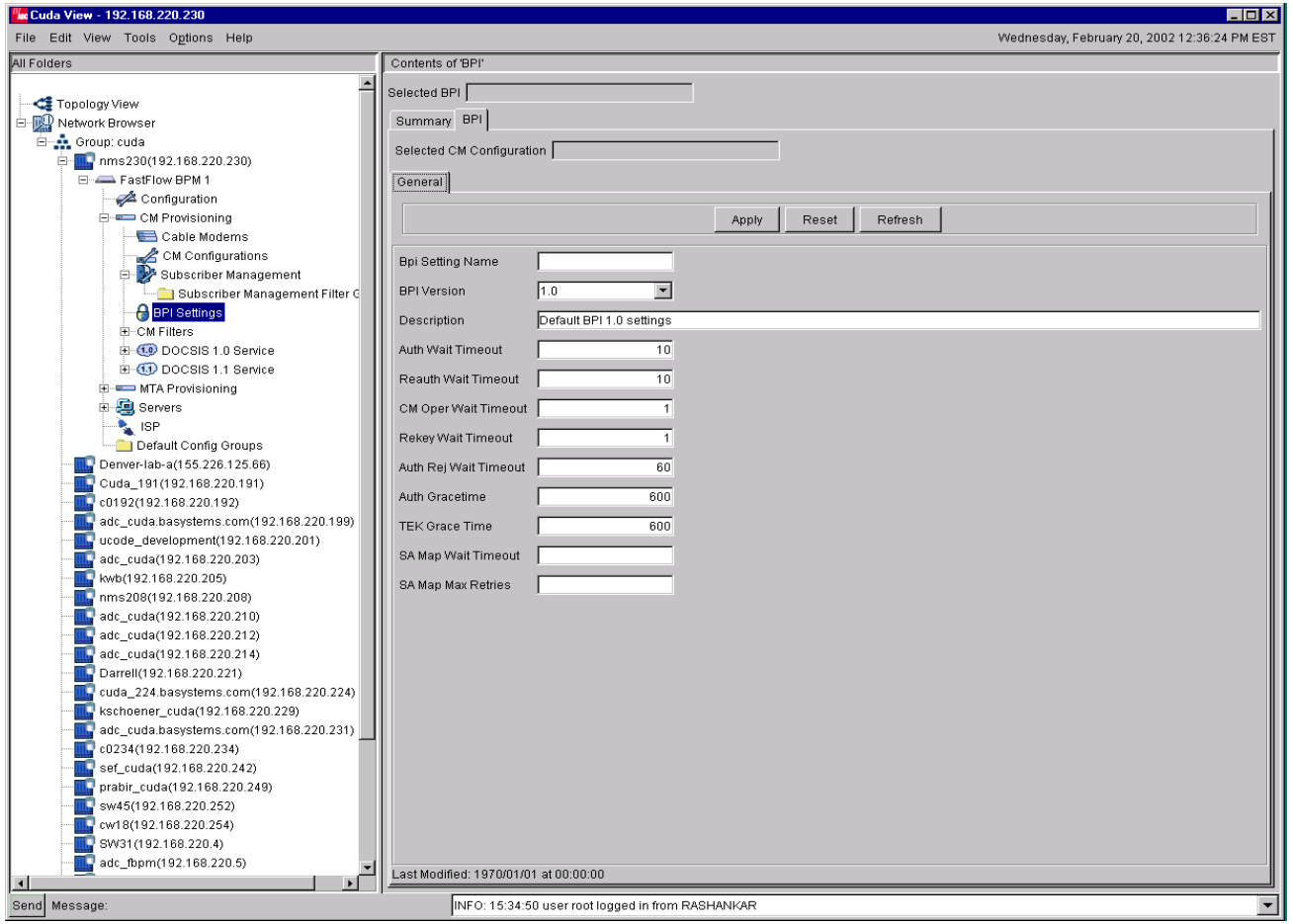
ROBERT GORDON UNIVERSITY ABERDEEN

Week 3: PACT, HTA & Conceptual Designs

What's wrong here?

Developer Centred Interface



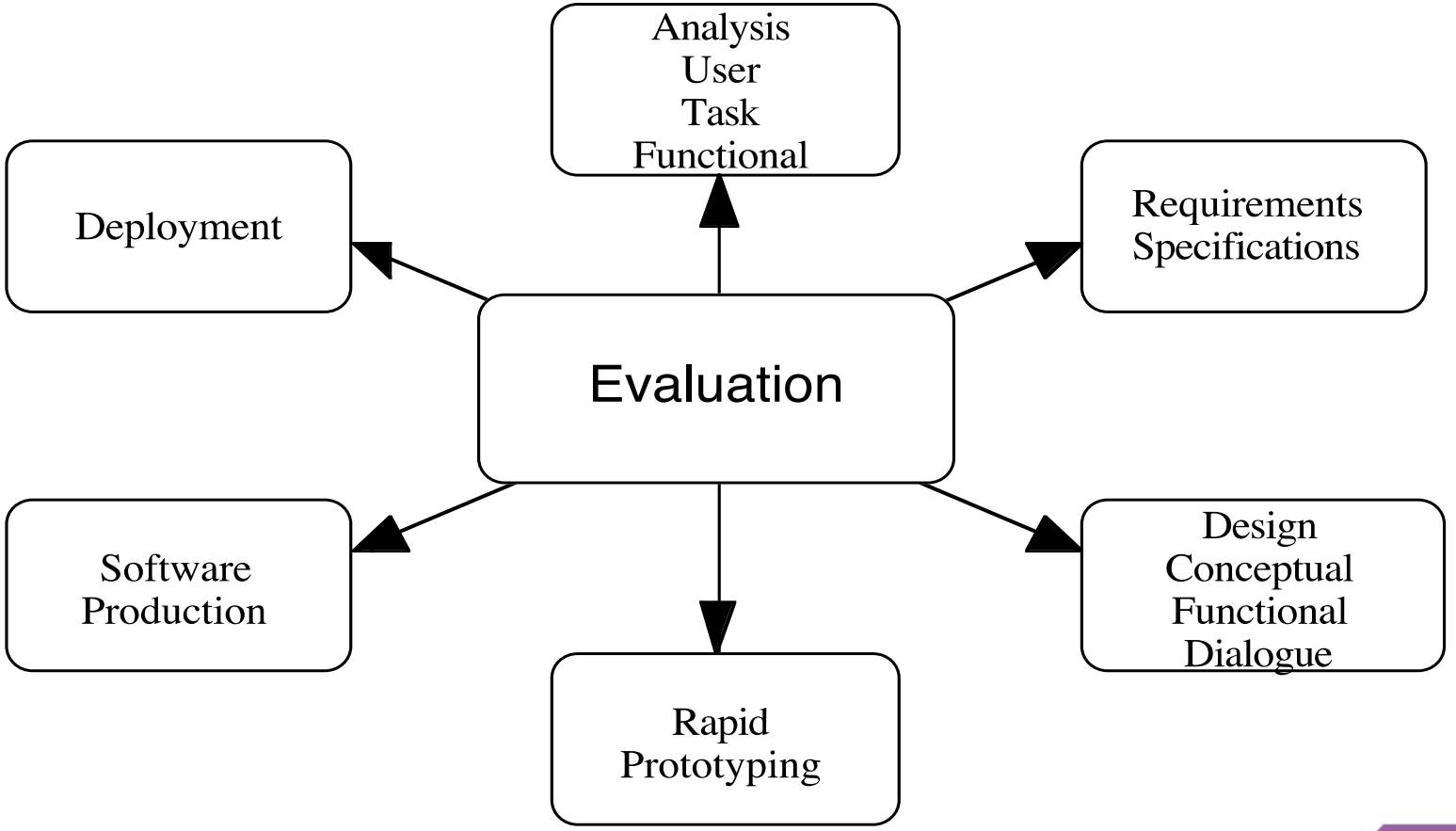
What makes this design more developer centric then user centric design?

- Organization of the information

- Use of technical jargon

- Similarity to other interfaces

Hix & Hartson Star Model



Why consider it?

- User Interface design is still a craft!
  - Our ability to predict human behaviour during interaction with a given UI is limited
    - Cognitive evaluations
  - Users' needs extend over several domains including personal, professional, organisational
  - Users' needs change from situation to situation and time to time
  - Our understanding of the processes underlying HCI are limited

- We need users' input and **feedback**!

Benefits of Involving the User

- User inform designers about their jobs
  - What is involved in their jobs?
  - What tools they use?
  - How these tools are used?
- Users help developers identify what could be useful
  - What is missing in current application or tool?
- Users try prototype and comment on it
  - Developers make incremental changes and iterate!

Factors to Consider

Organizational Factors
Training, job design, politics

Environmental Factors
Noise, lighting

Health & Safety Factors
Stress

The User
Cognitive process / capabilities

Comfort Factors
Equipment layout

User Interface technologies
Input devices, output displays, use of color, etc.

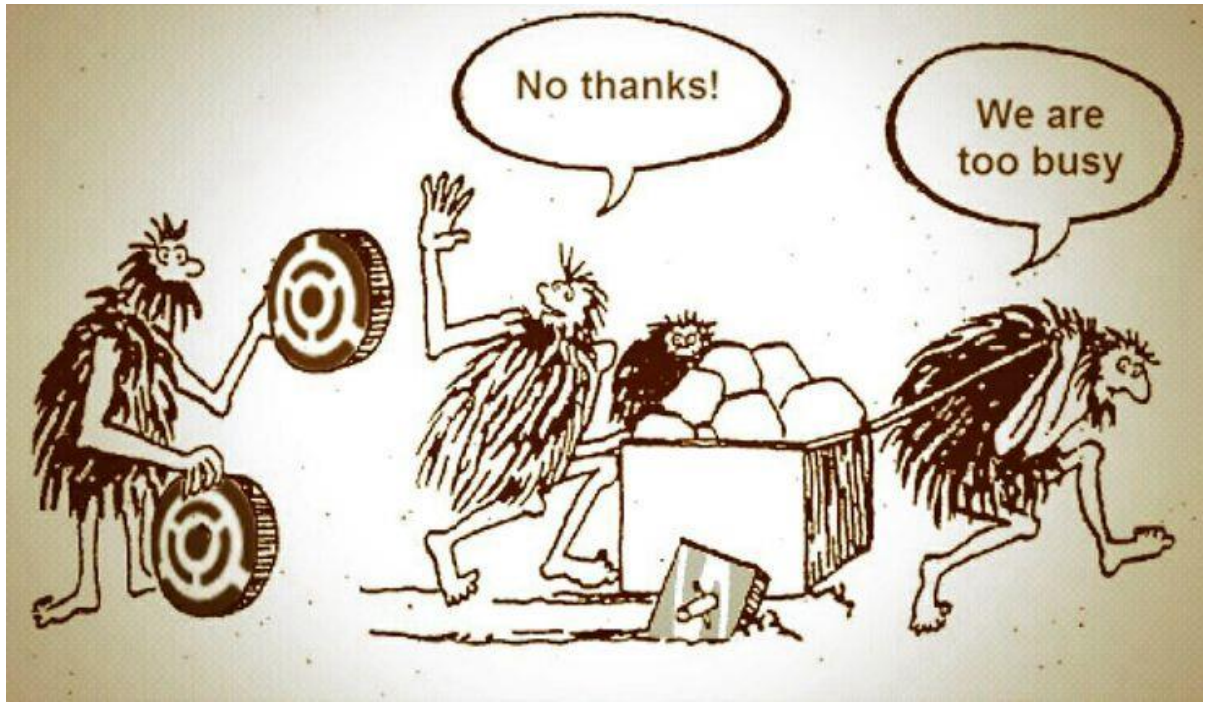
Task Factors
Easy, complex, novel

HCI at its best?

- Developers working with target users
  - Help define what the system will do and how
  - Lots of iterative exploration and feedback
- Think of the world from users' perspective
  - Users and customers are not the same person
- Understand work process
  - Points where human and computers interact

Prototypes

Invention & Realism



PACT

PACT

Before we can go on imaging some situations where potential users will be involved with a proposed system, there is a lot of information that need to be gathered.

- Goal: Build user profiles
- Potential users are part of your target audience. This needs to be defined and reflected upon.
- One exercise that you do undertake is PACT
  - **People**: Includes relevant user characteristics and skills
  - **Activities**: What are the tasks involved to complete a specific goal
  - **Context**: Activities undertaken in one environment could lead to a different outcome given another one
  - **Technologies**: What tools are used to complete these activities
- When PACT analysis is complete, you can create users profiles!

If you want to use it for the coursework

- **People:** Target audience
- **Activities:** Technical abilities, uses and habits
- **Context**: Socioeconomical status, laws, environment, etc
- **Technologies:** Tools available for you to design and for users to interact

Before or after the idea?

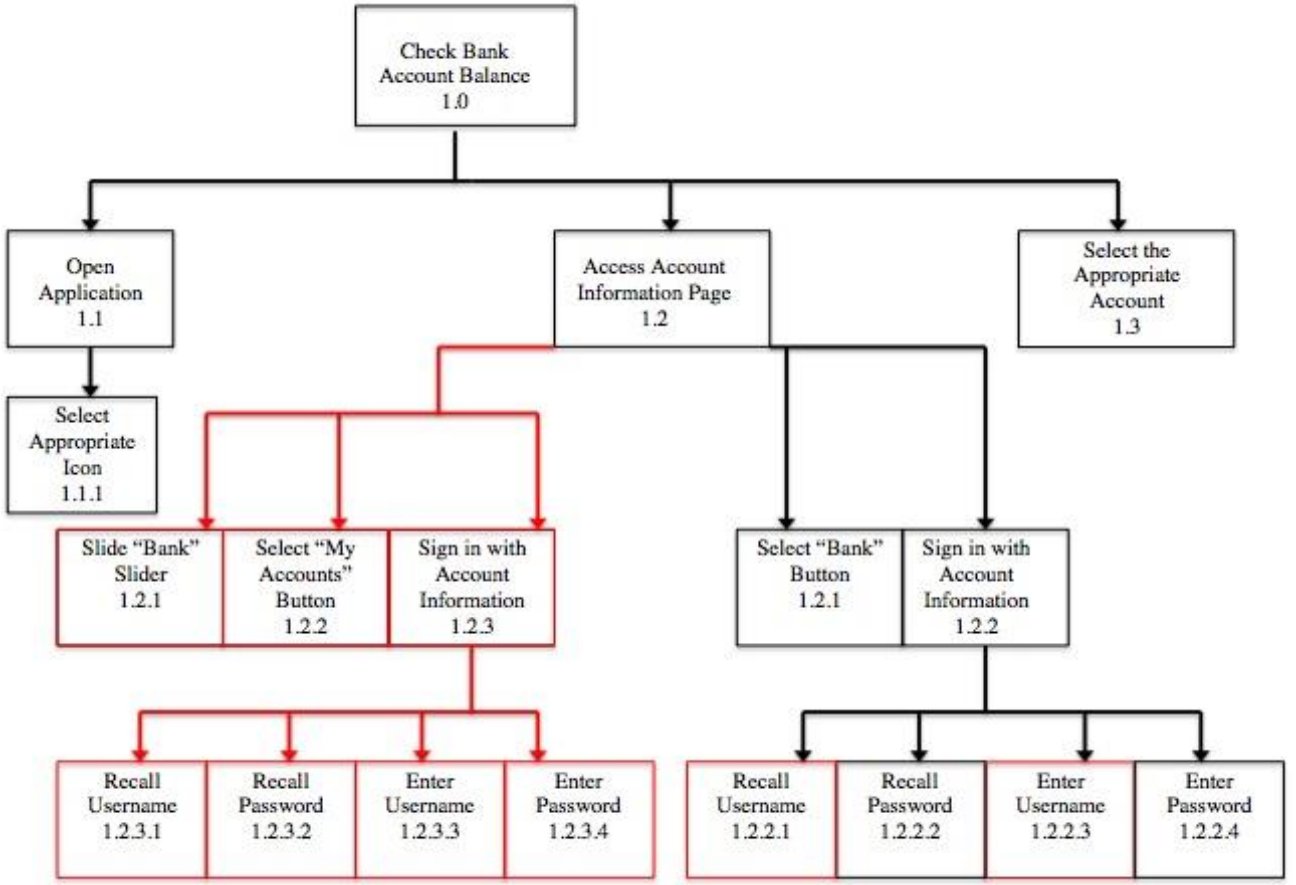
+ Accessibility

HCI Guide (page 10 to 14)

HTA

What is HTA?

- Description of tasks and sub-tasks to illustrate potential sequences that may occur through the interaction with a system
- Strict hierarchical structure with the primary goal represented as the top task
- Illustrates conditions that have to happen to get to certain subtasks
- First step for further analyses (SHERPA, critical path, etc)
- Steps are numbered (like creating sections for a coursework)



https://hfacmethods.wordpress.com/hierarchical-task-analysis/

Book Ordering Example

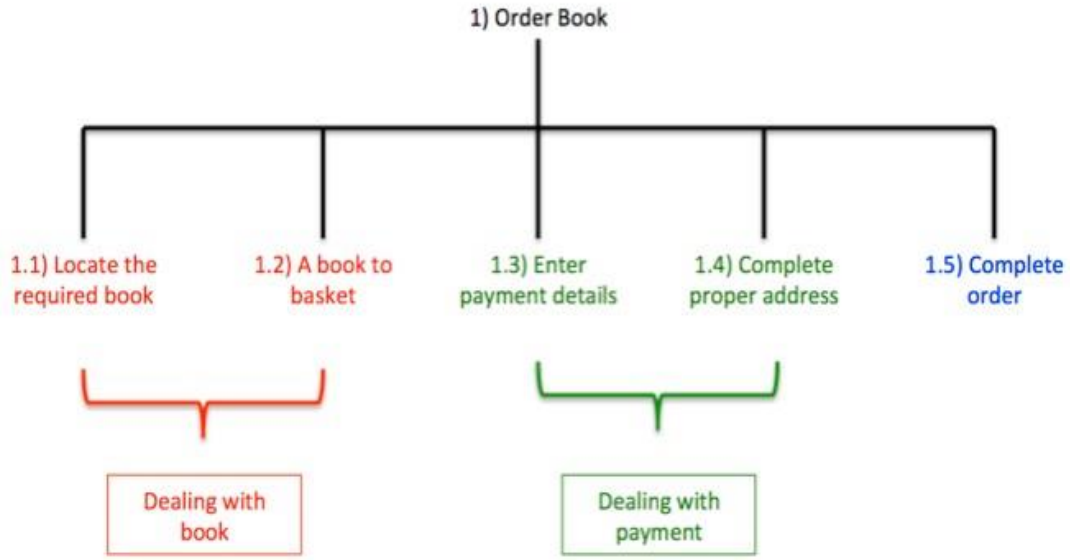
In this diagram, the main task has been broken down into subtasks.

A number scheme was used to help us understand the relationship between the main tasks i.e. ordering a book and the subtasks that are necessary in order to accomplish this main task.

This is a very rough analysis from the user point of view, as it does not communicate anything pertaining to user interaction with the system in order to complete these subtasks.

What it does however, is to provide a clear understanding of the tasks' high-level steps that are required.

Thus, the next step is to create a more complete diagram that includes users' interaction with the system.



Book Ordering Example

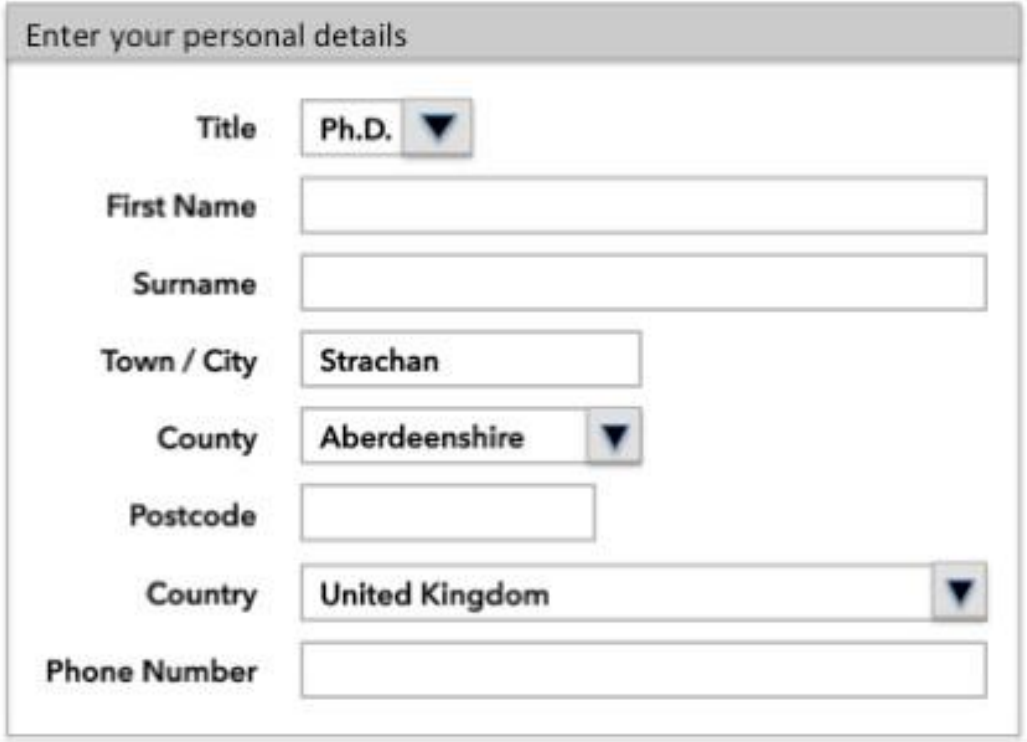
As an example, let's look at the ordering side of the systems, where the user types his/her personal information in order to buy what was selected.

- Locate title field
- Move the mouse/pen/finger to insertion point to the field
- Select adequate title i.e. Mrs, Mr, Dr, etc
- Locate the first name field
- Locate the county field i.e. Aberdeen, Aberdeenshire, Angus, etc
- Move the insertion point to the field
- Type or select the county
- Locate the postal code field
- Move the insertion point to the field

PLENTY OF DETAIL MISSING!

Book Ordering Example

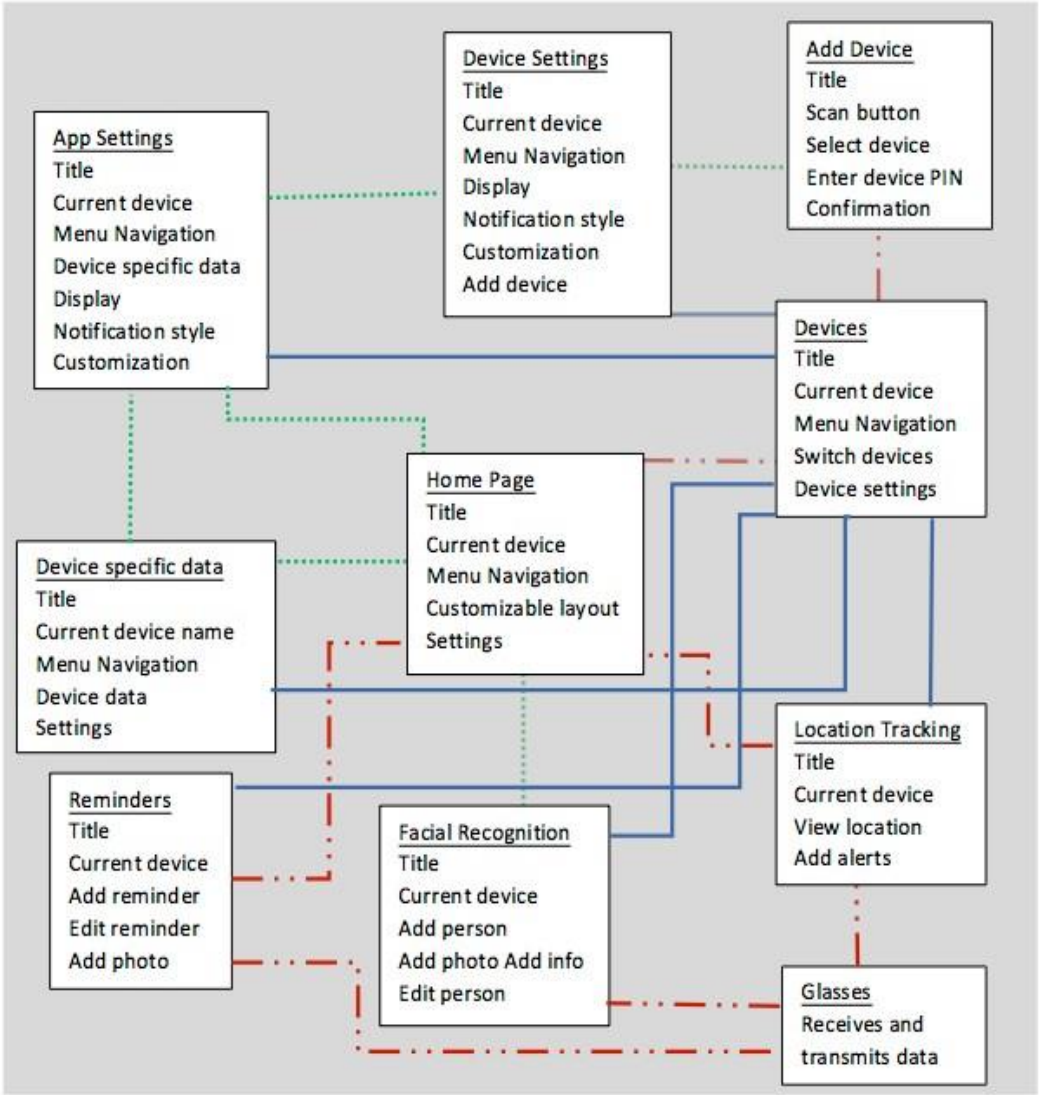
- The address is missing!
- However, this provides a good starting point
- Next, we can develop a prototype design
- Looking at the HTA, we have to assume that this goal is only achievable if a user has been registered
- Second, to arrive to the later stages of the transaction, the user must be signed in
- So, this brings us to the concepts of plans and strategies. For instance, one may think of developing the following strategies
    - User is new to the system
    - User has already registered
    - User has already registered and is signed in
- This provides opening to understanding some of the complexity of a system, which at the beginning where thought to be simple



Conceptual Design

Conceptual Design

- Before the processes of creating user profiles, scenarios and storyboards, you can create your first **Conceptual Design**
- Diagram/Schematic that allows you to visualise the interactions between your screens
- Considers the type of interactions, as well as the operations used
- Like an entity relational diagram in SQL



HTA → CD

- As you can see, a good HTA analysis can lead to a proper conceptual design
- Your main function ("section" 1) which is at the top of your HTA becomes the centre of your conceptual design (e.g. home page)
- From there, the "sub-sections" emanate!
- It is recommended to distinguish at the HTA step which relations *link to, retrieve data* or **do both**!