Facebook University Internship Application 2017

**How to Apply**

1. Upload a PDF copy of your Resume
2. Upload a PDF copy (unofficial is ok) of your High School Transcripts
3. Upload a PDF copy (unofficial is ok) of your College Transcripts
4. Upload a PDF copy of your response to the Essay Question

**"How should software engineers think about the ethical impact of their work?"**

**-**not always up to you to fix (hard to see big picture with small pieces)

-when creating apps, or hopefully know ultimate goal

-not something that breaks the law (opening iPhone)

-if you see something that’s clearly unethical don’t do it obviously

-it is complicated (not everyone will agree)

-think about possible consequences either way (and think about users) (empathy is important)

-implications, how used

-don’t do things just because you can  
--get the least amount of information that you can

-Facebook clarifies privacy settings now

-there is no black and white

-start thinking about ethics in design (brainstorming)

-learn critical thinking

-do you only need to ethically consider what people can do with your programs?

--not exploiting information you have (sensitive)

--AI

-gray areas (ads, data mining, data to customize)

-allegory/thought experiment

Once upon a time in a parallel world people did not consider ethics in their inventions. People innovated for the sake of innovation without any attention for consequences. This world constantly marveled in the new inventions, and was in awe of the human capacity to create. However, this soon spiraled out of control, and the inventions caused problems. People abused them to take advantage of other people, to do harm in the world. On top of this, the inventions began to create issues on their own. They began to make their own decisions, which grew to be unethical despite their creators’ initial intent. The world grew to chaos until a revolution occurred, the outcome of which would be totalitarian control, with no civil liberties, or destruction of all technology and new inventions, and life with the least amount of technology as the people could remember. Either way, the world would suffer.

Ethics is not black and white. There are many possible outcomes of actions. However, regardless, it is important for people, especially creators such as software engineers, to consider these possibilities before doing something. For example, for software engineers, when a product is in design, before production has even begun, it is important to think about what consequences could result either way. Whether from creating it, or creating it in a certain way, or even from not creating it. Then attempt to weigh these issues and consider how they may affect the users. Think about whether certain questionable aspects are necessary or helpful at all for the users and the purpose. Although these decisions may not always work out, it is vital that software developers consider them and make the best decisions they can, using their skills of logic and empathy, before proceeding. Then, even if a mistake has been made, continuing to consider these issues with any new developments that occur, or once a product has been released can continue to help create ethical products. While there are many **grey** areas that make this difficult, cutting out obvious unethical practices or software, and continuing to use critical thinking to pay attention to what one is doing will already make a big step towards avoiding the unethical world outlined above.

This advice applies even to those who do not get to participate in design meetings and make decisions. Innovation can often take large teams of people to create and discover. Even someone who receives a small task to undertake from the larger whole can and should consider ethics. First, it is ideal that this person would know the goal of the larger product that their piece would fit into, and could consider its implications before beginning their assignment, and then take appropriate actions. However, even if the software engineer does not know what exactly their work will be used for, if for example they were developing part of a package or a language, they can consider ethics on this smaller scale piece. They may not be able to decide whether the larger product is ethical, but they should still think about what could be done with the piece that they have coded. If it has capabilities to enable crime or sketchy actions, the developer should probably look into changing it or making it safe so that code can only be used for its original purpose and not be reused in bad ways.

It can be a difficult thing to effectively consider the ethics of what one is doing. Even the most well meaning creations can be abused. Even so, I maintain that as the creators, software engineers, we can do our best to think of our users, and do what will be best for them, and the community as a whole, instead of innovating blindly. We can strive for innovation that will further us as a community and bring us together, (like Facebook), instead of driving wedges to pull us apart.

public boolean decide(boolean ethical){

if(ethical){

do();

return true;

}

else{

return false;

}

}

Oh if only it were that simple. Understanding the ethical implications of a program is complicated. There are always many aspects of a problem to consider. It is not always clear whether something is helpful or harmful. While ethics is not black and white, it is possible for software engineers to take steps to make ethical decisions.

This process begins with design. Before software is even created its ethics should be considered. Think about possible effects of the software, how it could be abused and how it could be beneficial. For example, you might write a program to get into your phone if you forget your password, but other people might use this same software to break into other people’s phones. You might then decide that it would be unethical to publish this software. After this conclusion, software engineers may do their best to curb some of these negative effects by creating protective measures in their program. They should weigh the real life benefits of this program for their users with its capacity to harm and then make the best judgment call that they can. They should use as much of their empathy, logic and critical thinking as they can in their analysis.

However, even if a programmer is not a part of this initial design, they can and should still practice ethics. When they receive their task, they can consider how it will be used in the larger framework of the software. They can do the same process of cost-benefit analysis for their small piece as for the larger program as a whole. They can consider how what they code may possibly **harm individuals, the community, the environment, etc.**

Another important consideration is how this **individual piece of code could be used in other contexts. While a software engineer may decide that the intent of this code in this program in ethical, in other contexts it may be harmful.** If someone else were to take this code and use it for something, could they use it to harm people? If so, the engineer should consider how to change it or take precautions against its leaking, even if it is ethical in its intended context. This could be an especially pertinent problem if the software engineer created this code as part of a new coding language, package or other open source materials.

The ethical impact of anything is often hard to judge. Hindsight is 20-20 after all, and it can be hard to predict exactly what can come of any given action or creation. However, this means it is doubly important to consider the implications of what we do. Some things can be easily prevented because they can cause obvious problems if people stop to consider the ethics. Other things are more difficult to predict, but they can only be prevented if those with the power to create, the software engineers, take the time to think critically and carefully about how they can affect other people. If we use our empathy for our fellow humans to try to create things that will only help and not be opportunity for harm, we can really create great innovations that can bring us closer together.