RAD

1. Introduktion

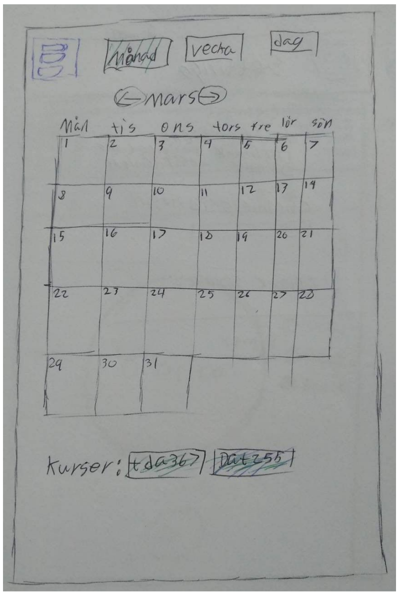
// En mobilapplikation med syfte att hjälpa studenter att hålla koll på deadlines och vara produktiva, genom att ha en lättöverskådlig bild av kurserna och kursmomenten samt hjälpmedel som att bocka av moment, mäta nedlagd tid m.m

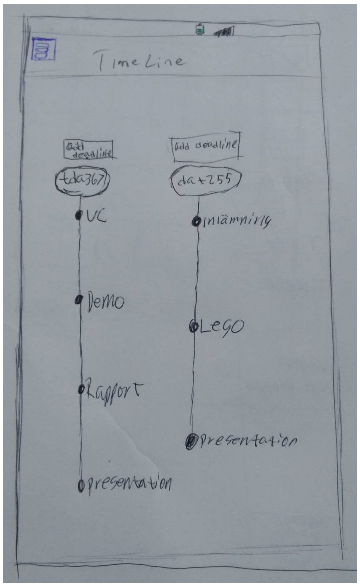
1.2 Definitions, acronyms and abbreviations

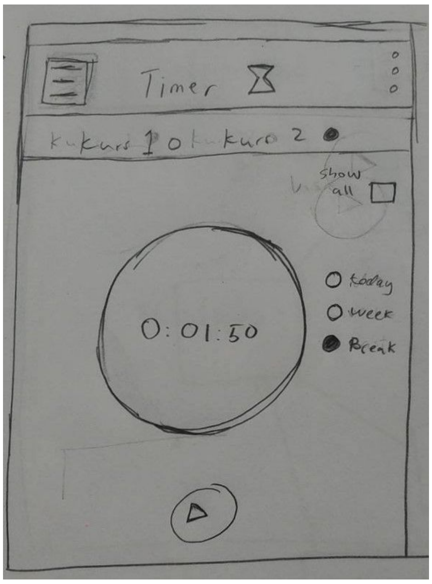
2 Requirements

2.1 User interface

The user interface will consist of separate views, mainly a calendar, timer and a timeline.







2.2 Functional requirements

Schedule view:

- The user shall be able to import schedule for each course
- The user shall be able to add event to schedule
- The user shall be able to view more information about event

- The user shall be able to edit event
- The user shall be able to remove event from schedule

Timeline view:

- The user shall be able to mark deadline as finished
- The user shall be able to view the users progress of the course on timeline

Timer view:

- The user shall be able to add a weekly goal with a given time to study for each course
- The user shall be able to set a time interval for study breaks
- The user shall be able to start and stop timer for each course

(Optional)

- The user shall be able to see an overview for his/her way to graduation
- The user shall get notifications as reminders

2.3 Non-functional requirements

- 9/10 novice users should be able to add an event
- 9/10 novice users should be able to add a deadline
- 9/10 novice users should be able to import a schedule
- 9/10 novice users should be able to view more information about an event or deadline
- 9/10 novice users should be able to start the timer
- 9/10 novice users should be able to navigate in the side bar
- 8/10 novice users should be able to set a weekly goal
- 8/10 novice users should be able to set a time interval for breaks
- App should open and be functional within 3 seconds after user starts app
- App should crash in max 1 out of 15 usage sessions

3 Use cases

3.1 Use case listing

UC1: Import schedule

Summary: The user can import schedule from timeedit for a given course
Priority: High
Extends:
Includes:
Participators: User

Normal flow

	User	System
1	User tells app which course to import	
2		System searches for course and fills calendar with course schedule

(1-2): Alternate flow - Invalid course name

	User	System
1	User tells invalid course name	
2		System returns error message

2: Alternate flow - Course doesn't exist

	User	System
2		System notices course name does not exists and returns error message

UC2: Add event to schedule

Summary: The user can add additional events to schedule, either within a course or stand-alone.
Priority: High
Extends:
Includes:
Participators: User

Normal flow

	User	System
1	User navigates to event form	
2		Presents event form
3	User gives name, date, starttime, endtime for event Optional: Description	
4		Adds event to schedule

(3-4)a: Alternate flow - Add deadline

	User	System
3	User gives name, date and deadline for event Optional: Description	
4		Adds deadline to schedule and timeline

4a: Alternate flow - Event collision

	User	System
4		System notices collision in schedule and returns warning

(3-4)b: Alternate flow - Invalid date

	User	System
3	User gives invalid date, starttime or endtime	
4		System returns error message

UC3: Remove event

Summary: The user can remove an event from the schedule
Priority: High
Extends:
Includes:
Participators: User

Normal flow

	User	System
1	Navigates to event	
2		Presents event details
3	Removes event	
4		Removes event from schedule

UC4: View event information

Summary: The user shall be able to view more details about an event in the schedule
Priority: High
Extends:
Includes:
Participators: User

Normal flow

	User	System

1	User navigates to event	
2		System presents event details

UC5: Mark deadline as finished

Summary: The user shall be able to mark a deadline as finished and thereby see his/her progress in the course
Priority: High
Extends:
Includes:
Participators: User

Normal flow

	User	System
1	user navigates to deadline	
2		System presents deadline details
3	User marks deadline as finished	
4		System updates progress on timeline

UC6: Remove deadline

Summary: The user shall be able to remove a deadline altogether.
Priority: High
Extends:
Includes:
Participators: User

Normal flow

	User	System

1	User navigates to deadline	
2		System presents deadline details
3	User deletes deadline	
4		System updates timeline with deadline removed

UC7: Add weekly goal

Summary: The user can add a weekly goal with a given time to spend on each course.
Priority: Medium
Extends:
Includes:
Participators: User

Normal flow

	User	System
1	User navigates to Timer	
2		System presents event form
3	User provides time and course	
4		System updates timeline with new goal

UC8: Set time interval for study breaks

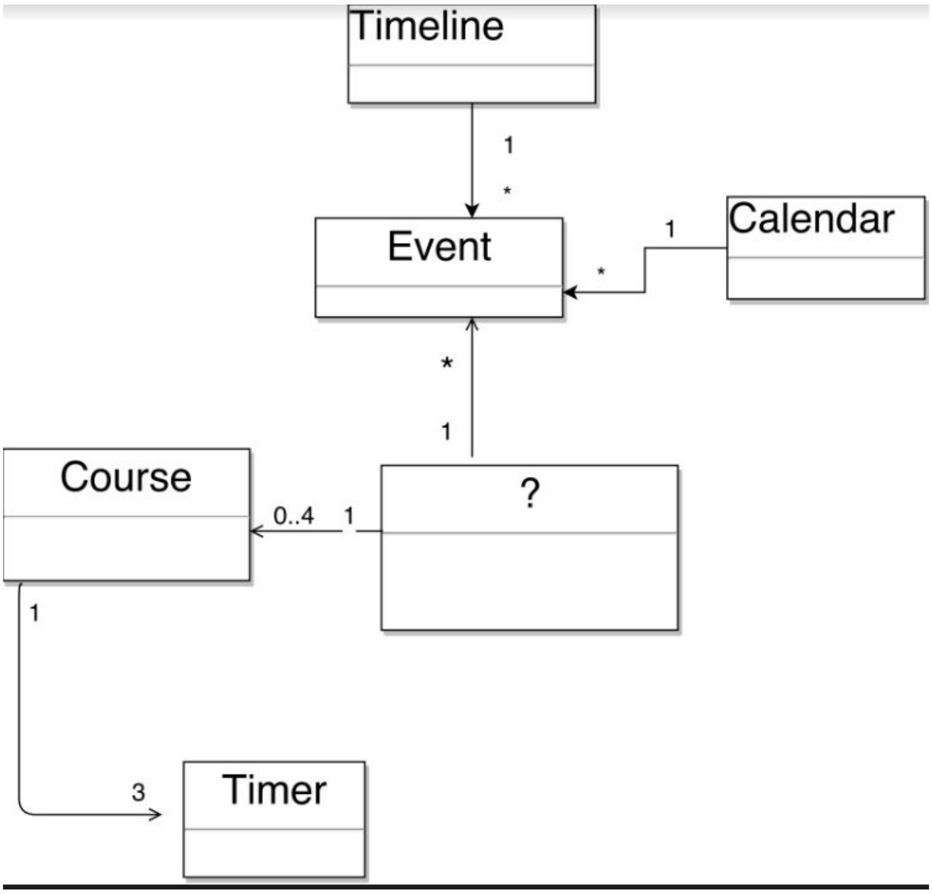
Summary: The user can set up a time intervarl for when to take breaks.
Priority: Medium
Extends:
Includes:
Participators: User

Normal flow

	User	System

1	User navigates to Timer event	
2		System presents current timer
3	User navigates to Settings	
4		System presents Settings options
5	Provides time interval	
6		System updates Timer view with new interval included

4 Domain model



4.1 Class responsibilities

5 References

TimeEdit