

Identifying Architecture Drivers

Lotfi ben Othmane

Architecture Drivers

- The architecture drivers define the **what** and **why** about the architecture
- They include:
 1. Design purpose
 2. Primary functionality
 3. Quality attributes
 4. Architectural concerns
 5. Architectural constraints

Architecture Design

Architecture drivers

Design purposes

Quality attributes

Primary functionalities

Architectural concerns

Constraints

Design concepts

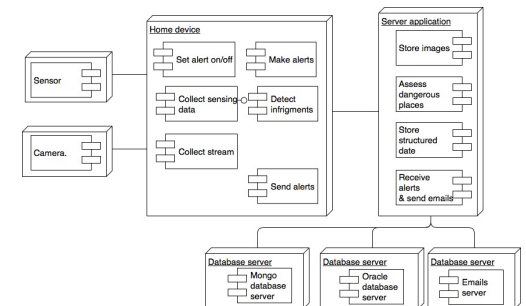
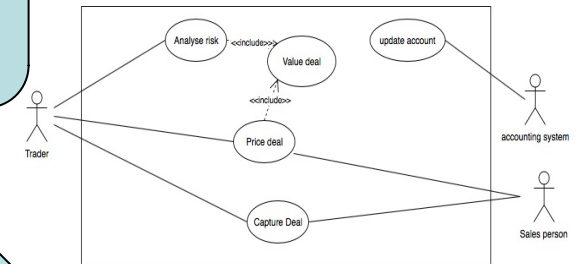
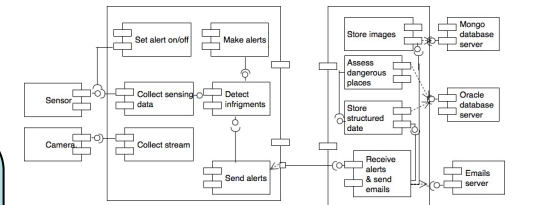
Selects and instantiates

Candidate design decisions



Architect

Architecture structures



Elicitation of Architecture Drivers

1. Analysis of the business goals
2. Analysis of the requirements documents
3. Use of existing knowledge
4. Quality attributes workshop

Analysis of the Business Goals

Support online courses

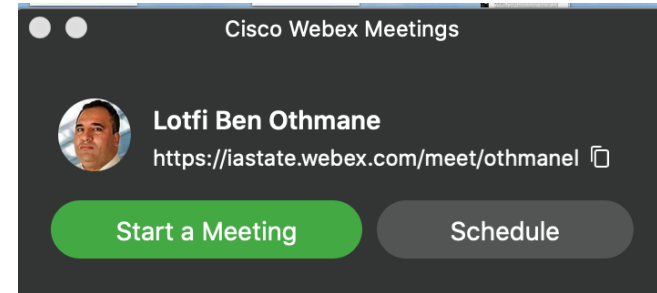
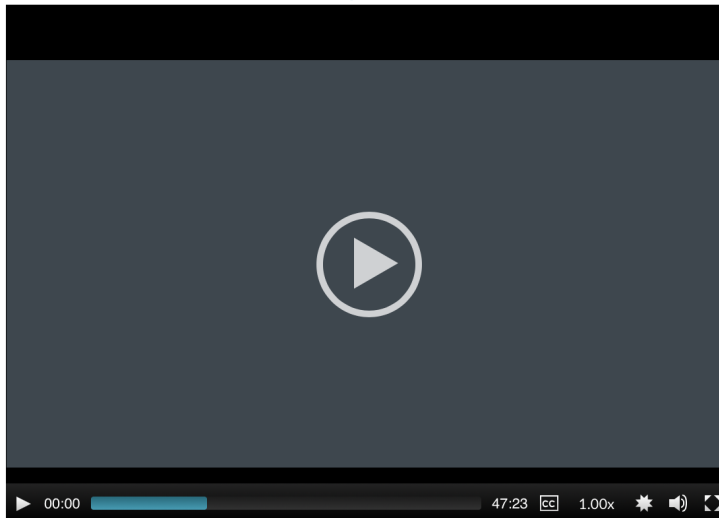
F2020-S E-339-1 > Files > SE339-20201001.mp4

Fall 2020

- Home
- Announcements
- Syllabus
- Modules
- Piazza
- Assignments
- Quizzes
- Discussions
- Grades
- People
- Webex
- ISU AdminTools
- My Surveys

SE339-20201001.mp4

[Download SE339-20201001.mp4](#) (52.8 MB)



Join a Meeting ⓘ

Enter the meeting information

Upcoming Meetings

Tuesday, October 6, 2020

5:00 PM | Practice sessions
5:30 PM | Lotfi Ben Othmane

Start

Thursday, October 8, 2020

1:00 PM | Big Data Brain Initiative
2:30 PM | (BDBI) and Magnetics Res...
Oluwaponmile Afuwape

Analysis of the Business Goals



Help

English (US) United States \$ USD

Log in

✈ Flights

🏨 Hotels

🚗 Car Rental

Where to next?

Roundtrip One way Multi-city

Map

From	To	Depart	Return	Cabin Class & Travelers
Ames (AMW)	Country, city or airport	10/11/20	10/18/20	1 adult, Economy

Add nearby airports

Add nearby airports

Non-stop flights only

Flexible tickets only

Search flights →

Analysis of Requirements Document

The main functionalities of the home security system are:

1. Set a system on and off
2. Collect data using sensors and camera
3. Analyze the data to detect intruders
4. Make alerts
5. Send alerts to the central system
6. Keep data to assess dangerous neighborhoods
7. Store structured data into an Oracle database
8. Store images in a MongoDB database

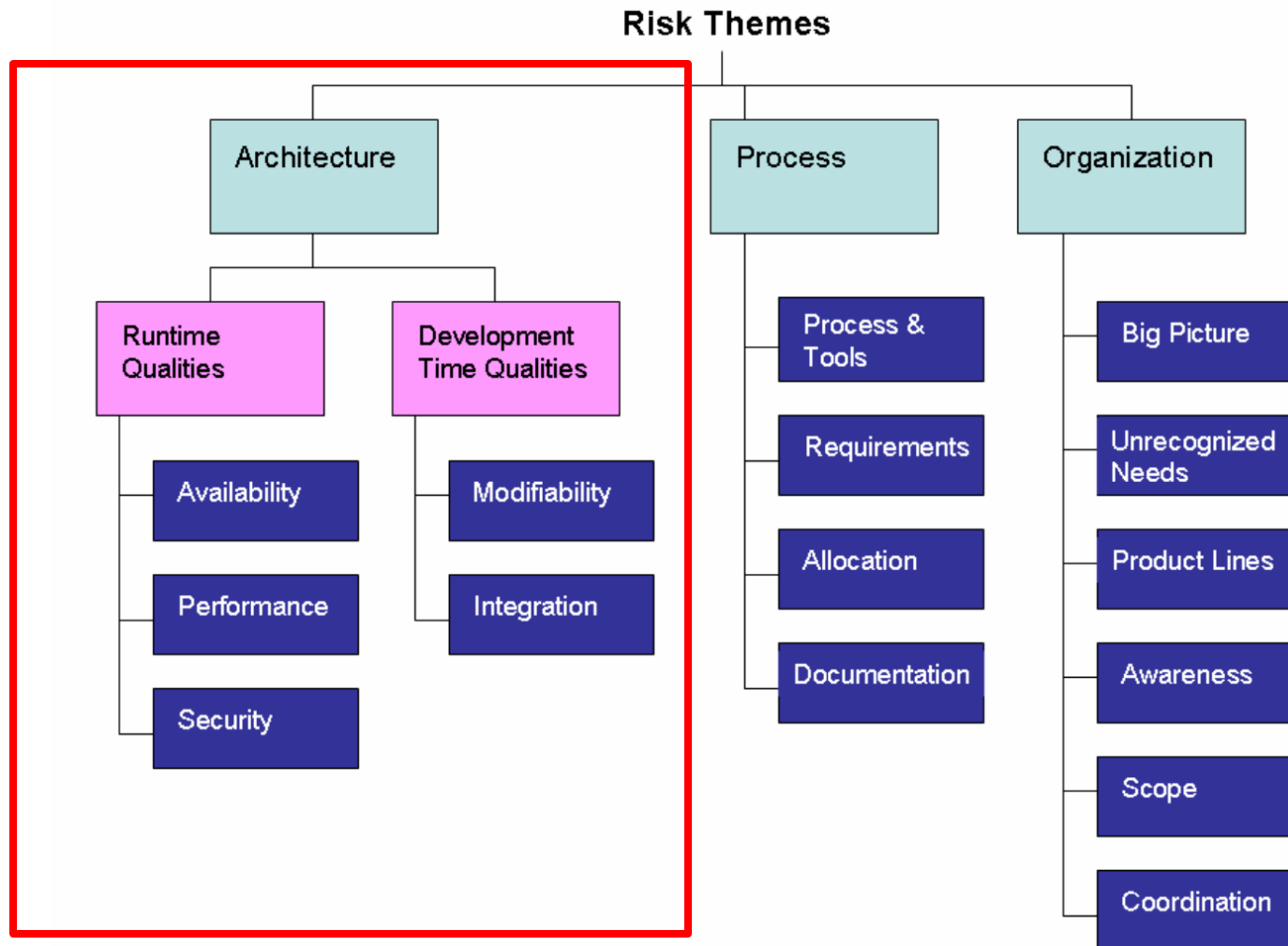
Analysis of Requirements Document

Some requirements have system-wide impact and others have local impact.

functional questionnaire

Line Number	Category	Specific Criteria
POS-001	Counter Functions	PC Software needs to work with current equipment; standard PC's and monitors, Windows 10, Dymo Labelwriter Printers, Gryphon Bar code scanners, cash drawers or hardware with similar functionality.
POS-002	Counter Functions	Receipt date and time
POS-003	Counter Functions	Duplicate receipt indicator
POS-004	Counter Functions	Receipt displays accounting date
POS-005	Counter Functions	Receipt displays a calendar date
POS-006	Counter Functions	Cashier # and register # on receipt
POS-007	Counter Functions	Type of sale (void, exchange, adjustment)
POS-008	Counter Functions	Ability to capture customer information, including but not limited to Date of Birth, Name, Address by transaction
POS-009	Counter Functions	G/L account up to a size of 50 characters
POS-010	Counter Functions	Scan bar codes, magnetic strips, etc. personal data/PT PIN/Inventory
POS-011	Counter Functions	Ability to set up transaction code by individual department
POS-012	Counter Functions	Ability to set up payment type by individual department
POS-013	Counter Functions	Software must be able to work on any PC including back office for reporting and management
POS-014	Counter Functions	Ability to report/query on any new transaction data
POS-015	Counter Functions	Must be able to capture and report in previous transaction history
POS-016	Features	Ability to accept multiple forms of payment including but not limited to cash, check, credit card, money orders, debit card, credit memo, ACH and Journal Entries.
POS-017	Features	Ability to accept electronic wallet and payment by Smart Phone
POS-018	Features	Ability to take combination of multiple payments in a single transaction
POS-019	Features	ABILITY TO PROCESS CREDIT CARD TRANSACTIONS ACCORDING TO State processes including convenience fees. Two MID's (Merchant ID) per credit card transaction calculate convenience fee, produce a two signature receipt
POS-020	Features	Ability of software to provide interface with 3rd party credit card processing vendors (passing encrypted account number/transaction)
POS-021	Features	Ability to produce customer receipts upon demand
POS-022	Features	Ability to produce receipts using a receipt printer

Use of Existing Knowledge



Use of Existing Knowledge

1. Avoid single point of failure
2. Use of known mechanisms to support security goals
3. Support the addition and deletion of different devices

Use of Existing Knowledge

Design Decision Category

Look for Requirements Addressing . . .

Allocation of Responsibilities

Planned evolution of responsibilities, user roles, system modes, major processing steps, commercial packages

Coordination Model

Properties of the coordination (timeliness, currency, completeness, correctness, and consistency)

Names of external elements, protocols, sensors or actuators (devices), middleware, network configurations (including their security properties)

Evolution requirements on the list above

Data Model

Processing steps, information flows, major domain entities, access rights, persistence, evolution requirements

Management of Resources

Time, concurrency, memory footprint, scheduling, multiple users, multiple activities, devices, energy usage, soft resources (buffers, queues, etc.)

Scalability requirements on the list above

Mapping among Architectural Elements

Plans for teaming, processors, families of processors, evolution of processors, network configurations

Binding Time Decisions

Extension of or flexibility of functionality, regional distinctions, language distinctions, portability, calibrations, configurations

Choice of Technology

Named technologies, changes to technologies (planned and unplanned)

Use of Existing Knowledge - Security Design Decisions

- Allocation of responsibilities – assess access needs
- Coordination models – assess secure collaboration
- Data model – ensure security of sensitive data
- Assess the impacts of mapping architecture elements on security
- Resource management – control accesses to monitor and manage the resources
- Binding time – identify binding cases and their security impacts
- Choice of technology – identify the impact of chosen technology on the security of the system

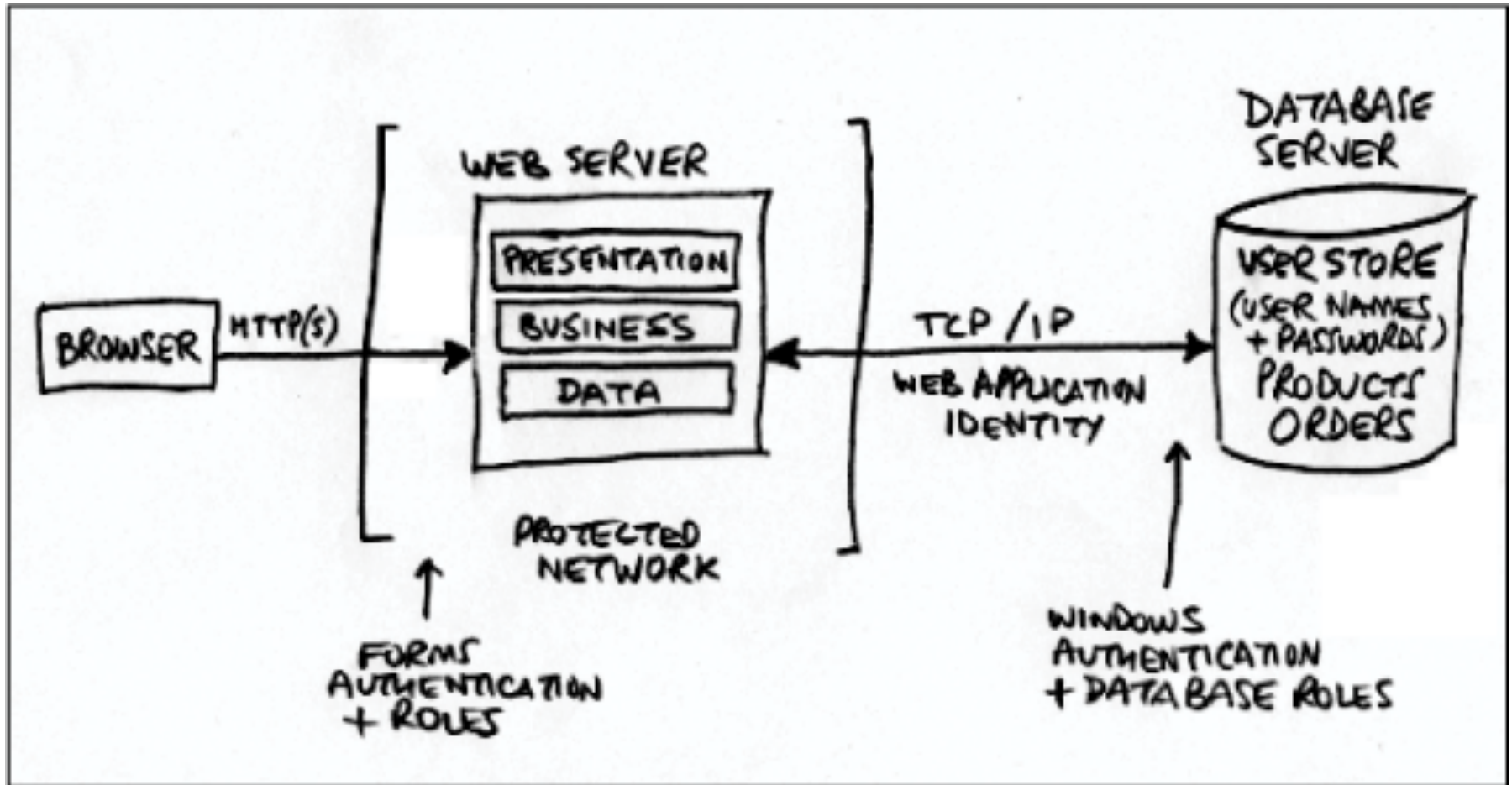
Quality Attribute Workshop

Quality attribute workshop - brainstorming session involving stakeholders for eliciting, specifying, prioritizing, and achieving consensus on QAs

QA Workshop Steps

1. Quality Attribute Workshop (QAW) presentation and introduction
2. Business goals
3. Architecture plan
4. Identification of architecture drivers
5. Scenarios brainstorming
6. Scenarios consolidations
7. Scenarios prioritization
8. Scenarios refinement

Architecture Plan



QA Workshop Steps

1. Quality Attribute Workshop (QAW) presentation and introduction
2. Business goals
3. Architecture plan
4. Identification of architecture drivers
5. Scenarios brainstorming
6. Scenarios consolidations
7. Scenarios prioritization
8. Scenarios refinement

Architecture Drivers Utility Tree

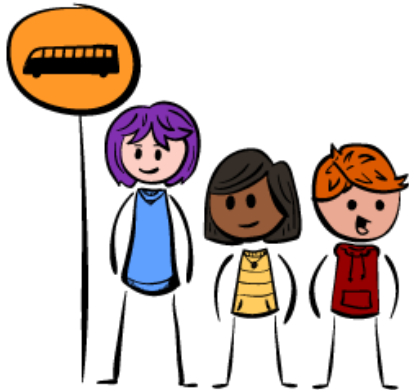
Quality Attribute	Attribute Refinement	ASR
Performance	Transaction response time	<p>A user updates a patient's account in response to a change-of-address notification while the system is under peak load, and the transaction completes in less than 0.75 second. (H,M)</p> <p>A user updates a patient's account in response to a change-of-address notification while the system is under double the peak load, and the transaction completes in less than 4 seconds. (L,M)</p>
	Throughput	At peak load, the system is able to complete 150 normalized transactions per second. (M,M)
Usability	Proficiency training	<p>A new hire with two or more years' experience in the business becomes proficient in Nightingale's core functions in less than 1 week. (M,L)</p> <p>A user in a particular context asks for help, and the system provides help for that context, within 3 seconds. (H,M)</p>
	Normal operations	A hospital payment officer initiates a payment plan for a patient while interacting with that patient and completes the process without the system introducing delays. (M,M)
Configurability	User-defined changes	A hospital increases the fee for a particular service. The configuration team makes the change in 1 working day; no source code needs to change. (H,L)
Maintainability	Routine changes	A maintainer encounters search- and response-time deficiencies, fixes the bug, and distributes the bug fix with no more than 3 person-days of effort. (H,M)
		A reporting requirement requires a change to the report-generating metadata. Change is made in 4 person-hours of effort. (M,L)

Key:
H=high
M=medium
L=low

Exercise – Pay With Travel Card



C1-VCC
C2-RST
C3-CLK
C4-
C5-GND
C6-VPP
C7-I/O
C8-



The photo ate licensed under [CC BY-SA](#)

Exercise – Business Goal

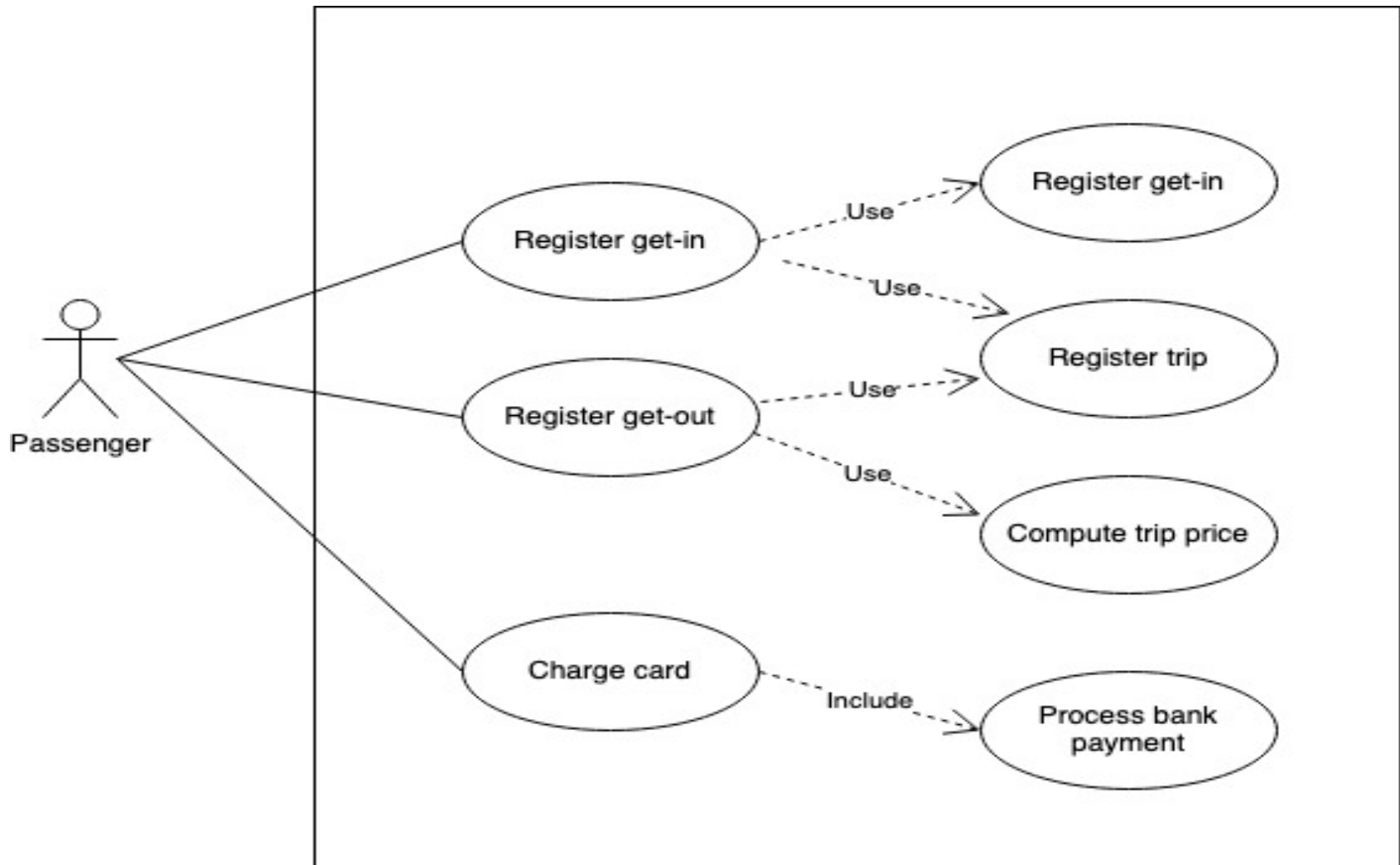
The city wants to simplify and control the payments of the use of public transportation

Exercise - Pay With Travel Card

Architecture drivers are:

1. Passengers can charge their cards,
2. Passengers scan their cards when they get-in,
3. Passengers scan their cards when they get-out,
4. The in-bus device computes the trip price, deducts the price from the balance in the card and registers the trip in the card and in the remote service.

Exercise - Pay With Travel Card



Exercise 2 - Pay With Travel Card

Examples of questions

Architecture driver categories

1. Primary functionality
 2. Design purpose
 3. **Quality attributes**
 4. Architectural concerns
 5. Architectural constraints
1. How long would a customer wait to get in?
 2. Would the company tolerate unpaid trips?
 3. Do you tolerate unavailability of internet

Exercise 2 - Pay With Travel Card

Examples of questions

1. How long would a customer wait to get in?
 2. Would the company tolerate unpaid trips?
 3. Do you tolerate unavailability of internet
- Only users with valid cards shall use the bus. Each card should include a secret key that the bus terminal can verify its validity.
 - A user should not wait more than 2 min to get-in. Up to 30 persons could wait for the bus in a given bus stop.
 - The system should work properly when Internet connection is not available for up to 15 min .
 - The terminal should restart immediately after exceptions and return the ID of the last processed card.

Exercise 2 - Pay With Travel Card

Architecture driver categories

1. Primary functionality
 2. Design purpose
 3. Quality attributes
 4. **Architectural concerns**
 5. Architectural constraints
- Establish overall system structure
 - Record bus terminal activities for **debugging**
 - The bus terminal should **log** all current transactions when an **exception** is raised

Exercise 2 - Pay With Travel Card

Architecture driver categories

1. Primary functionality
 2. Design purpose
 3. Quality attributes
 4. Architectural concerns
 5. **Architectural constraints**
- The system should support bad network connections
 - The developers have experience with Java and limited experience with C/C++

Self-Check

- What are the main four techniques to identify architecture drivers?
- Why the requirements document of a project is not sufficient in the identification of architecture drivers?
- What are the steps of the QA workshop?

Thank you

Questions?