Lecture 1 - Legal Perspectives on the Software Industry in a Surveillance Economy

**Key point 1: The legal system**

Legal systems are **complex**, being **contextual to time and place**, and are made up of **many features**. Firstly, cases are broken into **criminal cases**, where you are prosecuted by the state for a specific offense, versus **civil**, where the state facilitates the case, but it is between individuals, companies, regulatory bodies etc. A **jurisdiction** gives you the current laws for the current place and time, but sometimes even these can be superseded by **contracts**, which are deals struck between parties. It was noted however that non-unilateral contracts (one parts has all the power) can sometimes been overthrown by court. The **difference between laws and codes** was also highlighted via the example of the credit act 1984, which was a law that prevented a bank from being able to charge interest unless they’re acting lawfully. The banks lobbied to change this to a **code**, because a code is **not enforceable by law**.

**Key point 2: Rule of law**

There are **many protections placed on the law** so that it **does not become biased** towards a particular party. Firstly, **no one is above the law** and must be accountable to it. Secondly, the 3 **wings of the government must be separate to prevent collusion**, especially the legislature (who make the laws) and the executive government (who use the laws), otherwise they can just enact all laws in their favor and abolish all others. **Natural justice must also be upheld**, which ensures the accused are fully informed of and gives them the right to have their say, which prevents unfair enforcement. App stores were cited as an exampled of natural justice failing in tech, as apps are often taken down without a clear message as to why, and there’s nothing the developer can do to remedy it. Other measures include **the courts decision being binding** (must do as they say), **statutes** being **interpreted** **according** **to** **known principles** (need of lawyers), and **all interests** being genuinely **considered**.

**Key point 3: Professional liability and software engineering**

Professions must actually be **recognized as so in the eyes of the law**. To be a professional, you then must be a **registered member** of a profession. These professional groups have a bunch of **qualities** like **self-regulation**, where members regulate one another to protect the reputation of the field, possibly banning those who are seen as incompetent, and **standards**, which govern what methods are acceptable for use and what outcomes are acceptable. **In mature fields** like architecture, these **standards are well defined** as we’ve known how to build buildings that are structurally integral for thousands of years. The field of **software engineering is not so mature** however, and so the standards upon which methods to use in which scenarios (agile vs waterfall etc.) aren’t yet known, and as a result we **cannot yet be classified as professionals**.

**Key point 4: Concept of proportionality – OzTrack, Snowden, COVIDSafe app**

This was one of the major points discussed during the second part of the talk, where the focus was on dataveillance and privacy. It is a **principle that argues the benefits gained when violating the privacy of others are only justified if they are well proportionate enough**. The lecturer mentioned **OzTrack**, the financial tracking body in Australia, and how they are **now monitoring all transactions**, not just largescale ones. The **benefit is catching organized crime**, and considering they **encrypt data until it passes a threshold of suspicion** (which lowers risk), it’s reasonable to argue this is a **well proportionate benefit**. Conversely, the **NSA’s data retention and analysis programs** that Snowden whistle blew, which **also aimed to identify criminal activity**, were shown to be **severely ineffective**, despite **collecting data on not just Americans**. There is **no case for well proportionality** here, and it was **shut down**. The argument for the COVIDsafe app was similar, except it highlighted a few additional points, the main being **whether it is okay to lie to increase the effectiveness of a measure**, which was shown to be **ethically wrong** because it **breaks the gold standard of informed consent**.

Lecture 2 - Intellectual Property and Software Patents

**Key point 1: IP overview**

Intellectual property (IP) refers to a bunch of different aspects that are concerned with **legally protecting the right to the product of your mind**, i.e. the ownership of your own ideas, and are **an acknowledgement that mental effort is required to formulate an original idea**. These different aspects include **patents**, which are used to protect functionality via a set of specific claims and can last up to 20 years, **designs**, which are used to protect appearance and can last 10-25 years, **trademarks**, which protect brands’ identifiers indefinitely e.g. apple logo, **copyright**, which protects the expression of an idea automatically on materialization until 70 years after death, and **trade** **secrets**, which protect confidential information via confidences like NDA’s or other such obligations for as long as possible. Not only do they differ in what they protect and their duration, but they’re acquired and enforced differently.

**Key point 2: A closer look at patents**

Patents are **an agreement between the inventor and the government** that **prevent others from exploiting their invention** and so essentially **capture innovation as a property right**. Therefore, to be eligible for a patent, you **must show that your idea is new** (hasn’t been done before), **has utility** (is useful), and **is non-obvious** (intellectual, not trivial) i.e. it’s innovative. Furthermore, you **must fully disclose the invention** so that the patent alone could be used as a manual for replication. Patents have many **benefits**, including encouraging research and development, encouraging disclosure of inventions, attacking via royalties and cease and desists, defending via deterrence as a result from uncertain enforcement, and attracting investments/raising valuations. They are fairly **expensive** however, ranging from **$20-30 thousand** and so it’s often necessary to **prove your market** before obtaining a patent or **utilizing trade secrets instead**. Furthermore, specific relationships like employee-employer can affect who actually owns the patent, and licenses effect who has rights to use.

**Key point 3: Patent validity and infringement – chairs hypothetical**

This part of the lecture aimed to explore patent validity and infringement via following the process of getting a patent for a new type of chair. Chairs were assumed to only have 3 legs to begin with, and so the first suggested patent was for a chair with at least 4 legs that are secured to the seat such that it is substantially horizontal. This **generic wording** prevents patents being avoided on technicalities e.g. screwing vs nailing legs in. The patent office found a chair with 4 diagonal legs already exists and so the legs were specified to be vertical in the patent. This change may have been rejected in reality for being too obvious, but it’s assumed to have been granted. The important takeaway is that **our patent cannot be used to claim the diagonal legged chair, because if it could, then our patent shouldn’t have been granted because it already existed**. It was also discussed that If your patent and a competitors mutually barred one another, you could either agree to not sue one another, one buy the other out, or join forces to monopolize the chair industry.

**Key point 4: Software and business method patents**

Not only do patent requests have to be innovative, they must also be in an **eligible subject area**. Basically, there are **certain fields that patents will not be issued in, ever**, regardless of how innovative you are. These fields include a lot of biological areas, but business methods and computer implemented inventions are also excluded from patenting. Computer implemented inventions refers to using computer programs to provide a business process e.g. eBay. Programs must have a technical advantage, be it, processing speed, power consumption, improved communications or security, if they are to be patented. Examples of these include RSA key algorithms, Wi-Fi, MP3 and PageRank.