Assignment 2
TU/e Parking System

Group 30

By:
Matej Grobelnik (1281674)
Bram van de Wall (0911838)

Stakeholders

We have identified the following stakeholders:

- External users (can use only ticketing system)
- Internal users (employees and students)
- TU/e (Various departments: Dienst Huisvesting, Dienst interne zaken, Safety & Security)
- (H)eerlijk parkeren
- Companies located on the campus (can provide exit cards)
- Emergency vehicles
- Payment processors
- Hardware/Software vendors

We will not go into detail for all of them.

Extra-functional requirements

Security: Since system stores personal data (e.g. payment info, license plate, user data), it should have sufficient level of security required by the law and security guidelines.

Availability: It is important that the gates components work at every moment of the day. Individual components going down should not disable the whole system. If the administrative system or an individual gate goes down the system should continue to function. It is also important that the online account system does not fails, since people may not be able to leave the campus.

Scalability: Gates, Payment Booths, Availability Indicators and Fields can be added or removed without upgrading the rest of the system.

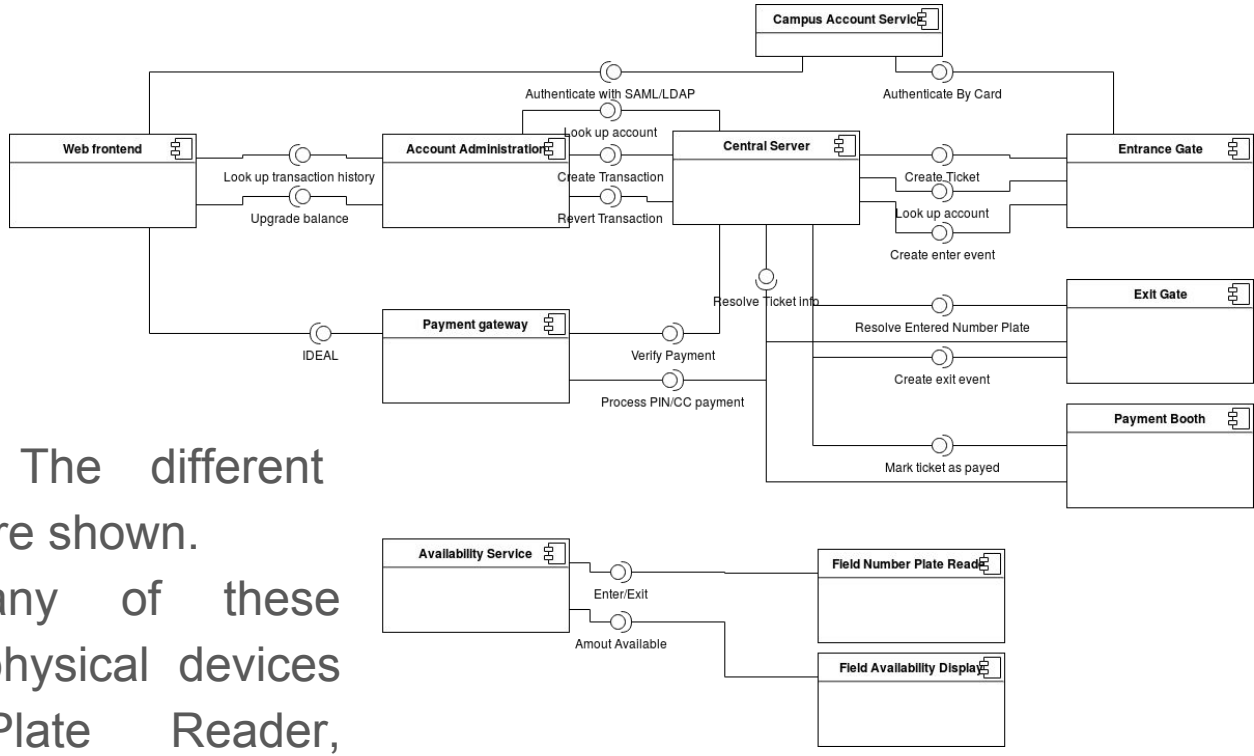
Maintainability: System should allow maintenance by a third party company and use of common components that have high availability of spare parts.

System overview

Diagram Description:
The general components that the system consists out of.

Viewpoints and stakeholder:

- **Development View:** The different software components are shown.
- **Physical View:** Many of these components map to physical devices (Gates, Number Plate Reader, Availability Display)



Architectural style

Client-Server

We use this style as we can map the components fairly easily to different servers. It may also be the case that not all components run on hardware owned by the same stakeholder. For example (H)Eerlijk parking may own the parking related services, but needs to communicate with a TU/e owned authentication server.

Interaction style

Request- Reply / Remote Procedure Call: We think that the most important interaction style used is the request- reply style or remote procedure style. This style is used because of the client server connection.

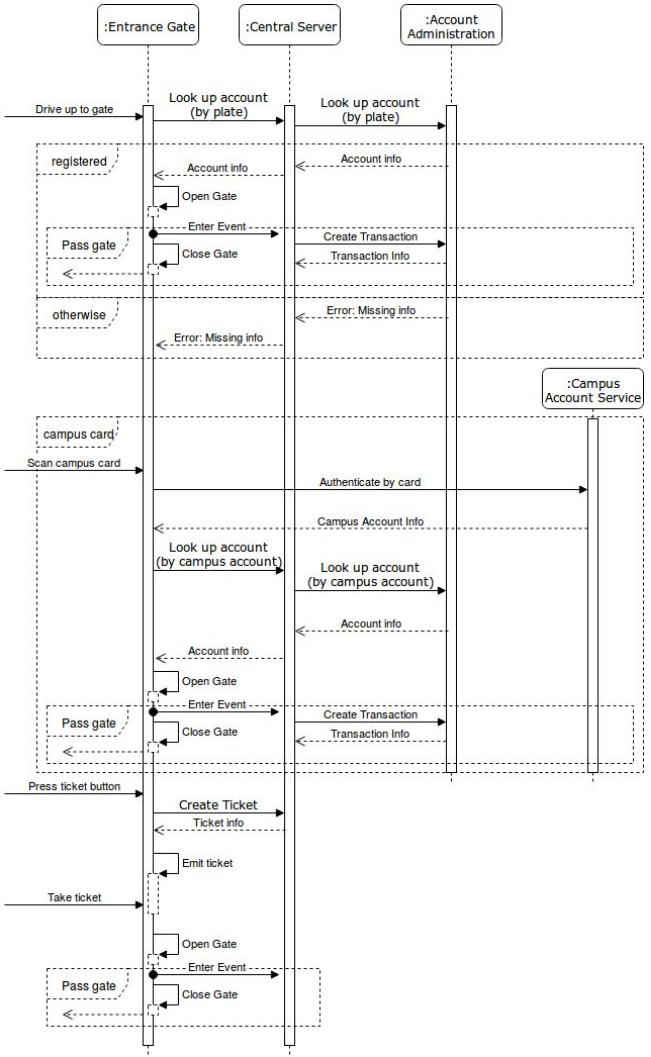
Users log in on their account (client) and make requests about their accounts to the server. This style is very common for Remote procedure calls, it has good portability of code, it reduces the language and operating system dependence and accesses transparency. Especially the dependence on OS is important since users might log in on their account using different browsers and/or OS.

Message Sequence Chart

Diagram Description: The various ways a user can enter the premises (Use Cases) and what communication occurs.

Viewpoints and stakeholder:

- **Process view:** Interesting for testers and integrators that have an interest in putting the system together and verifying that the parts interact in a proper fashion.

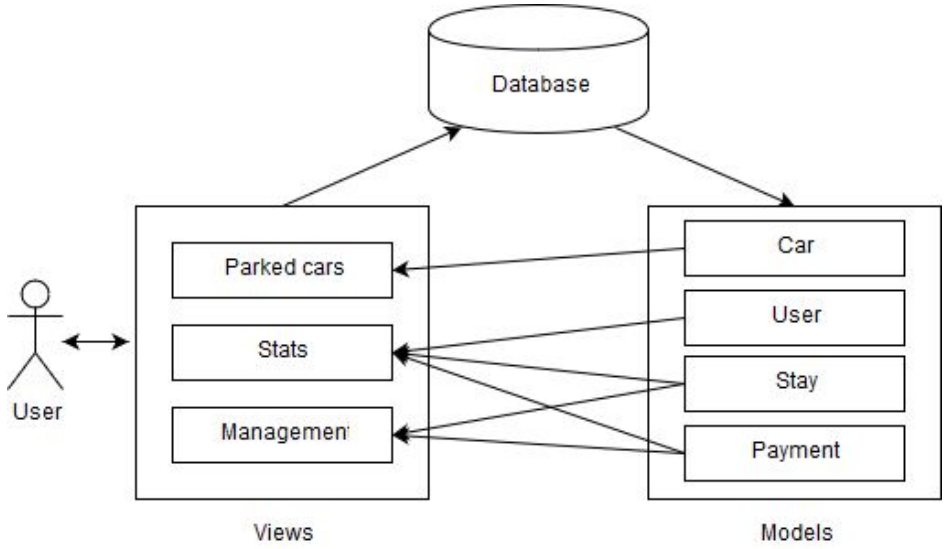


Management application - MVC architecture

Viewpoints:

- **Logical view:** A global overview on the system is shown to developers.
- **Development view:** The idea of data partition is shown in the view.

Stakeholders:
Interesting for developers and programmers to know how data is partitioned.



Availability Service (Activity Diagram)

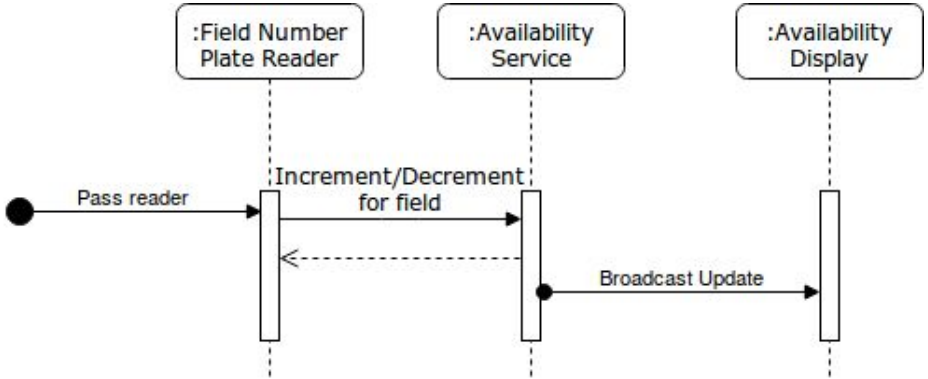
Viewpoints:

- **Process view:** Shows the communication between components.

Stakeholders: Developers, Testers, System integrators

Interaction Styles:

- **RPC:** For the update call.
- **Pub/Sub:** For the availability display.



Undesired scenarios

“Make it very hard to determine the exact amount to be paid.”
Unless the user forgets their ticket, the system can always know on entry how much will be paid for users with an account. For users without an account the amount is simply determined by the time present.

“Make it very hard to enforce that that amount indeed will be paid.”
In the design described in the previous slides, the entrance gate will always look up the account, and thus the balance, before letting a car in. This is open to a race condition when multiple cars enter with the same account, the balance is not yet deducted until they actually pass the gate. This will however leave the account with a negative balance and therefore not worth the effort to abuse.