the visualization techniques used in our solution, they did however have experience with using web and mobile applications and were accustomed to the interactions of the different components. The steps taken in the evaluation process are as followed:

- We explained the users the aim and scope of this thesis and project.
- We prepared the environment for the evaluation.
- The users were allowed a short time to get accustomed to the application's navigation, layout, and functionality.
- We asked the users questions regarding the visualization techniques, comparison of multiple resumes, and interactivity of the solution.
- We asked users to give their honest feedback regarding the visualization techniques used, the application's layout, and functionality. The results are highlighted in section 5.2.
- Feedback is used as a basis of improvements for the prototype. The improvements and changes are described in section 5.3.

Initially the users were given a short introduction into IV and VA. We provided the users with an overview of the thesis and motivation for the prototype, the prototype's scope, and the aim of our work.

The next step in the evaluation process is to prepare the prototype and environment for the user study. We have done this by hosting the prototype and the database online to imitate the conditions of a real application, including latency, loading, and error handling. For the evaluation setup we used one computer, as a client, connected to a 24" display. The computer has the following specifications:

- OS: Windows 10 64bit
- CPU: i7-6700 @ 3.40 GHz
- GPU: Nvidia GTX 970
- RAM: 16 GB
- Browser: Firefox 53.0.3

The client provides controlled environment settings and the user's actions are observed by the evaluators. In our evaluation we have opted to use the Concurrent Think Aloud method, introduced by Clayton Lewis [Lew82]. The think aloud method requires users to voice their thoughts as they interact with the prototype. The goal of this is to encourage participants to keep a running stream of consciousness as they solve the given set of

tasks. Evaluators gain insights on how the users interact with the application through the use of this method.

For the evaluation we have compiled a list of tasks that the users must complete. These tasks were thought of as typical tasks hiring managers usually have when comparing candidates. The tasks are described in table 5.1 and are associated with filtering, comparing, and analyzing candidates.

Task	Description
#1	Find all candidates in a London, United Kingdom with experience in HTML
#2	Find out how long it took for CLARICE STERLING to finish her education
#3	Add five candidates to the compare view
#4	Find where RACHEL SMITH completed her education
#5	How long has BERT SIGON spent in a the Leipzig, Germany?
#6	Add GORDON REED and CLARICE STERLING to the compare view. Remove GORDON REED from compare view and add LANCE MCCANISTER and JENNIFER BRIGHTHOUSE
#7	What is the difference in experience between BERT SIGON and CLARICE STERLING?
#8	How many countries has RACHEL SMITH been in?
#9	How many languages do BERT SIGON and LANCE MCCANISTER know (which and how well)?

Table 5.1: User tasks for the evaluation of the prototype

For the test data, we have populated the database with multiple personas, each having a different set of attributes. These personas are used for testing, evaluating, and demo purposes only.

As test metrics for the evaluation of the prototype we observe the following:

- Successful task completion

  - A task is successfully completed by the participant.
- Unsuccessful task completion
  - The participant could not complete the task.
- Time spent on task
  - The time that the participant has spent on the task. This has been measured by using a stop watch.
- Likes, dislikes, and recommendations
  - Participants provide their honest feedback, including what they liked and disliked about the prototype.

5.2 Evaluation Results

In this section we will describe the results of the tasks from the user study and discuss general feedback gained from the participants of the evaluation. The full results of participant's tasks can be found in the Appendix section A.2

5.2.1 Task Results

The results from the user study are depicted in the following table 5.2. As can be seen from the results most tasks could be executed quickly and successfully. Some users had difficulty in executing tasks 2 and 5, related to where an event has occurred and finding how long a candidate has spent in a given location. This is due to the fact that the participants have had no previous experience with the prototype, but after exploring the application for a bit they were able to successfully complete these tasks as well. During the process of the evaluation we discovered some usability issues, related to the layout and interactions, which have been resolved. The overall feedback from users was positive. The time required for executing tasks is averaged across all participants in the study. When comparing the recorded times with empirical data, gained from interviews and surveys with the participants, we see clear evidence that the time required for the successful completion of tasks is considerably lower than if they were to extract the information from a standard CV.

5.2.2 User Feedback

As previously mentioned the overall feedback from users regarding the design, layout, and interactivity of our proposed solution was positive. The users could quickly extract relevant and important information from the IV techniques used and could navigate in the prototype with ease. The use of the visualizations in the detailed and compare view have been perceived as intuitive, easily comprehensible, and visually appealing to the users. Based on the results presented in section 5.2 we can state that the prototype

Task	Successful	Unsuccessful	Time (average in minutes)
#1	6	-	0.7
#2	3	3	2
#3	6	-	0.4
#4	6	-	0.4
#5	3	3	1.9
#6	6	-	0.6
#7	6	-	0.5
#8	6	-	0.75
#9	6	-	0.6

Table 5.2: Results of the user study

provides a clean overview, with options to filter and display details on demand, and assists users in their tasks of recognizing and extracting information in a quick and efficient manner. We believe that our solution has succeeded in supporting Shneiderman's information seeking mantra and in proving the main hypothesis of this work, that the use of IV and VA techniques can support the visual exploration and comparison of multiple instances of resumes. We have received some additional feature requests and ideas for the improvement of certain components of the prototype. We have implemented changes based on this feedback and describe these in the following section 5.3.

5.3 Improvements

In this section we will describe the improvements and changes that we have implemented in the prototype based on the user's feedback.

5.3.1 Map markers

The first improvement is related to the marker plotting in the maps section of the compare view. The issue there is that at large zoom levels markers that are located close to each other overlap and are perceived as one marker. We have solved this issue by detecting overlapping markers and displaying them at different scales depending on the zoom level, as well as applying a small offset in both latitude and longitude. This change gives a

clear and intuitive overview about multiple events that occur in the same location and still manages to display the path of events upon zooming into a more detailed view. In figure 5.1 we can see the problem with overlapping markers and our proposed solution.

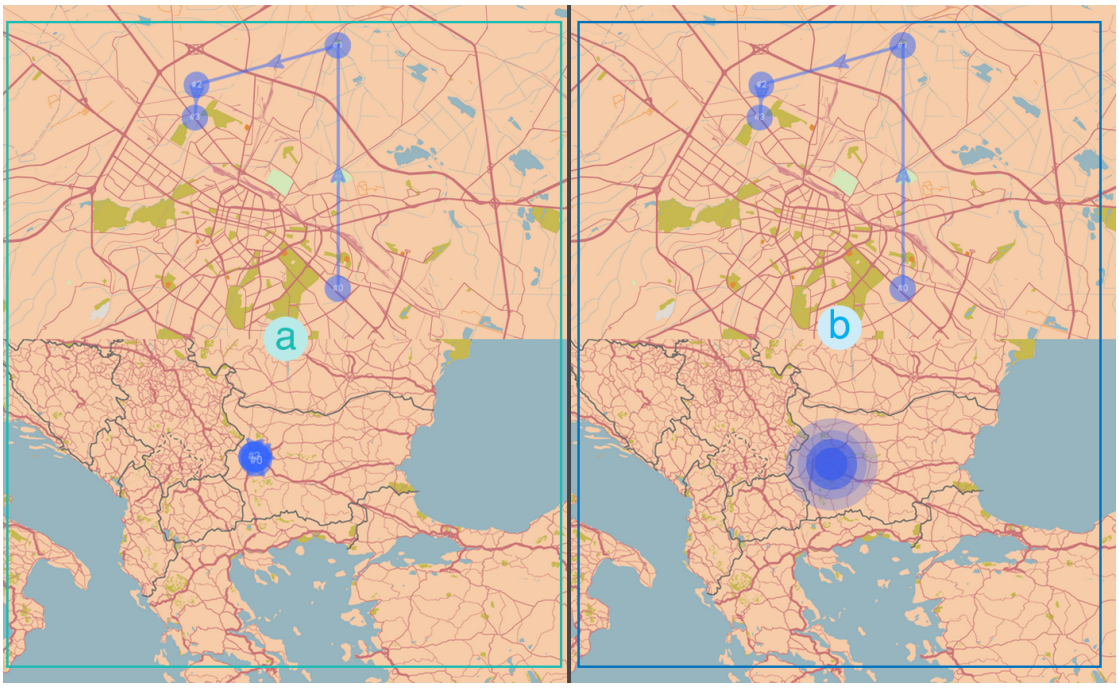


Figure 5.1: Improving overlapping markers. a) Problem: overlapping markers at varying zoom levels. b) Solution: markers visualized at different scale and opacity.

5.3.2 History

The second improvement we have implemented based on user feedback is a history list in the compare view. This list contains information related to candidates added and/or removed from the compare view by the user. Although this functionality is out of the scope of IV, we thought it provides a good overview of the user's interaction with the prototype and could assist her/him in exploring and analyzing multiple instance of resumes. The history list can be found on the right hand side of the application and is depicted in the following figure 5.2. The plus signs depict the candidates that have been added to the compare view and the minus signs are candidates that have been removed.

5.3.3 Filters

During the development and design of the prototype we have received feedback regarding the possibility of advanced filtering for the list view and the timeline representation of the compare view. We have implemented more complex filters allowing users the possibility to filter candidates based on their location, occupation, and skillset in the list view and

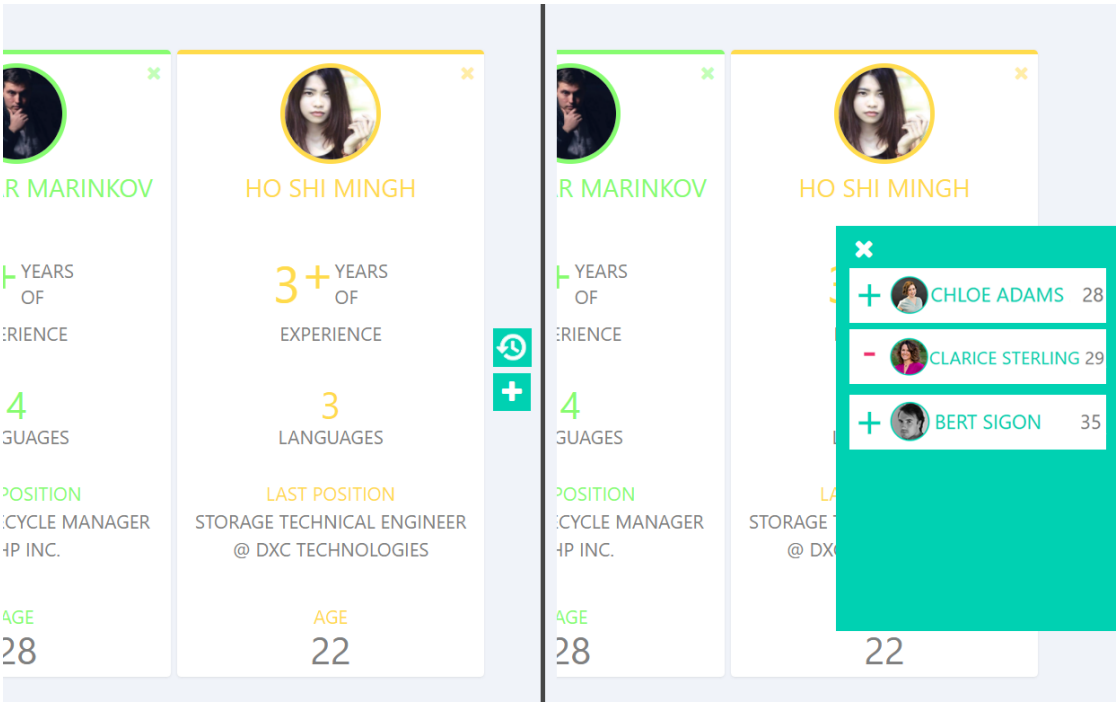


Figure 5.2: History list - displays a list of the users interactions with the prototype.

to filter events by their type and location in the timeline of the compare view. The filters are illustrated in figure 5.3 and have greatly improved the interactivity of the prototype and provide users with new insights about candidates by being able to explore and filter information quickly and intuitively.

5.3.4 Linked Views

The last improvement that has been implemented is providing linked views of the timeline and map components in the compare view. When a user selects an event in the timeline view the corresponding marker in the map view opens a dialog box with the event's details. The linked views have proven to be useful as they connect the temporal data present in the timeline representation and the corresponding spatial data in the map representation. An example of the linked views is depicted in figure 5.4. We believe that this additional improvement provides a clear connection between the timeline and map representation of the data and offers users another approach to explore multiple instances of resumes.

5.4 Summary

In this section we have described the individual steps taken in a small scale user study and presented the results as well as the user's feedback. We have outlined issues that have

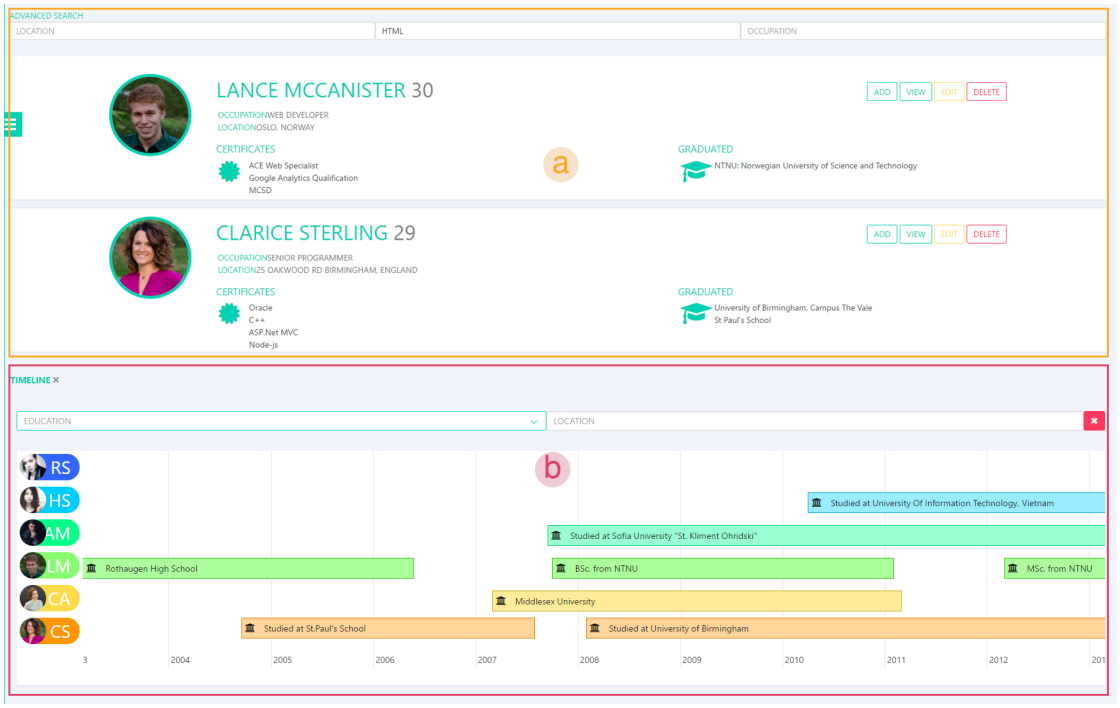


Figure 5.3: Improved filters. a) Advanced search filter for list view. b) Event filter for the timeline in the compare view.

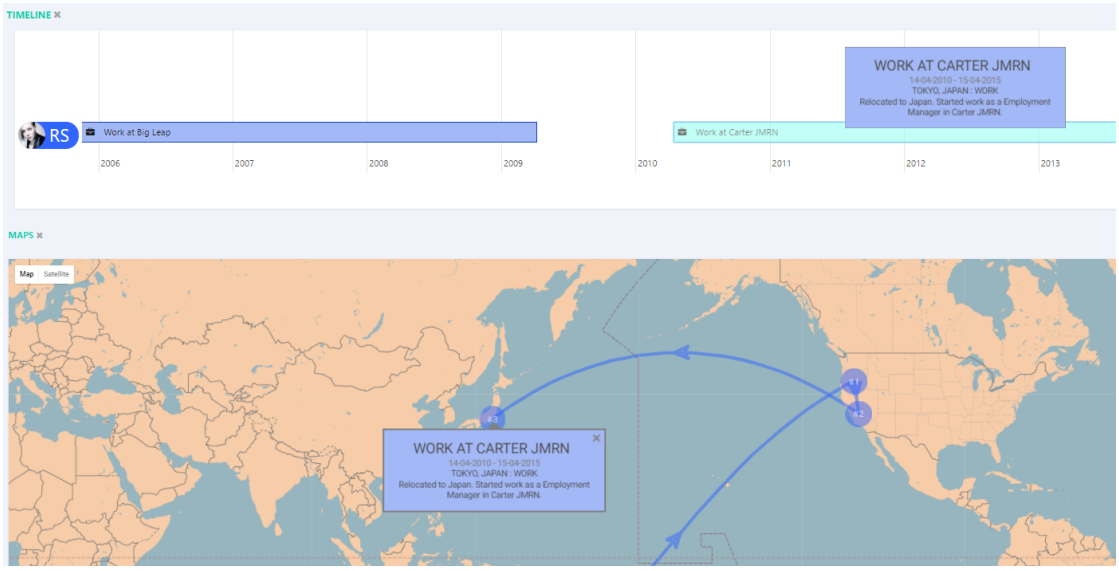


Figure 5.4: Linked views - the events from the timeline and map representation are linked together.

been discovered during the process of the evaluation and discussed the improvements to the prototype. The evaluation shows that using IV in the context of resume exploration and comparison does indeed provide users with a clear and intuitive overview of the individual candidates and highlights the individual applicant's strengths and weaknesses well. Providing users with an interactive environment to explore, analyze, and compare multiple instances of resumes improves the efficiency of the hiring process and selecting the right candidate for a given position. In conclusion we have proven that VA and IV does support the visual exploration and comparison of resumes.

CHAPTER 6

Conclusion

6.1 Summary

In this thesis we have discussed and analyzed state of the art visualization techniques for representing temporal and spatial data in the context of resume visualization. We have also described our approach to the design and development of a prototype that offers possible visualizations of that data and allows for an interactive experience to analyze, compare, and explore multiple instances of resumes.

Can VA and IV support the visual exploration and comparison of chronological events across several instances of resumes in a meaningful way?

To answer the main research question of this thesis, we have developed a prototype that utilizes state of the art visualization techniques in an interactive environment and have evaluated it by conducting a small scale user study. Based on the feedback from the user study and evaluation we have implemented improvements, that enhance the basic set of functions that our solution provides. The results of the evaluation show that IV and VA can indeed support the exploration and comparison of multiple instances of resumes and can assist users, that work either as hiring managers or in the HR department of a company, with their tasks in a meaningful way.

What kind of data do the users work with?

To obtain information about what data is present in resumes, we have performed a literary research and documentary analysis on standardized resumes and map that data to data types and attributes that are used in our solution. Information and research gathered in this step serves as a basis for the visualization techniques used in this project. The types of data in resumes are described in section 2.1.

What kind of tasks do users have?

What kind of results do users expect?

To gain insights on what type of tasks the users have, the general work process of hiring a candidate and what kind of results are expected, we have conducted interviews with people who have experience working in the HR department at varying positions. This step provides us with more detailed information on what type of data we will be visualizing and what type of tasks need to be executed. Helping us better understand what requirements the users have and what they expect from the approach leads to improved visualization techniques and features supporting the information discovery and analysis process.

How can we best visualize that data to support the user in her/his tasks?

How can the information seeking mantra of Shneiderman be supported?

To answer the questions of how we can support the users in her/his tasks and how we can best support the information seeking mantra of Shneiderman, we have performed exhaustive research in the field of IV related to time-oriented and resume data. The visualization techniques used provide a clear and intuitive overview of the data present in resumes and support the information seeking mantra of Shneiderman [Shn96] overview first, zoom and filter, then details on demand.

To the best of our knowledge as of writing this thesis there do not exist other products or applications that are capable of providing an interactive experience for visualizing, comparing, and analyzing multiple instances of resumes. Thus we believe this approach is a step towards new possibilities and IV techniques for the comparison of multiple instances of resumes.

6.2 Future work

The scope of our prototype, including the design, development, and features provided, is exclusively related to the use of IV and VA techniques that provide users with an interactive environment for exploring and comparing multiple instances of resumes. This provides a lot of possibilities for extending the set of features and integrating the prototype with other services in future work.

Numerous improvements and additional features, that are out of the scope of this thesis and are not a part of the design requirements, can be implemented in future work:

- Linking and integrating the prototype with professional social networking services such as LinkedIn and Xing
- Providing user management and functions for hiring managers to rate, comment on, or save lists of candidates.

- Improving visualization and interaction techniques used in the prototype.
- Implementing more complex functions related to information retrieval and filtering of applicants.
- Extending the set of attributes represented in a resume.
- Improvements for the usability of the prototype and the development of a mobile platform.

List of Figures

List of Tables

Appendix

A.1 Resume Model

Personal and general information

```
name:  String
picture:  String
website:  String
email:  String
phone:  String
address:  String
facebook:  String
twitter:  String
linkedin:  String
birthday:  String
gender:  String
nationality:  String
location:  String
aboutme:  String
occupation:  String
```

Languages and professional skills

```
languages:  Array of languages
    language:
        name :  String,
        level:  String
skills:  Array of skills
    skill:
        name:  String,
        level:  String
hobbies:  Array of hobbies
    hobby:
        name:  String,
```

```
        level:  String
```

Timeline information

```
timeline:  Array of events
event:
    title:  String,
    description:  String,
    media:  String,
    start_date:  String,
    end_date:  String,
    location:  String,
    category:  String,
    position:  String,
    company:  String,
    coords:  Array of coordinates
        coord:
            lat:  Number,
            lng:  Number


certificates:  Array of certificates
    certificate:
        name:  String
schools:  Array of schools
    school:
        name:  String
```

A.2 User study results

User # 1 (expert)

Task	Successful	Unsuccessful	Time (in minutes)
# 1	x	-	0.5
# 2	-	x	5
# 3	x	-	0.25
# 4	x	-	0.5
# 5	-	x	2.5
# 6	x	-	0.5
# 7	x	-	0.75
# 8	x	-	1
# 9	x	-	0.4

User # 2 (expert)

Task	Successful	Unsuccessful	Time (in minutes)
# 1	x	-	0.5
# 2	x	-	2
# 3	x	-	0.25
# 4	x	-	0.5
# 5	-	x	3
# 6	x	-	1
# 7	x	-	1
# 8	x	-	1
# 9	x	-	1

User # 3 (expert)

Task	Successful	Unsuccessful	Time (in minutes)
# 1	x	-	1
# 2	x	-	0.6
# 3	x	-	0.5
# 4	x	-	0.2
# 5	x	-	2
# 6	x	-	0.5
# 7	x	-	0.25
# 8	x	-	0.5
# 9	x	-	0.3

User # 4 (non-expert)

Task	Successful	Unsuccessful	Time (in minutes)
# 1	x	-	0.5
# 2	x	-	1
# 3	x	-	0.2
# 4	x	-	0.25
# 5	x	-	0.5
# 6	x	-	0.5
# 7	x	-	0.3
# 8	x	-	0.3
# 9	x	-	0.3

User # 5 (non-expert)

Task	Successful	Unsuccessful	Time (in minutes)
# 1	x	-	1
# 2	-	x	1.3
# 3	x	-	0.8
# 4	x	-	0.7
# 5	-	x	2.5
# 6	x	-	0.4
# 7	x	-	0.5
# 8	x	-	0.7
# 9	x	-	0.9

User # 6 (non expert)

Task	Successful	Unsuccessful	Time (in minutes)
# 1	x	-	0.6
# 2	-	x	2
# 3	x	-	0.5
# 4	x	-	0.3
# 5	x	-	1
# 6	x	-	0.7
# 7	x	-	0.5
# 8	x	-	1
# 9	x	-	0.6

Acronyms

API Application programming interfaces. 51, 53, 54, 55

ATS Applicant Tracking System. 2, 5, 24, 26, 29, 59

CEFRL Common European Framework of Reference for Languages. 36

CRUD Create, Read, Update, and Delete. 51

CSS Cascading Style Sheets. 52

CV Curriculum Vitae. 1, 2, 5, 11, 12, 31, 62

E-CV Electronic Curriculum Vitae. 1

GIS Geographic Information System. 23

HR Human Resources. 1, 30, 38, 42, 59, 69

HTML HyperText Markup Language. 52, 55

HTTP Hypertext Transfer Protocol. 51, 52, 54

IV Information Visualization. 1, 2, 3, 5, 7, 9, 18, 21, 24, 26, 27, 29, 30, 40, 59, 60, 62, 64, 65, 69, 70

JSON JavaScript Object Notation. 51, 55, 57

MVC Model-View-Controller. 51

NIH National Institutes of Health. 38

RAD Rapid Application Development. 34

RDLs Rapid Development Languages. 34

SPA Single Page Application. 51

SVG Scalable Vector Graphics. 52

UI User Interface. 35, 51, 56, 57

URI Uniform Resource Identifier. 54, 55, 56

VA Visual Analytics. 1, 2, 3, 16, 18, 29, 59, 60, 62, 65, 69, 70

Bibliography

[AMTB05] Wolfgang Aigner, Silvia Miksch, Bettina Thurnher, and Stefan Biffl. Planninglines: novel glyphs for representing temporal uncertainties and their evaluation. In *Information Visualisation, 2005. Proceedings. Ninth International Conference on*, pages 457–463. IEEE, 2005.

[BT76] Thomas E. Bell and Thomas A. Thayer. Software requirements: Are they really a problem? In *Proceedings of the 2Nd International Conference on Software Engineering*, ICSE '76, pages 61–68, Los Alamitos, CA, USA, 1976. IEEE Computer Society Press.

[CDW+16] Isaac Cho, Wewnen Dou, Derek X. Wang, Eric Sauda, and William Ribarsky. Vairoma: A visual analytics system for making sense of places, times, and events in roman history. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):210–219, Jan 2016.

[CMS99] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[Häg70] Torsten Hägerstrand. *What about People in Regional Science? I Carlestam, G. & Solbe, B.(red)(1991): Om tidens vidd och tingens ordning.* Texter av Torsten Hägerstrand. Stockholm: Byggforskningsrådet, 1970.

[JFS16] Stefan Jänicke, Josef Focht, and Gerik Scheuermann. Interactive visual profiling of musicians. *IEEE transactions on visualization and computer graphics*, 22(1):200–209, 2016.

[Lew82] Clayton Lewis. *Using the Thinking Aloud Method in Cognitive Interface Design*. IBM T.J. Watson Research Center, 1982.

[LM11] Saturnino Luz and Masood Masoodian. Comparing static gantt and mosaic charts for visualization of task schedules. In *2011 15th International Conference on Information Visualisation*, pages 182–187, July 2011.

[MA14] Silvia Miksch and Wolfgang Aigner. A matter of time: Applying a data-users-tasks design triangle to visual analytics of time-oriented data. *Computers & Graphics*, 38:286 – 290, 2014.

[Mar91] James Martin. *Rapid Application Development*. Macmillan Publishing Co., Inc., Indianapolis, IN, USA, 1991.

[Mun09] Tamara Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, Nov 2009.

[Nan] Ayaz Nanji. Eye-tracking study: How recruiters view resumes. `http://www.marketingprofs.com/charts/2014/25190/eye-tracking-study-how-recruiters-view-resumes`. Accessed: 2017-03-20.

[PSB10] Andrey A. Puretskiy, Gregory L. Shutt, and Michael W. Berry. *Survey of Text Visualization Techniques*, pages 105–127. John Wiley & Sons, Ltd, 2010.

[Shn96] Ben Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343, Sep 1996.

[SKK+08] Christin Seifert, Barbara Kump, Wolfgang Kienreich, Gisela Granitzer, and Michael Granitzer. On the beauty and usability of tag clouds. In *2008 12th International Conference Information Visualisation*, pages 17–25, July 2008.

[TC06] James J. Thomas and Kristin A. Cook. A visual analytics agenda. *IEEE Computer Graphics and Applications*, 26(1):10–13, Jan 2006.

[Web] Lauren Weber. Your resume vs. oblivion. `https://www.wsj.com/news/articles/SB10001424052970204624204577178941034941330`. Accessed: 2017-03-20.

[WGP+11] Krist Wongsuphasawat, John Alexis Guerra Gómez, Catherine Plaisant, Taowei David Wang, Meirav Taieb-Maimon, and Ben Shneiderman. Lifeflow: Visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1747–1756, New York, NY, USA, 2011. ACM.