

Requirements and Analysis Document for HabitHets

Group 16:

Oscar Helgesson, Tina Samimian, Norbert Laszlo, Elin Nilsson,
Jacob Messinger

2019

Contents

1	Introduction	3
1.1	Definitions, acronyms, and abbreviations	3
2	Requirements	3
2.1	Epic 1	3
2.2	Epic 2	3
2.3	Epic 3	3
2.4	Epic 4	3
2.5	Epic 5	3
2.6	Epic 6	4
2.7	Epic 7	4
2.8	User Stories	4
2.9	Definition of Done	9
2.10	User interface	9
3	Domain model	11
3.1	Class responsibilities	11
4	References	12

1 Introduction

It all begins in the younger years when we start school and therefor also face our first academic challenge. As the years go, we are expected to take on more responsibility and suddenly, the only person we can rely on to make sure we get somewhere is no other than ourselves. There are some who have tried to solve the ongoing question about how to get our lives in order. Our application HabitHets might be the solution to the problem but without being forced to use several applications with different functions but the same goal. HabitHets is your friend. It helps you to collect your events, habits, notes and all of the other things you otherwise would try to keep in your head. With this one application you can structure and ease your life while it helps you to build your future habits at the same time.

1.1 Definitions, acronyms, and abbreviations

User Stories = US

2 Requirements

2.1 Epic 1

As a user, I want to be able to add, remove and edit my habits on my calendar so that I can create new habits i life and reflect how well I preformed.

2.2 Epic 2

As a user, I easily want to be able to create lists of checkable things I will do, without them being attached to my calendar so that I don't forget to do everyday stuff.

2.3 Epic 3

As a user, I want to be able to write done notes so that I can get my thoughts in my HabitHets.

2.4 Epic 4

As a user, I want to be able to add, remove and edit events so that I can structure my days and don't forget important activities.

2.5 Epic 5

As a user, I want to be able to see my day, week and month in my calendar so that I can get a overview of my life and see planed events.

2.6 Epic 6

As a user, I want to be able to change my preferences, such as color theme, so that my application fits my requests.

2.7 Epic 7

As a user, I want to be able to share my calendar och chosen events, habits and to-do's with friends to easier be able to know when we can hang out and also get inspired of each others lives and routines.

2.8 User Stories

Implemented

ID to be set in later stages of RAD iterations

ID X

As a user I want to be able to view an entire month to be able to plan further than just a week and to get an overview of the month.

Acceptance criteria

- Graphically portray a given month.
- Display week numbers and dates.
- Display a dot on days that have an activity.

ID X

As a user I want to be able to create and delete habits on my calendar so that I can have all the habits I want to.

- Graphically portray created habits.
- Be able to save habits.
- Be able to delete habits.

ID X

As a developer, I want to create a view for habits so that habits can be represented graphically given a list.

- Display a view containing the important information of habits.

ID X

As a user, I want to be able to see the streaks of my performed habits so that I can be motivated and continue to do my habits.

ID X

As a user I want to be able to view a week forward, starting from any day of the week of my choosing, to be able to structure my day to day activities.

Acceptance criteria

- Show a week
- Provide option to choose which day to start at, plus the six following days after that.

ID X

As a user I want my notes to be placed in a specific day of the week to be able to swiftly view and put down any thoughts of my liking.

Acceptance criteria

- Graphically portray notes in the day-view.
- Take input from user.
- Save written notes for a later occurrence.

ID X

As a user I want to be able to write a note in order to quickly get my thoughts down on to HabitHets.

Acceptance criteria

- Provide a place from which a note may be entered.
- Display written note.

ID X

As a user I want to be able to add and remove a todo, so that I can be able to see what I need to do.

Acceptance criteria

- Add a todo.
- Remove a todo.
- Update an existing todo.

ID X

As a user, I want to be able to see what day and date it is in order to easily understand where I am in the calendar.

Acceptance criteria

- Show a graphical representation of a day.
- Show the day of the week in close proximity to that day.
- Also show the date of that day.

Not implemented

ID X

As a user I want to be able to change my preferences and settings in order to customize the programme to my liking.

Acceptance criteria

- Create a setting page
- Be able to change things eg. Theme of entire page and colour of events.

ID X

As a user, I want to be able to save a habit by pressing a button on the application so that I can don't lose my habits.

ID X

As a user, I want to be able to edit my habits on my calendar so that I can update my habits as my everyday life constantly is changing.

ID X

As a user I want to be able to save a todo in the GUI, so that I easily can see my todolist.

Acceptance criteria

- Graphically portray todos in the application.
- Be able to write a new todo.
- Be able to save a written todo.

ID X

As a developer I want to create a GUI for the todos, to be able to represent the todo-list.

ID 8

As a user I want to be able to quickly add a todo in the GUI, to be remember my upcoming tasks.

ID X

As a developer i want to create a first version of the week view in the graphical interface of the calendar.

Acceptance criteria

-

ID X

As a user i want to be able to see which day it currently is and other dates so i can plan ahead of time.

Acceptance criteria

- See the current date in the expanded day view.
- See the current week in the week view.

ID X

I as a user want to see a horizontal line in the view that indicates what time of the day it is so I can see what events are upcoming and ongoing at a certain time.

Acceptance criteria

- Have a absolute horizontal line in the interface for the days in the expanded day view and in the week view.
- Have the horizontal line show what time of the day it is

- Have the horizontal lines' height get updated periodically and align to current time.

ID X

I as a user want to have a year view in my calendar so I can have a nice overview of the months in a year.

- Have a dedicated view for the 12 months of a year show up (containing their days), given a year.

ID X

As a user, I want to be able to add events to my calendar and be able to see them so that I can better plan my time.

Acceptance criteria

- Display a button that can be pressed in order to create an event
- Show the event on the right day and at the right time

ID X

As a user, I want to be able to quickly and easily navigate back and forth between different days, weeks, months or years in order for me to more easily use the calendar.

Acceptance criteria

- Display an icon or label that shows what view the user is in
- Show a label that states what day, week, month or year the user is viewing
- Display two buttons that navigates the user back and forth when clicked, like example between months

ID X

As a user, I want to be able to edit and delete events from my calendar in order to update and manage my calendar.

Acceptance criteria

- Display a button that when pressed enables the user to delete events from their calendar
- Display a button on each single event that when pressed deletes the event
- Display a button on each event that makes the event enable for editing and another button to save what has been edited

2.9 Definition of Done

In this section you list the acceptance criteria that are common for all user stories. For example, the code should be reviewed and tested, it should be under version control, etc.

- To be considered done, all US should first be unit tested to a sufficient degree of satisfaction.
- The code should be reviewed by more than one developer independently to decrease the possibility of unexpected behaviour.
- The product owner, in this case our group, should all be acceptant of the US.

2.10 User interface

The application have three main parts, habits, calander and a todo list. These three parts will always be visible in the view.

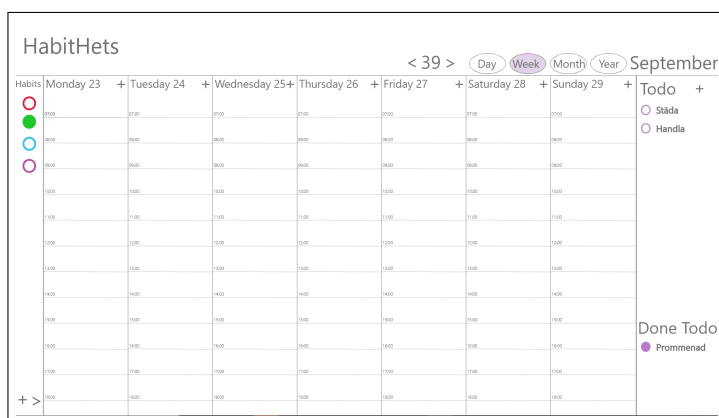


Figure 1: Front page of Habithets.

Figure 1 shows HabitHets frontpage. The center of the application is the main part of HabitsHetz, the calander. When opening the application the startpage shows the current week, to give the user an overview of the upcoming events. The user can easily navigate between weeks by using the arrows in the header.

The user's habits are represented to the left of the application. There are round check-boxes that enables the user to confirm a that a habit is done for the day. Each habit is given a color to easily help the user to recognize what habit it is, without having to get more information from the expanded habit-view. To get more information about the habits, the user have to press on $\dot{}$.

On the right side of the application are a todo-list that shows upcoming tasks. By pressing on the +, a new Todo is created. When a Todo is finished, the user can easily mark the todo as done and it will then be moved to Done todos, where the last five accomplished todos are visibale.

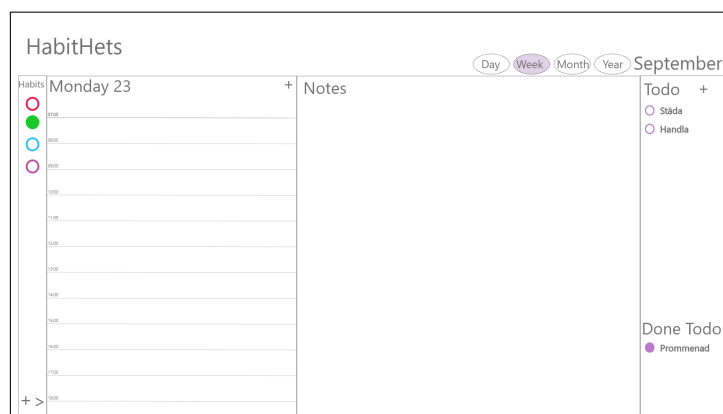


Figure 2: Habithets dayview.

There are four different calendar views, dayview, weekview, monthview and yearview, and there are buttons in the header to change between the different views. The dayview contains the events for that specific day and a notepad where the user can write down thing to remember.

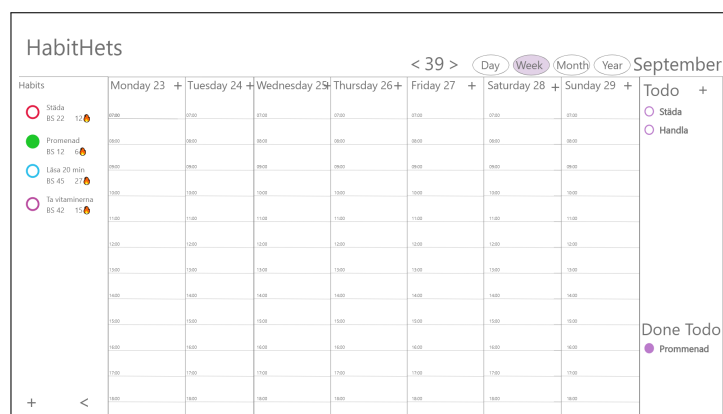


Figure 3: Extended view of habits.

In extended habits the user can see more information about the habits. Every habit has a title, a best streak and current streak. By showing the streak the user will get motivated to continue with the habit.

Include sketches, drawings and explanations of the application's user interface. Describe the navigation between the different views.

3 Domain model

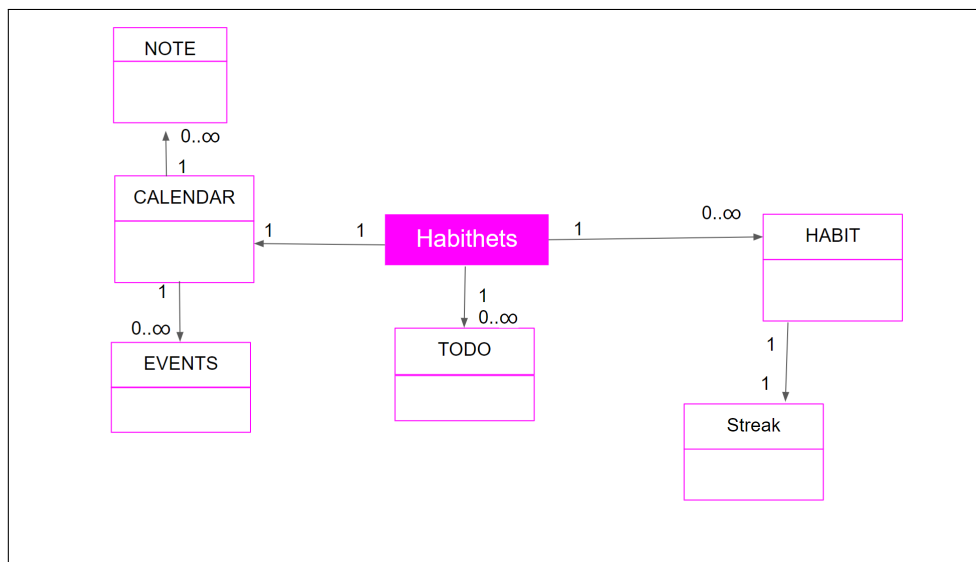


Figure 4: Domain model.

Give a high level view overview of the application using a UML diagram.

3.1 Class responsibilities

The entry point of the application is Habithets, which have the responsibility to connect all parts of Habithets.

Calendar: Calendar will have all of the events and have everything related to a day.

Events: Have a start and end time. Is connected to a day.

Note: Every day will have a note: Contains a string.

Todo: Has a list of Todos, checkable objects, and will have all logic connected to todos.

Habit: Has a list of Habits, checkable objects, and will have all logic connected to habits.

Streak: Every habit will have a streak connected to it. The streak symbolise how many days the habit has been completed. Holds the habits history.

Explanation of responsibilities of classes in diagram.

4 References

List all references to external tools, platforms, libraries, papers, etc.