Homework: Domain Facets
Prof. Marko Schutz

Exercise 11.1 *Intrinsics.* For the fixed topic, selected by you, identify and describe

1. some intrinsic entities,
2. some intrinsic functions,
3. some intrinsic events and
4. some intrinsic behaviours.

Intrinsics for the Domain: University

scheme Uni =
class
type
Campus, Student, Professor, Class, Classroom, Test, Grade, Dept
value
//Observe the number of students given the classroom assuming every student is unique.
obs_Student: Classrom —> Student**-set**
//Observe the list of Professors given the department and the class they offer. This could be useful when searching for classes.
obs_Prof: Dept x Class —> Professor*****
// Obtain the set of classrooms given the class
obs_Classroom: Class —> Classroom**-set**
// Given the Student, the professor and the class obtain the set of test he/she has taken
obs_Test: Student x Professor x Class —> Test**-set**
//Observe the grade given the student and the test he/she took
obs_Grade: Student x Test —> Grade
//Observe the set of Departments in a campus
obs_Dept: Campus —> Dept**-set**
//True if the campus is empty,empty, false otherwise
is_empty: Campus —> **Bool**
//True of the student belongs to class, false otherwise
belongsToClass: Student —> Bool
//Checks whether the student has failed or not
fails:Student —> Bool
//Checks whether the student has dropped a specific class or not
has_dropped: Student x Class —> Bool

axiom
∀ s:Student, p:Professor , c:Classroom, t:Test, cr:Classroom, d:Dept, cap:Campus
// if a student a dropped it is always true that he does not belong in the course
has_dropped(s,c) —> belongsToClass(s) ≡ **chaos**
// The campus is never empty (I'm making this up)
is_empty(c) ≡ **chaos**
// If a student fails it implies he is no longer in a course

fails_dropped(s) —> belongsToClass(s) ≡ **chaos**

// There is a professor for all classes in the campus
∃ p: Professor ∀c: Class ∈ cmp:Campus

//Describing events and behaviors

Events:

- The event where a student receives his grade and faints.
- The event where a professor enters the classroom
- The event where a hurricane strikes and there are no classes
- The event where a student receives a homework or a test.
- When a student fails a test of a particular class he or she is taking. Such an event occurs when the student takes an exam and was not prepared for such an exam.Both the student and the professor are the main cause of the event.
- A student receives funding from federal aid. This happens when a student is confirmed to be enrolled in several classes and a transfer of money is approved.
- The event that classes have been cancelled. This event occurs when the student or professor is notifies that classes have been cancelled for a specific event for the day. The main cause of this event is usually another event within the university system
- The event that a professor asks a question to the class. This could occur for several reasons. Usually it is due to lack of student understanding or involvement in the professor class.

Behaviors:

- A professor wants to assign a grade to a student using the grading system provided by the university. The system will interact with professor by providing signal that his or her task has been completed
- A student searching for a research fellowship. A student will be searching for such fellowships and making decisions based on the information provided.
- A student taking an exam. A student will try to will behave according to the rules of the university when taking the exam. A proctor will be present to ensure this behavior is being followed.
- A professor during office hours. A professor shall uphold ethics behavior established by the university. He or she must be as respectful to the student and try to help him or her as much as possible as established by the university.

Exercise 11.2 *Business Processes.* For the fixed topic, selected by you, identify and describe two ("as different as is reasonable") business processes.

// Business Processes Schema

scheme La Interamericana = **extend** Uni **with**
class
type
Administrator, Secret, Tuition

//By administrator we mean the person in charge of administrating something within the university domain.
// Similarly the secretary, we denote a person in charge with some business and documentation aspects of the university domain.

value
//Given the student and the administrator, he or she charges the tuition to the student
charge: Administrator x Student —> Tuition
//Given the student, this function tells you whether the student is in debt
is_in_debt: Student —> **Bool**
// Checks whether the student paid tuition
payed_tuition: Student x Tuition —> **Bool**
// Amount of money needed in the loan
get_loan: Student —> **Nat**

// Behavior description

Behaviors:

- The university will often charge a student a tuition, he or she has to somehow pay for this. The student will receive a bill(either electronic or physical) from the university, with the amount to be paid.
- The student will often search for a student aid or a fellowship, such as to help him subsidize the tuition.
- If the student doesn't find any studentship he will most likely have to get a loan. In order to get a loan, a student must contact a loan provider and then pay the tuition.
- Once all transactions have been done, the student can the take classes.

Exercise 11.3 *Support Technologies.* For the fixed topic, selected by you, identify and describe two ("as different as is reasonable") support technologies.

Rough Sketch

- **Payment System**
  - In the old days, universities kept the payment logs in an old fashion booklet. This used to be manual labor, a tedious, expensive, error prone process
  - The new technology uses database management system to keep better track of the process. Using a DBMS allows the university to have the same functionality of the payment logs. However, using this technology the university can lower costs, minimize risks, errors and the system become cheaper. The fact the university, has this technology also allows for tracking system which keeps track of all changes made to the management system

- **Online classes**
  - The traditional way of teaching in the university hasn't changed much, when compared to the number of years universities have existed.
  - The internet has allowed the university to offer a new way to impart teaching. Online classes, allow the student and the professor to be anywhere in the world and still be able to receive/impart teaching. This is beneficial for all parties because, the student and the

professor have greater flexibility and the university can scale the number of students it can fit into the “classroom”, thus increasing the profits of the university.

Exercise 11.4 *Management and Organisation.*

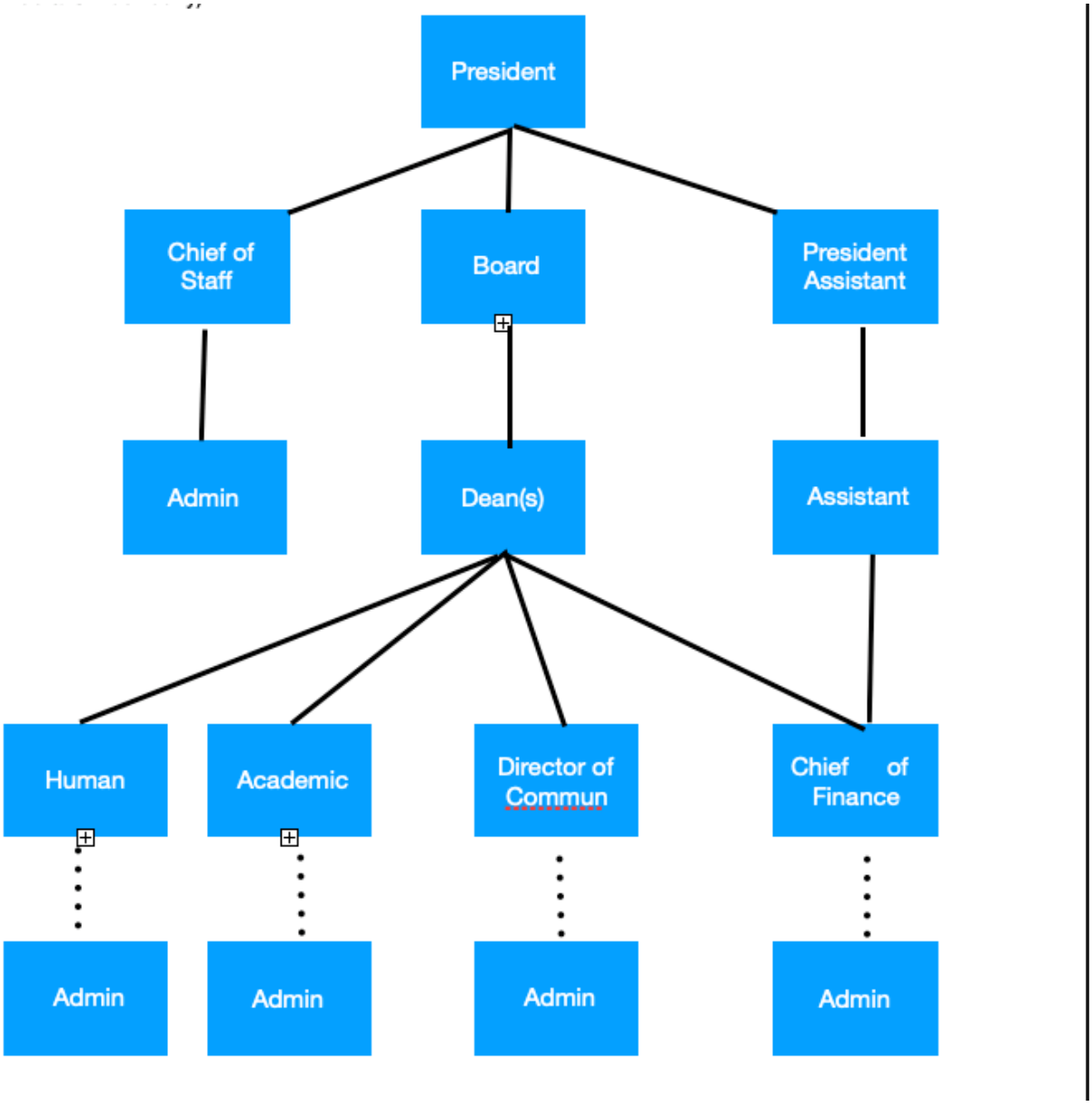
1. For the fixed topic, selected by you, identify and describe management entities, functions, events and behaviours.
2. Identify and describe a possible organisational structure of your chosen domain.

1. First Part

// Management Schema
scheme UPR = **extend** Uni **with**
class
type
Administrator, Secret, Dean, Report, President
//By administrator we mean the person in charge of administrating something within the university domain.
// Similarly the secretary, we denote a person in charge with some business and documentation aspects of the university domain.
// The Dean we denote a high ranking person in charge, within the university, of a department in the U.S.
// By president we denote the head of the entire university system under study
value
//Send a management report
send_mgmt_rpt: Administrator —> Report
// **Approve the report**
approve: Report —> **Bool**
// Send financial summary report to the Dean of the Department
send_fin_dean: Dean —> Report
// Send overall report to the President of the University
send_overall_rpt: President —> Report
// Behavior description

Behaviors:
- The Administrator will perform several queries on financial and institutional data. He/She will generate a report that includes the latest details of each subcomponent
- The administrator will also send various reports to the dean and other high level officials within the university system.
- With the knowledge of the report, the administrators will make decisions using their sound judgment and for the benefit of the university (they rarely do this)
- The dean’s of the departments as well as other high level officials will send reports to the president of the university.
- He/She will make decisions of management with other high level officials to maintain the university as organized as possible.

2. Second Part
// Dotted lines mean there is probably some other management professional in between. This depends on the university.
//Organizational structure of the Domain. (Note this is a rough sketch I might get some of the details incorrectly)



Exercise 11.5 *Rules and Regulations.* For the fixed topic, selected by you, identify and describe three to four rules and corresponding regulations.

- **Cheating on exams and plagiarism**
  1. **Rule:** Any person within the university cannot cheat on exams, homework, project or any other assignment where a grade will be handed.
  2. **Regulation:** If a person is caught cheating or committing plagiarism, he or she will be charged with plagiarism if so. If he/she is caught cheating in some other form the student will enter a probation or be expelled (this depends on university law)

- **Tuition**
  1. **Rule:** A student must have payed tuition in other to be properly enrolled in the courses he/she selected.
  2. **Regulation:** If the student hasn't payed his tuition by the end of the deadline the system will automatically drop the student from the enrollment list. The student will also be notified that such action has been taken.
- **Harassment Policy**
  1. **Rule:** No person within the university shall harass another, under any circumstance.
  2. **Regulation:** If a person is found harassing another, he or she shall be referred to the state authorities to proceed with charges. If found guilty the person shall be expelled from the university. A formal apology on behalf of the university shall be given to the harassed person.
- **Grading Policy**
  1. **Rule:** The professor of the course shall have complete freedom to establish the leading policy and grade distribution. The university will provide and encourage the use of a standard grading system. The grading policy must be established on the first 2 weeks of the semester.
  2. **Regulation:** If a student thinks the professor did not follow the policy, he or she has the right to summit a case to student affairs to review the case. If this is so the final grade will be the one established by university law under such circumstances.

Exercise 11.6 *Scripts.* For the fixed topic, selected by you, identify and describe a possible script language (hint at a syntax, and rough sketch or narrate a semantics).

I will first try to describe the script informally and the use RSL for the formal script.

Informal:

- At any time a "full time" student shall have at least 12 credit hours enrolled for the semester.
- A professor must either be a teaching professor, a research professor, or both. A professor must report the distribution of tasks he/she will be doing for the semester
- A professor will be the one in charge of creating, handing and grading the tests.
- If a student receives a grade lower than 25 % he/she will be automatically dropped from the course.
- A student will be able to drop and enroll courses.

Formal (Using RSL):

scheme Uni =
class
type
Student, Professor, Class, Classroom, Test, Grade
value

//True if the campus is empty,empty, false otherwise
is_empty: Campus —> **Bool**
//True of the student belongs to class, false otherwise
belongsToClass: Student —> **Bool**
//Checks whether the student has failed or not
fails:Student —> Bool
//Checks whether the student has dropped a specific class or not
has_dropped: Student x Class —> **Bool**
// Checks whether a student is a full time student or not given the number of credit
full_time: Student x **Nat** —> **Bool**
//Checks if the student must be forced dropped from the course.
force_dropped: Grade x Student —> Bool
//Drop the course.
drop: Student x Class —> Student

axiom
∀ s:Student, p:Professor , c:Classroom, t:Test, cr:Classroom, d:Dept, cap:Campus
// if a student a dropped it is always true that he does not belong in the course
has_dropped(s,c) —> belongsToClass(s) ≡ **chaos**
// The campus is never empty (I'm making this up)
is_empty(c) ≡ **chaos**
// If a student fails it implies he is no longer in a course
// If a student has 25 or greater number credits it true he is a full time student
full_time(s,n) [n > = 25]
fails_dropped(s) —> belongsToClass(s) ≡ **chaos**

// For every classroom, there is a grade and student there will be forced to drop the course.
∃ s:Student, g:Grade ∀cls: Class
force_dropped(g,s)

Exercise 11.7 *Human Behaviour.* For the fixed topic, selected by you, identify and describe:

1. specifically desirable human behaviours, and
2. specifically undesirable human behaviours.

1. Informal Description of desirable human behavior.
   - When a student gets a HW, we expect him to turn an original work
   - The student will log in to the system and pay his or her tuition before the deadline
   - Harassment by any person within the university will be reported to the relevant personnel.
   - All course projects should have a reasonable time of completion, if this is not the case an extension of the deadline will occur.
2. Informal Description of undesirable human behavior:
   - When a students gets a homework, project or report he/she hands in plagiarized work
   - The students has to go through late tuition payment procedures because he missed the deadline.
   - If a person commits some form of harassment and the person fails to report the incident to the authorities.
   - If the project is really complex and the professor decides not to extend the deadline. Go out and cry.

3. Formal Description for both undesirable and desirable behavior. (In General)

scheme Uni =
class
type
Student, Professor, Class, Classroom, Test, Grade, Claim, Report, Tuition, Delinquent, Sloppy
student_claim == mkClaim(s:Student, c:Claim, p:Professor)
plagiarism_claim == mkClaim(s:Student, c:Claim, p:Professor)
sloppy_claim == mkClaim(s:Student, sup;Sloppy)
value

//True if the campus is empty,empty, false otherwise
is_empty: Campus —> **Bool**
//True of the student belongs to class, false otherwise
belongsToClass: Student —> **Bool**
//Checks whether the student has failed or not
fails:Student —> Bool
//Checks whether the student has dropped a specific class or not
has_dropped: Student x Class —> **Bool**
// Checks whether a student is a full time student or not given the number of credit
full_time: Student x **Nat** —> **Bool**
//Checks if the student must be forced dropped from the course.
force_dropped: Grade x Student —> Bool
//Drop the course.

drop: Student x Class —> Student
//If there is claim of plagiarism, then the student is a delinquent.
delinquent_student: plagiarism_claim x **Bool —> Bool**
//Project is hard
hard: student_claim —> **Bool**
// Extend deadline
extend: student_claim —> **Bool**

Exercise 11.8 *A Comprehensive Domain Description.* For the fixed topic, selected by you, collate the descriptions that you have produced in answers to Exercises 11.1–11.7 into one comprehensive domain description.

The university domain is said to be composed of entities, functions, behaviors, etc. We start this description by describing the domain intrinsics. The domain intrinsics for this domain will be, and perhaps rather obvious, are Students, Professors, Departments, Campus, Lecture, Reseach, Course, Test, etc. Such intrinsics are at the core of what every university has or is. The domain will have, more or less, the following behavior Behaviors:

A professor wants to assign a grade to a student using the grading system provided by the university. The system will interact with professor by providing signal that his or her task has been completed A student searching for a research fellowship. A student will be searching for such fellowships and making decisions based on the information provided. A student taking an exam. A student will try to will behave according to the rules of the university when taking the exam. A proctor will be present to ensure this behavior is being followed. A professor during office hours. A professor shall uphold ethics behavior established by the university. He or she must be as respectful to the student and try to help him or her as much as possible as established by the university.

The Domain will have several rules and regulations in place. Some of the rules include, include not cheating in exams, not committing plagiarism in homework. For any in the university and outside this domain there are rules for harassment and corruption. Under this domain, such actions are punishable by law. The university will also establish several regulations on the case such rules were to be broken. Many of the rules and regulations are subject to human behavior within the domain. If we obtain undesirable human behavior, thus and most likely a set of rules will be broken. Another undesirable human behavior would be the lack of action or regulation when such rules have been broken. In the optimal domain, we could create software to ensure that everything works with the desired behaviors and minimize the need of human intervention. If human intervention were required our domain software implementation should also aid in such a process.

Log

Time Log

13/Sep/2018

date	start	stop	errupt	net	act	comment	mpleted	nits
10-Sep	7:25	9:00		95	repare	read news, breakfast		
	9:00	9:30		30	park	find parking space		
	9:30	0:20	10	40	class	re and waisting time on twitter		
	0:30	2:10		40	eat	lunch		
	2:30	1:20		50	class	lecture		
	1:30	3:00	10	80	search	read assigned papers	x	2
	3:00	5:00	30	90	study	tructions for HW3, break, phone		
	5:00	6:00	30	30	study	read ch3	x	1
	6:00	7:30		90	class	lecture		
	7:00	8:30	20	70	prog	esearch team & chat with team	x	2
	8:40	1:30	20	50	study	quiz prep, chat, leisure	x	1
date	start	stop	errupt	net	act	comment	mpleted	nits
11-Sep	7:25	8:30		55	repare	read news, breakfast		
	8:30	8:55	10	15	park	rking time and chat with friends		
	9:00	0:20		80	class	lecture		
	0:30	2:20		10	search	research meeting		
	2:30	1:15		45	eat	lunch		
	1:15	4:30	30	65	study	uiz prep, began reading chapter	x	1
	4:30	4:55	15	15	prog	read requirements for project		
	5:00	6:30		90	class	lecture		
	6:40	7:20		40	eat	supper		
	7:30	0:40	40	50	study	read ppt before class		
	0:50	2:00		70	ercise	go for a run		
date	start	stop	errupt	net	act	comment	mpleted	nits
12-Sep	7:25	9:00	10	85	repare	read news, breakfast		
	9:00	9:25		25	park	find parking space		
	9:30	0:20	2	48	class	lecture		
	0:30	2:05		95	eat	unch with friends and colleagues		
	2:06	2:25	10	9	prog	arch code and chat with friends		
	2:25	1:27		62	class	lecture		
	1:30	4:35	30	55	search	meeting and programming		
	4:35	4:50		15	eat	offee and waist time on twitter		
	4:50	6:10		80	study	prep, read notes before lecture	x	1
	6:30	7:20		50	class	lecture		
	7:20	0:30	20	70	prog	HW	x	1
	0:30	2:00	5	85	ercise	go for a run		
date	start	stop	errupt	net	act	comment	mpleted	nits

Weekly Activity Summary

week #	Task Date	Class	Prepare	Park	Eat	Study	Prog	Research	Exercise
2	M	180	95	30	40	270	70	80	0
3	T	170	55	15	85	315	70	110	70
4	W	160	85	40	15	80	179	155	85
5	T	170	65	20	40	270	70	0	69
8	Totals	680	300	105	180	935	389	345	224
9	Average	170	75	26.25	45	233.75	97.25	86.25	56
10	Min	160	55	15	15	80	70	0	0
11	Max	180	95	40	85	315	179	155	85

Proposed Schedule for New Tasks

Time	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
7:00							
8:00							
9:00							
10:00	Code Res		Code Res		Code Res		
11:00	Code Res		Code Res				
12:00	HW	HW	HW	HW	HW		
13:00	HW		HW		HW		
14:00							
15:00		HW		HW			
16:00							
17:00							

Categrory Percentages

Total Est Hr	Time	Class	Prepare	Park	Eat	Study	Prog	Research	Exercise
3980	Total	680	300	870	180	935	389	345	224
	Percentage	17%	8%	22%	5%	23%	10%	9%	6%