



HAL
open science

Adapting Mobile User Interfaces based on users' application usage behaviors

Hajer Dammak, Meriem Riahi, Faouzi Moussa

► **To cite this version:**

Hajer Dammak, Meriem Riahi, Faouzi Moussa. Adapting Mobile User Interfaces based on users' application usage behaviors. IJAHUC - International Journal of Ad Hoc and Ubiquitous Computing, 2023. hal-03984562

HAL Id: hal-03984562

<https://hal.science/hal-03984562v1>

Submitted on 12 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adapting Mobile User Interfaces based on users' application usage behaviors

Abstract: Understanding how users use smartphones can improve their quality of life and user experience and aid in the development of better smartphones. Manufacturers and researchers assume mobile users behave similarly. However, different interests or activities lead to diverse application usage behaviors. Additionally, mobile interfaces are static and do not self-adapt to the user's app usage. To this end, it's important to divide users into groups based on how they use mobile applications and then adapt the interfaces accordingly. This paper presents a two-phased UI adaptation approach. The first phase trains the model and identifies user clusters, while the second adapts the user interface. Two adaptations are offered. Primary adaptation is applied automatically once the user's cluster is identified, while advanced adaptation requires user approval. Our research could improve the overall smartphone experience by making it easier to create, develop, and adapt mobile interfaces for both individuals and groups of users.

Keywords: Adaptation; User Interface; Mobile Application; Usage Behavior; Clustering; User Groups.

1 Introduction

The smartphone is considered an essential component of modern life [1]. It offers a wide range of functions, and users can benefit from mobile phone apps to achieve many affordable goals [2, 3]. It can be transformed into GPS, musical instruments, and credit cards, among others [4], and it helps to achieve a variety of activities such as ordering food, shopping online, and managing health [5, 6, 7, 8].

Generally, a smartphone is associated with one person. Thus, the usage of smartphone apps can communicate a lot of personal information such as needs, interests, habits, and activities. This can provide new insight to better profile users.

In the last few years, several studies have tried to profile users from various points of view based on their app usage behavior, such as inferring demographics and personality [9, 10], and identifying the interests, preferences, and habits of users [11, 12, 13].

Understanding app usage behavior has the potential to improve the user experience (abbr., UX) and the user's life quality. Many researchers have in recent years invested in understanding users' behavior with their smartphone apps, known as "*app usage behavior*".

However, they either scratched the surface to distinguish different categories of users, examined the usage of the apps, or discovered the variety of users. They made no attempt to investigate any differences in app usage behavior amongst user groups. Furthermore, they have not explored the app's usage behavior for further purposes, such as adapting the mobile user interface (abbr., UI). Despite the remarkable technological evolution of smartphones, it continues to fall short of users' expectations.

The mobile UI is a critical component of the UX for mobile devices and applications. User-friendly, efficient, and responsive mobile UIs are essential. In today's fast-paced,

competitive mobile industry, users expect to be able to accomplish their tasks quickly and easily. Thus, understanding and adapting to mobile app users' behavior is crucial.

Adapting the interface of a mobile device based on the user's app usage behavior is crucial for several reasons. Firstly, mobile UIs are typically static and do not self-adapt to the user's needs, meaning that the app groupings and icons on the current screen are determined by the device manufacturer. Besides, the user's most used or most time-consuming apps are not considered either. This can lead to a disjointed UX, where the user has to navigate through multiple screens to access the apps they use most frequently. Regardless of user behavior, they remain unchanged.

The significance of adapting the mobile device's interface extends beyond a single app and encompasses several key benefits. In fact, adapting the interface for a particular app is not enough. A user's behavior can vary significantly between different apps, and their needs and preferences may change over time. A mobile UI that is tailored to the user's app usage behavior across all applications will provide a more seamless and consistent experience, making it easier for users to complete their tasks quickly and efficiently.

Adapting the mobile device's interface is crucial as it enhances the overall UX, increases consistency and coherence, addresses the challenges of the mobile form factor, and accommodates the needs of diverse user groups. In fact, an adaptive interface can cater to the diverse needs and preferences of different user groups, such as elderly users, those with disabilities, or those with different cultural backgrounds.

This requires first identifying user behavior groups based on mobile app usage and then adapting mobile UIs for each group.

We believe that the concept of adaptive UI based on the app usage behaviors of users and their clustered groups is an interesting research direction. The findings could help in designing, developing, and recommending apps for specific user groups, and improving the overall smartphone usage experience. The goal of this study is to identify various groups of users in order to adapt the UI for the different groups of users, and then for each user individually.

The remainder of the paper is organized as follows: Section 2 gives a synopsis of the related works. This section covers the exploited user information in profiling as well as the data of apps for user profiling. Section 3 presents our two-phased approach for adapting the UI based on the user app usage behavior. Section 4 is dedicated to the first phase of the approach where we present our method for collecting log files. We analyze the collected data and expose the experiment results of the trained model and we present our analysis of the obtained clusters. In Section 5, we present the proposed two levels of adaptation and the adaptation rules. Section 6 discusses the benefits and limits of our research work. Finally, Section 7 concludes the paper and outlines future perspectives.

2 Related work

This section presents the most relevant research on collecting data, app usage behavior, and UI adaptation.

2.1 Data Collection

Adaptation requires data collection. Log studies provide the most complete and accurate view of mobile user activity [14]. Logs are captured easily at scale. Log files record what

users do rather than why. They can also test researchers' hypotheses about user behavior. Table 1 presents the findings of several studies involving log files and data collection.

	Fernandez et al.[15]	Ma et al. [16]	Kluth,et al. [17]	Marczal et al. [18]	Holzmann et al. [19]	Ferre et al. [20]	Riegler et al. [21]
Platform	Android	Android	iOS	Android+iOS	Android	Android	Android
Log Format	CSV	-	-	-	CSV	-	-
Storage	Mobile device	Central server	Central server	server	Mobile device	GAMA server	Web server
Instrumentation	Yes (manually added code)	Yes (requires code modification)	Yes	-	No	Yes	-
Data collection	Triggered by the user	-	Auto	Auto (service)	Triggered by the user	Auto	Triggered by the user
Scaling	No	No	-	Yes	No	Yes	No
Collected information	Interaction info	Interaction Data:UI events	Interaction info	Interaction info+Mobility (GPS, data sensor)	Context of use+ Interaction info	Interaction info	Visited app's screenshot + Interaction info
Test-evaluation	-	Real usage+ Lab	Lab	Real usage	Real usage	Real usage + Lab	Lab
Purpose	Usability analysis	Usability evaluation	Usability evaluation	Behavior analysis	UI evaluation	Usability evaluation	Usability
Study object	Mobile apps	Mobile apps	Mobile apps	Mobile apps	Mobile device	Mobile apps	Mobile UI

Table 1 Research studies using log files for data collection

The table presents several key features, including the platform of the participants (Android or iOS), the format of the collected data, and the storage location of the data. The feature of *Instrumentation* refers to the process of adding extra code to a software program to gather and measure data. This code can log data to a file, send data to a remote server, or perform real-time analysis of the collected data.

The *Test-evaluation* feature indicates the evaluation approach used by the authors. In a Lab study, participants are brought into a laboratory setting to perform pre-determined tasks. In contrast, Log studies (i.e., Real usage) are more natural observations as the systems are used without any influence from experimenters or observers.

The *Scaling* feature highlights whether the research works consider the process of increasing the data collection, storage, and analysis efforts. This table also covers whether the data collection process is initiated automatically or triggered by the user, the type of collected data, the purpose of the studies, and the study object.

We conclude that the vast majority of studies that make use of log files are application-oriented. They are mostly concerned with determining the usefulness of a certain app. Even though the work of [21] focused on mobile UI, they were limited to creating an Android prototype for calculating the user interface complexity of any mobile application.

2.2 Apps usage behavior studies

The use of smartphone apps has the potential to transmit a considerable amount of personal information. As indicated previously, an increasing number of studies have been conducted in recent years to detect how users utilize smartphone apps.

The purpose of studying app usage behavior varies among research works. Some work such as [22] used mobile app use and lagged-ensemble machine learning to predict work fatigue and boredom. Other works such as [23] assessed the relationship between different

personality traits and mobile app usage behavior, and this by using the well-known Big Five personality traits taxonomy. Other researchers such as [24] conducted an empirical study in order to identify the types of learners based on the use of learning apps. The COVID-19 pandemic served as a catalyst for this study since it has shifted people's attention back to digital learning, and many online courses and digital language learning systems have attracted numerous language learners.

We synthesize in the table 2 below that there are many more complex and diverse behaviors that make up the smartphone user population. The table includes the used data type, population size, the operating system type of the user's smartphone, and the time period of the conducted experiment. The main goal of each research work as well as the disadvantages of the methods are listed.

As we can see from table 2, the study scale can vary in a large range, from 14 users to more than hundreds of users. The data duration also varies, from 1 week to around 1 year or more. The most commonly used data are user interactions and app usage records, with several works emphasizing the significance of location and time of use. The user's phone in the observed studies varies, ranging from iPhone to Android to Windows Mobile.

While some have demonstrated considerable variety in app usage [26, 25], others have concentrated on examining and reporting app usage averages or ranges of app use across users. Jones et al. [30] identified three distinct clusters of users based on their app revisitation patterns (*Checkers*, *Waiters* and *Responsives*) while Banovic et al. [31] identified four types of users based on their actions on emails displayed on their lock screen (*Non-users*, *Normal users*, *Power users*, and *Cleaners*). However, their scope is limited by only examining relatively simple behaviors such as app revisitation and actions performed on emails.

2.3 *UI adaptation*

When it comes to adapting mobile interfaces, we realize that the main research question revolves around the development of adaptive mobile apps that can be used in context-aware environments. The majority of research focuses on examining the impact of different design mechanisms in assisting the development of mobile apps. These apps can be adaptable to fit different user needs, situations, modes of interactive design, and technological limitations. Indeed, numerous approaches address the continuous adaptation of app interfaces. They demonstrate the wide range of software artifacts that can be used to adapt mobile app interfaces dynamically.

Mobile UI adaptation can have a significant impact on a wide range of industries (e.g., healthcare, finance, education, retail, travel, entertainment, etc.). For instance, mobile UI adaption can enable healthcare providers to access patient data, track vital signs, and collaborate with other healthcare professionals.

Kellett et al. [32] examine the use of a smartphone mobile app technology to record activity times. The purpose is to support time-driven activity-based costing (abbr., TDABC) studies, which are used to determine the cost of providing healthcare services (e.g., time spent on patient care, administrative tasks, and training). The adaption of UI allows for making informed decisions about how to allocate resources and improve the cost-effectiveness of healthcare delivery. However, this study has some potential operational barriers to implementation and does not consider adapting the overall interface of the mobile device itself. It is dedicated to a particular app.

Some research work aimed to improve mental healthcare by introducing new adaptive intervention technology. Lamo et al. [33] collaborated with the end-users by applying user-

Work	Used data	Users	Period	Work Synthesis	Critics
Falaki et al. [25]	User interactions + app use	33 Android + 222 Windows mobile	7-28 Weeks	Explored the mean number of interactions per day for a user, the mean interaction length in seconds, and the number of used apps.	Does not examine the differences in app usage between different types of users.
Xu et al. [26]	App usage records	National level (USA)	1 week	Showed how, where, and when smartphone apps are used at an aggregate level	Did not explore how app usage differs across users or user groups.
Jesdabodi et al. [6]	App usage records	24 iPhone	1 year	Focused on characterizing the usage states that exist between usage sessions + Analyzed how it differ within and between the users.	Did not analyze the app usage behavior of different kinds of users.
Yan et al. [27]	App usage + Location + DateTime	34 iPhones	14 months	Studied mobile device usage and engagement (the time period between unlocking and re-locking the device).	Did not explore how app usage varies across different kinds of users.
Rahmati et al. [28]	App usage records + Location	14 users	4 months	Studied what apps the users used and how their usage changed over time, showing that the usage is highly location-dependent.	Did not identify the differences in usage behaviors among various user groups.
Shin et al. [29]	Sensory data + Mobile app use	23 Android	1 month	Studied the length of usage sessions, the number of sessions per hour or per day, and the number of apps per day at an aggregate level.	Did not observe the variations in usage behavior among different user groups
Jones et al. [30]	App usage records	165 Android	3 months	Identified users clusters based on app revisitation patterns (how often users return to a particular app).	Examined only a relatively simple behavior: app revisitation.
Banovic et al. [31]	User interactions	27 Android	4 weeks	Identified four types of users by analyzing users' actions on emails displayed on their lock screens.	They only look at one pretty straightforward behavior (actions taken on emails)

Table 2 Synthesis of Mobile app usage behaviors research works

centered design (abbr., UCD) principles, developed by Abras et al. [34], to create and evaluate engaging UIs. The authors developed new machine learning (abbr., ML) algorithms and techniques for detecting and classifying patients' behavior and symptoms. The method has been used to tailor the treatment to individual patient's needs and preferences.

The adaptation of user interfaces can enhance the field of Extended Reality (XR) by adapting interface elements to the user's context, resulting in improved user experiences. Evangelista et al. [35] introduced a toolkit to simplify the design and implementation of optimization-based XR adaptation policies. The toolkit addresses implementation challenges such as manual scripting and limitations in realistic XR environments. The toolkit streamlines the design process, making adaptation policy implementation more manageable and efficient.

In [36], the authors employed XML-based UI descriptions to create UIs that adapt to the context of use at run-time. Authors in [37] suggested a UI architecture for run-time adaptations in mobile apps based on mobile agents, and authors in [38] created an XML vocabulary to model dynamic UI adaptations for mobile computing devices. Abrahao et al. [39] developed an ontology-based framework for representing, designing, and supporting the adaptation of UIs in mobile apps using design patterns that are adjusted to the user's needs or preferences as well as the context in which they are used. Abrahao et al. [39] defines a conceptual reference framework for intelligent UI adaptation for model-based UI adaptation.

The referenced research, however, exclusively dealt with personalizing the UI for a specific app. Furthermore, the majority of UI adaptation studies focused evaluate existing and new web app designs and provide developers and designers with a benchmark for choosing the most user-friendly and effective design. To the best of our knowledge, there haven't been any studies that cover mobile UI adaptation based on an app's usage behavior for different groups of users.

2.4 *Review*

Despite the fact that many studies have been conducted on mobile app usage behaviors, they have only scratched the surface in terms of identifying distinct types of users, analyzing app usage in the aggregate, or examining the range of users. They haven't looked into any variations in app usage behavior among different user groups.

Most of the mobile computing studies made no attempt to investigate any differences in app usage behavior across user groups. They investigate the ways users download, install, and use various apps [40], the length of the average app session [31, 4], the number of daily interactions individuals have with apps [25] and the way context affects the use of apps [41, 42].

Furthermore, many studies have been conducted to investigate smartphone apps usage patterns, such as the frequency with which apps are used together [7] and the frequency with which a particular app is revisited [30]. Other works have focused on predicting which apps users are intending to install [43], and use [44, 45, 46]. They treat all smartphone users as though they are the same, which is a convenient and simplifying assumption for app developers and phone manufacturers, besides the fact that it does not correspond to reality.

Adapting the mobile UI based on app usage behavior, especially after identifying various groups of users, is crucial for providing an optimized and personalized user experience. By first identifying user behavior groups based on mobile app usage and then adapting the mobile UI for each group, the interface becomes more responsive to the needs and

preferences of each user. This can lead to a more intuitive, efficient, and seamless experience, where the user's most frequently used or time-consuming apps are given priority in the interface design. Therefore, adapting the mobile UI based on app usage behavior and user behavior groups is an essential step in creating a user-friendly and successful mobile device

By identifying distinct types of mobile users based on their use of smartphone apps, it is possible to develop more accurate user profiles and, as a result, a more accurate interface adaptation. Profiling a user through a smartphone app involves a number of steps, such as collecting data, preprocessing it, and modeling the user. The quality of the user profile is affected by the data that is collected, and informational data helps to build an accurate user profile.

In this study, we aim to identify various types of users and redesign the mobile UI for the different groups of users in a dynamic and customizable way. From this point of view, we decided to collect our own data using an open-source Android toolkit that we upgraded so that log files could be sent automatically and periodically to a predefined Gmail box. We processed and analyzed the data we collected, and we used a few machine learning algorithms on our dataset to find out what kinds of smartphone users there are based on how they use apps.

3 The proposed approach

Our research purpose is to adapt the mobile UI based on the interactions of users with their mobile apps. To achieve our goal, we propose an approach based on two phases.

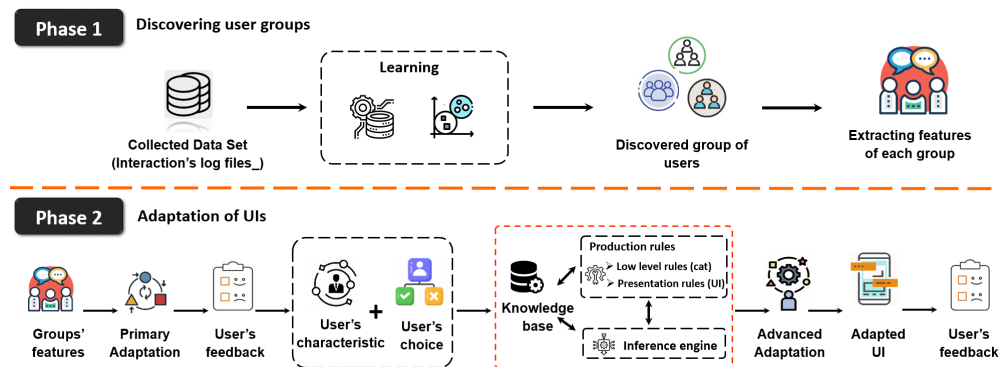


Figure 1 The proposed approach for the adaptation process

As mentioned in Figure 1, the first phase consists of training the model and discovering the different clusters of users. The second phase involves adapting the UI.

We provide two levels of adaptation: primary adaptation and advanced adaptation. The first is performed automatically. This type of adaptation is associated with the user's cluster. It is reliant upon the cluster's characteristics on the one hand, and the UI state on the other (i.e., an overloaded interface). Each cluster has its own adaptation according to its

characteristics. This adaptation is performed once the user’s cluster has been determined. This is covered in detail in section 5.1.1. The second type of adaptation no longer concerns all users in a given cluster. It requires user approval. The user is involved in the UI adaptation process, selecting the adaptation that he finds useful. We go into more detail in section 5.1.2.

4 Phase 1: Discovering users’ clusters

The first phase of our approach aims to train the model and discover the different clusters of users’ behaviors as depicted in Figure 2. Our attention so far, in this phase, is oriented towards achieving the following steps: 1) Gathering data from different participants, 2) Processing the collected data, 3) Building the model by selecting the right ML algorithms, 4) Gaining insights from the model’s results, and 5) Transforming results into visual graphs (i.e., data visualization).

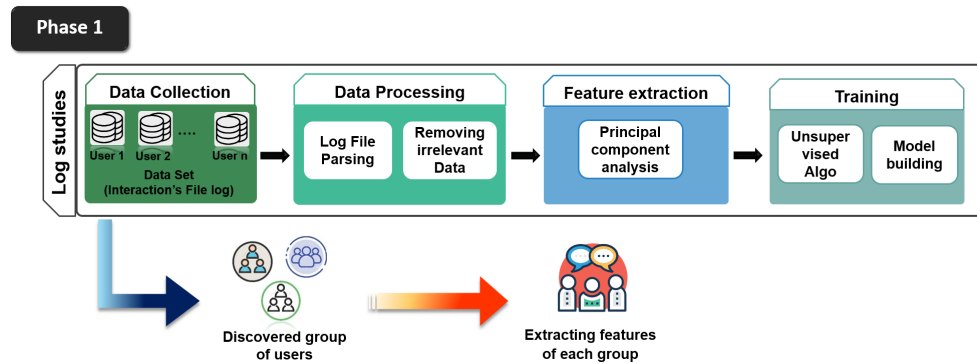


Figure 2 Process of discovering users’ clusters

This phase has row-collected data as input and the newly identified user groups are the output. We extract the features of each of the discovered groups. Subsequently, the distinctive characteristics of each group are identified and utilized as input for the second phase, in order to deliver tailored adaptations to each user group. We go through this section with the different steps mentioned above. We present an updated toolkit for collecting user log files, as well as the clustering methods that were used. We provide an overview and analysis of the collected data. We present an overview of the performance of the model and visualize the results obtained.

4.1 Data Collection

We focus on the Android OS due to its dominance among mobile operating systems and the availability of all essential UI elements found on competing mobile platforms, including app notifications, lock screens, and home screens. Collecting data from diverse mobile users is the first step in building our model. To accomplish this task, we use an open-source Android toolkit called *Automate* proposed by Holzmann et al. [19]. The toolkit allows for flexible use, such as pausing/resuming data collection and modifying settings by selecting

which data to track. The collected data will be stored locally and can be exported as a ZIP archive of CSV files.

We chose this tool because it meets our needs for collecting large numbers of log files from volunteers. However, this tool contains flaws that could hinder our research. The participants have to manually send the generated CSV every 24 hours. This causes the individual to get bored and possibly give up on the experience. To automatically and regularly send the log files to a predefined mailbox, we had to update the app. Figure 3 depicts the log file-sending process after revision. The source code is available on *GitHub*^a.

Accordingly, we took the following steps:

- *Updating of the dependencies (SDK, libraries):* the last activity on the *Automate* code dates back to August 2017. Thus, it was necessary to make an update to cope with Android's latest versions.
- *Adding **UserActivityBroadcastReceiver** component:* It starts the tracking process when the user activates his phone. So the collected data will be sent, this component operates when two conditions are checked: 1) the logs collected are substantial enough (i.e., 8 hours from the last send) 2) the phone's connectivity to send these logs. If two conditions are satisfied, this component will request *MailSenderService* to send the logs.
- *Incorporating the **MailSenderService** service:* It is called each time the *UserActivityBroadcastReceiver* estimates that an email carrying logs should be sent. A Java application (Web or Android) is not able to manage mail sending and receiving unless it is integrated with a mail server. This server has to be always accessible and use standard mail-sending protocols.

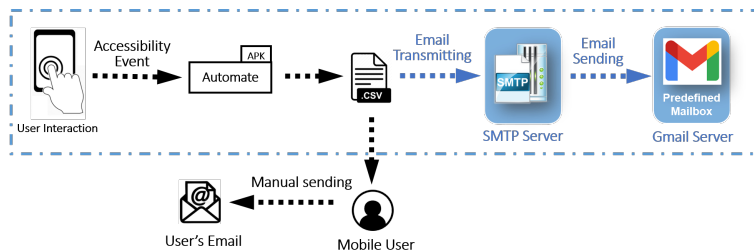


Figure 3 After-revision log file sending process

The *MailSenderService* allows the connection to an *SMTP* server via the *SMTP* protocol using an URL (server URL), a username, and a password. Afterward, this service builds a Message type Object that includes: 1) subject of the message, 2) body of the message, 3) the recipients, and 4) the attachment (i.e., a zip file containing the CSV files).

^a<https://github.com/mint-hagenberg>.

The mail server will return a positive response if the mail is correctly sent to the recipient. If a problem occurs, it will return a negative response and assume that the logs were not sent, and consequently, it will try to send them again until this operation is completed successfully.

The processing of the data is handled by managers, where each one collects a specific type of data and transforms it into various types of structured data (CSV). In our case, we studied the file generated by the manager *App Sequence Manager*.

4.2 Ethical Considerations

We are fully aware of the privacy ramifications of using the acquired data for research purposes. We have taken all necessary precautions to protect the privacy of the involved participants. Additionally, the toolkit is an open-source project and maintained on *GitHub*^b, allowing users to examine it easily. To allay user concerns about privacy, participants are informed of the data collection prior to downloading *Automate*. They provide their consent via their devices.

Participants were recruited through online postings and social media. After installing the APK file, the user has to enable the *accessibility service* when the app is started for the first time. After the permissions are granted, the tracking starts automatically and can be modified in the tracking settings. Hence, no action is taken unless the user gives his or her approval.

The conducted experiment is anonymous. As the user installs the toolkit, a generated user-specific identifier ID is assigned to that user. We only have access to his ID. Thus we can not correlate the user's ID with physical individuals.

4.3 Clustering method

Generally, the choice of algorithms is related to the size of the dataset. In our context, we selected algorithms that correspond best to small and medium-sized datasets.

There are two different clustering methods. The first requires that the number of clusters be fixed beforehand. We tested different algorithms, such as *K-means* and *Spectral clustering* with a different number of clusters. The second automatically determines the number of clusters. We tested *Affinity Propagation* and *Mean Shift*, and we compared the results in section 4.5.

We evaluate the performance of the chosen unsupervised ML algorithms by using *Silhouette score*, *Calinski Harabasz score*, and *Davies Bouldin score*.

4.4 Data Overview

In this section, we describe and analyze our dataset, and we present the exploited features. The basic analysis reveals app usage-related information and allows us to determine the features that could influence user behavior.

4.4.1 Dataset

The dataset we collected contains 37313 usage records in total (entries) resulting from 2212 log files collected from 71 volunteers over a 1-month period of time. The data was

^b<http://mint-hagenberg.github.io/automate-documentation>.

collected and sent nearly every eight hours by the *Automate* Toolkit. The collected log files have been processed to extract the relevant information. Consequently, we needed to delete, add, and translate some attributes.

We were inspired by Zhao et al.'s work [47] in segmenting the daytime and studying users' activity throughout the day: morning (6:01 am to 12:00 pm), afternoon (12:01 pm to 6:00 pm), evening (6:00 pm to 00:00 am), and night (00:01 am to 6:00 am).

In our context, app **category** (e.g., social media, photography, communication, etc.) is an important feature because it has greater significance than just using the application name. The use of the categories of mobile apps leads to a more general and less complex representation of the user's behavior.

In order to identify the type of app, we used *Kaggle* datasets ^c where the categories of mobile apps crawled from Google Play-Store ^d. We distinguish 23 groups of apps. We removed redundancy and added apps to existing categories (i.e., apps used across the country).

4.4.2 Dataset Analysis

The participants in our experiment are heterogeneous. It includes people of different genders, ages, backgrounds, different occupations, and different types of smartphones. After installing the app *Automate* successfully, the participant is invited to fill out a form to get more information about the demographic data. Table 3 depicts the percentages of users in the following demographic attributes: gender, age range, and function.

Percentage of users with a specific demographic attribute									
Gender		Age range				Function			
Female	Male	20-29	30-39	40-49	50-59	Student	Employee	Employer	Unemployed
56.2%	43.8%	68.7%	22.9%	6.4%	2%	54.1%	38.7%	5%	2.2%

Table 3 Percentage of users in our dataset in demographic attribute

Based on the remaining users' demographics, we see that our dataset has more female users than male users. Almost 69 (68.7%) of users are between the ages of 20 and 29, with the majority of them being students. We divided the dataset into two sub-datasets. The first contains all of the categories, while the second only considers the top 12 (less than 200 uses). The rest are put in a category that we call "Others".

The analysis carried out shows that users have the most activities in the afternoon (from 12:01 pm to 6:00 pm) and much less at night (from 12:01 am to 6:00 am), as illustrated in figure 4.

By analyzing both datasets, we noticed the following:

- The top five categories having the highest number of occurrences among the datasets are, respectively, *Social Media*, *Communication*, *Productivity*, *Life Style*, and *Map and Navigation*.
- Users tend to use their smartphones more in the afternoon than in the evening (with a slight difference in the morning).

^c<https://www.kaggle.com/bandikarthik/googleplaystoreapps>

^d<https://www.kaggle.com/gauthamp10/google-playstore-apps?select=Google-Playstore.csv>

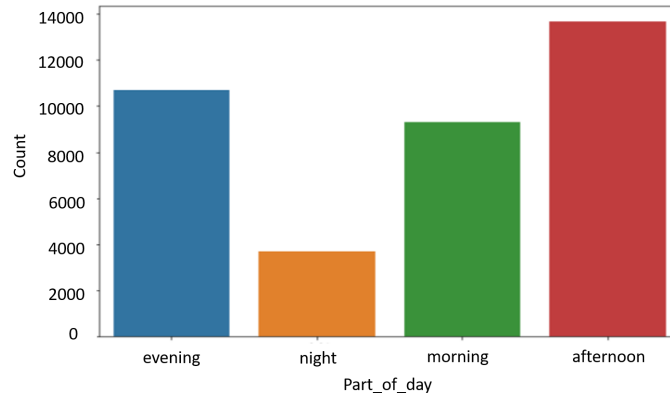


Figure 4 Frequency of use in each part of the day

- The top 5 categories where the user spends more time are: *Life Style*, *Social Media*, *Personalisation*, and *Weather Games*.
- The top five categories having the highest frequency of use are, respectively *Life Style*, *Personalisation*, *Social Media* and *Weather, Games*.

Analyzing the collected data is a necessary step to learning more about app usage behavior and deciding which features influence users' behavior the most.

4.5 Experiment Results

Our dataset includes features such as the “*Frequency*” and “*Duration*” of use of each category per day, and in each part of the day. We tried to merge “*Frequency*” and “*Duration*” as an added feature. We split our dataset into two sub-datasets. The first contains all categories (23 categories), while the second contains only categories below the threshold of 200 uses (12 categories). We combined similar categories like “*Games*” with “*Entertainment*”, the “*Calendar*” with “*Productivity*”. The rest (e.g., education, sports) are grouped under “*Others*”.

We collected log files from 71 volunteers. However, some users' collection periods are as short as a few hours. We deleted these and then tested both datasets on 61 users.

We measure, in the following subsections, the performance of the applied algorithms on our two sub-datasets through the evaluation metrics mentioned in section 4.3. We examine user behavior clusters and report algorithm performance in each dataset with various features: duration, frequency, frequency & duration per day, as well as frequency, duration, and frequency & duration in each part of the day.

4.5.1 Dataset 1 results

We summarize the performance of the various algorithms in table 4.

The results show that the algorithms *K-means* and *Spectral Clustering* outperform the other algorithms based on *Silhouette Score* and *Calinski Harabasz Score*. The algorithm

Adapting Mobile User Interfaces based on users' application usage behaviors 13

Features & Metrics	Duration	Frequency	Frequency & Duration	Dur part_of_day	Freq part_of_day	Freq & Dur part_of_day
3*Silhouette Score	K-means/ Spec Clust 0.38 k=2	K-means/ Spec Clust 0.28 k=2	K-means 0.36 k=2	K-means / Spec Clust 0.536 k=2	K-means 0.48 k=2	K-means /Spec Clust 0.5 k=2
3*Calinski Harabasz Score	K-means 10.0 k=2	K-means 12.31 k=2	K-means 9.58 k=2	K-means 11.84 k=2	K-means 10.36 k=2	K-means 11.04 k=2
3*Davies Bouldin Score	Mean Shift 0.48 k=9	Mean Shift 0.6 k=10	Mean Shift 0.5 k=10	Mean Shift 0.6 k=11	Mean Shift 0.5 k=12	Mean Shift 0.6 k=13

Table 4 Dataset 1 summary table

K-means has the best scoring with the attribute *Frequency part_of_day*(0.53). We obtained 2 clusters, the first with 48 users and the second with 8 users. The best score of *Calinski Harabasz* measurement is derived with the attribute *Frequency*, where $k = 2$ and the distribution is 36 users and 20 users.

On the other hand, Mean Shift provides better results with the feature *Duration* according to *Davies Bouldin*' measure (0.48). However, the distribution of clusters ($k = 9$) is not balanced. The first cluster contains 47 users, and the other clusters each contain only one user.

4.5.2 Dataset 2 results

The performance of algorithms is summarized in table 5.

Features & Metrics	Duration	Frequency	Frequency & Duration	Dur part_of_day	Freq part_of_day	Freq & Dur part_of_day
3*Silhouette Score	Agglomerative Clustering 0.37 k=2	K-means 0.55 k=2	K-means / Spectral Clustering 0.49 k=2	Spectral Clustering 0.51 k=2	Spectral Clustering 0.47 k=2	Spectral Clustering 0.46 k=2
3*Calinski Harabasz Score	K-means 11.56 k=2	K-means 14.45 k=2	K-means 11.97 k=2	K-means 13.99 k=2	K-means 11.59 k=2	K-means 12.4 k=2
3*Davies Bouldin Score	Agglomerative Clustering 0.47 k=2	Agglomerative Clustering 0.5 k=8	Mean Shift 0.4 k=10	Mean Shift 0.6 k=2	Mean Shift 0.5 k=12	Mean Shift 0.6 k=13

Table 5 Dataset 2 summary table

The *K-means* and *Spectral Clustering* algorithms are the most efficient using the *Frequency* feature according to the *Silhouette Score* (0.55) and *Calinski Harabasz Score* (14.45) metrics with $k = 2$. The first cluster has 46 users. while the second has 15. On one hand, the *Agglomerative Clustering* algorithm works well with the *Duration* feature according to *Davies Bouldin Score*. On the other hand, a good result is obtained by using the *Mean Shift* and *Agglomerative Clustering* algorithms, according to *Davies Bouldin*'s measure.

4.5.3 Cluster analysis

We chose the best-performing features based on the assessment metrics to analyze users' behavior. We used the *Calinski Harabasz Score* and *Silhouette Score* scores to avoid the unbalanced distribution of *Davies Bouldin* scores.

Observing the results, the feature *Frequency* has the best results in both datasets for the assessment measure *Calinski Harabasz Score* respectively (12.314) and (14.45) with a balanced distribution. We chose to study the first dataset attribute, *Frequency Part_of_day*.

The use of apps during the day varies between the two clusters. Users in the first cluster, as shown on the left side of figure 5 are more active in the morning and in the afternoon, while those in the second cluster, on the right side, are more active in the afternoon and the evening. In addition, users in the second cluster are more active at night than those in the first one.

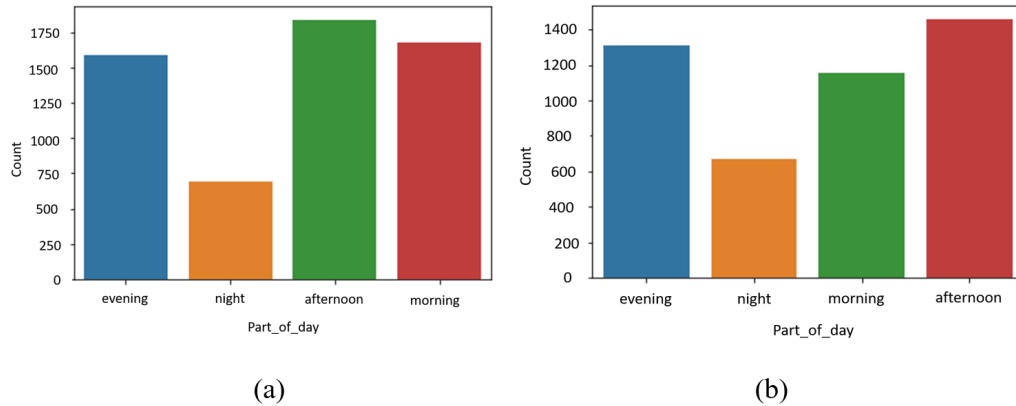


Figure 5 Period of day usage statistics: (a) Frequency of use in each part of the day for cluster 1; (b) Frequency of use in each part of the day for cluster 2

By examining the groups obtained, we observe the following points:

- The first group's users are more communicative. The most used categories of apps are *Social Media*, *Communication*, and *Phone and SMS*. The category *Social Media* is the most used in the evening with an average frequency of use of 147, while *Communication* in the afternoon with an average frequency of use of 89. The average frequency of use of the category *Phone and SMS* equals 47.
- Compared to the second group of users, the first group of users uses the *Productivity* category more frequently in the morning, afternoon, and evening, with an average frequency of use equal to 65. They prefer the *Entertainment* category in the morning and evening, and the *Games* category in the afternoon. Users of this group use the *Maps and navigation* category much more than the second group's users in the morning and afternoon.
- Users of the first group use the *Weather* and *Music and Audio* category less than those of the second group. On the other hand, the users of the second cluster are less communicative and more active during the night hours.
- The users of the second group use the *Lifestyle* and *Shopping* apps much more than the users of the first group and this is during the morning, afternoon, and evening with an average frequency of use equal to 98. They also use the *Weather*, *Video Players and Editors* categories in the evening, and *Music and audio* more than the first group.

5 Phase 2: Interface adaptation

Our research goal is to adapt the mobile UI based on how users interact with their mobile apps. We introduce in this section the adaptation phases, and we present the adaptation rules that contribute to the delivery of an adapted user interface. We also present some case studies.

5.1 Adaptation phases

We suggest two stages of adaptation: the primary adaptation, which is carried out automatically once the user's cluster is detected, and the advanced adaptation, which requires the user's approval.

5.1.1 Primary adaptation

This type of adaptation is based on the cluster's features on the one hand and on the UI state on the other (i.e., an overloaded interface). Each cluster has its own adaptation according to its characteristics, such as the most and least used categories, the most and least used apps, and the most frequently used apps in each part of the day. Hence, it is necessary to first identify the user's cluster.

(A) **Adaptation method:** A smartphone typically consists of multiple screens, including the home screen and other screens. Thus, it is crucial to identify “*which adaptation should be applied to which screen?*”. As an answer to this question, we propose the following:

– **Adaptation performed on home screen:**

We presumed that mobile users make the most use of the home screen. For accessibility purposes, we assume that the user moves to the home screen apps that he or she deems essential.

Therefore, we propose: *i) relocating apps* where the most frequently used apps are moved to the home screen (if they are currently on another screen and are among the TOP 5 most frequently used apps), *ii) zooming-in* the icons of the most utilized apps at each time of day.

– **Adaptation performed on other Screen:**

Similarly, we assume that the other screens are used less frequently than the home screen. We propose: *i) relocating apps* so that the apps with the lowest usage are moved to other screens, leaving the home screen with only the apps the user requires, *ii) ungrouping* a frequently used app from an existing group and facilitating access to the app, *iii) grouping* the five apps with the lowest frequency of use and labeling the group “least used apps”.

The user may have several screens. We decided to apply these changes to the screen that appears after the home screen.

We would like to point out that both of the adaptations mentioned above are realized simultaneously.

(B) **Adaptation algorithm:**

The input for our algorithm includes the user's home screen and the screen they use most frequently, and information about their app usage patterns compared to their cluster. The output of the algorithm is either one or two adapted interfaces, with the latter being generated only if the user has a secondary screen beside their home screen.

As *cluster's features* and the *interface overcrowdness* are considered in primary adaptation, we set as an input the state of the screens to be adopted.

According to the user's cluster, we start extracting clusters' features, such as the top 5 most used categories ($C_iTop5cat[]$), the top 5 most used apps ($C_iTop5apps[]$), as well as the least used apps ($C_iLeast5apps[]$) and the top 5 most used apps in each part of the day ($Top5appsByPoD[]$).

Algorithm 1 Primary adaptation

Require: $\exists(\text{User}_j \in \text{Cluster } C_i), \text{overloadedHS}, \text{overloadedScreen}, \text{distributed}$

- 1: $\text{change} \leftarrow \text{false}$
- 2: $\text{New_}C_i\text{Top5cat}[] \leftarrow \text{null}$
- 3: $\text{part_of_day} \leftarrow \{\text{morning}, \text{afternoon}, \text{evening}, \text{night}\}$
- 4: **for** (features in C_i) **do**
- 5: $C_i\text{Top5cat}[] \leftarrow \{\dots\}$;
- 6: $C_i\text{Top5apps}[] \leftarrow \{\dots\}$;
- 7: $C_i\text{Least5apps}[] \leftarrow \{\dots\}$;s
- 8: **end for**
- 9: **if** ($\text{New_}C_i\text{Top5cat}[] \neq \text{null}$) **then**
- 10: $\text{change} \leftarrow \text{true}$;
- 11: **else**
- 12: $\text{New_}C_i\text{Top5cat}[] \leftarrow C_i\text{Top5cat}[]$;
- 13: **end if**
- 14: **while** ($\text{change} = \text{false}$) **do**
- 15: **for** (time in part_of_day) **do**
- 16: Highlight $\text{Top5appsByPoD}[]$;
- 17: **end for**
- 18: **for** ($C_i\text{Top5cat}[]$) **do**
- 19: **for** ($C_i\text{Least5apps}[]$) **do**
- 20: /* Home screen adaptation*/
- 21: **if** ($C_i\text{Least5apps}[k] \in \text{homeScreen}$) **then**
- 22: Relocate the app to the other screen ;
- 23: **end if**
- 24: **if** ($\text{distributed} = \text{true}$) & ($\text{overloadedHS} = \text{false}$) **then**
- 25: Relocate the app to the home screen ;
- 26: **end if**
- 27: /* The other screen adaptation*/
- 28: **if** ($\text{overloadedScreen} = \text{false}$) & ($\text{Top5appsByPoD}[l] \in \text{group}$) **then**
- 29: Ungroup the app from the group ;
- 30: **end if**
- 31: **if** ($\text{overloadedScreen} = \text{true}$) **then**
- 32: Group least used apps and label it “Least usedApp”;
- 33: **end if**
- 34: **end for**
- 35: **end for**
- 36: **end while**

During this adaptation phase, we check the screen state to be adapted, where “*overloadedHS*” and “*overloadedScreen*” reports respectively whether the home screen and the other screen are overloaded with apps or not. The *distributed* variable shows if the Top 5 apps are spread out or if they are all on the same screen.

To cope with the cluster’s changes, we made sure to update the cluster’s features after adaptation. If the list of the top 5 most used categories changes, we update the list

($New_C_iTop5cat[]$); otherwise, if ($C_iTop5cat[]$) remains the same as when we started the adaptation, *change* remains *false* and the adaptation continues.

The user might not like the changes that have been made. He might cancel some of the provided adaptations. Consequently, these changes won't be made for this user during the next phase of adaptation. This implicit feedback may serve as a launching point for advanced adaptation.

5.1.2 Advanced adaptation

The user may not appreciate the primary adaptations, which are based on the cluster's features and the interface state (i.e., overcrowded). The advanced adaptation takes the user's preferences into account when making the adaptation. We think that involving the user in the process of UI adaptation could help increase the smartphone's usefulness. He selects the adaptation based on what he desires or what he considers useful. A user profile is created based on the gathered data, which includes:

- **User characteristics:** It includes details, namely the user's age (needs associated with the elderly), physical disabilities (such as visual or auditory impairment, difficulty in selection), and expertise level (beginner, expert).
- **User choices:** It includes the choices made by the user. The user can choose the interface on which to implement the adaptation (the home screen or other screens). The user has the option to accept or decline the various adaptations. If the user declines, the subsequent adaptation suggestion is displayed. The suggested adaptations are listed below.
 - **Grouping apps:** It entails grouping the most and least used apps into one group, each separately.
 - **Ungrouping apps:** It consists of dissociating one or more of the frequently used apps from an existing group. It is crucial to note that we take into account whether or not the target screen is overloaded. We also ensure that the group was not recently formed.
 - **Relocating apps:** It consists of transferring one or more apps from one screen to another. The user can choose to:
 - * Relocate the *most used apps* from an existing screen to the home screen. If it is overloaded, the adaptation will be canceled.
 - * Relocate the *least used apps* from the home screen to another screen that the user chooses. If the target screen is overloaded, the adaptation will be canceled.
 - * Relocate both the *most used apps* and the *least used apps*. The same conditions will be applied.
 - **Rearranging screen:** It consists of reorganizing the apps on the user-selected screen. For the frequently used apps, the user can choose the area of the screen (top, middle, or bottom) and the order of realignment (from right to left or the other way around).
 - **Zooming-in apps:** It consists of increasing the most-used apps' icon size according to each part of the day.

The purpose of this adaptation is to provide easy access to the apps that users use the most. In light of this, we set up adaptation rules to make sure that each user got the right adaptations.

5.2 *Adaptation rules*

Our main focus in this stage is the selection of consistent adaptation rules considering the user's choice and the current context information such as UI state. In artificial intelligence, various production systems have been found in literature, such as rule-based expert systems [48]. The rule's syntax is composed of two parts:

IF($condition_1 * condition_2 * \dots * condition_n$)**THEN***conclusion*

The proposed production system is made up of four subsets of production rules: general rules, interface rules, user rules, and app category rules.

5.2.1 *General rules*

We have established rules regarding the battery level. The main goal is to save energy so that the user can interact with the system for as long as possible. If the battery level is below 20%, then offer to close apps operating in the background until the next recharge. If the battery level reaches 5%, then force the shutdown of apps running in the background, and this until the next recharge.

5.2.2 *Interface rules*

The interface rules consider the interface's state and the user's choices. It consists of the selected screen and the various adaptation types outlined in the preceding section 5.1.2. Depending on the time of the day, the provided adaptation is applied. The user can choose to apply the adaptation either on the main screen or on another screen. If the user's smartphone has multiple screens, the user will select the screen that best suits him.

5.2.3 *User rules*

It includes the user's features such as disability, age, and expertise level. User disability encompasses visual and auditory problems in addition to selection inability.

We thought about both the case where the user is blind and the case where the user has just trouble seeing. When the user is unable to see, he is offered many adaptations, like reciting the notifications aloud, vibrating when an app is selected, and pronouncing the app's name when selecting it.

When the user is visually impaired, various adaptations are provided, like increasing system volume (i.e., ringing phone, notification sound) and enhancing the font and notification size. Users should be able to visualize all the screen items, and the notifications should be easy to read. We provide further insight in the table 6.

We have included the user's age in the list of user characteristics in order to meet the requirements specific to the elderly. We offer less crowded UIs (i.e., informational density). We assist seniors by providing them with graphical support and guidance.

For users who have trouble selecting items, we recommend making the selection fields larger so that they can press the appropriate spot on the screen. We also considered the user's level of expertise, particularly for novices where permanent assistance is offered.

	Context	Resulted adaptation	Type ^a
Primary adaptation	UI	Grouping the most/least used apps into a group	visual
	UI	Ungrouping a frequently used app from an existing group	visual
	UI	Relocating apps from one screen to another - Relocating the most used apps to the home screen - Relocating the least used apps to an existing screen	visual
(Automatic adaptation)	UI	Zooming-in apps: increasing the apps' icon size	visual
	UI	Reorganizing the apps on the user-selected screen (all the aforementioned adaptations)	visual
	UI	Hide notifications from other apps	visual
	App category	Increase the currently operating app' volume	auditory
	App category	Reduce the other apps' volume	auditory
	App category	Switch to silent mode	auditory
	App category	Mute the upcoming notifications (display notifications)	auditory
	App category	Temporarily pause background apps	system
	App category	Set a reminder and add it to the calendar	system
	App category	Vibrate when an app is selected	sensorial
Advanced adaptation (With user approbation)	User criterion	Vibrate when notifications are received	sensorial
	User criterion	Enhance the font size and size of notifications	visual
	User criterion	Only keep the most frequently used apps on the home screen	visual
	User criterion	Opting for a flat screen (without grouping)	visual
	User criterion	Increase the screen size	visual
	User criterion	Flashing screen when receiving a call or notification	visual
	User criterion	Provide voice guidance	auditory
	User criterion	Provide I/O auditory interfaces	auditory
	User criterion	Pronounce the app's name when selecting it	auditory
	User criterion	Recite the received notifications aloud	auditory
	User criterion	Increase the system's volume (ring phone, notification sound)	auditory
	User criterion	Provide permanent voice assistance	auditory

^aNote: On occasion, a hybrid adaptation is offered.

Table 6 A summarizing table of the proposed adaptations

5.2.4 App category rules

The category-specific adaptation is meant to make sure that the user stays focused on the task at hand and gives the app being used his or her full attention.

We chose to cover adaptation for 11 of the 23 categories, which are as follows: *productivity, education, photography, health & fitness, communication, sport, games, music & audio, entertainment, maps & navigation, and video players and editors.* Table 6 lists all of the possible adaptations that we propose.

After laying out the rules for UI adaptation, the next section shows some case studies for both basic and advanced adaptations.

5.3 Case studies

As stated previously, the user has the ability to personalize the UI as he deems appropriate. We reviewed examples of users with distinct preferences here:

- **User 1:** is a 34-year-old male user with no physical disabilities. He belongs to the first cluster. He set up his screens to suit his needs. The user prefers that the changes be made on the home screen. As a form of adaptation, he chooses to remove a frequently used app from a group and move other apps from one screen to another. He agreed on 1) hiding notifications from other apps 2) increasing the currently operating app's volume and reducing the other apps' volume 3) muting the upcoming notifications (display notifications).



Figure 6 The adaptation made to the first user's interface: (a) Home screen primary adaptation; (b) Second used screen primary adaptation; (c) Home screen advanced adaptation

The left side of the figure 6 depicts the primary adaptation, where the top 5 most-used categories are listed on the home screen. The middle figure shows the second-most-used screen primary adaptation, where the top 5 least-used apps are grouped and labeled. The right side of the figure depicts the advanced adaptation, where the app "strong" is moved to the home screen as it is among the 5 most used applications.

User 1 prefers the advanced adaptation over the primary adaptation, according to a 1–5 scale (1: not satisfied to 5: very satisfied). He gave the primary adaptation a score of 2 and the advanced adaptation a score of 3.

- **User 2:** is a 24-year-old male user. He belongs to the second cluster. He has no particular organization for the screen, and he gets used to the initial configuration. The user prefers that the changes be made on the home screen. He chooses the option of relocating apps from one screen to another. He agreed to only mute but display the upcoming notifications.

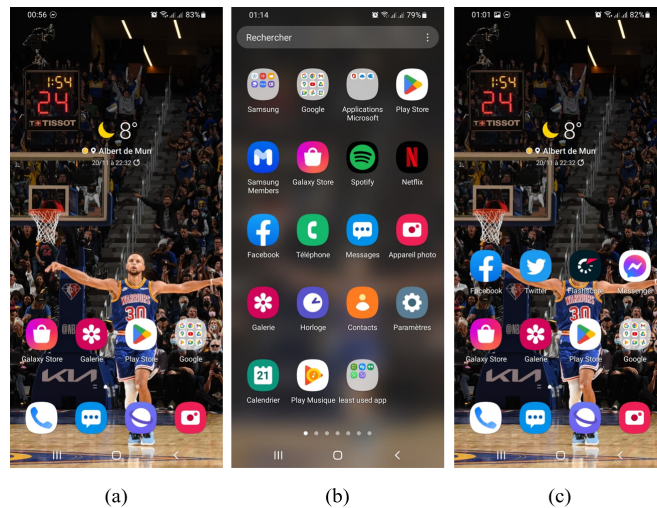


Figure 7 The adaptation made to the second user’s interface: (a) Home screen primary adaptation; (b) Second used screen primary adaptation; (c) Home screen advanced adaptation

Figure 7 represents the user’s main screen prior to adaptation. According to her chosen adaptation, the most-used apps were grouped and transferred to the home screen, as depicted on the right side of the figure.

The left side of the figure 7 depicts the primary adaptation, where the top 5 most-used categories are listed on the home screen. The middle figure shows the second-most-used screen primary adaptation, where the top 5 least-used apps are grouped and labeled. The right side of the figure depicts the advanced adaptation, where the apps “Flashscore”, “Twitter”, “Facebook”, and “Messenger” are moved to the home screen as they are among the 5 most-used applications. User 2 is neutral. He rated both the primary adaptation and the advanced adaptation as 3.

- **User 3:** is a 21-year-old female. She belongs to the first cluster. She modified the initial configuration to accommodate her app usage and places the most important apps on the home screen. The user prefers to adapt the second-most-used screen. She chooses the option of ungrouping a frequently used app from an existing group and grouping the least used apps. She agreed on hiding notifications from other apps and switch to silent mode.

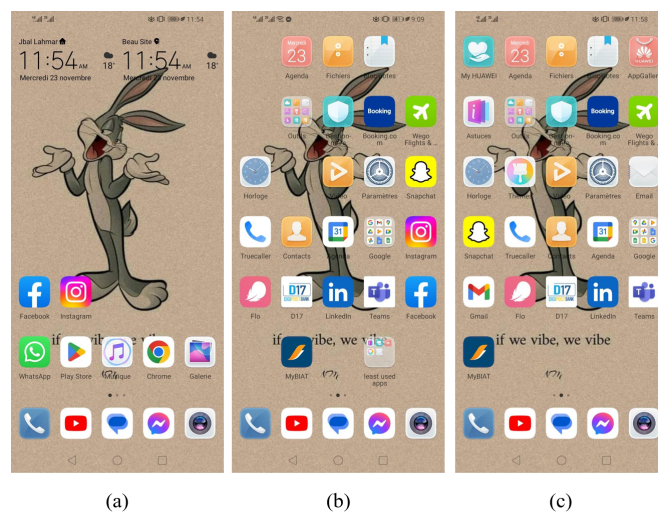


Figure 8 The adaptation made to the third user's interface: (a) Home screen primary adaptation; (b) Second used screen primary adaptation; (c) Second used screen advanced adaptation

The left side of the figure 8 depicts the primary adaptation, where the top 5 most-used categories are listed on the home screen. The middle figure shows the second-most-used screen primary adaptation, where the top 5 least-used apps are grouped and labeled. On the right side of the figure, the advanced adaptation is shown, in which the app "Gmail" is separated from the group called "Google". The third user seems to not like the adaptations since she gave both the primary adaptation and the advanced adaptation a 1 rating.

We presented a series of case studies involving 15 participants, including individuals with disabilities such as visual impairment and hearing difficulties. Our experiment revealed that users have varied approaches to organizing their interface. Some made minimal changes to their screens, while others rearranged apps for improved accessibility and personalized ease of use.

The feedback we received showed a diverse range of preferences. Some participants had no preference, some preferred the advanced adaptation over the primary adaptation, and others chose to keep the original adaptation provided. These findings have important implications for the design of smartphone interfaces. They suggest that there is a need for more customization options in interfaces to meet the unique needs and preferences of different users.

Through the received feedback, we noticed the importance of considering accessibility requirements, particularly for individuals with disabilities, when designing interfaces. Interfaces that are easy to use and accessible to all users, including those with disabilities, can lead to a better overall UX.

6 Discussion

Understanding users through smartphone app usage analysis is considered an encouraging approach. Even still, some challenges remain. In this section, we discuss the outcomes of our work as well as our research's shortcomings and challenges.

6.1 Benefits

Most studies that analyze smartphone usage data to understand users use an existing dataset based on a recent task list [49, 40, 50, 51, 47]. Its low sampling rate (data is collected once every hour) causes limitations. In fact, tracking app usage duration, frequency, and order of task list changes is not feasible. The logs we gathered reveal much about user behavior, including how long each app is used, how often it is used, and the order of the most utilized apps, either by the time of day or throughout the day.

We discovered two user groups. Users in the first group are more active in the afternoon and morning and are more communicative, with social media and communication being the most popular categories. This cluster's users utilize more productivity-related apps than the other cluster's users. Aside from that, users in the second group are more active in the afternoon and evening. Lifestyle apps are the most popular in this group, and users check the weather more frequently than users in the first cluster.

Identifying more than one group shows that treating all smartphone users the same is erroneous. Whoever has influence over the types of apps installed on phones and how users choose which apps to use can no longer treat users as a homogeneous group.

To counter this assumption, numerous parties may be involved, such as smartphone manufacturers and mobile app developers. Involving these parts can be advantageous, as each can contribute to improving smartphone user interaction.

This paper demonstrates the following advantages of the proposed approach when compared to the existing works: (1) adapting the smartphone's user interface based on how the user interacts with apps, as opposed to previous works that adapted the UI for a specific app; (2) adapting the UI by taking into account human factors such as user characteristics (i.e., difficulty in selection, visual or auditory impairment, etc.) and other factors including the app category and interface overcrowding; (3) Unlike other research works that propose approaches for pre-conception interface adaptation, our approach can be applied to post-interface conception.

6.2 Limitations

Although user behavior is used to identify distinct categories of users from collected log data, this study has drawbacks. Many studies stress that demographic information has a significant impact on app usage behavior. However, the conducted experiment is anonymous. Hence, we did not benefit from users' gender, age, or profession.

During data processing, we observed that the data collection period is not uniform (normally 1 month). Some users' collection periods are as short as a few hours. Others range from 11 to 28 days. This could be due to the user uninstalling or deactivating the app, or the OS shutting it down because it runs in the background.

Users of the two clusters have many app usage behaviors in common, making it difficult to label each cluster. As a consequence, we notice that the model requires minor adjustments to improve its performance.

7 Conclusion

Our research intends to design an adaptive mobile UI for distinct user groups depending on their app usage behaviors. We proposed, accordingly, a two-phased approach. The first phase involves discovering the different clusters of users by exploring the potential of Machine Learning (ML) algorithms, while the second involves adapting the UI.

We offer two levels of adaptation. The primary adaptation is based on the cluster's features. It is applied automatically once the user's cluster is detected. The Advanced type no longer concerns all users of a given cluster. The user is engaged in the process, and some adaptation requires his approval. We provided a variety of adaptations, including visual, audio, sensory, hybrid, and system adaptations. We established adaptation rules to provide each user with the appropriate adaptations.

Through the obtained result, we prove that the simplified hypothesis that all smartphone users are similar should not be applied and that mobile users can no longer be treated as if they are all part of one big group. We discussed that many parties could be involved and benefit from identifying the different groups of users.

We demonstrate through the result that the existence of different interface preferences, including the choice between the home screen or another screen, refutes the previously held belief that all smartphone users have a uniform preference for maintaining an unchanged home screen. This proves the importance of considering adapting the overall interface of the mobile device itself, rather than adapting the interface of a particular mobile app as most of the research work does. The observation that some participants chose to maintain the provided adaptation indicates that our hypothesis holds promise and can be further developed by incorporating certain relevant variables.

As a perspective, we plan to collect data from a targeted user group to test our approach and ensure that our experiments and results can be reproduced. We intend to enhance our model's performance and add a third phase that uses fine-tuning for continuous improvement and sends the noticed changes to the model so it can cope with them. We intend to investigate the *frequency* and *periodicity* of the adaptation in order to deliver the appropriate amount of adaptation at the appropriate time.

References

- [1] Rafael Pérez-Torres, César Torres-Huitzil, and Hiram Galeana-Zapién. Power management techniques in smartphone-based mobility sensing systems: A survey. *Pervasive and Mobile Computing*, 31:1–21, 2016.
- [2] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell. A survey of mobile phone sensing. *IEEE Communications magazine*, 48(9):140–150, 2010.
- [3] Marco Conti, Andrea Passarella, and Sajal K Das. The internet of people (iop): A new wave in pervasive mobile computing. *Pervasive and Mobile Computing*, 41:1–27, 2017.
- [4] Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. Falling asleep with angry birds, facebook and kindle: a large scale study on mobile application usage. In *Proceedings of the 13th international conference on Human computer interaction with mobile devices and services*, pages 47–56, 2011.

- [5] Matthias Böhmer, Christian Lander, and Antonio Krüger. What's in the apps for context? extending a sensor for studying app usage to informing context-awareness. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, pages 1423–1426, 2013.
- [6] Chakajkla Jesdabodi and Walid Maalej. Understanding usage states on mobile devices. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 1221–1225, 2015.
- [7] Vijay Srinivasan, Saeed Moghaddam, Abhishek Mukherji, Kiran K Rachuri, Chenren Xu, and Emmanuel Munguia Tapia. Mobileminer: Mining your frequent patterns on your phone. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, pages 389–400, 2014.
- [8] Vladan Radosavljevic, Mihajlo Grbovic, Nemanja Djuric, Narayan Bhamidipati, Daneo Zhang, Jack Wang, Jiankai Dang, Haiying Huang, Ananth Nagarajan, and Peiji Chen. Smartphone app categorization for interest targeting in advertising marketplace. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 93–94, 2016.
- [9] Zhen Qin, Yilei Wang, Hongrong Cheng, Yingjie Zhou, Zhengguo Sheng, and Victor CM Leung. Demographic information prediction: A portrait of smartphone application users. *IEEE Transactions on Emerging Topics in Computing*, 6(3):432–444, 2016.
- [10] Abdulrahman Al-Molegi, Izzat Alsmadi, and Antoni Martínez-Ballesté. Regions-of-interest discovering and predicting in smartphone environments. *Pervasive and Mobile Computing*, 47:31–53, 2018.
- [11] Nicholas Micallef, Erwin Adi, and Gaurav Misra. Investigating login features in smartphone apps. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pages 842–851, 2018.
- [12] Yi Ouyang, Bin Guo, Tong Guo, Longbing Cao, and Zhiwen Yu. Modeling and forecasting the popularity evolution of mobile apps: a multivariate hawkes process approach. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(4):1–23, 2018.
- [13] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, Francisco Flórez-Revuelta, Susanna Spinsante, and Maria Canavarro Teixeira. Identification of activities of daily living through data fusion on motion and magnetic sensors embedded on mobile devices. *Pervasive and Mobile Computing*, 47:78–93, 2018.
- [14] Susan Dumais, Robin Jeffries, Daniel M Russell, Diane Tang, and Jaime Teevan. Understanding user behavior through log data and analysis. In *Ways of Knowing in HCI*, pages 349–372. Springer, 2014.
- [15] Florence Balagtas-Fernandez and Heinrich Hussmann. A methodology and framework to simplify usability analysis of mobile applications. In *2009 IEEE/ACM International Conference on Automated Software Engineering*, pages 520–524. IEEE, 2009.

- [16] Xiaoxiao Ma, Bo Yan, Guanling Chen, Chunhui Zhang, Ke Huang, Jill Drury, and Linzhang Wang. Design and implementation of a toolkit for usability testing of mobile apps. *Mobile Networks and Applications*, 18(1):81–97, 2013.
- [17] Wolfgang Kluth, Karl-Heinz Krempels, and Christian Samsel. Automated usability testing for mobile applications. In *WEBIST (2)*, pages 149–156, 2014.
- [18] Denise Marczal and Plinio Thomaz Aquino Junior. Behavioural variables analysis in mobile environments. In *International Conference of Design, User Experience, and Usability*, pages 118–130. Springer, 2015.
- [19] Clemens Holzmann, Dustin Steiner, Andreas Riegler, and Christian Grossauer. An android toolkit for supporting field studies on mobile devices. In *Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia*, pages 473–479, 2017.
- [20] Xavier Ferre, Elena Villalba, Héctor Julio, and Hongming Zhu. Extending mobile app analytics for usability test logging. In *IFIP Conference on Human-Computer Interaction*, pages 114–131. Springer, 2017.
- [21] Andreas Riegler and Clemens Holzmann. Measuring visual user interface complexity of mobile applications with metrics. *Interacting with Computers*, 30(3):207–223, 2018.
- [22] Damien Lekkas, George D Price, and Nicholas C Jacobson. Using smartphone app use and lagged-ensemble machine learning for the prediction of work fatigue and boredom. *Computers in Human Behavior*, 127:107029, 2022.
- [23] Farid Huseynov. Understanding usage behavior of different mobile application categories based on personality traits. *Interacting with Computers*, 32(1):66–80, 2020.
- [24] Duen-Huang Huang and Hao-En Chueh. Behavioral intention to continuously use learning apps: A comparative study from taiwan universities. *Technological Forecasting and Social Change*, 177:121531, 2022.
- [25] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. Diversity in smartphone usage. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pages 179–194, 2010.
- [26] Qiang Xu, Jeffrey Erman, Alexandre Gerber, Zhuoqing Mao, Jeffrey Pang, and Shobha Venkataraman. Identifying diverse usage behaviors of smartphone apps. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 329–344, 2011.
- [27] Bo Yan and Guanling Chen. Appjoy: personalized mobile application discovery. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*, pages 113–126, 2011.
- [28] Ahmad Rahmati and Lin Zhong. A longitudinal study of non-voice mobile phone usage by teens from an underserved urban community. *arXiv preprint arXiv:1012.2832*, 2010.

- [29] Choonsung Shin, Jin-Hyuk Hong, and Anind K Dey. Understanding and prediction of mobile application usage for smart phones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 173–182, 2012.
- [30] Simon L Jones, Denzil Ferreira, Simo Hosio, Jorge Goncalves, and Vassilis Kostakos. Revisitation analysis of smartphone app use. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1197–1208, 2015.
- [31] Nikola Banovic, Christina Brant, Jennifer Mankoff, and Anind Dey. Proactivetasks: the short of mobile device use sessions. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, pages 243–252, 2014.
- [32] Kyle J Kellett, Ricardo Bertoglio Cardoso, Ana Paula Beck da Silva Etges, Mitchell H Tsai, and Brian M Waldschmidt. A mobile app for the precise measurement of healthcare provider activity times to support time-driven activity-based costing studies. *Journal of Medical Systems*, 46(6):30, 2022.
- [33] Yngve Lamo, Suresh K Mukhiya, Fazle Rabbi, Amin Aminifar, Svein I Lillehaug, Jim Tørresen, Minh H Pham, Ulysse Côté-Allard, Farzan M Noori, Frode Guribye, et al. Towards adaptive technology in routine mental health care. *Digital Health*, 8:20552076221128678, 2022.
- [34] Chadia Abras, Diane Maloney-Krichmar, Jenny Preece, et al. User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications*, 37(4):445–456, 2004.
- [35] João Marcelo Evangelista Belo, Mathias N Lystbæk, Anna Maria Feit, Ken Pfeuffer, Peter Kán, Antti Oulasvirta, and Kaj Grønþæk. Auit—the adaptive user interfaces toolkit for designing xr applications. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pages 1–16, 2022.
- [36] Karin Coninx, Kris Luyten, Chris Vandervelpen, Jan Van den Bergh, and Bert Creemers. Dygimes: Dynamically generating interfaces for mobile computing devices and embedded systems. In *International conference on Mobile human-computer interaction*, pages 256–270. Springer, 2003.
- [37] Nikola Mitrović, Carlos Bobed, and Eduardo Mena. Dynamic user interface architecture for mobile applications based on mobile agents. In *OTM confederated international conferences" on the move to meaningful internet systems"*, pages 282–292. Springer, 2016.
- [38] Mario Bisignano, Giuseppe Di Modica, and Orazio Tomarchio. Dynamic user interface adaptation for mobile computing devices. In *2005 Symposium on Applications and the Internet Workshops (SAINT 2005 Workshops)*, pages 158–161. IEEE, 2005.
- [39] Silvia Abrahão, Emilio Insfran, Arthur Sluÿters, and Jean Vanderdonckt. Model-based intelligent user interface adaptation: challenges and future directions. *Software and Systems Modeling*, pages 1–15, 2021.

- [40] Huoran Li, Xuan Lu, Xuanzhe Liu, Tao Xie, Kaigui Bian, Felix Xiaozhu Lin, Qiaozhu Mei, and Feng Feng. Characterizing smartphone usage patterns from millions of android users. In *Proceedings of the 2015 Internet Measurement Conference*, pages 459–472, 2015.
- [41] Trinh Minh Tri Do, Jan Blom, and Daniel Gatica-Perez. Smartphone usage in the wild: a large-scale analysis of applications and context. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 353–360, 2011.
- [42] Denzil Ferreira, Jorge Goncalves, Vassilis Kostakos, Louise Barkhuus, and Anind K Dey. Contextual experience sampling of mobile application micro-usage. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, pages 91–100, 2014.
- [43] Chunhui Zhang, Xiang Ding, Guanling Chen, Ke Huang, Xiaoxiao Ma, and Bo Yan. Nihao: A predictive smartphone application launcher. In *International Conference on Mobile Computing, Applications, and Services*, pages 294–313. Springer, 2012.
- [44] Ke Huang, Chunhui Zhang, Xiaoxiao Ma, and Guanling Chen. Predicting mobile application usage using contextual information. In *proceedings of the 2012 ACM conference on ubiquitous computing*, pages 1059–1065, 2012.
- [45] Philip Leroux, Klaas Roobroeck, Bart Dhoedt, Piet Demeester, and Filip De Turck. Mobile application usage prediction through context-based learning. *Journal of Ambient Intelligence and Smart Environments*, 5(2):213–235, 2013.
- [46] Zhung-Xun Liao, Yi-Chin Pan, Wen-Chih Peng, and Po-Ruey Lei. On mining mobile apps usage behavior for predicting apps usage in smartphones. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 609–618, 2013.
- [47] Sha Zhao, Julian Ramos, Jianrong Tao, Ziwen Jiang, Shijian Li, Zhaohui Wu, Gang Pan, and Anind K Dey. Discovering different kinds of smartphone users through their application usage behaviors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 498–509, 2016.
- [48] NL Griffin and FD Lewis. *A rule-based inference engine which is optimal and VLSI implementable*. Citeseer, 1989.
- [49] Jiaxin Huang, Fengli Xu, Yujun Lin, and Yong Li. On the understanding of interdependency of mobile app usage. In *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pages 471–475. IEEE, 2017.
- [50] Xuanzhe Liu, Huoran Li, Xuan Lu, Tao Xie, Qiaozhu Mei, Feng Feng, and Hong Mei. Understanding diverse usage patterns from large-scale appstore-service profiles. *IEEE Transactions on Software Engineering*, 44(4):384–411, 2017.
- [51] Huoran Li and Xuan Lu. Mining device-specific apps usage patterns from large-scale android users. *arXiv preprint arXiv:1707.09252*, 2017.