Part I:

What does a software architect do? Due to the complexity of today's information system, software architects need to build the architecture of software correctly. In definition: "architects are responsible for designing and making technological decisions in the software development process" [1].

1. The overall role for an architect is about providing technical leadership and being responsible for the technical success of the project[2]. Apart from the other developer such as front-end developer or cloud engineer, system architect needs a high-level overview of the system in order to design a better-connected system. Moreover, he needs to consider the business objectives as well. Architect has responsibility for ensuring coherence of all aspects of the project as an integrated system [3].

2. Reducing risk where there is high complexity or uncertainty. For example, the software architect role for a finance company is about identifying, and owning the high priority technical risks in order to make sure the all the transaction runs safely [2]. Moreover, they need also to analyze and provide reviews on the magnitude of shortcomings and shortages in the systems, approaches to mitigate risks, resolve security alerts, and forecast possible adverse outcomes of the weaknesses, such as preventing random user using some web crawler to fetch customer data on the website [5].

3. The system architects need enough confidence and experience to lead the team and allocate tasks. Also, an architect should be a very good team player to support the team players. The software architect should also be partially responsible for providing general education for those lower level software engineers, for example, he can give workshops on specific software development skills during the development cycle. Educating the team and inspiring the other team members become a natural extension of the responsibility of the software architect [1].

4. Able to work with customers and solve customer-oriented tasks: [1]. By interviewing eight architects from four different firms, Sofia's team concluded that one of the tasks that is essential for software architects is working with customers, gathering requirements from marketing based on the understanding the engineering workflows and needs. This helps architects to introduce more user-friendly capabilities to the system[4]. For instance, the system architect of Apple should take more customer complains so that they could build a more energy-efficient IOS environment.

References

[1] McBride, M.R.: The software architect. Comm. of the ACM 50(5), 75â81 (2007) [*Online Access*]. https://link.springer.com/content/pdf/10.1007%2F978-3-319-07869-4_20.pdf

[2] Simon Brown. *Software Architecture*. Technical leadership and the balance with agility [*Online Access*]. `https://leanpub.com/software-architecture-for-developers/read#leanpub-auto-architects`

[3] Simon Brown. *The key difference between developer and architect roles.* [*Online Access*]. `http://www.codingthearchitecture.com/2008/02/11/the_key_difference_between_developer_and_architect_roles.html`

[4] Ameller, D., Ayala, C., Cabot, J., Franch, X.: How do software architects consider non- functional requirements: An exploratory study. In: 2012 20th IEEE International Re- quirements Engineering Conference (RE) 2012, pp. 41â50. IEEE (September 2012)

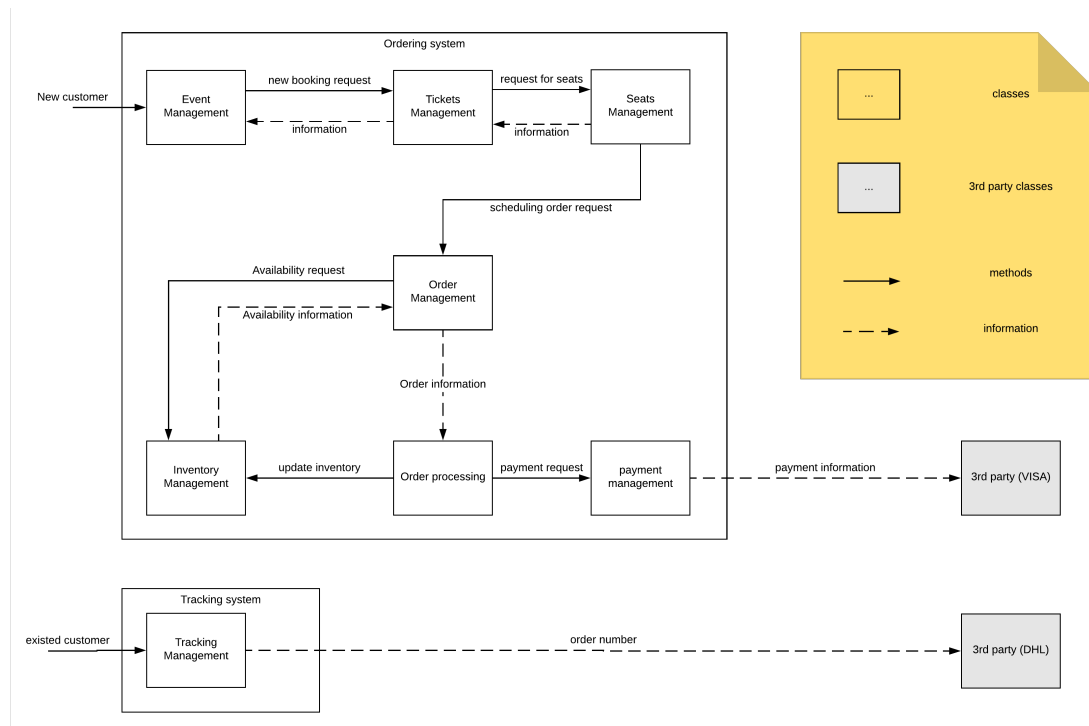[5] Field Engineer*Senior System Architect.*. [*Online Access*]. `https://www.fieldengineer.com/skills/senior-system-architect`

Part II:

Figure 1 content:

Ordering system

New customer → Event Management — new booking request → Tickets Management — request for seats → Seats Management

Event Management ← information — Tickets Management ← information — Seats Management

scheduling order request

Availability request — Order Management

Availability information

Order information

Inventory Management — update inventory — Order processing — payment request — payment management — payment information → 3rd party (VISA)

Legend (classes note):
... classes
... 3rd party classes
→ methods
⇢ information

Tracking system

existed customer → Tracking Management — order number — → 3rd party (DHL)

Figure 1: Functional view for ticket booking ststem

Figure 2 content:

Clients
<<device>>: UserClient
<<device>>: Browser
<<artifact>> HTML 5

Server
<<device>>: Web Server
<<artifact>> Online Shopping Cart
<<artifact>> Payment Gateway
<<artifact>> Tracking Gateway

<<device>>: DB Server
Ticket-allocation Database
Users-Orders Database

Payment Portal: VISA
Tracking Portal: DHL

https:80

Notes
Node — devices
Object — Artifacts
— deployment connection
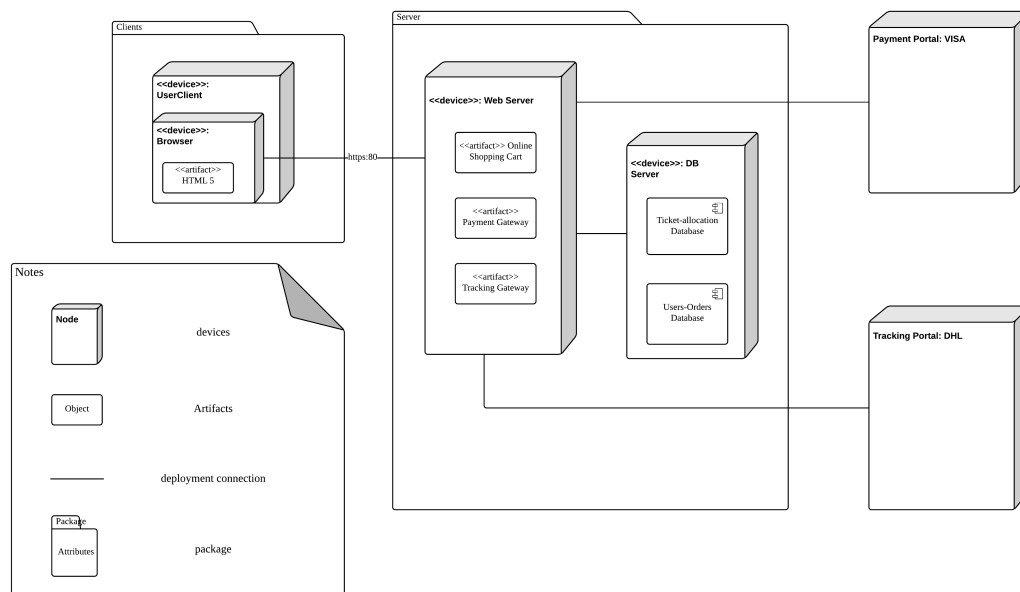Package / Attributes — package

Figure 2: Deployment view for ticket booking ststem

Figure 1 shows interaction between internal components. Its main target audience are new customers who have few professional knowledges in computing.

Figure 2 illustrates deployment for different devices. Its main target audience are software developers, such as front-end developers, database developers, network engineers, product managers.