Smart METU Campus

Software Requirements Specification for the Smart METU Campus Project

PROJECT EXECUTING TEAM

Project Member	METU Number	Signature
Kürşat Öztürk	e2171874	
Beste Burhan	e2171395	

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Purpose of the System

This project aims to handle daily problems easily that students and instructors face in daily life of university. Such problems are knowing ring location information and time it arrives to the stations, seeing and reserving available seats & books in the library, and attendance to the classes.

1.2 Scope

- Project will include an interface that provides students, instructors and staff to see ring information. Not only exact location, but also the approximate time information that ring will arrive to the stations via web application.
- Project will include an interface that provides students and instructors to see and reserve available books and seats in library via an IoT located in the entrance of library or web application.
- Project will include an interface that provides to students that they can give attendance to the classes they attend.
- Project will provide instructors to take attendance in an easy and without any cheating, that often occurs in schools.
- System will hold attendance information in database so that if course syllabus requires it, instructor giving the course can check students' attendance information at the end of semesters.
- System will hold a login interface for users. Users can login either via reading ID Card to certain IoT devices or entering the username & password of ODTU system created when they registered to the school.

1.2 System Overview

1.2.1 System Perspective

Smart METU Campus product includes Web Application, IoT Devices and Cloud server. Users will be able to interact with IoT devices via their METU Card and Web Application with their browser capable devices. GPS Devices will be in the Rings and not be in the interaction with the users. It sends real-time information to the server. The other IoT Devices which will be located in the classes and library can interact with users and server to

send and get information.

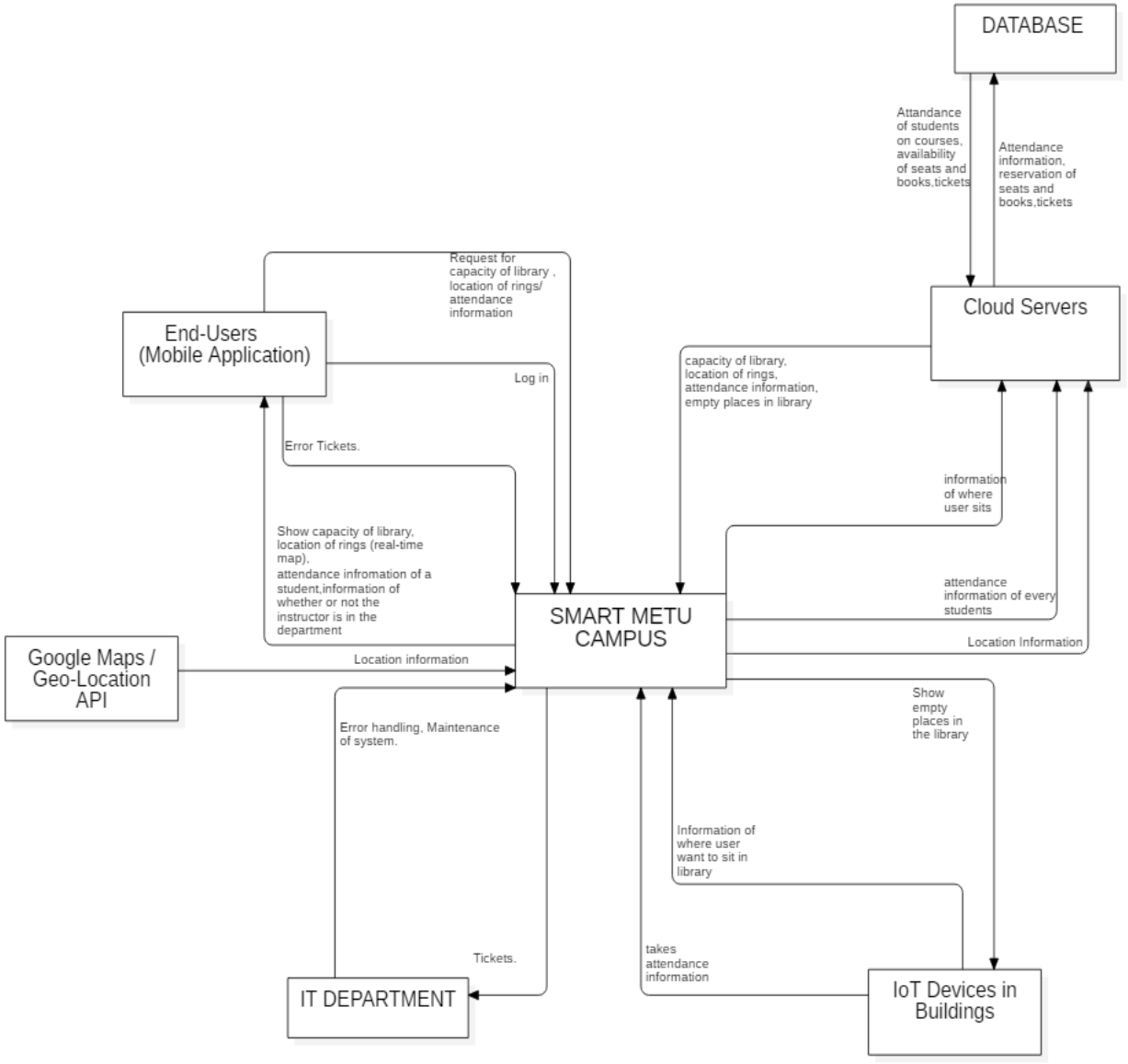

Figure 1 Context Diagram

1.3.1.1 System Interfaces

Backend Taking Attendance Information Interface: Students gives participating information to the system via IoT Devices. This interface collects this information and sends them to the Database. In addition, it can get them from Database so, a student's attendance to a class can be tracked efficiently.

Backend Real-Time Location Information Interface: This interface provides tracking the ring services. One can see the ring's live location and can see the approximate time it will take that it comes to particular ring station.

Backend Library Availability Tracking Interface: The users of library will see the available desks via an IoT device. Also, it will track those taking a break and calculates time it took and if it is more than 30 minutes, it will mark the desk as empty.

Backend Ticket Interface: Users will create tickets if any process goes wrong. This interface collects these and sends them to the Database. IT Staff can fetch the tickets from Database.

1.3.1.2 User Interfaces

METU ID Card Interface: Students, instructors and staff members' METU Card's will be the login key to the system.

Web Application Interface: This allows users to track rings, to see availability of library.

IoT Device Interface: With the help of IoT devices in entrance of classes it collects attendance information of students. Also, in library it provides the user that the list of available desks in the library.

1.3.1.3 Hardware Interfaces

METU Servers: This holds the personal data and authorization methods in its and uses METU ID Card as unique identity.

Database Servers: Smart METU Campus requires a database to store capacity and/or attendance information, tickets and availability of library seats. This data will be stored in the Database servers.

A Device for Web App: Any device capable of opening web browser.

1.3.1.4 Software Interfaces

Database Management System: Tickets, tracking and attendance data will be served by DBMS.

Web Application: Users will be able to track rings, to see availability of library.

IoT Device Software (Library only): It shows available desks to users and marks the seat chosen by user.

1.3.1.5 Communications Interfaces

It requires an active internet connections. IoT Device communicates with the cloud via HTTP. Also Web Application uses the same way to communicate with the cloud.

1.3.1.6 Memory Constraints

Although memory is not a huge problem, there will be a lot of attendance data that may be hold at least for a semester. Besides, for real-time usage availability of books and seats in the library will be stored for short-term such that if any of them is reserved, it will be written to database, if become available it will be deleted, so the memory usage is bounded with the number of books and seats in this case. Servers must have adequate hardware for this memory requirements.

1.3.1.7 Operations

Student Operations:

- Log In
- Show instant locations of rings and time it takes to arrive to a station
- Show availability of Library
- Choose a seat in Library
- Be Count in attendance

Staff Operations:

- Log In
- Show instant locations of rings and time it takes to arrive to a station

Instructor Operations:

- Log In
- Show instant locations of rings and time it takes to arrive to a station
- Show availability of Library
- Reserve a seat in Library
- See the attendance list of students to courses

IT Staff:

- View tickets.

1.3.2 System Functions

Functionalities (use cases) of Metu Smart Campus is summarized as what they do in below. More advanced and detailed version can be found in Functions section (3.2) with their complete description tables.

Functionality	Summary
Log In	Users log in Web Application.
Show attendance of a students on a course	Show attendance of students on course during semester.
Show users' break time information	If users give break when they are in library, they can see how many time passed.
Show location of rings	Get exact location of rings and time of arrivals.
Show availability of library	Get availability of seats or get availability of given book.
Open Tickets	Users, or IoT devices create tickets for errors.
Choose a seat	Users can reserve a seat from library to study.
Give class attendance information	When students attend to classes, they inform the system that they attended the class from IoT device.
View tickets	IT Staff can view tickets and take action to correct errors.

Table 1:System Functions

1.3.3 User Characteristics

The potential users can be divided in four group in this system. First group is students. Students are the most crowded group and have biggest acces range on interfaces. Second group is instructors. Instructors also have almost access all of interfaces, but apart from students, they can take attendance and check them retrospectively. Third group is staff. They have very limited access on interfaces. Last group is IT Staff, which are the admin of system. In any erroneous situation they access to the system and solve it.

1.3.4 Limitations

Regulatory policies: System uses the user information that already exists/created in METU Servers. The only data stored is attendance information, and no information is shared to outside.

Hardware limitations: System requires IoT devices on entrance of each class and library. Those IoT devices should have access to internet. Besides, there is a server on the cloud and a database management system for holding the attendance data and availability of seats and books in the library. Although the data will be dealing with is small, amount of is may be very high and hence database server should have huge storage capacity.

Interface to other applications: System will interact with google Geo-Location api via http requests.

Parallel operations: System should be able to handle each individual request asynchronously, parallelism would help, but not a must.

Audit functions: There is no finance operations.

Control functions: A control mechanism should be constructed for IT Staff use only. Only exception is that, instructor can track attendance of students, but can't add, change or delete anything.

Higher-order language requirements: System should be written in fast and reliable Object-Oriented language such as C++. Golang may be used in microservices, and a proper javascript library must be used in web application.

Signal handshake protocols: HTTPS will be required in all of interaction between components, so data sent can be encrypted. Besides, SSH connection will be provided to IT staff for situations in case any error. TCP will be used for database connection.

Quality requirements: Data should be kept safe, so system should use a backup policy.

Criticality of application: Except the loss of data, any kind of error can be tolerated and/or handled by IT Staff.

Safety and security considerations: Both cloud server and database should be immune to any kind of hacking attempts; however, critical user data(username, password, personal info etc.) will never be stored in those, will be fetched from METU Servers if necessary.

Physical/mental considerations: Since there is attendance, students should be able to make IoT devices in the entrance of classes and library read their ID card. Other than that, accessing any device capable of opening browser should be enough.

1.4 Definitions

DBMS	A NoSql Database Management Systems
IT Staff	Information Technology Department's Staff
API	Application Programming Interface.
METU	Middle East Technical University
GUI	Graphical User Interface
METU ID Card	Smart Card given by METU to members.
METU ID	Unique number to identify each person given by METU.

Table 2: Definitions

2. References

This document is written with respect to IEEE 29148-2011 standard:

IEEE. (2011, December 1). 29148-2011 - ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes --Requirements engineering. Retrieved from http://ieeexplore.ieee.org/document/6146379/ on March 12, 2018. doi: 10.1109/IEEESTD.2011.6146379

Other sources:

Microsoft Garcon Project, 2018, October
https://www.youtube.com/watch?v=Ad_EHDcomR8&t=20s

3. Specific Requirements

3.1 External Interfaces

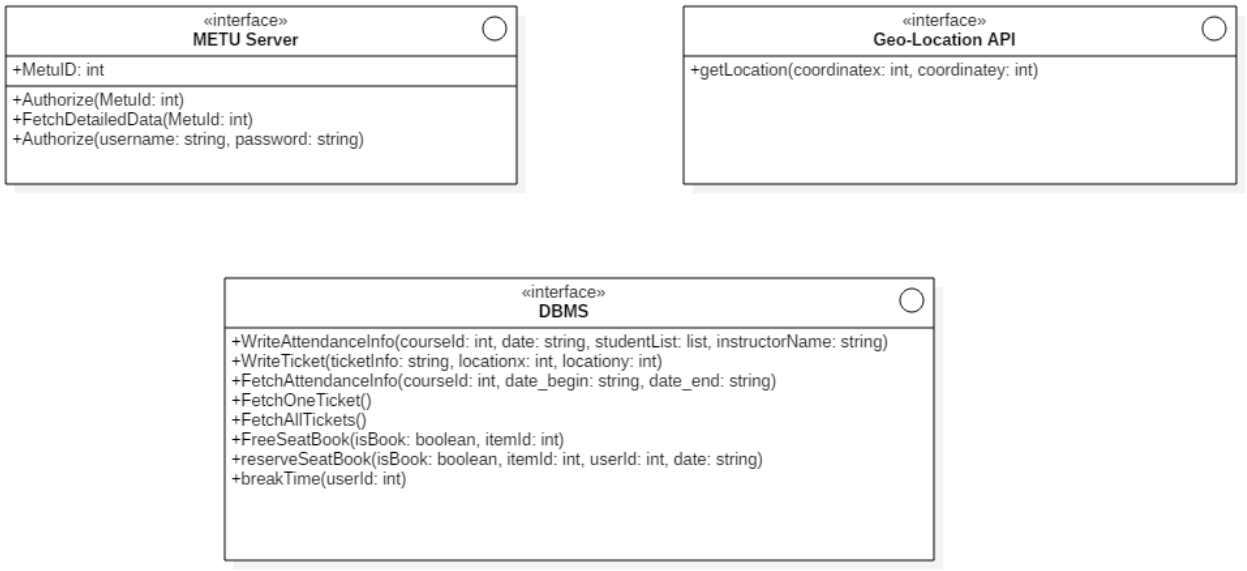

Figure 2: External Interfaces Class Diagram

Metu Server Interface: Smart Metu Campus is apart from METU internal servers. Detailed information will never be stored in Smart Metu Campus Servers. If it become needed, it will fetched from METU Servers. Besides, Smart Metu Campus has no its own authorization service. It uses METU internal servers to authorize users since target user is METU members.

Geo-Location API Interface: To be able to locate rings in the map, google's geo-location api should be used. Here, the coordinates fetched from the gps devices attached to rings, will be sent to the API and the exact location will be fetched.

DBMS Interface: Since authorization is not Smart Metu Campus project's concern, there is little task left here. Only Ticket, attendance and availability of library will be stored.

3.2 Functions

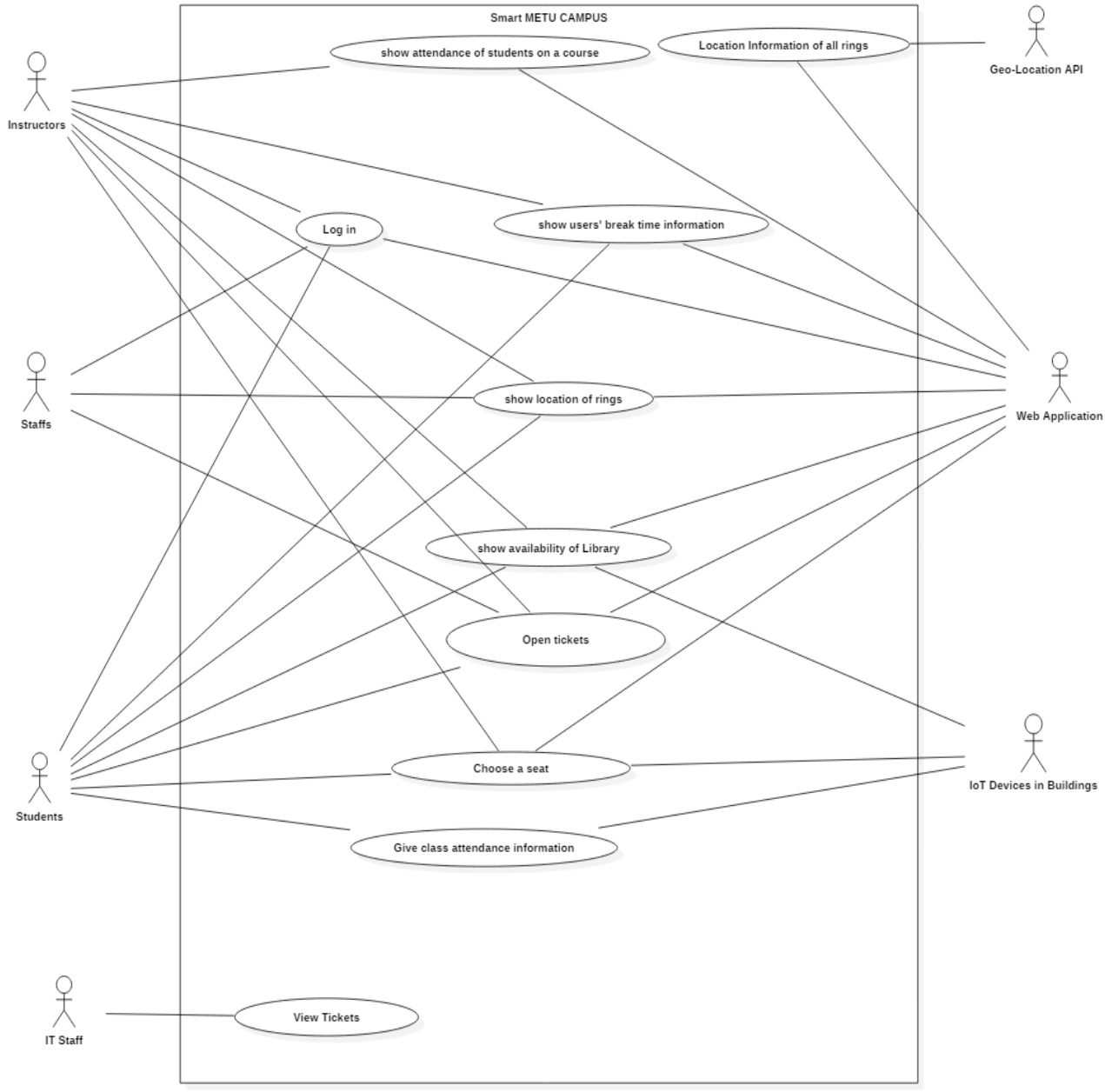

Figure 3: Use Case Diagram

Use case name	Show location of rings
Actors	Student, Instructors, Staffs -Web Application
Description	If an user send request for location of rings, according to information from IoT devices, server returns the coordinates to the map(in Web application) with real-time ring monitoring thanks to the Geo-Location API. Besides, It will calculate approximate time that the ring(s) come in to specified in parameters.
Data	Ring color(optional, if not specified all ring data will be returned). Station number/location(must)
Preconditions	IoT Devices sends their position information.
Stimulus	User requests the data from Web application.
Basic Flow	1 - User requests the data from web application. 2 - Web application fetches coordinates of rings. 3 - Web application sends the coordinates to the API. 4 - Location data sent back to the Web application. 5 - Web application sends the locations to the user. 6 - User see the map with position of ring with constantly updated information and time left to reach the station.
Alternative Flow	1 - User demands for location of all rings from application. 2 - From server, necessary information is sent 3 - User see the map with position of all rings with constantly updated information and time left to reach the station.
Exception Flow	If Geo-Location API returns an error, it is also written to error log file. If IoT devices don't work (so if data doesn't come), then it is written error log file. In addition, error tickets are sent to the IT staff.
Postconditions	Google map appears with positions of all rings.

Table 3:Show Location of Rings

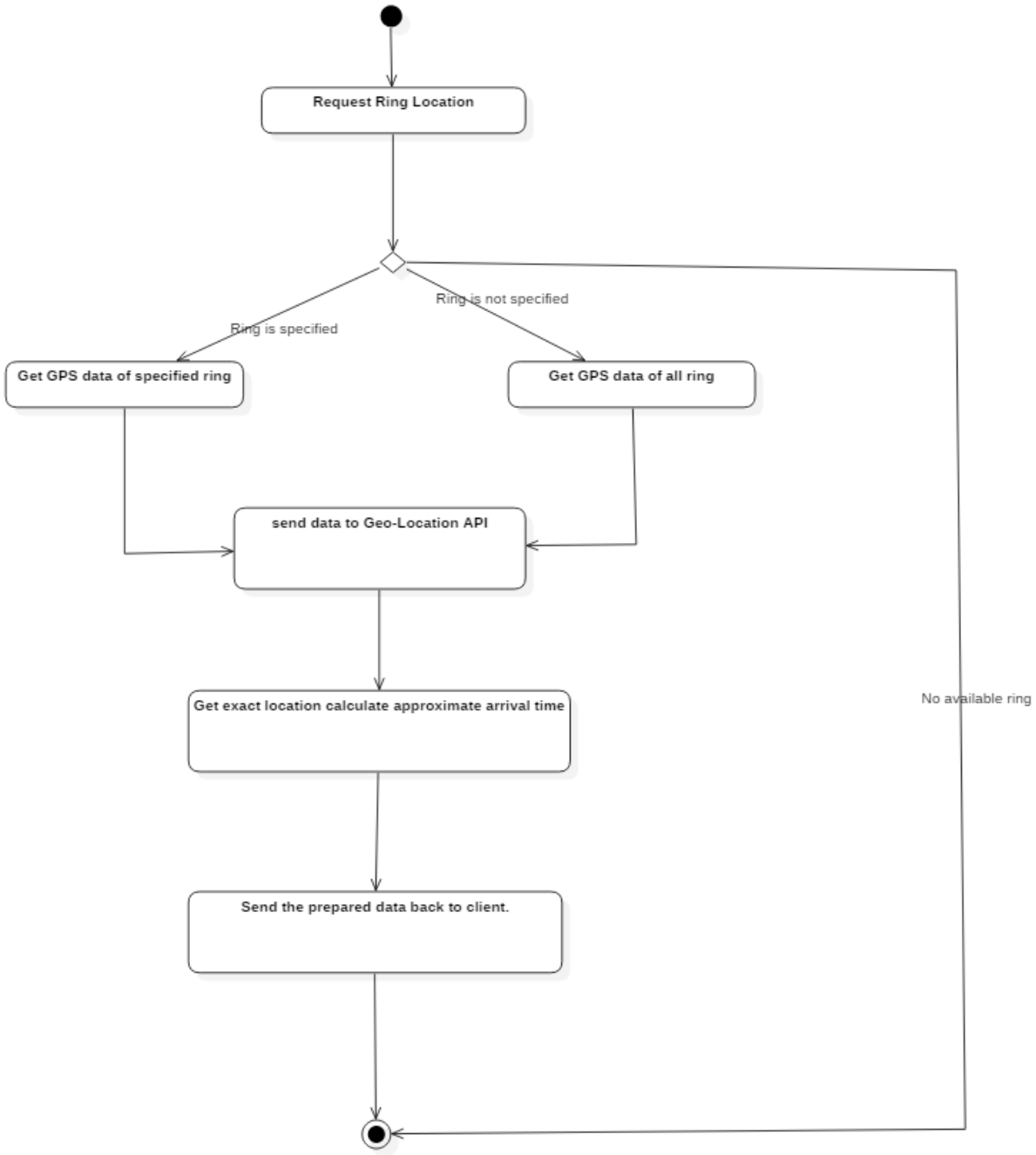

Figure 4: Activity Diagram for Show Location of Rings

Use case name	Show availability of Library
Actors	Student, Instructors- Web Aplication
Description	When a user wants to see available places on the Library, empty places are displayed by pulling the required information from the database.
Data	-
Preconditions	Users must be logged In.
Stimulus	-
Basic Flow	1-User clicks availability of Library button on the application. 2-Data will be sent to the application.
Alternative Flow	-
Exception Flow	If database connection is lost, error message will send to the user. Developers will be informed. In addition, error tickets are sent to the IT staff.
Postconditions	Visualized location information of Library will be shown.

Table 4: Show Availability of Library

Use case name	Give Class Attendance Information
Actors	Student, IoT Device
Description	Students will show their METU ID Card to the IoT Devices which are in the entrance of the classes in beginning and finishing time of course.
Data	-
Preconditions	-
Stimulus	IoT Device sensors will read the students' METU ID Card.
Basic Flow	1-When the course time start, Students coming to take course show their ID Card to sensor. 2-After class dismissed students must show their ID Card again to complete attendance taking.
Alternative Flow	In any alternative flow, students will not count as attended.
Exception Flow	If database connection is lost, error message will send to the user. Developers will be informed. In addition, error tickets are sent to the IT staff.
Postconditions	-

Table 5: Give Class Attendance Information

Use case name	View Tickets
Actors	IT Staff
Description	If a user or IoT device sent error ticket, IT Staff can get those from Database.
Data	Tickets
Preconditions	-
Stimulus	Records on the Database.
Basic Flow	1-IT Staff check for new tickets from Server. 2-They attract them from Database.
Alternative Flow	-
Exception Flow	-
Postconditions	IT Staff will handle errors.

Table 6: View Tickets

Use case name	Show Attendance of Students on a Course
Actors	Instructors, Web Application
Description	Instructor will be able to fetch all attendance data from server, so he/she can evaluate students attendance.
Data	Course Name
Preconditions	-
Stimulus	-
Basic Flow	1-Instructor requests the attendance data of a course. 2-Data sent back to the instructor from the server.
Alternative Flow	-
Exception Flow	If database connection is lost, error message will send to the user. Developers will be informed. In addition, error tickets are sent to the IT staff.
Postconditions	-

Table 7: Show Attendance of Students on a Course

Use case name	Show User's Break Time Information
Actors	Student, Instructors, Web Application
Description	When a user who holds a desk in library takes a break, there starts a countdown (30 mins) and user can see how many minutes they got.
Data	-
Preconditions	User must be logged in.
Stimulus	-
Basic Flow	1-User requests the data from Web application. 2-Data sent back to the user.
Alternative Flow	1-User requests the data from Web application. 2-If the countdown is finished, server raises Error since user will no longer holds any desk from library.
Exception Flow	If database connection is lost, error message will send to the user. Developers will be informed. In addition, error tickets are sent to the IT staff.
Postconditions	-

Table 8: Show User's Break Time Information

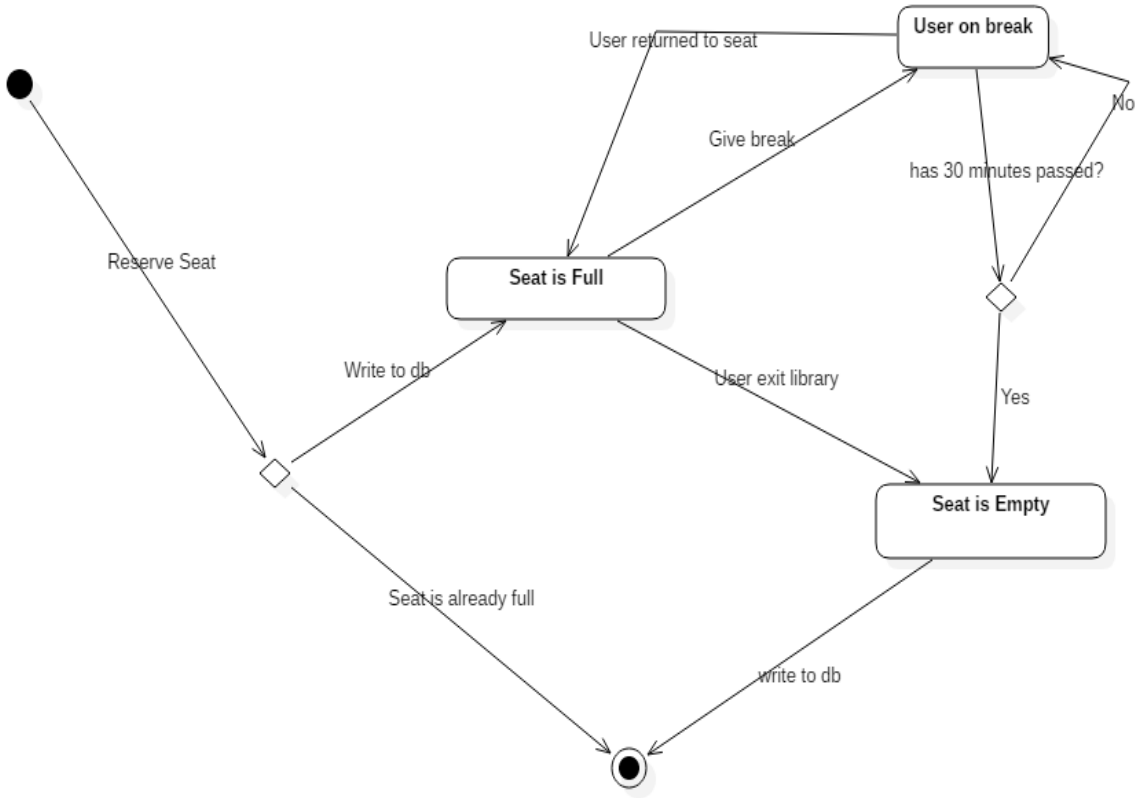

Figure 5: State Transition Diagram for Show User's Break Time Information

Use case name	Location Information of All Rings
Actors	Geo-Location API, Web Application
Description	When a user requests ring location, Web application fetches the coordinates of rings from server, and sends the data to Geo-Location API to get the location of rings on map.
Data	Rings' coordinates.
Preconditions	-
Stimulus	Show location of rings
Basic Flow	1-User requests the data from Web application. 2- Web application fetches coordinates of rings. 3- Web application sends the coordinates to the API. 4-Location data sent back to the Web application. 5- Web application sends the locations to the user. 6-User see the map with position of ring with constantly updated information and time left to reach the station.
Alternative Flow	-
Exception Flow	If database connection is lost, error message will send to the user. Developers will be informed. In addition, error tickets are sent to the IT staff.
Postconditions	-

Table 9: Location Information of All Rings

3.3 Usability Requirements

- Every IoT devices should be able to read users ID Card.
- In library, IoT device should be have a gui to show available seats and books.
- In library, IoT device should be able to provide to user a gui to choose a seat or book to reserve.
- In web application, exact ring locations must be seen in a map (or image) of METU campus.
- In web application, a search utility should be provided to get specific ring's location.
- In web application, a course page that instructor can see attendance information of students should be present.
- Users should be able to use system if and only if their ip is METU ip, meaning that they connected to the internet provided by METU.
- IT Staff should have a database manipulating interface(any program available can be used) to handle data related errors.

3.4 Performance requirements

- System should be able to handle minimum of 50000 request asynchronously.
- Latency should not be more than 1000 ms in reservation operations and ring tracking operations since they are real time, however other operations does not need low latency.
- Cloud server must have at least 25 Gbps bandwidth and 1 Gbps upload/download speed.

3.5 Logical database requirements

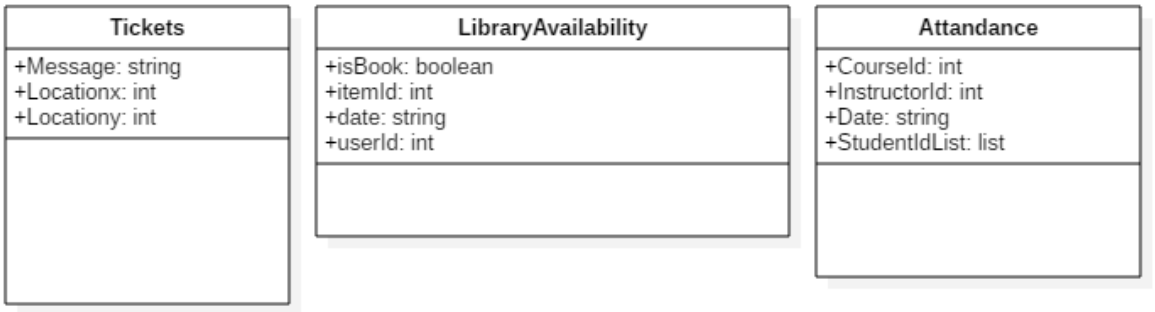

Figure 6: Database Logic Diagram

- Only IT Staff can access all table.
- Instructors can get attendance data.
- Attendance will be written in each class when students show their cards to IoT devices before and after the class hours.
- Tickets will be held in database till IT Staff reads them.
- Library availability data will be dynamic. If a seat is reserrverd, it will be written. If it is freed, it will be deleted from database.

3.6 Design constraints

- Only data stored will be users Metu ID's in attendance and availability of books & seats in library, which will be used to fetch data from METU Servers if necessary.
- The system will be designed according to law of privacy.

3.7 Software system attributes

Reliability:

- System should back up the data every night, at 00:00 am.
- Other than METU ips, any user should be able to connect to system. However, when a user tries to connect with METU ips and fails, IT Staff should handle the error, in less than 1 hour.

Availability:

- System will be available except the maintenance time for METU ips only.

Privacy & Security:

- No email, password or personal info will be stored. Only METU ID will be hold, and METU ID will be used to fetch personal info from METU Servers if necessary.

Maintainability:

- Whole project should be well documented and written as microservices so in any case of need to change, It should be easily changed.

Portability:

- Any device capable of opening web browser can access to web application, however in attendance case, user must be present in close to IoT devices.

3.8 Supporting information