Kandidat NR: 1044

Emnekode: BU5300

Emnenavn: IT-prosjektledelse

Innleveringsdato: 03.05.23

Individuell hjemmeeksamen

Høyskolen Kristiania Semester våren 2023

Erklæring: Denne Oppgaven er gjennomført som en del av utdannelsen ved Høyskolen Kristiania. Høyskolen er ikke ansvarlig for oppgavens metoder, resultater, konklusjoner eller anbefalinger. Eventuelt påføre om oppgaven skal være konfidensiell.

Health and Fitness App

The Document is 10 pages in total with the front-page

1.

The health and fitness tracking application being built is an information system project. The IS project falls under the category Software Development Project. We have a group of people that define the requirements needed for the project, these requirements are in place so the project can succeed. Then the project will design how our solution will look, both for the mobile application for IOS and Android as well as define how the design will look on the integrated version for the Fitbit and Apple Watch. The development of the project will then be based on the requirements and design. We also must test the software. We are creating this solution for potential customers and our expected end users in the health and fitness market.

The problem with software is often that things are intangible. This means that my idea of some functionality might not correspond to your idea of the same thing. This is often not dealt with in traditional settings, if we create a blueprint of a building, we can both agree on that's how the building should be built. When we try to do the same for software, if I say that we need functionality like daily goals the idea might differ based on whom we ask. Software development shares many of its ideas with traditional project management but differentiates at this specific point. Because of this, a Software Development Project needs flexibility, so we can discuss and change functionalities without much hassle. The changes to the project can come from user reviews or the main investors/funders that don't agree with how things are developed. It's also important in these cases to understand who the customer is and mediate this if it's to benefit the end users.

Even if this project would be put under the software development project category it could differentiate on some key factors:

If the project was not funded and developed by the health and fitness chain and its internal IT department it could be an out-sourcing project, and expertise that the IT department doesn't have can still be out-sourced. The same goes for Consultancy and Business Analysis Assignment project, I assume that the company chain has enough resources to handle market research and IT solutions on its own if needed. If we need to consult another market in the chain, we probably can if needed.

I think the best Development Lifecycle in this case would be based on that we need flexibility to change functionality and requirements.

I would use the Agile Approach as a good way to solve our solution, SCRUM would keep some structure with meetings and sprint reviews while also being flexible. The requirements and design also help the team understand what's needed and keep a certain level of structure. The employees also usually like this kind of work environment. There will also be some bad aspects if the SCRUM teams are not managed right, right now we have allocated 15 developers 5 people in quality assurance and 5 people in design. Splitting these into scrum teams can break cooperation between the designers and those in quality assurance because we need to split into teams of 5-8. It's also easier to get scope creep which means that we make too many changes or try to implement new things that make the scope bigger, resulting in not delivering on time.

Another note-worthy traditional solution could have been the V-model because it's rigid and well documented with quality assurance in place, but it's not flexible at all so the documentation phase of what's needed specifically as requirements would need to be noted.

2.

A Project works within three constraints, Budget, Scope, and Time and the quality resides in the middle of the triangle. If we are to define this project's Budget it is estimated to be 30 million NOK, since this is an estimation, it might be plausible for this number to be a bit higher or lower. Considering the time aspect of the triangle, the project is expected to take 12 months to complete. The scope can vary depending on what kind of solution we implement around supporting both Android, IOS, and integrated systems for Apple Watch/Fitbit but assuming we use a framework that supports both IOS and Android solutions while having to create an app for Fitbit/Apple Watch separate so the user experience is fluid.

And then adhering to the project specifications around the modules for the scope of the project.

Technical skills needed would then be separated into what the project manager needs and what the project needs.

The project manager needs communication skills, project management methodology, risk management, and problem-solving skills, all while tracking and monitoring stakeholders, time, budget, and cost. The project manager should also be knowledgeable about the current field we are working in which is the health and fitness field.

For the employees, we need them also to have communication skills, problem-solving skills around their expertise, should know the health and fitness market somewhat, and innovative skills based on that the market is highly competitive. The designers should provide good designs that go in correspondence with the company's high standards, same goes for good and strict quality assurance. The developers should have skillsets in the frontend, backend, full stack, IOT, databases, security, testing, programming language that the company defines, data structures, and algorithms for a fluid experience, while also having communication, teamwork, and problem-solving skills. Attention to detail in the work is important for any employee considering the company's high expectations.

3.

Our one-time costs would be:

- Initial Software purchase
- Initial Hardware purchase
- Initial Search Engine Optimization
- Initial Web hosting
- Initial API cost

Our reoccurring cost would be:

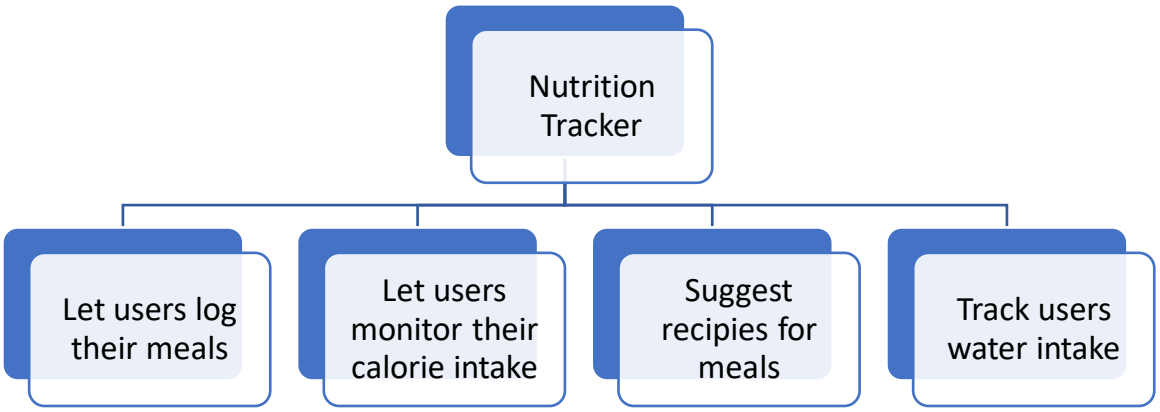
- API cost
- Database hosting
- App hosting

- Software fees
- Hardware maintenance
- Staff payment
- Office renting

One-time costs are often hard to define because it depends on if the project succeeds or not. If we see the use for search engine optimization again it moves to reoccur same goes for web hosting, and some APIs have an initial cost, while some have to reoccur that's why it's in both categories.

Some things might not be reoccurring either based on what the chain already has, if it provides the office the IT department usually uses there is no need for renting a place.

4.



Letting users log their meals, would probably consider having a meal database so the most common meals can be logged without a lot of manual labor.

The same goes for the calorie intake from the meals, the data for the calories will come from the same database. It will also be available to log your meals manually.

Suggesting recipes will be an algorithm developed based on the meals you have eaten.

Tracking the user's water intake will also be based on the workout plan and how active the user has been throughout the day.

5.

Estimation is exactly what it implies, it's an estimate. This means even if you estimate and plan there is no guarantee for the estimation to work out. But to try to calculate this we will be using Story points, since we suggested that this is a SCRUM project.

For letting the user log their meals I would assume we know something because this resembles the most likely already implemented set daily goal in the activity tracker or creating a custom workout from the workout tracker. It will have a few dependencies based on if we are using an internal database, we create from the ground up, or we use some kind of different API to be able to log the meals. And I'd assume this will probably take a week to implement. So based on this average I would give this task 5 story points.

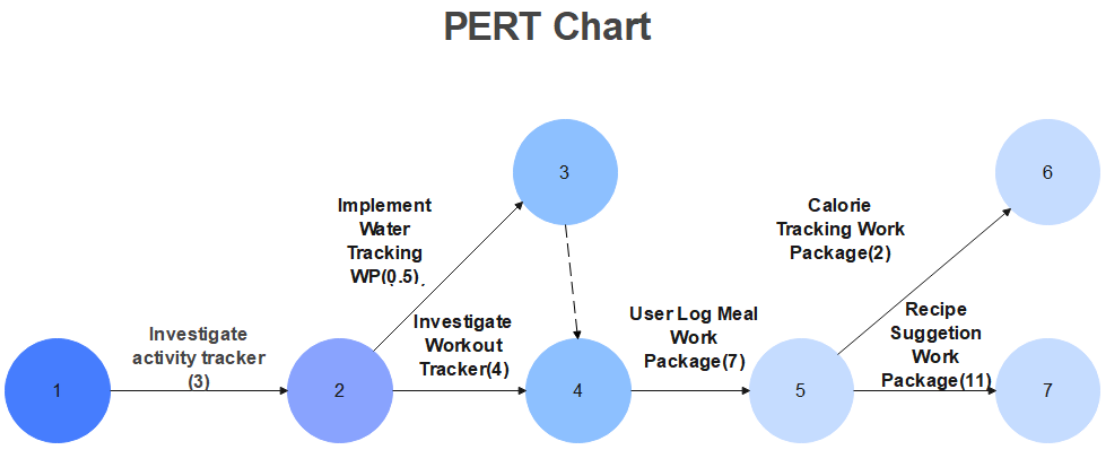
The calorie tracking would rely on the same modules set up for how the user logs their meal so if this is done after the user log implementation, I would say we know almost everything. It almost has no dependencies based on if we use the same systems already created. And it would take up to 2 days. Giving this task 3 story points.

Suggesting users' recipes based on recommendations would be challenging because we probably would need an algorithm based on what the user wants to achieve as their goal, while also considering their eating habits. I'd say we know something, we have more than a few dependencies and it would probably take a week or more to implement this algorithm properly. Giving this task 8 story points and the task should then be further broken down, probably into packages like creating a database/API for recipes and storing user data, and then creating this into an algorithm.

Tracking the user's water intake, we know almost everything. It has some dependencies based on that we probably need to know the activity level of the user to track the water level accordingly. I would say this probably takes half a day. So, I give this task 2 story points.

Based on if we always round the story points up to the worst-case scenario, I believe the first meal logger will take 8 story points, the calorie tracker will still have 3 story points, the recipe algorithm gets 13 story points and the water tracking get 3 story points.

6.



The pert charts critical path is usually what takes the longest, but in this case, the best case and critical path stay the same because we work in parallel. But the critical path would look something like 1->2->4->5->7 which is 3+4+7+11 which is 25 days.

7.

Nonfunctional requirements for performance, availability, security, and usability.

For performance, the page listing meals from the database should load within a second so the user sees relevant entries of the same meal from what he searched. If the user searches for pizza the page should return different types of pizza within a second.

For availability, all systems that rely on databases should have a backup database and an implemented alert on the app that says there are minor issues on the app.

For security all relevant user data should be encrypted, implement a 2-factor authorization check for deleting user data, providing information that user suggestions will be worse.

For usability I want to keep track of how much the work packages get used, if a functionality developed gets used under 10%, I want to keep track of that and maybe remove this, so the end user is happier.

8.

three main key stakeholders to this project with both high interest and high power are project investors/funders, end users which are our customers, and the dev team. More stakeholders fall into this category, like the app store or regulatory bodies, but I will go in-depth on the three mentioned previously.

The project investors/funders are most likely the leading fitness and health chain, so we need to report our progress and have clear project milestones. We need to involve them in key decisions for the project because without funding the project can't take place.

End users are the ones using the app, if they don't see any reason for using your app or think your app is bad, we have a serious problem. The app should be intuitive and user-friendly so all age groups can use it without hassle. It should be reliable on multiple phone models and reliable in the app itself. We should conduct user research throughout the project for user feedback and involve some users in testing. Without the end users we have wasted a lot of time and resources for a project that doesn't succeed.

The Dev Team should see the benefit in working on the project, clear requirements and timeline with the scope and time should give an idea to our dev team. Hopefully, the project gives some growth opportunities, and with SCRUM the team should be able to be involved in project planning and provide feedback in meetings while reviewing their recent work. Without the dev team or a dev team with bad morale, we can experience hardships with delivering the project to the time it should be delivered. If someone quits in the middle of the project this is a major problem for the whole project.

9.

Risk Factor	Probability of occurring (1-5)	Severity rating (1-5)	Evaluation (probability*severity)
Funders don't like a decision you made	2	5	10
End users don't like your application	3	5	15
The end user doesn't understand how to use your app	1	5	5
Someone quits their job in the dev team	1	5	5
The dev team has low morale	3	5	15

10.

We can avoid that the funders don't like a decision you made by constantly communicating with them, and clarifying where we are leading the project, so they always are involved in the project and can give you input.

We can try to mitigate that the end users don't like your app by doing thorough research on what the market needs, and have users involved in testing and design. Have users give feedback while the app is in development of wants and needs.

We can try to mitigate that the end users more often the elderly don't understand how to use the app by properly explaining how the app works on a small page for how to use and have a frequently asked question page. It would also help to have these users give feedback while testing the app.

Someone quitting their job is hard to deal with, because we can try to avoid it by providing a good work environment, and clear requirements of what's needed. We can also in the worst-case scenario just accept the risk and then something like the contingency reserve might be used to hire someone new and train them.

That the Dev Team has low morale can come from many things, we should try to mitigate this from happening by providing as said above a good work environment. Clear communication and goals for the team, while also trying to mitigate any individual problems someone might have with each other.

11.

The course is named IT-prosjektledelse so I was expecting a lot of content on how to be a project manager and managing different aspects of a project as a leader. I was expecting a lot more details about managing the people around the project, as in the projects team and investors and stakeholders which we have talked a lot about of course, but I was not expecting the course to teach how to create a great project in general, not relating to IT at all. The contents of the course and the teachings of the course would allow me to create a project not IT-relevant without being a project manager and still probably do a good job, the course reminds me of an entrepreneur course. This is beneficial for me because I already do some projects on the side, and it gives me insight on how to succeed or approach my own projects with who my potential customers are, how should I work, what are the risks of my project, how can I satisfy the needs of my customers.