



Bachelor Degree Project

The State of Progressive Web Applications

*- an investigation of the experiences and
opinions of developers in the industry*



Author: Adam Elfström

Supervisor: Rüdiger Lincke

Semester: VT 2021

Subject: Computer Science

Abstract

Mobile applications can be developed using a variety of different techniques and technologies. One of the most recent of these techniques is the Progressive Web Application (PWA), a cross-platform solution that is built exclusively using common web technologies. The technique has great potential to become a major competitor to native applications but is currently held back by a few rather significant limitations.

This project was initiated because of a significant lack of academic research on the topic of PWA, and a perceived poor level of knowledge in the industry about the technique. The goal of the project was to determine if PWA deserved broader utilization or if the current low level of adoption was justified.

During the project, two surveys were conducted. The first survey asked mobile application developers from companies in different countries about things such as their knowledge of, experience with, and opinions of PWA. The second survey asked similar questions but was instead answered by lecturers in higher education in Sweden only.

The results of this project show that the average level of knowledge of PWA is very low and that developers' opinions of the technique are quite negative. The limitations of PWA were found to be few but crippling to its potential to achieve widespread adoption.

Keywords: Progressive Web Applications, Cross-Platform Applications, Native Applications, Mobile Application Development, Mobile Application Education

Contents

1	Introduction	1
1.1	Background	1
1.2	Related Work	2
1.3	Problem Formulation	2
1.4	Motivation	3
1.5	Result Types	3
1.6	Scope	4
1.7	Target Group	4
1.8	Outline	4
2	Method	5
2.1	Research Project Overview	5
2.2	Research Methods	5
2.2.1	Literature Study	6
2.2.2	Surveys	6
2.3	Reliability and Validity	6
2.3.1	Literature Study	6
2.3.2	Surveys	7
2.4	Ethical Considerations	7
3	Theoretical Background	8
3.1	Responsive Web Design	8
3.2	Native Mobile Applications	8
3.3	Mobile Web Applications	9
3.4	Hybrid Applications	10
3.5	Progressive Web Applications	10
3.5.1	Web Workers & Service Workers	11
3.5.2	HTTPS	12
3.5.3	Web App Manifest	12
3.5.4	PWA Architecture	12
3.6	Mobile App Technologies - Feature Comparison	13
4	Research Project	14
4.1	Literature Study	14
4.2	Survey Design	15
4.2.1	Survey Tool: Google Forms	15
4.2.2	Developer Survey Questions	15
4.2.3	Education Survey Questions	18
4.3	Survey Questions Relation to Research Questions	19
4.4	Survey Participant Selection	20
4.4.1	Participating Countries - Developer Survey	20
4.4.2	The Clutch.co Directory	21
4.4.3	Selection Process - Developer Survey	21
4.4.4	Selection Process - Education Survey	22

5	Results	23
5.1	Literature Study	23
5.2	Developer Survey	23
5.2.1	Part 1 - Personal Details	24
5.2.2	Part 2 - Experience	25
5.2.3	Part 3 - PWA	26
5.2.4	Part 4 - Your Company & Clients	31
5.2.5	Part 5 - Mobile Applications	32
5.3	Education Survey	33
5.3.1	Part 1 - Personal Details	33
5.3.2	Part 2 - Mobile Applications	35
5.3.3	Part 3 - PWA	38
6	Analysis	42
6.1	Literature Study	42
6.2	Developer Survey	42
6.2.1	Part 1 - Personal Details	42
6.2.2	Part 2 - Experience	43
6.2.3	Part 3 - PWA	43
6.2.4	Part 4 - Your Company & Clients	45
6.2.5	Part 5 - Mobile Applications	46
6.3	Education Survey	46
6.3.1	Part 1 - Personal Details	46
6.3.2	Part 2 - Mobile Applications	47
6.3.3	Part 3 - PWA	47
6.4	Checklist of Criteria for Widespread Adoption	48
7	Discussion	49
7.1	Findings & Answers to Research Questions	49
7.2	Related Work	51
8	Conclusions	52
8.1	Summary of Conclusions	52
8.2	Future Work	52
	References	54
A	Appendix A - Developer Survey	A
B	Appendix B - Education Survey	E
C	Appendix C - DSQ 20 Diagrams	H
D	Appendix D - ESQ 9 Diagrams	K

1 Introduction

We live in a digital age where mobile phones have become a major part of our daily lives. Modern mobile phones, so-called smartphones, are predominantly running either the Android or iOS operating systems. Since both platforms have large user bases companies often need to provide their applications, or apps, on both platforms to not miss out on potential users. The process of developing an app for two distinctly different platforms while keeping the same functionality and appearance can be both costly and time-consuming. Instead of developing native apps for each platform, a more cost-effective solution is to develop a single app that can run on multiple platforms; a so-called cross-platform app. There are many different options for cross-platform development, but this research project will only focus on Progressive Web Applications (PWA). PWAs are built on the same technologies that power websites, but can in contrast to regular web apps be installed on devices, much like native apps. Compared to other cross-platform alternatives, PWA is somewhat unknown and infrequently used despite having high potential.

1.1 Background

In 2007, Steve Jobs of Apple Inc. revealed their latest product, a mobile phone they called the "iPhone". During the announcement, Jobs also presented their proposed solution to application development on this new platform. By using a web browser and what was at the time state-of-the-art web technologies, developers could create web-based applications that behaved just like native applications [1]. However, a year later they introduced an application marketplace, the App Store, which completely replaced the previously announced development technique [2]. It was not until 2015 that the concept of installable web apps resurfaced in an article published by Google Chrome developer Alex Russel, now named as Progressive Web Applications [3].

A PWA is a cross-platform web application that is in essence a mobile-friendly website that can be installed locally on a device. PWAs are not a new technology, but rather a development technique that adheres to the three major qualities of installability, capability, and reliability [4]. The application is installable and is able to run in its own stand-alone window instead of being constrained to a browser tab, and can therefore become indistinguishable from a native application. Once installed, it has capabilities much greater than that of a normal website, such as hardware access and proper push notifications. Since its inception, the capabilities of PWAs have greatly improved, and only a few features are still missing compared to its native counterparts [5]. Finally, the application is fast and reliable, regardless of network connection.

Important to note is that there exists a technology similar to PWA, so-called hybrid applications. These are also built on web technologies but are wrapped in a native "shell" that allows them to run on mobile devices, while PWA's exclusively use web technologies. Mobile application technologies are further discussed in Chapter 3.

PWA has been called "the future of mobile applications", something that will take over the mobile landscape [6], [7]. Even with a potentially promising future, published research on PWA is mainly about a small number of topics, such as its performance and size in comparison to native applications. It proves a challenge to find research related to topics such as the rate of adoption, how widespread the knowledge about the technique is, and general insights into the attitudes towards the technique in the industry. Causes for, and solutions to, issues related to these areas are as expected also mostly absent because of this.

1.2 Related Work

In 2016, Steczko investigated the differences that exist between native and cross-platform development. He interviewed several companies about their experiences and opinions on the two development alternatives. He concluded that while both options have their pros and cons, native was the preferred choice [8]. His project investigated cross-platform development in a general sense, but he never mentions PWA in the report, presumably since the technique was mostly unknown at the time.

In 2020, Olsson and Gustafsson investigated the differences between PWA and native development, extending the work done four years prior by Steczko. They also conducted interviews but instead focused on PWA in particular, rather than cross-platform development as a whole. They also extended the research by using a tool to determine the rate of adoption in Swedish companies, concluding that PWA appears to be drastically under-utilized [9].

Many other case studies directly compare a PWA with a native or web app counterpart. For the comparison with web apps, the studies mainly highlight that the initial speed at which the application loads was greatly reduced and that user retention was increased. When comparing to native counterparts, the main advantage was a drastic reduction in app size [6], [10]–[13].

Thacker and Dharani mention that due to the capabilities of PWA not differing meaningfully from native, the adoption of the technique is expected to rise. They also mention that PWAs are not yet widely known by end-users [13].

Majchrzak et al. discuss the possibility of PWA becoming the “Definite Approach to Cross-Platform Development” meaning that PWA would replace all other cross-platform technologies in the future. While they did not draw a definitive conclusion to the question, they discussed many different aspects of PWAs in great detail and encouraged further research on the topic to fill the existing knowledge gaps. One of the most significant hindrances to increased PWA adoption they identified was the poor support the tech has on iOS devices [14].

In a usability test by Sedkowska, native social media applications that also had a PWA alternative were compared and concluded that the participants slightly favored the PWA alternative but that no major usability differences could be found [15].

1.3 Problem Formulation

The research on the topic of PWA is sparse and not very varied. E.g., it proves somewhat challenging to find papers about the state-of-the-art. Furthermore, previous research indicates that PWA is infrequently used and not as well-known as expected. I.e., the technology seems to be rarely known and thus heavily under-utilized.

To understand why this is the case, this thesis aims to investigate the current state of PWA in the industry and whether it deserves broader utilization by gathering opinions and knowledge from mobile application developers. On a smaller scale, the opinions of lecturers in higher education will also be collected.

In order to do so, we answer the research questions in Table 1.1.

Research questions	
RQ 1	Are the most important characteristics of mobile applications realized by PWA?
RQ2	Do developers experience the same problems described in the literature?
RQ 3	What do developers think about PWA and its future?
RQ 4	What needs to change with PWA to achieve more widespread adoption?
RQ 5	Do customers influence the decision of native vs PWA?
RQ 6	What is the state of PWA coverage in web & app development in higher education?

Table 1.1: All research questions the thesis aims to answer

1.4 Motivation

Poor knowledge of PWA in the industry can mean a missed opportunity to build the best app for a company's purpose. The results of this thesis might lead companies to decide to offer PWA as a solution to customers and incentivize them to inform their customers about the advantages the tech can offer them.

In conjunction with the results from also surveying higher education, further knowledge of the underlying causes for the perceived low adoption of PWA in the industry can be gained. The results of this thesis might also incentivize lecturers in higher education to incorporate PWA into their courses, improving the variety of development techniques and technologies taught in higher education and avoiding failing to meet the needs of the industry.

1.5 Result Types

This thesis uses surveys as the method for obtaining data about the current state of PWAs which means that the primary result will be the data collected from said surveys. From an evaluation of this data, conclusions can be drawn about what the state of PWA as a development technique is from the perspective of the industry. The surveys contribute to answering all of the research questions.

We list a ranking of the most common issues with PWA according to the literature. While this list does not on its own answer any research questions, it creates knowledge that is useful particularly when creating the surveys.

RQ 1 and 4 are further answered by the creation of a checklist containing the most important changes needed for PWAs to achieve more widespread adoption. This checklist is based on a literature study and a survey question that ask developers to rank the importance of different mobile app characteristics and features. See Section 4.2 for more details about the survey questions and Appendix A and B for the full surveys. Since the checklist is created from the survey responses it is therefore validated by the collected real-world data.

1.6 Scope

In this thesis, the current state of PWA is investigated by interviewing companies and higher educational institutes about the technique. The investigation is limited by only surveying companies offering mobile application development services. The companies are further limited to be based in one of four different countries (Sweden, USA, Poland, and India). In the investigation of the education sector only higher educational institutes in Sweden are to be surveyed.

There are a number of things this project will not involve, such as investigating the numerical rate of adoption in any of the selected countries, describing the process of developing a PWA or converting a regular web app to a PWA, or creating an exposition of current PWA capabilities and advantages.

1.7 Target Group

The target group of this research project is mostly application developers and their employing companies since they get an opportunity to take part in perspectives from other companies in the industry. Another group that might have an interest in the thesis is lecturers in higher education as they might currently have an interest, or develop one based on the results, in adding something about PWA to their curricula.

1.8 Outline

The rest of this report will have the following structure:

- Chapter 2 includes discussions about the scientific approach, research methods, and ethical considerations of the project.
- Chapter 3 explains concepts of mobile applications and PWA, forming a theoretical background of the technology.
- Chapter 4 describes all activities performed before and during data collection.
- Chapter 5 presents the collected data.
- Chapter 6 contains the full analysis of the results.
- Chapter 7 discusses the findings of the project and whether the problem has been solved. The results are also compared to related work.
- Chapter 8 concludes the project gives suggestions for future work.

2 Method

This chapter contains a more detailed description of the scientific methods used for this thesis project. Section 2.1 introduces the project by outlining what methods will be used for which parts of the project and gives an overview of how the project will be carried out. Section 2.2 gives more details about the methods that were introduced in Section 2.1, as well as briefly mentioning what methods were chosen to not be used. Section 2.3 discusses the reliability and validity of the project, by describing what threats exist that might lead to a degradation of the quality of the results and subsequent conclusions drawn from these results. Finally, Section 2.4 discusses ethical considerations for the project, such as how to preserve the anonymity of survey participants.

2.1 Research Project Overview

This research project aims to answer questions about the current state of PWA, both in terms of developer's opinions and in terms of offerings in the education sector. A multi-method approach was chosen for this project, and since the creation of any artifacts apart from the survey are not needed, this approach appears sufficient.

To answer the research questions, the project is split into two major stages that each produce something for the report:

1. A literature study will answer some of the questions in the survey for the industry, and also produce some of the questions for the survey for universities.
2. Conducting the surveys will finally allow us to answer our research questions.

Getting an overview of PWA as a technique, and more specifically what problem areas exist, is the first step necessary to create the questions of the surveys. The chosen method for this step is a literature study on both scientific research and grey literature which aims to give the knowledge necessary to create the survey. Not all questions are based on the study however; some are purely based on the research questions or the information presented in Chapter 3.

The first step of conducting the survey is the process of selecting which companies to contact for participation. A company suitable for participation must offer the service of mobile application development, although it is not necessary that this is their primary service. The method of finding suitable companies and the correct people in said companies to contact is to use the search function built into the business-to-business (B2B) website Clutch [16]. Section 4.4 contains more information about this website and the exact steps taken when selecting the companies.

The chosen method for data collection is as previously mentioned surveys. When contacting companies a link to the survey will be provided in the email. If a company wishes to participate in an alternative way, this can be done by scheduling a phone call in which the questions of the survey will instead be asked during an interview. If an interview is conducted, the call will be recorded and transcribed, and later added to the other survey answers. The choice of including alternative means of participation is based on our wish to increase the likelihood of companies participating in the study.

2.2 Research Methods

The following sections each describe in more detail the methods chosen.

2.2.1 Literature Study

A literature study will initiate the project by providing the information necessary to formulate the questions for the surveys. This study is to be conducted primarily on so-called grey literature, meaning articles written and published by non-academic sources. While some academic literature will be included in the study, it will be a minority because of its relative scarcity.

This method was chosen since it was deemed to be the most effective way of acquiring the information needed. Alternative methods included interviewing experts in the field and conducting case studies about the pros and cons of using the technique. Both of these alternative methods were deemed to not be feasible for this project due to time constraints.

2.2.2 Surveys

The developer survey contains questions that are directly related to the development process of mobile apps, experience, and opinions. The creation process of the questions for the developer survey is described in Section 4.2.2.

Before the surveys can be conducted, a selection of companies to be participants in the surveys needs to be made. This selection process will include selecting companies from different countries to improve the diversity of participants. If a company is willing to participate, they are expected to forward the survey to one of the relevant persons in the company to answer in their free time. If a company wishes to schedule a phone interview instead, that is also possible.

The survey to be conducted in the education sector will be limited to only include lecturers from higher education in Sweden. Ten institutes will be selected for participation. The selection will primarily be from Swedish universities, but since not all universities have courses about mobile or web development there might be a need to involve polytechnics as well.

Surveys were chosen as the primary method for obtaining data since they allow for easier collection of larger amounts of data. Purely doing interviews over the phone, or some other voice-communication medium, require more time and might therefore result in fewer participants. The survey was also deemed more fitting for having a mix of quantitative and qualitative questions, which for this project is desirable.

2.3 Reliability and Validity

This section contains discussions of the reliability and validity of the project, with a separate section for the literature study and the interviews.

2.3.1 Literature Study

Since the literature study was conducted as a precursor to the construction of the survey material, from which the results could then be derived, it is somewhat important that the material that was studied was of adequate quality. Since the scientific literature on this subject is limited, grey literature needs to be used which can be argued greatly increases the risk of the results having potential problems with validity. Something that decreases this risk is that many sources of grey literature mention the same aspects of PWAs. A consensus on what the most important aspects of the technique appear to exist, so the use of grey literature becomes easier to justify. The study does not exclusively examine grey literature however, a few scientific sources are also included.

2.3.2 Surveys

The results derived from the surveys will most likely suffer from both external validity and have issues with reliability. The process of selecting the companies to contact and the lecturers to contact is completely manual, meaning that the person doing the selections has an impact on which people are contacted. Additional external validity issues arise because the directory of app development companies used for finding companies to contact is owned by a business. This business might for instance change their terms of service a few months from now which could result in a substantial difference in which companies are included in the directory.

To mitigate the problem of reliability, all steps taken in the process of selecting participants are documented in Section 4.4. The issue of external validity will however still be present and would only be completely solved if the directory was fully static and the selection process was fully automated.

As for the survey aimed at the education sector, the selection process is once again not automated. All of the institutes listed as being Universities, or similar, in Sweden are considered for participation. Since this list is unlikely to drastically change soon, the selection process becomes more likely to replicate. The selection is limited to higher educational institutes in Sweden, and only lecturers in courses related to mobile app and/or web development and design are contacted.

2.4 Ethical Considerations

All persons participating in the survey will be made aware of who is behind the survey, why it is being conducted, and how their participation will be anonymous before they start. No participants are forced to take part in this study, their participation is completely voluntary. The name of all participants, be it individual persons or company names, will not be presented in this report to preserve their anonymity. If any interviews are conducted, the recordings of said interviews will be deleted at the end of the project.

Which parties choose to participate in the survey might result in a bias emerging. This is mitigated by only contacting a company asking for only one of their developers to participate, and to contact companies from different countries.

3 Theoretical Background

This chapter describes in more detail the concepts of responsive web design, mobile applications, and different cross-platform development technologies and techniques. Understanding these concepts and how they related to each other is helpful for understanding the design of the surveys and the results they produced.

3.1 Responsive Web Design

From a visual perspective, there exist two distinct types of websites; fixed-width websites and responsive websites. Before smartphones became popular, websites were typically designed around a minimum target resolution by using fixed sizes for all its elements, often expressed in a number of pixels [17]. This type of website is not capable of adapting its content to fit screens of different sizes; it stays static regardless of the device that is viewing the site. This causes problems with devices having particularly small screens or large screens since elements containing text might become illegible.

A responsively designed website uses relative sizes for elements, such as percentage widths instead of fixed pixel widths. Responsive websites are therefore fluid and flexible, allowing scaling to both ends of the screen-size spectrum. Additionally, so-called “media queries” are used to further optimize the site and remove issues visible above or below certain screen sizes [17].

Figure 3.1 illustrates how a website can use media queries to optimize its appearance for different devices and screen sizes.

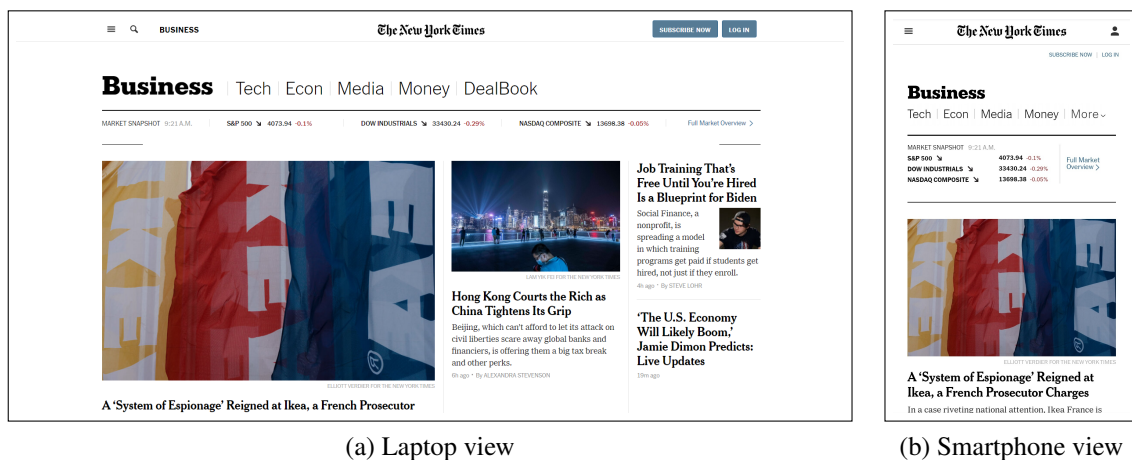


Figure 3.1: New York Times website responsiveness

3.2 Native Mobile Applications

A native application, or app, is a program developed to run on a specific type of mobile platform, developed using programming languages, application programming interfaces (APIs), and software development kits (SDKs) specific to that platform [18], [19]. A native app has full access to the device’s hardware and APIs, enabling the use of features such as the camera, storage, Bluetooth, and near-field communication (NFC).

This approach to mobile development results in apps that can only be run natively on one specific platform. This means that if an app is to be run on another platform, it needs to be remade with the tools and languages specific to the new platform. For example, if an Android application developed using the Java language is to be compatible with

Apple devices using the iOS operating system, it needs to be recreated using the Swift programming language instead.

A native mobile application is distributed through a mobile storefront, also called an app store, which differs depending on the platform. The storefronts have policies and rules dictating what content is allowed on the store which developers must adhere to. Today, the most popular storefronts are Google’s “Play Store” for Android devices and Apple’s “App Store” for iOS devices [20].

If an app is installed through one of these storefronts, updates to that app will be published and installed through that same storefront. Each update needs to be developed separately for each OS and published on its corresponding storefront, a process that might take different amounts of time depending on the storefronts’ policies. Maintaining consistency between versions running on different OS’s can therefore be difficult [21].

These qualities result in developing and maintaining native apps for multiple platforms at once being both time-consuming and costly [19].

Table 3.2 summarises the development stacks for Android and iOS. Since the two platforms do not share programming languages, UI frameworks, or development tools, developing for both platforms is time consuming. An app developed to natively run on Android can not natively run on iOS, and vice versa.

	Android	iOS
Programming language	Java, Kotlin	Objective-c, Swift
UI frameworks	Jetpack Compose, Android UI	UIKit, SwiftUI
Development Tools	Android Studio	Xcode, AppCode

Table 3.2: Android an iOS development stack [22]

3.3 Mobile Web Applications

A mobile web application is a web app that is hosted on a remote server containing logic to serve content in a specific way to mobile devices [19], [23]. Contrary to a native app, a web app is not installed on a mobile device but is instead accessed through a Uniform Resource Locator (URL) over the Hypertext Transfer Protocol (HTTP), and runs exclusively in the web browser [18], [23].

A web application, and by extension also a mobile web app, runs on a web server that handles all of the data processing and application logic. This leaves only the rendering of the user interface (UI) to the browser. Additionally, this also means that updates to the application are not required to be downloaded and installed by the accessing device, since all data resides on the webserver [18], [23].

Since common web technologies like HTML5, CSS and JavaScript are used, the app is only developed once and can thereafter be accessed by any device using any sufficiently up-to-date browser, making it platform independent [18]. Adversely, this dependence on browsers means that the app has many limitations such as inability to access device hardware and software, inability to be published on a mobile storefront, high dependency on network connectivity, and potentially poor performance [18], [23].

In Figure 3.2, the architecture of a web app is seen. The browser fetches all necessary data from the web server that is hosting the application. Since a web app is built com-

pletely out of web technologies and runs exclusively in the browser, a user's device does not need to install anything else.

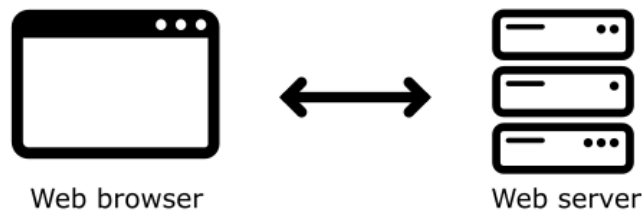


Figure 3.2: Web app architecture

3.4 Hybrid Applications

A hybrid app is essentially a mix of the native and web approaches to app development. The app is developed using the same tools as a web app, i.e., common web technologies, instead of native tools [19]. A hybrid app executes inside a native container, or shell, and uses the device's browser engine to render the common web elements inside a Web View [23]. This shell allows the hybrid app to function just like a native app would, allowing for the two approaches to be virtually indistinguishable for users. The shell also allows for the core of the app, such as the UI elements, to be reused across multiple platforms.

By making use of an abstraction layer, the web technologies that make up the app can access device hardware and software features through JavaScript APIs [23]. While a native app and a hybrid app have access to the same device features, this does not imply that the two are equivalent in performance. Since a hybrid app uses the browser engine, performance can be noticeably lower than its native counterpart [18], [23].

A hybrid app can be distributed through platform-specific storefronts and cannot be used directly through the browser like a web app, a download of the full application is required to use it [19], [23]. Just like a native app published on a storefront, updates are therefore also distributed through these same storefronts.

In Figure 3.3, the architecture of a hybrid app is provided. The core app is a mobile web app wrapped in a Web View. Through the use of an abstraction layer, native functionality can be achieved.

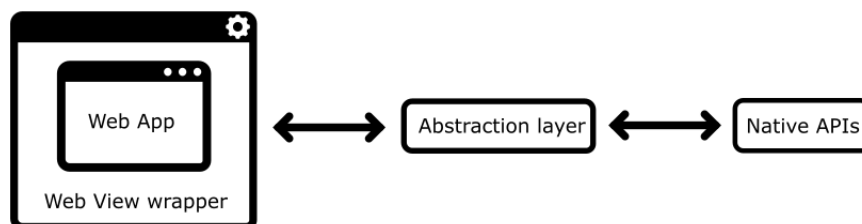


Figure 3.3: Hybrid app architecture

3.5 Progressive Web Applications

While progressive web apps are not a new technology per se, they are mobile web apps that follow certain best practices and conventions. Since a PWA is built with the same technologies as normal mobile web apps they share many similarities:

- a PWA can be accessed by any device with a sufficiently up-to-date web browser, making it platform-independent,
- only one web app needs to be developed and maintained,
- updates can be automatically applied without requiring additional downloads or installations,
- no forced dependence on mobile storefronts.

A PWA can be installed on a device and used like a native app, instead of being locked to a browser window like regular mobile web apps. When visiting a website, a prompt might pop up on the bottom of the screen containing the message “Add to home screen”. If the website is a PWA, clicking on the prompt will start an installation procedure which finishes with an icon appearing on the device’s home screen. This icon will then be able to launch the installed app, instead of simply being a shortcut to a website. Figure 3.4 depicts the visual differences between the home screen of the native Twitter Android app, Twitter Lite (PWA), and the mobile-friendly website. As expected, the mobile website and the PWA appear identical, while the native app has a slightly different appearance.

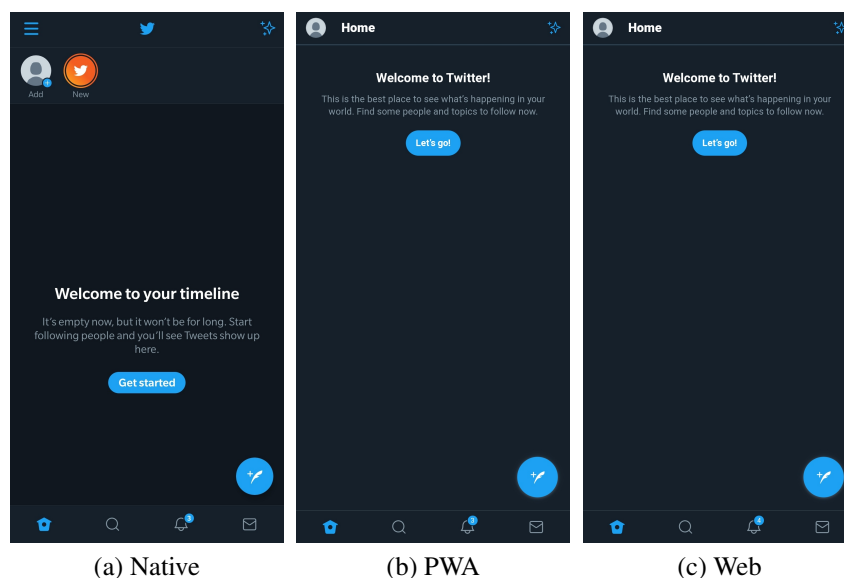


Figure 3.4: Native and PWA versions of Twitter’s mobile app

PWAs were originally only installable from the browser, but today the three most popular mobile storefronts allow for the publishing of PWAs, including the Windows Store [24]. This makes PWAs the most flexible in terms of usage options compared to native, hybrid, and mobile web apps.

To be considered a PWA, a web app must be served over HTTPS, have at least one service worker, and provide a web app manifest [25]. More on this in the following sections.

3.5.1 Web Workers & Service Workers

JavaScript is a single-threaded programming language, all instructions that are to be executed are placed on a stack and executed one by one [26]. Poor programming can therefore lead to greatly reduced performance in more complex applications.

A JavaScript web worker is an event-driven script that runs independently from the main document of a web page, effectively allowing for multiple threads of execution to be running concurrently. Web workers are intended to have a long lifespan and be used in lower numbers since they have both a high startup performance cost and a high memory cost while active. Web workers are therefore fit to handle complex tasks in the background to not slow down the main document thread that handles the normal UI functionality [27].

A service worker is a type of JavaScript web worker that can also act as a network proxy between the web page and the network. By sending the network requests to the worker the default network behavior can be overridden. One of the main benefits of this is the option to redirect network requests from the app to the cache instead of the server, effectively enabling offline access to the app. Additionally, the ability to deliver push notifications and sync data in the background while the app is not open is also enabled by the service worker [28].

3.5.2 HTTPS

On the internet, users with malicious intent can exploit essentially everything that gets sent over an unsecured connection, even data that might appear benign such as images or regular HTML. The Hypertext Transfer Protocol Secure, or just HTTPS for short, ensures that all requests to and from a website are encrypted and therefore secure [29].

While protecting the integrity of a website's users is important, HTTPS is also a requirement for many technologies that PWAs are using. Among others, service workers and many APIs, such as the Geolocation API, are not usable without a secure HTTPS connection [29].

3.5.3 Web App Manifest

A web app manifest is a JavaScript Object Notation (JSON) file including information about the app. The contents of the file are necessary for the app to function as a PWA since without it the web app would not be able to be downloaded and used as a native app.

The manifest includes information about the name, app icon, version, and background color, among other things. Data such as the app icon and name are especially important since without them the downloaded app would not properly be displayed like native apps on the device. Another important field is the URL which dictates what page the app should show when it is launched from the shortcut [30].

A manifest file can have either the ".webmanifest" or ".json" file extensions and is linked in the head of HTML files in the web app in the same way that a CSS file would also be included [30].

3.5.4 PWA Architecture

In Figure 3.5 the architecture of a PWA is displayed. When the application makes a request to fetch some form of content from the remote web server, the request instead goes to the service worker. The service worker first checks whether the content exists in the cache; if so it is returned to the app and no network activity is needed. Only when the content has not been cached (or the cache has expired) does the request go to the webserver.

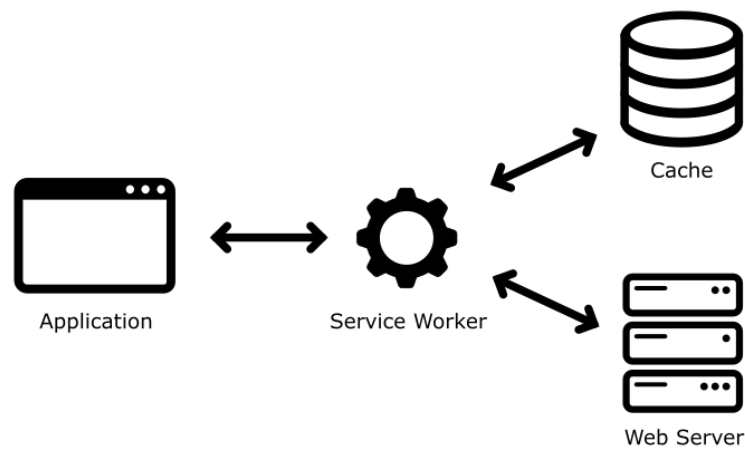


Figure 3.5: PWA architecture

3.6 Mobile App Technologies - Feature Comparison

Table 3.3 is a summary of what has been presented about the different mobile app and web development technologies and techniques in Sections 3.1 to 3.5. Sources [19], [31], [32] were primarily used when deciding which rows were to be included in the table. Note that the column “MWA” refers to Mobile Web Apps. The table is structured as a checklist to compare the features of each technique, with three possible checkmarks: “✓” indicates excellent support, “~” indicates partial support, and “✗” indicates poor or no support.

	Native	Hybrid	MWA	PWA
Installable	✓	✓	✗	✓
Offline access	~	~	✗	✓
Storefront distribution available	✓	✓	✗	✓
Mobile push notifications	✓	✓	✗	✓
Full platform API access	✓	✓	✗	~
Background sync	✓	✓	~	✓
Linkable, search indexable	✗	✗	✓	✓
No manual user updates	✗	✗	✓	✓
Fast and easy to distribute	~	~	✓	✓
Desktop availability	✗	✗	✓	✓
Great performance	✓	✗	✗	~
Cross-platform	✗	✓	✓	✓
Low cross-platform dev cost	✗	~	✓	✓

Table 3.3: Feature comparison of mobile app technologies

4 Research Project

This chapter contains detailed descriptions of all activities performed during the project both leading up to and during data collection.

4.1 Literature Study

When searching for grey literature on the subject, Google was the search engine of choice since it is currently the most widely used search engine [33]. The search terms used during the literature study are seen in Table 4.4. Before entering any search terms in the search engine, the precautions seen in the following list were taken:

- No Google account was signed in to avoid bias or optimizations.
- Only results from the past 2 years were shown to filter out outdated sources.
- Only results in English were shown.
- Location/region-specific search results were disabled.

Identifier	Search query
ST1	Progressive web application pros and cons
ST2	PWA pros and cons
ST3	PWA cons

Table 4.4: Literature study search terms

The final search term, ST3, proved to be the most effective at finding the desired information since ST1 and ST2 included a greater number of results that focused on the positive aspects and advantages. For this short literature study, the disadvantages of the technique were more relevant, since a greater quantity of the questions in the survey is to focus on drawbacks and desired changes from developers. Since many of the sources included only a small number of disadvantages, 15 sources were decided upon being a fitting sample size. If the number of disadvantages in each source was greater, a smaller number of sources would have been selected.

When searching for scientific sources, Google Scholar was used with the “Advanced search” feature. The search results were filtered to only be from 2019 - 2021, and to include at least one of the terms “PWA” or “Progressive web app”. Additional terms such as “features”, “comparison”, “vs native”, “disadvantages” and “evaluation” were also used to further narrow the searches. Even while using some of the additional terms, relevant sources were somewhat elusive. The three sources chosen for the study were sources [31], [32], [34].

Once the material was analysed, many issues were identified. The issues were then grouped together based on what the perceived problem area of each issue was. These groups were then summarized into a small number of different concrete problem areas, which can be seen in Section 5.1. The problem areas that were only mentioned in a single source were omitted.

4.2 Survey Design

This section details the creation of the survey questions. The questions are presented individually in Sections 4.2.2 and 4.2.3, but do not contain all of the options the participants have to choose from. The full questionnaires with all questions and options can be found in Appendix A and B.

4.2.1 Survey Tool: Google Forms

The surveys were created using the online survey tool Google Forms [35]. The choice of survey tool has a substantial impact on what type of questions can be asked, how the survey can be distributed, how the results are presented, and how the results can be exported. Google Forms was chosen due to its ease of use, aesthetics, and relatively strong feature set.

Google Forms allows for both open (qualitative) and closed (quantitative) questions, but having a mix between the two in one single question is not possible. Qualitative questions were mainly preferred in this project since they allow for the participants to in more detail explain their thoughts and opinions. Due to this preference, there was a need to extend some of the quantitative questions to include the option for the participant to further clarify their opinion or thought processes. In Google Forms, quantitative questions that give “Yes/No/Maybe” answers can not be extended to instead be “Yes: explain/No: explain/Maybe: explain”. A separate follow-up question asking for further clarification is needed to be added instead. This solution works but results in the questions being asked in a slightly different way. For example, there is now a need to add follow-up questions such as “Please explain your reasoning behind your answer to the previous question”. Not a drawback that makes certain questions unaskable, but a drawback that results in a greater number of questions being asked.

4.2.2 Developer Survey Questions

The decision to split the surveys into distinct sections, or parts, was made before the questions were written. This was done both for the benefit of the project and the participants since the questions themselves become easier to create and answer if they are grouped together based on specific topics. The parts that were decided upon for the developer survey were the following six:

1. Personal Details
2. Experience
3. PWA
4. Your Company & Clients
5. Mobile applications
6. Final Thoughts

The questions for the different parts of the survey are shown and discussed in the coming paragraphs and tables, except for Part 5. This part does not include questions like the other parts: it is only included to allow for participants to add any comments, suggestions, and thoughts they might have.

The questions for Part 1 and Part 2 of the survey are shown in Table 4.5 and Table 4.6 respectively. The questions in these parts do not contribute directly to the answers for the research questions but do instead provide details about the demographic. All of these questions are quantitative to give a clear picture of the demographic without the need for deep analysis of qualitative answers. Note that in the table, “DSQ 1” should be read as “Developer survey, question 1” or something similar.

Number	Question
DSQ 1	In which country do you work?
DSQ 2	Company name
DSQ 3	What is your age?

Table 4.5: Questions for dev Part 1: Personal Details

Number	Question
DSQ 4	How many years have you worked as a developer?
DSQ 5	What Best describes your current profession?
DSQ 6	What type of applications have you previously developed or worked on?
DSQ 7	Have you ever worked with any of the following cross-platform frameworks?

Table 4.6: Questions for dev Part 2: Experience

The questions in Part 3 of the survey are shown in Table 4.7 and all directly asking the developers about their opinion, experience, and knowledge of PWA. The questions primarily contribute to answering RQs 2, 3, and 4.

Question 8 is asked to gain more general knowledge about developers’ relation to PWA and to allow for more in-depth discussion, not to directly answer a research question. This question also acts as a guide for how an individual participant’s answers should be evaluated. A developer with great knowledge of PWA should be able to provide more relevant answers than that of a developer with little or no knowledge.

Questions 10 to 13 contribute to the answer to RQ 2 (if developers experience the same issues as the literature describes). Questions 10 and 11 are qualitative and allow the participant to describe their development preferences, which might include mentioning specific issues. Questions 12 and 13 are instead qualitative and directly ask about specific issues identified in the conducted literature study. Out of the issue categories seen in Table 5.14, CAT 7 was deemed not important enough to include in question 12.

Questions 9 to 11 and 14 to 16 all contribute to answering RQ 3 (developers’ opinions of PWA today and of its future). These questions are a mix of quantitative and qualitative to give a wider variety of information that allows for greater opportunities for discussion and analysis.

Number	Question
DSQ 8	How good is your knowledge of PWA?
DSQ 9	Would you consider developing a PWA in the future?
DSQ 10	Is there anything that makes you prefer other cross-platform solutions over PWA?
DSQ 11	Is there anything that makes you prefer developing natively over PWA?
DSQ 12	Have you ever encountered any of the following problems with PWA?
DSQ 13	When was the last time you encountered any of the problems above?
DSQ 14	What is your opinion on PWA as a development technique today?
DSQ 15	How much do you agree with this statement: "PWA is the future of mobile app development"
DSQ 16	If you disagree, how do you think PWAs need to change before the statement becomes true?

Table 4.7: Questions for dev Part 3: PWA

Part 4 of the survey asks questions about the developers' companies and their clients, which is seen in Table 4.8. Questions 17 and 18 are purely quantitative and contribute to RQ 4 (if clients influence the decision of PWA versus other solutions). Question 19 on the other hand does not answer any research questions but is yet another additional point of discussion.

Number	Question
DSQ 17	Has a client ever asked for a PWA?
DSQ 18	Have you ever proposed PWA as a solution to a client?
DSQ 19	What type of apps does your company offer?

Table 4.8: Questions for dev Part 4: Your company & clients

Finally, Part 5 of the survey asks the participant to rank the importance of several characteristics and features of mobile apps and can be seen in Table 4.9. This question is quantitative and answers RQ 1 (if PWA fulfills the most important characteristics of mobile apps).

Number	Question
DSQ 20	Please rank the importance of the following qualities of mobile apps

Table 4.9: Questions for dev Part 5: Mobile apps

4.2.3 Education Survey Questions

The survey was divided into 5 different parts, much like the developer survey. The parts that were decided upon for the education survey were the following four:

- Personal details & experience
- Mobile applications
- PWA
- Final Thoughts

Just like for the developer survey, the final part is not for asking questions but is instead for feedback. Note that the education survey is completely separate from the developer survey, which is why some questions appear duplicated.

The questions for Part 1 of the survey are shown in Table 4.10. Just like for the developer survey, the first part is only to give a better understanding of the participants. Note that in the table, “ESQ 1” should be read as “Education survey, question 1” or something similar. Additional note about ESQ 1 is that it is redundant since all participants in the education survey are from Sweden. This question is here because the survey constructed before the decision to limit the survey to only Swedish education was made.

Number	Question
ESQ 1	In which country do you work?
ESQ 2	Where do you work?
ESQ 3	What is your age?
ESQ 4	In which of the following subjects do you teach?
ESQ 5	How long have you been teaching in these subjects?

Table 4.10: Questions for education survey Part 1: Personal details & experience

The questions for Part 2 of the survey are seen in Table 4.11 and are about mobile applications. Questions 6 to 8 are purely quantitative and all contribute to answering RQ 6 (if PWA is covered in higher education courses about app development).

Question 9 is the same as the last question in the developer survey; it asks the participant to rank the importance of mobile app characteristics. This question can contribute to answering RQ 1 and its results might be extra interesting if it differs significantly from the rankings that developers gave.

Number	Question
ESQ 6	What type of applications do your courses cover?
ESQ 7	If your courses do not cover cross-platform apps, do you think it should be covered in the future?
ESQ 8	If your courses do cover cross-platform apps, does it cover any of the following frameworks?
ESQ 9	Please rank the importance of the following features and qualities of mobile apps

Table 4.11: Questions for education survey Part 2: Mobile applications

Part 3 of the survey is about PWAs and all its questions can be seen in Table 4.12. Since only one of the research questions are specifically about the education sector, all of the questions in this part aims to answer RQ 6.

Number	Question
ESQ 10	How good is your knowledge of PWA?
ESQ 11	Is PWA mentioned in the required literature in any of your courses?
ESQ 12	Is PWA part of the curriculum in any of your courses?
ESQ 13	If not, please give a short explanation
ESQ 14	Has it ever been requested that PWA be included in the curriculum?
ESQ 15	If yes, from whom?
ESQ 16	Do you think PWA should be included in the curriculum?

Table 4.12: Questions for education survey Part 3: PWA

4.3 Survey Questions Relation to Research Questions

Table 4.13 depicts how the survey questions contribute to the answer of the research questions. Questions 1-8 and 19 in the developer survey as well as questions 1-5 in the education survey do not appear in the table. The omitted questions do not directly contribute to the answer of any of the research questions but do instead help to understand the demographic that was covered by the survey. These questions also open up for further possible analysis and conclusion drawing at the end of the project.

	RQ 1	RQ 2	RQ 3	RQ 4	RQ 5	RQ 6
DSQ 9			X			
DSQ 10		X	X			
DSQ 11		X	X	X		
DSQ 12-13		X				
DSQ 14			X			
DSQ 15			X			
DSQ 16			X	X		
DSQ 17					X	
DSQ 18					X	
DSQ 20	X					
ESQ 6-8						X
ESQ 9	X					
ESQ 10-16						X

Table 4.13: Feature comparison of mobile app technologies

4.4 Survey Participant Selection

This Section details the process of selecting participants.

4.4.1 Participating Countries - Developer Survey

From the following four countries, the participants for the survey were selected:

- Sweden
- Poland
- USA
- India

These countries are located on three different continents; Europe, North America, and South Asia. They were chosen to represent as broad of a demographic as possible to avoid an overwhelming bias in favor of one particular country or community. Since both of the surveys are completely in English, an additional consideration when selecting countries was that the general knowledge of the English language of the population needed to be high.

Poland was not originally a part of the investigation. During the selection process it was discovered that several companies have offices in both Sweden and Poland. The apparent prevalence of Polish developers resulted in a slight expansion of the scope of the investigation to further include this demographic.

4.4.2 The Clutch.co Directory

For finding companies to contact, the application developer directory offered by Clutch [16] was chosen as the source. This directory offers the option to filter companies based on which countries they have offices in via the “Location” input. Other useful information can also be found such as what “Service focus” a company has and in which country their headquarters are located. Service focus refers to services such as “Social media marketing” or “Web development” and is expressed as percentages of their overall business.

This directory offers an easier way to search than if a traditional search engine such as Google or Bing was used since irrelevant results and advertisements are not shown. There are two major drawbacks with Clutch that can affect the results of the investigation. Firstly, the companies shown first by default are labeled as “Sponsors” of the platform. This problem can be rectified by changing the sorting option to for instance be “Number of reviews” or “Company name”. Secondly, showing up in the Clutch directory requires a manual application process. The number of companies that have been through this process varies greatly between countries which might mean that many potential candidates for participation are not considered at all. If the investigation had a broader scope, this issue might have been deemed big enough to include companies from other sources as well.

4.4.3 Selection Process - Developer Survey

The following requirements need to be fulfilled before a company could be selected to be a participant in the survey:

- The company must have specified that at least 15% of their service focus is “Mobile App Development”.
- The company must have an office in the country in question, preferably its headquarters.
- The company must have a functioning website. Sites that did not load or were automatically blocked as being “unsafe” by Google Chrome were not further pursued.
- The company must **not** have specified that their business is strictly in Native development.
- The company must have either a contact form or a specified contact email that is **not** exclusively for sales or business inquiries.

The directory is paginated with 40 companies per page. Sweden had only about 2 pages on compared to the around 100 pages each for the other three countries. Due to the much smaller number of companies, requirement 1 was deemed not necessary to fulfill for a Swedish company to be considered for participation.

If a company fulfilled all requirements the following details were saved in a document:

- Company name,
- Contact email address or contact form link,
- Website link,
- Any additional notes to keep in mind when contacting them,

- Have they been contacted?

The goal during this process was to save the information about 35 companies from each country for a grand total of 140 different companies. About 20 of the companies were not contacted on the basis of the email address not being reachable or the contact form not being functional.

4.4.4 Selection Process - Education Survey

A list of higher educational institutes in Sweden was used to find people to contact for the education survey [36]. Only the schools listed under the headings "Universitet" and "Högskolor" were included in the search.

At the website of each school, the main information that was searched for was either a list of courses or a search function for courses. Filtering options to only show courses related to computer science or IT were used whenever possible. A course believed to be fit for further investigation had its syllabus read through. If the syllabus mentioned that the course was about web or mobile app development it was considered for the survey. Some courses that were only about information or interface design were not included.

If a contact person was explicitly given on the course page they were directly contacted about the survey. If no contact person was given an email asking for the contact information for the courses was sent to the school instead. Out of the total 10 courses found fit for the survey, only the contact person for one of them was not able to be reached.

5 Results

This chapter presents the results from the short literature study and the survey answers.

5.1 Literature Study

The literature study was split into two parts, with the first part examining 15 sources of grey literature and the second part examining three sources of scientific literature. The problem areas identified in all the literature were organized and grouped together based on what the underlying issue was. These groups, or more accurately categories, can be seen in the Table 5.14. Additionally, the table includes how many times these issues were mentioned in both the grey literature and the scientific literature, expressed both as a raw number and as a percentage of the total.

Category	Issue/problem area	Grey mentions	Scientific mentions	Total
CAT 1	Hardware access limitations	11	3	78% (14)
CAT 2	Additional limitations on iOS	10	2	67% (12)
CAT 3	Visibility/marketplace presence	9	1	56% (10)
CAT 4	Software limitations	7	1	44% (8)
CAT 5	Higher battery consumption	7	0	39% (7)
CAT 6	Legacy device/browser support	4	3	39% (7)
CAT 7	Cross login/Inter-app communication	2	0	11% (2)

Table 5.14: Feature comparison of mobile app technologies

Categories 1-3 were all mentioned in above 50% of the sources, categories 4-6 were mentioned in around 40% of sources and finally, Category 7 was mentioned in only 11% of sources.

A difference between most of the grey literature and the scientific literature is the type of content they typically included and how it was presented. Most grey sources used bullet lists for presenting the many advantages and disadvantages at once to allow for an easy overview. The scientific sources tended to include a smaller number of issues that were discussed in much greater detail.

5.2 Developer Survey

This section presents the results from the survey aimed at application developers. The survey reached out to 122 different companies that offer app or website development to their clients. The contacted companies were from Sweden, USA, India, and Poland, with about an even split between the four countries. From the 122 companies contacted, 16 different developers participated in the survey.

5.2.1 Part 1 - Personal Details

This part of the survey was not aimed at directly answering any of the research questions. The questions asked in this part were for the purpose of enabling more in-depth analysis and discussion and to help identify potential biases.

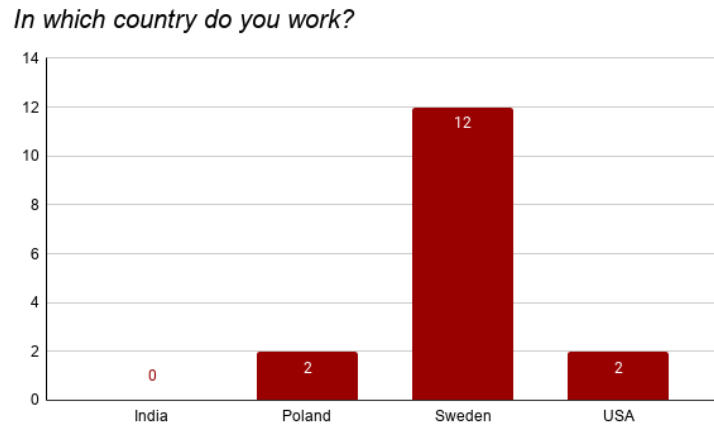


Figure 5.6: Answers for DSQ 1

The first question asked respondents about which country they work in, results seen in Figure 5.6. Most participants (12/16) were from Sweden, while Poland and the USA both had two participants. None of the Indian companies participated in the survey.

DSQ 2 is not shown in a diagram like all the other survey questions. DSQ 2 was an optional question that asked respondents to specify which company they work for. For the sake of the privacy of the respondents this information is omitted from the report. Most respondents (10/16) did not answer this question, but of the ones who did 4 were from the same company.

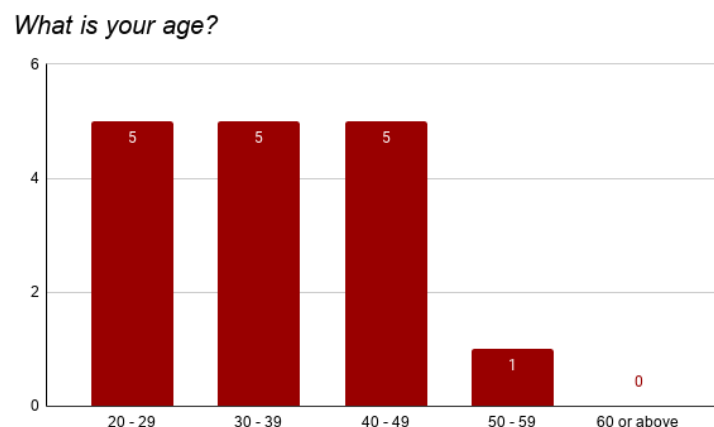


Figure 5.7: Answers for DSQ 3

DSQ 3 asked for which age group the participants belonged to, results seen in Figure 5.7. Only one respondent was between the age of 50 to 59 years old. five respondents each belonged to the age groups 20-29, 30-39, and 40-49.

5.2.2 Part 2 - Experience

Much like Part 1 of the survey, Part 2 also asked questions for the purpose of allowing for greater discussion. The questions in Part 2 of the survey helped paint the picture of how experienced respondents were with native and cross-platform development.



Figure 5.8: Answers for DSQ 4

DSQ 4 asked how many years of experience as a developer the respondents had, results seen in Figure 5.8. The main purpose of this question was to see if developers of different experience levels have different preferences and opinions of cross-platform development and PWA. Two respondents had between 5 and 9 years of experience while 4 respondents had more than 15 years of experience. The remaining 10 respondents were evenly split between 10 to 14 years of experience and less than five years of experience.

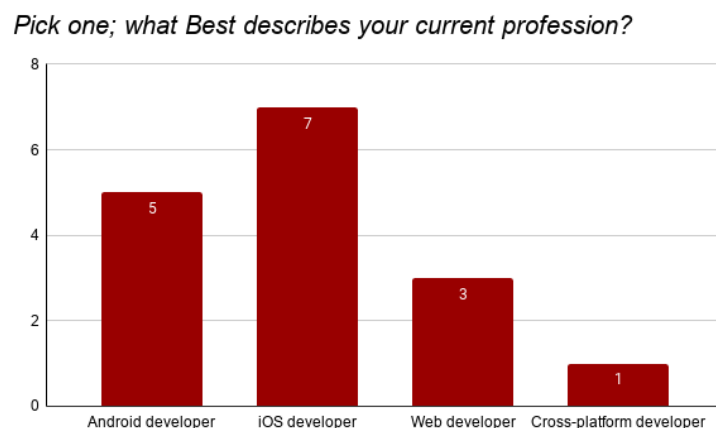


Figure 5.9: Answers for DSQ 5

DSQ 5 asked what best describes the current profession of the respondent, results seen in Figure 5.9. The purpose of this questions was to was to see how for instance Android developers' opinions of PWA differs from that of iOS developers. Five respondents were Android developers, seven were iOS developers, three were web developers, and only once was a cross-platform developer.

DSQ 6 asked respondents what type of applications they have previously worked on, results seen in Figure 5.10. Much like DSQ 4 and 5, this question was asked to see how developers experience might influence their answers to later questions. Most respondents

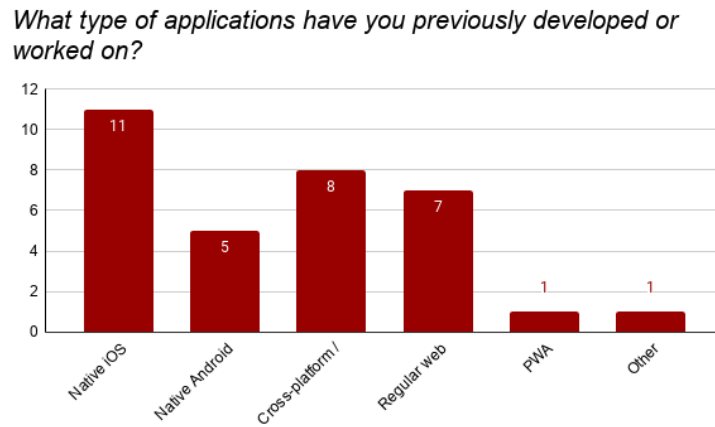


Figure 5.10: Answers for DSQ 6

(11/16) had worked on iOS apps before, while only some respondents (5/16) had worked on Android apps before. Eight respondents had worked on cross-platform apps and seven had worked on regular web apps. Only one respondent had worked on a PWA before. One respondent added that they had also worked on applications not included in the list of options which was software for embedded Linux systems.

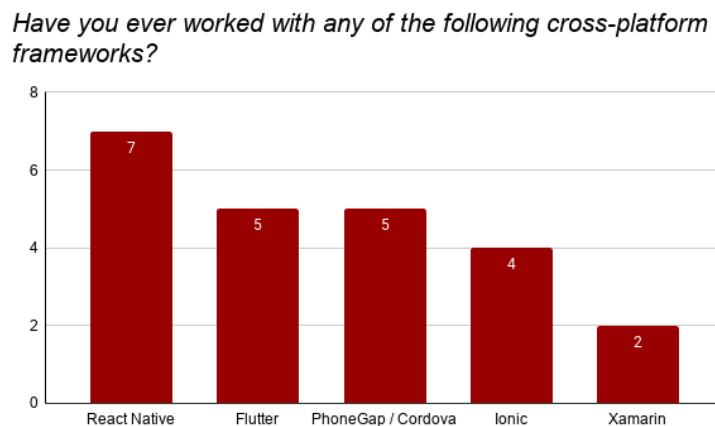


Figure 5.11: Answers for DSQ 7

DSQ 7 asked respondents what frameworks for cross-platform development they have used in the past, results seen in Figure 5.11. This question was asked to gauge the general popularity of different frameworks. A similar question was also asked in the education survey to see what frameworks are included in curricula, so education versus real-world popularity can be compared. React Native was the most popular of the frameworks at seven responses. Flutter, PhoneGap or Cordova, and Ionic were the second most popular at five, five, and 4 responses respectively. Xamarin was the least used framework at only two responses.

5.2.3 Part 3 - PWA

This part of the survey had the goal of answering RQs 2, 3, and 4. The questions were a mix of quantitative and qualitative and were quite mixed in what they were asking. All questions in this part were about PWA.

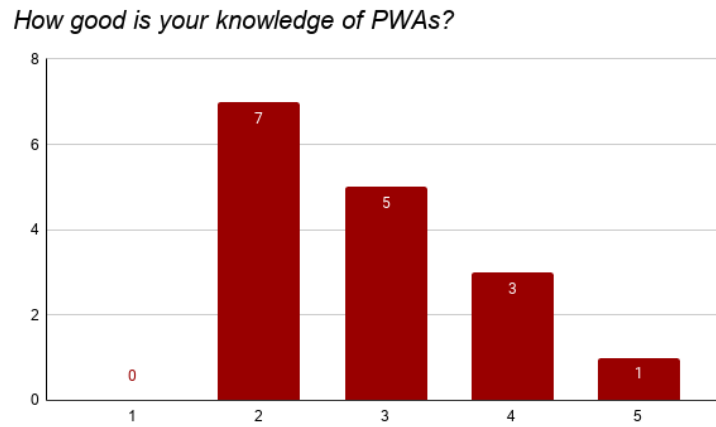


Figure 5.12: Answers for DSQ 8

DSQ 8 asked the respondents about their perceived level of knowledge of PWA, results seen in Figure 5.12. The answers were given on a scale of one to five, with one being "None" and five being "Very good". All respondents had at least some knowledge of PWA, since 0/16 responded that their knowledge was a 0/5. A majority of the respondents said their knowledge of PWA was either a 2/5 (7/16 respondents) or 3/5 (5/16 respondents), which would correspond to "low" and "ok" respectively. Three responded with a 4/5 while only one responded with a 5/5, scores that correspond to "good" and "very good" level of knowledge respectively.

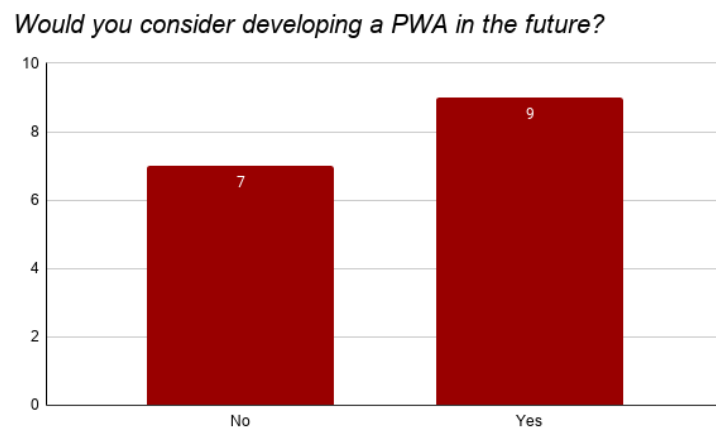


Figure 5.13: Answers for DSQ 9

DSQ 9 asked if the respondents would consider developing a PWA in the future, results seen in Figure 5.13. The question received a very split result: a small majority (9/16) responded with "Yes" while a small minority (7/16) responded with a "No". RQ 3 is about what developers think of PWA and its future. RSQ 9 does not directly ask their opinion on PWA, but the answers the respondents give can still be used to answer the research question.

Is there anything that makes you prefer other cross-platform solutions over PWA?

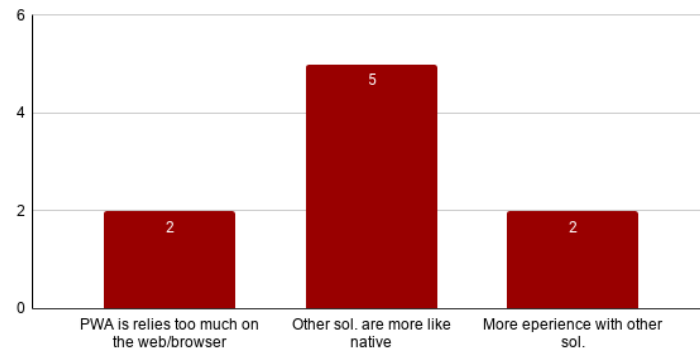


Figure 5.14: Summary of answers for DSQ 10

DSQ 10 asked if there is anything that makes the respondents prefer development with other cross-platform solutions, a summary of the results are seen in Figure 5.14. The answers were split according to what the issues brought up were related to. This question was optional and received responses from 8 respondents, with one response fitting into two categories at once. A majority of respondents (5/8) mentioned that a cross-platform solution that is "more like native" was preferable. Examples include mentioning that React Native uses "native components" and that other solutions "take less effort to integrate with the native platform". Two responses mentioned that PWA relied too much on the browser, with one respondent saying that this made PWA "unreliable". Two responses mentioned that their preference was based on them having more experience with other solutions. This question helps answer both RQ 2 and 3 since some responses mentioned opinions of PWA (RQ 3) and some mentioned problems with PWA (RQ 2).

Is there anything that makes you prefer developing natively over PWA?

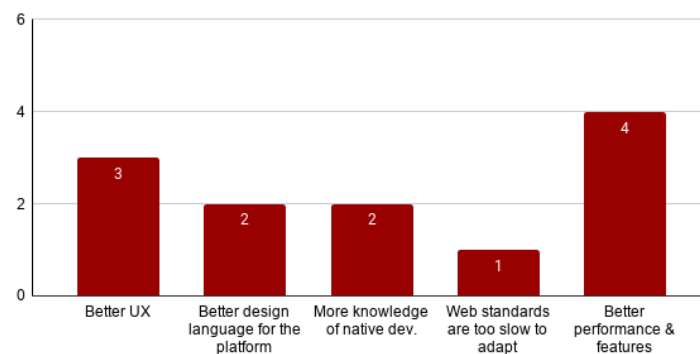


Figure 5.15: Answers for DSQ 11

DSQ 11 asked if there is anything that makes the respondents prefer development of native apps to that of PWA, a summary of the results are seen in Figure 5.15. The answers were split according to what the issues brought up were related to. This question was optional and received responses from 11 respondents, with several responses fitting into two or more categories at once. Four responses mentioned that native has either better performance or access to better features. Three responses mentioned that native offers a better user experience (UX) in general. Two responses mentioned that the design language

with native is better fit for the target platform. Two responses mentioned that they prefer native development since they have more experience with it. One response mentioned that web standards "can take ages to get adopted by browser vendors" while for native the access to new features is much quicker. This question helps answer RQs 2, 3, and 4 since the responses include current problems with PWA (RQ 2 & RQ 3) and what is holding it back from being more widely used (RQ 4).

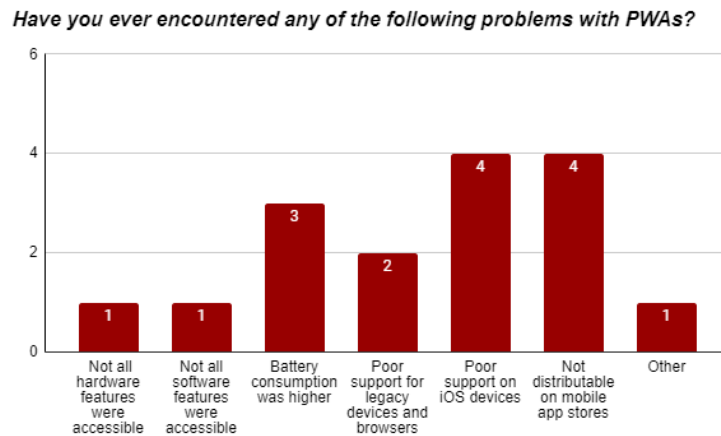


Figure 5.16: Answers for DSQ 12

DSQ 12 asked if the respondents had experienced any of a specific set of problems with PWA, answers seen in Figure 5.16. Respondents could select multiple options. "Poor support on iOS" and "Not distributable on app stores" were each selected four times. "Battery consumption was higher" was selected three times. "Poor support for legacy devices" was selected two times. Being selected one time each was that hardware and software access was limited. One respondent gave an answer in the "Other" option: a PWA "Doesn't feel at home on the phone". This question directly answers RQ 2.

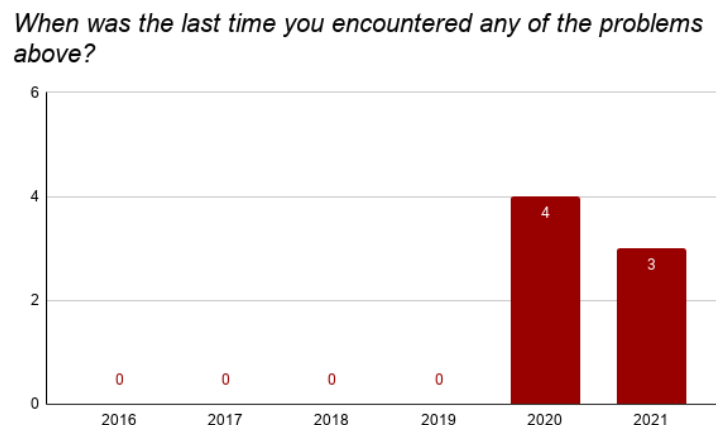


Figure 5.17: Answers for DSQ 13

DSQ 13 was a follow-up to DSQ 12 and asked when the last time any of the mentioned problems was encountered, results seen in Figure 5.17. None of the respondents answered 2016, 2017, 2018, or 2019. Four of the respondents answered 2020 and three answered 2021.

What is your opinion on PWA as a development technique today?

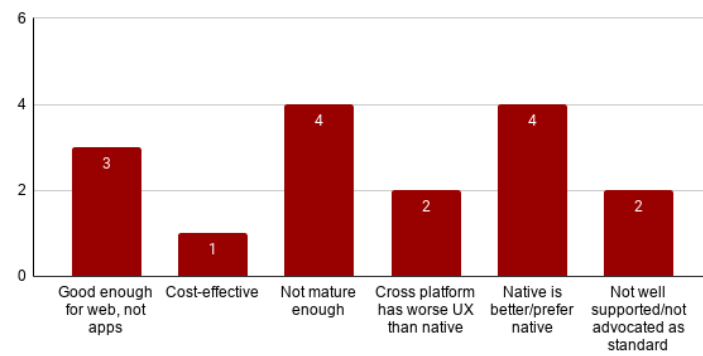


Figure 5.18: Summary of answers for DSQ 14

DSQ 14 asked respondents of their opinion on PWA today, a summary of the results are seen in Figure 5.18. Four respondents mentioned that they liked native better or that native development is better. Four respondents mentioned that the main problem with PWA was the it is not mature enough yet. Three respondents said that PWA is good enough for the web, but native development is better for apps.

How much do you agree with this statement: "PWA is the future of mobile app development"

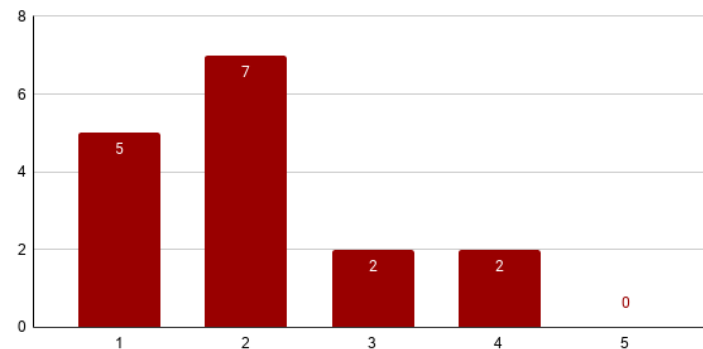


Figure 5.19: Answers for DSQ 15

DSQ 15 asked respondents to to rate their agreement with the statement that "PWA is the future of mobile development", results seen in Figure 5.19. The answers were given on a scale of one to five, with one being "Strongly disagree" and five being "Strongly agree". A vast majority answered either 1 (5/16) or (7/16), meaning that disagreed with the statement. The options 3 and 4 were each chosen two times while the option 5 was not chosen by any respondents.

If you disagree, how do you think PWAs need to change before the statement becomes true?

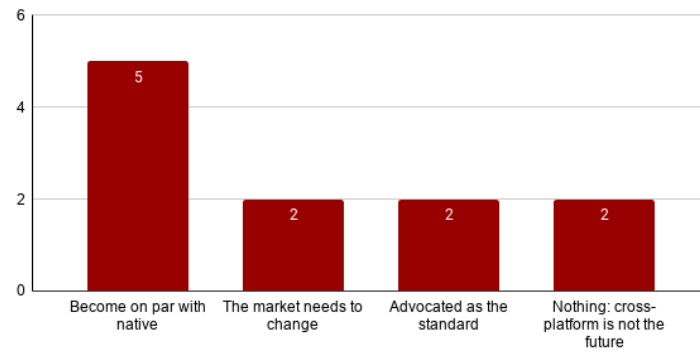


Figure 5.20: Summary of answers for DSQ 16

DSQ 16 was a follow-up question to DSQ 15, asking respondents that disagreed with the statement in DSQ 15 to try and give an answer as to what is needed to change for the statement to become true, a summary of the results are seen in Figure 5.20. Five respondents said that PWA needs to become at least on par with native in terms of for instance device API access. Two respondents pondered that maybe it is the market that needs to change to fit PWA instead of the other way around, while two respondents stated that since PWA is cross-platform, it cannot be the future. Two respondents talked about how PWA needs to be the standard on a platform, or at least advocated by Apple or Google. DSQ 15 and DSQ 16 together help answer RQ 3, while DSQ 16 additionally help answer RQ 4.

5.2.4 Part 4 - Your Company & Clients

This part of the survey only asked a small number of question about the developers' employing companies and their clients. These questions together help answer RQ 5, if clients influence the decision of using PWA or not.

Has a client ever asked for a PWA?

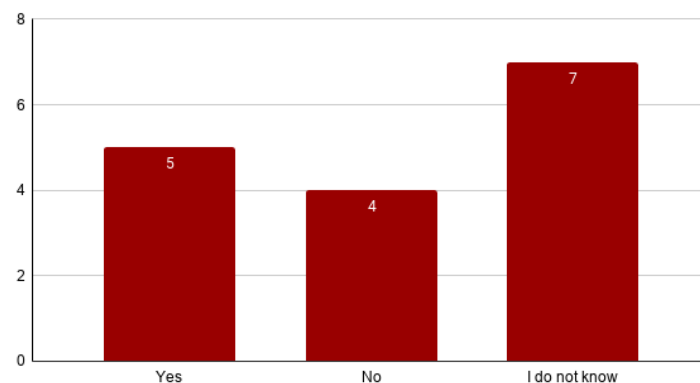


Figure 5.21: Answers for DSQ 17

DSQ 17 asked if a client ever requested a PWA, results seen in Figure 5.21. Seven respondents answered that they do not know if a client has request PWA. Five respondents answered "Yes" and four answered "No", a very evenly split result.

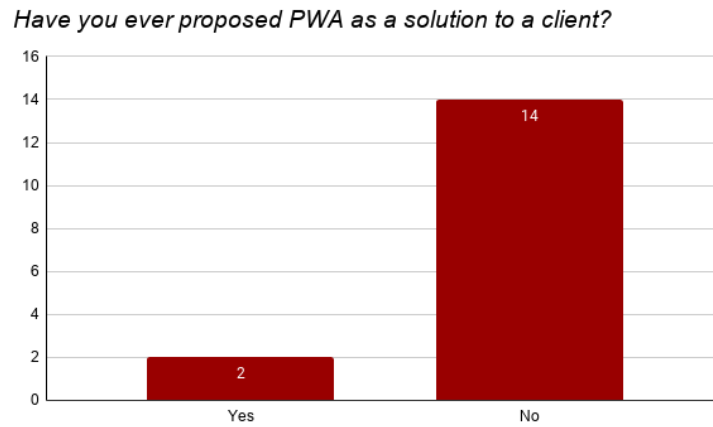


Figure 5.22: Answers for DSQ 18

DSQ 18 asked if the respondent had ever proposed PWA to a client, results seen in Figure 5.22. An overwhelming majority of respondents (14/16) answered that they had not proposed PWA as a solution to a client while only two respondents that they had proposed it.

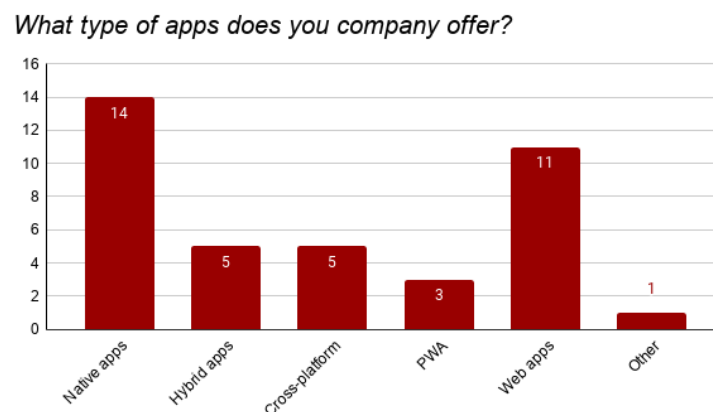


Figure 5.23: Answers for DSQ 19

DSQ 19 asked what types of applications the respondents' employing companies offered to clients, results seen in Figure 5.23. This question was mainly asked to see if the respondents' experience correlates with what the company offers.

5.2.5 Part 5 - Mobile Applications

This part of the survey contained only one question about how important certain aspects of mobile apps are according to developers. The question contains 9 sub-questions, one for each of the aspects/features that was to be ranked. This result is only presented in Table 5.15 and will not be further described in this chapter. Individual diagrams for each of the sub-questions can be seen in Appendix C.

One respondent added as a comment that two additional things that could have been asked in this question were "Native controls" and "Be a priority for the platform vendor to support". There was only one respondent that mentioned these two issues.

	Not at all important	Slightly important	Important	Fairly important	Very important	No opinion
Available on app stores	0	0	1	7	8	0
Installable (app store or web download)	1	0	1	6	8	0
Push notifications	0	1	0	2	13	0
Background sync	0	2	1	4	9	0
High performance	0	0	0	7	9	0
Low battery usage	0	0	2	5	9	0
Usable while offline	0	5	3	3	5	0
Cross-platform	5	3	2	5	1	0
Discoverable from web-search	4	4	6	0	2	0

Table 5.15: Answers for DSQ 20

5.3 Education Survey

The education survey reached out to nine different contact persons for courses in web or app development. The courses were taught in different institutes of higher education in Sweden. Out of the total of nine contacted persons, two participated in the survey.

5.3.1 Part 1 - Personal Details

This part of the survey was not aimed at directly answering any of the research questions, just like the first part of the Developer survey. The questions asked in this part were for the purpose of enabling more in-depth analysis and discussion and to help identify potential biases. Although since the number of responses was so low, this is not really possible in this case.

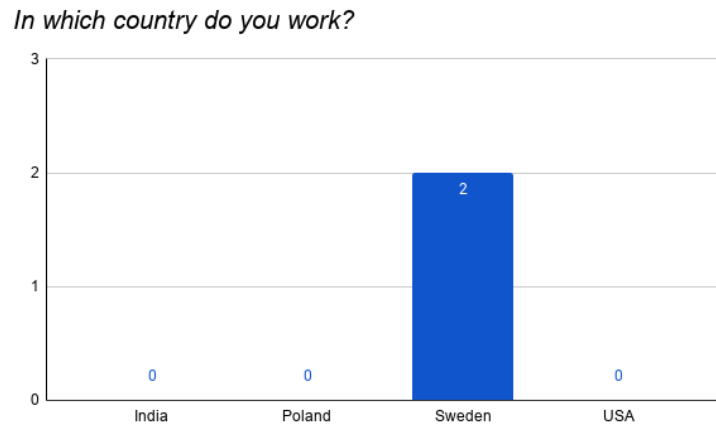


Figure 5.24: Answers for ESQ 1

ESQ 1 asked the respondents to answer which country they work in, results seen in Figure 5.24. Both respondents were from Sweden.

ESQ 2 was optional and asked respondents which institute of higher education they worked in. The results of this question is not shown to protect the privacy of the respondents.

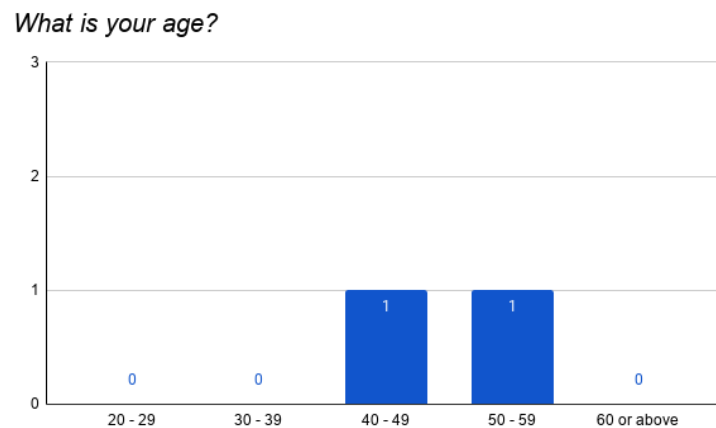


Figure 5.25: Answers for ESQ 3

ESQ 3 asked respondents which age group they belonged to, results seen in Figure 5.25. One respondent was between 40-49 years old while the other was between 50-59 years old.

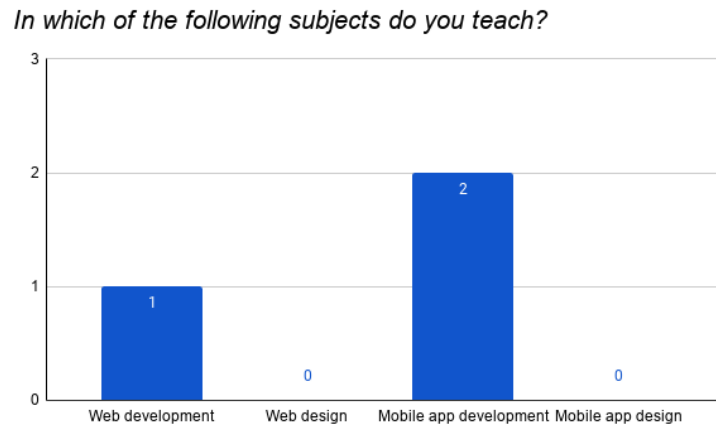


Figure 5.26: Answers for ESQ 4

ESQ 4 asked respondents what type of courses they teach, results seen in Figure 5.26. Both respondents teach in "Mobile app development" courses, and one respondent also teach in "Web development" courses.

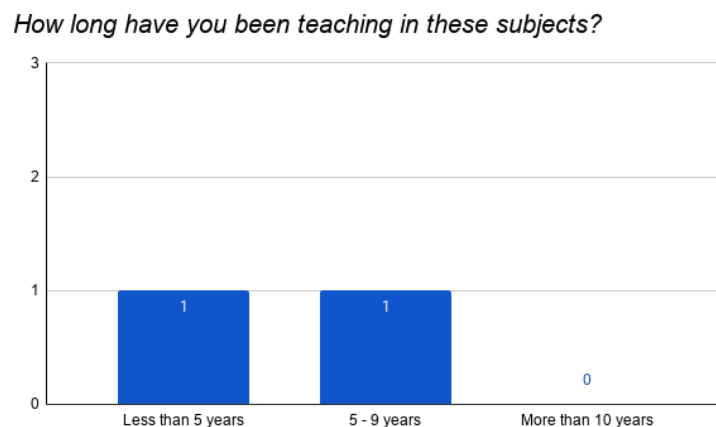


Figure 5.27: Answers for ESQ 5

ESQ 5 asked respondents for how long they had been teaching the subjects specific in ESQ 4, results seen in Figure 5.27. One respondent answered that they had been teaching the subjects for less than 5 years while the other answered 5-9 years.

5.3.2 Part 2 - Mobile Applications

This part of the survey asked questions about PWA, cross-platform in general, and the importance of mobile app qualities/features. These questions were asked so the answers could be directly compared to the answers given by developers.

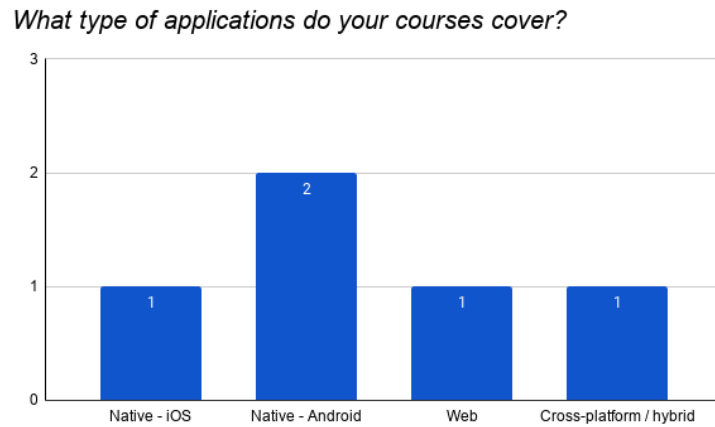


Figure 5.28: Answers for ESQ 6

ESQ 6 asked respondents what type of applications their courses covered, results seen in Figure 5.28. Both respondents had courses in native android development. Native iOS, Web, and cross-platform/hybrid apps were covered in courses by one respondent. This question was asked to help answer research question 6 which is about the state of PWA coverage in higher education.

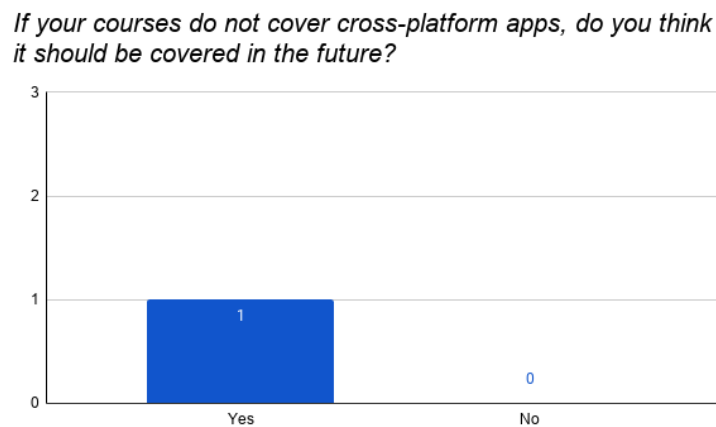


Figure 5.29: Answers for ESQ 7

ESQ 7 was a follow-up question to ESQ 6 and it asked respondents whose courses do not cover cross-platform if they think cross-platform development should be covered, results seen in Figure 5.29. The respondent answered that Yes, they do think cross-platform should be covered. This question was asked in conjunction with ESQ 6 to help answer RQ 6.

If your courses do cover cross-platform apps, does it cover any of the following frameworks?

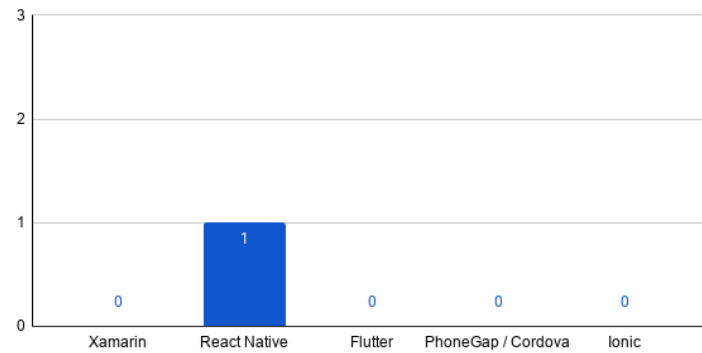


Figure 5.30: Answers for ESQ 8

ESQ 8 was a follow-up question to ESQ 6 and it asked respondents whose courses do cover cross-platform what type of frameworks they cover, results seen in Figure 5.30. The respondent answered that the framework "React Native" was covered in their courses. This question was asked in conjunction with ESQ 6 to help answer RQ 6.

	Not at all important	Slightly important	Important	Fairly important	Very important	No opinion
Available on app stores	0	1	1	0	0	0
Installable (app store or web download)	0	0	0	0	2	0
Push notifications	0	0	0	2	0	0
Background sync	0	2	0	0	0	0
High performance	0	0	0	1	1	0
Low battery usage	0	1	1	0	0	0
Usable while offline	0	1	0	0	1	0
Cross-platform	0	0	0	2	0	0
Discoverable from web-search	0	1	0	0	1	0

Table 5.16: Answers for ESQ 9

ESQ 9 is a duplicate of DSQ 20, meaning that the question contains 9 sub-questions, one for each of the aspects/features of mobile apps that was to be ranked. Just like for DSQ 20, this result is only presented in Table 5.16 and will not be further described in this chapter. Individual diagrams for each of the sub-questions can be seen in Appendix D.

5.3.3 Part 3 - PWA

This part of the survey had the goal of answering RQ 6. All questions in this part were about PWA.

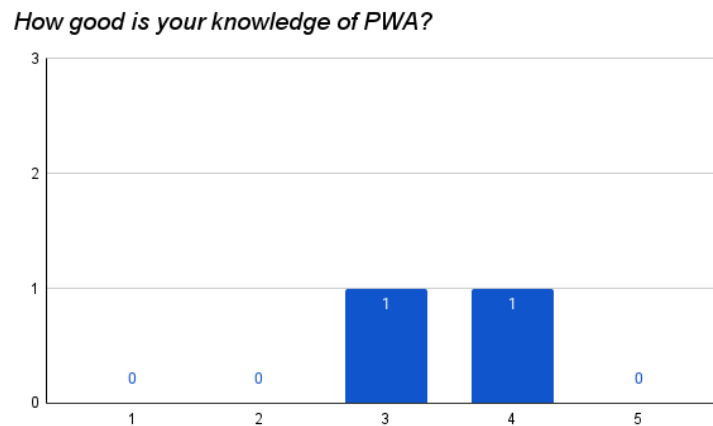


Figure 5.31: Answers for ESQ 10

ESQ 10 is a duplicate of DSQ 8 and asked the respondents about their perceived level of knowledge of PWA, results seen in Figure 5.14. The answers were given on a scale of one to five, with one being "None" and five being "Very good". One respondent answered 3/5 and one respondent answered 4/5, which corresponds to an "ok" and "good" level of knowledge respectively.

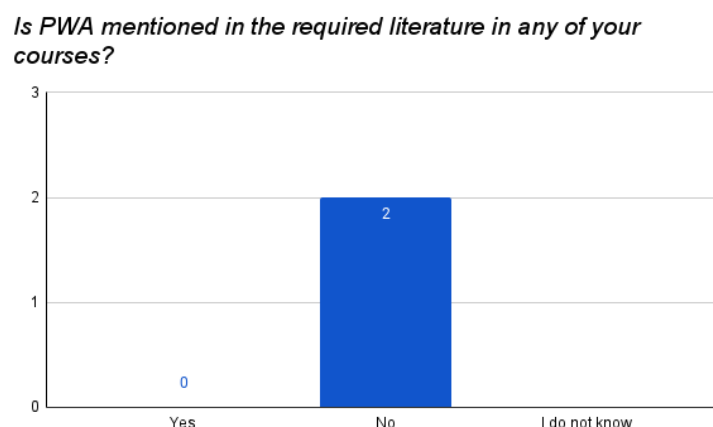


Figure 5.32: Answers for ESQ 11

ESQ 11 asked respondents whether or not PWA was covered by the required literature of their courses, results seen in Figure 5.32. Both respondents answered "No" to the question.

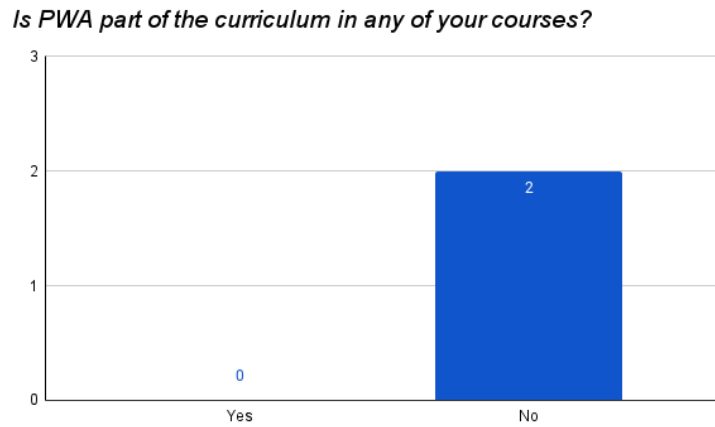


Figure 5.33: Answers for ESQ 12

ESQ 12 asked respondents whether or not PWA was mentioned in the curriculum of their courses, results seen in Figure 5.33. Both respondents answered "No" to the question.

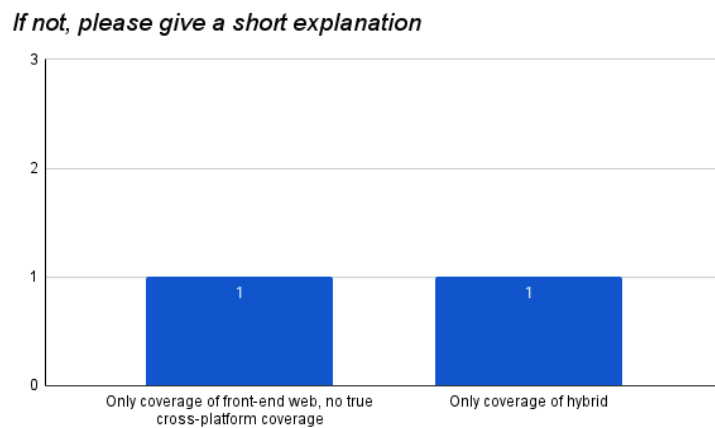


Figure 5.34: Summary of answers for ESQ 13

ESQ 13 was a follow-up to ESQ 12 and asked respondents to elaborate on why PWA was not covered in their courses, results seen in Figure 5.34. One respondent answered that the only cross-platform they covered was front-end web with JavaScript frameworks, but that no "true" cross-platform development was covered. One respondent answered that the only cross-platform apps covered by their courses were hybrid apps with the "React Native" framework.

Has it ever been requested that PWA be included in the curriculum?

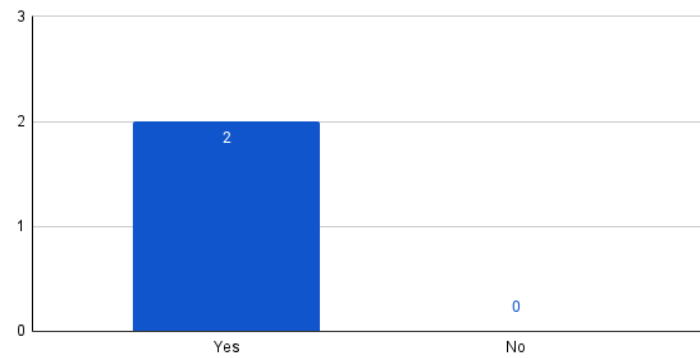


Figure 5.35: Answers for ESQ 14

ESQ 14 asked respondents whether or not it had ever been request that PWA was added to their courses, results seen in Figure 5.35. Both respondents answered "Yes" to the question.

If yes, from whom?

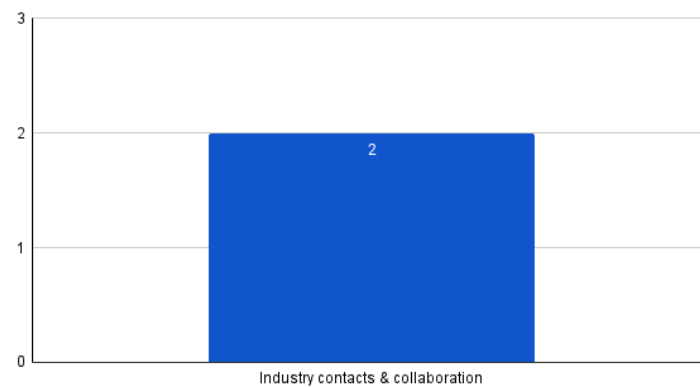


Figure 5.36: Summary of answers for ESQ 15

ESQ 15 was a follow-up to ESQ 14 and asked respondents to describe whom had made the request to add PWA to their courses, results seen in Figure 5.36. Both respondents answers mentioned that the request had been made from outside of the courses or programs. One respondent further clarified that the request was made from actors that provide assignments from students in project courses.

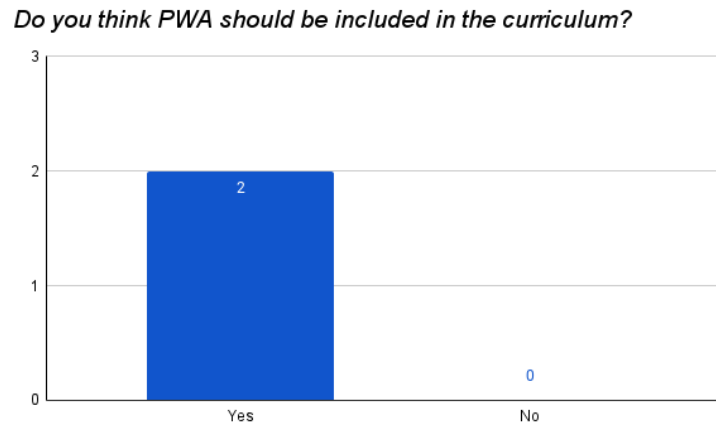


Figure 5.37: Answers for ESQ 16

ESQ 16 asked respondents whether or not they believe PWA should be included in the curriculum for their courses, results seen in Figure 5.37. Both respondents answered "Yes" to the question.

6 Analysis

This chapter contains analysis of the collected data.

6.1 Literature Study

This section analyses the results of the literature study performed at the start of the research project. The formulation of the issues in DSQ 12 was not identical to the formulation of the issue categories identified by the study, but they still contained the same core issues. Figure 6.38 shows a comparison of the issues mentioned by the literature and the issues mentioned by the developers. The drastic difference in number of mentions is simply a result of the number of respondents being lower than expected. Interestingly enough, the literature and the developers do not mention the different issues the same amount relative to the total amount of mentions. The most mentioned issue in the literature was that hardware access was restricted, while only one single developer mentioned this as an issue. The second and third most mentioned issues in the literature were instead the first and second most mentioned by the developers. One reason for these differences could be that the developers were simply not experience enough with PWA to have encountered the same issues as the literature describe. Issues with hardware and software access is probably only encountered if you are actually developing the application, while issues such as poor iOS support and troubles with publishing on app stores is something that an end user can encounter.

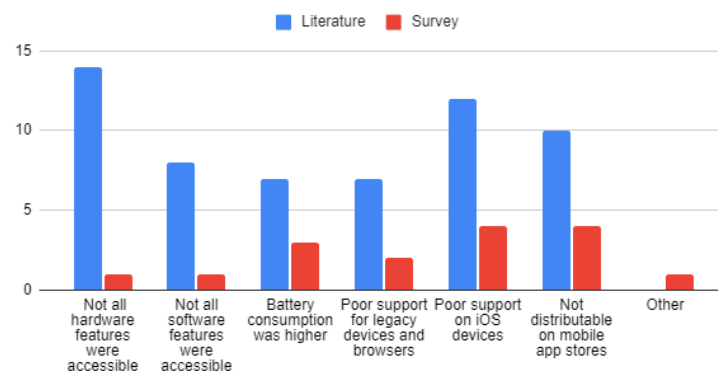


Figure 6.38: Issues mentioned in literature versus issues mentioned by developers

6.2 Developer Survey

This section contains the analysis of the results gathered from the survey given to application developers. The survey was answered by 16 respondents, which was deemed an acceptable amount for drawing generalized conclusions.

6.2.1 Part 1 - Personal Details

The first part of the survey consisted of questions DSQ 1 - DSQ 3.

Most of the respondents (14/16) were from within Europe while only 2 respondents were from outside of Europe. More specifically, 12/16 respondents were from Sweden alone. The vast difference in the response rate from the different countries might be a result of cultures and more specifically how companies treat their business email. An additional factor to consider is that Swedish companies received an email written in Swedish

while the all other companies received an email written in English. The external validity of the findings is greatly impacted by the Swedish bias and it cannot be said that the results represent the broader application development industry.

An additional bias in the results emerged from the responses to DSQ 2, in which respondents could choose to include the name of their company. Several respondents provided the same company name, making the results slightly skewed towards developers of one company instead of developers in general.

The third question asked about the age group the respondents belonged too and in contrast the the first two questions, the answers were almost perfectly split between the three age groups of 20-29, 30-39, and 40-49. This even distribution indicate that at least from an age perspective, the results represent developers generally. Section 6.2.3 contains further analysis of this question in combination with DSQ 8.

6.2.2 Part 2 - Experience

The second part of the survey consisted of questions DSQ 4 - DSQ 7.

This part of the survey was introduced by asking the developers how many years they had been in the profession and what their primary development platform currently is. These two questions will not be individually analysed in this section since they do not provide interesting data if viewed in a vacuum. These question will instead be analysed in combination with the results from DSQ 8, which can be seen in Section 6.2.3.

The next question followed up by asking what type of applications the respondents had worked on before. The results show a somewhat balanced number of answers for iOS, Android, web app, and cross-platform, which indicates that the demographic had a good balance of different experiences. What is most notable is that only one respondent had ever worked on a PWA before. All questions about opinions and experiences are therefore not answered with the expertise of someone with intimate experience of actually working with PWA.

The respondents who said that they had worked on cross-platform were then asked to specify which frameworks they had experience with. The different frameworks were mostly equally popular with React Native sticking out from the rest as the most popular and Xamarin as the least popular. This result was more balanced than expected, as during the process of searching for companies the most frequently mentioned framework was by far React Native. This might be a result influenced by the mostly Swedish respondents.

6.2.3 Part 3 - PWA

The third part of the survey consisted of questions DSQ 8 - DSQ 16.

Asking developers if they would consider developing a PWA in the future resulted in nine respondents answering Yes while seven answered No. If the individual responses are analysed by also examining DSQ 3, 4, 5, and 8 at the same time, some more interesting results emerge. Figure 6.39 shows a clear trend that developer tends to be more negative towards developing a PWA the older they get. One can argue that this result is a reflection of broader trends in society; younger people being more open to trying new things while older people are less open to changes.

An almost exact replica of this trend can be identified if the question is instead analysed in combination with years of experience, as seen in Figure 6.40. Analysing the results derived from combining the question with the respondents' primary development work is much easier.

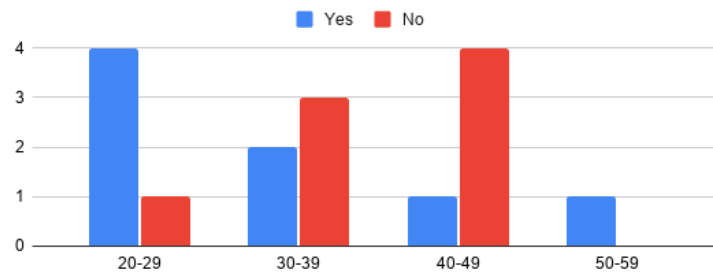


Figure 6.39: Combination of results from DSQ 3 and DSQ 9

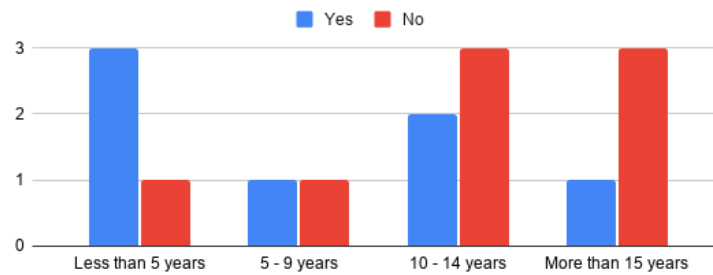


Figure 6.40: Combination of results from DSQ 4 and DSQ 9

As seen in Figure 6.41, iOS developers were much more likely to be against the idea of developing PWA, something that is easily explained by the knowledge of how much more restricted PWA is on iOS compared to Android. The final analysis for this question is on the results gained from combining the question with the self-reported knowledge of PWA as seen in Figure 6.42.

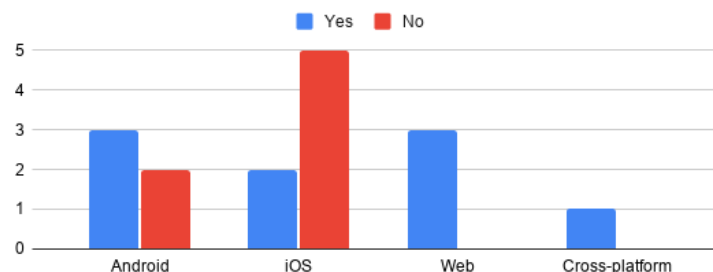


Figure 6.41: Combination of results from DSQ 5 and DSQ 9

All respondents with a 2/5 level of knowledge of PWA were much more likely to be against using the technology in the future compared to respondents that answered 3/5 or higher. A high level of knowledge should mean that the respondent much more aware of not only the drawbacks, but also all the capabilities of PWA. This might indicate that a well-informed respondent is more likely to realize what type of projects PWA is fit for, and is therefore able to see it as a viable alternative.

When asked about what made other cross-platform alternatives preferable to PWA, by far the most common response was that they were closer to native in design or simply that they used native components. Two respondents said that their preference was for alternatives that they had more experience with, while two respondents mentioned that PWA relies too much on the browser. All of these responses are expected since most of the respondents were primarily native developers.

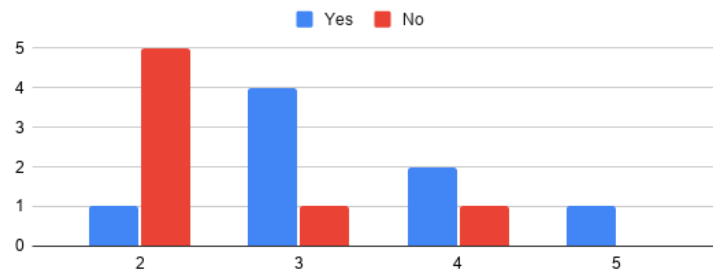


Figure 6.42: Combination of results from DSQ 8 and DSQ 9

When asked the same question but instead about native development, the results were more varied. While answers about better design language for the platform, user experience, features, and performance were all expected, one respondent mentioned that web standards were slow to adapt and to be adopted by browsers. This was a point nobody else had mentioned, and it highlights the important issue of day-1 support for new features.

The question about what issues the respondents had encountered with PWA has been analysed in Section 6.1. It was followed up by asking when the issues were last encountered. That all respondents answered either 2020 or 2021 was surprising since it was expected that at least some of the developers had come in contact with PWA at a time when it was less competitive than today, and stayed away from it since then.

The answers gained from asking what the respondents' opinion of PWA was were very varied. Most of the answers can be interpreted as developers proffering native solutions and that PWA will not be their preferred choice until it becomes a more mature and widely used alternative. The opinions of PWA not being good enough for installable apps probably stems from a mistrust in the capabilities of browser engines.

Most respondents did not agree with the statement that "PWA is the future [...]". By analysing the previous responses, this does not come as a surprise since the respondents appear very reluctant to consider cross-platform as a better alternative to native apps.

All respondents who disagreed were then asked to explain what they believe is necessary for the statement to become true. Most of the respondents expectedly answered that PWA needs to become at least on par with native in essentially all regards. What can be seen from all the answers is that PWA is considered to be sub-par to native and that very large changes need to take place before it becomes the preferred option. The most interesting results were the two respondents who mentioned that PWA perhaps should not change, but the market might instead need to change to better fit PWA. This indicates that these respondents were well-aware of the possibilities of PWA, but they think that as the market is today the need for it is not present. A similar response to this was that PWA needs to be advocated as the standard before it can be the goto alternative, which can also be interpreted as need for a change in the market.

6.2.4 Part 4 - Your Company & Clients

The fourth part of the survey consisted of questions DSQ 17 - DSQ 19.

The first question in this part of the survey asked if a client had ever asked for a PWA directly. As expected, a large number of the respondents did not have any knowledge about it and could not give an answer. That five developers answered Yes to the question raises some questions about how much the clients can influence the decision of underlying technology. Since only one developer had ever worked on a PWA, this answer can be interpreted as the client's suggestion not having any real influence.

The question was then asked in reverse; had the developers themselves proposed PWA? Only two developers answered Yes to the question.

The next question asked what type of apps the respondents' companies offered. These two question have answers that reflect each other well: only three companies offer PWA. A developer working at a company that does not offer PWA naturally does not consider suggesting it to a client. That most companies offered native and web apps is also in line with the type of apps the developers primarily worked on. Out of all 16 respondents, there were only two that stated that their company only offers web apps to their clients. The other answers given by these respondents were almost polar opposites; one of them was very open towards the idea of PWA while the other criticized the tech heavily on the basis of poor iOS support. The respondent that was against PWA also answered yes to the question of if a customer had ever asked for the tech to be used.

6.2.5 Part 5 - Mobile Applications

The fifth and final part of the survey consisted of one multi-part question, DSQ 20.

The results seen in Table 5.15 are not as varied as what was originally expected. The results almost follow a descending order of importance without the need to re-arrange the order of any table rows. While the features in the question were taken from Table 3.3, the order was re-arranged to create one group of features better supported on native and one group of features better supported on PWA. In hindsight, it might have been better to completely randomize the ordering to avoid the potential bias from the order the rows are read. As it stands now, there is a risk that the result has been influenced by the features with better PWA support being placed last.

The features that were ranked as the least important were offline usability, cross-platform support, and that the app is discoverable from a web search. All three of these are features that PWAs provide somewhat easily, and they are often mentioned as good characteristics of the tech. Important to note is that offline usability and cross-platform support were the most divisive features in the entire question since the answers were mostly evenly split apart from the spikes at both extremes. It seems like these two features were either deemed as important if the respondent was more familiar with PWA while they were deemed as quite unimportant if the respondent was primarily a native developer. With all that being said, the results are in line with what the respondents have answered in previous questions about their opinions on PWA, making the result less unexpected.

The results from this question can quite easily be summarized as follows: the respondents rank features with excellent native support as more important than features with excellent PWA support.

6.3 Education Survey

This section contains the analysis of the results gathered from the survey given to lecturers in higher education. Since there were only two respondents for this survey, the analysis will be briefer than the analysis for the developer survey. Four to five respondents would have been deemed a suitable number for conducting deeper analysis and discussion.

6.3.1 Part 1 - Personal Details

The first part of the survey consisted of questions ESQ 1 - ESQ 5.

Unsurprisingly, both respondents teach in Sweden. The intent was to send the survey to lecturers in higher education in the same countries as the developer survey. The question was left in instead of removed and can at best be a form of quality control. The results of this question can at least be interpreted as having been answered by the correct persons. The assurance that the survey had been answered by the correct persons was further strengthened by the fact that the respondents both gave the name of their institutes. Both of the answers given matched with the contact list.

Despite the respondents being between the ages of 40-49 and 50-59, they had only been teaching the subjects of mobile app development for less than five years and 5-9 years respectively. Although the somewhat short experience in the subjects does not degrade the quality of the answers, the more experience the respondents have the better when it comes to their opinions given in later questions.

6.3.2 Part 2 - Mobile Applications

The second part of the survey consisted of questions ESQ 6 - ESQ 9.

This part of the survey was introduced by asking what types of applications were covered by the respondents' courses. This question was asked to get an overview of the current coverage of application types to see if the industry matches with the education sector in terms of what applications are relevant. Since all options were selected at least once we can interpret this as the education having good and varied coverage. If more responses were gained it would have been interesting to see if the coverage would have remained equally good or if these two respondents are an anomaly.

One of the respondents did not have any coverage of cross-platform apps in their courses and was asked if they think this topic should be covered in the future. Since the respondent answered "Yes" to the question, we can assume that perhaps the coverage will be even more varied in the future. The other respondent was instead asked if they covered any specific cross-platform framework in their courses, to which they answered "React Native". Since React Native was the most commonly used framework by developers, the answers to the two surveys match up in this case.

Just like in the developer survey, the education survey also asked the respondents to rank the importance of mobile features and characteristics. Since there were only two respondents, the results can not be analysed in same way as the results from the same question in the developer survey. No real trends can be identified because of this. What can be said is that for the most part, the two respondents answered very similarly to each other. The only two features that differed more than one ranking were offline usability and web-search discoverability. Summarizing all the rankings given in this survey is not as easy as for the developer survey, since here the responses do not follow an easily identifiable trend. This time the results might simply be a case of personal preference instead.

6.3.3 Part 3 - PWA

The final part of the survey covered ESQ 10 - ESQ 16 and was introduced by asking how good the respondent's knowledge of PWA was, in which the respondents answered 3/5 and 4/5 respectively. This is a higher average than for the developer survey, which is something that might influence the opinions expressed about the tech to be more positive than if the knowledge would have been lower.

Earlier the question of cross-platform coverage was asked but to get an overview of PWA in particular three additional questions were asked about its coverage. None of

the respondents' courses covered PWA in any fashion; neither the curricula or literature had any coverage of it. While this answers the question of how good the coverage of PWA is, it does not give any indication as to the underlying reasons. When asked for a further explanation, one of the respondents answered that they only cover one type of cross-platform app while the other answered that they do not have any coverage of cross-platform at all. In conjunction with a question of whether the respondents think PWA should be added to their courses, the answers can be interpreted as follows: the respondents feel a need to add PWA to their courses, but do not currently have the ability to add it to their existing courses. There might be a need for adding new courses if the topic is to be covered at all.

Finally, it was asked if there has been any requests to add PWA to their courses at all. Since both respondents had received such a request from collaborative sources in the industry, there appears to be a need for PWA to be covered in higher education. What raises further questions is that this need is not at all mirror by the results from the developer survey. Clear conclusions appear difficult to draw from these results.

6.4 Checklist of Criteria for Widespread Adoption

The checklist seen in Table 6.17 was created after the analysis of the literature study and the surveys was completed. It is essentially a condensed interpretation of the results from Chapter 5. The list contains criteria that need to be fulfilled before PWA has a good chance of achieving widespread adoption as a mobile app development technique.

The criteria were first selected based on what the literature study found as current issues with the tech. With the data from the surveys, what criteria were included in the list was changed to be what is seen in the table. Finally, the the list was ranked in descending order of importance; the further down the list the less important the criteria are.

Since not all criteria are of equal importance, this list might be seen as more of a "wish list" of what developers want with PWA. This means that not all criteria need to be fulfilled, as they are suggestions for needed changes rather than the final solution to the problem of low adoption.

Number	Criterion
1	iOS support must become equal to Android support
2	Device hardware and software access must become equal to native
3	Performance must become equal or exceed native
4	Distribution on app stores must become easier
5	The UX offered by web standards must become equal to that of native

Table 6.17: Checklist of criteria for widespread adoption

7 Discussion

This chapter discusses our findings and compares the results of this research project to related work done by other researchers

7.1 Findings & Answers to Research Questions

While the survey aimed at the education sector was originally planned to have the same scope as the developer survey, i.e., it was to be given to lecturers in different countries, it proved nigh impossible to get the information needed to find relevant lecturers in countries other than Sweden. Since finding relevant companies was equally easy for all selected countries, only the scope of the education survey was changed. The developer survey had 16 respondents while the education survey only had two respondents, as mentioned in previous chapters. The number of developers that participated in the developer survey is sufficient for drawing conclusions, but the findings are not as generalizable as was the intention at the start of the project. Since a large majority of the respondents were from Sweden, the effort to try and involve countries of different regions was not fruitful. The results of the developer survey can only be used to draw conclusions about Sweden in particular and can not be applied to the wider application development industry of the world.

With that said, the findings do provide enough for answering most of the research questions. From the analysis of the data from DSQ 20 and ESQ 9, it is very clear that PWA only provides partial support for the most important features and characteristics of mobile applications. Features such as the ability to be published on mobile app stores and be installed to devices were ranked as very important and have full support on PWA. However, the feature ranked as most important out of all of them was the ability to send proper push notifications; a feature not supported on PWA on iOS devices. As can be seen numerous times throughout this report, poor iOS seems to be the most crippling disadvantage of PWA compared to other solutions. All this simply means that PWA can never fully realize all of the most important features of mobile apps as long as iOS continues to poorly support the tech. It should also be mentioned that the results from the developer survey and the education survey differs slightly when it comes to the rankings of importance. Background sync, cross-platform support, and low battery usage were the outliers from the education survey. While these results differ, it might simply be a result of the on average higher level of knowledge of PWA that the lecturers had. If that is the reason for the different results, the developer survey might have had drastically different rankings had they average level of knowledge been high.

By looking at Figure 6.38 it is evident that the respondents do experience the same problems as described in the literature, at least to some degree. While the frequency of the issues differs, all issues have some overlap. This result does not give a definitive all-encompassing answer to the question however, since most of the developers lacked actual experience of developing PWA. The only thing the results can tell us is that developers with experience of *using* a PWA do experience the same problems as the literature describes. Missing is still the answer of whether PWA developers experience the same issues or not. A partial answer seems to be the best we can get from our results

Even greater insight into how developers feel about PWA was gained by letting the developers express their thoughts freely in their own words. The analysis shows that almost all of the respondents have a very negative view of PWA and that they do not think very highly of its future. Most of these opinions seem to come from either their lack of knowledge of the technique or simply that the problems that exist today are simply

too severe to ignore. Since there was an almost universal consensus in the results, the question of what developers think about PWA and its future was clearly answered.

The findings of this research indicates that there are currently five goals that need to be achieved before PWA can have high adoption in the market. The goals, or criteria, can be seen in Table 6.17. The fact that this checklist can be created somewhat easily means that there are problems with PWA that must be solved before it can become a true competitor to not only other cross-platform apps, but also to native apps. The feasibility of the tech ever becoming the definitive solution to mobile app development can be heavily questioned upon reviewing the checklist. As it stands today, the most difficult hurdle to overcome is that Apple does not grant enough access to utilize the capabilities of their iOS devices. Without Apple's permission, the limitations on iOS will forever keep PWA to only be fit for certain applications on the platform. This also hinders PWA from becoming a go to solution for Android development, since the major advantage of the tech is that it is cross-platform. The other items on the checklist are much more achievable since the necessary improvements can be made regardless of Apple's support. An interesting point that one of the respondents brought up was that a disadvantage of PWA is the reliance on web standards since they are exceptionally slow to be adopted by browser vendors. This was not included in the checklist since the information comes from only one source, but shines some light on an additional issue that can hinder the adoption of PWA. Even with the problems the tech currently faces, its potential is still very high. Advantages such as (almost) full cross-platform support, low application size, and cheap and efficient development can not be understated. If the features that are currently limited on iOS are not needed, the use of a PWA should be seriously considered.

How customers/clients influence the decision of the underlying tech for their apps could not get a definitive answer from this research. This is mostly due to poor planning on what data was necessary to answer the question, and not the fault of the respondents. What the survey needed was one or two more questions about the topic since the two questions asked did not allow for any qualitative answers to be given. While this research question was included to gain a bit more knowledge around the topic of PWA, it was not deemed as one of the primary questions the research was to focus on. That is however not an excuse for not properly attempting to answer the question. With that being said, what we can state is that our findings do not indicate that customers/clients influence the decision to any large degree. This was derived from the fact that almost none of the developers had worked on PWA or even had good knowledge of it, despite the fact that a somewhat sizable number of clients had requested the tech to be used. We cannot state this in good confidence though, for the reasons previously mentioned.

Our research does not provide a definitive answer to the state of PWA coverage in higher education either. Our findings indicate that the coverage is exceptionally poor, to the point of non-existence. With only two lecturers partaking in the survey, we can not say that our results are generalizable to any degree. A higher number of respondents would have been needed to draw reliable conclusions. Even though only there were only two respondents, they agreed to a degree that makes us able to at least discuss the results somewhat instead of disregarding them as fully inconclusive. Our findings indicate that while coverage is low today, it might change in the coming years since lecturers are not only open to the idea of including PWA in their courses, but also feel like it is something that should be done. That the sources requesting that PWA is taught comes from the industry is also a very interesting result, since it to an extent contradicts what we have found from the developer survey. If almost no companies offer PWA and almost no developers have any experience working with the tech, why is it requested that it be taught in

higher education? Perhaps there are some sources in the industry that are pushing for the adoption of the technology that we somehow avoided in our search for app developers.

7.2 Related Work

This research project is similar to the work done in 2016 by Steczko [8] and in 2020 by Olsson and Gustafsson [9], as both of these projects investigated similar topics. Steczko's work mainly included interviews with developers that investigated the opinions of cross-platform and native development. While Olsson and Gustafsson conducted a very similar study, their area of investigation was on PWA. While both of their studies primarily used interviews for data collection, this project used surveys instead. Both of their studies agree with the findings of this research project; that developers prefer native development over cross-platform development. While many of the results in this project are similar to results from Olsson and Gustafsson, one of the most significant differences is developers' thoughts on the future of PWAs. Olsson and Gustafsson found a majority of the developers they interviewed believed that PWA is the future of mobile app development which is the opposite of our findings which are that an overwhelming majority disagreed with the statement.

Thacker and Dharani concluded that they expect PWA adoption to rise since PWA does not differ meaningfully from native apps, and even offers some unique benefits [13]. Our findings do not suggest that a rise in adoption is likely to occur since the opinions of developers seems to be very negative. The heavy restrictions on iOS devices further decreases the likelihood of an increase in PWA adoption, since it appears to be one of the most significant disadvantages present. While our research was mostly focused on what problems PWA faces today, Thacker and Dharani mostly focused on the potential benefits the tech has.

Majchrzak et al. [14]: In the paper from 2018 by Majchrzak et al. they discuss how PWA has the potential to become "definite approach to cross-platform development". Our findings are to some extent very similar, as they also identified that poor iOS support is a disadvantage that could greatly hinder the adoption of PWA in the market. We agree that PWA offers a great benefits for the market as a whole, but that the possibility the tech becoming the future of mobile app development is uncertain.

The investigation of user experience of PWA versus native done by Sedkowska in 2020 concluded that the only differences that could be identified were minor and would not be noticeable unless the two apps were directly compared [15]. While our findings suggest that there are some major disadvantages with PWA, the difference between our results is mainly an effect of the different approaches taken and the different goals we had. While we investigated what developers think of the tech in general, Sedkowska investigated the usability and user experience on an Android device only. The disadvantages our findings identified could not really be noticed in the type of test Sedkowska performed, especially since no iOS device was used.

None of the related works discussed in this chapter contain studies or investigations about the education sector that are similar to ours.

8 Conclusions

This chapter contains a summary of the discussion and conclusions, as well as suggestions for future work.

8.1 Summary of Conclusions

This research project has taken a closer look into the opinions that mobile app developers and lecturers in higher education have about Progressive Web Applications. Before the project began, our suspicion was that PWA was under-utilized in the industry and that the knowledge that app developers have about the tech was poor. This suspicion was based findings from the, admittedly very limited, academic research previously conducted on this topic.

To start the investigation of whether PWA deserves broader utilization or not we first conducted a literature study. The study found that PWA has many advantages over native applications, but that even though the disadvantages are few they are quite severe. The knowledge gained from the study was used to help create the questions for the surveys.

There were two surveys conducted; one answered by mobile app developers and one answered by lecturers in higher education in Sweden. The developer survey received a total of 16 responses (12 of which were Swedish developers) while the education survey only received two responses. From the surveys we found that developers' knowledge of PWA was very poor and that their general opinions on the tech was mostly negative. Developers with a higher level of knowledge had a tendency to have more positive opinions of PWA.

This project has contributed to the general knowledge of PWA and could be of interest to not only the industry of application developers but also lecturers in higher education. By highlighting what the current issues that face PWA are in conjunction with the opinions of tech, we want to encourage the technology to be further developed to become the best it can be. To make the necessary improvements more concrete, a short checklist containing the biggest issues has been provided and ordered according to the perceived importance of each issue.

The validity and significance of the findings in this report are negatively impacted by the low response rate by developers that were not from Sweden as well as the low response rate of the lecturers. The main improvement to this study would be more aggressive when trying to receive responses to increase the total number of participants in the study.

8.2 Future Work

Since there were several aspects of this research project that did not go according to plan there are many suggestions for future work that can be made.

This project did not manage to generalize the results to more than just one country, so a similar study could be performed in the future in which the process of contacting companies involves much more extensive work than just sending one email. Additionally, an extended version of this project should attempt to gain a much larger amount of responses to further increase the validity of the results. Since the PWA is still a relatively unknown solution a revisit of the topic with a similar study in a few years' time could be of interest. If the tech improves enough to no longer have the major restrictions seen today it would be interesting to see if the negative opinions seen today will have changed.

It might also be interesting to conduct research that instead has the focus on the technology itself. This research could for instance interview some of the persons involved

with the development of PWA to gather their opinions and thoughts on the tech and its current restrictions. Asking the question of whether they think PWA is the future or not might yield vastly different results from these persons than what was gained from surveying the industry.

A suggestion is to also do a study that is exclusively about the education sector to uncover more about if there is currently a missed opportunity to provide students with the knowledge they need when they enter the industry. Finally, a study could be conducted about PWA from the perspective of a client that is ordering a mobile app from a developer since this study did not have any conclusive results about this topic. It could be interesting to get the full picture of the exchange to see if perhaps there is a need to change how decisions about which tech to use are made.

References

- [1] Ramonak. “Steve jobs introducing pwa in 2007”. 22 Oct 2019. Accessed on: 13 Feb 2021, [Online]. Available: https://www.youtube.com/watch?v=QvQ9JNm_qWc.
- [2] gamingandtechnology. “Steve jobs introduces the app store - iphone sdk keynote”. 14 Mar 2008. Accessed on: 4 Mar 2021, [Online]. Available: https://www.youtube.com/watch?v=xo9cKe_Fch8.
- [3] A. Russell. (Jun. 2015). “Progressive web apps: Escaping tabs without losing our soul”, [Online]. Available: <https://infrequently.org/2015/06/progressive-apps-escaping-tabs-without-losing-our-soul/>. [accessed 14 Feb 2021].
- [4] S. Richard, P. LePage. (Feb. 2020). “What are progressive web apps?”, [Online]. Available: <https://web.dev/what-are-pwas/>. [accessed 7 Mar 2021].
- [5] A. Bar. “What web can do today”, [Online]. Available: <https://whatwebcando.today/>. [accessed 14 Feb 2021].
- [6] Mia. (Jun. 2019). “Progressive web apps (pwa): Top 3 case studies”, [Online]. Available: <https://www.simicart.com/blog/pwa-case-studies/>. [accessed 15 Feb 2021].
- [7] Intersog. (Aug. 2020). “Is a progressive web app the future of multi-platform mobile development?”, [Online]. Available: <https://intersog.com/blog/is-progressive-web-app-pwa-the-future-of-multi-platform-development>. [accessed 1 May 2021].
- [8] J. Steczko, “Analysis of companies’ experience with cross-platform development compared to native development for mobile devices”, B.S. Thesis, Dept. of Comp. Science, Linnaeus University, Växjö, Sweden, 2016. [Online]. Available: <http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A945760>.
- [9] V. Gustafsson and M. Olsson, “Progressiva webbapplikationer: En analys av erfarenheter och omfattning av tekniken i sverige”, B.S. Thesis, Dept. of Comp. Science and Media Tech., Linnaeus University, Växjö, Sweden, 2020. [Online]. Available: <http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1446696>.
- [10] S. J, “How does the user experience of a progressive web application compare to native application?: A case study on user’s attitude in context of social media”, B.S. Thesis, Dept. of Comp. Science, Linnaeus University, Växjö, Sweden, 2017. [Online]. Available: <http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1105475>.
- [11] Google. (Nov. 2018). “Nikkei achieves a new level of quality and performance with their multi-page pwa”, [Online]. Available: <https://developers.google.com/web/showcase/2018/nikkei>. [accessed 6 Mar 2021].
- [12] Google. (Feb. 2018). “George.com enhances the mobile customer experience with new progressive web app”, [Online]. Available: <https://developers.google.com/web/showcase/2018/asda-george>. [accessed 6 Mar 2021].
- [13] P. Thacker and D. Dharani, “Realization of native apps using Progressive Web Apps”, *JETIR*, vol. 7, pp. 774–782, 2020.

- [14] A. Biørn-Hansen, T. A. Majchrzak, and T. Grønli, “Progressive Web Apps: the Definite Approach to Cross-Platform Development?”, in *HICSS*, 2018, pp. 5735–5744.
- [15] S. J., “How does the user experience of a progressive web application compare to native application?: A case study on user’s attitude in context of social media”, B.S. Thesis, Sch. of Eng., Jönköping University, Jönköping, Sweden, 2020. [Online]. Available: <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1455626>.
- [16] Clutch. “Top mobile app development companies”, [Online]. Available: <https://clutch.co/directory/mobile-application-developers>. [accessed 21 Apr 2021].
- [17] B. S. Gardner, “Responsive Web Design: Enriching the User Experience”, *Sigma Journal*, vol. 11, pp. 13–19, 2011. [Online]. Available: <http://www.webdesignblog.gr/wp-content/uploads/2012/03/5.pdf>, [accessed 7 Apr 2021].
- [18] W. S. El-Kassas, B. A. Abdullah, A. H. Yousef, and A. M. Wahba, “Taxonomy of Cross-Platform Mobile Applications Development Approaches”, *Ain Shams Engineering Journal*, vol. 8, pp. 163–190, 2017.
- [19] N. Serrano, J. Hernantes, and G. Gallardo, “Mobile Web Apps”, *IEEE Software*, vol. 30, pp. 22–27, 2013.
- [20] Statista. (Feb. 2021). “Number of apps available in leading app stores as of 4th quarter 2020”, [Online]. Available: <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>. [accessed 7 April 2021].
- [21] B. Pan, K. Xiao, and L. Luo, “Component-based mobile web application of cross-platform”, in *2010 10th IEEE International Conference on Computer and Information Technology*, 2010, pp. 5735–5744.
- [22] R. Furman. “A definitive overview on mobile app technology stack”, [Online]. Available: <https://uptech.team/blog/mobile-app-technology-stack>. [accessed 9 Apr 2021].
- [23] C. Raj and R. Tolety, “A study on approaches to build cross-platform mobile applications and criteria to select appropriate approach”, in *INDICON*, 2012, pp. 625–629.
- [24] L. Vu. (Jan. 2021). “Publishing pwass to major app stores: The whys and hows”, [Online]. Available: <https://www.simicart.com/blog/pwa-app-stores/>. [accessed 7 Apr 2021].
- [25] I. Malavolta, G. Procaccianti, P. Noorland, and P. Vukmirovic, “Assessing the Impact of Service Workers on the Energy Efficiency of Progressive Web Apps”, in *MOBILESoft*, 2017, pp. 35–45.
- [26] MDN. (2021). “Javascript”, [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>. [accessed 7 Apr 2021].
- [27] WHATWG. (2021). “Html standard: Web workers”, [Online]. Available: <https://html.spec.whatwg.org/multipage/workers.html>. [accessed 7 Apr 2021].
- [28] W3C. (Nov. 2019). “Service workers 1”, [Online]. Available: <https://www.w3.org/TR/service-workers/>. [accessed 7 Apr 2021].

- [29] K. Basques. (Nov. 2020). “Why https matters”, [Online]. Available: <https://web.dev/why-https-matters/>. [accessed 8 Apr 2021].
- [30] MDN. (2021). “Web app manifests”, [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/Manifest>. [accessed 8 Apr 2021].
- [31] O. Adetunji, C. Ajaegbu, N. Otuneme, and O. J. Omotosho, “Dawning of Progressive Web Applications (PWA): Edging Out the Pitfalls of Traditional Mobile Development”, *ASRJETS*, vol. 68, pp. 85–99, 2020.
- [32] V. S. Magomadov, “Exploring the role of progressive web applications in modern web development”, *Journal of Physics: Conference Series*, vol. 1679, p. 022 043, 2020.
- [33] StatCounter. (2020). “Search engine market share worldwide”, [Online]. Available: <https://gs.statcounter.com/search-engine-market-share/all/worldwide/2020>. [accessed 28 Mar 2021].
- [34] A. Biørn-Hansen, T. A. Majchrzak, and T. Grønli, “Progressive Web Apps for the Unified Development of Mobile Applications”, in *Web Information Systems and Technologies*, Cham: Springer International Publishing, 2018, pp. 64–86, ISBN: 978-3-319-93527-0.
- [35] Google. “Google forms”, [Online]. Available: <https://docs.google.com/forms/>. [accessed 30 Apr 2021].
- [36] Universitetskanslersämbetet. (Jun. 2020). “Lista över universitet, högskolor och enskilda utbildningsanordnare”, [Online]. Available: <https://www.uka.se/fakta-om-hogskolan/universitet-och-hogskolor/lista-over-universitet-hogskolor-och-enskilda-utbildningsanordnare.html>. [accessed 23 Apr 2021].

A Appendix A - Developer Survey

Part 1 - Personal Details

In which country do you work? *

Company name

Your age *

Part 2 - Experience

How many years have you worked as a developer? *single-choice* *

- Less than 5 years
- 5 - 9 years
- 10 - 14 years
- More than 15 years

Pick one; what Best describes your current profession? *single-choice* *

- iOS developer
- Android developer
- Web developer
- Cross-platform developer

What type of applications have you previously developed or worked on?

- Native iOS
- Native Android
- Cross-platform / hybrid app
- PWA
- Regular web app

Have you ever worked with any of the following cross-platform frameworks?

- React Native
- Flutter
- PhoneGap / Cordova
- Xamarin
- Ionic
- Other:

Part 3 - PWA

How good is your knowledge of PWAs? *single-choice **

None Very good

1	2	3	4	5
---	---	---	---	---

Would you consider developing a PWA in the future? *single-choice **

- Yes
- No
- Other:

Is there anything that makes you prefer other cross-platform solutions over PWA?

Is there anything that makes you prefer developing natively over PWA?

Have you ever encountered any of the following problems with PWAs?

- Not all hardware features were accessible, such as Bluetooth or NFC
- Not all software features were accessible, such as contacts or SMS
- Battery consumption was higher
- Poor support for legacy devices and browsers
- Poor support on iOS devices
- Not distributable on mobile app stores
- Other:

When was the last time you encountered any of the problems above? *single-choice*

- 2016
- 2017
- 2018
- 2019
- 2020
- 2021

What is your opinion on PWA as a development technique today? *

For example: "It is a somewhat decent cross-platform solution, but until it has proper iOS support I will not consider it".

How much do you agree with this statement: "PWA is the future of mobile app development" *single-choice **

Strongly disagree

Strongly agree

1	2	3	4	5
---	---	---	---	---

If you disagree, how do you think PWAs need to change before the statement becomes true?

Part 4 - Your Company & Clients

Has a client ever asked for a PWA? *single-choice* *

- Yes
- No
- I do not know

Have you ever proposed PWA as a solution to a client? *

- Yes
- No

What type of apps does your company offer? *

- Native apps
- Hybrid apps
- Cross-platform apps
- PWAs
- Web apps
- Other:

Part 5 - Mobile Applications

Please rank the importance of the following features and qualities of mobile apps

	Not at all important	Slightly important	Important	Fairly important	Very important	No opinion
Available on app stores						
Installable (app store or web download)						
Push notifications						
Background sync						
High performance						
Low battery usage						
Usable while offline						
Cross-platform						
Discoverable from web-search						

Final thoughts

If you have any feedback about the survey or just want to add some final thoughts, please do so [here](#)

B Appendix B - Education Survey

Part 1 - Personal Details

In which country do you work? *

Where do you work?

The name of you university, polytechnic, college, etc.

What is your age? *single-choice* *

- 20 - 29
- 30 - 39
- 40 - 49
- 50 - 59
- 60 or above

In which of the following subjects do you teach? *

- Web development
- Web design
- Mobile app development
- Mobile app design
- Other:

How long have you been teaching in these subjects? *single-choice* *

- Less than 5 years
- 5 - 9 years
- More than 10 years

Part 2 - Mobile Applications

What type of applications do your courses cover?

- Native iOS
- Native Android
- Web
- Cross-platform / hybrid
- Other:

If your courses do not cover cross-platform apps, do you think it should be covered in the future?

- Yes
- No
- No opinion

If your courses do cover cross-platform apps, does it cover any of the following frameworks?

- Xamarin
- React Native
- Flutter
- PhoneGap / Cordova
- Ionic
- Other:

Please rank the importance of the following features and qualities of mobile apps

	Not at all important	Slightly important	Important	Fairly important	Very important	No opinion
Available on app stores						
Installable (app store or web download)						
Push notifications						
Background sync						
High performance						
Low battery usage						
Usable while offline						
Cross-platform						
Discoverable from web-search						

Part 3 - PWA

How good is your knowledge of PWAs? *single-choice **

None Very good

1	2	3	4	5
---	---	---	---	---

Is PWA mentioned in the required literature in any of your courses? *single-choice **

- Yes
- No
- I do not know

Is PWA part of the curriculum in any of your courses? *single-choice **

- Yes
- No

If not, please give a short explanation

For example: "The course does not cover cross-platform development, only native Android"

Has it ever been requested that PWA be included in the curriculum? *single-choice **

- Yes
- No

If yes, from whom?

For example: "A student expressed interest in learning more about it after they learned that 'Twitter Lite' is a PWA".

Do you think PWA should be included in the curriculum?

- Yes
- No
- No opinion

Final thoughts

If you have any feedback about the survey or just want to add some final thoughts, please do so here

C **Appendix C - DSQ 20 Diagrams**

Please rank the importance of the following features and qualities of mobile apps

[Available on app stores]

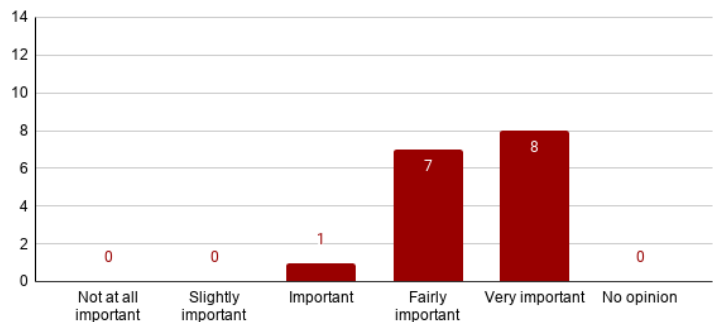


Figure 3.43: Answers for DSQ 20, part 1

Please rank the importance of the following features and qualities of mobile apps

[Installable (from app store or web download)]

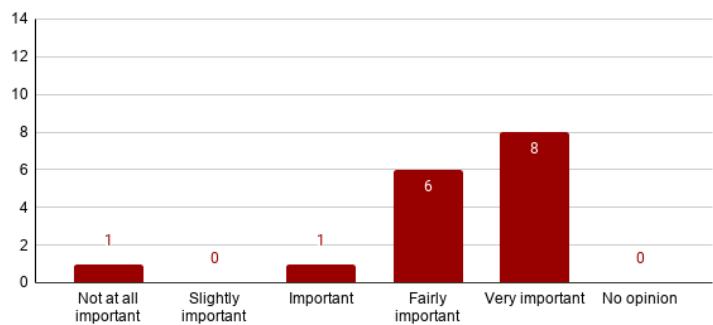


Figure 3.44: Answers for DSQ 20, part 2

Please rank the importance of the following features and qualities of mobile apps

[Push notifications]

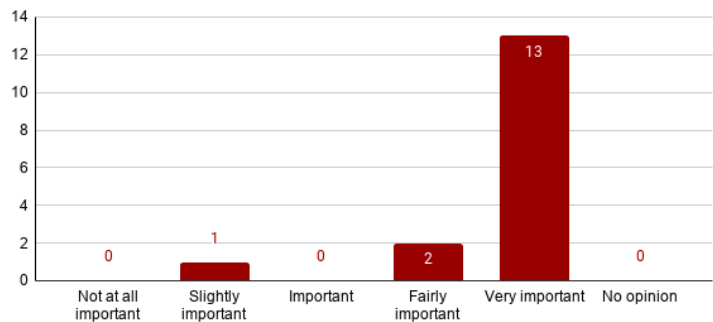


Figure 3.45: Answers for DSQ 20, part 3

Please rank the importance of the following features and qualities of mobile apps

[Background sync]

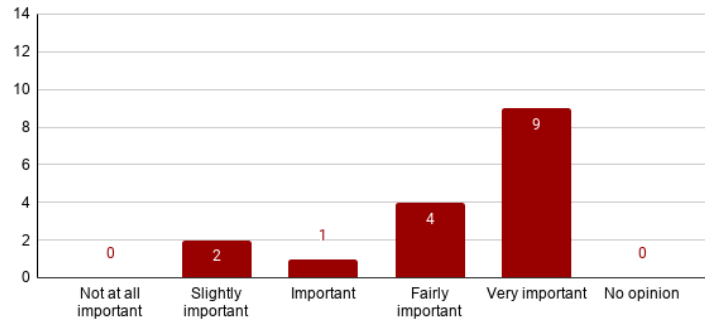


Figure 3.46: Answers for DSQ 20, part 4

Please rank the importance of the following features and qualities of mobile apps

[High performance]

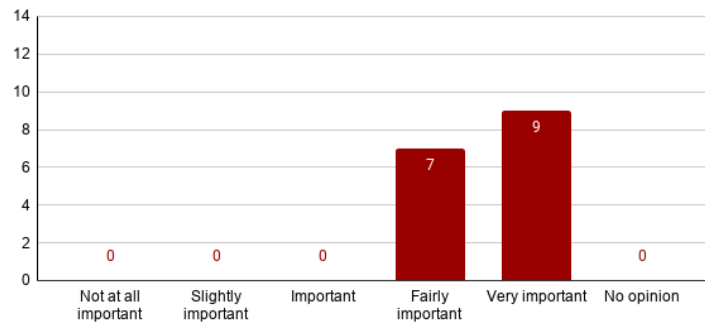


Figure 3.47: Answers for DSQ 20, part 5

Please rank the importance of the following features and qualities of mobile apps

[Low battery usage]

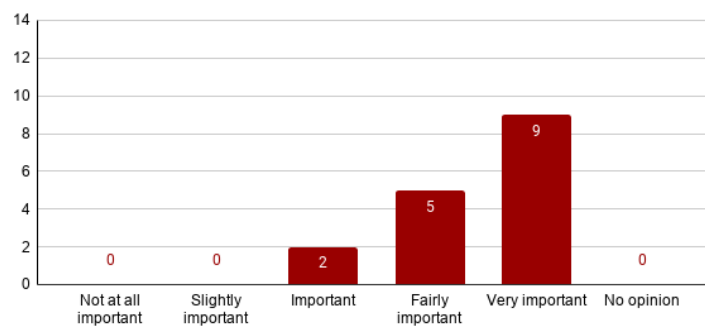


Figure 3.48: Answers for DSQ 20, part 6

Please rank the importance of the following features and qualities of mobile apps

[Usable while offline]

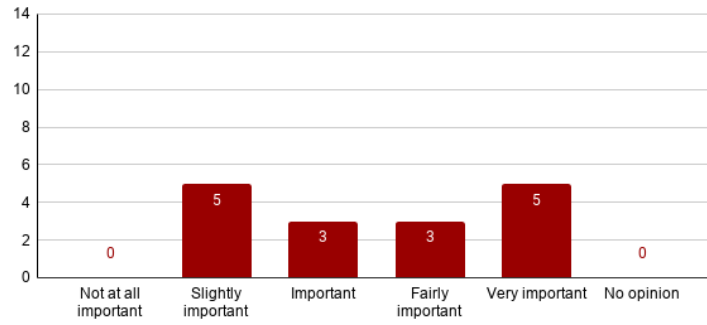


Figure 3.49: Answers for DSQ 20, part 7

Please rank the importance of the following features and qualities of mobile apps

[Cross-platform]

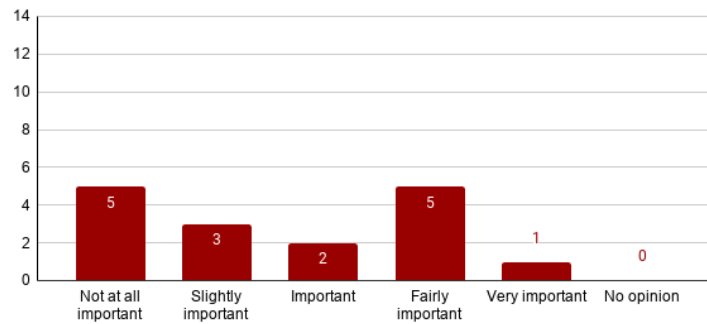


Figure 3.50: Answers for DSQ 20, part 8

Please rank the importance of the following features and qualities of mobile apps

[Discoverable from web-search]

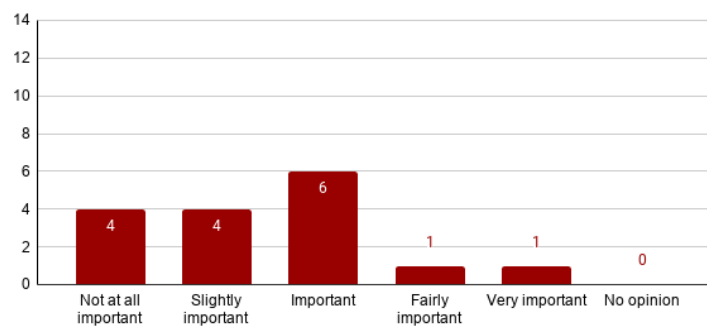


Figure 3.51: Answers for DSQ 20, part 9

D Appendix D - ESQ 9 Diagrams

Please rank the importance of the following features and qualities of mobile apps

[Available on app stores]

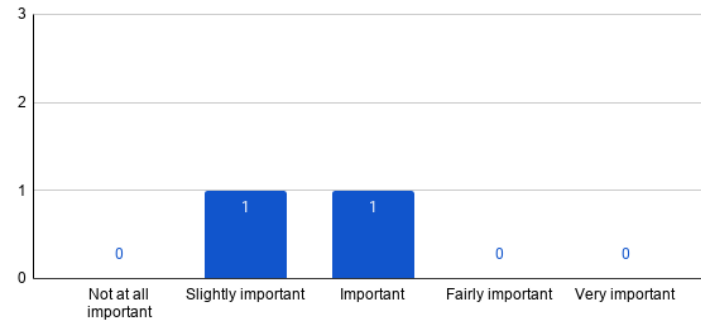


Figure 4.52: Answers for ESQ 9, part 1

Please rank the importance of the following features and qualities of mobile apps

[Installable (from app store or web download)]

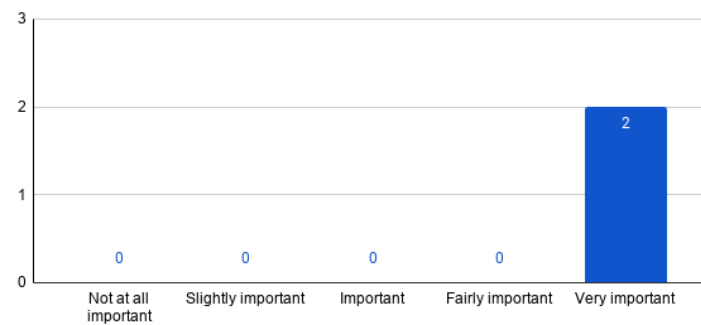


Figure 4.53: Answers for ESQ 9, part 2

Please rank the importance of the following features and qualities of mobile apps

[Push notifications]

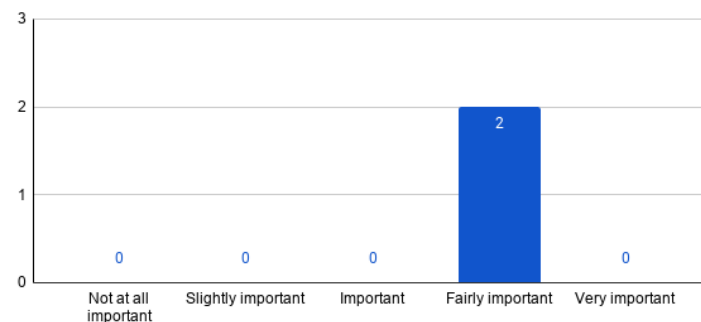


Figure 4.54: Answers for ESQ 9, part 3

Please rank the importance of the following features and qualities of mobile apps

[Background sync]

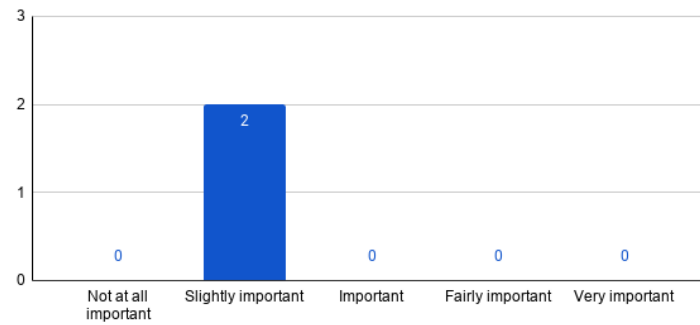


Figure 4.55: Answers for ESQ 9, part 4

Please rank the importance of the following features and qualities of mobile apps

[High performance]

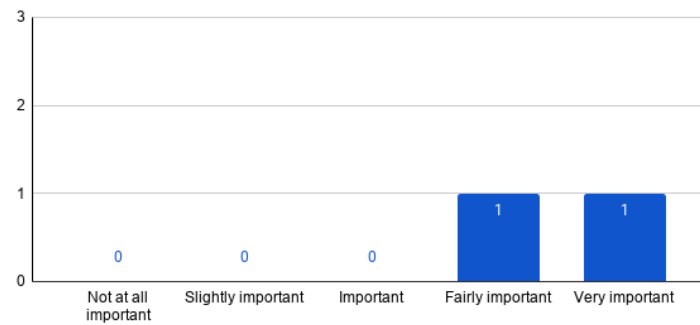


Figure 4.56: Answers for ESQ 9, part 5

Please rank the importance of the following features and qualities of mobile apps

[Low battery usage]

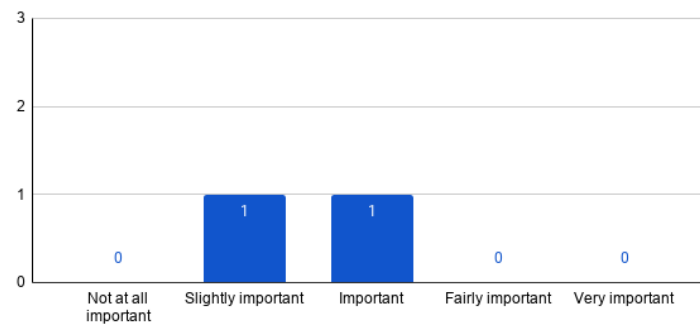


Figure 4.57: Answers for ESQ 9, part 6

Please rank the importance of the following features and qualities of mobile apps

[Usable while offline]

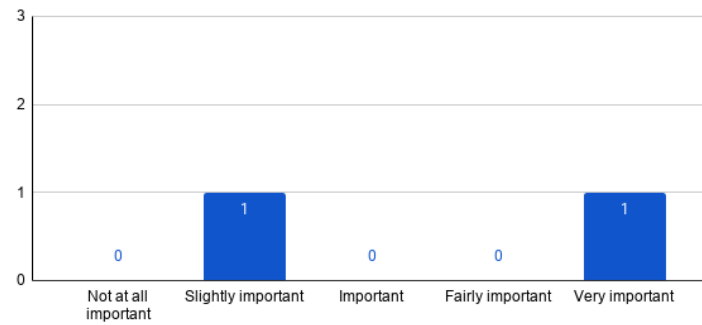


Figure 4.58: Answers for ESQ 9, part 7

Please rank the importance of the following features and qualities of mobile apps

[Cross-platform]

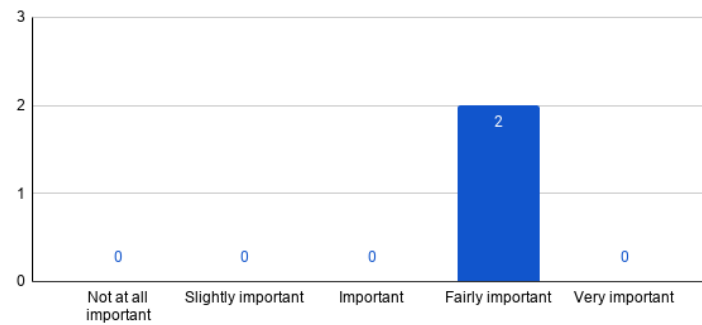


Figure 4.59: Answers for ESQ 9, part 8

Please rank the importance of the following features and qualities of mobile apps

[Discoverable from web-search]

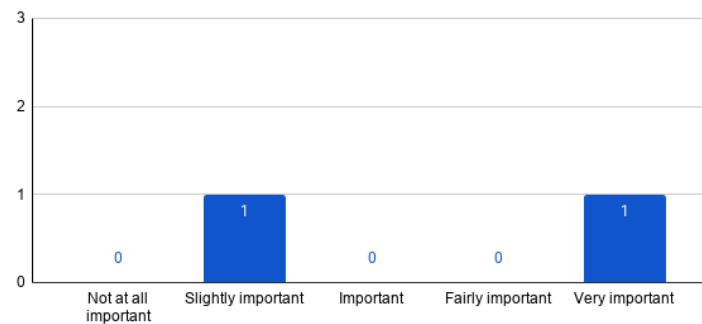


Figure 4.60: Answers for ESQ 9, part 9