

18 June 2021

Competition Commission of South Africa
77 Meintjies St
Sunnyside, Pretoria 0002
South Africa
ccsa@compcom.co.za

RE: *Comments of ACT | The App Association to the Competition Commission of South Africa “Online Intermediation Platforms Market Inquiry, Statement of Issues”*

ACT | The App Association appreciates the opportunity to provide input to the Competition Commission of South Africa (CCSA) on its draft titled ‘Online Intermediation Platforms Market Inquiry, Statement of Issues.’¹

The App Association represents more than 5,000 small business application developers and connected device companies, located both within the Republic of South Africa and across the globe. These companies drive a global app economy presently worth more than ZAR 25.21 trillion, and this economy continues to grow. App Association members leverage the connectivity of smart devices to create innovative solutions that introduce new efficiencies across consumer and enterprise use cases and rely on a predictable and fair approach to platform regulation in order to grow their businesses and create new jobs; therefore, CCSA’s inquiry into online intermediation platforms is directly relevant to us, and we urge for the careful consideration of our views.

The App Association shares CCSA’s goals of advancing competition and innovation in digital platforms. We offer the perspectives and recommendations below on the proposed scope of the CCSA’s inquiry and surrounding issues and welcome the opportunity to assist CCSA in its efforts moving forward.

¹ www.compcom.co.za/online-intermediation-platforms-market-inquiry/.

I. The Impact of Platforms on Software Distribution: What Makes an Ecosystem Work?

It is often easy to forget the journey once we arrive at the destination. We forget the bumps in the road and often overlook factors that made the trip possible. The app economy's trajectory is no different. In a little more than a decade of existence, the app ecosystem grew exponentially alongside the rise of the smartphone. Valued at ZAR 25.21 trillion, the app economy is driven by app developers and innovators who rely on software platforms to reach consumers around the globe. In 2018, the total number of app downloads was 194 billion (up from 178 billion in 2017), and the reach of software applications continues to grow.

The single most important factor in the app ecosystem's dynamic growth and unrivalled success is the presence of curated platforms, or app stores. Trusted app stores serve as a vital foundation for the growing uses of apps across industries and enterprises. Three key attributes led to the revolution in software distribution:

1. The provision of a bundle of services that reduces overhead costs;
2. Instantaneous and cost-effective consumer trust mechanisms; and
3. Cost-effective access to a global market.

Today every successful platform for mobile, desktop, gaming, and even cloud computing must provide these features or risk failing in the marketplace.

II. How Developers Distributed Software Before Platforms

Much has changed for consumers and developers since the early days of software applications. In the early 1990s, consumers were tasked with the challenge of locating and then travelling to a brick-and-mortar store that happened to sell software. Once internet connectivity became a standard feature in most private residences, consumers began to download applications from the comfort of their homes without having to step foot in a physical store. Despite the changes brought by internet connectivity, the golden age of personal computer (PC) software pales in comparison to the size and scale of the mobile app revolution during which software developers evolved into app developers. And consumers were often unable to trust software downloaded from the internet because the vetting function of platforms had not yet been introduced.

Before the ubiquity of mobile platforms, the software ecosystem ran on PCs, and software companies had to cobble together a distribution plan, including the creation of consumer trust from the ground up. This forced early app companies, often with teams of one to two developers, to wear many hats to develop, market, and benefit from the sale of their products. App companies were not only required to write code for their products, but they were also responsible for:

1. Managing their public websites;
2. Hiring third parties to handle financial transactions;
3. Employing legal teams to protect their intellectual property; and
4. Contracting with distributors to promote and secure consumer trust in their product.

The skillsets required to manage the overhead of online software distribution were often not “core competencies” of small development companies, and the additional steps cost app developers valuable time and money, with little tangible benefit.

In the internet economy, immediate consumer trust is almost impossible without a substantial online reputation, and not attaining it spells death for any app company. However, what does “trust” mean? In this context, trust refers to an established relationship between the app company and consumer where the consumer demonstrates confidence to install the app and disclose otherwise personal information to an app company. Prior to platforms, software developers often had to hand over their products to companies with a significant reputation to break through the trust barrier.

Developers in a pre-app store world experienced difficult and oppressive distributor requirements placed on software developers that predated the platform ecosystem. When dealing with retail distributors, these small businesses were required to guarantee a competitive price, pay 3-6 percent of sales as a marketing fee in addition to ZAR 160,000 for product launch marketing, shipping to deliver their products to distributors, and buying back unsold products. Once contracts were negotiated, software developers were often required to spend additional money so that in-store catalogues would feature their product or retail stores would place their product on an endcap display, all before consumers even saw the products.

However, with the advent of the smartphone and app stores, the experience of these innovative small businesses became a relic of the past. The smartphone, in its brief history, revolutionised the economy at large and established a symbiotic relationship between software platforms and developers. The fact that developers have a choice in which platform to use to reach their consumers and clients underscores that platforms compete not only as app marketplaces but as developer services providers.

III. The Applicability of Antitrust Law to Software Platforms: Two-Sided Market Analysis

The application of antitrust law to new markets is already difficult, and it is even more so in the instance of software platforms and the digital economy. Antitrust analyses are very often fact-specific, giving rise to a dynamic that has invited undefined calls for antitrust law's expanded application to software platforms, sometimes from those simply seeking to leverage antitrust claims to influence commercial disputes. CCSA should responsibly ignore these calls and recognise that the application of antitrust law should provide the flexibility to address new and challenging market definitions and determinations of market power and monopoly power. A thoughtful analysis will conclude that the harms claimed by those advocating for antitrust remedies on software distribution platforms are either absent or easily outweighed by their pro-consumer benefits.

a. Software Platforms and Market Definitions

A market definition should precede a determination of market power and abuse. While a market definition should consider antitrust foundations such as the existence of substitutes, such an analysis must be fact-specific and traditional antitrust analysis is not easily applied to platforms that very often are multi-sided markets.

Traditionally, antitrust analyses on two-sided markets (e.g., newspapers) have focused on only one side of the market because of the limited impact of network effects. Where platforms experience more indirect network effects with linked demands and pricing—such as in the case of software app distribution platforms—including both sides in the relevant antitrust market is appropriate. Mobile platform markets likely require consideration of at least three distinct markets (possibly four if one considers wireless carriers) to perform one transaction, as is the case for software platforms. But even where multi-sided platforms have demonstrable competition on both sides of a transaction, using traditional constructs such as the “small but significant non-transitory increase in price test” (SSNIP) on one side of the transaction would lead to the misapplication of antitrust law.

CCSA should provide the flexibility for case-by-case market definitions, and a full understanding of a market is required in order to appropriately apply antitrust law to multi-sided digital platforms. Novel economic and legal approaches can and should address the complexities of multi-sided platforms.

b. Software Distribution Platforms, Market Power, and Monopoly Power

Once a market has been appropriately defined, an antitrust analysis would turn to a determination of market and monopoly power. Market power and monopoly power are related concepts but are not the same. Market power is the seller's ability to raise prices above those that would be charged in a competitive market, while monopoly power occurs when a firm has the power to control prices and exclude competition. CCSA should distinguish the two concepts as a matter of degree, monopoly power being higher. However, a firm's mere possession of either market power or monopoly power is not enough for CCSA to find competitive harm; CCSA must demonstrate that the firm unfairly values its products that yield harms to consumers and competitors. Demonstration of such abuse is critical to properly determining whether antitrust remedies are appropriate, and if so, to what degree.

Platforms play an important role in tech-driven markets as well as across a variety of economic sectors, bundling sets of services together for sellers and connecting those sellers with specific categories of buyers. South African antitrust policy should reflect that market power assessments should be more holistic and rely on factors past market share alone, and that new digital platforms illustrate that the application of traditional antitrust fact patterns to complex software platforms is ill-advised. Over-reliance on basic market share (e.g., the relative size of a user base) breakdowns wrongly equates *share* with *power*, ignoring unique attributes of multi-sided platforms such as the ability to benefit from multiple services on the same platform, a low barrier to substitution, and ease of market entry by new competitors. Such characteristics minimise the lock-in effect on users. Further, a proper antitrust analysis should also demonstrate that the monopoly power at issue is not short-lived. Such a determination will, again, be highly fact-dependent and should be holistic.

IV. The Software Side of the Market

Turning to the different sides of the software platform market, the most visible side for the general public is the one characterised by software sellers (app developers) selling to software consumers (businesses and individual consumers). One of the most often-cited alleged competitive deficiencies in this side of the market is the practice of self-preferencing by platforms. Considering the unique nature of software distribution platforms, self-preferencing is in most cases pro-competitive because it is an example of vertical integration. We urge CCSA to conclude that where vertical integration or self-preferencing can lead to greater efficiency, better quality, or lower costs for consumers, there are minimal antitrust issues when users can easily switch to another platform.

Considering that smartphones are music players, cameras, and multimodal communications devices, a narrowly focused view of one of these features without recognising the integration of the same into the devices is incompatible with the way consumers experience them. Moreover, CCSA should expect competition to discipline examples where self-preferencing is bad for consumers because those consumers can leave the platform due to demonstrably low switching costs. Just like other categories of market activity, an antitrust inquiry into self-preferencing is generally only appropriate where the company at issue has market power and where it is using that market power to harm competition and consumers. Unfortunately, in other jurisdictions such as the European Union (EU), policymakers have proposed flipping the burden onto platforms to show that self-preferencing has no long-run exclusionary effects and either the absence of adverse effects on competition or an overriding efficiency rationale. The App Association discourages such an approach in South Africa because it would chill market activity that is likely to benefit consumers.

V. The Developer Services Side of the Market

Aside from the antitrust attacks on platform activity in the software half of the two-sided market, critics also allege competition abuses in the developer services side of the market. CCSA should be especially wary of populist calls for the overapplication of antitrust law to digital platform activity in this side of the market. Some are seeking to leverage this trend to use the antitrust laws to punish their competitors and tend to overstate the problems they identify. For example, advocates for antitrust intervention point to the cost of the services software platforms provide to developers as evidence that policymakers should expand antitrust law. To show that paying for developer services is unfair, they compare the cost of software distribution to the cost of payment processing. Similarly, payment processing is just one element of the array of services you get on a software platform, which include: immediate availability through hundreds of millions of people's devices; marketing through the app store; privacy features embedded in the platform; assistance with intellectual property protection; and security features built into the platform. Complaints about the costs of developer services paid to platforms are overstated because such costs are being compared to a much less substantial service and do not warrant an expansion of antitrust law or the creation of a new regulatory regime to reduce the price of developer services.

The other evidence advocates offer to show harm to competition is that making software available on the open internet is free when it is not, whereas software distribution on a platform generally costs money. As discussed above, selling software on the open internet requires the seller to take on several tasks the software platform bundles together (including marketing, intellectual property policing, privacy controls, security features, and payment processing). And even taking it at face value, the premise has the inconvenient characteristic of proving the opposite point—that is, selling software on the open internet can be a substitute for selling software on a platform. Not only that, detractors of software platforms say they have no choice but to submit to software platform demands and then openly admit that they need not submit to software platform demands because they sell their software on the open internet instead. It is hard to

imagine that this internal inconsistency goes unnoticed, and observers likely cannot help but discern from this that software sellers have options. Indeed, many other developers have made the transition off platforms without claims of anticompetitive conduct. Substitutes, even when they are not identical, are common in market economies and tend to signal healthy competition.

The other conclusion CCSA should draw from these arguments is that policymakers should be wary of opportunistic behaviour by well-resourced competitors disguised as antitrust concern. Those that are most vocal often imply they are speaking for the app economy as a whole, but in reality, they tend to be larger companies seeking to use antitrust law or other policy levers to undermine competitors. Right now, the largest software platforms generally charge the same (as a percentage of revenue) for developer services regardless of the company's size or political clout, or in some cases less for smaller developers. Smaller developers have the advantage in either of these arrangements because they do not have the leverage to negotiate better terms on their own, as larger companies do. Overtures to have CCSA or other policymakers involve themselves in developer-platform relations, therefore, may benefit the largest software companies on the platforms but may actually make small developers like App Association members worse off. If large software companies convince CCSA to require software platforms to give them a better one-off deal, App Association members and their clients and customers are forced to subsidise the resulting discount for these larger companies. Adding insult to injury, many App Association member companies compete with these larger firms, so the benefit handed to the larger companies could directly disadvantage App Association members.

Even as the antitrust concerns expressed in this area are often overstated, a competition analysis of these dynamics is not always the final say, and antitrust concerns may conflict with countervailing policy priorities. For example, policymakers have raised alarms over measures software platforms use to protect consumer privacy. In one instance, a software platform faced antitrust concerns after a decision to curtail apps' ability to track a consumer's location even when the app is not running unless the consumer clearly consents. Advocates exert a steady stream of pressure on software companies and platforms to improve their privacy practices, especially with respect to location data, often pointing to how companies collect such sensitive personal information. In reality, privacy controls at the platform level ameliorate this perceived problem by making it easier to set collection rules for all or specific apps.

Policymakers have long made it clear that companies should embed privacy into the design of their products and services. Accordingly, the purpose of a privacy prompt from the platform's operating system should not be to confuse a consumer into selecting an option that gives away more data than they intended. It follows that requiring platforms to make it easier to provide location data even when an app is not running than it is to protect that data—because doing so would help a specific app developer—runs headlong into the policy imperative of privacy by design. Looking at the issue solely from a competition lens is, therefore, an incomplete view. Moreover, the more privacy-protective approach of one software platform differentiates it competitively from other platforms that arguably make it easier for developers to collect sensitive data. In resolving these policy tangles, the focus should be on what works best for consumers. Antitrust law by itself rightfully addresses consumer welfare — it does not seek to benefit competitors. So, if a platform has an offering that a consumer prefers over the offering of an independent developer, CCSA should ask whether the complaints of powerful competitors necessitate legislating away that choice.

App Association members are selective about the markets they enter, but they compete aggressively. And the presence of a powerful and well-resourced competitor is not always enough to totally discourage entry. Having plentiful resources is an undeniable advantage as a competitor (whether it is a platform or not), but our member companies exist because they fill a niche with a differentiated product, they can compete on price, or they can simply outmanoeuvre the larger competitors. The continued existence and success of camera apps on app stores is an example of companies competing directly with a platform.

But that is not to say a company with a competing offering should never be purchased by a larger company. There are three main definitions of success for a small company: passing the company along to the next generation; being purchased by a larger company; or (much less often) an initial public offering (IPO). Being purchased is often the best of these three options for the business owner and consumers — after all, IPOs are expensive and fraught with risk. A purchase that helps produce better products or services for consumers is both a natural and beneficial end for some companies and healthy from a competition perspective.

VI. The Developer Services Market: Background

At first, developers were reluctant to join platforms, worried that the model might not accommodate their ability to launch fast and iterate their apps. But successful platforms changed the app ecosystem by providing app developers with ubiquitous access to a broader swath of consumers. Platforms provide a centralised framework for app developers to engage and secure visibility with 5 billion app users worldwide. With lower costs and barriers to entry, both fledgeling and established app developers can find success.

One of the central markets at issue is the market for developer services, where a developer pays a platform for assorted services including distribution, marketing, etc. This market also experiences vigorous competition. There is a tendency to include only a few platforms in this category of competitors, but for developers, the market is much wider. For example, game developers can choose additional platforms just for games, and enterprise developers can look to hundreds of proprietary, custom platforms or could create their own.

a. How Software Developers Established Consumer Trust Before Platforms

Before the introduction of the smartphone, software developers built consumer trust slowly and at great expense, and that trust was and remains essential for a software developer to bring a product to market. Most did not have a widely recognisable brand to endorse the software. Prior to mobile platforms, software developers often had to break through the trust barrier by handing over their products to companies with a significant reputation.

Even shareware products that could be digitally distributed would end up partnering with reputable brands to gain consumer trust. Today, consumers can download games like these for free on platforms. These platforms not only lower cost by taking care of the significant overhead involved in selling their product, but they can also reach consumers much more easily.

But the trust mechanism provided by the platforms is not merely an aspect of size. Consumer trust requires constant maintenance and vigilance because the loss of trust hurts both the platforms and the developers who rely on them. The immediate consumer trust embedded into platform brands worth billions of dollars allows developers to clear the critical hurdle of achieving trust from consumer adoption.

b. How Software Developers Dealt with Piracy Before Platforms

Before the age of platforms, software developers struggled to safeguard their intellectual property (IP) against piracy and theft. Software companies faced serious challenges in protecting their products in retail stores because the licensing codes remained active and easy to steal. Once developers overcame the significant barriers to bring their products to market, they were faced with the threat of piracy and theft which limited their volume of business and hurt their bottom line. As far back as 2006, it was estimated that, on average, software developers lost ZAR 110 million in revenue per year.

Before software developers could leverage dispute resolution mechanisms provided by platforms, developers were left with the significant burden of intellectual property infringement litigation in court, which could leave the legitimate IP owner with several thousand ZAR per month in legal fees and months or years diverted from company matters. When the infringement originated abroad, software developers were at the mercy of foreign judicial systems, some lacking rule of law and impartiality. Software developers and copyright holders continue to benefit from platforms' cost-effective avenues, such as their dispute resolution mechanisms referenced above, to distribute and protect the integrity of their products.

Despite all these platform-enabled advantages, for developers looking to reach a general audience, using the web is an alternative, especially for companies that are looking for different kinds of distribution or search services than those available on platforms. It is worth noting, however, that there are some important distinctions between software platforms—some provide a marketplace for software apps, while social media platforms or “aggregators” connect people with information and run on data. Aggregators connect people with information and other people (and generate valuable data in the process), while the app stores provide a marketplace for consumers and app developers to transact directly. These differences illustrate the diversity in the market for distribution methods, as developers may prefer one model over another.

Software platform safety and security are essential elements of developer services, particularly for enterprise app developers. Software platforms' security features have improved markedly over the course of their existence yet must continually adapt to address new vectors and threats. While unlocking a device used to require simply a four-digit passcode, devices are now capable of biometric authentication and software platforms make these authentication measures available to developers as well so that they can also offer these heightened security measures to their customers to build and maintain trust. But the game of cat-and-mouse between cybersecurity professionals and hackers will never end, and security must continue to evolve to meet and beat the threats. Although some platforms do not control device security, developers want the platform's security features to work seamlessly with any relevant hardware and that they account for all attack vectors. Software platforms should continue to improve their threat sharing and gathering capabilities to ensure they protect developers across the platform, regardless of where threats originate. Moreover, they should approve and deploy software updates with important security updates rapidly to protect consumers as well as developers and their clients and users.

VII. Signs of Competitive Health: Platforms Unlock New Markets

As successful as the past 12 years have been for the app economy, the next decade could be even better. In just the third quarter of 2019, the major app stores generated more than ZAR 329 million in revenue — a robust 23 per cent year-over-year increase from the third quarter of 2018. This growth suggests the developer-platform model is still succeeding. Moreover, app economy growth is likely to endure because developers are continuing to create new products, services, and markets that did not exist prior to platforms. A notable example of the app economy's ingenuity is in combatting the COVID-19 pandemic. Mobile apps have been effectively utilised for contact tracing notifications to assist in minimising the spread of the disease, saving countless lives.

Perhaps most importantly, the universe of platforms is continuing to evolve and expand as diverse kinds of hardware connect to the network. For example, new platforms are cropping up for wearables. Connected home devices and cars drive cross-platform interoperability so that voice-assisted capabilities can communicate with other devices — further weighing against conceptions of platform markets where a single player wields market power and indicating that developer services will continue to improve and evolve along with demand.

Another area where platforms enable developers to reach new audiences is through accessibility tools. Mobile operating systems are built with powerful accessibility tools for developers to use in creating apps that enhance the lives of the disabled. Whether it is voice directions in a mapping app for the visually impaired or text to speech tools for those with a speech-language disorder, offering these tools as part of a developer tool kit assists any app in reaching a wider audience.

In its Statement of Issues (Section 3.5), CCSA addresses featuring and ranking in app stores. App Association app developer members often are featured based on their designing of a sleek user interface and intuitive user experience, updating their app(s) regularly, optimising app localisations, making the app accessible to those with disabilities, gathering reviews, and creating an app preview. On the App Store, building an innovative app that stands out and letting the App Store editorial team know about it (through <https://developer.apple.com/contact/app-store/promote/>) is the best way to get featured. The Google Play store is more algorithm-driven (rather than editorial-driven); on Google platforms, it is more important to get discovered by users and start trending to be noticed. The app title, number of downloads, good ratings, and price are the main factors that determine search rank. Generally, platform transparency, including with respect to ranking and featuring in app stores, is important to our members and any business users to increase their ability to plan ahead and attain legal certainty for their business but is not crucial to our members' success in a platform. The App Association believes that there are different levels of transparency and notes that while more information on some levels can be beneficial (e.g., technical specifications, tools available to business users), platforms should not be obligated to disclose all their business operational details, such as their ranking-specific algorithms. Full and complete transparency would make search ranking manipulation trivial and fill the app

stores with spam. It is important to allow the platforms enough flexibility to continue to optimise their search and ranking algorithms and stay ahead of those who are trying to game the system.

VIII. What Does All of This Mean?

The extraordinary rise of the app economy happened in tandem with the development of the smartphone and software platforms. The presence of established, centralised platforms helps to drive the app ecosystem's dynamic growth and unrivalled success. Platforms serve as vital foundations and databases for the growing uses of apps across industries and enterprises. Software platforms do three things for app developers:

1. Reduce overhead costs across the board;
2. Provide instantaneous consumer trust mechanisms; and
3. Enable cost-effective access to a global market.

Today every successful platform for mobile, desktop, gaming, and even mainframe computing must provide those features, or they fail in the marketplace.

Apps serve as the driving force in both the popularity and development of the smartphone and in turn, platforms offer lower barriers to entry for software developers into markets worldwide. The two entities' successes are symbiotic, and we look forward to continued success into the next decade.

The App Association appreciates the opportunity to provide its views to CCSA and urges for careful consideration of our interests. We are committed to working with CCSA to bring the benefits of the dynamic app economy to all consumers, including those in South Africa, through the development of balanced consumer protection and anti-monopoly policies.

Sincerely,

A handwritten signature in black ink, appearing to read 'B. Scarpelli', with a stylized flourish at the end.

Brian Scarpelli
Senior Global Policy Counsel
bscarpelli@actonline.org

ACT | The App Association
1401 K St NW (Ste 501)
Washington, DC 20005
United States