TSU H & N

Help and Navigate

Purpose and Insight?

TSU H & N is basically an application that is designed to help (as self explained) the students of TSU navigate (as the name suggests) around the city and during their educational stay in Russia as a whole.

A very close application and similar in usage may be the already well established TSU time, which basically has some minimal functions. Well the main difference that would be acknowledged at this point is the fact that TSU time can help you with mainly academic purposes. TSU Help and Navigation helps with both academic and social issues to TSU students. Furthermore it can be noted noted that it has a B2B (back to back) usage.

Lastly the TSU H and N basically has three main participants :
- Students
- Guides
- Lecturer Representatives

Functional and Non-functional Requirements

Usability - basic interface of the application

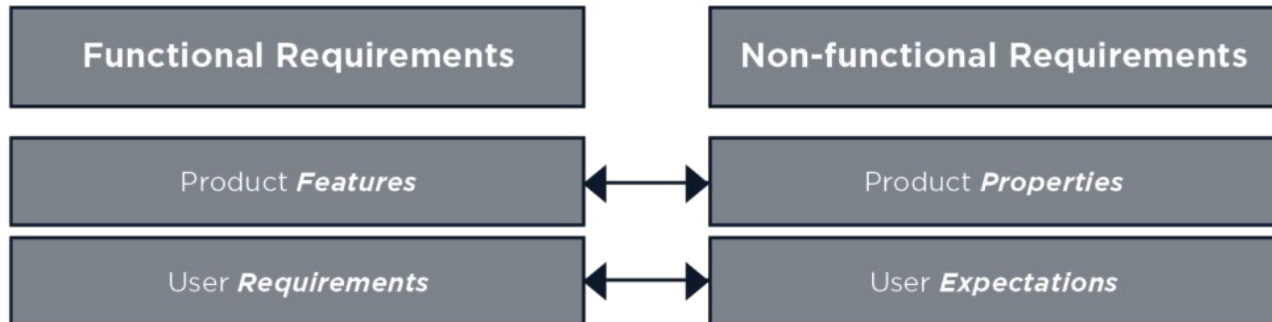**Reliability/Availability**- it needs to work 24/7

Performance- how fast will it work(transform messages and HELP)

If a user has not changed their password for 56 days, then the system shall require a password change upon login

All web pages shall load within 4 seconds

The background colour for all screens shall be #fff4b6.

System programming shall not use deprecated code

Functional Requirements		Non-functional Requirements
Product *Features*	⟷	Product *Properties*
User *Requirements*	⟷	User *Expectations*

Tools and Framework

Mobile Support and Web Support:

❖   Android Studio
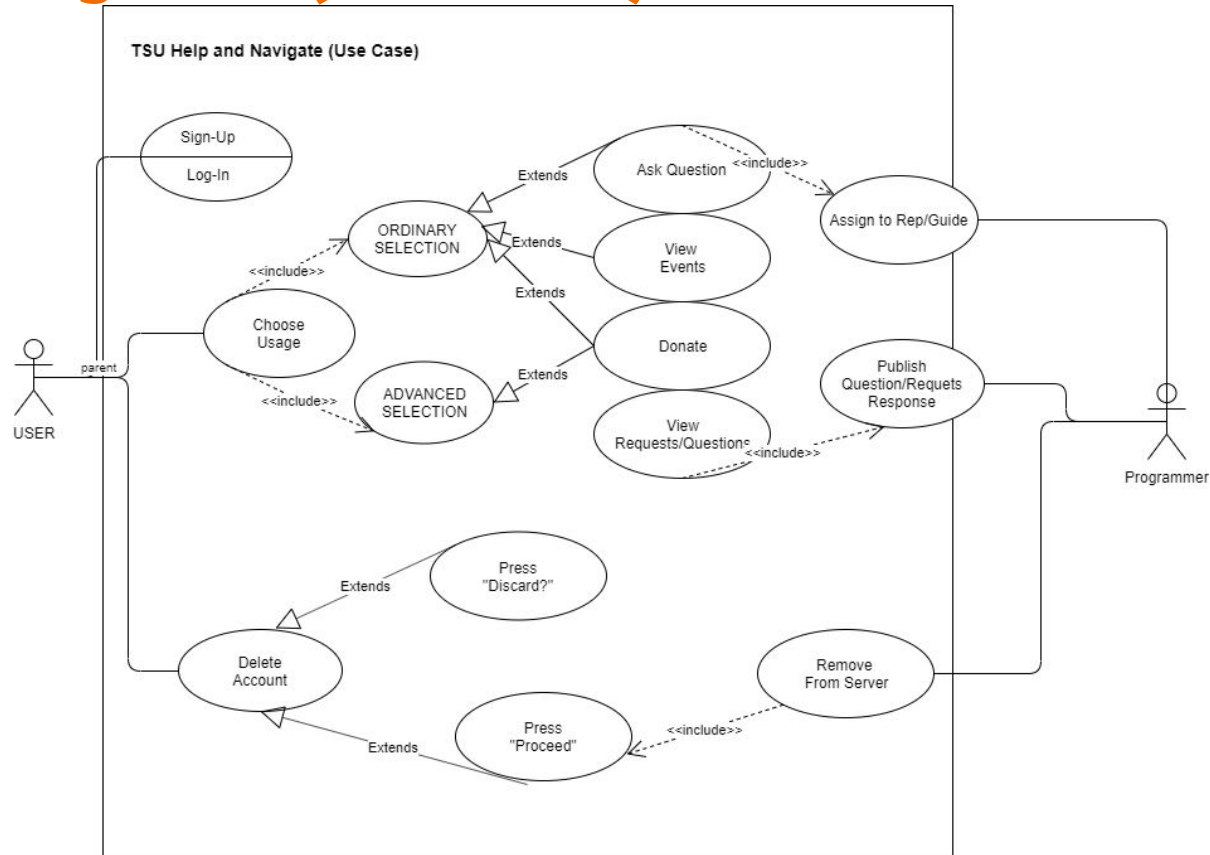❖   Visual Studio
❖   ASP.Net
❖   Buildfire.js

Framework:

Framework 7-Framework 7 used to be iOS only, but now offers Android support as well.
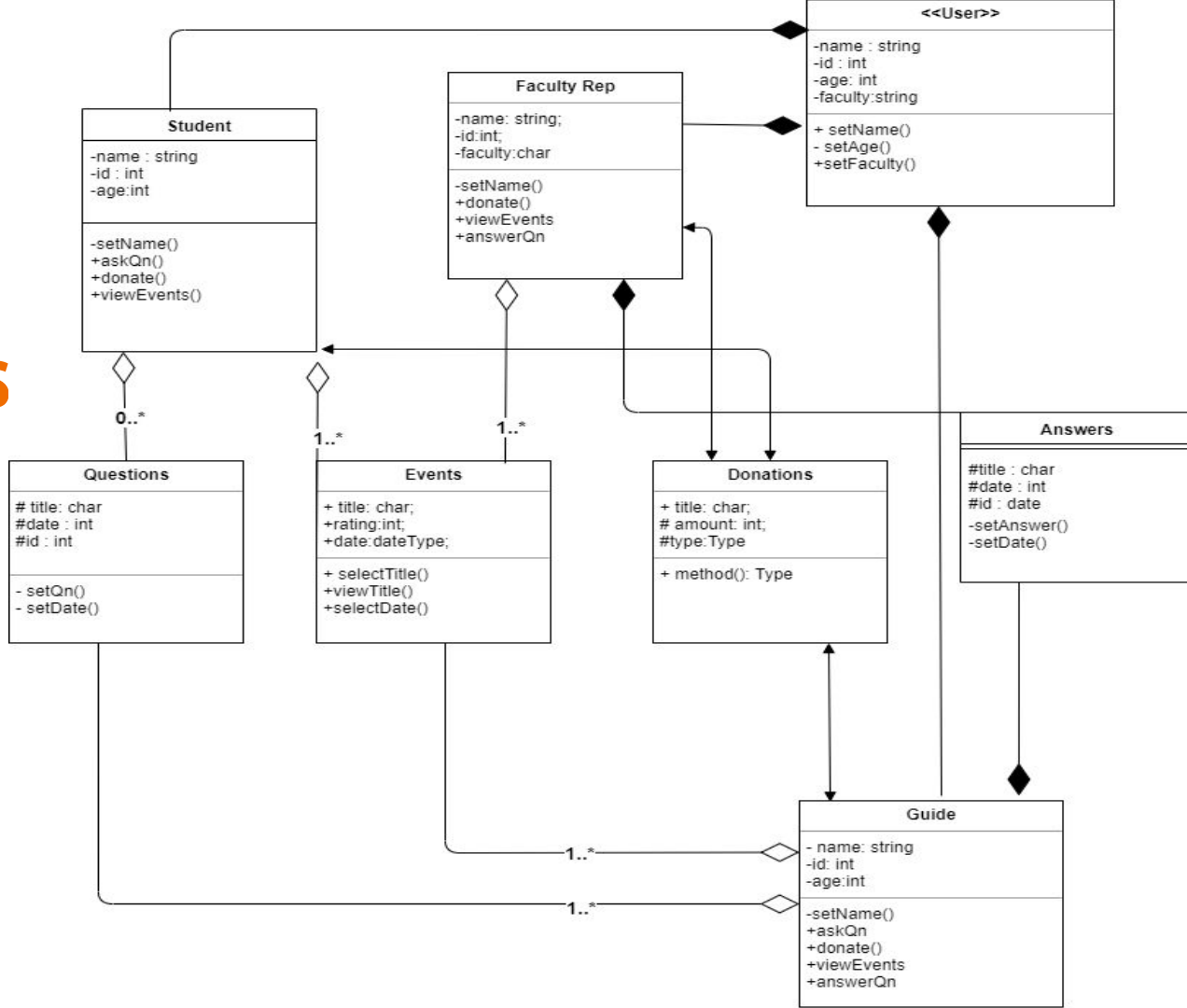
LANGUAGE:

❏   Java

UML Diagrams (Use Case)

TSU Help and Navigate (Use Case)

UML Diagrams(Class-Diagram)

<<User>>

-name : string
-id : int
-age: int
-faculty:string

+ setName()
- setAge()
+setFaculty()

Student

-name : string
-id : int
-age:int

-setName()
+askQn()
+donate()
+viewEvents()

Faculty Rep

-name: string;
-id:int;
-faculty:char

-setName()
+donate()
+viewEvents
+answerQn

Questions

title: char
#date : int
#id : int

- setQn()
- setDate()

Events

+ title: char;
+rating:int;
+date:dateType;

+ selectTitle()
+viewTitle()
+selectDate()

Donations

+ title: char;
# amount: int;
#type:Type

+ method(): Type

Answers

#title : char
#date : int
#id : date

-setAnswer()
-setDate()

Guide

- name: string
-id: int
-age:int

-setName()
+askQn
+donate()
+viewEvents
+answerQn

0..*
1..*
1..*
1..*
1..*

Glossary

Guides- these are people who basically help others with their way around (could be students too)
Functional Requirements-basic system behaviour. Essentially, they are **what** the system does or must not do, and can be thought of in terms of how the system responds to inputs.
Non-Functional Requirements- do not affect the basic functionality of the system (hence the name, *non*-functional requirements). Even if the non-functional requirements are not met, the system will still perform its basic purpose
UML- Unified Modeling Language, general puprpose develpmental , modeling language in the field of software engineering.