**Engineering For Equity**

we’ll discuss the unique responsibilities of an engineer when designing products for a broad base of users.

Further, we evaluate how an organization, by diversity, can design systems that work for everyone, and avoid harm against our users.

We’re newer still at understanding the impact it has on underrepresented people and diverse societies.

Understanding how to engineer products that empower and respect all our users is still something Google is learning to do.

more equitable products begin with evaluating our own failures and encouraging growth.

You likely aspire to be an exceptional engineer. You want to solve problems. You aspire to build products that drive positive outcomes for the broadest base of people, including people who are the most difficult to reach.

To do this, you will need to consider how the tools you build will change the humanity, hopefully for the better.

[**Bias Is the Default**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#bias_is_the_default)**:**

When engineers do not focus on users of different nationalities, ethnicities, races, genders, ages, socioeconomic statuses, abilities, and belief systems, even the most talented staff will inadvertently fail their users.

Such failures are often unintentional; all people have certain biases, and social scientists have recognized over the past several decades that most people exhibit unconscious bias, enforcing and promulgating existing stereotypes.

Unconscious bias is insidious and often more difficult to mitigate than intentional acts of exclusion.

Even when we want to do the right thing, we might not recognize our own biases.

[**Understanding the Need for Diversity**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#understanding_the_need_for_diversity)

Focus on bringing diverse perspectives into product design and implementation.

Googlers contribute to building a more representative workforce. it is important to learn how biased outcomes happen in hiring.

There are significant prerequisites for understanding how to anticipate harm and prevent it.

To get to the point where we can build for everyone, we first must understand our representative populations.

We need to disrupt the idea that only people with computer science degrees can design and build products. this approach is insufficient for inclusive and equitable engineering.

they need to understand the population demographics of their users.

In the absence of diverse representation on engineering teams, individual engineers need to learn how to build for all users.

[**Building Multicultural Capacity**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#building_multicultural_capacity)

how products can advantage and disadvantage different groups of human beings.

Engineers are expected to have technical aptitude, but they should also have the discernment to know when to build something and when not to.

Discernment includes building the capacity to identify and reject features or products that drive adverse outcomes. This is a lofty and difficult goal. We need to extend our focus to the next billion users or to current users who might be disenfranchised or left behind by our products.

tools that influence how people think about the value of human lives, tools that monitor human activity, and tools that capture and persist sensitive data, such as images of their children and loved ones, as well as other types of sensitive data.

As an engineer, you might have more power than you realize: the power to literally change society. It’s critical that on your journey to becoming an exceptional engineer, you understand the innate responsibility needed to exercise power without causing harm.

The first step is to recognize the default state of your bias caused by many societal and educational factors.

After you recognize this, you’ll be able to consider the often-forgotten use cases or users who can benefit or be harmed by the products you build.

Focusing on underrepresented users is a clear opportunity to promote equity.

Yet most technologists do not have the requisite perspective of underrepresented groups to understand the impact of racial variance in facial recognition or to understand how applying AI can drive harmful and inaccurate results.

Currently, AI-driven facial-recognition software continues to disadvantage people of colour or ethnic minorities.

Our research is not comprehensive enough and does not include a wide enough range of different skin tones. We cannot expect the output to be valid if both the training data **&&** those creating the software represent only a small subsection of people.

In those cases, we should be willing to delay development in favour of trying to get more complete and accurate data, and a more comprehensive and inclusive product.

Data science itself is challenging for humans to evaluate, however. Even when we do have representation, a training set can still be biased and produce invalid results.

Although the software is being developed and deployed at ever-increasing rates, the independent testing is not.

**[we need much time to test than to develop and deploy]**

we need to have the integrity to slow down and ensure that our inputs contain as little bias as possible.

Therefore, shifting the focus of your industry experience to include more comprehensive, multicultural, race and gender studies education is not only your responsibility, but also the responsibility of your employer.

requirement is not that one individual take it upon themselves to learn about other cultures or other demographics alone. change requires development that builds capacity to understand the diverse experiences throughout humanity.

[**Making Diversity Actionable**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#making_diversity_actionable)

equity & fairness are attainable if we accept that we are all accountable for the systemic discrimination.

We are accountable for the failures in the system.

It is also irresponsible to fully attribute dynamics to the larger societal issues that contribute to inequity.

“We are working hard to fix {}, but accountability is hard. How do we combat {} of historical discrimination?”

This is away from focused efforts to improve work conditions or outcomes.

focus on equity and fairness. Every technology lead / manager can augment equity on their teams.

It is important that we acknowledge that, it is our problem to fix.

[**Reject Singular Approaches**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#reject_singular_approaches)

We can’t keep solutions that present a single philosophy or methodology for fixing inequity in the technology sector.

**[there’s no one-size-fits-all solution]**

lack of representation in the workforce can be addressed by fixing the hiring pipelines.

We need to recognize inequity in progression while focusing on more representative hiring & educational disparities across lines of race, gender, and socioeconomic and immigration status, for example.

Fully understanding of a problem is critical to determining how to fix it.

This holds true from a critical data migration to the hiring of a representative workforce.

Consider whether your recruiters are demonstrating the ability to identify strong candidates who are women as well as men.

If you manage a diverse engineering team, focus on psychological safety and invest in increasing multicultural capacity on the team so that new team members feel welcome.

Instead, by building in inclusive design from the start & make tools delightful and accessible for people who struggle to access technology, we enhance the experience for all users.

Designing for the user who is least like you is not just wise, **it’s a best practice**.

There are pragmatic and immediate next steps that avoid disadvantaging or underrepresenting users.

It begins with more comprehensive user-experience research.

groups that are multilingual and multicultural and span multiple countries, socioeconomic class, abilities, and age ranges.

Focus on the most difficult or least represented use case first.

[**Challenge Established Processes**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#challenge_established_processes)

Challenging yourself to build more equitable systems goes beyond designing more inclusive product.

Building equitable systems means challenging established processes that drive invalid results.

At Google, teams worked to build a global hiring requisition system. The system supports both external hiring and internal mobility.

To drive efficiency, the recruiters asked the engineering team to include a feature that would highlight performance ratings—specifically lower ratings—to the hiring manager and recruiter as soon as an internal transfer expressed interest in a job.

On its face, expediting the evaluation process and helping jobseekers save time is a great goal. So where is the potential equity concern? The following equity questions were raised:

*Are developmental assessments a predictive measure of performance?*

*Are the performance assessments being presented to prospective managers free of individual bias?*

*Are performance assessment scores standardized across organizations?*

If the answer to any of these questions is “no,” presenting performance ratings could still drive inequitable, and therefore invalid, results.

candidates who received a poor performance rating were likely to overcome the poor rating if they found a new team. they were just as likely to receive a satisfactory or exemplary performance rating as candidates who had never received a poor rating.

performance ratings are indicative only of how a person is performing in their given role at the time they are being evaluated.

Ratings, an important way to measure performance during a specific period, are not predictive of future performance and should not be used to gauge readiness for a future role or qualify an internal candidate for a different team.

[**Values Versus Outcomes**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#values_versus_outcomes)

we continually evaluate our processes to improve equity and inclusion.

More broadly, our core values are based on respect and a commitment to a diverse and inclusive workforce.

The struggle to improve our equitable outcomes persists despite the policies and programs in place to help support inclusion initiatives and promote excellence in hiring and progression.

The failure point is not in (values, intentions, or investments) in the application of those policies at the implementation level.

Old habits are hard to break. The users you might be used to designing for today—the ones you are used to getting feedback from—might not be representative of all the users you need to reach.

So, what’s the way out?

**Take a hard look in the mirror.** At Google, we have the brand slogan, “**Build for Everyone**.” How can we build for everyone when we do not have a representative workforce or engagement model that centralizes community feedback first?

We failed to protect our most vulnerable users from racist, antisemitic, and homophobic content.

**Don’t build for everyone. Build with everyone.** We are not building for everyone yet. That work does not happen in a vacuum, we need to put the most vulnerable communities at the centre of our design.

They should not be an afterthought.

**Design for the user who will have the most difficulty using your product.** Building for those with additional challenges will make the product better for everyone. Another way of thinking about this is:

don’t trade equity for short-term velocity.

**Don’t assume equity, measure equity throughout your systems.** Recognize that decision makers are also subject to bias and might be undereducated about the causes of inequity.

You might not have the expertise to identify or measure the scope of an equity issue.

Catering to a single userbase might mean disenfranchising another; these trade-offs can be difficult to spot and impossible to reverse.

**Change is possible.** The problems today, from surveillance to disinformation to online harassment, are verwhelming.

We can’t solve these with the failed approaches of the past or with just the skills we already have. We need to change.

[**Stay Curious, Push Forward**](https://learning.oreilly.com/library/view/software-engineering-at/9781492082781/ch04.html#stay_curiouscomma_push_forward)

The path to equity is long and complex. However, we can and should transition from simply building tools and services to growing our understanding of how the products we engineer impact humanity.

Challenging our education, influencing our teams and managers, and doing more comprehensive user research are all ways to make progress.

Although change is uncomfortable and the path to high performance can be painful, it is possible through collaboration and creativity.

Lastly, as future exceptional engineers, we should focus first on the users most impacted by bias and discrimination.