STUDENT

0027-ERR

TENTAMEN

TIA315 Technology

Kurskod	TIA315
Bedömningsform	--
Starttid	01.10.2025 08:00
Sluttid	01.10.2025 10:00
Bedömningsfrist	--
PDF skapad	17.10.2025 13:22
Skapad av	Golsa Nouri Hosseini

ⁱ HT25: Welcome to examination!

This examination consists of open questions, sometimes referred to as 'essay questions'. It includes **7 questions**, each rewarded with a maximum of 3 points. The examination is limited to **2 hours.** To pass it you need **13 points** (60%)

The grading will focus on the overall capability to explain and apply ideas. It will assess whether (1) answers are logically coherent, whether (2) they are grounded in readings, lectures, and discussions, and, when applicable, that (3) answers engage the key concepts of the course.

Advice:

Please, read the questions carefully! Each question normally consists of multiple requests, such as: "**Define**... and **explain**... **illustrate** ... with an example..."

If you feel unsure about how to respond or don't remember exactly what papers or lectures said, trust your instincts and stay focused on the question. Explain in your own words, to the best of your abilities. Do _not_ start to broadly talk about related things, to show you know other things.

It could be worth noting that longer answers are not necessarily better answers, since long texts increase the risk of ending up with inconsistencies.

1 HT25, Question 1

Delegation and Prescription. Your colleague, Linda, argues that a digital artifact, like a password management app, only performs actions because humans program it. In that vein, she argues, it has no agency.

Latour introduces the concepts of *delegation* – the transferring of human action to non-human artifacts – and *prescription* – the capacity to impose behaviors back on human users. Use these concepts to explain how a digital password management app exercises material agency. Provide specific examples of:

1. what gets delegated to the app, and
2. how it prescribes behaviors back to users.

Skriv in ditt svar här

Agency is the capacity to act. ANT proposes general symmetry where both artificats (in this case the app) and humans (the user) both have the capacity to act and should be seen as equal. Password management app exercises material agency on the user because without the password management app the user is not able to access certain places / items, there is not specifically digital meant so this also could be physical, it is used as a tool.The knowledge (passwords) of the user has been delegated to the app and this makes the app an operand resource, where the user has to act upon the app to make certain actions possible. It will prescribe back at the user what password the user should use as input to a certain programme or physical matter and if the user wants to unlock this programme or physical matter it should listen to the prescription. The app also could prescribe behaviour like 'this is not the correct fingerprint to unlock the app' to prescribe that the user needs to use a different finger to unlock a certain password.

Ord: 184

Besvarad.

HT25, Question 2

Affordances. Maria is working on her thesis about digital design. Studying the literature, she learns that affordances emphasize meanings over specific, inherent attributes of artifacts. While a focus on what users can do with artifacts seems attractive, she becomes frustrated reading that affordances are defined in *relation* to an observer or user. If an affordance is something relative, it seems pointless to her. How can such a subjective concept be useful for design?

What is Maria overlooking about affordances? Address specifically:

1. Why affordances, despite being relational, can still describe and explain general aspects of material artifacts, and
2. How this relational quality actually makes affordances useful for understanding and designing digital artifacts.

Skriv in ditt svar här

Maria is overlooking that affordances are defined in a relation to an user to an object that allows the user to perform a certain action. The user does not have to perform the action or what is allowed. If we take for example a guitar, the guitar can just be seen as a physical object but the acoustic resonance wood is allowing us to perform on the guitar, it makes us understand and describe general aspects of material artifacts that could be overlooked just by looking at the physical object. The relational quality actually makes affordances useful for designing because if we take this exam for example, the designers of this software were thinking: How can we make the students not press the submit button early, well they grayed it out! This is indicating you did not complete all the tasks yet, and even though you can click on it (affordance), it will most likely provide an error message. Looking at what a button for in this example could do 'affords', could understand the designing of digital artifacts in many ways.

Ord: 181

Besvarad.

Digital Materiality. David sees no fundamental differences between physical products, like a car, and digital products, like a smartphone. To him, both are manufactured products. Explain to David why digital artifacts are inherently **unbounded**, **open-ended**, and **incomplete**, in ways that physical products are not. Specifically:

1. Explain what each of these three characteristics means in the context of digital artifacts, and
2. Resonate on the properties of digital technology that enable these characteristics. Use a concrete example, if helpful.

Skriv in ditt svar här

These characteristics unbounded, open-ended and incomplete, obviously are key characteristics for emergent processes. I will explain the following concepts:

Unbounded: limitless, this could be in the coding, think about it like like a 'digital' universe, it keeps expanding.

Open-ended: There is no end goal, it is all due uncoordinated autonomous individuals / communities that bring new projects (think about github), and for example new programs (while using github as an example).

Incomplete: It is not finished! There could be room for improving the code for example, or building a completely different code.

An example could be github, here the users can share certain applications, code, programs, etc. There is no specific end-goal, the outcomes are not planned and not foreseeable, it is a flexible system rather than a rigid system that allows emergent processes. In github the code is reproducible and reprogrammable, this makes the system very flexible, allowing diversity (different programmes), malleability (updating certain programmes), scalability (it is possible to share and work with multiple people) and transparency (everyone can look into pull/push request). It is unbounded, everyone can do/post whatever code they have. It is open-ended, everyone can use the programme without a specific goal, purpose. It is incomplete, people can build new stuff all the time on Github. Making it a perfect example for this case.

Ord: 220

Besvarad.

Defining features of generative technology. During a product development meeting, Sarah argues that their company's new photo editing software is highly generative because it has many advanced features and can perform complex tasks efficiently. Her colleague responds that having sophisticated functionality doesn't necessarily make a system generative.

Using Zittrain's framework, address specifically:

1. What are the four characteristics that define generative technology, and
2. Apply each of these four characteristics to explain to Sarah what makes a product generative.

Skriv in ditt svar här

The four characteristics that define generative technology are:

Leverage: If the programme saves effort, if the technology saves a lot of effort developers for example could build more due to the fact it saved effort, causing more generativity to be created.

Accessability: Is it possible to access it, and the information necessary to use the technology. This will allow more people being able to use the technology, thus creating more generativity.

Adaptability: If the technology can be used for different system / broader domains, this will allow more people to work with the certain technology making the technology have a broader purpose, thus creating more generativity .

Ease of masteyr: How easy is it to learn the technology, a lower barrier to entry will create more generativity.

Example specifically for Sarah's photo editing software:

Leverage: The programme should save effort, well this is the case because editing pixels with google paint will take a lot of effort.

Accessability: Is the editing software free to use, freemium?, accessable from any country and is there avaible knowledge on the advanced feature, and is this knowledge easily retrievable?

Adaptability: Can we use it for editing frames in video's as well, can we apply it to a video editing software, this would make the programme adaptable and have multiple purposes.

Ease of mastery: Is the programme easy to use, there are many advanced features, so this software could have quite a big learning curve.

This would explain to Sarah what makes a product generative and even an example specifically for Sarah.

Ord: 257

Besvarad.

The Paradox of Control and Generativity. Liam believes that creative freedom and openness are the primary drivers of innovation. To him, any form of control will inevitably hamper creativity. He is therefore confused by the academic literature on *generative governance*, suggesting that control can actually reinforce (strengthen) generativity.

Explain this apparent paradox to Liam. Address specifically:

1. How can control mechanisms increase generativity and innovation in sociotechnical systems, and
2. What distinguishes generative control from traditional, restrictive control. Use a concrete example, if helpful.

Skriv in ditt svar här

Control and openess often have to be balanced. Too much control would often lead in less generativity, due to rules/ guidelines that restrict the user. Openess would give the users much freedom but makes the process vulnarable for malicious behaviour.

It could be hard to believe that control could actually reinforce generativity since it may seem like it is stopping users from expressing / building. But if we look at it from a different perspective, control also could guide the user into building / expressing something freely. And in addition it prevents malicious behaviour that would usually lead into generativity being lost/ abandoned. Resulting in an increase of generativity and innovation in sociotechnical systems. Think for example about a minecraft server, there are servers with no rules, these servers do not specifically have more generativity, arguably even less, because there are people destroying the work of others. While in servers where there are rules this is not the case, leading to more generativity, because the buildings / things stay intact. And destroying other people's stuff could not result in hidden processes / stuff in the game. Generative control is often being done automatically, when you try to destroy a block of someone in the server, the block will automatically placed back into the spot it was by an algorithm, and you may receive a warning sign in your screen. While, on the other hand, traditional restrictive control would be in this case, an admin/person telling you not to destroy the block and punishing manually.

Ord: 254

Besvarad.

HT25, Question 6

Resourcing and Securing. The company HomeEase has developed a new platform for the smart home. It aims to establish an open ecosystem, allowing third-party developers to create innovative applications. As the platform owner, HomeEase faces a key challenge: how to develop and deploy platform boundary resources (PBRs) that can both attract third-party developers and preserve platform control.

In the context of platform governance:

1. Explain what *resourcing* is and describe a scenario where HomeEase might engage in resourcing by designing new platform boundary resources.
2. Explain what *securing* is and describe a scenario where HomeEase might engage in securing through platform boundary resource (re)design.

Skriv in ditt svar här

Resourcing is enriching the developers/users with certain tools/ stuff that helps the developers to build.

Securing is as the word says securing the given environment of malicious behaviour, this could be done by restricting certain functions/ behaviours/ etc.

For HomeEase might engage resourcing by designing new PBRs (platofrm boundary resources) like a new API (application programming interface) this will provide the developers with a set of toold in the API that they can use to adjust certain functions/ code, or even make new stuff (generative) that could be beneficial for HomeEase as an app / platform, by integrating the newly build stuff by the developers / users of the API.

HomeEase might engage in securing by restricting certain features in the API. The department of HomeEase could have put restrictions on for example allowing third-party phones to the same smart home. This would prevent malicious developers into creating backdoors by allowing multiple people controlling the same smart home app which is very possible.

Ord: 163

Besvarad.

HT25, Question 7

Ontological Reversal. Explain the basic idea behind ***ontological reversal*** and reflect on the consequences of ontological reversal for the design of digital systems.

Skriv in ditt svar här

The basic idea of ontological reversal is that the digital product / entity becomes more 'real' than the physical matter. The the airplane ticket for example, it is just a retrieved entry out of e database and the digital pass could be seen as more real, since this is the direct entry out of the database, compared to the physical paper. The consequences of ontological reversal are crucial for the design of digital systems. The world is becoming more and more digital, and more things like an airplane ticket do have no physical resemblance. This causes designers to be careful of the things that they build. For example, VR (virtual reality), the idea is that you put on the VR set and that you feel like you just have been spawned into a new world. The design of this digital system must feel as close as possible to the real world to enhance the user experience. The design is very carefully done to make sure it is either a completely different world or quite similar to the same world, so people will feel the 'reality'. As the word already says it feels like reality. This could for some users in some programmes feel as more real than the real world, an example would be driving schools using VR and a simulator to prepare rookies for their first lessons in a car by providing examples that feel real, and the rookie has to act upon.

Ord: 244

Besvarad.