UNIVERSITY OF DUISBURG-ESSEN

FACULTY OF BUSINESS ADMINISTRATION AND ECONOMICS

MASTER THESIS

Realizing Business Dashboards on Smartphones

A Visualization Approach

Submitted by:
Nils Höll

Supervisor:
Prof. Dr. Fabian Beck

Secondary Examiner:
Prof. Dr. Stefan Eicker

May 26, 2021

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Abstract

While more and more tasks in both private and business environments can be performed on modern smartphones, important dashboards presenting critical business metrics are still mostly limited to desktop PCs. In this thesis, we explore the use cases, requirements, possibilities, and limits of mobile dashboards based on a design science approach. Following a structured literature review the current state of the art in mobile visualizations, Business Intelligence and dashboard design was established. Afterwards, a series of expert interviews was conducted to assess current requirements and possible use cases of visualizing business data on smartphones. To assess the feasibility of mobile dashboards in general, the abstract use cases and requirements were used to build a Proof of Concept application, including adapted and novel visualizations optimized for the mobile context.

Our results show that using business dashboards on mobile devices could support an enterprises goals for data-driven decision making and serve as an entry point for novice users into more complex BI systems.

Zusammenfassung

Obwohl immer mehr Aufgaben sowohl im privaten als auch im geschäftlichen Umfeld auf modernen Smartphones erledigt werden können, sind wichtige Berichte, die geschäftskritische Metriken darstellen, meist noch auf Desktop-PCs beschränkt. In dieser Thesis werden die Anwendungsfälle, Anforderungen, Möglichkeiten und Grenzen von mobilen Dashboards mithilfe eines Design Science Ansatzes untersucht. Basierend auf einer strukturierten Literaturrecherche wurde der aktuelle Stand der Forschung in den Bereichen mobile Visualisierungen, Business Intelligence und Dashboard Design ermittelt. Anschließend wurde eine Interviewreihe mit Experten durchgeführt, um aktuelle Anforderungen und mögliche Anwendungsfälle der Visualisierung von Geschäftsdaten auf Smartphones zu bewerten. Um die Machbarkeit von mobilen Dashboards im Allgemeinen zu bewerten, wurden die abstrakten Anwendungsfälle und Anforderungen verwendet, um eine Proof-of-Concept-Anwendung zu erstellen, die sowohl angepasste als auch neue Visualisierungen enthält, welche für die mobile Nutzung optimiert wurden.

Unsere Ergebnisse zeigen, dass der Einsatz von Business Dashboards auf mobilen Geräten die Ziele eines Unternehmens zur datengetriebenen Entscheidungsfindung unterstützen und als Einstiegspunkt für unerfahrene Benutzer in komplexere BI-Systeme dienen kann.

Contents

List of Figures

List of Tables

Listings

List of Abbreviations

API	Application Programming Interface
BI	Business Intelligence
BIA	Business Intelligence and Analytics
BoK	Body of Knowledge
D3	Data Driven Documents
DSS	Decision Support System
DOM	Document Object Model
ERP	Enterprise Resource Planning
HTML	Hypertext Markup Language
HCI	Human-Computer Interaction
IBCS	International Business Communication Standards
IT	Information Technology
IS	Information Systems
IV	Information Visualization
KPI	Key Performance Indicator
OS	Operating System
PoC	Proof of Concept
PWA	Progressive Web App
SSBIA	Self-Service Business Intelligence and Analytics
SQL	Structured Query Language
URL	Uniform Resource Locator
UI	User Interface

1 Introduction

VISUALIZATIONS and dashboards are essential for businesses, as they provide critical information about the past, present, and possible futures of those corporations [59]. As the recipients of these reports, mostly executives and managers, are required to travel more [25], the demand for mobile availability of this information is increasing. Although the possibility to use mobile devices such as smartphones was already discussed at the start of the millennial [13, 49], the technology of the time severely limited the options for mobile reporting and detailed interactive visualizations. With growing screen sizes in smartphones [68], better displays, and significantly increased computing power these limitations are no longer a concern.

In combination with a projected worldwide revenue of Business Intelligence (BI) systems of 274 billion US dollars in 2022 [24], upcoming 5G high-speed mobile networks, and fast real-time enterprise databases like SAP's S/4HANA, mobile reporting might get even more feasible. Small visualizations for mobile dashboards have been discussed by various scholars, although their focus is mostly on domain-specific, simple data like electricity usage [2], medical data [54] or personal movement analysis [16].

We will address the challenges in mobile reporting and mobile dashboards from two perspectives: On the one side is the discipline of *Visualization*, or more precisely *Information Visualization*, focusing on how to represent and communicate data and information through the use of visual interfaces [37]. While bringing visualizations to mobile devices, researchers and practitioners face challenges regarding screen space, resizing, and scaling of graphics, possible interactions on touchscreens, and information encoding [41, 70].

Some scholars are developing small to word-sized graphics (*Microvisualizations*) as a means of conveying a large amount of information with little screen space. Current research about those visualizations and their interactivity focuses mainly on desktop PCs, or in general larger displays that are controlled via keyboard and mouse, rather than small touchscreen displays as found in smartphones. Commonly described interaction methods like tooltips or mouse hover actions as well as classical brush and link interactivity [27, 40, 4] are practically impossible on such devices. Additionally, the current focus on small, simple data sets for mobile dashboards as shown above leaves out further challenges for designing more complex dashboards, such as screen layout and navigation concepts.

The other side is formed by the business perspective, mostly the field of Business Intelligence, that provides insights into what data and information is needed, how performance indicators are calculated and finally how dashboards can be designed. In general, BI describes a set of applications, tools, and best practices for analyzing business data [39], of which mobile applications could be a valuable part.

Although visualizations play an important role in BI, many scholars take a management perspective, researching for example the implementation of BI systems [18] or possible data sources [38, pp. 15 sq.] rather than the presentation of information. Furthermore, common design guidelines like the ones provided by Smuts, Scholtz, and Calitz [64] are not suitable for mobile devices due to their focus on in-depth visual analytics.

This leaves a gap in between those two disciplines, leading to the following research questions:

> **RQ1:** *How and in which scenarios can mobile devices support data-driven decision making?*
> **RQ2:** *How can dashboards for business data be realized on mobile devices?*

Those questions were addressed using a design science approach: to get a good understanding of the state of the art in mobile visualizations and interactions, a concept-centric literature review was conducted. Those results were then further enriched with a set of expert interviews from different industries to derive both use cases of mobile dashboards, as well as requirements for such an application. The combined findings of those two perspectives were then transferred into a prototype application that serves as a technical demonstration to show the possibilities and limitations of mobile reporting. Lastly, the prototype was evaluated using a set of application scenarios derived from the interviews to have a realistic frame of reference.

2 Foundations

To better understand the following chapters and to establish a common definition of core concepts for this thesis the scientific foundations are presented in this chapter. Additionally, previous research related to the topic is analysed to establish a state of the art as well as to show the scientific gap justifying this work.

2.1 Business Intelligence

BI software has been a very large topic in enterprises since the 2000's, as both the amount of data collected in a business as well as the available computational power started to grow significantly [71]. BI systems are often seen as the evolution of Decision Support Systems (DSSs), following a more data-centric approach of obtaining business-critic information, therefore being driven by the information requirements of an organization [71]. In recent years those systems evolved to Business Intelligence and Analytics (BIA) and Self-Service Business Intelligence and Analytics (SSBIA) like Microsofts *Power BI*[1] or *Tableau*[2], that allow for explorative visual analytics of very large data sets, as well as standardized reporting.

In general, BI is usually an umbrella term for a set of software applications, tools, and processes for analyzing data to improve operational performance and to support decision making in enterprises [60, 44, 11]. The data is usually collected from multiple heterogenous data sources like Enterprise Resource Planning (ERP) systems, financial consolidation systems, data warehouses or external contractors [44, 38, pp. 15 sqq.].

If set up correctly, the data aggregated in a BI system can be used to analyze and explore processes, segments, or business units regarding performance or business targets, leading to management decisions based on this information. Due to advanced graphical interfaces, users can often easily assemble ad-hoc reports to get specific insights into the enterprises performance. Furthermore, reports from the BI system can show larger trends in the firms development, and support with planning and controlling strategies. To be able to deliver all information needed, the data base has to be both broad and deep - all relevant business units and departments have to report detailed numbers on a regular basis to generate a holistic and up-to-date dataset [39, pp. 31 sq.]. Therefore, according to Kemper, Baars, and Mehanna, BI projects are generally very broad and both horizontally as well as vertically integrated in the company.

To put it short, everything that is able to present data collected in an enterprise to users in a way that supports any decision-making process in that enterprise could be considered BI. Accordingly, putting BI capabilities on mobile devices like smartphones has been discussed since the early 2010's as described by Chen, Chiang, and Storey [11], although the interest dropped significantly after an initial hype [64], leaving lots of research questions open [39, p. 61].

[1] https://powerbi.microsoft.com/en-us/
[2] https://www.tableau.com/

In BI systems, especially financial data is often categorized into *scenarios* that represent different comparison categories. According to Hichert and Faisst [33, pp. 7 sq.], these are usually *Actual* (AC) for current data, *Previous Year* (PY) for data relating to the last financial year, *Budget* (BU) for target values, and *Forecast* (FC) for predicted future values. For clarity, those terms are also used throughout this thesis when the distinction is necessary.

2.2 Visualization

Information Visualization (IV) is a key discipline for this thesis, as the visual representation of data is one of the core concepts of BI in general and dashboards in particular. The first descriptions of this research area came up in the 1980's with McCormick [43] defining it as "encoding and abstracting data using symbols or graphics to generate visual representations".
Or to put it another way: by turning data into images or graphics, the amount of information that can be perceived just by looking at said image can be a lot higher than with raw numbers [65]. The field of visualization is broad and interdisciplinary, connecting design, cognition, Human-Computer Interaction (HCI), data processing, and various others. For the purposes of this work the sub-discipline of visual analytics is especially important, defined by Thomas and Cook [66] as "Analytical reasoning facilitated by interactive visual interfaces".

Although we do not aim to build a full-scale visual analytics system, but rather a simple dashboard due to the mobile context of this research, many insights of visual analytics research such as Shneidermans [63] famous visual information seeking mantra still apply:

> "*Overview first, zoom and filter, then details-on-demand*"
> \- Shneiderman, 2003

Within the intersection of IV, visual analytics, and BI is one particularly interesting visualization approach: dashboards. While driving a car, the dashboard has one main purpose: keeping the driver informed about every important metric such as the current speed or RPMs while displaying alerts for measures only relevant if they are out of bounds, say oil temperature or tire pressure. This concept of rapidly monitoring critical metrics, in busines contexts often referred to as Key Performance Indicators (KPIs), was mixed with the insights from IV research to create a class of visualizations with quite special properties [59].

As defined by Few, dashboards are single-page visual displays that provide information at a glance [59, 22, 21, p. 27] to provide the viewer with a fast answer to a specific problem. Therefore, knowing which information to display on a dashboard and which in other reports or visual analytics tools is a major challenge, since the dashboard itself should be tailored to the audience to be useful [21, p. 28]. As with the car analogy, these displays are generally used to maintain *situational awareness*, thus allowing a user to be aware what is going on based on information that is important to that particular scenario [21, pp. 31 sq.].

According to Sarikaya et al. [59], dashboards are built in nearly every industry, administration, and by individuals to support data-driven decision making, often as part of BI systems. They further argue that, while not exactly within the definition of Few, multi-paged dashboards with interactive interfaces are currently emerging in practice since they allow to monitor multiple different areas of interest.

For the design of dashboards the International Business Communication Standards (IBCS) by Hichert and Faisst [33] provide a standardized notation for specific data types, scenarios and situations that are widely used and known by practitioners and recognized in literature as a useful framework [30].

2.3 Related Work

The first scientific publications regarding visualizations on small mobile devices emerged in the early 2000's, only shortly after the term "smartphone" was even introduced [1, p. 278].

Noirhomme-fraiture et al. [49] published a paper in 2005 describing different possible interactions, visualizations, and limitations of mobile information systems with a focus on stock prices. Since the first smartphones were mostly used in business contexts [1, p. 278], the visualization of business data makes sense as a research objective at that time. While the limitations of those early devices are not comparable to todays smartphones, some of the findings still apply: Noirhomme-fraiture et al. advise to minimise screen changes and interactions, and to implement vertical scrolling whenever a single page is not sufficient to display the information. Furthermore, they introduce a heatmap-like visualization called a "Brick Wall" that is able to visualize daily changes in a metric over large timespans, similar to GitHubs "Contribution Calendar"[23]. Their evaluation shows it to be better regarding user performance compared to monthly bar graphs due to the high density, while still allowing for quick comparisons to other values. Lastly, the authors state that using contrast boundaries instead of lines (e.g. around graphics) is better to distinguish shapes and should therefore be preferred.

A more general view on mobile visualizations was provided by Chittaro (2006) [13], who describe the mobile use cases to be on a higher aggregation level than the standrad desktop ones, meaning that data should either be compressed or heavily filtered to be used on mobile devices. According to Chittaro, this is mainly due to the limited cognitive resources available in mobile scenarios, where users do not want to navigate through the usual data cube operations. The author also mentions the importance of selecting the right data, i.e. the right measures and their level of detail, as a key challenge in the field, while still emphasizing the need for some amount of interaction.

Cuttone, Lehmann, and Larsen (2013) [16] provide further insights into visualizing time-series information, although their research is centered around social interactions instead of busines data. However, they show that using a spiral timeline also works on mobile devices and is suitable for encoding bivariate data while still providing useful information even for unexperienced users. The authors state that visualizing patterns and especially outliers is easily done with spiral graphs, and that displaying different time frames (for example 24 hours or seven days) is feasible. Although different interactions like double-tapping or pinching were implemented in the prototype application, their performance is not evaluated in the paper.

While not a scientific publication, the collection of mobile visualizations and design guidelines collected by Ros in 2014[3] [58] are a valuable resource for further development. Especially the design patterns, based on previous research, best practices, and crowd-sourced examples include interesting advice, such as:

Vertical Bar Charts Use column charts instead of bar charts

Use Vertical Scrolling No horizontal scrolling, as already described by [49]

Stack Table Cells Use a more list-based display instead of tables

Carousel Instead of Tabs Carousels are more space-efficient and allow for swiping gestures to navigate

Fix Tooltips to Area of Screen Use a designated area for additional details

Use Touch Zones Provide clearly defined zones for touch interactions

Another interesting paper that is more closely related to our research was published by Apperley and Kalyan in 2015 [2]. While their research objective is a personal, not a business dashboard, the overall premises are similar to our work, since the need to communicate important information at a glance while the user is mobile is present in both settings. The authors conclude that even in a high-level scenario like theirs, having the ability to drill down for more details is a requirement for mobile dashboards. Furthermore, they suggest using tiles or cards with large numbers as an entry point for such a Drill-Down.

In the same year Watson and Setlur [70] wrote about emerging research in mobile visualization and the future of the field, citing the status of mobile devices as the dominant form of computing as a main driver. They come to the conclusion that visualizations need to become more useable for smartphones and tablets, since users increasingly demand to have easy access to measurements and data on the go.

A good review of the state of the art in 2016 is provided by Blumenstein et al. [8], who conclude that the amount of research in the field does not meet the demand among practitioners, even more so when focusing only on smartphones. They further say that the environment of actual users is seldom taken into account by researchers, although it is way more relevant than with desktop visualizations, since the attention spans for any tasks on mobile devices are usually a lot smaller. This overall view of the state of the art with mobile visualizations, especially for smartphones and in business contexts, aligns with the findings presented in this section - there is a clear gap with increasing practical relevance, but rather little attention from the scientific community, although a couple of workshops were conducted at the respective conferences in recent years [41, 14].

Very recent research seems to focus more on other devices like wearables with even smaller displays [7] or augmented reality lenses [73], that provides little value for the problem described in this thesis, at least from an interaction perspective.

Some other insights could be taken from the field of Microvisualizations, since the high information density of word-sized graphics could provide lots of benefits for displaying multiple measures on a single screen [40, 4].

Chen [12] focus on rapid monitoring tasks on smart watches, and point out the importance of reducing both the data and display complexity as far as possible. They also

[3] http://mobilev.is/

state that aggregations and mean values are important to achieve that goal, and propose to keep any interactions as simple as possible.

The work of Blascheck et al. [7] takes this one step further, suggesting that all labels, axes or legends should be omitted to keep the display clean. However, since their research was also conducted on smart watches, the results are not entirely applicable, as smartphone screens provide a lot more screen space for such annotations.

Another important angle are the interactions that users could perform to operate mobile visualizations, defined by the field of HCI. However, much of the research here focuses either on larger screens like tablets [17, 35] or much smaller ones such as smartwatches [50, 12] with a few exceptions.

One more general work is presented by Chang et al. [10] who researched users touch behaviour on mobile phones. While their results are not entirely applicable to our work, they provide a good foundation of common touch interactions and their discoverability, such as using a long touch for invoking context menus.

Schwab et al. [61] researched interactions to navigate large time-series data, coming to the conclusion that the default pinch interaction should be preferred over hierarchical scales for zooming in on data. They further say that for navigating those timelines they should be oriented along the longer axis of the device to increase interaction efficiency.

2.4 Definitions

2.4.1 Mobile Devices

As the term *Mobile Device* is rather ambiguous but critical to this work, a clear definition is necessary. In both the introduction and above in Section 2.3 we already set the scope for our research to smartphones only. Therefore, *mobile* excludes any form of tablets, laptops, and smart watches, although those are technically mobile since they can be used anywhere. For the sake of simplicity, if not explicitly stated, whenever we write about *mobile devices*, *mobile dashboards*, *mobile reporting* or *mobile BI*, we only refer to handheld devices with a screen size of four to seven inches, as this covers almost the entire smartphone market as of 2019 [68].

2.4.2 Key Performance Indicators

To objectively evaluate and monitor the performance of an enterprise complex relationships need to be reduced to a few metrics [26, p. 231]. If those are used to control and steer the organization, and therefore reflect its goals and strategy, they are often referred to as Key Performance Indicators (KPIs) [26, p. 233, 57]. Therefore, those measurements are critical to a company's success, and thus usually collected in dashboards for an easy overview [21]. For this thesis, a more general definition is applied, since the definition of which metrics are actually important depends on the industry, department, and audience of the dashboard. That means every indicator that might be important for a user is considered a KPI, even if it is not a key measure for the whole organization.

3 Research Approach

To close the gaps in the current Body of Knowledge (BoK) and answer the research questions as presented in Chapter 1, we employed an iterative approach based on the design science strategy described by Hevner et al. [32].

This research paradigm is often utilized in Information Systems (IS) research that focuses on development of Information Technology (IT) artefacts and can lead to results of a high practical relevance, as Oates [51] describes it:

> "*These projects explore and exhibit the possibilities of digital technology.*"
> - Oates, p. 109, 2006

Therefore, this approach was the best fit for our goals, since the creation of an IT artifact, namely a mobile application for business dashboards, is one of our core contributions. Our application is a scientific contribution as it shows that - and how - requirements which are relevant to todays businesses can be realized on modern smartphones as a dashboard application. The software is a novel artefact as described by Recker [55, pp. 106 sqq.] since it aims for a very broad target audience, as opposed to other mobile reporting products that have been developed for scientific purposes as shown in Chapter 2.

For structuring the whole approach, Hevner et al. [31] propose a method based on three cycles:

Relevance Cycle Provides the problem to be addressed as well as requirements, acceptance criteria, and evaluation of artifact iterations.

Rigor Cycle Responsible for a solid grounding of the research in the current BoK as a base for innovation.

Design Cycle Iterative construction and evaluation of the artifact based on theories and methods from the Rigor Cycle as well as requirements from the Relevance Cycle.

In our case, the *Rigor Cycle* is represented by a standard concept centric literature review using scientific publications from both the fields of visualization and businesses intelligence to get a thorough understanding of the state of the art and open questions in mobile visualizations, interactions, and dashboards. This base was then extended by a set of semi-structured expert interviews for the *Relevance Cycle* with the goal to get a better understanding of possible use cases for mobile reporting and requirements for an application.

Those findings were then transferred into a Proof of Concept (PoC) mobile application (see Figure 3.1) that was used to validate and improve the initial results. Its purpose is to demonstrate different types of visualizations, the feasibility of real-time mobile dashboards as well as the usefulness of notifications for special business events. The prototype development followed an iterative, agile approach that fits very well into the design science strategy.

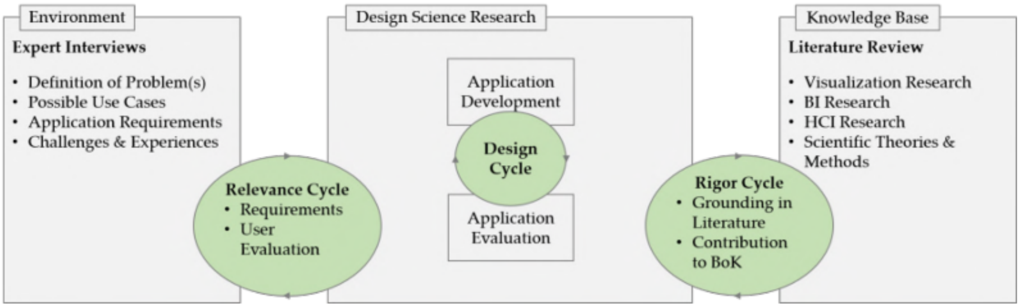


Figure 3.1: Overview of the three design science research cycles, based on Hevner [31]

3.1 Literature Review

The literature review was conducted as the first step in our research, mostly to get a solid understanding of the state of the art in the various involved fields and to identify a clear gap in the research.

We utilized a structured, concept-centric approach as proposed by Webster and Watson, a very common strategy in IS research [72, 3]. We started by reviewing material from renowned journals and conferences, such as IEEE VIS[1] or ACM CHI[2], and made use of scientific search engines like IEEE Xplore[3], AISeL[4], and ScienceDirect[5].

To thoroughly understand the foundations of each topic we conducted backward searches from our initial sample. Furthermore, forward searches ensured that we had the most recent findings at hand. In total, we were able to find 37 relevant sources for the field of visualization  (shown from 2003 to 2021), 22 for BI  , and another 9 for HCI  , with the majority of publications (over 50 %) from 2015 to 2020, as shown in Figure 3.2.

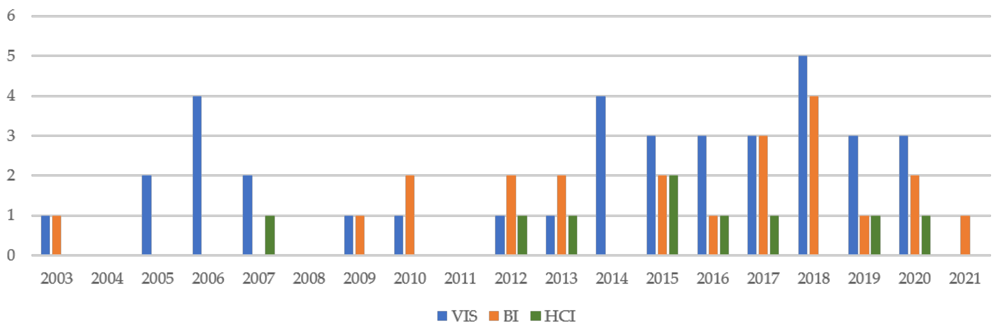


Figure 3.2: Distribution of scientific sources over publication years and fields

One major finding while specifically searching for previous work about mobile reporting was out of the surprisingly little research into this topic a large part was conducted in the early 2010's [69], rendering it irrelevant due to vastly differing hardware capabilities. This effect was later confirmed by various interview participants with consulting

[1]https://virtual.ieeevis.org/index.html
[2]https://sigchi.org/
[3]https://ieeexplore.ieee.org/
[4]https://aisel.aisnet.org/
[5]https://www.sciencedirect.com/

background, stating that the release of the first and subsequent iPhones around this time [20] created a "hype" around mobile reporting, that later died down due to a lack of usefulness of these applications, driven by inadequate screen resolutions and computing power.

The results of the literature review mostly influenced Chapter 2, since they form the scientific baseline upon which this work is built. In addition to that, we could also make use of previous work to back up and complete the requirements listed in Chapter 4, and to support the evaluation of the prototype.

3.2 Interviews

The second cornerstone of our research is a series of interviews with experts from various companies. These were conducted to get a good understanding of the possible use cases, challenges, and especially requirements associated with reporting business data on smartphones. Interviewing experts in the field ensures a high practical relevance of our research, and resulted in rich insights into the needs of users.

To get both a comparable structure between individual interviews, while still remaining open for new and explorative inputs, we decided to utilize a *semi-structured* approach [55, pp. 50 sq.]. This means that although we provided participants with a fixed set of key questions that acted as a reference frame for guiding the interview, we were able to ask follow-up questions about interesting aspects or leave out unnecessary parts where applicable. Therefore, the interviews were essentially structured into three parts: First, we would talk about the current use of BI systems in the participants company, then about possible use cases for mobile dashboards and lastly about requirements for such an application.

Prior to the actual interviews, all participants were informed about the purpose and topic of the interview, as well as their anonymity in this work. To make this more efficient and allowing the interviewees to prepare, we handed out interview guides containing all relevant formal information as well as a short introduction to the topic of this thesis and a set of key questions aligned with the planned structure of the interview. Those guides are available for reference in both German and English in Appendix C. In addition to the interview guides we prepared short presentations containing essentially the same information to serve as visual aid for the participants.

The procedure described above is largely based on the work of Hove and Anda [34] who specifically researched semi-structured interviews for software engineering. The interviews were conducted using various videoconferencing tools, since face-to-face meetings were not a viable option due to the Covid-19 pandemic.

The analysis of the interviews was conducted as a two-step qualitative coding process as described by [55, pp. 92 sq.] using RQDA as a coding tool. In the first iteration, a rough code structure was set up to classify most of the codes created while analysing the interview notes. Afterwards, this structure was revised based on the actual codes that turned out to be relevant, as well as the demands for further development. Existing codes were then sorted into those new categories, and merged or redefined where needed, completing the second round of the coding process. The final code structure, along with all used codes and their descriptions, is listed in Appendix D.

The results from the coding process were then used as the base for building both abstract use cases and deriving more specific requirements, both of which are presented in Chapter 4.

3.3 Prototype Development

The core of the design science strategy, the *Design Cycle* as shown in Figure 3.1, is represented by the development of a prototype application. The goal of this software is to show how the requirements derived from both literature and the interviews can be realised in an actual mobile application. The focus is hereby on the frontend side, more specifically the interaction patterns and visualizations required to make common dashboard tasks both useful and usable within the restricted capabilities of smartphones.

To quickly get a working PoC an iterative approach was employed, since it worked quite well with the also iterative nature of the design science method. This means that a basic layout and structure of the application could be developed while the requirements research as described above was not finished. To simplify and speed up development, we made use of existing frameworks, libraries, and best practices where possible.

Parts of the application that do not contribute to answering the research questions have not been developed or reduced to a minimum, even if they would be necessary for enterprise-grade software. This includes everything related to user and identity management, authentication and authorization, application security in general as well as production-ready data models and back-end software.

4 Findings

A necessary prerequisite for a useful application is a solid set of use cases and requirements that reflect the actual needs of potential users. Therefore, as described in the previous chapter, a combination of the expert interviews and existing literature forms the base for the application development.

After a summary of the interviews we present the most prevalent usage scenarios as well as a set of high-level requirements. Since the development part of this thesis only aims at presenting a prototype application and not production-ready enterprise software, that high level of abstraction is sufficient for our needs.

For the sake of simplicity, references to interviews or the participants will be given in the format #01, with the ID referring to that in table 4.2. Interview participants will be referred to as *participants*, *interviewees*, or *experts*.

4.1 Interview Results

In total, 20 interviews were conducted, of which the first (#00) was the only unstructured interview. As its goal was to get a first impression about the topic from an expert view, we marked it as #00 to show its special role as a pilot among the other conversations with experts which were conducted as semi-structured interviews as described in section 3.2. In addition to that, it provided us with an opportunity to test the interview guide that became the base for our later interviews, hence acting as a test run for the core part of our research.

4.1.1 Classification of Interviews

As seen in Table 4.1 we interviewed experts from 13 companies across 7 industries, ranging from consulting firms over logistics- and production-oriented corporations to financial institutions, providing rich and diverse insights into the topic.

The participating companies range from only six to over 140 thousand employees with a large focus on medium-sized enterprises as represented by a median size of 2000. Based on the data presented above, the companies essentially split into two main categories: ***Consultancies*** that either implement or develop BI software (companies **A** to **F**) and ***Potential Users***, that would deploy such a system (companies **G** to **M**). Even though consultancies might very well also use reporting software, this differentiation makes sense since participants from the former category can again provide insights from various different clients, while the latter is able to give more in-depth information about their specific case, leading to both broad and deep results. This effect is further amplified by the fact that multiple employees from the companies **G**, **H**, and **L** participated in the research, which allowed us to take an even more detailed look at those specific use cases.

Table 4.1: Information about the participants employers

ID	Size (Employees)	Industry
A	144000	Computer Software
B	6	Consulting
C	11	Consulting
D	45	Consulting
E	4400	Consulting
F	11000	Consulting
G	900	Chemicals Distributor
H	3800	Consumer Goods
I	1000	Finance
J	10200	Finance
K	300	Medical Technology
L	2000	Medical Technology
M	5000	Pharmaceutics

The actual interviews were conducted from December 2020 to the end of February 2021 and took between 30 and 60 minutes (median: 40 minutes) depending mostly on the availability of the participants. However, most interview partners stated that they had read the interview guide, enabling much more efficient interviews by directly jumping to the parts relevant for our research and leaving out long introductions or formalities.

Table 4.2: Details of all expert interviews and participants

ID	Company	Role	**Duration** (min.)
00	F	Senior Manager BI	45
01	H	BI Project Manager	45
02	H	BI Specialist	35
03	H	Report Owner Technical Performance	30
04	H	Sales Controlling	30
05	L	Corporate Development	50
06	G	Managing Director	30
07	G	Senior Business Controller	60
08	B	Managing Director	40
09	G	Managing Director	30
10	G	Managing Director	40
11	C	Managing Director	45
12	J	Competence Development Officer	40
13	E	Senior Consultant BI	35
14	D	Senior Director	45
15	I	Project Manager	55
16	M	VP Controlling Systeme & BI	30
17	L	SVP Manufacturing	45
18	A	Data & AI Lead	30
19	K	VP Sales and Business Development	40

A little over half of the participants stated they worked with some sort of dashboard or reporting tool at least daily, another six reported at least a weekly use, leaving only 15% of the sample with either little dashboard experience or not commenting on their usage.

Out of the **Consultants**, every single interviewee noted they had experience in dashboard design, while that number dropped to around 30% for the group of **Potential Users**.

4.1.2 Descriptive Codes

After the second iteration of coding as described in Chapter 3.2 a total of 47 codes over seven categories could be identified, as seen in Figure 4.1[1]. Almost half (22) of those codes are directly related to requirements, and another six to possible use cases. Those codes are more deeply analysed in the following Sections 4.2 and 4.3 since they directly impact development.

The nature of the data to represent is described by a total of seven codes, of which four relate to the timeliness (e.g. daily or monthly updates) of said data, and another three represent the different business functions. While the business functions are almost equally distributed, the codings show that most of the important data is updated either daily or more often, with only around 22% updated monthly. In most cases, those monthly updates are for financial data that needs to be consolidated across a whole group, which is usually done for the monthly financial statement. Nonetheless, some interviewees (#01, #06, #12, #17) expressed a desire to get even that rather small but important part to a higher update frequency, for example using pre-consolidated data.

Three different although equally important target audiences could be identified: standard employees, middle management, and top management such as executives.

The biggest challenge reported by participants is the quality of the available data, with interviewees #03, #04, #07, and #10 specifically citing quality issues with master data as a main issue of the current reporting systems. This can lead to, as described in interview #19, people getting different results for seemingly the same queries and reports, possibly resulting in wrong decisions or miscommunications.

While the data quality issue was only mentioned by participants from the group of **Potential Users**, multiple experts across both groups stated that a low level of *data literacy* among employees also poses a challenge. This can result in either misconfigurations of complex BI systems or misinterpretations of data, increasing the risk of suboptimal decisions. Furthermore, a total of 16 ideas for interactions as well as seven possible visualizations for mobile dashboards were identified through the interviews.

[1]The column *Frequency* displays how often a code was used, while *Interviews* shows in how many interviews that code was used at least once. The table is grouped by *(Sub-)Category* and sorted by *Interviews*.

Category	Sub-Category	Code	Frequency	Interviews
Functional Requirements	General	Push Notifications	20	14
		Personalization	16	12
		Offline Usage	10	9
		Usability	9	8
		Visualizations	9	8
		Favourites	3	3
	Interaction	Comparison to History	20	16
		Deviation Analysis	21	14
		Collaboration Features	13	11
		Drill-Down	11	9
		Timeframe Selection	2	2
		Dimension Selection	1	1
	Visualization	Information at a Glance	23	16
		Trendlines	16	13
		Entity Dashboard	9	8
		Forecasts	7	7
		Data Context	6	6
		Top/Flop Lists	4	3
Non-Functional Requirements		Platform-Independent Use	9	9
		Application Security	9	6
		Access Control	4	3
		Performance	2	2
Data Structure	Business Function	Financial Data	13	12
		Sales Data	13	11
		Operational Data	12	8
	Timeliness	Updated Daily	14	12
		Updated Intraday	10	8
		Updated Monthly	7	7
		Realtime or Streaming Data	1	1
Target Audience		Top Management	6	6
		Employees	7	5
		Middle Management	5	5
Use Cases		Easier Access	17	13
		Monitoring	18	11
		Field Work	9	9
		Increase Data Interaction	8	5
		Meetings	5	5
		Operational Support	9	5
Dashboard Experience		Daily Use	11	11
		Dashboard Design	11	11
		Weekly Use	6	6
		Little Dashboard Experience	1	1
Challenges		Data Quality	8	6
		Data Literacy	5	5
		Economic Viability	4	3
		KPI Selection	4	3
		Level of Details	1	1
Demand		Demand for Mobile Reporting	11	11

Figure 4.1: Overview of codes, categories, and their frequencies

4.2 Use Cases

Based on the qualitative analysis of the interviews as shown above, the most important findings are the use cases for mobile dashboards. While the previous section explained why such an application might be useful in general, finding use cases with associated tasks and target audiences is vital for a justification and evaluation of the whole concept as well as a prioritisation of requirements.

During the analysis of the interviews we quickly realized that specific scenarios for mobile reporting vary greatly between target industries, business models, and even departments of the same company. Therefore, we decided to abstract and group those use cases presented by the interviewees, and to generalize them into three rather high-level cases as shown below to ensure a higher applicability of our findings to most industries and enterprises.

1: Easy Access to Data

This use case is largely based on two codes from the analysis, “Easier Access” and “Increase Data Interaction”, since those essentially have a very similar underlying problem: users who want to access certain data or information, but neither have the expertise nor understand or even find it due to complex BI tools. Therefore, this use case also aims at solving the challenge of “Data Literacy” which was mentioned in five interviews.

The main goal, however, is to enable a more data-driven decision making, simply by making said data more accessible. Increasing the ease of use and availability of information was cited as a main driver for mobile reporting by #02, #03, #12, #14, and #19. A good example of why this is necessary in some cases was presented during interview #05, in which the participant reported that a former client used to distribute monthly reporting packages so extensive that no one worked with them - a fact proven by an experiment, in which the participant decided to not publish a reporting package at all for one month, without receiving any complaints or inquiries about the missing data.

While this example might be an outlier, it illustrates the issues with some reporting solutions very well: complex and versatile analytics platforms are good for deep insights by trained experts, but often too much of a hassle to operate if a user only occasionally needs access to few KPIs. This leads, as reported in the interviews #02, #05, #07, and #09, to low efficiencies and higher error rates due to misconfigurations of for example filters. In those cases, having very quick access to a set of pre-defined measures enriched by easily understandable visualizations could help to increase the awareness of those numbers, leading to decision making based on objectively comparable data (#14).

Expert #05 further argued that an increased transparency due to easier access could lead to discussions about and therefore interaction with the data, as well as a better visibility of (personal) targets. Wiener and Namazci refer to this concept of bringing information to every employee as “Data Democracy” or “Citicen BI” [74].

With a total of 15 interviews that were tagged with either one of the codes mentioned above, and no restrictions on the target audience (except for specialists who make regular use of analytics tools anyway), this is easily one of the most convincing use cases.

Bringing critical data onto devices that are continuously used throughout the work day, presented with visualizations that support easy interpretation at a glance, utilizing familiar interactions that do not allow for misconfiguration seems to be a perfect scenario for mobile dashboards.

2: Monitoring

While the former use case is concerned with a general accessability of information for status updates and getting employees to use that information, in some cases much more current data is required for short-term reactions to specific business events.

For this use case the codings for "Monitoring" (11 interviews) and "Operational Support" (5 interviews) are combined, leading to a total of 14 experts mentioning either one of these cases as relevant. The general idea is to provide a dashboard that is a lot closer to the definition of Few [22] and therefore allows for rapid monitoring of very current data, e.g. hourly updated or even streaming data. The main focus is on operational information like production data, since it does not make sense to calculate and report sales or even financial KPIs on such a regular basis, as reported by #04, #05, #08, #16, #17, and various others indirectly.

Especially in production scenarios, for example in a factory, quick reactions to certain events can be very important. A good example for this was provided by employees from company **H** that mass-produces consumer goods: failures in machines, deviations in product quality or even just a high count of sick employees that would require re-staffing of shifts are events that need to be reported to the responsible managers as quickly as possible. Participant #15 referred to such a system as a "portable control center" that would inform a user about critical business events which require stopping whatever that user is currently working on to address the problem.

Even the companies **K** and **L** that produce comparatively low volumes of medical devices reported a need for quick notifications of management in case of disruptions - currently, employees are often too busy fixing the problem that it might only get reported hours later. The mobile aspect is especially important in those scenarios, as reported by #11, #14, #15, and #17, since they need to travel a lot between manufacturing plants and cannot really use a laptop on a production floor. Therefore, smartphones are heavily used by the target audience for their daily work, increasing the usefulness for reporting on the same devices.

Another scenario was brought up during interviews #05 and #08 that mentioned marketing campaigns, especially online-marketing, as a possible beneficiary of mobile dashboards. According to the interviewees, such campaigns are highly time-critical, and an hour of money spend ineffectively on advertisements could lead to notable financial impacts. Additionally, participant #02 suggested live monitoring of critical (IT) infrastructure on mobile phones to quickly react to any incidents or outages.

To summarise, the monitoring use case is mostly about middle management making sure that critical business functions like production are up and running, essentially providing a dashboard that allows for a quick check if everything is in order, and actively notifying users if not.

3: Meeting Preparation

As with the previous use cases, this last one can also be abstracted from two codes from the interviews, in this case "Field Work" and "Meetings".

The former refers mostly to traveling sales people that actually *need* mobile access to certain information, but could also be applied to service technicians and other highly mobile employees, although only the sales case was mentioned during the interviews. This use case was reported as relevant during 9 interviews, of which #01, #04, #07, #09, #12, and #13 specifically mentioned the importance of accurate and up-to-date customer data sheets as a vital preparation task for sales employees that meet with clients.

The interviewee #19 also mentioned regular problems with outdated customer information for employees working in remote locations, since the currently used tool requires all data of all customers to be downloaded, which again requires the laptop to be booted, a stable internet connection, and lastly a search for the correct data. With a smartphone application this could easily run over mobile internet instead of WiFi or broadband, since only the relevant customer data would be downloaded and could be presented and interpreted a lot easier and especially faster.

This is in line with the five interviewees that mentioned meetings in general as a possible use case, citing the reduced need to bring and set up laptops for quick meetings as an advantage for mobile reporting. Since the field work cases are essentially meeting preparations too, although with very different circumstances than a quick project meeting at the company, combining those two cases makes sense.

4.3 Requirements

In addition to the rather abstract use cases described above, various more specific requirements could be derived from the interviews. Those are separated into the categories non-functional and functional, with the latter further split up into general requirements and those specific to visualizations and interactions.

The requirements below are largely prioritised by their associated coding frequency, shown in parentheses after their name[2]. However, those numbers should rather be seen as an orientation, since due to the nature of semi-structured interviews not all of the possible requirements came up during every interview to check whether they are important for a participant.

There is no hierarchical relationship between use cases and requirements, meaning that, if not specified otherwise, all requirements listed below are equally relevant for all use cases.

4.3.1 Functional Requirements

The requirements listed in this section describe different parts of the functionality of the application that mostly contribute to the realization of the use cases described in Chapter 4.2.

[2] Requirements marked with an "*" are derived from more than one coding, so the count of either one of the codes is displayed.

General Requirements

One of the general features (see Figure 4.2) most requested are push notifications (**FR-1.1**), i.e. a way of actively notifying a user of certain important events even when the application is not being used. Such functionality set mobile reporting systems apart from standard desktop ones, as relevant information can be pushed to the end device. This makes it a critical requirement for the monitoring use case, but can also be relevant for the first one by possibly increasing user interaction with the application as described by #10.

FR-1 – General Requirements

FR-1.1 Push Notifications (14): Notify the user about important events; especially important for Use Case 2

FR-1.2 Personalization (13)*: Individualised selection of displayed information and favourites

FR-1.3 Offline Usage (9): Local storage of important information to access without internet connection

Figure 4.2: Overview of general requirements and their code frequencies

Since one of the main goals of having a mobile dashboard is to provide easy access to information relevant to that person (as opposed to giving access to all data and letting users pull relevant information out of it), customization (**FR-1.2**) is quite relevant. Users should, for example, be able to select KPIs that are relevant to them as favourites for faster access or to change the landing page to the best suited dashboard. Multiple participants especially from the consultants group pointed out the importance of individualisation to increase both the relevance of the application as well as decrease interaction times.

The last genereal requirement is concerned with the availability of data in situations without internet access (**FR-1.3**), as it might happen during travel. Therefore, it is mostly relevant for the last use case, more specifically for field work as conducted by for example sales people. However, participants even from the same company did not agree on whether or not this was needed, with interviewee #09 stating that "nobody is actually offline anywhere it matters", while their colleague #10 saw offline capabilities as a very important feature for mobile reporting. In general, the participants agreed that very important data should be available at all times, since the low number of KPIs needed by each user would result in rather little data to store.

Interactions Requirements

The following interaction requirements (see Figure 4.3) are mostly abstract representations of actual tasks participants wanted to be able to perform on a mobile dashboard.

Among those, comparison tasks (**FR-2.1**) like deviations to targets or historical values were requested most often by the interviewees. This makes sense especially in the first two use cases, since the evaluation of a currently measured value only gets useful with some context on how it relates to a goal or how it developed over time. In the end, deviations and deltas are more important than the actual, absolute numbers of a KPI (#08), since in the end those differences lead to corrective actions.

Another important requirement that could set a mobile dashboard apart from the classic desktop version are collaboration features (**FR-2.2**). According to #18 those are especially relevant in a mobile context, since the user might not be able to take any action right away - thus, sharing the current view or alert with other members of the team might be a very useful feature.

FR-2 – Interaction Requirements

FR-2.1 Deviation Analysis (14)*: Display deviations between current and target or historical data

FR-2.2 Collaboration Features (11): Exporting, sharing and commenting abilities

FR-2.3 Drill-Down (9): Switching to other hierarchy or detail levels

FR-2.4 Time Frame Selection (2): Selecting the displayed time frame

FR-2.5 Dimension Selection (1): Switching the base dimension

Figure 4.3: Overview of interaction requirements and their code frequencies

A more standard feature of BI systems is the *Drill-Down* (**FR-2.3**), an interaction that lets users change the hierarchy level of a visualization, for example by going from an aggregate overview of product lines to a per-product view of a single product line. Most participants agreed that this is a very important requirement, but that the number of levels should be limited to keep the complexity low.

Lastly, two additional requirements could be identified that were only mentioned once or twice in the interviews, but relate closely to other features and were therefore included here. Selecting the time frame displayed (**FR-2.4**) is important for visual comparison tasks to historical values (relating to **FR-2.1**) since looking at a month worth of data might paint a different picture than comparing to a whole year. Although rarely explicitly stated as a requirement, the analysis showed that interviewees had quite different expectations of the time frames to be displayed, ranging from a week for monitoring use cases (#02) to multiple years worth of data for detecting seasonal effects (#06, #12, #19). Furthermore, being able to switch the base dimension **FR-2.5** is a relevant feature regarding Drill-Down operations, since it enables a user to quickly change the perspective on the data presented.

Visualization Requirements

Taking into account the planned use cases for a mobile dashboard it makes sense that the most popular requirement from a visualization perspective (see Figure 4.4) is to get information at a glance (**FR-3.1**). This is in line with the general literature regarding dashboards, especially as defined by Few [21, 22], since the whole point of having a dashboard is to get relevant information as quickly as possible. Especially in a mobile scenario, being able to just glance at the device to interpret the visualizations is a critical feature, as the interaction times are bound to be low. Participant #18 put the average

FR-3 – Visualization Requirements

FR-3.1 Information at a Glance (16): Getting the most important information with one look at the dashboard

FR-3.2 Trend Lines (13): Showing recent trends based on moving averages

FR-3.3 Entity Dashboards (8): Special dashboards to display information about a specific entity like a customer or project; especially important for Use Case 3

FR-3.4 Forecasts (7): Display possible future scenarios

FR-3.5 Data Context (6): Show context of the currently displayed information (e.g. dimensions, time frame etc.)

FR-3.6 Top/Bottom Lists (3): Display the top or bottom items of a specific dimension based on one KPI

Figure 4.4: Overview of interaction requirements and their code frequencies

time to complete a task at under a minute, leading to high demands for both visualizations and interactions.

One particular type of visualization regarding comparisons to historical data (**FR-2.1**) that was requested often are trend lines (**FR-3.2**). While effectively being line charts, their main purpose is to quickly show the development of a KPI over an arbitrary time frame, with the option to smoothen the curve using moving averages.

Requirement **FR-3.3** is not about a specific type of visualization, but rather a whole concept that closely relates to the meeting preparation use case. The general idea is to design a whole dashboard that displays a fixed set of KPIs, filtered down to a single object of interest. This could be a customer, so a sales person in the field can prepare for a visit, a dashboard including critical information about a projects progress, or any other business entity that is of relevance to the company deploying the application.

A little less important than previous requirements is the ability of the dashboard to incorporate possible future values, i.e. forecasts, into existing visualizations (**FR-3.4**). This is mainly relevant for deviation analyses regarding target values, as comparing to a monthly target in the second week of the month is of little use without an indication of where the measure might be at the end of that period.

Especially when implementing more complex interactions like Drill-Downs and filters, users might loose track of what information they are currently viewing. According to Verkooij and Spruit [69] as well as various participants, this is particularly important in a mobile scenario, as filters and time frames have to become apparent at a glance just like the rest of the visualizations. Therefore, implementing a feature to display the current context (**FR-3.5**) makes sense from a usability perspective, even more so since labels, legends and even axes will be reduced to a minimum on the actual visualization.

Building upon the idea of entity dashboards (**FR-3.3**), some participants requested another layer on top for an easier selection. It seems intuitive to start with a list of customers, sorted by sales volume or another KPI, and then drill down into the actual dashboard. Especially for scenarios with lots of possible business objects, say a product line manager with around 10 products, this gives a good first overview based on the most important metric, and allows for details when needed.

4.3.2 Non-Functional Requirements

To round things up, a couple of non-functional or technical requirements could be extracted from the interviews, as shown in Figure 4.5. Most notably (and quite obvious for any application) various participants pointed out the criticality of having a high usability, especially for the first use case that aims at users that are not familiar with reporting or BI tools. More specific, as defined by Nielsen, that means the system has to be:

- Easy to learn
- Efficient to use
- Easy to remember
- Mostly error-free
- Subjectively pleasing

Or, as #11 put it: "No one actually *wants* to use mobile reporting, and no one will if it's not very intuitive". This is further supported by other interviewees, stating that usage will be very low if there is any hassle (#03), configuration should be minimized (#09), and that design should be close to existing apps (#18). To ensure this requirement is met, we utilized the usability guidelines for mobile websites as defined by Shitkova et al. [62] for all features where they are applicable.

Closely related to this is requirement **NFR-2** which states that the application should be able to run independent from the platform, as both Android and iOS devices are prevalent in the business world.

NFR – Non-Functional Requirements

NFR-1 Usability (8): Intuitive usage of the application

NFR-2 Platform-Independent Use (9): The application should run on both Android and iOS Operating Systems (OSs) on recent devices

NFR-3 Application Security (7)*: Access control, strong authentication and local encryption of data

NFR-4 Performance (2): Fast loading times

Figure 4.5: Overview of interaction requirements and their code frequencies

Although mentioned quite often, anything related to application security (**NFR-3**), i.e. access control, authentication, encryption etc. is considered out of scope for the PoC we try to build. Since it is a very important part of any enterprise system, especially regarding the potentially highly sensitive data handled by the software, this should be taken into account for any production-ready application.

Lastly, performance was included as a requirement since the short timespans of interactions demand a significantly lower response time than compared to a desktop system, as described by #16.

4.4 The Case for Mobile Reporting

One of the most important codes regarding the justification of this thesis is “Demand for Mobile Reporting”, which we found in eleven of 20 interviews (over 60% from possible users and over 40% from consultants). However, this does not mean that the remaining nine participants deemed mobile reporting unnecessary, as only the explicit need for such an application was tagged with this code. In the interview #05, for example, the participant stated that they saw very high potential in mobile dashboards, but could not see an actual *demand* at their company. Nevertheless, this shows that there are clear needs regarding some sort of mobile data representation in todays businesses, so we will further explore what exactly those companies want and expect.

The few participants who have actually used current mobile reporting solutions reported a multitude of problems and challenges, ranging from scaling problems even on tablets (#07), unusable interactions and filtering options (#04), a general lack of understanding of how those dashboards could look like (#14), to outright not being able to do anything useful (#10). This is often attributed to mobile dashboards being just scaled down versions of their desktop counterparts instead of being specifically designed for smartphone screens and touch operations. This is important not only because the sizes, resolutions, and aspect ratios are different, but mostly since mobile phones are used very differently than desktop PCs or laptops. While the latter are used for most of the actual daily work, mobile devices are very good for a quick glance at some new information - they’re always available, regularly used anyway, and require almost no effort or time to look at, which is especially useful when the operator is on the move.

One argument against the need for any form of mobile reporting is the rather slow update rate of the data to monitor, with only a single participant stating they even had access to real-time data. In most cases, having information from the last day was current enough for the interviewees’ reporting needs, raising the question whether someone actually *needs* business data on their smartphone, since they would probably check on their PC before making any decision anyway. This specific concern was explicitly raised during interview #13 and at least mentioned in some form from a few others, especially from the consulting side.

On the other hand, participant #10 reported a clear trend towards using dashboards as a first step into more complex visual analytics from their customers, so todays businesses are aware of the potentials of enabling some sort of BI for most of their employees. As shown in Section 4.2, we were able to identify quite different compelling use cases in which mobile dashboards could significantly support the daily work of employees mostly by reducing the time and complexity to get to important information. This also makes a difference with data which is only updated daily or even monthly, since the main issue is not the actuality, but rather the accessability of the former.

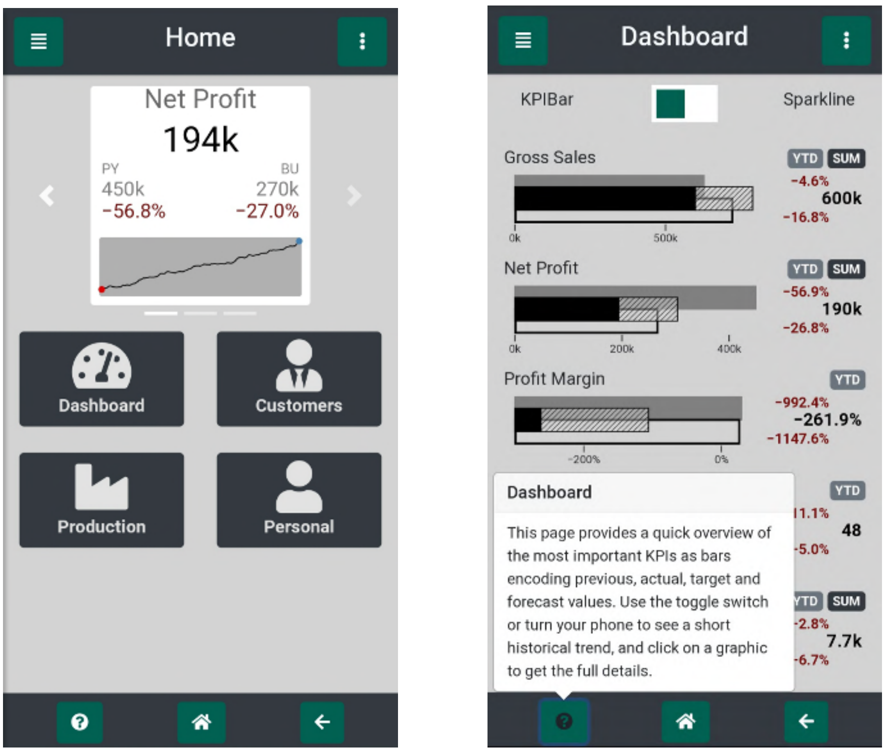
5 Prototype Application

The application developed based on the requirements from the interviews as well as the input from literature is one of the most important results of this thesis, as it allows for a basic feasibility assessment of the overall concept of mobile dashboards and provides a basis for future user evaluations.

For reference, some early drafts of the interface and possible visualizations can be found in Appendix B.

5.1 Application Overview

Since we have only developed a technical demonstration to show how the various requirements could be implemented, the overall structure of the application is mostly defined by the usage scenarios described in section 6.1. However, as those scenarios are derived from the interviews and represent realistic applications of our software, a production deployment could look very similar.



(a) Default home page with quick links to other dashboards and a carousel of KPITiles

(b) The "Dashboard" page with the help opened for further information

Figure 5.1: Basic layout and navigation features

Starting off from a home screen (see Figure 5.1a) that provides information about the most important KPIs as well as short links to various task-specific dashboards, users can either drill down to the former, or jump to the latter for another view. Those dashboards are the main views of the application, and usually combine a list of four to six KPIs relevant to the scenario associated with each dashboard.

Such a list can be seen in Figure 5.1b, where the *KPIBar* visualization is activated. The toggle button at the top of the page allows a user to switch between different chart types that encode various properties of a KPI.

As is quite common in mobile applications and recommended by Shitkova et al. [62], the main menu with a list of all dashboards and supplemental pages can be opened using the button on the top left featuring a hamburger icon, while the context menu that is usually specific to the current page is toggled using the three dots on the top right. The footer contains a page-specific help feature as seen in Figure 5.1b along buttons to navigate to the home screen and the last visited page to allow users to correct accidental interactions.

5.2 Architecture

At a high level, the whole system can be separated into four parts as seen in Figure 5.2:

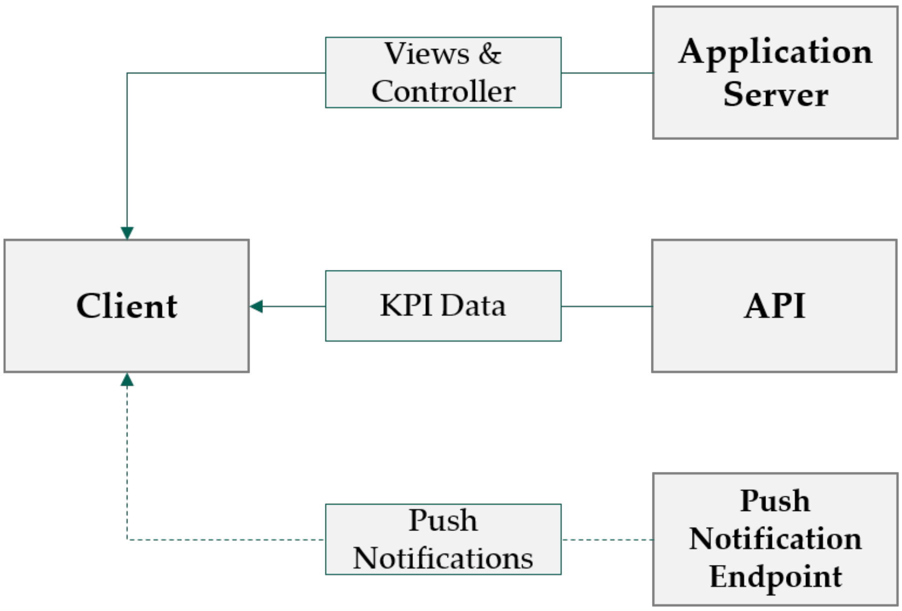


Figure 5.2: High-level overview of the system architecture

On the front-end side is the *Client*, i.e. the actual application responsible for rendering and displaying any content to the end user. Those views, their layout, and styling as well as the code that renders the visualizations is delivered by the *Application Server*. It additionally serves as a standard webserver, performing tasks such as routing and delivery of static assets like images.

The data that should be displayed, e.g. KPI master data as well as the actual measures, is delivered on request by an *Application Programming Interface (API)* so it can be updated independently of the rest of the application. Lastly, a separate endpoint for sending push notifications like Google Firebase or any other service implementing the Web Push protocol would be required in a real world implementation of this prototype, but has been omitted in favour of a purely client side implementation to reduce complexity.

5.2.1 Data Model

To be able to focus on the core parts of interest for the research, mainly the visualizations and interactions, the database used for storing the KPIs master- and transaction data should be as simple as possible. Therefore, the data model was reduced to only three dimensions: the KPI itself, *Products*, and *Partners*, with the latter being a synonym for customers in this specific case.

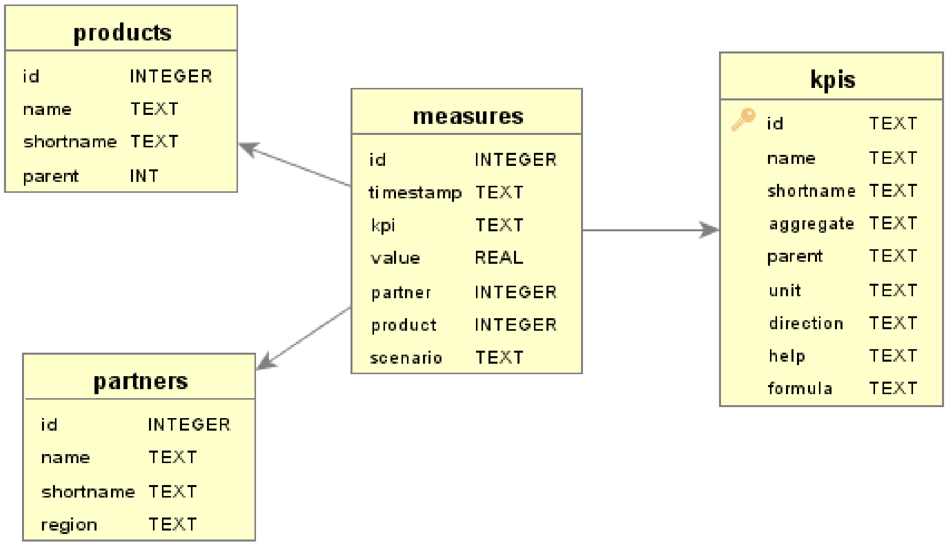
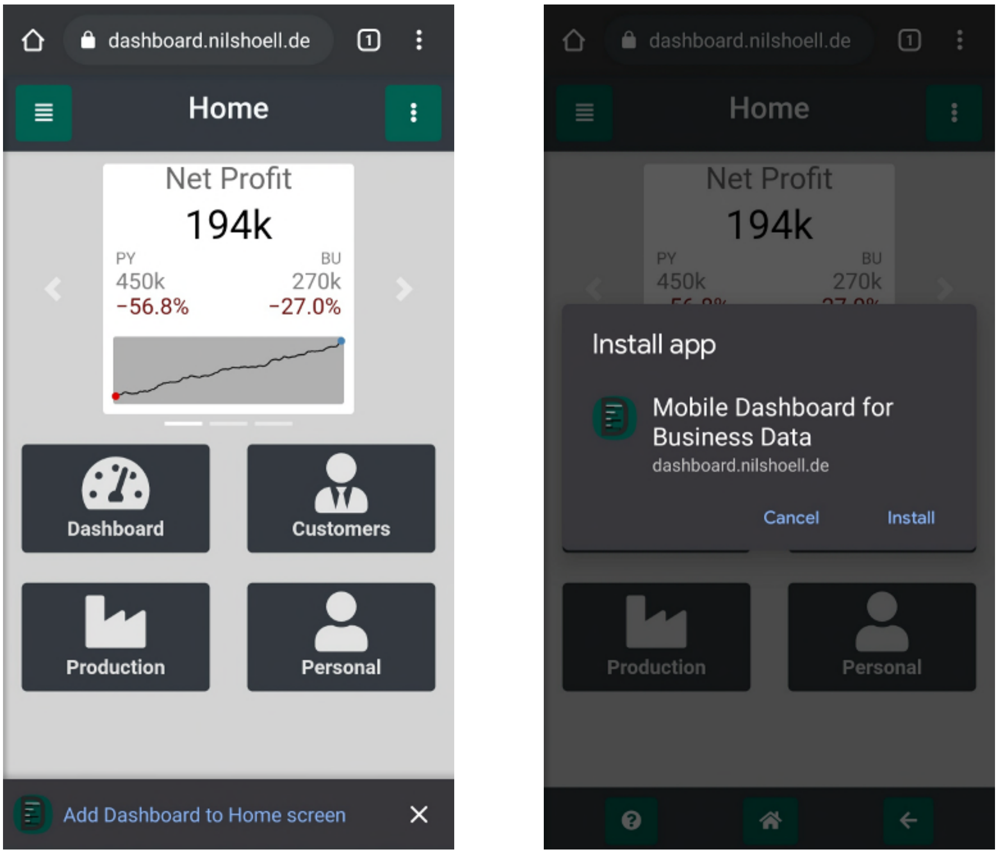


Figure 5.3: Simplified entity-relationship model of the applications database

As seen in Figure 5.3 the resulting model is quite simple, with the tables storing master data about the KPI and the actual measures being the most detailed. The former is the most important, since it contains various important attributes that define how KPIs are displayed and how data is aggregated. The partner and product dimensions are mostly included to show some Drill-Down and filtering capabilities and can be easily replaced or expanded with anything important to the users, such as business units or sales areas.

The used test data was generated for the thesis and is not related to any actual numbers from any of the interviewee's employers. However, at least the selection of KPIs is based on the interviews to get a realistic impression of what a user might want to see, allowing for a better evaluation later on.

A full definition of the data model in SQLite can be found in Listing A.1 in Appendix A.



(a) Chrome on Android displays a hint when visiting the web page

(b) After confirming the dialogue, the PWA is installed to the users App screen

Figure 5.4: Steps to install a PWA with Google Chrome for Android

5.3 Implementation Details

The prototype application shown above was developed using web technologies to ensure an easy cross-platform deployability without the need to develop multiple native apps, as stated in requirement **NFR-2**. The full source code is publicly available on GitHub at `https://github.com/nilshoell/dashboard-pwa` and published under the MIT license[1]. The commit associated with this thesis is tagged as `v1.0` or directly identified by its Commit-ID `57ec73e`. For easy testing of the whole application it was deployed on `https://dashboard.nilshoell.de`, it is however not guaranteed that this page will stay public or won't be updated in the future.

More specifically, we developed a Progressive Web App (PWA), a special type of web application that aims to combine the easy, platform-independent development and deployment of web sites with the deep integration into target devices that usually only comes with native apps. Biørn-Hansen, Majchrzak, and Grønli [6] show that both installation sizes and loading times for PWAs can be significantly smaller than for other cross-platform approaches. Google, which is a large driver behind PWAs, further claims that loading times are also faster compared to traditional websites due to better caching of static assets [29, 52], although scientific evaluations of the framework are scarce.

However, based on a wide availability of more technical sources from both companies

[1]More information about this license at `https://choosealicense.com/licenses/mit/`.

like Google and Facebook as well as countless of other resources like [42, 19, 48, 36], we can conclude that PWAs have the following properties that make it a good choice for this project:

Cross-Platform Work in Chrome and Firefox for Android, Safari for iOS and most other major browsers (requirement **NFR-2**).

Push Notifications PWAs enable native system notifications including custom actions (requirement **FR-1.1**).

Offline Mode Due to access to on-device storage the application can be launched when offline, falling back to the latest available data (requirement **FR-1.3**).

Installable Can be installed on the users home screen like a native app for fast launching as seen in Figure 5.4 (contributes to **NFR-1**).

Caching Assets can be easily cached on the device, accelerating subsequent starts (requirement **NFR-4**).

Fullscreen Mode When launched from the home screen, PWAs can disable all browser-side toolbars leading to a native experience (contributes to **NFR-1**).

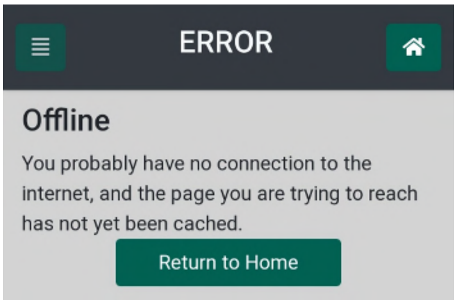
Security PWAs enforce using HTTPS thus guaranteeing at least transport-level encryption (requirement **NFR-3**).

PWAs utilize *Service Workers*, a type of web workers that can intercept or trigger network requests to perform caching, background requests or to display notifications [46]. Especially caching is very relevant to this type of application, as it can greatly increase performance (**NFR-4**) and allows for a full offline usage (**FR-1.3**)[15]. Loading times for a standard dashboard were quite high in the beginning, due to the large amount of single API requests that were executed in a blocking manner. By using asynchronous fetches and parallelizing the requests and chart drawing initial load times could be reduced from over 4.5 seconds to around two seconds. This was further reduced to less than 500 ms by implementing an aggressive caching strategy as seen in listing 5.1.

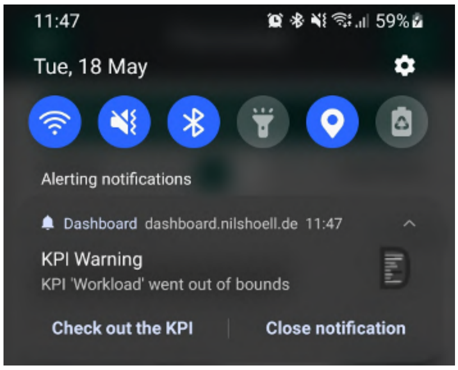
Listing 5.1: Service Worker Caching

```
self.addEventListener("fetch", (event) => {
  event.respondWith(
    caches.open("dashboard-pwa-cache").then(async (cache) => {
      const cache_response = await cache.match(event.request);
      return cache_response || fetch(event.request)
        .then((network_response) => {
            cache.put(event.request, network_response.clone());
            return network_response;
        }).catch(() => {
            return caches.match("/offline.html");
        });
    })
  );
});
```

The service worker intercepts network requests using the `fetch` event listener, and will try to load the resource from the cache before falling back to an actual network request.



(a) The hint that is shown when a user visits a non-cached site while being offline



(b) An example of a system notification on Android

Figure 5.5: Different features of service workers

The following response is then put into the cache so subsequent requests to the same Uniform Resource Locator (URL) are fetched from cache. While browsing the various dashboards and pages, the total cache size amounts to around 4-5MB in total, so the applications size should never become a problem. If the device has no network connection all previously visited pages can still be opened, while the rest simply displays an offline notice as seen in Figure 5.5a.

However, this aggressive caching comes with a downside: Once visited, every page will look exactly the same, even if the data for the visualizations changes on the server, since no more requests are send over the network. In a production-ready application this could be easily solved by either regularly purging data caches, background requests to periodically update data using service workers, or implementing the "Cache then Network" strategy, meaning that data is provided by the cache first, but a network request is still performed to update shortly after [28].

Those techniques require a lot of fine tuning to the specific application, so for our prototype we simply implemented a button on the "About" page to manually purge the data cache.

As stated above, another interesting component of service workers is the *Notification API* that allows to send native system notifications to the end device, that can even be triggered if the application is closed [45]. To show that the technology is capable of fulfilling requirement **FR-1.1** that is very important especially in the monitoring use case, a simple demonstration was implemented on the "Personal" dashboard. To speed up development the feature was only implemented on the client side, meaning there is no server endpoint that externally triggers a notification, since such a setup would have been too complex for this thesis. Nonetheless, after enabling notifications users can trigger them themself with a simple button and subsequently interact with them as seen in Figure 5.5b, for example to open the detailed page of a KPI that went out of bounds.

5.3.1 Technologies

As described above, the prototype was developed as a PWA building on various web technologies as seen in table 5.1. Most notably, the application server and API endpoint are driven by Node.js to enable both frontend and backend development in JavaScript or, as in this case, TypeScript. To store the test data, a small SQLite database file is delivered with the software and hard-coded into the backend to allow for fast deployment without a separate database server, while still providing an easily adaptable SQL interface to access the KPIs.

On the client side, basic layouting and styling is provided by Bootstrap, while the actual visualizations are implemented using Data Driven Documents (D3).

Table 5.1: Core technologies and dependencies of the project

Name	Description	Version	Homepage
Node.js	JavaScript Runtime	`15.14.0`	`https://nodejs.org/en`
D3.js	Visualization Library	`6.7.0`	`https://d3js.org`
Pug	Templating Engine	`3.0.2`	`https://pugjs.org`
jQuery	JavaScript Library	`3.6.0`	`https://jquery.com`
Bootstrap	CSS Framework	`4.6.0`	`https://getbootstrap.com`
SQLite	Database	`5.0.2`	`https://sqlite.org/index.html`
TypeScript	Typed JavaScript	`4.2.4`	`https://www.typescriptlang.org`

The first attempts at implementing the application were conducted using React, a User Interface (UI) framework developed by Facebook that is often used for PWAs [19]. However, due to both React and D3 manipulating the Document Object Model (DOM), the combination of those frameworks presented immense challenges especially when trying to re-draw graphics after resizing the screen or switching between visualizations. Since D3 is vital for developing custom visualizations, React was dropped in favour of a more standard web development approach using Pug.js for HTML templating.

5.3.2 Visualizations

Being the core part of communicating relevant information, a special focus was put on different types of visualizations. While most of them are well established both in literature and practice, they had to be adapted for use on smaller displays.

In general, we tried to reduce the amount of labels, annotations, and axes to a minimum to avoid cluttering the display and allow the user to focus on the actual information displayed. For the same reason colors have only been rarely used; mostly to highlight specific points of interest or deviations as proposed by Bera [5]. Instead, the encodings were selected as described in the IBCS by Hichert and Faisst: bars that represent actual data should be solid black, previous years are to be encoded with solid grey, budget or target data uses only a black outline, and forecasts should be displayed as hatched [33, pp. 7–11]. All of those variations can be easily distinguished even by people with impaired color perception, and are, according to multiple interview partners, virtually standard in business contexts, so the target audience will understand the different meanings out of the box. In the few cases where colors were used to actually encode information only the lightness and not hue was varied, again to ensure color perception is not a limiting factor.

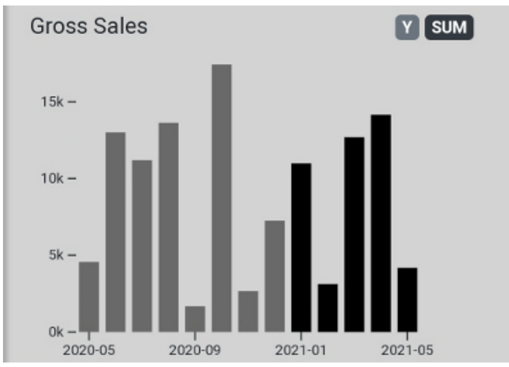
To add some context about filters and time frames as specified in requirement **FR-3.5** to all visualizations a couple of standardized labels were deployed: one in light grey encoding the time period currently displayed, using industry-standard abbreviations such as `MTD` for *Month-to-Date* or `Y` for a full year of data. The other, optional label in dark grey shows whether the data was aggregated over said time frame (`SUM`), or if a moving average was applied. The latter is only relevant for line charts and displays the number of days over which the moving average was calculated, for example `Ø 5d` for a five-day moving average.

As the focus of this thesis is on smartphone usage, all visualizations and their annotations were designed for screen sizes between 4.5" and 6.7" covering almost all current smartphones [68, 53]. While the application can be used in landscape orientation, most of the optimization was done for portrait mode, since the devices are used in that orientation for about 85% of the time [56].

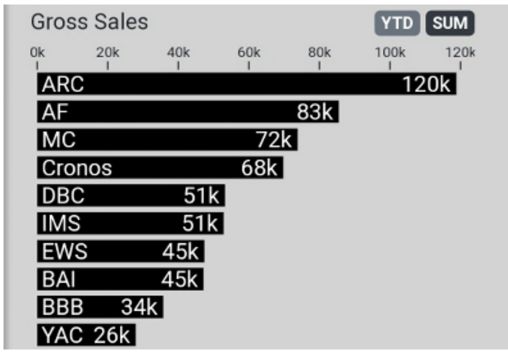
The implementations of the different visualizations described in alphabetical order below are located in the repository under `/src/public/js/charts/`.

Bar/Column Chart

Bar charts are probably the most important type of visualization in business dashboards, at least based on the insights from the interviews, and generally easily understood by all audiences.



(a) Bar chart displaying sales grouped by months



(b) Column chart showing sales of the current year broken down by customers

Figure 5.6: Bar- and Column chart samples

However, they face a couple of problems on smartphone screens, especially in portrait mode, since the restricted horizontal screen space limits the number of categories to be displayed. Therefore, column charts as presented in Figure 5.6b should be used instead [58], enabling better label placement parallel to or inside the bars, and a much larger amount of categories that would only require vertical scrolling. The ability to place the axis containing the value on top of the chart instead of the side is also more space-efficient for the same reason.

One exemption was made for data that is aggregated over time, for example by months as shown in Figure 5.6a, since users expect time to be plotted on the x-axis [33, p. 12] and the number of possible categories can be better controlled.

Brick Wall

As described by [49], a *Brick Wall* is a calendar-based heatmap visualization, encoding values using differently colored shapes that are aligned in columns for each week.
This approach allows for a high information density, since almost every pixel of the chart area is used, providing a good overview of comparatively large timespans as seen in

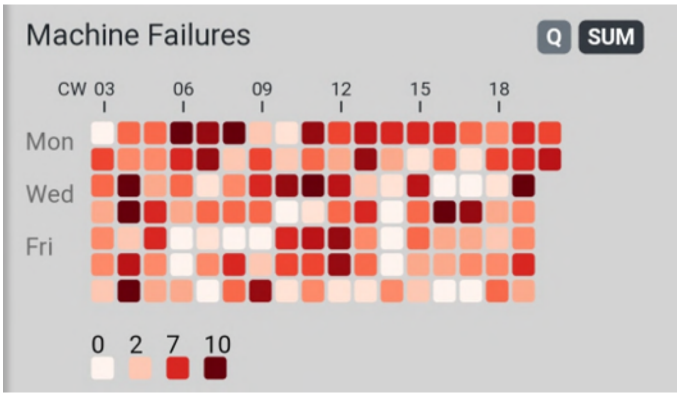


Figure 5.7: Sample Brick Wall visualization displaying the daily frequency of machine failures over the last quarter

Figure 5.7. Due to the alignment of week days in rows, getting insights about periodical effects is quite easy. However, since only color is used to differentiate between values, exact readings and comparisons of non-adjacent shapes can be hard, especially with large ranges of values.

KPIBar

The *KPIBar* is a special type of visualization developed based on the requirements from the interviews. It displays the current value of a single KPI along with the most important reference points, i.e. the current target and a historical value, usually the last year or month depending on the selected time frame. Therefore, this representation is vital to fulfill the requirements for analyzing deviations (**FR-2.1**) as well as for providing information at a glance (**FR-3.1**), while it additionally includes forecasts (**FR-3.4**).

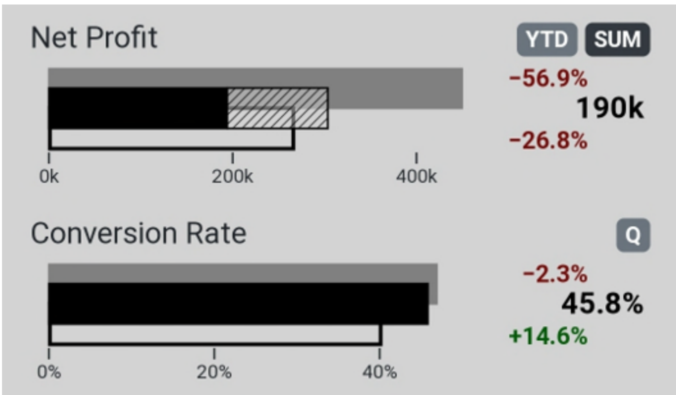


Figure 5.8: KPIBar visualizations showing both absolute and relative numbers over different time periods

While being based on Fews *Bullet Graph* [21, p. 151], the use of distinct bars instead of symbols to mark comparison values makes the distinction easier at a glance as seen in Figure 5.8, especially since the colors and styles of the IBCS could be applied.

By placing the bar with the actual value on top of the others it's importance is emphasised while improving space efficiency. The positioning of the comparison bars is also not arbitrary: The top one, and therefore the first to see, represents the value of the previous period and therefore also the earliest value based on time, while the future-oriented target or budget is below. Additionally, a semi-transparent forecast value can be added to the right of the one for actual data, making it easy to assess the predicted outcome of a KPI compared to its target.

To allow for exakt readings, labels for all three bars were placed on the right side of the screen at the respective height for each bar. However, the comparison values are only displayed as deviations in percent of the actual one, since relative deltas are the most important measure according to the expert interviews.

KPITile

While not a visualization itself, the usage of tiles to represent single KPIs has been recommended both in literature [2] and in some of the interviews (#08, #18).

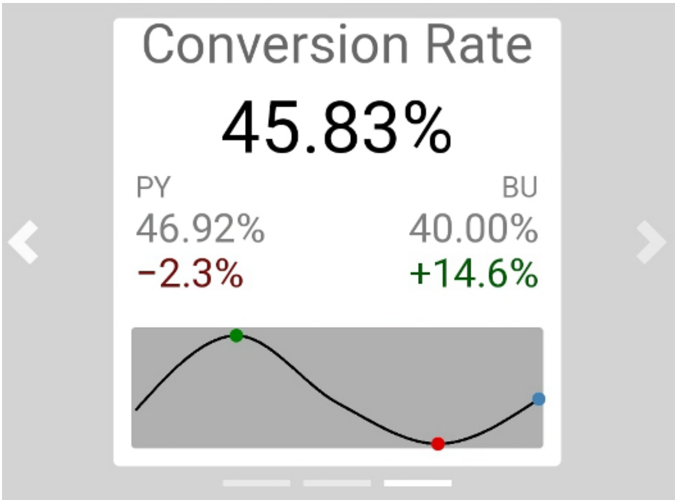


Figure 5.9: KPITile displaying current, previous (PY) and target (BU) values along a short history

As with the aforementioned *KPIBar*, in addition to the current value, both previous and target values are displayed on the tile (see Figure 5.8). To get a quick reading of recent history, a small Sparkline including annotations for minimum and maximum values in the selected time frame is displayed underneath the raw values.

Due to their quadratic form factor, KPITiles have the advantage of being perfect for use in a carousel as advised by [58], allowing a user to quickly swipe between the most important KPIs. Using multiple tiles at once in a dashboard is however not advised, since the combination of KPI name, five numbers, and a Sparkline might result in a cluttered view if used too often.

Sparkline

Since their first description by Tufte [67] in 2006, Sparklines have often been used to visualize trends in dashboards. However, due to their size, they are often used without any labels [21, 4], making it harder to see how much the measurement has actually changed, and over what time frame.
We therefore tried to find a middle ground between the high data density and glanceability of traditional Sparklines and the precision and context of standard line graphs by adding minimal axes and labels. Since the goal of these visualizations is not to be able to read a value on a specific date, but rather to get an impression of how the KPI

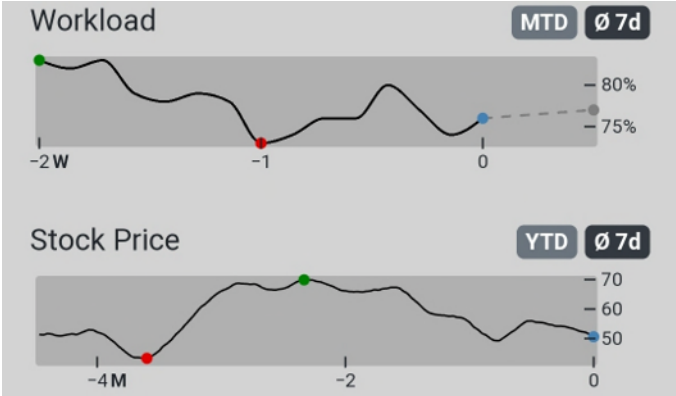


Figure 5.10: Sample Sparklines visualizing absolute and relative KPIs over different time periods including a forecast

developed over time, the x-axis only shows how far the data reaches back in time. This is achieved by converting the number of days displayed into either weeks, months or years depending on the time frame, and only showing that number. For example, if the Sparkline shows 30 days of data points, this is represented by a `-1M` on the left side of the axis, which allows for a very easy assessment of the time component of the data without the need to display long date labels. The code used to convert regular dates into this specific format can be found in Listing A.2 in Appendix A.

For reference, Figure 5.10 shows two weeks of data for the top chart, and almost five months on the bottom one, with both graphics rendered as a seven-day moving average of the original data as indicated by the label `Ø 7d`. To encode future trends a dashed line, visually similar to the hatched bar of a KPIBar, can be drawn after the latest data point, taking up to 25% of the Sparklines horizontal space, as seen in the top visualization of Figure 5.10.

To retain context for the value and its volatility, at least two labels are displayed on the y-axis, allowing for an estimation of the values span while still being as compact as the original Sparklines. Additionally, three points are used to mark the minimum, maximum, and current value on the chart.

Timeline

For those cases where the above mentioned Sparklines are not sufficient and enough space is available, a traditional line graph can also be used to visualize time-series data. Timelines have the benefit of providing more details especially in the vertical axis, and therefore allow for a better analysis of precise values.

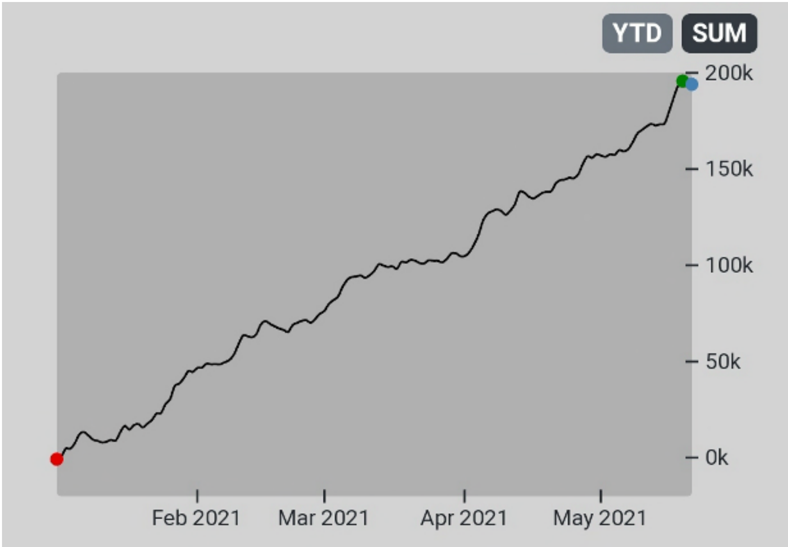


Figure 5.11: An example of a Timeline showing almost five months of data

Additionally, they provide a more realistic context for the amount and rate of change over time since the vertical axis starts at zero as opposed to the minimal value like in Sparklines. Nonetheless, the Timlines as implemented in this application are mostly large versions of their Sparkline counterparts, including the same annotations for minimum, maximum, and current values as well as the value labels on the right to ensure the latest values can be read with the highest precision.

5.3.3 Interactions

Most interactions implemented are standard for mobile phones and therefore intuitive, such as scrolling, opening the hamburger menu on the top left, or swiping to change the KPI presented on the home screen. Additionally, a long touch interaction was implemented on charts to display further details like master data about the KPI along with the current filters on a pop-over modal as seen in Figure 5.12. According to Chang et al. [10], the long touch is a standard interaction, so users should have little trouble discovering this feature.

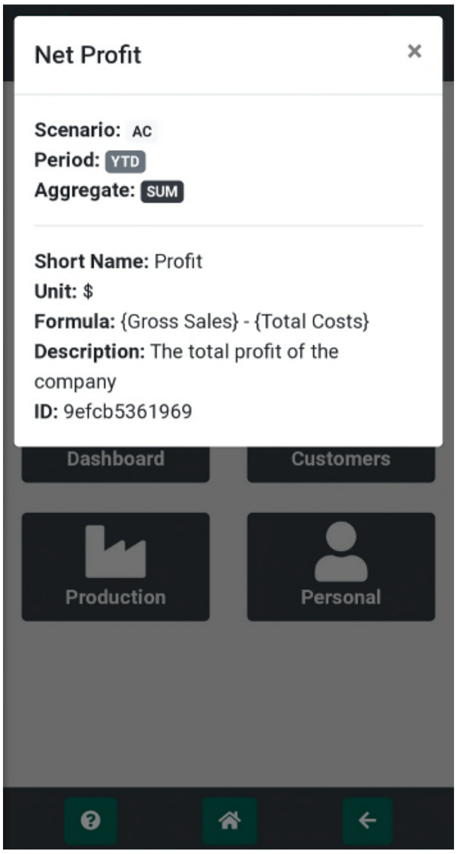
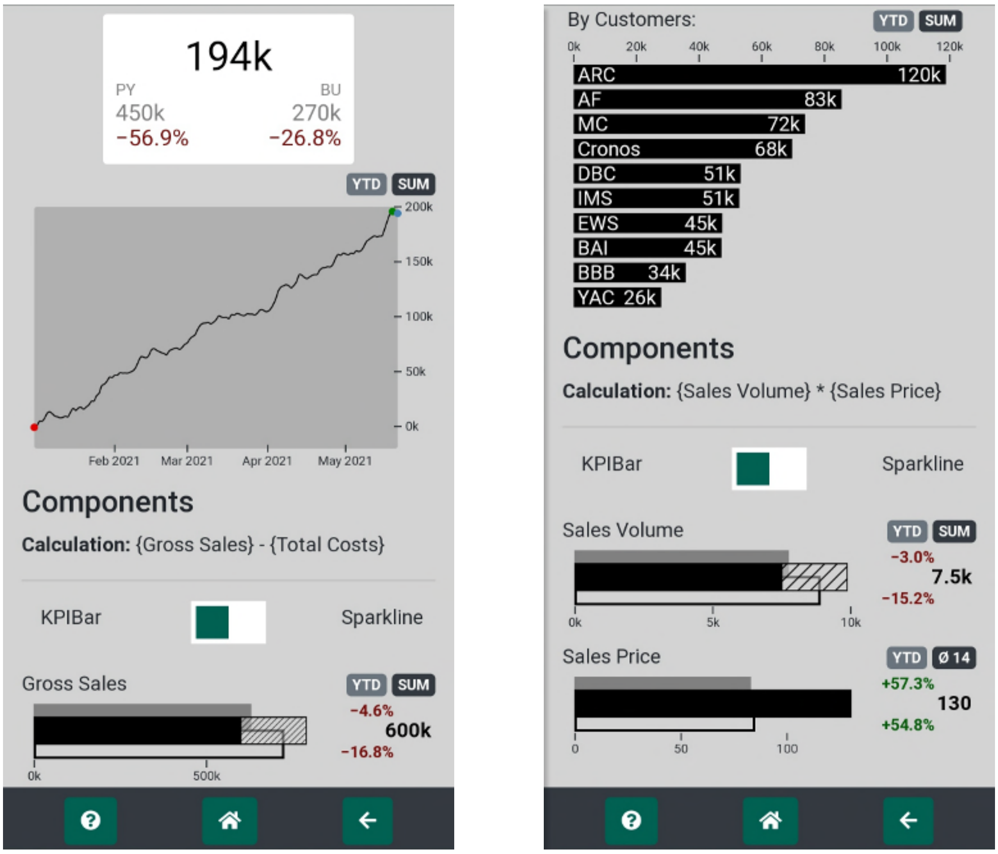


Figure 5.12: A modal triggered by a long touch on the visualization that quickly shows further details

To implement Drill-Downs as specified in requirement **FR-2.3** a simple click, or rather touch interaction is used. In general, a user can tap on any visualization to open a detail page (see Figure 5.13) of the respective KPI that will display actual, previous years, and budget values above a timeline visualization, the data split up by customer if applicable, and the components of which the metric is composed as shown in Figure 5.13b. One exception to this rule is the "Customers" dashboard, where a click on one of the bars drills down to a detail page of the associated customers, which is to be expected on a page specifically designed to show detailed customer information.



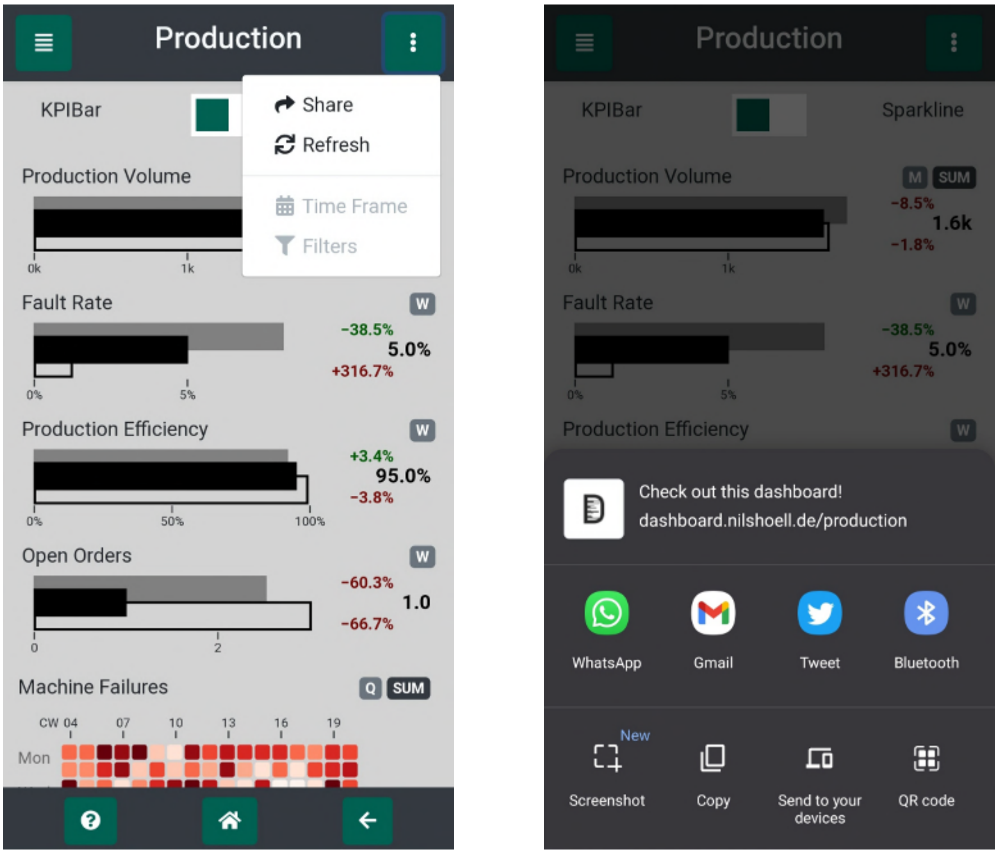
(a) Detail page of the Net Profit KPI with a simplified KPITile on top

(b) Bottom part of the Gross Sales KPI, showing the data broken down by customer as well as the KPIs components

Figure 5.13: Detail pages of different KPIs

Another important aspect of interacting with the application is the option to easily collaborate with others, a feature often requested in the interviews (▌**FR-2.2**) and further highlighted in literature [64, 74]. This was realized using a simple "Share" button in the context menu that will open the devices native sharing functionality, or simply copy the current URL to the clipboard if the former is not supported, as seen in Figure 5.14. Since every single view has its own unique URL, sharing that adress is sufficient to let a receiver open look at the same data and visualizations.

The context menu also implements a "Refresh" button to manually update the displayed data, as well as placeholders for future, dashboard-wide settings.



(a) The context menu with "Share" and "Refresh" buttons

(b) Native share-menu on Android that opens after klicking the "Share" button

Figure 5.14: Sharing dashboards using native operations

6 Evaluation

To assess how well the requirements found in Chapter 4 could be met by the prototype and ultimately answer the research questions, our findings and the resulting application are evaluated in this chapter.

6.1 Application Scenario Evaluation

To allow for an in-depth evaluation of the applications usefulness, four different possible application scenarios are used as a reference. Those scenarios represent realistic situations derived from the interviews, and are aligned to the more abstract use cases as presented in Chapter 4.2. As *Easy Access* is the most important use case we developed two separate application scenarios for different audiences to get more detailed results, while the other use cases have one scenario each to get broad insights into all relevant situations.

Each of the application scenarios consists of a short description and a series of tasks that a user might need to perform on a mobile dashboard. To ensure a high practical relevance of the scenarios, a feedback session on the latter and the application itself was conducted with another expert from company **L** that was not available for the initial interviews. The participant, an executive from the financial department, confirmed that the scenarios are indeed realistic and proposed some changes to the KPIs displayed on the various dashboards that have been implemented prior to the evaluation.

The tests were conducted on a Samsung Galaxy S10 running Android 11 with both Mozilla Firefox (version `88.1.3`) and Google Chrome (version `90.0.4430.210`) and an iPhone 6s running iOS 14.4.2 with Safari 14 as a browser to test cross-platform functionality.

6.1.1 Application Scenario I - Personal Targets

As described in section 4.2, increasing user engagement with data by providing a very easy and quick access is one of the most important uses for mobile dashboards. Therefore, this application scenario targets standard employees that would normally not interact with BI systems. The main goal for those employees would be to stay informed about their personal targets as well as information related to their department, team or project, i.e. topics immediately relevant to their daily work.

Application Scenario Definition

Name Personal Targets

Use Case Easy Access

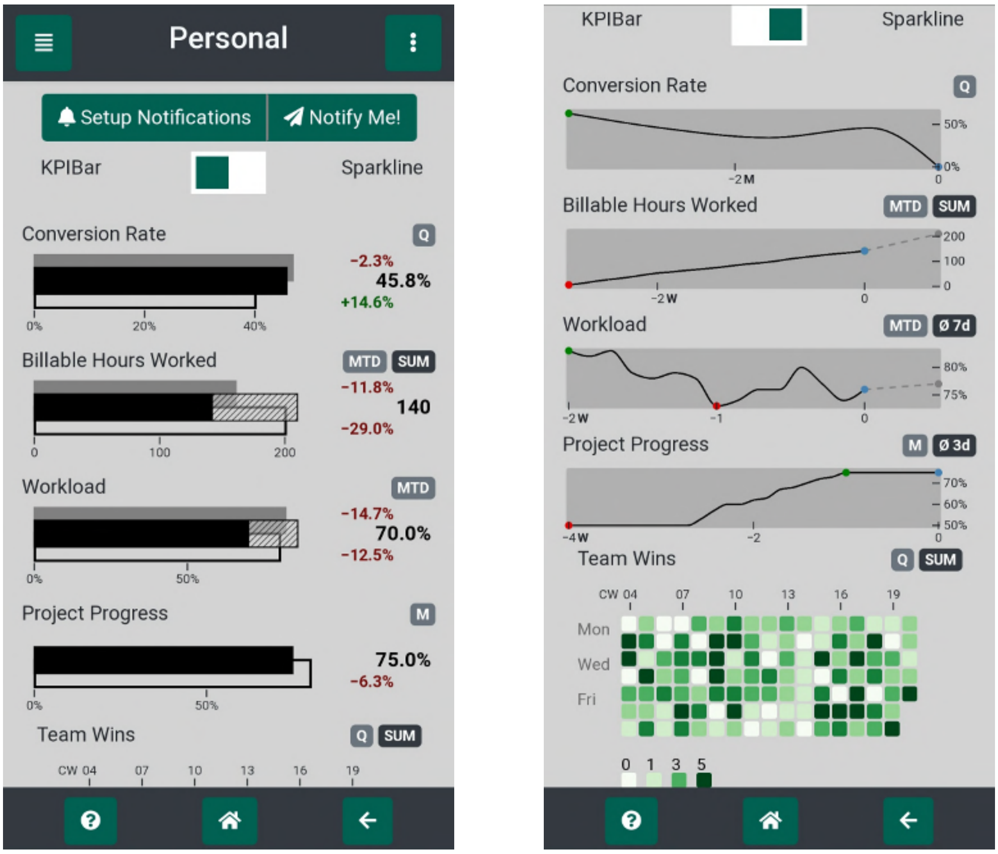
Description The user wants to monitor personal targets and KPIs relevant to their department

Audience Employees, Middle Management

Tasks

- Get an overview of current targets
- Compare to historic and actual values
- Check the projects status
- Get notified on a regular basis

The dashboard designed to fulfill those tasks can be reached from the home screen or menu under "Personal" or directly via the relative URL `/personal`. Since the actual metrics depend on the industry and department we decided to provide a mix to show the versatility of the application: The *Conversion Rate* is a standard KPI for sales people, while *Billable Hours* and *Workload* are often used in consulting or other project-driven work like (software) development. To get informed about the overall status in the department, the *Team Wins* (i.e. new customers) and a simple project progress indicator are used.



(a) The personal dashboard to display individual targets using KPIBars

(b) Switched to Sparklines as main visualization

Figure 6.1: Overview of the Personal dashboard

The default visualization for the KPIs displayed on the dashboard is the *KPIBar*, that immediately allows to see current values, targets, and historical data for each indicator (see Figure 6.1a). The user can compare their performance to goals and the previous year at a glance for each of the metrics at once. Additionally, the switch on top of the page or turning the phone into landscape mode will reveal Sparklines (Figure 6.1b) to get even more historical context as well as a recent trend.

As seen in Figure 6.1b, one can easily identify that the Workload dropped to a minimum of less than 75% a week ago, and is projected to slowly rise in the next few days. The difference of the current value as seen in the KPIBar (70%, Figure 6.1a) versus the Sparkline (over 75%) is due to the 7-day moving average used to smoothen the curve, indicated by the label reading `∅ 7d` above the Chart. A different view is used for visualizing the Team Wins (see Figure 6.1b at the bottom): in this case, a Brick Wall is used since the span of values is rather small and color encoding therefore useable.

The top of the page contains buttons to set up (i.e. request permission for) push notifications and to send one. In a production environment those buttons would not be used, as push notifications would be send by an external notification endpoint, so those are only implemented to show the technical capability of displaying native push notifications. In the current scenario, notifications could be used to increase engagement, for example by reminding users to check their dashboard, or to alert them if one of the metrics goes out of pre-defined bounds.

6.1.2 Application Scenario II - Corporate Overview

The second scenario was tailored to top management like managing directors or executives, and therefore needs to display information at a highly aggregated level. While the whole company and its performance is the main concern of the audience, simple Drill-Downs to get more details about where deviations are coming from are also important.

Application Scenario Definition

Name Corporate Overview

Use Case Easy Access

Description The user is interested in the overall performance of the enterprise as well as trends

Audience Top Management

Tasks

- Check all important KPIs at once
- Quickly switch to a line chart displaying history and trends
- Drill-down to assess any deviations
- Share certain views for more detailed analysis

The page implementing the main view for this scenario is simply called "Dashboard" on the home screen and main menu and can be directly accessed using the URL `/dashboard`. Again, this dashboard utilizes KPIBars (see Figure 6.2) to display the five most relevant KPIs at once, encoding previous years, actual, budget, and possibly forecast values in a compact visualization, enabling quick comparisons between important references. Switching to Sparklines to see historic development and therefore recent trends is possible using a single toggle switch, while Drill-Downs are possible by simply clicking on a visualization.

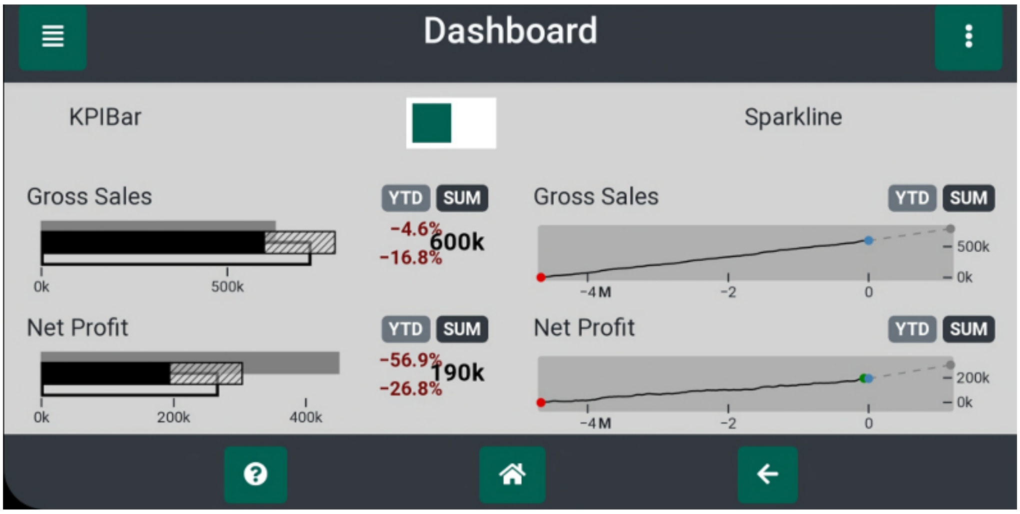


Figure 6.2: The main dashboard as seen in landscape mode, showing both KPIBars and Sparklines

Although those detailed views (for example `/kpi/9efcb5361969` for Net Profit as shown inf Figure 5.13a) provide quick insights into the sub-components which might be the cause for a deviation, they are currently lacking any sort of filtering or changing the base dimension. An important feature could be displaying a KPI split up by customers, sales areas or product groups, and filtering along those dimensions. Due to time constraints we were only able to implement an overview by customer for the measures that have partner information assigned in the database, but the interaction should be as easy as a toggle switch.

Lastly, sharing the current view is possible using the context menu and the "Share" button that will open the devices native sharing menu. Deep links shared using this method open directly in the app if it is installed on the target device, showing the same view as seen by the sender.

6.1.3 Application Scenario III - Production Monitoring

For the monitoring use case a production scenario seemed to be the most realistic one based on the expert interviews, since especially in mass production any incident like a machine failure might require immediate response. Therefore, push notifications are a critical feature in this case to allow for a close and rapid monitoring of KPIs. In addition to that, the default time frame of all metrics would be a lot smaller, since the most recent numbers and trends are also the most relevant. However, the ability to visualize certain data over a larger time period might still be useful to detect patterns, for example with Machine Failures.

Application Scenario Definition

Name Production Monitoring

Use Case Monitoring

Description The user wants to monitor production efficiency, fault rates, and incidents.

Audience (Production) Management

Tasks

- Get an overview of all relevant metrics over the last week
- Detect anomalies and analyze the deviation
- Check the number of incidents over the last month
- Stay notified about incidents
- Share certain views for more detailed analysis

Since both the notification and sharing features have already been described in Section 6.1.1 and Section 6.1.2 respectively, we will omit them here. The production dashboard is located in the application under "Production" or directly under the URL `/production`. Aside from the Production Volume the main KPIs are displayed for the last week to allow for a closer monitoring as seen in Figure 6.3. This can be easily determined by the indicators on the top right of the KPIBars (`W`), that also provide the information that the Sparklines values are displayed as a three-day moving average to smooth out some extremes and show trends (`Ø 3d`).

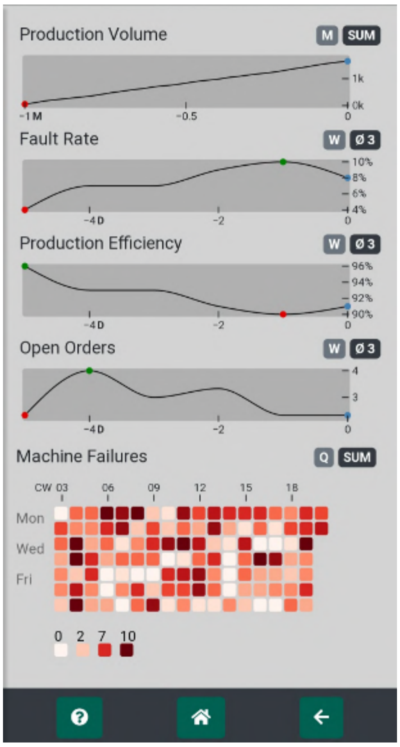


Figure 6.3: The Production dashboard in Sparkline mode

Finally, we implemented a Brick Wall to show the number of incidents (Machine Failures) for each day over the time of a quarter year. This is especially useful, as this view allows to asses the overall situation, trends, and specifically periodical occurrences of failures, for example every friday. Ideally, clicking on this visualization would lead to a detailed view specific from which machines these failures originated, but our simplified data model does not allow storing this information.

6.1.4 **Application Scenario IV** - Customer Meeting

Regarding the Meeting Preparation use case, the scenario of a traveling sales person that has to prepare for a customer meeting was by far the strongest in the interviews, and therefore seems like a suitable test case. It is rather short as compared to the others, but relies on the feature of *Entity Dashboards* (**FR-3.3**) to display various information about a specific customer or partner.

Application Scenario Definition

Name Customer Meeting

Use Case Meeting Preparation

Description A sales person in the field wants to prepare for a meeting with a customer

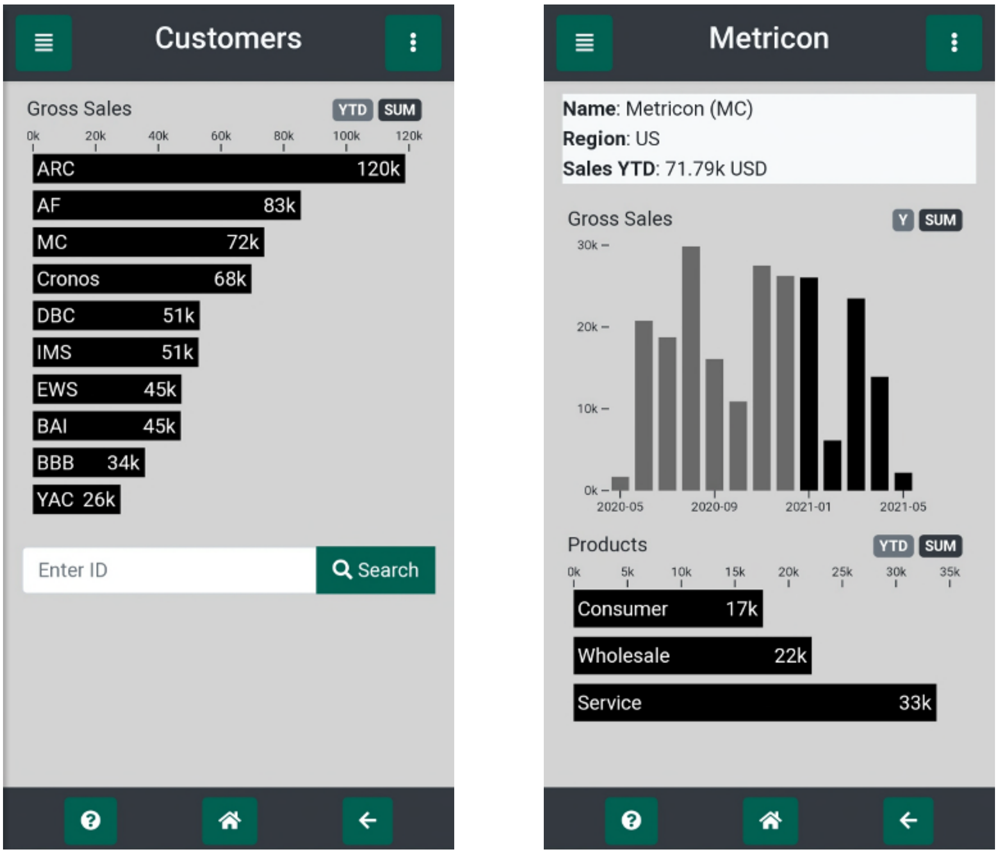
Audience Employees

Tasks

1. Get an overview of all assigned customers for context
2. Open details for the specific customer
3. Compare planned against actual sales volume
4. Compare to historic values

The entry point is located under "Customers" on the home screen, or directly under the URL `/customers`. In the first view (Figure 6.4a) the users sees an overview of all their assigned customers, sorted by the gross sales they generated in the current financial year. By either clicking on one of the bars or entering a partner ID into the search field a detailed view of that entity is loaded, which can be seen in Figure 6.4b.

It consists of a Bar Chart displaying the total gross sales grouped by month over the last year, and a column chart where the same KPI is split up by product groups. The use of a Bar Chart despite its problems as described in section 5.3.2 is justified in this case, as the x-axis represents time, and the number of possible values is fixed to 12.



(a) Summary of all customers ordered by Gross Sales YTD on top of a search field

(b) Detailed customer view with Gross Sales by month and product group

Figure 6.4: Overview of the Views Associated with the Customer Meeting Scenario

6.2 Requirements Fulfillment

The above section already evaluates how most of the requirements from Chapter 4 are met in the application, so the focus in this chapter is on providing an overview as well as highlighting open challenges.
Figure 6.5 shows that the majority of requirements, especially from the visualization category, could be implemented in the prototype.

While usability is quite important as discussed earlier, it is hard to evaluate using the application scenarios. As mentioned in Chapter 4, we used the work of Shitkova et al. [62] as a reference for designing many aspects of the interface, so it makes sense to also utilize their guidelines for evaluation. To keep this assessment short and concise, we will only discuss the items most relevant to this application, since a full usability evaluation would require a user study anyway. Shitkova et al. split up their guideline into four main categories as seen and summarized below.

Category		Requirement	Name	Fulfilled?
Functional Requirements	General Requirements	FR-1.1	Push Notifications	Yes
		FR-1.2	Personalization	No
		FR-1.3	Offline Usage	Yes
	Interaction Requirements	FR-2.1	Deviation Analysis	Yes
		FR-2.2	Collaboration Features	Yes
		FR-2.3	Drill-Down	Partly
		FR-2.4	Time Frame Selection	No
		FR-2.5	Dimension Selection	No
	Visualization Requirements	FR-3.1	Information at a Glance	Yes
		FR-3.2	Trend Lines	Yes
		FR-3.3	Entity Dashboards	Yes
		FR-3.4	Forecasts	Yes
		FR-3.5	Data Context	Yes
		FR-3.6	Top/Bottom Lists	Partly
Non-Functional Requirements		NFR-1	Usability	Yes
		NFR-2	Platform-Independent Use	Yes
		NFR-3	Application Security	N/A
		NFR-4	Performance	Yes

Figure 6.5: Overview of all requirements and their fulfillment

Layout Content should be arranged vertically and on separate pages, to avoid using tabs or horizontal scrolling.

Navigation Only one level, simple navigation menus should be used, with important pages directly reachable from the start page. The latter should be accessible from everywhere, and the current location should be visible.

Design The overall design should be simple and consistent, utilizing easily identifiable icons and large enough buttons.

Content The start page should not show too much information, while content on other pages should be ordered by its importance. Any user actions should be quickly revertible, and the state of the application should be saved when closing to be restored on later use.

Most of the remaining guideline items that have not been summarized above are concerned with login, editing, forms or longer text, and thus not applicable to our application. The only relevant suggestion the prototype cannot fulfill is restoring the last open page after it has been fully closed. However, we consider this to be a minor problem, and therefore conclude that the application meets the usability requirement as far as we can judge that.

Most of the other open requirements actually track back to the time constraints of this thesis as well as our data model, and are discussed in detail below.

Personalization (FR-1.2)

While both the interviews and literature have shown that personalization is very important for a mobile dashboard application, it had to be moved out of scope for the prototype. A useful configuration feature would be user-specific, which would in turn require some form of user authentication and management. Implementing a simple mock up would have been possible, but the additional scientific insights were considered to be rather few, so focusing on the development of the visualizations and interactions was prioritized.

Time Frame Selection (FR-2.4)

Changing the time period for individual visualizations might not be the most important, but still a relevant feature. The ability to select the time frame is essentially a zooming functionality for all time-series charts and an easy filter for the rest. The labels showing the current period (see Section 5.3.2) were designed as an entry point to such a selection, and initial feedback confirmed that users would expect to be able to switch time frames by clicking on it.

To keep the selection lean and fast, only a pre-defined set of periods such as `M`, `MTD`, `Y` or `YTD` should be possible, as picking individual start and end dates would require much more complex interfaces. Unfortunately, the feature could not be implemented due to time constraints for the development phase.

Dimension Selection (FR-2.5)

In most of the visualizations for the above usage scenarios the displayed KPIs are represented as single, total values, or over time. However, there are some cases in which data is broken down by customer or product group to show more details. Being able to switch that base dimension would add quite some flexibility to the analytical capabilities of the dashboard, without adding too much complexity.

Similar to the time frame selection presented above, a simple label showing the current base dimension could indicate the possibility to select another dimension and serve as an entry point for such a selection. Based on the interviews the range of those dimensions is usually quite small, with participant #15 stating that four to five should be enough for a good data model.

When adding options to sort the data (completing requirement **FR-3.6**) and maybe even drill down along, for example, a product hierarchy, the application could provide lots of features usually present in a desktop BI system. This would in turn increase the complexity of interactions and possibly clutter the interface, making it less visually pleasing. Therefore, such advanced functionality would not be present on the initial dashboards, but rather only on detail views, thus moving the focus away from glanceable visualizations. Based on the interviews, most users would not need or even want any detailed analytics options, since those tasks are better performed on a PC anyway. Therefore, although being interesting in general, any such features were not considered for implementation in the prototype, but might be relevant in the future if they do not impact the default dashboards.

Platform-Independent Use (NFR-2)

In general, the requirement for cross-platform usage of the application was met by implementing it as a PWA. The web technologies are well standardized and widely supported in both iOS and Android, and on all major browsers on those platforms. Most important, service workers, the basic requirement for any PWA feature, have been implemented in all relevant mobile browsers reaching over 96% of the users [46, 9].

Due to the constraints in development time, the focus of optimizations was on Chrome for Android, since it could be easily debugged by the authors. This lead to a couple of minor bugs on iPhones, such as the hatched pattern for encoding forecast values on KPIBars not being rendered, or a worse performance when switching to Sparklines on the dashboard views. The only bigger problem is the fact that as of now, no browser on iOS supports the Web Notification API [45] used to send native push notifications, which means that feature does not work.

6.3 Discussion of Results

Initially, our research was driven by two main research questions. To answer the first one, "*How and in which scenarios can mobile devices support data-driven decision making?*", the expert interviews provide a good answer: mobile dashboards can indeed be useful in various, very different use cases as presented in Section 4.2. Most notably, the ability to provide an easy and intuitive access for virtually all employees to information most relevant to them is a convincing scenario. In the remaining use cases, the deployment of mobile dashboards would have a more direct impact on the performance and efficiency of employees, since the time it takes to get important information or alerted after an incident would be reduced.

For the second part, "*How can dashboards for business data be realized on mobile devices?*", the prototype application is one of many possible solutions. It shows that with modern smartphones and standardized web technologies, it is definitely possible to bring informative yet simple, high-resolution visualizations on to comparatively small screens. By adapting existing and introducing a new visualization optimized for smartphones the ability to communicate important information is further enhanced.

6.3.1 Relevance

Due to 20 interviews with experts from different industries and departments, with varying grades of experience in designing and using dashboards, a high practical relevance is ensured. This is further supported by the initial literature review that shows a significant gap in recent research into this field. With mobile devices being the dominant form of computing [70] and a clear demand by the experts for usable mobile dashboards our results are quite relevant for both scholars and practitioners.

6.3.2 Validity

As the interviews are a core part of the subsequent results and findings, taking a closer look to assess possible biases is necessary. The participants mostly have a financial or sales background, with a few exceptions in production or human resources. This might lead to an over-representation of requirements coming from those fields, which in turn could reduce the applicability of the findings to, for example, research and development or logistics jobs. However, the high number of participants with a consulting background, that in turn have lots of experience with very different industries and departments, should be enough to guarantee generalizability for most situations.

The requirements derived from the interviews are abstracted high enough to be widely applied, a real-world implementation of a mobile dashboard would however require a much more detailed and organization-dependent specification.

While the application scenarios as presented above were cross-checked by another expert, they still only provide a rather small and generic subset of possible applications and usages, and might still be influenced by the expectations of the researchers. This also leads to the biggest issue with the evaluation and the findings in general: a detailed user study, preferably with the same experts as in the interviews, would significantly increase the validity of our research, but could not be conducted due to time constraints.

6.4 Further Research

As stated above, our research should first be validated by performing a user evaluation to better assess the usability and usefulness of the prototype. Subsequently, a real-life deployment as part of a field experiment would yield valuable insights into more specific benefits and challenges of using mobile dashboards as part of a BI strategy. This would also require the integration of the software into an existing BI infrastructure, including development of strong authentication and on-device encryption to ensure the protection of sensible data. Ideally, the mobile application would be an extension of a desktop BI system, allowing a user to seamlessly work with the same data on multiple devices. A desktop application could further be used for more complex personalization and setup of the mobile version, allowing users to tailor it to their individual needs.

Additionally, the implementation of more complex interactions and features as described in Section 6.2 especially for more experienced users could be of interest to increase the capabilities and potentially the usefulness of mobile dashboards. Currently, a holistic evaluation of what amount of complexity is too much in which scenario and for which target audience is missing from the scientific BoK, but would be a valuable addition to the results presented in this thesis.

7 Conclusion

To answer the research questions as described in Chapter 1 the problems were approached from multiple perspectives. An initial literature review provided a solid understanding of the state of the art in mobile visualizations and dashboards in general, as well as references for later developments.

The core source for the later results, however, were the 20 expert interviews conducted with participants from 7 industries, employed at companies of different sizes and perspectives on Business Intelligence (BI). This variety of interview partners ensured broad and generalizable insights into the needs of businesses regarding mobile dashboards. Additionally, three of the companies provided multiple participants to allow for a deeper, more detailed view on the requirements of those enterprises. The interviews resulted in a total of 18 rather high-level, general requirements that are well suited to describe the features and properties needed to design a useful and useable mobile dashboard application. While they are certainly not detailed enough to thoroughly specify such a system, they did provide necessary guidance for the later development.

Even more important from a contribution perspective are the three abstract use cases that could be derived from the interviews, as those show the quite different possible modes of operation of mobile dashboards. Especially Use Case 1, concerned with the ease of access to Key Performance Indicators (KPIs) and targets, is relevant for supporting data-driven decision making by all employees of an enterprise. In this case, a simple, intuitive, and quickly accessible mobile BI system could help in significantly reducing the time and effort it takes to get to exactly the information needed by each user individually, while also serving as a first step into more complex visual analytics tools.

A more direct impact on the daily business is likely represented in the second use case labeled “Monitoring”, since push notifications in case of incidents as well as the ability to rapidly monitor critical metrics could lead to faster reaction times, allowing users to quickly decide whether an incident is important enough to stop the current task and shift their focus.

Lastly, mobile dashboards could also provide an additional way of consuming aggregated data about a specific business entity such as a customer or project, to allow for meeting preparations on the go without the need to use any devices other than a smartphone.

The feasibility of those requirements and use cases was evaluated by developing a prototype application as a technical demonstration and test bench for a total of four more concrete application scenarios. To allow for cross-platform usage and a mostly native experience on all devices, while still keeping the development effort manageable within the given time constraints, the prototype was developed as a Progressive Web App (PWA). This approach enabled important features usually reserved for native applications, such as push notifications and offline usage, in a rather simple web project.

To visually represent the data common visualizations were adapted to fit the small screen and low aspect ratio of smartphones, while still providing enough information to users without cluttering the interface. Additionally, a novel visualization called *KPIBar* was developed specifically for the use in mobile dashboards. It encodes up to four of the most important analysis targets of a KPI in a single, compact visualization to allow users to quickly perceive potential deviations of multiple KPIs at once.

In combination with the application scenarios we were able to show that the prototype is indeed able to handle realistic usage situations, as well as demonstrate the general feasibility and usefulness of mobile dashboards.

Future researchers interested in this field could further validate our results by conducting a representative user evaluation or even a field deployment of a production-ready version of a mobile dashboard. Furthermore, the integration into existing BI systems as well as more complex interactions should be researched to evaluate the full potential of those applications.

A Sourcecode

Listing A.1: Data Model

```
BEGIN TRANSACTION;
CREATE TABLE IF NOT EXISTS `kpis` (
        `id` TEXT NOT NULL,
        `name` TEXT NOT NULL,
        `shortname` TEXT,
        `aggregate` TEXT,
        `parent` TEXT NOT NULL,
        `unit` TEXT,
        `direction` TEXT DEFAULT '+',
        `help` TEXT,
        `formula` TEXT DEFAULT 'Direct Measurement',
        PRIMARY KEY(`id`)
);
CREATE TABLE IF NOT EXISTS `partners` (
        `id` INTEGER NOT NULL,
        `name` TEXT NOT NULL,
        `shortname` TEXT NOT NULL,
        `region` TEXT NOT NULL,
        PRIMARY KEY(`id`)
);
CREATE TABLE IF NOT EXISTS `products` (
        `id` INTEGER NOT NULL,
        `name` TEXT NOT NULL,
        `shortname` TEXT NOT NULL,
        `parent` INTEGER NOT NULL,
        PRIMARY KEY(`id`)
);
CREATE TABLE IF NOT EXISTS `measures` (
        `id` INTEGER NOT NULL,
        `timestamp` TEXT NOT NULL,
        `kpi` TEXT NOT NULL,
        `value` REAL NOT NULL,
        `partner` INTEGER,
        `product` INTEGER,
        `scenario` TEXT DEFAULT 'AC',
        PRIMARY KEY(`id`),
        FOREIGN KEY(`kpi`) REFERENCES `kpis`(`id`),
        FOREIGN KEY(`product`) REFERENCES `products`(`id`),
        FOREIGN KEY(`partner`) REFERENCES `partners`(`id`)
);
COMMIT;
```

Listing A.2: Sparkline Date Conversion

```
convertDates() {
    const data = this.chartData.data;
    const now = new Date();
    const today = new Date(now.getFullYear() + "-" + String(now.getMonth() + 1).
        padStart(2,"0") + "-" + String(now.getDate()).padStart(2,"0"));
    const secondsInDay = 86400000;

    // Calculate day differences
    const dayDiffArr = data.map((d:any) => {
        const date = new Date(d.date);
        const dayDiff = (today.getTime() - date.getTime()) / secondsInDay;
        return {date: dayDiff, val: d.val};
    });

    // Convert from days to weeks, months or years
    const maxDays = Number(d3.max(dayDiffArr, (d) => d["date"]));
    let divisor = 365;
    let suffix = "Y";

    switch (true) {
        case maxDays < 14:
            divisor = 1;
            suffix = "D";
            break;

        case maxDays < 30:
            divisor = 7;
            suffix = "W";
            break;

        case maxDays < 365:
            divisor = 30;
            suffix = "M";
            break;
    }

    // Set suffix and new time scale
    this.chartData["suffix"] = suffix;
    this.chartData["data"] = dayDiffArr.map(d => {
        return {date: d["date"]/divisor * -1, val: d["val"]};
    });
}
```

B Drafts

To demonstrate the evolution of some of the basic concepts and visualizations, some initial drafts are shown here.

The Figures B.1 and B.2 represent early versions of the home page and a generic dashboard.

Figure B.1 already features the KPITiles, although not in a carousel, as well as direct links to different dashboards. The dashboard in Figure B.2 is quite different from the later implementations, using some sort of progress bars to visualize deviations to a target, as well as a Sparkline simultaneously.

The detail pages for a single KPI as seen in Figures B.3 and B.4 also show many of the ideas later implemented, such as a Timeline as the central visualization, along with the sub-components presented in the same compact form like on a main dashboard.

Lastly, Figure B.5 shows what would later become the central visualization for the main dashboards: a combination of KPIBars and Sparklines as well as experiments with the time axis of the latter.

In contrast, Figure B.6 contains an interaction concept that would link a Sparkline displaying the full time range with a Timeline chart showing the selected time frame. However, such detailed time frame selection was deemed unnecessary later on.

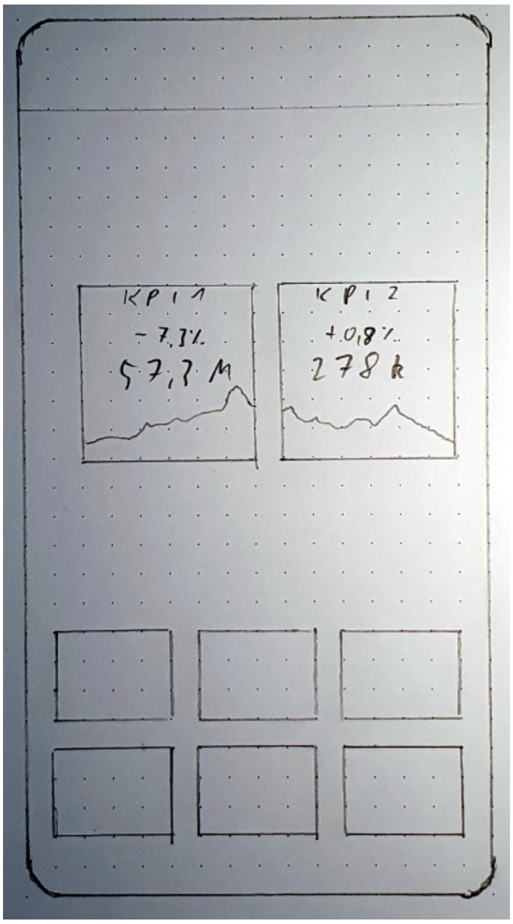


Figure B.1: Draft for the applications home screen featuring an initial version of the KPITiles and navigation buttons

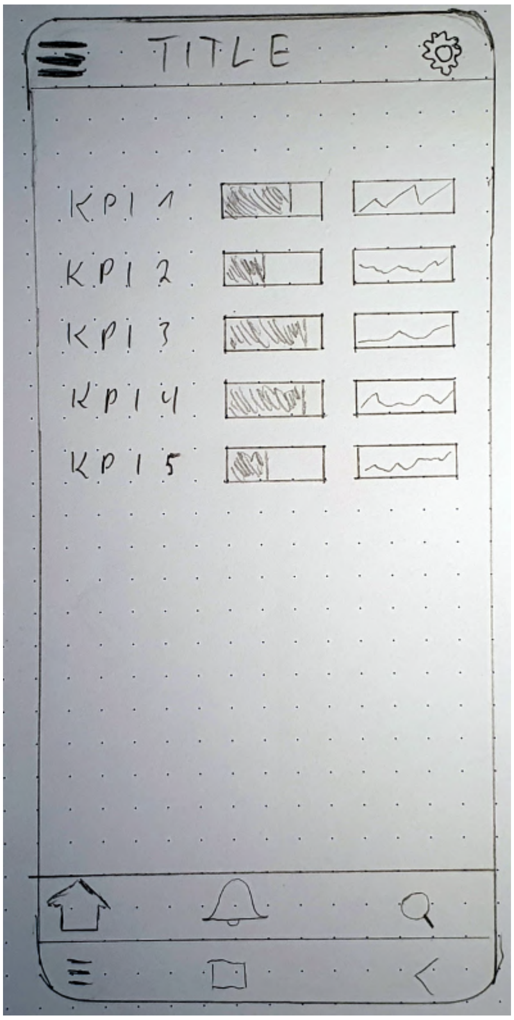


Figure B.2: Concept for displaying multiple KPIs on a dashboard with different visualizations

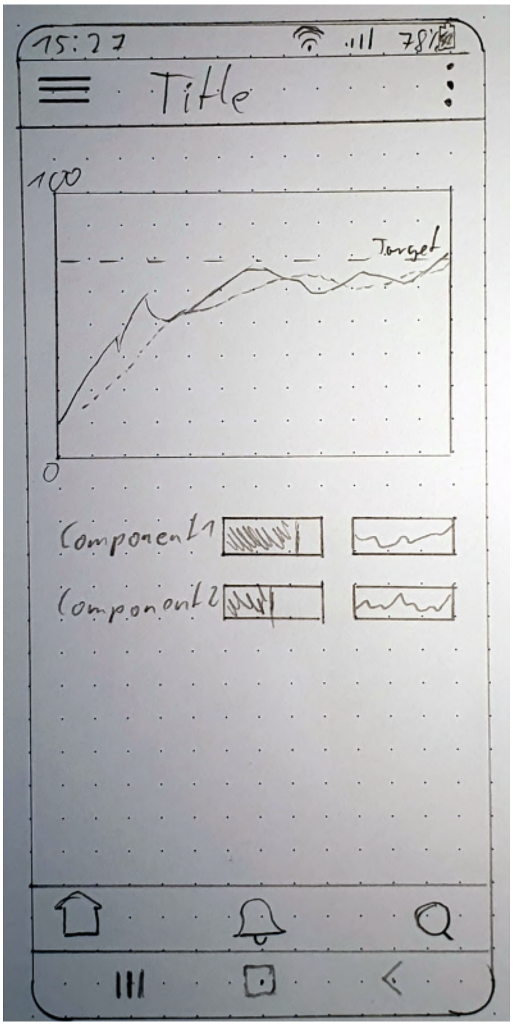


Figure B.3: Detail page of a KPI visualizing its sub-components

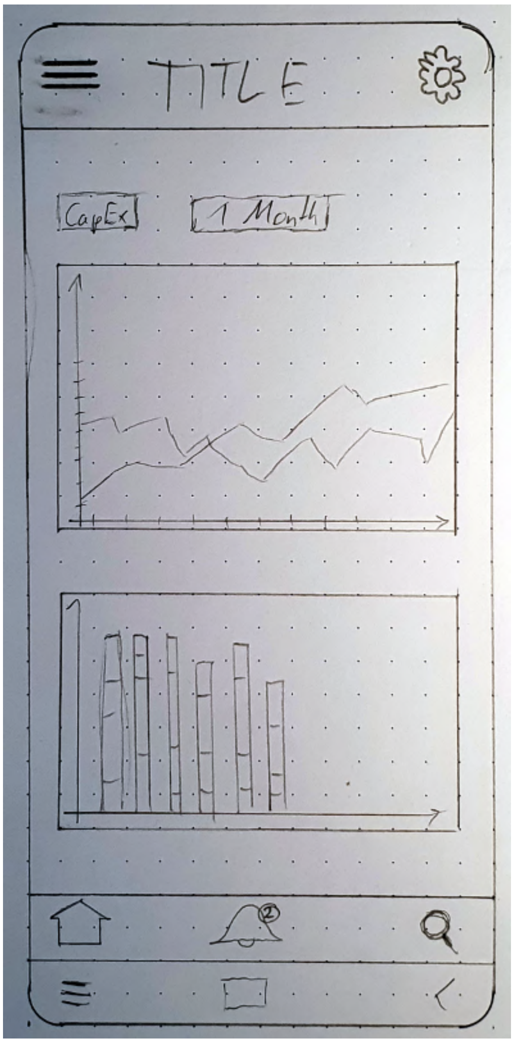


Figure B.4: More complex detail page using a bar chart for breaking down a dimension

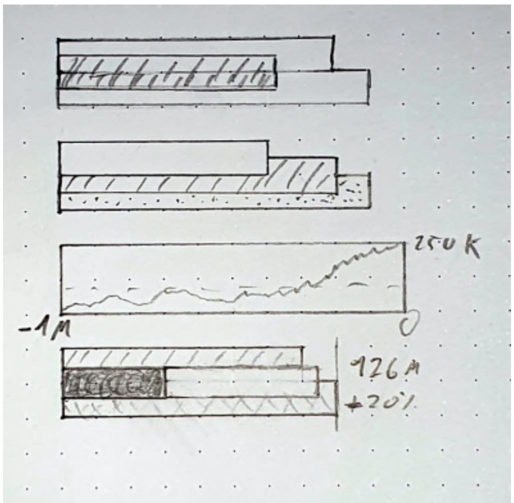


Figure B.5: Different versions of the KPIBar and Sparkline axes

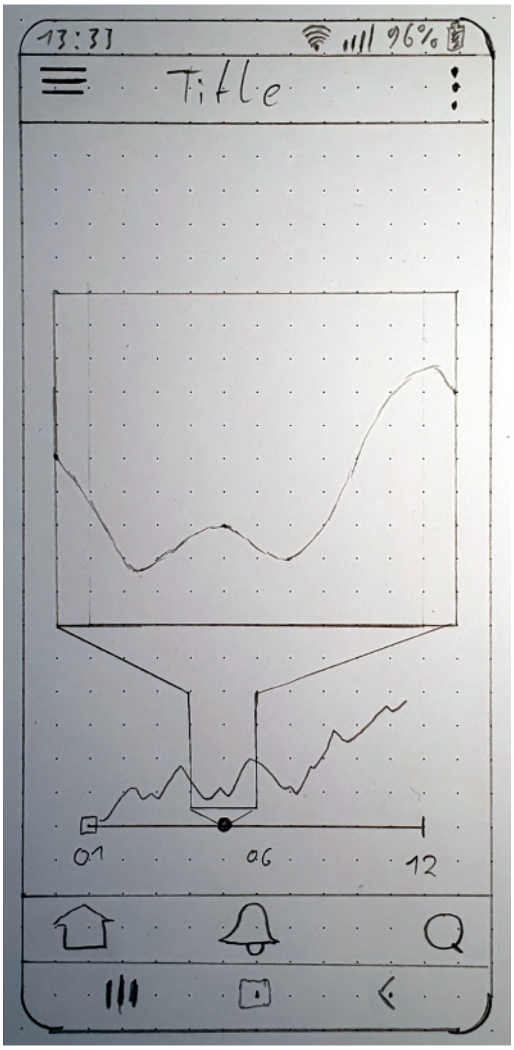


Figure B.6: An interaction concept for scrubbing through large time-series data

C Interview Guides

The interview guides were distributed as separate documents. To retain the original formatting, headers, page numbers and pages, they were scaled down to 95% to fit inside this document.
Direct link to english version: page 59
Direct link to german version: page 62

Interview Guide

Nils Höll

University of Duisburg-Essen

January 14, 2021

I. Introduction

Conducting expert interviews is an important part of the scientific research for my Master Thesis, which aims at exploring and researching possible use cases, data categories and requirements to create a dashboard application taylored to mobile devices like smartphones.
To support those interviews and help potential participants to prepare, this guide gives an overview of the topic, goals and key questions of my project.

The main focus of my work is on researching and developing visualization techniques for monitoring of Key Performance Indicators (KPIs). Conducting interviews with experts from different industries, departements and roles helps to asses current challenges and requirements for mobile reporting, thus ensuring practical relevance of the research.

With the participants permission, the interview will be recorded so the interviewer can focus on the discussion. Any recording will be deleted after the thesis is finalized and graded. Both the participants and their employers are fully anonymized, only the industry, department and role will be noted in the thesis. Direct quotes are only published with explicit permission of the participants; this permission, as well as the permission to use any information from the interviews in the thesis can be revoked at any point prior to finalization (2021-05-20). All material gained from the interview will only be used for research purposes.
The final thesis is reviewed by KPMG AG Wirtschaftsprüfungsgesellschaft (KPMG) to ensure no sensitive or de-anonymizing information is published. Furthermore, the interviewer and all interviews are subject to the standard non-disclosure agreement for KPMG consultants.

II. Goals

As mentioned above, I am conducting research in the field of mobile visualizations and mobile reporting to develop a dashboard application for monitoring and reporting of business data on smartphones.
The main research question that drives my work is: *"How can dashboards for monitoring of business data be realized on mobile devices?"*
To support answering that question, expert interviews are conducted achieve the following goals:

- Learn about possible use cases and szenarios for mobile dashboards
- Explore data categories that should/could be displayed in a mobile dashboard
- Develop suitable visualization and interaction techniques for these data and szenarios
- Collect possible requirements (functional and quality) for a mobile dashboard application

III. Participant Information

To asses the quality and generalizability of the interviews, I ask you for some basic information about your area of expertise:

- In what industry do you work?
- In what department and in what position do you work?
- How often do you need to interact with visualizations or dashboards?

IV. Key Questions

The following questions are a guideline for the interview. Depending on time constraints and the course of the conversation not all of these questions will be asked or need to be answered.

i. Reporting and Business Intelligence

- Do you currently use any reporting or dashboarding tools in your daily work?
- What data is most important to you (in terms of KPIs vs. detailed data, financial or operational numbers etc.)?
- How current is this data? Is this current enough for your work?
- Do you have any good or bad examples, challenges or highlights within your current reporting environment?

ii. Mobile Dashboards

- Are there any scenarios in your work day where you might need mobile access to certain data?
- Do you already have mobile access to any data? If yes, is it data you need?
- What types of data would make sense to use in a mobile environment?
- How current does the information have to be? From the last day, hour, or real-time?
- How much historical data do you need (last month, Year to Date (YTD), Year over Year (YoY))?
- Would you use the application rather for monitoring of single KPIs, or even simple analyses like comparisons?
- Do you believe that mobile dashboards could provide value for your corporation or department?
- Do you have any other whishes regarding the functions of the application?

iii. Application Requirements

- On what devices does the application need to be run (in terms of size, operating system, age)?
- Is there any need for offline usage? For example, when mobile data is not available?
- Would you want to save, export or send views for later analysis?
- Would you like to get push notifications about important business events?
- Are there any other technical requirements you can think of?

V. Open Discussion

If you have any other input, questions or suggestions you are very welcome to express them at any point in the interview.

Thank you very much for participating in this interview and helping me with in my research.

Interviewleitfaden

Nils Höll

Universität Duisburg-Essen

14. Januar 2021

I. Einleitung

Das Durchführen von Experteninterviews ist ein wichtiger Teil der wissenschaftlichen Forschung für meine Masterarbeit, welche darauf abzielt, mögliche Anwendungsfälle, Datenkategorien und Anforderungen zu erforschen und zu untersuchen, um eine Dashboard-Anwendung zu erstellen, die auf mobile Geräte wie Smartphones zugeschnitten ist.
Um diese Interviews zu unterstützen und potenziellen Teilnehmern bei der Vorbereitung zu helfen, gibt dieser Leitfaden einen Überblick über das Thema, die Ziele und die wichtigsten Fragen meines Projekts.

Der Schwerpunkt der Arbeit liegt auf der Erforschung und Entwicklung von Visualisierungstechniken für das Monitoring von Key Performance Indicators (KPIs). Die Durchführung von Interviews mit Experten aus verschiedenen Branchen, Abteilungen und Positionen hilft uns dabei, aktuelle Herausforderungen und Anforderungen an mobiles Reporting zu erfassen und somit den Praxisbezug meiner Forschung zu gewährleisten.

Mit dem Einverständnis der Teilnehmer wird das Interview aufgezeichnet, damit sich der Interviewer auf das Gespräch konzentrieren kann. Jegliche Aufnahme wird nach Fertigstellung und Benotung der Arbeit gelöscht. Sowohl die Teilnehmer als auch ihre Arbeitgeber werden vollständig anonymisiert, lediglich die Branche, die Abteilung und die Rolle werden in der Thesis vermerkt. Direkte Zitate werden nur mit ausdrücklicher Erlaubnis der Teilnehmer veröffentlicht; diese Erlaubnis sowie die Erlaubnis, Informationen aus den Interviews in der Abschlussarbeit zu verwenden, kann bis zur Fertigstellung (2021-05-25) jederzeit widerrufen werden. Alle im Interview gesammelten Informationen werden ausschließlich zu Forschungszwecken verwendet. Die Abschlussarbeit wird von der KPMG AG Wirtschaftsprüfungsgesellschaft (KPMG) geprüft, um sicherzustellen, dass keine sensiblen oder de-anonymisierenden Informationen veröffentlicht werden. Darüber hinaus unterliegen der Interviewer und alle Interviews der Standard-Geheimhaltungsvereinbarung für KPMG-Berater.

II. Ziele

Wie bereits erwähnt, forsche ich im Bereich der mobilen Visualisierung und des mobilen Reportings, um eine Dashboard-Anwendung für Monitoring und Reporting von Business-Daten für Smartphones zu entwickeln. Die Hauptforschungsfrage, die diese Arbeit antreibt, ist:
„Wie können Dashboards zur Überwachung von Geschäftsdaten auf mobilen Geräten realisiert werden?“
Um die Beantwortung dieser Frage zu unterstützen, werden Experteninterviews durchgeführt, um insbesondere die folgenden Ziele zu erreichen:

- Mögliche Anwendungsfälle und Szenarien für mobile Dashboards ermitteln

- Datenkategorien, die in einem mobilen Dashboard angezeigt werden sollten finden
- Geeignete Visualisierungs- und Interaktionstechniken für diese Daten und Szenarien entwickeln
- Sammeln von Anforderungen (funktional bzw. fachlich und qualitativ) für eine mobile Dashboard-Anwendung

III. Informationen über Teilnehmer

Um die Qualität und Allgemeingültigkeit des Interviews beurteilen zu können, bitten ich Sie um einige grundlegende Informationen zu Ihrem Fachgebiet:

- In welcher Branche arbeiten Sie?
- In welcher Abteilung und in welcher Position arbeiten Sie?
- Wie oft arbeiten Sie mit Visualisierungen oder Dashboards?

IV. Leitfragen

Die folgenden Fragen dienen als Leitfaden für das Interview. In Abhängigkeit vom Verlauf des Interviews und zeitlichen Einschränkungen werden nicht unbedingt alle Fragen gestellt.

i. Berichtswesen und Business Intelligence

- Verwenden Sie derzeit irgendwelche Reporting- oder Dashboarding-Tools bei Ihrer täglichen Arbeit?
- Welche Daten sind für Sie am wichtigsten (in Bezug auf KPIs vs. detaillierte Daten, finanzielle oder operative Kennzahlen usw.)?
- Wie aktuell sind diese Daten?
- Haben Sie gute oder schlechte Beispiele, Herausforderungen oder Highlights innerhalb Ihrer aktuellen Berichtsumgebung?

ii. Mobile Dashboards

- Gibt es Szenarien in Ihrem Arbeitsalltag, in denen Sie möglicherweise mobilen Zugriff auf bestimmte Daten benötigen?
- Haben Sie bereits mobilen Zugriff auf bestimmte Daten? Wenn ja, handelt es sich um Daten, die Sie benötigen?
- Welche Arten von Daten wären sinnvoll, um sie in einer mobilen Umgebung zu nutzen?
- Wie aktuell müssen die Informationen sein? Vom letzten Tag, der letzten Stunde oder in Echtzeit?
- Wie viele historische Daten benötigen Sie (letzter Monat, Year to Date (YTD), Year over Year (YoY))?
- Würden Sie eine solche Anwendung nur zum Monitoring einzelner KPIs verwenden, oder würden Sie einfache Analysen wie Vergleiche durchführen wollen?
- Glauben Sie, dass mobile Dashboards einen Mehrwert für Ihr Unternehmen oder Ihre Abteilung bieten könnten?
- Haben Sie weitere funktionalen Wünsche oder Anmerkungen zur Applikation?

iii. Anwendungsanforderungen

- Auf welchen Geräten muss die Anwendung ausgeführt werden (in Bezug auf Größe, Betriebssystem, Alter)?
- Gibt es Bedarf an Offline-Funktionalitäten, zum Beispiel bei fehlendem mobilen Internet?
- Würden Sie gerne einzelne Sichten speichern, exportieren or verschicken können?
- Würden Sie gerne Push-Nachrichten bei wichtigen Ereignissen erhalten?
- Gibt es weitere technische Anforderungen an eine solche Anwendung?

V. Diskussion

Wenn Sie weitere Anregungen, Fragen oder Vorschläge haben, können Sie diese gerne an jeder Stelle des Gesprächs äußern.

Vielen Dank für die Teilnahme am Interview und Ihre Hilfe bei meinen Forschungen.

D Coding Reference

D.1 Code Details

The table below lists all codes used for the qualitative analysis of the interviews, grouped by categories and sorted by alphabet, with additional descriptions. For a full coding table including frequencies and allocations to interviews please refer to appendix D.2.

Table D.1: Code Details

Code Category	Code	Explanation
Functional Requirements	Collaboration Features	Exporting, sharing and commenting of views.
	Comparison to History	Comparison of the current value of a KPIs to the previous month or year.
	Data Context	Showing what data or information is currently displayed, e.g. the time frame, filters or hierarchy level
	Deviation Analysis	Comparison of values to targets.
	Dimension Selection	Possibility to change the base dimension for multivariate data.
	Drill-Down	Exploring details of a specific sub-selection.
	Entity Dashboard	A dashboard that shows information about for example a customer or person.
	Favourites	User-selected important KPIs.
	Forecasts	Visualizing possible future values.
	Information at a Glance	Getting the most important information without interaction.
	Offline Usage	Caching of data to allow for application use without internet.
	Personalization	Configuration of views, KPIs and landing page.
	Push Notifications	Sending push notifications to client devices for special events.
	Time Frame Selection	Selection of the currently displayed time frame
	Top/Flop Lists	Lists to show the top or bottom X customers, products, suppliers etc.
	Trendlines	Additional metric that indicates trends.
	Usability	Simple and intuitive use of the application.
	Visualizations	Whether visualizations are required (in contrast to numbers only).

Non-Functional Requirements	Access Control	Control who can see what data.
	Application Security	Authentication and authorisation of users, Application Programming Interfaces (APIs) and data on server and clients.
	Performance	Application speed.
	Platform-Independent Use	Ability to use application on different operating systems and device sizes.
Data Structure	Financial Data	Accounting and financial controlling data.
	Operational Data	Measures regarding production and logistics.
	Sales Data	Sales volumes, prices and other sales-related KPIs
	Realtime or Streaming Data	Measures are updated in real time.
	Updated Intraday	Measures are updated multiple times a day.
	Updated Daily	Measures are updated once a day (e.g. over night).
	Updated Monthly	Measures are updated monthly.
Use Cases	Easier Access	Simplifying access to and evaluation of information.
	Field Work	Support of e.g. sales people or service employees that travel a lot.
	Increase Data Interaction	Increase employee engagement and use of data.
	Monitoring	Status or health checks of systems and KPIs; possibly in real time.
	Operational Support	Information of users on for example shop floors or in productions plants.
Target Audience	Employees	Creating dashboards for almost every user.
	Middle Management	Main focus on business units or product lines.
	Top Management	Aggregation on company level.
Challenges	Data Literacy	Ability of users to understand and use the presented data and visualizations.
	Data Quality	Accuracy and correctness of the collected data.
	Economic Viability	Which KPIs to display.
	KPI Selection	Which KPIs to display.
	Level of Details	How many details and levels to enable.
Dashboard Experience	Daily Use	Participant uses dashboards daily.
	Weekly Use	Participant uses dashboards 2-3 times a week.
	Dashboard Design	Participant actively develops dashboards.
	Little Dashboard Experience	Participant uses dashboards irregularly or never.
Other	Demand for Mobile Reporting	Whether there is any demand for mobile dashboards.
	Interaction Idea	Possibly interesting interaction concept.
	Visualization Idea	Possibly interesting visualization concept.

D.2 Coding Table

Category	Sub-Category	Code	Frequency	Interviews	Possible Users														Consultancies							
					1	2	3	4	5	6	7	9	10	12	15	17	19	Sum	0	8	11	13	14	16	18	Sum
Functional Requirements	General	Push Notifications	20	14	1	1	2	-	-	1	1	1	2	2	1	3	-	**15**	-	1	-	-	1	1	2	**5**
		Personalization	16	12	-	1	-	1	1	-	2	1	2	-	-	-	1	**9**	2	1	1	-	2	-	1	**7**
		Offline Usage	10	9	1	-	1	-	-	-	2	-	1	1	-	-	1	**7**	1	-	1	-	-	1	-	**3**
		Usability	9	8	-	-	2	-	-	-	1	1	1	-	-	-	-	**5**	-	1	1	-	-	1	1	**4**
		Visualizations	9	8	-	-	-	-	2	1	-	1	1	-	-	1	-	**6**	-	1	-	1	-	-	1	**3**
		Favourites	3	3	1	-	-	-	-	-	-	-	-	-	-	-	-	**1**	-	-	-	-	-	1	1	**2**
	Interaction	Comparison to History	20	16	1	1	2	1	1	2	1	2	1	-	1	1	1	**15**	-	-	1	-	1	1	2	**5**
		Deviation Analysis	21	14	1	1	1	1	3	2	2	-	2	-	-	1	2	**16**	1	1	2	-	-	1	-	**5**
		Collaboration Features	13	11	1	1	1	1	1	1	-	-	2	-	-	-	1	**9**	-	-	1	-	1	-	2	**4**
		Drill-Down	11	9	-	-	-	-	2	-	1	-	1	-	-	1	-	**5**	1	-	1	-	2	1	1	**6**
		Timeframe Selection	2	2	-	-	-	-	-	1	-	-	-	-	-	-	-	**1**	-	-	-	-	-	-	1	**1**
		Dimension Selection	1	1	-	-	-	-	-	-	-	1	-	-	-	-	-	**1**	-	-	-	-	-	-	-	**0**
	Visualization	Information at a Glance	23	16	1	2	2	1	2	2	1	2	-	-	-	1	1	**15**	2	1	1	-	2	1	1	**8**
		Trendlines	16	13	-	1	1	-	1	2	1	1	1	1	-	2	1	**12**	-	2	-	-	1	1	-	**4**
		Entity Dashboard	9	8	1	-	-	1	1	-	1	-	-	1	-	-	1	**6**	-	-	2	1	-	-	-	**3**
		Forecasts	7	7	-	-	-	-	-	1	1	-	1	1	-	1	1	**6**	-	-	-	-	-	1	-	**1**
		Data Context	6	6	-	-	-	-	-	-	-	1	1	-	1	-	-	**3**	-	-	1	-	1	-	1	**3**
		Top/Flop Lists	4	3	1	-	-	-	-	2	-	-	-	-	-	-	-	**3**	-	-	1	-	-	-	-	**1**
Non-Functional Requirements		Platform-Independent Use	9	9	1	-	-	-	1	1	1	-	1	1	-	1	-	**7**	-	-	1	-	1	-	-	**2**
		Application Security	9	6	-	-	-	-	1	-	2	-	-	2	-	-	-	**5**	-	-	1	-	2	-	1	**4**
		Access Control	4	3	-	-	-	-	-	-	2	-	-	-	-	-	1	**3**	-	-	-	-	1	-	-	**1**
		Performance	2	2	-	-	-	-	-	-	1	-	-	-	-	-	-	**1**	-	-	-	-	-	1	-	**1**
Data Structure	Business Function	Financial Data	13	12	2	1	-	1	-	1	1	1	-	-	-	-	1	**8**	-	-	1	1	1	1	1	**5**
		Sales Data	13	11	-	1	-	1	1	3	1	-	1	-	-	-	1	**9**	-	-	1	1	1	-	1	**4**
		Operational Data	12	8	3	1	2	-	-	-	1	-	-	-	-	1	-	**8**	-	-	-	1	1	2	-	**4**
	Timeliness	Updated Daily	14	12	1	-	-	-	1	2	1	1	1	-	1	-	-	**8**	1	-	2	1	-	1	1	**6**
		Updated Intraday	10	8	1	-	2	-	-	-	-	-	-	1	1	-	-	**5**	-	-	2	1	1	1	-	**5**
		Updated Monthly	7	7	1	-	-	-	1	1	1	-	-	1	-	1	-	**6**	-	-	-	-	1	-	-	**1**
		Realtime or Streaming Data	1	1	-	-	-	1	-	-	-	-	-	-	-	-	-	**1**	-	-	-	-	-	-	-	**0**

Figure D.1: Full coding table (Part 1)

Category	Sub-Category	Code	Frequency	Interviews	Possible Users 1	2	3	4	5	6	7	9	10	12	15	17	19	Sum	Consultancies 0	8	11	13	14	16	18	Sum
Target Audience		Top Management	6	6	-	1	1	1	-	-	-	-	-	-	-	1	-	**4**	-	1	-	-	-	1	-	**2**
		Employees	7	5	-	-	1	1	2	-	1	-	-	-	-	-	-	**5**	-	-	2	-	-	-	-	**2**
		Middle Management	5	5	-	-	1	-	1	-	-	-	-	-	-	1	-	**3**	-	1	1	-	-	-	-	**2**
Use Cases		Easier Access	17	13	-	1	2	-	1	-	1	1	1	1	1	2	1	**12**	2	1	-	-	2	-	-	**5**
		Monitoring	18	11	1	2	1	-	3	1	-	1	-	-	3	1	-	**13**	2	2	-	-	-	1	-	**5**
		Field Work	9	9	1	-	1	1	1	-	1	1	-	-	-	-	1	**7**	1	-	-	1	-	-	-	**2**
		Increase Data Interaction	8	5	-	-	-	1	4	-	-	-	1	-	-	-	1	**7**	-	-	-	-	-	-	1	**1**
		Meetings	5	5	1	-	-	1	-	-	1	-	-	1	-	-	-	**4**	1	-	-	-	-	-	-	**1**
		Operational Support	9	5	-	-	-	-	2	-	-	-	-	2	-	-	-	**4**	-	3	1	-	1	-	-	**5**
Dashboard Experience		Daily Use	11	11	1	1	-	-	-	1	1	1	-	-	1	-	1	**7**	1	1	1	-	-	-	1	**4**
		Dashboard Design	11	11	1	1	-	-	-	-	1	-	-	-	1	-	-	**4**	1	1	1	1	1	1	1	**7**
		Weekly Use	6	6	-	-	-	1	1	-	-	-	1	-	-	-	-	**3**	-	-	-	1	1	1	-	**3**
		Little Dashboard Experience	1	1	-	-	-	-	-	-	-	-	-	1	-	-	-	**1**	-	-	-	-	-	-	-	**0**
Challenges		Data Quality	8	6	1	-	2	1	-	-	1	-	1	-	-	-	2	**8**	-	-	-	-	-	-	-	**0**
		Data Literacy	5	5	-	-	-	-	-	-	-	-	-	-	-	1	1	**2**	-	1	-	1	1	-	-	**3**
		Economic Viability	4	3	-	1	-	-	-	-	-	-	-	-	2	-	-	**3**	-	-	-	1	-	-	-	**1**
		KPI Selection	4	3	-	2	1	-	1	-	-	-	-	-	-	-	-	**4**	-	-	-	-	-	-	-	**0**
		Level of Details	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	**1**	-	-	-	-	-	-	-	**0**
Other		Demand for Mobile Reporting	11	11	1	1	1	1	-	-	1	-	-	-	1	1	1	**8**	-	-	-	1	1	1	-	**3**
		Interaction Idea	16	9	3	-	-	1	1	-	3	1	-	-	-	-	-	**9**	1	2	-	1	-	-	3	**7**
		Visualization Idea	7	6	-	-	-	1	1	1	-	-	-	-	-	-	-	**3**	2	-	-	1	-	-	1	**4**
		Other Requirements	3	3	-	-	-	-	-	1	-	-	-	-	-	-	-	**1**	-	-	-	-	1	-	1	**2**

Figure D.2: Full coding table (Part 2)

Bibliography

[1] Kim Normann Andersen et al. *Electronic Government and the Information Systems Perspective: Second International Conference, EGOVIS 2011, Toulouse, France, August 29 – September 2, 2011, Proceedings*. Springer Science & Business Media, Aug. 19, 2011. 422 pp. ISBN: 978-3-642-22960-2. Google Books: Xd6HDE5wUIoC.

[2] Mark Apperley and Jishaal Kalyan. "A Mobile Personal Residential Electricity Dashboard". In: *2015 19th International Conference on Information Visualisation*. 2015 19th International Conference on Information Visualisation. July 2015, pp. 195–199. DOI: 10.1109/iV.2015.43.

[3] Wasana Bandara et al. "Achieving Rigor in Literature Reviews: Insights from Qualitative Data Analysis and Tool-Support". In: *Communications of the Association for Information Systems* 37.1 (Aug. 1, 2015). ISSN: 1529-3181. DOI: 10.17705/1CAIS.03708.

[4] Fabian Beck and Daniel Weiskopf. "Word-Sized Graphics for Scientific Texts". In: *IEEE Transactions on Visualization and Computer Graphics* 23.6 (Jan. 7, 2017), pp. 1576–1587. DOI: 10.1109/TVCG.2017.2674958.

[5] Palash Bera. "How Colors in Business Dashboards Affect Users' Decision Making". In: *Communications of the ACM* 59.4 (Mar. 23, 2016), pp. 50–57. ISSN: 0001-0782. DOI: 10.1145/2818993.

[6] Andreas Biørn-Hansen, Tim A. Majchrzak, and Tor-Morten Grønli. "Progressive Web Apps: The Possible Web-Native Unifier for Mobile Development:" in: *Proceedings of the 13th International Conference on Web Information Systems and Technologies*. 13th International Conference on Web Information Systems and Technologies. Porto, Portugal: SCITEPRESS - Science and Technology Publications, 2017, pp. 344–351. ISBN: 978-989-758-246-2. DOI: 10.5220/0006353703440351.

[7] Tanja Blascheck et al. "Glanceable Visualization: Studies of Data Comparison Performance on Smartwatches". In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (Jan. 2019), pp. 630–640. ISSN: 1941-0506. DOI: 10.1109/TVCG.2018.2865142.

[8] Kerstin Blumenstein et al. "Evaluating Information Visualization on Mobile Devices: Gaps and Challenges in the Empirical Evaluation Design Space". In: *Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization*. BELIV '16. New York, NY, USA: Association for Computing Machinery, Oct. 24, 2016, pp. 125–132. ISBN: 978-1-4503-4818-8. DOI: 10.1145/2993901.2993906.

[9] *Can I Use... Support Tables for HTML5, CSS3, Etc*. May 9, 2021. URL: https://caniuse.com/serviceworkers (visited on 05/20/2021).

[10] Youli Chang et al. "Understanding Users' Touch Behavior on Large Mobile Touch-Screens and Assisted Targeting by Tilting Gesture". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. New York, NY, USA: Association for Computing Machinery, Apr. 18, 2015, pp. 1499–1508. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702425.

[11] Hsinchun Chen, Roger H. L. Chiang, and Veda C. Storey. "Business Intelligence and Analytics: From Big Data to Big Impact". In: *MIS Quarterly* 36.4 (2012), pp. 1165–1188. ISSN: 0276-7783. DOI: 10.2307/41703503. JSTOR: 41703503.

[12] Yang Chen. "Visualizing Large Time-Series Data on Very Small Screens". In: *EuroVis 2017 - Short Papers*. Eurographics Conference on Visualization (EuroVis) 2017. The Eurographics Association, 2017. ISBN: 978-3-03868-043-7. DOI: 10.2312/eurovisshort.20171130.

[13] L. Chittaro. "Visualizing Information on Mobile Devices". In: *Computer* 39.3 (Mar. 2006), pp. 40–45. ISSN: 1558-0814. DOI: 10.1109/MC.2006.109.

[14] Eun Kyoung Choe et al. "Mobile Data Visualization (Dagstuhl Seminar 19292)". Version 1.0. In: (2019). In collab. with Michael Wagner, 16 pages. DOI: 10.4230/DAGREP.9.7.78.

[15] Mozilla Corporation. *Cache - Web APIs | MDN*. Feb. 10, 2021. URL: https://developer.mozilla.org/en-US/docs/Web/API/Cache (visited on 05/15/2021).

[16] Andrea Cuttone, Sune Lehmann, and Jakob Eg Larsen. "A Mobile Personal Informatics System with Interactive Visualizations of Mobility and Social Interactions". In: *Proceedings of the 1st ACM International Workshop on Personal Data Meets Distributed Multimedia*. PDM '13. New York, NY, USA: Association for Computing Machinery, Oct. 22, 2013, pp. 27–30. ISBN: 978-1-4503-2397-0. DOI: 10.1145/2509352.2509397.

[17] Steven M. Drucker et al. "TouchViz: A Case Study Comparing Two Interfaces for Data Analytics on Tablets". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. New York, NY, USA: Association for Computing Machinery, Apr. 27, 2013, pp. 2301–2310. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2481318.

[18] Florian Eder and Stefan Koch. "Critical Success Factors for the Implementation of Business Intelligence Systems". In: *International Journal of Business Intelligence Research (IJBIR)* 9.2 (July 1, 2018), pp. 27–46. ISSN: 1947-3591. DOI: 10.4018/IJBIR.2018070102.

[19] Facebook, Inc. *Create React App*. Facebook, Nov. 29, 2020. URL: https://github.com/facebook/create-react-app (visited on 11/29/2020).

[20] Dan Farber. *When iPhone Met World, 7 Years Ago Today*. CNET. Jan. 9, 2014. URL: https://www.cnet.com/news/when-iphone-met-world-7-years-ago-today/ (visited on 03/23/2021).

[21] Stephen Few. *Information Dashboard Design : Displaying Data for at-a-Glance Monitoring*. 2. ed. Burlingame, Calif.: Analytics Press, 2013. ISBN: 1-938377-00-1.

[22] Stephen Few. *There's Nothing Mere About Semantics*. Visual Business Intelligence. Dec. 13, 2017. URL: http://www.perceptualedge.com/blog/?p=2793 (visited on 11/27/2020).

[23] GitHub, Inc. *Viewing Contributions on Your Profile - GitHub Docs*. 2021. URL: https://docs.github.com/en/github/setting-up-and-managing-your-github-profile/viewing-contributions-on-your-profile#viewing-contributions-from-specific-times (visited on 05/07/2021).

[24] *Global Big Data and Business Analytics Revenue 2015-2022*. Statista. Apr. 2019. URL: https://www.statista.com/statistics/551501/worldwide-big-data-business-analytics-revenue/ (visited on 10/13/2020).

[25] *Global Business Travel Spending Growth Forecast 2015-2020*. Statista. July 2017. URL: https://www.statista.com/statistics/324786/global-business-travel-spending-growth-forecast/ (visited on 10/13/2020).

[26] A. Goedecke et al. *Das Controlling-Konzept: Die Gestaltung Eines Wirkungsvollen Controllingsystems*. 8th ed. dtv München, 2016. ISBN: 978-3-423-50949-7.

[27] Pascal Goffin et al. "Interaction Techniques for Visual Exploration Using Embedded Word-Scale Visualizations". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. New York, NY, USA: Association for Computing Machinery, Apr. 21, 2020, pp. 1–13. ISBN: 978-1-4503-6708-0. DOI: 10.1145/3313831.3376842.

[28] Google LLC. *Caching Files with Service Worker*. Google Developers. May 1, 2019. URL: https://developers.google.com/web/ilt/pwa/caching-files-with-service-worker (visited on 05/17/2021).

[29] Google LLC. *Unit 01: Why build Progressive Web Apps*. Google Docs. URL: https://docs.google.com/presentation/d/18IOdAivMSRDsqEnwvJL_eaEWhqBgFUjFWEamoWZ5djc/edit?usp=embed_facebook (visited on 04/24/2021).

[30] Christian Grund and Michael Schelkle. "Developing a Serious Game for Business Information Visualization". In: *AMCIS 2016 Proceedings* (Aug. 11, 2016). URL: https://aisel.aisnet.org/amcis2016/HCI/Presentations/4.

[31] Alan Hevner. "A Three Cycle View of Design Science Research". In: *Scandinavian Journal of Information Systems* 19.2 (Jan. 1, 2007). URL: https://aisel.aisnet.org/sjis/vol19/iss2/4.

[32] Alan R. Hevner et al. "Design Science in Information Systems Research". In: *MIS Quarterly* 28.1 (2004), pp. 75–105. ISSN: 0276-7783. DOI: 10.2307/25148625. JSTOR: 25148625.

[33] Rolf Hichert and Jürgen Faisst. *International Business Communication Standards: Conceptual, Perceptual, and Semantic Design of Comprehensible Business Reports, Presentations, and Dashboards*. 1.1. CreateSpace Independent Publishing Platform, July 17, 2017. 168 pp. ISBN: 978-1-974669-19-6. URL: https://www.ibcs.com/standards/#%3F=.

[34] S. E. Hove and B. Anda. "Experiences from Conducting Semi-Structured Interviews in Empirical Software Engineering Research". In: *11th IEEE International Software Metrics Symposium (METRICS'05)*. 11th IEEE International Software Metrics Symposium (METRICS'05). Sept. 2005, 10 pp.–23. DOI: 10.1109/METRICS.2005.24.

[35] Jaemin Jo et al. "TouchPivot: Blending WIMP & Post-WIMP Interfaces for Data Exploration on Tablet Devices". In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. New York, NY, USA: Association for Computing Machinery, May 2, 2017, pp. 2660–2671. ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025752.

[36] Wafaa S. El-Kassas et al. "Taxonomy of Cross-Platform Mobile Applications Development Approaches". In: *Ain Shams Engineering Journal* 8.2 (June 1, 2017), pp. 163–190. ISSN: 2090-4479. DOI: 10.1016/j.asej.2015.08.004.

[37] D. A. Keim et al. "Challenges in Visual Data Analysis". In: *Tenth International Conference on Information Visualisation (IV'06)*. Tenth International Conference on Information Visualisation (IV'06). July 2006, pp. 9–16. DOI: 10.1109/IV.2006.31.

[38] Hans-Georg Kemper, Henning Baars, and Walid Mehanna. *Business Intelligence - Grundlagen Und Praktische Anwendungen*. 3rd ed. Wiesbaden: Vieweg+Teubner Verlag, 2010. ISBN: 978-3-8348-0719-9.

[39] Andreas Klein and Jens Gräf. *Reporting Und Business Intelligence*. München: Haufe Gruppe, 2017. ISBN: 978-3-648-08826-5.

[40] Shahid Latif and Fabian Beck. "Visually Augmenting Documents With Data". In: *Computing in Science Engineering* 20.6 (Nov. 2018), pp. 96–103. ISSN: 1558-366X. DOI: 10.1109/MCSE.2018.2875316.

[41] Bongshin Lee et al. "Data Visualization on Mobile Devices". In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18: CHI Conference on Human Factors in Computing Systems. Montreal QC Canada: ACM, Apr. 20, 2018, pp. 1–8. ISBN: 978-1-4503-5621-3. DOI: 10.1145/3170427.3170631.

[42] Google LLC. *Progressive Web Apps Training*. Progressive Web Apps Training | Google Developers. May 1, 2019. URL: https://developers.google.com/web/ilt/pwa (visited on 04/24/2021).

[43] B. H. McCormick. "Visualization in Scientific Computing". In: *ACM SIGBIO Newsletter* 10.1 (Mar. 1, 1988), pp. 15–21. ISSN: 0163-5697. DOI: 10.1145/43965.43966.

[44] Sven Michalczyk et al. "A State-of-the-Art Overview and Future Research Avenues of Self-Service Business Intelligence and Analytics". In: *Proceedings of the 28th European Conference on Information Systems (ECIS)*. ECICS 2020. Association for Information Systems, 2020, p. 21. URL: https://aisel.aisnet.org/ecis2020_rp/46.

[45] Mozilla Corporation. *Notifications API - Web APIs | MDN*. Feb. 26, 2021. URL: https://developer.mozilla.org/en-US/docs/Web/API/Notifications_API (visited on 05/15/2021).

[46] Mozilla Corporation. *Service Worker API - Web APIs | MDN*. Aug. 20, 2021. URL: https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API (visited on 05/15/2021).

[47] Jakob Nielsen. *Usability Engineering*. Morgan Kaufmann, Oct. 7, 1994. 382 pp. ISBN: 978-0-12-518406-9. Google Books: 95As2OF67f0C.

[48] Chidume Nnamdi. *How to Build a React Progressive Web Application (PWA)*. Medium. July 23, 2019. URL: https://blog.bitsrc.io/how-to-build-a-react-progressive-web-application-pwa-b5b897df2f0a (visited on 01/23/2021).

[49] Monique Noirhomme-fraiture et al. "Data Visualizations on Small and Very Small Screens". In: *In ASMDA 2005: Proceedings of Applied Stochastic Models and Data Analysis*. 2005, p. 2005.

[50] Ian Oakley et al. "Beats: Tapping Gestures for Smart Watches". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. New York, NY, USA: Association for Computing Machinery, Apr. 18, 2015, pp. 1237–1246. ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702226.

[51] Briony J. Oates. *Researching Information Systems and Computing*. Los Angeles: Sage, 2006. ISBN: 1-4129-0223-1.

[52] Addy Osmani. *The App Shell Model*. The App Shell Model. May 14, 2019. URL: https://developers.google.com/web/fundamentals/architecture/app-shell (visited on 11/24/2020).

[53] Phone Sized. *Compare Phone and Tablet Sizes Side by Side*. Phone Sized. Apr. 23, 2021. URL: https://phonesized.com/compare/#1447,1582,1583,1669,1668,1566,1329 (visited on 04/27/2021).

[54] Mufizur Rahman et al. "GEAR Analytics: A Clinician Dashboard for a Mobile Game Assisted Rehabilitation System". In: *2016 4th International Conference on User Science and Engineering (i-USEr)*. 2016 4th International Conference on User Science and Engineering (i-USEr). Aug. 2016, pp. 193–198. DOI: 10.1109/IUSER.2016.7857959.

[55] Jan Recker. *Scientific Research in Information Systems - a Beginner's Guide*. 1st ed. Heidelberg: Springer Verlag, 2013. ISBN: 978-3-642-30047-9.

[56] Rahul Reddy. *Smartphone vs Tablet Orientation: Who's Using What?* ScientiaMobile. July 28, 2017. URL: https://www.scientiamobile.com/smartphone-vs-tablet-orientation-whos-using-what/ (visited on 03/30/2021).

[57] F. John Reh. *How Key Performance Indicators Work*. The Balance Careers. Mar. 19, 2020. URL: https://www.thebalancecareers.com/key-performance-indicators-2275156 (visited on 05/19/2021).

[58] Irene Ros. *MobileVis*. June 2014. URL: http://patterns.mobilev.is/ (visited on 11/17/2020).

[59] Alper Sarikaya et al. "What Do We Talk About When We Talk About Dashboards?" In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (Jan. 2019), pp. 682–692. ISSN: 1941-0506. DOI: 10.1109/TVCG.2018.2864903.

[60] Reinhard Schütte. "Management of Large Enterprise Systems". Lecture. Management of Large Enterprise Systems (University of Duisburg-Essen). Oct. 2018.

[61] Michail Schwab et al. "Evaluating Pan and Zoom Timelines and Sliders". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. New York, NY, USA: Association for Computing Machinery, May 2, 2019, pp. 1–12. ISBN: 978-1-4503-5970-2. DOI: 10.1145/3290605.3300786.

[62] Maria Shitkova et al. "Towards Usability Guidelines for Mobile Websites and Applications". In: *12th International Conference on Wirtschaftsinformatik*. 2015, p. 15.

[63] Ben Shneiderman. "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations". In: *The Craft of Information Visualization*. Ed. by BENJAMIN B. Bederson and BEN Shneiderman. Interactive Technologies. San Francisco: Morgan Kaufmann, Jan. 1, 2003, pp. 364–371. ISBN: 978-1-55860-915-0. DOI: 10.1016/B978-155860915-0/50046-9.

[64] Martin Smuts, Brenda Scholtz, and Andre Calitz. "Design Guidelines for Business Intelligence Tools for Novice Users". In: *Proceedings of the 2015 Annual Research Conference on South African Institute of Computer Scientists and Information Technologists*. Conference on South African Institute of Computer Scientists and Information Technologists. SAICSIT '15. New York, NY, USA: Association for Computing Machinery, Sept. 28, 2015, pp. 1–15. ISBN: 978-1-4503-3683-3. DOI: 10.1145/2815782.2815788.

[65] Robert Spence. "Introduction". In: *Information Visualization: An Introduction*. Ed. by Robert Spence. Cham: Springer International Publishing, 2014, pp. 1–20. ISBN: 978-3-319-07341-5. DOI: 10.1007/978-3-319-07341-5_1.

[66] J.J. Thomas and K.A. Cook. "A Visual Analytics Agenda". In: *IEEE Computer Graphics and Applications* 26.1 (Jan. 2006), pp. 10–13. ISSN: 1558-1756. DOI: 10.1109/MCG.2006.5.

[67] Edward R. Tufte. *Beautiful Evidence*. Cheshire, Conn.: Graphics Press, 2006. ISBN: 0-9613921-7-7. URL: http://digitale-objekte.hbz-nrw.de/storage/2008/01/15/file_53/2263821.pdf.

[68] *US Smartphone Sales by Display Size 2017-2019*. Statista. Aug. 2019. URL: https://www.statista.com/statistics/1042669/us-smartphone-sales-by-display-size/ (visited on 10/13/2020).

[69] Kim Verkooij and Marco Spruit. "Mobile Business Intelligence: Key Considerations for Implementations Projects". In: *Journal of Computer Information Systems* 54.1 (Sept. 1, 2013), pp. 23–33. ISSN: 0887-4417. DOI: 10.1080/08874417.2013.11645668.

[70] Benjamin Watson and Vidya Setlur. "Emerging Research in Mobile Visualization". In: *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '15. New York, NY, USA: Association for Computing Machinery, Aug. 24, 2015, pp. 883–887. ISBN: 978-1-4503-3653-6. DOI: 10.1145/2786567.2786571.

[71] Hugh Watson. "Tutorial: Business Intelligence – Past, Present, and Future". In: *Communications of the Association for Information Systems* 25.1 (Nov. 1, 2009). ISSN: 1529-3181. DOI: 10.17705/1CAIS.02539.

[72] Jane Webster and Richard Watson. "Analyzing the Past to Prepare for the Future: Writing a Literature Review". In: *Management Information Systems Quarterly* 26.2 (June 1, 2002). ISSN: 0276-7783. URL: https://aisel.aisnet.org/misq/vol26/iss2/3.

[73] M. Whitlock, K. Wu, and D. A. Szafir. "Designing for Mobile and Immersive Visual Analytics in the Field". In: *IEEE Transactions on Visualization and Computer Graphics* 26.1 (Jan. 2020), pp. 503–513. ISSN: 1941-0506. DOI: 10.1109/TVCG.2019.2934282.

[74] Andreas Wiener and Ekrem Namazci. *Power BI, Data Und AI - Im Gespräch Mit Ekrem Namazci, Microsoft*. Podcast. Feb. 17, 2021. URL: https://open.spotify.com/episode/4IZ7KhyTkE8empxGXSkES1 (visited on 02/17/2021).