Project Plan

Abra Cadabra

Philips

Date: 04-01-2021
Version: V0.4
State: Fourth draft for approval
Authors: Lars van den Brandt, Kevin Bevers, Kristian Lachev, Robbe Bryssinck, Denny Cox, Tijmen Coenders, Ivan Banov, Mureşeanu Gabriel

Version History

Version	Date	Author(s)	Changes	State
0.1	10-09-2020	Everyone	Removed unnecessary subjects	First draft for approval
0.2	15-09-2020	Everyone	Filled in most sections following the template descriptions	Second draft for approval
0.3	18-09-2020	Everyone	Completed all the sections in the document	Third draft for approval
0.4	05-10-2020	Everyone	Updated the document to reflect our changes to the project idea we made.	Fourth draft for approval

Distribution

Version	Date	Receivers
0.3	30-10-20	Carli Kleijnen, Alexander Heuts, Liesbeth Bergmans

Contents

1. Project Assignment

1.1 Context

Our company consists of student software engineers from the Demand Based course in Fontys. We were contacted by Philips to deliver a software solution to their company issue.

1.2 Goal of the Project

The goal of our project is to make a friendly and trustful environment where users can ask questions for several subjects and get answers from experts. The main reason we're doing this project is because we wanted a trustworthy way to get information from a trustworthy source. This subject is a good base for us to start developing on more complex techniques. Our preferred situation after our project would be that we created a website which creates safe and trustworthy ways to ask questions. Having this website would be helpful to all kinds of people who have questions that be it about cooking, programming or math. The advantage of our project would be the experts, who will create a trustworthy experience. The value this project would add to Philips is that we show a secure way to host a Q&A site with honest answers. In the end the project will realize: a secure way to verify experts and a safe and friendly environment for users.

1.3 Scope and Preconditions

Inside scope:	**Outside scope:**
1. User acceptance test	1. Performance Tests
2. Security measures	2. Guidance
3. Useability tests	3. Profile pages for users
4. User stories	4. Mobile application

1.4 Strategy

The strategy we choose for this project is the agile approach 'SCRUM'; the reason we choose this is so that we can deliver value often, we can continuously improve upon our project and commit to quality. We are going to work in sprints. In each sprint we focus on a few chosen user stories and complete these before the next delivery. Every sprint delivery, we will perform a demo for the project owner, so that they can give us feedback on the UI, the UX, and the features, so that the end product is something that the product owner will be satisfied with. We will also do a stand-up every time we start the day for our group project.

2. Project Organization

2.1 Stakeholders and Team Members

Name	Abbreviation	Role and functions	Availability
Carli Kleijnen		*Stakeholder*	*Tuesday 09:00 – 16:00, Thursday morning*
Alexander Heuts		*Project owner*	*Wednesday morning 10:00 – 11:30*
Liesbeth Bergmans		*Project owner*	*Wednesday morning 10:00 – 11:30*
Robbe Bryssinck		*Scrum Master, Back-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Kristian Lachev		*Stenographer, Back-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Kevin Bevers		*Front-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Ivan Banov		*Front-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Lars van den Brandt		*Front-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Denny Cox		*Back-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Tijmen Coenders		*Front-end Dev*	*Monday, Tuesday 09:00 – 16:00*
Mureşeanu Gabriel		*Back-end Dev*	*Monday, Tuesday 09:00 – 16:00*

3. Activities and Time Plan

3.1 Phases of the Project

Our project exists out of 3 phases, they are described below:

1. In the first phase of the project we analyse our risks, what techniques we want to use and why, during this phase a lot can change and we are focused on finding the best options for our project and how we can create our project in the fastest and simplest way possible.

2. In the second phase of the project we are focused on programming and creating the project, during this time there will be little documentation done. Only the necessary things like progress management and quick tech sketch ups are done. This phase is divided in multiple sprints of 3 weeks. Each sprint we focus on different user stories to complete.

3. In the third phase of the project we shift our attention to wrapping up and reflecting on our work process and the project itself. In this phase we deliver the project to the project owner, and hand over all our documentation.

3.2 Time Plan and Milestones

Every 3 weeks we will hold a sprint review/demo with our project holders and teachers. After that we will also have a retrospective to see what went good, bad and what we can do better. Every workday in the week we will also hold a stand-up with our entire team to discuss what we will do that day and how it went yesterday.

Phasing (Sprints)	Start date	Finish date
1 Phase 1 (Sprint 1)	21 September	6 October
2 Phase 2 (Sprint 2)	12 October	3 November
3 Phase 2 (Sprint 3)	9 November	24 November
4 Phase 2 (Sprint 4)	30 November	15 December
5 Phase 3 (Sprint 5)	4 January	19 January

Commented [BI1]: *Week 6 is between 5 to 11 October*

4. Testing Strategy and Configuration Management

4.1 Testing Strategy

During our project we are going to be doing a mix of user acceptance testing and unit testing. The reason being is that we can have a good balance between system tests, which will ensure that the software won't crash, and user acceptance test which would make sure that the software solution handles everything correctly.

4.2 Configuration Management

During the project we will use CI/CD practices, to ensure that we will be using GitHub. Additionally, we will apply branching strategies to allow for the continuous integration and deployment of our product. We will employ multiple branches and pull requests to allow for the smooth development operations, while keeping the latest release of the product available and accessible to the public. For CI, we will set up Github Actions. These will automatically build our code and run tests for each component. For CD, we will containerize the application with Docker, so it is easily deliverable to other machines.

If any unknown bugs were to appear in a release of our product, that need to be fixed immediately, we could employ a hotfix branch that would aim at resolving the problem as quick as possible, and allow for the issue not to persist on the live version, until a new release is deployed. Additionally, we will employ testing during development, to try and prevent occurrences of bugs appearing in a release.

5. Finances and Risk

5.1 Risk and Mitigation

Risk	Prevention activities	Mitigation activities
1 One of the team members gets sick so that work lags behind on a specific part of the project.	Have at least two people that have knowledge of the same subject.	Have a different person with enough knowledge take over.
2 A specific part of the project is technically too difficult, so we are not going to deliver in a sprint.	Research the subject to find out if it is not too difficult.	Decide to use a different technology that Is not too difficult.
3 One of the team members does not work as hard as the other ones so the amount of work is not spread equally.	Have a stand-up meeting every workday to know what people are working on individually.	Notate the hours of work and what was done in that time.